



کتابچه مقالات

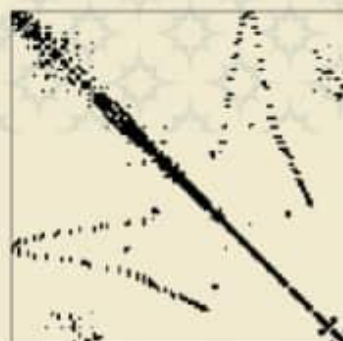
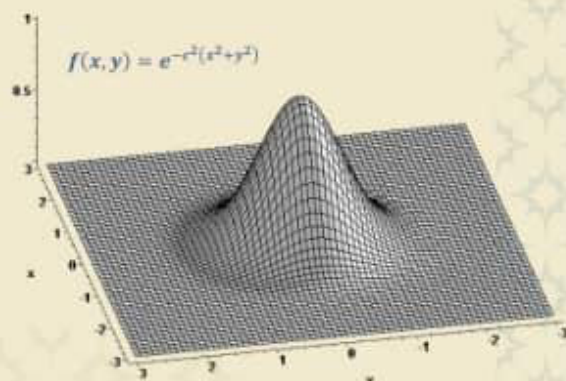


پایتخت علمی
مؤسسه تحقیقات و توسعه
مؤسسه مدیریت و برنامه ریزی استان کردستان



هشتمین سمینار آنالیز عددی و کاربردهای آن

۱۷ الی ۲۰ فروردین ۱۴۰۰



ویراستاران:

دکتر ارسلان رحمانی
سپیده جعفری رشتی
مسعود پندار

Website: conf.uok.ac.ir/sna2020

Email: sna2020@uok.ac.ir

Phone: 087-33660076

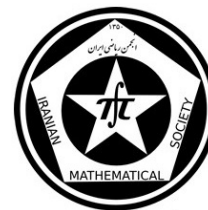
آدرس: ایران، کردستان، سنندج، بلوار پاسداران، دانشگاه کردستان - کدپستی: ۶۶۱۷۷-۱۵۱۷۵





8th Seminar on Numerical Analysis and its Applications

University of Kurdistan, Sanandaj, Iran, April 06-09 2021



8th Seminar on Numerical Analysis and its Applications

April 06-09, 2021

University of Kurdistan

Address: University of Kurdistan, Pasdaran Blvd, Sanandaj, Kurdistan, Iran

Phone: +98-87-33660076

mail: sna2020@uok.ac.ir

Scientific Committee

- Esmail Babolian – Kharazmi University
- Saeid Abbasbandy – Imam Khomeini International University
- Morad Ahmadnasab – University of Kurdistan
- Amjad Alipanah – University of Kurdistan
- Kayvan Amini – Razi University
- Amir Amiraslani – Capilano University
- Pouria Assari – Bu-Ali Sina University
- Mansour Dana – University of Kurdistan
- Mohammad taghi Darvishi – Razi University
- Shahrokh Esmaeili – University of Kurdistan
- Ali Foroush Bastani – Institute for Advanced Studies in Basic Sciences (IASBS)
Zanjan-Iran
- Mohammad Ghasemi – University of Kurdistan
- Mahmoud Hadizadeh Yazdi – K. N. Toosi University of Tech.
- Gholam Reza Hojjati – University of Tabriz
- Reza Jalilian – Razi University
- Davod Khojasteh Salkuyeh – University of Guilan
- Mehrdad Lakestani – University of Tabriz
- Nezam Mahdavi-Amiri – Sharif University of Technology
- Davoud Mirzaei – University of Isfahan
- Yadollah Ordokhani – Alzahra University
- Arsalan Rahmani – University of Kurdistan
- Jalil Rashidnia – Iran University of Science and Technology
- Davoud Rostamy – Imam Khomeini International University
- Fardin Saedpanah – University of Kurdistan
- Ali Sangavi – Charmo University
- Sedaghat Shahmorad Moghanloo – University of Tabriz
- Mostafa Shamsi – Amirkabir University of Technology
- Kamal Shanazari (Chair of Scientific Committee) – University of Kurdistan
- Alireza Sohili – Ferdowsi University of Mashhad
- Mohammad Reza Yaghooti – Guilan University
- Sohrabali Yousef – Shahid Beheshti University
- Ali Zakeri – Tossi University of Technology

Organizing Committee

- Amjad Alipanah (Chair of Organizing Committee)
- Morad Ahmadnasab
- Mohammad Ghasemi
- Arsalan Rahmani
- Eghbal Ghaderi
- Mohammad Esmael Samei – Bu-Ali Sina University
- Mohammad Ali Ardalani
- Amanj Khoramian
- Kaveh Sadeghi
- Somayeh Jafarrameshti
- Behareh Noori
- Neda Nouri
- Shahla Zandinejad
- Masoud Pendar
- Keivan Mohammadi
- Mehran Mohammadi
- Himan Rezaee

Keynote Speakers

- ★ Prof. Esmail Babolian – Kharazmi University
- ★ Prof. John C. Butcher – University of Auckland
- ★ Prof. Robert M. Corless – University of Western Ontario
- ★ Prof. Nezam Mahdavi-Amiri – Sharif University of Technology
- ★ Prof. Jalil Rashidnia – Iran University of Science and Technology
- ★ Prof. Mostafa Shamsi – Amirkabir University of Technology

	Participants	University
1	Akbar Shirilord	Amirkabir University of Technology
2	Ali Ebrahimijahan	Amirkabir University of Technology
3	Ali Habibirad	Shiraz University of Technology
4	Ali Janmohammadi	Semnan University
5	Amin Abrishami Moghadam	Ferdowsi University of Mashhad
6	Amin Faghih	Sahand University of Technology
7	Azam Ebadi	Urmia University
8	Azam Sadat Hashemi	Yazd University
9	Behzad Nemati Saray	Institute for Advanced Studies in Basic Sciences (IASBS)
10	Davod Khojasteh Salkuyeh	University of Guilan
11	Donya Haghighi	Imam Khomeini International University
12	Ebrahim Azhdari	Salman Farsi University of Kazerun
13	Elyas Shivanian	Imam Khomeini International University
14	Eman Tavakoli	Ferdowsi University of Mashhad
15	Erfan Bahmani	University of Zanjan
16	Fakhroddin Hashemi	University of Guilan
17	Farzad Rahpeymaii	Payame Noor University
18	Fateme Ghomanjani	Kashmar Higher Education Institute
19	Fateme Torabi	Damghan University
20	Fazlollah Soleymani	Institute for Advanced Studies in Basic Sciences (IASBS)
21	Hadis Azin	University of Hormozgan
22	Hajar Masoumi	University of Bonab
23	Hamid Momeni	University of Mazandaran
24	Hamid Reza Khodabandehlo	Imam Khomeini International University
25	Haniye Dehestani	Alzahra University
26	Hassan Majidian	Tarbiat Modares University
27	Iraj Fahimi Khalilabad	University of Tabriz
28	Jamshid Saeidian	Kharazmi University
29	Kolsum Mozdgir	Urmia University
30	Maeddeh Pourbagher	University of Guilan
31	Mahdi Jalili	Tarbiat Modares University
32	Mahmood Dadkhah	Payame Noor University
33	Mahshid Foroughipour	Shahrekord University
34	Mansour Hoseiniazar	Urmia University
35	Maryam Kargarfard	Vali-e-Asr University of Rafsanjan
36	Maryam Sheikhi	Urmia University
37	Masoumeh Zeinali	University of Tabriz
38	Mehran Namjoo	Vali-e-Asr University of Rafsanjan
39	Mina Mohammadi	Urmia University of Technology
40	Mohadese Sarvari	Shahid Bahonar University of Kerman

41	Mohamad Javad Fazeli	University of Birjand
42	Mohammad Mahdi Izadkhah	Birjand University of Technology
43	Mohammad Tafakkori-Bafghi	Yazd University
44	Mohammad Zaher Kazemi Baneh	University of Kurdistan
45	Mohsen Riahi	Saravan Integrated Education
46	Monireh Nosrati Sahlan	University of Bonab
47	Moslem Jamalpour MalekAbadi	K. N. Toosi University of Technology
48	Nadia Dolatabady	Hakim Sabzevari University
49	Narges Taheri	Islamic Azad University, South Tehran Branch
50	Naser Azizi	Damghan University
51	Nasim Maddah Shariati	University of Guilan
52	Nasrin Samadyar	Alzahra University
53	Nayyer Jangi-bahador	University of Tabriz
54	Neda Biabannavard	University of Bojnord
55	Razie Gholampour	Alzahra University
56	Razieh Delpasand	Shahid Bahonar University of Kerman
57	Raziye Erfanifar	Malayer University
58	Reyhane Mohamadinejad	University of Guilan
59	Reza Jalilian	Razi University
60	Salameh Sedaghat	Buein Zahra Technical University
61	Samira Shahsavari	University of Guilan
62	Seyed Mehdi Shafiof	Payame Noor University
63	Somaye Sheikhi	University of Mazandaran
64	Somayeh Fazeli	University of Tabriz
65	Somayeh Majidifar	Yasouj University
66	Somayyeh Jangjooye Shaldehi	Alzahra University
67	Soraya Torkaman	Yazd University
68	Tahereh Molaee	Alzahra University
69	Vali Torkashvand	Farhangian University
70	Zohre Azimi	University of Mazandaran

Keynote Speakers Abstracts

Introduction to G-symplectic methods

John C. Butcher – University of Auckland

Abstract

G-symplectic methods are an alternative to symplectic Runge-Kutta methods. This talk will discuss the development of G-symplectic methods in collaboration with Y. Habib, A. Hill, G. Imran, T. Norton, H. Podhaisky and others. In particular, a sixth order method, will be presented, together with numerical experimental results. To support the analysis of G-symplectic, and other general linear methods, a discussion will be included of trees, forests, elementary differentials and B-series.

Compact Finite Differences and Cubic Splines

Rob Corless

University of Western Ontario

Abstract

Recently an unexpected connection was found between compact finite differences and cubic splines, which has led to a new and possibly useful way of dealing with the edge conditions (typically dealt with by using the not-a-knot condition). In exploring this connection, we prove that a certain family of parameterized tridiagonal matrices is totally positive, which means that the computation of cubic splines and compact finite differences can be done in a fast and numerically stable way. This talk is intended to be widely accessible, but the audience will have to believe that cubic splines, such an elementary topic, might still be interesting.

A non-monotone filter exact penalty approach using inexact solutions of sequential quadratic programming problems for solving constrained nonlinear programming problems

Nezam Mahdavi-Amiri and Hani Ahmadzadeh
Sharif University of Technology

Abstract

Sequential Quadratic Programming (SQP) algorithms for solving smooth constrained nonlinear programming (CNLP) problems have been popular for several decades, mainly due to their fast local convergence. These algorithms, however, having the engagement of all the constraints in every iteration, require high computational costs. Recently, non-monotone filter SQP algorithms have been introduced for solving large CNLP problems. Here, to speed up the iterations, we present a new version of such an algorithm, making use of inexact solutions of the SQP problems. The proposed method, making use of an exact penalty function, considers the use of two directions, the so-called steering and predictor directions. In a global iteration, with an infeasible iterate at hand, both directions are computed. The steering direction, computed as a minimizer of a linear approximation of the constraint violations over a trust region, is used to construct a convex feasible quadratic sub-problem for computing the predictor direction, which serves to be a descent direction for the penalty function. A combined steering and predictor direction is also specified to serve as a descent direction for the constraint violations. This direction is also made to be descent for the penalty function by a proper updating of the penalty parameter. In a local iteration, with a feasible iterate at hand, an inexact solution of a convex quadratic model is computed to serve as a descent direction for the objective function. To avoid the Maratos effect, a non-monotone filter strategy is employed in local iterations. A superlinear local rate of convergence is established under standard assumptions. Comparative numerical results obtained by an implementation of the proposed method on a collection of available test problems in the literature affirm the approach to be competitive.

Sinc Function Approximation

Jalil Rashidinia

School of Mathematics

Iran University of Science and Technology

Abstract

In this talk, a brief survey of Sinc approximation to numerical solution of differential, integral equations and fractional differential equations has been investigated. We will emphasize on recent development in this field as well as the works which have been developed at Iran University of Science and Technology. This talk has six major parts, first part is devoted to the definition and main concepts of Sinc approximation and second part is concern with application of Sinc function for numerical solution of integral equations. The third part focused on solution of singularly perturbed parabolic convection-diffusion problems by Sinc basis function. The fourth part is devoted to numerical solution of ordinary differential equation. The fifth part is concerned with the Sinc- Galerkin and Sinccollocation method for solving nonlinear two point boundary value problems. The sixth part is concern with application of Sinc function for numerical solution of fractional differential equations.

POD reduced-order method for solving optimization problems constrained by Partial Differential Equations

Mostafa Shamsi

Amirkabir University of Technology

Abstract

PDE-constrained optimization is a rapidly developing field of mathematical optimization where at least one PDE appears in the constraints of the problem. This field plays a central role in optimal design and control of systems in aerodynamics, computational fluid dynamics, image segmentation, inverse problems, etc. However, these problems are often hard to tackle numerically because their discretization leads to very large-scale optimization problems. In this regard, the reduced-order methods are developed to approximate these problems by smaller ones that are tractable with less computational time and memory. Proper Orthogonal Decomposition (POD) is probably the most efficient one for PDE-constraint optimization among reduced order techniques. In this talk, after a brief introduction to PDE optimization, a successful application of using POD for solving a PDE-optimization problem is demonstrated.

Articles

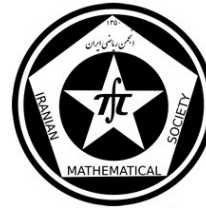
- * A coordinate transformation method for solving differential equations on unbounded domain 1
- * An Efficient Solution of System of Generalized Abel Integral Equations Using Jacobi Wavelets 7
- * A new double-step iteration method for solving indefinite complex linear system of equations 12
- * An Extension of Jung's Theorem 18
- * An iteration method for solving the systems arisen from finite element discretization of PDE-constrained optimization problems 22
- * An iterative scheme for generalized Lyapunov matrix equation 28
- * An operational matrix method for solving a class of nonlinear Volterra integral equations arising in steady activation of a skeletal muscle 34
- * An operation matrix method based on the Boubaker polynomials for solving nonlinear integro-differential equations 40
- * A numerical method for one-dimensional Schrodinger equation using compact local integrated radial basis function method 46
- * A P_0 WEAK GALERKIN FINITE ELEMENT METHOD FOR SECOND-ORDER PROBLEMS 52
- * A point-collocation approach for solving ordinary differential equations with implicit Runge-Kutta methods 58
- * Application of Caputo fractional derivative operator for solving the fractional optimal control problems 65
- * Applying the Meshless Fragile Points Method (FPM) to solve Two Dimensional Linear Schrödinger Equation 72
- * CGNE-like iterative method for nearness problem of generalized Sylvester-transpose matrix equations 79
- * Compact finite difference method to solve the problem of electrohydrodynamic flow of a fluid in an ion drag configuration in a circular cylindrical conduct 85
- * Development of wavelets method for numerical treatment of two-dimensional nonlinear Volterra integral equations 91



* Differential transform and exponential spline function approximations for fractional sub-diffusion equations	97
* Direct meshless local Petrov-Galerkin for the nonlinear coupled Burgers' equation	110
* Existence and multiplicity of solutions to the nano boundary layer flows with nonlinear Navier boundary condition	116
* Generalization of Sarrus rule	122
* Higher order finite difference method based on linear barycentric rational interpolation to solve sub-diffusion equations	128
* Hyperbolic contour to numerical approximation of fractional Volterra integro-differential equation	134
* Identification of a space-dependent source term in a time-fractional wave equation	140
* Implementation of Lucas wavelet functions for solving fractional differential equations	145
* Jacobi polynomials based collocation method for solving singular fractional Lane-Emden equation	151
* Legendre wavelet method combined with the Gauss quadrature rule for numerical solution of fractional integro-differential equations	156
* Lucas polynomials collocation method for solving fractional order electrohydrodynamics flow model	162
* Mathematical model of drug release to the posterior segment of the eye after topical drug delivery	168
* Modified Filon-Clenshaw-Curtis rules for regular oscillatory integrals	174
* Multi-scale representation of Galerkin method for solving linear mixed Volterra-Fredholm integral equations	180
* Numerical study of 2D variable-order time fractional mobile/immobile advection-diffusion model using a truly meshless method	186
* Numerical Solution of The Three-Asset Black-Scholes Option Pricing Model	191
* Numerical simulation of stochastic population growth model via operational matrix method	197
* Numerical solution of Bagley-Torvik equation using fractional Chebyshev collocation method	203

* Numerical solution of nonlinear multi-order fractional differential equations using a novel shifted Jacobi operational matrix	208
* Numerical solution of spread option pricing model based on jump-diffusion models by ADI method	214
* Numerical solution of the fractional boundary value problems by the Sinc-Collocation method	221
* Numerical solution of the inverse Gardner equation	227
* Numerical Solution of Inverse Time Fractional Telegraph Equation	233
* Numerical Study of Inverse System of Coupled Burgers Equations by Chebyshev Wavelet Method	239
* Numerical methods for Ordinary differential equations based on second derivative collocation	245
* Nonstandard finite difference- θ scheme for stochastic partial differential equations	251
* Nonhomogeneous linear second order fully fuzzy differential equation under MMCE-product	257
* On a Bzier-Like Model with Shape Parameter	263
* On RBF-HFD formulas under multiquadric function on non-equidistant grids	269
* Solving a new class of nonlinear optimal control problems generated by Atangana-Baleanu-Caputo variable order fractional derivative	274
* Solving the Volterra functional integral equations with non-vanishing delays	279
* Sine-Cosine wavelets method for solving DrinfeldSokolov system	285
* Some optimization methods for solving absolute value equation	291
* The Crank-Nicolson Type Methods for Fractional Partial Differential Equations	297
* Tumor Hyperthermia treatment: Effect of cooling period on thermal dose	302

۳۰۸	برآورد جواب مساله مقدار مرزی غیرخطی ناشی از بینایی سنجی
۳۱۴	برآورد نوسانات تصادفی بازارهای مالی با استفاده از مدل ارنشتین آهلنبرگ با نویز لوی
۳۲۰	تحقق انتهای طیف ها به وسیله ی ماتریس های سه قطری حقیقی متقارن
۳۲۸	تصحیح بهینه دستگاه نشدنی از معادلات قدرمطلق کلی با استفاده از مینیمم تغییرات درماتریس های ضرایب و بردار سمت راست
۳۳۶	تقریب جدید برای معادله ی برگرز کسری زمانی با مشتق کاپاتو
۳۴۵	توسعه روش های کانگ-تراب دو گامی بهینه بدون حافظه به حالت آداپتیو با حافظه
۳۵۸	حرارت درمانی تومور سرطانی: رویکرد مساله معکوس
۳۶۴	حل دستگاه معادلات انتگرال آبل با استفاده از چندجمله ای های ژاکوبی انتقال یافته
۳۷۰	حل عددی معادلات آب کم عمق تحت یک جریان برشی خطی و ساحل شیب دار
۳۸۱	حل عددی معادلات دیفرانسیل کسری مرتبه متغیر با استفاده از تقریب ریتز و روش هم محلی
۳۸۷	حل عددی معادله تلگراف کسری با روش اجزای حجمی محدود
۳۹۴	حل مدل بازهای بلک-شولز با استفاده از روش گسسته سازی
۴۰۲	حل مسأله میانگین-واریانس مارکوفیز با روش های بهینه سازی درجه دوم
۴۱۱	حل و استخراج معادلات دیفرانسیل مشتقات جزئی داده محور از دیدگاه یادگیری ماشین
۴۲۱	روش بلوکی چندگامی برای حل عددی معادلات دیفرانسیل تأخیری درجه دوم با شرایط مرزی
۴۳۱	روش تقریب های پیشرو چندجمله ای های چیشف نوع اول برای حل عددی معادلات دیفرانسیل معمولی مرتبه دوم
۴۳۹	روش هم محلی بر اساس پایه های چند ترازوی برای حل دستگاه های مرکب معادلات انتگرال ولترا-فردهلم نوع اول و دوم
۴۴۶	روش های هم محلی چندگامی برای حل معادلات انتگرال ولترای اتو-کانولوشن نوع دوم
۴۵۴	روش هم محلی بر اساس چند جمله ایهای تکه ای برای حل معادلات جبری انتگرال تاخیری نیمه صریح
۴۶۲	کاربرد روش توابع پایه شعاعی برای حل مساله معکوس سهموی خطی
۴۶۹	کاربرد روش های پروکسیمال در حل دستگاه معادلات قدر مطلق
۴۷۷	کاربردهایی از دترمینان
۴۸۷	مدل سازی بیماری کرونا
۴۹۵	مشخص سازی لایه های شبکه چشم انسان با استفاده از موجک در ماشین بردار پشتیبان
۵۰۳	همگرایی سراسری روش گرادیان مزدوج PRP برای توابع غیر محدب
۵۱۱	یک روش تکراری موثر بدون معکوس برای محاسبه تجزیه قطبی ماتریس



A coordinate transformation method for solving differential equations on unbounded domain

H. Azin[†], F. Mohammadi^{‡*}

Department of Mathematical, University of Hormozgan, Hormozgan, Iran and h.azin.phd@hormozgan.ac.ir[†].

Department of Mathematical, University of Hormozgan, Hormozgan, Iran and f.mohammadi62@hotmail.com[‡].

Abstract

This study presents an efficient numerical method to solve differential equations in unbounded domains with vanishing boundary conditions. First, by using properly designed domain transformations, the actual differential equation in the unbounded physical domain is transformed to a mapped differential equation in computational bounded domain. Then, the mapped Legendre polynomial is then used to solve the differential equation in the computational bounded domain. To demonstrate the accuracy and efficiency of the proposed method, it has been implemented to solve some illustrative examples.

AMS subject Classification 2010: 33C45, 34B40, 34K28.

Keywords: Coordinate transformation, Unbounded domains, Legendre polynomials, Differential equations.

1 Introduction

Infinite or unbounded domains are often encountered in mathematical models associated with acoustic, aerodynamic, geophysical, electromagnetic and many other problems. The domain truncating and artificial boundary conditions are the most commonly used approaches for solving this kind of problems. A more satisfactory treatment consists of mapping the original domain into a bounded one and using these standard discretization techniques. In [1] a non reflecting boundary condition (NRBC) on the chosen artificial boundary is proposed for solving 2D wave equations. Moreover, the method of truncating unbounded domain by using artificial boundaries is investigated in [2]. Shen and Wang [3] introduced spectral methods with mapped Jacobi, Laguerre and Hermite functions to solve differential equations on unbounded domain. In this paper, an efficient numerical method based on coordinate transformation is proposed to solve differential equations in unbounded domains with vanishing boundary conditions.

*H. Azin



2 Preliminaries

Legendre polynomials

The Legendre polynomial $L_i(x)$ of degree $i \in \mathbb{N} \cup \{0\}$ can be defined by recurrence relation:

$$L_{i+1}(x) = \frac{2i+1}{i+1}xL_i(x) - \frac{i}{i+1}L_{i-1}(x), \quad x \in [-1, 1], \quad (1)$$

where $L_0(x) = 1$ and $L_1(x) = x$. Also, Legendre polynomials are orthogonal with respect to the weight function $w(x) = 1$, i.e.

$$\int_{-1}^1 L_i(x)L_j(x)dx = \frac{2}{2i+1}\delta_{ij}, \quad (2)$$

where δ_{ij} is the Kronecker function.

3 Approximation on unbonded domain

Let $y(x)$ is smoothly continuous in $\mathbb{R}$. Hence, it can be approximated as

$$y(x) = \sum_{k=0}^{+\infty} a_k f_k(x),$$

which $f_k(x)$ are orthonormal function and complete in $\mathbb{R}$. That is

$$\int_{-\infty}^{+\infty} f_k(x)f_s(x)dx = \delta_{ks}. \quad (3)$$

In particular, it can be considered as follows:

$$y(x) \simeq \sum_{k=0}^m a_k f_k(x). \quad (4)$$

4 The computational method

Transformation function using mapped basis set

The main aim of this section is to present a mapping for transforming unbounded domain to a bounded domain by the intermediate of a mapped basis set. First, the variable x is transfered to a finite interval by using the function $g(u)$, i.e.

$$x = g(u) \implies u = g^{-1}(x), \quad (5)$$

where $u \in [a, b]$. The basis functions should be orthogonal in both intervals. As a result, we have

$$f_i(x) = P_i(u)\mu(u), \quad (6)$$

where $\{P_i(u)\}$ is an orthogonal basis set with respect to the weight function $\omega(u)$ that is

$$\int_a^b P_i(u)P_k(u)\omega(u)du = c_i\delta_{ik}, \quad (7)$$



and $\mu(u)$ is an auxiliary function that either $P_i(u)$ or $\mu(u)$ and or both must vanish as $x \rightarrow \pm\infty$. We choose $\mu(u)$ that vanishes at infinities. Applying the coordinate transformation (6) in (3)

$$\int_a^b P_i(u)P_k(u)\mu^2(u)\frac{dx}{du}du = \delta_{ik}. \quad (8)$$

Relations (3) and (8) result in

$$\frac{dx}{du} = \frac{\omega(u)}{c_i\mu^2(u)}. \quad (9)$$

The $\{P_i(u)\}$ is chosen to be coefficient of Legendre polynomial, $P_i(u) = \sqrt{i + \frac{1}{2}}L_i(u)$. Thus

$$\frac{dx}{du} = \frac{1}{\mu^2(u)}. \quad (10)$$

The transformation function obtain by applying the $\mu(u) = \sqrt{1-u^2}$ and Eq.10 as follow

$$x = g(u) = \tanh^{-1}(u). \quad (11)$$

Application in differential equations

Consider the generalized form

$$Ly(x) = F(x), \quad y(x) \rightarrow 0 \quad \text{as } x \rightarrow \pm\infty, \quad (12)$$

where L is a differential operator on $y(x)$ and $F(x)$ is a known function. Both $y(x)$ and $F(x)$ are smoothly continuous. we approximate $y(x)$ according with (4). Using the inner products $(f_k(x)/\sum_{i=1}^N a_i Lf_i(x))$ and $(f_k(x)/F(x))$, relation (12) can be expressed in the martix form $A\Upsilon = B$ where A is the differential operator matrix with elements $A_{ij} = (f_i(x)/Lf_j(x))$, $\Upsilon = [c_1 \quad \cdots \quad c_N]^T$ and $B = [b_1 \quad \cdots \quad b_N]^T$ in which $b_i = (f_i(x)/F(x))$. Now consider the operator L as follows:

$$L = \frac{d^n}{dx^n} + \alpha x^n, \quad (13)$$

which α is a constant and x a variable. Thus, we have

$$A_{ij} = \int_{-\infty}^{+\infty} f_i(x) \frac{df_j(x)}{dx} dx = \int_{-1}^1 P_i(u)\mu(u) \frac{d(P_j(u)\mu(u))}{du} du, \quad (14)$$

$$b_i = \int_{-\infty}^{+\infty} f_i(x)F(x)dx = \int_{-1}^1 \frac{P_i(u)F(g(u))}{\mu(u)} du, \quad (15)$$

and

$$X_{ij} = \int_{-\infty}^{+\infty} f_i(x)f_j(x)xdx = \int_{-1}^1 P_i(u)P_j(u)g(u)du. \quad (16)$$

The matrix form of (13) is given by $A^n + \alpha X^n$.

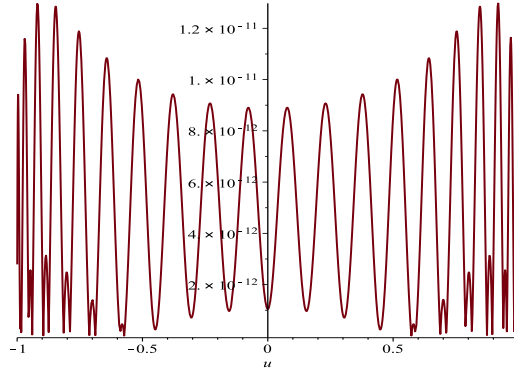


Figure 1: Absolute error function for Example 0.1 with $m=40$ whenever $\kappa = 10$.

3 Numerical example

Example 0.1. Consider the problem

$$\frac{d^2 y}{dx^2} = 2\kappa e^{-\kappa x^2} (2\kappa x^2 - 1), \quad (17)$$

whit the exact solution $y(x) = e^{-\kappa x^2}$. Graph of the absolute error is shown in Figure 1.

Example 0.2. Consider the differential equation

$$-\frac{d^2 y}{dx^2} + \alpha y(x) = F(x),$$

where α is a constant. The exact solution of this problem is $y(x) = \cos(kx)e^{-\beta x^2}$ and $y(x) = \frac{\cos(kx)}{(1+x^2)^\beta}$ with and without oscillation at infinity with $k = 2$ and $k = 0$, respectively. Figures 2 to 5 show the absolute error functions for Example 0.2 with $\alpha = 2$.

References

- [1] S. Falletta and G. Monegato, *An exact NRBC for 2D wave equation problems in unbounded domains*, (2013).
- [2] H. Han and X. Wu, *Artificial boundary method*, Tsinghua Univ. Press, (2013).

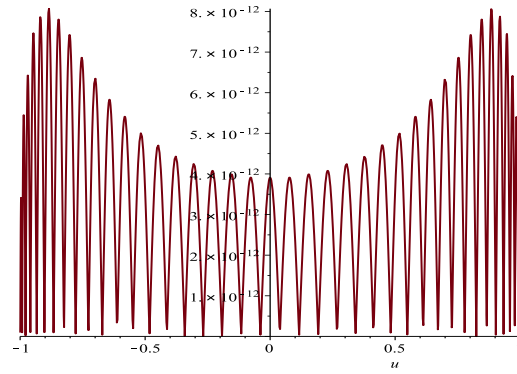


Figure 2: Numerical results obtained of exponential decay without oscillation for Example 0.2 with $\beta = 10$ and $m = 40$.

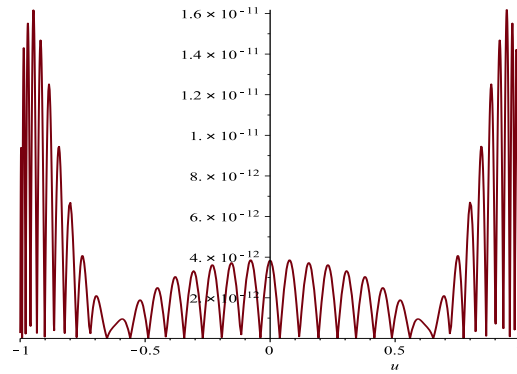


Figure 3: Numerical results obtained of exponential decay with oscillation for Example 0.2 with $\beta = 10$ and $m = 40$.

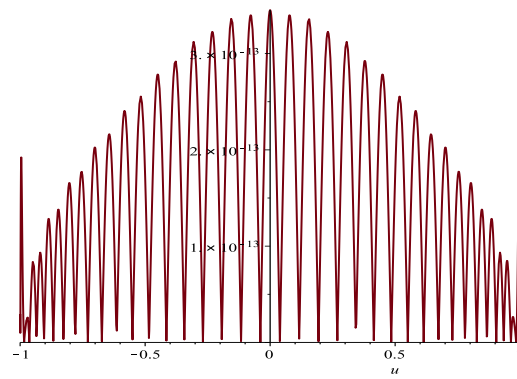


Figure 4: Numerical results obtained of algebraic decay without oscillation for Example 0.2 with $\beta = 10$ and $m = 40$.

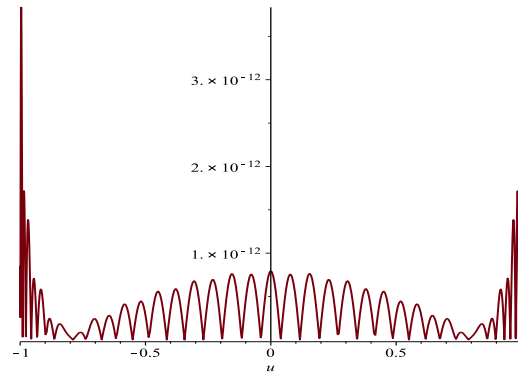
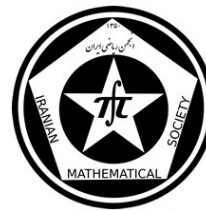


Figure 5: Numerical results obtained of algebraic decay with oscillation for Example 0.2 with $\beta = 10$ and $m = 40$.

- [3] J. Shen and Li. Wang, *Some recent advances on spectral methods for unbounded domains*, communications in computational physics, **5** (2009), 195-241.



An Efficient Solution of System of Generalized Abel Integral Equations Using Jacobi Wavelets

S. Sedaghat ¹

Buein Zahra Higher Education Institute of Technology, Imam Khomeini International University, Qazvin, Iran

Abstract

Based on Jacobi wavelets polynomials, an operational method is proposed to solve the generalized Abels integral equations. An approximation of the unknown function is considered in terms of Jacobi wavelets functions with unknown coefficients, which must be determined. By substituting this approximation into the equation, a system of linear algebraic equations is obtained. The applicability of earlier numerical inversion methods was restricted to the one portion of generalized Abel integral equations. The proposed method is quite accurate. Numerical example is given to illustrate the applicability, efficiency, and accuracy of the new scheme. AMS subject Classification 2010: 41A10, 65G99.

Keywords: Jacobi wavelets; Abel inversion; system of generalized Abel integral equations.

1 Introduction

In this paper, we propose an approximate numerical inversion for the system of generalized Abel integral equations given by

$$\begin{cases} \rho_1(s) \int_0^s \frac{u^{\beta-1} \varepsilon_1(u)}{(s^\beta - u^\beta)^\alpha} du + \rho_2(s) \int_s^1 \frac{u^{\beta-1} \varepsilon_2(u)}{(u^\beta - s^\beta)^\alpha} du = I_1(s) \\ \omega_1(s) \int_s^1 \frac{u^{\beta-1} \varepsilon_1(u)}{(u^\beta - s^\beta)^\alpha} du + \omega_2(s) \int_0^s \frac{u^{\beta-1} \varepsilon_2(u)}{(s^\beta - u^\beta)^\alpha} du = I_2(s) \end{cases} ; s \in (0, 1); 0 < \alpha < 1; \beta \geq 1, \quad (18)$$

where coefficients ρ_1 , ρ_2 , ω_1 , ω_2 cannot vanish simultaneously and $I_1(s)$, $I_2(s)$ are intensity functions.

Abels equation is an integral equation derived directly from a concrete problem in physics, without satisfying a differential equation. This integral equation is found in the mathematical modeling of several models in physics, astrophysics, solid mechanics, scattering theory, spectroscopy, stereology, elasticity theory, and plasma physics. Many approaches have been proposed for determining the numerical solution to Abel equation [1]. For example, Fettis [2] proposed a numerical form of the solution to Abel equation by using the Gauss-Jacobi quadrature rule. Piessens and Verbaeten [3] and Piessens [4] developed an approximate solution to Abel equation by means of the Chebyshev polynomials of the first kind. When input signal is with noisy error, Murio et al. [5] suggested a stable numerical solution. Furthermore, Garza et al. [6] and Hall et al. [7] used the wavelet method to invert the inversion of noisy Abel equation. The aim of this paper is to give a new algorithm for the numerical inversion of the system of generalized Abel integral equation, based on Jacobi polynomials wavelet bases. Numerical example is given to demonstrate the efficiency and accuracy of the proposed method.

¹Speaker



2 Preliminaries

The class of Jacobi polynomials, $P_i^{(v,\gamma)}(t)$, $(-1, 1)$ [8] are a set of orthogonal functions

$$P_i^{(v,\gamma)}(t) = \sum_{k=0}^i \binom{i+v}{k} \binom{i+\gamma}{i-k} \left(\frac{t-1}{2}\right)^{i-k} \left(\frac{t+1}{2}\right)^k,$$

with respect to the weight function

$$\omega^{(v,\gamma)}(t) = (1-t)^v(1+t)^\gamma, \quad (19)$$

with the following orthogonality property:

$$\int_{-1}^1 \omega^{(v,\gamma)}(t) P_i^{(v,\gamma)}(t) P_j^{(v,\gamma)}(t) dt = h_i^{(v,\gamma)} \delta_{ij},$$

where δ_{ij} is the Kronecker delta and

$$h_i^{(v,\gamma)} = \frac{2^{v+\gamma+1} \Gamma(v+i+1) \Gamma(\gamma+i+1)}{i! (v+\gamma+2i+1) \Gamma(v+\gamma+i+1)}.$$

We define the generalized Jacobi wavelets functions on the interval $[0, T)$ as follows:

$$\psi_{n,m}^{(v,\gamma)}(t) = \begin{cases} 2^{\frac{k}{2}} \sqrt{\frac{1}{h_m^{(v,\gamma)}}} P_m^{(v,\gamma)}\left(\frac{2^k}{T}t - 2n + 1\right), & \frac{n-1}{2^{k-1}}T \leq t \leq \frac{n}{2^{k-1}}T, \\ 0, & o.w, \end{cases}$$

where $k = 1, 2, 3, \dots$ is the level of resolution, $n = 1, 2, 3, \dots, 2^{k-1}$, $m = 0, 1, 2, \dots$, is the degree of the Jacobi polynomial, and t is the normalized time.

An arbitrary function $f \in L_2[0, T)$ may be approximated using Jacobi wavelet functions as

$$f(t) \simeq \psi_{k,M}^{(v,\gamma)}(t) = \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M c_{n,m} \psi_{n,m}^{(v,\gamma)}(t) = C^T \psi(t), \quad (20)$$

where C and ψ are $(M+1)2^{k-1} \times 1$ order matrices given by

$$C = [c_{10}, c_{11}, \dots, c_{1M}; \dots; c_{(2^{k-1})0}, c_{(2^{k-1})1}, \dots, c_{(2^{k-1})M}]^T,$$

$$\psi(t) = [\psi_{10}, \psi_{11}, \dots, \psi_{1M}; \dots; \psi_{(2^{k-1})0}, \psi_{(2^{k-1})1}, \dots, \psi_{(2^{k-1})M}]^T,$$

and

$$\begin{aligned} c_{n,m} &= \left\langle f(t), \psi_{n,m}^{(v,\gamma)}(t) \right\rangle_{\omega_k^{(v,\gamma)}} \\ &= \int_0^T \omega_k^{(v,\gamma)}(t) f(t) \psi_{n,m}^{(v,\gamma)}(t) dt \\ &= \int_{\frac{n-1}{2^{k-1}}T}^{\frac{n}{2^{k-1}}T} \omega_{n,k}^{(v,\gamma)}(t) f(t) \psi_{n,m}^{(v,\gamma)}(t) dt. \end{aligned}$$

The integral operator for Jacobi wavelet polynomials can be approximated by

$$\int_0^t \psi(t') dt' = \psi(t, 1). \quad (21)$$



3 Solution of the system of generalized Abel integral equations

In this section, we are discussing the solution of the system of generalized Abel integral equations in equation (1.1) by using Jacobi polynomials wavelet bases.

Now, let us take the unknown functions $\varepsilon_1(u)$, $\varepsilon_2(u)$ and the intensity functions $I_1(s)$, $I_2(s)$ from equation (1.1) and approximates to this by using equation (2.2), such as

$$\begin{aligned}\varepsilon_1(u) &= C_1^T \psi(u), & \varepsilon_2(u) &= C_2^T \psi(u), \\ I_1(s) &= F_1^T \psi(s), & I_2(s) &= F_2^T \psi(s),\end{aligned}\tag{22}$$

where C_1 , C_2 , F_1 , F_2 are coefficient matrices. Substituting these approximated value of functions from equation (3.1) in equation (1.1) we get

$$\begin{cases} \rho_1(s) C_1^T \int_0^s \frac{u^{\beta-1} \psi(u)}{(s^\beta - u^\beta)^\alpha} du + \rho_2(s) C_2^T \int_s^1 \frac{u^{\beta-1} \psi(u)}{(u^\beta - s^\beta)^\alpha} du = F_1^T \psi(s), \\ \omega_1(s) C_1^T \int_s^1 \frac{u^{\beta-1} \psi(u)}{(u^\beta - s^\beta)^\alpha} du + \omega_2(s) C_2^T \int_0^s \frac{u^{\beta-1} \psi(u)}{(s^\beta - u^\beta)^\alpha} du = F_2^T \psi(s). \end{cases}\tag{23}$$

Let

$$\int_0^s \frac{u^{\beta-1} \psi(u)}{(s^\beta - u^\beta)^\alpha} du = W_1 \psi(s), \quad \int_s^1 \frac{u^{\beta-1} \psi(u)}{(u^\beta - s^\beta)^\alpha} du = W_2 \psi(s).\tag{24}$$

The matrix W_1 and W_2 are operational matrix of integration of wavelet bases $\psi(u)$ of order $(M+1)2^{k-1} \times (M+1)2^{k-1}$. By using equation (3.3) in equation (3.2), yields

$$\begin{cases} \rho_1(s) C_1^T W_1 \psi(s) + \rho_2(s) C_2^T W_2 \psi(s) = F_1^T \psi(s), \\ \omega_1(s) C_1^T W_2 \psi(s) + \omega_2(s) C_2^T W_1 \psi(s) = F_2^T \psi(s). \end{cases}\tag{25}$$

By solving this system using a direct method, the unknown coefficients C_1^T , C_2^T are obtained. Finally, an approximation of the solution of (1.1) is given by (3.1).

4 Best approximation errors

The aim of this section is to give some estimates for the error of the Jacobi wavelets approximation of a function u in terms of Sobolev norms and seminorms. With this purpose, we extend to the case of Jacobi wavelets some results which were obtained in [8] for the best approximation error by Jacobi polynomials in Sobolev spaces. The main result of this section is Theorem, which establishes a relationship between the regularity of a given function and the convergence rate of its approximation by Jacobi wavelets. At the first, We recall that the Sobolev norm of integer order $\mu \geq 0$ is given by

$$\|u\|_{H_w^r(a,b)} = \left(\sum_{j=0}^r \int_a^b |u^{(j)}(t)|^2 w(t) dt \right)^{\frac{1}{2}} = \left(\sum_{j=0}^r \|u^{(j)}\|_{L_w^2(a,b)}^2 \right)^{\frac{1}{2}},\tag{26}$$



where $u^{(j)}$ denotes the derivative of u of order j .

Theorem: Suppose that $u \in H_{\omega_k}^\mu$ with $\mu \geq 0$

$$\psi_{k,M}(u) = \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M u_{n,m} \psi_{n,m}(t),$$

is the best approximation of u based on the Jacobi wavelets. Then,

$$\|u - \psi_{k,M}(u)\|_{L_{\omega_k}^2(0,T)} \leq CM^{-\mu} |u|_{H_{\omega_k}^{0;\mu;M;k}(0,T)},$$

where the constant C denotes a positive constant that is independent of M and k but depends on the length T .

5 Illustrative Example

In this section, an example is given to demonstrate the applicability and accuracy of our method.

Example.

Consider equation (1.1) with ρ_1, ρ_2, ω_1 and ω_2 are unity and $\beta = 1, \alpha = \frac{1}{3}$ for the pairs:

$$I_1(s) = e^s \left[\Gamma\left(\frac{2}{3}\right) - \Gamma\left(\frac{2}{3}, s\right) \right] + \frac{e^{2s}(1-s)^{\frac{2}{3}} \left[\Gamma\left(\frac{2}{3}\right) - \Gamma\left(\frac{2}{3}, s-1\right) \right]}{2^{\frac{2}{3}}(s-1)^{\frac{2}{3}}},$$

$$I_2(s) = \frac{e^s(1-s)^{\frac{2}{3}} \left[\Gamma\left(\frac{2}{3}\right) - \Gamma\left(\frac{2}{3}, s\right) \right]}{(s-1)^{\frac{2}{3}}} + \frac{e^{2s}(1-s)^{\frac{2}{3}} \left[\Gamma\left(\frac{2}{3}\right) - \Gamma\left(\frac{2}{3}, s-1\right) \right]}{2^{\frac{2}{3}}},$$

the exact solutions are $\varepsilon_1(u) = e^u, \varepsilon_2(u) = e^{2u}$.

Fig 1. Comparison of absolute errors with noises for $m = 7$.

6 Conclusion

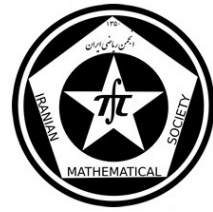
We have introduced a numerical algorithm for solving the system of generalized Abel integral equations. The error analysis carried out in this paper indicates the validity and applicability of the proposed method for smooth functions.



References

- [1] R. Gorenflo, S. Vessella, *Abel Integral Equations: Analysis and Applications*, Springer-Verlag, Berlin, 1991.
- [2] M. E. Fettis, *On numerical solution of equations of the Abel type*, Math. Comp., 18 (1964) 491–496.
- [3] R. Piessens, P. Verbaeten, *Numerical solution of the Abel integral equation*, JBIT, 13 (1973) 451–457.
- [4] S. Piessens, *Computing integral transforms and solving integral equations using Chebyshev polynomial approximations*, J. Comput. Appl. Math., 121 (2000) 113–124.
- [5] D.A. Murio, D.G. Hinestroza, C.W. Mejia, *New stable numerical inversion of Abels integral equation*, Comput. Math. Appl., 11 (1992) 3–11.
- [6] J. Garza, P. Hall, F.H. Ruymagaart, *A new method of solving noisy Abel-type equations*, J. Math. Anal. Appl., 257 (2001) 403–419.
- [7] P. Hall, R. Paige, F.H. Ruymagaart, *Using wavelet methods to solve noisy Abel-type equations with discontinuous inputs*, J. Multivariate. Anal., 86 (2003) 72–96.
- [8] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang, *Spectral methods, Scientific Computation*, Springer- Verlag, Berlin, 2006.

s.sedaghat@bzte.ac.ir



A new double-step iteration method for solving indefinite complex linear system of equations

Maeddeh Pourbagher[†], Davod Khojasteh Salkuyeh^{‡2}

Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran^{†‡}

mapourbagher@gmail.com[†], khojasteh@guilan.ac.ir[‡]

Abstract

We present a new iteration method, called symmetric positive definite and negative stable splitting (SNSS) method for solving indefinite complex symmetric linear systems. Theoretical analysis shows that the proposed method is convergent under suitable conditions. In each iteration of the method two subsystems should be solved. One of them can be solved inexactly using the conjugate gradient method, and the second one by the Chebyshev acceleration method in conjunction with the well-known PRESB preconditioner. Numerical experiments are reported to demonstrate the efficiency of the SNSS method.

AMS subject Classification 2010: 65F10, 65F50, 65F08.

Keywords: Complex, symmetric, symmetric positive definite, negative stable, indefinite.

1 Introduction

We consider the equation

$$M\ddot{q} + C\dot{q} + Kq = p, \quad (27)$$

where q is configuration vector, p is the vector of generalized components of dynamic forces, matrices M , K and $C \in \mathbb{R}^{n \times n}$ are the inertia, stiffness and viscous damping matrices, respectively. Also, $p(t) = f(\omega)e^{i\omega t}$ is complex harmonic excitation and ω is driving circular frequency such that $\omega > 0$. The steady state solution of Eq. (27) is $q(t) = \tilde{q}(\omega)e^{i\omega t}$, with $\tilde{q}(\omega)$ being the solution of the linear system $E(\omega)\tilde{q}(\omega) = f(\omega)$, in which $E(\omega)$ is the dynamic impedance matrix. Substituting $q(t) = \tilde{q}(\omega)e^{i\omega t}$ into Eq. (27) leads to the following complex symmetric linear system of equations

$$A\tilde{q} \equiv (-\omega^2 M + K + i\omega C)\tilde{q} = f, \quad (28)$$

where M , K and C are symmetric positive definite (SPD) matrices, and $i = \sqrt{-1}$. For simplicity in writing, we indicate $\tilde{q}(\omega)$ by $\tilde{q}$. This type of complex symmetric linear systems come from many problems in scientific computing and engineering applications, such as structural dynamics [3] and Helmholtz equations [4]. There are several methods for solving the system (28), when both of the matrices $-\omega^2 M + K$ and ωC are symmetric positive semidefinite (SPSD) with at least one of them being positive definite. For example Salkuyeh et al. applied the successive overrelaxation (SOR) method for solving this system [5]. The transformed matrix iteration method was presented by Axelsson and Salkuyeh in [1]. However, when the driving circular frequency ω is large enough, $-\omega^2 M + K$ is symmetric indefinite and then all

²Speaker



the aforementioned methods may fail to converge. In this case, Li and Wu in [6] proposed the modified positive/negative stable splitting (MPNS) method. In this paper, we present a two-parameter iteration method which is called the symmetric positive definite and negative stable splitting (SNSS) method when $-\omega^2 M + K$ is symmetric indefinite.

2 The SNSS method

Multiplying both sides of Eq. (28) by -1 and then adding $\alpha\omega C$, we get

$$(\alpha\omega C + \omega^2 M)\tilde{q} = ((\alpha + i)\omega C + K)\tilde{q} - f, \quad (29)$$

On the other hand, by adding $i\beta\omega C$ to the both sides of (28), we obtain

$$(i(\beta + 1)\omega C + K)\tilde{q} = (i\beta\omega C + \omega^2 M)\tilde{q} + f. \quad (30)$$

Now using Eqs. (29) and (30) we state the SNSS iteration method as

$$\begin{cases} (\alpha\omega C + \omega^2 M)\tilde{q}^{(k+\frac{1}{2})} = ((\alpha + i)\omega C + K)\tilde{q}^{(k)} - f, \\ (i(\beta + 1)\omega C + K)\tilde{q}^{(k+1)} = (i\beta\omega C + \omega^2 M)\tilde{q}^{(k+\frac{1}{2})} + f, \end{cases} \quad (31)$$

where $\tilde{q}^{(0)} \in \mathbb{C}^n$ is an initial guess and α is a given positive constant. By eliminating the vector $\tilde{q}^{(k+\frac{1}{2})}$ from Eq. (31), we get $\tilde{q}^{(k+1)} = G_{\alpha,\beta}\tilde{q}^{(k)} + f_{\alpha,\beta}$, where

$$G_{\alpha,\beta} = (i(\beta + 1)\omega C + K)^{-1}(i\beta\omega C + \omega^2 M)(\alpha\omega C + \omega^2 M)^{-1}((\alpha + i)\omega C + K),$$

and $f_{\alpha,\beta} = \omega(\alpha - i\beta)(i(\beta + 1)\omega C + K)^{-1}C(\alpha\omega C + \omega^2 M)^{-1}f$, in which $G_{\alpha,\beta}$ is the iteration matrix of the SNSS method. If we define

$$\begin{aligned} B_{\alpha,\beta} &= \frac{1}{\omega(\alpha - i\beta)}(\alpha\omega C + K)C^{-1}(i(\beta + 1)\omega C + \omega^2 M), \\ C_{\alpha,\beta} &= \frac{1}{\omega(\alpha - i\beta)}(i\beta\omega C + \omega^2 M)C^{-1}((\alpha + i)\omega C + \omega^2 M), \end{aligned}$$

then

$$A = B_{\alpha,\beta} - C_{\alpha,\beta}, \quad \text{and} \quad G_{\alpha,\beta} = B_{\alpha,\beta}^{-1}C_{\alpha,\beta}. \quad (32)$$

It follows from Eq. (32) that $B_{\alpha,\beta}^{-1}A = I - G_{\alpha,\beta}$. Therefore, the matrix $B_{\alpha,\beta}$ can be used as a preconditioner for the system (28). The next theorem states the convergence of the SNSS iteration method.

Theorem 0.3. *Let $A = -\omega^2 M + K + i\omega C$ such that M , K and C are SPD matrices. Let also $\lambda_{\max}$ be the largest eigenvalue of $\tilde{M} = \omega C^{-\frac{1}{2}} M C^{-\frac{1}{2}}$ and $\mu_{\min}$ be the smallest eigenvalue of $\tilde{K} = \frac{1}{\omega} C^{-\frac{1}{2}} K C^{-\frac{1}{2}}$. If $\beta > \max\{0, \frac{1}{2}(\lambda_{\max}^2 - \mu_{\min}^2 - 1)\}$, and α is large enough, then $\rho(G_{\alpha,\beta}) < 1$, where $G_{\alpha,\beta}$ is the iteration matrix of the SNSS method.*



Proof. Clearly,

$$\begin{aligned}
\rho(G_{\alpha,\beta})^2 &= \rho \left((i\beta\omega C + \omega^2 M)(\alpha\omega C + \omega^2 M)^{-1}((\alpha + i)\omega C + K)(i(\beta + 1)\omega C + K)^{-1} \right)^2 \\
&= \rho \left((\omega C)^{\frac{1}{2}}(i\beta I + \tilde{M})(\alpha I + \tilde{M})^{-1}((\alpha + i)I + \tilde{K})(i(\beta + 1)I + \tilde{K})^{-1}(\omega C)^{-\frac{1}{2}} \right)^2 \\
&= \rho \left((i\beta I + \tilde{M})(\alpha I + \tilde{M})^{-1}((\alpha + i)I + \tilde{K})(i(\beta + 1)I + \tilde{K})^{-1} \right)^2 \\
&\leq \left\| (i\beta I + \tilde{M})(\alpha I + \tilde{M})^{-1}((\alpha + i)I + \tilde{K})(i(\beta + 1)I + \tilde{K})^{-1} \right\|_2^2 \\
&\leq \left\| (i\beta I + \tilde{M})(\alpha I + \tilde{M})^{-1} \right\|_2^2 \left\| ((\alpha + i)I + \tilde{K})(i(\beta + 1)I + \tilde{K})^{-1} \right\|_2^2 \\
&= \max_{\lambda_i \in \sigma(\tilde{M})} \frac{\beta^2 + \lambda_i^2}{(\alpha + \lambda_i)^2} \max_{\mu_j \in \sigma(\tilde{K})} \frac{(\alpha + \mu_j)^2 + 1}{(\beta + 1)^2 + \mu_j^2}, \tag{33}
\end{aligned}$$

where $\tilde{M} = (\omega C)^{-\frac{1}{2}}(\omega^2 M)(\omega C)^{-\frac{1}{2}} = \omega C^{-\frac{1}{2}} M C^{-\frac{1}{2}}$ and $\tilde{K} = (\omega C)^{-\frac{1}{2}} K (\omega C)^{-\frac{1}{2}} = \frac{1}{\omega} C^{-\frac{1}{2}} K C^{-\frac{1}{2}}$. Evidently, the matrices $\tilde{M}$ and $\tilde{K}$ are SPD. Therefore, for every $\lambda_i \in \sigma(\tilde{M})$ and $\mu_j \in \sigma(\tilde{K})$ we have $\lambda_i > 0$ and $\mu_j > 0$. There exist $1 \leq k \leq n$ and $1 \leq r \leq n$ such that

$$\max_{\lambda_i \in \sigma(\tilde{M})} \frac{\beta^2 + \lambda_i^2}{(\alpha + \lambda_i)^2} = \frac{\beta^2 + \lambda_k^2}{(\alpha + \lambda_k)^2} \quad \text{and} \quad \max_{\mu_j \in \sigma(\tilde{K})} \frac{(\alpha + \mu_j)^2 + 1}{(\beta + 1)^2 + \mu_j^2} = \frac{(\alpha + \mu_r)^2 + 1}{(\beta + 1)^2 + \mu_r^2}.$$

Therefore, it follows from Eq. (33) that

$$\rho(G_{\alpha,\beta})^2 \leq \frac{\beta^2 + \lambda_k^2}{(\alpha + \lambda_k)^2} \frac{(\alpha + \mu_r)^2 + 1}{(\beta + 1)^2 + \mu_r^2} = \frac{\beta^2 + \lambda_k^2}{(\beta + 1)^2 + \mu_r^2} \frac{(\alpha + \mu_r)^2 + 1}{(\alpha + \lambda_k)^2} =: f(\beta)g(\alpha), \tag{34}$$

where $f(\beta) = \frac{\beta^2 + \lambda_k^2}{(\beta + 1)^2 + \mu_r^2}$ and $g(\alpha) = \frac{(\alpha + \mu_r)^2 + 1}{(\alpha + \lambda_k)^2}$. It is easy to see that $f(\beta) < 1$ is equivalent to $\beta > (\lambda_k^2 - \mu_r^2 - 1)/2$. Hence to have $f(\beta) < 1$, it is enough to have $\beta > \max\{0, (\lambda_{\max}^2 - \mu_{\min}^2 - 1)/2\}$. On the other hand, $\lim_{\alpha \rightarrow +\infty} \frac{(\alpha + \mu_r)^2 + 1}{(\alpha + \lambda_k)^2} = 1$. Thus, for every $\epsilon > 0$, there exists a $P > 0$ such that for all $\alpha > P$, $\left| \frac{(\alpha + \mu_r)^2 + 1}{(\alpha + \lambda_k)^2} - 1 \right| < \epsilon$. So for $\alpha > P$, we have $0 \leq g(\alpha) = \frac{(\alpha + \mu_r)^2 + 1}{(\alpha + \lambda_k)^2} < 1 + \epsilon$. Therefore, from (34) we deduce that $\rho(G_{\alpha,\beta})^2 \leq f(\beta)g(\alpha) < f(\beta)(1 + \epsilon)$. Hence, for the convergence we need to have $f(\beta)(1 + \epsilon) < 1$ and this can be achieved if the parameter ϵ satisfies $\epsilon < \frac{1}{f(\beta)} - 1$. $\square$

3 Inexact version of SNSS

Let $\tilde{q}^{(k)}$ be the computed solution at iteration k . Setting $\tilde{q}^{(k+\frac{1}{2})} = \tilde{q}^{(k)} + \gamma^{(k)}$ and substituting $\tilde{q}^{(k+\frac{1}{2})}$ in the first half-step of (31), we get $(\alpha\omega C + \omega^2 M)\gamma^{(k)} = A\tilde{q}^{(k)} - f =: -r^{(k)}$. For computing $\gamma^{(k)}$, since the matrix $\alpha\omega C + \omega^2 M$ is SPD, we can use the conjugate gradient (CG) method or its preconditioned version. Similarly, by setting $\tilde{q}^{(k+1)} = \gamma^{(k+\frac{1}{2})} + \tilde{q}^{(k+\frac{1}{2})}$ and substituting $\tilde{q}^{(k+1)}$ in the second half-step in (31), we get $(i(\beta + 1)\omega C + K)\gamma^{(k+\frac{1}{2})} = f - A\tilde{q}^{(k+\frac{1}{2})} =: r^{(k+\frac{1}{2})}$. This system is equivalent to

$$\mathcal{A}u \equiv \begin{pmatrix} K & -(\beta + 1)\omega C \\ (\beta + 1)\omega C & K \end{pmatrix} \begin{pmatrix} \gamma_1^{(k+\frac{1}{2})} \\ \gamma_2^{(k+\frac{1}{2})} \end{pmatrix} = \begin{pmatrix} r_1^{(k+\frac{1}{2})} \\ r_2^{(k+\frac{1}{2})} \end{pmatrix} \equiv c, \tag{35}$$



where $\gamma^{(k+\frac{1}{2})} = \gamma_1^{(k+\frac{1}{2})} + i\gamma_2^{(k+\frac{1}{2})}$, $r^{(k+\frac{1}{2})} = r_1^{(k+\frac{1}{2})} + ir_2^{(k+\frac{1}{2})}$. To solve the system (524), we can use the preconditioner matrix PRESB (see [1, 2]). In [2], Axelsson et al. showed that the eigenvalues of $\mathcal{P}_{PRESB}^{-1}\mathcal{A}$ are contained in the interval $[\frac{1}{2}, 1]$, where $\mathcal{P}_{PRESB}$ is the PRESB preconditioner. In this case, the lower and upper bounds for the eigenvalues $\mathcal{P}_{PRESB}^{-1}\mathcal{A}$ are $\eta = \frac{1}{2}$ and $\zeta = 1$, respectively. So the Chebyshev acceleration method (see [8]) can be used for solving the system (524).

Algorithm 1. The ISNSS iteration method

1. Choose an initial guess $\tilde{q}^{(0)}$;
2. For $k = 0, 1, 2, \dots$, until convergence, Do
3. Compute $r^{(k)} = f - A\tilde{q}^{(k)}$;
4. Solve system $(\alpha\omega C + \omega^2 M)\gamma^{(k)} = -r^{(k)}$ approximately for $\gamma^{(k)}$ using PCG;
5. $\tilde{q}^{(k+\frac{1}{2})} = \tilde{q}^{(k)} + \gamma^{(k)}$;
6. Compute $r^{(k+\frac{1}{2})} = f - A\tilde{q}^{(k+\frac{1}{2})}$;
7. Solve $\mathcal{A}u = c$ using Chebyshev acceleration with preconditioner $\mathcal{P}_{PRESB}$;
8. $\tilde{q}^{(k+1)} = \gamma^{(k+\frac{1}{2})} + \tilde{q}^{(k+\frac{1}{2})}$;
9. EndDo

4 Parameters estimation

By using the idea of Ren and Cao [7], we present a strategy for estimating the iteration parameters α and β of the SNSS iteration method. We rewrite the matrix $C_{\alpha,\beta}$ as

$$C_{\alpha,\beta} = -i^2 C_{\alpha,\beta} = \frac{1}{\omega(i\beta - \alpha)}(-\beta\omega C + i\omega^2 M)C^{-1}(\alpha i\omega C + iK - \omega C),$$

and define the function ψ as

$$\psi(\alpha, \beta) = (-\beta\omega\|C\|_F + \omega^2\|M\|_F)\|C^{-1}\|_F(\alpha\omega\|C\|_F + \|K\|_F - \omega\|C\|_F).$$

To estimate the iteration parameters α and β , we set $-\beta\omega\|C\|_F + \omega^2\|M\|_F = 0$ and $\alpha\omega\|C\|_F + \|K\|_F - \omega\|C\|_F = 0$. Since the matrix C is SPD, we have $\|C\|_F \neq 0$ and $\|C^{-1}\|_F \neq 0$. Hence, we obtain the following estimations for the iteration parameters α and β of the SNSS iteration method

$$\alpha_{est} = 1 - \frac{\|K\|_F}{\omega\|C\|_F} \quad \text{and} \quad \beta_{est} = \frac{\omega\|M\|_F}{\|C\|_F}.$$

Note that if $\alpha_{est} < 0$, then we apply the above strategy for the function $-\psi$. In this case, we get $\alpha_{est} = \frac{\|K\|_F}{\omega\|C\|_F} - 1$. Summarizing the above discussion results in the the following estimations for the parameters:

$$(\alpha_{est}, \beta_{est}) = \left(\left| 1 - \frac{\|K\|_F}{\omega\|C\|_F} \right|, \frac{\omega\|M\|_F}{\|C\|_F} \right). \quad (36)$$



5 Numerical experiments

We consider the system (28) with $M = I$ and $C = \omega C_V + C_H$, where $C_V = 5M$, $C_H = \mu K$ with a damping coefficient $\mu = 0.02$ and K the five-point centered difference matrix approximating the negative Laplacian operator with homogeneous Dirichlet boundary conditions, on a uniform mesh in the unit square $[0, 1] \times [0, 1]$ with the mesh size $h = 1/(m+1)$. In this case, we have $K = (I \otimes V_m + V_m \otimes I) \in \mathbb{R}^{n \times n}$ with $V_m = h^{-2} \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$. Hence, the total number of variables is $n = m^2$. In addition, the right-hand side vector f is to be adjusted such that $f = (1+i)Ae$, where $e = (1, 1, \dots, 1)^T \in \mathbb{R}^n$.

We compare the numerical results of ISNSS with those of the GMRES(10) and the incomplete version of MPNS (IMPNS) methods. In both of the ISNSS and IMPNS methods, the first subsystem is solved using the preconditioned CG (PCG) method incorporated with the incomplete Cholesky factorization with dropping tolerance 10^{-2} as a preconditioner. In the MATLAB notation the following command can be used for computing the incomplete Cholesky factor

```
L=ichol(Z,struct('type','ict','droptol',1e-2));
```

where Z is a given SPD matrix. The second subsystem is solved by the Chebyshev acceleration method in conjunction with the PRESB preconditioner. Note that in all of the methods, in implementation of $\mathcal{P}_{PRESB}$, we solve inner systems exactly using the sparse Cholesky factorization incorporated with the symmetric approximate minimum degree permutation. To do so, the `symamd.m` command of MATLAB are applied. We use a null vector as an initial guess. The outer iteration is stopped as soon as the residual 2-norm is reduced by a factor of 10^6 and the inner iteration by a factor of 10^2 . The maximum number of inner and outer iterations are set to be 10000. A dagger ($\dagger$) shows that the method fails to converge in the maximum number of iterations. Numerical results are presented in Table 1 for $m = 256$ and different values of ω . In this table we report the number of outer iterations (“Iters”), elapsed CPU time in seconds (“CPU”) and $R_k = \|r^{(k)}\|_2 / \|f\|_2$, where $r^{(k)}$ is the residual vector of the computed solution. For the ISNSS method, we use the estimation values of parameters α and β presented in (36). As the numerical results show the ISNSS outperforms the GMRES(10) and IMPNS method from both the CPU time and the iterations point of view. On the other hand, by increasing the parameter ω , the number of iterations are increased moderately.

References

- [1] O. Axelsson, D.K. Salkuyeh, A new version of a preconditioning method for certain two-by-two block matrices with square blocks, BIT Numer. Math., 59 (2018) 321–342.
- [2] O. Axelsson, M. Neytcheva and Z.-Z. Liang, Parallel solution methods and preconditioners for evolution equations, Math. Model. Anal., 23 (2018) 287–308.
- [3] Z.-Z. Bai, M. Benzi and F. Chen, Modified HSS iteration methods for a class of complex symmetric linear systems, Computing, 87 (2010) 93–111.
- [4] A. Bayliss, C.I. Goldstein, E. Turkel, An iterative method for Helmholtz equation, J. Comput. Phys., 49 (1983) 443–457.



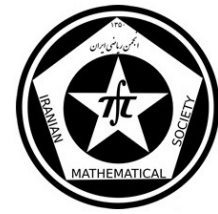
Table 1: Numerical results for $m = 256$ ($n = 65536$).

Method		$\omega = 1$	$\omega = 3$	$\omega = 5$	$\omega = 7$	$\omega = 9$
GMRES(10)	Iter	10000	6423	3068	1740	1119
	CPU	109.24	65.90	31.58	18.35	11.52
	R_k	1.30e-06	9.99e-07	9.98e-07	9.97e-07	9.94e-07
IMPNS	α_{opt}	720	1400	2410	3570	4780
	Iter	240	225	203	182	174
	CPU	28.52	25.55	20.18	18.04	17.04
	R_k	9.96e-07	9.86e-07	9.78e-07	9.92e07	9.65e-07
ISNSS	α_{est}	48.9621	15.6288	8.9622	6.1051	4.5179
	β_{est}	0.0001	0.0005	0.0008	0.0011	0.0015
	Iter	3	5	6	7	8
	CPU	0.80	1.01	1.07	1.27	1.35
	R_k	2.88e-07	8.94e-07	8.73e-07	5.55e07	2.75e-07

- [5] D.K. Salkuyeh, D. Hezari, V. Edalatpour, Generalized SOR iterative method for a class of complex symmetric linear system of equations, Intern. J. Comput. Math., 92 (2015) 802–815.
- [6] C.-X. Li, S.-L. Wu, A splitting method for complex symmetric indefinite linear system, J. Comput. Appl. Math., 313 (2017) 343–354.
- [7] Z.-R. Ren, Y. Cao, An alternating positive-semidefinite splitting preconditioner for saddle point problems from time-harmonic eddy current models, IMA J. Numer. Anal, 36 (2016) 922–946.
- [8] Y. Saad, Iterative methods for sparse linear systems, Second Edition, SIAM, (2003).

mapourbagher@gmail.com

khojasteh@guilan.ac.ir



An Extension of Jung's Theorem

Somayyeh Jangjooye Shaldehi [†]Faculty of Mathematical Sciences, Alzahra University, s.jangjoo@alzahra.ac.ir[†]

Abstract

We extend Jung's result about the relations among bi-closing, open and constant-to-one codes between general shift spaces to closing codes. We also show that any closing factor code $\varphi : X \rightarrow Y$ has a degree d , and it is proved that d is the minimal number of preimages of points in Y . Some other properties of closing codes are provided.

AMS subject Classification 2010: 37B10, 54H20.

Keywords: closing code, degree, open, constant-to-one.

1 Introduction

Closing codes are important in symbolic dynamics. A classical consideration of them is in [3, 4]. Jung showed that these maps have a close relation with open codes [2]. They also have a very natural description from the viewpoint of hyperbolic dynamics [1]: right-closing codes are injective on unstable sets while left-closing codes are injective on stable sets.

There are some relations among bi-closing, open and constant-to-one codes. In [5], Nasu showed that in the category of irreducible shifts of finite type, these conditions are equivalent. Jung extended this result to the general shift spaces and showed that any two of these properties imply the third [2]. In Section , we prove that any closing factor code $\varphi : X \rightarrow Y$ has a degree d and also, we show that d is the minimal number of preimages of points in Y (Theorem 0.6). Then, Corollary 0.8 states that any open closing code from a shift space to an irreducible shift space is bi-closing. After that, we extend Jung's result to closing codes (Theorem 0.10).

2 Background and Notations

This section is devoted to the basic definitions in symbolic dynamics. The notations has been taken from [4]. Given a non-empty finite set $\mathcal{A}$, the full $\mathcal{A}$ -shift, denoted by $\mathcal{A}^{\mathbb{Z}}$, is the collection of all bi-infinite sequences of symbols from $\mathcal{A}$. A word (or block) over $\mathcal{A}$ is a finite sequence of symbols from $\mathcal{A}$. The *shift map* on $\mathcal{A}^{\mathbb{Z}}$ is the map $\sigma : \mathcal{A}^{\mathbb{Z}} \rightarrow \mathcal{A}^{\mathbb{Z}}$ where $(\sigma x)_i = x_{i+1}$. The pair $(\mathcal{A}^{\mathbb{Z}}, \sigma)$ is called the *full shift*. Let $\mathcal{F}$ be a set of words over $\mathcal{A}$. Define $X_{\mathcal{F}}$ to be the subset of $\mathcal{A}^{\mathbb{Z}}$ which do not contain any word in $\mathcal{F}$. A *shift space* is a closed subset X of $\mathcal{A}^{\mathbb{Z}}$ such that $X = X_{\mathcal{F}}$ for some set $\mathcal{F}$ of forbidden words. A shift space X is called *irreducible* if for every ordered pair of words $u, v \in \mathcal{B}(X)$, there is a word $w \in \mathcal{B}(X)$ such that $uwv \in \mathcal{B}(X)$.



Let $\mathcal{B}_n(X)$ be the set of all admissible n -words and $\mathcal{B}(X) = \bigcup_{n=0}^{\infty} \mathcal{B}_n(X)$. Given $u \in \mathcal{B}(X)$, the *cylinder* ${}_l[u]$ is the set $\{x \in X : x_{[l, l+|u|-1]} = u\}$. For $l \geq 0$ and $|u| = 2l + 1$, ${}_l[u]$ is called a *central* $2l + 1$ *cylinder*.

Let $\mathcal{A}$ and $\mathcal{D}$ be alphabets and X a shift space over $\mathcal{A}$. For $m, n \in \mathbb{Z}$ with $-m \leq n$, the $(m+n+1)$ -*block map* $\Phi : \mathcal{B}_{m+n+1}(X) \rightarrow \mathcal{D}$ is defined by

$$y_i = \Phi(x_{i-m}x_{i-m+1}\dots x_{i+n}) = \Phi(x_{[i-m, i+n]}) \quad (37)$$

where $y_i \in \mathcal{D}$. Then, the map $\varphi = \Phi_{\infty}^{[-m, n]} : X \rightarrow \mathcal{D}^{\mathbb{Z}}$ defined by $\varphi(x) = y$ with y_i given by (37) is called the *code induced by* Φ . If $m = n = 0$, then φ is called a *1-block code* and $\varphi = \Phi_{\infty}$. An onto (resp. invertible) code $\varphi : X \rightarrow Y$ is a *factor code* (resp. *conjugacy*). Then, X is called an *extension* of Y .

A code $\varphi : X \rightarrow Y$ is *finite-to-one* if there is a positive integer M such that $|\varphi^{-1}(y)| \leq M$ for every $y \in Y$ and if the points in the image have the same number of preimages, φ is called *constant-to-one*.

3 Main Results

In this section, we investigate some properties of closing codes. First we show that any closing factor code $\varphi : X \rightarrow Y$ has a degree and it is the minimal number of preimages of points in Y . Then, we extend Jung's result [2] about relations among bi-closing, open and constant-to-one codes to the class of closing codes and conclude that any open closing code is bi-closing.

A 1-block code $\varphi = \Phi_{\infty} : X \rightarrow Y$ is called *right-resolving* if whenever $ab, ac \in \mathcal{B}_2(X)$ with $\Phi(b) = \Phi(c)$, then $b = c$. A pair $x, \bar{x}$ of points in a shift space X is *left-asymptotic* if there exists an integer N for which $x_{(-\infty, N]} = \bar{x}_{(-\infty, N]}$. A code $\varphi : X \rightarrow Y$ is called *right-closing* if whenever x and $\bar{x}$ are left-asymptotic and $\varphi(x) = \varphi(\bar{x})$, then $x = \bar{x}$. A right-closing code can be recoded to a right-resolving code. *Right-asymptotic points* and *left-closing* (resp. *resolving*) *codes* are defined similarly. If φ is left or right-closing, it is called *closing* and we call φ *bi-closing* (resp. *bi-resolving*) if it is both left and right-closing (resp. resolving).

A point x in a subshift X is called *doubly transitive* if every word in X appears in x infinitely often to the left and to the right. Let $\varphi : X \rightarrow Y$ be a factor code. If there exists a $d \in \mathbb{N}$ such that every doubly transitive point of Y has exactly d preimages, d is called the *degree* of φ . Then, φ is d -to-one almost everywhere.

Even when a factor code $\varphi : X \rightarrow Y$ is finite-to-one, the degree of φ need not be defined. Jung [2] proved that bi-closing codes have a degree. Now we extend this result to closing factor codes. For this, we will use quantity d^* .

Definition 0.4. Suppose $\varphi = \Phi_{\infty} : X \rightarrow Y$ is a 1-block code. For $w = w_1 \dots w_m \in \mathcal{B}_m(Y)$ and $1 \leq i \leq m$, we define $d^*(w, i)$ to be the number of distinct symbols that you can see at coordinate i in preimages of the word w . Now put

$$d^* = \min\{d^*(w, i) : w \in \mathcal{B}(Y), 1 \leq i \leq |w|\}.$$

A *magic word* is a word w such that $d^*(w, i) = d^*$ for some i . Then, the index i is called a *magic coordinate*.

Theorem 0.5. Let $\varphi : X \rightarrow Y$ be a resolving 1-block factor code and Y an irreducible shift space. Then, φ has a degree d and $d = d^*$. Furthermore, $|\varphi^{-1}(y)| \geq d$ for any $y \in Y$.



Proof. Let w be a magic word with the magic coordinate i . So if $y \in Y$, then for each m , there exist at least d^* distinct symbols that we can see at coordinate 0 in the preimages of $y_{[-m,m]}$. Therefore, by compactness, y has at least d^* preimages.

Now we show that any doubly transitive point has at most d^* preimages. Let $y \in Y$ be a doubly transitive point. By definition, w appears in y infinitely often to the left and to the right. Since φ is resolving, the number of preimages of y is at most d^* . Thus, $d = d^*$. $\square$

Recall that if $\varphi : X \rightarrow Y$ is a right-closing code, then there exist a shift space X' and a conjugacy $\psi : X' \rightarrow X$ such that $\varphi \circ \psi$ is right-resolving. So by Theorem 0.5, we have:

Theorem 0.6. *Let $\varphi : X \rightarrow Y$ be a closing factor code and Y an irreducible shift space. Then, φ has a degree d and $|\varphi^{-1}(y)| \geq d$ for any $y \in Y$.*

Jung showed that any open bi-closing code from a shift space to an irreducible subshift is constant-to-one [2, Corollary 2.8]. We give an analogue of this result for closing codes. For this, first recall that any open code from a shift space to an irreducible shift space is onto [2, Lemma 2.1].

Theorem 0.7. *Let $\varphi : X \rightarrow Y$ be an open closing code and Y irreducible. Then, φ is constant-to-one.*

Proof. By Theorem 0.6, φ has a degree d and $|\varphi^{-1}(y)| \geq d$ for all $y \in Y$. Since φ is closing, it is finite-to-one [4, Proposition 8.1.11]. So openness implies that $|\varphi^{-1}(y)| \leq d$ for all $y \in Y$ [2, Lemma 2.5]. $\square$

Corollary 0.8. *Any open closing code from a shift space to an irreducible shift space is bi-closing.*

Proof. Theorem 0.7 implies that such codes are constant-to-one and by [2, Proposition 2.9], any open constant-to-one code is bi-closing. $\square$

A continuous map $f : Y \rightarrow X$ is called a *cross section* of a code $\varphi : X \rightarrow Y$ if $\varphi(f(y)) = y$ for all $y \in Y$. We say that φ has d disjoint cross sections if there are d cross sections $f_i : Y \rightarrow X$ such that $f_i(Y) \cap f_j(Y) = \emptyset$, for all $i \neq j$. Using cross sections, Jung showed that any constant-to-one bi-closing code is open [2, Proposition 4.5]. Using Theorem 0.6, the main ingredients of the proof of [2, Proposition 4.5] works for closing codes as follows.

Theorem 0.9. *Let $\varphi : X \rightarrow Y$ be a d -to-one closing factor code and Y an irreducible shift space. Then, φ is open.*

Proof. We claim that φ has d disjoint cross sections such that the union of their images is X . Without loss of generality, we can assume that $\varphi = \Phi_\infty$ is a right-resolving 1-block code. For each $n \geq 0$ and $w \in \mathcal{B}_{2n+1}(Y)$, define

$$\mathcal{D}(w) = \{x_0 : x \in X, \Phi(x_{[-n,n]}) = w\}.$$

Then by Definition 0.4, $d(w) = |\mathcal{D}(w)| \geq d^*$ and so Theorem 0.5 gives $|\mathcal{D}(w)| \geq d$. Now define $Y_n = \{y \in Y : d(y_{[-n,n]}) > d\}$. Each Y_n is closed and the family of Y_n 's is nested. If for all n , $Y_n \neq \emptyset$, then there is a $y \in \cap Y_n$. Therefore, for each n , there are at least $d+1$ distinct symbols that appear at coordinate 0 in the preimages of $y_{[-n,n]}$ and hence by compactness, y has at least $d+1$ preimages which is a contradiction. So there is an n such that $Y_n = \emptyset$. Thus, $|\mathcal{D}(w)| = d$ for all $w \in \mathcal{B}_{2n+1}(Y)$.



For this n , we can define $g_1, \dots, g_d : \mathcal{B}_{2n+1}(Y) \rightarrow \mathcal{B}_1(X)$ such that for any $w \in \mathcal{B}_{2n+1}(Y)$, $\{g_1(w), \dots, g_d(w)\} = \mathcal{D}(w)$. Then, we define cross sections $f_1, \dots, f_d : Y \rightarrow X$ such that for each $1 \leq i \leq d$, $f_i(y)$ is the unique point in $\varphi^{-1}(y)$ where $f_i(y)_0 = g_i(y_{[-n,n]})$. Now it is sufficient to show continuity.

For each $1 \leq i \leq d$, define an open and closed set

$$V_i = \cup \{_{-n}[u] : u \in \mathcal{B}_{2n+1}(X), (g_i \circ \Phi)(u) = u_0\}.$$

Then, $x \in V_i$ if and only if $x = (f_i \circ \varphi)(x)$. Therefore, $f_i^{-1} = \varphi|_{V_i}$ is an invertible continuous map between compact metric spaces. So, f_i is continuous and the claim is proved.

Now suppose φ is not open. Then, there exist an open set $U \subseteq X$ such that $\varphi(U)$ is not open. So there are a point $x \in U$ and a sequence $\{y^{(n)}\}$ in Y such that $y^{(n)} \rightarrow \varphi(x)$ and $y^{(n)} \notin \varphi(U)$ for any $n \in \mathbb{N}$. Let f be a cross section with $x \in f(Y)$ and let $z^{(n)} = f(y^{(n)})$ for all n . Then, we have $z^{(n)} \rightarrow x$ and since U is open, $z^{(n)} \in U$. But then, $y^{(n)} \in \varphi(U)$ for all large enough n , which is a contradiction. $\square$

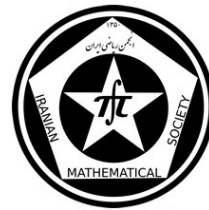
Theorems 0.7, 0.9 and [2, Proposition 2.9] give the following result.

Theorem 0.10. *Let $\varphi : X \rightarrow Y$ be a factor code between shift spaces and Y an irreducible shift space. Then, any two of the following conditions imply the third:*

1. φ is open;
2. φ is constant-to-one;
3. φ is closing.

References

- [1] M. Brin and G. Stuck, *Introduction to Dynamical Systems*, Cambridge: Cambridge University Press, 2002.
- [2] U. Jung, *Open maps between shift spaces*, Ergod. Th. & Dynam. Sys. **29** (2009), 1257-1272.
- [3] B. Kitchens. *Symbolic Dynamics. One-Sided, Two-Sided and Countable State Markov Shifts*, New York: Springer-Verlag, 1998.
- [4] D. Lind and B. Marcus, *An introduction to symbolic dynamics and coding*, Cambridge Univ. Press, 1995.
- [5] M. Nasu, *Constant-to-one and onto global maps of homomorphisms between strongly connected graphs*, Ergod. Th. & Dynam. Sys. **3** (1983), 387-413.



An iteration method for solving the systems arisen from finite element discretization of PDE-constrained optimization problems

Davod Khojasteh Salkuyeh^{†3}, Hamid Mirchi[‡]

Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran[†]

khojasteh@guilan.ac.ir, hamidmirchi@gmail.com

Abstract

A stationary iterative method is presented for solving the system of linear equations arisen from finite element discretization of a distributed optimal control problem with time-periodic parabolic equations. It is shown that under some conditions the method is convergent. Numerical results are presented to show the efficiency of the proposed method.

AMS subject Classification 2010: 49M25, 49K20, 65F10, 65F50.

Keywords: Preconditioner, GMRES, finite element, PDE-constrained, optimization.

1 Introduction

We consider the following distributed control problems: (see [6])

$$\begin{aligned} \min_{y,u} \quad & \frac{1}{2} \int_0^T \int_{\Omega} |y(x,t) - y_d(x,t)|^2 dx dt + \frac{\nu}{2} \int_0^T \int_{\Omega} |u(x,t)|^2 dx dt, \\ \text{s.t.} \quad & \frac{\partial}{\partial t} y(x,t) - \Delta y(x,t) = u(x,t) \quad \text{in } Q_T, \\ & y(x,t) = 0 \quad \text{on } \Sigma_T, \\ & y(x,0) = y(x,T) \quad \text{on } \partial\Omega, \\ & u(x,0) = u(x,T) \quad \text{in } \Omega, \end{aligned}$$

where Ω is an open and bounded domain in $\mathbb{R}^d$ ($d \in \{1, 2, 3\}$), with Lipschitz-continuous boundary $\partial\Omega$. In the above problem, we introduce the space-time cylinder $Q_T = \Omega \times (0, T)$ and its lateral surface $\Sigma_T = \partial Q \times (0, T)$. We also have $T > 0$, $y_d(x, t)$ is a desired state and ν is a regularization parameter. We can assume that $y_d(x, t)$ is time-harmonic, i.e., $y_d(x, t) = y_d(x) e^{i\omega t}$, with $\omega = 2\pi m/T$ for some $m \in \mathbb{Z}$. Substituting $y_d(x, t)$ in the problem and discretizing it by the finite element method, gives the following system of linear equations

$$\begin{pmatrix} M & 0 & K - i\omega M \\ 0 & \nu M & -M \\ K + i\omega M & -M & 0 \end{pmatrix} \begin{pmatrix} \bar{y} \\ \bar{u} \\ \bar{p} \end{pmatrix} = \begin{pmatrix} M \bar{y}_d \\ 0 \\ 0 \end{pmatrix}, \quad (38)$$

³D.K. Salkuyeh



where the symmetric positive definite (SPD) matrices M and K are the mass and stiffness matrices, representing the L^2 -inner product in V_h and the discretized negative Laplacian, respectively. From the second equation in Eq. (38) we have, $\bar{u} = \frac{1}{\nu}\bar{p}$ and by substituting $\bar{u}$ in the third equation, we obtain the following system

$$\begin{cases} M\bar{y} + (K - i\omega M)\bar{p} = M\bar{y}_d, \\ (K + i\omega M)\bar{y} - \frac{1}{\nu}M\bar{p} = 0, \end{cases}$$

which is itself equivalent to

$$\mathcal{A}x = \begin{pmatrix} M & \sqrt{\nu}(K - i\omega M) \\ \sqrt{\nu}(K + i\omega M) & -M \end{pmatrix} \begin{pmatrix} \bar{y} \\ \bar{q} \end{pmatrix} = \begin{pmatrix} M\bar{y}_d \\ 0 \end{pmatrix} = b, \quad (39)$$

where $\bar{q} = \bar{p}/\sqrt{\nu}$. In [3], Krendl proposed the real block diagonal and the alternative indefinite preconditioners for the system (39). Zheng et al. in [6] proposed the block alternating splitting (BAS) iteration method which can be written as

$$\begin{cases} (\alpha V + H_1)x^{k+\frac{1}{2}} = (\alpha V - S_1)x^k + \mathcal{P}_1 b, \\ (\alpha V + H_2)x^{k+1} = (\alpha V - S_2)x^{k+\frac{1}{2}} + \mathcal{P}_2 b. \end{cases}$$

where $\alpha > 0$, $V = \text{blkdiag}(M, M)$, $H_1 = \text{blkdiag}(M, -M)$, $H_2 = \text{blkdiag}(\sqrt{\nu}K, \sqrt{\nu}K)$,

$$S_1 = \frac{1}{1 + \omega^2\nu} \begin{pmatrix} -i\omega\nu K & \sqrt{\nu}K \\ -\sqrt{\nu}K & i\omega\nu K \end{pmatrix}, \quad S_2 = \begin{pmatrix} i\sqrt{\nu}\omega M & -M \\ M & -i\sqrt{\nu}\omega M \end{pmatrix},$$

and

$$\mathcal{P}_1 = \frac{1}{1 + \omega^2\nu} \begin{pmatrix} -I & -i\omega\sqrt{\nu}I \\ i\omega\sqrt{\nu}I & -I \end{pmatrix}, \quad \mathcal{P}_2 = \frac{1}{1 + \omega^2\nu} \begin{pmatrix} 0 & I \\ I & 0 \end{pmatrix}.$$

Numerical results presented in [6] show that the BAS iteration method outperform the GMRES method. They also showed that the parameter $\alpha = 1 + \nu\omega^2$ often gives quite suitable results. Therefore, if $\nu\omega^2 \ll 1$, then $\alpha = 1$ is a good choice.

In this paper we present a new iterative method for the system (39) and investigate its convergence properties.

2 The new method

Let

$$\mathcal{P} = \begin{pmatrix} -i\omega\sqrt{\nu}I & I \\ I & -i\omega\sqrt{\nu}I \end{pmatrix},$$

where $i = \sqrt{-1}$ and I is the identity matrix. Pre-multiplying both sides of (39) by $\mathcal{P}$ gives the following system

$$\mathbf{A}x = \begin{pmatrix} \sqrt{\nu}K & -i\omega\nu K - (1 + \nu\omega^2)M \\ -i\omega\nu K + (1 + \nu\omega^2)M & \sqrt{\nu}K \end{pmatrix} \begin{pmatrix} \bar{y} \\ \bar{q} \end{pmatrix} = \begin{pmatrix} -\sqrt{\nu}i\omega M\bar{y}_d \\ M\bar{y}_d \end{pmatrix} = \mathbf{b}.$$

We now consider the splitting $\mathbf{A} = \mathcal{H} - \mathcal{S}$ with

$$\mathcal{H} = \begin{pmatrix} \sqrt{\nu}K & -(1 + \nu\omega^2)M \\ (1 + \nu\omega^2)M & \sqrt{\nu}K \end{pmatrix}, \quad \mathcal{S} = \begin{pmatrix} 0 & i\omega\nu K \\ i\omega\nu K & 0 \end{pmatrix},$$



and state the single block splitting (SBS) iteration method

$$x^{(k+1)} = \mathcal{H}^{-1} \mathcal{S} x^{(k)} + \mathcal{H}^{-1} \mathbf{b}, \quad (40)$$

In each iteration of the method a system of linear equations with the coefficient matrix $\mathcal{H}$ should be solved. We will shortly see that it can be efficiently solved.

Lemma 0.11. *Assume that the matrices M and K are SPD. Then, the matrix $\mathcal{H}$ is nonsingular.*

Proof. It is easy to see that

$$\begin{aligned} \det(\mathcal{H}) &= \det(\sqrt{\nu}K) \det(\sqrt{\nu}K - (1 + \nu\omega^2)M(\sqrt{\nu}K)^{-1}(-(1 + \nu\omega^2)M)) \\ &= \det(\sqrt{\nu}K) \det\left(\sqrt{\nu}K + \frac{(1 + \nu\omega^2)^2}{\sqrt{\nu}}MK^{-1}M\right) > 0. \end{aligned}$$

So the matrix $\mathcal{H}$ is nonsingular. $\square$

Theorem 0.12. *Assume that the matrices M and K are SPD. Then, the SBS iterative method (40) is convergent if and only if*

$$\frac{\nu\omega\mu_{\max}}{\sqrt{(1 + \omega^2\nu)^2 + \nu\mu_{\max}^2}} < 1,$$

where $\mu_{\max}$ is the largest eigenvalue of $Q = M^{-\frac{1}{2}}KM^{-\frac{1}{2}}$.

Proof. Let $(\lambda, \mathbf{w})$ be an eigenpair of the matrix $\mathcal{G} = \mathcal{H}^{-1}\mathcal{S}$, where $0 \neq \mathbf{w} = (x; y) \in \mathbb{C}^{2m}$. Then, we have $\mathcal{S}\mathbf{w} = \lambda\mathcal{H}\mathbf{w}$, which is itself equivalent to

$$\begin{cases} i\omega\nu Q\bar{y} = \lambda\sqrt{\nu}Q\bar{x} - \lambda(1 + \nu\omega^2)\bar{y}, \\ i\omega\nu Q\bar{x} = \lambda\sqrt{\nu}Q\bar{y} + \lambda(1 + \nu\omega^2)\bar{x}, \end{cases} \quad (41)$$

where $Q = M^{-\frac{1}{2}}KM^{-\frac{1}{2}}$, $\bar{x} = M^{\frac{1}{2}}x$ and $\bar{y} = M^{\frac{1}{2}}y$. From Eq. (41) it is easy to see that $\lambda \neq 0$, $\bar{x} \neq 0$ and $\bar{y} \neq 0$. Computing $\bar{y}$ from the second equation in (41) and substituting in the first one gives

$$(\lambda^2\nu + (\nu\omega)^2)Q^2\bar{x} = -\lambda^2(1 + \omega^2\nu)^2\bar{x}. \quad (42)$$

Now if $\lambda^2\nu + (\nu\omega)^2 = 0$, then it follows from (42) that $\bar{x} = 0$, which is a contradiction. Therefore,

$$Q^2\bar{x} = \mu^2\bar{x}, \quad \text{with} \quad \mu^2 = -\frac{\lambda^2(1 + \omega^2\nu)^2}{(\nu\omega)^2 + \lambda^2\nu}. \quad (43)$$

which shows that μ^2 is an eigenvalue of matrix Q^2 . Therefore, it follows from Eq. (43) that $\lambda = \pm f(\mu)i$, where

$$f(\mu) = \frac{\mu\omega\nu}{\sqrt{(1 + \nu\omega^2)^2 + \nu\mu^2}}.$$

Obviously, $|\lambda| = f(\mu)$. Since, the function f is increasing for $\mu > 0$, we deduce that the method is convergent if $f(\mu_{\max}) < 1$, which completes the proof. $\square$

Corollary 0.13. *Under the conditions of Theorem 0.12 the SBS iterative method is convergent in the following cases:*

- (i) $\sqrt{\nu}\omega < 1$;
- (ii) $\nu\omega\mu_{\max} < 4$.



Proof. To prove (i), we have

$$f(\mu_{\max}) \leq \frac{\nu\omega\mu_{\max}}{\sqrt{\nu\mu_{\max}^2}} = \sqrt{\nu}\omega.$$

Now if $\sqrt{\nu}\omega < 1$, then the method is convergent. To prove (ii), using the *Arithmetic Mean-Geometric Mean* inequality we get

$$f(\mu_{\max})^2 = \frac{(\nu\omega\mu_{\max})^2}{(1 + \nu\omega^2)^2 + \nu\mu_{\max}^2} \leq \frac{(\nu\omega\mu_{\max})^2}{2\sqrt{\nu}\mu_{\max}(1 + \nu\omega^2)} \leq \frac{(\nu\omega\mu_{\max})^2}{4\nu\omega\mu_{\max}} = \frac{1}{4}\nu\omega\mu_{\max},$$

which completes the proof. $\square$

3 Implementation issue

We rewrite Eq. (40) as $\mathcal{H}x^{(k+1)} = \mathcal{S}x^{(k)} + \mathbf{b}$. Letting $\delta^{(k)} = x^{(k+1)} - x^{(k)}$, this equation can be written as

$$\mathcal{H}\delta^{(k)} = \mathbf{b} - \mathbf{A}x^{(k)} =: \mathbf{r}^{(k)}. \quad (44)$$

Therefore, in each iteration of the SBS iteration method we first solve Eq. (44) for $\delta^{(k)}$ and then set $x^{(k+1)} = x^{(k)} + \delta^{(k)}$. Since the matrix $\mathcal{H} + \mathcal{H}^T$ is SPD, we deduce that the matrix $\mathcal{H}$ is positive real. Hence, the system (44) can be solved using a Krylov subspace method like GMRES. To expedite the convergence of the GMRES method we can use the PRESB preconditioner (see [1, 2])

$$\mathcal{B} = \begin{pmatrix} \sqrt{\nu}K & -(1 + \nu\omega^2)M \\ (1 + \nu\omega^2)M & \sqrt{\nu}K + 2(1 + \nu\omega^2)M \end{pmatrix}. \quad (45)$$

As the authors of [1] showed the eigenvalues of the preconditioned matrix $\mathcal{B}^{-1}\mathcal{H}$ are contained in the interval $[\frac{1}{2}, 1]$. Hence, the GMRES method as well as the Chebyshev acceleration method [5] are quite suitable for solving the preconditioned system $\mathcal{B}^{-1}\mathcal{H}\delta^{(k)} = \mathcal{B}^{-1}\mathbf{r}^{(k)}$.

In the implementation of the preconditioner $\mathcal{B}$ in each iteration of GMRES or the Chebyshev acceleration method a linear system of equations of the form

$$\mathcal{B} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \\ s \end{pmatrix},$$

should be solved, which is equivalent to

$$\begin{cases} \sqrt{\nu}Kx - (1 + \nu\omega^2)My = r, \\ (1 + \nu\omega^2)Mx + (\sqrt{\nu}K + 2(1 + \nu\omega^2)M)y = s. \end{cases} \quad (46)$$

Adding both sides of the second equation in (46) to the first one, results in the system

$$Tz = r + s, \quad (47)$$

where $T = \sqrt{\nu}K + (1 + \nu\omega^2)M$ and $z = x + y$. The matrix T is SPD. Therefore, this system can be solved exactly using the Cholesky factorization or inexactly using the conjugate gradient method. On the other hand, from the second equation in (46) we obtain

$$Ty = s - (1 + \nu\omega^2)Mz, \quad (48)$$

and solving the system can be treated similar to Eq. (47). Using the above results, to compute the vectors x and y we first solve the system (47) for z . Then from the system (48) we obtain the vector y . Finally, we calculate the vector x using $x = z - y$.

Table 2: Numerical results for $h = 2^{-8}$.

Method	ν	$\omega = 0.1$		$\omega = 1$		$\omega = 10$		$\omega = 100$	
		Its	CPU	Its	CPU	Its	CPU	Its	CPU
SBS-GMRES	10^{-5}	4	1.66	4	1.63	6	2.48	16	6.46
	10^{-6}	4	1.62	4	1.66	4	1.65	8	3.21
	10^{-7}	4	1.69	4	1.66	4	1.73	6	2.45
	10^{-8}	4	1.64	4	1.74	4	1.64	4	1.67
SBS-Cheb	10^{-5}	4	0.80	4	0.81	6	1.31	16	3.54
	10^{-6}	4	0.82	4	0.84	5	1.01	8	1.87
	10^{-7}	4	0.84	4	0.79	4	0.84	6	1.33
	10^{-8}	4	0.84	4	0.85	4	0.80	4	0.90
BAS	10^{-5}	48	7.19	48	7.22	48	7.23	46	6.84
	10^{-6}	47	7.17	47	7.01	47	7.02	47	7.10
	10^{-7}	48	7.25	48	7.11	48	7.13	48	7.17
	10^{-8}	50	7.48	50	7.36	50	7.33	50	7.36

4 Numerical experiments

We consider the distributed control problem in two-dimensional case. The computational domain is the unit square $\Omega = (0, 1) \times (0, 1) \in \mathbb{R}^2$. The target state is chosen as

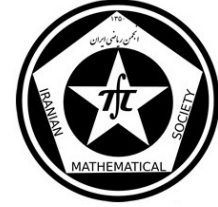
$$y_d(x, y) = \begin{cases} (2x - 1)^2(2y - 1)^2, & \text{if } (x, y) \in (0, \frac{1}{2}) \times (0, \frac{1}{2}), \\ 0, & \text{otherwise.} \end{cases} \quad (49)$$

We present the numerical results for above example. To generate the system (38) we have used the codes of the paper [4] which is available at www.numerical.rl.ac.uk/people/rees/. We solve the system $\mathbf{A}x = \mathbf{b}$ using the SBS iterative method. In each iteration of the method we solve the sub-systems using the GMRES and the Chebyshev acceleration methods in conjunction with the PRESB preconditioner, which hereafter are denoted by SBS-GMRES and SBS-Cheb, respectively. The SBS and the BAS iterative methods are terminated as soon as the residual 2-norm is reduced by a factor of 10^8 . For the sub-systems the iterations are stopped when the residual 2-norm is reduced by a factor of 10^2 and the maximum number of iterations is set to be 10. We always use a zero vector as an initial guess. In the implementation of the PRESB preconditioner the systems with the coefficient T and all the subsystems in the BAS iteration method are solved using the Cholesky factorization of the coefficient matrix in conjunction with the symmetric minimum degree reordering using the `symamd.m` command of MATLAB. In the BAS iteration method the parameter α is set to be 1. All runs are implemented in MATLAB R2017, equipped with a Laptop with 1.80 GHz central processing unit (Intel(R) Core(TM) i7-4500), 6 GB RAM and Windows 7 operating system. We set $h = 2^{-8}$. In this case, the coefficient matrix of the system (38) is of size 195075×195075 . Numerical results have been presented in Table 2. In this table, “Its” and “CPU” stand for the number of iterations and the CPU time (in seconds), respectively. As we see, both of the SBS-GMRES and SBS-Cheb methods outperform the BAS iteration method. It is necessary to mention that the GMRES method (complete version) has not been converged in 200 iterations for the all the tested systems.



References

- [1] O. Axelsson, M. Neytcheva and Z.-Z. Liang, Parallel solution methods and preconditioners for evolution equations, *Math. Model. Anal.*, **23** (2018), 287–308.
- [2] O. Axelsson and D.K. Salkuyeh, A new version of a preconditioning method for certain two-by-two block matrices with square blocks, *BIT Numer. Math.*, **59** (2018), 321–342.
- [3] W. Krendl, V. Simoncini and W. Zulehner, Stability estimates and structural spectral properties of saddle point problems, *Numer. Math.*, **124** (2013), 183–213.
- [4] T. Rees, H.S. Dollar and A.J. Wathen, Optimal solvers for PDE-constrained optimization, *SIAM J. Sci. Comput.*, **32** (2010), 271–298.
- [5] Y. Saad, *Iterative Methods for Sparse Linear Systems*, Second Edition, SIAM, 2003.
- [6] Z. Zheng, G.-F. Zhang and M.-Z. Zhu, A block alternating splitting iteration method for a class of block two-by-two complex linear systems, *J. Comput. Math. Appl.*, **288** (2015), 203–214.



An iterative scheme for generalized Lyapunov matrix equation

Akbar Shirilord^{†4}, Mehdi Dehghan[†] [†]Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences,

Amirkabir University of Technology, No. 424, Hafez Ave., 15914, Tehran, Iran
Emails: akbar.shirilord@aut.ac.ir mdehghan@aut.ac.ir

Abstract

Here we propose a new iterative method for solving the generalized Lyapunov matrix equation on the complex matrices. In each step of this method, two standard Lyapunov matrix equations with real coefficient matrices should be solved. This method uses a real parameter α . Hence we obtain the condition for convergence of the proposed method. Then a numerical test is given and solved via the new method.

AMS subject Classification 2010: 15A30, 15A69, 65F10, 65F30.

Keywords: Iterative scheme, Generalized Lyapunov matrix equation, Spectral radius, Convergence.

1 Introduction

Linear matrix equations play an important role in various applications in sciences and technology. Consider generalized Lyapunov matrix equation

$$\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{A}^T + \sum_{j=1}^m \mathbf{N}_j \mathbf{X} \mathbf{N}_j^T = \mathbf{C}, \quad (50)$$

where $\mathbf{A}$, $\mathbf{N}_j$ ($j = 1, \dots, m$), $\mathbf{C} \in \mathbb{C}^{n \times n}$ are known and $\mathbf{X} \in \mathbb{C}^{n \times n}$ must be determined. Moreover $\mathbf{A}^T$ represents the transpose of matrix $\mathbf{A}$. Eq. (654) reduces to standard Lyapunov equation when $\mathbf{N}_j = 0$ ($j = 1, \dots, m$). Generalized Lyapunov matrix equation (654) has various roles in sciences such as, controllability and model simplification of bilinear systems [4], stability analysis of linear stochastic systems [3]. Linear matrix equation (654) is equivalent to the following linear system

$$(\mathbf{I} \otimes \mathbf{A} + \mathbf{A} \otimes \mathbf{I} + \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j) \text{vec}(\mathbf{X}) = \text{vec}(\mathbf{C}), \quad (51)$$

where $\otimes$ is Kronecker product and $\text{vec}(\mathbf{X})$ is an operator defined as

$$\text{vec} : \mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_n] \mapsto [\mathbf{x}_1^T, \dots, \mathbf{x}_n^T]^T, \quad (52)$$

⁴Speaker



where $\mathbf{x}_k$ is the k -th column of $\mathbf{X}$. Linear system (655) can be solved by many methods such as Krylov subspace methods. Clearly, this transformation increases computational complexity when the sizes of the coefficient matrices in (654) are large. Hence for large problems this approach cannot be applied directly. For example see [1] for numerical Krylov subspace methods for solving (654). Also Dehghan and Shirilord [1] by parameterizing the MHSS method presented a generalized MHSS (GMHSS) iteration method for solving a special case of (654).

2 The EDSS method for solving generalized Lyapunov matrix equation

In this section we extend **DSS** iteration method [6] (also see a good paper in [5]) for solving Eq. (654). Let $\mathbf{A} = \mathbf{U} + i\mathbf{V}$, where $\mathbf{U}$ and $\mathbf{V}$ are symmetric positive definite matrices. We write Eq. (654) as

$$(\mathbf{U} + i\mathbf{V})\mathbf{X} + \mathbf{X}(\mathbf{U} + i\mathbf{V})^T = \mathbf{C} - \sum_{j=1}^m \mathbf{N}_j \mathbf{X} \mathbf{N}_j^T. \quad (53)$$

Now suppose that α is a positive real constant. By multiplying $\alpha - i$ in (661) we get

$$(\alpha\mathbf{U} + \mathbf{V})\mathbf{X} + \mathbf{X}(\alpha\mathbf{U} + \mathbf{V}) = i(\mathbf{U} - \alpha\mathbf{V})\mathbf{X} + i\mathbf{X}(\mathbf{U} - \alpha\mathbf{V}) + (\alpha - i)\mathbf{C} - (\alpha - i) \sum_{j=1}^m \mathbf{N}_j \mathbf{X} \mathbf{N}_j^T, \quad (54)$$

Multiplying $1 - i\alpha$ in (661) concludes

$$(\alpha\mathbf{V} + \mathbf{U})\mathbf{X} + \mathbf{X}(\alpha\mathbf{V} + \mathbf{U}) = i(\alpha\mathbf{U} - \mathbf{V})\mathbf{X} + i\mathbf{X}(\alpha\mathbf{U} - \mathbf{V}) + (1 - i\alpha)\mathbf{C} - (1 - i\alpha) \sum_{j=1}^m \mathbf{N}_j \mathbf{X} \mathbf{N}_j^T. \quad (55)$$

Then by considering relations (239) and (55) we obtain the following scheme for solving Eq. (654).

The EDSS procedure for solving Eq. (654)

Compute $\mathbf{X}^{(k+1)} \in \mathbb{C}^{n \times n}$ for $k = 0, 1, 2, \dots$ by using the following scheme such that $\{\mathbf{X}^{(k)}\}_{k=0}^{\infty} \subseteq \mathbb{C}^{n \times n}$ converges:

$$\begin{cases} (\alpha\mathbf{U} + \mathbf{V})\mathbf{X}^{(k+\frac{1}{2})} + \mathbf{X}^{(k+\frac{1}{2})}(\alpha\mathbf{U} + \mathbf{V}) = i(\mathbf{U} - \alpha\mathbf{V})\mathbf{X}^{(k)} + i\mathbf{X}^{(k)}(\mathbf{U} - \alpha\mathbf{V}) \\ \quad + (\alpha - i)\mathbf{C} - (\alpha - i) \sum_{j=1}^m \mathbf{N}_j \mathbf{X}^{(k)} \mathbf{N}_j^T, \\ (\alpha\mathbf{V} + \mathbf{U})\mathbf{X}^{(k+1)} + \mathbf{X}^{(k+1)}(\alpha\mathbf{V} + \mathbf{U}) = i(\alpha\mathbf{U} - \mathbf{V})\mathbf{X}^{(k+\frac{1}{2})} + i\mathbf{X}^{(k+\frac{1}{2})}(\alpha\mathbf{U} - \mathbf{V}) \\ \quad + (1 - i\alpha)\mathbf{C} - (1 - i\alpha) \sum_{j=1}^m \mathbf{N}_j \mathbf{X}^{(k)} \mathbf{N}_j^T, \end{cases} \quad (56)$$

where $\alpha > 0$ is constant and $\mathbf{X}^{(0)} \in \mathbb{C}^{n \times n}$ is an initial guess. It is clear that the matrices $\alpha\mathbf{U} + \mathbf{V}$, $\alpha\mathbf{U} + \mathbf{V}$, $\alpha\mathbf{V} + \mathbf{U}$ and $\alpha\mathbf{V} + \mathbf{U}$ are symmetric positive definite, therefore, the two half-steps involved in iterative method (116) can be solved effectively using efficient direct algorithms. Using Kronecker product, the scheme (116) will have the following structure:



$$\left\{ \begin{aligned} &(\mathbf{I} \otimes (\alpha \mathbf{U} + \mathbf{V}) + (\alpha \mathbf{U} + \mathbf{V}) \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) = (\mathbf{I} \otimes (i\mathbf{U} - i\alpha \mathbf{V}) + (i\mathbf{U} - i\alpha \mathbf{V}) \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k)} \right) \\ &\quad + (\alpha - i) \text{vec}(\mathbf{C}) - (\alpha - i) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right), \\ &(\mathbf{I} \otimes (\alpha \mathbf{V} + \mathbf{U}) + (\alpha \mathbf{V} + \mathbf{U}) \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+1)} \right) = (\mathbf{I} \otimes (i\alpha \mathbf{U} - i\mathbf{V}) + (i\alpha \mathbf{U} - i\mathbf{V}) \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) \\ &\quad + (1 - i\alpha) \text{vec}(\mathbf{C}) - (1 - i\alpha) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right), \end{aligned} \right. \quad (57)$$

or

$$\left\{ \begin{aligned} &(\alpha \mathbf{I} \otimes \mathbf{U} + \alpha \mathbf{U} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{V} + \mathbf{V} \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) = i(\mathbf{I} \otimes \mathbf{U} + \mathbf{U} \otimes \mathbf{I} - \alpha \mathbf{I} \otimes \mathbf{V} - \alpha \mathbf{V} \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k)} \right) \\ &\quad + (\alpha - i) \text{vec}(\mathbf{C}) - (\alpha - i) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right), \\ &(\mathbf{I} \otimes \mathbf{U} + \mathbf{U} \otimes \mathbf{I} + \alpha \mathbf{I} \otimes \mathbf{V} + \alpha \mathbf{V} \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+1)} \right) = i(\alpha \mathbf{I} \otimes \mathbf{U} + \alpha \mathbf{U} \otimes \mathbf{I} - \mathbf{I} \otimes \mathbf{V} - \mathbf{V} \otimes \mathbf{I}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) \\ &\quad + (1 - i\alpha) \text{vec}(\mathbf{C}) - (1 - i\alpha) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right). \end{aligned} \right. \quad (58)$$

Hence

$$\left\{ \begin{aligned} &(\alpha \mathbf{G} + \mathbf{F}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) = i(\mathbf{G} - \alpha \mathbf{F}) \text{vec} \left(\mathbf{X}^{(k)} \right) + (\alpha - i) \text{vec}(\mathbf{C}) - (\alpha - i) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right), \\ &(\alpha \mathbf{F} + \mathbf{G}) \text{vec} \left(\mathbf{X}^{(k+1)} \right) = i(\alpha \mathbf{G} - \mathbf{F}) \text{vec} \left(\mathbf{X}^{(k+\frac{1}{2})} \right) + (1 - i\alpha) \text{vec}(\mathbf{C}) - (1 - i\alpha) \left(\sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right) \text{vec} \left(\mathbf{X}^{(k)} \right), \end{aligned} \right. \quad (59)$$

where

$$\mathbf{G} = \mathbf{I} \otimes \mathbf{U} + \mathbf{U} \otimes \mathbf{I}, \quad \text{and} \quad \mathbf{F} = \mathbf{I} \otimes \mathbf{V} + \mathbf{V} \otimes \mathbf{I}.$$

After straightforward derivations we can reformulate the method (116) into the standard form

$$\text{vec} \left(\mathbf{X}^{(k+1)} \right) = \Theta(\alpha) \text{vec} \left(\mathbf{X}^{(k)} \right) + \Phi(\alpha) \text{vec}(\mathbf{C}), \quad (60)$$

where

$$\Theta(\alpha) = \Psi(\alpha) - \Phi(\alpha) \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j, \quad (61)$$

and

$$\Psi(\alpha) = (\alpha \mathbf{F} + \mathbf{G})^{-1} (\alpha \mathbf{G} - \mathbf{F}) (\alpha \mathbf{G} + \mathbf{F})^{-1} (\alpha \mathbf{F} - \mathbf{G}), \quad \Phi(\alpha) = 2\alpha (\alpha \mathbf{F} + \mathbf{G})^{-1} (\mathbf{G} - i\mathbf{F}) (\alpha \mathbf{G} + \mathbf{F})^{-1}. \quad (62)$$

Suppose that λ_i are the eigenvalues of $\hat{\mathbf{F}} = \mathbf{G}^{-\frac{1}{2}} \mathbf{F} \mathbf{G}^{-\frac{1}{2}}$. Then the spectral radius of the iteration matrix can be obtained as:

$$\begin{aligned} \rho(\Psi(\alpha)) &= \rho((\alpha \mathbf{F} + \mathbf{G})^{-1} (\alpha \mathbf{G} - \mathbf{F}) (\alpha \mathbf{G} + \mathbf{F})^{-1} (\alpha \mathbf{F} - \mathbf{G}) - \Phi(\alpha) \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j) \\ &\leq \rho((\alpha \mathbf{F} + \mathbf{G})^{-1} (\alpha \mathbf{G} - \mathbf{F}) (\alpha \mathbf{G} + \mathbf{F})^{-1} (\alpha \mathbf{F} - \mathbf{G})) + \rho(\Phi(\alpha) \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j) \\ &\leq \|(\alpha \mathbf{G} - \mathbf{F}) (\alpha \mathbf{G} + \mathbf{F})^{-1} (\alpha \mathbf{F} - \mathbf{G}) (\alpha \mathbf{F} + \mathbf{G})^{-1}\|_2 + 2\alpha \left\| \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right\|_2 \|\mathbf{G} - i\mathbf{F}\|_2 \|(\alpha \mathbf{G} + \mathbf{F})^{-1}\|_2 \|(\alpha \mathbf{F} + \mathbf{G})^{-1}\|_2 \\ &\leq \max_{\lambda \in sp(\hat{\mathbf{F}})} \left| \frac{(\alpha - \lambda)(\alpha\lambda - 1)}{(\alpha + \lambda)(\alpha\lambda + 1)} \right| + 2\alpha \left\| \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right\|_2 \|\mathbf{G}\|_2 \|\mathbf{G}^{-1}\|_2^2 \|\mathbf{I} - i\hat{\mathbf{F}}\|_2 \|(\alpha \mathbf{I} + \hat{\mathbf{F}})^{-1}\|_2 \|(\alpha \hat{\mathbf{F}} + \mathbf{I})^{-1}\|_2, \end{aligned}$$



where $sp(\mathbf{A})$ denotes the spectrum of matrix $\mathbf{A}$. Suppose that

$$f(x) = x + 1/x, \quad x > 0,$$

$$f_{\min} = \min_{\lambda} f(\lambda), \quad f_{\max} = \max_{\lambda} f(\lambda), \quad \lambda_{\min} = \min_{\lambda_i} \{\lambda_i\}, \quad \lambda_{\max} = \max_{\lambda_i} \{\lambda_i\},$$

then it can be shown that

$$\rho(\Psi(\alpha)) \leq \max \left\{ \frac{|f(\alpha) - f_{\min}|}{f(\alpha) + f_{\min}}, \frac{|f(\alpha) - f_{\max}|}{f(\alpha) + f_{\max}} \right\} + \frac{\alpha\theta}{(\alpha + \lambda_{\min})(\alpha\lambda_{\min} + 1)}, \quad (63)$$

where

$$\theta = 2 \left\| \sum_{j=1}^m \mathbf{N}_j \otimes \mathbf{N}_j \right\|_2 \|\mathbf{G}\|_2 \|\mathbf{G}^{-1}\|_2^2 \sqrt{\lambda_{\max}^2 + 1}.$$

Theorem 0.14. Let $\mathbf{A} = \mathbf{U} + i\mathbf{V}$, where $\mathbf{U}$ and $\mathbf{V}$ be real symmetric positive definite matrices. Also let $\alpha > 0$ be constant. Then under condition

$$\theta < \frac{1}{\alpha} \left\{ (\alpha + \lambda_{\min})(\alpha\lambda_{\min} + 1) \left(1 - \max \left\{ \frac{|f(\alpha) - f_{\min}|}{f(\alpha) + f_{\min}}, \frac{|f(\alpha) - f_{\max}|}{f(\alpha) + f_{\max}} \right\} \right) \right\},$$

method (116) converges unconditionally to the unique exact solution $\mathbf{X} \in \mathbb{C}^{n \times n}$ of Eq. (654).

Example 0.15. Consider the equation $\mathbf{A}\mathbf{X} + \mathbf{X}\mathbf{A}^T + \mathbf{N}_1\mathbf{X}\mathbf{N}_1^T = \mathbf{C}$, where

$$\mathbf{U} = \begin{pmatrix} 25 & 2 & -2 & -4 & 6 & & & & & \mathbf{O} \\ 2 & 25 & 2 & -2 & -4 & 6 & & & & \\ -2 & 2 & 25 & 2 & -2 & -4 & 6 & & & \\ -4 & -2 & 2 & 25 & 2 & -2 & -4 & 6 & & \\ 6 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 6 \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -4 \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & -2 \\ & & & & 6 & -4 & -2 & 2 & 25 & 2 \\ \mathbf{O} & & & & & 6 & -4 & -2 & 2 & 25 \end{pmatrix}_{n \times n},$$



$$\mathbf{V} = \begin{pmatrix} 5 & -\frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & & & & & & \mathbf{O} \\ -\frac{1}{2} & 5 & -\frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & & & & & \\ \frac{1}{3} & -\frac{1}{2} & 5 & -\frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & & & & \\ \frac{1}{4} & \frac{1}{3} & -\frac{1}{2} & 5 & -\frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & & & \\ \frac{1}{5} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & \frac{1}{5} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \frac{1}{5} & \\ & & \frac{1}{5} & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \frac{1}{4} & \\ & & & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \frac{1}{3} & \\ & & & & \frac{1}{5} & \frac{1}{4} & \frac{1}{3} & -\frac{1}{2} & 5 & -\frac{1}{2} & \\ \mathbf{O} & & & & & \frac{1}{5} & \frac{1}{4} & \frac{1}{3} & -\frac{1}{2} & 5 & \end{pmatrix}_{n \times n},$$

$$\mathbf{N}_1 = \begin{pmatrix} 2 & -\frac{1}{3} & -\frac{1}{2} & & & & & & & & \mathbf{O} \\ \frac{1}{4} & 2 & -\frac{1}{3} & -\frac{1}{2} & & & & & & & \\ \frac{1}{5} & \frac{1}{4} & 2 & -\frac{1}{3} & -\frac{1}{2} & & & & & & \\ & \frac{1}{5} & \frac{1}{4} & 2 & -\frac{1}{3} & -\frac{1}{2} & & & & & \\ & & \frac{1}{5} & \frac{1}{4} & 2 & -\frac{1}{3} & -\frac{1}{2} & & & & \\ & & & \frac{1}{5} & \ddots & \ddots & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & & & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & & & & \ddots & \ddots & \ddots & \ddots & -\frac{1}{2} \\ & & & & & & & \ddots & \ddots & \ddots & -\frac{1}{3} \\ \mathbf{O} & & & & & & & & \frac{1}{5} & \frac{1}{4} & 2 \end{pmatrix}_{n \times n},$$

and right hand side matrix $\mathbf{C}$ is such that $X^\star = (x_{i,j})$ with

$$x_{i,j} = \exp\left(-((-1 + 2(i-1)/(n-1)^2 + (-1 + 2(j-1)/(n-1)^2))\right), \quad i, j = 1, 2, \dots, n,$$

be exact solution.

Table 3: Numerical results for $\alpha = 5.1$ and $n = 30$.

α	4.2	4.4	4.6	4.8	5	5.2	5.4	5.6	5.8	6
$\rho(\Psi(\alpha))$	0.4914	0.4724	0.4537	0.4354	0.4175	0.4057	0.4196	0.4330	0.4458	0.4582
<i>Iteration</i> (= k)	1	2	3	4	5	6	7	...	14	15
$\log_{10}(\ X^\star - X^{(k)}\ _F / \ X^\star\ _F)$	-1.19	-2.35	-3.33	-4.10	-4.78	-5.41	-6.01	...	-9.20	-9.62
<i>CPU-time</i> (second)	0.0045	0.0047	0.0049	0.0052	0.0053	0.0054	0.0056	...	0.0064	0.0066

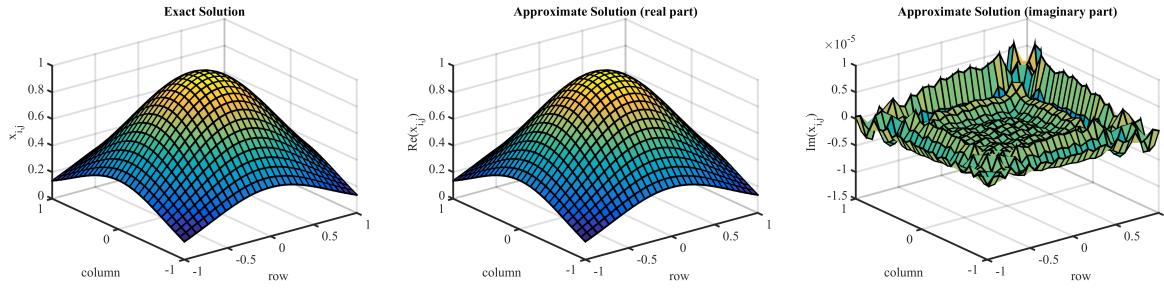


Figure 6: Exact solution and approximate solution after 5 iteration.

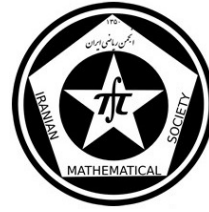
References

- [1] A. Bouhamidi, K. Jbilou, A note on the numerical approximate solutions for generalized Sylvester matrix equations with applications, *Appl. Math. Comput.* 206 (2008) 687–694.
- [2] M. Dehghan, A. Shirilord, A generalized modified Hermitian and skew-Hermitian splitting (GMHSS) method for solving complex Sylvester matrix equation, *Appl. Math. Comput.* 348 (2019) 632–651.
- [3] D.L. Kleinman, On the stability of linear stochastic systems, *IEEE Trans. Automat. Control.* 14 (1969) 429–430.
- [4] D.A. Paolo, I. Alberto, R. Antonio, Realization and structure theory of bilinear dynamical systems, *SIAM J. Control Optim.* 12 (1974) 517–535.
- [5] D.K. Salkuyeh, Two-step scale-splitting method for solving complex symmetric system of linear equations, *arXiv:1705.02468*
- [6] Z. Zheng, F.-L. Huang, Y.-C. Peng, Double-step scale splitting iteration method for a class of complex symmetric linear systems, *Appl. Math. Lett.* 73 (2017) 91–97.



8th Seminar on Numerical Analysis and its Applications

University of Kurdistan, Sanandaj, Iran, April 06-09 2021



An operational matrix method for solving a class of nonlinear Volterra integral equations arising in steady activation of a skeletal muscle

S. Torkaman^{†5}, G. B. Loghmani[†], M. Heydari[†]

[†]Department of Mathematics, Yazd University, Yazd, Iran

s.torkaman@stu.yazd.ac.ir, loghmani@yazd.ac.ir, m.heydari@yazd.ac.ir

Abstract

In this research, the Volterra integral equation for the steady activation of a skeletal muscle is investigated using an effective operational matrix-based method. For this purpose, the operational matrix of integration is derived for the barycentric rational cardinal functions. Then, by applying this operational matrix and without using any collocation points, the governing Volterra integral equation is reduced to a system of nonlinear algebraic equations. The numerical results confirm the accuracy and efficiency of the proposed method.

AMS subject Classification 2010: 65R20, 45G10

Keywords: Volterra integral equation; Barycentric rational interpolants; Operational matrices.

1 Introduction

The steady activation of a skeletal muscle is a phenomenon with complex and subtle physiological processes that can be described by Volterra integral equation [2, 3]. The motor units (MUs) develop a skeletal muscle that each MUs consisting of a motoneuron (MN) and the muscle fibers it innervates. Electrical signals that coming from higher motor centers and propagated to the motoneurons along a network of nerve fibers compose the input of MUs. In this problem, MUs are ordered according to their maximal (tetanic) contraction forces t . Consider $f(t)$ be the muscle force as a function of the last recruited MU for an arbitrary but fixed network. Clearly, $f(t)$ is an increasing function of t . As mentioned in [2], the muscle force $f(t)$ satisfies the following Volterra integral equation:

$$f(t) = \int_a^t k(s, f(s), f(t)) ds. \quad (64)$$

Let ρ be a density function that represents the MU population. Suppose that t_{min} and t_{max} be the tetanic forces of the weakest and strongest MUs and $\int_{t_{min}}^{t_{max}} \rho(s) ds$ be the number of MUs in the pool.

⁵Speaker



The Volterra integral equation for the (64) takes the form [2]:

$$f(t) = \int_{t_{min}}^t h(s) \left(1 - c \exp \left(-\alpha \frac{f(t) - f(s)}{f(s) + \Delta} \right) \right) ds, \quad t \in [t_{min}, t_{max}], \quad (65)$$

where $h(t) = t\rho(t)$, $\alpha > 0$, $\Delta > 0$ and $0 < c < 1$. Because ρ is strictly positive function almost everywhere, the function $H(t) = \int_{t_{min}}^t h(s)ds$, $t \in [t_{min}, t_{max}]$ is strictly increasing and hence invertible. So, by considering the change of variable $s = H^{-1}(x)$, the Volterra integral equation (65) can be rewritten as

$$Y(t) = \int_0^t \left(1 - c \exp \left(-\alpha \frac{Y(t) - Y(x)}{Y(x) + \Delta} \right) \right) dx, \quad t \in [0, T], \quad (66)$$

where $Y(t) = f(H^{-1}(t))$ and $T = H(t_{max})$. Finally, by considering the above notation we obtain $f(t) = Y(H(t))$.

2 Linear barycentric rational interpolation

Let $f : [a, b] \rightarrow \mathbb{R}$, $n \in \mathbb{N}$, $\{t_j\}_{j=0}^n$ be a set of equidistant nodes where $t_j = a + jh$, $j = 0, 1, \dots, n$, $hn = b - a$ and $\{f(t_j)\}_{j=0}^n$ be the corresponding $n + 1$ given data. A linear barycentric rational interpolation for this data can be written as follows

$$r_n(t) = \sum_{j=0}^n f(t_j) \phi_j(t), \quad \phi_j(t) = \frac{\frac{w_j}{t - t_j}}{\sum_{i=0}^n \frac{w_i}{t - t_i}}, \quad j = 0, 1, \dots, n, \quad (67)$$

where $\{w_j\}_{j=0}^n$ are arbitrary nonzero set of weights. The basis functions $\{\phi_j(t)\}_{j=0}^n$ have the following properties:

- **Partition of unity:** $\sum_{j=0}^n \phi_j(t) = 1$.
- **Lagrange property:** $\phi_j(t_i) = \delta_{ij}$, $i, j = 0, 1, \dots, n$ where δ_{ij} is the Kronecker delta.

Let $0 \leq d \leq n$ be an integer parameter. Floater and Hormann [1] proposed a family of rational interpolants based on the barycentric weights w_j as follows:

$$w_j = (-1)^j \sum_{i=\max(j-d, 0)}^{\min(j, n-d)} \left(\prod_{k=i, k \neq j}^{i+d} \frac{1}{|t_j - t_k|} \right), \quad j = 0, 1, \dots, n. \quad (68)$$

By choosing interpolation weights (68), interpolant (67) has no real poles [1]. Due to the Lagrange property of the basic functions (67), any function f on the interval $[a, b]$ can be approximated as:

$$f(t) \simeq \sum_{j=0}^n f(t_j) \phi_j(t). \quad (69)$$

By considering the barycentric weights (68), the approximation error of (69) can be estimated as follows:



Theorem 0.16. [1] Consider $f \in C^{d+2}[a, b]$. Then

$$\| f(t) - \sum_{j=0}^n f(t_j) \phi_j(t) \|_{\infty} \leq \begin{cases} h^{d+1} (1 + \gamma \mu) (b - a) \frac{\| f^{(d+2)} \|_{\infty}}{d+2}, & (n - d) \text{ odd} \\ h^{d+1} (1 + \gamma \mu) \left((b - a) \frac{\| f^{(d+2)} \|_{\infty}}{d+2} + \frac{\| f^{(d+1)} \|_{\infty}}{d+1} \right), & (n - d) \text{ even} \end{cases}$$

where

$$\gamma = \begin{cases} 1, & d = 0 \\ 0, & d > 1 \end{cases}, \quad h = \max_{0 \leq i \leq n-1} (t_{i+1} - t_i), \quad \mu = \max_{1 \leq i \leq n-2} \min \left(\frac{t_{i+1} - t_i}{t_i - t_{i-1}}, \frac{t_{i+1} - t_i}{t_{i+2} - t_{i+1}} \right).$$

3 Operational matrix of integration

Suppose that $\Phi(t)$ is a $(n + 1) \times 1$ vector as follows:

$$\Phi(t) = [\phi_0(t), \phi_1(t), \dots, \phi_n(t)]^T, \quad (70)$$

where $\{\phi_j(t)\}_{j=0}^n$ is defined in (67). By integrating the vector $\Phi(t)$, we have

$$\int_a^t \Phi(s) ds = \left[\int_a^t \phi_0(s) ds, \int_a^t \phi_1(s) ds, \dots, \int_a^t \phi_n(s) ds \right]^T. \quad (71)$$

Using (69), any components of the vector (0.17) can be approximated as:

$$\int_a^t \phi_i(s) ds \simeq \sum_{j=0}^n p_{ij} \phi_j(t), \quad i = 0, 1, \dots, n. \quad (72)$$

Substituting (72) in (0.17), we get

$$\int_a^t \Phi(s) ds \simeq \left[\sum_{j=0}^n p_{0j} \phi_j(t), \sum_{j=0}^n p_{1j} \phi_j(t), \dots, \sum_{j=0}^n p_{nj} \phi_j(t) \right]^T = \mathbf{P} \Phi(t).$$

Hence, the operational matrix of integration can be defined as follow:

Definition 0.17. Let $\Phi(t)$ be the vector stated in (70). The operational matrix of integration $\mathbf{P}$ based on the barycentric basis functions can be represented as

$$\int_a^t \Phi(s) ds \simeq \mathbf{P} \Phi(t),$$

where $\mathbf{P} = (p_{ij})$ and p_{ij} can be computed as:

$$p_{ij} = \int_a^{t_j} \phi_i(s) ds, \quad i, j = 0, 1, \dots, n.$$



Remark 0.18. Let A be a $(n+1) \times (n+1)$ matrix. By using (10), the function $\Phi^T(t)A\Phi(t)$ can be approximated as following

$$\Phi^T(t)A\Phi(t) \simeq \sum_{j=0}^n \Phi^T(t_j)A\Phi(t_j)\phi(t).$$

Due to the Lagrange property of the basic function (67), we get

$$\Phi^T(t)A\Phi(t) \simeq \hat{\mathbf{A}}\Phi(t), \quad \hat{\mathbf{A}} = [A_{00}, A_{11}, \dots, A_{nn}],$$

where $\hat{\mathbf{A}}$ is a row vector composed of the diagonal elements of the matrix A .

4 Description of the numerical method

In this section, a numerical approach based on the operational matrix of integration for solving the nonlinear Volterra integral equations (64) is described. For this purpose, by using (69), we approximate the function $f(t)$ as

$$f(t) \simeq f_n(t) = \sum_{i=0}^n f(t_i)\phi_i(t) = F^T\Phi(t), \quad F = [f(t_0), f(t_1), \dots, f(t_n)]^T, \quad (73)$$

where the vector F is unknown. By using (73), the residual function for (64) can be written as

$$\mathbf{RES}(t) = f_n(t) - \int_a^t k(s, f_n(s), f_n(t))ds. \quad (74)$$

In order to determine the unknown vector F , we proceed as follow:

At first, by applying (73), the function $k(s, f_n(s), f_n(t))$ relative to the variable s is estimated as

$$k(s, f_n(s), f_n(t)) \simeq K(t)^T\Phi(s), \quad (75)$$

where

$$K(t) = \left[k(s_0, f_n(s_0), f_n(t)), k(s_1, f_n(s_1), f_n(t)), \dots, k(s_n, f_n(s_n), f_n(t)) \right]^T.$$

According to the Lagrange property, $K(x)$ can be rewritten as

$$K(t) = \left[k(s_0, f(s_0), f_n(t)), k(s_1, f(s_1), f_n(t)), \dots, k(s_n, f(s_n), f_n(t)) \right]^T.$$

Again using (69), each component of the vector $K(t)$ is estimated as

$$k(s_i, f(s_i), f_n(t)) \simeq K_i^T\Phi(t), \quad i = 0, 1, \dots, n, \quad (76)$$

where

$$K_i = \left[k(s_i, f(s_i), f(t_0)), k(s_i, f(s_i), f(t_1)), \dots, k(s_i, f(s_i), f(t_n)) \right]^T.$$

By substituting (333) in (75), we get

$$k(s, f_n(s), f_n(t)) \simeq \Phi^T(t)K^T\Phi(s), \quad (77)$$



where K is $(n+1) \times (n+1)$ matrix and $K_{ij} = k(s_i, f(s_i), f(t_j))$. By substituting (77) in (74), we have

$$F^T \Phi(t) - \Phi(t)^T K^T \int_a^t \Phi(s) ds = 0.$$

Using the integral operational matrix that defined in Definition 0.17, we obtain

$$F^T \Phi(t) - \Phi(t)^T K^T \mathbf{P} \Phi(t) = 0.$$

Finally, by applying Remark 0.18 and canceling $\Phi(t)$, we get

$$F^T - \hat{\mathbf{A}} = 0, \quad (78)$$

where $\hat{\mathbf{A}}$ is a row vector composed of the diagonal elements of the matrix $K^T \mathbf{P}$. So, by solving the nonlinear system of algebraic equations (78), the unknown vector F can be obtained.

5 Numerical results and conclusion

In this section, the barycentric rational cardinal functions with associated integral operational matrix were applied to approximate the solution of the Volterra integral equations (66). To verify efficiency and high accuracy of the proposed method, the residual function is plotted. Consider $t_{min} = 10$, $t_{max} = 20$, $c = 0.9$, $\alpha = 1.14$ and $\rho(t) = \frac{1}{t}$, $t \in [t_{min}, t_{max}]$. The approximation solution $Y(t)$ and the residual function $\mathbf{RES}(t)$ with $n = 20$ and $d = 18$ are illustrated in Figure 42. Table 0.52 reports the numerical

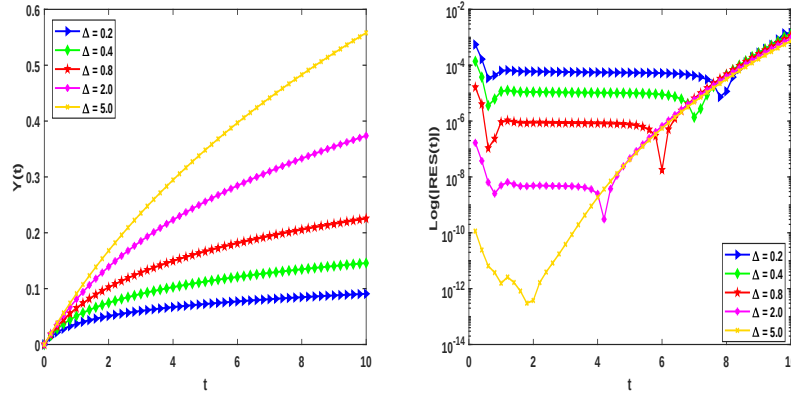


Figure 7: Graphs of $Y(t)$ and $\mathbf{RES}(t)$ with $n = 20$ and $d = 18$.

results for the Volterra integral equation (66) for $\Delta = 2$ and 5. Graphical and tabulated results indicate that the proposed method can be successfully applied for solving a wide class of the nonlinear Volterra integral equations arising in mathematical and engineering problems.

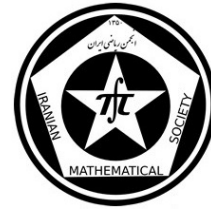


Table 4: Numerical results with $n = 20$ and $d = 19$.

t	$ \mathbf{RES}(t) $	
	$\Delta = 2$	$\Delta = 5$
0.0	0.000000	0.0000000
2.0	$3.90e - 8$	$1.68e - 11$
4.0	$2.70e - 8$	$8.88e - 09$
6.0	$6.71e - 7$	$5.92e - 07$
8.0	$3.00e - 5$	$2.42e - 05$
10	$2.18e - 3$	$1.77e - 03$

References

- [1] M. S. Floater, K. Hormann, Barycentric rational interpolation with no poles and high rates of approximation, *Numerische Mathematik*, 107 (2) (2007) 315-331.
- [2] J. P. Gabriel, L. M. Studer, D. G. Rüegg, M. A. Schnetzer, A mathematical model for the steady activation of a skeletal muscle, *SIAM Journal on Applied Mathematics*, 68 (2008) 869-889.
- [3] L. M. Studer, D. G. Ruegg, J. P. Gabrie, A model for steady isometric muscle activation, *Biological Cybernetics*, 80 (1999) 339-355.



An operation matrix method based on the Boubaker polynomials for solving nonlinear integro-differential equations

Mohsen Riahi⁶

Department of Applied Mathematics, Higher Education Complex of Saravan, Saravan, Iran, m.riahi@saravan.ac.ir

Abstract

In this paper, a new scheme for the formulation of operational matrix form of the integration, differentiation and approximate functions in any arbitrary interval which will be used in the Galerkin method is presented. Our aim is to determine the exact operational matrices for the Boubaker polynomials. This method transforms the nonlinear Volterra-Fredholm integro-differential equations (VFIDEs) into a system of nonlinear algebraic equations which will be solved easily by the Galerkin procedure. To illustrate the efficiency and capability of the method, one example is presented and their results are compared with the results of some other well-known methods.

AMS subject Classification 2010: 45G10, 41A10, 45D05, 45B05

Keywords: Boubaker polynomials, Operational matrices, VFIDEs.

1 Introduction

Many problems in physics and engineering give rise to integral and integro-differential equations. When a physical system is modeled under the differential sense; it appears as a differential equation, an integral equation or an integro-differential equation. In recent years, many methods have been used to estimate the solution of nonlinear Volterra-Fredholm integral equations, such as collocation methods [1, 5], Triangular functions methods [4], rationalized Haar functions method [6] and etc.

This paper considers a specific case of nonlinear VFIDEs of the form

$$\begin{cases} \sum_{n=0}^m p_n(x)u^{(n)}(x) + \lambda_1 \int_a^b k_1(x, t)H_1(t, u(t)) dt + \lambda_2 \int_a^x k_2(x, t)H_2(t, u(t)) dt = f(x), \\ u^{(n)}(a) = \alpha_n, \quad 1 \leq n \leq m, \end{cases} \quad (79)$$

where the functions $H_1(t, u(t))$ and $H_2(t, u(t))$ are given nonlinear continuous functions with respect to $u(t)$, $k_i(x, t) \in L^2[a, b]$, λ_1 and λ_2 are constants, $p_n(x)$ and $f(x)$ are defined on $L^2[a, b]$ and $u(x)$ is an unknown function which will be determined. For solving these equations, a new set of functions which

⁶Speaker



were introduced by Boubaker will be used. Also, we present new matrix forms of operational matrix for integration, differentiation, expansion of functions of one and two variables with respect to Boubaker functions. By using these representations a nonlinear VFIDEs can be easily reduced to a nonlinear system of algebraic equations. Afterwards, this system will be solved by the Galerkin procedure.

2 The Boubaker polynomials and some of their properties

The Boubaker polynomials were established by Boubaker et al. [2] as a guide for solving a one-dimensional formulation of heat transfer equation $f_{xx}(x, t) = kf_t(x, t)$. The well-known Boubaker polynomials are defined on any arbitrary interval and can be determined

$$b_0(x) = 1, \quad b_1(x) = x, \quad b_2(x) = x^2 + 2, \quad b_m(x) = xb_{m-1}(x) - b_{m-2}(x), \quad \text{for } m > 2. \quad (80)$$

3 Matrix representation of the Boubaker polynomials

Using Eq. (609), the Boubaker vector

$$\mathbf{B}(x) = [b_0(x), b_1(x), \dots, b_N(x)]^T, \quad (81)$$

can be written in the form $\mathbf{B}(x) = \mathbf{A}\mathbf{\Lambda}(x)$, where $\mathbf{\Lambda}(x) = [1, x, x^2, \dots, x^N]^T$ and $\mathbf{A}$ is $(N+1) \times (N+1)$ lower triangular matrix which its main diagonal elements are all 1.

4 Function approximation

Suppose that $u(x)$ is an any arbitrary function in $H = L^2[a, b]$, then it can be approximated as

$$u(x) \simeq u_N(x) = \sum_{n=0}^N u_n b_n(x) = \mathbf{U}^T \mathbf{B}(x), \quad (82)$$

where $\mathbf{B}(x)$ is the Boubaker vector defined in (610) and $\mathbf{U}$ is the Boubaker coefficients vector as

$$\mathbf{U} = [u_0, u_1, \dots, u_N]^T. \quad (83)$$

Vector $\mathbf{U}$ can be computed by $\mathbf{U}^T \langle \mathbf{B}(x), \mathbf{B}^T(x) \rangle = \langle u(x), \mathbf{B}^T(x) \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the standard inner product in H . Setting ${}_a\mathbf{Q}_b = \langle \mathbf{B}(x), \mathbf{B}^T(x) \rangle = \int_a^b \mathbf{B}(x) \mathbf{B}^T(x) dx$, one obtains $\mathbf{U}^T = \left(\int_a^b u(x) \mathbf{B}^T(x) dx \right) {}_a\mathbf{Q}_b^{-1}$. The method to calculate ${}_a\mathbf{Q}_b$ will be presented in the next subsection. We approximate the function $k(x, t) \in L^2([a, b] \times [a, b])$ as follows

$$k(x, t) \simeq k_{N,N}(x, t) = \sum_{n=0}^N \sum_{m=0}^N k_{nm} b_n(x) b_m(t) = \mathbf{B}^T(x) \mathbf{K} \mathbf{B}(t), \quad (84)$$

where $\mathbf{K}$ is a $(N+1) \times (N+1)$ matrix which can be determined by the following relation

$$\mathbf{K} = {}_a\mathbf{Q}_b^{-1} \langle \mathbf{B}(x), \langle k(x, t), \mathbf{B}^T(t) \rangle \rangle {}_a\mathbf{Q}_b^{-1}. \quad (85)$$



5 Main operational matrices

The derivative of the vector $\mathbf{B}(x)$ in Eq. (610) is given by $\frac{d}{dx}\mathbf{B}(x) = \mathbf{D}\mathbf{B}(x)$, where $\mathbf{D} = [d_{ij}]$ is the $(N+1) \times (N+1)$ operational lower triangular matrix for derivative and is given by the following recursive relations

$$\begin{cases} d_{i,i-1} = i, & \text{for } i = 1, 2, \dots, N, \\ d_{i,0} = -3d_{i-2,0} - 2, & \text{for } i = 3, 5, \dots, k, \quad k = \begin{cases} N, & N \text{ is odd,} \\ N-1, & N \text{ is even,} \end{cases} \\ d_{2n+i,i+1} = d_{2n+i-1,i} + 1, & \text{for } i = 0, 1, \dots, N-2n, \quad n = 2, 3, \dots, \left\lfloor \frac{N}{2} \right\rfloor, \\ 0, & \text{Otherwise.} \end{cases}$$

Also, the n -th order derivative of this expression is $\frac{d^n}{dx^n}\mathbf{B}(x) = \mathbf{D}^n\mathbf{B}(x)$. Therefore,

$$u^{(n)}(x) \simeq \mathbf{U}^T \frac{d^n}{dx^n} \mathbf{B}(x) = \mathbf{U}^T \mathbf{D}^n \mathbf{B}(x). \quad (86)$$

The matrix ${}_a\mathbf{Q}_b$ defined in subsection is named the dual matrix of $\mathbf{B}(x)$ and can be obtained as

$${}_a\mathbf{Q}_b = \langle \mathbf{B}(x), \mathbf{B}^T(x) \rangle = \int_a^b \mathbf{B}(x) \mathbf{B}^T(x) dx = \mathbf{A} \left(\int_a^b \mathbf{\Lambda}(x) \mathbf{\Lambda}^T(x) dx \right) \mathbf{A}^T = \mathbf{A} \mathbf{\Gamma}_b \mathbf{A}^T, \quad (87)$$

where $\mathbf{\Gamma}_b = \int_a^b \mathbf{\Lambda}(x) \mathbf{\Lambda}^T(x) dx = [\gamma_{ij}]$ is a $(N+1) \times (N+1)$ symmetric matrix as follows

$$\gamma_{ij} = \frac{1}{i+j+1} (b^{i+j+1} - a^{i+j+1}), \quad i, j = 0, 1, \dots, N.$$

6 Numerical solution of VFIDE by the Boubaker polynomials

In this section, we consider the nonlinear VFIDE of the form (689). Assume that

$$H_1(t, u(t)) = y(t), \quad H_2(t, u(t)) = w(t), \quad a \leq t \leq b. \quad (88)$$

Using Eq. (611), the functions $u(t)$, $y(t)$ and $w(t)$ are approximated as follows

$$u(t) = \mathbf{U}^T \mathbf{B}(t), \quad y(t) = \mathbf{Y}^T \mathbf{B}(t), \quad w(t) = \mathbf{W}^T \mathbf{B}(t), \quad (89)$$

where $\mathbf{Y}$ and $\mathbf{W}$ are $(N+1) \times 1$ unknown column vectors similar to $\mathbf{U}$ in (612). Hence, by using relations (611) and (603), one can approximate the functions $p_n(x)$, $n = 0, 1, 2, \dots, m$, $f(x)$, $k_1(x, t)$ and $k_2(x, t)$ as

$$p_n(x) \simeq \mathbf{P}_n^T \mathbf{B}(x), \quad f(x) = \mathbf{F}^T \mathbf{B}(x), \quad k_1(x, t) = \mathbf{B}^T(x) \mathbf{K}_1 \mathbf{B}(t), \quad k_2(x, t) = \mathbf{B}^T(x) \mathbf{K}_2 \mathbf{B}(t), \quad (90)$$



where $\mathbf{P}_n^T$ and $\mathbf{F}_n^T$ are $(N+1) \times 1$ known column vectors similar to $\mathbf{U}$ in (612), $\mathbf{K}_1$ and $\mathbf{K}_2$ are $(N+1) \times (N+1)$ known matrices similar to $\mathbf{K}$ in (604). Using (408)-(543) one obtain

$$\begin{aligned} \int_a^b k_1(x, t) H_1(t, u(t)) dt &= \int_a^b \mathbf{B}^T(x) \mathbf{K}_1 \mathbf{B}(t) \mathbf{B}^T(t) \mathbf{Y} dt = \mathbf{B}^T(x) \mathbf{K}_1 \left(\int_a^b \mathbf{B}(t) \mathbf{B}^T(t) dt \right) \mathbf{Y} \\ &= \mathbf{B}^T(x) \mathbf{K}_1 ({}_a\mathbf{Q}_b) \mathbf{Y} = \mathbf{B}^T(x) \mathbf{K}_1 \mathbf{A} \mathbf{\Gamma}_b \mathbf{A}^T \mathbf{Y}, \end{aligned} \quad (91)$$

$$\begin{aligned} \int_a^x k_2(x, t) H_2(t, u(t)) dt &= \int_a^x \mathbf{B}^T(x) \mathbf{K}_2 \mathbf{B}(t) \mathbf{B}^T(t) \mathbf{Y} dt = \mathbf{B}^T(x) \mathbf{K}_2 \left(\int_a^x \mathbf{B}(t) \mathbf{B}^T(t) dt \right) \mathbf{Y} \\ &= \mathbf{B}^T(x) \mathbf{K}_2 ({}_a\mathbf{Q}_x) \mathbf{Y} = \mathbf{B}^T(x) \mathbf{K}_2 \mathbf{A} \mathbf{\Gamma}_x \mathbf{A}^T \mathbf{Y} \end{aligned} \quad (92)$$

Substituting (538), (543)-(616) in (689), results in

$$\sum_{n=0}^m \mathbf{P}_n^T \mathbf{B}(x) \mathbf{U}^T \mathbf{D}^n \mathbf{B}(x) + \lambda_1 \mathbf{B}^T(x) \mathbf{K}_1 \mathbf{A} \mathbf{\Gamma}_b \mathbf{A}^T \mathbf{Y} + \lambda_2 \mathbf{B}^T(x) \mathbf{K}_2 \mathbf{A} \mathbf{\Gamma}_x \mathbf{A}^T \mathbf{W} = \mathbf{F}^T \mathbf{B}(x). \quad (93)$$

This equation is a system of $N+1$ algebraic equations with $3(N+1)$ unknowns for $\mathbf{U}$, $\mathbf{Y}$ and $\mathbf{W}$ vectors given by (540). To find the solution $u(x)$, we utilize the equations $H_1(t, \mathbf{U}^T \mathbf{B}(t)) = \mathbf{Y}^T \mathbf{B}(t)$, $H_2(t, \mathbf{U}^T \mathbf{B}(t)) = \mathbf{W}^T \mathbf{B}(t)$ to construct $2(N+1)$ algebraic equations. This can be written as

$$\langle H_1(t, \mathbf{U}^T \mathbf{B}(t)), b_i(t) \rangle = \langle \mathbf{Y}^T \mathbf{B}(t), b_i(t) \rangle, \quad \langle H_2(t, \mathbf{U}^T \mathbf{B}(t)), b_i(t) \rangle = \langle \mathbf{W}^T \mathbf{B}(t), b_i(t) \rangle, \quad (94)$$

for $i = 0, 1, 2, \dots, N$. By applying the Galerkin method in equation (546) (similar to relation (412)), and combining the result with Eq. (412), we obtain a total of $3(N+1)$ algebraic equations with the same number of unknowns for $\mathbf{U}$, $\mathbf{Y}$ and $\mathbf{W}$.

Initial conditions $y^{(n)}(a) = \alpha_n$, consists only m ($m \leq N$) algebraic equation for $N+1$ unknowns of $\mathbf{U}$, due to this restriction the matrix form of these conditions are as follows

$$\mathbf{U}^T \mathbf{D}^n \mathbf{B}(a) = \alpha_n. \quad (95)$$

Consequently, by replacing the rows of (618), with the rows of the algebraic system and solving the results for the unknown coefficients vectors in $\mathbf{U}$, $\mathbf{Y}$ and $\mathbf{W}$, one can obtain the solution as follows

$$u(x) = \mathbf{U}^T \mathbf{B}(x).$$

7 Numerical example

Example 0.19. [5] Suppose that $f(x) = \left(-\frac{2}{3}x + \frac{1}{9}\right) e^{3x} + (2x+1)e^x + \left(\frac{4}{3} - e\right)x + \frac{8}{9}$ and consider the following nonlinear VFIDE

$$u'(x) + 2xu(x) - \int_0^1 (x-t)u(t)dt - \int_0^x (x+t)u^3(t)dt = f(x), \quad (96)$$

with the initial condition $u(0) = 0$. This problem has the exact solution $u(x) = e^x$.

Following the procedure in Section 36, we obtain its approximate solutions for various values of N . In Table 54 the numerical results for this example with various values of N are displayed together with the



Table 5: Comparison MAE between the methods in [5] and presented method for Eq. (96).

N	By ∞ norm [5]	By theorem 5.2 in [5]	Present method
3	$7.8000e - 003$	$2.1200e - 002$	$8.2673e - 004$
4	$4.3534e - 004$	$2.3000e - 003$	$3.9792e - 005$
5	$3.6607e - 005$	$1.6574e - 004$	$1.0760e - 006$
6	$1.5551e - 006$	$1.0404e - 005$	$4.0701e - 008$
7	$9.1675e - 008$	$5.1426e - 007$	$1.3143e - 009$
8	$3.1820e - 009$	$2.3157e - 008$	$3.0021e - 011$
9	$1.4238e - 010$	$8.9254e - 010$	$7.8349e - 013$
10	$4.1851e - 012$	$3.1997e - 011$	$1.8533e - 014$
11	$1.4611e - 013$	$1.0175e - 012$	$3.4225e - 016$
12	$1.0658e - 014$	$3.0437e - 014$	$6.7863e - 018$

Table 6: The numerical results of Equation (96) with different methods for $N = 7$.

x_i	Exact solution	Taylor Method [3]		Bernstein series Method [5]		Present method	
	$u_7(x_i)$	$u_7(x_i)$	Absolute error	$u_7(x_i)$	Absolute error	$u_7(x_i)$	Absolute error
0	1	1	0	1	0	1	0
0.2	1.22140275816	1.2214031	$3.4184e - 007$	1.22140275634	$1.8156e009$	1.221402757477	$6.8285e - 010$
0.4	1.49182469764	1.4918253	$6.0236e - 007$	1.49182469631	$1.3355e009$	1.491824697790	$1.4933e - 010$
0.6	1.82211880039	1.8221197	$8.9961e - 007$	1.82211879934	$1.0530e009$	1.822118800494	$1.0310e - 010$
0.8	2.22554092849	2.2255420	$1.0715e - 006$	2.22554092747	$1.0261e009$	2.225540927818	$6.7415e - 010$
1	2.71828182846	2.7182814	$4.2846e - 007$	2.71828173678	$9.1675e008$	2.718281825164	$3.2947e - 009$

results obtained in [5] by the Bernstein series method. In addition, we compare the exact solution with the results of the Taylor method [3], the Bernstein series method [5] and the present method for $N = 7$ in Table 39. Fig. 42 display the comparisons of the actual absolute error function for different values of N . Considering these Tables and graph, one can see that the error will be decreased when N is increased.

8 Conclusion

In this paper, a numerical method based on the Boubaker polynomials for solving Volterra-Fredholm integro-differential equations was proposed. This efficient scheme reduced Volterra-Fredholm integro-differential equations to a nonlinear system of algebraic equations by using the Galerkin method with an arbitrary Boubaker polynomials. Furthermore, the exact operational matrices (derivative, integral, approximate function and product) for the Boubaker polynomials were constructed. Moreover, Numerical example showed that the proposed method was reliable and more efficient than the other well-known methods.

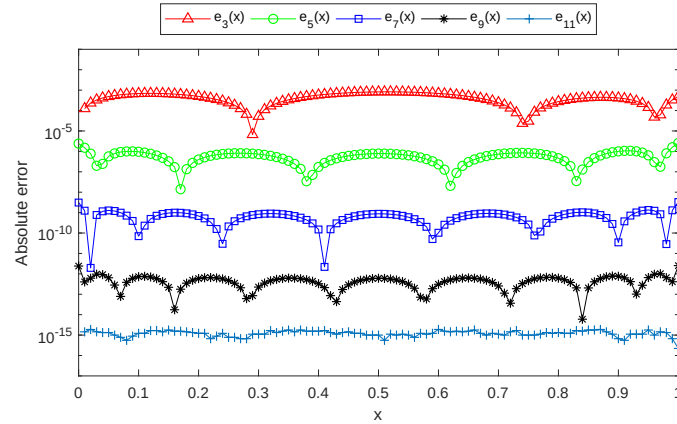
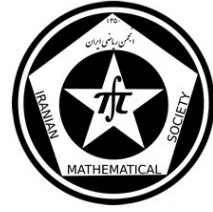


Figure 8: Comparison of the actual absolute error functions for $u(x)$ in problem (96).

References

- [1] F. Mirzaee and S. F. Hoseini, *Application of fibonacci collocation method for solving volterra fredholm integral equations*, Applied Mathematics and Computation, 273 (2016), 637– 644.
- [2] K. Boubaker, *On modified boubaker polynomials: some differential and analytical properties of the new polynomials issued from an attempt for solving bi-varied heat equation*, Trends in Applied Sciences Research, 2 (2007), 540–544.
- [3] K. Maleknejad and Y. Mahmoudi, *Taylor polynomial solution of high-order nonlinear volterra fredholm integro-differential equations*, Applied Mathematics and Computation, 145 (2003), 641–653.
- [4] K. Maleknejad, H. Almasieh and M. Roodaki, *Triangular functions (tf) method for the solution of nonlinear volterra fredholm integral equations*, Communications in Nonlinear Science and Numerical Simulation, 15 (2010), 3293–3298.
- [5] Ş. Yüzbaşı, *A collocation method based on bernstein polynomials to solve nonlinear fredholm volterra integro-differential equations*, Applied Mathematics and Computation, 273 (2016), 142–154.
- [6] Y. Ordokhani and M. Razzaghi, *Solution of nonlinear volterra fredholm hammerstein integral equations via a collocation method and rationalized haar functions*, Applied Mathematics Letters, 21 (2008), 4–9.



A numerical method for one-dimensional Schrodinger equation using comapct local integrated radial basis function method

Ali Ebrahimijahan^{†7}, Mehdi Dehghan[‡]

Amirkabir University of Technology, No. 424, Hafez Ave.,15914, Tehran, Iran and ebrahimijahan.ali@aut.ac.ir[†]

Amirkabir University of Technology, No. 424, Hafez Ave.,15914, Tehran, Iran and mdehghan@aut.ac.ir[‡]

Abstract

In this paper, we propose a numerical scheme to solve the one-dimensional (1D) time-dependent Schrodinger equation using compact local integrated radial basis function method. For time discretion, we use Crank-Nicolson scheme. The results of numerical experiment are presented and are compared with analytical solution to confirm the good accuracy of the presented scheme.

AMS subject Classification 2010: 35-XX , 65Mxx, 65Nxx

Keywords: Schrodinger equation, Compact local integrated radial basis function, Meshhless Method.

1 Introduction

Solving the time dependent Schrodinger equation with arbitrary potential function is of fundamental importance in quantum dynamics calculations [4, 3] and has received considerable attention because of its usefulness as a model that describes several important physical and chemical phenomena.

We will consider the one-dimensional Schrodinger equation with $w = w(x)$ potential

$$-i\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + w(x)u, \quad a \leq x \leq b, \quad 0 < t \leq T, \quad (97)$$

with initial and boundary conditions

$$u(x, 0) = g(x), \quad a \leq x \leq b, \quad u(a, t) = f_0(t), \quad u(b, t) = f_1(t), \quad 0 < t \leq T. \quad (98)$$

2 A brief review of RBF and IRBF methods

The first research on the integrated radial basis function (IRBF) method was done by Mai Duy that can be referred to [5]. Tien and et al.[7] proposed a three-point coupled compact local integrated radial basis function (CLIRBF) scheme. In [6], Mai-Duy and et al. presented a compact 5-point stencil for the discretization of second-order PDEs in two-dimensional spaces.

⁷Speaker



3 Compact local IRBF method

Suppose that the interval $[a, b]$ is divided as follows

$$[a, b] = [x_1, x_2, \dots, x_{n-2}, x_{n-1}, x_n],$$

where $x_1 = a$ and $x_n = b$. Consider a three-point stencil $[x_1, x_2, x_3]$. Suppose that we have the second-order derivative approximation as follows.

$$\frac{d^2 u(x)}{dx^2} = \sum_{j=1}^3 \lambda_j \Psi_j(x), \quad (99)$$

in which $\{\Psi_j\}_{j=1}^3$ is the set of RBFs and $\{\lambda_j\}_{j=1}^3$ is the set of coefficients to be determined. The use of integration leads to obtain the following approximations

$$\frac{du(x)}{dx} = \sum_{j=1}^3 \lambda_j \Phi_j(x) + w_1, \quad (100)$$

$$u(x) = \sum_{j=1}^3 \lambda_j \bar{\Phi}_j(x) + w_1 x + w_2, \quad (101)$$

where $\Phi_j(x) = \int \Psi_j(x) dx$ and $\bar{\Phi}_j(x) = \int \Phi_j(x) dx$ and w_1, w_2 are the integration constants that will be obtained. At interior nodes, for the compact approximation of the function, further information is collected by second-order derivative from two boundary points at per stencil. Collocating grid point in the stencil of $[x_1, x_2, x_3]$, yields

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \frac{d^2 u_1}{dx^2} \\ \frac{d^2 u_3}{dx^2} \end{pmatrix} = \begin{bmatrix} \bar{\Phi}_1(x_1) & \bar{\Phi}_2(x_1) & \bar{\Phi}_3(x_1) & x_1 & 1 \\ \bar{\Phi}_1(x_2) & \bar{\Phi}_2(x_2) & \bar{\Phi}_3(x_2) & x_2 & 1 \\ \bar{\Phi}_1(x_3) & \bar{\Phi}_2(x_3) & \bar{\Phi}_3(x_3) & x_3 & 1 \\ \Psi_1(x_1) & \Psi_2(x_1) & \Psi_3(x_1) & 0 & 0 \\ \Psi_1(x_3) & \Psi_2(x_3) & \Psi_3(x_3) & 0 & 0 \end{bmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ w_1 \\ w_2 \end{pmatrix} = \mathcal{C}_{inner} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ w_1 \\ w_2 \end{pmatrix}. \quad (102)$$

The second-order derivative at the center point i.e. x_2 is computed by

$$\frac{d^2 u_2}{dx^2} = [\Psi_1(x_2), \Psi_2(x_2), \Psi_3(x_2), 0, 0] \mathcal{C}_{inner}^{-1} \left(u_1, u_2, u_3, \frac{d^2 u_1}{dx^2}, \frac{d^2 u_3}{dx^2} \right)^T. \quad (103)$$

The first-order derivative at the center point i.e. x_2 is specified by

$$\frac{du_2}{dx} = [\Phi_1(x_2), \Phi_2(x_2), \Phi_3(x_2), 0, 0] \mathcal{C}_{inner}^{-1} \left(u_1, u_2, u_3, \frac{d^2 u_1}{dx^2}, \frac{d^2 u_3}{dx^2} \right)^T. \quad (104)$$

For boundary points i.e. x_1 and x_n , we consider four point stencils $[x_1, x_2, x_3, x_4]$ and $[x_{n-3}, x_{n-2}, x_{n-1}, x_n]$, respectively. So for nodes x_1 and x_n , the approximations can be written by the following subsequent relations

$$\frac{d^2 u(x)}{dx^2} = \sum_{j=1}^4 \lambda_j \Psi_j(x), \quad \frac{du(x)}{dx} = \sum_{j=1}^4 \lambda_j \Phi_j(x) + w_1, \quad u(x) = \sum_{j=1}^4 \lambda_j \bar{\Phi}_j(x) + w_1 x + w_2.$$



Collocating nodes $[x_1, x_2, x_3, x_4]$ into $u(x)$, yields

$$\begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \frac{d^2 u_2}{dx^2} \\ \frac{d^2 u_3}{dx^2} \end{pmatrix} = \begin{bmatrix} \bar{\Phi}_1(x_1) & \bar{\Phi}_2(x_1) & \bar{\Phi}_3(x_1) & \bar{\Phi}_4(x_1) & x_1 & 1 \\ \bar{\Phi}_1(x_2) & \bar{\Phi}_2(x_2) & \bar{\Phi}_3(x_2) & \bar{\Phi}_4(x_2) & x_2 & 1 \\ \bar{\Phi}_1(x_3) & \bar{\Phi}_2(x_3) & \bar{\Phi}_3(x_3) & \bar{\Phi}_4(x_3) & x_3 & 1 \\ \bar{\Phi}_1(x_4) & \bar{\Phi}_2(x_4) & \bar{\Phi}_3(x_4) & \bar{\Phi}_4(x_4) & x_3 & 1 \\ \Psi_1(x_2) & \Psi_2(x_2) & \Psi_3(x_2) & \Psi_4(x_2) & 0 & 0 \\ \Psi_1(x_3) & \Psi_2(x_3) & \Psi_3(x_3) & \Psi_4(x_3) & 0 & 0 \end{bmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ w_1 \\ w_2 \end{pmatrix} = \mathcal{C}_{border} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ w_1 \\ w_2 \end{pmatrix}. \quad (105)$$

The second-order derivative at the border point i.e. x_1 gives

$$\frac{d^2 u_1}{dx^2} = [\Psi_1(x_1), \Psi_2(x_1), \Psi_3(x_1), \Psi_4(x_1), 0, 0] \mathcal{C}_{border}^{-1} \left(u_1, u_2, u_3, u_4, \frac{d^2 u_2}{dx^2}, \frac{d^2 u_3}{dx^2} \right)^T. \quad (106)$$

The first-order derivative at the boundary point i.e. x_1 can be shown by

$$\frac{du_1}{dx} = [\Phi_1(x_1), \Phi_2(x_1), \Phi_3(x_1), \Phi_4(x_1), 0, 0] \mathcal{C}_{border}^{-1} \left(u_1, u_2, u_3, u_4, \frac{d^2 u_2}{dx^2}, \frac{d^2 u_3}{dx^2} \right)^T. \quad (107)$$

Same as before, for x_n by collocating nodes $[x_{n-3}, x_{n-2}, x_{n-1}, x_n]$ into (105), we will have

$$\begin{pmatrix} u_{n-3} \\ u_{n-2} \\ u_{n-1} \\ u_n \\ \frac{d^2 u_{n-2}}{dx^2} \\ \frac{d^2 u_{n-1}}{dx^2} \end{pmatrix} = \underbrace{\begin{bmatrix} \bar{\Phi}_{n-3}(x_{n-3}) & \bar{\Phi}_{n-2}(x_{n-3}) & \bar{\Phi}_{n-1}(x_{n-3}) & \bar{\Phi}_n(x_{n-3}) & x_{n-3} & 1 \\ \bar{\Phi}_{n-3}(x_{n-2}) & \bar{\Phi}_{n-2}(x_{n-2}) & \bar{\Phi}_{n-1}(x_{n-2}) & \bar{\Phi}_n(x_{n-2}) & x_{n-2} & 1 \\ \bar{\Phi}_{n-3}(x_{n-1}) & \bar{\Phi}_{n-2}(x_{n-1}) & \bar{\Phi}_{n-1}(x_{n-1}) & \bar{\Phi}_n(x_{n-1}) & x_n & 1 \\ \bar{\Phi}_{n-3}(x_n) & \bar{\Phi}_{n-2}(x_n) & \bar{\Phi}_{n-1}(x_n) & \bar{\Phi}_n(x_n) & x_n & 1 \\ \Psi_{n-3}(x_{n-2}) & \Psi_{n-2}(x_{n-2}) & \Psi_{n-1}(x_{n-2}) & \Psi_n(x_{n-2}) & 0 & 0 \\ \Psi_{n-1}(x_{n-1}) & \Psi_{n-2}(x_{n-1}) & \Psi_{n-1}(x_{n-1}) & \Psi_n(x_{n-1}) & 0 & 0 \end{bmatrix}}_{\mathcal{C}_{border}} \begin{pmatrix} \lambda_{n-3} \\ \lambda_{n-2} \\ \lambda_{n-1} \\ \lambda_n \\ w_1 \\ w_2 \end{pmatrix}.$$

The second-order derivative at the boundary point i.e. x_n is computed by

$$\frac{d^2 u_n}{dx^2} = [\Psi_{n-3}(x_n), \Psi_{n-2}(x_n), \Psi_{n-1}(x_n), \Psi_n(x_n), 0, 0] \mathcal{C}_{border}^{-1} \left(u_{n-3}, u_{n-2}, u_{n-1}, u_n, \frac{d^2 u_{n-2}}{dx^2}, \frac{d^2 u_{n-1}}{dx^2} \right)^T.$$

The first-order derivative at the boundary point i.e. x_n is calculated using

$$\frac{du_n}{dx} = [\Phi_{n-3}(x_n), \Phi_{n-2}(x_n), \Phi_{n-1}(x_n), \Psi_n(x_n), 0, 0] \mathcal{C}_{border}^{-1} \left(u_{n-3}, u_{n-2}, u_{n-1}, u_n, \frac{d^2 u_{n-2}}{dx^2}, \frac{d^2 u_{n-1}}{dx^2} \right)^T.$$

The assembled matrices are attained by collocating all grid nodes. The final second-order derivative vector of u is as follows

$$\left[\frac{d^2 u_1}{dx^2}, \frac{d^2 u_2}{dx^2}, \frac{d^2 u_3}{dx^2}, \dots, \frac{d^2 u_{n-1}}{dx^2}, \frac{d^2 u_n}{dx^2} \right]^T = \mathcal{D}_{xx} [u_1, u_2, u_3, \dots, u_n]^T, \quad (108)$$

The assembled first-order derivative is demonstrated as

$$\left[\frac{du_1}{dx}, \frac{du_2}{dx}, \frac{du_3}{dx}, \dots, \frac{d^2 u_{n-1}}{dx}, \frac{d^2 u_n}{dx} \right]^T = \mathcal{D}_x [u_1, u_2, u_3, \dots, u_n]^T. \quad (109)$$



3 Implementation CLIRBF method for governing equation

First, let us discretize (97) according to the following Crank-Nicolson scheme:

$$-i \frac{u(x, t + \tau) - u(x, t)}{\tau} = \frac{1}{2} [u_{xx}(x, t + \tau) + w(x)u(x, t + \tau)] + \frac{1}{2} [u_{xx}(x, t) + w(x)u(x, t)], \quad (110)$$

where τ is the time step size. Rearranging (110), using the notation $u^n = u(x, t^n)$ in which $t^{n+1} = t^n + \tau$, we achieve

$$-iu^{n+1} - \frac{1}{2}\tau [u_{xx}^{n+1} + w(x)u^{n+1}] = -iu^n + \frac{1}{2}\tau [u_{xx}^n + w(x)u^n]. \quad (111)$$

According by CLIRBF that was explained in Sec. , we rewrite (111) as follows

$$-iU^{n+1} - \frac{1}{2}\tau [\mathcal{D}_{xx}U^{n+1} + w(x)U^{n+1}] = -iU^n + \frac{1}{2}\tau [\mathcal{D}_{xx}U^n + w(x)U^n]. \quad (112)$$

Assume that $U^n = U_r^n + iU_i^n$ and r and i denote the real part and imaginary part of complex vectors, respectively. Using this notation, (112) can be written as

$$\begin{aligned} -i(U_r^{n+1} + iU_i^{n+1}) - \frac{1}{2}\tau [\mathcal{D}_{xx}(U_r^{n+1} + iU_i^{n+1}) + w(x)(U_r^{n+1} + iU_i^{n+1})] = \\ -i(U_r^n + iU_i^n) + \frac{1}{2}\tau [\mathcal{D}_{xx}(U_r^n + iU_i^n) + w(x)(U_r^n + iU_i^n)], \end{aligned}$$

or

$$\begin{aligned} -\frac{1}{2}\tau \mathcal{D}_{xx}U_r^{n+1} - \frac{1}{2}\tau w(x)U_r^{n+1} + U_i^{n+1} + i \left(-U_r^{n+1} - \frac{1}{2}\tau \mathcal{D}_{xx}U_i^{n+1} - \frac{1}{2}\tau w(x)U_i^{n+1} \right) = \\ \frac{1}{2}\tau \mathcal{D}_{xx}U_r^n + \frac{1}{2}\tau w(x)U_r^n + U_i^n + i \left(-U_r^n + \frac{1}{2}\tau \mathcal{D}_{xx}U_i^n + \frac{1}{2}\tau w(x)U_i^n \right) \end{aligned} \quad (113)$$

The complex equation (113) can be [1] rewritten by

$$\begin{bmatrix} -0.5\tau(\mathcal{D}_{xx} + wI) & I \\ -I & -0.5\tau(\mathcal{D}_{xx} + wI) \end{bmatrix} \begin{bmatrix} U_r \\ U_i \end{bmatrix}^{n+1} = \begin{bmatrix} 0.5\tau(\mathcal{D}_{xx} + wI) & I \\ -I & 0.5\tau(\mathcal{D}_{xx} + wI) \end{bmatrix} \begin{bmatrix} U_r \\ U_i \end{bmatrix}^n,$$

where I is identity matrix.

4 Numerical results

Example 0.20. We investigate Eq. (97) in the region $a \leq x \leq b$ with potential function

$$w(x) = 3 - 2 \tanh^2(x).$$

The exact solution of this equation is [2]

$$u(x, t) = \frac{i \exp(it)}{\cosh(x)}.$$

The initial and boundary conditions can be found by the above analytical solution.

Table 54 depicts $L_2 = \|\cdot\|_2, L_\infty, RMS$ and $CPU - time$ with $\tau = 0.01$ for varied values of h . Table 39 shows L_∞, RMS and $CPU - time$ for varied regions and various values of τ, h and final time T for Example 0.20. Fig. 42 demonstrates real and imaginary part of approximation solution by the proposed method.

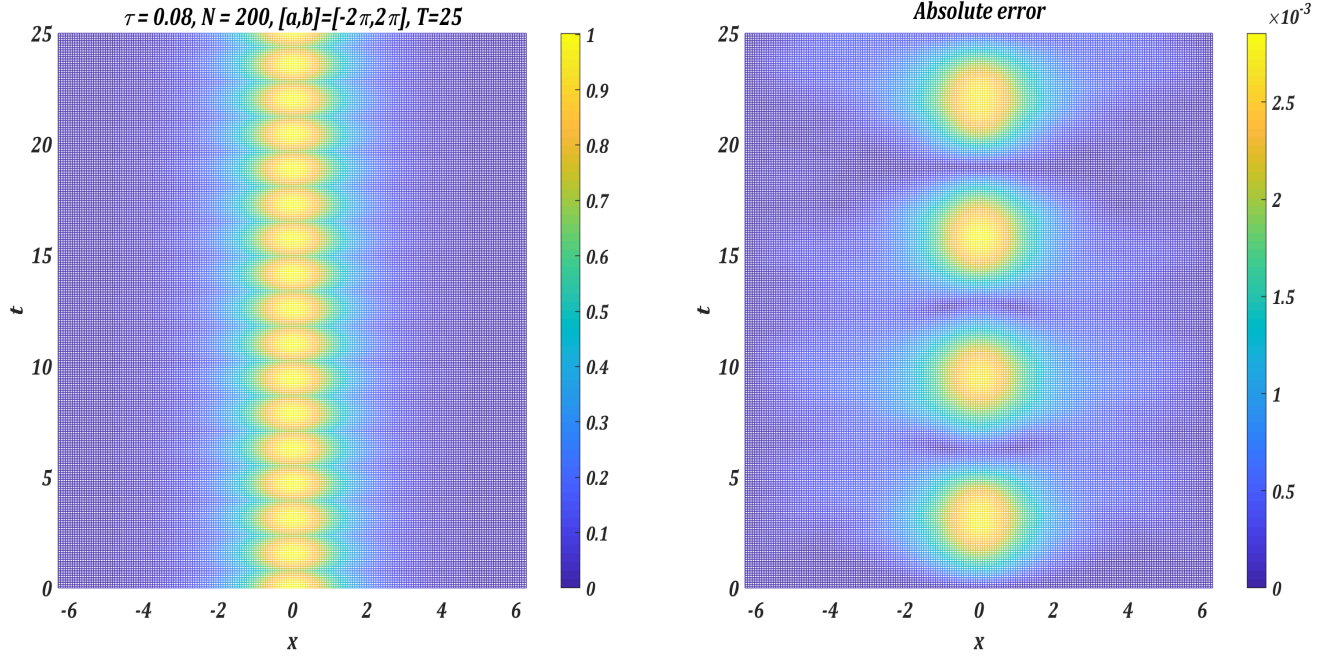


Figure 9: Graph approximation solution and absolute error obtained by CLIRBF method for Example 0.20.

Table 7: Error obtained by present method on $[-1, 1]$ for various values of h with $\tau = \frac{1}{100}$ at final time $T = 1$ for Example 0.20.

h	L_2	L_∞	RMS	$CPU - time$
$\frac{1}{5}$	3.2012×10^{-3}	1.9494×10^{-3}	6.4025×10^{-4}	0.01
$\frac{1}{10}$	5.6295×10^{-3}	2.6667×10^{-3}	5.6295×10^{-4}	0.07
$\frac{1}{20}$	5.8301×10^{-4}	1.9904×10^{-4}	2.9150×10^{-5}	0.08
$\frac{1}{40}$	1.5545×10^{-4}	3.5851×10^{-5}	3.8862×10^{-6}	0.11
$\frac{1}{80}$	1.5452×10^{-4}	2.4560×10^{-5}	1.9316×10^{-6}	0.15
$\frac{1}{160}$	2.1737×10^{-4}	2.4328×10^{-5}	1.3585×10^{-6}	0.28
$\frac{1}{320}$	3.1557×10^{-4}	2.4007×10^{-5}	9.8616×10^{-7}	1.40

References

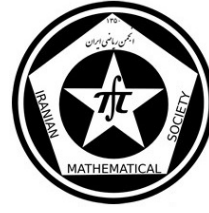
- [1] M. Dehghan, *Finite difference procedures for solving a problem arising in modeling and design of certain optoelectronic devices*, Math. Comput. Simul. 71 (2006), 16–30.



Table 8: Error obtained by present method for varied interval and varous values of h, τ and final time T for Example 0.20.

$[a, b]$	T	h	τ	L_∞	RMS	$CPU - time$
$[0, 1]$	32	$\frac{1}{200}$	$\frac{32}{100}$	5.5033×10^{-3}	2.7807×10^{-4}	0.53
$[0, 4]$	16	$\frac{4}{200}$	$\frac{16}{100}$	1.7486×10^{-2}	8.8789×10^{-4}	0.54
$[0, 8]$	8	$\frac{8}{200}$	$\frac{8}{100}$	2.9938×10^{-4}	1.4832×10^{-4}	0.60
$[0, 16]$	4	$\frac{16}{200}$	$\frac{4}{100}$	2.6743×10^{-4}	9.2514×10^{-6}	0.57
$[0, 32]$	2	$\frac{32}{200}$	$\frac{2}{100}$	2.6421×10^{-3}	5.0049×10^{-5}	0.58
$[0, 64]$	1	$\frac{64}{200}$	$\frac{1}{100}$	2.2111×10^{-2}	2.6496×10^{-4}	0.53

- [2] M. Dehghan, A. Shokri. *A numerical method for two-dimensional Schrödinger equation using collocation and radial basis functions*, Comput. Math. Appl. 54 (1) (2007), 136–146.
- [3] F. Y. Hajj, *Solution of the Schrodinger equation in two and three dimension*, J. Phys. B. At. Mol. Phys. 18 (1985), 1–11.
- [4] L. Gr. Ixaru, *Operations on oscillatory functions*, Comput. Phys. Commun. 105 (1997), 1–9.
- [5] N. Mai-Duy, T. Tran-Cong, *An efficient indirect RBFN-based method for numerical solution of PDEs*, Numer. Meth. PDE. 21 (4) (2005), 770–790.
- [6] N. Mai-Duy, T. Tran-Cong, *A compact five-point stencil based on integrated RBFs for 2D second-order differential problems*, J. Comput. Phys., 235 (2013), 302–321.
- [7] C. Tien, N. Thai-Quang, N. Mai-Duy, C.-D. Tran, T. Tran-Cong *A three-point coupled compact integrated RBF scheme for second-order differential problems*, Comput. Model. Eng. Sci., 104 (6) (2015), 425–469.



A P_0 WEAK GALERKIN FINITE ELEMENT METHOD FOR SECOND-ORDER PROBLEMS

AllahBakhsh Yazdani Cherati[†], Mohammed S. Cheichan[‡], Hamid momeni⁸

Department of Mathematics, University of Mazandaran, Babolsar, Iran; E-mail : yazdani@umz.ac.ir[†]

Department of Mathematics, Ministry of Education, General Directorate of Education in Basrah, Iraq;

E-mail : cheichan.mohammed@yahoo.com[‡]

Abstract

A lowest-order weak Galerkin (WG) finite element method is proposed and analyzed for solving the second order elliptic problems. This scheme features piecewise-constant on each element and piecewise constant on each edge or face of the element. The piecewise constant WG scheme achieves $O(h)$ and $O(h^2)$ convergence in the H^1 and L^2 norms, respectively. Numerical experiment is conducted to confirm the theoretical results.

Keywords: discrete gradient, finite element methods, second-order elliptic problems, weak Galerkin.

1 Introduction

We consider the second-order elliptic problem, which seeks an unknown function $u(x)$ satisfying,

$$-\nabla \cdot (a \nabla u) = f, \quad \text{in } \Omega, \quad (114)$$

$$u = g, \quad \text{on } \partial\Omega, \quad (115)$$

where Ω is a polytopal domain in $\mathbb{R}^d$ (polygonal or polyhedral domain for $d = 2, 3$), ∇u denotes the gradient of the function u , and a is a symmetric $d \times d$ matrix-valued function in Ω . Assume that there exist two positive numbers $\lambda_1, \lambda_2 > 0$ such that

$$\lambda_1 \xi^t \xi \leq \xi^t a \xi \leq \lambda_2 \xi^t \xi, \quad \forall \xi \in \mathbb{R}^d. \quad (116)$$

We are well aware that in the variational form of the Eq. (654), there is a gradient operator or, in other words the first-order derivative. If v is a constant function, it is clearly $\nabla v = 0$. Therefore, for finite element method (FEM), the degrees of approximation polynomials must be at least one or higher. As an extension of the standard FEM, we use the WG method in which classical derivatives are replaced in the variational equation by the weak derivatives defined on weak finite element functions. By approximating the weak gradient operator in a piecewise-constant space, due to the flexibility of the discrete weak gradient operator, it is possible to use constant functions as approximation functions to

⁸Speaker



obtain the approximate solution of the Eq. (654) by the weak FEM. The main feature of this method is that it allows the use of totally discontinuous finite element function and the trace of finite element function on element boundary may be independent with its value in the interior of element. For second order elliptic equations, a weak function has the form $v = \{v_0, v_b\}$ where v_0 representing the value of v in the interior of each element and v_b on the boundary of the element. Denote by $P_\ell(T)$ and $P_s(\partial T)$ the approximating polynomial spaces for $\{v_0, v_b\}$, where ℓ and s are non-negative integers. In the WG method, we denote the approximate space of the weak partial derivatives with $P_m(T)$. Thus, each combination of $(P_\ell(T), P_s(\partial T), [P_m(T)]^d)$ leads to a WG method for the PDEs under consideration. In [1], a combination of $(P_1(T), P_0(\partial T), [P_0(T)]^2)$ and in [2] combinations of $(P_0(T), P_0(\partial T), [P_1(T)]^2)$ was set as the WG finite element space. In this paper, we introduce a WG scheme in which we use a combination of $(P_0(T), P_0(\partial T), [P_0(T)]^2)$.

2 Weak Galerkin finite element schemes

We define the WG finite element space and its subspace as follows,

$$V_h = \{v = \{v_0, v_b\} : v_0|_T \in P_0(T), v_b|_e \in P_0(e), e \in \partial T, T \in \mathcal{T}_h\}, \quad (117)$$

and

$$V_h^0 = \{v : v \in V_h, v_b = 0 \text{ on } \partial\Omega\}. \quad (118)$$

Definition 0.21. For any $v = \{v_0, v_b\} \in V_h$, the weak gradient $\nabla_w v|_T \in [P_0(T)]^d$ is a constant vector satisfying

$$(\nabla_w v, q)_T = \langle v_b, q \cdot \mathbf{n} \rangle_{\partial T}, \quad \forall q \in [P_0(T)]^d. \quad (119)$$

For each element $T \in \mathcal{T}_h$, denote by Q_0 the L^2 projection from $L^2(T)$ onto $P_0(T)$ and by Q_b the L^2 projection from $L^2(e)$ to $P_0(e)$. Let $\mathbb{Q}_h$ be the L^2 projection of $[L^2(T)]^d$ onto $[P_0(T)]^d$. We define a projection operator $Q_h : H^1(\Omega) \rightarrow V_h$ as follows

$$Q_h u = \{Q_0 u, Q_b u\}, \quad \forall u = \{u_0, u_b\} \in H^1(\Omega).$$

For $v = \{v_0, v_b\}$ and $w = \{w_0, w_b\}$ in V_h , we introduce a bilinear form

$$a(v, w) = \sum_{T \in \mathcal{T}_h} (a \nabla_w v, \nabla_w w)_T + s(v, w), \quad (120)$$

where

$$s(v, w) = \rho \sum_{T \in \mathcal{T}_h} h_T^{-1} \langle v_0 - v_b, w_0 - w_b \rangle_{\partial T}. \quad (121)$$

Weak Galerkin algorithm. Find $u_h = \{u_0, u_b\} \in V_h$ such that $u_b = Q_b g$ on $\partial\Omega$ and satisfying the following equations:

$$a(u_h, v) = (f, v_0), \quad \forall v = \{v_0, v_b\} \in V_h^0. \quad (122)$$

We define the triple bar norm as follows

$$v = \sqrt{a(v, v)}. \quad (123)$$



Lemma 0.22. Let Q_h and $\mathbb{Q}_h$ be the L^2 projection operators defined in previous sections. Then, on each element $T \in \mathcal{T}_h$, we have the following commutative property

$$\nabla_w(Q_h\phi) = \mathbb{Q}_h(\nabla\phi), \quad \forall \phi \in H^1(T). \quad (124)$$

Lemma 0.23. Let $\phi \in H^1(\Omega)$. Then for all $v \in V_h$, we have

$$\sum_{T \in \mathcal{T}_h} (a\nabla\phi, \nabla v_0)_T = \sum_{T \in \mathcal{T}_h} (a\nabla_w(Q_h\phi), \nabla_w v)_T + \sum_{T \in \mathcal{T}_h} \langle (a\mathbb{Q}_h(\nabla\phi)) \cdot \mathbf{n}, v_0 - v_b \rangle_{\partial T}. \quad (125)$$

Lemma 0.24. Let $e_h = Q_h u - u_h$ be the error of the WG finite element solution arising from (122). Then, for any $v \in V_h^0$ we have

$$a(e_h, v) = \ell_u(v) + s(Q_h u, v), \quad (126)$$

where

$$\ell_u(v) = \sum_{T \in \mathcal{T}_h} \langle a(\nabla u - \mathbb{Q}_h \nabla u) \cdot \mathbf{n}, v_0 - v_b \rangle_{\partial T}.$$

Lemma 0.25. Let $\mathcal{T}_h$ be a finite element partition of Ω that is shape regular. Then, for any $u \in H^1(\Omega)$, we have

$$\sum_{T \in \mathcal{T}_h} \|u - Q_0 u\|_T^2 + \sum_{T \in \mathcal{T}_h} h_T^2 \|\nabla(u - Q_0 u)\|_T^2 \leq Ch^2 \|u\|_1^2, \quad (127)$$

$$\sum_{T \in \mathcal{T}_h} \|\nabla u - \mathbb{Q}_h(\nabla u)\|_T^2 + \sum_{T \in \mathcal{T}_h} h_T^2 \|\nabla(\nabla u - \mathbb{Q}_h(\nabla u))\|_T^2 \leq C \|u\|_1^2. \quad (128)$$

Lemma 0.26. Suppose $u \in H^2(\Omega)$ be the exact solution of the problem (654)-(662). Then for $v \in V_h^0$,

$$|\ell_u(v)| \leq Ch \|u\|_2 v, \quad (129)$$

$$|s(Q_h u, v)| \leq Ch \|u\|_2 v. \quad (130)$$

Theorem 0.27. Let $u_h \in V_h$ be the WG finite element solution of the problem (654)-(662) arising from (122). Assume the exact solution $u \in H^2(\Omega)$. Then, there exists a constant C such that

$$u_h - Q_h u \leq Ch \|u\|_2. \quad (131)$$

Proof. By letting $v = e_h$ in (126), we have

$$e_h^2 = \ell_u(e_h) + s(Q_h u, e_h). \quad (132)$$

It then follows from (129) and (130) that

$$e_h^2 \leq Ch \|u\|_2 e_h. \quad (133)$$

This completes the proof. $\square$



We consider a dual problem that seeks $\Phi \in H_0^1(\Omega)$ satisfying

$$-\nabla \cdot (a \nabla \Phi) = e_0, \quad \text{in } \Omega. \quad (134)$$

Theorem 0.28. *Let $u_h \in V_h$ be the WG finite element solution of the problem (654)-(662) arising from (122). Assume the exact solution $u \in H^2(\Omega)$. In addition, assume that the dual problem (134) has the usual H^2 -regularity. Then, there exists a constant C such that*

$$\|u - u_0\| \leq ch^2 \|u\|_2. \quad (135)$$

Proof. By testing (134) with e_0 we obtain

$$\begin{aligned} \|e_0\|^2 &= -(\nabla \cdot (a \nabla \Phi), e_0) \\ &= \sum_{T \in \mathcal{T}_h} (a \nabla \Phi, \nabla e_0)_T - \sum_{T \in \mathcal{T}_h} \langle (a \nabla \Phi) \cdot \mathbf{n}, e_0 - e_b \rangle_{\partial T}. \end{aligned} \quad (136)$$

Setting $\phi = \Phi$ and $v = e_h$ in (239) yields

$$\sum_{T \in \mathcal{T}_h} (a \nabla \Phi, \nabla e_0)_T = \sum_{T \in \mathcal{T}_h} (a \nabla_w(Q_h \Phi), \nabla_w e_h)_T + \sum_{T \in \mathcal{T}_h} \langle (a \nabla_h(\nabla \Phi)) \cdot \mathbf{n}, e_0 - e_b \rangle_{\partial T}. \quad (137)$$

Substituting (137) into (136) gives

$$\begin{aligned} \|e_0\|^2 &= \sum_{T \in \mathcal{T}_h} (a \nabla_w e_h, \nabla_w(Q_h \Phi))_T + \sum_{T \in \mathcal{T}_h} \langle a(Q_h(\nabla \Phi) - \nabla \Phi) \cdot \mathbf{n}, e_0 - e_b \rangle_{\partial T} \\ &= \sum_{T \in \mathcal{T}_h} (a \nabla_w e_h, \nabla_w(Q_h \Phi))_T + \ell_\Phi(e_h). \end{aligned} \quad (138)$$

Using equation (120) and the error equation (126), we can get

$$(a \nabla_w e_h, \nabla_w(Q_h \Phi)) = \ell_u(Q_h \Phi) + s(Q_h u, Q_h \Phi) - s(e_h, Q_h \Phi). \quad (139)$$

By combining (139) with (138), we arrive at

$$\|e_0\|^2 = \ell_u(Q_h \Phi) + s(Q_h u, Q_h \Phi) - s(e_h, Q_h \Phi) + \ell_\Phi(e_h). \quad (140)$$

$$\begin{aligned} |\ell_u(Q_h \Phi)| &= \left| \sum_{T \in \mathcal{T}_h} \langle a(\nabla u - Q_h \nabla u) \cdot \mathbf{n}, Q_0 \Phi - Q_b \Phi \rangle_{\partial T} \right| \\ &\leq \left| \sum_{T \in \mathcal{T}_h} \langle a(\nabla u - Q_h \nabla u) \cdot \mathbf{n}, Q_0 \Phi - \Phi \rangle_{\partial T} \right| + \left| \sum_{T \in \mathcal{T}_h} \langle a(\nabla u - Q_h \nabla u) \cdot \mathbf{n}, \Phi - Q_b \Phi \rangle_{\partial T} \right|. \end{aligned} \quad (141)$$

By using the definition of Q_b and the fact that $\Phi = 0$ on $\partial\Omega$, we obtain

$$\sum_{T \in \mathcal{T}_h} \langle a(\nabla u - Q_h \nabla u) \cdot \mathbf{n}, \Phi - Q_b \Phi \rangle_{\partial T} = \sum_{T \in \mathcal{T}_h} \langle a \nabla u \cdot \mathbf{n}, \Phi - Q_b \Phi \rangle_{\partial T} = 0. \quad (142)$$



From the trace inequality and the estimate (127) we have

$$\begin{aligned} \left(\sum_{T \in \mathcal{T}_h} \|Q_0 \Phi - \Phi\|_{\partial T}^2 \right)^{1/2} &\leq Ch^{3/2} \|\Phi\|_2, & \left(\sum_{T \in \mathcal{T}_h} \|a(\nabla u - Q_h \nabla u)\|_{\partial T}^2 \right)^{1/2} &\leq Ch^{1/2} \|u\|_2. \\ \left| \sum_{T \in \mathcal{T}_h} \langle a(\nabla u - Q_h \nabla u) \cdot \mathbf{n}, Q_0 \Phi - \Phi \rangle_{\partial T} \right| &\leq C \left(\sum_{T \in \mathcal{T}_h} \|a(\nabla u - Q_h \nabla u)\|_{\partial T}^2 \right)^{1/2} \\ &\quad \cdot \left(\sum_{T \in \mathcal{T}_h} \|Q_0 \Phi - \Phi\|_{\partial T}^2 \right)^{1/2} \leq Ch^2 \|u\|_2 \|\Phi\|_2. \end{aligned} \quad (143)$$

Combining (141) with (142), and (143) yields

$$|\ell_u(Q_h \Phi)| \leq Ch^2 \|u\|_2 \|\Phi\|_2. \quad (144)$$

$$\begin{aligned} |s(Q_h u, Q_h \Phi)| &\leq \sum_{T \in \mathcal{T}_h} h_T^{-1} |\langle Q_0 u - Q_b u, Q_0 \Phi - Q_b \Phi \rangle_{\partial T}| \\ &= \sum_{T \in \mathcal{T}_h} h_T^{-1} |\langle Q_0 u - u, Q_0 \Phi - \Phi \rangle_{\partial T}| \\ &\leq C \left(\sum_{T \in \mathcal{T}_h} h_T^{-1} \|Q_0 u - u\|_{\partial T}^2 \right)^{1/2} \left(\sum_{T \in \mathcal{T}_h} h_T^{-1} \|Q_0 \Phi - \Phi\|_{\partial T}^2 \right)^{1/2} \\ &\leq Ch^2 \|u\|_2 \|\Phi\|_2. \end{aligned} \quad (145)$$

The estimates (130) and (131) imply

$$|s(e_h, Q_h \Phi)| \leq Ch \|\Phi\|_2 e_h \leq Ch^2 \|u\|_2 \|\Phi\|_2. \quad (146)$$

Similarly, it follows from (129) and (131) that

$$|\ell_\Phi(e_h)| \leq Ch^2 \|u\|_2 \|\Phi\|_2. \quad (147)$$

Now substituting (144)-(147) into (140) yields

$$\|e_0\|^2 \leq Ch^2 \|u\|_2 \|\Phi\|_2, \quad (148)$$

which, combined with the regularity assumption and the triangle inequality, provides the required error estimate (135). $\square$

Example 0.29. we consider the model problem (654)-(662). The boundary condition $u|_{\partial\Omega} = g$ and f are chosen such that the exact solution is given by $u = x(1-x)y(1-y)$ and

$$a = \begin{pmatrix} x+y & 0 \\ 0 & x+y \end{pmatrix}.$$

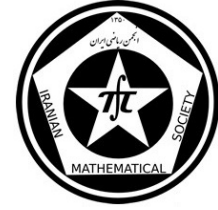


Table 9: Numerical result for the first example

h	e_h	Order	$\ e_0\ $	Order
1/2	2.5652e-02		1.6638e-02	
1/4	1.3064e-02	0.9735	4.4030e-03	1.9179
1/8	6.5541e-03	0.9951	1.1164e-03	1.9796
1/16	3.2797e-03	0.9988	2.8048e-04	1.9929
1/32	1.6403e-03	0.9996	7.1135e-05	1.9793
1/64	8.2045e-04	0.9995	2.0685e-05	1.7820

References

- [1] L.Mu, J.Wang, X. Ye, *A weak Galerkin finite element method with polynomial reduction*, J. Comput. Appl.Math. vol. 285 (2015) pp. 45-48.
- [2] L. Mu, J. Wang, X. Ye, *A hybridized formulation for the weak Galerkin mixed finite element method*, J. Comput. Appl.Math.vol. 307 (2016) pp. 335-345.



A point-collocation approach for solving ordinary differential equations with implicit Runge-Kutta methods

Amin Abrishami Moghadam^{†9}, Ali R. Soheili[‡]

Department of Applied Mathematics, Ferdowsi University of Mashhad, Mashhad, Iran

amin_abrishamimoghadam@mail.um.ac.ir^{†*}

soheili@um.ac.ir[‡]

Abstract

For solving an initial value problem with implicit Runge-Kutta methods, we encounter with a system of nonlinear equations. However solving a system of nonlinear equations will increase the difficulty of computations. Instead in this paper we present a point-collocation method for integrating the differential function.

AMS subject Classification 2010: 65L06, 42C10, 65L60, 34A12

Keywords: Implicit Runge-Kutta, Integration, Collocation, Initial value problem(IVP)

1 Introduction

An initial value problem is a differential equation $y^{(n)} = f(x, y, y', y'', \dots, y^{(n-1)})$ with $f: \Omega \subset \mathbb{R} \times \mathbb{R}^n \times \dots \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ where Ω is an open set of $\mathbb{R} \times \mathbb{R}^n \times \dots \times \mathbb{R}^n$ together with a condition $y(x_0) = y_0$ which means a point in the domain of f , $(x_0, y_0) \in \Omega$ is called an initial condition. Also $y(x)$ denotes the exact solution of the differential equation.

According to [1] the Butcher tableau of an s-stage implicit Runge-Kutta methods is a full matrix $A = (a_{i,j})$, and the method has the form:

$$y_{n+1} = y_n + h \sum_{i=1}^s b_i k_i,$$

where k_i for $i = 1, \dots, s$ are:

$$k_i = f(x_n + c_i h, y_n + h \sum_{j=1}^s a_{ij} k_j).$$

Since we are applying the implicit Runge-Kutta methods for solving ODEs then the above formulation results a system of nonlinear equations.

⁹Speaker



2 Main Results

In this section, we describe the main idea of our approach for solving ODEs. To illustrate the basic idea of point-collocation method, we first define the collocation points within the notation of x_i^j as follows:

$$x_i^j = x_{i-1} + c_j h_i, \quad (149)$$

where $0 \leq c_j \leq 1$ are called collocation points for $j = 1, \dots, s$ and h_i are step lengths for $i = 1, \dots, n$ that are not essentially equal. When the collocation points are added to our partition, our partition gets refined. Though we define the Lagrange basis polynomials at the collocation points then we'll show how to find the Lagrange interpolation function of $y'(x)$.

Definition 0.30. Lagrange basis polynomials at the collocation points $0 \leq c_j \leq 1$ for $j = 1, \dots, s$ are as follows:

$$L_i(c) = \prod_{\substack{j=1 \\ j \neq i}}^s \frac{c - c_j}{c_i - c_j},$$

and if the function $g(c)$ at the points $0 \leq c_i \leq 1$ for $i = 1, \dots, s$ has values $g(c_i)$ then the function $g(c)$ can be written in the following form:

$$g(c) = P(c) + R(c),$$

where $P(c)$ is the Lagrange interpolation function of g and $R(c)$ is called interpolation error:

$$P(c) = \sum_{i=1}^s g(c_i) L_i(c).$$

If function $g(c) \in C^{(s)}[a, b]$ and $\zeta \in [a, b]$ then the interpolation Error is as follows:

$$R(c) = \frac{\prod_{j=1}^s (c - c_j) g^{(s)}(\zeta)}{s!}.$$

Now if function $y'(x)$ or $f(x, y)$ is sufficiently differentiable on $[a, b]$ meaning that $y'(x) \in C^{(s)}[a, b]$ then:

$$y'(x) = \sum_{j=1}^s y'(x_i^j) L_j(c) + R(c), \quad (150)$$

for every $x \in [x_{i-1}, x_i]$, (149) can be rewritten as follows:

$$x = x_{i-1} + c h_i, \quad (151)$$

where $0 \leq c \leq 1$. Clearly $c = \frac{x - x_{i-1}}{h_i}$ though (150) can be rewritten as follows:

$$y'(x) = \sum_{j=1}^s y'(x_i^j) L_j\left(\frac{x - x_{i-1}}{h_i}\right) + R\left(\frac{x - x_{i-1}}{h_i}\right).$$



Now we're in the position to prove the main theorem which is an extension of theorem based on Runge-Kutta methods (See [1, 2])

Theorem 0.31. *Consider the following IVP*

$$\begin{cases} y'(x) = f(x, y) \\ y(x_0) = y_0 \end{cases}$$

satisfies the relations:

$$y(x_i) = y(x_{i-1}) + h_i \sum_{j=1}^s b_j k_{ij},$$

$$y(x_i^j) = y(x_{i-1}) + h_i \sum_{k=1}^s a_{jk} k_{ik},$$

$$k_{ij} = f(x_i^j, y(x_i^j)) = f(x_{i-1} + c_j h_i, y(x_{i-1}) + h_i \sum_{k=1}^s a_{jk} k_{ik}). \quad i = 1 \dots, n, \quad j = 1, \dots, s$$

Proof. By the definition of the IVP we first integrate both sides of $y'(x) = f(x, y)$ on $[x_{i-1}, x_i]$:

$$\begin{aligned} \int_{x_{i-1}}^{x_i} y'(x) dx &= \int_{x_{i-1}}^{x_i} f(x, y) dx, \\ y(x_i) - y(x_{i-1}) &= \int_{x_{i-1}}^{x_i} f(x, y) dx, \\ y(x_i) &= y(x_{i-1}) + \int_{x_{i-1}}^{x_i} f(x, y) dx. \end{aligned}$$

It is clear that for every $x \in [x_{i-1}, x_i]$, $[x_{i-1}, x] \subseteq [x_{i-1}, x_i]$. If the whole integration process done on the subset $[x_{i-1}, x]$ then:

$$y(x) = y(x_{i-1}) + \int_{x_{i-1}}^x f(x, y) dx = y(x_{i-1}) + \int_{x_{i-1}}^x y'(x) dx.$$

By (150), we substitute the Lagrange interpolation function of $y'(x)$ in the above integral:

$$y(x) = y(x_{i-1}) + \int_{x_{i-1}}^x \left[\sum_{j=1}^s y'(x_i^j) L_j(c) + R(c) \right] dx, \quad (152)$$

$$y(x) = y(x_{i-1}) + \left[\sum_{j=1}^s y'(x_i^j) \int_{x_{i-1}}^x L_j(c) dx + \int_{x_{i-1}}^x R(c) dx \right]. \quad (153)$$

According to (151) Let:

$$\begin{aligned} \mu = c &= \frac{(x - x_{i-1})}{h_i}, \\ d\mu &= \frac{dx}{h_i}. \end{aligned}$$



If $x = x_{i-1}$ then $\mu = \frac{(x_{i-1}-x_{i-1})}{h_i} = 0$. Though $0 \leq \mu \leq c$ and $h_i d\mu = dx$

$$\begin{aligned}\int_{x_{i-1}}^x L_j(c)dx &= h_i \int_0^c L_j(\mu)d\mu, \\ \int_{x_{i-1}}^x R(c)dx &= h_i \int_0^c R(\mu)d\mu.\end{aligned}$$

Though (153) becomes:

$$y(x) = y(x_{i-1}) + h_i \left[\sum_{j=1}^s y'(x_i^j) \int_0^c L_j(\mu)d\mu + \int_0^c R(\mu)d\mu \right]. \quad (154)$$

Now let $x = x_i^k$ in (154)

$$y(x_i^k) = y(x_{i-1}) + h_i \left[\sum_{j=1}^s y'(x_i^j) \int_0^{c_k} L_j(\mu)d\mu + \int_0^{c_k} R(\mu)d\mu \right]. \quad (155)$$

The following coefficients can be defined as follow:

$$a_{kj} := \int_0^{c_k} L_j(\mu)d\mu, \quad (156)$$

$$E_k := \int_0^{c_k} R(\mu)d\mu. \quad (157)$$

By the above coefficients a_{kj} , E_k from (156) and (157) the equation (155) can be rewritten as follows:

$$y(x_i^k) = y(x_{i-1}) + h_i \left[\sum_{j=1}^s y'(x_i^j) a_{kj} + E_k \right]. \quad (158)$$

If we let $x = x_i$ in (154) then

$$y(x_i) = y(x_{i-1}) + h_i \left[\sum_{j=1}^s y'(x_i^j) \int_0^1 L_j(\mu)d\mu + \int_0^1 R(\mu)d\mu \right]. \quad (159)$$

The following coefficients can be defined as follow:

$$b_j := \int_0^1 L_j(\mu)d\mu, \quad (160)$$

$$E := \int_0^1 R(\mu)d\mu. \quad (161)$$

By the above coefficients b_j , E from (160) and (161) the equation (159) can be rewritten as follows:

$$y(x_i) = y(x_{i-1}) + h_i \sum_{j=1}^s y'(x_i^j) b_j + h_i \sum_{j=1}^s E. \quad (162)$$



Clearly the relation between b_j and a_{kj} from (156) and (160) can be interpreted as follows:

$$b_j = \int_0^1 L_j(\mu) d\mu = \int_0^{c_k} L_j(\mu) d\mu + \int_{c_k}^1 L_j(\mu) d\mu,$$

$$b_j = a_{kj} + \int_{c_k}^1 L_j(\mu) d\mu.$$

By neglecting the error terms E_k from (157) for $k = 1, \dots, s$ then $E = 0$ in (162):

$$y(x_i) \approx y(x_{i-1}) + h_i \sum_{j=1}^s y'(x_i^j) b_j. \quad (163)$$

Similarly in (158), the value of function $y(x)$ at the collocation points $x = x_i^j$ equals:

$$y(x_i^j) \approx y(x_{i-1}) + h_i \sum_{k=1}^s y'(x_i^k) a_{jk}.$$

If we want to calculate the values of $y'(x_i^j)$ then:

$$y'(x_i^j) = f(x_i^j, y(x_i^j)) = f(x_{i-1} + c_j h_i, y(x_{i-1}) + h_i \sum_{k=1}^s a_{jk} k_{ik}),$$

where $k_{ik} = y'(x_i^k)$ and $k_{ij} = y'(x_i^j)$. □

Example 0.32. Consider the following IVP

$$\begin{cases} y'(x) = 1 + y^2(x) & x \in [0, 1] \\ y(0) = 0. \end{cases}$$

The exact solution of the above problem is $y(x) = \tan(x)$. Now in this example we would like to apply a 3-stage implicit Runge-Kutta method based on the Butcher tableau of [1] which has order of 6:

$\frac{1}{2} - \frac{\sqrt{15}}{10}$	$\frac{5}{36}$	$\frac{2}{9} - \frac{\sqrt{15}}{15}$	$\frac{5}{36} - \frac{\sqrt{15}}{30}$
$\frac{1}{2}$	$\frac{5}{36} + \frac{\sqrt{15}}{24}$	$\frac{2}{9}$	$\frac{5}{36} - \frac{\sqrt{15}}{24}$
$\frac{1}{2} + \frac{\sqrt{15}}{10}$	$\frac{5}{36} + \frac{\sqrt{15}}{30}$	$\frac{2}{9} + \frac{\sqrt{15}}{15}$	$\frac{5}{36}$
	$\frac{5}{18}$	$\frac{4}{9}$	$\frac{5}{18}$

From the above Butcher tableau one can easily obtain the collocation points, $c_1 = \frac{1}{2} - \frac{\sqrt{15}}{10}$, $c_2 = \frac{1}{2}$ and $c_3 = \frac{1}{2} + \frac{\sqrt{15}}{10}$. We now apply our approach by using (163) and letting $n = 10$ or $h = 0.1$. The numerical results can be observed in Table 10. Also the comparison of absolute errors of our approach, explicit RK6, explicit RK4 are illustrated in Figure 35.



x	Explicit RK4 Method	Explicit RK6 Method	Implicit RK3(Our Approach)	y(x)=tan(x)
0	0	0	0	0
0.1	0.100334589	0.100334671	0.100334672	0.100334672
0.2	0.202709878	0.202710030	0.202710035	0.202710035
0.3	0.309336039	0.309336236	0.309336249	0.309336249
0.4	0.422792992	0.422793189	0.422793218	0.422793218
0.5	0.546302307	0.546302432	0.546302489	0.546302488
0.6	0.684136756	0.684136699	0.684136807	0.684136808
0.7	0.842288569	0.842288175	0.842288379	0.842288379
0.8	1.029639061	1.029638167	1.029638554	1.029638557
0.9	1.260158782	1.260157459	1.260158210	1.260158219
1.0	1.557406442	1.557406199	1.557407702	1.557407725

Table 10: Numerical results and comparison of our approach with explicit RK4, explicit RK6 and the exact solution for $0 \leq x \leq 1$.

3 Conclusion

In this paper we have presented a point-collocation method by using Lagrange interpolation for integrating the function $f(x, y)$. Although we have used a 3-stage implicit Runge-Kutta of order 6 as an example of our approach. According to Figure 35, this could be seen that our approach is much more accurate than sixth order explicit Runge-Kutta method. In Example 0.32 the maximum error of this approach is $2.17678\text{e-}08$.

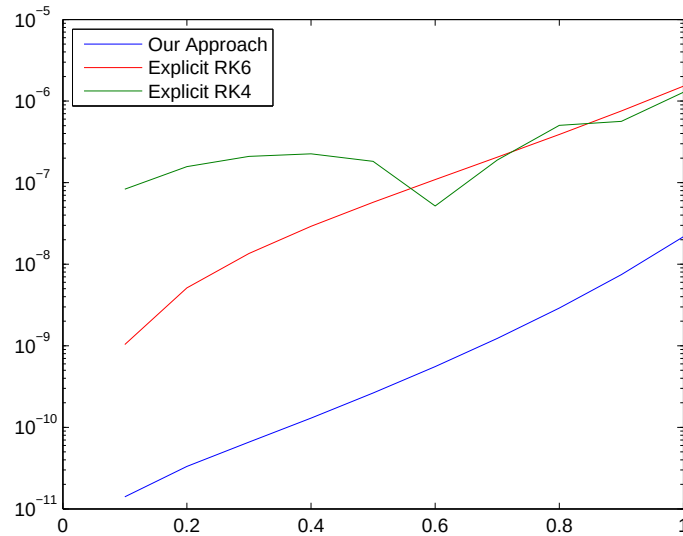
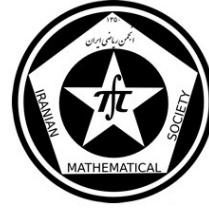


Figure 10: Comparison of absolute errors of our approach, explicit RK6, explicit RK4



References

- [1] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems The Initial Value Problem*, Wiley, 1991.
- [2] David F. Griffiths and Desmond J. Higham, *Numerical Methods for Ordinary Differential Equations Initial Value Problems*, Springer, 2010.
- [3] F. Costabile and A. Napoli, *A class of collocation methods for numerical integration of initial value problems*, Computers and Mathematics with Applications, 62 (2011) 3221-3235.
- [4] M. Ghabarpoor, M. T. Kajani and M. Allame, *A collocation method for differential equations with one-point Taylor initial conditions*, Mathematical Methods in the Applied Sciences, (2019) 1-13.
- [5] F. Costabile and A. Napoli, *A class of Birkhoff-Lagrange-collocation methods for high order boundary value problems*, Applied Numerical Mathematics, 116 (2017) 129-140.



Application of Caputo fractional derivative operator for solving the fractional optimal control problems

Seyed Mehdi Shafiof^{†, 10}

sm.shafiof@gmail.com[†]

Abstract

A numerical method is presented based on the hybrid functions including, Bernoulli polynomials and Block-pulse functions for solving a class of fractional optimal control problems (FOCPs). A new approach is proposed for calculating the Caputo fractional derivative (CFD) operator of the Bernoulli hybrid functions. First, CFD operator and hybrid functions are used to convert the FOCP into an unrestricted optimization problem. Then, by calculating the necessary optimality conditions, a system of algebraic equations is obtained. From solving the resulted system, unknown coefficients of the state variable are determined. The efficiency and effectiveness of the proposed method are shown by some test problems.

AMS subject Classification 2010: 11B68, 26A33, 49M05.

Keywords: Bernoulli hybrid functions. Fractional optimal control. Caputo fractional derivative

1 Introduction

In recent years, fractional calculus has been considered for describing and modeling many scientific phenomena [6, 2]. A FOCP is a particular case of an optimal control problem in which the objective function or constraints of the problem, or both, contain at least one fractional differential equations [1]. Many researchers have implemented hybrid functions for solving various problems because they are powerful mathematical tools to approximate unknown functions [6]. In the current article, we propose a direct numerical method based on the CFD operator for the Bernoulli hybrid functions to solve FOCPs, including boundary conditions with the final point. We consider the following FOCP

$$\min J = \int_0^1 f(t, x(t), u(t)) dt, \quad (164)$$

¹⁰Department of Mathematics, Payame Noor University, P.O.Box 19395-3697, Tehran, Iran



subject to

$$Z_1 \dot{x}(t) + Z_2 {}^C D_t^\alpha x(t) = h(t, x(t)) + e(t)u(t), \quad (165)$$

with boundary conditions

$$x(0) = x_0, x(1) = x_1, 0 < \alpha \leq 1. \quad (166)$$

Where $f(t, x(t), u(t))$, $h(t, x(t))$ and $e(t)$ are smooth, $e(t)$ is nonzero and, Z_1 and Z_2 are scalar numbers. In order to obtain the numerical solution for the problem mentioned in equations (689)-(610), first, the CFD operator for the Bernoulli hybrid functions is calculated by the new method. Then, the state variable is expanded by the hybrid functions with unknown coefficients. The control variable is derived from the dynamical system according to the state variable using the CFD operator. The boundary conditions are added to the objective function by using the undetermined Lagrange multipliers. By applying the Legendre-Gauss quadrature rule to approximate the integral and the necessary optimality conditions for the problem, the given FOCP, is transformed to gain the solution of a system of nonlinear equations.

2 Preliminaries and notation

In this section, we present some of the basic definitions and some required properties.

Fractional calculus

Let $f : [a, b] \mapsto R$ be a function, and $\alpha > 0$ be the order of fractional derivative and $i = [\alpha] + 1$.

Definition 0.33. The Caputo fractional derivative is defined by [3]

$${}_a^C D_t^\alpha f(t) = \frac{1}{\Gamma(i - \alpha)} \int_a^t (t - \tau)^{i - \alpha - 1} f^{(i)}(\tau) d\tau, \quad (167)$$

also, we have

$${}_0^C D_t^\alpha f(t) = \frac{1}{\Gamma(i - \alpha)} t^{i - \alpha - 1} * f^{(i)}(t), \quad (168)$$

where $t^{(i - \alpha - 1)} * f^{(i)}(t)$ is the convolution of $t^{(i - \alpha - 1)}$ and $f^{(i)}(t)$ and $\Gamma(\alpha)$ is the Gamma function. For Caputo fractional derivative, we have [6]

$${}_a^C D_t^\alpha C = 0, \quad (C \text{ is constant}) \quad {}_a^C D_t^z f(t) = f^{(z)}(t), \quad z \text{ is integer.} \quad (169)$$

Bernoulli polynomials

Bernoulli polynomials of order m can be defined as: [9]

$$\beta_m(t) = \sum_{k=0}^m \binom{m}{k} \alpha_{m-k} t^k, \quad (170)$$

where $\alpha_k, k = 0, 1, 2, \dots, m$, are Bernoulli numbers. These polynomials satisfy the following relation:

$$\beta_m(1 - t) = (-1)^m \beta_m(t). \quad (171)$$



Hybrid of Block-pulse functions and Bernoulli polynomials

Hybrid functions $b_{nm}(t)$ for $n = 1, 2, \dots, N, m = 0, 1, 2, \dots, M$ are given over the interval $[0, 1]$ as: [4]

$$b_{nm}(t) = \begin{cases} \beta_m(Nt - n + 1), & t \in [\frac{n-1}{N}, \frac{n}{N}), \\ 0, & \text{otherwise,} \end{cases} \quad (172)$$

where n and m are the orders of the Block-pulse functions and Bernoulli polynomials, respectively. Approximating $f(t)$ by means of the Bernoulli hybrid functions yields:

$$f(t) \simeq \sum_{n=1}^N \sum_{m=0}^M a_{nm} b_{nm}(t) = A^T B(t),$$

where A and $B(t)$ are the following vectors:

$$B^T(t) = [b_{10}(t), b_{20}(t), \dots, b_{N0}(t), b_{11}(t), \dots, b_{N1}(t), \dots, b_{1M}(t), \dots, b_{NM}(t)], \quad (173)$$

$$A^T = [a_{10}, a_{20}, \dots, a_{N0}, a_{11}, \dots, a_{N1}, \dots, a_{1M}, \dots, a_{NM}]. \quad (174)$$

3 CFD operator

The first step in developing our numerical method is to compute the CFD operator of the Bernoulli hybrid functions to be applicable in the proposed method. We display this operator by Δ^α . We consider the interval, be $[0, 1]$ and $0 < \alpha \leq 1$. By using the definition of the Laplace transform and Bernoulli hybrid functions (536) and equations (603) and (604), we have

$$\begin{aligned} L\{b_{nm}(t)\} &= \int_0^\infty e^{-st} b_{nm}(t) dt = \int_{\frac{n-1}{N}}^{\frac{n}{N}} e^{-st} \beta_m(Nt - n + 1) dt = \int_{\frac{n-1}{N}}^{\frac{n}{N}} e^{-st} \beta_m(Nt - n + 1) dt \\ &\quad - \int_{\frac{n}{N}}^\infty e^{-st} \beta_m(Nt - n + 1) dt = e^{-\frac{n-1}{N}s} \int_{\frac{n-1}{N}}^\infty e^{-s(t - \frac{n-1}{N})} \beta_m\left(N\left(t - \frac{n-1}{N}\right)\right) dt \\ &\quad - e^{-\frac{n}{N}s} \int_{\frac{n}{N}}^\infty e^{-s(t - \frac{n}{N})} \beta_m\left(1 - (-N)\left(t - \frac{n}{N}\right)\right) dt \\ &= e^{-\frac{n-1}{N}s} \left(\int_{\frac{n-1}{N}}^\infty \left(\sum_{k=0}^m e^{-s(t - \frac{n-1}{N})} \binom{m}{k} \alpha_{m-k} N^k \left(t - \frac{n-1}{N}\right)^k \right) dt \right) \\ &\quad - e^{-\frac{n}{N}s} \left(\int_{\frac{n}{N}}^\infty \left(\sum_{k=0}^m \binom{m}{k} \alpha_{m-k} (-1)^m (-N)^k e^{-s(t - \frac{n}{N})} \left(t - \frac{n}{N}\right)^k \right) dt \right), \end{aligned}$$

Using integration by the change of variable, the Laplace transform is obtained as follows:

$$L\{b_{nm}(t)\} = e^{-\frac{n-1}{N}s} \left(\sum_{k=0}^m \binom{m}{k} \alpha_{m-k} N^k \frac{\Gamma(k+1)}{s^{k+1}} \right) - e^{-\frac{n}{N}s} \left(\sum_{k=0}^m (-1)^m \binom{m}{k} \alpha_{m-k} (-N)^k \frac{\Gamma(k+1)}{s^{k+1}} \right).$$

By applying the derivative Laplace transform properties and above equation, we get

$$\begin{aligned} L\{(b_{nm}(t))'\} &= sL\{b_{nm}(t)\} - b_{nm}(0) = e^{-\frac{n-1}{N}s} \left(\sum_{k=0}^m \binom{m}{k} \alpha_{m-k} N^k \frac{\Gamma(k+1)}{s^k} \right) \\ &\quad - e^{-\frac{n}{N}s} \left(\sum_{k=0}^m (-1)^m \binom{m}{k} \alpha_{m-k} (-N)^k \frac{\Gamma(k+1)}{s^k} \right) - b_{nm}(0). \end{aligned} \quad (175)$$



By using equations (394) and (611) and taking laplace transform of the Convolution product, we obtain

$$\begin{aligned} L [{}^C_0 D_t^\alpha (b_{nm}(t))] &= L \left[\frac{1}{\Gamma(1-\alpha)} \int_0^t (t-\tau)^{-\alpha} (b_{nm}(t))' d\tau \right] = L \left[\frac{1}{\Gamma(1-\alpha)t^\alpha} * (b_{nm}(t))' \right] \\ &= \frac{1}{s^{1-\alpha}} \times L \{ (b_{nm}(t))' \} = e^{-\frac{n-1}{N}s} \left(\sum_{k=0}^m \binom{m}{k} \alpha_{m-k} N^k \frac{\Gamma(k+1)}{s^{k+1-\alpha}} \right) \\ &\quad - e^{-\frac{n}{N}s} \left(\sum_{k=0}^m (-1)^m \binom{m}{k} \alpha_{m-k} (-N)^k \frac{\Gamma(k+1)}{s^{k+1-\alpha}} \right) - \frac{b_{nm}(0)}{s^{1-\alpha}}. \end{aligned} \quad (176)$$

By taking inverse Laplace transform of relation (543), the CFD operator is derived as follows:

$$\begin{aligned} \Delta^\alpha (b_{nm}(t)) &= u_{\frac{n-1}{N}}(t) \sum_{k=0}^m \binom{m}{k} \alpha_{m-k} N^k \frac{\Gamma(k+1)}{\Gamma(k+1-\alpha)} \left(t - \frac{n-1}{N} \right)^{k-\alpha} \\ &\quad - u_{\frac{n}{N}}(t) \sum_{k=0}^m (-1)^m \binom{m}{k} \alpha_{m-k} (-N)^k \frac{\Gamma(k+1)}{\Gamma(k+1-\alpha)} \left(t - \frac{n}{N} \right)^{k-\alpha} - \frac{b_{nm}(0)}{\Gamma(1-\alpha)t^\alpha}. \end{aligned} \quad (177)$$

Where $u_r(t)$ is a unit step function defined as:

$$u_r(t) = \begin{cases} 1 & t \geq r, \\ 0 & t < r. \end{cases}$$

Remark 0.34. If $\alpha = 1$, according to equation (612), Δ^1 operator is calculated directly from relation (540).

4 Numerical method

In this section, we describe the numerical method for solving FOCP given in equations (689) to (610), using the hybrid functions and CFD operator. We expand the state variable with respect to the hybrid functions as follows:

$$x(t) = A^T B(t), \dot{x}(t) = A^T \Delta^1 B(t), {}^C_0 D_t^\alpha x(t) = A^T \Delta^\alpha B(t), \quad (178)$$

A and B are the vectors defined in (535) and (538). The control variable can be calculated using equation (609). Then, by substituting relations (616) into control variable, we obtain

$$u(t) = \frac{1}{e(t)} \left(Z_1 (A^T \Delta^1 B(t)) + Z_2 (A^T \Delta^\alpha B(t)) - h(t, A^T B(t)) \right). \quad (179)$$

The substitution of these approximations into the performance index (689) yields:

$$J(A) = \int_0^1 f \left(t, A^T B(t), \frac{1}{e(t)} \left(Z_1 (A^T \Delta^1 B(t)) + Z_2 (A^T \Delta^\alpha B(t)) - h(t, A^T B(t)) \right) \right) dt. \quad (180)$$

Let

$$J^* = J(A) + \mu_0 (x(0) - A^T B(0)) + \mu_1 (x(1) - A^T B(1)), \quad (181)$$



where μ_0 and μ_1 are the unknown Lagrange multipliers. We first evaluate J numerically using the Legendre-Gauss quadrature formula [2]. Then, the necessary optimality conditions for the extreme J^* are obtained as:

$$\frac{\partial J^*}{\partial a_{ij}} = 0, \quad i = 1, 2, \dots, N, j = 0, 1, \dots, M, \quad \frac{\partial J^*}{\partial \mu_k} = 0, k = 0, 1. \quad (182)$$

The unknown coefficients and subsequently the state and control variables are achieved by solving the system of algebraic equations (619).

5 Illustrative Examples

In this section, we solve two examples using the proposed method.

Example 0.35. Consider the following nonlinear problem [5]

$$\begin{aligned} \min J &= \int_0^1 (u^2(t) - 4x(t))^2 dt \\ \dot{x}(t) + {}^C D_t^{0.5} x(t) &= u(t) + \frac{8}{3\sqrt{\pi}} t^{1.5}, \quad x(0) = 0, \quad x(1) = 1. \end{aligned} \quad (183)$$

For this problem, $(x(t), u(t)) = (t^2, 2t)$ is the exact solution. To implement the CFD method, for $N = 1, M = 2$, the hybrid function approximations of equations (616) are written as follows:

$$\begin{aligned} x(t) &= A^T B(t) = a_{10}b_{10}(t) + a_{11}b_{11}(t) + a_{12}b_{12}(t), \\ {}^C D_t^{0.5} x(t) &= a_{10}\Delta^{0.5}b_{10}(t) + a_{11}\Delta^{0.5}b_{11}(t) + a_{12}\Delta^{0.5}b_{12}(t), \\ \dot{x}(t) &= a_{10}\Delta^1b_{10}(t) + a_{11}\Delta^1b_{11}(t) + a_{12}\Delta^1b_{12}(t). \end{aligned}$$

By using these equations, the control variable is obtained from equation (620) as:

$$\begin{aligned} u(t) &= a_{10}\Delta^1b_{10}(t) + a_{11}\Delta^1b_{11}(t) + a_{12}\Delta^1b_{12}(t) + a_{10}\Delta^{0.5}b_{10}(t) + a_{11}\Delta^{0.5}b_{11}(t) \\ &+ a_{12}\Delta^{0.5}b_{12}(t) - \frac{8}{3\sqrt{\pi}}t^{1.5}. \end{aligned}$$

By substituting these relations into equation (618), we get

$$\begin{aligned} J^* &= J + \mu_0 (x(0) - a_{10}b_{10}(0) - a_{11}b_{11}(0) - a_{12}b_{12}(0)) \\ &+ \mu_1 (x(1) - a_{10}b_{10}(1) - a_{11}b_{11}(1) - a_{12}b_{12}(1)). \end{aligned}$$

By solving the system (619), the unknown coefficients $x(t)$ are achieved as follows: $a_{10} = \frac{1}{3}, a_{11} = 1, a_{12} = 1$. That, we obtain the exact values of $x(t)$ and $u(t)$ while in [5], the maximum absolute errors of the state and control variables are reported using the state parameterization method based on shifted Legendre polynomials with $m = 4$ are 4.4817×10^{-5} and 4.1937×10^{-4} , respectively.

Example 0.36. Consider the following FOCP with the quadratic performance index [2]

$$\min J = \frac{1}{2} \int_0^1 (x^2(t) + u^2(t)) dt,$$

subject to the given constraint and boundary conditions

$$\frac{1}{2}\dot{x}(t) + \frac{1}{2} {}^C D_t^\alpha x(t) = -x(t) + u(t), \quad x(0) = 1, \quad x(1) = \cosh(\sqrt{2}) + \omega \sinh(\sqrt{2}).$$



The exact solution for $\alpha = 1$, is given by

$$x(t) = \cosh(\sqrt{2}t) + \omega \sinh(\sqrt{2}t),$$

$$u(t) = (1 + \sqrt{2}\omega) \cosh(\sqrt{2}t) + (\sqrt{2} + \omega) \sinh(\sqrt{2}t), \omega = -\frac{\cosh(\sqrt{2}) + \sqrt{2} \sinh(\sqrt{2})}{\sqrt{2} \cosh(\sqrt{2}) + \sinh(\sqrt{2})}.$$

We apply the proposed numerical method with $(N = 1, M = 3)$ to solve this problem. Figure 11 display the exact and approximate solutions for state and control variables with different values for α . Table 52 shows the absolute errors resulting in $\alpha = 1$.

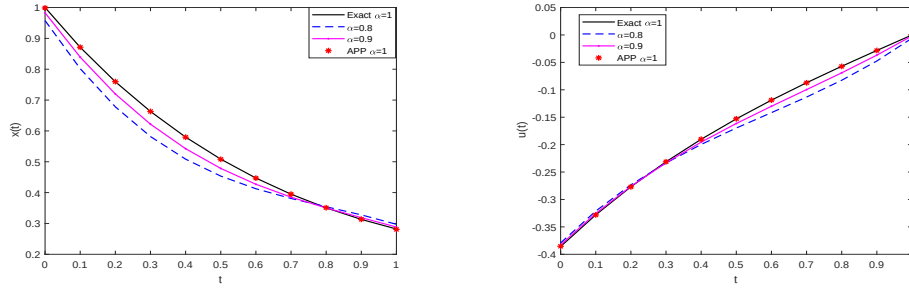


Figure 11: Exact and approximate solutions for state (left) and control (right) variables with different values for α in Example 5.2.

Table 11: Absolute errors for state, control variables and J in $\alpha = 1$ for Example 5.2.

t	$x(t)$	$u(t)$
0	1.3×10^{-3}	3.7709×10^{-4}
0.1	2.8958×10^{-4}	1.1005×10^{-4}
0.2	4.5066×10^{-4}	1.4242×10^{-4}
0.3	4.7553×10^{-5}	8.8760×10^{-6}
0.4	3.6022×10^{-4}	1.1282×10^{-4}
0.5	4.7923×10^{-4}	1.3773×10^{-4}
0.6	2.4878×10^{-4}	5.7246×10^{-5}
0.7	1.8773×10^{-4}	7.5836×10^{-5}
0.8	5.0745×10^{-4}	1.6098×10^{-4}
0.9	2.2903×10^{-4}	6.2862×10^{-5}
1	1.2695×10^{-3}	3.7750×10^{-4}
J	1.5499×10^{-5}	

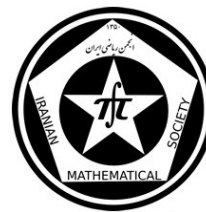
6 Conclusion

In this paper, CFD operator of the Bernoulli hybrid functions is obtained directly and without any approximation. Then, this operator has been used to approximate numerical solution of FOCPs with boundary conditions. The numerical solutions resulted using this method show that the proposed method can solve the FOCPs effectively.



References

- [1] O.P. Agrawal, A general formulation and solution scheme for fractional optimal control problems, *Nonlinear Dyn.*, 38 (1-4) (2004), 323-337.
- [2] R. Dehghan, State Parametrization Method Based on Shifted Legendre Polynomials for Solving Fractional Optimal Control Problems, *Int. J. Appl. Comput. Math.*, (2018) 4: 37.
- [3] A. A. Kilbas, H. M. Srivastava and J. J. Trujillo, *Theory and Applications of Fractional Differential Equations*, Vol. 204 (North-Holland Mathematics Studies), Elsevier Science Inc., New York, NY, 2006.
- [4] S. Mashayekhi and M. Razzaghi, Numerical solution of the fractional Bagley-Torvik equation by using hybrid functions approximation, *Math. Meth. Appl. Sci.* 39, (2015), 353-365.
- [5] J. Mathews and K. Fink (ed.), *Numerical Methods Using Matlab*, Prentice Hall, 1999.
- [6] M. Razzaghi and H. R. Marzban, Direct method for variational problems via hybrid of Block-pulse and Chebyshev functions, *Math. Probl. Eng.*, 6 (2000), 8597.
- [7] S. G. Samko, A. A. Kilbas, and O. I. Marichev, *Fractional Integrals and Derivatives: Theory and Applications*, Switzerland: Gordon and Breach Science Publishers, 1993.
- [8] N.H. Sweilam and T.M. Al-Ajami, Legendre spectral-collocation method for solving some types of fractional optimal control problems, *J. Adv. Res.*, 6 (3), (2015), 393-403.
- [9] C. Viola, *An Introduction to Special Functions*, Springer International Publishing Switzerland, 2016.
- [10] Y. Zhou, *Basic Theory of Fractional Differential Equations*, World Scientific Publishing Co. Pte.Ltd. 2014.



Applying the Meshless Fragile Points Method (FPM) to solve Two Dimensional Linear Schrödinger Equation

Donya Haghighi¹¹, Saeid Abbasbandy

Department of Applied Mathematics, Imam Khomeini International University, Qazvin, Iran

Abstract

In this paper, the meshless Fragile Points Method (FPM) is applied to find the numerical solutions of the Schrödinger equation. This method is based on Galerkin weak-form formulation and the generalized finite difference method has been used to obtain the test and trial functions. For partitioning the problem domain into subdomains, Voronoi Diagram has been applied. These functions are simple, local, and discontinuous polynomials. Because of the discontinuity of test and trial functions, FPM may be inconsistent. To deal with these inconsistencies, we use numerical flux corrections. Finally, numerical results are presented for some examples to demonstrate the accuracy, reliability, and efficiency.

AMS subject Classification 2010: 35J10, 65M99, 65M20, 65N99.

Keywords: Fragile Points Method; Numerical Fluxes; Schrödinger equation; Voronoi Diagram.

1 Introduction

In many problems, numerical analysis researchers use the finite element method (FEM) [1], Finite Volume Method (FVM) [4] and Boundary Element Method (BEM) [7] to discretize the spatial dimension. In these methods, the accuracy of the method may be affected by deforming the elements or meshes. Another group of numerical methods known as meshless methods that do not require mesh for discretization such as Element Free Galerkin (EFG) [3] and Meshless Local Petrov-Galerkin (MLPG) [2] that in these methods the trial and test functions must be continuous. Usually, the test and trial functions in these methods are complicated.

Hence by analyzing and comparing the other numerical methods, Leiting Dong and colleagues provided a general method for solving problems so that the test and trial functions are simple, local, and discontinuous polynomials [5]. This new method is called the Fragile Points Method (FPM), which we will study in this article.

Solving the Schrödinger equation is very important in quantum dynamic calculations and it has received a lot of attention as a model that describes several important chemical and physical phenomena [6]. We consider the two-dimensional time-dependent Schrödinger equation with the following form:

$$-i \frac{\partial u}{\partial t}(\mathbf{x}, t) = \nabla^2 u(\mathbf{x}, t) + w(\mathbf{x})u(\mathbf{x}, t) \quad \mathbf{x} \in \Omega, \quad (184)$$

¹¹Speaker



with initial conditions

$$u(\mathbf{x}, 0) = g(\mathbf{x}), \quad (185)$$

and the boundary conditions

$$u(\mathbf{x}, t) = h_1(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_D, \quad \nabla u \cdot \mathbf{n}(\mathbf{x}, t) = h_2(\mathbf{x}, t), \quad \mathbf{x} \in \Gamma_N. \quad (186)$$

In the above equation, $w(\mathbf{x})$ is an arbitrary potential function.

2 Polynomial discontinuous trial and test functions

In this section, we describe how to obtain local, simple, discontinuous, polynomial, point-based trial and test functions. Inside the domain Ω and its boundary $\partial\Omega$, several points are distributed sporadically. Using these points, the domain is divided into subdomains that have nothing in common and in each subdomain consists of only one point. There are various procedures for the partition of the domain e.g., the Voronoi Diagram partition, quadrilateral and triangular partition (in 2D), tetrahedron and hexahedron partition (in 3D), etc. In this study, the Voronoi Diagram method has been selected. Note that there are no elements or mesh in the present Fragile Points Method for discretization and only random or uniform distributed points on the domain are applied.

The trial function u_h in the subdomain E_0 which includes the point P_0 can be written as

$$u_h(\mathbf{x}, t) = u_0(\mathbf{x}, t) + (\mathbf{x} - \mathbf{x}_0) \nabla u(\mathbf{x}, t)|_{P_0}, \quad \mathbf{x} \in E_0. \quad (187)$$

In the above equation, u_0 is the value of u_h at P_0 and $\mathbf{x}_0$ denotes the coordinate of the point P_0 .

The gradient of ∇u at P_0 is the yet unknown. We employ the Generalized Finite Difference (GFD) method to calculate ∇u at P_0 in terms of the values of u_h at several neighboring points of P_0 . We name these neighboring points as $q_1, q_2, \dots, q_m$. In the following, to calculate the amount of the gradient of ∇u at P_0 , we minimize a weighted discrete L^2 norm $\mathbf{J}$ so that

$$\mathbf{J} = \sum_{i=0}^m \left(\nabla u|_{P_0} \cdot (\mathbf{x}_i - \mathbf{x}_0)^T - (u_i - u_0) \right)^2 w_i, \quad (188)$$

where w_i denotes the value of weight function at q_i , $\mathbf{x}_i$ is the coordinate vector of q_i , and u_i is the value of u_h at q_i , ($i = 1, 2, \dots, m$). For convenience, we assume that w is constant. Due to the stationarity of $\mathbf{J}$ we have

$$\nabla u = (A^T A)^{-1} A^T (\mathbf{u}_m - u_0 \mathbf{I}_m) \quad (189)$$

where

$$A = \begin{bmatrix} x_1 - x_0 & y_1 - y_0 \\ x_2 - x_0 & y_2 - y_0 \\ \dots & \dots \\ x_m - x_0 & y_m - y_0 \end{bmatrix}, \quad \mathbf{u}_m = \begin{bmatrix} u_1 \\ u_2 \\ \dots \\ u_m \end{bmatrix}, \quad \mathbf{I}_m = \begin{bmatrix} 1 \\ 1 \\ \dots \\ 1 \end{bmatrix}_{m \times 1}.$$

Also equation (502) can be expressed at point P_0 as follows:

$$\nabla u = \mathbf{B} \mathbf{u}_E, \quad (190)$$



where

$$\mathbf{B} = (A^T A)^{-1} A^T \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ -1 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -1 & 0 & \dots & 0 & 1 \end{bmatrix}_{m \times (m+1)}, \quad \mathbf{u}_E = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_m \end{bmatrix}.$$

Also by substituting (603) into (611) the relation between u_h and $\mathbf{u}_E$ will be obtained as

$$u_h = \mathbf{N} \mathbf{u}_E, \quad \forall \mathbf{x} \in E_0, \quad \mathbf{N} = [\mathbf{x} - \mathbf{x}_0] \mathbf{B} + [1, 0, \dots, 0]_{1 \times (m+1)}. \quad (191)$$

3 Implementation of numerical flux corrections

We can rewrite Schrödinger equation (689)-(610) using mixed form as following,

$$\begin{cases} \sigma(\mathbf{x}, t) = \nabla u(\mathbf{x}, t), & \text{in } \Omega, \\ -\nabla \cdot \sigma(\mathbf{x}, t) = i \frac{\partial u}{\partial t}(\mathbf{x}, t) + w(\mathbf{x}) u(\mathbf{x}, t), & \text{in } \Omega, \\ u(\mathbf{x}, t) = h_1(\mathbf{x}, t), & \text{in } \Gamma_D, \\ \sigma \cdot \mathbf{n}(\mathbf{x}, t) = h_2(\mathbf{x}, t), & \text{in } \Gamma_N. \end{cases} \quad (192)$$

By multiplying the first and second equations in (536) by test functions τ and ν respectively and integrating it on the subdomain E ,

$$\int_E \sigma_h \cdot \tau d\Omega = \int_E \nabla u_h(\mathbf{x}, t) \cdot \tau d\Omega, \quad (193)$$

$$\int_E -\nabla \cdot \sigma \nu d\Omega = i \int_E \frac{\partial u}{\partial t}(\mathbf{x}, t) \nu d\Omega + \int_E w(\mathbf{x}) u(\mathbf{x}, t) \nu d\Omega \quad (194)$$

using the Green formula and by summing these equations over all subdomains we have

$$\int_{\Omega} \sigma_h \cdot \tau d\Omega = - \int_{\Omega} u_h \nabla \cdot \tau d\Omega + \sum_{E \in \Omega} \int_{\partial E} \hat{u}_h n \cdot \tau d\Gamma, \quad (195)$$

$$\int_{\Omega} \sigma_h \cdot \nabla \nu = \sum_{E \in \Omega} \int_{\partial E} \hat{\sigma}_h \cdot n \nu d\Gamma + i \int_{\Omega} \frac{\partial u}{\partial t}(\mathbf{x}, t) \nu d\Omega + \int_{\Omega} w(\mathbf{x}) u(\mathbf{x}, t) \nu d\Omega. \quad (196)$$

In the above equations values $\hat{\sigma}_h$ and $\hat{u}_h$ represent approximations σ_h and u_h on ∂E . These values are named Numerical Fluxes. To simplify Eqs. (535) and (538), we define operators the *average* and the *jump* which by these operators, we can manage the numerical fluxes. As regards $\Gamma = \Gamma_h + \Gamma_D + \Gamma_N$ (Γ_h is the set of all internal boundaries of subdomains), Table 3.1 in [5] and by substituting the Interior Penalty Numerical Fluxes (IPNF), we have

$$\begin{aligned} & \sum_{E \in \Omega} \int_E \nabla u_h \cdot \nabla \nu d\Omega - \sum_{e \in \Gamma_h \cup \Gamma_D} \int_e (\{\nabla u_h\}[\nu] + \{\nabla \nu\}[u_h]) d\Gamma + \sum_{e \in \Gamma_h \cup \Gamma_D} \frac{\eta}{h_e} \int_e [\nu][u_h] d\Gamma \\ &= \int_{\Omega} w(\mathbf{x}) u(\mathbf{x}, t) \nu d\Omega + \sum_{e \in \Gamma_D} \int_e \left(\frac{\eta}{h_e} \nu - \nabla \nu \cdot \mathbf{n} \right) h_1(\mathbf{x}, t) d\Gamma + \sum_{e \in \Gamma_N} \int_e \nu h_2(\mathbf{x}, t) d\Gamma + i \int_{\Omega} \frac{\partial u}{\partial t}(\mathbf{x}, t) \nu d\Omega. \end{aligned}$$



The above equation is the formula of FPM, which is called FPM-Primal method.

The method matrix form can be expressed as follows:

$$(\mathbf{K} - \mathbf{W})\mathbf{u} - i\mathbf{C}\dot{\mathbf{u}} = \mathbf{F}, \quad (197)$$

Using finite difference schemes, the above equation can be written as follows:

$$(\mathbf{K} - \mathbf{W})(\alpha\mathbf{u}^{n+1} + (1 - \alpha)\mathbf{u}^n) - i\mathbf{C}\left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t}\right) = \mathbf{F}^n. \quad (198)$$

In Equation (198), $\mathbf{u}^k(\mathbf{x}) = \mathbf{u}(\mathbf{x}, k\Delta t)$ that Δt is time step and $0 \leq \alpha \leq 1$. By Substituting values $\mathbf{B}$ instead of $\nabla\nu$ and ∇u , $\mathbf{N}$ instead of u_h and ν in the formula of FPM-Primal, the point stiffness matrices $\mathbf{K}$, $\mathbf{C}$, $\mathbf{W}$ and also the right vector $\mathbf{F}$ will be achieved as follows

$$\mathbf{W} = \int_E \mathbf{N}^T \mathbf{N} w(\mathbf{x}) d\Omega, \quad \mathbf{C} = \int_E \mathbf{N}^T \mathbf{N} d\Omega, \quad \mathbf{K}_E = \int_E \mathbf{B}^T \mathbf{B} d\Omega, \quad E \in \Omega, \quad (199)$$

$$\begin{aligned} \mathbf{K}_h &= \frac{-1}{2} \int_e (\mathbf{B}_1^T \mathbf{n}_1^T \mathbf{N}_1 + \mathbf{N}_1^T \mathbf{n}_1 \mathbf{B}_1) d\Gamma + \frac{\eta}{h_e} \int_e \mathbf{N}_1^T \mathbf{N}_1 d\Gamma \\ &\quad + \frac{-1}{2} \int_e (\mathbf{B}_2^T \mathbf{n}_2^T \mathbf{N}_2 + \mathbf{N}_2^T \mathbf{n}_2 \mathbf{B}_2) d\Gamma + \frac{\eta}{h_e} \int_e \mathbf{N}_2^T \mathbf{N}_2 d\Gamma \\ &\quad + \frac{-1}{2} \int_e (\mathbf{B}_2^T \mathbf{n}_1^T \mathbf{N}_1 + \mathbf{N}_2^T \mathbf{n}_2 \mathbf{B}_1) d\Gamma + \frac{\eta}{h_e} \int_e \mathbf{N}_1^T \mathbf{N}_2 d\Gamma \\ &\quad + \frac{-1}{2} \int_e (\mathbf{B}_1^T \mathbf{n}_2^T \mathbf{N}_2 + \mathbf{N}_1^T \mathbf{n}_1 \mathbf{B}_2) d\Gamma + \frac{\eta}{h_e} \int_e \mathbf{N}_2^T \mathbf{N}_1 d\Gamma, \quad e \in \partial E_1 \cap \partial E_2, \\ \mathbf{K}_D &= - \int_e (\mathbf{B}^T \mathbf{n}^T \mathbf{N} + \mathbf{N}^T \mathbf{n} \mathbf{B}) d\Gamma + \frac{\eta}{h_e} \int_e \mathbf{N}^T \mathbf{N} d\Gamma, \quad e \in \Gamma_D, \end{aligned} \quad (200)$$

and we can also be written

$$\begin{aligned} F_N &= \int_e \mathbf{N}^T h_2(\mathbf{x}, t) d\Gamma, \quad e \in \Gamma_N, \\ F_D &= \int_e \left(\frac{\eta}{h_e} \mathbf{N}^T - \mathbf{B}^T \mathbf{n} \right) h_1(\mathbf{x}, t) d\Gamma, \quad e \in \Gamma_D. \end{aligned} \quad (201)$$

4 Numerical results

In this section, we will study some numerical examples and using the results obtained from the applying of FPM on these examples, the accuracy and stability of the method is investigated. The relative errors used in this section are defined as follows:

$$r_0 = \frac{\|u^h - u\|_{L^2}}{\|u\|_{L^2}}, \quad r_1 = \frac{\|\nabla u^h - \nabla u\|_{L^2}}{\|\nabla u\|_{L^2}}.$$



Table 12: The relative errors of the method for Example 4.1 at $T = 1$ and $\Delta t = 0.01$.

points	Computational parameters	r_0	r_1	cpu time
$N = 121$	$h_e = 0.065, \eta = 1$	7.8263×10^{-3}	2.8068×10^{-2}	1.1s
$N = 676$	$h_e = 2.5, \eta = 1$	9.8615×10^{-4}	5.5903×10^{-3}	11s
$N = 2601$	$h_e = 0.15, \eta = 1$	8.7575×10^{-4}	2.4872×10^{-3}	123s

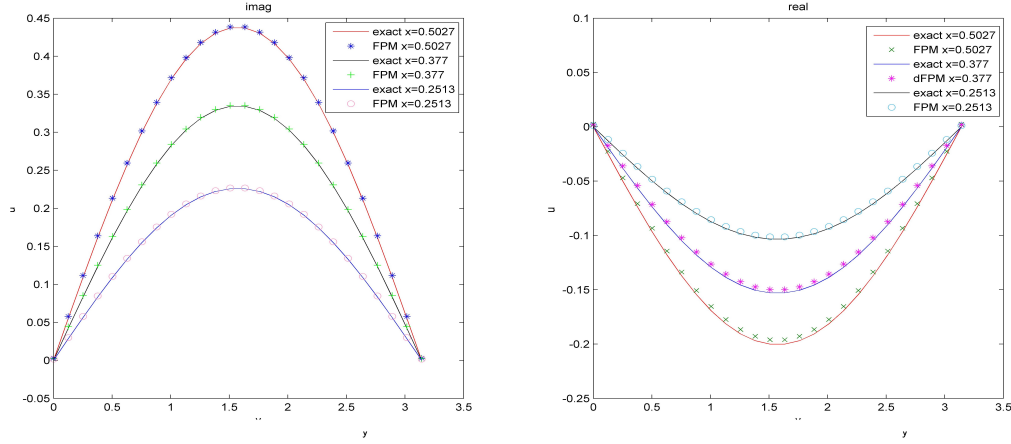


Figure 12: Numerical and exact solutions related to Example 4.2 for $dt = 0.01$, $N = 676$, $T = 2$

Example 0.37. In this example, we consider the time-dependent Schrödinger (689)-(610) in $(x, y) \in \Omega = [0, \pi] \times [0, \pi]$ with initial condition $u(x, y, 0) = \sin(x)\sin(y)$ and Dirichlet boundary conditions which are zero on all sides, with the given potential function as $w(x, y) = 3, (x, y) \in \Omega$. The analytical solution is as $u(x, y, t) = e^{it}\sin(x)\sin(y)$.

As you can see in Table 1, for $\alpha = 0.58$, the method has the appropriate accuracy to solve this example.

Example 0.38. In the previous example, we consider the boundary conditions as

$$\begin{aligned} u(0, y, t) &= 0, \quad \nabla u.n(\pi, y, t) = -e^{it}\sin(y), \\ \nabla u.n(x, 0, t) &= \nabla u.n(x, \pi, t) = -e^{it}\sin(x). \end{aligned}$$

The figure 1 shows the accuracy of FPM for this example for $N = 676$, $T = 2s$, $dt = 0.01$, and $\alpha = 0.58$.

Example 0.39. In this example, we consider Equation (689) as follows:

$$-i\frac{\partial u}{\partial t}(\mathbf{x}, t) = \nabla^2 u(\mathbf{x}, t) + (3 - 2\tanh^2(x) - 2\tanh^2(y))u(\mathbf{x}, t) \quad \mathbf{x} \in \Omega = [0, 1] \times [0, 1],$$

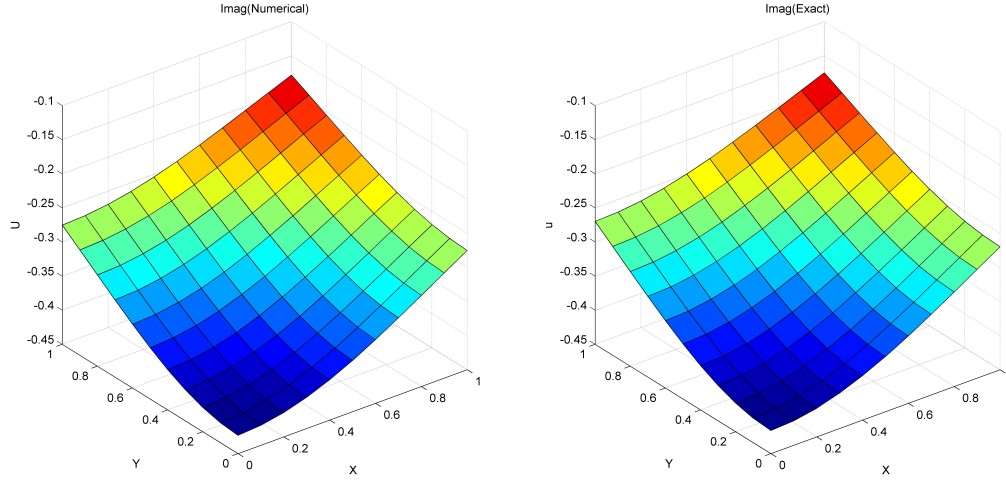


Figure 13: Plot of imaginary part numerical and exact solutions for Example 4.3 so that $N = 121$, $dt = 0.01$, $T = 1$.

that the analytical solution of this equation is $u(x, y, t) = \frac{ie^{it}}{\cosh(x)\cosh(y)}$. if we set all of the boundary conditions as Neumann, we have

$$\begin{aligned}\nabla u(0, y, t) \cdot n &= \nabla u(x, 0, t) \cdot n = 0, & 0 \leq x, y \leq 1, \\ \nabla u(1, y, t) \cdot n &= -ie^{it} \tanh(1) \operatorname{sech}(1) \operatorname{sech}(y), & 0 \leq y \leq 1, \\ \nabla u(x, 1, t) \cdot n &= -ie^{it} \tanh(1) \operatorname{sech}(1) \operatorname{sech}(x), & 0 \leq x \leq 1,\end{aligned}$$

with the initial condition $u(x, y, t) = \frac{i}{\cosh(x)\cosh(y)}$, $0 \leq x, y \leq 1$. In the curve shown in Figure 2, we set the values $N = 121$, $T = 2$ and $dt = 0.01$ and the numerical and exact solutions related to the imaginary part of the solutions are drawn.

According to the results of the table and comparison of the curves obtained by FPM with the exact curves, we can be seen that the method is stable and has good precision. Also, the method does not have much computational cost and depending on the number of points used, it will achieve numerical solutions with good accuracy in a short time that this is an advantage over finite element. Other advantages of this method over other numerical methods are described in detail in Table 1 in [8].

References

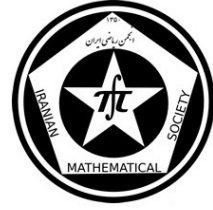
- [1] M. Asadzadeh, An introduction to the finite element method (FEM) for differential equations, Chalmers: Lecture notes. 2010.
- [2] S.N. Atluri, T. Zhu, A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics, Comput. Mech. 22(2) (1998) 117-127.



- [3] T. Belytschko, Y.Y. Lu, L. Gu, Element Free Galerkin Methods, Int. J. Numer. Methods. Eng. 37(2) (1994) 229-256.
- [4] J.C. Chai, H.S. Lee, S.V. Patankar, Finite volume method for radiation heat transfer, J. Thermophys. Heat. Trans. 8(3) (1994) 419-425.
- [5] L. Dong, T. Yang, K. Wang, S.N. Atluri, A new Fragile Points Method (FPM) in computational mechanics, based on the concepts of Point Stiffnesses and Numerical Flux Corrections, Eng. Anal. Bound. Elem. 107 (2019) 124-133.
- [6] M. Subai, On the finite differences schemes for the numerical solution of two dimensional Schrödinger equation, Numer. Meth. for Partial Differential Equations: An Int. J 18.6 (2002): 752-758.
- [7] L.C. Wrobel, The Boundary Element Method, Volume 1: Applications in Thermo-Fluids and Acoustics. Vol. 1. John Wiley & Sons. 2002.
- [8] T. Yang, L. Dong, S.N. Atluri, A simple Galerkin meshless method, the Fragile Points method using point stiffness matrices, for 2D linear elastic problems in complex domains with crack and rupture propagation, Int. J. Numer. Meth. Eng.

E-mail 1: *Haghighi.donya@edu.ikiu.ac.ir*

E-mail 2: *Abbasbandy@ikiu.ac.ir*



CGNE-like iterative method for nearness problem of generalized Sylvester-transpose matrix equations

Mohammad Mahdi Izadkhah^{†12}

[†]Department of Computer Science, Faculty of Computer and Industrial Engineering,
Birjand University of Technology, Birjand, Iran, and Email address: izadkhah@birjandut.ac.ir

Abstract

In the present paper, we propose a CGNE-like iterative method for solving nearness problem of generalized Sylvester-transpose matrix equations. We first consider CGNE-like iterative method for generalized Sylvester-transpose matrix equations. Then, some theoretical results are established for the residual and direction matrices of the introduced method. Next, We investigate nearness problem of generalized Sylvester-transpose matrix equations. Numerical illustrations will also be given to show the effectiveness and accuracy of the proposed CGNE-like method in comparison with other ones.

AMS subject Classification 2010: 65F10, 15A24, 15A09.

Keywords: Sylvester-transpose matrix equations, least Frobenius norm solution, CGNE method.

1 Introduction

Consider a generalized Sylvester-transpose matrix equation of the following form

$$AXB + CX^T D = F, \quad (202)$$

where $A \in \mathbb{R}^{r \times m}$, $B \in \mathbb{R}^{n \times s}$, $C \in \mathbb{R}^{r \times n}$, $D \in \mathbb{R}^{m \times s}$, and $F \in \mathbb{R}^{r \times s}$ are the given matrices and $X \in \mathbb{R}^{m \times n}$ is the unknown matrix to be determined. In the case that $r = s = n$ and $B = C = I_n$, the matrix equation (202) reduces to the Sylvester-transpose matrix equation of the form $AX + X^T D = F$, where I_n stands for the identity matrix of order n . For two matrices A and B in $\mathbb{R}^{m \times n}$, the inner product $\langle A, B \rangle_F$ is defined as $\langle A, B \rangle_F = \text{trace}(B^T A)$, and the associated norm is the Frobenius norm denoted by $\|\cdot\|$. For the vector $\mathbf{x} = (x_i) \in \mathbb{R}^n$, we consider Euclidean norm as $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^n |x_i|^2}$. If the matrix equation (202) is consistent, then for any initial matrix $X^{(0)}$, a solution can be obtained within finite iteration steps of our method, in the absence of round-off errors. Here, we also consider the solution of the matrix nearness problem

$$\min_{X \in \mathcal{S}_X} \|X - \tilde{X}\|^2, \quad (203)$$

¹²Speaker



where $\tilde{X} \in \mathbb{R}^{m \times n}$ is a given matrix, and $\mathcal{S}_X$ is the solution set of the linear matrix equation (202). The best estimation $\hat{X}$ is the matrix that is not only a solution of the linear matrix equation (202), but also is the best approximation of the matrix $\tilde{X}$ [3]. Problems (202) and (203) frequently arise from applications in control and system theory, stability, analysis of bilinear systems and, signal and image processing [1, 6]. Salkuyeh and Beik considered the problem of finding the minimum norm least-squares solution of a quite general class of coupled linear matrix equations defined over field of complex numbers, and they examined a gradient-based approach with the convergence properties of their proposed algorithm [5]. The aim of this paper is to present CGNE-like iterative method for solving Problems (202) and (203).

2 CGNE-like method and theoretical results

In this section, we propose an efficient iterative algorithm based on the CGNE method, to solve generalized Sylvester-transpose matrix equation (202). The proposed algorithm is described in Algorithm .

CGNE-like iterative method

1. Choose $X^{(0)} \in \mathbb{R}^{m \times n}$, and tolerance ϵ .
 2. Compute $R^{(0)} = F - AX^{(0)}B - CX^{(0)T}D$ and $P^{(0)} = A^T R^{(0)} B^T + DR^{(0)T} C$.
 3. **For** $k = 0, 1, \dots$ until convergence, **Do**
 4. $X^{(k+1)} = X^{(k)} + \frac{\|R^{(k)}\|^2}{\|P^{(k)}\|^2} P^{(k)}$
 5. $R^{(k+1)} = R^{(k)} - \frac{\|R^{(k)}\|^2}{\|P^{(k)}\|^2} (AP^{(k)}B + CP^{(k)T}D)$
 6. $P^{(k+1)} = A^T R^{(k+1)} B^T + DR^{(k+1)T} C + \frac{\|R^{(k+1)}\|^2}{\|R^{(k)}\|^2} P^{(k)}$
 7. **If** $\frac{\|R^{(k+1)}\|}{\|R^{(0)}\|} < \epsilon$ **then STOP**
 8. **EndDo**
-

Algorithm possesses the following properties. The proof of the following Lemma is motivated from Lemma 2.1 in [4].

Lemma 0.40. *Assume that the linear matrix equation (202) is consistent, and X^* is its solution, then for any initial matrix $X^{(0)}$, the sequences $X^{(k)}, P^{(k)}$ and $R^{(k)}$ generated by Algorithm satisfy*

$$\langle P^{(k)}, X^* - X^{(k)} \rangle_F = \|R^{(k)}\|^2, \quad k = 0, 1, 2, \dots$$

Remark 0.41. From the formula of $P^{(k)}$ in Algorithm and Lemma 0.113, we know that if the linear matrix equation (202) is consistent, then $R^{(k)} = 0$ if and only if $P^{(k)} = 0$. So, $\|R^{(k)}\| \neq 0$ if and only if $\|P^{(k)}\| \neq 0$. This result implies that if there exist a positive number k such that $P^{(k)} = 0$ but $R^{(k)} \neq 0$, then the linear matrix equation (202) is inconsistent. This means that we can solve the linear matrix equation (202) in the absence of round-off errors.

Lemma 0.42. *Assume that the linear matrix equation (202) is consistent and the sequences $R^{(i)}$ and $P^{(i)}$, where $R^{(i)} \neq 0 (i = 0, 1, 2, \dots, k)$, are generated by Algorithm . Then*

$$\langle R^{(i)}, R^{(j)} \rangle_F = 0, \quad \langle P^{(i)}, P^{(j)} \rangle_F = 0, \quad (i, j = 0, 1, 2, \dots, k, \quad i \neq j).$$



Remark 0.43. Since $R^{(0)}, R^{(1)}, R^{(2)}, \dots$ are orthogonal matrices in the finite dimension matrix space $\mathbb{R}^{r \times s}$, so we achieve the exact solution at most rs iteration steps. In fact, we have $R^{(k)} = 0$, for some $k \leq rs$.

3 Investigation of the nearness problem

To facilitate the statement of the nearness problem (203), we recall the following theorems.

Theorem 0.44 ([2]). *Suppose that $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ and $\mathcal{X} = \{x \in \mathbb{R}^n | x = \operatorname{argmin}_{y \in \mathbb{R}^n} \|Ay - b\|_2\}$, then, $x \in \mathcal{X}$ if and only if $A^T A x = A^T b$. Moreover, $x^* = A^\dagger b$ is the unique solution of the problem $\min_{x \in \mathcal{X}} \|x\|_2$, where $A^\dagger$ is the MoorePenrose generalized inverse(pseudoinverse) of A .*

Theorem 0.45 ([4]). *Let $A \in \mathbb{R}^{m \times n}$ and $b \in \operatorname{Range}(A)$. Suppose that the system of linear equations $Ax = b$ has a solution $x^* \in \operatorname{Range}(A^T)$. Then x^* is the unique least Euclidean norm solution of the system of linear equations, e.i., $x^* = A^\dagger b$.*

For $A \in \mathbb{R}^{r \times m}$, $B \in \mathbb{R}^{n \times s}$, $C \in \mathbb{R}^{r \times n}$, $D = [d_1, d_2, \dots, d_s] \in \mathbb{R}^{m \times s}$, let us assume that $\mathcal{M} = B^T \otimes A + \begin{pmatrix} C \otimes d_1^T \\ C \otimes d_2^T \\ \vdots \\ C \otimes d_s^T \end{pmatrix}$, where $A \otimes B$ is the Kronecker product of the matrices A and B , that is specified as the

$rn \times ms$ matrix $A \otimes B = [a_{ij}B]$. Straightforward computation shows that the Sylvester-transpose matrix equation (202) is equivalent to the following linear system of equations $\mathcal{M}\operatorname{vec}(X) = \operatorname{vec}(F)$, where the 'vec($\cdot$)' operator, transforms a matrix A of size $r \times m$ to a vector $a = \operatorname{vec}(A)$ of size $rm \times 1$ by stacking the columns of A . Furthermore, for any vector $\mathbf{x}$ of size $mn \times 1$, we use notation $\widetilde{\operatorname{vec}}_{m,n}(\mathbf{x})$ for a matrix of size $m \times n$ by separating all entries of the vector $\mathbf{x}$ as $\widetilde{\operatorname{vec}}_{m,n}(\mathbf{x}) = [\mathbf{x}(1:m), \mathbf{x}(m+1:2m), \dots, \mathbf{x}((n-1)m+1:mn)]$, where $\mathbf{x}(i:j)$ denotes the elements of the positions i -th to j -th of the vector $\mathbf{x}$. Obviously, for arbitrary matrix $H \in \mathbb{R}^{r \times s}$ and $C = [c_1, c_2, \dots, c_n] \in \mathbb{R}^{r \times n}$, we have

$$\begin{aligned} \operatorname{vec}(A^T H B^T + D H^T C) &= \left[B \otimes A^T + \begin{pmatrix} D \otimes c_1^T \\ D \otimes c_2^T \\ \vdots \\ D \otimes c_n^T \end{pmatrix} \right] \operatorname{vec}(H), \\ &= \mathcal{M}^T \operatorname{vec}(H) \in \operatorname{Range}(\mathcal{M}^T). \end{aligned}$$

We see that if we take the matrix $X^{(0)} = A^T H B^T + D H^T C$ as an initial guess, where H is arbitrary, then all $X^{(k)}$ generated by Algorithm satisfy $\operatorname{vec}(X^{(k)}) \in \operatorname{Range}(\mathcal{M}^T)$. From Theorem 0.111, we understand that for $\mathbf{w} = \mathcal{M}^\dagger \operatorname{vec}(F)$ we have

$$X^* = \widetilde{\operatorname{vec}}_{m,n}(\mathbf{w}). \quad (204)$$

So, we summarize the above statements, in the following theorem, which its proof is omitted. It follows immediately from the definitions of Frobenius norm and Euclidean norm that

$$\|F - AXB - CX^T D\| = \|\operatorname{vec}(F) - \mathcal{M}\operatorname{vec}(X)\|_2. \quad (205)$$



Theorem 0.46. Suppose that the generalized Sylvester-transform matrix equation (202) is consistent, then for any initial guess matrix $X^{(0)}$, the matrix sequences $X^{(k)}$, generated by Algorithm , converges to the exact solution within at most rs iteration steps. Furthermore, if we choose the initial guess matrix $X^{(0)} = A^T H B^T + D H^T C$, where H is arbitrary, then according to (205), the exact solution of the Algorithm is the least Frobenius norm solution of the problem (203), which can be expressed as (204).

When the linear matrix equation (202) is consistent, the solution set $\mathcal{S}_X$ is non-empty, then for $X \in \mathcal{S}_X$ and a given matrix $\tilde{X}$ we have

$$A X B + C X^T D = F \Leftrightarrow A(X - \tilde{X})B + C(X - \tilde{X})^T D = F - A\tilde{X}B + C\tilde{X}^T D.$$

Let $\bar{X} = X - \tilde{X}$ and $\bar{F} = F - A\tilde{X}B - C\tilde{X}^T D$, then the nearness problem (203) is equivalent to finding the least Frobenius norm solution $\bar{X}^*$ of the generalized Sylvester-transpose matrix equation

$$A\bar{X}B + C\bar{X}^T D = \bar{F}. \quad (206)$$

Therefore, by using Algorithm , we obtain an approximate solution $\bar{X}^{(N)}$ for (206). Then according to Theorem 0.110, the unique solution $\hat{X}$ of the nearness problem (203) can be approximated as

$$\hat{X}^{(N)} = \bar{X}^{(N)} + \tilde{X}.$$

4 Numerical illustrations

In this section, a numerical example is presented to illustrate the effectiveness of the proposed CGNE-like method to solve (202) and (203) compared with accelerated gradient based iterative algorithm (AGB) [1] and modified gradient based (MGB) algorithm [6]. All results were computed with MATLAB 8.5. Initial guess for the presented method is zero matrix, and the method is terminated if the prescribed iteration number $k_{\max} = 2000$ is exceeded, or the current iterations satisfy $ERR = \frac{\|R^{(k)}\|}{\|R^{(0)}\|} \leq 10^{-6}$.

Example 0.47. We consider generalized Sylvester-transpose matrix equation (202), where

$$A = \begin{pmatrix} 4 & -1 & \cdots & -1 \\ -1 & 4 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ -1 & \cdots & -1 & 4 \end{pmatrix} \in \mathbb{R}^{n \times n}, \quad C = \begin{pmatrix} 8 & -2 & \cdots & -2 \\ -2 & 8 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ -2 & \cdots & -2 & 8 \end{pmatrix} \in \mathbb{R}^{n \times n},$$

and, $B = T_{2,n}$ and $D = T_{3,n}$, where $T_{d,n} = \text{tridiag}\left(-1 + \frac{10}{n+1}, d, -1 + \frac{10}{n+1}\right) \in \mathbb{R}^{n \times n}$. The matrix F was taken such that $X = \text{tridiag}(-1, 2, 1)$ is the exact solution of (202). The numerical results are given in Table 54. In this table, IT and CPU stand for the number of iteration and the elapsed time of computation, respectively. Also dagger ($\dagger$) means that the method is not achieved to the stop criterion in 2000 iterations. Also the comparative results are depicted in Figure 42. Furthermore, we let $\tilde{X} = \text{tridiag}(1, 1, 1) \in \mathbb{R}^{n \times n}$ in nearness problem (203), and use CGNE-like method for computing best approximate solution with $H = \text{tridiag}(1, 2, 3) \in \mathbb{R}^{n \times n}$ for initial guess. The results are presented in Table 39.



Table 13: Performance of CGNE-like method compared with AGB[1] and MGB[6] for different n in Example 0.106

Method	n	5	8	20	80
CGNE-like	IT	22	26	63	108
	CPU	0.0003	0.0004	0.0018	0.0294
	ERR	7.3194e-07	7.955e-07	9.8282e-07	9.1239e-07
AGB	IT	1852	†	†	†
	CPU	0.0376	0.0420	0.0879	1.0561
	ERR	9.95583e-07	9.51277e-05	5.35573e-02	3.13055e-01
MGB	IT	927	1760	†	†
	CPU	0.0189	0.0364	0.0841	1.1117
	ERR	9.93280e-07	9.975763e-07	1.99271e-02	2.24993e-01

Table 14: Performance of CGNE-like method for Example 0.106 compared with AGB[1] and MGB[6] for nearness problem (203)

	Method	IT	CPU-time	$\min_{X \in \mathcal{S}_X} \ X - \tilde{X}\ ^2$
$n = 50$	CGNE-like	84	0.0143	2.2136e+01
	AGB	†	0.5025	2.0818e+02
	MGB	†	0.4978	1.5614e+02
$n = 100$	CGNE-like	95	0.0574	3.1464e+01
	AGB	†	2.1166	4.0721e+02
	MGB	†	2.1053	3.4788e+02

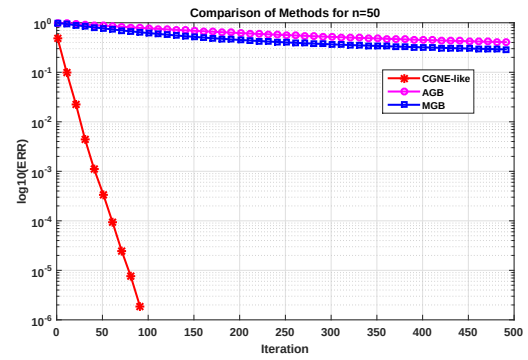
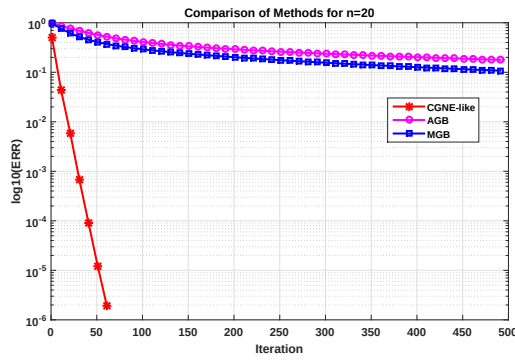


Figure 14: Convergence curves of GCNE-like, AGB and MGB for Example 0.106 with $n = 20$ (left) and $n = 50$ (right)

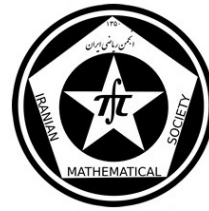


5 Conclusion

We presented a CGNE-like iterative method to solve generalized Sylvester-transpose matrix equations and the corresponding nearness problem. Numerical experiments indicate that CGNE-like method gives better number of iteration and CPU-time than accelerated gradient based (AGB) and modified gradient based (MGB) iterative algorithms.

References

- [1] A.M.E. Bayoumia, M.A. Ramadanb, M. Nili Ahmadabadi, An accelerated gradient based iterative algorithm for solving systems of coupled generalized Sylvester-transpose matrix equations, *J. Linear Topol. Algebra*, 8 (2019), No. 2, 117–126.
- [2] A. Berman and R.J. Plemmons, *Nonnegative Matrices in the Mathematical Sciences*, SIAM, Philadelphia, 1994.
- [3] N.J. Higham, Computing a nearest symmetric positive semidenite matrix. *Linear Algebra Appl.*, 103 (1988), 103–118.
- [4] Z.-Y. Peng, and Y.-X. Peng, An efficient iterative method for solving the matrix equation $AXB + CYD = E$, *Numer. Linear Algebra Appl.* 13 (2006), 473–485.
- [5] D.K. Salkuyeh, F.P.A. Beik, Minimum norm least-lquares solution to general complex coupled linear matrix equations via iteration, *Filomat*, 29:6 (2015), 1389–1407.
- [6] K. Wang, Z. Liu, C. Xu, A modified gradient based algorithm for solving matrix equations $AXB + CX^T D = F$, *J. Appl. Math.*, vol. 2014, Article ID 954523, 6 pages, 2014.



Compact finite difference method to solve the problem of electrohydrodynamic flow of a fluid in an ion drag configuration in a circular cylindrical conduct

Azam Sadat Hashemi¹³, Ghasem Barid Loghmani, Mohammad Heydari

Department of Mathematics, Yazd University, Yazd, Iran

as.hashemi@stu.yazd.ac.ir, loghmani@yazd.ac.ir, m.heydari@yazd.ac.ir

Abstract

In this article, we focus on finding the numerical solution of the nonlinear singular second order boundary value problem for the electrohydrodynamic flow of a fluid in an ion drag configuration in a circular cylindrical conduit. For this purpose, the quasilinearization method combined with the fourth order compact finite difference scheme is applied to solve the equation numerically. AMS subject Classification 2010: 65L12, 34B16, 76E25.

Keywords: Electrohydrodynamic flow, Nonlinear singular boundary value problem, Compact finite difference method, Quasilinearization method.

1 Introduction

The effect of electric and magnetic fields on the flow of liquids is included in electrohydrodynamics (EHD) and magnetohydrodynamics (MHD). EHD flow of a fluid in an "ion drag" configuration in a circular cylindrical conduit is described by a set of partial differential equations which can be reduced to a single ordinary differential equation in the stationary and radially symmetric case [6]. Due to the singularity, many researchers have become interested in solving this problem. Recently, Roul et al. [7, 8] obtained their approximate solutions by a sixth-order numerical method and an iterative algorithm.

Following [8], we consider the nonlinear singular boundary value problem (SBVP)

$$(xy'(x))' = -xH^2 \left(1 - \frac{y(x)}{1 - \alpha y(x)} \right), \quad (207)$$

$$y'(0) = 0, \quad y(1) = 0, \quad (208)$$

where $y(x)$, x and H denote the fluid velocity, the radial distance from the center of the cylindrical conduit and Hartmann electric number, respectively. The nonlinear parameter α is a measure of the strength of the nonlinearity. The singularity arises when $x = 0$ and $y(x) = \frac{1}{\alpha}$.

¹³Speaker



2 Linearization method

In this section, we describe the quasilinearization method developed by Bellman and Kalaba [1, 3]. This technique linearizes the nonlinear boundary value problems and provides a sequence of linear differential equations. To do this, consider the nonlinear boundary value problem

$$y'' = f(x, y, y'), \quad x \in [a, b], \quad (209)$$

$$y'(a) = c_0, \quad y(b) = c_1, \quad (210)$$

where c_0 and c_1 are real constants. Employing the quasilinearization method for nonlinear BVP (655) and boundary conditions (457), we obtain the following sequence of linear differential equations obtained,

$$y''_{n+1} = f(x, y_n, y'_n) + (y_{n+1} - y_n)f_y(x, y_n, y'_n) + (y'_{n+1} - y'_n)f_{y'}(x, y_n, y'_n), \quad (211)$$

with boundary conditions

$$y'_{n+1}(a) = c_0, \quad y_{n+1}(b) = c_1. \quad (212)$$

Solving the recurrent relation (610) yields the solution of the original nonlinear equation (655)[1].

Theorem 0.48. [5] Suppose that $\delta y_{n+1} = y_{n+1} - y_n$ denotes the difference between two subsequence iterations in (610)-(394). Then the quasilinearization method is convergent quadratically, i.e.

$$\|\delta y_{n+1}\| \leq C \|\delta y_n\|^2,$$

where C is a positive constant.

3 Compact finite difference method

In this work, the discretization of the equation (689) will be performed according to the compact finite difference method (CFDM) proposed by Lele[4] which is a linear cell-centered compact scheme. For the first order derivative, this scheme is given by

$$\hat{\beta}y'_{i-2} + \hat{\alpha}y'_{i-1} + y'_i + \hat{\alpha}y'_{i+1} + \hat{\beta}y'_{i+2} = \hat{a}\frac{y_{i+1} - y_{i-1}}{2h} + \hat{b}\frac{y_{i+2} - y_{i-2}}{4h} + \hat{c}\frac{y_{i+3} - y_{i-3}}{6h}. \quad (213)$$

This formula contains the spatial derivatives y'_i and the function values y_i at the grid points x_i , $i = 0, 1, \dots, N$ where $N \in \mathbb{N}$ and $h = (b - a)/N$. The relationships between the coefficients $\hat{\alpha}$, $\hat{\beta}$, $\hat{a}$, $\hat{b}$ and $\hat{c}$ are derived by matching the Taylor series coefficients of various orders. In this way, a one-parameter family of fourth order schemes is obtained by choosing $\hat{\beta} = \hat{c} = 0$. This family has

$$\hat{\beta} = 0, \quad \hat{c} = 0, \quad \hat{a} = \frac{2}{3}(\hat{\alpha} + 2), \quad \hat{b} = \frac{1}{3}(4\hat{\alpha} - 1),$$

where the truncation error is $T_1 = \frac{4}{5!}(3\hat{\alpha} - 1)h^4y^{(5)}$. Furthermore, for $\hat{\alpha} = \frac{1}{4}$, the standard Padé method is recovered.

A similar relation is provided for the second order derivatives as follows

$$\begin{aligned} \tilde{\beta}y''_{i-2} + \tilde{\alpha}y''_{i-1} + y''_i + \tilde{\alpha}y''_{i+1} + \tilde{\beta}y''_{i+2} = & \tilde{a}\frac{y_{i+1} - 2y_i + y_{i-1}}{h^2} + \tilde{b}\frac{y_{i+2} - 2y_i + y_{i-2}}{4h^2} \\ & + \tilde{c}\frac{y_{i+3} - 2y_i + y_{i-3}}{9h^2}, \end{aligned} \quad (214)$$



where the unknown coefficients $\tilde{\alpha}$, $\tilde{\beta}$, $\tilde{a}$, $\tilde{b}$ and $\tilde{c}$ are obtained in the same manner as the first derivative. A One-parameter family of fourth order schemes has

$$\tilde{\beta} = 0, \quad \tilde{c} = 0, \quad \tilde{a} = \frac{4}{3}(1 - \tilde{\alpha}), \quad \tilde{b} = \frac{1}{3}(10\tilde{\alpha} - 1),$$

where the truncation error is $T_2 = -\frac{4}{6!}(11\tilde{\alpha} - 1)h^4y^{(6)}$. For $\tilde{\alpha} = \frac{1}{10}$, $\tilde{b}$ vanishes and the classical Padé scheme is obtained.

4 Method description

The aim of this section is to solve the singular nonlinear BVP (689)-(609) via a combination of the methods described in the two previous sections. For this purpose, we suppose that equation (689) has the following form

$$y''(x) + \frac{1}{x}y'(x) = f(x, y(x)), \quad x \in [0, 1], \quad (215)$$

where $f(x, y) = -H^2 \left(1 - \frac{y}{1-\alpha y}\right)$. Our scheme to compute the numerical solution of the equation (604) will be accomplished in two steps. At first, we apply the quasilinearization method to obtain the following linear structure

$$y''_{n+1}(x) + \frac{1}{x}y'_{n+1}(x) - f_y(x, y_n(x))y_{n+1}(x) = f(x, y_n(x)) - f_y(x, y_n(x))y_n(x), \quad (216)$$

$$y'_{n+1}(0) = 0, \quad y_{n+1}(1) = 0. \quad (217)$$

Then, using the fourth order CFDM formulas (611)-(612) with $\hat{\beta} = \hat{c} = 0$, $\tilde{\beta} = \tilde{c} = 0$ for the first and second derivatives, respectively, we present a discrete system for solving linear differential equation (536) as follows:

$$\begin{cases} M_1 Y' = \frac{1}{h} A_1 Y + \frac{1}{h} B_1, \\ M_2 Y'' = \frac{1}{h^2} A_2 Y + \frac{1}{h^2} B_2, \\ Y'' + B_3 Y' + C_3 Y = C, \end{cases} \quad (218)$$

where B_3 and C_3 are diagonal matrices and M_1 and M_2 are tridiagonal invertible matrices if $\hat{\alpha}$ and $\tilde{\alpha}$ don't equal to 1. Furthermore, A_1 and A_2 are tridiagonal matrices but their first and last rows may be recalculated due to boundary conditions (457) to meet the order of the accuracy. In this way, to have the fourth order accuracy, the first and last rows of Matrix A_1 and A_2 have the following relations

$$\begin{cases} y'_1 + \hat{\alpha}y'_2 = \frac{1}{h} (\hat{a}_0y_0 + \hat{a}_1y_1 + \hat{a}_2y_2 + \hat{a}_3y_3 + \hat{a}_4y_4), \\ \hat{\alpha}y'_{N-2} + y'_{N-1} = \frac{1}{h} (\hat{b}_0y_{N-4} + \hat{b}_1y_{N-3} + \hat{b}_2y_{N-2} + \hat{b}_3y_{N-1} + \hat{b}_4y_N), \end{cases} \quad (219)$$

$$\begin{cases} y''_1 + \tilde{\alpha}y''_2 = \frac{1}{h^2} (\tilde{a}_0y_0 + \tilde{a}_1y_1 + \tilde{a}_2y_2 + \tilde{a}_3y_3 + \tilde{a}_4y_4 + \tilde{a}_5y_5), \\ \tilde{\alpha}y''_{N-2} + y''_{N-1} = \frac{1}{h^2} (\tilde{a}_5y_{N-5} + \tilde{a}_4y_{N-4} + \tilde{a}_3y_{N-3} + \tilde{a}_2y_{N-2} + \tilde{a}_1y_{N-1} + \tilde{a}_0y_N), \end{cases} \quad (220)$$

where

$$\begin{cases} \hat{a}_0 = \frac{1}{12}\hat{\alpha} - \frac{1}{4}, \hat{a}_1 = -\frac{2}{3}\hat{\alpha} - \frac{5}{6}, \hat{a}_2 = \frac{3}{2}, \hat{a}_3 = \frac{2}{3}\hat{\alpha} - \frac{1}{2}, \hat{a}_4 = -\frac{1}{12}\hat{\alpha} + \frac{1}{12}, \\ \hat{b}_0 = \frac{1}{12}\hat{\alpha} - \frac{1}{12}, \hat{b}_1 = -\frac{2}{3}\hat{\alpha} + \frac{1}{2}, \hat{b}_2 = -\frac{3}{2}, \hat{b}_3 = \frac{2}{3}\hat{\alpha} + \frac{5}{6}, \hat{b}_4 = -\frac{1}{12}\hat{\alpha} + \frac{1}{4}, \\ \tilde{a}_0 = \frac{-1}{12}\tilde{\alpha} + \frac{5}{6}, \tilde{a}_1 = \frac{4}{3}\tilde{\alpha} - \frac{5}{4}, \tilde{a}_2 = \frac{-5}{2}\tilde{\alpha} - \frac{1}{3}, \tilde{a}_3 = \frac{4}{3}\tilde{\alpha} + \frac{7}{6}, \tilde{a}_4 = \frac{-1}{12}\tilde{\alpha} - \frac{1}{12}, \tilde{a}_5 = \frac{1}{12}. \end{cases} \quad (221)$$



Other rows of A_1 and A_2 are calculated by relations (611) and (612), respectively. So, we have:

$$M_1 = \begin{bmatrix} 1 & \hat{\alpha} & 0 & \cdots & 0 \\ \hat{\alpha} & 1 & \hat{\alpha} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \hat{\alpha} & 1 & \hat{\alpha} \\ 0 & \cdots & 0 & \hat{\alpha} & 1 \end{bmatrix}, \quad A_1 = \frac{1}{h} \begin{bmatrix} \hat{a}_1 & \hat{a}_2 & \hat{a}_3 & \hat{a}_3 & \cdots & 0 \\ -\frac{\hat{a}}{2} & 0 & \frac{\hat{a}}{2} & \frac{\hat{b}}{4} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -\frac{\hat{b}}{4} & -\frac{\hat{a}}{2} & 0 & \frac{\hat{a}}{2} \\ 0 & \cdots & \hat{b}_0 & \hat{b}_1 & \hat{b}_2 & \hat{b}_3 \end{bmatrix},$$

$$M_2 = \begin{bmatrix} 1 & \tilde{\alpha} & 0 & \cdots & 0 \\ \tilde{\alpha} & 1 & \tilde{\alpha} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \tilde{\alpha} & 1 & \tilde{\alpha} \\ 0 & \cdots & 0 & \tilde{\alpha} & 1 \end{bmatrix}, \quad A_2 = \frac{1}{h^2} \begin{bmatrix} \tilde{a}_1 & \tilde{a}_2 & \tilde{a}_3 & \tilde{a}_4 & \tilde{a}_5 & \cdots & 0 \\ \tilde{a} & -2(\tilde{a} + \frac{\tilde{b}}{4}) & \tilde{a} & \frac{\tilde{b}}{4} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \frac{\tilde{b}}{4} & \tilde{a} & -2(\tilde{a} + \frac{\tilde{b}}{4}) & \tilde{a} \\ 0 & \cdots & \tilde{a}_5 & \tilde{a}_4 & \tilde{a}_3 & \tilde{a}_2 & \tilde{a}_1 \end{bmatrix}.$$

According to boundary condition $y'(0) = 0$, we approximate y_0 by the following equations

$$y_0 = hy'_0 + \hat{\beta}_1 y_1 + \hat{\beta}_2 y_2 + \hat{\beta}_3 y_3 + \hat{\beta}_4 y_4 + \hat{\beta}_5 y_5, \quad (222)$$

$$y_0 = hy'_0 + \tilde{\beta}_1 y_1 + \tilde{\beta}_2 y_2 + \tilde{\beta}_3 y_3 + \tilde{\beta}_4 y_4 + \tilde{\beta}_5 y_5 + \tilde{\beta}_6 y_6. \quad (223)$$

So, two first rows of A_1 and A_2 will be updated by formulas (222) and (223), respectively. Hence, the vector of unknowns $Y = [y_1, \dots, y_{N-1}]^T$ is calculated by solving the following linear system

$$\left(\frac{1}{h^2} M_2^{-1} A_2 + \frac{1}{h} B_3 M_1^{-1} A_1 + C_3 \right) Y = C - \frac{1}{h^2} M_2^{-1} B_2 - \frac{1}{h} B_3 M_1^{-1} B_1. \quad (224)$$

Multiplying both sides of equation (224) by h^2 implies that if $h \rightarrow 0$ then the coefficient matrix $\frac{1}{h^2} M_2^{-1} A_2 + \frac{1}{h} B_3 M_1^{-1} A_1 + C_3$ tends to the matrix $M_2^{-1} A_2$ and we have $A_2 Y = B_2$. Using elementary row operations, A_2 will be converted to a tridiagonal matrix. So, assuming that $\tilde{\alpha} \neq 1, 10$, the matrix A_2 is invertible (see Theorem 6.31 in [2]).

5 Numerical results

In this section, we investigate our method accuracy by considering different values for the nonlinearity parameter α and the Hartmann electric number H . To solve the problem (689), the described method is applied with five iterations ($n = 5$) and $N = 100$ knots. We set $\hat{\alpha} = \frac{1}{4}$ and $\tilde{\alpha} = \frac{1}{10}$ to reach fourth order accuracy. Figure 42 (a) shows the curve of the approximate solution and the numerical solutions obtained by the `bvp4c` command in Matlab.

Now we discuss the velocity profile behavior under variation of the two relevant parameters, the electric Hartmann number H and the nonlinearity α . First, we plot the velocity fields for $H^2 = 0.5, 1, 2, 4$ and $\alpha = 0.5$. Figure 42 (b) shows that the conduit centreline velocity increases with the increase in Hartmann electric numbers. The effect of nonlinearity on the velocity α for different values of H^2 is shown in 40. From the figures, it can be observed that the velocity decreases as the α increases.

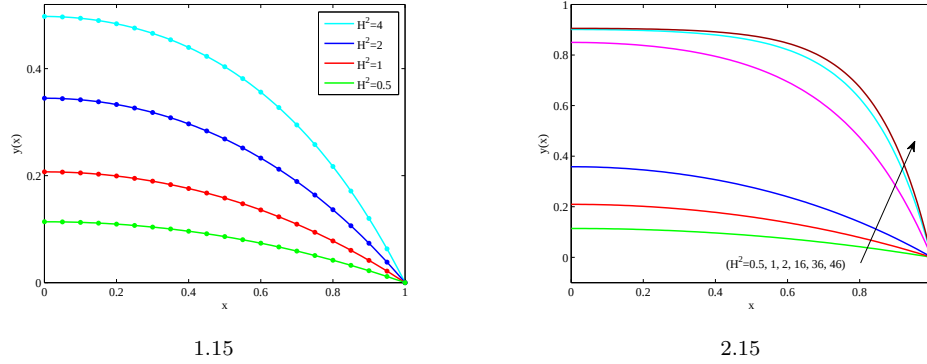


Figure 15: (a) The curve of the solutions for $\alpha = 0.5$ and $H^2 = 0.5, 1, 2, 4$. The approximate solutions computed by proposed method (line), The numerical solutions computed by bvp4c command (dots), (b) Approximate solution for $\alpha = 0.1$ and $H^2 = 0.5, 1, 2, 16, 36, 46$

Table 15: The numerical solutions obtained by the proposed method and bvp4c command.

x_i	Proposed method	bvp4c	Absolute error
0	0.113745650138407	0.113745572240118	7.79E-8
0.2	0.109342497299210	0.109342515373067	1.81E-8
0.4	0.0960590437022219	0.096061267167261	2.22E-6
0.6	0.0736742300556366	0.073674974049292	7.43E-7
0.8	0.0418227662917361	0.041825788857662	3.02E-6

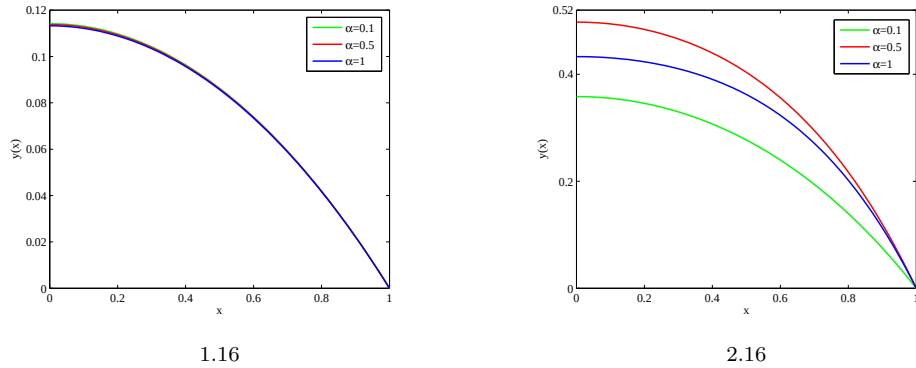


Figure 16: Approximate solutions for different values of the electric Hartmann and the strength of nonlinearity, $\alpha = 0.1, 0.5, 1$: (a) and $H^2 = 0.5$, (b) $H^2 = 4$.



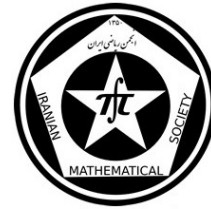
References

- [1] R.E. Bellman and R. Kabala, *Quasilinearization and nonlinear boundary-value problems*, Elsevier Publishing Company, New York, 1965.
- [2] R.L. Burden and J.D. Faires, *Numerical Analysis 9th edn.*, Boston: Brooks/Cole-Cengage Learning, 2010.
- [3] R. Kalaba, *On nonlinear differential equations, the maximum operation, and monotone convergence*, J. Math. Mech., (1959), 519–574.
- [4] S.K. Lele, *Compact Finite Difference Schemes with Spectral-Like Resolution*, J. Comput. Phys. , **103** (1992), 16–42.
- [5] V.B. Mandelzweig and F. Tabakin, *Quasilinearization approach to nonlinear problems in physics with application to nonlinear ODEs*, Computer Physics Communications, **1141** (2001), 268–281.
- [6] S. Mckee, R. Watson, J.A. Cuminato, J. Caldwell and M.S. Chen, *Calculation of electrohydrodynamic flow in a circular cylindrical conduit*, Z. Angew. Math. Mech. **77** (1997), 457–465.
- [7] P. Roul, M. Harshita and K. Klaus, *A new iterative algorithm for a strongly nonlinear singular boundary value problem*, J. Comput. Appl. Math., **35** (2019), 167–178.
- [8] P. Roul, M. Harshita and K. Klaus, *A sixth-order numerical method for a strongly nonlinear singular boundary value problem governing electrohydrodynamic flow in a circular cylindrical conduit*, Appl. Math. Comput., **350** (2019), 416–433.



8th Seminar on Numerical Analysis and its Applications

University of Kurdistan, Sanandaj, Iran, April 06-09 2021



Development of wavelets method for numerical treatment of two-dimensional nonlinear Volterra integral equations

K. Mozdgir¹⁴ and S. Sohrabi

Department of Mathematics, Faculty of Science, Urmia University, Urmia, Iran

Abstract

The main focus of this paper is to present an effective numerical method for solving the two-dimensional nonlinear Volterra integral equations (2DNVIEs), which appear in many phenomena in physics and engineering. The method is based on two-dimensional Legendre wavelets, which reduces 2DNVIEs to a nonlinear algebraic system of equations in collocation. The convergence analysis of the method is investigated and a numerical example presented to confirm the accuracy and computational efficiency of the method.

AMS subject Classification 2010: 65R20; 45F15.

Keywords: Two-dimensional integral equations; Legendre wavelets; Collocation method.

1 Introduction

We consider the two-dimensional nonlinear Volterra integral equation of the second kind as follows:

$$f(x, y) = g(x, y) + \int_0^y \int_0^x k(x, y, s, t, f(s, t)) ds dt \quad (x, y) \in [0, 1] \times [0, 1], \quad (225)$$

where the functions g and k are known continuous functions defined on $[0, 1] \times [0, 1]$ and $\Omega = [0, 1]^4 \times \mathbb{R}$, respectively, and $f(x, y)$ is an unknown real-valued function to be determined. Equations of this type appear in many practical applications such as physics, biology and engineering. McKee et al. [7] showed that a class of nonlinear telegraph equations is equivalent to two-dimensional Volterra integral equations. Several authors have written a number of papers which establish numerical techniques for finding an approximation of the 2DNVIEs. Recently, a number of algorithms have been developed for the numerical solution of two-dimensional integral equations. These methods can be categorized into radial basis functions methods (see, e.g. [8, 1, 2]), triangular functions method (see, e.g. [5]), Legendre polynomials method (see, e.g. [9]), hybrid functions method (see, e.g. [6]) and the rationalized Haar functions method

¹⁴Speaker



(see, e.g. [3, 4]).

In this paper we apply the wavelet collocation method using Legendre wavelets for numerical treatment of this equation. The analysis of the accuracy estimation of the method is given and the performance of the proposed method is illustrated by means of a numerical example.

2 Two-dimensional Legendre wavelets

Definition

Consider the well-known Legendre polynomials, which are orthogonal with respect to the weight function $w(x) = 1$ and derived from the following recursive formula

$$L_0(x) = 1,$$

$$L_1(x) = x,$$

$$L_m(x) = \frac{2m-1}{m}xL_{m-1}(x) - \frac{m-1}{m}L_{m-2}(x), \quad m = 2, 3, \dots$$

Legendre wavelets $\psi_{n,m}(x) = \psi(k, \hat{n}, m, x)$ have four arguments, $k = 2, 3, \dots, \hat{n} = 2n-1, n = 1, 2, 3, \dots, 2^{k-1}, m = 0, 1, 2, \dots, M-1$ is the order of Legendre polynomials and M is a fixed positive integer. They are defined on the interval $[0, 1)$ as follows:

$$\psi_{n,m}(x) = \begin{cases} \sqrt{(m + \frac{1}{2})} 2^{\frac{k}{2}} L_m(2^k x - \hat{n}), & \frac{\hat{n}-1}{2^k} \leq x < \frac{\hat{n}+1}{2^k} \\ 0, & \text{elsewhere.} \end{cases} \quad (226)$$

We define the two-dimensional Legendre wavelet $\psi_{n_1 m_1 n_2 m_2}(x, y)$ as:

$$\begin{cases} \sqrt{(m_1 + \frac{1}{2})(m_2 + \frac{1}{2})} 2^{\frac{k_1+k_2}{2}} L_{m_1}(2^{k_1} x - \hat{n}_1) L_{m_2}(2^{k_2} y - \hat{n}_2), & \frac{\hat{n}_1-1}{2^{k_1}} \leq x < \frac{\hat{n}_1+1}{2^{k_1}} \\ & \frac{\hat{n}_2-1}{2^{k_2}} \leq y < \frac{\hat{n}_2+1}{2^{k_2}} \\ 0, & \text{elsewhere.} \end{cases} \quad (227)$$

where $m_1 = 0, 1, \dots, M_1-1, m_2 = 0, 1, \dots, M_2-1$. Here $\hat{n}_1$ and $\hat{n}_2$ are defined similarly to $\hat{n}$, k_1 and k_2 are positive integers, L_{m_1} and L_{m_2} are Legendre polynomials of order m_1 and m_2 , respectively.

The expansion of a function

A function $f(x, y)$ defined over $[0, 1) \times [0, 1)$ may be expanded as

$$f(x, y) = \sum_{i_1=1}^{\infty} \sum_{j_1=0}^{\infty} \sum_{i_2=1}^{\infty} \sum_{j_2=0}^{\infty} c_{i_1 j_1 i_2 j_2} \psi_{i_1 j_1 i_2 j_2}(x, y). \quad (228)$$

If the infinite series in Eq. (654) is truncated, then Eq. (654) can be written as

$$f(x, y) \simeq \sum_{i_1=1}^{2^{k_1-1}} \sum_{j_1=0}^{M_1-1} \sum_{i_2=1}^{2^{k_2-1}} \sum_{j_2=0}^{M_2-1} c_{i_1 j_1 i_2 j_2} \psi_{i_1 j_1 i_2 j_2}(x, y) = \mathbf{C}^T \Psi(x, y) = \Psi^T(x) \mathbf{F} \Psi(y), \quad (229)$$



where $\Psi(x)$ and $\Psi(y)$ are $2^{k_1-1}M_1 \times 1$ and $2^{k_2-1}M_2 \times 1$ matrices respectively given by

$$\Psi(x) = [\psi_{10}(x), \dots, \psi_{1M_1}(x), \psi_{20}(x), \dots, \psi_{2M_1}(x), \dots, \psi_{2^{k_1-1}0}(x), \dots, \psi_{2^{k_1-1}M_1}(x)]^T, \quad (230)$$

$$\Psi(y) = [\psi_{10}(y), \dots, \psi_{1M_2}(y), \psi_{20}(y), \dots, \psi_{2M_2}(y), \dots, \psi_{2^{k_2-1}0}(y), \dots, \psi_{2^{k_2-1}M_2}(y)]^T. \quad (231)$$

$\mathbf{F}$ is a $2^{k_1-1}M_1 \times 2^{k_2-1}M_2$ matrix whose elements can be calculated from

$$\int_0^1 \int_0^1 \Psi_{i_1 j_1}(x) \Psi_{i_2 j_2}(y) f(x, y) dy dx,$$

with $i_1 = 1, \dots, 2^{k_1-1}$, $j_1 = 0, \dots, M_1 - 1$, $i_2 = 1, \dots, 2^{k_2-1}$, $j_2 = 0, \dots, M_2 - 1$.

3 Methodology

In this section, we consider the two-dimensional nonlinear Volterra integral equation (655) and perform the expansion of the unknown function $f(x, y)$ using two-dimensional Legendre wavelets expansion given in Eq. (229) such that $k_1 = k_2 = k$, $M_1 = M_2 = M$, and $N = 2^{k-1}$. Then we have

$$f(x, y) \simeq \bar{f}_{N,M}(x, y) = \sum_{i_1=1}^N \sum_{j_1=0}^{M-1} \sum_{i_2=1}^N \sum_{j_2=0}^{M-1} f_{i_1 j_1 i_2 j_2} \psi_{i_1 j_1 i_2 j_2}(x, y) = \mathbf{F}^T \Psi(x, y), \quad (232)$$

Substituting Eq. (457) into Eq. (655) gives

$$\bar{f}_{N,M}(x, y) = g(x, y) + \int_0^y \int_0^x k(x, y, s, t, \bar{f}_{N,M}(s, t)) ds dt \quad (233)$$

Now, we discretize system (662) at the set of collocation nodes (x_i, y_j) for $i, j = 1, 2, \dots, NM$ as follows:

$$g(x_i, y_j) = \bar{f}_{N,M}(x_i, y_j) - \int_0^{y_j} \int_0^{x_i} k(x_i, y_j, s, t, \bar{f}_{N,M}(s, t)) ds dt \quad (234)$$

where

$$x_i = \frac{1}{2} \left(\cos\left(\frac{(2i-1)\pi}{2NM}\right) + 1 \right), \quad i = 1, 2, \dots, NM \quad (235)$$

$$y_j = \frac{1}{2} \left(\cos\left(\frac{(2j-1)\pi}{2NM}\right) + 1 \right), \quad j = 1, 2, \dots, NM \quad (236)$$

are zeros of the shifted Chebyshev polynomials $T_{NM}(2x-1)$ and $T_{NM}(2y-1)$, respectively. Now the Gauss-Legendre quadrature formula is employed to approximate the integral operator in Eq. (658). For this purpose, linear transformations must be applied with the following forms

$$\tau = \frac{2}{x_i} s - 1, \quad s \in [0, x_i], \quad (237)$$



$$\eta = \frac{2}{y_j}t - 1, \quad t \in [0, y_j]. \quad (238)$$

Then

$$g(x_i, y_j) = \bar{f}_{N,M}(x_i, y_j) - \frac{x_i y_j}{4} \int_{-1}^1 \int_{-1}^1 k\left(x_i, y_j, \left(\frac{x_i}{2}(\tau + 1), \frac{y_j}{2}(\eta + 1), \bar{f}_{N,M}\left(\frac{1}{2}(\tau + 1), \frac{1}{2}(\eta + 1)\right)\right)\right) d\tau d\eta, \quad (239)$$

By applying the Legendre–Gauss–Lobatto integration formula, we obtain

$$g(x_i, y_j) = \bar{f}_{N,M}(x_i, y_j) - \frac{x_i y_j}{4} \sum_{n=1}^{r_1} \sum_{m=1}^{r_2} w_n w_m \left(k\left(x_i, y_j, \left(\frac{x_i}{2}(\tau_m + 1), \frac{y_j}{2}(\eta_n + 1), \bar{f}_{N,M}\left(\frac{x_i}{2}(\tau_m + 1), \frac{y_j}{2}(\eta_n + 1)\right)\right)\right), \right. \\ \left. i = 1, 2, \dots, NM \quad j = 1, 2, \dots, NM, \right) \quad (240)$$

where τ_m and η_n are the Legendre–Gauss points, zeros of Legendre polynomials of degrees r_1 and r_2 in $[-1, 1]$, respectively, and w_m s and w_n s are the corresponding weights. Eq. (240) generates a nonlinear system of $(NM)^2$ algebraic equations that can be solved using Newton's iteration method.

4 Convergence analysis

In this section we obtain an error estimate for the best approximation of the function f based on Legendre wavelets and describe the convergence behavior of the proposed numerical method. For this purpose, we present the following results. The first theorem provides an error term for the best approximation of $f(x, y)$.

Theorem 0.49. *Suppose that $f \in C^M([0, 1] \times [0, 1])$ is a real-valued function. Then $f_{NM}(x, y) = \mathbf{C}^T \Psi(x, y)$ approximates $f(x, y)$ with the following error bound*

$$\|f(x, y) - \mathbf{C}^T \Psi(x, y)\|_2 \leq \frac{\gamma 2^M}{N^M M!}, \quad \gamma = \max_{\substack{x, y \in [0, 1] \\ 0 \leq k \leq M}} \left| \frac{\partial^M f(x, y)}{\partial x^k \partial y^{M-k}} \right|. \quad (241)$$

Theorem 0.50. *Consider the Eq. (655) and suppose that $k \in C^1(\Omega)$, $\Omega = [0, 1]^4 \times \mathbb{R}$, with $C_0 = \sup_{\Omega} |k_z(x, y, s, t, z)| < \infty$. Moreover, let $f \in C^M([0, 1] \times [0, 1])$ ($M > 2$) be the exact solution of the Eq. (655). Then, we have,*

$$\|f(x, y) - \bar{f}_{NM}(x, y)\|_2 \leq \frac{2^M}{N^N M!} \left(\gamma + C_1 \|\mathbf{A}^{-1}\|_2 (NM)^2 ((M) + 1) \right), \quad (242)$$

where $\bar{f}_{NM}(x, y)$ is the approximation of f and

$$\mathbf{A} = \left[\psi_{i_1 j_1 i_2 j_2}(x_i, y_j) - \int_0^{y_j} \int_0^{x_i} k_z(x_i, y_j, s, t, \zeta) \psi_{i_1 j_1 i_2 j_2}(s, t) ds dt \right] \begin{matrix} 1 \leq i_1, i_2 \leq N \\ 0 \leq j_1, j_2 \leq M-1, \end{matrix}$$

that assumed to be nonsingular.



Table 16: Absolute errors for Example 5.1.

$(x, y) = (\frac{1}{2^l}, \frac{1}{2^l})$	$k = 2, M = 2$	$k = 2, M = 3$	method in [9] with $M = 3$
$l = 1$	1.86×10^{-1}	8.67×10^{-16}	2.8×10^{-6}
$l = 2$	2.51×10^{-2}	2.43×10^{-17}	1.7×10^{-4}
$l = 3$	3.20×10^{-2}	2.78×10^{-17}	1.3×10^{-5}
$l = 4$	1.20×10^{-2}	1.39×10^{-17}	3.5×10^{-5}
$l = 5$	3.78×10^{-3}	1.39×10^{-17}	3.2×10^{-5}
$l = 6$	1.31×10^{-2}	1.39×10^{-17}	1.9×10^{-5}
$\ e_{N,M}(x, y)\ _2$	2.95×10^{-2}	1.67×10^{-15}	2.49×10^{-4}

5 Numerical example

In this section, we apply the presented method for a numerical example (selected from [9]) to demonstrate the accuracy and effectiveness of the method. For different values of k, M , the absolute values of errors are presented and compared with the results obtained in [9] using Legendre polynomials in Table 1, which confirms that the proposed method in this paper is more effective than the method in [9].

Example 0.51. Consider the following two-dimensional nonlinear Volterra integral equation of second kind:

$$f(x, y) = g(x, y) + \int_0^y \int_0^x f^2(s, t) ds dt, \quad (x, y) \in [0, 1) \times [0, 1),$$

where

$$g(x, y) = x^2 + y^2 - \frac{1}{45}xy(9x^4 + 10x^2y^2 + 9y^4),$$

with the exact solution $f(x, y) = x^2 + y^2$.

References

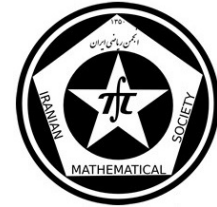
- [1] A. Alipanah, S. Esmaeili, *Numerical solution of the two-dimensional fredholm integral equations by gaussian radial basis function*, J. Comput. Appl. Math. **235** (2011) 5342–5347.
- [2] H. Almasieh, J. Nazari Meleh, *Numerical solution of a class of mixed two-dimensional nonlinear volterra-fredholm integral equations using multiquadric radial basis functions*, J. Comput. Appl. Math. **260** (2014) 173–179.
- [3] I. Aziz, Siraj-ul-Islam, F. Khan, *A new method based on haar wavelet for the numerical solution of two-dimensional nonlinear integral equations*, J. Comput. Appl. Math. **272** (2014) 70–80.
- [4] E. Babolian, S. Bazm, P. Lima, *Numerical solution of nonlinear two-dimensional integral equations using rationalized haar functions*, J. Comput. Appl. Math. **16** (2011) 1164–1175.
- [5] K. Maleknejad, Z. JafariBehbahani, *Applications of two-dimensional triangular functions for solving nonlinear class of mixed volterra-fredholm integral equations*, Math. Comput. Model. **55** (2012) 1833–1844.



- [6] K. Maleknejad, E. Saeedipoor, function method and convergence analysis for two-dimensional non-linear integral equations, *J. Comput. Appl. Math.* **322** (2017) 96–108.
- [7] S. McKee, T. Tang, T. Diogo, *An euler-type method for two-dimensional volterra integral equations of the first kind*, *IMA. J. Numer. Anal.* **20** (2000) 423–440.
- [8] F. Mirzaee, E. Hadadiyan, *Numerical solution of linear fredholm integral equations via twodimensional modification of hat functions*, *Appl. Math. Comput.* **250** (2015) 805–816.
- [9] S. Nemati, P. Lima, Y. Ordokhani, *Numerical solution of a class of two-dimensional nonlinear volterra integral equations using legendre polynomials*, *J. Comput. Appl. Math.* **242** (2013) 53–69.

email: kolsummozdgir@gmail.com

email: s.sohrabi@urmia.ac.ir



Differential transform and exponential spline function approximations for fractional sub-diffusion equations

Parisa Noori[†], Reza Jalilian^{†15}

Department of Mathematics, Razi University Tagh Bostan, Kermanshah 6714967346 Kermanshah, Iran

nooriniledgolbarg.parisa@stu.razi.ac.ir

Department of Mathematics, Razi University Tagh Bostan, Kermanshah 6714967346 Kermanshah, Iran

reza.jalilian72@gmail.com

Abstract

In this paper, we shall develop a numerical scheme for a fractional sub-diffusion problem using new exponential spline and differential transform method. The convergence analysis of the method has been discussed. Numerical examples are presented to illustrate the applications of the method and to compare the computed results with other known methods.

AMS subject Classification 2010: 34B15; 33F05; 65D20; 41A15

Keywords: Fractional sub-diffusion equation, Exponential spline functions, Differential transform method.

1 Introduction

In recent years, the derivation of solutions to fractional differential equations have become a hot topic in many fields of applied sciences and engineering. A large number of applied problems are formulated by fractional differential equations because for most real world problems fractional derivatives provide more accurate models than integer order derivatives do. It frequently appears in many fields such as viscoelastic and viscoelastic flow, control theory, transport problems, tumor development, random walks, continuum mechanics and turbulence [1]- [8]. There are many publications on the numerical solutions of fractional differential equations, see [9]- [34] and the references cited therein. In [9] have developed non-polynomial spline scheme for solving the fractional sub-diffusion equations and show that the convergence of the proposed scheme is at least $O(\tau^{2-\gamma} + h^{4.5})$, in maximum norm. In [10] an approximation technique using trigonometric cubic B-splines was presented for the numerical solution of the fractional sub-diffusion equation. In [11] have used Grünwald formula for the approximation of space fractional derivatives of order $0 < \gamma < 1$. The numerical solutions of the fractional diffusion equation by a weighted average finite difference method has derived In [12]. For fractional equations with one spatial variable, parametric, polynomial and non-polynomial spline developed in [13] and [14]. A Fourier method for the fractional diffusion equation describing sub-diffusion was studied by Chen *et al.* In [15]. In [16], Cui raised a high order compact difference scheme and its convergence was discussed, another similar approach was studied in [17] for the fractional sub-diffusion equation by G. Gao and Z. Sun. In [18] Chen *et al.* studied numerical schemes with high spatial accuracy for a variable-order anomalous sub-diffusion equation, The finite difference approximation was considered for a class of fractional sub-diffusion equations on a space unbounded domain In [19]. Gao *et al.* established stability and convergence of finite

¹⁵Speaker



difference schemes for a class of time-fractional sub-diffusion equations based on certain super convergence in [20] and a finite difference scheme for fractional sub-diffusion equations on an unbounded domain using artificial boundary conditions was analyzed in [21]. The accuracy and stability of an implicit solution method for the fractional diffusion equation was developed by Langlands and Henry in [22]. Lin and Xu studied Finite difference/spectral approximations for the time-fractional diffusion equation in [23]. A high-order and unconditionally stable scheme for the modified anomalous fractional sub-diffusion equation with a nonlinear source term was analyzed by Mohebbi *et al.* in [24]. D.A. Murio in [25] was studied implicit finite difference approximation for time fractional diffusion equations. Wang and Vong extended compact difference schemes for the modified anomalous fractional sub-diffusion equation and the fractional diffusion-wave equation in [26]. Orthogonal spline collocation method was proposed for the two-dimensional fractional sub-diffusion equation by Yang *et al.* in [27]. Alternating direction implicit schemes for the two-dimensional fractional sub-diffusion equation was analyzed by Zhang and Sun in [28]. X. Zhao and Z. Sun developed A box-type scheme for fractional sub-diffusion equation with Neumann boundary conditions in [29]. Orthogonal spline collocation methods for the sub-diffusion equation was applied by Tinggang Zhao *et al.* in [30]. K. Al-Khaled studied numerical solution of time-fractional partial differential equations using sumudu decomposition method in [31]. A. Arikglu and I. Ozkol studied solution of fractional differential equations by using differential transform method in [32]. Cubic polynomial spline method applied to an approximate solution for a class of boundary value problem of fractional order to Dirichlet boundary conditions by W. K. Zahra and S. M. Elkholy in [33]. The existence, uniqueness and continuous dependence of solutions for a boundary value problem of the nonlinear fractional differential equation have been discussed in [34].

In this paper, we consider fractional sub-diffusion problem with the boundary condition in the following form:

$$\begin{cases} D_t^\alpha u(x, t) + d \frac{\partial^2 u(x, t)}{\partial x^2} = f(x, t), \\ u(x, 0) = \psi_1(x), \\ (x, t) \in \Omega = [a, b] \times [0, T]. \end{cases} \quad (243)$$

with following two types of boundary conditions:

1. Dirichlet boundary conditions:

$$u(a, t) = h_a(t), \quad u(b, t) = h_b(t),$$

2. Neumann's boundary conditions:

$$\frac{\partial u(a, t)}{\partial x} = g_a(t), \quad \frac{\partial u(b, t)}{\partial x} = g_b(t),$$

where

$$D_t^\alpha u(x, t) = \frac{1}{\Gamma(1 - \alpha)} \int_0^t (t - \xi)^{-\alpha} \frac{\partial u(x, \xi)}{\partial \xi} d\xi,$$

is the Caputa fractional derivatives, d is constant and $0 < \alpha \leq 1$. we shall propose an exponential spline method for a fractional sub-diffusion problem that proves to be of higher order than the other methods



in the maximum norm. We focus on the high-order numerical solutions for time-space fractional sub-diffusion equations by using differential transform method and exponential spline schemes. The paper is organized as follows: A exponential spline approach is presented in Section 2. The differential transforms method for time-fractional derivative and converges analysis are discussed in Section 3. The convergence of the scheme is analyzed in sections 4. Finally, in section 5, numerical evidence is included to demonstrate the practical usefulness of our methods and confirm their theoretical behavior.

2 Exponential spline function

We introduce the set of grid points in the interval $[a, b] \times [0, T]$ in the following form:

$$x_0 = a, \quad x_i = a + ih, \quad h = \frac{b-a}{N}, \quad i = 1, 2, \dots, N, \quad x_N = b,$$

$$t_0 = 0, \quad t_j = j\tau, \quad \tau = \frac{T}{M}, \quad j = 0, 1, 2, \dots, M, \quad t_M = T.$$

Let $\bar{u}(x, t)$ denote the exact solution of problem (522) and we shall use the notations $\bar{u}_i^j = \bar{u}(x_i, t_j)$, and $f_i^j = f(x_i, t_j)$, further let u_i^j , denote an approximation of $\bar{u}_i^j$. We shall approximate the solution $\bar{u}(x, t_j)$, At each fixed $t = t_j, 0 \leq j \leq M$, by using exponential spline $Q_i(x, t_j) \in C^\infty[a, b]$ in the subinterval $[x_i, x_{i+1}]$ which is defined in the following form:

$$Q_i(x, t_j) = a_i(t_j)e^{\beta(x-x_i)} + b_i(t_j)e^{-(\beta)(x-x_i)} + c_i(t_j)e^{2\beta(x-x_i)} + d_i(t_j)e^{-2\beta(x-x_i)}, \quad (244)$$

where β is a free parameter of the spline functions which can be real or pure imaginary and which will be used to raise the accuracy of the method see [35] (page 386 equation 2.9 and page 390 equation 4.5).

Let u_i^j be an approximation to $\bar{u}_i^j$ obtained by the segment $Q_i(x, t_j)$ passing through the points (x_i, u_i^j) and (x_{i+1}, u_{i+1}^j) .

To derive the coefficients of the Eq. (527), we first define:

$$\begin{cases} Q_i(x_i, t_j) = u_i^j, & Q_i^{(2)}(x_i, t_j) = M_i^j, \\ Q_i(x_{i+1}, t_j) = u_{i+1}^j, & Q_i^{(2)}(x_{i+1}, t_j) = M_{i+1}^j. \end{cases} \quad (245)$$

By algebraic manipulation we get

$$\begin{aligned} a_i(t_j) &= \frac{M_i - e^\theta M_{1+i} - 4\beta^2 u_i + 4e^\theta \beta^2}{3(-1 + e^{2\theta})\beta^2}, \\ b_i(t_j) &= \frac{e^\theta (-e^\theta M_i + M_{1+i} + 4e^\theta \beta^2 u_i - 4\beta^2)}{3(-1 + e^{2\theta})\beta^2}, \\ c_i(t_j) &= -\frac{M_i - e^{2\theta} M_{1+i} - \beta^2 + e^{2\theta} \beta^2}{3(-1 + e^{4\theta})\beta^2}, \\ d_i(t_j) &= -\frac{e^{2\theta} (-e^{2\theta} M_i + M_{1+i} + e^{2\theta} \beta^2 - \beta^2)}{3(-1 + e^{4\theta})\beta^2}, \end{aligned}$$



where $\theta = h\beta$. Applying the continuity of the first derivative of $Q'_i(x, t_j) = Q'_{i-1}(x, t_j)$ at $x = x_i$ we get the following consistency relation:

$$u_{i+1}^j + \lambda_3 u_i^j + u_{i-1}^j = h^2 [\lambda_1 M_{i-1}^j + \lambda_2 M_i^j + \lambda_1 M_{i+1}^j], i = 1, \dots, N-1, \quad (246)$$

where

$$\begin{aligned} \lambda_1 &= \frac{(-1 + e^\theta)^2}{2(2 - e^\theta + 2e^{2\theta})\theta^2}, \\ \lambda_2 &= \frac{e^{-\theta}(-1 + e^{2\theta})^2}{2(2 - e^\theta + 2e^{2\theta})\theta^2}, \\ \lambda_3 &= -\frac{4e^{2\theta-2\theta}(1 + 4e^{2\theta} + e^{4\theta})}{(4e^\theta(2 - e^\theta + 2e^{2\theta}))}, \end{aligned}$$

Similarly we obtain

$$h(\alpha_1 m_{i-1}^j + \beta_1 m_i^j + \alpha_1 m_{i+1}^j) = (-\gamma_1 u_{i-1}^j + \gamma_1 u_{i+1}^j), i = 1, \dots, N-1, \quad (247)$$

where $Q'_i(x_i, t_j) = m_i^j$,

$$\begin{aligned} \alpha_1 &= -\frac{6e^\theta(1 + e^\theta)\theta}{-1 - 3e^\theta + 3e^{2\theta} + e^{3\theta}}, \\ \beta_1 &= -\frac{6(1 + e^\theta)^3\theta}{-1 - 3e^\theta + 3e^{2\theta} + e^{3\theta}}, \\ \gamma_1 &= -\frac{12e^\theta(1 + e^\theta + e^{2\theta})\theta^2}{(-1 + e^\theta)^2(1 + 4e^\theta + e^{2\theta})}, \end{aligned}$$

In the limiting case when $\theta \rightarrow 0$, the relations (246) and (247) reduce into ordinary cubic spline relation:

$$\begin{cases} \frac{h^2}{6}[M_{i-1}^j + 4M_i^j + M_{i+1}^j] = u_{i+1}^j - 2u_i^j + u_{i-1}^j, & i = 1, \dots, N-1, \\ -2h(m_{i-1}^j + 4m_i^j + m_{i+1}^j) = (6u_{i-1}^j - 6u_{i+1}^j), & i = 1, \dots, N-1, \end{cases} \quad (248)$$

The proposed differential equation (522) in the mesh point (x_i, t_j) , may be discretized by:

$$M_i^j = f_i^j - D_t^\alpha u_i^j - Lu_i^j, i = 1, \dots, N-1, j = 0, 1, 2, \dots, M. \quad (249)$$

By the Taylor expansion the local truncation errors $t_i^j, i = 1, \dots, N-1$, associated with the (246) are given by

$$\begin{aligned} t_i^j &= (-2 - \lambda_3)u_i^j + h^2(-1 + 2\lambda_1 + \lambda_2)\left(\frac{\partial^2 u}{\partial x^2}\right)_i^j + \frac{1}{12}h^4(-1 + 12\lambda_1)\left(\frac{\partial^4 u}{\partial x^4}\right)_i^j \\ &\quad + \frac{1}{360}h^6(-1 + 30\lambda_1)\left(\frac{\partial^6 u}{\partial x^6}\right)_i^j + \frac{h^8(-1 + 56\lambda_1)}{20160}\left(\frac{\partial^8 u}{\partial x^8}\right)_i^j + O(h^9). \end{aligned} \quad (250)$$

For different choices of parameters $\lambda_i = 1, 2, 3$ we can obtain classes of methods.

(i) **Second-Order Method** For $\lambda_3 = -2, \lambda_1 = \frac{1}{6}$ and $\lambda_2 = \frac{2}{3}$ at each fixed $t = t_j, 0 \leq j \leq M$, gives



$$t_i^j = \frac{1}{12}h^4\left(\frac{\partial^4 u}{\partial x^4}\right)_i^j, i = 1, \dots, N-1.$$

(i) **Fourth-Order Method** For $\lambda_3 = -2, \lambda_1 = \frac{1}{12}$ and $\lambda_1 = \frac{5}{6}$ at each fixed $t = t_j, 0 \leq j \leq M$, gives

$$t_i^j = \frac{1}{240}h^6\left(\frac{\partial^6 u}{\partial x^6}\right)_i^j, i = 1, \dots, N-1.$$

3 The fractional differential transform method

In this section, we shall derive the differential transform method for the time-fractional derivative. The proposed method is based on Taylor's formula.

Consider a function $v(t)$, on the basis of the properties of differential transform, the function $v(t)$, can be represented as

$$v(t) = \sum_{k=0}^{\infty} V(k)(t - t_0)^{k\alpha}, \quad (251)$$

where $k \in N, 0 < \alpha < 1$ and t_0 is first point in the interval $[0, T]$

$$V(k) = \frac{1}{\Gamma(\alpha k + 1)} [(D_{t_0}^{\alpha} v(t))_{t=t_0}], t \in (t_0, t_0 + r), r > 0, \quad (252)$$

which is also called the differential transformation of $v(t)$ and can be represented by a finite-term series plus a remainder as shown below [32]:

$$v(t) = \sum_{k=0}^K V(k)(t - t_0)^{k\alpha} + R_{K+1}(t). \quad (253)$$

we will recall some basics definitions needed in the following theorem[32].

Theorem 0.52. Suppose that $V_1(k), V_2(k)$ and $V_3(k)$ are the differential transformations of the functions $v_1(t), v_2(t)$ and $v_3(t)$, respectively:

- (a) If $v_1(t) = v_2(t) \pm v_3(t)$, then $V_1(k) = V_2(k) \pm V_3(k)$,
- (b) If $v_1(t) = av_2(t)$, $a \in R$, then $V_1(k) = aV_2(k)$,
- (c) If $v_1(t) = (t - t_0)^{n\alpha}$, then $V_1(k) = \delta(k - n)$,
- (d) If $v_1(t) = D^{\alpha}v_2(t)$, $0 < \alpha < 1$, then $V_1(k) = \frac{\Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k + 1)} V_2(k + 1)$.

by applying differential transform to (246) and (249) we have

$$\left\{ \begin{array}{l} U_{i+1}(k) + \lambda_3 U_i(k) + U_{i-1}(k) = -h^2 \left[\frac{\Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k + 1)} (\lambda_1 U_{i+1}(k+1) + \lambda_2 U_i(k+1) \right. \\ \quad \left. + \lambda_1 U_{i+1}(k+1)) - (\lambda_1 F_{i-1}(k) + \lambda_2 F_i(k) \right. \\ \quad \left. + \lambda_1 F_{i+1}(k)) \right], i = 1, 2, \dots, N-1, \\ k = 0, 1, 2, \dots, \end{array} \right. \quad (254)$$

where F and U is the differential transformations of the functions f and u .



3 Convergence analysis

In this section, we investigate the convergence analysis of the proposed numerical scheme in maximum norm. We shall show that the convergence analysis of exponential spline in the spatial direction and convergence of differential transform method in the time direction. Let

$$E_i(k) = U_i(k) - \bar{U}_i(k), i = 1, 2, 3, \dots, N - 1,$$

$$U = (U_1(k), \dots, U_{N-1}(k))^t, E = (E_1(k), \dots, E_{N-1}(k))^t,$$

where $\bar{U}_i$ and U_i are the differential transformations of the exact solution and approximate solution. Equation (246), (249) and theorem (0.52) yields the systems of equations, and may be written in matrix form as

$$\frac{h^2 \Gamma(\alpha(k+1) + 1)}{\Gamma(\alpha k + 1)} A_0 U(k+1) = -BU(k) - h^2 A_0 F + C, \quad (255)$$

in (255) the matrices A_0, B, F and C are order $N - 1$ in the following forms:

$$A_0 = (a_{ij}) = \begin{cases} \lambda_1 & i = j = 1, 2, 3, \dots, N - 1, \\ \lambda_2 & |i - j| = 1, \\ 0 & \text{otherwise.} \end{cases}, \quad (256)$$

$$B = (b_{ij}) = \begin{cases} -2 & i = j = 1, 2, 3, \dots, N - 1, \\ 1 & |i - j| = 1, \\ 0 & \text{otherwise.} \end{cases}, \quad (257)$$

$$C = \begin{pmatrix} -\frac{h^2 \Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k+1)} \lambda_1 U_0(k+1) - U_0(k) + h^2 \lambda_1 (F_0) \\ \vdots \\ -\frac{h^2 \Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k+1)} \lambda_1 U_N(k+1) - U_N(k) + h^2 \lambda_1 (F_N) \end{pmatrix},$$

$F = \text{diag}(-F_i(k)), i = 1, \dots, N - 1$, is diagonal matrix of order $N - 1$ and we consider

$$\frac{h^2 \Gamma(\alpha(k+1) + 1)}{\Gamma(\alpha k + 1)} A_0 \bar{U}(k+1) = -BU(k) - h^2 A_0 F + T, \quad (258)$$

where T is vector of truncation error, $\bar{F} = \text{diag}(-F_i(k)), i = 1, \dots, N - 1$, and $U = (U_1(k), \dots, U_{N-1}(k))^t$. By using (255) and (258) we get

$$\frac{h^2 \Gamma(\alpha(k+1) + 1)}{\Gamma(\alpha k + 1)} A_0 E = T, \quad (259)$$

where $\text{Max}(U(k+1), U(k)) = U$, $\text{Max}(\bar{U}(k+1), \bar{U}(k)) = \bar{U}$ and $\text{Max}(E(k+1), E(k)) = E$. By using Usmani and Warst [36] the inverse of A_0 exists and is bounded as follows

$$\begin{cases} \|(A_0)^{-1}\| \leq 3, & \text{for } \lambda_1 = \frac{1}{6}, \lambda_2 = \frac{4}{6}, \\ \|(A_0)^{-1}\| \leq \frac{3}{2}, & \text{for } \lambda_1 = \frac{1}{12}, \lambda_2 = \frac{10}{12}. \end{cases} \quad (260)$$

Now suppose that $\bar{A}$ is in the following form:

$$\bar{A} = \frac{h^2 \Gamma(\alpha(k+1) + 1)}{\Gamma(\alpha k + 1)} A_0. \quad (261)$$



Lemma 4.1 If $\overline{M}_n$ is strictly diagonally dominant, then $\overline{M}_n$ is invertible. By using (259) and Lemma 4.1 we get $\overline{A}$ is invertible.

$$E = (\overline{A})^{-1}T. \quad (262)$$

Using Eq. (250) we get:

$$\begin{cases} \|T\| = \|t_i^j\| \leq \frac{1}{240}h^6 M_6, & \text{for } \lambda_1 = \frac{1}{12}, \lambda_2 = \frac{10}{12}, \\ \|T\| = \|t_i^j\| \leq \frac{1}{24}h^4 M_4, & \text{for } \lambda_1 = \frac{1}{6}, \lambda_2 = \frac{2}{3}, \end{cases} \quad (263)$$

where $|(\frac{\partial^6 u}{\partial x^6})|_{(\xi_i, t)} = M_6$, and $|(\frac{\partial^4 u}{\partial x^4})|_{(\xi_i, t)} = M_4$, at fixed t_j , where $u(\xi_i, t), x_i < \xi_i < x_{i+1}, i = 1, 2, \dots, N-1, t \in [0, T]$.

Theorem 0.53. Let $\overline{U}(x)$ be the exact solution of the boundary value problem (522) and $u_i, i = 1, 2, \dots, N-1$ be the numerical solution obtained by solving the system (255) Then we have:

$$\begin{cases} \|E\| \equiv O(h^4), & \text{for } \lambda_1 = \frac{1}{12}, \lambda_2 = \frac{10}{12}, \\ \|E\| \equiv O(h^2), & \text{for } \lambda_1 = \frac{1}{6}, \lambda_2 = \frac{2}{3}. \end{cases} \quad (264)$$

Proof. By using equation (262) and taking norm we get,

$$\|E\| \leq \|(\overline{A})^{-1}\| \|T\|. \quad (265)$$

Substituting (260) and (263) into (265) we can find,

$$\begin{aligned} \|E\| &\leq \|(\overline{A})^{-1}\| \|T\| = \frac{1}{h^2} \left| \frac{\Gamma(\alpha k + 1)}{\Gamma(\alpha(k+1) + 1)} \right| \|A_0^{-1}\| \|T\|, \\ \begin{cases} \|E\| \leq \frac{3\omega h^6 M_6}{480h^2} \equiv O(h^4), & \text{for } \lambda_1 = \frac{1}{12}, \lambda_2 = \frac{10}{12}, \\ \|E\| \leq \frac{3\omega h^4 M_4}{24h^2} \equiv O(h^2), & \text{for } \lambda_1 = \frac{1}{6}, \lambda_2 = \frac{2}{3}, \end{cases} \end{aligned} \quad (266)$$

where $\omega = \left| \frac{\Gamma(\alpha k + 1)}{\Gamma(\alpha(k+1) + 1)} \right|$.

Now if we suppose that $\overline{U}(x, t_j)$ be the exact solution of (1) and $U(x, t_j)$ be the numerical approximation to this solution by our numerical process at each fixed $t = t_j$, then we have:

$$\begin{cases} \|\overline{U}(x, t_j) - U(x, t_j)\| \equiv O(h^2), \\ \|\overline{U}(x, t_j) - U(x, t_j)\| \equiv O(h^4). \end{cases} \quad (267)$$

□



4 Numerical results

In this section, we present the results obtained by applying the numerical methods discussed in previous sections to the following two-point boundary-value problems. Examples 1-3 has been solved using our methods with different values of $\lambda_1 = \frac{1}{6}, \lambda_2 = \frac{4}{6}, \lambda_3 = -2, h, \tau, \alpha$ and also compared the obtained solution with the exact solution. The maximum absolute errors in solutions of the method are tabulated in Tables and compared with methods in [27], [29] and [30].

Example 1. Consider the following time-fractional equation:

$$D_t^\alpha u(x, t) - \frac{\partial^2}{\partial x^2} u(x, t) = 2t^\alpha - 2, \quad (268)$$

with the initial and boundary conditions

$$u(x, 0) = x^2, u(0, t) = \frac{2\Gamma(\alpha + 1)}{\Gamma(2\alpha + 1)} t^{2\alpha}, u(1, t) = 1 + \frac{2\Gamma(\alpha + 1)}{\Gamma(2\alpha + 1)} t^{2\alpha}. \quad (269)$$

From equation (268), then by applying differential transform to (530) we have

$$\left\{ \begin{array}{l} U_{i+1}(k) - 2U_i(k) + U_{i-1}(k) = \frac{h^2}{6} \left[\frac{\Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k+1)} (U_{i+1}(k+1) + 4U_i(k+1) \right. \\ \quad \left. + U_{i-1}(k+1)) - (F_{i-1}(k) + 4F_i(k) \right. \\ \quad \left. + F_{i+1}(k)) \right], i = 1, 2, \dots, N-1, \\ k = 0, 1, 2, \dots, \end{array} \right. \quad (270)$$

where F and U are the differential transformations of the functions f and u . We use (270) for equation (268) where

$$F_i(k) = 2(\delta(k-1) - \delta(k)), i = 1, 2, \dots, N-1$$

and differential equation for initial and boundary conditions (269) are as follows:

$$U_i(0) = (ih)^2, i = 1, 2, \dots, N-1, U_0(k) = \frac{2\Gamma(\alpha + 1)}{\Gamma(2\alpha + 1)} \delta(k-2), \quad (271)$$

$$U_N(k) = 1 + \frac{2\Gamma(\alpha + 1)}{\Gamma(2\alpha + 1)} \delta(k-2).$$

The exact solution of (268) is

$$u(x, t) = x^2 + \frac{2\Gamma(\alpha + 1)}{\Gamma(2\alpha + 1)} t^{2\alpha}.$$

Example 1 has been solved using our methods with different values of $\lambda_1 = \frac{1}{6}, \lambda_2 = \frac{4}{6}, \lambda_3 = -2, h = \frac{1}{10}, \tau = \frac{1}{10}, \alpha = 0.4, 0.5, 0.6$ and also compared the obtained solution with the exact solution. The maximum absolute errors in the solutions of the method is tabulated in Table 1.

Example 2. Consider the following time-fractional equation:

$$D_t^\alpha u(x, t) - \frac{\partial^2}{\partial x^2} u(x, t) = \frac{\Gamma(\alpha + 3)}{2} t^2 e^x x^2 (1-x)^2 - e^x t^{\alpha+2} (2 - 8x + x^2 + 6x^3 + x^4), \quad (272)$$



Table 17: L^∞ errors for $h = \frac{1}{10}$ and $\tau = \frac{1}{10}$ for example 1.

T	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$
1	$2.4973e - 11^*$	$2.2777e - 11$	$2.0244e - 11$
2	$4.5391e - 11$	$4.8066e - 11$	$4.9602e - 11$
3	$6.3654e - 11$	$7.3353e - 11$	$8.2375e - 11$

* $2.4973e - 11 = 2.4973 \times 10^{-11}$

with the initial and Neumann's boundary conditions

$$u(x, 0) = 0, u_x(0, t) = 0, u_x(1, t) = 0. \quad (273)$$

We use (270) for equation (272) where

$$\begin{cases} F_i(k) = \frac{\Gamma(\alpha+3)}{2} \delta(k - 2/\alpha) e^{ih} (ih)^2 (1 - ih)^2 \\ - e^{ih} \delta(k - (1 + 2/\alpha)) (2 - 8ih + (ih)^2 + 6(ih)^3 + (ih)^4) \\ i = 1, 2, \dots, N - 1. \end{cases}$$

and differential equation for initial and boundary conditions (273) is

$$U_i(0) = 0, i = 1, 2, \dots, N - 1, U_0(k) = U_1(k), U_N(k) = U_{N-1}(k). \quad (274)$$

The exact solution of (272) is

$$u(x, t) = e^x x^2 (1 - x)^2 t^{\alpha+2}.$$

The maximum absolute errors in the solutions of example 2 for different values of $\lambda_1 = \frac{1}{6}, \lambda_2 = \frac{4}{6}, \lambda_3 = -2, h = \frac{1}{200}, \frac{1}{400}, \frac{1}{800}, \frac{1}{1600}, \frac{1}{3200}, \tau = \frac{1}{20}, \frac{1}{40}, \frac{1}{80}, \frac{1}{160}, \frac{1}{320}, \alpha = 0.5$ are reported in Table 2 and compared with method in [29].

Table 18: L^∞ errors for $\alpha = 0.5$ and $T = 1$ for example 2.

τ	h	Our method	[29]
$\frac{1}{20}$	$\frac{1}{200}$	$4.0570e - 5$	$3.877e - 4$
$\frac{1}{40}$	$\frac{1}{400}$	$1.0219e - 5$	$1.488e - 4$
$\frac{1}{80}$	$\frac{1}{800}$	$2.5644e - 6$	$5.554e - 5$
$\frac{1}{160}$	$\frac{1}{1600}$	$6.4231e - 7$	$2.037e - 5$
$\frac{1}{320}$	$\frac{1}{3200}$	$1.6073e - 7$	$7.388e - 6$

Example 3. Consider the following time-fractional equation:

$$D_t^\alpha u(x, t) - \frac{\partial^2}{\partial x^2} u(x, t) = \frac{\Gamma(3)}{\Gamma(3 - \alpha)} t^{2-\alpha} \sin(2\pi x) + (2\pi)^2 t^2 \sin(2\pi x), \quad (275)$$



with the initial and boundary conditions

$$u(x, 0) = 0, u(0, t) = 0, u(1, t) = 0. \quad (276)$$

We use (270) for equation (275) where

$$\begin{cases} F_i(k) = \frac{\Gamma(3)}{\Gamma(3-\alpha)} \delta(k - (2/\alpha - 1)) \\ \sin(2\pi i h) + (2\pi)^2 \delta(k - 2/\alpha) \sin(2\pi i h), \\ i = 1, 2, \dots, N - 1, \end{cases}$$

and differential equation for initial and boundary conditions (275) is

$$U_i(0) = 0, i = 1, 2, \dots, N - 1, U_0(k) = 0, U_N(k) = 0. \quad (277)$$

The exact solution of (275) is

$$u(x, t) = t^2 \sin(2\pi x).$$

The L^2 -norm of the errors in solutions are tabulated in Table 3, with different values of $\lambda_1 = \frac{1}{6}, \lambda_2 = \frac{4}{6}, \lambda_3 = -2, h = \frac{1}{16}, \tau = \frac{1}{4096}, \alpha = 0.5$ and $T = \frac{1}{8}, \frac{1}{4}, \frac{1}{2}, 1$ and also compared with method [30]. The L^2 -norm and L^∞ -norm of the errors in solutions with different values of $h = \tau = \frac{1}{10}, \frac{1}{20}, \frac{1}{40}, \frac{1}{80}, \frac{1}{160}, \alpha = 0.5, 0.8$ of example 3, are tabulated in Table 4 and 5. Our results compared with the results obtained [27],

Table 19: L^2 errors for $h = \frac{1}{16}$ and $\tau = \frac{1}{4096}$ for example 3.

α	0.5	0.5
T	Our method	[30]
$\frac{1}{8}$	$1.0061e - 7$	$1.9703e - 7$
$\frac{1}{4}$	$3.8261e - 7$	$7.7365e - 7$
$\frac{1}{2}$	$1.5304e - 6$	$3.0519e - 6$
1	$6.1217e - 6$	$1.2083e - 5$

Table 20: L^2 and L^∞ errors with $h = \tau, \alpha = 0.5$ for example 3.

N	L^2 Our method	L^∞ Our method	L^2 [27]	L^∞ [27]
10	5.6889e-4	6.6789e-4	$3.2503e - 4$	$4.3717e - 4$
20	1.5499e-4	1.6670e-4	$9.3315e - 5$	$1.3179e - 4$
40	3.9889e-5	4.2739e-5	$3.1994e - 5$	$4.5247e - 5$
80	1.0499e-5	1.1520e-5	$1.1332e - 5$	$1.6025e - 5$

5 Conclusion

We have developed new non-polynomial spline method for finding the numerical solution of time fractional partial differential equations and we get a class of new methods based on a exponential spline function. we



Table 21: L^2 and L^∞ errors with $h = \tau, \alpha = 0.8$ for example 3.

N	L^2 Our method	L^∞ Our method	$L^2[27]$	$L^\infty[27]$
10	5.6729e-4	2.8829e-4	$9.2939e-4$	$1.2500e-3$
20	1.4270e-4	7.1270e-5	$3.7578e-4$	$5.3143e-4$
40	3.7327e-5	1.8829e-5	$1.6204e-4$	$2.2916e-4$
80	1.0032e-5	4.9221e-6	$7.0496e-5$	$9.9696e-5$
160	2.6320e-6	1.2032e-6	$3.0698e-5$	$4.3413e-5$

Table 22: L^2 and L^∞ errors with $h = \tau, \alpha = 0.2$ for example 3.

N	L^2 Our method	L^∞ Our method	$L^2[27]$	$L^\infty[27]$
20	4.0234e-5	6.0201e-5	$1.9514e-5$	$2.7497e-5$
40	1.0129e-5	1.5121e-5	$4.7354e-6$	$6.6968e-6$
80	2.6896e-6	3.9602e-6	$1.3497e-6$	$1.9088e-6$
160	6.9876e-7	9.8120e-7	$3.9795e-7$	$5.6280e-7$

discretize the problem in time variable using fractional differential transform method and an approximate the space direction by using non-polynomial spline method. Boundary condition and convergence analysis of the methods are discussed. Numerical results are shown that our method is more efficient than the method in [27], [29] and [30].

References

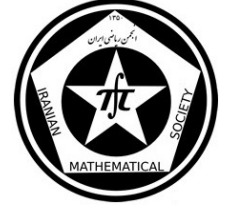
- [1] K. Diethelm, A.D. Freed, *On solution of nonlinear fractional order differential equations used in modelling of viscoplasticity*, *Scientific Computing in Chemical Engineering II: Computational Fluid Dynamics, Reaction Engineering and Molecular Properties*, Springer Verlag, Heidelberg, 1999, pp. 217–224.
- [2] R. Hilfer (Ed.), *Applications of Fractional Calculus in Physics*, World Scientific Publishing, River Edge, NJ, USA, 2000.
- [3] I.M. Sokolov, J. Klafter, A. Blumen, *Fractional kinetics*, *Phys. Today* **55** (11) (2002) 48-54.
- [4] A. Iomin, *Fractional transport of tumor cells*, *WSEAS Trans. Biol. Biomed* **2** (2005) 82-86.
- [5] R. Metzler, J. Klafter, *The random walk's guide to anomalous diffusion: a fractional dynamics approach*, *Phys. Rep.* **339** (1) (2000) 1–77.
- [6] F. Mainardi, *Fractals and Fractional Calculus Continuum Mechanics*, Springer Verlag, 1997, pp. 291-348.
- [7] I.M. Sokolov, J. Klafter, A. Blumen, *Ballistic versus diffusive pair-dispersion in the Richardson regime*, *Phys. Rev. E* **61** (3) (2000) 2717–2722.



- [8] W. Chen, *A speculative study of $\frac{2}{3}$ -order fractional Laplacian modeling of turbulence: some thoughts and conjectures*, *Chaos* **16** (2006) 023126.
- [9] L. Xuhao, Patricia J.Y. Wong, *A higher order non-polynomial spline method for fractional sub-diffusion problems*, *J. Comput. Phys.* **328** (2017) 46–65.
- [10] M. Yaseena, M. Abbas, A. Izani Ismail, T. Nazir, *A cubic trigonometric B-spline collocation approach for the fractional sub-diffusion equations*, *Appl. Math. Comput.* **293** (2017) 311–319.
- [11] M. Meerschaert, C. Tadjeran, *Finite difference approximations for fractional advection-dispersion flow equations*, *J. Comput. Appl. Math.* **172** (2004) 65–77.
- [12] S.B. Yuste, *Weighted average finite difference methods for fractional diffusion equation*, *J. Comput. Phys.* **216** (2006) 264–274.
- [13] T.S. El-Danaf, A.R. Hadhoud, *Parametric spline functions for the solution of the one time fractional Burgers equation*, *Appl. Math. Model.* **36** (2012) 4557–4564.
- [14] S.M. Hosseini, R. Ghaffari, *Polynomial and nonpolynomial spline methods for fractional sub-diffusion equations*, *Appl. Math. Model.* **38** (2014) 3554–3566.
- [15] C. Chen, F. Liu, I. Turner, V. Anh, *A Fourier method for the fractional diffusion equation describing sub-diffusion*, *J. Comput. Phys.* **227** (2007) 886–897.
- [16] M. Cui, *Compact finite difference method for the fractional diffusion equation*, *J. Comput. Phys.* **228** (2009) 7792–7804.
- [17] G. Gao, Z. Sun, *A compact finite difference scheme for the fractional sub-diffusion equations*, *J. Comput. Phys.* **230** (2011) 586–595.
- [18] C. Chen, F. Liu, I. Turner, V. Anh, *Numerical schemes with high spatial accuracy for a variable-order anomalous subdiffusion equation*, *SIAM J. Sci. Comput.* **32** (2010) 1740–1760.
- [19] G. Gao, Z. Sun, *The finite difference approximation for a class of fractional sub-diffusion equations on a space unbounded domain*, *J. Comput. Phys.* **236** (2013) 443–460.
- [20] G. Gao, H. Sun, Z. Sun, *Stability and convergence of finite difference schemes for a class of time-fractional sub-diffusion equations based on certain super convergence*, *J. Comput. Phys.* **280** (2015) 510–528.
- [21] G. Gao, Z. Sun, Y. Zhang, *A finite difference scheme for fractional sub-diffusion equations on an unbounded domain using artificial boundary conditions*, *J. Comput. Phys.* **231** (2012) 2865–2897.
- [22] T. Langlands, B. Henry, *The accuracy and stability of an implicit solution method for the fractional diffusion equation*, *J. Comput. Phys.* **205** (2005) 719–736.
- [23] Y.M. Lin, C.J. Xu, *Finite difference/spectral approximations for the time-fractional diffusion equation*, *J. Comput. Phys.* **225** (2007) 1533–1552.



- [24] A. Mohebbi, M. Abbaszadeh, M. Dehghan, *A high-order and unconditionally stable scheme for the modified anomalous fractional sub-diffusion equation with a nonlinear source term*, J. Comput. Phys. **240** (2013) 36–48.
- [25] D.A. Murio, *Implicit finite difference approximation for time fractional diffusion equations*, Comput. Math. Appl. **56** (2008) 1138–1145.
- [26] Z. Wang, S. Vong, *Compact difference schemes for the modified anomalous fractional sub-diffusion equation and the fractional diffusion-wave equation*, J. Comput. Phys. **277** (2014) 1–15.
- [27] X. Yang, H. Zhang, D. Xu, *Orthogonal spline collocation method for the two-dimensional fractional sub-diffusion equation*, J. Comput. Phys. **256** (2014) 824–837.
- [28] Y. Zhang, Z. Sun, *Alternating direction implicit schemes for the two-dimensional fractional sub-diffusion equation*, J. Comput. Phys. **230** (2011) 8713–8728.
- [29] X. Zhao, Z. Sun, *A box-type scheme for fractional sub-diffusion equation with Neumann boundary conditions*, J. Comput. Phys. **230** (2011) 6061–6074.
- [30] C. Li, Tinggang Zhao, Weihua Deng, Yujiang Wu, *Orthogonal spline collocation methods for the sub-diffusion equation*, J. Comput. Appl. Math. **255** (2014) 517–528.
- [31] K. Al-Khaled, *Numerical solution of time-fractional partial differential equations using sumudu decomposition method*, Romanian journal of physics **60** (2015) 99–110.
- [32] A. Arikglu, I. Ozkol, *Solution of fractional differential equations by using differential transform method*, Solution fractals **34** (2007) 1473–01481.
- [33] W. K. Zahra, S. M. Elkholy, *Cubic spline solution of fractional bagley-torvik equation*, Electron. J. Differential Equations, **1(2)** (2013) 230–241.
- [34] Su X., Zhang S., *Solution to boundary value problem for nonlinear differential equations of fractional order*, Electron. J. Differential Equations, **26** (2009) 1–15.
- [35] M. Van Daele, G. Vanden Berghe and H. De Meyer, *A smooth approximation for the solution of a fourth-order boundary value problem based on nonpolynomial splines*, J. Comput. Appl. Math. **51** (1994) 383–394.
- [36] R. A. Usmani and S. A. Wasrt, *Quintic spline solutions of boundary value problems*, Comput. Math. with Appl., **6256** (1980) 197–203.



Direct meshless local Petrov-Galerkin for the nonlinear coupled Burgers' equation

Ali Habibirad¹⁶, Esmail Hesameddini,

Department of Mathematics, Shiraz University of Technology, Shiraz, Iran

a.habibirad@sutech.ac.ir, hesameddini@sutech.ac.ir

Abstract

The coupled Burgers equations is an important class of partial differential equations. In this work we apply an accurate meshless method based on the generalized moving least squares (GMLS) approximation for numerical solution of the coupled Burgers equations. In this method to discretizes this equation in time and space variables we use finite difference method and GMLS approximation, respectively. Also, we compute local weak form for every point instead of computation of global weak form for the global domain. By using this method, such problems will be reduced to a system of algebraic equations. To show the accuracy of the present method an example is obtained.

AMS subject Classification 2010: 34A45, 65M12, 65M60

Keywords: Coupled Burgers equations, generalized moving least squares, meshless method.

1 Introduction

The coupled Burgers equation is an important class of such Partial Difference Equations (PDEs). Certain cases of this class are solved in [4] by Hopf-Cole transformation. There are many numerical solutions to the Burgers equations for example [5] and reference therein. In this paper we discuss a meshless method to obtain numerical solution of the following nonlinear coupled Burgers equation:

$$\begin{cases} u_t(\mathbf{x}, t) + u(\mathbf{x}, t)u_x(\mathbf{x}, t) + v(\mathbf{x}, t)u_y(\mathbf{x}, t) = \alpha\Delta u(\mathbf{x}, t), \\ v_t(\mathbf{x}, t) + u(\mathbf{x}, t)v_x(\mathbf{x}, t) + v(\mathbf{x}, t)v_y(\mathbf{x}, t) = \beta\Delta v(\mathbf{x}, t), \\ \mathbf{x} = (x, y) \in \Omega \subset \mathbb{R}^2, \quad t \in [0, T], \end{cases} \quad (278)$$

subject to initial conditions

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \quad v(\mathbf{x}, 0) = v_0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (279)$$

and boundary conditions of Dirichlet type

$$u(\mathbf{x}, t) = u_1(\mathbf{x}), \quad v(\mathbf{x}, t) = v_1(\mathbf{x}), \quad \mathbf{x} \in \partial\Omega. \quad (280)$$

¹⁶Speaker



In the above equation T is the final time, α and β are positive viscosity parameters, and they are inverse of Reynolds Re (if $\alpha = \beta$ then $\alpha = \beta = \frac{1}{Re}$). Also, u_0, u_1, v_0 and v_1 are known functions, u and v are the velocity components in 2D. Δ is the Laplacian differential operator. Many researchers discussed about various differential quadrature schemes for the numerical solution of the 2D nonlinear/linear PDEs. Also, meshfree techniques have become a brand issue for solving differential equations, numerically. To display the different between meshfree schemes and mesh-based methods, authors of [1] discussed a meshfree technique based on radial basis function (RBF) and Kansa method for the numerical solution of the time-fractional nonlinear Kline-Gordon and Sine-Gordon equations. One of the famous meshless methods is Meshless Local Petrov-Galerkin (MLPG) scheme [6]. Shokri and Habibirad applied the MLPG method for nonlinear Kline-Gorden equation in [2]. The main benefit of MLPG method is to transform a global approximation to a system of algebraic equations for the numerical solution of a problem. Also, it uses local weak form in each sub-domain instead of a global weak form. There are six types of MLPG techniques that they created by changing the test function which are named MLPG1, MLPG2,..., MLPG6. In [3] Mirzaei and Schaback used a generalized moving least squares (GMLS) approximation instead of moving least squares, to obtain DMLPG methods (DMLPG1, DMLPG2,...,DMLPG6). In this paper, we apply DMLPG2 method for the numerical solution of nonlinear coupled Burgers' equation.

The organization of this work is as follows: In Section 2, we give a brief review of the GMLS. In Section 3, we explain the time discretization and numerical performance of the DMLPG technique for the nonlinear coupled Burgers' equation. A numerical experiment is performed for the mentioned equation in Section 4. Finally, a brief conclusion is given in Section 5.

2 The GMLS scheme

Suppose that $v(\mathbf{x})$ be a function in the global domain $\Omega \subseteq \mathbb{R}^2$. We discretized Ω with some of nodal points $\mathbf{x}_i$, $i = 1, 2, \dots, n$ which surrounding node $\mathbf{p}$ and have the influence on $v(\mathbf{p})$. Using the GMLS technique, $v^h(\mathbf{x})$ an approximate value of $v(\mathbf{x})$ can be written as follows [3]

$$v(\mathbf{x}) \approx v^h(\mathbf{x}) = \sum_{i=1}^N \phi_i(\mathbf{x})v_i = \Phi(\mathbf{x})\mathbf{V}, \quad \mathbf{x} \in \Omega_x, \quad (281)$$

in which

$$\Phi(\mathbf{x}) = [\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \dots, \phi_N(\mathbf{x})], \quad \mathbf{V} = [v(\mathbf{x}_1), v(\mathbf{x}_2), \dots, v(\mathbf{x}_N)]^T. \quad (282)$$

Considering v^h be the best approximation for the smooth function v , the MLS shape functions $\phi_i(\mathbf{x})$ will be obtained. Also, suppose $W = W(\mathbf{x}) = \text{diag}\{w(\mathbf{x}_j, \mathbf{x})\} \in \mathbb{R}^{N \times N}$. In this paper, we choice the following weight function

$$w(\mathbf{x}, \mathbf{x}_i) = \begin{cases} 1 - 6d_i^2 + 8d_i^3 - 3d_i^4, & d_i \leq 1, \\ 0, & d_i > 1, \end{cases} \quad (283)$$

where $d_i = \frac{\|\mathbf{x} - \mathbf{x}_i\|}{r_i}$, in which r_i is the size of support for the weight function. The shape functions $\Phi(\mathbf{x})$ are obtained as



$$\Phi(\mathbf{x}) = p(\mathbf{x})H(\mathbf{x}), \quad (284)$$

in which $H(\mathbf{x}) = (P^T W(\mathbf{x})P)^{-1} P^T W(\mathbf{x})$. Here $\mathbf{p}(\mathbf{x})$ is a vector of the m basis functions as follows

$$\mathbf{p}^T(\mathbf{x}) = [p_1(\mathbf{x}) \cdots p_m(\mathbf{x})], \quad (285)$$

, we consider cube polynomial $p^T(\mathbf{x}) = [1, x, y, xy, x^2, y^2]$. Moreover

$$P = \begin{pmatrix} p_1(\mathbf{x}_1) & \cdots & p_m(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ p_1(\mathbf{x}_N) & \cdots & p_m(\mathbf{x}_N) \end{pmatrix}, \quad (286)$$

in which $m = 6$. In GMLS approximation, we approximate directly $\lambda(v)$ from the nodal values $v(\mathbf{x}_1), v(\mathbf{x}_2), \dots, v(\mathbf{x}_N)$. This method is different from the classical MLS approximation in which v is approximated first by v^h and then λ acts on v^h . Indeed in GMLS, we have

$$\lambda(v) \approx \lambda(v^h) = \sum_{i=1}^N \phi_i(\lambda) v(\mathbf{x}_i), \quad (287)$$

where $\phi_i(\lambda)$ are the shape functions related to the given functional $\phi_i(\lambda)$. Setting $v = y_i, i = 1, 2, \dots, k$, the approximate relation (287) will be transferred to the exact solution. Using [3] and (284), one obtains $\Phi(\lambda) = [\phi_1(\lambda), \phi_2(\lambda), \dots, \phi_N(\lambda)] = \lambda(y)(P^T W P)^{-1} P^T W$, where $\lambda(y) = [\lambda(y_1), \lambda(y_2), \dots, \lambda(y_k)]$. Therefore, $\lambda(v)$ can be approximated directly by the nodal values $v(\mathbf{x}_1), v(\mathbf{x}_2), \dots, v(\mathbf{x}_N)$ without any additional computations for the shape functions. For more details about the GMLS argument see [3].

3 Numerical implementation

In this section, we propose a meshfree scheme based on the GMLS method for the numerical solution of Eq. (689). At first, we generate a fully discrete scheme for Eq. (689) in the time variable. So, we suppose $\Delta t = T/n$ as the time step size and $t_k = k\Delta t, k = 0, 1, 2, \dots, n$ are the time discretization points where n is an arbitrary positive integer. Now, to discretize the time derivative we use the following backward finite difference relations

$$u_t(\mathbf{x}, t) \simeq \frac{u^{k+1}(\mathbf{x}) - u^k(\mathbf{x})}{\Delta t}, \quad \Delta u(\mathbf{x}) = \frac{\Delta u^{k+1}(\mathbf{x}) + \Delta u^k(\mathbf{x})}{2}, \quad (288)$$

and the same relations for v and its derivative. Substituting these relations in (689)

$$\begin{cases} \frac{u^{k+1}(\mathbf{x}) - u^k(\mathbf{x})}{\Delta t} + u^k(\mathbf{x})u_x^k(\mathbf{x}) + v^k(\mathbf{x})u_y^k(\mathbf{x}) = \alpha \frac{\Delta u^{k+1}(\mathbf{x}) + \Delta u^k(\mathbf{x})}{2}, \\ \frac{v^{k+1}(\mathbf{x}) - v^k(\mathbf{x})}{\Delta t} + u^k(\mathbf{x})v_x^k(\mathbf{x}) + v^k(\mathbf{x})v_y^k(\mathbf{x}) = \beta \frac{\Delta v^{k+1}(\mathbf{x}) + \Delta v^k(\mathbf{x})}{2}, \end{cases} \quad (289)$$

and simplifying this relation one obtains

$$\begin{cases} u^{k+1}(\mathbf{x}) - 0.5\alpha\Delta t\Delta u^{k+1}(\mathbf{x}) = u^k(\mathbf{x}) - \Delta t u^k(\mathbf{x})u_x^k(\mathbf{x}) - \Delta t v^k(\mathbf{x})u_y^k(\mathbf{x}) + 0.5\alpha\Delta t\Delta u^k(\mathbf{x}), \\ v^{k+1}(\mathbf{x}) - 0.5\beta\Delta t\Delta v^{k+1}(\mathbf{x}) = v^k(\mathbf{x}) - \Delta t u^k(\mathbf{x})v_x^k(\mathbf{x}) - \Delta t v^k(\mathbf{x})v_y^k(\mathbf{x}) + 0.5\beta\Delta t\Delta v^k(\mathbf{x}). \end{cases} \quad (290)$$



Now, we discretize the space variables by using GMLS approximation. For this propose, we assume, $\Omega \subseteq \mathbb{R}^2$ be the global domain and $\mathbf{X}^* = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n\}$ be an arbitrary scattered points in this domain. In this work, instead of calculation the global weak form, we introduce weak form over a local subdomain such as $\Omega_{\mathbf{x}}$ which is a small region over each node in the general domain Ω . These subdomains can have any desired geometric shapes and size [7]. They overlap and cover the entire domain Ω . We assume they are circular in 2D. Now, for any random point $\mathbf{x} \in \mathbf{X}^*$ we introduce the local weak form of (689) in dependent subdomain $\Omega_{\mathbf{x}} \subset \Omega$. Assuming only N points ($N \leq m$) in the neighborhood corresponding node $\mathbf{x}$ and effect on the numerical solution, result in

$$\begin{aligned}\Delta u(\mathbf{x}, t) &= \sum_{j=1}^N \Delta \phi_j(\mathbf{x}) \hat{u}_j = \Delta \Phi(\mathbf{x}) \mathbf{u}(t), \quad \mathbf{x} \in \Omega_{\mathbf{x}}, \\ u_x(\mathbf{x}, t) &= \sum_{j=1}^N \frac{\partial \phi_j(\mathbf{x})}{\partial x} \hat{u}_j = \Phi_x(\mathbf{x}) \mathbf{u}(t), \quad \mathbf{x} \in \Omega_{\mathbf{x}}, \\ u_y(\mathbf{x}, t) &= \sum_{j=1}^N \frac{\partial \phi_j(\mathbf{x})}{\partial y} \hat{u}_j = \Phi_y(\mathbf{x}) \mathbf{u}(t), \quad \mathbf{x} \in \Omega_{\mathbf{x}}, \\ u(\mathbf{x}, t) &= \sum_{j=1}^N \phi_j(\mathbf{x}) \hat{u}_j = \Phi(\mathbf{x}) \mathbf{u}(t), \quad \mathbf{x} \in \Omega_{\mathbf{x}}.\end{aligned}$$

Also, the same relations will be used for v and its derivatives. Substituting the above GMLS equations in (394) and doing this process for all nodes, one arrive at the following matrix form equation

$$CU^{k+1} = AU^k - \mathfrak{d}tB^k, \quad (291)$$

where

$$C = \begin{pmatrix} [C_1]_{N \times N} & 0_{N \times N} \\ 0_{N \times N} & [C_2]_{N \times N} \end{pmatrix}_{2N \times 2N}, \quad A = \begin{pmatrix} [A_1]_{N \times N} & 0_{N \times N} \\ 0_{N \times N} & [A_2]_{N \times N} \end{pmatrix}_{2N \times 2N}, \quad B = ([B_1]_{N \times 1} [B_2]_{N \times 1})^T, \quad (292)$$

in which

$$\begin{aligned}(C_1)_{ij} &= \phi_i(\mathbf{x}_j) - 0.5\alpha \mathfrak{d}t \Delta \phi_i(\mathbf{x}_j), & (C_2)_{ij} &= \phi_i(\mathbf{x}_j) - 0.5\beta \mathfrak{d}t \Delta \phi_i(\mathbf{x}_j), \\ (A_1)_{ij} &= \phi_i(\mathbf{x}_j) + 0.5\alpha \mathfrak{d}t \Delta \phi_i(\mathbf{x}_j), & (A_2)_{ij} &= \phi_i(\mathbf{x}_j) + 0.5\beta \mathfrak{d}t \Delta \phi_i(\mathbf{x}_j), \\ (B_1)_i &= u^k(\mathbf{x}_i) u_x^k(\mathbf{x}_i) + v^k(\mathbf{x}_i) u_y^k(\mathbf{x}_i), & (B_2)_i &= u^k(\mathbf{x}_i) v_x^k(\mathbf{x}_i) + v^k(\mathbf{x}_i) v_y^k(\mathbf{x}_i).\end{aligned}$$

4 example

In this section, we consider an example to illustrate the validity and capability of this scheme. To show the accuracy of the proposed method, the L_∞ error is considered as follows

$$L_\infty = \max_{1 \leq i \leq N} |u(\mathbf{x}_i) - u^h(\mathbf{x}_i)|, \quad (293)$$

where $u(\mathbf{x}_i)$ and $u^h(\mathbf{x}_i)$ are the exact and numerical solutions at node $\mathbf{x}_i$.



Table 23: The L_∞ errors between the presented method and analytical solutions of this Example

T	$u(\mathbf{x}, t)$	$v(\mathbf{x}, t)$	CPU-time
1	$9.8986e - 06$	$6.4721e - 06$	3
3	$5.1808e - 06$	$3.1908e - 06$	5
5	$8.3626e - 06$	$7.3456e - 06$	7

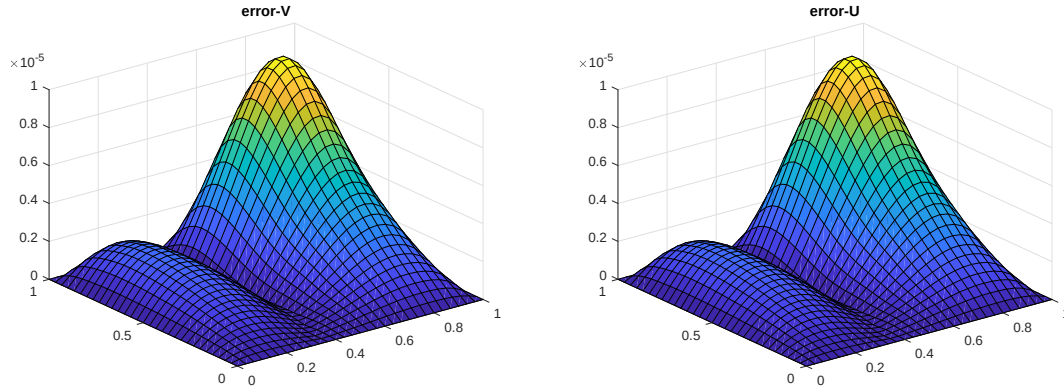


Figure 17: The absolute error for this Example at $T = 1$.

Suppose the nonlinear coupled Burgers equation (689) with $\alpha = \beta = 1/Re$ with the following exact solution

$$u(\mathbf{x}, t) = 0.25 \left(3 - \frac{1}{1 + \exp((-4x + 4y - t)\frac{Re}{32})} \right), \quad v(\mathbf{x}, t) = 0.25 \left(3 + \frac{1}{1 + \exp((-4x + 4y - t)\frac{Re}{32})} \right),$$

where $\Omega = [0, 1] \times [0, 1]$ and $\Delta t = 0.01$. The initials and boundary conditions will be obtained from the exact solution. We choose 31×31 uniform nodes in Ω . The first column of Table 54 is final time and columns two and three are L_∞ errors of the presented method for solving this example. The last column is CPU-time. Figure 17 shows the absolute error at $T = 1$ for the presented scheme. This table and Figure 17 reveal that our scheme is accurate and efficient for solving this problem.

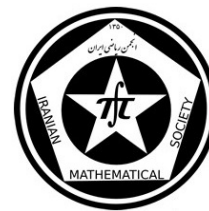
5 Conclusion

The nonlinear coupled Burgers equation is one of the most important mathematical models that have been considered by many researchers in recent years. The numerical solution of these equations is of great importance. In this work, we presented a meshless scheme for the numerical solution of this equation. We used GMLS approximation in this method to estimate this equation in space variables. Also, we used finite difference relations to approximate the time derivative. The capability and accuracy of this method were shown through an example.



References

- [1] M. Dehghan, M. Abbaszadeh, and A. Mohebbi, “An implicit RBF meshless approach for solving the time fractional nonlinear sine-gordon and klein–gordon equations,” *Engineering Analysis with Boundary Elements*, vol. 50, pp. 412–434, 2015.
- [2] A. Shokri and A. Habibirad, “A moving kriging-based MLPG method for nonlinear Klein–Gordon equation,” *Mathematical Methods in the Applied Sciences*, vol. 39, no. 18, pp. 5381–5394, 2016.
- [3] D. Mirzaei and R. Schaback, “Direct Meshless Local Petrov–Galerkin (DMLPG) method: a generalized MLS approximation,” *Applied Numerical Mathematics*, vol. 68, pp. 73–82, 2013.
- [4] C. A. Fletcher, “Generating exact solutions of the two-dimensional Burgers’ equations,” *International Journal for Numerical Methods in Fluids*, vol. 3, pp. 213–216, 1983.
- [5] A. Jafarabadi and E. Shivanian, “Numerical simulation of nonlinear coupled Burgers equation through meshless radial point interpolation method,” *Engineering Analysis with Boundary Elements*, vol. 95, pp. 187–199, 2018.
- [6] Habibirad A, Hesameddini, Heydari M H and Roohi R *An efficient meshless method based on the Moving Kriging interpolation for two-dimensional variable-order time fractional mobile/immobile advection-diffusion model*. Mathematical Methods in the Applied Sciences, 2020.
- [7] S. N. Atluri and T. Zhu, “A new meshless local petrov-galerkin (mlpg) approach in computational mechanics,” *Computational mechanics*, vol. 22, no. 2, pp. 117–127, 1998.



Existence and multiplicity of solutions to the nano boundary layer flows with nonlinear Navier boundary condition

Elyas Shivanian¹⁷

Department of Applied Mathematics, Imam Khomeini International University, Qazvin, 34149-16818, Iran
shivanian@sci.ikiu.ac.ir

Abstract

In the present work, the classical laminar boundary layer equation of the flow away from the origin past a wedge with the no-slip boundary condition replaced by a nonlinear Navier boundary condition is revisited. This boundary condition includes an arbitrary index parameter, denoted by $n > 0$, which appears in the coefficients of the differential equation to be solved. It is proved corresponding to the value $n = \frac{1}{3}$, there are exactly three situations for the problem: (i) there is no solution; (ii) there exist two solutions; (iii) there exist four solutions. Furthermore, the exact analytical solution of the problem is given in terms of parabolic cylinder functions for further physical interpretations.

AMS subject Classification 2010: 34B15; 34B40.

Keywords: Exact analytic solution; Parabolic cylinder function; Existence; Multiplicity.

1 Introduction and problem formulation

Consider the steady two-dimensional boundary layer equations

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \quad (294)$$

$$u \frac{\partial u}{\partial x} + v \frac{\partial v}{\partial y} = U \frac{dU}{dx} + \frac{\partial^2 u}{\partial y^2}, \quad (295)$$

subject to the boundary conditions

$$|u| = l \left(\left| \frac{\partial u}{\partial y} \right| \right)^n \quad \text{and} \quad v = 0 \quad \text{at} \quad y = 0, \quad (296)$$

$$u \equiv U(x) = ax^m \quad \text{as} \quad y \rightarrow +\infty, \quad (297)$$

¹⁷Speaker



where x and y are the dimensionless Cartesian coordinates measured along the plate and normal to it. u and v are the velocity components along the x and y axes. U is a given external inviscid velocity field. The parameters a , m and n are constants. The case $a > 0$ is of main interest when describing flows away from the origin, and $a < 0$ when the external stream flows towards the origin (for more details see [5, 2]). The more interesting problems arising out of finding a solution of boundary layer equation is when the investigation of the conditions under which two solutions are similar. Matthews and Hill [3, 7] introduced the following similarity transformation, by using Lie symmetries analysis

$$\eta = x^{-\frac{n-1}{3n-2}} y, \quad (298)$$

and the stream function $f(\eta)$ defined by

$$\frac{u}{U} = \frac{1}{a} f', \quad x^{\frac{2n-1}{3n-2}} \frac{v}{U} = \frac{1}{a} \left(\frac{n-1}{3n-2} \eta f' - \frac{2n-1}{3n-2} \eta f \right), \quad (299)$$

with $m = \frac{n}{3n-2}$ and $n \neq \frac{3}{2}$. By these transformations, Eqs. (294) and (295) can be written as

$$f''' + \frac{2n-1}{3n-2} f f'' - \frac{n}{3n-2} (f'^2 - a^2) = 0, \quad (300)$$

where the prime denotes differentiation with respect to η . The boundary conditions depend on the value of n . For $n \neq \frac{1}{2}$, the boundary conditions are

$$f = 0 \quad \text{and} \quad |f'| = l (|f''|)^n \quad \text{at} \quad \eta = 0, \quad (301)$$

$$f' \rightarrow a \quad \text{as} \quad \eta \rightarrow +\infty. \quad (302)$$

A further simplification shows the parameter a could be removed from the governing equation and the second boundary condition (302) if η and l are multiplied by $\sqrt{|a|}$ and $|a|^{\frac{3n-2}{2}}$, respectively; and f is multiplied by $\frac{1}{\sqrt{a}}$ for $a > 0$. Therefore, the differential equation of the flow away from the origin past a wedge is converted to

$$f''' + \frac{2n-1}{3n-2} f f'' - \frac{n}{3n-2} (f'^2 - 1) = 0, \quad (303)$$

with the boundary condition

$$f(0) = 0, \quad |f'(0)| = l (|f''(0)|)^n, \quad f'(+\infty) = 1. \quad (304)$$

In the next sections, we consider the above boundary value problem when $n = \frac{1}{3}$. In this case, Matthews and Hill [3, 4] have shown that the solutions could be non-unique.

2 The exact analytical solution

Assuming $n = \frac{1}{3}$, the problem (303) and (304) is converted to the following one

$$f''' + \frac{1}{3} f f'' + \frac{1}{3} (f'^2 - 1) = 0, \quad (305)$$



with the boundary condition

$$f(0) = 0, \quad |f'(0)| = l\sqrt[3]{|f''(0)|}, \quad f'(+\infty) = 1. \quad (306)$$

One easily sees that (305) admits the first integral

$$f'' + \frac{1}{3}ff' - \frac{1}{3}\eta = C, \quad (307)$$

where C is the integral constant. Using the first boundary condition (306), Eq. (307) yields

$$C = f''(0) = \gamma, \quad (308)$$

Equation (307) still admits the integral, then using notation (308), we obtain

$$f' + \frac{1}{6}f^2 - \frac{1}{6}\eta^2 = \gamma\eta + D, \quad (309)$$

where D is the integral constant. Again using the first boundary condition (306), Eq. (309) gives

$$D = f'(0) = \alpha, \quad (310)$$

Therefore, Eq. (309) is converted to

$$f' + \frac{1}{6}f^2 - \frac{1}{6}\eta^2 = \gamma\eta + \alpha. \quad (311)$$

The above equation is a kind of well-known Riccati equation. Any Riccati equation can be transformed to a second-order linear ODE. If the latter can be solved explicitly, then a solution for the Riccati equation can be derived. The general solution of Eq. (311) with the condition $f(0) = 0$ is

$$f(\eta) = \frac{\begin{pmatrix} -6i\text{ParabolicCylinderD}\left(\mu, \frac{i(3\gamma+\eta)}{\sqrt{3}}\right) \left(\sqrt{3}\gamma\text{ParabolicCylinderD}\left(\psi, \sqrt{3}\gamma\right) - 2\text{ParabolicCylinderD}\left(\xi, \sqrt{3}\gamma\right)\right) \\ -6\text{ParabolicCylinderD}\left(\xi, \frac{3\gamma+\eta}{\sqrt{3}}\right) \left(2i\text{ParabolicCylinderD}\left(\mu, i\sqrt{3}\gamma\right) + \sqrt{3}\gamma\text{ParabolicCylinderD}\left(\phi, i\sqrt{3}\gamma\right)\right) \\ + (3\gamma + \eta)\text{ParabolicCylinderD}\left(\psi, \frac{3\gamma+\eta}{\sqrt{3}}\right) \left(2i\sqrt{3}\text{ParabolicCylinderD}\left(\mu, i\sqrt{3}\gamma\right) + 3\gamma\text{ParabolicCylinderD}\left(\phi, i\sqrt{3}\gamma\right)\right) \\ - (3\gamma + \eta)\text{ParabolicCylinderD}\left(\phi, \frac{i(3\gamma+\eta)}{\sqrt{3}}\right) \left(3\gamma\text{ParabolicCylinderD}\left(\psi, \sqrt{3}\gamma\right) - 2\sqrt{3}\text{ParabolicCylinderD}\left(\xi, \sqrt{3}\gamma\right)\right) \end{pmatrix}}{\begin{pmatrix} \text{ParabolicCylinderD}\left(\psi, \frac{3\gamma+\eta}{\sqrt{3}}\right) \left(2i\sqrt{3}\text{ParabolicCylinderD}\left(\mu, i\sqrt{3}\gamma\right) + 3\gamma\text{ParabolicCylinderD}\left(\phi, i\sqrt{3}\gamma\right)\right) \\ + \text{ParabolicCylinderD}\left(\phi, \frac{i(3\gamma+\eta)}{\sqrt{3}}\right) \left(3\gamma\text{ParabolicCylinderD}\left(\psi, \sqrt{3}\gamma\right) - 2\sqrt{3}\text{ParabolicCylinderD}\left(\xi, \sqrt{3}\gamma\right)\right) \end{pmatrix}}, \quad (312)$$

where

$$\phi = \frac{1}{4}(2\alpha - 3\gamma^2 - 2), \quad (313)$$

$$\psi = \frac{1}{4}(-2\alpha + 3\gamma^2 - 2), \quad (314)$$

$$\xi = \frac{1}{4}(-2\alpha + 3\gamma^2 + 2), \quad (315)$$

$$\mu = \frac{1}{4}(2\alpha - 3\gamma^2 + 2), \quad (316)$$



and ParabolicCylinderD is so-called parabolic cylinder function which is related to the Hermite functions defining as

$$\text{ParabolicCylinderD}(v, z) = 2^{-\frac{v}{2}} \exp\left(-\left(\frac{z}{2}\right)^2\right) H_v\left(\frac{z}{\sqrt{2}}\right). \quad (317)$$

The solution (312) still contains two parameters α and β which should be determined in the next section.

3 Existence results and discussions

Using the second boundary condition (306) and taking in to account Eqs. (308) and (310), we obtain

$$\gamma = \pm \frac{1}{l^3} |\alpha|^3. \quad (318)$$

Setting above equation into (311) yields

$$f' + \frac{1}{6} f^2 - \frac{1}{6} \eta^2 = \pm \frac{1}{l^3} |\alpha|^3 \eta + \alpha. \quad (319)$$

Now, if we consider the third condition in (306) i.e. $f'(+\infty) = 1$, then Eq. (319) gives the following equation in infinity

$$\lim_{\eta \rightarrow +\infty} f(\eta) = \sqrt{\eta^2 \pm \frac{6}{l^3} |\alpha|^3 \eta + 6\alpha - 6}. \quad (320)$$

On the other hand, Eq. (307) using Eqs. (308) and (318) becomes

$$f'' + \frac{1}{3} f f' - \frac{1}{3} \eta = \pm \frac{1}{l^3} |\alpha|^3, \quad (321)$$

since $f'(+\infty) = 1$ then $f''(+\infty) = 0$, now using these facts Eq. (321) is converted to an equation in infinity as follows

$$\lim_{\eta \rightarrow +\infty} f(\eta) = \eta \pm \frac{3}{l^3} |\alpha|^3. \quad (322)$$

Finally, Eqs. (320) and (322) with together gives the following relation between α and l

$$\frac{3}{2} \alpha^6 = (\alpha - 1) l^6. \quad (323)$$

Moreover, later equation and Eq. (318) yields a relation between between γ and l as follows

$$\frac{3}{2} |\gamma|^2 = l^3 \sqrt{|\gamma|} - 1. \quad (324)$$

By using Eqs. (323) and (324), α and γ as function of constant slip length l have been plotted in Figures 1 and 2, respectively. An inspection of these Figures emphasizes three remarkable features, namely:

1. Corresponding to the critical value $l = l^* = 1.6788998099469905$, as indicated in Figures, we have $f'(0) = 1.1999999170030182$ and $f''(0) = \pm 0.3651483778986363$ which concludes there exist two solutions.



2. For each value l belongs to the interval $(0, l^*)$, the problem does not admit any solution.
3. For a given l in the interval $(l^*, +\infty)$, there are corresponding two $f'(0)$, and for each $f'(0)$ there are corresponding two $f''(0)$ from the Eq. (318). Therefore, there exist four solutions for the problem (305)-(306) as indicated in Figure 2.

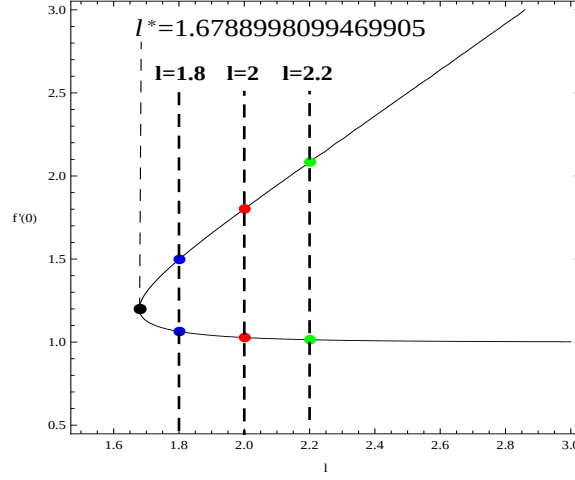


Fig. 1: Diagram of $\alpha = f'(0)$ as function of l .

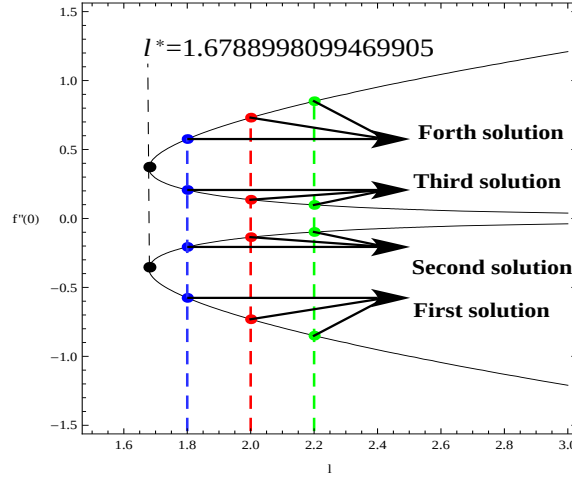


Fig. 2: Diagram of $\gamma = f''(0)$ as function of l .

We mention here that after obtaining the parameters α and γ from Eqs. (323) and (324) for a given l , the solution is given by (312). In Figure 3, we have shown the solutions $f'(\eta)$ and $f(\eta)$ for $l = 1.8$, $l = 2$ and $l = 2.2$ as indicated in Figures 1 and 2 with the colours blue, red and green, respectively.

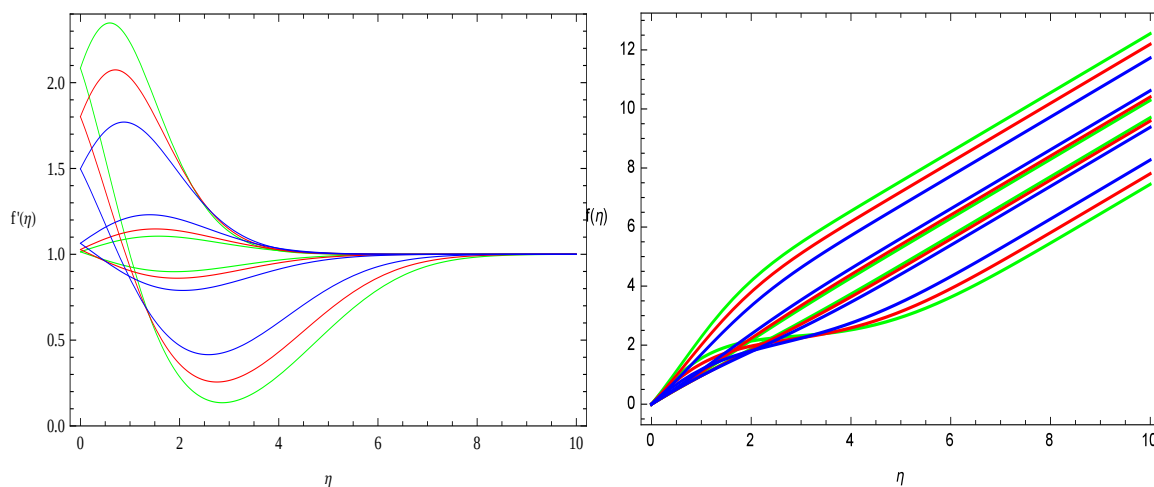
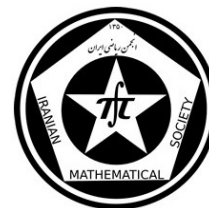


Fig. 3: Plot of $f'(\eta)$ and $f(\eta)$ for three different l namely $l = 1.8$, $l = 2$ and $l = 2.2$.

References

- [1] H. Schlichting, *Boundary Layer Theory*, McGraw-Hill, New York, 1979.
- [2] J. Cheng and S. Liao and R. Mohapatra and K. Vajravelu, *Series solutions of nano boundary layer flows by means of the homotopy analysis method*, J. Math. Anal. Appl. **343** (2008) 233245.
- [3] M. Matthews and J. Hill, *Nano boundary layer equation with nonlinear Navier boundary condition*, J. Math. Anal. Appl. **333** (2007) 381400.
- [4] M. Matthews and J. Hill, *A note on the boundary layer equations with linear slip boundary layer condition*, Appl. Math. Lett. **21** (2008) 810813.
- [5] E. Shivanian, *Existence results for nano boundary layer flows with nonlinear Navier boundary condition*, Phys. Lett. A **377** (41) (2013) 2950-2954



Generalization of Sarrus rule

M.Z. Kazemi Baneh^{†18}, S. Haidarian[†],

zaherkazemi@uok.ac.ir[‡]

sedigh.haidarian@gmail.com[†]

Abstract

We extend the Sarrus rule to calculate the determinant of a square matrix of degrees higher than three by permutation groups.

AMS subject Classification 2010: 65F40.

Keywords: Determinant, Permutation, Sarrus rule.

1 Introduction and Preliminaries

Like function derivatives, matrix determinant in most of the cases can give us good information about the matrix itself and its characteristic. The concept of determinant play a prominent role in linear algebra, geometry and analysis. Determinant could be useful in the invertibility of a matrix, the solution of system of n -equations n -variables, in calculating the triangle area and vector made parallelogram, for volume of tetrahedron and parallel-piped made of 3 vectors. For a piece of a kD subspace of $\mathbb{R}^n$, how to calculate the content of the kD ? With the help of the determinant vector outer products in $\mathbb{R}^3$, mathematicians produce these concepts for 3D and the arrive to a common definition of length, area volume and again with the help of the determinant of a $n \times n$ matrix, they generalize the definitions of nD , Right-handedness left-handedness of an ordered n tuple of vectors to the nD space of $\mathbb{R}^n$.

We all know that the determinant has useful features and its application appears in various branches of mathematics. Some of its applications could be found in the following discussions : proving the inequalities, algebraic identities, decomposition of symmetric algebraic expressions, roots of 2nd 3rd grade equation in algebra, divisibility on prime numbers, the solution of some of number theories, and specification of the convexity of functions. The determinant give the ability to us to determine the Wronskian of functions, differential equations of curves and also the type of extremum points or saddle points in functions several variables.

In this paper we generalize the Sarrus rule to calculate the square matrix determinant of bigger than 3 by permutation groups, while in most of the linear algebra textbooks, its clearly claimed that the Sarrus rule could only applied for square matrix determinant of 3.

¹⁸Speaker



For more clarifications we will first discuss this rule for matrices of 2×2 and 3×3 and then we will generalize it for the dets of higher than 3. We have:

Theorem 0.54. 1 Let A be a square-order matrix with rank n and S_n the permutations group of order n . Then

$$\det(A) = \sum_{\varphi \in S_n} (\text{Sgn}(\varphi)) a_{1\varphi(1)} a_{2\varphi(2)} \dots a_{n\varphi(n)}.$$

Proof. It can be verified using induction. For square matrices, rank 2, 3, 4, and 5 will be calculated. $\square$

We know that the 2×2 matrix det is calculated as factorial elements on principal diagonal minus total sum of elements on the secondary diagonal. Which is :

$$\begin{vmatrix} + & - \\ a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21}$$

First we will place another copy of the det on the right side of the main det

$$\begin{vmatrix} + & + & + & - & - & - \\ a_{11} & a_{12} & a_{13} & a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} & a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} & a_{31} & a_{32} & 6a_{33} \end{vmatrix}$$

$$\det(A) = (a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32}) - (a_{11}a_{23}a_{32} + a_{12}a_{21}a_{33} + a_{13}a_{22}a_{31}) = \sum_{\varphi \in S_3} (\text{Sgn}(\varphi)) a_{1\varphi(1)} a_{2\varphi(2)} a_{3\varphi(3)}.$$

2 Generalization of Sarrus for calculating the Det 4×4

Now in this stage we calculate the product of entries of main diagonal and three parallel lines with the sing of diagonal. We repeat for the secondary diagonal and three parallel lines. The result is called A_1 . Then we have:

$$\begin{vmatrix} + & - & + & - & - & + & - & + \\ a_{11} & a_{12} & a_{13} & a_{14} & a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix} = A_1$$



$$\begin{aligned}
A_1 = & a_{11}a_{22}a_{33}a_{44} - a_{12}a_{23}a_{34}a_{41} + a_{13}a_{24}a_{31}a_{41} \\
& - a_{14}a_{21}a_{32}a_{43} - a_{11}a_{24}a_{33}a_{42} \\
& + a_{12}a_{21}a_{34}a_{43} - a_{13}a_{22}a_{31}a_{44} + a_{14}a_{23}a_{32}a_{41}
\end{aligned}$$

In the second stage, one of the determinant lines for example because of making work easier fixed the fourth line, and the next three lines which means lines 1,2,3 We give permutations to each other. Which means the place of first lines will exchange with second one, second one with third one and the third one with the first one again. This operation does not affect the relative positions of other rows and columns (In this step for relocation of lines, we set the permutation or the round for the collection of 1, 2, 3 and 4. Which is a round with odd length, then because of this the round itself and permutation is a pair, so the sign of determinant will not change.) And then we will perform all the first step operations as below for this step. The result is called A2.

$$\begin{vmatrix}
+ & - & + & - & - & + & - & + \\
a_{31} & a_{32} & a_{33} & a_{34} & a_{31} & a_{32} & a_{33} & a_{34} \\
a_{11} & a_{12} & a_{13} & a_{14} & a_{11} & a_{12} & a_{13} & a_{14} \\
a_{21} & a_{22} & a_{23} & a_{24} & a_{21} & a_{22} & a_{23} & a_{24} \\
a_{41} & a_{42} & a_{43} & a_{44} & a_{41} & a_{42} & a_{43} & a_{44}
\end{vmatrix} = A_2$$

$$\begin{aligned}
A_2 = & a_{12}a_{23}a_{31}a_{44} - a_{13}a_{24}a_{32}a_{41} + a_{11}a_{22}a_{34}a_{43} \\
& - a_{14}a_{21}a_{33}a_{42} - a_{14}a_{23}a_{31}a_{42} \\
& + a_{11}a_{24}a_{32}a_{43} - a_{12}a_{21}a_{33}a_{44} + a_{13}a_{22}a_{34}a_{41}
\end{aligned}$$

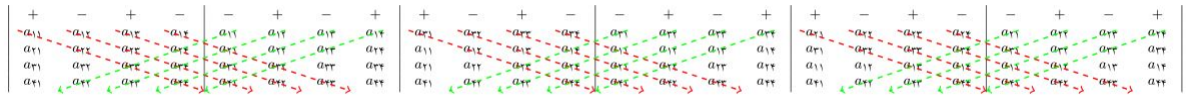
. Now, in the third stage, we still hold the fourth row of the current determinism, the second-order determinism, and re-position the next three lines That is, in A2, we repeat the same, this means that in the determinant A2, we put the first row in place of the second row and the second row in place of the third row and the third row in place of the first row. It's like replacing the first row with the third row and the third row with the second row and replacing the second row with the first line in the initial determinant. In this case I consider permutations or rounds (1, 2, 3) that are pairwise to get a definition like A3. For A3, we repeat the steps in the previous steps as follows.

$$\begin{vmatrix}
+ & - & + & - & - & + & - & + \\
a_{21} & a_{22} & a_{23} & a_{24} & a_{21} & a_{22} & a_{23} & a_{24} \\
a_{31} & a_{32} & a_{33} & a_{34} & a_{31} & a_{32} & a_{33} & a_{34} \\
a_{11} & a_{12} & a_{13} & a_{14} & a_{11} & a_{12} & a_{13} & a_{14} \\
a_{41} & a_{42} & a_{43} & a_{44} & a_{41} & a_{42} & a_{43} & a_{44}
\end{vmatrix} = A_3$$



$$\begin{aligned}
 A_3 = & a_{13}a_{21}a_{32}a_{44} - a_{14}a_{22}a_{33}a_{41} + a_{11}a_{23}a_{34}a_{42} \\
 & - a_{12}a_{24}a_{31}a_{43} - a_{13}a_{21}a_{34}a_{42} \\
 & + a_{14}a_{22}a_{31}a_{43} - a_{11}a_{23}a_{32}a_{44} + a_{12}a_{24}a_{33}a_{41}
 \end{aligned}$$

In the final step, we combine the three values obtained in the first, second and third steps. Therefore, the value of the determinant A is equal to A1+A2+A3. Consequently, if the matrix forms the determinants A1, A2 and A3 from left to right, all steps can be done one by one in the form below Where S_4 is the group of permutations of degree 4.



$$\begin{aligned}
 \det(A) = & (a_{11}a_{22}a_{33}a_{44} + a_{11}a_{23}a_{34}a_{42} + a_{11}a_{24}a_{32}a_{43} + a_{12}a_{21}a_{34}a_{43} \\
 & + a_{12}a_{24}a_{33}a_{41} + a_{12}a_{23}a_{31}a_{44} + a_{13}a_{24}a_{31}a_{42} + a_{13}a_{21}a_{32}a_{44} + a_{13}a_{22}a_{34}a_{41} \\
 & + a_{14}a_{23}a_{32}a_{41} + a_{14}a_{22}a_{31}a_{43} + a_{14}a_{21}a_{33}a_{42}) - (a_{11}a_{24}a_{33}a_{42} + a_{11}a_{23}a_{32}a_{44} \\
 & + a_{11}a_{22}a_{34}a_{43} + a_{12}a_{23}a_{34}a_{41} + a_{12}a_{24}a_{31}a_{43} + a_{12}a_{21}a_{33}a_{44} + a_{13}a_{22}a_{31}a_{44} \\
 & + a_{13}a_{21}a_{34}a_{42} + a_{13}a_{24}a_{32}a_{41} + a_{14}a_{21}a_{32}a_{43} + a_{14}a_{22}a_{33}a_{41} + a_{14}a_{23}a_{31}a_{42}) \\
 \implies \det(A) = & \sum_{\varphi \in S_4} (Sgn(\varphi)) a_{1\varphi(1)} a_{2\varphi(2)} a_{3\varphi(3)} a_{4\varphi(4)} .
 \end{aligned}$$

3 The generalization of the Sarrus rule to calculate the 5×5 determinants

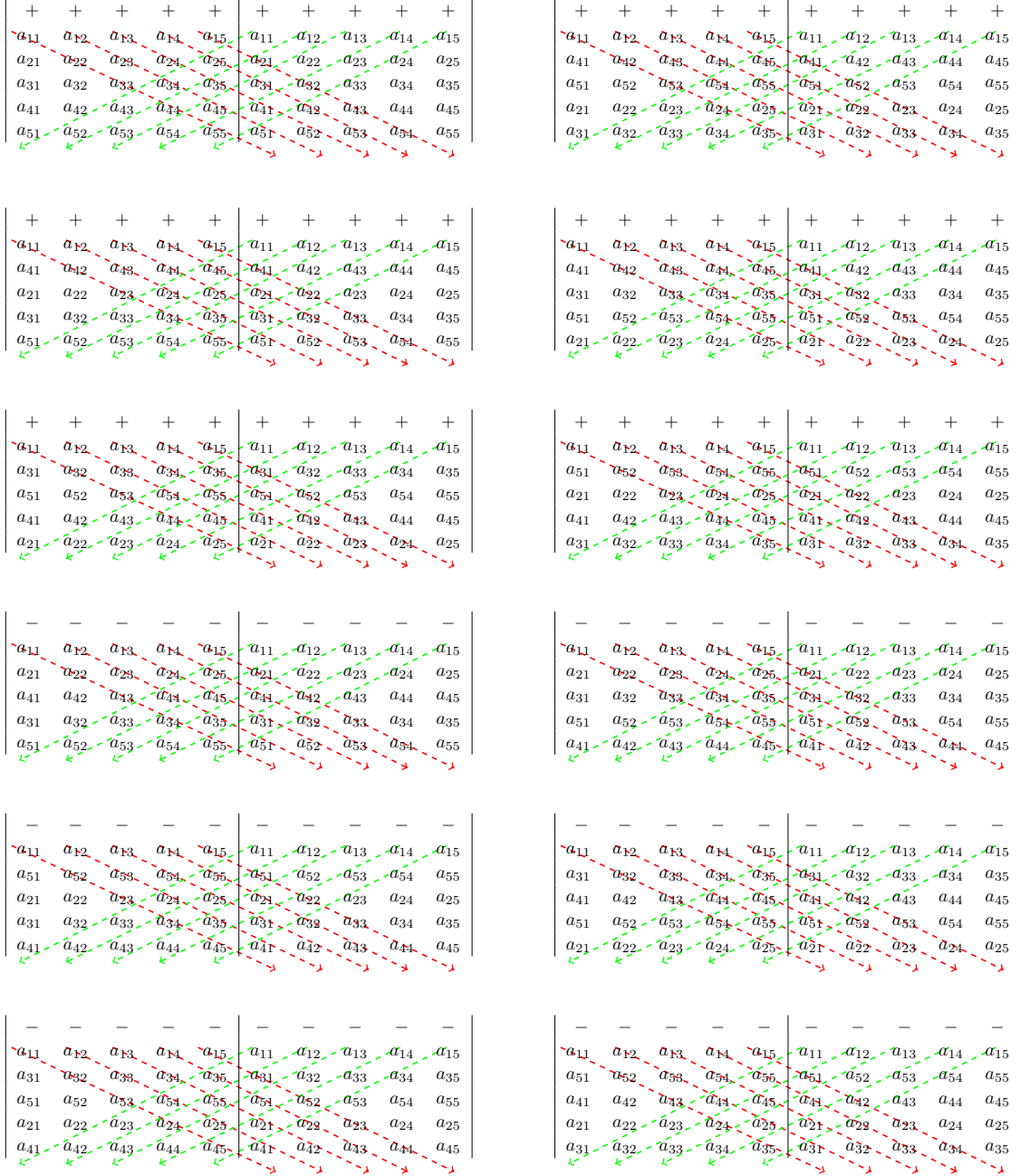
To calculate the determinants of a 5×5 matrix, such as A

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{pmatrix}$$

By using the Sarrus pattern or rule, first, for the convenience of work, the first line of the determinant A constant is kept constant and we place the next four rows, the second, third, fourth, and fifth rows together. So that we apply half of the permutations of the even and half of the permutations of the odd (2, 3, 4, 5) for the displacement of these four rows. For example, even permutations are intended to move these four-line A matrix: (2, 3, 5), (2, 5, 3), (2, 5, 4), (2, 3, 4), (2, 4)(3, 5), e and odd permutation are intended like this: (2, 4, 3, 5)(2, 4, 5, 3), (2, 5, 4, 3), (2, 3, 4, 5), (4, 5), (3, 4) Like the



4×4 matrix determinant, you do not need to add an extra template as follows.



$$\det(A) = \sum_{\varphi \in S_5} (\text{Sgn}(\varphi)) a_{1\varphi(1)} a_{2\varphi(2)} a_{3\varphi(3)} a_{4\varphi(4)} a_{5\varphi(5)}$$



4 Sarrus rule for calculating 6×6 and the above ranks determinant

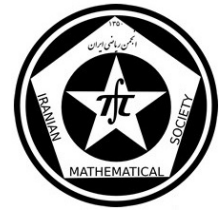
In the expansion of determinants by the method of Sarrus's patterns, when we calculate a 2×2 determinant we consider only one determinant. In expansion by means of the Sarrus model, we extend this to more than 2-rank determinism. When we calculate a 3×3 determinant, we can do it with one determinant and copy of same determinant in the right site. And If we want to get a 4×4 determinant with this method we must use 3 determinants and copy of each of them in their right hand. If we want to calculate 5×5 determinant by this way, we need to use 12 determinants with the copy of each of them next to them. And if we want to calculate a 6×6 determinant, we can do it with 60 determinants, with a copy from each of them beside themselves. Therefore, in general, to calculate a $n \times n$ deterministic method in Sarrus pattern, the number of $\frac{(n-1)!}{2}$ determinants and their copy must be put together and the calculation of the determinant must be done according to the description given.

Acknowledgement

Thanks to Mr. Masoud Pendar for his computational helps and drawing graphs.

References

- [1] P.J. Cameron, *Permutation Groups*, LMS Student Text 45. Cambridge University Press, Cambridge, 1999.
- [2] K. Haffman, R. Kunze, *Linear Algebra, second edition*, Prentice-Hall, Inc., 1971.
- [3] H. Kurzweil, B. Stellmacher , *The theory of finite groups: an introduction* , Springer-Verlag New York, Inc., 2004.
- [4] A. Seress, *Permutation group algorithms*, Cambridge Tracts in Mathematics, 152. Cambridge University Press, Cambridge, 2003.
- [5] S.Haidarian, *Generalization of Sarrus and Laplace rule for expansion the determinant and applications of determinant*, M.Sc. thesis, University of Kurdistan, 2016.



Higher order finite difference method based on linear barycentric rational interpolation to solve sub-diffusion equations

Iraj FahimiI-Khalilabad ¹⁹ , Safar Irandoust and Somayeh Abdi Mazraeh

Emails: 0.73, 0.33, 0.83i.fahimi@tabrizu.ac.ir[†],

s.irandoust@tabrizu.ac.ir^{†*},

s.abdi.m@tabrizu.ac.ir^{†**} shivanian@sci.ikiu.ac.ir

Abstract.

Finding the analytical solution of fractional partial differential equations (FPDEs) is difficult or impossible, so the approaches of numerical methods for fractional differential equations are the main topics in the study of fractional calculus. In recent years, various numerical methods have been used to solve partial differential equations of fractional order. Such as the finite difference method and the finite element method. In general, Riemann-Liouville derivatives and commonly Caputo derivatives are more used in the problem formulation. Riemann-Liouville derivatives are often approximated using the generalized Grunwald-Letnikov formula. So far, several numerical methods have been devised for Caputo derivatives. In this paper, an approximate multistep method for discretization of Caputo derivative is provided using the barycentric Lagrange interpolation weights for numerical solution of FPDEs such as fractional sub-diffusion equation.

Keywords: fractional differential equations, fractional partial differential equations, Riemann-Liouville derivatives , Caputo derivatives, the barycentric Lagrange interpolation.

1 Introduction

Consider the following type of time-fractional diffusion equation:

$$\begin{cases} \delta_t \Upsilon = {}^{RL}_0 D_t^{1-\alpha} (\kappa_\alpha \delta_x^2 \Upsilon) + f(x, t), & (x, t) \in (a, b) \times (0, T], \\ \Upsilon(x, 0) = \phi_0(x), & x \in (a, b), \\ \Upsilon(a, t) = \Upsilon_a(t), \Upsilon(b, t) = \Upsilon_b(t), & t \in (0, T], \end{cases} \quad (325)$$

¹⁹Speaker



where $\kappa_\alpha > 0$, $0 < \alpha < 1$ and ${}_0^{RL}D_t^{1-\alpha}$ denote the Riemann-Liouville derivative of order $1 - \alpha$ [6, 7]. The equation (1.1) is called Riemann-Liouville type sub-diffusion equation [7]. The fractional diffusion equation, when the order of the differentiation is between 0 and 1, is called the fractional sub-diffusion equation [10].

Clearly, if $\alpha \rightarrow 1$, the above order is reduced to the classical diffusion equation.

If $\Upsilon(x, t)$ is suitably smooth in time and the fractional integral ${}_0D_t^{-(1-\alpha)}$ is applied to both sides of (1, 1), the following Caputo type time-fractional diffusion can be obtained [3, 7, 8]

$$\begin{cases} {}_0^CD_t^\alpha \Upsilon = \kappa_\alpha \partial_x^2 \Upsilon + g(x, t), & (x, t) \in (a, b) \times (0, T], \\ \Upsilon(x, 0) = \phi_0(x), & x \in (a, b), \\ \Upsilon(a, t) = \Upsilon_a(t), \Upsilon(b, t) = \Upsilon_b(t), & t \in (0, T], \end{cases} \quad (326)$$

where $g(x, t) = {}_0^{RL}D_t^{\alpha-1}f(x, t)$.

In the spatial direction, the derivative is discretized by the classical methods such as the central difference method or the compact difference method. If $0 < \alpha < 1$ and the solution $\Upsilon(x, t)$ is suitably smooth in time, then

${}_0^CD_t^\alpha \Upsilon(x, t) = {}_0^{RL}D_t^\alpha (\Upsilon(x, t) - \Upsilon(x, 0))$. Hence, a common way to discretize the Caputo derivative in (1.2) is to use the Grunwald-Letnikov approximation, or the $L1$ method, or the fractional linear multistep method [2, 3, 4, 10]. In this paper, in the time direction, the Caputo derivative has been discretized based on fractional linear barycentric rational interpolation method (FLBRI) [5]. The finite difference method for solving equation (1.2) is as follows:

$$\begin{cases} \delta_t^\alpha(v_i^n - v_i^0) = \kappa_\alpha \partial_x^2 u_i^n + g_i^n, & i = 1, 2, \dots, N-1, \\ v_i^n = \phi_0(x_i), & i = 0, 1, 2, \dots, N, \\ v_0^n = \Upsilon v_a(t_n), v_N^n = \Upsilon v_b(t_n), \end{cases} \quad (327)$$

where $\delta_t^\alpha(v_i^n - v_i^0) = \frac{1}{\Delta t^\alpha} \sum_{k=0}^n \omega_{n-k}^{(-\alpha)}(v_i^n - v_i^0)$, $\omega_k^{(-\alpha)}(k = 0, 1, \dots, n)$ are weights introduced in the next section.

The stability analysis and the convergence properties of the proposed scheme are analyzed while the Lax-Richtmyer definition and the maximum local truncation error being, respectively [6, 9].

After that, it has been obtained that the convergence order is $O(\Delta t^p + \Delta x^2)$ where $1 \leq p \leq 7$, Δt the temporal step size and Δx the spatial step size.

2 Main section and results

In this section, some definitions, theorems, and conclusions are considered.

Definition 2.1. ([6]) For the function $\tau(z)$ the following equations are called Riemann-Liouville integrals, Riemann-Liouville derivative, and Caputo derivative, respectively.

$${}_{z_0}I_z^\alpha \tau(z) = {}_{z_0}^{RL}D_z^{-\alpha} \tau(z) = \frac{1}{\Gamma(\alpha)} \int_{z_0}^z (z-c)^{\alpha-1} \tau(c) dc, \quad 0 < \alpha, \quad (328)$$



$${}^{RL}D_z^\alpha \tau(z) = \frac{1}{\Gamma(n-\alpha)} \left(\frac{d}{dz}\right)^n \int_{z_0}^z (z-c)^{n-\alpha-1} \tau(c) dc, \quad n-1 \leq \alpha < n, \quad (329)$$

$${}_C D_z^\alpha \tau(z) = \frac{1}{\Gamma(n-\alpha)} \int_{z_0}^z (z-c)^{n-\alpha-1} \tau^{(n)}(c) dc, \quad n-1 < \alpha \leq n, \quad (330)$$

Remark ([6]) The Riemann-Liouville derivative and the Caputo derivative of the function τ have the following relation:

$${}^{RL}D_t^\alpha \left(\tau(z) - \sum_{i=0}^{m-1} \frac{\tau^{(i)}(z-z_0)^i}{\Gamma(i+1)} \right) = {}_C D_z^\alpha \tau(z), \quad m-1 < \alpha < m, \quad m \in N. \quad (331)$$

Definition 2.2.([5]) If $s, k \in N$, then

$$\vartheta_k = \frac{(-1)^{k-s}}{2^s} \sum_{r \in A_k} \binom{s}{k-r}, \quad A_k = \{r \in \{0, 1, \dots, m-s\}, k-s \leq r \leq k\}, \quad (332)$$

and

$$\psi_{r,k} = \begin{cases} \frac{\vartheta_k}{(r-k)\vartheta_r}, & k \neq r, \\ -\sum_{\substack{t=0 \\ t \neq r}}^m \psi_{r,t}, & k = r. \end{cases} \quad (333)$$

Therefore, LBRI weights and FLBRI weights are defined as follows:

$$\omega_k^{(-1)} = \psi_{m,m-k}, \quad (k = 0, 1, \dots, m), (\text{LBRI}),$$

$$\omega_i^{(-\alpha)} = (\psi_{m,m})^\alpha \omega_i, \quad i = 0, 1, \dots, \quad (\text{FLBRI}),$$

$$\omega_0 = 1,$$

$$\omega_i = (\xi_i - 1) \frac{\psi_{m,m-1}}{\psi_{m,m}} \omega_{i-1} + (2\xi_i - 1) \frac{\psi_{m,m-2}}{\psi_{m,m}} \omega_{i-2} + \dots + (m\xi_i - 1) \frac{\psi_{m,0}}{\psi_{m,m}} \omega_{i-m},$$

$$\text{where } \xi_i = \frac{\alpha-1}{i}, \quad i = 0, 1, \dots,$$

Definition 2.3.([9]) Let Δt be the time step size, n_T a positive integer number where $\Delta t = T/n_T$, $t_n = n\Delta t$ for $n = 0, 1, \dots, n_T$, N a positive integer number, Δx the spatial step size, meaning $\Delta x = x_{i+1} - x_i$ for $i = 0, 1, \dots, N$ and $u^{n+1} = Au^n + f^n$ for $n = 0, 1, \dots, n_T$, the matrix representation of the finite difference method. According to Lax and Richtmyer's definition is stable whenever there exists a positive number m , independent of $n, \Delta x$ and Δt so that the norm of matrix A is less than or equal to m .

Theorem 2.1 The proposed finite difference method is unconditionally stable.

Theorem 2.2 The proposed finite difference method (1.3) is convergent, and its order of temporal convergence and that of spatial convergence is $O(\Delta t^p + \Delta x^2)$, where $1 \leq p \leq 7$.



Numerical example

In this subsection, we provide a numerical example to verify the proposed finite difference method.

Example: We consider the following subdiffusion equation.

$$\begin{cases} {}^C D_t^\alpha \Upsilon = \delta_x^2 \Upsilon + f(x, t), & (x, t) \in (0, 1) \times (0, 1], 0 < \alpha \leq 1, \\ \Upsilon(x, 0) = 0, & x \in (a, b), \\ \Upsilon(0, t) = \Upsilon(1, t) = 0, & t \in [0, 1], \end{cases} \quad (334)$$

Substituting the suitable right-hand side function $f(x, t)$ in the above equation (2.1.1), we get the exact solution $\Upsilon(x, t) = t^{10+t} \sin(\pi x)$.

The maximum L^2 error is defined as follows:

$$\max_{0 \leq n \leq n_T} \|v - \Upsilon^n\|_N = \max_{0 \leq n \leq n_T} \sqrt{\Delta x \sum_{j=0}^{N-1} (\Upsilon(x_j, t_n) - v_j^n)^2}.$$

For $\alpha = 0.45$, $\Delta x = 0.001$, $\Delta t = \frac{1}{64}$, $r = 6$, we solved equation (2.1.1) by using (1.3) method. Figures show that obtained results are useful. In figure1 m is equal to 4 and in figure2 m is equal to 3. Figures 1.a and 2.a show analytical solution and figures 1.b and 2.b show approximate solution achieved by using (1.3) method. By comparing analytical and approximate solution, high accuracy of the method can be concluded. Numerical results verify theories explained in previous sections.

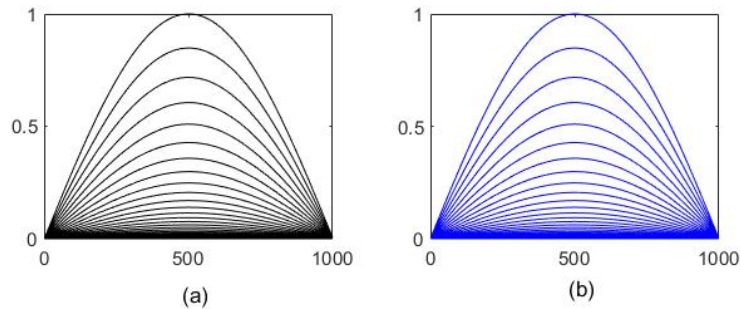


Figure 18: the ,maximum , L^2 error=0.002219228110674.

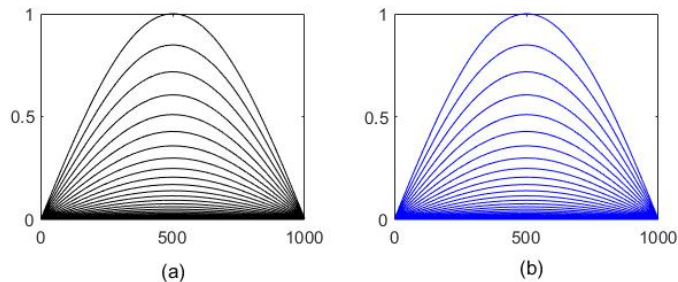


Figure 19: the ,maximum , L^2 ,error=0.002219243238113.

3 Conclusion

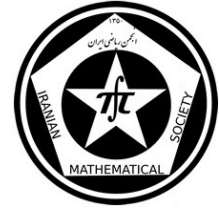
In this paper, a multistep method based on linear barycentric rational interpolation to solve Caputo-type sub-diffusion equations is presented. This method is more accurate than the methods previously used for this purpose, so that the convergence order in time direction is better and more effective in comparison with previous methods.

References

- [1] Berrut, J.-P., Hosseini, S. A., and Klein, G. The Linear Barycentric Rational Quadrature Method for Volterra Integral Equations. *SIAM Journal on Scientific Computing*, 36(1), (2014). A105A123.
- [2] Du, R., Cao, W. R., and Sun, Z. Z. A compact difference scheme for the fractional diffusion-wave equation. *Applied Mathematical Modelling*, 34(10), (2010). 29983007.
- [3] Gao, G.-, and Sun, Z.-. A compact finite difference scheme for the fractional sub-diffusion equations. *Journal of Computational Physics*, 230(3), (2011). 586595.
- [4] Hu, X., and Zhang, L. Implicit compact difference schemes for the fractional cable equation. *Applied Mathematical Modelling*, 36(9), (2012). 40274043.
- [5] Irandoust-pakchin, S., Abdi-mazraeh, S., and Kheiri, H. Construction of new generating function based on linear barycentric rational interpolation for numerical solution of fractional differential equations. *Journal of Computational and Applied Mathematics*, 375, (2020). 112799.



-
- [6] Li, C., and Ding, H. Higher order finite difference method for the reaction and anomalous-diffusion equation. *Applied Mathematical Modelling*, 38(1516), (2014). 38023821.
 - [7] Li, Changpin, and Fanhai Zeng. *Numerical methods for fractional calculus*. Vol. 24. CRC Press, (2015).
 - [8] Metzler, R., and Klafter, J. (2000). The random walks guide to anomalous diffusion: a fractional dynamics approach. *Physics Reports*, 339(1), (2000). 177.
 - [9] G. D. Smith, *Numerical solution of partial differential equations: finite difference methods*. Oxford university press, Bruned University, (1985).
 - [10] Zhao, X., and Sun, Z.-. A box-type scheme for fractional sub-diffusion equation with Neumann boundary conditions. *Journal of Computational Physics*, 230(15), (2011). 60616074.



Hyperbolic contour to numerical approximation of fractional Volterra integro-differential equation

Mahshid Foroughipour^{†20}, Mohammad Shafi Dahaghin[‡]

mahshid.fp64@gmail.com[†]

msh-dahaghin@sku.ac.ir[‡]

Abstract

In this paper, we consider the fractional Volterra integro-differential equations and represent the solution of this type of equations in the form of contour integral in the complex plane. Then we select the hyperbolic contour as an optimal contour to approximate this integral. Further, an example to show absolute errors for various fractional orders by using our numerical scheme on hyperbolic contour is given.

AMS subject Classification 2010: 26A33, 34C20.

Keywords: Fractional Volterra integro-differential equations, Hyperbolic contour, Numerical approximation.

1 Introduction and Preliminaries

The fractional Volterra integro-differential equations have many applications in physics, engineering, economics, diffusion problems. Finding the exact solutions of fractional Volterra integro-differential equations is sometimes difficult. So, numerical methods are proposed to obtain the solution of such types of equations [1, 5].

In this work, a numerical scheme is constructed to approximate the fractional Volterra integro-differential equations based on Laplace transform. Then we select the hyperbolic contour and use trapezoidal rule with equal step size to approximate this integral. Finally, an example for the application of method is given.

Definition 0.55. The Riemann-Liouville fractional integral of order $\alpha > 0$ ($m-1 \leq \alpha < m$, $m \in \mathbb{N}$) is defined as follows [6]

$$I_{t_0}^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_{t_0}^t (t-\tau)^{\alpha-1} f(\tau) d\tau = (f * K_\alpha)(t), \quad (335)$$

where $t > t_0$, $K_\alpha(t) = \frac{t^{\alpha-1}}{\Gamma(\alpha)}$, and $*$ is the Laplace convolution integral given by

$$(f * g)(t) = \int_0^t f(t-\tau)g(\tau) d\tau. \quad (336)$$

²⁰Speaker



Definition 0.56. The Caputo fractional derivative of order $\alpha > 0$ ($m - 1 \leq \alpha < m$, $m \in \mathbb{N}$) is defined as [6]

$${}^C D_{t_0}^\alpha f(t) = I_{t_0}^{m-\alpha} \frac{d^m}{dt^m} f(t) = \frac{1}{\Gamma(m-\alpha)} \int_{t_0}^t (t-\tau)^{m-1-\alpha} \frac{d^m}{dt^m} f(\tau) d\tau, \quad t > t_0. \quad (337)$$

Lemma 0.57. The Laplace transform of Caputo fractional derivative of order $\alpha > 0$ ($m - 1 \leq \alpha < m$, $m \in \mathbb{N}$) is given by [6]

$$\mathcal{L}\{{}^C D_{t_0}^\alpha f(t); t \rightarrow s\} = s^\alpha \mathcal{L}\{f(t); s\} - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)}(t_0). \quad (338)$$

2 Numerical approximation of fractional Volterra integro-differential equation

Consider the following fractional Volterra integro-differential equation:

$${}^C D_{t_0}^\alpha f(t) = g(t) + \int_0^t k(t, \tau) f(\tau) d\tau, \quad m - 1 \leq \alpha < m, \quad m \in \mathbb{N}. \quad (339)$$

For the convolution type of equation (339), the kernel will be of the form $k(t, \tau) = k(t - \tau)$. So the equation (339) becomes

$${}^C D_{t_0}^\alpha f(t) = g(t) + \int_0^t k(t - \tau) f(\tau) d\tau. \quad (340)$$

Taking the Laplace transform on the both side of (340) with respect to t , we have

$$\mathcal{L}\{{}^C D_{t_0}^\alpha f(t); t \rightarrow s\} = \mathcal{L}\{g(t); t \rightarrow s\} + \mathcal{L}\left\{\int_0^t k(t - \tau) f(\tau) d\tau; t \rightarrow s\right\}. \quad (341)$$

By using the convolution theorem and the Laplace transform of Caputo fractional derivative, we have

$$s^\alpha F(s) - \sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)}(t_0) = G(s) + K(s)F(s),$$

where $F(s) = \mathcal{L}\{f(t); s\}$, $G(s) = \mathcal{L}\{g(t); s\}$ and $K(s) = \mathcal{L}\{k(t); s\}$. Finally, we get

$$F(s) = \frac{G(s)}{s^\alpha - K(s)} + \frac{\sum_{k=0}^{m-1} s^{\alpha-k-1} f^{(k)}(t_0)}{s^\alpha - K(s)}.$$

Taking inverse Laplace, the problem reduces to compute the following integral in the complex plane

$$f(t) = \frac{1}{2\pi i} \int_{c-i\infty}^{c+i\infty} F(s) e^{st} ds. \quad (342)$$

The numerical method for inversion of the Laplace transform is based on the approximation of the Bromwich complex contour integral. We select the contour of integration to approximate the path



from $c - i\infty$ to $c + i\infty$. To this end we consider hyperbolic contour [5, 6]. The parametric equation of hyperbolic contour is given by [7, 5, 6]

$$s = \beta \left(1 - \sin(\lambda + c) \cosh(\zeta) \right) + i\beta \cos(\lambda + c) \sinh(\zeta), \quad -\infty < \zeta < +\infty, \quad (343)$$

where β , λ and c are parameters and need to be optimized for better accuracy. More details about the hyperbolic contour are given in [7, 6]. The numerical solution can be represented in the following form

$$f(t) = \mathcal{L}^{-1}\{F(s); s \rightarrow t\} = \frac{1}{2\pi i} \int_{\Gamma} F(s(\zeta)) e^{s(\zeta)t} s'(\zeta) d\zeta. \quad (344)$$

If we use equal weight quadrature rule, i.e trapezoidal rule with step size h , then the equation (344) can be approximated as

$$f_N(t) = \frac{h}{2\pi i} \sum_{j=-N}^N F(s(\zeta_j)) e^{s(\zeta_j)t} s'(\zeta_j), \quad 1 < \alpha < 2, \quad \zeta_j = jh. \quad (345)$$

Example

Consider the following fractional Volterra integro-differential equation

$$\begin{cases} {}^C D_{t_0}^{\alpha} f(t) = t + \int_0^t (t - \tau) f(\tau) d\tau, \\ t_0 = 0, \quad f(0) = 0, \quad f'(0) = 1, \end{cases} \quad (346)$$

where $\alpha \in (1, 2)$. The exact solution to the fractional Volterra integro-differential (346) is

$$f(t) = t E_{\alpha+2,2}(t^{\alpha+2}) + t^{\alpha+1} E_{\alpha+2,\alpha+2}(t^{\alpha+2}), \quad 1 < \alpha < 2,$$

where $E_{\alpha+2,2}(t^{\alpha+2})$ and $E_{\alpha+2,\alpha+2}(t^{\alpha+2})$ are the Mittag-Leffler functions which are defined generally as follows [6]:

$$E_{\alpha,\beta}(t) = \sum_{n=0}^{\infty} \frac{t^n}{\Gamma(\alpha n + \beta)}, \quad \alpha, \beta \in \mathbb{C}, \Re(\alpha) > 0. \quad (347)$$

To give the approximate solution of (346), we use the presented numerical scheme. Applying the Laplace transform to equation (346) with respect to t , we have

$$s^{\alpha} F(s) - s^{\alpha-1} f(0) - s^{\alpha-2} f'(0) = \frac{1}{s^2} + \frac{1}{s^2} F(s).$$

Since $f(0) = 0$ and $f'(0) = 1$, so we get

$$F(s) = \frac{s^{\alpha} + 1}{s^{\alpha+2} - 1}, \quad 1 < \alpha < 2. \quad (348)$$

From the above equation and equation (345), we get the approximate solution as

$$f_N(t) = \frac{h}{2\pi i} \sum_{j=-N}^N \frac{(s(\zeta_j))^{\alpha} + 1}{(s(\zeta_j))^{\alpha+2} - 1} e^{s(\zeta_j)t} s'(\zeta_j), \quad 1 < \alpha < 2, \quad \zeta_j = jh.$$



N	α	Absolute error
10	1.90	$1.4517e - 04$
15	1.90	$5.9696e - 06$
20	1.90	$2.1923e - 07$
25	1.90	$7.5541e - 09$
30	1.90	$2.4999e - 10$
35	1.90	$8.0454e - 12$
40	1.90	$2.5368e - 13$
45	1.90	$7.8700e - 15$
50	1.90	$2.3939e - 16$
60	1.90	$1.5179e - 18$
70	1.90	$2.1685e - 19$
100	1.90	$3.9031e - 18$

Table 24: Absolute errors of equation (346) using our numerical scheme and hyperbolic contour for $\alpha = 1.90$, $t = 0.001$ and different N .

For our numerical experiments, we chose the parameters such that the absolute error has the least value. See [5]. So, we take the parameters as follows

$$\begin{cases} c = 0.3, & \lambda = 1.1721, \\ h = \frac{1}{N}\vartheta_\lambda, \\ \beta = \frac{1}{\vartheta_\lambda t}((4\pi\lambda - \pi^2)N), \end{cases}$$

where

$$\vartheta_\lambda = \cosh^{-1} \left(\frac{2\lambda}{(4\lambda - \pi) \sin(\lambda)} \right).$$

In order to evaluate the Mittag-Leffler functions $E_{\alpha+2,2}(t^{\alpha+2})$ and $E_{\alpha+2,\alpha+2}(t^{\alpha+2})$ in the exact solution and approximate solution, we use MATLAB software. Table (52) and Figures (20)-(21) show the absolute errors for approximate solution. Table (52) and Figure (20) show absolute errors of fractional Volterra integro-differential equation (346) for $\alpha = 1.90$, $t = 0.001$ and various N . Figure (21) shows the absolute errors for various fractional orders and different times.

4 Conclusions

In this paper we stated a numerical scheme for approximating the fractional Volterra integro-differential equations. The proposed numerical scheme is based on Laplace transform and quadrature

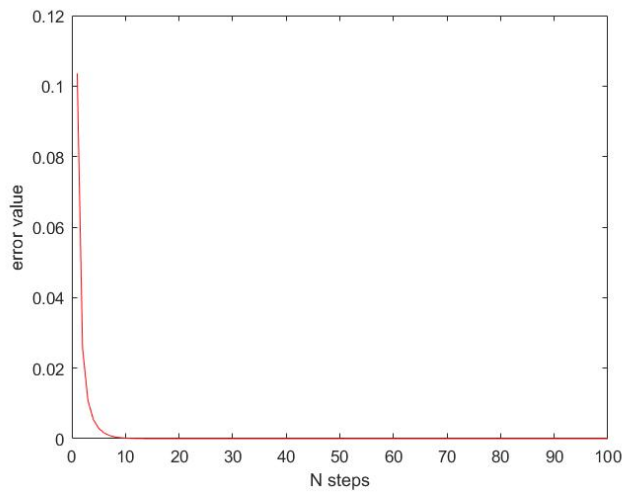


Figure 20: Absolute error for $\alpha = 1.90$ and $t = 0.001$.

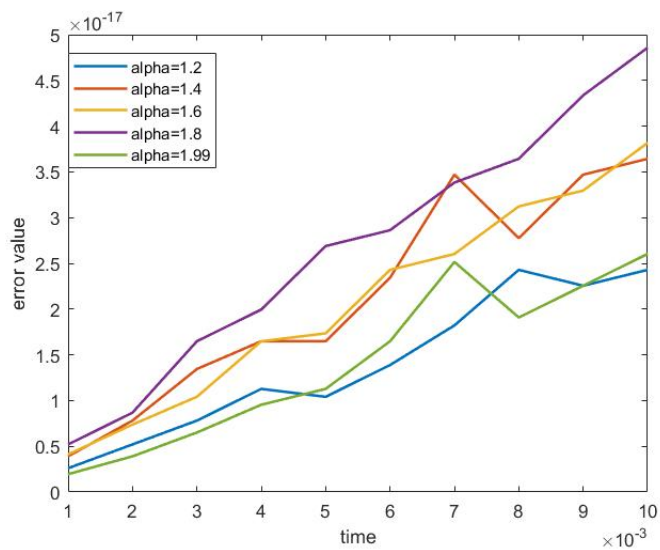


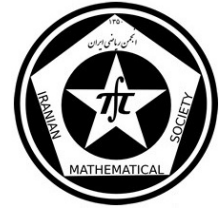
Figure 21: Absolute errors for various fractional orders and different times.



rule. Then we gave an example and showed the absolute value errors for approximating the solution of fractional Volterra integro-differential equation. The proposed method is a very well alternative for solving fractional problems in science and engineering.

References

- [1] M. Akbar, R. Nawaz, S. Ahsan, K.S. Nisar, A.H. Abdel Aty and H. Eleuch, *New approach to approximate the solution for the system of fractional order Volterra integro-differential equations*, Results Phys., **19** (2020), 103453.
- [2] W. McLean and V. Thomée, *Numerical solution via Laplace transforms of a fractional order evolution equation*, J. Integral Equ. Appl., **22** (2010), 57–94
- [3] I. Podlubny, *Fractional Differential Equations*, Academic Press San Diego, 1999.
- [4] N. Rajagopala, S. Balajia, R. Seethalakshmi and V.S. Balajib, *A new numerical method for fractional order Volterra integro-differential equations*, Ain Shams Eng. J., **11** (2020), 171–177.
- [5] M. Uddin and M. Uddin, *On the numerical approximation of Volterra integro-differential equation using Laplace transform*, Comput. Methods Differ. Equ., **8** (2020), 305–313.
- [6] J. Weideman and L. Trefethen, *Parabolic and hyperbolic contours for computing the Bromwich integral*, Math. Comput., **76** (2007), 1341–1356.



Identification of a space-dependent source term in a time-fractional wave equation

Tahereh Molaei^{†21}, Alimardan Shahrezaei[‡]

Department of Mathematics, Faculty of Mathematical Sciences, University of Alzahra, Tehran, Iran

t.molaei@alzahra.ac.ir[†]

ashahrezaei@alzahra.ac.ir[‡]

Abstract

In this work, an inverse space-dependent source problem for a time-fractional wave equation on a finite domain is considered. The time-fractional derivative is defined in Caputo's sense. The method of fundamental solutions is applied for the time-fractional wave problem to approximate the solution of the problem and the source function. Because the resulting matrix equation is ill-conditioned, the Tikhonov regularization is applied. In the numerical results section, the discontinuous source function is approximated by the proposed algorithm. There are few papers on the inverse source problem for a time fractional wave equation and they provide numerical results for a continuous source function. Numerical examples are presented to show the effectiveness of the proposed method.

AMS subject Classification 2010: 35R11, 35R30, 65M32.

Keywords: Time-fractional wave equation; Inverse space-dependent source problem.

1 Introduction

Different sciences use fractional calculus in phenomena modeling to better describe unusual behaviors of the phenomena. Deep applications of fractional calculus are found in many fields of science and engineering [1]. In recent decades, the time-fractional wave equations have been attracted the attention of many researchers. However, there are only a few papers on inverse source problems for fractional wave equations on a bounded domain [2, 3, 4, 5, 6, 7, 8].

Inverse source problems are a class of inverse problems where a couple consisting of a source function and a solution of the equation is unknown.

In this work, we consider the following inverse space-dependent source problem for time-fractional wave equation:

$$\begin{cases} {}^c D_t^\alpha u(x, t) = u_{xx}(x, t) + f(x), & 1 < \alpha < 2, 0 < x < L, 0 < t < T, L \in \mathbb{R} > 0 \\ u(x, 0) = \phi(x), & 0 \leq x \leq L, \\ \frac{\partial u(x, 0)}{\partial t} = \zeta(x), & 0 \leq x \leq L, \\ u(0, t) = u(L, t) = 0, & 0 \leq t \leq T, \end{cases} \quad (349)$$

²¹Speaker



with overspecified condition

$$u(x, T) = \xi(x), \quad 0 \leq x \leq L, \quad (350)$$

where unknown functions u and f represent the state variable and space-dependent source term, respectively, T denotes final time, ${}^c D_t^\alpha$ denotes the Caputo fractional derivative in [9] as follows

$${}^c D_t^\alpha u(x, t) = \frac{1}{\Gamma(2-\alpha)} \int_0^t (t-s)^{1-\alpha} \frac{\partial^2 u(x, s)}{\partial s^2} ds, \quad 1 < \alpha < 2, \quad (351)$$

The inverse source problem (349) is an ill-posed problem [8], meaning that small errors in the measured data may lead to large errors in the solution, a fact which makes inverse problems difficult to solve numerically.

In the following, in Section 2, a numerical procedure, in Section 3, some numerical examples and in Section 4, a conclusion is presented.

2 Numerical procedure

In order to use the method of fundamental solutions (MFS) to solve the problem (349), first, we find a suitable transformation which change the inverse problem into a direct problem containing only one unknown function and then apply the method of fundamental solutions for the resulted problem. For this purpose, we define the following transformation:

$$v(x, t) = u(x, t) + r(x), \quad (352)$$

where

$$r(x) = \int_0^x (x-\alpha) f(\alpha) d\alpha. \quad (353)$$

By the transformation (352) and overspecified condition (350), the problem (349) is transformed into the following problem

$$\frac{\partial^\alpha v}{\partial t^\alpha} = \frac{\partial^2 v}{\partial x^2}, \quad 1 < \alpha < 2, 0 < x < L, 0 < t < T, \quad (354)$$

$$v(x, 0) - v(x, T) = \phi(x) - \xi(x), \quad 0 \leq x \leq L, \quad (355)$$

$$\frac{\partial v(x, 0)}{\partial t} = \zeta(x), \quad 0 \leq x \leq L, \quad (356)$$

$$v(0, t) - v(0, T) = -\xi(0), \quad 0 \leq t \leq T, \quad (357)$$

$$v(L, t) - v(L, T) = -\xi(L), \quad 0 \leq t \leq T. \quad (358)$$

Now we use the MFS to solve the problem (354)-(358). The method of fundamental solutions considers the numerical solution of a problem as a linear combination of fundamental solutions of the governing equation of the problem which should be imposed at a finite number of collocation points. The fundamental solution of Equation (354) is given [10] as

$$G(x, t) = \frac{1}{2t^{\alpha/2}} M\left(\frac{\alpha}{2}; \frac{|x|}{t^{\alpha/2}}\right), \quad (359)$$



where

$$M(\alpha; z) = \sum_{k=0}^{\infty} \frac{(-1)^k z^k}{z! \Gamma[-\alpha k + (1 - \alpha)]}. \quad (360)$$

The time shift function

$$\psi(x, t) = G(x, t + \tau), \quad (361)$$

is a non-singular solution of Equation (354) in the domain $[0, L] \times [0, T]$ where $\tau > T$. We assume $\chi_B = \{(x_k, t_k)\}_{k=1}^N$ is a set of $N = n + m + s$ scattered nodes on boundaries as

$$\chi_B = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3, \quad (362)$$

where

$$\Gamma_1 = \{(x_k, t_k); 0 \leq x_k \leq L, t_k = T, k = 1, \dots, n\}, \quad (363)$$

$$\Gamma_2 = \{(x_k, t_k); x_k = 0, 0 \leq t_k \leq T, k = n + 1, \dots, n + m\}, \quad (364)$$

$$\Gamma_3 = \{(x_k, t_k); x_k = L, 0 \leq t_k \leq T, k = n + m + 1, \dots, n + m + s\}. \quad (365)$$

The method of fundamental solutions expresses the approximation of $v(x, t)$ function as follows

$$\tilde{v}(x, t) = \sum_{k=1}^N \lambda_k \psi(x - x_k, t - t_k), \quad (366)$$

where $\psi(x, t)$ is given by (361) and unknown coefficients of λ_k , $k = 1, \dots, N$ are obtained by imposing boundary data at collocation points. We consider the total number of collocation points equal to the total number of source points and express the resulting system in matrix form

$$A\lambda = b. \quad (367)$$

The $(N \times N)$ matrix A , column vector b with N entries are defined as follows

$$A = \begin{pmatrix} \psi(x_i - x_k, 0 - t_k) + \frac{\partial \psi(x_i - x_k, 0 - t_k)}{\partial t} - \psi(x_i - x_k, T - t_k) \\ \psi(0 - x_k, t_j - t_k) - \psi(0 - x_k, T - t_k) \\ \psi(L - x_k, t_r - t_k) - \psi(L - x_k, T - t_k) \end{pmatrix}, \quad (368)$$

$$b = [\phi(x_i) + \zeta(x_i) - \xi(x_i), -\xi(x_j), -\xi(x_r)]^T, \quad (369)$$

where $i = 1, \dots, n, j = n + 1, \dots, n + m, r = n + m + 1, \dots, n + m + s$.

Because the inverse source problem (349) is ill-posed, we use the Tikhonov regularization method. By considering $\lambda^* = [\lambda_1^*, \lambda_2^*, \dots, \lambda_N^*]^T$ as the regularized solution, the numerical solution of the problem (354)-(357) is obtained as

$$\tilde{v}^*(x, t) = \sum_{k=1}^N \lambda_k^* \psi(x - x_k, t - t_k). \quad (370)$$

The numerical solution of the problem (349) is then given as

$$\tilde{u}^*(x, t) = \tilde{v}^*(x, t) - r(x), \quad (371)$$

where

$$r(x) = \tilde{v}^*(x, T) - u(x, T), \quad (372)$$

and

$$f^*(x) = r''(x). \quad (373)$$



Table 25: Numerical results of Example 0.106 and Example 0.59.

error	Example 0.106			Example 0.59
	$\alpha = 1.15$	$\alpha = 1.5$	$\alpha = 1.9$	$\alpha = 1.8$
RMS(u)	7.22e-2	1.88e-2	8.35e-2	0.1349
RMS(f)	1.1e-3	8.09e-2	7.5e-3	0.1353

3 Numerical examples

For numerical purposes, we consider $T = 1$ and we apply the proposed procedure on two test problems. The numerical error produced due to the performance of the scheme is measured with the root mean square (RMS) error, defined as

$$RMS(u) = \sqrt{\frac{1}{N_t} \sum_{k=1}^{N_t} (u(x_k, t_k) - \tilde{u}(x_k, t_k))^2}, \quad (374)$$

where N_t is the total number of testing points in domain $[0, L] \times [0, T]$, $u(x, t)$ and $\tilde{u}^*(x_k, t_k)$ are exact and numerical values at these points, respectively. Similarly, the RMS error is defined for $f(x)$.

Example 0.58. We consider the inverse problem (349) with $L = \pi$ and a exact source term as

$$f(x) = \sin(\pi x), \quad x \in [0, \pi],$$

The numerical result for $\tau = 1.2$, $m = n = s = 3$ and dfferent α is peresented in Figure 42 and Table 52.

Example 0.59. In this example, we consider the inverse problem (349) with $L = \pi$ and a discontinuous exact source term as

$$f(x) = \begin{cases} 0, & 0 \leq x \leq \pi/4, \\ 1, & \pi/4 < x \leq \pi/2, \\ -1, & \pi/2 < x \leq 3\pi/4, \\ 0, & 3\pi/4 < x \leq \pi, \end{cases}$$

Figure 42 and Table 52 display the numerical result for $\alpha = 1.8$, $\tau = 1.2$ and $m = n = s = 3$.

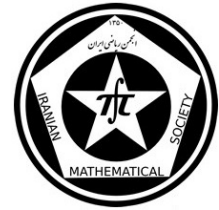
4 Conclusion

In this paper, we developed the method of fundamental solutions for the time-fractional wave problem to recover a space-dependent source function in a time-fractional wave inverse problem on a finite domain. The numerical results showed the applicability and practical efficiency of the algorithm for the problem (1) in terms of RMS error and the relative error. The presented algorithm is truly meshless, computationally inexpensive, in terms of implementation is relatively simple.



References

- [1] HG. Sun, Y. Zhang, D. Baleanu, W. Chen, YQ. Chen, *A new collection of real world applications of fractional calculus in science and engineering*, Commun. Nonlinear. Sci. Numer. Simulat., **64** (2018) 213–231.
- [2] J. Janno, Determination of time-dependent sources and parameters of nonlocal diffusion and wave equations from final data, *Fract. Calc. Appl. Anal.*, **23** (2020) 1678–1701.
- [3] X. Yan, T. Wei, *Determine a Space-Dependent Source Term in a Time Fractional Diffusion-Wave Equation*, Acta. Appl. Math., **165** (2020) 163–181.
- [4] K. Sisková, M. Slodicka, *A source identification problem in a time-fractional wave equation with a dynamical boundary condition*, Comput. Math. Appl., **75** (2018) 4337–4354.
- [5] K. Sisková, M. Slodicka, *Identification of a source in a fractional wave equation from a boundary measurement*, J. Comput. Appl. Math., **349** (2019) 172–186.
- [6] K. Sisková and M. Slodicka, *Recognition of a time-dependent source in a time-fractional wave equation*, Appl. Numer. Math., **121** (2017) 1–17.
- [7] S. Jiang, Y. Wu, *Recovering space-dependent source for a time-space fractional diffusion wave equation by fractional Landweber method*, Inverse. Probl. Sci. Eng., (2020) 1–22.
- [8] T. Wei, X. Yan, *Recovering a space-dependent source term in a time-fractional diffusion wave equation*, J. Appl. Anal. Comput., **9** (2019) 1801–1821.
- [9] I. Podlubny, *Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*, Elsevier, 1998.
- [10] F. Mainardi, A. Mura, G. Pagnini, *The M-Wright Function in Time-Fractional Diffusion Processes: A Tutorial Survey*, Int. J. Differ. Equ., **2010** (2010) 104505.



Implementation of Lucas wavelet functions for solving fractional differential equations

Haniye Dehestani²², Yadollah Ordokhani

Department of Mathematics, Faculty of Mathematical Sciences, Alzahra University, Tehran, Iran

Abstract

In this study, the Lucas wavelet technique is presented for the solution of fractional differential equations. According to Lucas wavelet functions, modified operational matrix of integration and pseudo-operational matrix of the derivative, we reduced the proposed equations to a system of algebraic equations. Also, the numerical outcomes compare with the results of the other methods in the last section. Illustrative examples demonstrate the validity and applicability of the computational technique.

AMS subject Classification 2010: 34A08, 65M70.

Keywords: Lucas wavelet functions, Modified operational matrix, Fractional differential equations.

1 Introduction

Fractional differential equations have appeared in the modeling of many phenomena in real life, such as electromagnetism [1], fluid-dynamic models [2], and etc. Due to the useful properties of wavelet methods, many researchers have studied in this field (see [3, 4, 5] and references therein).

2 Requirement concepts

Lucas wavelet functions

The Lucas wavelet functions are defined over the interval $[0, 1)$ as follows [5, 6]:

$$\Psi_{nm}(t) = \begin{cases} 2^{\frac{K-1}{2}} \mathfrak{L}_m(2^{K-1}t - n + 1), & \frac{n-1}{2^{K-1}} \leq t < \frac{n}{2^{K-1}}, \\ 0, & \text{otherwise,} \end{cases} \quad m = 0, 1, \dots, M, \quad n = 1, 2, \dots, 2^{K-1}. \quad (375)$$

²²Speaker



where

$$\mathfrak{L}_m(2^{K-1}t - n + 1) = \frac{1}{\sqrt{\int_0^1 L_m^2(x) dx}} L_m(2^{K-1}t - n + 1),$$

and $L_m(x)$ denotes the Lucas polynomials [5, 6]

$$L_0(x) = 2, \quad L_m(x) = \sum_{k=0}^{\lfloor \frac{m}{2} \rfloor} \frac{m}{m-k} \binom{m-k}{k} x^{m-2k}, \quad m > 0. \quad (376)$$

Modified operational matrix of integration

For achieving the aim of this paper, we compute the modified operational matrix of integration (MOMI). Hence, according to the Lucas wavelet functions and their properties, we get [5]

$$\int_0^t \Psi(\xi) d\xi = \Theta \Psi(t) + \Lambda(t). \quad (377)$$

where Θ and $\Lambda(t)$ are MOMI and complement vector, respectively. It should be noted that each component of the matrix is obtained as follows[5]:

$$\int_0^t \Psi_{nm}(\xi) d\xi \simeq \begin{cases} 0, & t < \frac{n-1}{2^{K-1}}, \\ \sum_{s=1}^{2^{K-1}} \sum_{i=0}^M \theta_{si}^{nm} \Psi_{si}(t), & \frac{n-1}{2^{K-1}} \leq t < \frac{n}{2^{K-1}}, \quad m = 0, 1, \dots, M-1, \\ \sum_{s=1}^{2^{K-1}} \sum_{i=0}^M \theta_{si}^{nM} \Psi_{si}(t) + \lambda_M^n, & \frac{n-1}{2^{K-1}} \leq t < \frac{n}{2^{K-1}}, \quad m = M, \\ \sum_{s=1}^{2^{K-1}} \theta_{s0}^{nM} \Psi_{s0}(t), & t \geq \frac{n}{2^{K-1}}. \end{cases} \quad (378)$$

3 Pseudo-operational matrix of derivative

To compute the fractional derivative of unknown function in the problem, we obtain the pseudo-operational matrix of derivative as follows [5]:

$$D^\nu \Psi(t) = t^{q-\nu} \Upsilon \Psi(t), \quad (379)$$

where Υ is called the pseudo-operational matrix of derivative.

4 Numerical technique

This section discusses the numerical approach for solving the following equation:

$$\begin{cases} D^\nu u(t) = F(t, u(t), u'(t), u''(t), D^\gamma u(t)), & 0 < \nu \leq 1, \quad 1 < \gamma \leq 2, \\ u(0) = a_0, \quad u'(0) = a_1. \end{cases} \quad (380)$$



To get the approximate solution, we assume

$$u''(t) = A^T \Psi(t). \quad (381)$$

Then, by repeated integrations, we obtain:

$$u'(t) = A^T (\Theta \Psi(t) + \Lambda(t)) + a_1, \quad (382)$$

and

$$u(t) = A^T \left(\Theta^2 \Psi(t) + \Theta \Lambda(t) + \int_0^t \Lambda(\xi) d\xi \right) + a_1 t + a_0. \quad (383)$$

Furthermore, from the pseudo-operational matrix of derivative, we get

$$D^\nu u(t) = A^T \left(t^{q-\nu} \Theta^2 \mathbf{r} + \Theta D^\nu \Lambda(t) + D^\nu \left[\int_0^t \Lambda(\xi) d\xi \right] \right) + \frac{\Gamma(2)a_1}{\Gamma(2-\nu)} t^{1-\nu}. \quad (384)$$

By substituting Eqs. (381)-(384) to (560) and using Newton-Cotes nodes, we get the system of algebraic equations.

5 Illustrative examples

In this section, we compare the results with Chebyshev wavelets [5, 8] and fractional-order Bessel functions [9], which demonstrate that using the pseudo-operational matrix of the derivative and modified operational matrix of integration affect directly the accuracy of the method.

Example 0.60. Consider the following nonlinear multi-fractional differential equation [5, 8]

$$D^\nu u(t) + D^\gamma u(t) + u^2(t) = (t^2 - t)^2 + \frac{2t^{2-\nu}}{\Gamma(3-\nu)} + \frac{2t^{2-\gamma}}{\Gamma(3-\gamma)} - \frac{t^{1-\gamma}}{\Gamma(2-\gamma)}, \quad 1 < \nu \leq 2, \quad 0 < \gamma \leq 1,$$

subject to $u(0) = 0$, $u'(0) = -1$. The exact solution of this example is $u(t) = t^2 - t$. The results of this problem are illustrated in Table 26 and Figure 22. Table 26 shows the superiority of the method in comparison to Chebyshev wavelets [5, 8].

Example 0.61. Consider the following nonlinear fractional differential equation [9]

$$D^\nu u(t) = -u^2(t) + 1, \quad 0 < \nu \leq 1,$$

subject to $u(0) = 0$. The exact solution of this example for $\nu = 1$, is $u(t) = \frac{\exp(2t)-1}{\exp(2t)+1}$. In Table 27, we compare the results with fractional-order Bessel functions [9], which shows that this method is more accurate than the method in [9]. Figure 23 illustrates that, as approach ν to 1, the approximate solutions converge to the exact solution.

6 Conclusion

In this paper, we present an efficient numerical algorithm based on Lucas wavelet functions for solving fractional differential equations. Also, we implemented the method for two problems which demonstrates the efficiency and accuracy of the proposed method.



Table 26: Comparison of absolute error obtained by the present method with Refs. [5, 8] for $\nu = 2, \gamma = 0.5$ of Example 6.1.

t	Present method		Ref. [5]	Ref. [8]
	$M = 2, K = 1$	$M = 2, K = 2$	$M = 3, k = 3$	$M = 3, k = 3$
0.1	3.1311×10^{-16}	4.3964×10^{-17}	1.4411×10^{-4}	9.2677×10^{-7}
0.2	1.2468×10^{-16}	1.7133×10^{-16}	1.4007×10^{-4}	3.6193×10^{-6}
0.3	2.7930×10^{-16}	3.7803×10^{-16}	1.3459×10^{-4}	6.3739×10^{-6}
0.4	4.9431×10^{-16}	6.6363×10^{-16}	1.2835×10^{-4}	9.1427×10^{-6}
0.5	7.6888×10^{-16}	1.0313×10^{-15}	1.2241×10^{-4}	1.2129×10^{-5}
0.6	1.1022×10^{-15}	1.4215×10^{-15}	1.1491×10^{-4}	1.4512×10^{-5}
0.7	1.4934×10^{-15}	1.7723×10^{-15}	1.0803×10^{-4}	1.7072×10^{-5}
0.8	1.9416×10^{-15}	2.0844×10^{-15}	1.0114×10^{-4}	1.9533×10^{-5}
0.8	2.4462×10^{-15}	2.5886×10^{-15}	9.4240×10^{-5}	2.1888×10^{-5}

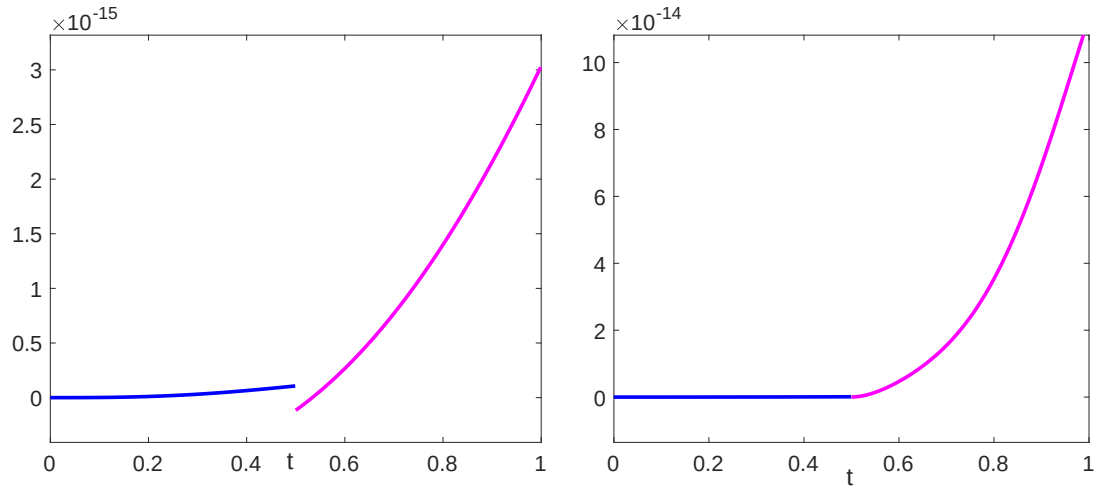


Figure 22: The absolute error for $M = 2$ (left) $M = 3$ (right) with $K = 2, \nu = 2, \gamma = 0.8$ of Example 6.1.

Table 27: Errors for various choices of M with $K = 1$ and $\nu = 1$ of Example 6.2.

	Present method		
	$M = 4$	$M = 6$	$M = 8$
L_2 -error	4.2802×10^{-4}	7.4311×10^{-6}	1.2909×10^{-7}
L_∞ -error	3.2158×10^{-4}	3.6721×10^{-6}	9.2888×10^{-8}
	Ref. [9]		
	$M = 4$	$M = 6$	$M = 9$
L_2 -error	6.33×10^{-4}	1.37×10^{-5}	2.69×10^{-7}
L_∞ -error	3.25×10^{-4}	6.80×10^{-6}	1.19×10^{-7}

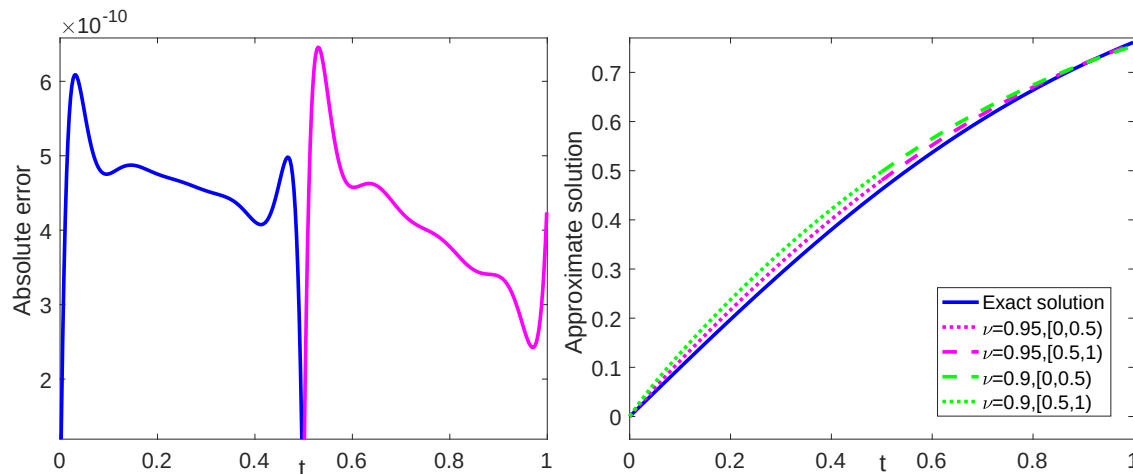


Figure 23: The absolute error (left) and approximate solution for different values of $\nu = 0.95, 0.9$ with exact solution (right) for $M = 7, K = 2$ of Example 6.2.

Acknowledgement

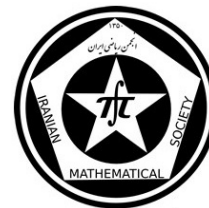
This work is supported by the National Elite Foundation and Alzahra University. We express our sincere thanks to the anonymous referees for their valuable suggestions that improved the final manuscript.

References

- [1] N. Engheta, *On fractional calculus and fractional multipoles in electromagnetism*, IEEE Trans. Antennas Propag. **44** (1996), 554–566.
- [2] J. He, *Approximate analytical solution for seepage flow with fractional derivatives in porous media*, Comput Methods Appl Mech Eng. **167** (1998), 57–68.
- [3] H. Dehestani, Y. Ordokhani and M. Razzaghi, *On the applicability of Genocchi wavelet method for different kinds of fractional-order differential equations with delay*, Numer Linear Algebra Appl., **26** (2019), e2259.
- [4] H. Dehestani, Y. Ordokhani and M. Razzaghi, *Fractional-order Bessel wavelet functions for solving variable order fractional optimal control problems with estimation error*, Int J Soc Syst Sci., **51**(6) (2020), 1032–1052.
- [5] H. Dehestani, Y. Ordokhani and M. Razzaghi, *Combination of Lucas wavelets with Legendre-Gauss quadrature for fractional Fredholm-Volterra integro-differential equations*, Int J Appl Comput Math., **382** (2021), 113070.



- [6] H. Dehestani, Y. Ordokhani and M. Razzaghi, *A novel direct method based on the Lucas multiwavelet functions for variable-order fractional reaction-diffusion and subdiffusion equations*, Numer Linear Algebra Appl., (2020), e2346.
- [7] Y. Li, *Solving a nonlinear fractional differential equation using chebyshev wavelets*, Commun Nonlinear Sci Numer Simul., **15**(9) (2009), 2284–2292.
- [8] M. H. Heydari, M. R. Hooshmandasl, C. Cattani, *A new operational matrix of fractional order integration for the Chebyshev wavelets and its application for nonlinear fractional Van der Pol oscillator equation*, Proc Math Sci., **128**(2) (2018), 26.
- [9] H. Dehestani, Y. Ordokhani and M. Razzaghi, *Fractional-order Bessel functions with various applications*, Appl Math., **64**(6) (2019), 637–662.



Jacobi polynomials based collocation method for solving singular fractional Lane-Emden equation

Monireh Nosrati Sahlan, Hajar Masoumi²³

Faculty of Mathematical Sciences, University of Bonab, Bonab, Iran

Abstract

The aim of this paper is solving fractional order of singular Lane-Emden equation by using spectral collocation method. Jacobi polynomials are the test functions of collocation method. The properties of Jacobi polynomials and collocation method are applied to reduce the fractional order Lane-Emden equation to a system of algebraic equations. The method is computationally attractive and applications are demonstrated through illustrative examples.

AMS subject Classification 2010: 26A33, 42C05

Keywords: Lane-Emden equation, fractional calculus, Jacobi polynomials.

1 Introduction

Since the beginning of stellar astrophysics, the research about stellar structure has been a significant problem. There have been continuous efforts to deduce the radial profiles of pressure, density and mass of a star and one of the key results that came out of these efforts are the Lane-Emden equations. First published by Jonathan Homer Lane in 1870 [3], the Lane-Emden equation describes the density profile of a gaseous star. Mathematically, the Lane-Emden equation is a second-order ODE with an arbitrary index, known as the polytropic index, involved in one of its terms. In astrophysics, the Lane-Emden equation is essentially a Poisson equation for the gravitational potential of a self-gravitating, spherically symmetric polytropic fluid [3, 5].

In this study we consider the singular fractional Lane-Emden equations in the form

$$\begin{cases} {}_c D^\gamma \nu(x) + \frac{\eta}{x^{\gamma-\mu}} {}_c D^\mu \nu(x) + \omega(x)\nu(x) = \psi(x), & 0 \leq x \leq 1, \\ \nu(0) = \lambda_1, & \nu'(0) = \lambda_2, \end{cases} \quad (385)$$

where ${}_c D^\gamma$ denotes the Caputo fractional derivative of order γ , η is non-negative real constant and $0 < \mu \leq 1$, $1 < \gamma \leq 2$ and $\omega(x), \psi(x) \in C[0, 1]$. In this paper, the unknown function ν is expressed by linear combination of orthogonal Jacobi polynomials with unknown coefficients. Utilizing collocation

²³Speaker



approach, the obtained equation transforms to an algebraic system. By solving the obtained system, the unknown coefficients and consequently the solution of equation (689) is achieved. Because of applying the Caputo fractional derivative in this paper, we briefly present its known definition.

Definition 0.62. The Caputo fractional derivative operator of order nonnegative α is defined as [6]

$${}_c D^\alpha f(x) = \frac{1}{\Gamma(n-\alpha)} \int_0^x \frac{f^{(n)}(t)}{(x-t)^{\alpha+1-n}} dt, \quad n-1 < \alpha \leq n, \quad n \in \mathbb{N}. \quad (386)$$

From the Caputo derivative definition, we get

$$\begin{aligned} {}_c D^\alpha x^\beta &= 0, \quad \beta \in \mathbb{N}_0, \quad \beta < [\alpha], \\ {}_c D^\alpha x^\beta &= \frac{\Gamma(\beta+1)}{\Gamma(\beta+1-\alpha)} x^{\beta-\alpha}, \quad \beta \in \mathbb{N}_0, \quad \beta \geq [\alpha] \text{ or } \beta \in \mathbb{R} - \mathbb{N}_0, \quad \beta > [\alpha]. \end{aligned} \quad (387)$$

2 Jacobi polynomials

Orthogonal Jacobi polynomials are the solutions of the following second order Jacobi differential equation

$$(1-x^2)y'' + (\beta - \alpha - (\alpha + \beta + 2)x)y' + n(n + \alpha + \beta + 1)y = 0,$$

for $\alpha, \beta > -1$. This polynomials are defined over interval $[-1, 1]$ and denoted by $J_n^{(\alpha, \beta)}(x)$. The n -th Jacobi polynomial is defined as follows

$$J_n^{(\alpha, \beta)}(x) = \frac{\Gamma(n + \alpha + 1)}{n! \Gamma(n + \alpha + \beta + 1)} \sum_{m=0}^n \binom{n}{m} \frac{\Gamma(n + \alpha + \beta + 1 + m)}{\Gamma(m + \alpha + 1)} \left(\frac{x-1}{2}\right)^m, \quad (388)$$

where

$$J_n^{(\alpha, \beta)}(1) = \frac{\Gamma(n + \alpha + 1)}{n! \Gamma(\alpha + 1)}, \quad J_n^{(\alpha, \beta)}(-1) = \frac{(-1)^n \Gamma(n + \beta + 1)}{n! \Gamma(\beta + 1)}. \quad (389)$$

This polynomials are orthogonal with weight function $\omega^{(\alpha, \beta)}(x) = (1-x)^\alpha (1+x)^\beta$ in their domain, that is

$$\int_{-1}^1 J_j^{(\alpha, \beta)}(x) J_i^{(\alpha, \beta)}(x) \omega^{(\alpha, \beta)}(x) dx = \delta_{ij} \frac{2^{\alpha+\beta+1} \Gamma(i + \alpha + 1) \Gamma(i + \beta + 1)}{(2i + \alpha + \beta + 1) i! \Gamma(i + \alpha + \beta + 1)}.$$

Also Rodrigues formula for Jacobi polynomials defined as

$$J_n^{\alpha, \beta}(x) = \frac{(-1)^n}{2^n n!} (1-x)^{-\alpha} (1+x)^{-\beta} \frac{d^n}{dx^n} [(1-x)^{n+\alpha} (1+x)^{n+\beta}].$$

3 Shifted Jacobi polynomial

Due to the domain of the unknown function of this study, by using the changing variable $x = 2\tilde{x} - 1$, the domain of Jacobi polynomials is shifted from $[-1, 1]$ to $[0, 1]$. That is we have Shifted Jacobi Polynomials (SJP) as

$$\tilde{J}_n^{(\alpha, \beta)}(x) := J_n^{(\alpha, \beta)}(2\tilde{x} - 1). \quad (390)$$



Analytical form of SJP of degree $n \geq 1$ is ([1]):

$$\tilde{J}_n^{(\alpha, \beta)}(x) = \frac{\Gamma(n + \beta + 1)}{\Gamma(n + \alpha + \beta + 1)} \sum_{m=0}^n \binom{n}{m} \frac{(-1)^{n-m} \Gamma(n + \alpha + \beta + 1 + m)}{\Gamma(m + \beta + 1)} x^m. \quad (391)$$

It is clear that SJP are orthogonal with relevant weight function $\tilde{\omega}^{(\alpha, \beta)}(x)$.

Theorem 0.63. ([2]) *The m -th derivative of the n -th SJP has the form*

$$\frac{d^m}{dx^m} J_n^{(\alpha, \beta)}(x) = \sum_{j=0}^{n-m} C_m(n, j, \alpha, \beta) J_j^{(\alpha, \beta)}(x), \quad (392)$$

where

$$C_m(n, j, \alpha, \beta) = \frac{(n + \alpha + \beta)_m (n + m + \alpha + \beta)_j (m + j + \alpha + 1)_{n-m-j} \Gamma(j + \alpha + \beta)}{2^m (n - m - j)! \Gamma(2j + \alpha + \beta)} \\ \times {}_3F_2 \left(\begin{matrix} -n + m + j & n + m + j + \alpha + \beta & j + \alpha + 1 \\ m + j + \alpha + 1 & 2j + \alpha + \beta + 1 \end{matrix} ; 1 \right),$$

and $(\cdot)_\lambda$ is the Pochhammer symbol and defined by $(a)_\lambda = \frac{\Gamma(a + \lambda)}{\Gamma(a)}$.

The value of the derivatives of this polynomials in 0 is

$$\left. \frac{d^m J_n^{(\alpha, \beta)}(x)}{dx^m} \right|_{x=0} = \frac{(-1)^{n-m} \Gamma(n + \beta + 1) (n + \alpha + \beta + 1)_m}{\Gamma(n - m + 1) \Gamma(m + \beta + 1)}, \quad m \leq n.$$

4 Function approximation

The function $f(t) \in L^2[0, 1]$ is approximated by SJP of degree M as

$$f(x) \simeq f_M(x) = \sum_{n=0}^M c_n \tilde{J}_n^{(\alpha, \beta)}(x) = C_M^T \tilde{\mathcal{J}}_M^{(\alpha, \beta)}(x),$$

where

$$C_M = (c_0, \dots, c_M)^T, \quad \tilde{\mathcal{J}}_M^{(\alpha, \beta)}(x) = \left(\tilde{J}_0^{(\alpha, \beta)}(x), \dots, \tilde{J}_M^{(\alpha, \beta)}(x) \right)^T,$$

where the coefficients c_i are determined by

$$c_n = \frac{1}{\tilde{\lambda}_n^{(\alpha, \beta)}} \int_0^1 \tilde{J}_n^{(\alpha, \beta)}(x) f(x) \tilde{\omega}^{(\alpha, \beta)}(x) dx.$$



5 Numerical implementation

In this section we use SJP and collocation method for solving (689). For this purpose, the unknown function is approximated by SJP as $\nu(x) \simeq \nu_M(x) = C_M^T \tilde{\mathcal{J}}_M^{(\alpha, \beta)}(x)$ and replaced in equation (689). Now we define the residual function $\mathcal{R}_M(x)$ as

$$\mathcal{R}_M(x) = {}_c D^\gamma \nu_M(x) + \frac{\eta}{x^{\gamma-\mu}} {}_c D^\mu \nu_M(x) + \omega(x)\nu(x) - \psi(x), \quad (393)$$

collocating the residual function in the points $\xi_j, j = 1, \dots, M-1$, where the points ξ_j are the roots of $\frac{d}{dx} \mathcal{J}_M^{(\alpha, \beta)}(x)$, and putting $\mathcal{R}_M(\xi_j) = 0$, we get

$$C_M^T \left[\left({}_c D^\gamma \tilde{\mathcal{J}}_M^{(\alpha, \beta)} \right) (\xi_j) + \frac{\eta}{\xi_j^{\gamma-\mu}} \left({}_c D^\mu \tilde{\mathcal{J}}_M^{(\alpha, \beta)} \right) (\xi_j) + \omega(\xi_j) \tilde{\mathcal{J}}_M^{(\alpha, \beta)}(\xi_j) \right] = \psi(\xi_j). \quad (394)$$

where the Caputo fractional derivatives ${}_c D^\gamma \tilde{\mathcal{J}}_M^{(\alpha, \beta)}$ and ${}_c D^\mu \tilde{\mathcal{J}}_M^{(\alpha, \beta)}$, are obtained using the definition of SJPs and (387). On the other hands for initial conditions, we can write

$$\nu_M(0) = C_M^T \tilde{\mathcal{J}}_M^{(\alpha, \beta)}(0) = \lambda_1, \quad \nu'_M(0) = C_M^T \left(\tilde{\mathcal{J}}_M^{(\alpha, \beta)} \right)'(0) = \lambda_2, \quad (395)$$

combining the equations (394) together with equations (611), we obtain a system of $M+1$ algebraic equation with $M+1$ unknowns. By solving this system, the solution of the fractional order Lane-Emden equation is achieved.

6 Illustrative example

Example 0.64. Consider the following form of equation (689):

$$\begin{cases} {}_c D^{1.5} \nu(x) + \frac{1}{x} {}_c D^{0.5} \nu(x) + \sqrt{x} \nu(x) = \psi(x), & 0 \leq x \leq 1, \\ \nu(0) = 0, & \nu'(0) = 0, \end{cases} \quad (396)$$

where

$$\psi(x) = \sqrt{\frac{x}{\pi}} \left(\frac{56x}{5} - \frac{20}{3} \right) + \sqrt{x} (x^3 - x^2).$$

The exact solution of equation (612) is $\nu(x) = x^3 - x^2$, this equation is solved by the presented method for $\alpha = -0.25$, $\beta = -0.75$ and $N = 5, 7, 10$ and the absolute error of numerical results are given in table 28 and compared with the results of [4], where applied power series technique via collocation method for solving fractional order Lane-Emden equation. The numerical results show the high accuracy of the presented approach.

Conclusion

In this paper the spectral collocation method based on shifted Jacobi polynomials are used for solving the fractional order Lane-Emden equation. The zeros of the first derivative of this polynomials are employed as collocation points. The method is computationally attractive and simple to use.



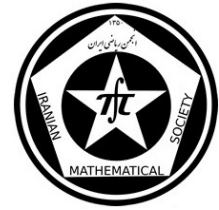
Table 28: Absolute error of approximated solution

x	$M = 5$	$M = 7$	$M = 10$	[4] $n = 10$
0.25	$3.1286e - 5$	$3.0407e - 7$	$1.9148e - 9$	$1.3232e - 5$
0.50	$5.5133e - 5$	$7.4236e - 7$	$5.2104e - 9$	$2.6342e - 5$
0.75	$3.4555e - 5$	$8.0003e - 7$	$1.7261e - 9$	$1.5634e - 6$
1.00	$9.1580e - 6$	$2.3514e - 7$	$6.1007e - 9$	$4.1443e - 5$

References

- [1] Y. Luke, The Special Functions and Their Approximations, vol. 2, Academic Press, New York, 1969.
- [2] E.H. Doha, On the construction of recurrence relations for the expansion and connection coefficients in series of Jacobi polynomials, J. Phys. A: Math. Gen. 37 (2004) 657675.
- [3] J.H. Lane, On the theoretical temperature of the sun under the hypothesis of a gaseous mass maintaining its volume by its internal heat and depending on the laws of gases known to terrestrial experiment, The American Journal of Science and Arts, 5 (1870), 57-74.
- [4] M.S. Mechee, N. Senu, Numerical Study of Fractional Differential Equations of Lane-Emden Type by Method of Collocation, Applied Mathematics, 3 (2012), 851-856.
- [5] E. Momoniat, and C. Harley, Approximation implicit solution of a Lane-Emden equation, New Astronomy, 11 (2006), 520-526.
- [6] I. Podlubny, Fractional differential equations: An introduction to fractional derivatives, fractional differential equations to methods of their solution and some of their applications, 1st ed. Academic Press, New York, 1998.

Authors email address: nosrati@ubonab.ac.ir
Masoomih99@gmail.com



Legendre wavelet method combined with the Gauss quadrature rule for numerical solution of fractional integro-differential equations

Mohsen Riahi²⁴

Department of Applied Mathematics, Higher Education Complex of Saravan, Saravan, Iran, m.riahi@saravan.ac.ir

Abstract

In this work, we propose a novel technique for solving the nonlinear fractional Volterra-Fredholm integro-differential equations (FVFIDEs). This method approximates the unknown function with the Legendre wavelets. To do this, the Legendre wavelets are used in conjunction with the quadrature rule for converting the problem into a nonlinear system of algebraic equations which can be easily solved by applying the mathematical programming techniques. Furthermore, the existence, uniqueness and convergence analysis of the solution are proved by preparing some theorems and lemmas. Moreover, one example is presented and their results are compared to the results of Chebyshev wavelet, Nyström and Newton-Kantorovitch methods to show the capability and accuracy of this scheme.

AMS subject Classification 2010: 47G20, 65T60, 34A08.

Keywords: Legendre wavelet, Gaussian quadrature, FVFIDEs.

1 Introduction

Applications of fractional differential equations and fractional integral equations create a wide area of research for many researchers. It has been applied to model the nonlinear oscillation of earthquakes, fluid-dynamic traffic and etc.. In order to better analysis these systems, it is required to obtain the solution of these equations. But, mostly finding the analytical solution of these equations cannot be possible, so considering more accurate numerical solutions can be helpful. In the literature, there are various technique for solving these kinds of equations. For example, Chebyshev wavelet method [1], Nyström and Newton-Kantorovitch methods [2], exp-function method [3], Euler wavelet method [4] and etc..

In this work, the Legendre wavelets are implemented to obtain an approximate solution for nonlinear FVFIDEs. We consider the fractional FVFIDEs in a Banach space as follows

$$({}_0^C D_x^\alpha y)(x) + Vy(x) + Fy(x) = g(x), \quad m-1 < \alpha \leq m, \quad m \in \mathbb{N}, \quad (397)$$

²⁴Speaker



where ${}_0^C D_x^\alpha$ is in the Caputo sense fractional derivative, $g(x)$ is a given continuous function, $x \in J = [a, b]$, $y(x)$ is unknown function $Vy(x) = \int_a^x k_1(x, \xi) N_1(y(\xi)) d\xi$, $Fy(x) = \int_a^b k_2(x, \xi) N_2(y(\xi)) d\xi$, in which $k_i : J \times J \rightarrow \mathbb{R}$, $i = 1, 2$ are continuous functions and $N_i : J \times \mathbb{R} \rightarrow \mathbb{R}$, $i = 1, 2$ are Lipschitz nonlinear continuous functions. Eq. (689) is subject to the following mixed boundary conditions

$$\sum_{j=1}^m \left[\lambda_{ij} y^{(j-1)}(a) + \eta_{ij} y^{(j-1)}(b) \right] = \gamma_i, \quad i = 1, 2, \dots, m. \quad (398)$$

The main characteristic of this technique is that it reduces these problems to a system of algebraic equations. This approach is based on converting the FVFIDEs into a mixed Volterra-Fredholm integral equations through integration, approximating various terms involved in the equation by truncated Legendre wavelet series and using the operational matrices to eliminate the integral, derivation and product operations.

2 Legendre wavelets and functions approximation

Properties of Legendre wavelets

Legendre wavelets $\psi_{n,m}(x) = \psi(k, \hat{n}, m, x)$ have four arguments; $\hat{n} = 2n - 1$ for $n = 1, 2, \dots, 2^{k-1}$, $k \in \mathbb{Z}^+$, m as the order of Legendre polynomials and x which is the normalized time. They are defined on the interval $[0, 1)$ by

$$\psi_{n,m}(x) = \psi(k, \hat{n}, m, x) = \begin{cases} 2^{\frac{k-1}{2}} (2m+1)^{\frac{1}{2}} L_m^*(x), & \frac{\hat{n}-1}{2^k} \leq x < \frac{\hat{n}+1}{2^k}, \\ 0, & \text{otherwise,} \end{cases} \quad (399)$$

where $m = 0, 1, \dots, M-1$, $n = 1, 2, \dots, 2^{k-1}$, the dilation and translation parameters are $a = 2^{-k}$ and $b = \hat{n}2^{-k}$, respectively. $L_m^*(x) = L_m(2^k x - \hat{n})$ are shifted Legendre polynomials on interval $[0, 1)$ which are orthogonal with respect to the weight function $w(x) = 1$.

3 Function approximation by Legendre wavelets

A function $y(x)$, may be expressed in terms of the Legendre wavelet in $[0, 1)$ as

$$y(x) \simeq \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} y_{n,m} \psi_{n,m}(x) = Y^T \Psi(x), \quad (400)$$

where $y_{n,m} = \langle y(x), \psi_{n,m}(x) \rangle$ and $\langle \cdot, \cdot \rangle$ denotes the inner product and

$$Y = [y_{1,0}, y_{1,1}, \dots, y_{1,M-1}, y_{2,0}, y_{2,1}, \dots, y_{2,M-1}, \dots, y_{2^{k-1},0}, y_{2^{k-1},1}, \dots, y_{2^{k-1},M-1}]^T, \quad (401)$$

$$\Psi(x) = [\psi_{1,0}(x), \psi_{1,1}(x), \dots, \psi_{1,M-1}(x), \dots, \psi_{2^{k-1},0}(x), \psi_{2^{k-1},1}(x), \dots, \psi_{2^{k-1},M-1}(x)]^T. \quad (402)$$

Similarly, we can approximate the functions $k_i(x, \xi) \in L^2([0, 1] \times [0, 1])$ as

$$k_i(x, \xi) = \Psi(x)^T K_i \Psi(\xi), \quad (403)$$

where K_i , $i = 1, 2$ are $2^{k-1}M \times 2^{k-1}M$ matrices.



4 Gauss quadrature Legendre wavelet method (GQLWM)

Let us consider the nonlinear FVFIDE with mixed boundary conditions (689)-(609). To approximate the functions $y(x)$ and $g(x)$ by $y(x) = Y^T \Psi(x)$ and $g(x) = G^T \Psi(x)$ assume that

$$N_1(y(\xi)) = u(\xi), \quad N_2(y(\xi)) = v(\xi), \quad (404)$$

where $u(\xi)$ and $v(\xi)$ are as follows

$$u(\xi) = U^T \Psi(\xi), \quad v(\xi) = V^T \Psi(\xi), \quad (405)$$

in which U^T and V^T defined similarly as in (612). Applying operator ${}_0I_x^\alpha$ on both sides of Eq. (689), results in

$$\begin{aligned} y(x) - \sum_{k=0}^{n-1} y^{(k)}(0^+) \frac{x^k}{k!} &= \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} g(\tau) d\tau - \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} \int_0^\tau k_1(\tau, \xi) u(\xi) d\tau d\xi \\ &\quad - \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} \int_0^1 k_2(\tau, \xi) v(\xi) d\tau d\xi. \end{aligned} \quad (406)$$

Replacing the exact solution $y(x)$ by $Y^T \Psi(x)$ and using their approximations by (611), (603)-(536), one gets

$$\begin{aligned} Y^T \Psi(x) - \sum_{k=0}^{n-1} Y^T \Psi^{(k)}(0^+) \frac{x^k}{k!} &= \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} G^T \Psi(\tau) d\tau \\ &\quad - \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} \int_0^\tau \Psi(\tau)^T K_1 \Psi(\xi) U^T \Psi(\xi) d\tau d\xi \\ &\quad - \frac{1}{\Gamma(\alpha)} \int_0^x (x-\tau)^{\alpha-1} \int_0^1 \Psi(\tau)^T K_2 \Psi(\xi) V^T \Psi(\xi) d\tau d\xi. \end{aligned} \quad (407)$$

Then, collocating (538) in $M2^{k-1}$ Legendre wavelet-Gauss collocation points $x_j, j = 1, 2, \dots, M2^{k-1}$ on interval $[0,1]$. In a similar way, the mixed boundary conditions (609) are approximated as follows

$$\sum_{j=1}^m \left[\lambda_{ij} Y^T \Psi^{(j-1)}(0) + \eta_{ij} Y^T \Psi^{(j-1)}(1) \right] = \gamma_i, \quad i = 1, 2, \dots, m. \quad (408)$$

Now, we use the quadrature rule to approximate the integral involved in this equation, which has zero error integration for polynomial integrands of degree less than or equal to $1 + M2^k$ with $M2^{k-1}$ Legendre-Gauss nodes. To do this, $M2^{k-1}$ intervals $[0, x_j]$ are transferred to a fixed interval $[-1,1]$ through the transformations $\tau = (x_j/2)(s+1)$. Taking into account Eqs. (604)-(536) and using collocation points $x_j, j = 1, 2, \dots, M2^{k-1}$, one obtain

$$N_1(Y^T \Psi(x_j)) = U^T \Psi(x_j), \quad N_2(Y^T \Psi(x_j)) = V^T \Psi(x_j). \quad (409)$$

Applying the Gaussian quadrature rule on system (538) and combining Eqs. (538)-(543), the main problem is reduced to a system of nonlinear algebraic equations. By solving this system, the unknown values Y^T will be obtained and by relation (611), the approximate solution of system (689) will be determined.



5 Existence, uniqueness and convergence analysis

Lemma 0.65. Suppose that for $y(x) \in [0, 1]$, there exist $\beta_y \in \mathbb{R}$ such that $|y(x)| \leq \beta_y$. Then, the sum of Legendre coefficients of $y(x)$ defined in (611) is absolutely convergent if $|y_{n,m}| \leq \sqrt{2^{1-k}} \beta_y$.

Theorem 0.66. The series solution $\tilde{y}(x) = \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} y_{n,m} \psi_{n,m}(x)$ defined by (611) is convergent with respect to L^2 -norm on $[0, 1]$ if the sum of absolute values of the Legendre coefficients $\sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} |y_{n,m}|$ for continuous function $y(x)$ forms a convergent series.

Proof. Let $L^2(\mathbb{R})$ be a Hilbert space and $\psi_{n,m}$ defined in (610) form an orthogonal basis with $\psi_{n,m}(x) = \Delta_j(x)$ and $\delta_j = \langle \tilde{y}(x), \Delta_j(x) \rangle$. Define a sequence of partial sums $\{S_n\}$ as: $S_n(x) = \sum_{j=0}^n \delta_j \Delta_j(x)$. Now, for every $\epsilon > 0$ there exists $N_\epsilon > 0$ such that for every $n > m > N_\epsilon$,

$$\|S_n(x) - S_m(x)\|_2^2 = \int_0^1 \left| \sum_{l=m+1}^n \delta_l \Delta_l(x) \right|^2 dx \leq \sum_{l=m+1}^n |\delta_l|^2 \int_0^1 |\Delta_l(x)|^2 dx = \sum_{l=m+1}^n |\delta_l|^2.$$

Using Lemma 0.65, $\sum_{l=0}^\infty |\delta_l|^2$ is absolutely convergent. Hence, according to Cauchy criterion we have $\sum_{l=m+1}^n |\delta_l|^2 < \epsilon^2$. So, $\|S_n(x) - S_m(x)\|_2 \leq \sqrt{\epsilon^2} = \epsilon$. Thus, the sequence of partial sum of the series is a Cauchy sequence. Therefore, it convergent with respect to L^2 -norm and this completes the proof. $\square$

In order to study the uniqueness of the solution we consider the FVFIDEs (689) in an operator form as

$$({}_0^C D_x^\alpha y)(x) = g(x) - \mathcal{K}_1 \mathcal{N}_1 y - \mathcal{K}_2 \mathcal{N}_2 y, \quad (410)$$

where $\mathcal{K}_1 \mathcal{N}_1 y = \int_a^x k_1(x, \xi) N_1(y(\xi)) d\xi$, $\mathcal{K}_2 \mathcal{N}_2 y = \int_a^b k_2(x, \xi) N_2(y(\xi)) d\xi$. Applying operator ${}_0 I_x^\alpha$ on both sides of Eq. (615), one obtains

$$y(x) = \sum_{k=0}^{n-1} y^{(k)}(0^+) x^k / k! + {}_0 I_x^\alpha [g(x) - \mathcal{K}_1 \mathcal{N}_1 y - \mathcal{K}_2 \mathcal{N}_2 y], \quad (411)$$

This equation can be written as $\mathcal{T}y = y$. Let $(C[0, 1], \|\cdot\|_\infty)$ be a Banach space of all continuous functions with the norm $\|h\|_\infty = \max_x |h(x)|$. Also, the operators $\mathcal{N}_1$ and $\mathcal{N}_2$ satisfies the Lipschitz condition on $[0, 1]$ with Lipschitz constants J_1 and J_2 .

Theorem 0.67. Suppose that the nonlinear operators $\mathcal{N}_1$ and $\mathcal{N}_2$ satisfies in the relation $J_1 \|\mathcal{N}_1\|_\infty + J_2 \|\mathcal{N}_2\|_\infty < \Gamma(\alpha + 1)$. Then the FVFIDE (689) has an unique solution.

Proof. Let $\mathcal{T} : C[0, 1] \rightarrow C[0, 1]$ be defined as

$$\mathcal{T}y(x) = f(x) + \frac{1}{\Gamma(\alpha)} \int_0^x (x - \xi)^{\alpha-1} [g(\xi) - \mathcal{K}_1 \mathcal{N}_1 y(\xi) - \mathcal{K}_2 \mathcal{N}_2 y(\xi)] d\xi. \quad (412)$$

Suppose that $\tilde{y}, y \in C[0, 1]$, then for every positive x we have

$$\mathcal{T}\tilde{y}(x) - \mathcal{T}y(x) = \frac{1}{\Gamma(\alpha)} \int_0^x (x - \xi)^{\alpha-1} [\mathcal{K}_1 \mathcal{N}_1 y(\xi) - \mathcal{K}_1 \mathcal{N}_1 \tilde{y}(\xi) + \mathcal{K}_2 \mathcal{N}_2 y(\xi) - \mathcal{K}_2 \mathcal{N}_2 \tilde{y}(\xi)] d\xi.$$



Therefore, one obtain

$$\begin{aligned} |\mathcal{T}\tilde{y}(x) - \mathcal{T}y(x)| &\leq \frac{1}{\Gamma(\alpha)} \int_0^x |x - \xi|^{\alpha-1} [|\mathcal{K}_1| |\mathcal{N}_1 y(\xi) - \mathcal{N}_1 \tilde{y}(\xi)| + |\mathcal{K}_2| |\mathcal{N}_2 y(\xi) - \mathcal{N}_2 \tilde{y}(\xi)|] d\xi \\ &\leq \frac{1}{\Gamma(\alpha)} \int_0^x |x - \xi|^{\alpha-1} [|\mathcal{K}_1| J_1 |\tilde{y}(\xi) - y(\xi)| + |\mathcal{K}_2| J_2 |\tilde{y}(\xi) - y(\xi)|] d\xi \\ &\leq \frac{1}{\Gamma(\alpha+1)} (\|\mathcal{K}_1\|_\infty J_1 + \|\mathcal{K}_2\|_\infty J_2) \|\tilde{y}(\xi) - y(\xi)\|_\infty. \end{aligned}$$

Hence, $\|\mathcal{T}\tilde{y}(x) - \mathcal{T}y(x)\|_\infty \leq L \|\tilde{y}(\xi) - y(\xi)\|_\infty$, where $L = [1/\Gamma(\alpha+1)] (\|\mathcal{K}_1\|_\infty J_1 + \|\mathcal{K}_2\|_\infty J_2)$. Since $L < 1$ by contraction mapping theorem, this problem has an unique solution in $C[0, 1]$. $\square$

Example 0.68. Consider the following nonlinear FVFIDEs

$$\left({}^C_0 D_x^{\frac{\sqrt{7}}{2}} y \right) (x) - \int_0^x \frac{1+2\xi}{1+y(\xi)} d\xi - \int_0^1 (1+2\xi)y(\xi)d\xi = g(x), \quad (413)$$

subject to the boundary conditions $y(0) = 0$ and $y(1) = 2$ in which

$$g(x) = \frac{2x^{2-\frac{\sqrt{7}}{2}}}{\Gamma\left(3-\frac{\sqrt{7}}{2}\right)} - \ln(x^2 + x + 1) + 1 - e^2.$$

The exact solution of this problem is $y(x) = x^2 + x$. Numerical results for example 0.106 are indicated in Table 54, which show that the results of our method are better than the results obtained in [2, 1] by Chebyshev wavelet method (CWM), Nyström and Newton-Kantorovitch methods, respectively. Table 30 shows the maximum absolute error (MAE) between the exact and approximate solutions for various choices of M and k . The outcomes reveal that the results of GQLWM by using only a small number of bases are very promising and superior to CWM and Nyström and Newton-Kantorovitch methods.

Table 29: Comparison between our method, CWM [1] and method in [2] for example 0.106.

x		0.1	0.3	0.5	0.7	0.9
Error [2]	$k = 3, n = 160$	$7.81E - 06$	$1.55E - 05$	$1.65E - 05$	$1.27E - 05$	$5.07E - 06$
	$k = 3, n = 320$	$1.92E - 06$	$3.84E - 06$	$4.10E - 06$	$3.15E - 06$	$1.25E - 06$
	$k = 3, n = 640$	$4.65E - 07$	$9.31E - 07$	$9.92E - 07$	$7.65E - 07$	$3.04E - 07$
Error [1] for $k = 1, M = 19$		$3.47E - 12$	$6.08E - 12$	$6.21E - 12$	$4.68E - 12$	$1.84E - 12$
GQLWM	$k = 2, M = 4$	$7.81E - 06$	$1.55E - 05$	$1.65E - 05$	$1.27E - 05$	$5.07E - 06$
	$k = 3, M = 5$	$1.92E - 06$	$3.84E - 06$	$4.10E - 06$	$3.15E - 06$	$1.25E - 06$
	$k = 4, M = 6$	$4.65E - 07$	$9.31E - 07$	$9.92E - 07$	$7.65E - 07$	$3.04E - 07$
	$k = 1, M = 10$	$4.65E - 07$	$9.31E - 07$	$9.92E - 07$	$7.65E - 07$	$3.04E - 07$
	$k = 1, M = 19$	$4.65E - 07$	$9.31E - 07$	$9.92E - 07$	$7.65E - 07$	$3.04E - 07$



Table 30: Comparison between our method, CWM [1] and method in [2] for example 0.106.

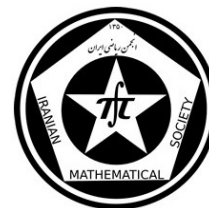
M		4	7	10	13	16	19
MAE	$k = 1$	$4.41E - 04$	$4.06E - 08$	$1.03E - 12$	$4.02E - 16$	$3.01E - 23$	$1.57E - 28$
	$k = 2$	$7.68E - 05$	$9.41E - 10$	$3.05E - 15$	$3.96E - 21$	$2.55E - 27$	$1.61E - 34$
	$k = 3$	$5.46E - 06$	$8.23E - 12$	$3.39E - 18$	$5.49E - 25$	$4.42E - 32$	$2.02E - 39$

6 Conclusion

In this paper, the GQLWM which is an efficient technique for the numerical solution of nonlinear FVFIDEs has been proposed. Using this method, the system of nonlinear FVFIDEs was reduced to a system of algebraic equations. By solving this nonlinear system, the numerical solution was obtained. Moreover, the existence, uniqueness and convergence analysis of the proposed method were discussed. It was evident that, for a certain value of k , as M increased, the accuracy of the GQLWM was increased. Also, for a certain value of M , as k increased, the accuracy was increased, as well. Moreover, the results showed that the GQLWM was suitable and a powerful technique for solving the nonlinear FVFIDEs.

References

- [1] S.T. Mohyud-Din, H. Khan, M. Arif, M. Rafiq, *Chebyshev wavelet method to nonlinear fractional volterra fredholm integro-differential equations with mixed boundary conditions*, Advances in Mechanical Engineering, 9 (2017), 1687814017694802.
- [2] M. Jahanshahi et al., *numerical solution of nonlinear fractional volterra-fredholm integro-differential equations with mixed boundary conditions*, Inter-national Journal of Industrial Mathematics, 7 (2015), 63–69.
- [3] O. Guner, A. Bekir, *Exp-function method for nonlinear fractional differential equations*, Non-linear Sci Lett, 8 (2017), 41–49.
- [4] Y. Wang, L. Zhu, *Solving nonlinear volterra integro-differential equations of fractional order by using euler wavelet method*, Advances in Difference Equations, 27 (2017).
- [5] L. Zhu, Q. Fan, *Solving fractional nonlinear fredholm integro-differential equations by the second kind chebyshev wavelet*, Communications in Nonlinear Science and Numerical Simulation 17 (2012), 2333–2341.



Lucas polynomials collocation method for solving fractional order electrohydrodynamics flow model

Monireh Nosrati Sahlan²⁵, , Hajar Masoumi

Faculty of Mathematical Sciences, University of Bonab, Bonab, Iran.

Abstract

The aim of this paper is studying the fractional order nonlinear differential equations arising in electrohydrodynamics flow of a fluid in a circular cylindrical conduit. For this purpose Lucas polynomials are employed via collocation method for reducing the nonlinear singular differential equation of fractional order to an algebraic system. The obtained system is solved by Newton iteration method. The method is computationally simple and attractive.

AMS subject Classification 2010: 76W05, 26A33, 42C05.

Keywords: Electrohydrodynamics flow, Caputo fractional derivative, Lucas polynomials, collocation method.

1 Introduction

Because of extensive applications in different fields of science and engineering, electrohydrodynamics (EHD) fluid systems have attracted the attention of researchers in recent years. The study of the motion of ionized fluid particles and their interactions with electric field and the surrounding fluid that controls the transport phenomenon in the fluid flow is called EHD fluid. The mathematical model of EHD fluid is a nonlinear ordinary differential equation and the governed fractional differential equation of EHD flow is given in [4] by

$${}^c D^\beta \nu(\tau) + \frac{1}{\tau} {}^c D^\mu \nu(\tau) + \mathcal{H}^2 \left(1 - \frac{\nu(\tau)}{1 - \alpha \nu(\tau)} \right) = 0, \quad 0 < \tau < 1, \quad (414)$$

subject to the following boundary conditions

$$\nu'(0) = 0, \quad \nu(1) = 0,$$

where $\nu(\tau)$ is the unknown of equation (689), states the fluid speed, and the variable τ is the radial range measured from the centre of the conduit, also $\mathcal{H}$ is Hartmann electric number and nondimensional variable α denotes the degree of nonlinearity. The fractional derivative in equation (689) is considered in Caputo sense, where β and μ denote the order of fractional derivatives. In the following the fractional order boundary value problem (689) is solved by Lucas polynomials via collocation method.

²⁵Speaker



2 Lucas polynomials

The n -th order Lucas polynomial, $u_n(\tau)$, is the solution of the following differential equation [5]

$$(\tau^2 + 4) \frac{d^2 u}{d\tau^2} + \tau \frac{du}{d\tau} - n^2 u = 0.$$

If $L_0(\tau) = 2$ and $L_1(\tau) = \tau$, then the three term recurrence relation for these polynomials, is defined as

$$L_n(\tau) = \tau L_{n-1}(\tau) + L_{n-2}(\tau), \quad n \geq 2. \quad (415)$$

These polynomials are orthogonal with respect to the weight function $\omega(\tau) = \frac{1}{l\sqrt{\tau^2 + 4}}$ on the interval $[-2l, 2l]$:

$$\langle L_n, L_m \rangle = \int_{-2l}^{2l} \frac{L_n(\tau) L_m(\tau)}{l\sqrt{\tau^2 + 4}} d\tau = \begin{cases} 0; & m \neq n \\ \frac{\pi}{2}; & m = n > 0, \\ \pi; & m = n = 0. \end{cases} \quad (416)$$

Also the explicit power form representation of Lucas polynomial is [3]

$$L_n(\tau) = n \sum_{k=0}^{[n/2]} \frac{(n-k-1)!}{k!(n-2k)!} \tau^{n-2k}. \quad (417)$$

The Lucas polynomials satisfy

$$L_n(0) = 1 + (-1)^n, \quad L_n(1) = L_n,$$

where the L_n s are Lucas numbers and first few of them are

$$L_2 = 3, \quad L_3 = 4, \quad L_4 = 7, \quad L_5 = 11, \quad L_6 = 18, \dots$$

Caputo fractional derivative: Using explicit form for Lucas polynomials and the definition of Caputo fractional order derivative [6], we have

$${}^c D^\gamma L_n(\tau) = n \sum_{k=0}^{[n/2]} \frac{(n-k-1)!}{k! \Gamma(n-2k-\gamma+1)} \begin{cases} 0; & n-2k < \gamma \\ \tau^{n-2k-\gamma}; & \text{otherwise.} \end{cases}$$

Matrix representation: The Lucas polynomials $L_n(\tau)$ can be written in the matrix form as

$$\mathbf{L}_N(\tau) = P_N \Theta_N(\tau),$$

where

$$\mathbf{L}_N(\tau) = [L_0(\tau), L_1(\tau), \dots, L_N(\tau)]^T, \quad \Theta_N(\tau) = [1 \ \tau \ \tau^2 \dots \tau^N]^T,$$



and P_N is $(N + 1)$ -dimensional square lower triangular matrix, where its diagonal entries are 1. For odd and even N , the matrix P is defined differently [2]. For $N = 2k + 1$, matrix P has the following form

$$P_{2k+1} = \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & & & & \vdots \\ 2 & 0 & \frac{2k}{k+1} \begin{pmatrix} k+1 \\ k-1 \end{pmatrix} & 0 & \frac{2k}{k+2} \begin{pmatrix} k+2 \\ k-2 \end{pmatrix} & \dots & 0 \\ 0 & \frac{2k+1}{k+1} \begin{pmatrix} k+1 \\ k \end{pmatrix} & 0 & \frac{2k+1}{k+2} \begin{pmatrix} k+2 \\ k-1 \end{pmatrix} & 0 & \dots & 1 \end{pmatrix},$$

and for $N = 2k$, matrix P_{2k} is the first $(2k + 1) \times (2k + 1)$ block matrix of P_{2k+1} .

It should be noticed that the Lucas polynomials are related to the Chebyshev polynomials $T_n(\tau) = \cos(n \cos^{-1} \tau)$, by

$$L_n(\tau) = 2i^{-n} T_n\left(\frac{i\tau}{2}\right), \quad (418)$$

so the roots of Lucas polynomials are derived from the roots of Chebyshev polynomials in the form

$$\tau_k = 2i \cos\left((2k + 1)\frac{\pi}{2n}\right), \quad k = 0, 1, \dots, n - 1. \quad (419)$$

Function approximation: A function $\phi \in L^2(\mathbb{R})$ could be represented by Lucas polynomials as

$$\phi(\tau) \simeq \phi_N(\tau) = \sum_{j=0}^N \alpha_j L_j(\tau) = A_N^T \mathbf{L}_N(\tau), \quad (420)$$

where

$$\alpha_j = \frac{\langle \phi, L_j \rangle}{\langle L_j, L_j \rangle}, \quad A_N = [\alpha_0 \quad \alpha_1 \quad \dots \quad \alpha_N]^T.$$

By using matrix representation of Lucas polynomials, equation (420) is rewritten as

$$\phi(\tau) \simeq A_N^T P_N \Theta_N(\tau). \quad (421)$$

3 Numerical implementation

In this section first the unknown function of equation (689) is approximated by Lucas polynomials as

$$\nu(\tau) \simeq \nu_N(\tau) = C_N^T P \Theta_N(\tau), \quad (422)$$

where C_N is the unknown coefficients of approximation. The residual error function is defined by substituting the approximate solution $\nu_N(\tau)$ in the equation (689), as

$$Res_N(\tau) := {}^c D^\beta \nu_N(\tau) + \frac{1}{\tau} {}^c D^\mu \nu_N(\tau) + \mathcal{H}^2 \left(1 - \frac{\nu_N(\tau)}{1 - \alpha \nu_N(\tau)} \right), \quad (423)$$



now by collocating the equation (421) in the roots of Lucas polynomial $L_N(\tau)$, defined in (419), we get

$$Res_N(\tau_k) = \left[{}^c D^\beta \nu_N(\tau) + \frac{1}{\tau} {}^c D^\beta \nu_N(\mu) + \mathcal{H}^2 \left(1 - \frac{\nu_N(\tau)}{1 - \alpha \nu_N(\tau)} \right) \right]_{\tau=\tau_k}, \quad k = 1, \dots, N-1. \quad (424)$$

The residual function is forced to vanished at the given collocation points, that is $Res_N(\tau_k) = 0$, that is

$$\left[{}^c D^\beta \nu_N(\tau) + \frac{1}{\tau} {}^c D^\mu \nu_N(\tau) + \mathcal{H}^2 \left(1 - \frac{\nu_N(\tau)}{1 - \alpha \nu_N(\tau)} \right) \right]_{\tau=\tau_k} = 0, \quad (425)$$

we can rewrite the equation (425) as following

$$\left[(1 - \alpha \nu_N(\tau)) \left({}^c D^\beta \nu_N(\tau) + \frac{1}{\tau} {}^c D^\mu \nu_N(\tau) + \mathcal{H}^2 \right) - \mathcal{H}^2 \nu_N(\tau) \right]_{\tau=\tau_k} = 0,$$

by substituting (422) in the current relation, we get

$$\begin{aligned} & (1 - \alpha C_N^T P_N \Theta_N(\tau_k)) C_N^T P_N \left[({}^c D^\beta \Theta_N(\tau))_{\tau_k} + \frac{1}{\tau_k} ({}^c D^\mu \Theta_N(\tau))_{\tau_k} \right] \\ & + (1 - \alpha C_N^T P_N \Theta_N(\tau_k)) \mathcal{H}^2 - \mathcal{H}^2 C_N^T P_N \Theta_N(\tau_k) = 0. \end{aligned} \quad (426)$$

On the other hand using boundary conditions we have

$$\nu'(0) \simeq \nu'_N(0) = \left(C_N^T P_N D \Theta_N(\tau) \right)_{\tau=0}, \quad \nu(1) \simeq \nu_N(1) = C_N^T P_N \Theta_N(1). \quad (427)$$

System of equations (426) together with the relations (427) give a $(N+1) \times (N+1)$ nonlinear algebraic system. The obtained system is solved by Newton iteration method. For using Newton method, due to the initial conditions of problem (689), we put the initial value of method $(C_N^{(0)})$ as

$$C_N^{(0)} = \left(\frac{1}{2}, 0, \dots, 0 \right)^T.$$

4 Results

In this section we solve the boundary value problem (689) for integer order derivative $\beta = 2$, $\mu = 1$ and for fractional order derivative order $\beta = 1.9$, $\mu = 0.9$ and $N = 8$. The two important parameters in this problem are α and $\mathcal{H}^2$. For showig the affect of these parameters in the numerical solutions, different values for these parameters are choosed. The plot of results for $\beta = 2$ and $\mu = 1$ are reported in figure 1 and the numerical solutions achived by presented method (LPCM) for $\beta = 1.9$ and $\mu = 0.9$ and also the results of Reproducing Kernel Method (RKM) [1] in some arbitrary points are given in table 31. The comparison of the results of two method shows that the introduced method is in good agreement with RKM.

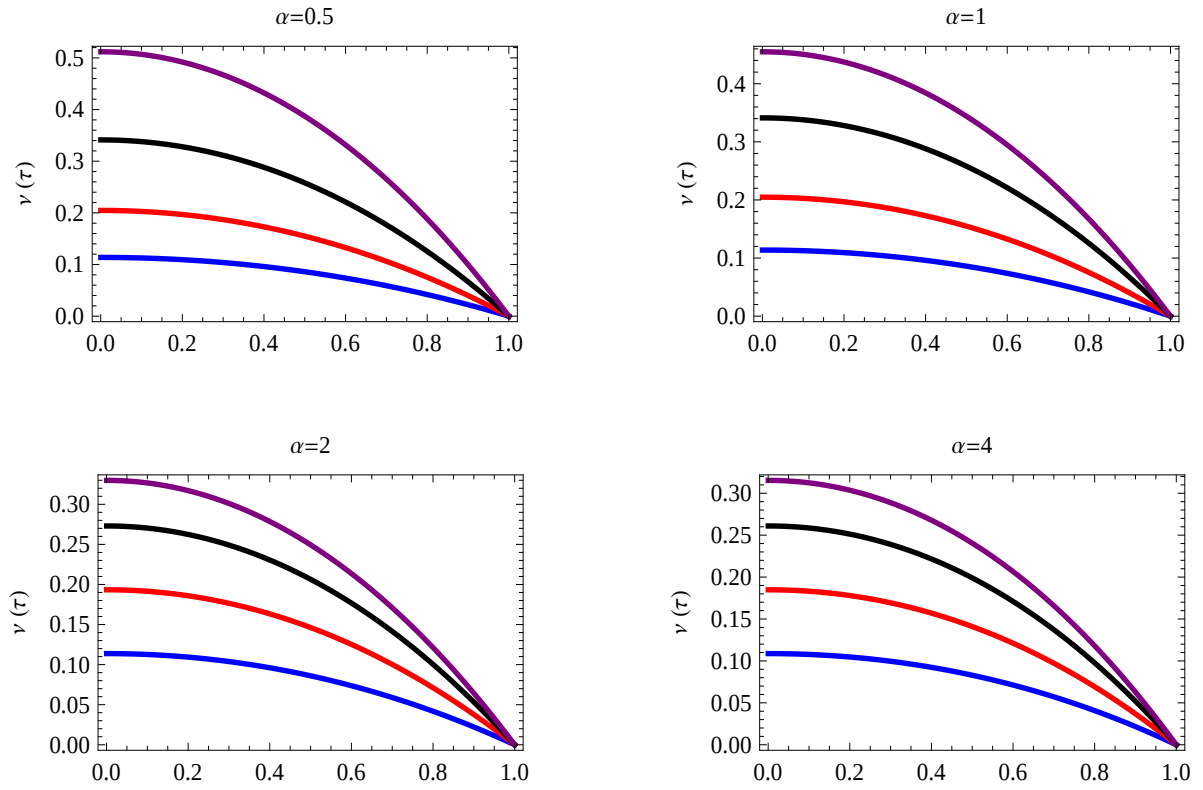


Figure 24: Approximated solution for $\beta = 2$ and $\mu = 1$ and different value of α , where the colors blue, red, black and purple are related plots for $\mathcal{H}^2 = 0.5, 1, 2, 4$ respectively.



Table 31: Approximated solutions for $\beta = 1.9$, $\mu = 0.9$ and $\mathcal{H}^2 = 1$.

τ	$\alpha = 0.5$		$\alpha = 1$	
	LPSM $N = 8$	RKM	LPSM $N = 8$	RKM
0.0	0.38124417	0.38123631	0.31765381	0.317659270
0.2	0.35993504	0.35996847	0.29780448	0.297816605
0.4	0.31576929	0.31572398	0.26541927	0.265485319
0.6	0.24453042	0.24454677	0.21540035	0.215406643
0.8	0.14105615	0.14105351	0.12834168	0.128333833

Conclusion

In this paper Lucas polynomials based collocation method are applied for solving the fractional order nonlinear differential equations arising in electrohydrodynamics flow of a fluid in a circular cylindrical conduit. In order to having sparse operational matrix, the roots of Lucas polynomials are choosen as collocation points. On the other hand the Caputo fractional derivative of Lucas polynomials are calculated simply because of their definition. The numerical results of the purposed method show that the presented approach is computationally easy and attractive and its results are in good agreement with the results of some other existing methods.

References

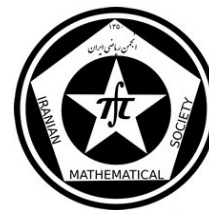
- [1] A. Akgul, D. Baleanu, F. Tchier, On the solutions of electrohydrodynamic flow with fractional differential equations by reproducing kernel method, *Open Physics*, **1**(1): 685-689, 2016.
- [2] M. Cetin, M. Sezer, C. Guler, Lucas polynomial approach for system of high-order linear differential equations and residual error estimation, *Mathematical Problems in Engineering*, 2015, Article ID 625984, 14 pages.
- [3] M.W. Elhameed, Y.H. Youssri, Spectral solutions for fractional differential equations via a novel Lucas operational matrix of fractional derivatives. *Rom. J. Phys.* **61**(56): 795813, 2016.
- [4] A.C. Eringen, G.A. Maugin, *Electrodynamics of continua*. Springer-Verlag; 1990.
- [5] G.E. Bergum, E. Vernrer, J.R. Hoggatt, Irreducibility of Lucas and generalized Lucas polynomials, *Fibonacci Quart.*, **12**: 95100, 1974.
- [6] I. Podlubny, *Fractional differential equations: An introduction to fractional derivatives, fractional differential equations to methods of their solution and some of their applications*, 1st ed. Academic Press, New York, 1998.

Authors E-mails: nosrati@ubonab.ac.ir
Masoomih99@gmail.com



8th Seminar on Numerical Analysis and its Applications

University of Kurdistan, Sanandaj, Iran, April 06-09 2021



Mathematical model of drug release to the posterior segment of the eye after topical drug delivery

Ebrahim Azhdari

Department of Mathematics, Salman Farsi University of Kazerun, Kazerun, Iran, e.azhdari@kazerunsfu.ac.ir

Abstract

In this paper, drug delivery to the posterior segment of the eye and retina is studied using a topical method. Mathematical modeling is performed using partial differential equations with boundary, interface and initial conditions. This model helps to measure the amount of drug that reaches the retina and the posterior segment of the eye from this method. A number of numerical simulations are given to justify this claim.

AMS subject Classification 2010: 65M06.

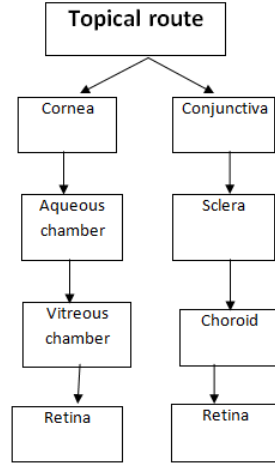
Keywords: Topical drug delivery, Mathematical model, Numerical simulation.

1 Introduction

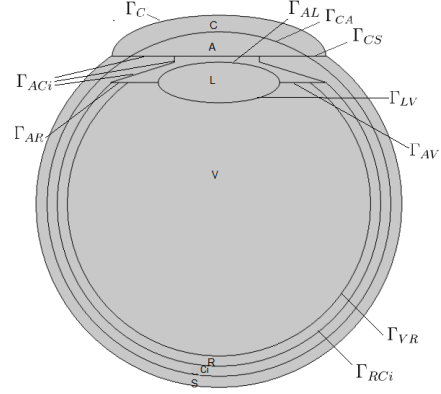
Topical delivery in the form of eye drops is one of the most common methods of medication for the treatment of eye diseases. Due to the anatomy of the eye, there are many barriers to the posterior of the eye to prevent the drug ([1, 2, 6]). After dropping the eye drop into the eye, there are two ways for the drug to pass into the eye: one through the cornea and the other through the non-corneal pathway (see Figure 251.25).

In this paper, I am going to focus on the corneal pathway. After pouring the drop into the eye, the drop enters the cornea, then enters the anterior chamber aqueous humor and from there enters the posterior chamber aqueous humor and then through the hyaloid membranes enters the vitreous chamber and finally enters the retina.

The sections of this article are summarized as follows: In Section 36, the mathematical model is described along with the initial, boundary and interface conditions. In Section 36, numerical simulations are provided with explanations. Finally, discussion and conclusion are presented in Section 36.



1.25 Drug delivery routes to the posterior of eye.



2.25 A schematic of the geometric shape of the problem.

Figure 25

2 Mathematical model

Geometrical of the model

In this paper, a two-dimensional mathematical model of the eye that includes the cornea (C), aqueous cavity (A), lens L , vitreous cavity (V), retina (R), iris-choroid (Co), and sclera (S) is presented (see Figure 252.25). Boundaries and interface boundaries are shown in this figure. However, different geometries have been used for eye anatomy (see [4, 5, 7]).

Cornea route

Cornea: Assuming that an eye drop is poured into the eye, which reaches the cornea after passing through the tear film of the eye. Therefore, the release of the drug into the cornea is described by the equation

$$\frac{\partial c_c}{\partial t} = -\nabla \cdot J_{c_c} - k_c c_c \text{ in } C \times (0, T], \quad (428)$$

where c_c is the drug concentration in the cornea, k_c is a coefficient that takes into account the metabolic consumption of the drug and $J_{c_c} = -D_c \nabla c_c$ is the flux of the drug concentration in the cornea, where D_c is the diffusion coefficient of drug in the cornea.



Aqueous cavity: To simulate drug release in the aqueous cavity, I use the following equation, which includes diffusion, convection, and reaction:

$$\frac{\partial c_a}{\partial t} = -\nabla \cdot J_{c_a} - k_a c_a \text{ in } A \times (0, T], \quad (429)$$

where c_a is the drug concentration in the aqueous cavity, k_a is the clearance of the drug in the aqueous cavity and $J_{c_a} = -D_a \nabla c_a + \mathbf{v}_a c_a$ is the flux of the drug concentration in the aqueous cavity, where D_a is the diffusion coefficient of drug in the aqueous cavity and $\mathbf{v}_a$ is the velocity vector for aqueous humor. The Navier-Stokes equation is used to describe the convective equation due to the motion of the aqueous humor that produces from the ciliary body and leaves the eye through the trabecular meshwork (Eq. (430)).

$$\begin{cases} \rho \frac{\partial \mathbf{v}_a}{\partial t} + \rho(\mathbf{v}_a \cdot \nabla) \mathbf{v}_a + \nabla p_a - \nabla \cdot \mu_a (\nabla \mathbf{v}_a + (\nabla \mathbf{v}_a)^T) = 0, \\ \nabla \cdot \mathbf{v}_a = 0 \text{ in } A \times (0, T], \end{cases} \quad (430)$$

where ρ , μ_a and p_a are density, viscosity and pressure, respectively.

Vitreous: The behavior of the drug concentration in the vitreous is simulated by the following equation

$$\frac{\partial c_v}{\partial t} = -\nabla \cdot J_{c_v} \text{ in } V \times (0, T], \quad (431)$$

where c_v is the drug concentration in the vitreous cavity and $J_{c_v} = -D_v \nabla c_v + \mathbf{v}_v c_v$ is the flux of the drug concentration in the vitreous cavity, where D_v is the diffusion coefficient of drug in the vitreous cavity and $\mathbf{v}_v$ is the convection velocity of the aqueous humor. The Darcy's law is used to describe this velocity as following

$$\begin{cases} \mathbf{v}_v = -\frac{k_v}{\mu_v} \nabla p_v, \\ \nabla \cdot \mathbf{v}_v = 0 \text{ in } V \times (0, T], \end{cases} \quad (432)$$

where k_v , μ_v and p_v are the permeability of the vitreous, viscosity of the aqueous and pressure, respectively.

Retina: Finally, the drug enters the retina after passing through the vitreous, which was the target of the drug delivery, and this phenomenon is described by the following equation:

$$\frac{\partial c_r}{\partial t} = -\nabla \cdot J_{c_r} - k_r c_r \text{ in } R \times (0, T], \quad (433)$$

where c_r is the drug concentration in the retina, k_r is the clearance of the drug in the retina due to the BRB and $J_{c_r} = -D_r \nabla c_r$ is the flux of the drug concentration in the retina, where D_r is the diffusion coefficient of drug in the retina.



The equations (428)-(433) are complemented with initial condition

$$c_c(0) = c_0, c_a(0) = 0, c_v(0) = 0, c_r(0) = 0, p_a(0) = p_0, \mathbf{v}_a(0) = 0, \quad (434)$$

boundary condition

$$\left\{ \begin{array}{l} J_{c_c} \cdot \eta = 0 \text{ on } \Gamma_C \cup \Gamma_{CS} \cup \Gamma_{out}, \\ J_{c_a} \cdot \eta = 0 \text{ on } \Gamma_{ACi} \cup \Gamma_{AR} \cup \Gamma_{in} \cup \Gamma_{AL}, \\ c_a = 0 \text{ on } \Gamma_{out} \cup \Gamma_{AR} \cup \Gamma_{in} \cup \Gamma_{AL}, \\ \mathbf{v}_a = 0 \text{ on } \Gamma_{ACi} \cup \Gamma_{AR} \cup \Gamma_{CA} \cup \Gamma_{AL}, \\ \mathbf{v}_a = v_{in} \text{ on } \Gamma_{in}, \\ p_a = p_{out} \text{ on } \Gamma_{out}, \\ J_{c_v} \cdot \eta = 0 \text{ on } \Gamma_{LV}, \\ \mathbf{v}_v = 0 \text{ on } \Gamma_{LV}, \\ p_v = 2000Pa \text{ on } \Gamma_{AV}, \\ p_v = 1200Pa \text{ on } \Gamma_{VR}, \\ J_{c_r} \cdot \eta = 0 \text{ on } \Gamma_{AR}, \end{array} \right. \quad (435)$$

and interface condition

$$\left\{ \begin{array}{l} J_{c_c} \cdot \eta = A_{ca} c_c \text{ on } \Gamma_{CA}, \\ J_{c_c} \cdot \eta = -J_{c_a} \cdot \nu \text{ on } \Gamma_{CA}, \\ J_{c_a} \cdot \eta = A_{av} c_2 \text{ on } \Gamma_{AV}, \\ J_{c_a} \cdot \eta = -J_{c_v} \cdot \nu \text{ on } \Gamma_{AV}, \\ J_{c_v} \cdot \eta = A_{vr} c_v \text{ on } \Gamma_{VR}, \\ J_{c_v} \cdot \eta = -J_{c_r} \cdot \nu \text{ on } \Gamma_{VR}, \\ J_{c_v} \cdot \eta = A_{rc} c_r \text{ on } \Gamma_{RCi}, \end{array} \right. \quad (436)$$

where A_i , $i = ca, av, vr, rc$, are the mass transfer coefficients and η is the exterior unitary normal to boundaries.

3 Numerical Simulations

In this paper, I have produced the computational meshes by using the commercial tool COMSOL 5.3 Multiphysics.

I use the following modules in the COMSOL:

- Laminar Flow (spf) in order to calculate the velocity and pressure of the aqueous chamber,
- Darcy's Law (dl) in order to calculate the velocity and pressure of the vitreous chamber,
- Coefficient Form PDE (c) in order to calculate the drug concentration.



Backward differential formulae with orders between 1 and 2 with a step length of $0.1s$ is used to integrate in time.

Adequacy of mesh density has been investigated using the convergence test. Considering the two meshes with different sizes, the convergence of the solution and the independence of the mesh have been investigated.

The values and descriptions of the parameters used in this paper are given in Table 32 ([3]).

Table 32: The values used in the model for the parameters.

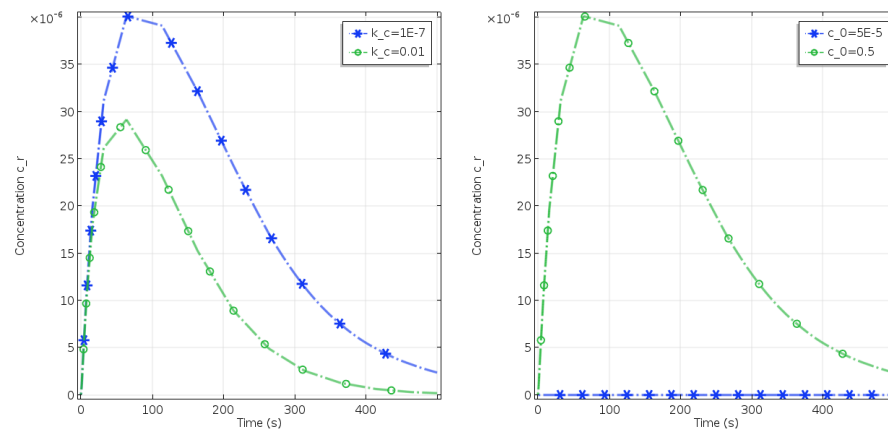
Variable/Parameter	Description	Value
c_0	initial concentration (mol/m^3)	0.5
D_c	diffusion coefficient (m^2/s)	5.74×10^{-11}
k_c	consumption rate ($1/s$)	1.0713×10^{-5}
A_{ca}	mass transfer coefficient (m/s)	1×10^{-5}
D_a	diffusion coefficient (m^2)	5×10^{-5}
k_a	clearance rate ($1/s$)	3×10^{-2}
A_{av}	mass transfer coefficient (m/s)	1×10^{-5}
v_{in}	inlet velocity (m/s)	1.2×10^{-3}
p_{out}	outlet pressure (Pa)	1200
ρ_a	density (kg/m^3)	1000
μ_a	viscosity (Kg/ms)	0.001
k_v	permeability (m^2)	8.4×10^{-14}
D_v	diffusion coefficient (m^2/s)	5.556×10^{-9}
k_r	clearance rate ($1/s$)	4×10^{-5}
D_r	diffusion coefficient (m^2/s)	3.9×10^{-11}
A_{vr}	mass transfer coefficient (m/s)	1×10^{-5}
A_{rc}	mass transfer coefficient (m/s)	1×10^{-5}
p_0	initial pressure (Pa)	1950

Figure 261.26 shows the effect of the parameter k_c on the drug concentration. When the ophthalmic drug reaches the cornea, some of it is used for metabolism, including corneal epithelium, rupture and thinning of tears, rupture of ruptured fluids. These cases are indicated by parameter k_c . It is clear that the larger the value of this parameter, the less drug reaches the target tissue in the retina.

Concentration and subsequent bioavailability of drugs used topically in the ocular position, due to factors such as limited capacity of the ocular space, rapid clearance of the drug by the fluid system of the eyes and lacrimal glands as well as the outflow of the drug due to blinking is reduced. Therefore, by reducing these barriers, that is, by increasing the amount of input drug using Figure 262.26, we see that the amount of drug that reaches the target tissue is a significant amount.

4 Discussion and Conclusion

The most common method of topical medication to the eyes is topical drug delivery. Due to the enclosure of the eyes by membranes and anatomical and physiological barriers, topical drug delivery faces the challenge of not penetrating drugs into different parts of the eye, so that its bioavailability will reach a maximum of 5, and this is the point where pharmaceutical weaknesses on achievements pharmacologically successful and in total ineffective ocular treatments are provided. Figure 26 also illustrate some of these problems.

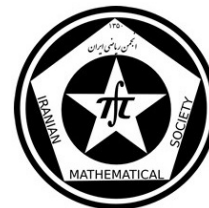


1.26 The influence of k_c on the drug release. 2.26 The influence of initial drug concentration on the drug release.

Figure 26

References

- [1] Eva M. del Amo and et al, *Pharmacokinetic aspects of retinal drug delivery*, Progress in Retinal and Eye Research, (2016), pp. 1-52, DOI: 10.1016/j.preteyeres.2016.12.001.
- [2] Ram K. Balachandran, *Computational modeling of drug transport in the posterior eye*, PhD thesis, University of Minnesota, (2010).
- [3] J.A. Ferreira, P. de Oliveira, P.M. da Silva, J.N. Murta, *Numerical simulation of aqueous humor flow: From healthy to pathologic situations*, Applied Mathematics and Computation, 226 (2014), pp. 777–792, DOI: 10.1016/j.amc.2013.10.070.
- [4] J.A. Ferreira, Paula de Oliveira, P.M. da Silva, R. Silva, *Mathematics of aging: Diseases of the posterior segment of the eye*, Computers and Mathematics with Applications, 73 (2017), pp. 11–26, DOI: 10.1016/j.camwa.2016.10.013.
- [5] J.A. Ferreira, Paula de Oliveira, P.M. da Silva, R. Silva, *Toward a precision ophthalmology : targetting the retina*, Siam J. Appl. Math., 78 (2018), pp. 2996–3023, DOI: 10.1137/17M1163165.
- [6] S. Hu, S. Koevary, *Efficacy of antibody delivery to the retina and optic nerve by topical administration*, Journal of ocular pharmacology and therapeutics, 32, (2016), pp. 203–210, DOI: 10.1089/jop.2015.0111.
- [7] J. Naghipoor, N. Jafary, T. Rabczuk, *Mathematical and computational modeling of drug release from an ocular iontophoretic drug delivery device*, International Journal of Heat and Mass Transfer, 123 (2018), pp. 1035–1049, DOI: 10.1016/j.ijheatmasstransfer.2018.03.021.



Modified Filon-Clenshaw-Curtis rules for regular oscillatory integrals

Hassan Majidian^{†26}

Department of Encyclopedia Research, Institute for Humanities and Cultural Studies, PO Box 14155-6419, Tehran, Iran[†]

h.majidian@ihcs.ac.ir[†]

Abstract

In the implementation of the Filon-Clenshaw-Curtis rules, when the oscillator of the oscillatory integral is nonlinear, its inverse should be evaluated at several interpolation points. In this paper, we suggest an approach based on interpolation, which leads to a class of modifications of the original Filon-Clenshaw-Curtis rules in such a way that the modified rules do not deal with the inverse of the oscillator function. Error estimates of this class are given theoretically, and their accuracy for some special modified rules is illustrated by some numerical experiments.

AMS subject Classification 2010: MSC 65D30, MSC 65T40.

Keywords: Modified Filon-Clenshaw-Curtis rule, oscillatory integral, nonlinear oscillator.

1 Introduction

Consider the oscillatory integrals

$$I_k^{[a,b]}(f, g) := \int_a^b f(x) \exp(kg(x)) x, \quad (437)$$

where $f \in L^1[a, b]$, $k > 0$, and $g \in C^m[a, b]$ for some positive integer m . This type of integrals especially appear in boundary integral methods for direct scattering problem of an acoustic incident plain wave with high frequency (see, e.g., [1, 2]).

Filon-type methods are among most efficient ones for computing (437). Among Filon-type methods, Filon-Clenshaw-Curtis (FCC) rules have many interesting advantages: they can be constructed by a fast and stable algorithm thanks to the FFT; error estimates are available in terms of both the wave number k and the degree of exactness of the rule; moreover, the error estimate reveals that how the Sobolev regularity of the amplitude f is related to the convergence rate. The $(N + 1)$ -point FCC rule for the oscillatory integrals

$$I_k(f) := \int_{-1}^1 f(x) \exp(kx) x, \quad (438)$$

²⁶Speaker



with the linear oscillator can be described as follows:

$$I_{k,N}(f) := \sum_{n=0}^N {}'' \alpha_{n,N}(f) \omega_n(k), \quad (439)$$

where the weights (modified moments) $\omega_n(k) = I_k(T_n)$, $n \geq 0$, can be computed recursively thanks to the three-term recurrence relation for the Chebyshev polynomials T_n . Here, $\sum {}''$ means that the first and the last terms of the sum are to be halved, and the coefficients $\alpha_{n,N}(f)$ can be written as

$$\alpha_{n,N}(f) = \frac{2}{N} \sum_{j=0}^N {}'' \cos(jn\pi/N) f(t_{j,N}), \quad n = 0, \dots, N. \quad (440)$$

For the general oscillatory integral (437) when g has no stationary points in $[a, b]$, by the change of variables $\tau = g(x)$,

$$\begin{aligned} I_k^{[a,b]}(f, g) &:= \int_{g(a)}^{g(b)} F(\tau) \exp(k\tau) \tau \\ &= l \exp(kc) I_{\tilde{k}}(\tilde{F}) \\ &= l \exp(kc) \int_{-1}^1 \tilde{F}(\tau) \exp(\tilde{k}\tau) \tau, \end{aligned} \quad (441)$$

where

$$F(\tau) = ((f/g') \circ g^{-1})(\tau), \quad \tau \in [g(a), g(b)] \quad (442)$$

$$c = \frac{g(b) + g(a)}{2}, \quad l = \frac{g(b) - g(a)}{2}, \quad (443)$$

$$\tilde{k} = lk, \quad \tilde{F}(\tau) = F(c + l\tau). \quad (444)$$

The efficient and robust algorithm of [4] for constructing the FCC rules can now be applied to (441) (see [3]). However, evaluating $g^{-1}(x)$ for some x is required in this process, and this is not of our favorite. For example, it may be impossible to express $g^{-1}(x)$ in a closed form. Then we should resort to Newton iterations, while their convergence is not guaranteed. Even if it converges rapidly, the idea imposes extra error and computational cost.

In this work, we develop a modification of the method in such a way that the need for computing $g^{-1}(x)$ is released.

2 The general idea

Employing the $(N+1)$ -point FCC rule (439) for computing $I_{\tilde{k}}(\tilde{F})$ (that is essential in (441)) necessitates evaluation of $\tilde{F}$ at the Clenshaw-Curtis points. We describe a method for accurate approximation of $\tilde{F}$ at the Clenshaw-Curtis points, in which g^{-1} is not needed to be evaluated at any point.



Let N be a positive integer and consider $N + 1$ Clenshaw-Curtis points $t_{0,N}, \dots, t_{N,N}$. Let $\ell_1 : [-1, 1] \rightarrow [a, b]$ and $\ell_2 : [-1, 1] \rightarrow [g(a), g(b)]$ be the onto affine-like mappings. By definition, $\tilde{F} = F \circ \ell_2$. For a large integer $N' > 1$, choose a set of arbitrary points $-1 = u_0 < \dots < u_{N'} = 1$ and define

$$d_j := \ell_2^{-1}(g(\ell_1(u_j))), \quad j = 0, \dots, N'. \quad (445)$$

Then $-1 = d_0 < \dots < d_{N'} = 1$. One can compute $\tilde{F}(d_j)$ without needing to compute d_j itself and to evaluate the inverse of g at any point. Indeed,

$$\tilde{F}(d_j) = F(\ell_2(d_j)) = \frac{f(\ell_1(u_j))}{g'(\ell_1(u_j))}. \quad (446)$$

Now, we can approximate $\tilde{F}(t_{j,N})$ by interpolation of $\tilde{F}$ at the nodes $d_0, \dots, d_{N'}$. Denote by $\tilde{Q}_{N'}$ the interpolating projection corresponding to some method of interpolation at the points $-1 = d_0 < \dots < d_{N'} = 1$, i.e.,

$$\tilde{Q}_{N'} \tilde{F}(d_j) = \frac{f(\ell_1(u_j))}{g'(\ell_1(u_j))}, \quad j = 0, \dots, N'. \quad (447)$$

Then,

$$\tilde{Q}_{N'} \tilde{F}(t_{j,N}) \approx \tilde{F}(t_{j,N}), \quad j = 0, \dots, N. \quad (448)$$

If $\tilde{k} \geq 1/2$, $k \geq 1/(g(b) - g(a))$, the $(N + 1)$ -point MFCC rule is defined by

$$I_{k,N}^{[a,b]}(f, g) := l \exp(kc) I_{\tilde{k},N}(\tilde{Q}_{N'} \tilde{F}) = l \exp(kc) \sum_{n=0}^N {}'' \alpha_{n,N}(\tilde{Q}_{N'} \tilde{F}) \omega_n(\tilde{k}), \quad (449)$$

where the coefficients are expressed by (440), i.e.,

$$\alpha_{n,N}(\tilde{Q}_{N'} \tilde{F}) = \frac{2}{N} \sum_{j=0}^N {}'' \cos(jn\pi/N) \tilde{Q}_{N'} \tilde{F}(t_{j,N}), \quad n = 0, \dots, N. \quad (450)$$

Indeed, the vector $(\alpha_{n,N}(\tilde{Q}_{N'} \tilde{F}))_{n=0}^N$ is the discrete cosine transform (DCT) of type I of $(\tilde{Q}_{N'} \tilde{F}(t_{j,N}))_{j=0}^N$, which can be computed by the FFT algorithm in $\mathcal{O}(N \ln N)$ flops. Also, the weights $\omega_n(\tilde{k})$ are stably and rapidly computed by the two-phase algorithm [4].

In summary, the MFCC rule for computing (437) is as follows:

- If $\tilde{k} \geq 1/2$,

$$I_k^{[a,b]}(f, g) = l \exp(kc) I_{\tilde{k}}(\tilde{F}) \approx l \exp(kc) I_{\tilde{k},N}(\tilde{F}) \approx l \exp(kc) I_{\tilde{k},N}(\tilde{Q}_{N'} \tilde{F}).$$

- If $\tilde{k} < 1/2$, the integral $I_{\tilde{k}}(\tilde{F})$ and then $I_k^{[a,b]}(f, g)$ is no longer oscillatory, and can be computed by the standard Clenshaw-Curtis rule:

$$I_{k,N}^{[a,b]}(f, g) := I_{0,N}(H_k), \quad H_k(x) := f(x) \exp(kg(x)).$$



3 Error estimates

The following theorem provides an error estimate for the MFCC rules (449):

Theorem 0.69. [5] Assume that f and g are so smooth that $F \in H_w^m[g(a), g(b)]$ for some integer $m \geq 1$. Then, for any $r \in [0, 2]$ with $m > \rho(r)$, the error of the $(N + 1)$ -point MFCC rule with $N \geq m - 1$ is estimated as follows:

$$\begin{aligned} \left| I_k^{[a,b]}(f, g) - I_{k,N}^{[a,b]}(f, g) \right| &\leq Ck^{-r}h^{m+1-r}N^{-m+\rho(r)} \\ &\quad + C'k^{-1}\sqrt{N} \sup_{\tau \in [-1,1]} \left| (\tilde{F} - \tilde{Q}_{N'}\tilde{F})(\tau) \right|, \end{aligned} \quad (451)$$

where $h := b - a$ is the length of the integration interval, and C, C' are constants independent of N and N' .

Corollary 0.70. [5] Assume that f and g are so smooth that $F \in H_w^{N+1}[g(a), g(b)]$ for some integer $N \geq 1$. Then, for any $r \in [0, 2]$, the error of the $(N + 1)$ -point MFCC rule is estimated as follows:

$$\begin{aligned} \left| I_k^{[a,b]}(f, g) - I_{k,N}^{[a,b]}(f, g) \right| &\leq Ck^{-r}h^{N+2-r} \frac{N^{-1+\rho(r)}}{N!2^N} \|g'\|_\infty^N \\ &\quad + C'k^{-1}\sqrt{N} \sup_{\tau \in [-1,1]} \left| (\tilde{F} - \tilde{Q}_{N'}\tilde{F})(\tau) \right|, \end{aligned} \quad (452)$$

where $h := b - a$, C, C' are constants independent of N, N' , and $\|g'\|_\infty = \sup_{x \in [a,b]} |g'(x)|$.

One way to increase accuracy of the MFCC rules is the compound rules developed as follows: Let $[a, b] = [-1, 1]$. Take a large integer M and divide the integration interval $[0, 1]$ into M subintervals of equal lengths h . Assume that $f \in C^{N+1}[0, 1]$ and $g \in C^{N+2}[0, 1]$ for some integer $N \geq 1$. In each panel $[x_{n-1}, x_n]$, we employ the $(N + 1)$ -point MFCC rule (449) with $N' = N$, $u_j = t_{N-j,N}$, and $\tilde{Q}_N$ being the Lagrange interpolating projection at the points $-1 = d_0 < \dots < d_N = 1$. In [5] it is proved that the absolute error is bounded by

$$Ck^{-r}h^{N+1-r} \frac{N^{-1+\rho(r)}}{N!2^N} \|g'\|_\infty^N + C'k^{-1}h^N, \quad (453)$$

where C, C' are independent of M .

4 Numerical experiments

Throughout this subsection, we consider the model integral $I_k^{[0,1]}(f, g)$ with

$$f(x) = \frac{x^{4.5}}{1+x^2}, \quad g(x) = \sqrt{x^2 + 3x + 4}, \quad (454)$$

with some $k > 0$. It can be seen that $g'(x) > 0$, $x \in [0, 1]$, $f \in C^4[0, 1]$ and $g \in C^\infty[0, 1]$. We employ Algorithm I and illustrate the error estimate (453).



Table 33: Application of Algorithm I to the model integral $I_{100}^{[0,1]}(f, g)$ with f and g defined by (454): Absolute error with its decaying rate.

	$N = 1$		$N = 2$		$N = 3$	
M	error	rate	error	rate	error	rate
2	1.87×10^{-4}	—	1.27×10^{-5}	—	2.22×10^{-6}	—
4	2.72×10^{-5}	2.8	3.87×10^{-6}	1.7	4.43×10^{-7}	2.3
8	3.42×10^{-5}	-0.3	8.11×10^{-8}	5.6	3.50×10^{-8}	3.7
16	3.98×10^{-5}	-0.2	1.40×10^{-7}	-0.8	1.41×10^{-9}	4.6
32	3.69×10^{-6}	3.4	2.37×10^{-9}	5.9	1.66×10^{-11}	6.4
64	8.25×10^{-7}	2.2	1.25×10^{-10}	4.2	7.41×10^{-13}	4.5

Experiment 1 Here we study the case when k and N are kept fixed and $h \rightarrow 0$. According to the error bound (453), the convergence of the method is of order $\mathcal{O}(h^N)$, which can be illustrated by the following example. Let $k = 100$. Since $f \in C^4[0, 1]$ and $g \in C^\infty[0, 1]$, $F \in C^4[g(0), g(1)]$. Thus, the error bound (453) is valid if the parameter N does not exceed 3. The absolute error with its decaying rate, as $h \rightarrow 0$, is given in Table 33 for $N = 1, 2, 3$. As established in [5], the distribution of the interpolation nodes in each panel is so that the corresponding Lebesgue constant is rather small. Thus, the convergence rate will be higher than what we expect. For coarser grids (corresponding to larger h), instead, the convergence rate of $\mathcal{O}(h^N)$ is not expected. Considering all these facts together, the numerical results of Table 33 are in agreement with the theory.

Experiment 2 The error bound (453) suggests that for a fixed parameters N and M , the maximum rate of convergence is $\mathcal{O}(k^{-1})$ as $k \rightarrow \infty$. In this experiment, we illustrate this rate by a numerical example. For each $M \in \{4, 8, 16, 32, 64, 128\}$ and $N \in \{1, 2, 3\}$, we apply Algorithm I to the model integral $I_k^{[0,1]}(f, g)$ with (454) when k varies in a wide band from 10 to 10^4 . The absolute error is scaled by k , and the scaled error for each N and M has been plotted as a function of k (Figure 27). As it is seen the error for each M and N does not deteriorate as k increases, and this observation is in agreement with the maximum convergence order $\mathcal{O}(k^{-1})$ suggested by the theory.

References

- [1] A. Asheim and D. Huybrechs, *Extraction of uniformly accurate phase functions across smooth shadow boundaries in high frequency scattering problems*, SIAM J. Appl. Math. **74** (2014), 454–476.
- [2] S. Chandler-Wilde, I. Graham, S. Langdon, and E. Spence, *Numerical-asymptotic boundary integral methods in high-frequency scattering*, Acta Numerica, **21** (2012), 89–305.



- [3] V. Domínguez, I. Graham, and T. Kim, *Filon-Clenshaw-Curtis rules for highly oscillatory integrals with algebraic singularities and stationary points*, SIAM J. Numer. Anal., **51** (2013), 1542–1566.
- [4] V. Domínguez, I. Graham, and V. Smyshlyaev, *Stability and error estimates for Filon-Clenshaw-Curtis rules for highly oscillatory integrals*, IMA J. Numer. Anal., **31** (2011), 1253–1280.
- [5] H. Majidian, *Modified Filon-Clenshaw-Curtis rules for oscillatory integrals with a nonlinear oscillator*, Preprint, (2021).

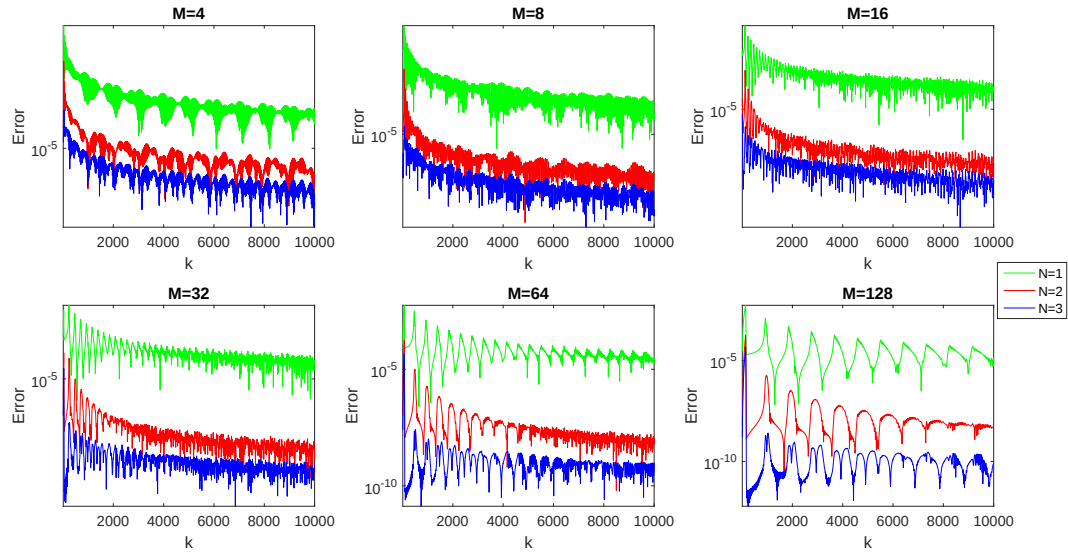
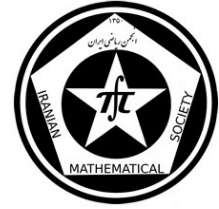


Figure 27: Absolute error scaled by k of Algorithm I with some M and N , when applied to the model integral $I_k^{[0,1]}(f, g)$ with (454).



Multi-scale representation of Galerkin method for solving linear mixed Volterra-Fredholm integral equations

Mehrdad Lakestani[†], Behzad Nemati Saray^{‡27}

Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Tabriz, Tabriz, Iran and
lakestani@tabrizu.ac.ir[‡]

Department of Mathematics, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan 45137-66731, Iran and
bn.saray@iasbs.ac.ir[†]

Abstract

This paper presents an efficient method for solving the linear mixed Volterra-Fredholm equations using multi-scale transformation. For this purpose, by changing the variables, the Fredholm-Volterra equation is discretized using wavelet Galerkin method. The convergence analysis has been investigated. The validity and applicability of the technique are illustrated by a series of numerical tests.

AMS subject Classification 2010: 65R20; 65T6; 34E13.

Keywords: Multi-scale transformation, Multiwavelets, Volterra-Fredholm integral equation.

1 Introduction

In this paper, an efficient numerical method is proposed for solving linear mixed Volterra-Fredholm integral equations using discontinuous wavelet Galerkin method. A linear mixed Volterra-Fredholm equation can be written as

$$u(t, x) - \int_{\alpha}^t \int_{\Omega} k(t, x, \tau, \xi) u(\tau, \xi) d\xi d\tau = f(t, x), \quad (t, x) \in \Omega \times \Omega. \quad (455)$$

We assume that the real value functions $f(x, t)$, $k(x, t, \tau, \xi)$ are continuous functions on $D := \Omega \times \Omega$ and $\mathcal{S} \times \mathbb{R}$ where $\mathcal{S} := \{(t, x, \tau, \xi) : t, \tau \in \Omega, x, \xi \in \Omega\}$ and further, this equation possesses a unique solution on $C(D)$ [3].

There are two reasons for using Alpert's multiwavelets as a tool for solving Volterra-Fredholm integral equations. First, the discretizations of integral operators lead to sparse matrices, and this ability reduces the cost of computing. Second, the scaling functions in this type of multiwavelets have the interpolating property, and this also reduces the computational time and cost. These two abilities are sufficient to use these bases to solve this equation.

²⁷Speaker



2 Alpert's multiwavelets

Let $\Omega := [0, 1] = \bigcup_{b \in \mathcal{B}_j} I_{j,b}$, where $I_{j,b} := [x_{j,b}, x_{j,b+1}]$, $x_{j,b} := 2^{-j}b$ and $\mathcal{B}_j := \{0, 1, \dots, 2^j - 1\}$ for $j \in \mathbb{Z}^+ \cup \{0\}$. In the Alpert's multiwavelets system, the multi-scaling basis are determined by

$$\phi^k(x) := \begin{cases} \sqrt{\frac{2}{\omega_k}} L_k(2x - 1), & x \in [0, 1], \\ 0, & \text{otherwise,} \end{cases} \quad k = 0, 1, \dots, r-1. \quad (456)$$

Here $L_k(x)$ denotes the Lagrange interpolating polynomial that passes through the r points $\{\tau_k, k = 0, 1, \dots, r-1\}$ where τ_k are the roots of Legendre polynomials $P_r(x)$ and $\{\omega_k := 2/(rP'_r(\tau_k)P_{r-1}(\tau_k)), k = 0, 1, \dots, r-1\}$ are the Gauss-Legendre quadrature weights. The function $f \in L^2(\Omega)$ can be written as an expansion of the multi-scaling bases, i.e.,

$$f \approx \mathcal{P}_J^r(f) = \sum_{b \in \mathcal{B}_J} \sum_{k=0}^{r-1} \langle f, \phi_{J,b}^k \rangle_{L^2(\Omega)} \phi_{J,b}^k, \quad J \geq 0,$$

where $\mathcal{P}_J^r$ is an orthogonal projection that maps $L^2(\Omega)$ onto V_J^r . This projection approximates the sufficiently smooth function f with mean error bounded as follows

$$\|\mathcal{P}_J^r(f) - f\| \leq 2^{-Jr} \frac{2}{4^r r!} \sup_{x \in \Omega} |f^{(r)}(x)|,$$

and converge to f as $J \rightarrow \infty$. To calculate the coefficients $f_{J,b}^k := \langle f, \phi_{J,b}^k \rangle_{L^2(\Omega)} = \int_{I_{J,b}} f(x) \phi_{J,b}^k(x) dx$, the interpolation property of Alpert's multi-scaling bases are applied as follows [1]

$$f_{J,b}^k \approx 2^{-J/2} \sqrt{\frac{\omega_k}{2}} f(2^{-J}(\hat{\tau}_k + b)), \quad b \in \mathcal{B}_J, \text{ and } k = 0, \dots, r-1, \quad (457)$$

where $\hat{\tau}_k := (\tau_k + 1)/2$. For a positive fixed number r and $J \geq 0$, suppose that $\Phi_{J,r,b} := [\phi_{J,b}^0, \dots, \phi_{J,b}^{r-1}]^T$ and $\Phi_J^r := [\Phi_{J,r,0}^T, \Phi_{J,r,1}^T, \dots, \Phi_{J,r,2^J-1}^T]^T$. Thus we can write $f \approx \mathcal{P}_J^r(f) = F_J^T \Phi_J^r$ where $F_J \in \mathbb{R}^N$ with $N := r2^J$.

Now, we introduce the multi-scale projection operator $\mathcal{M}_J^r$ that maps $L^2(\Omega)$ onto multiscaling spaces V_J^r via

$$f \approx \mathcal{M}_J^r(f) = (\mathcal{P}_0^r + \sum_{j=0}^{J-1} \mathcal{Q}_j^r)(f) = \sum_{k=0}^{r-1} f_{0,0}^k \phi_{0,0}^k + \sum_{j=0}^{J-1} \sum_{b \in \mathcal{B}_j} \sum_{k=0}^{r-1} \tilde{f}_{j,b}^k \psi_{j,b}^k,$$

where $f_{0,0}^k := \langle f, \phi_{0,0}^k \rangle$, $\tilde{f}_{j,b}^k := \langle f, \psi_{j,b}^k \rangle$ and $\mathcal{Q}_j^r$ is the orthonormal projection operator that maps $L^2(\Omega)$ onto multiwavelet spaces W_j^r . To avoid evaluating directly $\tilde{f}_{j,b}^k$, we apply the wavelet transform matrix T_J , introduced in [4]. This matrix helps us to find the multiwavelets by using the scaling functions $\Psi_J^r = T_J \Phi_J^r$, where $\Psi_J^r := [\Psi_{r,0,b}, \Psi_{r,0,b}, \Psi_{r,1,b}, \dots, \Psi_{r,J-1,b}]^T$ and $\Psi_{r,j,b} := [\psi_{j,b}^0, \dots, \psi_{j,b}^{r-1}]$, $b \in \mathcal{B}_j$.

Now we introduce a new space $V_J^{r,2} := V_J^r \times V_J^r$. This space is spanned by the orthonormal bases $\Phi_{r,J}^2 = \Phi_J^r \otimes \Phi_J^r$ where $\otimes$ is the kronecker product. For a given function $f \in L^2(D)$, we define the



orthonormal projection operator $\mathcal{P}_J^{r,2}$ that maps $L^2(D)$ onto $V_J^{r,2}$ via

$$\begin{aligned} f(x, t) &\approx \mathcal{P}_J^{r,2}(f)(x, t) = \sum_{b \in \mathcal{B}_J} \sum_{k=0}^{r-1} \sum_{b' \in \mathcal{B}_{J'}} \sum_{k'=0}^{r-1} F_{rb+(k+1), rb'+(k'+1)} \phi_{J,b}^k(x) \phi_{J,b'}^{k'} \\ &= \sum_{i=1}^n \sum_{j=1}^n F_{i,j} [\Phi_J^r]_i(x) [\Phi_J^r]_j(t) = \Phi_J^{r,T}(x) F \Phi_J^r(t), \end{aligned}$$

where $[\Phi_J^r]_i$ for $i = 1, \dots, n$ is the i th element of the vector function Φ_J^r , and $i := rb + (k + 1)$, $j := rb' + (k' + 1)$ for $k, k' = 0, \dots, r - 1$ and $b, b' \in \mathcal{B}$ also F is an $(n \times n)$ matrix with elements

$$F_{i,j} \approx 2^{-J} \sqrt{\frac{\omega_k}{2}} \sqrt{\frac{\omega_{k'}}{2}} f(2^{-J}(\hat{\tau}_k + b), 2^{-J}(\hat{\tau}_{k'} + b')) . \quad (458)$$

The orthonormal projection $\mathcal{P}_J^{r,2}(f)$ is convergent to f in $L^2(D)$ as $J \rightarrow \infty$. We may be find the boundary for error, as established by the following theorem.

3 Volterra-Fredholm integral equations

Assume that equation (455) possesses a unique solution on $C(D)$ [3]. By change of variables, we can rewrite this equation as

$$u(t, x) - \int_{\Omega} \int_0^t k(t, x, \tau, \xi) u(\tau, \xi) d\tau d\xi = f(t, x), \quad (t, x) \in D. \quad (459)$$

The operator form of this equation may be written $(I - \mathcal{K})u = f$, where the Volterra-Fredholm operator

$$\mathcal{K}(u)(t, x) := \int_{\Omega} \int_0^t k(t, x, \tau, \xi) u(\tau, \xi) d\tau d\xi. \quad (460)$$

is a compact operator on $L^2(D)$ [2]. To derive an approximate solution of Volterra-Fredholm equation (459), we assume that this solution can be written as an expansion of the Alpert's multiwavelets,

$$u(t, x) \approx \mathcal{M}_J^{r,2}(u)(t, x) = u_J^r(t, x), \quad (t, x) \in D. \quad (461)$$

Also, we can find the same expansion for functions f and $k(\cdot, \tau, \xi)$. Notice that $\|\mathcal{M}_J^{r,2}\| = 1$ and $u \in V_J^r$, $\mathcal{M}_J^r(u) = 0$ if and only if $\langle u, [\Psi_J^r]_i \rangle = 0$ for any $i = 1, \dots, n$. Now we introduce the residual as

$$r_J^r = (I - \mathcal{K}_J^r)u_J^r - f_J^r, \quad (462)$$

where $f_J^r := \mathcal{M}_J^r(f)$ and $\mathcal{K}_J^r := \mathcal{M}_J^r \mathcal{K}$. To find the unknown coefficients, we have the linear system

$$\langle (I - \mathcal{K}_J^r)u_J^r - f_J^r, \Gamma_{i,j} \rangle = 0, \quad i, j = 1, \dots, n, \quad (463)$$

where $\Gamma_{i,j} := [\Psi_J^r]_i [\Psi_J^r]_j$ and inner products can be obtained by (??) and wavelet transform matrix T_J . This is the multiwavelets Galerkin method for the approximate solution of (455). Assume that

$$\mathcal{M}_J^r(k)(x, t, \tau, \xi) := \Psi_J^r(x)^T \tilde{K}(x, t) \Psi_J^r(t), \quad (x, t) \in D,$$



where $\tilde{K}(x, t)$ is an $n \times n$ matrix whose entries obtained by (??) and wavelet transform matrix T_J . Substitute this into (460)

$$\begin{aligned}\mathcal{K}(u)(x, t) &:= \int_{\Omega} \int_0^t k(x, t, \tau, \xi) u(\tau, \xi) d\tau d\xi = \int_{\Omega} \Psi_J^r(\xi) U \left(\int_0^t \Psi_J^r(\tau)^T \Psi_J^r(\tau) d\tau \right) \tilde{K}(x, t) \Psi_J^r(\xi) d\xi \\ &= \int_{\Omega} \Psi_J^r(\xi) U \tilde{\Psi}(t) \tilde{K}(x, t) \Psi_J^r(\xi) d\xi.\end{aligned}$$

Because these multiwavelets are the orthonormal bases, we can write

$$\mathcal{K}(u)(x, t) := \text{tr}(U^T \tilde{\Psi}(t) \tilde{K}(x, t)) = \Psi_J^r(x)^T \hat{K} \Psi_J^r(t),$$

where $\text{tr}(\cdot) := \text{trace}(\cdot)$ and $\hat{K}$ is an $n \times n$ matrix. Also using the matrix to vector conversion, we can replace $\hat{K} := \bar{K} \tilde{U}$ and $F := \tilde{F}$ with $\bar{K} \in \mathbb{R}^{n^2 \times n^2}$ and $\tilde{F}, \tilde{U} \in \mathbb{R}^{n^2}$. Therefore we have

$$(I - \bar{K}) \tilde{U} = \tilde{F}. \quad (464)$$

The linear system (464) must be solved to find the unknowns coefficients U . Since the details coefficients decay at the rate of $2^{-JN_{\psi}^k}$ thus by the higher vanishing moments (N_{ψ}^k) and increasing refinement level J the more details coefficients may be discarded in the regions where the function is smooth. This gives rise to thresholding. By thresholding, the number of coefficients in the linear system (464) can be reduced while maintaining the specified accuracy ϵ .

In the abstract form, we have

$$\mathcal{M}_J^r(I - \mathcal{K}_J^r) u_J^r = \mathcal{M}_J^r f_J^r, \quad u_J^r \in V_J^{r,2}. \quad (465)$$

Due to $\mathcal{M}_J^r u_J^r = u_J^r$, if u_J^r is a solution of (465), equation (465) can be written as

$$(I - \mathcal{M}_J^r \mathcal{K}_J^r) u_J^r = \mathcal{M}_J^r f_J^r, \quad u_J^r \in V_J^{r,2}. \quad (466)$$

It is easy to show that the solution of this equation is also a solution of (465). Also we can able to show that $\|\mathcal{K}_J^r - \mathcal{M}_J^r \mathcal{K}_J^r\| \rightarrow 0$ when $J \rightarrow \infty$ because $\mathcal{K}_J^r$ is a compact operator.

Theorem 0.71. *Let $\mathcal{K} : L^2(D) \rightarrow L^2(D)$ be a compact operator and let $I - \mathcal{K}$ be injective. Assume that the sequence $\mathcal{K}_J^r : L^2(D) \rightarrow L^2(D)$ is collectively compact and pointwise convergent to $\mathcal{K}$. Further assume*

$$\|\mathcal{K}_J^r - \mathcal{M}_J^r \mathcal{K}_J^r\| \rightarrow 0, \quad \text{as } J \rightarrow \infty. \quad (467)$$

Then $(I - \mathcal{K}_J^r)^{-1}$ exist and uniformly bounded. Also the solution of (??) satisfies the error estimate

$$\|u - u_J^r\| \leq \frac{\|(I - \mathcal{K}_J^r)^{-1}\|}{1 - \epsilon_{J_0} \|(I - \mathcal{K}_J^r)^{-1}\|} (\|\mathcal{M}_J^r f - \mathcal{M}_J^r f_J^r\| + \|u - \mathcal{M}_J^r u\| + \|(\mathcal{M}_J^r \mathcal{K} - \mathcal{M}_J^r \mathcal{K}_J^r) u\|). \quad (468)$$

4 Numerical illustration

Consider the linear mixed Volterra-Fredholm integral equation (455) with

$$\begin{aligned}k(t, x, \tau, \xi) &= -\cos(t - \tau) e^{-(x - \xi)}, \\ f(t, x) &= \frac{1}{4} e^{-x} (2x \cos(t) - x \sin(t - 2) + x \sin(t) + 4 \cos(t)).\end{aligned}$$



Table 34: Sparsity and L_2 error with different values of r , J and σ .

r	J	$\sigma = 0$		$\sigma = 10^{-5}$		$\sigma = 10^{-3}$	
		S_σ	L_2	S_σ	L_2	S_σ	L_2
3	2	0	$3.22e-5$	94.04%	$3.26e-5$	97.15%	$9.34e-4$
	3	0	$4.04e-6$	99.14%	$7.32e-6$	99.58%	$9.37e-4$
4	2	0	$6.14e-7$	96.34%	$1.38e-5$	98.22%	$3.37e-4$
	3	0	$3.84e-8$	99.45%	$1.62e-5$	99.74%	$3.37e-4$

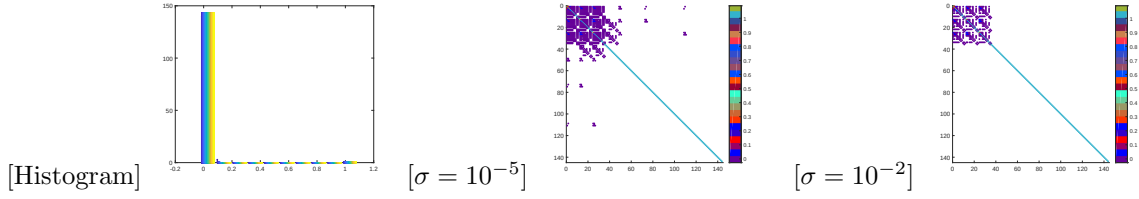


Figure 28: The effect of thresholding on the coefficients matrix.

Its exact solution is $u(t, x) = e^{-x} \cos(t)$.

In Table 54, we report the sparsity and L_2 error obtained taking different values of r , J and σ . We can find the effect of thresholding on the coefficients matrix $I - \bar{K}$ in Fig 40. Also sub-figures b and c of Fig 40, show the effect of decreasing σ on the coefficients matrix $I - \bar{K}$. Fig 41 is reported to illustrate the effect of multiresolution analysis depend on J and parameter r on the error. It is obvious that the error decreases when J and r increase.

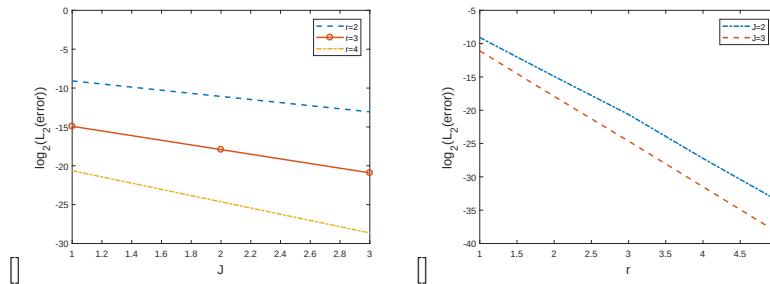
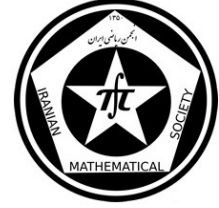


Figure 29: The effect of parameters J (a) and r (b) on the L_2 error.



References

- [1] Alpert B, Beylkin G, Gines D, Vozovoi L. *Adaptive solution of partial differential equations in multiwavelet bases*. J. Comput. Phys., **182** (2002), 149-190.
- [2] Atkinson KE. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press. 1997.
- [3] Brunner H. *On the numerical solution of nonlinear Volterra-Fredholm integral equations by collocation methods*. SIAM J. Numer. Anal. **27(4)** (1990), 987-1000.
- [4] B. N. Saray, M. Lakestani and M. Razzaghi, *Sparse Representation of System of Fredholm Integro-Differential Equations by using Alpert Multiwavelets*, Comput. Math. Math. Phys., **55(9)** (2015), 1468-1483.



Numerical study of 2D variable-order time fractional mobile/immobile advection-diffusion model using a truly meshless method

Ali Shokri[†], Erfan Bahmani^{‡28}

a.shokri@znu.ac.ir[†]

erfan.bahmani@znu.ac.ir[‡]

Abstract

In this paper, we propose a direct meshless local Petrov-Galerkin (DMLPG) method for solving the 2D variable-order time fractional mobile/immobile advection-diffusion equation. This method is based on a generalized moving least square (GMLS) and a local weak form of this equation. Also, to show the high capability and accuracy of the DMLPG method, a numerical example is provided in the last part of this article

AMS subject Classification 2010: 34K37,65M99,35K20.

Keywords: DMLPG method; Local weak form; GMLS approximation.

1 Introduction

In the past two decades, fractional calculus has demonstrated its capability in modeling unusual behaviors perceived in different kinds of engineering and science where the standard mathematical models of integer-order derivatives do not work adequately. Therefore, applications and theory of fractional calculus quickly expanded. It is very difficult or impossible to find the analytical solution for fractional differential equations (FDEs). So their computations have been the focus of interest for many researchers [1].

The 2D variable-order time fractional mobile/immobile advection-diffusion [2] is

$$u_t(\mathbf{x}, t) + \lambda D_t^\alpha u(\mathbf{x}, t) = V_1 \cdot \nabla^2(\mathbf{x}, t) - V_2 \cdot \nabla(\mathbf{x}, t) + f(\mathbf{x}, t), \quad (\mathbf{x}, t) \in \Omega \times (0, T], \quad (469)$$

where λ , V_1 and V_2 are constant. $D_t^\alpha u(\mathbf{x}, t)$ is the fractional derivative of order α ($0 < \alpha < 1$) with respect to t in the Caputo type as follows

$$D_t^\alpha u(\mathbf{x}, t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t (t-\tau)^{-\alpha} \frac{\partial u(\mathbf{x}, \tau)}{\partial \tau} d\tau. \quad (470)$$

²⁸Erfan Bahmani



The initial and boundary conditions are

$$u(\mathbf{x}, 0) = h(x, t) \quad \mathbf{x} = (x, y) \in \Omega \subset \mathbb{R}^2, \quad (471)$$

$$u(\mathbf{x}, t) = g(x, t) \quad \mathbf{x} \in \partial\Omega, \quad t \in [0, T]. \quad (472)$$

Many numerical methods have been used for solving fractional mobile/immobile equation like the MLPG method[2]. The authors in [3] used the discrete Hahn polynomials for the numerical solution of this equation. There are many meshless techniques based on the MLS approximation for the numerical solution of differential equations. The Meshless Local Petrov–Galerkin (MLPG) method is one of the popular meshless methods that has been used very successfully to solve several types of boundary value problems since the late nineties (see [4] and references therein).

The direct MLPG (DMLPG) technique that uses a generalized moving least squares (GMLS) approximation, was first introduced by Mirzaei and Schaback [4]. In the following, we recall the GMLS approximation[5].

2 GMLS approximation

Let $u \in \mathbb{C}^m(\Omega)$, for some $m \geq 0$, and let $\{\lambda_j(u)\}_{j=1}^N$ be a set of continuous linear functionals from the dual space $\mathbb{C}^m(\Omega)^*$ of $\mathbb{C}^m(\Omega)$. For a fixed functional $\lambda \in \mathbb{C}^m(\Omega)^*$, the GMLS method approximates the value of $\lambda(u)$ from the values $\{\lambda_j(u)\}_{j=1}^N$. The approximation $\widehat{\lambda(u)}$ of $\lambda(u)$ is a linear function of $\lambda_j(u)$ as follows

$$\widehat{\lambda(u)} = \sum_{j=1}^N a_j(\lambda) \lambda_j(u), \quad (473)$$

and the coefficients a_j should be linear in λ . As in classical MLS, we assume that the approximation equation (473) to be exact for a finite-dimensional subspace $\mathcal{P} = \text{span}\{p_1, p_2, \dots, p_Q\}$, in which $\mathcal{P}$ is the space of d -variate polynomials of degree at most m .

The GMLS approximation $\widehat{\lambda(u)}$ of $\lambda(u)$ can also be obtained as $\widehat{\lambda(u)} = \lambda(p^*)$, where $p^* \in \mathcal{P}$ is the minimizer of the weighted least-squares error functional

$$\sum_{j=1}^N (\lambda_j(u) - \lambda_j(p))^2 \omega_j, \quad (474)$$

among all $p \in \mathcal{P}$ and ω_j are given non-negative weights. Even if a different numerical method is used to minimize (474), the optimal solution $a^*(\lambda) \in \mathbb{R}^N$ can be written as

$$a^*(\lambda) = W P^T (P W P^T)^{-1} \lambda(\mathbf{p}), \quad (475)$$

where W is the diagonal matrix carrying the weights ω_j on its diagonal, P is the $N \times Q$ matrix of values $\lambda_j(p_k)$, $1 \leq j \leq N$, $1 \leq k \leq Q$, and $\lambda(\mathbf{p}) \in \mathbb{R}^Q$ is the vector with values $\lambda(p_1), \dots, \lambda(p_Q)$ of λ on the basis of $\mathcal{P}$. It should be noted that the weight function ω in GMLS depends on functional λ and since all our functionals are finally considered as point evaluation functionals at point x , we can choose the same $\omega(x)$ for all. Moreover, a small domain Ω_j containing x_j is associated with node j



such that $\omega(x, x_j)$ equals zero outside Ω_j . In this paper, the Gaussian weight function is used for all computations, which is

$$\omega(x, x_j) = \begin{cases} \frac{\exp(-(\|x - x_j\|_2 / c)^2) - \exp(-(\delta/c)^2)}{1 - \exp(-(\delta/c)^2)}, & 0 \leq \|x - x_j\|_2 \leq \delta, \\ 0, & \text{elsewhere} \end{cases} \quad (476)$$

where c is a constant controlling the shape of the weight function and δ is the size of the support domains.

3 Weak form and DMLPG formulation

In meshless methods, everything writes entirely in terms of scattered nodes as $X = \{x_1, x_2, \dots, x_N\}$ located in the spatial domain Ω and its boundary Γ . In every type of DMLPG methods, a small sub-domain $\Omega_s \subset \Omega \cup \Gamma$ is chosen around each node such that integrations over Ω_s are comparatively cheap. On these sub-domains, the V-O time fractional mobile/immobile advection-diffusion equation (469) including boundary condition are stated in the following weak form

$$\int_{\Omega_s} (u_t + \lambda D_t^\alpha u - V_1 \cdot \nabla^2 u + V_2 \cdot \nabla u - f) \nu d\Omega = 0, \quad (477)$$

where ν is an appropriate test function and $F(u) = u(1 - u)$. Applying integration by parts, we get

$$\frac{\partial}{\partial t} \int_{\Omega_s} u \nu d\Omega + \lambda D_t^\alpha \int_{\Omega_s} u \nu d\Omega = \int_{\Omega_s} V_1 \cdot \nabla^2 u \nu d\Omega - \int_{\Omega_s} V_2 \cdot \nabla u \nu d\Omega + \int_{\Omega_s} f \nu d\Omega, \quad (478)$$

The DMLPG method is based on the local weak form (478). All integrals in (478) can be approximated by GMLS method as

$$\lambda_{1,k}(u) := \int_{\Omega_s} u \nu d\Omega \approx \widehat{\lambda_{1,k}(u)} = \sum_{j=1}^N a_{1,j}(x_k) u(x_j), \quad (479)$$

$$\lambda_{2,k}(u) := \int_{\Omega_s} V_1 \cdot \nabla^2 u \nu d\Omega \approx \widehat{\lambda_{2,k}(u)} = \sum_{j=1}^N a_{2,j}(x_k) u(x_j), \quad (480)$$

$$\lambda_{3,k}(u) := \int_{\Omega_s} V_2 \cdot \nabla u \nu d\Omega \approx \widehat{\lambda_{3,k}(u)} = \sum_{j=1}^N a_{3,j}(x_k) u(x_j). \quad (481)$$

To approximate the fractional time derivative of (469), we use the following finite difference technique

$$D_t^\alpha u(\mathbf{x}, t^{(n+1)}) = a(\mathbf{x}, t^{(n+1)}) \left[u^{(n+1)} - u^{(n)} + \sum_{k=1}^n b_k(\mathbf{x}, t^{(n+1)}) (u^{(n-k+1)} - u^{(n-k)}) \right], \quad (482)$$

where $a(\mathbf{x}, t^{(n+1)}) = \frac{dt^{-\alpha(\mathbf{x}, t^{(n+1)})}}{\Gamma(2-\alpha(\mathbf{x}, t^{(n+1)}))}$, $b_k(\mathbf{x}, t^{(n+1)}) = (k+1)^{1-\alpha(\mathbf{x}, t^{(n+1)})} - (k)^{1-\alpha(\mathbf{x}, t^{(n+1)})}$ and

$$u_t = \frac{u^{(n+1)} - u^{(n)}}{dt}, \quad u = \frac{u^{(n+1)} + u^{(n)}}{2} \quad (483)$$



So, we will have the following equation in matrix form

$$\left(A^{(1)}/dt + \mu A^{(1)} + (A^{(3)} - A^{(2)})/2 \right) u^{(n+1)} = \left(A^{(1)}/dt + \mu A^{(1)} + (A^{(2)} - A^{(3)})/2 \right) u^{(n)} \quad (484)$$

$$-\mu A^{(1)} \sum_{k=1}^n b_k(\mathbf{x}, t^{(n+1)}) (u^{(n-k+1)} - u^{(n-k)}) + F(t), \quad (485)$$

where $\mu = a(\mathbf{x}, t^{(n+1)})$, $F(t)$ is the integration of the right hand sides and $A_{kj}^{(\ell)} = a_{\ell,j}(x_k)$, $\ell = 1, 2, 3$. The k -th row of $A^{(\ell)}$ is

$$a_k^{(\ell)} = WP^T(PWP^T)^{-1}\lambda_{\ell,k}(\mathbf{p}) \quad (486)$$

where

$$\lambda_{1,k}(\mathcal{P}) = \left[\int_{\Omega_s} p_1 \nu d\Omega, \int_{\Omega_s} p_2 \nu d\Omega, \dots, \int_{\Omega_s} p_Q \nu d\Omega \right]^T, \quad (487)$$

$$\lambda_{2,k}(\mathcal{P}) = \left[\int_{\Omega_s} V_1 \cdot \nabla^2 p_1 \nu d\Omega, \int_{\Omega_s} V_1 \cdot \nabla^2 p_2 \nu d\Omega, \dots, \int_{\Omega_s} V_1 \cdot \nabla^2 p_Q \nu d\Omega \right]^T, \quad (488)$$

$$\lambda_{3,k}(\mathcal{P}) = \left[\int_{\Omega_s} V_2 \cdot \nabla p_1 \nu d\Omega, \int_{\Omega_s} V_2 \cdot \nabla p_2 \nu d\Omega, \dots, \int_{\Omega_s} V_2 \cdot \nabla p_Q \nu d\Omega \right]^T. \quad (489)$$

4 Main Results

Consider the 2D V-O time fractional mobile/immobile advection-dispersion equation which $\mu = 1$ and $V_1 = V_2 = \vec{i} + \vec{j}$. The initial and Dirichlet boundary conditions will be obtained by the exact solution $u(x, y, t) = x^2 + y^2 + t^2$ and

$$f = 2t + \frac{2t^{2-\alpha(x,y,t)}}{\Gamma(3-\alpha(x,y,t))} + 2x + 2y - 4.$$

Furthermore, three cases for $\alpha(x, y, t)$ are considered such as 0.5, 0.7 and $\alpha(x, y, t) = 0.8 - 0.1 \cos(xt) \sin(x) - 0.1 \cos(yt) \sin(y)$. To solve this example, we consider uniformly distributed nodes on an irregular domain Ω with 833 uniform nodes (653 internal nodes and 180 boundary nodes) which define as follows

$$\Omega = \{(x, y) \in \mathbb{R}^2 : x = R_c \cos(\theta), y = R_c \sin(\theta)\}$$

where $0 \leq \theta \leq 2\pi$, and

$$R_c = \left(\cos(4\theta) + \sqrt{\frac{18}{5} - \sin^2(4\theta)} \right)^{1/3},$$

where nodal and boundary nodes distribution on Ω are displayed in Figure 35. Table 35 shows the maximum errors obtained by DMLPG1 method for different time steps and different values of $\alpha(x, y, t)$ at $T = 1$. Moreover, the exact and approximate solutions are plotted in Figure 35 with $\alpha = 0.5$ and $\Delta t = 0.001$ at $T = 1$.



Figure 30: nodal and boundary nodes distribution (left) and exact vs. approximate solutions of DMLPG1 in a 833 regular grid for $\alpha = 0.5$, $dt = 0.001$ at $T = 1$ (right).

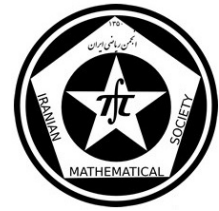
Table 35: Maximum errors on 833 regular nodes at $T = 1$ for DMLPG1

Δt	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha(x, y, t)$
0.01	2.62(-3)	2.99(-3)	8.03(-3)
0.01/2	1.29(-3)	1.45(-3)	9.17(-3)
0.01/4	6.42(-4)	7.13(-4)	9.86(-3)

In this paper, we used the DMLPG methods to numerically solve the 2D variable-order time-fractional mobile/immobile advection-diffusion equation. By observing the results, we can conclude that these methods have the appropriate efficiency and high accuracy of these methods.

References

- [1] M.Ramezania and M.Mojtabaeia and D.Mirzaei, *DMLPG solution of the fractional advection-diffusion problem*, Engineering Analysis with Boundary Elements, **59** (2015), 36–42.
- [2] A. Habibirad and E. Hesameddini and M.H Heydari and R. Roohi, *An efficient meshless method based on the moving Kriging interpolation for two-dimensional variable-order time fractional mobile/immobile advection-diffusion model*, Math Meth Appl Sci.,2020, 1-13.
- [3] F. Salehi and H. Saeedi and M Moghadam, *A Hahn computational operational method for variable order fractional mobile-immobile advection-dispersion equation*, Math Sci. **12** (2018), 91–101.
- [4] D. Mirzaei, R. Schaback, *Direct meshless local Petrov-Galerkin (DMLPG) method: A generalized MLS approximation*, Appl. Numer. Math., **68** (2013) 73–82.
- [5] D. Mirzaei, R. Schaback, M. Dehghan, *On generalized moving least squares and diffuse derivatives*, IMA J. Numer. Anal. **32** (2012) 983–1000.



Numerical Solution of The Three-Asset Black-Scholes Option Pricing Model

Razieh Delpasand^{†29}, Mohammad Mehdi Hosseini[‡]

Department of Applied Mathematics, Faculty of Mathematics, Shahid Bahonar University of Kerman, Kerman, Iran

r.delpasand@yahoo.com[†]

Department of Applied Mathematics, Faculty of Mathematics, Shahid Bahonar University of Kerman, Kerman, Iran

mhosseini@uk.ac.ir[‡]

Abstract

In this paper, a numerical method for solving three-asset Black-Scholes partial differential equation is presented. The model is based on the Crank-Nicolson and the radial basis functions methods. Also, the stability and convergence of the proposed method are proved. Implementation of the proposed method is specially studied on cash-or-nothing option. The numerical results show the efficiency and high accuracy of the proposed method. 91G80, 65M06, 65M12

Three-asset option pricing; Black-Scholes equation; Radial basis functions; Convergency; Stability.

1 Introduction

In the recent decades, financial derivatives have gained a lot of popularity in the market. A financial derivative is a contract with a value dependant on one or several underlying assets. Options are some of the most common derivatives. There are two main types of options: A call option gives its owner the right, but not the obligation, to buy some asset for a price referred to as the exercise price or strike price until a specified time called the expiration time of the option. A put option gives its owner the right, but not the obligation, to sell some asset for an exercise price until the expiration time of the option. Furthermore, options are classified into European and American options. European options can be exercised only on the expiration time, but American options can be exercised any time on or before the expiration time.

Black-Scholes and Merton introduced a parabolic partial differential equation (PDE) that the price of the European option satisfies under certain assumption. During the last decades, researchers have been presenting some numerical methods in order to solve Black-Scholes equation [6, 3, 4]. Pricing

²⁹Speaker



a multi-asset option is a noticable problem. Since finding the exact solution to these problems is difficult due to the high computational cost, many researchers in recent years have proposed numerical methods to solve such problems [2, 2].

In this paper we applied RBFs in order to solve the three-asset Black-Scholes PDE for the cash-or-nothing model. In order to this purpose we use Multiquadratic radial basis functions. The rest of this paper is organized as follows: In section 2, a proposed method based on θ method and RBFs method for solving three-asset Black-Scholes PDE is presented. In addition, stability and convergence of the proposed method are proved in section 3. In section 4, the cash-or-nothing option is introduced. The proposed method is applied to solve this problem and obtained results are presented.

2 Proposed method for the three-asset option pricing

In this paper, we consider the three-asset Black-Scholes PDE;

$$\begin{aligned} & \frac{\partial U}{\partial t}(x, y, z, t) - \frac{1}{2}\sigma_1^2 x^2 \frac{\partial^2 U}{\partial x^2}(x, y, z, t) - \frac{1}{2}\sigma_2^2 y^2 \frac{\partial^2 U}{\partial y^2}(x, y, z, t) - \frac{1}{2}\sigma_3^2 z^2 \frac{\partial^2 U}{\partial z^2}(x, y, z, t) \\ & - \sigma_1 \sigma_2 \rho_{xy} xy \frac{\partial^2 U}{\partial x \partial y}(x, y, z, t) - \sigma_2 \sigma_3 \rho_{yz} yz \frac{\partial^2 U}{\partial y \partial z}(x, y, z, t) - \sigma_3 \sigma_1 \rho_{xz} zx \frac{\partial^2 U}{\partial z \partial x}(x, y, z, t) \\ & - rx \frac{\partial U}{\partial x}(x, y, z, t) - ry \frac{\partial U}{\partial y}(x, y, z, t) - rz \frac{\partial U}{\partial z}(x, y, z, t) + rU(x, y, z, t) = 0. \end{aligned} \quad (490)$$

The domain for each asset price is $[0, +\infty)$, but in a numerical method, we usually truncate the domain to $[0, L_1]$, $[0, L_2]$ and $[0, L_3]$ for S_1, S_2 and S_3 respectively. The choice of L_1, L_2 and L_3 usually depends on the evaluation area we are interested in. We consider (527) with initial condition:

$$U(x, y, z, 0) = \text{payoff}(x, y, z), \quad (491)$$

and boundary conditions:

$$U(0, y, z, t) = \alpha(y, z, t), \quad (492)$$

$$U(L_1, y, z, t) = \beta(y, z, t), \quad (493)$$

$$U(x, 0, z, t) = \gamma(x, z, t), \quad (494)$$

$$U(x, L_2, z, t) = \delta(x, z, t), \quad (495)$$

$$U(x, y, 0, t) = \eta(x, y, t), \quad (496)$$

$$U(x, y, L_3, t) = \omega(x, y, t). \quad (497)$$

where $\alpha(y, z, t)$, $\beta(y, z, t)$, $\gamma(x, z, t)$, $\delta(x, z, t)$, $\eta(x, y, t)$ and $\omega(x, y, t)$ functions are consistent to the exact solution of (527). We discretize the domain with N division in x-axis, y-axis and z-axis, not necessarily uniform as,

$$\{x_i\}_{i=1}^N, \{y_i\}_{i=1}^N \text{ and } \{z_i\}_{i=1}^N, \quad (498)$$

and M time steps, such that interval $[0, T]$ is discretized with $\Delta t = \frac{T}{M}$, such that $t^n = \Delta t \cdot n$, $n = 0, 1, \dots, M$. Now, we approximate function U with RBF method as,

$$U(x, y, z, t) = \sum_{i=1}^{N^3} \lambda_i(t) \phi_i(x, y, z), \quad (499)$$



where $\phi(x, y, z) = [\varphi(r_{1,1,1}), \dots, \varphi(r_{1,1,N}), \varphi(r_{1,2,1}), \dots, \varphi(r_{1,2,N}), \dots, \varphi(r_{N,N,1}), \dots, \varphi(r_{N,N,N})]$, $r_{i,j,k} = \sqrt{(x - x_i)^2 + (y - y_j)^2 + (z - z_k)^2}$ and φ is a radial basis function. By defining the operator

$$D = -\frac{1}{2}\sigma_1^2 x^2 \frac{\partial^2}{\partial x^2} - \frac{1}{2}\sigma_2^2 y^2 \frac{\partial^2}{\partial y^2} - \frac{1}{2}\sigma_3^2 z^2 \frac{\partial^2}{\partial z^2} - \sigma_1 \sigma_2 \rho_{xy} xy \frac{\partial^2}{\partial x \partial y} - \sigma_2 \sigma_3 \rho_{yz} yz \frac{\partial^2}{\partial y \partial z} - \sigma_3 \sigma_1 \rho_{xz} xz \frac{\partial^2}{\partial z \partial x} - rx \frac{\partial}{\partial x} - ry \frac{\partial}{\partial y} - rz \frac{\partial}{\partial z} + r, \quad (500)$$

we can rewrite (527) to:

$$\frac{\partial U}{\partial t}(x, y, z, t) + Du(x, y, z, t) = 0. \quad (501)$$

Using the θ method, we have

$$\left(\frac{U(x, y, z, t + \Delta t) - U(x, y, z, t)}{\Delta t} + O(\Delta t) \right) + (1 - \theta)DU(x, y, z, t + \Delta t) + \theta DU(x, y, z, t) = 0, \quad (502)$$

where the parameter θ is chosen in interval $[0, 1]$. By rearranging (502) we have

$$[1 + (1 - \theta)\Delta t D]U^{n+1} = [1 - \theta\Delta t D]U^n, \quad (503)$$

where $U^n = U(x, y, z, t^n)$. Defining

$$A = 1 + (1 - \theta)\Delta t D \quad \text{and} \quad B = 1 - \theta\Delta t D, \quad (504)$$

we obtain

$$AU^{n+1} = BU^n. \quad (505)$$

By using RBF approximation, we find

$$U^{n+1} = \sum_{i=1}^{N^3} \lambda_i^{n+1} \phi_i(x, y, z), \quad (506)$$

$$U^n = \sum_{i=1}^{N^3} \lambda_i^n \phi_i(x, y, z). \quad (507)$$

Substituting values from (506) and (507) into (529) for all interial and boundary points of collocation points (498), we get the scheme in matrix form:

$$A\Phi\lambda^{n+1} = B\Phi\lambda^n + g^{n+1}, \quad (508)$$

where $\Phi = [\varphi(r_{i,j,k})]_{i,j,k=1}^N$ and g^{n+1} is a $N^3 \times 1$ vector, such that according to interial points its components are equal to zero and its other componets are obtained by substituting boundary points into their boundary conditions. Subsequently (508) can be written as

$$\lambda^{n+1} = (A\Phi)^{-1}(B\Phi)\lambda^n + (A\Phi)^{-1}g^{n+1}. \quad (509)$$

So,

$$\lambda^{n+1} = H\lambda^n + G, \quad (510)$$

where $H = (A\Phi)^{-1}(B\Phi)$ and $G = (A\Phi)^{-1}g^{n+1}$, from (499) and (510) it follows that

$$\mathcal{U}^{n+1} = \Phi H \Phi^{-1} \mathcal{U}^n + \Phi G^{n+1}. \quad (511)$$

In above relation $\mathcal{U}^0$ vector is obtained using initial condition (491).



3 The convergence of the proposed method

In this section, we prove the stability and convergence of the scheme (511). We define matrix $E = \Phi H \Phi^{-1}$. The components of the matrix E depends on the constant $\gamma = \frac{\Delta t}{h^s}$, where h is the distance between any two nodes, and s is the highest order of partial differential operator, where s is equal to 2 for mentioned problem (527). We know that $|U_t(x, y, z) - u_t(x, y, z)| \leq \beta_l h^{l-1} |u|_{\mathcal{N}_\varphi(\Omega)}$, where $l \in \mathbb{N}$, $\mathcal{N}_\varphi(\Omega)$ is a native space of RBF φ and $u^n(x, y, z)$ is the exact solution of (527) at time $n\Delta t$. We assume that (511) is accurate of order p , it yields that

$$\mathbf{u}^{n+1} = \Phi H \Phi^{-1} \mathbf{u}^n + \Phi G^{n+1} + O((\Delta t) + h^p), \quad \Delta t \rightarrow 0, h \rightarrow 0, \forall n \quad (512)$$

Now we define $e^n(x, y, z) = u^n(x, y, z) - U^n(x, y, z)$. By subtracting (511) from (512) we get:

$$\mathbf{e}^{n+1} = E \mathbf{e}^n + O((\Delta t) + h^p), \quad \Delta t \rightarrow 0, h \rightarrow 0 \quad (513)$$

By Lax-Richtmyer definition of stability the scheme in (511) is stable if

$$\|E\| \leq 1, \quad (514)$$

hence, there exist a constant η such that

$$\|\mathbf{e}^{n+1}\| \leq \|E\| \|\mathbf{e}^n\| + \eta((\Delta t) + h^p). \quad (515)$$

It is seen that $e^0 = 0$, using the initial condition. So we have

$$\|\mathbf{e}^{n+1}\| \leq (1 + \|E\|^2 + \dots + \|E\|^{n+1}) \eta((\Delta t) + h^p). \quad (516)$$

By considering the stability condition (514), we obtain

$$\|\mathbf{e}^{n+1}\| \leq n\eta((\Delta t) + h^p). \quad (517)$$

So convergence of the scheme is proved.

4 Implementation of the proposed method

In this section we introduce the cash-or-nothing option. The numerical solutions for the problem are further considered using the proposed method.

Cash-or-nothing option

The cash-or-nothing European call option pays at $t = T$ a fixed value A when at time T that value exceeds or is equal to the strike price E , and nothing if the value of the asset is bellow E . So the payoff function is defined by

$$V(S, T) = \begin{cases} A & x \geq E \\ 0 & x < E \end{cases}, \quad (518)$$



where $A > 0$ is fixed. In this paper we consider the three-asset cash-or-nothing call option. According to (518) the payoff function is defined by

$$V(S_1, S_2, S_3, 0) = \begin{cases} A & S_1 \geq E_1, S_2 \geq E_2, S_3 \geq E_3 \\ 0 & \text{otherwise} \end{cases}, \quad (519)$$

where E_1 , E_2 and E_3 are the strike prices for assets S_1 , S_2 and S_3 , respectively. The closed form solution of (527) is given by [2]:

$$u(S_1, S_2, S_3, t) = \frac{Ae^{-r(T-t)}}{\sigma_1\sigma_2\sigma_3\sqrt{(2\pi(T-t))^3\det\rho}} \int_{E_3}^{\infty} \int_{E_2}^{\infty} \int_{E_1}^{\infty} \frac{e^{-0.5\alpha\rho^{-1}\alpha^T}}{\xi_1\xi_2\xi_3} d\xi_1 d\xi_2 d\xi_3, \quad (520)$$

where

$$\rho = \begin{bmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{12} & 1 & \rho_{23} \\ \rho_{13} & \rho_{23} & 1 \end{bmatrix},$$

and $\alpha = (\alpha_1, \alpha_2, \alpha_3)$, where α_i for $i = 1, 2, 3$ is defined by

$$\alpha_i = \frac{\ln\left(\frac{S_i}{\xi_i}\right) + \left(r - \frac{1}{2}\sigma_i^2\right)(T-t)}{\sigma_i\sqrt{T-t}}.$$

The numerical solutions of cash-or-nothing option using proposed method is presented in the next section.

An example of cash-or-nothing option

In this section, we consider PDE (527) with initial condition (491) and boundary conditions (492)-(497), when

$$\begin{aligned} A &= 10, \sigma_1 = \sigma_2 = \sigma_3 = 0.2, \rho_{12} = \rho_{23} = \rho_{13} = 0.03, \\ T &= 0.5, r = 0.1, L_1 = L_2 = L_3 = 40, E_1 = E_2 = E_3 = 20. \end{aligned}$$

In order to apply the proposed method, we suppose $N = 8, M = 30$ and $\theta = 0.5$. This problem is solved by the proposed method by MQ RBF as, $\varphi(r) = \sqrt{c^2 + r^2}$, and appropriate shape parameter. The results of the proposed method for $N = 8, \Delta t = 0.0125, \theta = 0.5$ are shown in table 52. It can be seen that the proposed method has high accuracy.

It should be noted that the matrix of these problems is usually ill-conditioned, and to solve this problem, appropriate methods are used that are suitable for these cases.

5 Conclusion

The three-asset Black-Scholes PDE, depends on three underlying assets. In this paper, an efficient hybrid numerical method based on the Crank-Nicolson and the RBFs methods was introduced. Furthermore, the stability and convergence of the proposed method were proved. The results show the high accuracy of the proposed method.

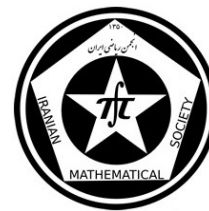


Table 36: Results for cash-or-nothing option example

S_1	S_2	S_3	Approx by proposed method	Exact solution
10	25	22	0.000011274273	0.000011274239
12	16	20	0.001046329783	0.001046329789
16	16	16	0.044237859126	0.044237859123
16	16	20	0.131393694918	0.131393694912
18	24	28	2.721333358812	2.721333358850
24	30	28	9.102486076824	9.102486076800

References

- [1] K. in't Hout, R. Volk, Numerical solution of a two-asset option valuation PDE by ADI finite difference discretization, *AID. Conf. Proc.*, 1648 (2015) 850054.
- [2] J. Kim, T. Kim, J. Jo, Y. Choi, S. Lee, H. Hwang, M. Yoo, D. Jeon, A practical finite difference method for the three-dimensional Black-Scholes equation, *Eur. J. Oper. Res.*, (2015) 1–8. 0
- [3] T. Pellegrino, P. Sabino, On the use of the moment-matching technique for pricing and hedging multi-asset spread options, *Energy. Econ.*, 45 (2014) 172–185.
- [4] V. Shcherbakov. E. Larsson, Radial basis function partition for unity methods for pricing vanilla basket options, *Comput. Math. Appl.*, 71 (1) (2016) 185–200.
- [5] Sh. Zhang, Radial basis functions method for valuing options: A multinomial tree approach, *J. Comput. Appl. Math.*, 319 (2017) 97–107.



Numerical simulation of stochastic population growth model via operational matrix method

Nasrin Samadyar^{† 30}, Yadollah Ordokhani[‡]

Department of Mathematics, Faculty of Mathematical Sciences, Alzahra University, Tehran, Iran and
n.samadyar@alzahra.ac.ir[†]

Department of Mathematics, Faculty of Mathematical Sciences, Alzahra University, Tehran, Iran and
ordokhani@alzahra.ac.ir[‡]

Abstract

Population growth is not constant at all times and it depends on some random environment effects. So, it is better to utilize stochastic differential equations instead of deterministic differential equations for modelling this problem. In this paper, we present an efficient numerical method, which is based on delta operational matrix method, to investigate the behavior of this model. Finally, some numerical experiments are reported to confirm accuracy, efficiency and applicability of presented method.

AMS subject Classification 2010: 60H10, 65L05, 60J65.

Keywords: Delta functions, Operational matrix method, Stochastic differential equations.

1 Introduction

Population growth model is one of the most important mathematical models which investigates the change of population over time. Although the term population growth can be used for any species, it is often used for humans and demonstrates the growth of the population of the world. In recent years, there has been a significant interest in studying population growth model and have been published several articles on this subject. For instance, stochastic logistic population growth models with immigration and multiple births have been studied in [4]. Ramezani et al. [7] applied composite spectral functions to solve Volterra population model. Adomians decomposition approach has been employed to approximate the solution of a fractional population growth model in a closed system [6]. The growth rate has been considered uncertain in [2] and stochastic exponential population growth model has been numerically solved in the referred paper.

This article focuses on the numerical solution of stochastic population growth model which is formulated as follows [1]

$$\begin{cases} dX(t) = X(t)(\lambda - X(t))dt + \sigma X(t)dB(t), \\ X(0) = X_0, \end{cases} \quad (521)$$

³⁰Speaker



where λ, σ, X_0 are known real numbers, $X(t)$ is an unknown stochastic process which should be approximated. Also, $B(t)$ denotes standard brownian motion [3] defined on probability space $(\Omega, \mathcal{F}, \mathcal{P})$. The exact solution of Eq. (521) is as follows [8]

$$X(t) = \frac{X_0 \exp((\lambda - \frac{1}{2}\sigma^2)t + \sigma B(t))}{1 + \int_0^t X_0 \exp((\lambda - \frac{1}{2}\sigma^2)s + \sigma B(s)) ds}.$$

The aim of this paper is approximating the solution of stochastic population growth model, which is one of the most important SDEs in biological science, via operational matrix method based on delta functions.

2 Delta functions

Definition 0.72. A set of $(N + 1)$ delta functions are defined on the interval $[0, T]$ as follows [5]

$$\Delta_i(t) = \begin{cases} \frac{t-(i-1)h}{h}, & (i-1)h \leq t < ih, \quad i \neq 0, \\ 1 - \frac{t-ih}{h}, & ih \leq t < (i+1)h, \quad i \neq N, \\ 0, & \text{otherwise,} \end{cases} \quad (522)$$

where N is a positive integer number, $i = 0, 1, \dots, N$ and $h = \frac{T}{N}$.

Theorem 0.73. An arbitrary function $f(t)$ defined on the closed interval $[0, T]$ can be expanded via delta functions as follows:

$$f(t) \simeq f_N(t) = \sum_{i=0}^N f_i \Delta_i(t) = F^T \Delta(t) = \Delta^T(t) F, \quad (523)$$

where $F = [f_0, f_1, \dots, f_N]^T$, $f_i = f(ih)$ and $\Delta(t) = [\Delta_0(t), \Delta_1(t), \dots, \Delta_N(t)]^T$.

Theorem 0.74. The ordinary and Itô integrals of delta functions vector $\Delta(t)$ can be estimated as follows:

$$\int_0^t \Delta(s) ds \simeq P_o \Delta(t), \quad \int_0^t \Delta(s) dB(s) \simeq P_s \Delta(t), \quad (524)$$

where P_o and P_s are called operational matrix and stochastic operational matrix of integration based on delta functions, respectively, and have been computed in [5].



3 Numerical method

Consider stochastic population growth model introduced in Eq. (521) with initial population $X(0) = X_0$. By integrating from both sides of Eq. (521) on the interval $[0, t]$ and using initial condition, we obtain the following stochastic Itô Volterra integral equation

$$X(t) = X_0 + \int_0^t X(s)(\lambda - X(s))ds + \int_0^t \sigma X(s)dB(s), \quad (525)$$

where X_0, λ, σ are known real numbers, $B(t)$ denotes standard Brownian motion process, and $X(t)$ is an unknown stochastic process which should be approximated.

Let $Z_1(t) = X(t)(\lambda - X(t))$ and $Z_2(t) = X(t)$. By inserting these notations into Eq. (525), we have

$$X(t) = X_0 + \int_0^t Z_1(s)ds + \sigma \int_0^t Z_2(s)dB(s). \quad (526)$$

Therefore, we have

$$\begin{aligned} Z_1(t) &= \left[X_0 + \int_0^t Z_1(s)ds + \sigma \int_0^t Z_2(s)dB(s) \right] \left[\lambda - X_0 - \int_0^t Z_1(s)ds - \sigma \int_0^t Z_2(s)dB(s) \right], \\ Z_2(t) &= X_0 + \int_0^t Z_1(s)ds + \sigma \int_0^t Z_2(s)dB(s). \end{aligned} \quad (527)$$

Now, we approximate the functions $Z_1(t)$ and $Z_2(t)$ in terms of delta functions as follows

$$\begin{aligned} Z_1(t) &\simeq \sum_{i=0}^N z_i^1 \Delta_i(t) = Z_1^T \Delta(t) = \Delta^T(t) Z_1, \\ Z_2(t) &\simeq \sum_{i=0}^N z_i^2 \Delta_i(t) = Z_2^T \Delta(t) = \Delta^T(t) Z_2, \end{aligned} \quad (528)$$

where $Z_1 = [z_0^1, z_1^1, \dots, z_N^1]^T$ and $Z_2 = [z_0^2, z_1^2, \dots, z_N^2]^T$. By substituting the approximations in Eq. (528) into Eq. (527) and using Eq. (524), we have

$$\begin{aligned} Z_1^T \Delta(t) &= \left[X_0 + Z_1^T P_o \Delta(t) + \sigma Z_2^T P_s \Delta(t) \right] \left[\lambda - X_0 - Z_1^T P_o \Delta(t) - \sigma Z_2^T P_s \Delta(t) \right], \\ Z_2^T \Delta(t) &= X_0 + Z_1^T P_o \Delta(t) + \sigma Z_2^T P_s \Delta(t). \end{aligned} \quad (529)$$

If we collocate Eq. (529) at the $(N+1)$ Newton-Cotes nodes, which are defined as follows:

$$t_i = \frac{2i+1}{2(N+1)}, \quad i = 0, 1, \dots, N, \quad (530)$$

we obtain the following nonlinear system of algebraic equations

$$\begin{aligned} Z_1^T \Delta(t_i) &= \left[X_0 + Z_1^T P_o \Delta(t_i) + \sigma Z_2^T P_s \Delta(t_i) \right] \left[\lambda - X_0 - Z_1^T P_o \Delta(t_i) - \sigma Z_2^T P_s \Delta(t_i) \right], \\ Z_2^T \Delta(t_i) &= X_0 + Z_1^T P_o \Delta(t_i) + \sigma Z_2^T P_s \Delta(t_i). \end{aligned} \quad (531)$$

After solving this nonlinear system with Newtons method and determining the unknown vectors Z_1 and Z_2 , we obtain the approximate solution of Eq. (525) as follows

$$X(t) = X_0 + Z_1^T P_o \Delta(t) + \sigma Z_2^T P_s \Delta(t). \quad (532)$$



Table 37: The effect of parameter N on the numerical solution.

t_i	$N = 10$			$N = 20$		
	Exact	Approximate	Error	Exact	Approximate	Error
0.00	0.5000	0.5000	0.0000	0.5000	0.5000	0.0000
0.25	0.6197	0.6578	0.0381	0.6006	0.6039	0.0033
0.50	0.6671	0.6729	0.0058	0.6617	0.6691	0.0074
0.75	0.7738	0.8214	0.0476	0.7787	0.7854	0.0067
1.00	0.8902	0.8976	0.0074	0.8672	0.8736	0.0064

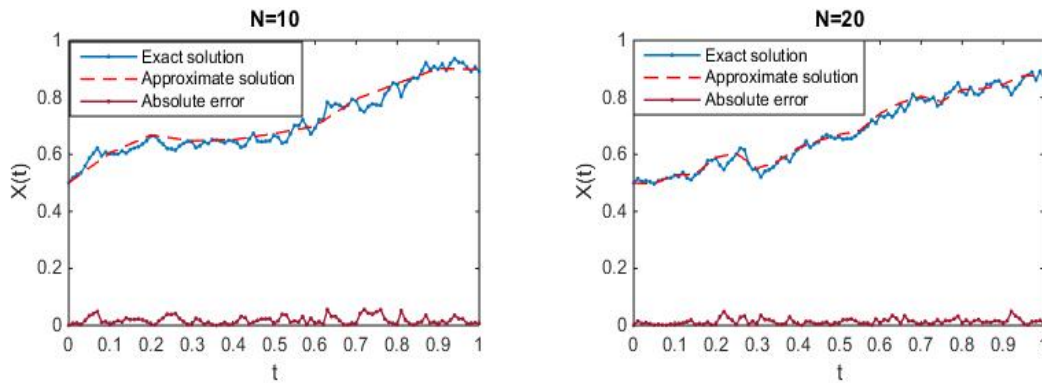


Figure 31: The effect of parameter N on the numerical solution.

4 Numerical experiments

In this section, we consider the numerical simulation of stochastic population growth model by the developed numerical method for different values of X_0 , λ , σ and N . The numerical experiments for $T = 1$, $X_0 = 0.5$, $\lambda = 1$, $\sigma = 0.25$, and two values of $N = 10, 20$ have been reported in Table 54. Also, the figures of exact solution, numerical solution and absolute error have been plotted in Figure 31. Also, the effect of parameters σ on numerical solution have been investigated by considering $T = 1$, $N = 20$, $X_0 = 0.5$, $\lambda = 1$ and obtained results have been reported in Table 39 and Figure 32.

5 Conclusion

In this paper, an efficient numerical scheme has been presented to solve stochastic population growth model, which is one of the most important mathematical models in biological science. To provide



Table 38: The effect of parameter σ on the values of absolute error.

t_i	$\sigma = 0$	$\sigma = 0.125$	$\sigma = 0.5$	$\sigma = 0.75$	$\sigma = 1$
0.25	0.0004	0.0006	0.0179	0.0177	0.0518
0.50	0.0019	0.0004	0.0335	0.0317	0.0766
0.75	0.0045	0.0004	0.0538	0.0335	0.0885
1.00	0.0084	0.0047	0.0538	0.0478	0.0489

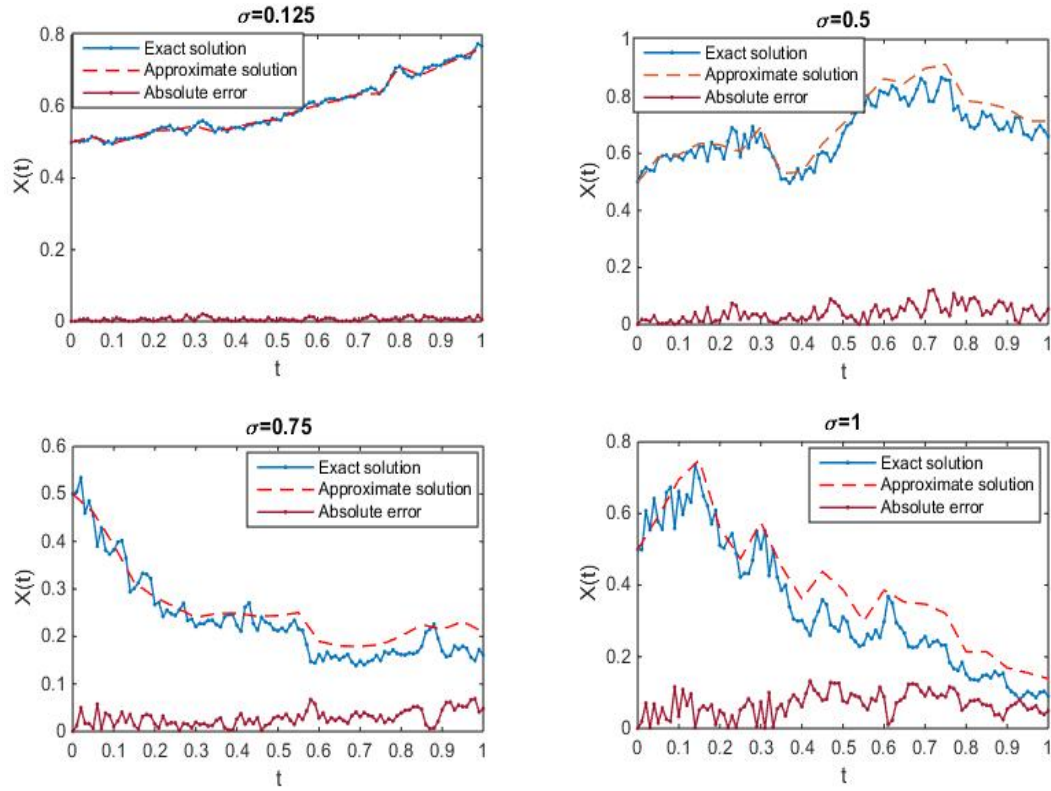


Figure 32: The effect of parameter σ on the numerical solution.



more accurate numerical results, we proceed two different ways:

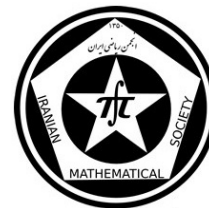
1. increasing the number of used delta functions N ,
2. reduce the amount of parameter σ .

Acknowledgement

This work is supported by the National Elites Foundation and Alzahra University. The authors would like to express our very great appreciation to reviewers for their valuable comments which have improved the quality of our paper.

References

- [1] M. Asgari and F. Hosseini Shekarabi, *Numerical solution of nonlinear stochastic differential equations using the block pulse operational matrices*, Math. Sci., **7** (2013), 1–5.
- [2] M. Khodabin, K. Maleknejad, M. Rostami and M. Nouri, *Interpolation solution in generalized stochastic exponential population growth model*, Appl. Math. Model., **36**(3) (2012), 1023–1033.
- [3] F.C. Klebaner, *Introduction to stochastic calculus with applications*, World Scientific Publishing Company, 2005.
- [4] J.H. Matis and T.R. Kiffe, *On stochastic logistic population growth models with immigration and multiple births*, Theor. Popul. Biol., **65**(1) (2004), 89–104.
- [5] F. Mirzaee and N. Samadyar, *On the numerical solution of stochastic quadratic integral equations via operational matrix method*, Math. Method. Appl. Sci., **41**(12) (2018), 4465–4479.
- [6] S. Momani and R. Qaralleh, *Numerical approximations and Padé approximants for a fractional population growth model*, Appl. Math. Model., **31**(9) (2007), 1907–1914.
- [7] M. Ramezani, M. Razzaghi and M. Dehghan, *Composite spectral functions for solving Volterra population model*, Chaos Solitons Fract., **34** (2007) 588–593.
- [8] Z. Sadati, *Application of new basis functions for solving nonlinear stochastic differential equations*, Int. J. Nonlinear Anal. Appl., **7**(2) (2016), 59–68.



Numerical solution of Bagley-Torvik equation using fractional Chebyshev collocation method

A. Faghhi^{†31}, P. Mokhtary[‡]

Department of Mathematics, Sahand University of Technology, Tabriz, Iran

Abstract

This paper is devoted to introducing a numerical approach based on fractional Chebyshev collocation method for the numerical solution of Bagley-Torvik equation. The validity of the method is demonstrated via an illustrative example.

AMS subject Classification 2010: 34A09, 65L05, 65L20.

Keywords: Bagley-Torvik equation, Fractional Chebyshev functions, Collocation method.

1 Introduction

We intend to propose an appropriate numerical method for solving the following Bagley-Torvik equation which has an important role in the fractional calculus

$$\begin{cases} a_1 y''(x) + a_2 D_C^{\frac{3}{2}} y(x) + a_3 y(x) = q(x), & x \in \Omega = [0, 1], \\ y(0) = y'(0) = 0, \end{cases} \quad (533)$$

where a_1, a_2 are real constants with $a_1 \neq 0$, $q(x)$ are continuous function on Ω , and D_C^p denotes the Caputo fractional derivative operator of order $p \in \mathbb{Q}^+$ given by [1]

$$D_C^p y(x) = I^{[p]-p} \partial_x^{[p]} y(x),$$

where $[p]$ denotes the ceiling function of p and

$$I^p y(x) = \frac{1}{\Gamma(p)} \int_0^x (x-t)^{p-1} y(t) dt,$$

is the fractional integral operator of order p .

The rest of this paper is organized as follows. In the next section, we introduce the fractional Chebyshev functions and design a new fractional interpolation operator. To approximate the solution of (689), an efficient fractional Chebyshev collocation method is developed in Section 3. In Section 4, an illustrative example is given to show the efficiency of the proposed method.

³¹Speaker



2 The fractional Chebyshev functions

The shifted Chebyshev polynomials on Ω denoted by $T_n(t)$ are orthogonal with respect to the shifted Chebyshev weight function $w(t) = t^{-\frac{1}{2}}(1-t)^{-\frac{1}{2}}$ i.e.,

$$\int_{\Omega} T_m(t)T_n(t)w(t)dt = h_n\delta_{mn}, \quad m, n \geq 0, \quad (534)$$

in which $h_n = \|T_n\|_w^2 = \frac{\Gamma^2(n + \frac{1}{2})}{2\Gamma^2(n + 1)}$, and δ_{mn} is the Kronecker delta.

Lemma 0.75. [4] (*Shifted Chebyshev-Gauss quadrature*) Suppose that $\{t_j\}_{j=0}^N$ are the zeros of $T_{N+1}(t)$, and

$$\{w_j\}_{j=0}^N = \frac{K_N}{\left(1 - (t_j)^2\right)\left(\partial_t T_{N+1}(t_j)\right)^2},$$

are the corresponding weights, where $K_N = \frac{\Gamma^2(N + \frac{3}{2})}{(N + 1)!\Gamma(N + 1)}$.

Then we have

$$\int_{\Omega} V(t)w(t)dt = \sum_{j=0}^N V(t_j)w_j, \quad \forall V \in \mathbb{P}_{2N+1}.$$

Here, $\mathbb{P}_{2N+1}$ is the space of all algebraic polynomials with degree at most $2N + 1$.

Further properties of the Chebyshev polynomials can be found in [4].

The fractional Chebyshev functions $T_n^{(\sigma)}(x)$ with $\sigma \in (0, 1]$ and $x \in \Omega$ are defined from the shifted Chebyshev polynomials through the coordinate transform $t = x^\sigma$ as follows [5]

$$T_n^{(\sigma)}(x) = T_n(x^\sigma).$$

In view of (609), it can easily be checked that these functions are mutually orthogonal concerning the weight function $w^{(\sigma)}(x) = \sigma x^{\frac{\sigma}{2}-1}(1-x^\sigma)^{-\frac{1}{2}}$, i.e.,

$$\int_{\Omega} T_m^{(\sigma)}(x)T_n^{(\sigma)}(x)w^{(\sigma)}(x)dx = h_n\delta_{mn}, \quad m, n \geq 0.$$

Utilizing Lemma 0.113, the following result can be achieved.

Lemma 0.76. [5] (*Fractional Chebyshev-Gauss quadrature*) Suppose that $\{t_j, w_j\}_{j=0}^N$ are the shifted Chebyshev-Gauss quadrature nodes and corresponding weights respectively. Considering $x_j^{(\sigma)} = (t_j)^{1/\sigma}$ and $w_j^{(\sigma)} = w_j$, we have

$$\int_{\Omega} v(x)w^{(\sigma)}(x)dx = \sum_{j=0}^N v(x_j^{(\sigma)})w_j^{(\sigma)}, \quad \forall v \in \Psi_{2N+1}^{(\sigma)},$$

where

$$\Psi_{2N+1}^{(\sigma)} = \text{Span}\{T_n^{(\sigma)}(x) : 0 \leq n \leq 2N + 1\}.$$



The fractional Chebyshev interpolation operator $I_N^{(\sigma)} : L_{w^{(\sigma)}}^2(\Omega) \rightarrow \Psi_N^{(\sigma)}$ is introduced as follows

$$(I_N^{(\sigma)} v)(x_j^{(\sigma)}) = v(x_j^{(\sigma)}), \quad 0 \leq j \leq N.$$

Since $I_N^{(\sigma)} v \in \Psi_N^{(\sigma)}$, we can write

$$(I_N^{(\sigma)} v)(x) = \sum_{j=0}^N \tilde{v}_j T_j^{(\sigma)}(x),$$

where the unknown coefficients $\tilde{v}_j$ are obtained from $\tilde{v}_j = \frac{1}{h_j} \langle v, T_j^{(\sigma)} \rangle_{N, w^{(\sigma)}}$, $0 \leq j \leq N$.

Here, $\langle v, u \rangle_{N, w^{(\sigma)}}$ is the fractional Chebyshev-Gauss discrete inner product formula defined by

$$\langle v, u \rangle_{N, w^{(\sigma)}} = \sum_{j=0}^N v(x_j^{(\sigma)}) u(x_j^{(\sigma)}) w_j^{(\sigma)}.$$

Clearly, from Lemma 0.76, we have

$$\langle u, v \rangle_{N, w^{(\sigma)}} = (u, v)_{w^{(\sigma)}}, \quad \forall uv \in \Psi_{2N+1}^{(\sigma)}.$$

3 Fractional Chebyshev collocation approach

From the well-known existence and uniqueness theorems, we can find that the solution of (689) suffer from singularity at origin and behave like $\frac{1}{\sqrt{x}}$ [3].

This drawback may lead to a challenge in obtaining accurate approximate solutions. Trivially, applying the spectral methods based on the infinitely smooth basis functions produces less accurate approximation. In this paper, to fix this difficulty, we design an efficient spectral collocation approach by introducing a new fractional interpolation operator to solve (689) that allows us to obtain an approximation with a satisfactory order of convergence regardless of the existence of a singularity behavior in the solution. For this purpose, we first transform the main problem (689) to the following equivalence system of multi-order fractional differential equations [1]

$$\begin{cases} D_C^{3/2} y_1(x) = y_2(x), \\ D_C^{1/2} y_2(x) = -d_1 y_1(x) - d_2 y_2(x) + d_3 q(x), \\ y_1(0) = y_1'(0) = y_2(0) = 0, \end{cases} \quad (535)$$

where $d_1 = \frac{a_3}{a_1}$, $d_2 = \frac{a_2}{a_1}$, $d_3 = \frac{1}{a_1}$, and $y_1(x) = y(x)$ is the exact solution of (689). We consider the fractional Chebyshev collocation solution of (535) as

$$y_{j,N}(x) = \sum_{i=0}^N c_{ji} T_i^{(\frac{1}{2})}(x) = \underline{c}_j T^{(\frac{1}{2})} = \underline{c}_j T^{(\frac{1}{2})} \underline{X}, \quad \underline{c}_j = [c_{j0}, c_{j1}, \dots, c_{jN}], \quad j = 1, 2, \quad (536)$$



where $\underline{T}^{(\frac{1}{2})} = [T_0^{(\frac{1}{2})}(x), T_1^{(\frac{1}{2})}(x), \dots, T_N^{(\frac{1}{2})}(x)]^T$ is the vector of fractional Chebyshev functions, $T^{(\frac{1}{2})}$ is an invertible lower triangular coefficient matrix and $\underline{X} = [1, x^{\frac{1}{2}}, x, \dots, x^{\frac{N}{2}}]^T$. Considering

$$q(x) = \sum_{i=0}^N q_i x^{\frac{i}{2}} = \underline{q}\underline{X}, \quad \underline{q} = [q_0, q_1, \dots, q_N], \quad (537)$$

and inserting (536) and (537) into the equivalence system of Volterra integral equations of (535), we have

$$\begin{cases} \underline{c}_1 T^{(\frac{1}{2})} \underline{X} = \underline{c}_2 T^{(\frac{1}{2})} I^{\frac{3}{2}} \underline{X}, \\ \underline{c}_2 T^{(\frac{1}{2})} \underline{X} = -d_1 \underline{c}_1 T^{(\frac{1}{2})} I^{\frac{1}{2}} \underline{X} - d_2 \underline{c}_2 T^{(\frac{1}{2})} I^{\frac{1}{2}} \underline{X} + d_3 \underline{q} I^{\frac{1}{2}} \underline{X}. \end{cases} \quad (538)$$

Therefore, Computing $I^{\frac{3}{2}} \underline{X}$ and $I^{\frac{1}{2}} \underline{X}$ the relation (538) can be written as

$$\begin{cases} \underline{c}_1 \underline{X} - \underline{c}_2 \underline{X}_{\frac{3}{2}} = 0, \\ \underline{c}_1 d_1 \underline{X}_{\frac{1}{2}} + \underline{c}_2 (\underline{X} + d_2 \underline{X}_{\frac{1}{2}}) = d_3 \underline{q} \underline{X}_{\frac{1}{2}}, \end{cases} \quad \underline{c}_j = \underline{c}_j T^{(\frac{1}{2})} = [\underline{c}_{j0}, \underline{c}_{j1}, \dots, \underline{c}_{jN}], \quad j = 1, 2, \quad (539)$$

where

$$\begin{aligned} \underline{X}_{\frac{3}{2}} &:= I^{\frac{3}{2}} \underline{X} = \left[I^{\frac{3}{2}} \left(x^{\frac{k}{2}} \right) \right]_{k=0}^N = \left[\frac{\Gamma(\frac{k}{2} + 1)}{\Gamma(\frac{k+5}{2})} x^{\frac{3+k}{2}} \right]_{k=0}^N, \\ \underline{X}_{\frac{1}{2}} &:= I^{\frac{1}{2}} \underline{X} = \left[I^{\frac{1}{2}} \left(x^{\frac{k}{2}} \right) \right]_{k=0}^N = \left[\frac{\Gamma(\frac{k}{2} + 1)}{\Gamma(\frac{k+3}{2})} x^{\frac{k+1}{2}} \right]_{k=0}^N, \end{aligned}$$

are the vectors of order $N + 1$.

The relation (539) has the following matrix formulation

$$[\underline{c}_1, \underline{c}_2] \begin{bmatrix} \underline{X} & d_1 \underline{X}_{\frac{1}{2}} \\ -\underline{X}_{\frac{3}{2}} & \underline{X} + d_2 \underline{X}_{\frac{1}{2}} \end{bmatrix} = [0, \tilde{q}(x)], \quad \tilde{q}(x) = d_3 \underline{q} \underline{X}_{\frac{1}{2}}.$$

In this step, the unknown vector $\underline{c}_j$ is obtained such that the system (539) is satisfied at a set of collocation points. We choose the fractional Chebyshev-Gauss nodes $\{x_r^{(\frac{1}{2})}\}_{r=0}^N$ introduced in Lemma 0.76. Consequently, we have the following system of algebraic equations of order $2(N + 1)$

$$\begin{aligned} [\underline{c}_1, \underline{c}_2] \begin{bmatrix} \underline{X}_{x_0^{(\frac{1}{2})}} & \cdots & \underline{X}_{x_N^{(\frac{1}{2})}} & (d_1 \underline{X}_{\frac{1}{2}})_{x_0^{(\frac{1}{2})}} & \cdots & (d_1 \underline{X}_{\frac{1}{2}})_{x_N^{(\frac{1}{2})}} \\ (-\underline{X}_{\frac{3}{2}})_{x_0^{(\frac{1}{2})}} & \cdots & (-\underline{X}_{\frac{3}{2}})_{x_N^{(\frac{1}{2})}} & (\underline{X} + d_2 \underline{X}_{\frac{1}{2}})_{x_0^{(\frac{1}{2})}} & \cdots & (\underline{X} + d_2 \underline{X}_{\frac{1}{2}})_{x_N^{(\frac{1}{2})}} \end{bmatrix} \\ = \overbrace{[0, \dots, 0, \tilde{q}(x_0^{(\frac{1}{2})}), \dots, \tilde{q}(x_N^{(\frac{1}{2})})]}^{N+1}. \end{aligned} \quad (540)$$

Due to the non-singularity of (540), this system of algebraic equations is solved using the Biconjugate Gradient (BCG) algorithm [4]. Finally, the fractional Chebyshev collocation approximations (536) can be calculated from solving the lower triangular system $\underline{c}_j = \underline{c}_j T^{(\frac{1}{2})}$, $j = 1, 2$. Consequently, the approximate solution $y_N(x)$ of the main problem (689) is given by $y_N(x) = y_{1,N}(x)$.



Figure 33: Semi-log representation of the numerical error of Example 0.77 versus N .

4 Numerical example

Example 0.77. [2] Considering $a_1 = a_3 = 1$, $a_2 = \frac{\sqrt{\pi}}{5}$, $q(x) = 0$, $y(0) = 1$ and $y'(0) = 0$ the exact solution of (689) is given by

$$y(x) = 1 - \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \frac{(-1)^j (-\frac{\sqrt{\pi}}{5})^i (i+j)! x^{2+2j+\frac{i}{2}}}{i! j! (2+2j+\frac{i}{2}) \Gamma(2+2j+\frac{i}{2})}. \quad (541)$$

We solve this problem using the proposed method and report the obtained results in Table 39 and Figure 33. To compute the numerical errors, 200-terms of the series of right-hand side of (541) are considered. As it can be inferred, the numerical results imply the efficiency of our approach.

Table 39: L^∞ -error of the presented approach for Example 0.77 versus N .

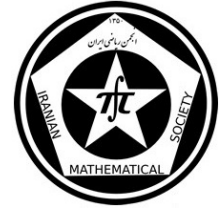
N	$\ e_N\ _{L^\infty}$	$CPU\text{-time (sec)}$
5	5.44×10^{-5}	2.23
10	1.40×10^{-10}	2.23
15	1.05×10^{-14}	3.23
20	7.05×10^{-16}	4.39

References

- [1] K. Diethelm, *The Analysis of Fractional Differential Equations*, Springer, Berlin, 2010.
- [2] A. Faghih, P. Mokhtary, *An efficient formulation of Chebyshev Tau method for constant coefficients systems of multi-order FDEs*, J. Sci. Comput., **82** (2020), 6.
- [3] P. Mokhtary, *Numerical treatment of a well-posed Chebyshev Tau method for Bagley-Torvik equation with high-order of accuracy*, Numer. Algo., **72** (2016), 875–891.
- [4] J. Shen, T. Tang, L. L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*, Springer, 2011.
- [5] M. A. Zaky, E. H. Doha, J. A. Tenreiro Machado, *A spectral framework for fractional variational problems based on fractional Jacobi functions*, Appl. Numer. Math., **132** (2018), 51–72.

a_faghih@sut.ac.ir[†]

mokhtary@sut.ac.ir[‡]



Numerical solution of nonlinear multi-order fractional differential equations using a novel shifted Jacobi operational matrix

H.R. Khodabandehlo^{†32}, Elyas Shivanian[‡]

Department of Mathematics, Imam Khomeini International University, Qazvin, Iran and

khodabandelo.hamidreza@yahoo.com[†]

Department of Mathematics, Imam Khomeini International University, Qazvin, 34149-16818, Iran and

shivanian@sci.ikiu.ac.ir[‡]

Abstract

This paper presents the generalized nonlinear multi-order fractional differential equations. In this article, a novel shifted Jacobi operational matrix technique is introduced for solving a class of these equations via reducing the main problem to an algebraic system of equations that can be solved numerically. The suggested technique is successfully developed for the aforementioned problem. Comprehensive numerical experiments are presented to demonstrate the efficiency, generality, accuracy of proposed scheme and the flexibility of this method. The numerical results compared it with other existing methods such as Reproducing Kernel Hilbert Space method (*RKHS*). Comparing the results of these methods as well as comparing the current method (*NSJOM*) with the exact solution, indicating the efficiency and validity of this method. Note that the procedure is easy to implement and this technique will be considered as a generalization of many numerical schemes. AMS subject Classification 2010: 34A08; 46E22; 65F25.

Keywords: Shifted Jacobi Operational Matrix technique, Nonlinear multi-order FDE.

16cm0.1pt

1 Introduction

The analysis and applications of fractional calculus is an active and the fastest growing region for research in the last three decades. It has currently become an important tool owing to their wide applications in various scientific disciplines, such as physics, regular variation in thermodynamics, etc. ([5] and references therein). The increasing development of appropriate and efficient method to solve *FDEs* has aroused more interest of researchers in this field. Solving the nonlinear *FDEs* is relatively difficult compared to its linear type. In this regard, analytical methods such as, homotopy perturbation (*HPM*), new iterative methods and Adomian decomposition, have been used in

³²Corresponding author's E-mail: khodabandelo.hamidreza@yahoo.com



the recent literature widely [2, 4, 5]. On the otherhand, some researchers like Diethelm et al [7, 5] have developed standard numerical methods, such as Adams-Bashforth methods have been used for numerical integration of *FDEs*.

In recent years, many attempts have been made to solve the *FDEs*, we refer the interested reader to [8, 9] and etc.

In this paper, the our aim is to generalize the orthogonal polynomials in the base of solution. In fact, we present a new shifted Jacobi operational matrix for the fractional derivative to solve the multi-order *FDEs* which as follow:

$$\sum_{j=1}^n \alpha_j D^{\eta_j} w(t) = F(t, w(t), D^{\eta_1} w(t), D^{\eta_2} w(t), \dots, D^{\eta_n} w(t)), \quad 0 \leq t \leq T, \\ w(t) = g(t), \quad t \in [-\tau, 0], \quad w^{(i)}(0) = 0, \quad i = 0, 1, 2, \dots, n-1, \quad n \in \mathbf{N}, \quad (542)$$

where $\alpha_j \in \mathbb{R} (j = 1, 2, \dots, n), 0 < T$. and $D^{\eta_j} w(t) (j = 1, 2, \dots, n)$ are the derivatives of variable-order fractional in the Caputo sense.

2 Fundamentals and preliminaries

This article is based on Caputo definition because, as well as know, only the Caputo sense has the same form as integer-order differential equations in initial conditions.

Definition 0.78. The left-sided Caputo fractional derivatives of order $\eta (n-1 < \eta \leq n)$ are defined as $D_+^{\eta} w(t) = \frac{1}{\Gamma(n-\eta)} \int_0^t \frac{U'(\tau)}{(t-\tau)^{\eta-n+1}} d\tau$.

Shifted Jacobi polynomials and their properties

Denote $P_n^{(\alpha, \beta)}(z); \alpha > -1, \beta > -1$ as the n -th order Jacobi polynomial in z defined on $[-1, 1]$. As all classic orthogonal polynomials, $P_n^{(\alpha, \beta)}(z)$ form an orthogonal system with respect to weight function $\omega^{(\alpha, \beta)}(z) = (1+z)^{\alpha} (1-z)^{\beta}$. If we implement the change of variable $z = (\frac{2t}{T} - 1)$, therefore, we have so-called the shifted Jacobi polynomials $P_n^{(\alpha, \beta)}(\frac{2t}{T} - 1)$ which denoted by $P_{T,i}^{(\alpha, \beta)}(t)$ on the interval $t \in [0, T]$. Also the i -th order Shifted Jacobi polynomial has the following analytical form [6]

$$P_{T,i}^{(\alpha, \beta)}(t) = \sum_{k=0}^i (-1)^{i-k} \frac{\Gamma(\alpha+i+1)\Gamma(\alpha+\beta+k+i+1)}{\Gamma(\alpha+\beta+i+1)\Gamma(\alpha+1+k)\Gamma(k+1)\Gamma(i-k+1)T^k} t^k. \quad (543)$$



3 Function approximation by shifted Jacobi polynomials

The function $w(t)$, square integrable with respect to $\omega_T^{(\alpha,\beta)}(t)$ in $[0, T]$, can be expanded as the following expression:[10, 6]

$$w(t) = \sum_{i=0}^{\infty} a_i P_{T,i}^{(\alpha,\beta)}(t), \quad (544)$$

where a_i (the coefficients of the series) are obtained by

$$a_i = \frac{1}{h_{T,j}^{(\alpha,\beta)}} \int_0^T \omega_T^{(\alpha,\beta)} P_{T,i}^{(\alpha,\beta)}(t) w(t) dt, \quad i = 0, 1, \dots \quad (545)$$

So, we can estimate the approximate solution by taking $(N+1)$ -terms of the series in Eq. (615) and we will have

$$w(t) \simeq w_N(t) = \sum_{i=0}^N a_i P_{T,i}^{(\alpha,\beta)}(t) = A^T \Phi_{T,N}(t), \quad (546)$$

where $A = [a_0, a_1, \dots, a_N]^T$, and $\Phi_{T,N}(t) = [P_{T,0}^{(\alpha,\beta)}(t), P_{T,1}^{(\alpha,\beta)}(t), \dots, P_{T,N}^{(\alpha,\beta)}(t)]^T$. Here, we suppose that $S(t) = [1, t, t^2, t^3, \dots, t^N]^T$. By equation (546), the vector $\Phi_{T,N}(t)$ can be presented as $\Phi_{T,N}(t) = B_{(\alpha,\beta)} S(t)$, where $B_{(\alpha,\beta)}$ is a square matrix of order $(N+1) \times (N+1)$ that given as follow

$$b_{i+1,j+1} = \begin{cases} (-1)^{i-j} \frac{(\alpha+i)!(\alpha+\beta+j+i)!}{(\alpha+\beta+i)!(\alpha+j)!(j!)(i-j)! T^j}, & i \geq j, \\ 0, & \text{otherwise.} \end{cases} \quad (547)$$

for $0 \leq i, j \leq N$. Hence, we get

$$S(t) = B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t). \quad (548)$$

4 Shifted Jacobi Polynomials Operational Matrix (SJOM)

At first, $D^{\eta_i(t)} \Phi_{T,N}(t)$, ($i = 1, 2, \dots, n$) can be deduced as the following: since $\Phi_{T,N}(t) = B_{(\alpha,\beta)} S(t)$, then we have

$$D^{\eta_i(t)} \Phi_{T,N}(t) = D^{\eta_i(t)} (B_{(\alpha,\beta)} S(t)) = B_{(\alpha,\beta)} D^{\eta_i(t)} [1, t, \dots, t^N]^T, \quad i = 1, 2, \dots, n. \quad (549)$$



Then

$$\begin{aligned}
 D^{\eta_i(t)} \Phi_{T,N}(t) &= B_{(\alpha,\beta)} D^{\eta_i(t)}(S(t)) = B_{(\alpha,\beta)} \left[0, \frac{\Gamma(2)t^{(1-\eta_i(t))}}{\Gamma(2-\eta_i(t))}, \dots, \frac{\Gamma(N+1)t^{(N-\eta_i(t))}}{\Gamma(N+1-\eta_i(t))} \right]^T \\
 &= B_{(\alpha,\beta)} \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{\Gamma(2)t^{-\eta_i(t)}}{\Gamma(2-\eta_i(t))} & 0 & \dots & 0 \\ 0 & 0 & \frac{\Gamma(3)t^{-\eta_i(t)}}{\Gamma(3-\eta_i(t))} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \frac{\Gamma(N)t^{-\eta_i(t)}}{\Gamma(N+1-\eta_i(t))} \end{bmatrix} \begin{bmatrix} 1 \\ t \\ t^2 \\ \vdots \\ t^N \end{bmatrix} \\
 &= B_{(\alpha,\beta)} Q_i(t) S(t), \quad i = 1, 2, \dots, n.
 \end{aligned} \tag{550}$$

Using Eq.(548), then

$$D^{\eta_i(t)} \Phi_{T,N}(t) = B_{(\alpha,\beta)} Q_i(t) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t), \quad i = 1, 2, \dots, n. \tag{551}$$

The operational matrix of $D^{\eta_i(t)} \Phi_{T,N}(t)$, ($i = 1, 2, \dots, n$.) is $B_{(\alpha,\beta)} Q_i(t) B_{(\alpha,\beta)}^{-1}$.

Now, we can estimate the multi-order fractional of the approximated function that obtained in Eq. (546). Hence the equation (689) turned into

$$\begin{aligned}
 \sum_{i=1}^n \alpha_i (A^T B_{(\alpha,\beta)} Q_i(t) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t)) + \alpha_{n+1} A^T \Phi_{T,N}(t - \tau) = \\
 F(t, A^T \Phi_{T,N}(t), (A^T B_{(\alpha,\beta)} Q_1(t) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t)), (A^T B_{(\alpha,\beta)} Q_2(t) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t)), \dots, \\
 (A^T B_{(\alpha,\beta)} Q_n(t) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t)), A^T \Phi_{T,N}(t - \tau)), \quad 0 \leq t \leq T, \tag{552}
 \end{aligned}$$

Finally, we use t_j ($j = 0, 1, 2, \dots, m$.) wher they are the roots of $P_{T,m+1}^{(\alpha,\beta)}(t)$. Then Eq. (623) can be converted into the following algebraic system

$$\begin{aligned}
 \sum_{i=1}^n \alpha_i (A^T B_{(\alpha,\beta)} Q_i(t_j) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t_j)) + \alpha_{n+1} A^T \Phi_{T,N}(t_j - \tau) = \\
 F(t_j, A^T \Phi_{T,N}(t_j), (A^T B_{(\alpha,\beta)} Q_1(t_j) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t_j)), (A^T B_{(\alpha,\beta)} Q_2(t_j) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t_j)), \dots, \\
 (A^T B_{(\alpha,\beta)} Q_n(t_j) B_{(\alpha,\beta)}^{-1} \Phi_{T,N}(t_j)), A^T \Phi_{T,N}(t_j - \tau)), \quad j = 0, 1, 2, \dots, m. \tag{553}
 \end{aligned}$$

So, the system in Eq. (553) can be solved numerically for determining the unknown vector A . Therefore, the numerical solution that presented in Eq. (546) can be obtained.



Table 40: Absolute errors of $w(t)$ with $\alpha = 0, \beta = 0$ and $T = 1$ for Ex.1.

$t \in [0, T]$	New technique, $N = 3$	Method in [3], $N = 50, w \in H^4$	Method in [3], $N = 50, w \in H^5$
0	4.163×10^{-17}	2.997×10^{-6}	1.178×10^{-6}
0.1	6.852×10^{-17}	2.997×10^{-6}	1.178×10^{-6}
0.2	7.459×10^{-17}	5.548×10^{-6}	3.389×10^{-6}
0.3	6.592×10^{-17}	2.784×10^{-5}	3.378×10^{-6}
0.4	4.501×10^{-17}	3.977×10^{-5}	1.431×10^{-5}
0.5	2.082×10^{-17}	5.259×10^{-5}	2.256×10^{-5}
0.6	0	6.034×10^{-4}	3.542×10^{-5}
0.7	2.775×10^{-17}	7.634×10^{-4}	5.157×10^{-4}
0.8	2.775×10^{-17}	5.114×10^{-4}	4.871×10^{-4}
0.9	2.775×10^{-17}	6.321×10^{-3}	1.456×10^{-3}
1.0	0	4.981×10^{-3}	2.641×10^{-3}

5 Numerical experinces

Example 0.79. [3] Consider the following multi-order fractional DE

$$D^{\eta_2} w(t) + D^{\eta_1} w(t) + w(t)^3 = \frac{2w(t)^{3-\eta_2}}{\Gamma(4-\eta_2)} + \frac{2w(t)^{3-\eta_1}}{\Gamma(4-\eta_1)} + \frac{t^9}{27},$$

$$w(t)^{(0)} = 0, \quad 0 < \eta_1 < \eta_2 < 1, \quad t \leq 0. \quad (554)$$

Note that $w(t) = \frac{t^3}{3}$ is the exact solution and $0 \leq t \leq T, T = 1, \eta_1 = \frac{1}{3}, \eta_2 = \frac{1}{2}$. By the concepts presented in Section 4, we consider the approximate solution with $(N+1)$ finite terms that presented in Eq.(546) for this problem and substitute in main problem. Then this problem is converted to form of the Eq.(623), finally using t_i , therefore, a system of algebraic equations emerges wich can be solved numerically to determine the unknown vector A with initial value $w_0 = 0$. The solution of the Eq.(554) approximated using new method compared to other methods, is in the best agreement with the exact solution. The absolute errors (at some nodal points) of this method and methods in [3](*RKHS*) are presented and compared in Table (54). From this Table, it is observed that the numerical results are very close to the exact solution and we achieved an excellent approximation for the exact solution by employing current scheme and it was found the our method in comparison with mentioned methods is better with view to utilization, accuracy and more time efficiency. In this example for $N = 3$, we have $A = [0.083333, 0.15, 0.083333, 0.016667]^T$, and the CPU time needed for this technique is 0.062400 s.

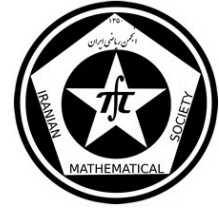


Conclusions

In the current paper, we have proposed the novel shifted Jacobi operational matrix (NSJOM) scheme for multi-order fractional differential equations via reducing the main problem to an algebraic system of equations that can be solved numerically. We have shown that the proposed method has good convergence, is easy to implement and its concepts are simple. The obtained results are excellent compared to other methods. Finally, the numerical results have been presented to clarify the effectiveness and accuracy of this technique.

References

- [1] K. Diethelm, N.J. Ford, A. D. Freed, *Detailed error analysis for a fractional Adams method*. Numer Algorithms 36 (1): 31-52, 2004.
- [2] G. Adomian, *A review of decomposition method in applied mathematics*. J. Math. Anal. Appl. 135, 501-544, 1988.
- [3] R. Khoshsiar Ghaziani, M. Fardi, M. Ghasemi, *Solving multi-order fractional differential equations by reproducing kernel Hilbert space method*, Computational Methods for Differential Equations, Vol. 4, No. 3, pp 170-190, 2016.
- [4] V. Daftardar-Gejji, H. Jafari, *An iterative method for solving nonlinear functional equations*. J. Math. Anal. Appl. 316, 753-763, 2006.
- [5] A. Hemeda, *Homotopy perturbation method of solving systems of nonlinear coupled equations*. Appl. Math. Sci. 6, 4787-4800, 20012.
- [6] A.A. El-Sayed, D. Baleanu, P. Agarwal, *A novel Jacobi operational matrix for numerical solution of multi-term variable-order fractional differential equations*, Journal of Taibah University for Science, 14:1, 963-974, 2020.
- [7] K. Diethelm, N.J. Ford, A.D. Freed, *A predictor-corrector approach for the numerical solution of fractional differential equations*. Nonlinear Dynamics 29, 3-22, 2002.
- [8] K. Diethelm, N.J. Ford, *Multi-order fractional differential equations and their numerical solution*, Appl. Math. Comput., 154 (2004), 621-640.
- [9] R. Gorenflo, *Fractional calculus: some numerical methods*, in: *Fractals and Fractional Calculus in Continuum Mechanics*, in: *CISM Lecture Notes*, Springer Verlag, Wien, 1997, pp.277290.
- [10] A.H. Bhrawy, M.A. Zaky, *A method based on the Jacobi tau approximation for solving multi-term time-space fractional partial differential equations*, J. Comput. Phys, 2014.



Numerical solution of spread option pricing model based on jump-diffusion models by ADI method

Reyhane Mohamadinejad^{†33}, Abdolsadeh Neisy[‡]

Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran.

e-mail: r.m.math81@gmail.com[†]

Department of Mathematics, Faculty of Mathematics Science and Computer, Allameh Tabatabai University, Tehran, Iran;

e-mail: a(neisy)@atu.ac.ir[‡]

Abstract

Jump models are common in pricing models, and it seems that economic, political and natural factors such as earthquake, flood, hurricane and others have a significant impact on the option price or underlying asset. If the jumps are not considered in the basic models, this can lead to incorrect results in option pricing and subsequently can cause great damage to the economy and has a negative impact on the market and investors. Long or short positions can control the risk of these high jumps. Therefore, we represent an appropriate and efficient strategy to describe the way we use the spread option on jump-diffusion Vasicek interest rate models and the credit spreads. First, we introduce two the jump-diffusion models and then the spread option is extracted from them. Finally, we solve the model by using the ADI method and implement it in MATLAB software.

Keywords: spread option, credit spread, Interest rate, ADI method.

1 Introduction

In financial definitions, a spread option is an option that its return is based on the price difference between the two underlying assets. The vertical spread option is a type of spread option which is divided into the credit spread and the debit spread. The amount of premium payable is reduced in the debit spread. In other words, when the market volatilities increase or when the implied volatility of certain stocks are high, the option premium can be very expensive.

The credit spread can reduce writing option risk to a large extent, because option writers take a significant risk to earn a relatively small amount of premium. A disastrous trade can wipe out the positive results of many successful options trading. Writing uncovered call options is one of the most dangerous option strategies, as the potential loss in the event of a business failure is theoretically unlimited. Writing Put options is relatively less risky, but a trader who has written puts on numerous stocks would be stuck with a large number of pricey stocks in a sudden market crash. The credit

³³Speaker



spread reduces this risk, although the cost of this reduction is less than the option premium.

Mohamadinejad et al (2021) have recently written a research article which is about pricing spread option based on the Vasicek interest rate and the credit spread [1]. The stability of the ADI method depends on the length of the time step, i.e., the smaller the step length, the more stable the method will be. So, in this article, we use the ADI method with $\frac{1}{10}$ step size to solve the model that is based on the two interest rates.

Zhou et al (2015) proposed a model for the credit spread option pricing which depended on time, the interest rate and the logarithm of the credit spread and was solved by the GARCH and Longstaff-Schwartz methods [2]. The weakness of this model was that it ignored the factors that caused the jumps.

In 2020, Mohamadinejad et al used the ADI method with $\frac{1}{6}$ step size to solve the spread option pricing model which was based on two Libor interest rates [3]. One of the interest rates had a jump-diffusion term. Safaei et al (2017) examined the ADI method with $\frac{1}{4}$ step size where the pricing was based on the Heston model [4].

This paper is organized as follows:

In Section 2, we study the modeling problem, the equation and the boundary and initial conditions where the PIDE depends on the Vasicek interest rate, the credit spread, and the time. In this section, this problem is solved numerically by applying the ADI method with $\frac{1}{10}$ step size, too. We implement this model on random data by using MATLAB software and represent the figures of some steps in section 3 and the stability of this method is proven by obtaining the distance between the solution steps in this section and we show that these differences are insignificant and converge to zero. Finally, section 4 presents conclusions and suggestions for future research.

2 Financial Modeling

In this section, we obtain the spread option model based on the Vasicek interest rate and the credit spread. Let dX is the logarithm of the credit spread and dr is the Vasicek interest rate which follow the dynamic models [2]:

$$dX = (a - bX)dt + sdW_1 + d\Sigma_{i=1}^{M_t} P_i, \quad dr = (\alpha - \beta r)dt + \sigma dW_2 + d\Sigma_{i=1}^{N_t} J_i. \quad (555)$$

Where a, b, s, α, β and σ are fixed numbers, W_1, W_2 are standard Brownian motions where $Corr(W_1, W_2) = \rho$, M_t and N_t are Poisson processes with the rates of λ_1 and λ_2 , respectively and P_i, J_i are Poisson distribution functions. Note that W_1, W_2 and M_t, N_t are pairwise independent and λ is positive.

Let π is the spread option price. The following PDE is obtained by using equation (555) and Itos Lemma and by considering τ as the backward time variable [3]:

$$\begin{aligned} \pi_\tau = & (a - bx - qs - s^2 + \frac{1}{\lambda_1} e^{-x} - \frac{1}{\lambda_1^2} e^{-2x}) \frac{\partial \pi}{\partial x} + (\alpha - \beta r - q\sigma - \sigma^2 + \frac{1}{\lambda_2} e^{-y} - \frac{1}{\lambda_2^2} e^{-2y}) \frac{\partial \pi}{\partial y} + \\ & \rho\sigma s \frac{\partial^2 \pi}{\partial x \partial y} + (s^2 + \frac{1}{\lambda_1^2} e^{-2x}) \frac{\partial^2 \pi}{\partial x^2} + (\sigma^2 + \frac{1}{\lambda_2^2} e^{-2y}) \frac{\partial^2 \pi}{\partial y^2} - (r + \lambda_1 + \lambda_2 - 2)\pi. \end{aligned} \quad (556)$$



The initial condition in $\tau = 0$ is equivalent to the payoff function, in other words

$$\pi(0, X, r) = \max\{0, X - K, r - K\}.$$

The boundary conditions for equation (556) will be as follows:

$$\frac{\partial^2 \pi}{\partial X^2}(0, r, \tau) = 0, \frac{\partial^2 \pi}{\partial x^2}(K, r, \tau) = 0, \frac{\partial^2 \pi}{\partial r^2}(X, 0, \tau) = 0, \frac{\partial^2 \pi}{\partial r^2}(X, H, \tau) = 0,$$

$$\frac{\partial \pi}{\partial X}(K, r, \tau) = 0, \frac{\partial \pi}{\partial r}(X, H, \tau) = 0, \pi(0, r, \tau) = \pi(X, 0, \tau) = 0.$$

3 ADI method with $\frac{1}{10}$ step size for numerical solution of spread option pricing model

First, we consider the PDE in equation (556). We define the $I\pi$ operator as follows:

$$I\pi = I^x\pi + I^{x,xx}\pi + I^{xx}\pi + I^y\pi + I^{yy}\pi + I^{y,yy}\pi + I^{xy}\pi + I^{x,xy}\pi + I^{y,xy}\pi + \Phi. \quad (557)$$

The coefficients of $I\pi$ are defined based on the coefficients of equation (556) and the methods for determining these coefficients can be different, but the sum of these coefficients must be consistent with the sum of the coefficients in equation (556).

If we write equation (556) as $\pi_\tau = A \frac{\partial \pi}{\partial x} + B \frac{\partial \pi}{\partial y} + C \frac{\partial^2 \pi}{\partial x \partial y} + D \frac{\partial^2 \pi}{\partial x^2} + E \frac{\partial^2 \pi}{\partial y^2} - F\pi$, the components of relation (557) are defined as follows [3,5]:

$$\begin{aligned} I^x\pi &= \frac{1}{3}A \frac{\partial \pi}{\partial x}, I^x\pi_{ij}^{n+\frac{1}{10}} = \frac{\pi_{ij}^{n+\frac{1}{10}} - \pi_{ij}^n}{\Delta\tau}, \quad I^{x,xx}\pi = \frac{1}{3}A \frac{\partial \pi}{\partial x} + \frac{1}{2}D \frac{\partial^2 \pi}{\partial x^2}, I^{x,xx}\pi = \frac{\pi_{ij}^{n+\frac{1}{5}} - \pi_{ij}^{n+\frac{1}{10}}}{\Delta\tau} \\ I^{xx}\pi &= \frac{1}{2}D \frac{\partial^2 \pi}{\partial x^2}, I^{xx}\pi_{ij}^{n+\frac{3}{10}} = \frac{\pi_{ij}^{n+\frac{3}{10}} - \pi_{ij}^{n+\frac{1}{5}}}{\Delta\tau}, \quad I^y\pi = \frac{1}{3}B \frac{\partial \pi}{\partial y}, I^y\pi_{ij}^{n+\frac{2}{5}} = \frac{\pi_{ij}^{n+\frac{2}{5}} - \pi_{ij}^{n+\frac{3}{10}}}{\Delta\tau} \\ I^{yy}\pi &= \frac{1}{2}E \frac{\partial^2 \pi}{\partial y^2}, I^{yy}\pi_{ij}^{n+\frac{1}{2}} = \frac{\pi_{ij}^{n+\frac{1}{2}} - \pi_{ij}^{n+\frac{2}{5}}}{\Delta\tau}, I^{y,yy}\pi = \frac{1}{3}B \frac{\partial \pi}{\partial y} + \frac{1}{2}E \frac{\partial^2 \pi}{\partial y^2}, I^{y,yy}\pi_{ij}^{n+\frac{3}{5}} = \frac{\pi_{ij}^{n+\frac{3}{5}} - \pi_{ij}^{n+\frac{1}{2}}}{\Delta\tau} \\ I^{xy}\pi &= \frac{1}{3}C \frac{\partial^2 \pi}{\partial x \partial y}, I^{xy}\pi_{ij}^{n+\frac{7}{10}} = \frac{\pi_{ij}^{n+\frac{7}{10}} - \pi_{ij}^{n+\frac{3}{5}}}{\Delta\tau}, I^{x,xy}\pi = \frac{1}{3}A \frac{\partial \pi}{\partial x} + \frac{1}{3}C \frac{\partial^2 \pi}{\partial x \partial y}, I^{x,xy}\pi_{ij}^{n+\frac{4}{5}} = \frac{\pi_{ij}^{n+\frac{4}{5}} - \pi_{ij}^{n+\frac{7}{10}}}{\Delta\tau} \\ I^{y,xy}\pi &= \frac{1}{3}B \frac{\partial \pi}{\partial y} + \frac{1}{3}C \frac{\partial^2 \pi}{\partial x \partial y}, I^{y,xy}\pi_{ij}^{n+\frac{9}{10}} = \frac{\pi_{ij}^{n+\frac{9}{10}} - \pi_{ij}^{n+\frac{4}{5}}}{\Delta\tau}, \Phi = -F\pi, \frac{\pi_{ij}^{n+1} - \pi_{ij}^{n+\frac{9}{10}}}{\Delta\tau} = \Phi. \quad (558) \end{aligned}$$

The next step in achieving a numerical solution is to establish the relationship between the finite differences and the relations in equation (557), which we solve step by step and recursively. That is,



the solution obtained from the first relation in equation (558) is replaced and used to solve the second relation in equation (558). Therefore, we obtain the following equations using the Adams-Moulton formula and the relations of equation (558) [3,4]:

$$\begin{aligned}
\frac{\pi_{ij}^{n+\frac{1}{10}} - \pi_{ij}^n}{\Delta\tau} &= I^x \pi_{ij}^{n+\frac{1}{10}}, \quad \frac{\pi_{ij}^{n+\frac{1}{5}} - \pi_{ij}^{n+\frac{1}{10}}}{\Delta\tau} = \frac{1}{2}(I^{x,xx} \pi_{ij}^{n+\frac{1}{10}} + I^{x,xx} \pi_{ij}^{n+\frac{1}{5}}), \\
\frac{\pi_{ij}^{n+\frac{3}{10}} - \pi_{ij}^{n+\frac{1}{5}}}{\Delta\tau} &= \frac{1}{12}(-I^{xx} \pi_{ij}^{n+\frac{1}{10}} + 8I^{xx} \pi_{ij}^{n+\frac{1}{5}} + 5I^{xx} \pi_{ij}^{n+\frac{3}{10}}), \\
&\vdots \\
\frac{\pi_{ij}^{n+\frac{9}{10}} - \pi_{ij}^{n+\frac{4}{5}}}{\Delta\tau} &= \frac{1}{3628800}(-33953I^{y,xy} \pi_{ij}^{n+\frac{1}{10}} + 312874I^{y,xy} \pi_{ij}^{n+\frac{1}{5}} - 1291214I^{y,xy} \pi_{ij}^{n+\frac{3}{10}} + \\
&3146338I^{y,xy} \pi_{ij}^{n+\frac{2}{5}} - 5033120I^{y,xy} \pi_{ij}^{n+\frac{1}{2}} + 5595358I^{y,xy} \pi_{ij}^{n+\frac{3}{5}} - 4604594I^{y,xy} \pi_{ij}^{n+\frac{7}{10}} + \\
&4467094I^{y,xy} \pi_{ij}^{n+\frac{4}{5}} + 1070017I^{y,xy} \pi_{ij}^{n+\frac{9}{10}}), \\
\frac{\pi_{ij}^{n+1} - \pi_{ij}^{n+\frac{9}{10}}}{\Delta\tau} &= -(\dot{r} + \lambda_1 + \lambda_2 - 2)\pi_{ij}^n.
\end{aligned} \tag{559}$$

We use the finite differences to solve the relations in equation (559).

$$\begin{aligned}
I^x \pi_{ij}^n &= \frac{\pi_{i+1j}^n - \pi_{ij}^n}{h}, \quad I^y \pi_{ij}^n = \frac{\pi_{ij+1}^n - \pi_{ij}^n}{k}, \\
I^{xx} \pi_{ij}^n &= \frac{\pi_{i+1j}^n - 2\pi_{ij}^n + \pi_{i-1j}^n}{h^2}, \quad I^{yy} \pi_{ij}^n = \frac{\pi_{ij+1}^n - 2\pi_{ij}^n + \pi_{ij-1}^n}{k^2}, \\
I^{xy} \pi_{ij}^n &= \frac{\pi_{i+1j+1}^n - \pi_{i+1j}^n - \pi_{ij+1}^n + 2\pi_{ij}^n - \pi_{i-1j}^n - \pi_{ij-1}^n + \pi_{i-1j-1}^n}{kh}.
\end{aligned} \tag{560}$$

Now it suffices to replace the above finite differences correctly in equation (560) and calculate the solution for $i = 0, 1, \dots, n$ and $j = 0, 1, \dots, m$. To summarize, solve the first and ninth equations as follows. From the first equation of equation (559), we get the following equation [3,6]:

$$(1 + A)\pi_{ij}^{n+\frac{1}{10}} - A\pi_{i+1j}^{n+\frac{1}{10}} = \pi_{ij}^n,$$

where $A = \frac{\Delta\tau}{3h}(a - bx_i - qs - s^2 + \frac{1}{\lambda_1}e^{x_i} - \frac{1}{\lambda_1^2}e^{-2x_i})$. Using the finite differences defined in equation (560), the ninth equation of relations (559) is written as follows [7]:

$$(1 - 1140034S\dot{N} + 1070017RS)\pi_{ij}^{n+\frac{9}{10}} - 1070017S(R\pi_{ij+1}^{n+\frac{9}{10}} + \dot{N}(\pi_{i+1j+1}^{n+\frac{9}{10}} - \pi_{i+1j}^{n+\frac{9}{10}} - \pi_{ij+1}^{n+\frac{9}{10}} - \pi_{i-1j}^{n+\frac{9}{10}}))$$



$$\begin{aligned}
& -\pi_{ij-1}^{n+\frac{9}{10}} + \pi_{i-1j-1}^{n+\frac{9}{10}})) = -33953S(R(\pi_{ij+1}^{n+\frac{1}{10}} - \pi_{ij}^{n+\frac{1}{10}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{1}{10}} - \pi_{ij+1}^{n+\frac{1}{10}} - \pi_{i+1j}^{n+\frac{1}{10}} + 2\pi_{ij}^{n+\frac{1}{10}} - \\
& \pi_{i-1j}^{n+\frac{1}{10}} - \pi_{ij-1}^{n+\frac{1}{10}} + \pi_{i-1j-1}^{n+\frac{1}{10}})) + 312874S(R(\pi_{ij+1}^{n+\frac{1}{5}} - \pi_{ij}^{n+\frac{1}{5}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{1}{5}} - \pi_{ij+1}^{n+\frac{1}{5}} - \pi_{i+1j}^{n+\frac{1}{5}} + 2\pi_{ij}^{n+\frac{1}{5}} - \\
& \pi_{i-1j}^{n+\frac{1}{5}} - \pi_{ij-1}^{n+\frac{1}{5}} + \pi_{i-1j-1}^{n+\frac{1}{5}})) - 1291214S(R(\pi_{ij+1}^{n+\frac{3}{10}} - \pi_{ij}^{n+\frac{3}{10}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{3}{10}} - \pi_{ij+1}^{n+\frac{3}{10}} - \pi_{i+1j}^{n+\frac{3}{10}} + 2\pi_{ij}^{n+\frac{3}{10}} - \\
& \pi_{i-1j}^{n+\frac{3}{10}} - \pi_{ij-1}^{n+\frac{3}{10}} + \pi_{i-1j-1}^{n+\frac{3}{10}})) + 3146338S(R(\pi_{ij+1}^{n+\frac{2}{5}} - \pi_{ij}^{n+\frac{2}{5}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{2}{5}} - \pi_{ij+1}^{n+\frac{2}{5}} - \pi_{i+1j}^{n+\frac{2}{5}} + 2\pi_{ij}^{n+\frac{2}{5}} - \\
& \pi_{i-1j}^{n+\frac{2}{5}} - \pi_{ij-1}^{n+\frac{2}{5}} + \pi_{i-1j-1}^{n+\frac{2}{5}})) - 5033120S(R(\pi_{ij+1}^{n+\frac{1}{2}} - \pi_{ij}^{n+\frac{1}{2}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{1}{2}} - \pi_{ij+1}^{n+\frac{1}{2}} - \pi_{i+1j}^{n+\frac{1}{2}} + 2\pi_{ij}^{n+\frac{1}{2}} - \\
& \pi_{i-1j}^{n+\frac{1}{2}} - \pi_{ij-1}^{n+\frac{1}{2}} + \pi_{i-1j-1}^{n+\frac{1}{2}})) + 5595358S(R(\pi_{ij+1}^{n+\frac{3}{5}} - \pi_{ij}^{n+\frac{3}{5}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{3}{5}} - \pi_{ij+1}^{n+\frac{3}{5}} - \pi_{i+1j}^{n+\frac{3}{5}} + 2\pi_{ij}^{n+\frac{3}{5}} - \\
& \pi_{i-1j}^{n+\frac{3}{5}} - \pi_{ij-1}^{n+\frac{3}{5}} + \pi_{i-1j-1}^{n+\frac{3}{5}})) - 4604594S(R(\pi_{ij+1}^{n+\frac{7}{10}} - \pi_{ij}^{n+\frac{7}{10}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{7}{10}} - \pi_{ij+1}^{n+\frac{7}{10}} - \pi_{i+1j}^{n+\frac{7}{10}} + 2\pi_{ij}^{n+\frac{7}{10}} - \\
& \pi_{i-1j}^{n+\frac{7}{10}} - \pi_{ij-1}^{n+\frac{7}{10}} + \pi_{i-1j-1}^{n+\frac{7}{10}})) + 4467094S(R(\pi_{ij+1}^{n+\frac{4}{5}} - \pi_{ij}^{n+\frac{4}{5}}) + \dot{N}(\pi_{i+1j+1}^{n+\frac{4}{5}} - \pi_{ij+1}^{n+\frac{4}{5}} - \pi_{i+1j}^{n+\frac{4}{5}} + \\
& 2\pi_{ij}^{n+\frac{4}{5}} - \pi_{i-1j}^{n+\frac{4}{5}} - \pi_{ij-1}^{n+\frac{4}{5}} + \pi_{i-1j-1}^{n+\frac{4}{5}})) + \pi_{ij}^{n+\frac{4}{5}},
\end{aligned}$$

where $R = \frac{1}{3}D$, $M = \frac{1}{3}\rho\sigma s$, $S = \frac{\Delta\tau}{3628800k}$, and $\dot{N} = \frac{M}{h}$.

4 Numerical result

The parameters and values used in MATLAB are given in table (1). We find the spread option price and the last step of the solution is shown in figure (1). (According to the $\frac{1}{10}$ step size, we will have 10 figures, but in order to summarize the content, we have included only 2 figures in this article.



Table 41: Data

<i>Parameters</i>	Definitions	Values
σ	volatility	0.3
K	maximum value of X	30
H	maximum value of r	30
$\dot{r}$	Interest rate	0.03
λ_1	intensity rate of Poisson process	0.1
λ_2	intensity rate of Poisson process	0.5
ρ	$corr(W_1, W_2)$	0.35
q	the market price of the risk	1
T	maximum value of t	0.5
a, b , and s	constant	0.5, 0.5, and 0.25

The difference between the solutions of successive steps with $\Delta x = \Delta r = 1, \Delta t = \frac{1}{14}$ step size is calculated in MATLAB and the solutions are as follows:

0.1218, 0.0216, 0.0132, 0.0102, 0.0037, 0.0033, 0.0026, 0.0020, 0.0000.

As can be seen, the difference is insignificant and eventually reaches zero. So, this method is suitable and stable for solving this type of pricing model and the stability of the method is proven.

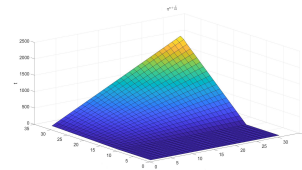
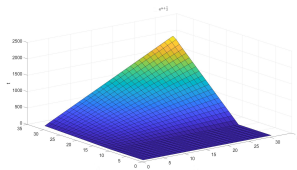


Figure 34: Numerical results using the ADI method with a step-size of $\frac{1}{10}$ for $n = \frac{1}{2}$, and $\frac{7}{10}$

5 coclusion

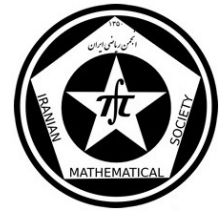
In this article, we studied the spread option pricing based on the models of the Vasicek interest rate and the credit spread. Both basic equations have a jump term. The existence of a jump in the modeling can cause many challenges in solving the model. To resolve these problems, we used the



ADI method with $\frac{1}{10}$ step size and Adams-Moulton coefficients. Note that the other models such as Heston and CIR can be used instead of the Vasicek interest rate model. Other methods such as wavelets, machine learning and RBF can be applied to numerically solve the PDE [8]. The PDE introduced in the article is new and has not been solved by any other methods. For future research, we recommend that the model is solved by other methods and compared with them and if it is possible the better method for solving the model is represented.

References

- [1] R. Mohamadinejad, A. Neisy, J. Biazar. *ADI method of credit spread option pricing based on jump-diffusion model*, Iranian Journal of Numerical Analysis and Optimization, (2021).
- [2] Zhou Rongxi, Du Sinan, Yu Mei, Yang Fengmei. *Pricing Credit Spread Option with Longstaff-Schwartz and GARCH Models in Chinese Bond Market*, J Syst Sci Complex, 28 (2015) 1363-1373.
- [3] R. Mohamadinejad, J. Biazar, A. Neisy. *Spread Option Pricing Using Two Jump-Diffusion Interest Rates*, U.P.B. Sci. Bull., Series A, Vol. 82, Iss. 1, (2020) 171-182.
- [4] M. Safaei at el. *New Splitting Scheme for Pricing American Options Under the Heston Model*, Springer, (2017).
- [5] G.I. Marchuk. *Splitting and Alternating Direction Methods*, 1990.
- [6] Randall J. Leveque. *Finite Difference Methods for Ordinary and Partial Differential Equations*, (2007).
- [7] Darae Jeong, Junseok Kim. *A comparison study of ADI and operator splitting methods on option pricing models*, Journal of Computational and Applied Mathematics, 247 (2013) 162-171.
- [8] Mohammad Karimnejad Esfahani, Abdolsadeh Neisy, Stefano De Marchi. *An RBF approach for oil futures pricing under the jump-diffusion model*, Journal of Mathematical Modeling, Vol. 9, No. 1, (2021), 8192.



Numerical solution of the fractional boundary value problems by the Sinc-Collocation method

N. Taheri^{†34} Z. Balali[†]

Department of Mathematics, Islamic Azad University, South Tehran Branch, Tehran, Iran [†]

Abstract

In this paper, we first convert a class of fractional differential equations (FDEs) to an equivalent fractional order Volterra integral equations, and then we applied Sinc-Collocation method. Errors in the solutions are demonstrated in the tables. We conclude that our approach converge rapidly with the exponential order.

AMS subject Classification 2010: 34A08, 45D05, 65L10, 65L60.

Keywords: Fractional differential equations, Integral equations, Sinc-Collocation method.

1 Introduction

We consider a class of fractional order boundary value problem with constant coefficients in the following form[3, 6]

$$y'(x) + \lambda {}^C D_{0+}^{\alpha} y(x) + \mu y(x) = g(x), \quad 0 < x \leq 1, \quad 0 < \alpha < 1, \lambda, \mu \in \mathbb{R} \quad (561)$$

subjected to homogeneous boundary conditions:

$$y(0) = 0, \quad y(1) = 0, \quad (562)$$

where y and g are continuous and bounded in interval $(0, 1]$, and ${}^C D_{0+}^{\alpha} y$ is an operator of the left Caputo fractional derivative of order α of y is defined by the following definition:

Definition 0.80. [3, 4] Let $y : [a, b] \rightarrow \mathbb{R}$ be a function, α a positive real number, n the integer satisfying $n - 1 < \alpha < n$, and Γ the Euler gamma function. Then, by considering the homogeneous boundary conditions (562), the left Caputo fractional integral and derivative of order α of $y(x)$ respectively is given as follows:

$$\begin{aligned} ({}^C I_{a+}^{n-\alpha} y)(x) &= \frac{1}{\Gamma(n-\alpha)} \int_a^x \frac{y(t)}{(x-t)^{\alpha-n+1}} dt, \quad (x > a; \mathcal{R}(\alpha) > 0), \\ ({}^C D_{a+}^{\alpha} y)(x) &= \frac{1}{\Gamma(n-\alpha)} \int_a^x \frac{y^{(n)}(t)}{(x-t)^{\alpha-n+1}} dt =: (I_{a+}^{n-\alpha} D^n y)(x). \end{aligned} \quad (563)$$

³⁴Speaker



In particular, when $0 < \alpha \leq 1$ and $y(x) \in (0, 1]$, and by considering $n = [\mathcal{R}(\alpha)] + 1$, the left Caputo fractional integral and derivative of order α of $y(x)$ respectively is given as follows:

$$({}^C I_{0+}^{1-\alpha} y)(x) = \frac{1}{\Gamma(1-\alpha)} \int_0^x \frac{y(t)}{(x-t)^\alpha} dt, \quad (564)$$

$$({}^C D_{0+}^\alpha y)(x) = \frac{1}{\Gamma(1-\alpha)} \int_0^x \frac{y'(t)}{(x-t)^\alpha} dt =: (I_{0+}^{1-\alpha} Dy)(x). \quad (565)$$

Lemma 0.81. [3] Let $\mathcal{R}(\alpha) > 0$ and let $n = [\mathcal{R}(\alpha)] + 1$. If $y(x)$ be continuous and bounded on the interval $[a, b]$, then

$$(I_{a+}^\alpha {}^C D_{a+}^\alpha y)(x) = y(x) - \sum_{k=0}^{n-1} \frac{y^{(k)}(a)}{k!} (x-a)^k.$$

In particular, if $0 < \alpha \leq 1$ and $y(x) \in (0, 1]$, then

$$(I_{a+}^\alpha {}^C D_{a+}^\alpha y)(x) = y(x) - y(0). \quad (566)$$

By using relation (564) for $0 < \alpha \leq 1$, Lemma 0.81, and homogeneous boundary conditions (562), we can formulate the FDEs (561)-(562) in the terms of Volterra integral equations as follows:

$$\begin{aligned} & \frac{1}{\lambda \Gamma(1-\alpha)} \int_0^x \frac{y'(t)}{(x-t)^\alpha} dt + y(x) + \frac{\mu}{\lambda \Gamma(1-\alpha)} \int_0^x \frac{y(t)}{(x-t)^\alpha} dt \\ &= \frac{1}{\lambda \Gamma(1-\alpha)} \int_0^x \frac{g(t)}{(x-t)^\alpha} dt. \end{aligned} \quad (567)$$

2 Sinc function preliminaries and the Sinc-Collocation method

We approximate solution of integral equation (567) on the interval $(0, 1]$, by the finite expansion of Sinc basis function:

$$\text{Sinc}(z) = \begin{cases} \frac{\sin(\pi z)}{\pi z}, & z \neq 0; \\ 1, & z = 0. \end{cases}$$

Definition 0.82. [7] We assume that $\mathcal{D} = \{z \in \mathbf{C} : |\text{Im}(z)| < d\}$ is a simply connected domain in the complex plane($\mathbf{C}$). We consider two separate points a and b of $\partial\mathcal{D}$ (boundary of $\mathcal{D}$). In (a, b) , we define a conformal map $\varphi_{a,b}(z) = \ln(\frac{z-a}{b-z})$, which has inverse $\psi_{a,b}(z) = \varphi_{a,b}^{-1}(z) = \frac{b-a}{2} \tanh(\frac{z}{2}) + \frac{b+a}{2}$. For $\varphi_{a,b}$, $\psi_{a,b}$, and a step size $h > 0$, we consider $z_k = \psi_{a,b}(kh)$, $k \in \mathbf{Z}$ which is the Sinc points.

Theorem 0.83. [8] We assume $\mathbf{L}_\beta(\mathcal{D})$ is the set of all analytic functions and $y \in \mathbf{L}_\beta(\mathcal{D})$ for $\beta > 0$. By considering $h = (\pi d/(\beta N))^{1/2}$, there exist a constant $k_1 > 0$, so

$$\left| y(z) - \sum_{j=-N}^N y(z_j) \Psi_j \right| \leq k_1 e^{-(\pi d \beta N)^{1/2}}. \quad (568)$$



We define the basis function Ψ_j as follows:

$$\Psi_j(\varphi(z)) = \begin{cases} \frac{1}{1+e^{\varphi(z)}} - \sum_{k=-N+1}^N \frac{1}{1+e^{kh}} S(k, h) \circ \varphi(z), & j = -N, \\ S(j, h) \circ \varphi(z), & j = -N+1, \dots, N-1, \\ \frac{e^{\varphi(z)}}{1+e^{\varphi(z)}} - \sum_{k=-N}^{N-1} \frac{e^{kh}}{1+e^{kh}} S(k, h) \circ \varphi(z), & j = N, \end{cases} \quad (569)$$

with $S(j, h)$ the shifted Sinc

$$S(j, h) \circ \varphi(z) = \text{Sinc}([\varphi(z) - jh]/h), \quad j = -N, \dots, N. \quad (570)$$

The following theorem involves bounding the error of $2N+1$ point Sinc quadrature of y on $\Lambda = [0, 1]$.

Theorem 0.84. [7] We consider $\frac{y}{\varphi'} \in L_\beta(\mathcal{D})$, $\beta > 0$, and $0 < d \leq \pi$. Let N be a positive integer, and let h be selected by the formula $h = (\pi d/(\alpha N))^{1/2}$. Then there exists a constant, $k_2 > 0$, independent of N , such that

$$\left| \int_\Lambda y(z) dz - h \sum_{k=-N}^N \frac{y(z_k)}{\varphi'(z_k)} \right| \leq k_2 e^{-(\pi d \beta N)^{1/2}}. \quad (571)$$

Sinc-Collocation method

We assume an approximate solution for $y(x)$ of the integral equation (567) by the finite expansion of sinc basis functions [1, 5, 8]

$$y_m(x) = \sum_{j=-N}^N c_j \Psi_j(\varphi(x)), \quad m = 2N+1, \quad (572)$$

where $\Psi_j(\varphi(x))$ is defined in (569). The unknown coefficients c_j in (572) are determined by Sinc-Collocation method. For this purpose, the first derivative of $y_m(x)$ is given by

$$\frac{d}{dx} y_m(x) = \sum_{j=-N}^N c_j \varphi'(x) \frac{d}{d\varphi} \Psi_j(\varphi(x)). \quad (573)$$

Now we try to approximate the integral parts of the equation (567)[2]. First by using the left Caputo fractional integral (564) and (565), we transform the given interval $(0, x)$ to $(-\infty, \infty)$ by the change of variable $s = \psi_{0,x}(z) = \frac{x-0}{2} \tanh(\frac{z}{2}) + \frac{x+0}{2}$, and definition 2.1. The integrals $\int_0^x \frac{y(t)}{(x-t)^\alpha} dt$ and $\int_0^x \frac{y'(t)}{(x-t)^\alpha} dt$ of equation (567) can be approximated as follows:

$$\begin{aligned} ({}^C I_{0+}^\alpha y^{(n)})(x) &= \frac{1}{\Gamma(\alpha)} \int_0^x \frac{y^{(n)}(s)}{(x-s)^\alpha} ds = \frac{1}{\Gamma(\alpha)} \int_{-\infty}^\infty \frac{y^{(n)}(\psi_{0,x}(z)) (\psi_{0,x})'(z)}{(x-\psi_{0,x}(z))^\alpha} dz \\ &= \frac{(x-0)^\alpha}{\Gamma(\alpha)} \int_{-\infty}^\infty \zeta(z) y^{(n)}(\psi_{0,x}(z)) dz; \quad n = 0, 1, \end{aligned} \quad (574)$$

where $\zeta(z) = \frac{1}{(1+e^{-z})(1+e^z)^\alpha}$, and applying the Sinc quadrature rule (571) for $z_k = \psi(kh)$, we obtain

$$({}^C I_{0+}^\alpha y^{(n)})(x) = \frac{x^\alpha}{\Gamma(\alpha)} h \sum_{k=-N}^N \frac{y^{(n)}(\psi_{0,x}(kh))}{(1+e^{-kh})(1+e^{kh})^\alpha}; \quad n = 0, 1, \quad (575)$$



where $h = (\pi d/(\beta N))^{1/2}$ is a mesh size depending on N .

Upon replacing $y(x)$ in the Volterra integral equation (567) with cardinal series $y_m(x)$ which has been defined in (572), Sinc function defined in (569), relation (575), substituting $\phi_{0,k}(x)$ instead of $\varphi_{0,1}(\psi_{0,x}(kh))$, setting points $x_l = \frac{e^{lh}}{1+e^{lh}}$, and $\phi_{0,k}(x_l) = \frac{x_l e^{lh}}{1+e^{lh}}$ finally, we get the following system for the determination of the unknown coefficients $c_{-N}, \dots, c_N$ of equation (572):

$$\begin{aligned} & \frac{x_l^\alpha}{\lambda \Gamma(\alpha)} h \sum_{k=-N}^N \sum_{j=-N}^N \frac{c_j}{(1+e^{-kh})(1+e^{kh})^\alpha} \left(\frac{d}{dx} (\Psi_j(\phi_{0,k}(x))) \right) \Big|_{x=x_l} \\ & + \sum_{j=-N}^N c_j \Psi_j(\varphi(x_l)) \\ & + \frac{\mu}{\lambda} \frac{x_l^\alpha}{\Gamma(\alpha)} h \sum_{k=-N}^N \sum_{j=-N}^N \frac{c_j}{(1+e^{-kh})(1+e^{kh})^\alpha} \Psi_j(\phi_{0,k}(x_l)) \\ & - \frac{x_l^\alpha}{\lambda \Gamma(\alpha)} h \sum_{k=-N}^N \frac{g(\phi_{0,k}(x_l))}{(1+e^{-kh})(1+e^{kh})^\alpha} = 0, \quad l = -N, \dots, N. \end{aligned} \quad (576)$$

3 Main results

In this section, all experiments were performed in Mathematica 11.0. Also, in order to show the errors and the accuracy of the approximation, on the set of sinc grid points

$$S = \{x_{-N}, \dots, x_N\}, \quad x_k = \frac{e^{kh}}{1+e^{kh}}, \quad k = -N, \dots, N,$$

we apply the following criteria:

1. The relative error is defined by

$$E_{rel} = \left| \frac{y_m(x_k) - y(x_k)}{y(x_k)} \right|. \quad (577)$$

2. The maximum absolute error is defined by

$$E_{abs} = \max_{-N \leq k \leq N} |y_m(x_k) - y(x_k)|. \quad (578)$$

3. The root mean square (RMS) error is defined for $M = 2N + 1$ by

$$RMS = \sqrt{\frac{1}{M} \sum_{k=-N}^N (y(x_k) - y_m(x_k))^2}. \quad (579)$$

4. The L_2 error norm is defined by

$$\|\cdot\|_2 = \sqrt{\sum_{k=-N}^N (y_m(x_k) - y(x_k))^2}. \quad (580)$$

In our presented method, we take $d = \frac{\pi}{2}$, $\beta = 1$ and we applied our procedure for $N = 20, 30, 40$, and 50 and by using $h = (\pi d/(\beta N))^{1/2}$ we can achieve h .



Table 42: Relative errors in the solution

x	$N = 20$	$N = 30$	$N = 40$	$N = 50$
0.1	6.35×10^{-2}	6.15×10^{-3}	1.01×10^{-3}	7.34×10^{-5}
0.3	3.77×10^{-3}	1.14×10^{-3}	5.09×10^{-4}	2.55×10^{-4}
0.5	3.52×10^{-3}	1.50×10^{-3}	6.88×10^{-4}	3.43×10^{-4}
0.7	4.79×10^{-3}	1.84×10^{-3}	8.41×10^{-4}	4.20×10^{-4}
0.9	6.31×10^{-3}	2.38×10^{-3}	1.09×10^{-3}	5.44×10^{-4}

Table 43: Errors in the solution

N	$h = \frac{\pi}{\sqrt{N}}$	$E_{abs}(h)$	$ \cdot _2$	RMS
10	0.993458	1.72×10^{-3}	2.72×10^{-3}	5.93×10^{-4}
20	0.702481	4.30×10^{-4}	9.35×10^{-4}	1.46×10^{-4}
30	0.573573	1.73×10^{-4}	4.23×10^{-4}	5.41×10^{-5}
40	0.496729	7.77×10^{-5}	2.13×10^{-4}	2.37×10^{-5}
50	0.444289	3.87×10^{-5}	1.15×10^{-4}	1.15×10^{-5}

Example 0.85. We consider the following FDE:

$$y'(x) + {}^C D_{0+}^{0.3} y(x) + y(x) = \frac{120}{\Gamma(5.7)} x^{4.7} - \frac{24}{\Gamma(4.7)} x^{3.7} + x^4(x-1) + 5x^4 - 4x^3,$$

with the boundary conditions $y(0) = 0$ and $y(1) = 0$. The exact solution is $y(x) = x^4(x-1)$.

For different values of N , the relative errors (577) in the solution in range $[0, 1]$ are tabulated in Table 1. Table 2 lists the errors (578), (579), and (580). Fig. 1 shows the graph of the approximate and exact solutions which are coincide together. Numerical results in the tables show that by increasing the number of terms N , the errors have been decreased.

References

- [1] S. Alkan, M. C. Serifoglu, and T. Yeloglu, *Comparison of sinc methods for the solution of fractional boundary value problems*, Journal of Abstract and Computational Mathematics, (2016), N.1, pp.13-23.
- [2] G. Baumann, and F. Stenger, *Fractional calculus and Sinc methods*, Fractional Calculus and Applied Analysis, (2011), 14, pp. 568–622.

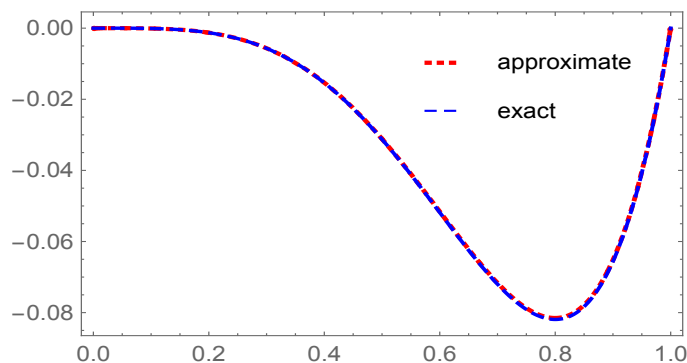
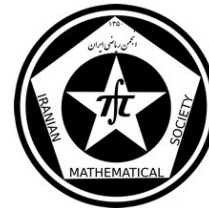


Figure 35: Comparison between the exact and approximate solutions

- [3] A.A. Kilbas, *Theory and Applications of Fractional Differential Equations*, Elsevier, Amsterdam, Netherlands, 2006.
- [4] T. Okayama, T. Matsuo, M. Sugihara, *Approximate Formula for Fractional Derivatives by Means of Sinc Methods*, Journal of Concrete and Applicable Mathematics, 8, (2010).
- [5] J. Rashidinia, M. Nabati, *Sinc-Galerkin and Sinc-Collocation methods in the solution of non-linear two-point boundary value problems*, Comput. Appl. Math., (2013), V.32, N.2, pp.315-330.
- [6] M.U.Rehman, R.A. Khan, *A numerical method for solving boundary value problems for fractional differential equations*, Appl. Math. Model., (2012), V.36, N.3, pp.894-907.
- [7] F. Stenger, *Numerical method based on sinc and analytic functions*, Springer-Verlag New York, Inc, 1993.
- [8] F. Stenger, *Handbook of Sinc Numerical Methods*. CRC Press, 2010.

Email: na_taheri@yahoo.com

Email: za.shadi899@gmail.com



Numerical solution of the inverse Gardner equation

Naser Azizi^{†35}, Reza Pourgholi[‡]

School of Mathematics and Computer Science, Damghan University, Damghan, Iran. E-mail: a.uni.a.13491349@std.du.ac.ir[†]

School of Mathematics and Computer Science, Damghan University, Damghan, Iran. E-mail: pourgholi@du.ac.ir[‡]

Abstract

In this paper, the numerical solution of the inverse Gardner equation will be considered. The Haar wavelet collocation method (HWCN) will be used to determine the unknown boundary condition which is estimated from an **over-specified** condition at a boundary. We determine a stable numerical solution to this equation. Numerical results show that a good estimation of the unknown functions of the inverse problem can be obtained within a couple of minutes of CPU time at **Pentium** IV-2.53 GHz PC.

AMS subject Classification 2010: 65M32, 65T60.

Keywords: Haar wavelet, Inverse problems, Numerical solution.

1 Introduction

Inverse problems have become more and more important in various fields of science and technology. They arise, for example, in the study of heat conduction processes, diffusion, control theory, and have certainly been one of the fastest-growing areas in applied mathematics over the last three decades due partly to its importance in applications. However, as inverse problems typically lead to mathematical models that are ill-posed, their solutions are unstable under data perturbations and classical numerical techniques fail to provide accurate and stable solutions.

Haar wavelets are based on the functions which were introduced by Hungarian mathematician Alfred Haar in 1910. The Haar wavelets are made up of piecewise constant functions and are mathematically the simplest among all the wavelet families. A good feature of these wavelets is the possibility to integrate them analytically at arbitrary times. They can be interpreted as a first-order Daubechies wavelet. Haar wavelets are the only compactly supported orthogonal wavelets that have explicit formulas. **Applications of the Haar wavelet method for solving the nonlinear systems in physics, biology, chemical reactions, and fluid mechanics are seen in [1, 5]. In the field of numerical solution of the inverse problems, using the Haar wavelet method, you can refer to references [2, 6].**

In this paper, as the main problem, we use the HWCN for solving the inverse Gardner equation in the dimensionless form;

$$u_t + 2\alpha uu_x - 3\beta u^2 u_x + u_{xxx} = 0, \quad \alpha, \beta > 0, \quad (581)$$

³⁵Speaker



with the initial condition

$$u(x, 0) = f(x), \quad x \in [0, 1], \quad (582)$$

the boundary conditions

$$u(0, t) = g(t), \quad u(1, t) = q(t), \quad u_x(1, t) = h(t), \quad t \in [0, t_F], \quad (583)$$

and the overspecified conditions

$$u(\ell, t) = k_1(t), \quad u_x(\ell, t) = k_2(t), \quad t \in [0, t_F], \quad (584)$$

where α and β are arbitrary constants, $0 < \ell < 1$ is fixed point, $f(x)$, $q(t)$, $h(t)$, $k_1(t)$, and $k_2(t)$ are piecewise known continuous functions, and t_F represents the final existence time, while function $g(t)$ is unknown which remains to be determined from some interior temperature measurements. **The existence and uniqueness of (581)-(584) are stated in [1].**

The paper is arranged as follows: in Section 36, the Haar wavelets family and their integrals are introduced. In Section 36, the application and procedure of implementation of the method, are presented. The numerical results are reported in Section 36 and finally, in Section 36, we conclude our work.

2 Haar wavelets

Definition 0.86. [4] The Haar wavelet family for $x \in [0, 1)$ is defined as follows:

$$h_i(x) = \begin{cases} 1, & x \in [\frac{k}{m}, \frac{k+0.5}{m}), \\ -1, & x \in [\frac{k+0.5}{m}, \frac{k+1}{m}), \\ 0, & o.w. \end{cases}$$

integer $m = 2^j$, ($j = 0, 1, \dots, J$) indicates the level of the wavelet, $k = 0, 1, \dots, m - 1$ is the translation parameter. Maximal level of resolution is J . The index i is calculated according the formula $i = m + k + 1$. In the case of minimal values $m = 1$, $k = 0$ we have $i = 2$, the maximal value of i is $2^{J+1} = M$. It is assumed that the value $i = 1$ corresponds to the scaling function for which $h_1 \equiv 1$ in $[0, 1)$. The integrals of Haar functions are obtained as follows:

$$p_{v,i}(x) = \underbrace{\int_0^x \int_0^x \dots \int_0^x}_{v\text{-times}} h_i(t) dt^v = \frac{1}{(v-1)!} \int_0^x (x-t)^{v-1} h_i(t) dt, \quad (585)$$

where, $v = 1, 2, \dots, n$ and $i = 1, 2, \dots, M$.

3 Overview of the Method

In this section, we present our method for solving inverse Gardner equation (581). Now, Let us divide the interval $[0, t_F]$ into N equal parts of length $\Delta t = \frac{t_F}{N}$ and denote $t_s = (s-1)\Delta t$, $s = 1, 2, \dots, N+1$.



We assume that $\dot{u}'''(x, t)$, can be expanded in terms of the Haar wavelets as [6],

$$\dot{u}'''(x, t) \cong c_1^s h_1(x) + \sum_{j=0}^J \sum_{k=0}^{2^j-1} c_{2^j+k+1}^s h_{2^j+k+1}(x) = C_M^T H_M(x), \quad (586)$$

where dot and prime mean differentiation with respect to t and x , respectively. Also, the vector C_M^T is constant in each subinterval $[t_s, t_{s+1}]$, $s = 1, 2, \dots, N$.

We integrate (586) one time with respect to t from t_s to t and three times with respect to x from ℓ to x . By using the boundary conditions (583) and overspecified conditions (584), and then, discretizing the results by assuming $x \rightarrow x_l = \frac{l-0.5}{M}$ and $t \rightarrow t_{s+1}$, we have the following schemes:

$$u'''(x_l, t_{s+1}) = \Delta t C_M^T H_M(x_l) + u'''(x_l, t_s), \quad (587)$$

$$\begin{aligned} \dot{u}(x_l, t_{s+1}) &= C_M^T \left[P_3(x_l) - \frac{(x_l - \ell)^2}{(1 - \ell)^2} P_3(1) \right] + \left[1 - \frac{(x_l - \ell)^2}{(1 - \ell)^2} \right] k_1'(t_{s+1}) \\ &\quad - (x_l - \ell) \frac{x_l - 1}{1 - \ell} k_2'(t_{s+1}) + \frac{(x_l - \ell)^2}{(b - \ell)^2} q'(t_{s+1}), \end{aligned} \quad (588)$$

$$\begin{aligned} u'(x_l, t_{s+1}) &= \Delta t C_M^T \left[P_2(x_l) - \frac{x_l - \ell}{b - \ell} P_2(1) \right] + u'(x_l, t_s) \\ &\quad + \frac{x_l - \ell}{1 - \ell} \left[h(t_{s+1}) - h(t_s) \right] - \frac{x_l - 1}{1 - \ell} \left[k_2(t_{s+1}) - k_2(t_s) \right], \end{aligned} \quad (589)$$

$$\begin{aligned} u(x_l, t_{s+1}) &= \Delta t C_M^T \left[P_3(x_l) - \frac{(x_l - \ell)^2}{(1 - \ell)^2} P_3(1) \right] + \left[1 - \frac{(x_l - \ell)^2}{(1 - \ell)^2} \right] \left[k_1(t_{s+1}) - k_1(t_s) \right] \\ &\quad - (x_l - \ell) \frac{x_l - 1}{1 - \ell} \left[k_2(t_{s+1}) - k_2(t_s) \right] + \frac{(x_l - \ell)^2}{(1 - \ell)^2} \left[q(t_{s+1}) - q(t_s) \right] + u(x_l, t_s), \end{aligned} \quad (590)$$

where, $P_2(x) = p_{2,i}(x)$ and $P_3(x) = p_{3,i}(x)$ are obtained from (585).



So, according to (581) and equations (708)–(590), we have

$$\begin{aligned}
C_M^T & \left[(1 + \mathcal{R}_1 \Delta t) \left[P_3(x_l) - \frac{(x_l - \ell)^2}{(1 - \ell)^2} P_3(1) \right] + \mathcal{R}_2 \Delta t \left[P_2(x_l) - \frac{x_l - \ell}{1 - \ell} P_2(1) \right] + \Delta t H_M(x_l) \right] = \\
& 2\alpha u(x_l, t_s) u_x(x_l, t_s) - 6\beta u^2(x_l, t_s) u_x(x_l, t_s) - u'''(x_l, t_s) \\
& - \left[\left[1 - \frac{(x_l - \ell)^2}{(1 - \ell)^2} \right] k_1'(t_{s+1}) - (x_l - \ell) \frac{x_l - 1}{1 - \ell} k_2'(t_{s+1}) + \frac{(x_l - \ell)^2}{(1 - \ell)^2} q'(t_{s+1}) \right] \\
& - \mathcal{R}_1 \left[\left[1 - \frac{(x_l - \ell)^2}{(1 - \ell)^2} \right] [k_1(t_{s+1}) - k_1(t_s)] - (x_l - \ell) \frac{x_l - 1}{1 - \ell} [k_2(t_{s+1}) - k_2(t_s)] \right. \\
& \quad \left. + \frac{(x_l - \ell)^2}{(1 - \ell)^2} [q(t_{s+1}) - q(t_s)] + u(x_l, t_s) \right] \\
& - \mathcal{R}_2 \left[u'(x_l, t_s) + \frac{x_l - \ell}{1 - \ell} [h(t_{s+1}) - h(t_s)] - \frac{x_l - 1}{1 - \ell} [k_2(t_{s+1}) - k_2(t_s)] \right], \tag{591}
\end{aligned}$$

where

$$\mathcal{R}_1 = 2\alpha u_x(x_l, t_s) - 6\beta u(x_l, t_s) u_x(x_l, t_s), \quad \mathcal{R}_2 = 2\alpha u(x_l, t_s) - 3\beta u^2(x_l, t_s).$$

From the equation (591), a system of M linear equations in the M unknown coefficients is obtained. This system can be written in the matrix vector form as follows:

$$\mathcal{A}\mathcal{X} = \mathcal{B}. \tag{592}$$

From (716), the wavelet coefficients can be calculated. Putting the calculated wavelet coefficients within the equation (590), we can obtain

$$\begin{aligned}
u(0, t_{s+1}) = g(t_{s+1}) = \Delta t C_M^T & \left[P_3(0) - \frac{\ell^2}{(1 - \ell)^2} P_3(1) \right] + u(0, t_s) + \left[1 - \frac{\ell^2}{(1 - \ell)^2} \right] [k_1(t_{s+1}) - k_1(t_s)] \\
& - \frac{\ell}{1 - \ell} [k_2(t_{s+1}) - k_2(t_s)] + \frac{\ell^2}{(1 - \ell)^2} [q(t_{s+1}) - q(t_s)],
\end{aligned}$$

$$\begin{aligned}
u(x_l, t_{s+1}) = \Delta t C_M^T & \left[P_3(x_l) - \frac{(x_l - \ell)^2}{(1 - \ell)^2} P_3(1) \right] + \left[1 - \frac{(x_l - \ell)^2}{(1 - \ell)^2} \right] [k_1(t_{s+1}) - k_1(t_s)] \\
& - (x_l - \ell) \frac{x_l - 1}{1 - \ell} [k_2(t_{s+1}) - k_2(t_s)] + \frac{(x_l - \ell)^2}{(1 - \ell)^2} [q(t_{s+1}) - q(t_s)] + u(x_l, t_s),
\end{aligned}$$

for $l = 1, 2, \dots, M$ and $s = 1, 2, \dots, N$.

4 Numerical results

In this section, we are going to demonstrate numerically, some of the results for the unknown boundary condition in the inverse Gardner equation (581)–(584). The purpose of this section is to



illustrate the applicability of the present method described in section 36. As expected the inverse problems is ill-posed and therefore it is necessary to investigate the stability of the present method by giving a test problem. To this end, we take $\alpha = \beta = 1$, $\ell = 0.5$, $t_F = 1$, $\Delta t = 0.01$, and the noisy data=input data+0.001×randn. The error norm L_∞ is calculated to measure the accuracy of the numerical scheme using the following formula:

$$L_\infty = \|u(x, t) - u^*(x, t)\|_\infty = \max_{1 \leq i \leq M} |u(x_i, t) - u^*(x_i, t)|,$$

where, $u^*(x, t)$ is the estimated value of $u(x, t)$.

Example 0.87. We consider the inverse Gardner equation (581)–(584) satisfying,

$$u_t + 2uu_x - 3u^2u_x + u_{xxx} = 0.$$

The exact solution of this problem is, ([5]),

$$u(x, t) = \frac{1}{3} \left[1 - \tanh\left(\frac{1}{3\sqrt{2}}\left(x - \frac{2}{9}t\right)\right) \right], \quad x \in [0, 1], \quad t \in [0, 1],$$

$$u(0, t) = \frac{1}{3} \left[1 + \tanh\left(\frac{\sqrt{2}}{27}t\right) \right], \quad t \in [0, 1].$$

The numerical results of the unknown boundary condition $u(0, t) = g(t)$ are reported in Table 44, for $J = 1, 4$. Furthermore, Table 45 reports the absolute error $u(x, t)$, for different time steps in $J = 1, 2, 3, 4$.

Table 44: The comparison among exact and numerical solutions for $g(t)$.

t	$g(t)$	$J = 1 \ (M = 4)$		$J = 4 \ (M = 32)$	
		$g^*(t)$	$ g(t) - g^*(t) $	$g^*(t)$	$ g(t) - g^*(t) $
0.1	0.335079	0.335079	$5.821170e - 07$	0.335079	$5.697485e - 07$
0.2	0.336825	0.336824	$1.171149e - 06$	0.336824	$1.146304e - 06$
0.3	0.338571	0.338569	$1.767145e - 06$	0.338569	$1.729682e - 06$
0.4	0.340316	0.340314	$2.369855e - 06$	0.340314	$2.319651e - 06$
0.5	0.342061	0.342058	$2.979024e - 06$	0.342058	$2.915963e - 06$
0.6	0.343806	0.343802	$3.594395e - 06$	0.343802	$3.518363e - 06$
0.7	0.345549	0.345545	$4.215701e - 06$	0.345545	$4.126593e - 06$
0.8	0.347293	0.347288	$4.842675e - 06$	0.347288	$4.740390e - 06$
0.9	0.349035	0.349030	$5.475044e - 06$	0.349030	$5.359487e - 06$
1	0.350777	0.350771	$6.112529e - 06$	0.350771	$5.983612e - 06$

5 Conclusion

In this paper, we applied the HWCW to estimate unknown boundary conditions for the inverse Gardner equation (581)–(584). Due to the numerical solutions which are presented in the Tables,



Table 45: The absolute error (L_∞) for different values of t .

t	$J = 1$ ($M = 4$)	$J = 2$ ($M = 8$)	$J = 3$ ($M = 16$)	$J = 4$ ($M = 32$)
0.1	$4.364209e - 11$	$3.073597e - 12$	$1.489231e - 11$	$2.190170e - 11$
0.3	$1.291322e - 10$	$4.176554e - 11$	$5.648260e - 12$	$1.895811e - 11$
0.5	$2.172547e - 10$	$8.116213e - 11$	$4.939937e - 12$	$1.668449e - 11$
0.7	$3.079276e - 10$	$1.178229e - 10$	$1.470157e - 11$	$1.454431e - 11$
1	$4.483989e - 10$	$1.674241e - 10$	$2.814687e - 11$	$1.159794e - 11$
CPU time	0.944683	2.376038	7.071569	24.244557

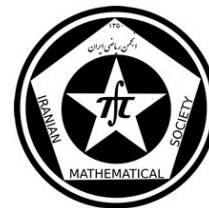
from which we observed that HWCN is the most accurate and stable concerning small perturbation in the input data. Generally, the obtained numerical solutions by the presented method are in good agreement with the exact solutions.

Acknowledgement

The present article was extracted from the thesis written by Naser Azizi and was financially supported by Damghan University.

References

- [1] M.A. Alejo and C. Kwak, *Global solutions and stability properties of the 5th order Gardner equation*, arXiv preprint arXiv:1901.03350, (2019).
- [2] S. Foadian, R. Pourgholi, S. H. Tabasi, and J. Damirchi, *The inverse solution of the coupled nonlinear reaction-diffusion equations by the Haar wavelets*, International Journal of Computer Mathematics, **96** (2019), 105–125.
- [3] M. Kumar and S. Pandit, *A composite numerical scheme for the numerical simulation of coupled burgers' equation*, Computer Physics Communications, **185** (2014), 809–817.
- [4] Ü. Lepik and H. Hein, *Haar wavelets: with applications*, Springer Science & Business Media, (2014).
- [5] A. Patra and S. S. Ray, *Two-dimensional haar wavelet collocation method for the solution of stationary neutron transport equation in a homogeneous isotropic medium*, Annals of Nuclear Energy, **70** (2014), 30–35.
- [6] R. Pourgholi, N. Tavallaie, and S. Foadian, *Applications of Haar basis method for solving some ill-posed inverse problems*, Journal of Mathematical Chemistry, **50** (2012), 2317–2337.
- [7] A.M. Wazwaz, *Exact and explicit travelling wave solutions for the nonlinear Drinfeld–Sokolov system*, Communications in Nonlinear Science and Numerical Simulation, **11** (2006), 311–325.



Numerical Solution of Inverse Time Fractional Telegraph Equation

F. Torabi^{†36}, R. Pourgholi[‡]

Ph. D. student of School of Mathematics and Computer Sciences,

Damghan University, f.torabi@std.du.ac.ir[†]

Academic member of School of Mathematics and Computer Sciences,

Damghan University, pourgholi@du.ac.ir[‡]

Abstract

In this paper, a numerical method is proposed to approximate the solution of the inverse time fractional telegraph equation. We apply extended cubic B-spline basis functions for the determination of an unknown boundary condition in the inverse problem which produce a system. We solve this system by using the Tikhonov regularization method. The aim of this paper is to show that the collocation method based on extended cubic B-spline is also suitable for the treatment of the inverse partial differential equations.

AMS subject Classification 2010: 65M32, 35K05

Keywords: Inverse problems, Time fractional telegraph equation, Extended CubicB-spline

1 Introduction

Inverse problems are encountered in many branches of engineering and science. Mathematically, the inverse problems belong to the class of problems called the ill-posed problems, i.e. small errors in the measured data can lead to large deviations in the estimated quantities. As a consequence, their solutions do not satisfy the general requirement of existence, uniqueness and stability under small changes to the input data [6, 7].

The time fractional telegraph equation (TFTE) has been used into the modeling of reaction dissemination, signal analysis for transference, random walk of suspension ow, propagation of electrical signals etc.

2 Main sections and results

We consider the following inverse one dimensional TFTE[1]

$$\frac{\partial^{2\alpha} u(x, t)}{\partial t^{2\alpha}} + 2\lambda \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + \mu u(x, t) = \nu \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t), \quad (x, t) \in [0, 1] \times [0, t_f], \quad (593)$$

³⁶Speaker



with initial and boundary conditions

$$u(x, 0) = f_1(x), \quad x \in [0, 1], \quad (594)$$

$$u_t(x, 0) = f_2(x), \quad x \in [0, 1], \quad (595)$$

$$u(0, t) = g_1(t), \quad t \in [0, t_f], \quad (596)$$

$$u(1, t) = g_2(t), \quad t \in [0, t_f], \quad (597)$$

where $0 < \alpha < 1$ and λ, μ, ν are arbitrary positive constants. and t_f represents the final time of interest for the time evolution of the problem, while the functions $f_1(x)$, $f_2(x)$, $g_1(t)$ and $f(x, t)$ are known functions, $g_2(t)$ and $u(x, t)$ are unknown.

Definition 0.88. The Caputos time fractional derivative of order α is dened as[3]

$$\frac{\partial^\alpha u(x, t)}{\partial t^\alpha} = \begin{cases} \frac{1}{\Gamma(n - \alpha)} \int_0^t \frac{\partial u(x, t)}{\partial t} \frac{d\xi}{(t - \tau)^{\alpha - n + 1}}, & \alpha \in (n - 1, n], n \in N, \\ \frac{\partial^n u(x, t)}{\partial t^n}, & \alpha = n \in N. \end{cases} \quad (598)$$

where Γ is the Euler Gamma function.

By using (609), we can easily derive the fractional derivative of order 2α .

Suppose $t_n = n\tau, n = 0, 1, \dots, M$ in which $\tau = \frac{t_f}{M}$ is the time step size.

Discretization of the Caputo derivative using a first and second order forward nite dierence method[11] can be written as

$$\frac{\partial^\alpha u(x, t)}{\partial t^\alpha} = \frac{1}{\Gamma(2 - \alpha)} \sum_{j=0}^n b_j^\alpha \frac{u(x, t_{n-j+1}) - u(x, t_{n-j})}{\tau^\alpha} + e_\tau^{n+1}, \quad (599)$$

$$\frac{\partial^{2\alpha} u(x, t)}{\partial t^{2\alpha}} = \frac{1}{\Gamma(3 - 2\alpha)} \sum_{j=0}^n d_j^{2\alpha} \frac{u(x, t_{n-j+1}) - 2u(x, t_{n-j}) + u(x, t_{n-j-1})}{\tau^{2\alpha}} + r_\tau^{n+1}, \quad (600)$$

where $b_j^\alpha = (j + 1)^{1-\alpha} - j^{1-\alpha}$, $d_j^{2\alpha} = (j + 1)^{2-2\alpha} - j^{2-2\alpha}$. The approximation of first and second order truncation error e_τ^{n+1} and r_τ^{n+1} bound are given in [4] as

$$|e_\tau^{n+1}| \leq C\tau^{2-\alpha}, \quad |r_\tau^{n+1}| \leq D\tau^{2-\alpha},$$

where C , D are a constant.

We solve the inverse problem (689)-(597) with the over-specified condition

$$u(a, t) = h_1(t), \quad t \in [0, t_f], \quad (601)$$

where $0 < a < 1$ is a fixed point.

Suppose $u^{n+1} = u(x, t_{n+1})$ and $f^{n+1} = f(x, t_{n+1})$, substituting Eqs. (599) and (600) into (689) thus, we have

$$\begin{aligned} & \beta_1(u^{n+1} - 2u^n + u^{n-1}) + \beta_2(u^{n+1} - u^n) + \beta_1 \sum_{j=1}^n d_j^{2\alpha} (u^{n+1-j} - 2u^{n-j} \\ & + u^{n-1-j}) + \beta_2 \sum_{j=1}^n b_j^\alpha (u^{n+1-j} - u^{n-j}) + \mu u^{n+1} = \nu \frac{\partial^2 u^{n+1}}{\partial x^2} + f^{n+1}, \end{aligned} \quad (602)$$



where $n = 0, 1, \dots, M$, $\beta_1 = \frac{1}{\tau^{2\alpha}\Gamma(3-2\alpha)}$, $\beta_2 = \frac{2\lambda}{\tau^\alpha\Gamma(2-\alpha)}$.

The solution domain $x \in [0, 1]$ is partitioned into a mesh of uniform length $h = x_{i+1} - x_i$ by the knots x_i where $i = 0, 1, \dots, N-1$ such that $0 = x_0 < x_1 < \dots < x_N = 1$ be the partition in $[0, 1]$. The extended cubic B-splines (ExCuBs) basis functions at the nodal point x_i can be presented as[9]

$$B_i(x, \eta) = \frac{1}{24h^4} \begin{cases} 4h(1-\eta)(x-x_{i-2})^3 + 3\eta(x-x_{i-2})^4, & x \in [x_{i-2}, x_{i-1}), \\ (4-\eta)h^4 + 12h^3(x-x_{i-1}) + 6h^2(2+\eta)(x-x_{i-1})^2 \\ -12h(x-x_{i-1})^3 - 3\eta(x-x_{i-1})^4, & x \in [x_{i-1}, x_i), \\ (4-\eta)h^4 + 12h^3(x_{i+1}-x) + 6h^2(2+\eta)(x_{i+1}-x)^2 \\ -12h(x_{i+1}-x)^3 - 3\eta(x_{i+1}-x)^4, & x \in [x_i, x_{i+1}), \\ 4h(1-\eta)(x_{i+2}-x)^3 + 3\eta(x_{i+2}-x)^4, & x \in [x_{i+1}, x_{i+2}), \\ 0, & \text{otherwise.} \end{cases} \quad (603)$$

where $\eta \in R$ is a free parametr and $x \in R$ is a variable. For $-8 \leq \eta \leq 1$, the ExCuBs basis and cubic B-spline possess the same properties. The basis function will be reduced to the basis function of cubic B-spline for $\eta = 0$. The set of functions $\{B_{-1}, B_0, \dots, B_{N+1}\}$ forms a basis over the considered domain on $[0, 1]$. The approximated solution $U(x, t)$ to $u(x, t)$ can be written in terms of the extended B-splines as[10]

$$U(x, t) = \sum_{i=-1}^{N+1} c_i(t) B_i(x, \eta), \quad (604)$$

where $c_i(t)$ are time-dependent quantities to be determined from the boundary and over-specified conditions and collocation from of the differential equations. The values of $B_i(x, \eta)$ and its derivatives may be tabulated as in Table 46.

x	x_{i-1}	x_i	x_{i+1}
$B_i(x, \eta)$	$\frac{4-\eta}{24}$	$\frac{8+\eta}{12}$	$\frac{4-\eta}{24}$
$B'_i(x, \eta)$	$\frac{1}{2h}$	0	$-\frac{1}{2h}$
$B''_i(x, \eta)$	$\frac{2+\eta}{2h^2}$	$-\frac{2+\eta}{h^2}$	$\frac{2+\eta}{2h^2}$

Table 46: Coefficient of extended cubic B-splines and its derivatives at knots x_i .

Using approximate function (604) and ExCuBs (603), the approximate values at the knots of $U(x)$ and its derivatives up to third order are determined in terms of the time parameters c_m as given



below:

$$U_m = \frac{4-\eta}{24}c_{m-1} + \frac{8+\eta}{12}c_m + \frac{4-\eta}{24}c_{m+1}, \quad (605)$$

$$U'_m = -\frac{1}{2h}c_{m-1} + \frac{1}{2h}c_{m+1}, \quad (606)$$

$$U''_m = \frac{2+\eta}{2h^2}c_{m-1} - \frac{2+\eta}{h^2}c_m + \frac{2+\eta}{2h^2}c_{m+1}. \quad (607)$$

The Caputo derivatives and ExCuBs are used to discretize the model problem. Using the approximation and its derivatives in Eq.(602), we have a system which solve by using Tikhonov regularization method, the coefficients c_i are obtained and using these coefficients, we can obtain the approximate solution.

3 Main Results

The purpose of this section is to illustrate the applicability of the present method described in Section 36 for solving the nonlinear inverse problem (689)-(597). We compare the exact and the approximate solutions by considering the total error S defined by[2]

$$S_{g_2} = \left[\frac{1}{N-1} \sum_{z=1}^N \left(g_2(t_z) - g_2^*(t_z) \right)^2 \right]^{\frac{1}{2}}, \quad S_u = \left[\frac{1}{N-1} \sum_{z=1}^N \left(u(t_z) - u^*(t_z) \right)^2 \right]^{\frac{1}{2}},$$

where N is the number of estimated values, g_2 and u are the exact values, g_2^* and u^* are the estimated values.

Example 0.89. In this example we solve the inverse TFTE (689) and the comparisons are made with the exact solutions given in [1], with the noisy data (input data+0.0001×rand(1)).

Consider the time-fractional telegraph equation of order 2α

$$\frac{\partial^{2\alpha} u(x, t)}{\partial t^{2\alpha}} + 40 \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + 100u(x, t) = \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t),$$

$$\begin{aligned} u(x, 0) &= f_1(x) = \sinh(x), & 0 \leq x \leq 1, \\ u(0, t) &= g_1(t) = 0, & 0 \leq t \leq t_f, \\ u_t(x, 0) &= f_2(x) = -2 \sinh(x), & 0 \leq x \leq 1, \\ f(x, t) &= 23e^{-2t} \sinh(x), & 0 \leq x \leq 1, \quad 0 \leq t \leq t_f, \end{aligned}$$

The exact solutions of this problem are,

$$\begin{aligned} u(x, t) &= e^{-2t} \sinh(x), & 0 \leq x \leq 1, \quad 0 \leq t \leq t_f, \\ u(1, t) &= g_2(t) = e^{-2t} \sinh(1) & 0 \leq t \leq t_f. \end{aligned}$$



t	<i>Exact</i>	Extended cubic B-spline	<i>Error</i>
0.1	$9.621734e-01$	$9.573555e-01$	$4.817820e-03$
0.2	$7.877609e-01$	$7.816076e-01$	$6.153296e-03$
0.3	$6.449641e-01$	$6.397938e-01$	$5.170293e-03$
0.4	$5.280519e-01$	$5.246856e-01$	$3.366309e-03$
0.5	$4.323324e-01$	$4.309353e-01$	$1.397016e-03$
0.6	$3.539638e-01$	$3.544056e-01$	$4.417858e-04$
0.7	$2.898010e-01$	$2.918264e-01$	$2.025349e-03$
0.8	$2.372690e-01$	$2.407073e-01$	$3.438222e-03$
0.9	$1.942595e-01$	$1.986835e-01$	$4.424021e-03$
1	$1.590462e-01$	$1.641909e-01$	$5.144705e-03$
<i>S</i>	-	-	0.0039
<i>Condition number</i> 17430839.853114			

Table 47: The values of $g_2(t)$ when $h = 0.1$, $\tau = 0.01$, $a^* = 0.8$ for $\alpha = 0.985$, $\eta = 10^{-5}$.

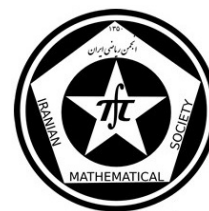
t	<i>Exact</i>	Extended cubic B-spline	<i>Error</i>
0.1	$3.362956e-01$	$3.347080e-01$	$1.587581e-03$
0.2	$2.753355e-01$	$2.738610e-01$	$1.474500e-03$
0.3	$2.254257e-01$	$2.246456e-01$	$7.800062e-04$
0.4	$1.845629e-01$	$1.845964e-01$	$3.352111e-05$
0.5	$1.511073e-01$	$1.518955e-01$	$7.882007e-04$
0.6	$1.237162e-01$	$1.251379e-01$	$1.421673e-03$
0.7	$1.012903e-01$	$1.032090e-01$	$1.918769e-03$
0.8	$8.292946e-02$	$8.521785e-02$	$2.288384e-03$
0.9	$6.789690e-02$	$7.043286e-02$	$2.535959e-03$
1	$5.558928e-02$	$5.828259e-02$	$2.693306e-03$
<i>S</i>	-	-	0.0017

Table 48: The values of $u(0.4, t)$ when $h = 0.1$, $\tau = 0.01$, $a^* = 0.8$ for $\alpha = 0.985$, $\eta = 10^{-5}$.



References

- [1] M.Asgari and R.Ezzati and T.Allahviranloo, *Numerical solution of time-fractional order telegraph equation by Bernstein polynomials operational matrices*, Mathematical Problems in Engineering, 2016.
- [2] J. Cabeza, G. María, J. García, M. Andrés and A. Rodríguez, *A sequential algorithm of inverse heat conduction problems using singular value decomposition*, International Journal of Thermal Sciences, 2005.
- [3] Caputo, Michele, *Elasticita e dissipazione*, Zanichelli, 1969.
- [4] L.Changpin and Z.Zhengang and Ch.YangQuan, *Numerical approximation of nonlinear fractional differential equations with subdiffusion and superdiffusion*, Elsevier, 2011.
- [5] A. Iserles and S.P. Norset, *On quadrature methods for highly oscillatory integrals and their implementation*, BIT, **44** (2004), 755–772.
- [6] Kozhanov, Aleksandr Ivanovich and Safiullova, Regina Rifkhatovna, *Linear inverse problems for parabolic and hyperbolic equations*, Journal of Inverse & Ill-Posed Problems, Berlin, 2010.
- [7] H.Lopushanska, V.Rapita, *Inverse coefficient problem for the semi-linear fractional telegraph equation*, Electronic Journal of Differential Equations, 2014.
- [8] S. Olver, *Theoretical Numerical Analysis*, Springer, Berlin, 2005.
- [9] Shengjun, Han Xuli Liu, *An Extension of the Cubic Uniform B-Spline Curve [J]*, Journal of Computer Aided Design & Computer GraphicsSpringer,2003.
- [10] A.Tayyaba and M.Abbas and I.Ahmad Izani and A.Norhashidah Hj M and Baleanu, Dumitru, *Extended cubic B-splines in the numerical solution of time fractional telegraph equation*, Springer, 2019.
- [11] Wang, Jinfeng and Zhao, Meng and Zhang, Min and Liu, Yang and Li, Hong, *Numerical analysis of an-Galerkin mixed finite element method for time fractional telegraph equation*, The Scientific World Journal, 2014.



Numerical Study of Inverse System of Coupled Burgers Equations by Chebyshev Wavelet Method

A. Janmohmmadi^{†37}, J. Damirchi[†] and S.M. Mahmoudi[‡]

Department of Mathematics, Semnan University, Semnan, Iran.[†]

Department of Statistics, Semnan University, Semnan, Iran. mahmoudi@semnan.ac.ir[‡]

Abstract

This paper is devoted to the numerical solution of an inverse system of coupled Burgers' equations by Chebyshev wavelet in collaboration with the collocation method. We have used the Chebyshev wavelet for the approximation of unknown dependent variables u , v , and their derivative w.r.t space variable x . Taylor's expansion is applied to handle nonlinearity terms. The collocation proposed method results in an ill-posed linear algebraic system. Thus, we use Tikhonov regularization technique with L-curve criteria to obtain a stable solution for this system. Finally, the test problem considered to demonstrate the ability and utility of the proposed scheme. The computed numerical results are in good agreement with the exact solution.

AMS subject Classification 2010: 47A52, 65M70, 35R30, 65T60.

Keywords: Chebyshev wavelet method, Coupled Burgers' equations, Inverse Problem, Regularization Method.

1 Introduction

Coupled Burgers equation was first derived by Esipov [2] which is a simple model of sedimentation or evolution of scaled volume concentrations of two kinds of particles in fluid suspensions or colloids, under the effect of gravity [5]. This equation has been solved by various approaches such as; Khater et al. [4] used the Chebyshev spectral collocation method to solve the equation.

In this paper, we propose a collocation-based numerical scheme to obtain approximate solutions of an inverse system of coupled Burgers equations. The scheme employs collocations of Chebyshev wavelets.

We consider the inverse system of Coupled Burgers equations to determine the unknown boundary conditions $u(0, t)$ and $v(0, t)$ in the following system

$$\begin{cases} u_t(x, t) = u_{xx}(x, t) + \mu_1 u(x, t)u_x(x, t) + \xi_1(u(x, t)v(x, t))_x, & 0 < x < 1, \quad 0 < t < T, \\ v_t(x, t) = v_{xx}(x, t) + \mu_2 v(x, t)v_x(x, t) + \xi_2(u(x, t)v(x, t))_x, & 0 < x < 1, \quad 0 < t < T, \end{cases} \quad (608)$$

³⁷Speaker



with the initial conditions

$$u(x, 0) = f_1(x), \quad v(x, 0) = f_2(x), \quad 0 \leq x \leq 1, \quad (609)$$

and the boundary conditions

$$u(1, t) = g_1(t), \quad v(1, t) = g_2(t), \quad 0 \leq t \leq T, \quad (610)$$

and overspecified data which are given by:

$$u(x_1, t) = k_1(t), \quad v(x_2, t) = k_2(t), \quad 0 \leq x_1, x_2 \leq 1, \quad 0 \leq t \leq T, \quad (611)$$

where $f_1(x), f_2(x), g_1(x), g_2(x), k_1(t), k_2(t)$ are considered as known continuous functions, T is the final time, and $\mu_1, \mu_2, \xi_1, \xi_2$ are real constants depending on the system parameters from physical point of view. The boundary conditions $u(0, t)$ and $v(0, t)$ are unknown functions and to be estimated from overspecified data.

2 A Review of the Chebyshev wavelet

Wavelets constitute a family of functions constructed from dilation and translation of a signal function called the mother wavelet. Chebyshev wavelet $\psi_{n,m}(x) = \psi(k, n, m, x)$ is defined on the interval $[-1, 1)$ as follows [6]:

$$\psi_{nm}(x) = \begin{cases} \gamma_m \frac{2^{\frac{(k-1)}{2}}}{\sqrt{\pi}} T_m(2^k x - 2n + 1), & \frac{n-1}{2^{k-1}} \leq x < \frac{n}{2^{k-1}}, \\ 0, & \text{elsewhere,} \end{cases} \quad (612)$$

where

$$\gamma_m = \begin{cases} \sqrt{2}, & m = 0, \\ 2, & m = 1, 2, \dots, \end{cases} \quad (613)$$

and $n = 1, 2, \dots, 2^{k-1}$, $k \in \mathbb{N}$, $m = 0, 1, 2, \dots, M-1$, and m is the degree of Chebyshev polynomials of first kind $T_m(x)$. It should be noted that any function $u(x) \in L^2[0, 1)$ can be expanded into Chebyshev wavelets as follows:

$$u(x) = \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} \psi_{nm}(x). \quad (614)$$

We denote the first and second integral of Eq. (612) by $p_{nm} = \int_0^x \psi_{nm}(s) ds$ and $q_{nm} = \int_0^x p_{nm}(s) ds$. More details about the calculation of the first and second integral of Chebyshev wavelets can be found in [1]. We will use these integrals in solution approach in next section.



3 Implementation of the method

Let us divide the interval $[0, T]$ into N equals parts of length $\Delta t = \frac{T}{N}$ and denote $t_n = (n-1)\Delta t$, $n = 1, 2, \dots, N$. Now, we expand the derivatives $u_{xx}(x, t_{n+1})$ and $v_{xx}(x, t_{n+1})$ in the Eqs. (689) in terms of Chebyshev wavelets. base on porposed method. We discretize only $u(x, t)$ below, same procedure can be applied to $v(x, t)$. We expand $u_{xx}(x, t_{n+1})$ as:

$$u_{xx}(x, t_{n+1}) = (u_{xx})_{n+1} = \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} \psi_{nm}(x). \quad (615)$$

Integrating Eq. (615) with respect to x from 0 to x , we have following equation

$$(u_x)_{n+1} = (u_x)_{n+1}(0) + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} p_{nm}(x). \quad (616)$$

Integrating Eq. (616) with respect to x from x_1 to 1, and by using boundary conditions (610) and overspecified conditions (611) at discrete point t_{n+1} , we obtain

$$u_{n+1}(1) - u_{n+1}(x_1) = (1 - x_1)(u_x)_{n+1}(0) + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} (q_{nm}(1) - q_{nm}(x_1)). \quad (617)$$

By using boundary conditions (610) and overspecified conditions (611) at discrete point t_{n+1} , we can get

$$g_1(t_{n+1}) - k_1(t_{n+1}) = (1 - x_1)(u_x)_{n+1}(0) + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M c_{nm} (q_{nm}(1) - q_{nm}(x_1)),$$

and so

$$(u_x)_{n+1}(0) = \frac{1}{1 - x_1} \left[g_1(t_{n+1}) - k_1(t_{n+1}) - \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} (q_{nm}(1) - q_{nm}(x_1)) \right]. \quad (618)$$

Substituting Eq. (618) into Eq. (616), yields

$$(u_x)_{n+1} = \frac{1}{1 - x_1} \left[g_1(t_{n+1}) - k_1(t_{n+1}) - \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M c_{nm} (q_{nm}(1) - q_{nm}(x_1)) \right] + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} p_{nm}(x). \quad (619)$$

By integrating Eq. (619) with respect to x from 0 to x , we have following equation

$$u_{n+1}(x) = u_{n+1}(0) + \frac{x}{1 - x_1} \left[g_1(t_{n+1}) - k_1(t_{n+1}) - \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} (q_{nm}(1) - q_{nm}(x_1)) \right] + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} q_{nm}(x), \quad (620)$$

by calculating $u_{n+1}(1)$ and $u_{n+1}(0)$ and known boundary conditions from Eq. (620), we obtain

$$\begin{aligned} u_{n+1}(x) &= g_1(t_{n+1}) + \frac{x-1}{1-x_1} \left[g_1(t_{n+1}) - k_1(t_{n+1}) \right] - \frac{x-1}{1-x_1} \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} (q_{nm}(1) - q_{nm}(x_1)) \\ &\quad + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M c_{nm} (q_{nm}(x) - q_{nm}(1)), \end{aligned} \quad (621)$$



and finally

$$u_{n+1}(x) = g_1(t_{n+1}) + \frac{x-1}{1-x_1} \left[g_1(t_{n+1}) - k_1(t_{n+1}) \right] + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^M c_{nm} \left[q_{nm}(x) + \frac{x_1-x}{1-x_1} q_{nm}(1) + \frac{x-1}{1-x_1} q_{nm}(x_1) \right]. \quad (622)$$

The values of $(u_{xx})_{n+1}$, $(u_x)_{n+1}$ and $u_{n+1}(x)$ can be calculated by Eqs. (615), (619) and (622). The same procedure was followed to obtain these equations for $v(x, t)$ and its derivative at t_{n+1} . Now, we apply finite difference approximation for Eq (689) at grid point (x, t_{n+1}) in the following form. It should be noted that we discretize only dependent variable $u(x, t)$ for simplicity and same procedure can be applied. We consider the following finite difference scheme for Eq. (689) for numerical approximation of $u(x, t)$

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{1}{2} [(u_{xx})_{n+1} + (u_{xx})_n] + \frac{\mu_1}{2} [(uu_x)_{n+1} + (uu_x)_n] + \frac{\xi_1}{2} [(u_x v + uv_x)_{n+1} + (u_x v + uv_x)_n]. \quad (623)$$

By using of Taylor's expansion and linearization of nonlinear terms we have the following equations

$$\begin{aligned} (uu_x)_{n+1} &\cong u_{n+1}(u_x)_n + u_n(u_x)_{n+1} - u_n(u_x)_n, \\ (u_x v)_{n+1} &\cong (u_x)_{n+1}v_n + (u_x)_n v_{n+1} - (u_x)_n v_n, \\ (uv_x)_{n+1} &\cong u_{n+1}(v_x)_n + u_n(v_x)_{n+1} - u_n(v_x)_n. \end{aligned}$$

By using Eq. (623) and approximations of $(uu_x)_{n+1}$, $(u_x v)_{n+1}$ and $(uv_x)_{n+1}$, we obtain

$$A_n u_{n+1} - B_n (u_x)_{n+1} - \frac{1}{2} (u_{xx})_{n+1} - C_n v_{n+1} - D_n (v_x)_{n+1} = \frac{u_n}{\Delta t} + \frac{1}{2} (u_{xx})_n, \quad (624)$$

where

$$A_n = \frac{1}{\Delta t} - \frac{\mu_1}{2} (u_x)_n - \frac{\xi_1}{2} (v_x)_n, \quad B_n = \frac{1}{2} (\mu_1 u_n + \xi_1 v_n) \quad C_n = \frac{\xi_1}{2} (u_x)_n, \quad D_n = \frac{\xi_1}{2} u_n.$$

Similarly for the dependent variable $v(x, t)$, we have

$$A_n^* v_{n+1} - B_n^* (v_x)_{n+1} - \frac{1}{2} (v_{xx})_{n+1} - C_n^* u_{n+1} - D_n^* (u_x)_{n+1} = \frac{v_n}{\Delta t} + \frac{1}{2} (v_{xx})_n, \quad (625)$$

where

$$A_n^* = \frac{1}{\Delta t} - \frac{\mu_2}{2} (v_x)_n - \frac{\xi_2}{2} (u_x)_n, \quad B_n^* = \frac{1}{2} (\mu_2 v_n + \xi_2 u_n), \quad C_n^* = \frac{\xi_2}{2} (v_x)_n, \quad D_n^* = \frac{\xi_2}{2} v_n.$$



Substituting Eqs. (615), (619) and (622) into Eq. (624), yields

$$\begin{aligned}
& \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} c_{nm} \left[\left(A_n \frac{x_1 - x}{1 - x_1} + \frac{B_n}{1 - x_1} \right) q_{nm}(1) + \left(A_n \frac{x - 1}{1 - x_1} - \frac{B_n}{1 - x_1} \right) q_{nm}(x_1) \right. \\
& \quad \left. + A_n q_{nm}(x) - B_n p_{nm}(x) - \frac{1}{2} \psi_{nm}(x) \right] \\
& + \sum_{n=1}^{2^{k-1}} \sum_{m=0}^{M-1} d_{nm} \left[\left(-C_n \frac{x_2 - x}{1 - x_2} + \frac{D_n}{1 - x_2} \right) q_{nm}(1) + \left(-C_n \frac{x - 1}{1 - x_2} - \frac{D_n}{1 - x_2} \right) q_{nm}(x_2) - C_n q_{nm}(x) - D_n p_{nm}(x) \right] \\
& = -A_n \left[(g_1)_{n+1} + \frac{x-1}{1-x_1} ((g_1)_{n+1} - (k_1)_{n+1}) \right] + \frac{B_n}{1-x_1} ((g_1)_{n+1} - (k_1)_{n+1}) \\
& + C_n \left[(g_2)_{n+1} + \frac{x-1}{1-x_2} ((g_2)_{n+1} - (k_2)_{n+1}) \right] + \frac{D_n}{1-x_2} ((g_2)_{n+1} - (k_2)_{n+1}) + \frac{u_n}{\Delta t} + \frac{1}{2} (u_{xx})_n. \tag{626}
\end{aligned}$$

In a similar way, we can obtain these equations for $v(x, t)$.

We collocate x at $x_i = \frac{i-0.5}{2^{k-1}M}$, $i = 1, 2, \dots, 2^{k-1}M$ collocation points in Eq. (626) and similar equation for $v(x, t)$. The linear system of equations in matrix form $AX = \mathbf{b}$ is derived for the unknown elements of $c_{nm}, d_{nm}, n = 1, 2, \dots, 2^{k-1}, m = 0, 1, \dots, M$. Since the resulting matrix is ill-conditioned so applying the regularization technique is necessary. We will apply Tikhonov regularization with the L-curve criterion to obtain stable solution for unknow coefficients [3]. The unknown functions $u(x, t)$ and $v(x, t)$ can be obtained by finding these coefficients.

4 Numerical Experiments and Results

This section included an examples to demonstrate the ability and validity of the proposed numerical approach. The stability of the proposed method for test example is investigated when some perturbed approximation of Eq. (611) is in hand at point t_i by adding noises $\epsilon_{i_1}, \epsilon_{i_2}$ to exact data as follows:

$$u_\epsilon(x_1, t_i) = k_1(t_i) + \epsilon_{i_1}, \quad v_\epsilon(x_2, t_i) = k_2(t_i) + \epsilon_{i_2}, \quad i_1, i_2 = 0, 1, \dots, N,$$

where $\epsilon_{i_1}, \epsilon_{i_2}$ are random variables which are generated from a Gaussian normal distribution with zero mean and standard deviation σ_1, σ_2 given by

$$\sigma_1 = \delta \times \max_{t \in [0,1]} |u(x_1, t)|, \quad \sigma_2 = \delta \times \max_{t \in [0,1]} |v(x_2, t)|,$$

where δ is represent the percentage of noise. The root mean square error (RMSE) is defined by:

$$RMSE(u(0, t)) = \sqrt{\frac{1}{N} \sum_{i=1}^N [u_{exact}(0, t_i) - u_{approx}(0, t_i)]^2}, \tag{627}$$

to measure the values of error in our computations.

Consider the following inverse problem (689)-(611) with input data

$$\begin{aligned}
\mu_1 &= \mu_1 = +2, \quad \xi_1 = \xi_2 = -1, \quad f_1(x) = f_2(x) = \sin(\sqrt{\lambda}(x - 0.5)), \quad \lambda \in \mathbb{R}, \\
g_1(t) &= g_2(t) = e^{-\lambda t} \sin(\sqrt{\lambda}/2), \quad u\left(\frac{1}{4}, t\right) = k_1(t) = -e^{-\lambda t} \sin(\sqrt{\lambda}/4), \quad v\left(\frac{1}{2}, t\right) = k_2(t) = 0,
\end{aligned}$$



Table 49: The comparison between the results of exact and numerical solution of $u(0, t) = v(0, t)$ for different values of noise level with $K = 3, M = 4, N = 5$.

t	Exact solution	Numerical solution $\delta=0\%$	Numerical solution $\delta=1\%$	Numerical solution $\delta=3\%$	Numerical solution $\delta=5\%$
0.0	-0.34623	-0.34623	-0.346230	-0.34623	-3.0000
0.2	-0.31329	-0.31471	-0.31360	-0.31137	-0.32954
0.4	-0.28347	-0.28341	-0.280588	-0.27491	-0.23602
0.6	-0.25650	-0.25763	-0.26534	-0.28079	-0.29358
0.8	-0.23209	-0.23217	-0.2229	-0.20430	-0.20050
1.	-0.21000	-0.21072	-0.21797	-0.23248	-0.22709
$RMSE$	—	$6.4169e - 07$	$3.9075e - 05$	$3.2404e - 04$	$8.6355e - 04$

and $T = 1$. The exact solutions of the inverse problem (689)-(611) are as follow:

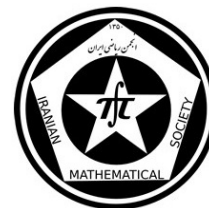
$$u(x, t) = v(x, t) = e^{-\lambda t} \sin(\sqrt{\lambda}(x - 0.5)).$$

The numerical results for comparison between the exact values of $u_{exact}(0, t)$ given in this example and the computed solution of $u_{approx}(0, t)$ for different values of δ with their $RMSE$ are shown in Table 54.

References

- [1] I. Celik, *Chebyshev wavelet collocation method for solving generalized Burgers-Huxley equation*, Mathematical methods in the applied sciences, **39**(3) (2016), 366–377.
- [2] S. E. Esipov, *Coupled Burgers equations: a model of polydisperse sedimentation*, Physical Review E, **52**(4) (1995), 3711.
- [3] P.C. Hansen, *Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems*, Numerical algorithms, **6**(1) (1994), pp. 1–35.
- [4] A. H. Khater, R. S. Temsah, and M. M. Hassan. *A Chebyshev spectral collocation method for solving Burgers-type equations*, Journal of Computational and Applied Mathematics **222**(2) (2008), 333–350.
- [5] J. Nee, and J. Duan, *Limit set of trajectories of the coupled viscous Burgers' equations*, Applied mathematics letters, **11**(1), (1998) 57–61.
- [6] P. Wojtaszczyk, *A mathematical introduction to wavelets*, Cambridge University Press, **37**, 1997.

Email: janmohammadiali@semnan.ac.ir damirchi@semnan.ac.ir mahmoudi@semnan.ac.ir



Numerical methods for Ordinary differential equations based on second derivative collocation

S. Fazeli ^{†38}, M. Zeinali[‡]

Marand Faculty of Engineering, University of Tabriz, Tabriz, Iran and fazeli@tabrizu.ac.ir[†]

Faculty of Mathematical Sciences, University of Tabriz, Tabriz, Iran and zeinali@tabrizu.ac.ir[‡]

Abstract

In this paper, we introduce second derivative multistep collocation methods for the numerical integration of ordinary differential equations to obtain methods with desirable stability properties of higher order. In these methods, the solution of the problem in each step depends on the numerical solution in some points in the r previous steps. The construction of technique, analysis of the order of accuracy and linear stability properties of the methods are described.

AMS subject Classification 2010: 65L05 .

Keywords: Collocation, Linear stability, Ordinary differential equation, Second derivative methods.

1 Introduction

The purpose of this paper is to discuss the construction of highly stable multistep collocation methods by using the second derivative of the solution for the numerical solution of system of ordinary differential equations (ODEs) in the form

$$\begin{cases} y' = f(y(t)), & t \in [t_0, T], \\ y(t_0) = y_0, \end{cases} \quad (628)$$

where the function $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is assumed to be sufficiently smooth, $y_0 \in \mathbb{R}^d$ is a given initial value and d is the dimensionality of the system.

The idea of multistep collocation methods was first introduced by Lie and Nørsett in [7]. These methods which depend on more parameters than classical ones, while not imposing more computational cost, have high convergence order and strong stability properties.

In the literature many authors (see [1] and references therein contained) have analyzed one-step collocation methods for ODEs. As it is well known, a collocation method is based on the idea of

³⁸Speaker



approximating the exact solution of a given integral equation with a suitable function belonging to a chosen finite dimensional space, usually a piecewise algebraic polynomial which satisfies the integral equation exactly on a certain subset of the integration interval (called the set of collocation points).

In this paper we propose a new extension to the multistep case by using the second derivatives of the solution in the collocation points. More precisely here we derive a special case of multistep collocation methods, which depend on a fixed number r of previous time steps and first and second derivative of the solution in m collocation points, with the aim of increasing the order of multistep collocation methods without increasing the computational cost. In analogy to the case of Ordinary Differential Equations (ODEs) [2, 3, 4]. Using more of the derivative of the approximated solution has been successfully applied to construct methods with higher order and extensive region of stability for numerical solution of ODEs [5], specially for stiff problems.

2 Construction of method

Let us discretize the interval $I := [t_0, T]$ by introducing a uniform mesh h in the form

$$I_h = \{t_0 + nh \mid n = 0, 1, \dots, N, \ h > 0, \ Nh = T - t_0\},$$

also consider the abscissa vector $c = [c_1, c_2, \dots, c_m]^T$ and define the collocation parameters by $t_{n,i} := t_n + c_i h$. The two-step continuous approximation to the solution of (628) is defined by

$$\begin{cases} P(t_n + sh) = \sum_{k=0}^{r-1} \varphi_k(s) y_{n-k} + h \sum_{j=1}^m \left(\psi_j(s) f(P(t_{n,j})) \right) + h^2 \sum_{j=1}^m \left(\chi_j(s) g(P(t_{n,j})) \right), \\ y_{n+1} = P(t_{n+1}), \end{cases} \quad (629)$$

with $s \in (0, 1]$ and $g(\cdot) = f'(\cdot)f(\cdot)$. Also, we assume that the approximate solution in the initial interval $[t_0, t_{r-1}]$ is computed by a sufficiently high order method. The functions $\varphi_k(s), \psi_j(s), \chi_j(s)$, $j = 1, \dots, m$, $k = 0, 1, \dots, r-1$, are polynomials of degree $2m + r - 1$ which define the method. The interpolation conditions for $P(t_n + sh)$ at the points t_{n-i} , lead to

$$P(t_{n-i}) = y_{n-i}, \quad i = 0, 1, \dots, r-1, \quad (630)$$

and the collocation conditions for the equation (628) and its derivative are written in the form

$$P'(t_{n,i}) = f(P(t_{n,i})), \quad P''(t_{n,i}) = g(P(t_{n,i})), \quad i = 1, 2, \dots, m. \quad (631)$$

The conditions (630) and (631) determine the polynomials which satisfy

$$\varphi_k(-i) = \delta_{ik}, \quad \psi_j(-i) = 0, \quad \chi_j(-i) = 0$$

and

$$\begin{aligned} \varphi'_k(c_i) &= 0, & \psi'_j(c_i) &= \delta_{ij}, & \chi'_j(c_i) &= 0, \\ \varphi''_k(c_i) &= 0, & \psi''_j(c_i) &= 0, & \chi''_j(c_i) &= \delta_{ij}, \end{aligned}$$



where δ_{ij} , $i, j = 1, 2, \dots, m$, is the usual Kronecker delta. The construction of these polynomials are obtained by Hermite–Birkhoff interpolation [8], also by means of poisoning condition [6], it can be shown that these interpolation problems have the unique solution.

3 Continuous order conditions

In this section, we investigate the continuous order conditions for the method (36) under the assumption that the collocation polynomial $P(t_n + sh)$ provides a uniform approximation of order p to $y(t_n + sh)$, $s \in [0, 1]$. By replacing $P(t_n + sh)$ by $y(t_n + sh)$ in (36), where $y(t)$ is the exact solution of (628), and subtracting the both sides of the obtained relation, we get the discretization error

$$\varepsilon(t_n + sh) = y(t_n + sh) - \sum_{k=0}^{r-1} \varphi_k(s)y(t_{n-k}) - h \sum_{j=1}^m \psi_j(s)f(y(t_{n,j})) - h^2 \sum_{j=1}^m \chi_j(s)g(y(t_{n,j})), \quad (632)$$

where $s \in [0, 1]$ and $n = 1, 2, \dots, N - 1$. By expanding local discretization error in Taylor series around the point t_n and collecting terms with the same powers of h we have the following theorem:

Theorem 0.90. *Assume that the function $f(y)$ is sufficiently smooth. Then the method (36) has uniform order p if the following conditions are satisfied*

$$\begin{cases} \sum_{k=0}^{r-1} \varphi_k(s) = 1, \\ \sum_{k=0}^{r-1} k\varphi_k(s) + \sum_{j=1}^m \psi_j(s) = s, \\ \sum_{k=0}^{r-1} \frac{(-k)^i}{i!} \varphi_k(s) + \sum_{j=1}^m \left(\psi_j(s) \frac{c_j^{i-1}}{(i-1)!} + \chi_j(s) \frac{c_j^{i-2}}{(i-2)!} \right) = \frac{s^i}{i!}, \end{cases} \quad (633)$$

where $s \in [0, 1]$, $i = 2, 3, \dots, p$.

The set of order conditions (633) constitutes a linear system of $p + 1$ equations in $2m + r$ unknown basic functions. As a consequence, in order to ensure the compatibility of the system (633), p can be at most equal to $2m + r - 1$. This leads to the following result.

Remark 0.91. The maximum attainable uniform order of convergence for new method (36) is $2m + r - 1$.

We next derive the conditions for zero-stability of the new method. For this purpose, we apply the method to the simple ODE $y' = 0$, where leads to the recurrence relation

$$y_{n+1} = \sum_{k=0}^{r-1} \theta_k y_{n-k},$$

where $\theta_k = \varphi_k(1)$.



4 Linear stability analysis

In this section, we investigate the stability behavior of the SDTCMs (36) with respect to the standard test equation of Dahlquist

$$y' = \lambda y, \quad (634)$$

where λ is a complex number with negative real part. Let us define

$$\begin{aligned} A &= [\psi_j(c_i)]_{i,j=1}^m, & \bar{A} &= [\chi_j(c_i)]_{i,j=1}^m, \\ v^T &= [\psi_i(1)]^T, & w^T &= [\chi_i(1)]^T, \\ (\varphi(c))_{j,i} &= [\varphi_j(c_i)], & j &= 0, 1, \dots, r-1, \quad i = 1, 2, \dots, m \\ P(t_n + ch) &= [P(t_n + c_i h)]_{i,j=1}^m. \end{aligned} \quad (635)$$

Theorem 0.92. *The new method (36), applied to the test equation (634), leads to the following recurrence relation*

$$\begin{bmatrix} y_{n+1} \\ y_n^{(r)} \end{bmatrix} = \begin{bmatrix} M_{11}(z) & 0 \\ \mathbf{I}_{r-1} & \mathbf{O}_{r-1,1} \end{bmatrix} \begin{bmatrix} y_n \\ y_{n-1}^{(r)} \end{bmatrix}, \quad (636)$$

where $z := \lambda h$ and

$$M_{11}(z) = (\varphi(1) + zv^T + z^2w^T) Q^{-1}\varphi(c),$$

$$Q = \mathbf{I}_m - zA - z^2\bar{A}.$$

We find that there are some A-stable Methods within this class of methods, for example in the case, $r = 2, m = 1$ when the collocation parameter c lies in the interval $[0.578, 1]$, the obtained method is A-stable of order 3. Also in the case $r = 2, m = 2$ by choosing $c_1 \in [0.214, 0.558]$ and $c_2 = 1$, we obtained A-stable method of order 5.

5 Numerical Experiments

In this section, we present numerical results showing the efficiency and accuracy of the new method, and validating the order of these methods in the integration of stiff systems. In what follows, we describe details of the implemented methods:

- **Method 1:** A-stable second derivative multistep collocation method of convergence order 3 with $r = 2, m = 1$, and collocation parameter $c = 1$.



- **Method 2:** A-stable second derivative multistep collocation method of convergence order 5 with $r = 2$, $m = 2$, and collocation parameters $c_1 = \frac{5}{10}$, $c_2 = 1$.

Also, we compared the results with the method in [4, 5], by implementation of the method

- **Method 3:** Second derivative two-step collocation method with $m = 1$, $p = 4m = 4$, $c = \frac{2}{3}$ and $q = 1$, introduced in [5].
- **Method 4:** Two-step almost collocation method with $m = 2$, $p = 2m = 4$, $c = [\frac{1}{2}, \frac{9}{10}]$, introduced in [4].

Computational experiments are done by applying the methods to the following problem:

P1. The nonlinear stiff ODE

$$\begin{cases} y_1'(t) = -10004y_1(t) + 10000y_2(t)^2, & y_1(0) = 1, \\ y_2'(t) = y_1(t) - y_2(t)(1 + y_2(t)^3), & y_2(0) = 1, \end{cases}$$

with $t \in [0, 1]$. The exact solution is $[y_1(t) \ y_2(t)]^T = [e^{-4t} \ e^{-t}]^T$.

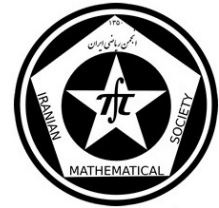
In our numerical experiments, we apply Method 1 and Method 2, to the given test problems with the fixed stepsizes $h = T/N$ for various values of N . The global error of the methods at the endpoint of the interval of integration is indicated by Err . Also, to verify theoretical results on the order of accuracy, we compute a numerical estimate to the order of accuracy of the methods by the formula $\log_2(\|Err_h(x)\|/\|Err_{h/2}(x)\|)$, where $Err_h(x)$ and $Err_{h/2}(x)$ are the errors at the endpoint of the interval of integration corresponding to the stepsizes h and $h/2$, respectively. Also by comparison results by methods introduced in [4, 5], one can see that the results are competitable by method 3 and the solution by this new methods are more precise than results of method 4 while the computational cost for the new method is less.

Table 50: The results of the methods for problem **P1**.

N		4	8	16	32	64
Method 1	Err	2.22×10^{-4}	3.40×10^{-5}	4.64×10^{-6}	6.04×10^{-7}	7.71×10^{-8}
	$p(h)$		2.71	2.87	2.94	2.97
Method 2	Err	2.12×10^{-7}	8.38×10^{-9}	2.93×10^{-10}	9.66×10^{-12}	3.10×10^{-13}
	$p(h)$		4.66	4.84	4.92	4.96
Method 3	Err	1.17×10^{-3}	6.14×10^{-5}	2.58×10^{-6}	1.01×10^{-7}	5.96×10^{-9}
	$p(h)$		4.26	4.57	4.67	4.59
Method 4	Err	1.51×10^{-3}	1.74×10^{-4}	1.49×10^{-5}	1.08×10^{-6}	7.08×10^{-8}
	$p(h)$		3.11	3.54	3.79	3.93

References

- [1] H. Brunner, *Collocation methods for Volterra integral and related functional equations*, Cambridge University Press, 2004.
- [2] J.C. Butcher, G. Hojjati, *Second derivative methods with RK stability*, Numer. Algorithms 40 (2005), 415–429.
- [3] J.R. Cash, *Second derivative extended backward differentiation formulas for the numerical integration of stiff systems*, SIAM J. Numer. Anal. **18** (1981), 21–36.
- [4] R. D'Ambrosio, M. Ferro, Z. Jackiewicz, B. Paternoster, *Two-step almost collocation methods for ordinary differential equations*, Numer. Algorithms, **53** (2010), 195–217.
- [5] S. Fazeli, G. Hojjati, *Second derivative two-step collocation methods for ordinary differential equations*, Appl. Numer. Math., **156** (2020), 514–527.
- [6] Ferguson, D. *Some interpolation theorems for polynomials*, J. Approx. Theory, **9** (1973), 327–348.
- [7] I. Lie, S.P. Nørsett, *Superconvergence for multistep collocation*, Math. Comp. **52** (1989), 65–79.
- [8] I.J. Schoenberg, *On Hermite-Birkhoff interpolation*, J. Math. Anal. Appl., **16** (1966), 538–543.



Nonstandard finite difference- θ scheme for stochastic partial differential equations

M. Namjoo^{1,3,4,5}, M. J. Mohammad Taghizadeh², M. Aminian³, M. Karami⁴, A. Mohebbian⁵

^{1,3,4,5}Department of Mathematics, Vali-e-Asr University of Rafsanjan, Rafsanjan, Iran

²Atherosclerosis Research Center, Ahvaz Jundishapur University of Medical Sciences, Ahvaz, Iran

namjoo@vru.ac.ir, dr.taghizadeh87@gmail.com, mehran.aminian@vru.ac.ir

m.karami@vru.ac.ir, a.mohebbian@stu.vru.ac.ir

Abstract

In this paper, numerical solutions of Itô stochastic partial differential equations (SPDEs) are presented based on the nonstandard finite difference θ (NSFD- θ) scheme. The consistency, stability and convergence of the NSFD- θ is discussed. The performance of the proposed NSFD- θ scheme is tested for SPDE and the accuracy and efficiency of the numerical scheme is demonstrated.

AMS subject Classification 2010: 60M15, 65M12.

Keywords: Nonstandard finite difference θ scheme, Stochastic partial differential equations, Consistency, Stability, Convergence.

1 Introduction

Although stochastic partial differential equation has a long history, its applications to mathematical finance, social sciences, biology and medical are a recent focus of interest. The description of some phenomena is more accurate when the SPDE is used. Since, in many situation, such SPDE is cannot be solved analytically, it is important to obtain their numerical solutions by using some numerical schemes. Whenever continuous models have been converted into discrete models, the properties of continuous models are not transferred completely to the discrete model in the case of large stepsize in the discrete models. However, if we use NSFD scheme, the properties of the continuous model can be preserved into its discrete model. In this paper, a new NSFD- θ scheme determined for solving SPDE as follows.

$$u_t(x, t) = \gamma u_{xx}(x, t) + \gamma u(x, t) \dot{W}(t), \quad x \in [0, 1], \quad t \in [0, 1]. \quad (637)$$

Subject to the initial and boundry conditions $u(x, 0) = u_0(x)$, $u(0, t) = f_1(t)$ and $u(1, t) = f_2(t)$, where the parameters γ and σ are positive constants. Here $W(t)$ is Wiener process such that the white noise $\dot{W}(t)$ is a Gaussian distribution with zero mean [1].

³⁹Speaker



2 NSFD- θ scheme for SPDE

Nonstandard finite difference schemes provide numerical solutions to ordinary and partial differential equations by constructing discrete models. They preserve the significant properties of their continuous analogues and consequently give reliable numerical results. NSFD schemes depend on the two rules. The first order derivative is replaced by a discrete form

$$\left. \frac{dx}{dt} \right|_{t=t_k} \approx \frac{x_{k+1} - \varphi_1(h)x_k}{\varphi_2(h)},$$

where

$$\varphi_1(h) = 1 + O(h), \quad \varphi_2(h) = h + O(h^2).$$

Considering a uniform space grid Δx and time grid Δt in the time-space lattice, we able to approximate the solution of the SPDE at the nodal points of this lattice. The value of the numerical solution at the nodal point $(x_k, t_n) = (k\Delta x, n\Delta t)$ will be denoted by u_k^n . In order to construct a NSFD scheme, we will require linear and nonlinear terms must, in general, be modeled by the nonlocal discrete representation on the lattice [3]. For example, linear and nonlinear terms can be considered as

$$u_k^n \approx 2u_{k+1}^n - u_k^n, \quad (u_k^n)^2 \approx u_k^n u_{k+1}^n, \quad (u_k^n)^3 \approx \frac{1}{2}(u_k^n)^2 (3u_{k+1}^n - u_k^n).$$

In order to construct a NSFD scheme, the time and space derivatives in the SPDE (637) and also the function $u(x, t)$ in stochastic part can be approximated as follows.

$$u_t(k\Delta x, n\Delta t) \approx \frac{u_k^{n+1} - u_k^n}{\Delta t}, \quad u_{xx}(k\Delta x, n\Delta t) \approx \frac{u_{k-1}^n - 2u_k^n + u_{k+1}^n}{\Delta x^2}, \quad u_k^n \approx (1 - \theta)u_{k+1}^n + \theta u_k^n.$$

Therefore, the NSFD scheme of SPDE (637) is given by

$$u_{k+1}^n = ru_{k-1}^n + (1 - 2r)u_k^n + ru_{k+1}^n + \sigma [(1 - \theta)u_{k+1}^n + \theta u_k^n] \Delta W_n, \quad (638)$$

where $r = \gamma \frac{\Delta t}{\Delta x^2}$, $\Delta W_n = W((n+1)\Delta t) - W(n\Delta t) \sim N(0, \Delta t)$ and $\theta \in [0, 1]$. We call the scheme (638) a NSFD- θ scheme. In the remainder of this paper we assume the increments of Wiener process are independent of the state u_k^n .

3 Consistency and stability analysis of stochastic NSFD- θ scheme

In numerical methods for SPDEs, consistency and stability are desirable properties of a stochastic scheme. Consider a SPDE of the form $Lv = G$ where L denotes the differential operator and G is source term. Let $L_k^n u_k^n = G_k^n$ denote the NSFD- θ scheme, where G_k^n is the approximation of the source term. In order to analysis consistency and stability the NSFD- θ scheme (638), it is necessary to consider a norm. Hence, for a sequence $\{u_k^n\}_{0 \leq k \leq M}$, we define $\|u^n\| = \sqrt{\sup_{0 \leq k \leq M} |u_k^n|^2}$. For the consistency, stability and convergence definitions of a stochastic NSFD scheme, we refer the reader to [2].



Definition 0.93. A NSFD scheme $L_k^n u_k^n = G_k^n$ is consistent with the SPDE $Lv = G$ at point (x, t) , if for any continuously differentiable function $\Phi(x, t)$ in the sense of mean square we have $\mathbb{E}\|u^{n+1} - v^{n+1}\| \rightarrow 0$, for $t = (n+1)\Delta t$ and $\Delta t, \Delta x \rightarrow 0$, where the vectors u^{n+1} and v^{n+1} are defined by $u^{n+1} = (u_0^{n+1}, u_1^{n+1}, \dots, u_M^{n+1})$ and $v^{n+1} = (v_0^{n+1}, v_1^{n+1}, \dots, v_M^{n+1})$.

Theorem 0.94. If $0 \leq r \leq \frac{1}{2}$ and $0 \leq \theta \leq 1$, then the NSFD- θ scheme (638) is consistent in the sense of mean square.

Proof. Let $\Phi(x, t)$ be a smooth function, hence we have

$$\begin{aligned} L(\Phi)|_k^n &= \Phi(k\Delta x, (n+1)\Delta t) - \Phi(k\Delta x, n\Delta t) \\ &\quad - \gamma \int_{n\Delta t}^{(n+1)\Delta t} \Phi_{xx}(k\Delta x, s) ds - \sigma \int_{n\Delta t}^{(n+1)\Delta t} \Phi(k\Delta x, s) dW(s), \end{aligned}$$

and

$$\begin{aligned} L_k^n \Phi &= \Phi(k\Delta x, (n+1)\Delta t) - \Phi(k\Delta x, n\Delta t) \\ &\quad - \gamma \frac{\Delta t}{\Delta x^2} \left(\Phi((k+1)\Delta x, n\Delta t) - 2\Phi(k\Delta x, n\Delta t) + \Phi((k-1)\Delta x, n\Delta t) \right) \\ &\quad - \sigma \left(\theta \Phi(k\Delta x, n\Delta t) + (1-\theta)\Phi((k+1)\Delta x, n\Delta t) \right) \Delta W_n. \end{aligned}$$

Now, if we use the square property of Itô integral, we will obtain

$$\begin{aligned} &\mathbb{E}|L(\Phi)|_k^n - L_k^n \Phi|^2 \\ &\leq 2\gamma^2 \mathbb{E} \left| \int_{n\Delta t}^{(n+1)\Delta t} \left(\Phi_{xx}(k\Delta x, s) \right. \right. \\ &\quad \left. \left. - \frac{1}{\Delta x^2} \left(\Phi((k+1)\Delta x, n\Delta t) - 2\Phi(k\Delta x, n\Delta t) + \Phi((k-1)\Delta x, n\Delta t) \right) \right) ds \right|^2 \\ &\quad + 2\sigma^2 \int_{n\Delta t}^{(n+1)\Delta t} \mathbb{E} \left| \Phi(k\Delta x, s) - (\theta \Phi(k\Delta x, n\Delta t) + (1-\theta)\Phi((k+1)\Delta x, n\Delta t)) \right|^2 ds. \end{aligned}$$

Since $\Phi(x, t)$ is a deterministic function, hence $\mathbb{E}|L(\Phi)|_k^n - L_k^n \Phi|^2 \rightarrow 0$, as $n, k \rightarrow \infty$. $\square$

Definition 0.95. A stochastic NSFD scheme is said to be stable in the sense of mean square, if there exist some positive constants $\overline{\Delta x_0}$ and $\overline{\Delta t_0}$ and nonnegative constants K and β such that $\mathbb{E}\|u^{n+1}\|^2 \leq K e^{\beta t} \mathbb{E}\|u^0\|^2$, for all $t = (n+1)\Delta t$, $0 \leq \Delta x \leq \overline{\Delta x_0}$, and $0 \leq \Delta t \leq \overline{\Delta t_0}$.

Theorem 0.96. The NSFD- θ scheme (638) with $t = (n+1)\Delta t$ and $0 \leq r \leq \frac{1}{2}$ is stable with respect to sup-norm.

Proof. Applying $\mathbb{E}|\cdot|^2$ in (638) and using the independence of the Wiener process increments, we obtain

$$\begin{aligned} \mathbb{E}|u_k^{n+1}|^2 &= \mathbb{E} \left| ru_{k-1}^n + (1-2r)u_k^n + ru_{k+1}^n \right|^2 + \sigma^2 \mathbb{E} \left| (1-\theta)u_{k+1}^n + \theta u_k^n \right|^2 \Delta t \\ &\leq (|r| + |1-2r| + |r|)^2 \sup_k \mathbb{E}|u_k^n|^2 + \sigma^2 (|1-\theta| + |\theta|)^2 \sup_k \mathbb{E}|u_k^n|^2 = (1 + \sigma^2 \Delta t) \sup_k \mathbb{E}|u_k^n|^2. \end{aligned}$$



Consequently,

$$\sup_k \mathbb{E}|u_k^{n+1}|^2 \leq (1 + \sigma^2 \Delta t) \sup_k \mathbb{E}|u_k^n|^2.$$

Since $\Delta t = \frac{t}{n+1}$, it follows that

$$\mathbb{E}\|u^{n+1}\|_\infty^2 \leq (1 + \frac{\sigma^2 t}{n+1})^{n+1} \mathbb{E}\|u^0\|_\infty^2 \leq e^{\sigma^2 t} \mathbb{E}\|u^0\|_\infty^2.$$

So, the NSFD- θ scheme (638) is stable, with $K = 1$ and $\beta = \sigma^2$. $\square$

Definition 0.97. A stochastic difference scheme $L_k^n u_k^n = G_k^n$ approximating the SPDE $Lv = G$ is convergent in the sense of mean square at time t , if $\mathbb{E}\|u^{n+1} - v^{n+1}\|^2 \rightarrow 0$, for $t = (n+1)\Delta t$, $\Delta x \rightarrow 0$ and $\Delta t \rightarrow 0$.

Theorem 0.98. The NSFD- θ scheme (638) for SPDE (637) is convergent in the sense of mean square with respect to sup-norm.

Proof. From the stochastic Lax-Richtmyer Theorem we concluded that the NSFD- θ scheme is convergent. $\square$

3 Numerical results

Analytical studies always remain incomplete without numerical verification of the results. In this section, numerical results from the implementation of the NSFD- θ scheme (638) are presented. Also, the stability and convergence of the proposed NSFD- θ scheme is numerically investigated.

Example 0.99. Consider the SPDE as follows

$$u_t(x, t) = \gamma u_{xx}(x, t) + \sigma u(x, t) \dot{W}(t), \quad x \in [0, 1], \quad t \in [0, 1], \quad (639)$$

subject to the following initial condition

$$u(x, 0) = \exp\left(-\frac{(x - 0.2)^2}{\gamma}\right), \quad x \in [0, 1],$$

with the boundary conditions

$$\begin{aligned} u(0, t) &= \frac{1}{\sqrt{4t+1}} \exp\left(-\frac{0.04}{\gamma(4t+1)}\right), \quad t \in [0, 1], \\ u(1, t) &= \frac{1}{\sqrt{4t+1}} \exp\left(-\frac{0.64}{\gamma(4t+1)}\right), \quad t \in [0, 1]. \end{aligned}$$

It is easy to verify that in the absence of the stochastic term, the exact solution is

$$u(x, t) = \frac{1}{\sqrt{4t+1}} \exp\left(-\frac{(x - 0.2)^2}{\gamma(4t+1)}\right).$$



In Figure 36 the approximated solution of the SPDE (639) using the NSFD- θ scheme is represented for $\gamma = 0.01$, $\sigma = 2$, $\theta = \frac{2}{3}$, $\Delta x = 0.01$, $\Delta t = 0.001$ and also $\gamma = 0.001$, $\sigma = 1$, $\theta = \frac{3}{4}$, $\Delta x = 0.01$, $\Delta t = 0.02$.

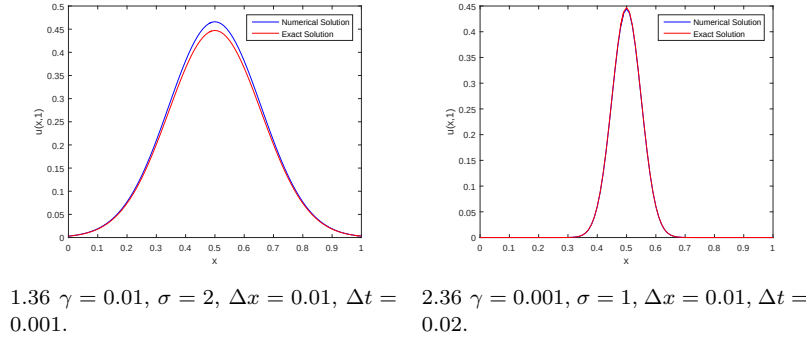


Figure 36: Comparison between deterministic and stochastic numerical solution of (639) using the proposed NSFD- θ scheme with $\gamma = 0.01$, $\sigma = 2$, $\theta = \frac{2}{3}$, $\Delta x = 0.01$ and $\Delta t = 0.001$ and with $\gamma = 0.001$, $\sigma = 1$, $\theta = \frac{3}{4}$, $\Delta x = 0.01$ and $\Delta t = 0.02$.

Let M and N be the total number of grid points for the space and time discretizations, respectively. The NSFD- θ scheme is conditionally stable for $0 \leq r \leq \frac{1}{2}$ and $0 \leq \theta \leq 1$. Therefore, if $M = 100$, and $\theta = \frac{2}{3}$ then for the stability condition, we must have $N \geq 100$. In Figure 37 stability of the NSFD- θ scheme at the end of time interval $t = 1$, for the fixed space grid points $M = 100$ and various time grid points $N = 50, 80, 100, 110$ is considered. Note that the NSFD- θ scheme is unstable for $N = 50$.

4 Conclusion

This paper has provided a class of stochastic NSFD schemes for the numerical solutions of stochastic partial differential equations. Consistency and stability of this class of schemes is established. Convergence conditions of the proposed NSFD scheme are given. Some numerical results are included to show the efficiency of the NSFD- θ scheme.

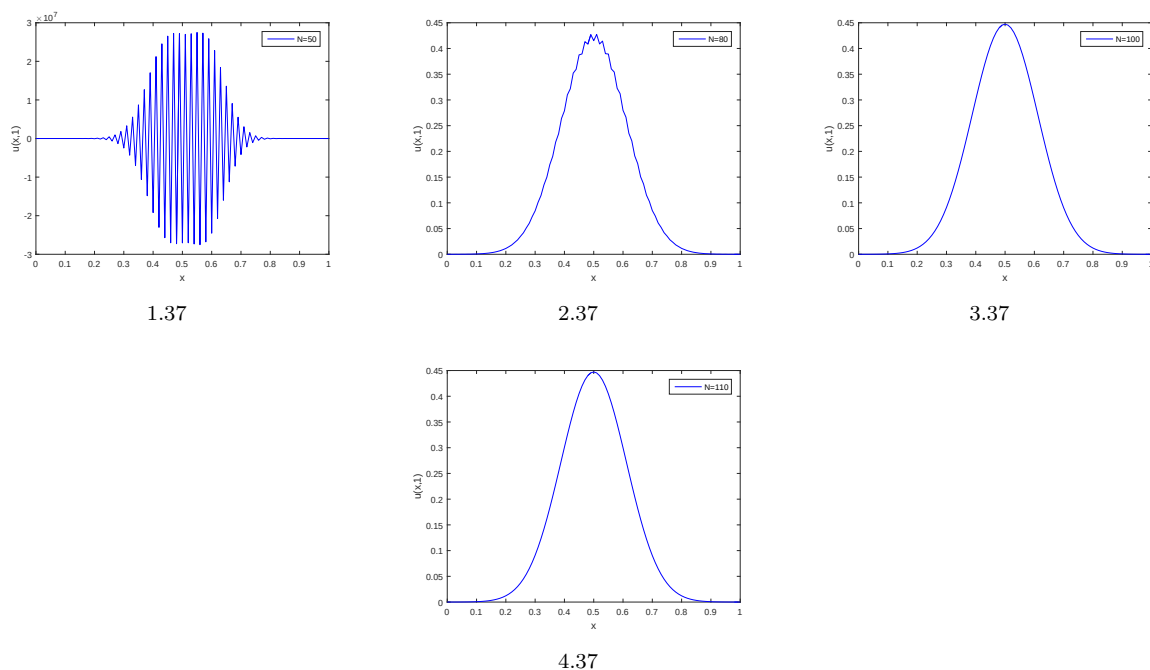
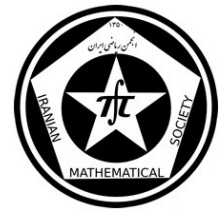


Figure 37: Representation of conditional convergence, $u(x, 1)$ for different values of N .

References

- [1] P. E. Kloeden, E. Platen, *Numerical solution of stochastic differential equations*, Springer, Berlin, 1995.
- [2] C. Roth, *Difference methods for stochastic partial differential equations*, Z. Angew. Math. Mech., 82 (2002), 821–830.
- [3] M. Namjoo, S. Zibaei, *A NSFD Scheme for the Solving Fractional-Order Competitive Prey–Predator System*, Thai J. Math., 18 (2020), 1933–1945.



Nonhomogeneous linear second order fully fuzzy differential equation under MMCE-product

Masoumeh Zeinali^{†40}, Somayeh Fazeli[‡]

Faculty of Mathematical Sciences, University of Tabriz, Tabriz, Iran and E-mail:zeinali@tabrizu.ac.ir[†]

Marand Technical College, University of Tabriz, Tabriz, Iran and E-mail:fazeli@tabrizu.ac.ir[‡]

Abstract

In this paper, the nonhomogeneous form of a linear second order differential equation with fuzzy constant coefficients and fuzzy initial conditions has been solved. The fuzzy derivative is assumed to be of the generalized Hukuhara type. Due to the fuzzy coefficients, the most complexity of this problem pertains to fuzzy multiplication. Considering the Zadeh extension based product, the resulted differential equation will be very complicated due to dependency of Zadeh extension-based product on the signs of the under product fuzzy numbers. In the present work, we use the modified form of the recently introduced MCE-product.

45JXX, 94DXX.

Keywords: MCE-product, Fully fuzzy differential equation, Second order differential equation.

1 Introduction

In this paper, we study the following fuzzy initial value problem

$$\begin{cases} y'' + p \circledast y' + q \circledast y = f(x), \\ y(0) = y_0, y'(0) = y'_0 \end{cases} \quad (640)$$

where $p, q, y_0, y'_0 \in \mathbb{R}_\tau$. The term fully fuzzy refers to a case in which all coefficients and initial values are fuzzy. If the coefficients are real, there is various works have been done on this issue [2, 3, 4]. In fully fuzzy cases, finding a solution to this equation is rare. It is due to the complexity of fuzzy multiplication. In this paper, using the recently introduced MCE-product, we propose a solution to this problem.

⁴⁰Speaker



2 Preliminaries

The spaces of fuzzy numbers and triangular fuzzy numbers are denoted by $\mathbb{R}_{\mathcal{F}}$ and $\mathbb{R}_{\tau}$, respectively.

Definition 0.100. Suppose v is a fuzzy number with the r -cut $v_r = [v_r^-, v_r^+]$. Let $m_v := \frac{v_-^1 + v_+^1}{2}$, the functions $\theta_v^-, \theta_v^+ : [0, 1] \rightarrow \mathbb{R}^+$ defined by

$$\theta_v^- := m_v - v_-^r, \quad \theta_v^+ := v_+^r - m_v$$

are bounded, decreasing, left-continuous on $(0, 1]$ and continuous at 0. The representation $v = (m_v; \theta_v^-, \theta_v^+)$ is called MCE-representation of v .

Definition 0.101. Let $v = (m_v; \theta_v^-, \theta_v^+)$ and $u = (m_u; \theta_u^-, \theta_u^+)$. Then $u \odot v$ defined by

$$u \odot v = (m_u m_v; \theta_u^- \theta_v^-, \theta_u^+ \theta_v^+)$$

is called MCE-product of u and v .

For the triangular fuzzy number $u = (a, b, c)$, the MCE-representation is of the form $u = (b; (b - a)(1 - r), (c - b)(1 - r))$. Therefore, it can be claimed that the MCE-representation of a triangular fuzzy number is always as follows

$$u = (m_u; k_u^-(1 - r), k_u^+(1 - r))$$

in which $k_v^-, k_v^+ \in \mathbb{R}^+$.

Definition 0.102. Let $v = (m_v; k_v^-(1 - r), k_v^+(1 - r))$ and $u = (m_u; k_u^-(1 - r), k_u^+(1 - r))$. Then

$$u \otimes v = (m_u m_v; k_u^- k_v^-(1 - r), k_u^+ k_v^+(1 - r)).$$

is called modified MCE-product of u and v .

The modified MCE-product preserves the shapes of fuzzy numbers, that is, the product of triangular fuzzy numbers is triangular. Also, it is distributive and independent of the sign involved fuzzy numbers.

Theorem 0.103. Let $g : \mathbb{R} \rightarrow \mathbb{R}_{\tau}$.

- (i) If g is (i)-GH-differentiable, then $g'(x) = (m'_g(x); (k_g^-)'(x)(1 - r), (k_g^+)'(x)(1 - r))$.
- (ii) If g is (ii)-GH-differentiable, then $g'(x) = (m'_g(x); -(k_g^+)'(x)(1 - r), -(k_g^-)'(x)(1 - r))$.

Theorem 0.104. Let $g : (a, b) \rightarrow \mathbb{R}_{\tau}$, $g(x) = (m_g(x), k_g^-(x)(1 - r), k_g^+(x)(1 - r))$ and $m_g, k_g^-, k_g^+ \in C^1(a, b)$. If $(k_g^-(x))'(k_g^+(x))' > 0$, then g is GH-differentiable and

$$f'(x) = ((m_g(x))', \max\{(k_g^-(x))', -(k_g^+(x))'\}(1 - r), \max\{(k_g^+(x))', -(k_g^-(x))'\}(1 - r)).$$



3 Main Results

Definition 0.105. We say that $y : \mathbb{R} \rightarrow \mathbb{R}_{\mathcal{F}}$ is a solution for problem (640) if it is GH-differentiable and satisfies problem (640) for all $x \in \mathbb{R}$. According to the type of GH-differentiability, four types of solution can be considered:

- A fuzzy solution y is called **(i)-(i)-solution**, if y and y' are (i)-GH-differentiable.
- A fuzzy solution y is called **(ii)-(ii)-solution**, if y and y' are (ii)-GH-differentiable.
- A fuzzy solution y is called **(ii)-(i)-solution**, if y is (ii)-GH-differentiable and y' is (i)-GH-differentiable.
- A fuzzy solution y is called **(i)-(ii)-solution**, if y is (i)-GH-differentiable and y' is (ii)-GH-differentiable.

Let

$$\begin{aligned} y(x) &= (m_y(x); k_y^-(x)(1-r), k_y^+(x)(1-r)), \\ p(x) &= (m_p(x); k_p^-(x)(1-r), k_p^+(x)(1-r)), \\ q(x) &= (m_q(x); k_q^-(x)(1-r), k_q^+(x)(1-r)), \\ f(x) &= (m_f(x); k_f^-(x)(1-r), k_f^+(x)(1-r)). \end{aligned}$$

(i)-(i)-solution: We consider y and y' are (i)-GH-differentiable. According to Theorem 0.110 we have

$$\begin{aligned} y' &= (m'_y; (k_y^-)'(1-r), (k_y^+)'(1-r)) \\ y'' &= (m''_y; (k_y^-)''(1-r), (k_y^+)''(1-r)). \end{aligned}$$

From the definition of MCE-product, we obtain

$$\begin{aligned} & (m''_y; (k_y^-)''(1-r), (k_y^+)''(1-r)) + (m_p; k_p^-(1-r), k_p^+(1-r)) \odot (m'_y; (k_y^-)'(1-r), (k_y^+)'(1-r)) \\ & + (m_q; k_q^-(1-r), k_q^+(1-r)) \odot (m_y; k_y^-(1-r), k_y^+(1-r)) = (m_f(x); k_f^-(x)(1-r), k_f^+(x)(1-r)) \\ & \Rightarrow \\ & \left(m''_y + m'_y m_p + m_y m_q; ((k_y^-)'' + k_p^- (k_y^-)' + k_q^- k_y^-)(1-r), ((k_y^+)'' + k_p^+ (k_y^+)' + k_q^+ k_y^+)(1-r) \right) \\ & = (m_f(x); k_f^-(x)(1-r), k_f^+(x)(1-r)) \end{aligned}$$

Thus, we have the following three crisp initial value problems

$$m''_y + m'_y m_p + m_y m_q = m_f \quad , \quad m_y(0) = y_0 \quad , \quad m'_y(0) = y'_0 \quad (641)$$

$$(k_y^-)'' + k_p^- (k_y^-)' + k_q^- k_y^- = k_f^- \quad , \quad k_y^-(0) = k_0^- \quad , \quad (k_y^-)'(0) = (k^-)'_0 \quad (642)$$



$$(k_y^+)'' + k_p^+(k_y^+) + k_q^+k_y^+ = k_f^+ \quad , \quad k_y^+(0) = k_0^+ \quad , \quad (k_y^+)'(0) = (k^+)'_0 \quad (643)$$

The above equations are second-order differential equations, and they can be easily solved. From Theorem 0.104, if $k_y^-, k_y^+ \geq 0$ and $(k_y^-)', (k_y^+) > 0$, then

$$y(x) = (m_y(x); k_y^-(x)(1-r), k_y^+(x)(1-r))$$

is the (i)-(i)-solution of the problem (640). Otherwise, it doesn't have (i)-(i)-solution.

(ii)-(i)-solution: In a similar way, the corresponding system of differential equations for (i)-(ii)-solution is

$$m_y'' + m_y' m_p + m_y m_q = m_f \quad , \quad m_y(0) = y_0 \quad , \quad m_y'(0) = y'_0 \quad (644)$$

$$-(k_y^+)'' - k_p^-(k_y^+) + k_q^-k_y^- = k_f^- \quad , \quad k_y^-(0) = k_0^- \quad , \quad (k_y^-)'(0) = (k^-)'_0 \quad (645)$$

$$-(k_y^-)'' - k_p^+(k_y^-) + k_q^+k_y^+ = k_f^+ \quad , \quad k_y^+(0) = k_0^+ \quad , \quad (k_y^+)'(0) = (k^+)'_0 \quad (646)$$

The equation (644) is second-order differential equations, and it is easily solved.

From the equation (645), we have

$$\begin{aligned} -(k_y^+)'' - k_p^-(k_y^+) + k_q^-k_y^- = k_f^- &\Rightarrow k_y^- = \frac{1}{k_q^-} ((k_f^-)' + (k_y^+)'' + k_p^-(k_y^+)') \Rightarrow \\ (k_y^-)' = \frac{1}{k_q^-} ((k_f^-)' + (k_y^+)''' + k_p^-(k_y^+)') &\Rightarrow (k_y^-)'' = \frac{1}{k_q^-} ((k_f^-)'' + (k_y^+)^{(4)} + k_p^-(k_y^+)''') \end{aligned}$$

and substantive $(k_y^-)'$ and $(k_y^-)''$, in (646), we have:

$$\begin{aligned} -\frac{1}{k_q^-} ((k_f^-)'' + (k_y^+)^{(4)} + k_p^-(k_y^+)''') - k_p^+ \frac{1}{k_q^-} ((k_f^-)' + (k_y^+)''' + k_p^-(k_y^+)') + k_q^+k_y^+ &= 0 \\ -\frac{1}{k_q^-} (k_y^+)^{(4)} + \left(\frac{-k_p^- - k_p^+}{k_q^-}\right)(k_y^+)''' - \frac{k_p^+k_p^-}{k_q^-}(k_y^+)'' + k_q^+k_y^+ &= \frac{1}{k_q^-} ((k_f^-)'' + k_p^+(k_f^-)') \end{aligned}$$

First we obtain the solution of the above equation using the methods available in the classical differential equations and then by substituting k_y^+ into the equation (646), this equation becomes a classical second differential equation with constant coefficients. By solving this equation, k_y^- is obtained. If k_y^- and k_y^+ are positive, then $y(x) = (m_y(x); k_y^-(x)(1-r), k_y^+(x)(1-r))$ is the (ii)-(i)-solution of the problem (640). Otherwise, it doesn't have (ii)-(i)-solution.

(i)-(ii)-solution: The corresponding system of differential equations for (i)-(ii)-solution is

$$m_y'' + m_y' m_p + m_y m_q = m_f, \quad m_y(0) = y_0, \quad m_y'(0) = y'_0, \quad (647)$$

$$-(k_y^+)'' + k_p^-(k_y^-)' + k_q^-k_y^- = k_f^-, \quad k_y^-(0) = k_0^-, \quad (k_y^-)'(0) = (k^-)'_0. \quad (648)$$

$$-(k_y^-)'' + k_p^+(k_y^+) + k_q^+k_y^+ = k_f^+, \quad k_y^+(0) = k_0^+, \quad (k_y^+)'(0) = (k^+)'_0 \quad (649)$$



The equation (647) can be easily solved. From the equation (648) and considering $u = \int \int k_y^-$, we have

$$\begin{aligned}
 -(k_y^+)' + k_p^-(k_y^-)' + k_q^-k_y^- &= 0 \Rightarrow (k_y^+)' = k_p^-(k_y^-)' + k_q^-k_y^- \\
 \Rightarrow (k_y^+)' &= k_p^- \int (k_y^-)' + k_q^- \int k_y^- \\
 \Rightarrow k_y^+ &= k_p^- \int (k_y^-)' + k_q^- \int \int k_y^- + cx \\
 \Rightarrow (k_y^-)' &= \frac{1}{k_q^-} ((k_y^+)' - k_p^-(k_y^-)') \\
 \Rightarrow (k_y^-)'' &= \frac{1}{k_q^-} ((k_y^+)'') + k_p^-(k_y^-)''
 \end{aligned}$$

and substantive $(k_y^-)'$ and $(k_y^-)''$, in (646), we have:

If k_y^- and k_y^+ are positive, then $y(x) = (m_y(x); k_y^-(x)(1-r), k_y^+(x)(1-r))$ is the (i)-(i)-solution of the problem (640). Otherwise, it doesn't have (i)-(ii)-solution.

(ii)-(ii)-solution: The corresponding system of (ii)-(ii)-solution is

$$m_y'' + m_y' m_p + m_y m_q = m_f, \quad m_y(0) = y_0, \quad m_y'(0) = y_0' \quad (650)$$

$$(k_y^-)'' - k_p^-(k_y^+) + k_q^-k_y^- = k_f^-, \quad k_y^-(0) = k_0^-, \quad (k_y^-)'(0) = (k^-)'_0 \quad (651)$$

$$(k_y^+)'' - k_p^+(k_y^-)' + k_q^+k_y^+ = k_f^+, \quad k_y^+(0) = k_0^+, \quad (k_y^+)'(0) = (k^+)'_0. \quad (652)$$

The equation (650) is second-order differential equations, and it is easily solved according to the above.

If k_y^- and k_y^+ are positive, then $y(x) = (m_y(x); k_y^-(x)(1-r), k_y^+(x)(1-r))$ is the (i)-(i)-solution of the problem (640). Otherwise, it doesn't have (i)-(i)-solution.

Example 0.106. Consider the following second order differential equation with the fuzzy coefficient and fuzzy initial conditions.

$$\begin{cases} y'' + p \otimes y' = (1; 2(1-r), 1-r), \\ y(0) = (1; 1-r, 1-r) \\ y'(0) = (1; 1-r, 1-r) \end{cases} \quad (653)$$

where $p = (2; 3(1-r), 1-r)$. In Figure 38, the solution of this problem is plotted.

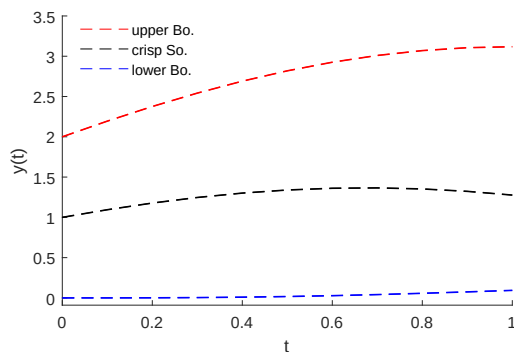
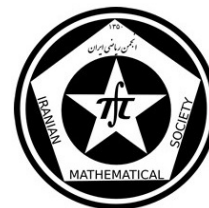


Figure 38: The (i)-(i)-solution of Example 0.106.

References

- [1] A. M. Bica, D. Fechete, and I. Fechete, *Towards the properties of fuzzy multiplication for fuzzy numbers.*, Kybernetika, 55 (2019), 44-62.
- [2] N. Gasilov, S. EmrahAmrahov, A. Fatullayev, *Solution of lineardifferential equations with fuzzy boundary values*, Fuzzy Sets and Systems 257 (2014) 169-183.
- [3] S.P. Mondal, and T.K. Roy, *Solution of second order linear differential equation in fuzzy environment*, Annals of Fuzzy Mathematics and Informatics, 10 (2015), 1-20.
- [4] R. Rodrguez-Lpez, *On the existence of solutions to periodic boundary value problems for fuzzy linear differential equations*, Fuzzy Sets and Systems 219 (2013) 1-26.



On a Be'zier-Like Model with Shape Parameter

Bahareh Nouri, Jamshid Saeidian⁴¹

Faculty of Mathematical Sciences and Computer, Kharazmi University, Tehran, Iran.

Abstract

A class of basis functions, equipped with a shape parameter, is presented. Employing the introduced basis functions, the corresponding curve is defined based on some control points. It is verified that the new curve has most of the features of the classic Bzier curve. The shape parameter helps to adjust the shape of the curve while the control points are fixed.

AMS subject Classification 2010: 65D17, 65D18.

Keywords: basis functions, Bzier curve, shape adjustment.

1 Introduction

As an important geometric modeling tool, Bzier curve has been widely used in Computer Aided Geometric Design (CAGD), Computer Graphics (CG) and Computer Aided Design (CAD). For a set of control points, a Bzier curve is defined based on Bernstein polynomials [1], The curve passes through the two end points and the whole curve lies in the polygon generated by control points. Once the control points are fixed, the shape of the classic Bzier curve cannot be changed. In order to overcome this deficiency of the classic Bzier curve, many researchers have tried to add shape parameters to the basis functions to create new curves whose structures are similar to the Bzier curve ([2, 3, 4, 5]). These new curves have many basic properties with the Bzier curve, but also have the flexible shape adjustable property.

Here, we propose a new parameter based Bzier-like curve which is in common with the classic Bzier curves in many features such as geometric invariance, endpoint interpolation, formal symmetry and convex hull property. Moreover, employing the shape parameter, one can adjust the shape of the curve.

⁴¹Speaker



2 Basis functions

Definition 0.107. The following three functions of t are called the ν -starting basis functions with the shape parameter ν

$$\begin{aligned} b_{2,0}(t) &= \frac{1}{2} - t + \varphi(t) \\ b_{2,1}(t) &= 1 - 2\varphi(t). \\ b_{2,2}(t) &= t - \frac{1}{2} + \varphi(t) \end{aligned} \quad (654)$$

where $0 \leq t \leq 1, 0 < \nu \leq 1$ and

$$\varphi(t) = \sqrt{(1-\nu)(t^2-t) + \frac{1}{4}}$$

We apply the recursion relation of the classical Bernstein basis functions to generate the ν - basis functions of order n ($n \geq 3$).

$$b_{n,i}(t) = (1-t)b_{n-1,i}(t) + tb_{n-1,i-1}(t), \quad (655)$$

where $0 \leq t \leq 1, 0 < \nu \leq 1, i = 0, 1, 2, \dots, n$. In the case $k = -1$ or $k > l$, we set $b_{l,k}(t) = 0$. Figure 42 shows the ν - basis functions generated with $n = 3, 4, 5, 6$ and value of ν is set for $\nu = 0.1, 0.2, \dots, 1$.

Theorem 0.108. The ν -basis functions defined in (654) and (655) have the following properties,

- (a) *Non-negativity:* $b_{n,i}(t) \geq 0$ ($i = 0, 1, 2, \dots, n$).
- (b) *Normalization:* $\sum_{i=0}^n b_{n,i}(t) = 1$.
- (c) *Symmetry:* $b_{n,i}(t) = b_{n,n-i}(1-t)$ ($i = 0, 1, 2, \dots, n$).
- (d) *Properties at the endpoints:*

$$b_{n,i}(0) = \begin{cases} 1 & i = 0; \\ 0, & i \neq 0. \end{cases} \quad b_{n,i}(1) = \begin{cases} 1 & i = n; \\ 0, & i \neq n. \end{cases} \quad (656)$$

$$b'_{n,i}(0) = \begin{cases} -n + \nu & i = 0; \\ n - 2\nu & i = 1; \\ \nu & i = 2; \\ 0 & O.W. \end{cases} \quad b'_{n,i}(1) = \begin{cases} n - \nu & i = n; \\ 2\nu - n & i = n - 1; \\ -\nu & i = n - 2; \\ 0 & O.W. \end{cases} \quad (657)$$

Proof. Mathematical induction is used to prove this theorem

- (a) Non-negativity is obvious by (654) and (655).

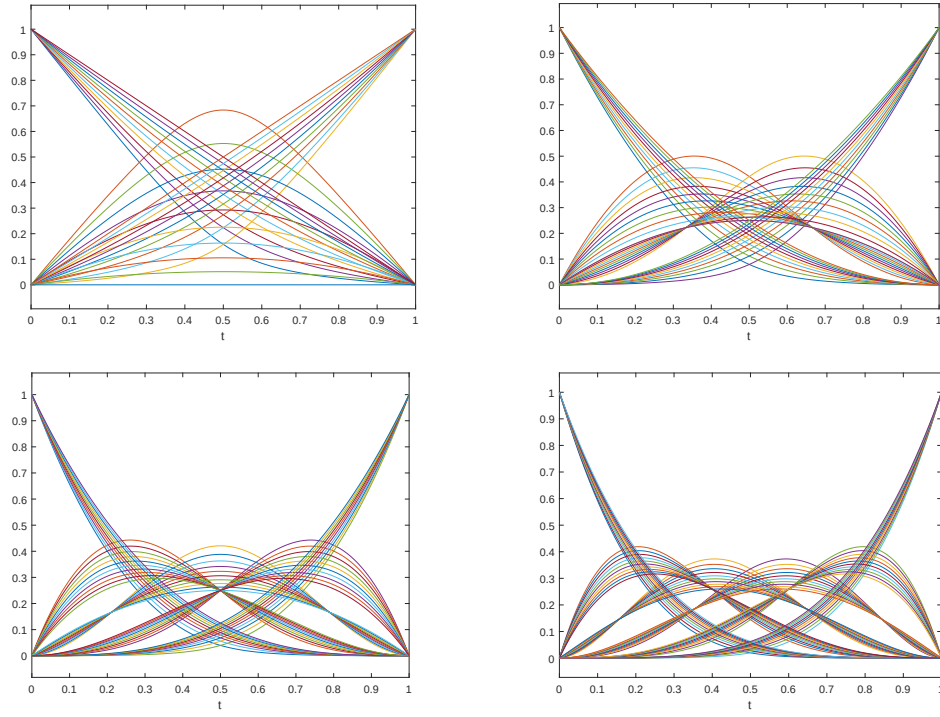


Figure 39: the ν -basis functions with different values of $\nu = 0.1, 0.2, \dots, 1$

- (b) From (654), we obtain $\sum_{i=0}^2 b_{2,i}(t) = 1$. Now assume that the equality holds for m , then according to formula (655) and the inductive hypothesis, we have

$$\sum_{i=0}^{m+1} b_{m+1,i}(t) = (1-t) \sum_{i=0}^{m+1} b_{m,i}(t) + t \sum_{i=0}^{m+1} b_{m,i-1}(t) = 1-t+t = 1.$$

- (c) The ν -starting basis functions are symmetrical. Suppose that the ν -basis functions of order m are symmetrical, then from this inductive hypothesis and recursion formula (655), one has

$$\begin{aligned} b_{m+1,i}(1-t) &= (1-(1-t))b_{m,i}(1-t) + (1-t)b_{m,i-1}(1-t) \\ &= tb_{m,m-i}(t) + (1-t)b_{m,m-i+1}(t) = b_{m+1,m-i+1}(t) \end{aligned}$$

- (d) From (654), we have

$$\begin{aligned} b'_{2,0}(t) &= -1 + \varphi'(t) \\ b'_{2,1}(t) &= -2\varphi'(t). \\ b'_{2,2}(t) &= 1 + \varphi'(t) \end{aligned} \tag{658}$$



where

$$\varphi'(t) = \frac{(1-\nu)(2t-1)}{2\sqrt{(1-\nu)(t^2-t) + \frac{1}{4}}}$$

By a simple deduction from (654) and (658), it conclude that the results in (656) and (657) hold for $n = 2$.

Suppose that the ν - basis functions hold the properties at the endpoints for m , then

$$b_{m+1,i}(0) = b_{m,i}(0) = \begin{cases} 1 & i = 0; \\ 0, & i \neq 0. \end{cases} \quad b_{m+1,i}(1) = b_{m,i}(1) = \begin{cases} 1 & i = m; \\ 0, & i \neq m. \end{cases} \quad (659)$$

can be obtained by the inductive hypothesis and (655).

From (655), we have

$$b'_{m+1,i}(0) = -b_{m,i}(0) + b'_{m,i}(0) + b_{m,i-1}(0), \quad (660)$$

$$b'_{m+1,i}(1) = -b_{m,i}(1) + b_{m,i-1}(1) + b'_{m,i-1}(1), \quad (661)$$

By the inductive hypothesis and (659), the following conclusions can be deduced from (660),

- (i) If $i = 0$, then $b'_{m+1,0}(0) = -b_{m,0}(0) + b'_{m,0}(0) + b_{m,-1}(0) = -1 - (-m + \nu) = -(m + 1) + \nu$
- (ii) If $i = 1$, then $b'_{m+1,1}(0) = -b_{m,1}(0) + b'_{m,1}(0) + b_{m,0}(0) = (m + 1) - 2\nu$
- (iii) If $i = 2$, then $b'_{m+1,2}(0) = -b_{m,2}(0) + b'_{m,2}(0) + b_{m,1}(0) = 0 + \nu + 0 = \nu$
- (iv) If $i \neq 0, 1, 2$, then $b'_{m+1,i}(0) = 0$

Similarly, from (661), the following conclusions can be are extracted.

- (i) If $i = m + 1$, then $b'_{m+1,m+1}(1) = (m + 1) - \nu$
- (ii) If $i = m$, then $b'_{m+1,m}(1) = -(m + 1) + 2\nu$
- (iii) If $i = m - 1$, then $b'_{m+1,m-1}(1) = -\nu$
- (iv) If $i \neq m - 1, m, m + 1$, then $b'_{m+1,i}(1) = 0$

□

3 ν - curve

Definition 0.109. Given control points $P_i (i = 0, 1, 2, \dots, n)$ in R^2 or R^3 , for $0 \leq t \leq 1$, $0 < \nu \leq 1$,

$$r_n(t) = \sum_{i=0}^n b_{n,i}(t) P_i \quad (662)$$

is called a ν - curve, where $b_{n,i}(t) (i = 0, 1, 2, \dots, n; n \geq 3)$ are the ν - basis functions expressed in (654) and (655).



From Theorem 0.108, the ν - curve expressed in (662) has the following properties,

- (a) Geometric property at the endpoints: From the properties at the endpoints of the ν -basis functions, we have

$$\begin{aligned} r_n(0) &= P_0, & r_n(1) &= P_n \\ r'_n(0) &= (\nu) (P_2 - 2P_1 + P_0) + n (P_1 - P_0), \\ r'_n(1) &= (-\nu) (P_n - 2P_{n-1} + P_{n-2}) + n (P_n - P_{n-1}) \end{aligned}$$

- (b) Symmetry: From the symmetry of the ν -basis functions, one has

$$\begin{aligned} r_n(t; P_0, P_1, \dots, P_n) &= \sum_{i=0}^n b_{n,i}(t) P_i = \sum_{i=0}^n b_{n,n-i}(1-t) P_i \\ &= \sum_{j=0}^n b_{n,j}(1-t) P_{n-j} = r_n(1-t; P_n, P_{n-1}, \dots, P_0) \end{aligned}$$

- (c) Geometric invariance: Because $r_n(t)$ is an affine combination of the control points, the shape of the ν - curve is independent of the choice of coordinate system.
- (d) Convex hull property: Because the ν - basis functions are nonnegative and sum to one, the ν - curve must lie inside the convex hull of its control polygon.

Figure 40 shows two specific curves generated by ν - curves, the control points are fixed and value of ν is set for $\nu = 0.1, 0.2, \dots, 0.9, 1$, respectively, from inside to outside. One can see that the ν - curves approach the control polygons with the increase of the shape parameter.

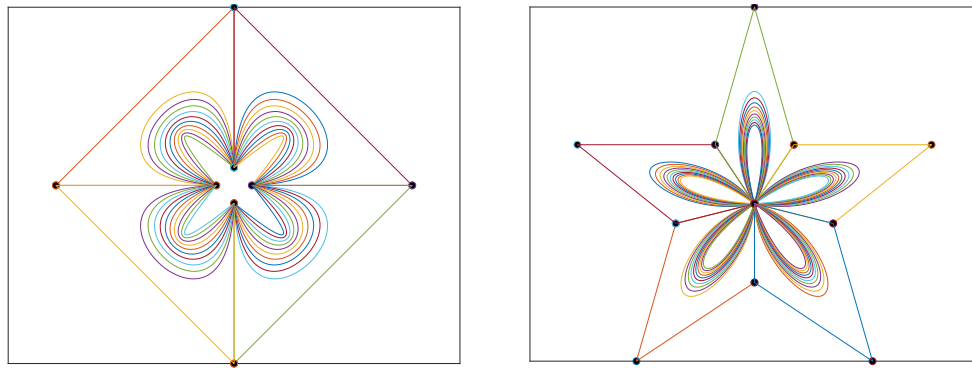


Figure 40: The shape consisting of ν - curve with different values of $\nu = 0.1, 0.2, \dots, 1$



4 ν - surface

Using tensor product, we can construct

$$S(s, t) = \sum_{i=0}^m \sum_{j=0}^n P_{i,j} b_i^m(t) b_j^n(t), \quad 0 \leq s, t \leq 1$$

in which $b_i^m(t)$, $b_j^n(t)$ are the ν - basis functions and $P_{i,j} \in \mathbb{R}^3$ is the control points.

Figure 41 shows the ν - surfaces and the effect on the surfaces by altering the values of the shape parameters at the same time under keeping the control net.

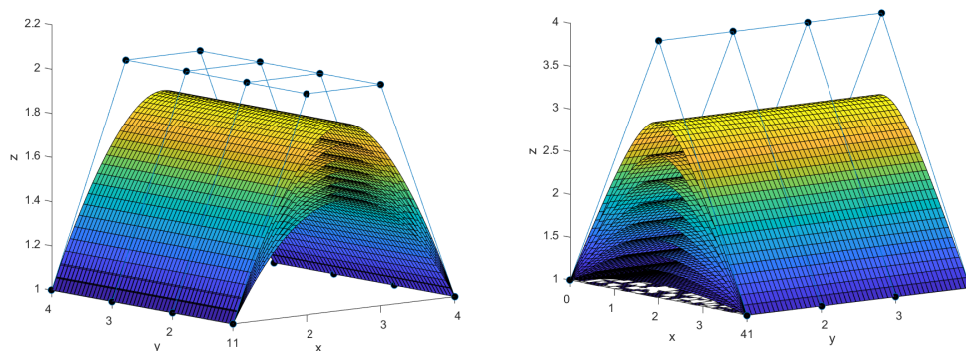


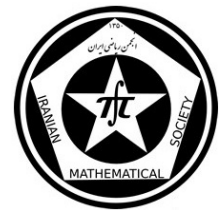
Figure 41: The shape of ν - surfaces with different shape parameters.

References

- [1] G.E. Farin and G. Farin, *Curves and surfaces for CAGD: a practical guide*. Morgan Kaufmann, 2002.
- [2] G. Hu, J. Wu and X. Qin, *A novel extension of the Bzier model and its applications to surface modeling*. Advances in Engineering Software **125** (2018), 27-54.
- [3] J. Li, *A novel Bzier curve with a shape parameter of the same degree*. Results in Mathematics **73**(4) (2018), 1-11.
- [4] K. Wang and G. Zhang, *Curve construction based on quartic Bernstein-like basis*. AIMS Mathematics **5.5** (2020), 5344-5363.
- [5] L. Yan and J. Liang, *An extension of the Bzier model*. Applied Mathematics and Computation **218**(6) (2011), 2863-2879.

Email: std_nouri411@khu.ac.ir

Email: j.saeidian@khu.ac.ir



On RBF–HFD formulas under multiquadric function on non-equidistant grids

F. Soleymani⁴²

Department of Mathematics, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran

Abstract

In this work, more generalized formulations for approximating derivatives of sufficiently smooth functions are constructed by considering two techniques, the radial basis function (RBF) and the Hermite finite difference (HFD). The RBF–HFD formulas lead to compact schemes that improve the existing FD and RBF–FD schemes for different types of optimization and PDE problems in terms of accuracy and stability.

AMS subject Classification 2010: 41A25.

Keywords: radial basis function; Hermite; multiquadric RBF; high order; weighting coefficients.

1 Introduction

It is well-known that in the Hermite interpolation the purpose is to calculate a polynomial that interpolates both the function values at neighboring points and the associated derivatives' values at the same or different points. However, as long as more than one derivative value from the Hermite approach is considered in the methodology of obtaining the weighting coefficients, more complicated symbolic computations should be done with more bulky formulas for the weights.

The results presented in this work can be applied to effectively tackle practical problems such as PDEs or PIDE problems in finance with multi dimensions via the RBF–HFD formulas which in fact result in sparse differentiation matrices (DMs), [4]. In fact, in such application the final discretization coefficient matrix is mainly dense but the convergence order is high as well as one can solve system at which both sides are matrices in order to reduce the cost of density. A good choice of c can also affect on accurate results. Numerical results have been provided to report the applicability of our new formulas for nonuniform grids with accurate and high rate of convergence.

Here, the weights for the multiquadric RBF generated HFD scheme is reported in closed form for both the first and second derivatives on a nonuniform mesh having three unequally spaced nodes. Moreover, it is shown that the formulations have higher convergence speed, which are very efficient and have been attained by employing only three nearby nodes of a grid. In fact, the results here in view of the other existing literature is that the weights are derived for the multiquadric RBF in closed form under the RBF–HFD methodology which possess higher rates of convergence.

⁴²Speaker, Email: fazlollah.soleymani@gmail.com.



2 RBF–HFD background

Let us consider the multiquadric RBF as follows [2]:

$$\phi(\|\mathbf{x} - \mathbf{x}_i\|_2) = \sqrt{c^2 + \|\mathbf{x} - \mathbf{x}_i\|_2^2}, \quad i = 1, 2, \dots, m, \quad (663)$$

wherein the shape parameter is c . This parameter in the RBF methodology has a direct influence in the accuracy of the results as well as in the ill-conditioning of the approximations specially when used for solving differential equations.

The significance of weighting coefficients in the RBF–HFD estimates is that by taking into account only three nearby even unequally-spaced nodes, one may can obtain higher rates of convergence for approximating derivatives. This is not possible in the estimates of the FD; and a higher order such as four can only be obtained when at least five nodes be considered, [3].

Here the concept of Hermite permits us to have not only the values of the function at each adjacent nodes, but also their derivative values, [6]. Bearing this methodology in mind, one is able to retrieve higher convergence orders. Now we use the RBF (663) to interlink with the HFD concept and find the weights of function derivative estimating. If the shape parameter is conveniently tuned, then the RBF–HFD is much more accurate than the Hermite-FD or RBF-FD schemes.

Let us consider a stencil having three non equidistant nodes as follows:

$$\{x_{i-1}, x_i, x_{i+1}\} = \{x_i - h, x_i, x_i + \omega_{i+1}h\}, \quad h > 0. \quad (664)$$

The new weights for the multiquadric RBF–HFD formulas will be optimized and contributed for the first time in the next section on the nonuniform mesh (664).

3 Optimizing the weights

At this moment, take into account the following formula of the RBF–HFD estimate:

$$\begin{aligned} g'(x_i) \simeq & \alpha_{i-1}g(x_{i-1}) + \alpha_i g(x_i) + \alpha_{i+1}g(x_{i+1}) \\ & + \beta_{i-1}g'(x_{i-1}) + \beta_{i+1}g'(x_{i+1}). \end{aligned} \quad (665)$$

The theorem below discussed the analysis of error of (665) in approximating the first derivative of a function.

Theorem 0.110. *The estimate of the RBF–HFD for the first derivative of a sufficiently differentiable function g by employing (665) is of quartic convergence order, as long as the weighting coefficients are given by:*



$$\alpha_{i-1} = \frac{\omega_{i+1}^2 (\omega_{i+1} (-14c^2 + 31h^2\omega_{i+1} + 62h^2) - 28c^2 - 121h^2)}{7c^2h(\omega_{i+1} + 1)^3}, \quad (666)$$

$$\alpha_i = \frac{(\omega_{i+1} - 1) \left(\frac{14}{\omega_{i+1}} - \frac{31h^2}{c^2} \right)}{7h}, \quad (667)$$

$$\alpha_{i+1} = \frac{\frac{h^2(\omega_{i+1}(121\omega_{i+1}-62)-31)}{c^2} + 14 \left(\frac{1}{\omega_{i+1}} + 2 \right)}{7h(\omega_{i+1} + 1)^3}, \quad (668)$$

$$\beta_{i-1} = -\frac{\omega_{i+1}^2 (7c^2 - 31h^2\omega_{i+1} + 45h^2)}{7c^2(\omega_{i+1} + 1)^2}, \quad (669)$$

$$\beta_{i+1} = \frac{h^2(31 - 45\omega_{i+1})\omega_{i+1} - 7c^2}{7c^2(\omega_{i+1} + 1)^2}. \quad (670)$$

Proof. For proving and obtaining the order of convergence this, we replace the function g by (663) at the stencil nodes to get a system of linear equations as follows:

$$\begin{aligned} \frac{2h\beta_{i-1}}{\sqrt{c^2 + 4h^2}} &= \alpha_{i+1}\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2} + \sqrt{c^2 + 4h^2}\alpha_{i-1} + \sqrt{c^2 + h^2}\alpha_i \\ &\quad + \frac{h\beta_{i+1}(\omega_{i+1} - 1)}{\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2}} + \frac{h}{\sqrt{c^2 + h^2}}, \end{aligned} \quad (671)$$

$$\frac{h\beta_{i-1}}{\sqrt{c^2 + h^2}} = \frac{\alpha_{i+1}(c^2 + h^2\omega_{i+1}^2) + h\beta_{i+1}\omega_{i+1}}{\sqrt{c^2 + h^2\omega_{i+1}^2}} + \sqrt{c^2 + h^2}\alpha_{i-1} + c\alpha_i, \quad (672)$$

$$\begin{aligned} \frac{h\omega_{i+1}}{\sqrt{c^2 + h^2\omega_{i+1}^2}} &= \alpha_{i-1}\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2} + \alpha_i\sqrt{c^2 + h^2\omega_{i+1}^2} \\ &\quad + \alpha_{i+1}\sqrt{c^2 + 4h^2\omega_{i+1}^2} + \frac{h\beta_{i-1}(\omega_{i+1} - 1)}{\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2}} + \frac{2h\beta_{i+1}\omega_{i+1}}{\sqrt{c^2 + 4h^2\omega_{i+1}^2}}, \end{aligned} \quad (673)$$

$$\begin{aligned} \frac{2h\alpha_{i-1}}{\sqrt{c^2 + 4h^2}} + \frac{h(c^2 + h^2)\alpha_i + c^2}{(c^2 + h^2)^{3/2}} &= \frac{h\alpha_{i+1}(\omega_{i+1} - 1)}{\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2}} \\ &\quad + \frac{c^2\beta_{i+1}}{(c^2 + h^2(\omega_{i+1} - 1)^2)^{3/2}} + \frac{c^2\beta_{i-1}}{(c^2 + 4h^2)^{3/2}}, \end{aligned} \quad (674)$$

$$(675)$$

$$\begin{aligned} \frac{c^2}{(c^2 + h^2\omega_{i+1}^2)^{3/2}} &= \frac{h\alpha_{i-1}(\omega_{i+1} - 1)}{\sqrt{c^2 + h^2(\omega_{i+1} - 1)^2}} + \frac{h\alpha_i\omega_{i+1}}{\sqrt{c^2 + h^2\omega_{i+1}^2}} \\ &\quad + \frac{2h\alpha_{i+1}\omega_{i+1}}{\sqrt{c^2 + 4h^2\omega_{i+1}^2}} + \frac{c^2\beta_{i-1}}{(c^2 + h^2(\omega_{i+1} - 1)^2)^{3/2}} + \frac{c^2\beta_{i+1}}{(c^2 + 4h^2\omega_{i+1}^2)^{3/2}}. \end{aligned} \quad (676)$$



Solving (671)–(676) by considering $c \gg h$ and some simplifications [5], we get (666)–(668). The reason of doing this transformation $c \gg h$ is the fact that the results are quite tedious and bulky as well as we wish to reduce the sensitivity of the shape parameter. So, by considering $c \gg h$, the results would be simplified and always an enough good shape parameter based on the fact that it should be greater than h can be derived and introduced.

This furnishes the error equation below to approximate the first derivative of the function:

$$\varepsilon(x_i) = -\frac{(\omega_{i+1}^2 (c^2 g^{(5)}(x_i) + 40g^{(3)}(x_i)))}{120c^2} h^4 + \mathcal{O}(h^5), \quad (677)$$

wherein

$$\varepsilon(x_i) = \hat{g}'(x_i) - g'(x_i). \quad (678)$$

The error equation (677) proves a quartic convergence of the proposed approximation (665) on even unequally-spaced meshes. The proof ends now. $\square$

To discuss the second derivative case, by employing the methodology of the RBF–HFD, we take into account (664) and write down:

$$\begin{aligned} g''(x_i) &\simeq \lambda_{i-1}g(x_{i-1}) + \lambda_i g(x_i) + \lambda_{i+1}g(x_{i+1}) \\ &\quad + \mu_{i-1}g''(x_{i-1}) + \mu_{i+1}g''(x_{i+1}). \end{aligned} \quad (679)$$

Now, we furnish the weights and their corresponding equation of error at the theorem below.

Theorem 0.111. *The formula of the RBF–HFD in approximating the second derivative of a sufficiently differentiable function g employing (665) is of cubic convergence order as long as the weights are defined properly (due to page limitation we do not provide closed form relations further).*

Proof. Having the proof of Theorem 0.110, one obtains the weights when $c \gg h$, we can finally obtain the following error equation:

$$\begin{aligned} \varepsilon(x_i) = & -\frac{(\omega_{i+1} (2\omega_{i+1}^3 + 3\omega_{i+1}^2 - 3\omega_{i+1} - 2) (c^4 g^{(5)}(x_i) + 40c^2 g^{(3)}(x_i) + 75g'(x_i)))}{30 (c^4 (\omega_{i+1}^2 + 3\omega_{i+1} + 1))} h^3 \\ & + \mathcal{O}(h^4), \end{aligned} \quad (680)$$

wherein

$$\varepsilon(x_i) = \hat{g}''(x_i) - g''(x_i). \quad (681)$$

The error equations (680) reveals a cubic convergence speed for the RBF–HFD approximation (679). $\square$



Table 51: Results of comparison on problem (682).

Step size	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
FD	0.08789	0.00093	9.3×10^6	9.3×10^{-8}	9.3×10^{-10}
COC (FD)	-	1.97	1.99	2.00	2.00
RBF-HFD	0.00301	4.2×10^{-7}	4.2×10^{-11}	4.2×10^{-15}	4.2×10^{-19}
COC (RBF-HFD)	-	3.85	3.99	3.99	4.00

4 An experiment

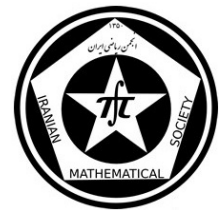
In this part, we check the above-mentioned discussions on an experiment to show the efficiency of the RBF-HFD method over the existing solvers. We employ the following test problem on the domain $[0, 1]$:

$$f(x) = \sin^{\exp(x)}(4x) - 10x^6 + \tanh(10x) - 1, \quad (682)$$

when the shape parameters is $c = 1$ at the position $x = 0.5$. Table 52 provides the convergence history along with the associated computational orders of convergence (COCs). Numerical results confirm the theoretical discussions.

References

- [1] V. Bayona, M. Moscoso, M. Kindelan, *Optimal constant shape parameter for multiquadric based RBF-FD method*, J. Comput. Phys. 230 (2011), 7384-7399.
- [2] G.E. Fasshauer, *Meshfree Methods, Handbook of Theoretical and Computational Nanotechnology*, vol. 27, American Scientific Publishers, (2006), 33-97.
- [3] J. Gu, J.-H. Jung, *Adaptive radial basis function methods for initial value problems*, J. Sci. Comput., 82 (2020), Article ID: 47.
- [4] F. Soleymani, A. Itkin, *Pricing foreign exchange options under stochastic volatility and interest rates using an RBF-FD method*, J. Comput. Sci., 37 (2019), Art. ID: 101028.
- [5] M. Trott, *The Mathematica Guidebook for Programming*, Springer, New York, NY, USA, 2004.
- [6] G.B. Wright, B. Fornberg, *Scattered node compact finite difference-type formulas generated from radial basis functions*, J. Comput. Phys., 212 (2006), 99-123.



Solving a new class of nonlinear optimal control problems generated by Atangana-Baleanu-Caputo variable order fractional derivative

F. Ghomanjani

Department of Mathematics, Kashmar Higher Education Institute, Kashmar, Iran, f.ghomanjani@kashmar.ac.ir

Abstract

The numerical solution of a new class of nonlinear optimal control problems (OCPs) generated by Atangana-Baleanu-Caputo (ABC) variable order (V-O) fractional derivative (FD) is found by Bezier curve method (BCM). The main goal behind this work is the use of the BCM. Using this method, it is possible to obtain BCM solution of the general form of multipoint boundary value problems. To shown the efficiency of the developed method, numerical results are stated as the main results in this study.

AMS subject Classification 2010: 65K10, 26A33, 49K15.

Keywords: variable order, Bezier curve, nonlinear optimal control problems (NOCPs).

1 Introduction

OCPs is one of the main topics refereed to the V-O fractional. These problems are related to the V-O fractional operators in their cost functional and dynamical system. Recently, many numerical methods have been stated such as in [1] and [2], the Bernstein functions for nonlinear V-O fractional OCPs is stated. In [3], generalized polynomials is studied for a kind of V-O fractional 2D OCPs.

In this sequel, the Bezier curve method is applied to solve V-O fractional OCPs (689).

Additionally some papers spent the Bezier curves. In [4] and [5], the authors utilized the Bezier curves for solving delay differential equation (DDE) and optimal control of switched systems numerically. In [6], the authors proposed the utilization of Bezier curves on some linear optimal control systems with pantograph delays. Also, to solve the quadratic Riccati differential equation and the Riccati differential-difference equation, the Bezier control points strategy is utilized (see [7]). Some other uses of the Bezier functions are found in (see [8]).

The organization of this study is classified as follows: Problem statement is introduced in Section 36. Also solving ABC V-O FD based on the Bezier curves is stated in Section 36. Convergence analysis is stated in Section 36. A numerical example is solved in Section 36. Finally, Section 36 will give a conclusion briefly.



2 Problem statement

In this paper, the following definition is considered

Definition 0.112. let $\alpha : [0, \tau_{max}] \rightarrow (0, 1)$ be a continuous function and $x \in C^1[0, \tau_{max}]$. The V-O FD of order $\alpha(\tau)$ in the ABC sense of $x(\tau)$ is defined as follows (see [9]):

$${}_0^{ABC}D_t^{\alpha(\tau)}x(\tau) = \frac{C(\alpha(\tau))}{1 - \alpha(\tau)} \int_0^\tau x'(s)E_{\alpha(\tau)}\left(\frac{-\alpha(\tau)(\tau - s)^{\alpha(\tau)}}{1 - \alpha(\tau)}\right) ds, \quad (683)$$

where

$$E_\lambda(\tau) = \sum_{j=0}^{\infty} \frac{\tau^j}{\Gamma(j\lambda + 1)}, \quad \lambda \in R^+, \quad \tau \in R,$$

$$C(\alpha(\tau)) = 1 - \alpha(\tau) + \frac{\alpha(\tau)}{\Gamma(\alpha(\tau))},$$

$${}_0^{ABC}D_t^{\alpha(\tau)}c = 0, \quad (\text{for any constant } c).$$

So, we focus on the following problem

$$\begin{aligned} \min J &= \int_0^{\tau_{max}} L(\tau, x_1(\tau), x_2(\tau), \dots, x_m(\tau), u(\tau)) d\tau, \\ \text{such that } {}_0^{ABC}D_t^{\alpha_i(\tau)}x(\tau) &= G_i(\tau, x_1(\tau), x_2(\tau), \dots, x_m(\tau), u(\tau)) \\ \alpha_i(\tau) &\in (0, 1), \quad i = 1, 2, \dots, m, \quad m \in N \\ x_i(0) &= x_{i,0}, \quad i = 0, 1, \dots, m, \quad \tau_{max} \in R. \end{aligned} \quad (684)$$

where L, G_i for $i = 1, 2, \dots, m$ are continuous operators, α_i for $i = 1, 2, \dots, m$ is a continuous function on $[0, \tau_{max}]$ and $x_{i,0}$ is a given real constant.

3 Solving ABC V-O FD based on the Bezeir curves

Our aim is utilizing Bezier curves to approximate the solutions $x(\tau)$ and $u(\tau)$ where $x(\tau)$ and $u(\tau)$ are given below. Define the Bezier polynomials of degree n over the interval $[\tau_0, \tau_f]$ as follows:

$$x(\tau) = \sum_{r=0}^n a_r B_{r,n}\left(\frac{\tau - \tau_0}{h}\right), \quad \tau_f = 1, \quad \tau_0 = 0, \quad (685)$$

$$u(\tau) = \sum_{r=0}^n b_r B_{r,n}\left(\frac{\tau - \tau_0}{h}\right), \quad (686)$$

where $h = \tau_f - \tau_0$, and

$$B_{r,n}\left(\frac{\tau - \tau_0}{h}\right) := \binom{n}{r} \frac{1}{h^n} (\tau_f - \tau)^{n-r} (\tau - \tau_0)^r$$



is the Bernstein polynomial of degree n over the interval $[\tau_0, \tau_f]$, and $a_r, b_r, r = 0, 1, \dots, n$, and they are unknown control points.

Also we have

$$\begin{aligned} \frac{dB_{r,n}(\tau)}{d\tau} &= n(B_{r-1,n-1}(\tau) - B_{r,n-1}(\tau)), \\ \frac{dx(\tau)}{d\tau} &= \sum_{r=0}^{n-1} na_r B_{r-1,n-1}(\tau) - \sum_{r=0}^{n-1} na_r B_{r,n-1}(\tau) \\ &= \sum_{r=0}^{n-1} na_{r+1} B_{r,n-1}(\tau) - \sum_{r=0}^{n-1} na_r B_{r,n-1}(\tau) = \sum_{r=0}^{n-1} B_{r,n-1}(\tau) n(a_{r+1} - a_r), \end{aligned}$$

then

$$\begin{aligned} {}_0^{ABC}D_t^{\alpha(\tau)}x(\tau) &= \frac{C(\alpha(\tau))}{1-\alpha(\tau)} \int_0^\tau \left(\sum_{r=0}^n a_r B_{r,n}(\tau) \right)' E_{\alpha(\tau)}\left(\frac{-\alpha(\tau)(\tau-s)^{\alpha(\tau)}}{1-\alpha(\tau)}\right) ds \\ &= \frac{C(\alpha(\tau))}{1-\alpha(\tau)} \int_0^\tau \sum_{i=0}^{n-1} B_{i,n-1}(\tau) n(a_{i+1} - a_i) E_{\alpha(\tau)}\left(\frac{-\alpha(\tau)(\tau-s)^{\alpha(\tau)}}{1-\alpha(\tau)}\right) ds \end{aligned} \quad (687)$$

By substituting $x(\tau)$ and $u(\tau)$ and Eq.(687) in Eq.(684), we obtain a simplified problem then we can solve this problem by Maple 16. Our goal is to solve the following optimization problem over the interval $[\tau_0, \tau_f]$ to find the entries of the vectors a_r, b_r , for $r = 0, 1, \dots, n$.

4 Convergence analysis

In this section, we can suppose the following problem

$$\begin{aligned} \min J &= \int_0^{\tau_{max}} x^T(\tau)P(\tau)x(\tau) + u^T(\tau)Q(\tau)u(\tau) d\tau, \\ \text{such that } {}_0^{ABC}D_t^{\alpha(\tau)}x(\tau) - A(\tau)x(\tau) - B(\tau)u(\tau) &= F(\tau), \\ x(0) = x_0 = a, u(0) = u_0 = b, \quad a, b &\in R, \tau_{max} = 1, \end{aligned} \quad (688)$$

where $P(\tau)$ and $Q(\tau)$ are given non-negative functions for $\tau \in [0, 1]$.

Lemma 0.113. For a polynomial in Bezier form

$$x(\tau) = \sum_{i=0}^{n_2} a_{i,n_2} B_{i,n_2}(\tau),$$

where a_{i,n_2+m_1} is the Bezier coefficient of $x(\tau)$ after being degree-elevated to degree $n_2 + m_1$. Now, we have

$$\frac{\sum_{i=0}^{n_2} a_{i,n_2}^2}{n_2 + 1} \geq \frac{\sum_{i=0}^{n_2+1} a_{i,n_2+1}^2}{n_2 + 2} \geq \dots \geq \frac{\sum_{i=0}^{n_2+m_1} a_{i,n_2+m_1}^2}{n_2 + m_1 + 1}.$$



Proof. For proof, we can use [6]. □

Theorem 0.114. If the problem (688) has a unique C^1 continuous solution $\bar{x}$, C^0 continuous control solution $\bar{u}$, then the approximate solution obtained by the control-point-based method converges to the exact solution $(\bar{x}, \bar{u})$ as the degree of the approximate solution tends to infinity.

Proof. For proof, we use [6]. □

5 Numerical application

Now, a numerical example of ABC V-O FD is stated to illustrate the BCM. All results are obtained by utilizing Maple 16.

Example 0.115. The following ABC V-O FD is considered (see [9])

$$\begin{aligned} \min J[u] &= \frac{1}{2} \int_0^1 x^2(\tau) + u^2(\tau) d\tau, \\ {}_0^{ABC}D_t^{\alpha(\tau)} x(\tau) &= u(\tau) - x(\tau), \\ x(0) &= 1, \end{aligned}$$

where the exact solution with $\alpha(\tau) = 1$ is followed as:

$$\begin{aligned} x_{exact}(\tau) &= \cosh(\sqrt{2}\tau) + \nu \sinh(\sqrt{2}\tau), \\ u_{exact}(\tau) &= (1 + \sqrt{2}\nu) \cosh(\sqrt{2}\tau) + (\sqrt{2} + \nu) \sinh(\sqrt{2}\tau), \\ \nu &= -\frac{\cosh(\sqrt{2}) + \sqrt{2} \sinh(\sqrt{2})}{\sqrt{2} \cosh(\sqrt{2}) + \sinh(\sqrt{2})} = -0.98, \\ J_{exact} &= 0.192909. \end{aligned}$$

By BCM with $\alpha = 0.95 + 0.04\sin(\tau)$, we obtain the following solution:

$$\begin{aligned} x_{approx}(\tau) &= -1.383424070\tau + 0.9779939940\tau^2 - 0.3971018840\tau^3 \\ &\quad + 0.08450149506\tau^4 + 0.9999999998, \\ u_{approx}(\tau) &= -2.365740323 - 0.06762181093\tau + 2.572114069\tau^2 - 8.284294539\tau^3 \\ &\quad + 3.832909652\tau^4, \\ J_{approx} &= 0.1929636321. \end{aligned}$$

Table 52 shows comparison of the values of $x(\tau)$ and $u(\tau)$ for proposed method, Chebyshev cardinal function method [9], and exact solution (with $\alpha = 0.95 + 0.04\sin(\tau)$).

6 Conclusions

This paper deals with the approximate solution of ABC V-O fractional derivative via BCM. The solution obtained using the suggested method is in very good agreement with the already existing ones and state that this approach can solve the problem effectively. Although we considered a model problem, the main idea and techniques used in this work are also applicable to many other problems.

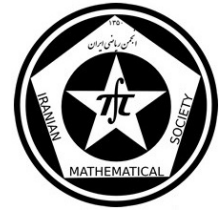


Table 52: Comparison of the value of $x(\tau)$ and $u(\tau)$ for proposed method, Chebyshev cardinal function method, and exact solution for Example 0.115

τ	$x(\tau)$ in proposed method	$x(\tau)$ in Chebyshev	$x(\tau)$ in exact	$u(\tau)$ in proposed method	$u(\tau)$ in Chebyshev	$u(\tau)$ in exact
0.2	0.7593933324	0.711981	0.759393	-0.2768738382	-0.305356	-0.276873
0.4	0.5798581284	0.549930	0.579944	-0.1854511779	-0.210872	-0.190227
0.6	0.4472007827	0.430800	0.447200	-0.1141705380	-0.129592	-0.118900
0.8	0.3504725480	0.343133	0.350472	-0.05714883910	-0.061000	-0.057148
1.0	0.2819695347	0.280370	0.281969	0.0	0.012271	0.0

References

- [1] H. Hassani, Z. Avazzadeh, Transcendental bernstein series for solving nonlinear variable order fractional optimal control problems, Appl Math Comput, 366 (2019), p. 124563.
- [2] H. Hassani, Z. Avazzadeh, J.A.T. Machado, Solving two-dimensional variable-order fractional optimal control problems with transcendental Bernstein series, J Comput Nonlinear Dyn, 14 (6) (2019), p. 061001.
- [3] F. Mohammadi, H. Hassani, Numerical solution of two-dimensional variable-order fractional optimal control problem by generalized polynomial basis, J Optim Theory Appl, 10 (2) (2019), 536-555.
- [4] F. Ghomanjani, M.H. Farahi, The Bezier control points method for solving delay differential equation, Intelligent Control and Automation, 3(2), (2012) 188-196.
- [5] F. Ghomanjani, M.H. Farahi, Optimal control of switched systems based on bezier control points. Int J Intell Syst Appl 4(7) (2012) 16-22 .
- [6] F. Ghomanjani, M.H. Farahi, A.V. Kamyad, Numerical solution of some linear optimal control systems with pantograph delays. IMA J Math Control Inf (2013).
- [7] F. Ghomanjani, E. Khorram, Approximate solution for quadratic Riccati differential equation, Journal of Taibah university for science, 11 (2017) 246-250.
- [8] F. Ghomanjani, A new approach for solving fractional differential-algebraic equations, Journal of Taibah university for science, (2017) DOI: /10.1016/j.jtusci.2017.03.006.
- [9] M.H.Heydari, Chebyshev cardinal functions for a new class of nonlinear optimal control problems generated by Atangana-Baleanu-Caputo variable-order fractional derivative, Chaos, Solitions and Fractals, (2020), DOI:10.1016/j.chaos.2019.109401.



Solving the Volterra functional integral equations with non-vanishing delays

F. Ghomanjani

Department of Mathematics, Kashmar Higher Education Institute, Kashmar, Iran, f.ghomanjani@kashmar.ac.ir

Abstract

In this work, a numerical method for Volterra functional integral equations (VFIEs) with non-vanishing delays is shown. This approach depends on Bernstein polynomial approximation. Numerical example is utilized to evaluate the accurate results.

AMS subject Classification 2010: 65K10, 26A33, 49K15.

Keywords: Volterra functional integral equations (VFIEs), non-vanishing delays.

1 Introduction

The following general Volterra functional integral equation (VFIE) is considered

$$\begin{aligned} y(t) &= g(t) + b(t)y(\theta(t)) + \int_{t_0}^t k_1(t, s)y(s)ds \\ &+ \int_{t_0}^{\theta(t)} k_2(t, s)y(s)ds, \quad t \in I := (t_0, T] = (t_0, t_f], \\ y(t) &= \phi(t), \quad t \in I_\theta := [\theta(t_0), t_0], \quad t_f = T, \end{aligned} \tag{689}$$

where $0 < q < 1$, and $k_1(t, s)$ and $k_2(t, s)$ are assumed to be continuous functions on their respective domains $D := \{(t, s) : t_0 \leq s \leq t \leq T, t \in I\}$ and $D_\theta := \{(t, s) : \theta(t_0) \leq s \leq \theta(t), t \in I\}$ (for more explanation see [1]).

The hp variant of the discontinuous Galerkin technique for the numerical solution of delay differential equations (DDEs) with nonlinear vanishing delays was introduced in [2]. In [3], the existence, uniqueness of solutions for VFIEs with the time delay were analyzed. Huang et al. [4] studied the superconvergence of the discontinuous Galerkin solutions for DDEs with proportional vanishing at $t = 0$.

Additionally many papers manage the Bezier curves. In [5] and [6], authors utilized the Bezier curves



in approximating functions. For solving differential equations (DEs) numerically, authors in [7] proposed the utilization of Bezier curves. Also, to solve delay differential equation, the Bezier control points strategy is utilized (see [8]). Some other uses of the Bezier functions are found in (see [9]). In the present work, we use the proposed method in [9] for solving VFIE. The outline of this sequel is as follow: In Section 36, problem statement is stated. Explanation of the problem is explained in Section 36. A numerical example is provided in Section 36. Finally, Section 36 will give a conclusion briefly.

2 Problem Statement

For finding an approximate solution of Eq. (689), one may consider the following $y(t)$:

$$y(t) \cong y_n(t) = \sum_{i=0}^n c_i B_{i,n}(t), \quad t_0 \leq t \leq t_f, \quad n \geq 1, \quad (690)$$

where

$$B_{r,n}\left(\frac{t-t_0}{h}\right) = \binom{n}{r} \frac{1}{h^n} (t_f - t)^{n-r} (t - t_0)^r, \quad t_0 \leq t \leq t_f, \quad 0 \leq i \leq n, \quad h = t_f - t_0,$$

and c_i , $i = 0, 1, \dots, n$ are the unknown control points. In this paper, one may utilize Bezier curve for solving VFIE. This technique is applied in [10], [11] for solving optimal control of problems (OCPs) and some linear OCPs with pantograph delays. The convergence of the proposed technique is proven in this sequel.

By substituting $y(t)$ in (689), one may define

$$f_{objective} = \sum_{i=0}^n c_i^2, \quad t \in [t_0, t_f], \quad (691)$$

Now, our goal is to solve the following problem to find the values c_r , for $r = 0, 1, \dots, n$.

$$\begin{aligned} \min \quad & f_{objective} \\ \text{s.t.} \quad & y(t) = g(t) + b(t)y(\theta(t)) + \int_{t_0}^t k_1(t, s)y(s)ds \\ & + \int_{t_0}^{\theta(t)} k_2(t, s)y(s)ds, \quad t \in I := (t_0, T], \\ & y(t) = \phi(t), \quad t \in I_\theta := [\theta(t_0), t_0], \end{aligned} \quad (692)$$

where "s.t." is abbreviation of "such that".



3 Explanation of the problem

Now, the following problem is considered

$$\begin{aligned}(Dy)(t) &= y(t) - g(t) + b(t)y(\theta(t)) + \int_0^t k_1(t, s)y(s)ds + \int_0^{\theta(t)} k_2(t, s)y(s)ds, \\ y_n(t) &= \sum_{i=0}^n c_i B_{i,n}(t), \quad 0 \leq t \leq t_f, \\ y(t) &= \phi(t), \quad t \leq 0.\end{aligned}$$

Without loss of generality, one may consider $t_0 = 0$. Firstly, we have

$$\begin{aligned}(Dy_n)(t) &= \sum_{i=0}^n c_i B_{i,n}(t) - g(t) + b(t) \sum_{i=0}^n c_i B_{i,n}(\theta(t)) \\ &+ \int_0^t k_1(t, s) \sum_{i=0}^n c_i B_{i,n}(s)ds + \int_0^{\theta(t)} k_2(t, s) \sum_{i=0}^n c_i B_{i,n}(s)ds \\ &= \sum_{i=0}^n c_i \left(B_{i,n}(t) + b(t)B_{i,n}(\theta(t)) + \int_0^t k_1(t, s)B_{i,n}(s)ds + \int_0^{\theta(t)} k_2(t, s)B_{i,n}(s)ds \right) - g(t) \\ &= \sum_{i=0}^n c_i \alpha_i(t) - g(t),\end{aligned}$$

where $\alpha_i = \left(B_{i,n}(t) + b(t)B_{i,n}(\theta(t)) + \int_0^t k_1(t, s)B_{i,n}(s)ds + \int_0^{\theta(t)} k_2(t, s)B_{i,n}(s)ds \right)$, $i = 0, 1, 2, \dots, n$.

Now, the following real function is defined

$$J = J(c_0, c_1, \dots, c_n) = \int_0^{t_f} (Dy_n)^2(t)dt,$$

according to the above equations, we have

$$\begin{aligned}\min J(C) \\ s.t. \quad I(C) &= \sum_{i=0}^n c_i \left(\sum_{k=0}^n c_k B_{k,n}(-\frac{j}{n}) \right) - \phi(-\frac{j}{n}), \quad j = 0, 1, \dots, n,\end{aligned} \tag{693}$$

where $C = (c_0, c_1, \dots, c_n)$. There are various techniques for solving this problem, such as the Newton's method, the gradient descent, the conjugate-gradient technique and the Lagrange technique. Here, the Lagrange-multiplier method is considered to solve the problem (693). By the Lagrange function, one may have the following form of

$$L = J(C) + \mu I(C),$$



Hence

$$\begin{aligned}
\frac{\partial L}{\partial c_i} &= 2 \int_0^{t_f} (Dy_n)(t) \frac{\partial (Dy_n)(t)}{\partial c_i} dt + \mu \frac{\partial I(C)}{\partial c_i} \\
&= 2 \left(\sum_{j=0}^n c_j \int_0^{t_f} \alpha_j(t) \alpha_i(t) dt - \int_0^{t_f} g(t) \alpha_i(t) dt \right) + \mu \sum_{k=0}^n c_k B_{k,n} \left(-\frac{i}{n} \right), \\
\frac{\partial L}{\partial \mu} &= I(C),
\end{aligned} \tag{694}$$

As we known, a necessary condition (693) is that

$$\begin{cases} \frac{\partial L}{\partial c_i} = 0, & 0 \leq i \leq n, \\ \frac{\partial L}{\partial \mu} = 0, \end{cases} \tag{695}$$

For simplification, one may define

$$(f, g) = \int_0^{t_f} f(t)g(t)dt, \tag{696}$$

Combining Eqs. (694)-(696), we have

$$\begin{aligned}
\sum_{j=0}^n 2c_j(\alpha_j, \alpha_i) + \mu \sum_{k=0}^n c_k B_{k,n} \left(-\frac{i}{n} \right) &= 2(g, \alpha_i), \quad 0 \leq i \leq n, \\
\sum_{i=0}^n c_i B_{i,n}(t) &= \phi(t), \quad t \leq 0,
\end{aligned} \tag{697}$$

we can rewrite Eq. (697) as the following form

$$\sum_{i=0}^n c_i B_{i,n} \left(-\frac{j}{n} \right) = \phi \left(-\frac{j}{n} \right), \quad -n \leq j \leq 0.$$

Now, we write

$$\begin{aligned}
G\tilde{C} &= F, \\
G &= \begin{bmatrix} 2(\alpha_0, \alpha_0) & 2(\alpha_1, \alpha_0) & \dots & 2(\alpha_n, \alpha_0) & \phi(0) \\ 2(\alpha_0, \alpha_1) & 2(\alpha_1, \alpha_1) & \dots & 2(\alpha_n, \alpha_1) & \phi(-\frac{1}{n}) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 2(\alpha_0, \alpha_n) & 2(\alpha_1, \alpha_n) & \dots & 2(\alpha_n, \alpha_n) & \phi(-\frac{n-1}{n}) \\ B_{0,n}(0) & B_{1,n}(0) & \dots & B_{n,n}(0) & 0 \end{bmatrix}, \\
\tilde{C} &= [c_0, c_1, \dots, c_n, \mu]^T, \\
F &= [2(g, \alpha_0), \dots, 2(g, \alpha_n), \phi(0)]^T.
\end{aligned} \tag{698}$$

the unique solution of linear system (698) is $y_n(t) = \sum_{i=0}^n c_i B_{i,n}(t)$ is called an optimal control approximation solution (OCAS). If $\lim_{n \rightarrow \infty} \int_0^{t_f} (Dy_n)^2(t)dt = 0$ then the OCAS converges to the exact solution.



3 Numerical applications

In this section, an example is given to demonstrate the quality of the sated technique in approximating the solution of delay Volterra integral equations (689).

Example 0.116. The following vanishing delay Volterra integral equation (VIE) is considered (see [13])

$$y(t) = \frac{1}{2}(1 + e^{-qt}) - \int_0^t y(s)ds + \frac{1}{2} \int_0^{qt} y(s)ds,$$

where $q = 0.2$, and $y_{exact}(t) = e^{-t}$. One may obtain $y(t) = 1 - 0.99941777t + 0.4953779080t^2 - 0.153598222t^3 + 0.0255175232t^4$ with this method by $n = 4$. Table 53 demonstrates the errors of the this technique where $\|error\|_\infty$ in [13] and this paper are respectively 2×10^{-4} (for $n = 4, m = 1$) and 0.0.

Table 53: The errors of the this method for Example 0.116

t	error of y
0.1	0.00002353764412
0.2	0.0000128515059
0.3	7.4×10^{-11}
0.4	0.000003726205
0.5	8×10^{-11}
0.6	0.00000360418
0.7	3×10^{-11}
0.8	0.0000116301
0.9	0.0000206027
1.0	0.0

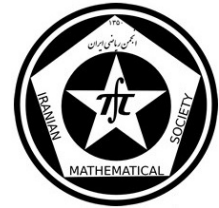
4 Conclusions

The aim of this sequel is to improve an effective and accurate technique for solving VFIE and fractional Bagley-Torvik equation. The Bezier curve strategy is utilized to obtain the approximate solution of this problem. The result is included to explain the validity of this technique.



References

- [1] W. Ming, C. Huang, Collocation methods for Volterra functional integral equations with non-vanishing delays *Applied Mathematics and Computation*, 296, 198–214 (2017).
- [2] Q. Huang, H. Xie, H. Brunner, The hp discontinuous Galerkin method for delay differential equations with nonlinear vanishing delay, *SIAM J. Sci. Comput.* 35(3) A1604–A1620 (2013).
- [3] H. Xie, R. Zhang, H. Brunner, Collocation methods for general Volterra functional integral equations with vanishing delays, *SIAM J. Sci. Comput.*, 33(6) 3303–3332 (2011).
- [4] Q. Huang, H. Xie, H. Brunner, Super convergence of a discontinuous Galerkin solutions for delay differential equations of pantograph type, *SIAM J. Sci. Comput.*, 33(5) 2664–2684 (2011).
- [5] K. Harada, E. Nakamae, Application of the Bezier curve to data interpolation, *Computer-Aided Design*, International Journal of Computer Mathematics, 14(1), 55-59(1982).
- [6] G. Nürnberger, F. Zeilfelder, Developments in bivariate spline interpolation, *Journal of Computational and Applied Mathematics*, Computers and Mathematics with Applications, 121(1-2), 125-152(2000).
- [7] J. Zheng, T.W. Sederberg, R.W. Johnson, Least squares methods for solving differential equations using Bezier control points, *Applied Numerical Mathematics*, 48, 237-252(2004).
- [8] F. Ghomanjani, M.H. Farahi, The Bezier control points method for solving delay differential equation, *Intelligent Control and Automation*, 3(2), 188-196(2012).
- [9] F. Ghomanjani, M.H. Farahi, M. Gachpazan, Bezier control points method to solve constrained quadratic optimal control of time varying linear systems, *Computational and Applied Mathematics*, 31(3), 1-24(2012).
- [10] F. Ghomanjani, M.H. Farahi, Optimal control of switched systems based on bezier control points. *Int J Intell Syst Appl* 4(7):16-22 (2012).
- [11] F. Ghomanjani, M.H. Farahi, AV. Kamyad, Numerical solution of some linear optimal control systems with pantograph delays. *IMA J Math Control Inf* (2013).
- [12] L.J. Xie, CL. Zhou, S. Xu, An effective numerical method to solve a class of nonlinear singular boundary value problems using improved differential transform method, *springerplus* 5:1066 (2016).
- [13] H. Laeli Dastjerdi, M. Nili Ahmadabadi, Moving least squares collocation method for Volterra integral equations with proportional delay, *International Journal of Computer Mathematics*, (2017), Doi:10.1080/00207160.2017.1283024.



Sine-Cosine wavelets method for solving Drinfeld- Sokolov system

Naser Azizi^{†43}, Reza Pourgholi[‡]

School of Mathematics and Computer Science, Damghan University, Damghan, Iran. E-mail: a.uni.a.13491349@std.du.ac.ir[†]

School of Mathematics and Computer Science, Damghan University, Damghan, Iran. E-mail: pourgholi@du.ac.ir[‡]

Abstract

In this paper, a numerical method is proposed for the numerical solution of DrinfeldSokolov (DS) system by using the Sine-Cosine wavelets (SCWs) method. We apply the SCWs method for discretizing the space derivatives and report the numerical results which show the efficiency and capability of the proposed method.

AMS subject Classification 2010: 65Nxx, 65T60.

Keywords: DrinfeldSokolov system, Numerical solution, Sine-Cosine wavelets.

1 Introduction

Nonlinear partial differential equations (NPDEs) frequently arise in formulating fundamental laws of nature and in mathematical analysis of a wide variety of problems naturally arising from meteorology, fluid dynamics, plasma physics, ocean and atmospheric waves, mathematical biology, chemistry, material science, etc. Exact solutions of NPDEs play an important role in the proper understanding of qualitative features of many phenomena and processes in the mentioned areas of natural science. Considerable efforts have been made by many mathematicians and physical scientists to obtain exact solutions of such nonlinear evolution equations and a number of powerful and efficient methods, such as trial equation method [1], multiple exp-function method [6], transformed rational function method [2], and etc. have been developed by those scientists.

In this paper, our main goal is to solve numerically the DS system. A generalized form of the DS system is given by:

$$\begin{cases} u_t + (v^2)_x = 0, \\ v_t - av_{xxx} + 3bu_xv + 3dvv_x = 0, \end{cases} \quad (699)$$

⁴³Speaker



where a , b , and d are constants. The exact solution of system (699) by Using the tanh method is the following [5]:

$$\begin{cases} u(x, t) = \frac{c}{2b+d} \tanh^2\left(\sqrt{\frac{c}{2a}}(x - ct)\right), \\ v(x, t) = \frac{c}{\sqrt{2b+d}} \tanh\left(\sqrt{\frac{c}{2a}}(x - ct)\right). \end{cases}$$

The wavelet analysis is the decomposition of a function into shifted and scaled versions of the basic wavelet. The wavelet basis is orthogonal basis for $\mathcal{L}^2(\mathbb{R})$ and is generated by the translation and dilatation of the basic wavelet. In the present paper, we apply the SCWs method for solving the DS system (699) on $\Omega = \Omega_x \times \Omega_t = (0, 1) \times (0, t_f)$, with the initial conditions

$$u(x, 0) = f_1(x), \quad v(x, 0) = f_2(x), \quad x \in \bar{\Omega}_x, \quad (700)$$

and the boundary conditions

$$\begin{aligned} u(0, t) &= g_1(t), & v(0, t) &= g_2(t), & t &\in \bar{\Omega}_t, \\ v(1, t) &= k_2(t), & v_x(0, t) &= w_2(t), & t &\in \bar{\Omega}_t, \end{aligned} \quad (701)$$

where t_f represents the final time, $f_1(x)$, $f_2(x)$, $g_1(t)$, $g_2(t)$, $k_2(t)$, and $w_2(t)$ are differentiable known functions.

The paper is arranged as follows: in Section 36, we describe the SCWs and function approximation. In Section 36, the application and procedure of implementation of the method, are presented. The numerical results are reported in Section 36 and finally, in Section 36, we conclude our work.

2 Sine-Cosine wavelets

Sine-Cosine wavelets are defined on interval $x \in [0, 1)$ as [3]:

$$\psi_{n,m}(x) = \begin{cases} 2^{\frac{k+1}{2}} f_m(2^k x - n), & x \in [\frac{n}{2^k}, \frac{n+1}{2^k}), \\ 0, & \text{elsewhere,} \end{cases} \quad (702)$$

where

$$f_m(x) = \begin{cases} \frac{1}{\sqrt{2}}, & m = 0, \\ \cos(2m\pi x), & m = 1, 2, \dots, L, \\ \sin(2(m-L)\pi x), & m = L+1, L+2, \dots, 2L, \end{cases} \quad (703)$$

and L is any positive integer, $k = 0, 1, 2, \dots$, is the level of resolution, $n = 0, 1, 2, \dots, 2^k - 1$, is the translation parameter, $m = 0, 1, 2, \dots, 2L$ and x is the normalized time. SCWs have compact support and forms a orthonormal basis of $\mathcal{L}^2([0, 1))$.

3 Function approximation

It is known that any integrable function $u(x) \in \mathcal{L}^2([0, 1))$ can be expanded by the SCWs with an infinite number of terms,

$$u(x) = \sum_{m=0}^{\infty} \sum_{n=0}^{2^k-1} c_{n,m} \psi_{n,m}(x), \quad (704)$$



where $c_{n,m} = \langle u, \psi_{n,m} \rangle$, with $\langle \cdot, \cdot \rangle$ denoting the inner product. If the infinite series of equation (704) is truncated, then it can be written as:

$$u(x) = \sum_{m=0}^{2L} \sum_{n=0}^{2^k-1} c_{n,m} \psi_{n,m}(x) = C^T \Psi(x),$$

where C and $\Psi(x)$ are $2^k(2L+1) \times 1$ matrices given by

$$C = \left[c_{0,0}, c_{1,0}, \dots, c_{2^k-1,0}, c_{0,1}, c_{1,1}, \dots, c_{2^k-1,1}, \dots, c_{0,2L}, c_{1,2L}, \dots, c_{2^k-1,2L} \right]^T,$$

$$\Psi(x) = \left[\psi_{0,0}, \psi_{1,0}, \dots, \psi_{2^k-1,0}, \psi_{0,1}, \psi_{1,1}, \dots, \psi_{2^k-1,1}, \dots, \psi_{0,2L}, \psi_{1,2L}, \dots, \psi_{2^k-1,2L} \right]^T. \quad (705)$$

The integration of the vector $\Psi(x)$ defined in (705) can be obtained as

$$\int_0^x \Psi(s) ds = P \Psi(x),$$

where P is $2^k(2L+1) \times 2^k(2L+1)$ operational matrix [4].

4 Overview of the Method

In this section, we present our method for solving DS system (699). Now, Let us divide the interval $[0, t_f]$ into N equal parts of length $\Delta t = \frac{t_f}{N}$ and denote $t_s = (s-1)\Delta t$, $s = 1, 2, \dots, N$. We assume that $\dot{u}'$ and $\dot{v}'''$ can be expanded in terms of SCWs as,

$$\dot{u}'(x, t) \cong \sum_{m=0}^{2L} \sum_{n=0}^{2^k-1} c_{n,m} \psi_{n,m}(x) = C^T \Psi(x), \quad (706)$$

$$\dot{v}'''(x, t) \cong \sum_{m=0}^{2L} \sum_{n=0}^{2^k-1} d_{n,m} \psi_{n,m}(x) = D^T \Psi(x), \quad (707)$$

where $\cdot$ and $'$ mean differentiation with respect to t and x , respectively. By integrating equations (706) once with respect to t from t_s to t and once with respect to x from 0 to x , we obtain

$$u'(x, t) = (t - t_s) C^T \Psi(x) + u'(x, t_s), \quad (708)$$

$$\dot{u}(x, t) = C^T P \Psi(x) + g_1'(t), \quad (709)$$

$$u(x, t) = (t - t_s) C^T P \Psi(x) + [g_1(t) - g_1(t_s)] + u(x, t_s), \quad (710)$$



Now, by integrating equation (707) once with respect to t from t_s to t and three times with respect to x from 0 to x , we obtain

$$v'''(x, t) = (t - t_s)D^T \Psi(x) + v'''(x, t_s), \quad (711)$$

$$v'(x, t) = (t - t_s)D^T P^2 \Psi(x) + v'(x, t_s) + (1 - 2x)[w_2(t) - w_2(t_s)] + 2x[k_2(t) - k_2(t_s)] - 2x[g_2(t) - g_2(t_s)], \quad (712)$$

$$\dot{v}(x, t) = D^T P^3 \Psi(x) + (1 - x^2)g_2'(t) + x(1 - x)w_2'(t) + x^2 k_2'(t), \quad (713)$$

$$v(x, t) = (t - t_s)D^T P^3 \Psi(x) + v(x, t_s) + (1 - x^2)[g_2(t) - g_2(t_s)] + x(1 - x)[w_2(t) - w_2(t_s)] + x^2[k_2(t) - k_2(t_s)]. \quad (714)$$

Discretizing the results by assuming $x \rightarrow x_l$ and $t \rightarrow t_{s+1}$, we have the following scheme:

$$\begin{cases} C^T \Gamma_1 + D^T \Gamma_2 = H_1(x_l, t_s), \\ C^T \Gamma_3 + D^T \Gamma_4 = H_2(x_l, t_s), \end{cases} \quad (715)$$

where $x_l = \frac{2l-1}{2M}$, $l = 1, 2, \dots, M = 2^k(2L + 1)$ are the collocation points for the SCWs and

$$\Gamma_1 = P\Psi(x_l),$$

$$\Gamma_2 = \left[2v'(x_l, t_s)\Delta t P^3 \Psi(x_l) + 2v(x_l, t_s)\Delta t P^2 \Psi(x_l) \right],$$

$$\Gamma_3 = \left[3bv(x_l, t_s)\Delta t \Psi(x_l) + 3dv'(x_l, t_s)\Delta t P \Psi(x_l) \right],$$

$$\Gamma_4 = \left[P^3 \Psi(x_l) - a\Delta t \Psi(x_l) + 3bu'(x_l, t_s)\Delta t P^3 \Psi(x_l) + 3du(x_l, t_s)\Delta t P^2 \Psi(x_l) \right],$$

$$H_1(x_l, t_s) = -2v(x_l, t_s)v'(x_l, t_s) - g_1'(t_{s+1})$$

$$- 2v'(x_l, t_s) \left[(1 - x_l^2)[g_2(t_{s+1}) - g_2(t_s)] + x_l(1 - x_l)[w_2(t_{s+1}) - w_2(t_s)] + x_l^2[k_2(t_{s+1}) - k_2(t_s)] \right]$$

$$- 2v(x_l, t_s) \left[(1 - 2x_l)[w_2(t_{s+1}) - w_2(t_s)] + 2x_l[k_2(t_{s+1}) - k_2(t_s)] - 2x_l[g_2(t_{s+1}) - g_2(t_s)] \right],$$

$$H_2(x_l, t_s) = av'''(x_l, t_s) - 3bv(x_l, t_s)u'(x_l, t_s) - \left[(1 - x_l^2)g_2'(t_{s+1}) + x_l(1 - x_l)w_2'(t_{s+1}) + x_l^2 k_2'(t_{s+1}) \right]$$

$$- 3bu'(x_l, t_s) \left[(1 - x_l^2)[g_2(t_{s+1}) - g_2(t_s)] + x_l(1 - x_l)[w_2(t_{s+1}) - w_2(t_s)] + x_l^2[k_2(t_{s+1}) - k_2(t_s)] \right]$$

$$- 3dv'(x_l, t_s)[g_1(t_{s+1}) - g_1(t_s) + u(x_l, t_s)]$$

$$- 3du(x_l, t_s) \left[(1 - 2x_l)[w_2(t_{s+1}) - w_2(t_s)] + 2x_l[k_2(t_{s+1}) - k_2(t_s)] - 2x_l[g_2(t_{s+1}) - g_2(t_s)] \right].$$

From the equations (715), a system of $2M$ linear equations in the $2M$ unknown coefficients is obtained.

This system can be written in the matrix-vector form as follows:

$$\begin{pmatrix} \Gamma_1 & | & \Gamma_2 \\ - & - & - \\ \Gamma_3 & | & \Gamma_4 \end{pmatrix}_{2M \times 2M} \begin{pmatrix} C \\ - \\ D \end{pmatrix}_{2M \times 1} = \begin{pmatrix} H_1 \\ - \\ H_2 \end{pmatrix}_{2M \times 1} \quad (716)$$

From (716), the coefficients C and D can be calculated. Finally, putting the calculated coefficients into the equations (710) and (714), we can successively calculate the approximate solutions.



5 Numerical results

In this section, the numerical example is discussed to demonstrate the capability, consistency, and efficiency of the presented method which described in section 36.

Example 0.117. In this example, we solve DS system (699) satisfying,

$$\begin{cases} u_t + (v^2)_x = 0, & 0 < x < 1, \quad 0 < t < 1, \\ v_t - v_{xxx} + 3u_x v + 3uv_x = 0, & 0 < x < 1, \quad 0 < t < 1, \end{cases}$$

Comparison among the exact and numerical solutions and difference between the exact and numerical results for $u(x, t)$ and $v(x, t)$, are reported in Table 54 and Figure 42, respectively, when $k = L = 2$ and $\Delta t = 0.01$.

Table 54: Comparison among the exact and numerical solutions for $u(x, t)$ and $v(x, t)$ at $x = 0.525$.

t	$u_{ex}(x, t)$	$u_{nu}(x, t)$	$ u_{ex}(x, t) - u_{nu}(x, t) $	$v_{ex}(x, t)$	$v_{nu}(x, t)$	$ v_{ex}(x, t) - v_{nu}(x, t) $
0.1	0.009058	0.009057	$1.452609e - 06$	0.067300	0.067271	$2.860078e - 05$
0.2	0.007305	0.007304	$1.243188e - 06$	0.060436	0.060387	$4.922403e - 05$
0.3	0.005725	0.005725	$1.868115e - 07$	0.053501	0.053439	$6.249404e - 05$
0.4	0.004325	0.004327	$2.426487e - 06$	0.046501	0.046433	$6.815124e - 05$
0.5	0.003112	0.003117	$5.098826e - 06$	0.039445	0.039379	$6.601905e - 05$
0.6	0.002092	0.002100	$7.868460e - 06$	0.032340	0.032284	$5.600799e - 05$
0.7	0.001270	0.001280	$1.044867e - 05$	0.025195	0.025157	$3.811816e - 05$
0.8	0.000649	0.000662	$1.260746e - 05$	0.018019	0.018006	$1.243997e - 05$
0.9	0.000234	0.000248	$1.417249e - 05$	0.010820	0.010841	$2.084678e - 05$
1	0.000026	0.000041	$1.503480e - 05$	0.003608	0.003670	$6.147569e - 05$

6 Conclusion

The SCWs method has been employed to solve the DS system (699). The numerical results showed that the SCWs method has the best performance. Generally, the obtained numerical solutions by the presented method are in good agreement with the exact solutions. Therefore, it follows that the proposed method is efficient to solve such systems.

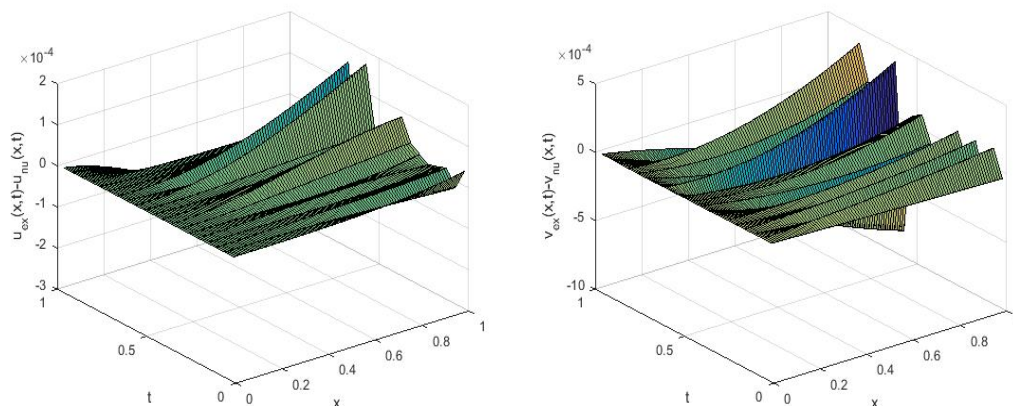


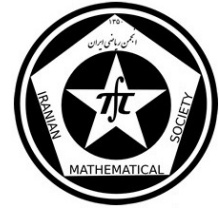
Figure 42: Difference between the exact and numerical results for $u(x, t)$ and $v(x, t)$.

Acknowledgement

The present article was extracted from the thesis written by Naser Azizi and was financially supported by Damghan University.

References

- [1] Y. Gurefe, A. Sonmezoglu, and E. Misirli, *Application of the trial equation method for solving some nonlinear evolution equations arising in mathematical physics*, Pramana, **77** (2011), 1023–1029.
- [2] Z. Huiqun and Ma. Wen-Xiu, *Extended transformed rational function method and applications to complexiton solutions*, Applied Mathematics and Computation, **230** (2014), 509–515.
- [3] I. Nagma and AH. Siddiqi, *Sine-cosine wavelets approach in numerical evaluation of Hankel transform for seismology*, Applied Mathematical Modelling, **40** (2016), 4900–4907.
- [4] M. Tavassoli Kajani, M. Ghasemi, and E. Babolian, *Numerical solution of linear integro-differential equation by using sine-cosine wavelets*, Applied Mathematics and Computation, **180** (2006), 569–574.
- [5] A.M. Wazwaz, *Exact and explicit travelling wave solutions for the nonlinear Drinfeld–Sokolov system*, Communications in Nonlinear Science and Numerical Simulation, **11** (2006), 311–325.
- [6] Ma. Wen-Xiu, H. Tingwen, and Z. Yi, *A multiple exp-function method for nonlinear differential equations and its application*, Physica Scripta, **82** (2010), pages: 065003.



Some optimization methods for solving absolute value equation

Farzad Rahpeymai^{†44}

[†]Department of Mathematics, Payame Noor University, PO BOX 19395-3697, Tehran, Iran
rahpeyma_83@yahoo.com

Abstract

Solving the system of absolute value equation is one of the most important problem in numerical analysis and optimization. In this paper, we present three optimization methods to solve absolute value equation which two methods are based on three-term conjugate subgradient method. Another method is the improved Levenberg-Marquardt algorithm. Numerical experiments show that the three-term spectral conjugate subgradient and improved Levenberg-Marquardt methods are more efficient.

AMS subject Classification 2010: 90C30, 93E24, 34A34.

Keywords: Absolute value equation, Conjugate gradient method, Levenberg-Marquardt approach.

1 Introduction

Consider the system of absolute value equation (AVE)

$$F(x) = Ax - |x| - b = 0, \quad (717)$$

where $A \in \mathbb{R}^{n \times n}$, $b \in \mathbb{R}^n$, and $|x|$ denotes the component-wise absolute value of $x \in \mathbb{R}^n$. It is shown that every solution of the problem (717) is a solution of the following unconstrained optimization problem [5]

$$\begin{aligned} \min \quad & f(x) = \frac{1}{2} \|F(x)\|^2 = \frac{1}{2} \|Qx - b\|^2 \\ \text{s.t.} \quad & x \in \mathbb{R}^n, \end{aligned} \quad (718)$$

where $\|\cdot\|$ denotes the Euclidean norm and $Q := A - \text{diag}(\text{sgn}(x))$ with

$$\text{sgn}(x) = \begin{cases} +1 & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ -1 & \text{if } x < 0. \end{cases}$$

⁴⁴Speaker



Conversely, if x^* is a minimizer of (718) such that $f(x^*) = 0$ then x^* is a solution of (717). Since $F(x)$ is not differentiable, we consider the Clarke generalized Jacobian $\partial|x|$ as [3]

$$D(x) = \partial|x| = \text{diag}(\text{sgn}(x)). \quad (719)$$

The generalized Jacobian of $F(x)$ is defined by $\partial F(x) = A - D(x)$. Now, we rewrite (717) as

$$F(x) = Ax - |x| - b = (A - D(x))x - b = \partial F(x)x - b.$$

Hence, the Clarke generalized gradient of f at x can be stated by

$$\partial f(x) = \partial F(x)^T F(x) = (A - D(x))^T ((A - D(x))x - b).$$

There are many methods for solving AVE such as iterative method [7], semi-smooth Newton method [6] and improved generalized Newton method [4].

2 Three-term conjugate subgradient method

The conjugate gradient (CG) method is to solve unconstrained optimization problems which use very little storage and hence are suitable for very large problems. For the first time, three-term conjugate gradient methods were proposed by Beale [2]. Now, we present three-term conjugate subgradient method (TTCS) to solve AVE. Let

$$\beta_k^1 = 1 + \gamma_k \frac{\partial f_k^T d_{k-1}}{y_{k-1}^T d_{k-1}}, \quad \beta_k^2 = \gamma_k \beta_k^{DY} + \beta_k^{PR}, \quad \beta_k^3 = -\frac{\partial f_k^T d_{k-1}}{\|\partial f_{k-1}\|^2}, \quad (720)$$

and

$$o_k = \partial f_k, \quad p_k = d_{k-1}, \quad q_k = y_{k-1}. \quad (721)$$

where $y_{k-1} = \partial f_k - \partial f_{k-1}$, $\beta_k^{PR} = \frac{\partial f_k^T y_{k-1}}{\|\partial f_{k-1}\|^2}$ and $\beta_k^{DY} = \frac{\|\partial f_k\|^2}{d_{k-1}^T y_{k-1}}$. Also

$$\gamma_k = \frac{\delta}{(1 + 5k)^\varepsilon}, \quad (722)$$

in which $\delta = 10^{-4}$ and $\varepsilon = 0.25$. Then, the new direction is presented by

$$d_k = \begin{cases} -\beta_k^1 o_k + \beta_k^2 p_k + \beta_k^3 q_k, & \text{angleCon1,} \\ -\partial f_k, & \text{otherwise,} \end{cases} \quad (723)$$

where

$$\text{angleCon1} = (k \geq 1 \ \& \ |\partial f_k^T d_k| \geq \Delta \|d_k\| \|\partial f_k\|), \quad (724)$$

with $\Delta \in (0, 1)$. Each iteration of TTCS is constructed by $x_{k+1} = x_k + \alpha_k d_k$ for which the step-size α_k satisfies the following Wolfe conditions:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \alpha_k c_1 \partial f_k^T d_k, \quad (725)$$

$$|\partial f(x_k + \alpha_k d_k)^T d_k| \leq -c_2 \partial f_k^T d_k, \quad (726)$$



where $0 < c_1 < c_2 < 1$. Therefore, the new three-term conjugate subgradient method (TTCS) is presented to solve absolute value equation as follows: New three-term conjugate subgradient algorithm (TTCS)

input: choose $x_0 \in \mathbb{R}^n$, $\Delta \in (0, 1)$, $0 < c_1 < c_2 < 1$, $\epsilon > 0$. compute f_0 and g_0 ; set $k = 0$;
while $\|\partial f_k\| > \epsilon$
if $k > 0$ & **angleCon1**,
obtain $\gamma_k \in (0, 1)$ by (722); set $s_{k-1} = x_k - x_{k-1}$; $y_{k-1} = \partial f_k - \partial f_{k-1}$;
compute β_k^i ($i = 1, 2, 3$) by (720); get o_k, p_k, q_k by (721); compute d_k by (723);
else
set $d_k = -\partial f_k$;
end;
determine **angleCon1** by (724); find α_k satisfying (725) and (726);
get x_{k+1} and compute f_{k+1} and g_{k+1} ; $k \leftarrow k + 1$;
end;
output: x_k and f_k

3 Three-term spectral conjugate subgradient method

In this section, a new three-term conjugate subgradient method (TTSCS) is combined with BB technique for solving (717). We first introduce the BB technique. The step-size of BB strategy using low computation cost is generated by [1]

$$\theta_k = \max \left\{ \theta_{\min}, \min \left\{ \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}, \theta_{\max} \right\} \right\}, \quad (727)$$

where $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = \partial f_k - \partial f_{k-1}$ and $0 < \theta_{\min} < \theta_{\max} < +\infty$. Let

$$\eta_k = 1 + \frac{\partial f_k^T d_{k-1}}{y_{k-1}^T d_{k-1}} \quad \lambda_k = \frac{\|\partial f_k\|^2}{d_{k-1}^T y_{k-1}}. \quad (728)$$

We now present how to construct the new direction by

$$d_k = \begin{cases} -\partial f_k, & \text{if } k = 0, \\ -\eta_k \partial f_k + \lambda_k d_{k-1} + \theta_k \partial f_{k-1}, & \text{if } k \geq 1 \text{ and } \partial f_k^T \partial f_{k-1} \leq 0, \\ -\eta_k \partial f_k + \lambda_k d_{k-1} - \theta_k \partial f_{k-1}, & \text{if } k \geq 1 \text{ and } \partial f_k^T \partial f_{k-1} > 0. \end{cases} \quad (729)$$

The new iteration is generated by $x_{k+1} = x_k + \alpha_k d_k$, where α_k satisfies the strong Wolfe conditions (725)-(726). At this point, the following three-term conjugate subgradient algorithm can be presented to solve AVE. Three-term spectral conjugate subgradient method (TTSCS)

input: choose $x_0 \in \mathbb{R}^n$, $0 < c_1 < c_2 < 1$, $\epsilon > 0$; compute f_0 , ∂f_0 and F_0 and set $k = 0$;
while $\|f_k\| > \epsilon$
compute d_k by (729); determine α_k satisfying in (725) and (726);
get $x_{k+1} = x_k + \alpha_k d_k$; compute f_{k+1} , ∂f_{k+1} and F_{k+1} ; set $k \leftarrow k + 1$;
end;
output: x_k and f_k



4 Levenberg-Marquardt method

In this section, we introduce a new approach to solve absolute value equation (717). Let us define the span of as the set of D all finite linear combination of elements of D , considered as

$$\text{span}(D) = \left\{ \sum_{i=1}^k \lambda_i d_i \mid k \in \mathbb{N}, d_i \in D, \lambda_i \in \mathbb{R} \right\}. \quad (730)$$

Let $\partial f_k = \partial f(x_k)$. Set

$$J_k = \partial F(x_k) = A - D(x_k), \quad J_k^T F_k = \partial f_k, \quad (731)$$

The CG methods for solving AVE combine the steepest descent direction $-\partial f_k$ and the previous direction, i.e., $d_k \in \text{span}\{-\partial f_k, d_{k-1}\}$. Since the steepest descent direction may not lead to good numerical performance, we replace $-\partial f_k$ by the Levenberg-Marquardt direction presented by

$$d_k^{LM} = -\left(J_k^T J_k + \lambda_k^{LM} I\right)^{-1} \partial f_k, \quad (732)$$

in which λ_k^{LM} , called the Levenberg-Marquardt parameter. The new direction is presented by

$$d_k \in \langle d_k^{LM}, d_{k-1}, y_{k-1} \rangle.$$

By introducing

$$\beta_k^4 = \frac{\partial f_k^T y_{k-1}}{\|\partial f_{k-1}\|^2}, \quad \beta_k^5 = \frac{\partial f_k^T d_{k-1}}{\|\partial f_{k-1}\|^2}, \quad (733)$$

and with choosing γ_k as a decreasing sequence such that $\lim_{k \rightarrow \infty} \gamma_k = 0$, we rewrite the new direction as

$$d_k = d_k^{LM} + \gamma_k \left(\beta_k^4 d_{k-1} - \beta_k^5 y_{k-1} \right). \quad (734)$$

Moreover, the new iterates is $x_{k+1} = x_k + \alpha_k d_k$ where the step-size α_k satisfies in Goldstein condition as follows:

$$f(x_k) + (1 - c)\alpha_k \partial f_k^T d_k \leq f(x_k + \alpha_k d_k) \leq f(x_k) + c\alpha_k \partial f_k^T d_k, \quad (735)$$

in which $c \in (0, 1)$. Based on the above information, our algorithm framework is as follows:

Improved Levenberg-Marquardt Algorithm (ILM)

input: choose $x_0 \in \mathbb{R}^n$, $c > 0$ and $\epsilon > 0$; set $k = 0$ and $d_k = -\partial f_k$;

while $\|f_k\| > \epsilon$

determine α_k by (735); set $x_{k+1} = x_k + \alpha_k d_k$; choose λ_{k+1}^{LM} ;

compute J_{k+1} by (731) and obtain d_{k+1}^{LM} by (732); obtain β_{k+1}^4 and β_{k+1}^5 by compute d_{k+1} by (734);

$k \leftarrow k + 1$;

end;

output: x_k and f_k The convergence of TTCS is established in [8] and for other methods is similar to it.



5 Numerical results

In this section, we compare TTCS, TTSCS and ILMN methods for solving absolute value equations (AVE). We generate the total of 12 absolute value equations test problems the Procedure T1. For solving absolute value equation, we use the parameters $\beta = 1.5$ and

$$\lambda_k^{LM} = \|F_k\|^\beta, \quad \gamma_k^{LM} = \frac{\delta}{(1+k)^{0.9}}.$$

Note that, all algorithms are stopped whenever $\|f_k\| < 10^{-7}$ or the total number of iterates exceeds 500. Numerical results are reported in Table 1. In these table, f^* and **sec** are the optimum value of objective function and CPU times. From Table 1, it is clear that TTCSC and ILMN in **sec** and f^* win about %100 and %75, respectively.

Procedure T1

```

A = 1 + rand(n, n);
for i = 1, ..., n;
    A(i, i) = 500;
for i = 1, ..., n;
    for j = 1, ..., n;
        if i > j;
            A(j, i) = A(i, j);
        end
    end
end
b = (A - eye(n)) * ones(n, 1);
x_0 = zeros(n, 1);

```

References

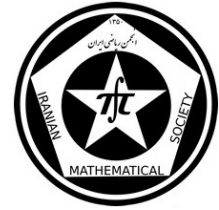
- [1] J. Barzilai, J. M. Borwein, Two point step size gradient method, IMA Journal of Numerical Analysis, 8 (1988) 141–148.
- [2] E.M.L. Beale, A derivative of conjugate gradients, in: F.A. Lootsma (Ed.), Numerical Methods for Nonlinear Optimization, Academic Press, London, (1972) 39–43.
- [3] F.H. Clarke, Optimization and nonsmooth analysis, volume 5 of Classics in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 1990.
- [4] J. Feng, S. Liu, An improved generalized Newton method for absolute value equations, Springer Plus, 5:1042 (2016) 1–10.



Table 55: The comparison TTCS, TTSCS and ILMN methods for absolute value equations

dim	TTCS		TTSCS		ILM	
	sec	f^*	sec	f^*	sec	f^*
100	0.0469	3.4e-11	0.0063	2.6e-11	0.1718	1.7e-11
200	0.0313	9.7e-11	0.0171	3.3e-13	0.1250	9.9e-11
300	0.1250	7.1e-12	0.0383	4.9e-12	0.1406	1.7e-17
400	0.3438	3.1e-11	0.1139	4.1e-11	0.2344	1.43e-16
500	0.5000	7.1e-11	0.1422	6.8e-11	0.3244	1.3e-15
600	0.6406	2.4e-13	0.2409	3.3e-13	0.3281	8.7e-15
700	0.9063	9.1e-13	0.4042	9.2e-13	0.5156	4.1e-14
800	1.1250	2.3e-12	0.5487	2.1e-12	0.7969	2.1e-13
900	1.4844	5.1e-12	0.5955	5.8e-11	1.0000	8.1e-13
1000	2.0313	7.6e-12	0.8464	9.1e-12	1.3125	2.8e-12
1100	2.7969	1.8e-11	0.0067	2.4e-11	1.6719	2.4e-11
1200	2.8750	3.1e-11	1.3457	3.7e-11	4.0156	6.8e-11

- [5] S. Ketabchi, H. Moosaei, An efficient method for optimal correcting of absolute value equations by minimal changes in the right hand side, *Computers & Mathematics with Applications*, 64 (2012) 1882–1885.
- [6] O.L. Mangasarian, A generalized Newton method for absolute value equations, *Optimization Letters*, 3(1) (2009) 101–108.
- [7] O. L. Mangasarian, R. R. Meyer, Absolute value equations, *Linear Algebra Appl*, 419 (2006) 359–367.
- [8] F. Rahpeymaii, K. Amini, T. Allahviranloo, M. Rostamy Malkhalifeh, A new class of conjugate gradient methods for unconstrained smooth optimization and absolute value equations, *Calcolo*, 56(2) (2019) 1–28.



The Crank-Nicolson Type Methods for Fractional Partial Differential Equations

Nayyer Jangi-bahador^{†45}, Safar Irandoust-pakchin[‡]

n.janghii@tabrizu.ac.ir[†]

s.irandoust@tabrizu.ac.ir[‡]

Abstract

The accuracy of the L_1 approximation for the $(1 - \gamma)$ -order Riemann-Liouville fractional derivative is obtained the $\mathcal{O}(\Delta t^{1+\gamma})$ local truncation error. However, the global accuracy is first order due to the backward Euler approximation for the first-order time derivative. In this paper, we adopt the L_1 approximation for the Riemann-Liouville fractional derivative. To obtain the numerical method with global accuracy $\mathcal{O}(\Delta t^{1+\gamma})$ in the temporal direction, we construct the Crank-Nicolson-type scheme. A Crank-Nicolson-type difference scheme is proposed for solving the subdiffusion equation with fractional derivative, and the truncation error is analyzed in detail. It is well known that the Riemann-Liouville fractional derivative of a smooth function may be unbounded at $t = 0$. In order to avoid the difficulty, we use Taylor expansion to obtain the numerical approximation at the first time level. The truncation error of the Crank-Nicolson-type scheme is $\mathcal{O}(\Delta t^{1+\gamma} + \Delta x^2)$.

AMS subject Classification 2010: 35R05, 35R11

Keywords: fractional derivative, subdiffusion equation, finite difference

1 Introduction

Fractional differential equations have attracted considerable interest due to their ability to model many phenomena. They are widely applied in materials and mechanics, signal processing and system identification, anomalous diffusion, control and robotics, wave propagation, turbulence, etc [7, 1, 4].

Derivation of the difference scheme. We consider the following type of time-fractional diffusion equation [6, 8, 9, 2, 11, 3, 5]

$$\begin{cases} \partial_t U = {}_{RL}D_{0,t}^{1-\gamma}(K_\gamma \partial_x^2 U) + f(x, t), & (x, t) \in (a, b) \times (0, T], \\ U(x, 0) = \phi_0(x), & x \in (a, b), \\ U(a, t) = U_a(t), U(b, t) = U_b(t), & t \in (0, T], \end{cases} \quad (736)$$

⁴⁵Speaker



where $0 < \gamma < 1$, K_γ is a positive constant, and $f(x, t)$ and $\phi_0(x)$ are sufficiently smooth functions.

Denote by $I = (a, b)$. Let Δt be the time step size and n_T be a positive integer with $\Delta t = T/n_T$ and $t_n = n\tau$ for $n = 0, 1, \dots, n_T$. Denote by $t_{n+\frac{1}{2}} = (t_n + t_{n+1})/2$ for $n = 0, 1, \dots, n_T - 1$. One can define the space step size $\Delta x = (b - a)/N$, N is a positive integer. The space grid point x_i is given by $x_i = a + i\Delta x$, $i = 0, 1, \dots, N$. Let $x_{i+\frac{1}{2}} = (x_i + x_{i+1})/2$. For the function $U(x, t) \in C(I \times [0, T])$, denote by $U^n = U^n(\cdot) = U(\cdot, t_n)$ and $U_i^n = U(x_i, t_n)$.

Next, we introduce the following notations that will be used in the description of the numerical schemes.

$$\delta_x U_{i+\frac{1}{2}}^n = \frac{U_{i+1}^n - U_i^n}{\Delta x}, \quad \delta_x^2 U_i^n = \frac{U_{i+1}^n - 2U_i^n + U_{i-1}^n}{\Delta x^2}. \quad (1.1)$$

$$\delta_t U_i^{n+\frac{1}{2}} = \frac{U_i^{n+1} - U_i^n}{\Delta t}. \quad (1.2)$$

Next, we list some formulas for the discretization of the Riemann-Liouville derivatives and Caputo derivatives.

- The γ th-order ($0 < \gamma < 1$) Riemann-Liouville derivative of $U(t)$, $t \in (0, T]$ at $t = t_n$ can be discretized by the $L1$ method as

$${}^{L1}_{RL} \delta_t^{(\gamma)} U^n = \frac{1}{\Delta t^\gamma} \left(\sum_{k=0}^{n-1} b_{n-k-1}^{(\gamma)} (U^{k+1} - U^k) + \frac{n^{-\gamma}}{\Gamma(1-\gamma)} U^0 \right), \quad b_k^{(\gamma)} = \frac{(k+1)^{1-\gamma} - k^{1-\gamma}}{\Gamma(2-\gamma)}. \quad (738)$$

- The γ th-order ($0 < \gamma < 1$) Caputo derivative of $U(t)$, $t \in (0, T]$ at $t = t_n$ can be discretized by the $L1$ method as

$${}^{L1}_C \delta_t^{(\gamma)} U^n = \frac{1}{\Delta t^\gamma} \left(\sum_{k=0}^{n-1} b_{n-k-1}^{(\gamma)} (U^{k+1} - U^k) \right), \quad b_k^{(\gamma)} = \frac{(k+1)^{1-\gamma} - k^{1-\gamma}}{\Gamma(2-\gamma)}. \quad (739)$$

2 Main Results

We know that the Crank-Nicolson (CN) method for the classical equation has second-order accuracy in time. The CN method for the classical diffusion equation can be constructed. we can derive the following CN type method

$$\begin{cases} \delta_t u_i^{n-\frac{1}{2}} = \frac{K_\gamma}{2} \left[\delta_t^{(1-\gamma)} \delta_x^2 u_i^{n-1} + \delta_t^{(1-\gamma)} \delta_x^2 u_i^n \right] + f \left(x_i, t_{n+\frac{1}{2}} \right), \\ i = 1, 2, \dots, N-1, \quad n = 1, 2, \dots, n_T \\ u_i^0 = \phi_0(x_i), \quad i = 0, 1, 2, \dots, N \\ u_0^n = U_a(t_n), u_N^n = U_b(t_n) \end{cases} \quad (740)$$

where $\delta_t^{(1-\gamma)}$ is the approximate operator of the time fractional derivative operator ${}^{L1}_{RL} D_{0,t}^{1-\gamma}$.



So we have [10]

$${}_0^{RL}D_t^\alpha u(t) = \sum_{j=0}^{m-1} \frac{u^{(j)}(0)t^{j-\alpha}}{\Gamma(1+j-\alpha)} + \frac{1}{\Gamma(m-\alpha)} \int_0^t \frac{u^{(m)}(s)ds}{(t-s)^{\alpha-m+1}}$$

$$\left[{}_0^{RL}D_t^{(1-\gamma)} \left(\frac{\partial^2 u}{\partial x^2} \right) \right] (x_i, t_n) = \frac{\Delta t^{\gamma-1}}{\Gamma(1+\gamma)} \left[\delta_x^2 U_i^n - \sum_{k=1}^{n-1} (a_{n-k-1} - a_{n-k}) \delta_x^2 u_i^k - a_{n-1} \delta_x^2 u_i^0 \right] + \frac{t_n^{\gamma-1}}{\Gamma(\gamma)} \delta_x^2 u_i^c + R_i^n,$$

subject to $(t_n = n\Delta t)$, $a_n = (n+1)^\gamma - n^\gamma$.

$$\partial_t U_i^{n-1/2} = \frac{1}{2} [({}_0^{RL}D_t^{1-\gamma} \partial_x^2 u)(x_i, t_n) + ({}_0^{RL}D_t^{1-\gamma} \partial_x^2 u)(x_i, t_{n-1})] + f(x_i, t_{n-1/2})$$

Replacing U_i^n by u_i^n and neglecting the truncation error in the above equation:

$$\begin{aligned} \delta_t u_i^{n-1/2} &= k_\gamma \frac{t_n^{\gamma-1} + t_{n-1}^{\gamma-1}}{2\Gamma(\gamma)} \delta_x^2 u_i^0 + \frac{k_\gamma \Delta t^{\gamma-1}}{2\Gamma(1+\gamma)} \left[\delta_x^2 u_i^n - \sum_{k=1}^{n-1} (a_{n-k-1} - a_{n-k}) \delta_x^2 u_i^k - a_{n-1} \delta_x^2 u_i^0 \right] \\ &\quad + \frac{k_\gamma \Delta t^{\gamma-1}}{2\Gamma(1+\gamma)} \left[\delta_x^2 u_i^{n-1} - \sum_{k=1}^{n-2} (a_{n-k-2} - a_{n-k-1}) \delta_x^2 u_i^k - a_{n-2} \delta_x^2 u_i^0 \right] + f_i^{n-1/2} \end{aligned}$$

where $f_i^{n-1/2} = \frac{1}{2} (f_i^n + f_i^{n-1})$ and if $n = 1$:

$$\delta_t u_i^{1/2} = \frac{k_\gamma \Delta t^{\gamma-1}}{\Gamma(\gamma+1)} [\delta_x^2 u_i^1 + (\gamma-1) \delta_x^2 u_i^0] + f_i^1$$

$$u_i^1 = u_i^0 + k_\gamma \frac{\Delta t^\gamma}{\Gamma(1+\gamma)\Delta x^2} [u_{i+1}^1 - 2u_i^1 + u_{i-1}^1] - (1-\gamma) (u_{i+1}^0 - 2u_i^0 + u_{i-1}^0) + \Delta t f_i^1$$

Let $s = \frac{K_\gamma \Delta t^\gamma}{\Gamma(1+\gamma)\Delta x^2}$ and if we continue from $i = 1, \dots, N-1$ then

$$A\mathbf{u}^1 = \mathbf{u}^0 - s(1-\gamma)B\mathbf{u}^0 + \Delta t\mathbf{f}^1$$

where

$$A = \begin{bmatrix} 1+2s & -s & & & \\ -s & 1+2s & -s & & \\ & \ddots & \ddots & \ddots & \\ & & -s & 1+2s & -s \\ & & & -s & 1+2s \end{bmatrix}_{(N-1) \times (N-1)}$$

and



$$B = \begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}_{(N-1) \times (N-1)}$$

Now we have for $n \neq 1$:

$$u_i^n = u_i^{n-1} + s[(u_{i+1}^{n-1/2} - 2u_i^{n-1/2} + u_{i-1}^{n-1/2}) - \sum_{k=1}^{n-1} (a_{n-k-1} - a_{n-k})(u_{i+1}^{k-1/2} - 2u_i^{k-1/2} + u_{i-1}^{k-1/2}) - a_{n-1}(u_{i+1}^0 - 2u_i^0 + u_{i-1}^0)] + k_\gamma \Delta t \frac{t_n^{\gamma-1} + t_{n-1}^{\gamma-1}}{2\Gamma(\gamma)} \delta_x^2 u_i^0 + \Delta t f_i^n$$

if $i = 1$ and $n = n\Delta t$ then

$$u_1^n = u_1^{n-1} + s[(u_2^{n-1/2} - 2u_1^{n-1/2} + u_0^{n-1/2}) - \sum_{k=1}^{n-1} (a_{n-k-1} - a_{n-k})(u_2^{k-1/2} - 2u_1^{k-1/2} + u_0^{k-1/2}) - a_{n-1}(u_2^0 - 2u_1^0 + u_0^0)] + \frac{s}{2}[(\gamma n^{\gamma-1} + \gamma(n-1)^{\gamma-1})(u_2^0 - 2u_1^0 + u_0^0)] + \Delta t f_1^n.$$

and if we continue to $i = N - 1$, it will be discretized in the following form

$$\tilde{A}\mathbf{u}^n = \mathbf{u}^{n-1} + \frac{s}{2}B\mathbf{u}^{n-1} - s \sum_{k=1}^{n-1} (a_{n-k-1} - a_{n-k})B\mathbf{u}^{k-\frac{1}{2}} + s\tilde{a}_{n-1}B\mathbf{u}^0 + \Delta t \mathbf{f}^{n-\frac{1}{2}}$$

where

$$\tilde{A} = \begin{bmatrix} 1+s & -s/2 & & & \\ -s/2 & 1+s & -s/2 & & \\ & \ddots & \ddots & \ddots & \\ & & -s/2 & 1+s & -s/2 \\ & & & -s/2 & 1+s \end{bmatrix}_{(N-1) \times (N-1)}$$

and $\tilde{a}_{n-1} = \frac{1}{2}[\gamma n^{\gamma-1} + \gamma(n-1)^{\gamma-1}] - a_{n-1}$, $a_{n-1} = n^\gamma - (n-1)^\gamma$.

It is known that $\frac{1}{2}([{}_0^{RL}D_t^{1-\gamma}u(t)]_{t=t_{n-1}} + [{}_0^{RL}D_t^{1-\gamma}u(t)]_{t=t_n})$ is not a good approximation to $[{}_0^{RL}D_t^{1-\gamma}u(t)]_{t=t_{n-\frac{1}{2}}}$.

References

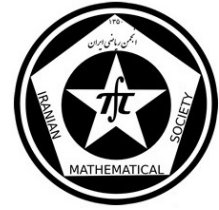
- [1] O.P. Agrawal, J.A.T. Machado, and J. Sabatier, *Introduction (special issue on fractional derivatives and their applications)*, Nonlinear Dyn, **38** (2004), 1–2 .



- [2] C.M. Chen, F. Liu, I. Turner and V. Anh, *A Fourier method for the fractional diffusion equation describing sub-diffusion*, J. Comput. Phys., **227**(2007), 886–897.
- [3] M. Cur, *Compact finite difference method for the fractional diffusion equation*, J. Comput. Phys., **228**(2009), 7792–7804.
- [4] R. Hilfer, ED., *Applications of Fractional Calculus in Physics*, World Scientific, Singapore, 2000.
- [5] Changpin Li and Fanhai Zeng, *Numerical methods for fractional calculus* Vol. 24. CRC Press, 2015.
- [6] R. Metzler and J. Klafter, *The random walk's guide to anomalous diffusion: A fractional dynamics approach*, Phys. Rep, **339**(2000), 1–77.
- [7] I. Podlubny, *Fractional Differential Equations*, Academic Press, Academic Press, New York, 1999.
- [8] S.B. Yuste and L. Acedo, *An explicit finite difference method and a new von Neumann type stability analysis for fractional diffusion equations*, SIAM J. Numer. Anal., **42** (2005), 1862–1874.
- [9] S.B. Yuste, *Weighted average finite difference methods for fractional diffusion equations*, J. Comput. Phys., **216**(2006), 264–274.
- [10] Y.N. Zhang, Z.Z. Sun and H.W. Wu, *Error estimates of CrankNicolson-type difference schemes for the subdiffusion equation* SIAM Journal on Numerical Analysis, 49(2011), 2302–2322.
- [11] P. Zhuang, F. Liu, V. Anh and I. Turner, *New solution and analytical techniques of the implicit numerical method for the anomalous subdiffusion equation*, SIAM J. Numer. Anal., **46**(2008), 1079–1095.

Email: n.janghii@tabrizu.ac.ir

Email: s.irandoust@tabrizu.ac.ir



Tumor Hyperthermia treatment: Effect of cooling period on thermal dose

Neda Biabannavard^{†46}, Mohammad Alinejadmoftad[‡], Mohammad Dehghan Nayyeri[°]

M.Sc. Graduate of Applied Mathematics, University of Bojnord, biabannavardneda@gmail.com[†]

Assistant Professor of Mathematics, University of Bojnord, m.alinejad@ub.ac.ir[‡]

Assistant Professor of Mathematics, University of Bojnord, m.dehghan@ub.ac.ir[°]

Abstract

Hyperthermia is an effective non-invasive way to treat cancer tumor. In this way, heat is generated by an extra-tissue generator such as ultrasound. With the development of hyperthermia for the treatment of cancer tumor, considering and studying effect of cooling period in which accumulation of thermal dose take place is very important in the treatment. In this paper, Hyperthermia treatment of a tumor is represented as an optimal control problem with a PDE constraint namely Pennes bioheat equation. Then the problem is solved numerically by finite difference method and the distribution of temperature and thermal dose in different locations of the tumor is calculated. Results show that in order to achieve the maximum thermal dose, it is better to use the maximum heat in a short time instead of using a constant heat throughout the process.

AMS subject Classification 2010: 46N40, 49J20.

Keywords: Hyperthermia, Thermal dose, Cooling period, Numerical solution of PDEs.

1 Introduction

Cancer means the unnormal growth of cells in the body, and the cancerous area in the body becomes a gland or tumor. In general, cancerous tumors are divided into benign and malignant. A malignant tumor attacks and spreads to its peripheral tissues.

One of the treatments for cancer is heat therapy or hyperthermia, so that the heat generated destroys the cancerous tissue and at the same time does not cause much damage to the healthy surrounding tissues. To kill the tumor, we will have a controlled increase in the temperature of the tumor from 37 to more than 40°C .

⁴⁶Speaker



Optimal control problem of the treatment

Using a time dependent heat generator with uniform distribution, we want to maximize the temperature in center of the tumor at the end of the process, i.e.

$$\max T(x_{center}, t_f). \quad (741)$$

where T is temperature, x_{center} is center of the tumor, t is time, t_f is end of the process. Assume that the total volumetric heat generation is constant during the process, that is,

$$\int_0^{t_f} Q_r(t) dt = Q_{total}, \quad (742)$$

Also, the control function, $Q_r(t)$, cannot be less than the specified minimum value (usually zero) and greater than the specified maximum value. In other words,

$$Q_{min} \leq Q_r(t) \leq Q_{max}. \quad (743)$$

Therefore, the optimal control problem according to the Pennes bioheat equation is as follows

$$\begin{aligned} & \max_{Q_r(t)} T(x_{center}, t_f) \\ \text{s. t. } & \frac{\partial T(x, t)}{\partial t} = \frac{k}{\rho c} \frac{\partial^2 T(x, t)}{\partial x^2} + \frac{w_b \rho_b c_b}{\rho c} (T_a - T(x, t)) + \frac{Q_m}{\rho c} + \frac{Q_r(t)}{\rho c}, \\ & T(x, 0) = T_0(x), \quad 0 \leq x \leq L, \\ & \int_0^{t_f} Q_r(t) dt = Q_{total}, \\ & Q_{min} \leq Q_r(t) \leq Q_{max}, \end{aligned} \quad (744)$$

where k is thermal conductivity, ρ is tissue density, c is specific heat of tissue, w_b is blood perfusion, ρ_b is blood density, c_b is specific heat of blood, T_a is arterial temperature, Q_m is metabolic heat generation and L is thickness of the heated layer of tissue.

Using the Pontryagin minimum principle, we can calculate the optimal $Q_r^*(t)$ for the above problem, which is [1]

$$Q_r^*(t) = \begin{cases} Q_{min}, & 0 \leq t \leq \frac{t_f}{2}, \\ Q_{max}, & \frac{t_f}{2} < t \leq t_f. \end{cases} \quad (745)$$

Now, substituting the $Q_r^*(t)$ in the first constraint of (744) and solving the equation with the second constraint (initial condition), the optimal temperature distribution in the tissue is also obtained.

Thermal dose calculation

The thermal dose at location x is calculated by the following formula

$$D(x) = \frac{1}{60} \int_0^{N \cdot t_f} R^{T_r - T(x, t)} dt, \quad (746)$$



where $T_r = 43^\circ C$ is reference temperature. The formula gives the time in minutes that would take to accumulate the thermal damage that occurs in temperature T . In (746), if $N = 1$, then thermal dose is only accumulated between zero and t_f and any accumulation of thermal dose during the cooling period is neglected. If $N > 1$, then thermal dose accumulation during the cooling period is accounted.

Constant function R in the equation (746) is defined as

$$R = \begin{cases} 0.5, & T(x, t) \geq T_r, \\ 0.25, & T(x, t) < T_r. \end{cases} \quad (747)$$

Numerical solution

Consider the optimal control problem of cancer tumor treatment (744) with the parameters in table 56. Here, we use the explicit finite difference method [3] for solving the first with second constraint of

Table 56: Problem parameter values [2]

Parameter	Value	Units	Parameter	Value	Units
c	3770	$J/kg^\circ C$	ρ	998	kg/m^3
c_b	3770	$J/kg^\circ C$	ρ_b	998	kg/m^3
h	6	$W/m^2^\circ C$	w_b	1000	$\frac{ml \text{ of blood}}{m^3 \text{ of tissue.s}}$
k	0.5	$W/m^\circ C$	t_f	600	s
L	0.01	m	T_r	43	$^\circ C$
Q_m	33800	W/m^3	x_{center}	0.005	m
T_∞	25	$^\circ C$	Q_{max}	220000	W/m^3
T_a	37	$^\circ C$	Q_{min}	0	W/m^3
			Q_{total}	$6.6 * 10^7$	W/m^3

(744) on $[0, L] \times [0, t_f]$ domain. We consider three different control functions including the optimal one and then obtain the temperature distribution and thermal dose for each of these functions for two cases $t_f = 600s$ and ten minutes after that. These functions are

$$Q_r^*(t) = \begin{cases} 0, & 0 \leq t \leq \frac{t_f}{2}, \\ Q_{max}, & \frac{t_f}{2} < t \leq t_f, \end{cases} \quad (748)$$

$$Q_r(t) = \frac{Q_{max}}{2}, \quad 0 \leq t \leq t_f, \quad (749)$$

and

$$Q_r(t) = \begin{cases} Q_{max}, & 0 \leq t \leq \frac{t_f}{2}, \\ 0, & \frac{t_f}{2} < t \leq t_f. \end{cases} \quad (750)$$



2 Main Results

Case 1: Figure 1 shows the temperature distribution (Left shape) and thermal dose (Right shape) at $t_f = 600$ seconds. It is obvious that the maximum temperature occurred for the optimal control function (748) as expected. But the maximum thermal dose occurred for the (750), where the highest energy production rate is used at the beginning of the process. This is because of the accumulation of thermal dose during the cooling period (the second 300 s of the process, when the heat generator is turned off). Therefore, the highest temperature does not necessarily mean the highest thermal dose.

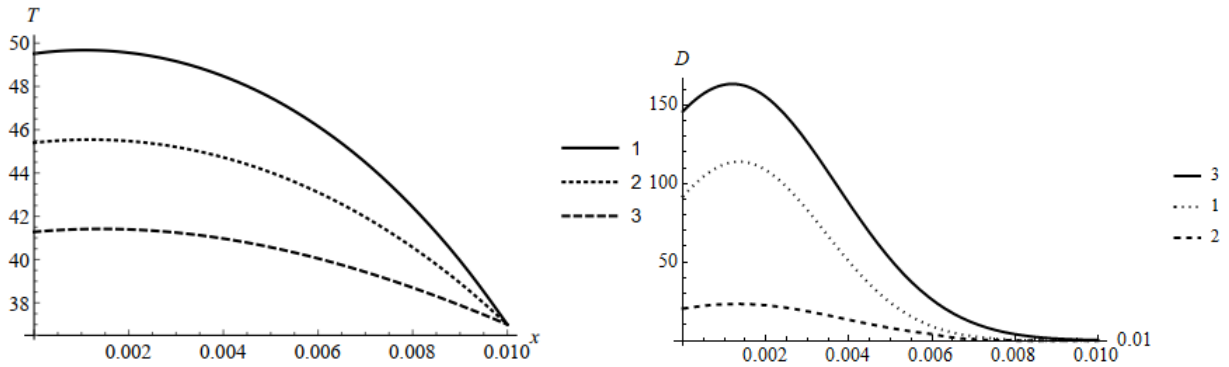


Figure 43: Temperature, $T(^{\circ}C)$ and thermal dose, D (min) in the tissue at $t_f = 600s$.
1:(748), 2:(749), 3:(750)

Case 2: Figure 2 (Left shape) shows the temperature distribution 10 minutes after the end of the process, i.e. at 1200 seconds. In all 3 cases, the temperature has dropped below $39.5^{\circ}C$, which means that no significant thermal dose accumulation occurs any more.

Calculating the thermal dose in this case, we substitute $N = 2$ in the (746) formula. Figure 2 (Right shape) shows the thermal dose. In this shape the lowest thermal dose occurs for a constant rate of energy production (749). Considering (750) and (748), it does not matter whether the maximum energy is at the beginning or the maximum energy at the end of the process because they have the same shapes, indicating the accumulation of thermal dose during cooling.

From this case, we also conclude that to maximize the thermal dose, it is better to use the maximum heat in a short period of time instead of using a constant heat for a long time.

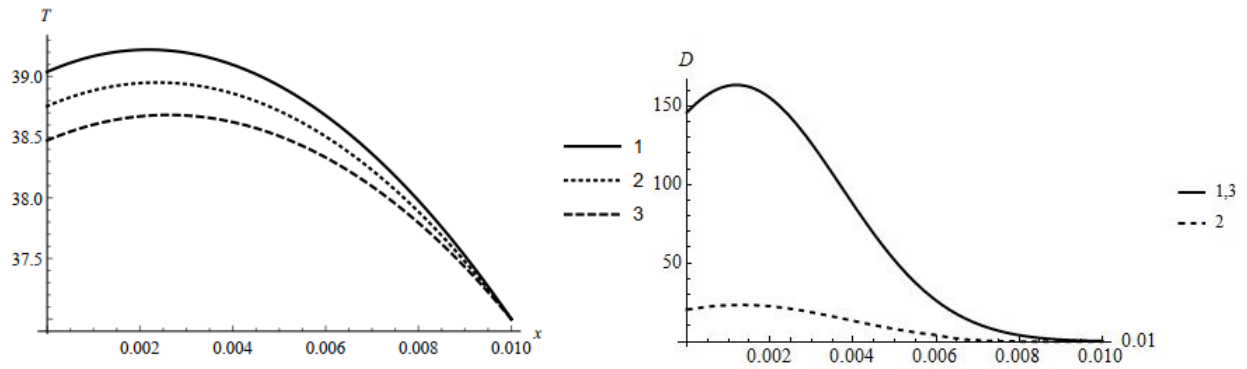


Figure 44: Temperature, $T(^{\circ}C)$ and thermal dose, D (min) in the tissue at $2t_f = 1200s$.
1:(748), 2:(749), 3:(750)

References

- [1] N. Biabannavard, *Optimization problems for bioheat equation*, Master Thesis, Faculty of Basic Sciences, University of Bojnord, 1397.
- [2] A.V. Kuznetsov, *Optimization problems for bioheat equation*, International Communications in Heat and Mass Transfer, **33** (2006), 537-543.
- [3] E. Kreyszig, *Advanced Engineering Mathematics*, Wiley, New York, 2011.



8th Seminar on Numerical Analysis and its Applications

University of Kurdistan, Sanandaj, Iran, April 06-09 2021



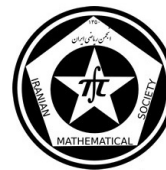
مقالات فارسی

هشتمین سمینار آنالیز عددی

و کاربردهای آن

دانشگاه کردستان – سنندج

۱۷ الی ۲۰ فروردین ۱۴۰۰



برآورد جواب مساله مقدار مرزی غیرخطی ناشی از بینایی سنجی

سمیه مجیدی[†] و رقیه کتانی[‡]

[†] دانشکده علوم پایه، دانشگاه یاسوج، یاسوج، ایران

[‡] دانشکده علوم پایه، دانشگاه یاسوج، یاسوج، ایران

چکیده

در این مقاله یک مدل ریاضی از شکل قرنیه چشم که بصورت مساله مقدار مرزی غیرخطی است، معرفی می‌شود. سپس از روش هم‌مکانی تکه‌ای خطی برای حل عددی معادله انتگرال غیرخطی حاصل از مساله مقدار مرزی استفاده خواهیم نمود و در نهایت مقاله را با تایید عددی نتایج تحلیلی پایان می‌دهیم.

واژه‌های کلیدی: مساله مقدار مرزی غیرخطی، مدل‌بندی قرنیه چشم، معادله انتگرال غیرخطی، روش هم‌مکانی.

طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

حدود دو سوم قدرت انکساری را قرنیه چشم فراهم می‌نماید و بسیاری از اختلالات بینایی از برخی خطاهای هندسه قرنیه ناشی می‌شود. موفقیت جراحی انکساری و لنزهای تماسی بستگی زیادی به دانش توپوگرافی قرنیه و مدل‌های توصیف کننده آن دارد. بنابراین، برای اهداف چشم پزشکی، توصیف دقیق شکل قرنیه بسیار مهم است.

مدل‌های متنوعی برای شکل قرنیه توسط پژوهش‌گران پیشنهاد شده است. دسته‌ای از این مدل‌ها براساس تئوری پوسته ساخته شده‌اند. این مدل‌ها معمولاً پیچیده هستند اما جزئیات فیزیکی زیادی را توصیف می‌کنند [۲]. برخی از مدل‌های کامپیوتری قرنیه براساس روش‌های اجزا محدود ارائه شده‌اند [۱]. در این مقاله مدلی که ابتدا در [۳] ارائه و سپس در [۵] تعمیم یافت را در نظر می‌گیریم. با تبدیل مساله مقدار مرزی بدست آمده در این مدل به یک معادله انتگرال، درصدد حل آن با روش هم‌مکانی تکه‌ای خطی خواهیم بود. در نهایت کارایی روش را با مقایسه نتایج عددی بدست آمده با نتایج ارائه شده در [۴] خواهیم آزمود.



شکل ۱: سطح مقطع قرنيه

۲ مدل ریاضی شکل قرنيه

در شروع فرض می‌کنیم هندسه قرنيه متقارن با شعاع R است. این فرضیه‌ای است که به طور گسترده استفاده می‌شود و تقریب خوبی برای شکل قرنيه فراهم می‌آورد. فرض می‌کنیم قرنيه یک غشاء است به این معنا که خمش نداشته و دارای تنش ثابتی است. در نهایت، تنوع در توزیع ضخامت را نادیده می‌گیریم و بنابراین کافی است فقط یک سطح از قرنيه مدل‌سازی شود.

با توجه به تقارن شعاع، تنها قسمت $h = h(x)[m]$ از قرنيه را در نظر می‌گیریم و برآیند نیروهای وارد بر آن را محاسبه می‌کنیم. همچنین فشار ثابت $P[\frac{N}{m}]$ که در جهت نرمال به برش مقطعی از قرنيه وارد می‌شود را حساب می‌کنیم. تنش $T[N]$ به صورت مماس و نیروهای ارتجاعی متناسب با h عمل می‌کنند (شکل ۱)، نیروهای ارتجاعی به دلیل ویژگی الاستیکی قرنيه برای مدل در نظر گرفته شده‌اند. فرض می‌کنیم $h \in C^2$ ، فاصله x از مرکز تقارن ثابت است و اختلال کوچک Δx را در نظر می‌گیریم. همچنین فرض می‌کنیم که فشار در طول این بخش ثابت است و نیروها در جهت y برآیند می‌شوند (شکل ۱)، در این صورت داریم

$$T \sin(\phi) + T \sin(\psi) + P \cos(\phi) \Delta x - kh \Delta x = 0, \quad (2.1)$$

که در آن k ثابت متناسب با نیروی ارتجاعی است. در ادامه با توجه به شکل و روابط مثلثاتی مربوط به $\sin(x)$, $\cos(x)$ داریم

$$\sin(\phi) = \frac{-h'(x)}{\sqrt{1 + (h'(x))^2}}, \quad \sin(\psi) = \frac{h'(x + \Delta x)}{\sqrt{1 + (h'(x + \Delta x))^2}}, \quad P \cos(\phi) = \frac{P}{\sqrt{1 + (h'(x))^2}},$$

با جایگذاری این روابط در رابطه (۲.۱) و سپس تقسیم دو طرف رابطه بر Δx و حدگیری با استفاده از قاعده هوییتال نتیجه می‌شود

$$\frac{-h''(x)}{(1 + (h'(x))^2)^{\frac{3}{2}}} + kh(x) = \frac{P}{\sqrt{1 + (h'(x))^2}} \quad (2.2)$$

در ادامه چون انحراف سطح قرنيه اندک است، برخی ساده‌سازی‌ها را انجام می‌دهیم به این معنا که می‌توان نوشت

$$\sin(\phi) \approx \tan(\phi) = -h'(x), \quad \sin(\psi) \approx \tan(\psi) = h'(x + \Delta x). \quad (2.3)$$

بنابراین مشابه قبل معادله ساده شده زیر برای توصیف شکل قرنيه حاصل می‌شود

$$-Th''(x) + kh(x) = \frac{P}{\sqrt{1 + (h'(x))^2}} \quad (2.4)$$

منحنی $h(x)$ یک نصف النهار از سطح دورانی است که هندسه قرنيه را توصیف می‌کند. به وضوح می‌خواهیم $h(R) = 0$ باشد چون R شعاع قرنيه است و می‌خواهیم در حاشیه ارتفاع نداشته باشیم. شرط مرزی دومی که تحمیل می‌کنیم، $h'(0) = 0$ است، چون می‌خواهیم یک ارتباط هموار بین نصف النهارها در مبدا و سطح محور تقارن داشته باشیم.



سرانجام با انتخاب متغیرهای جدید $x^* = \frac{x}{R}$, $h^* = \frac{h}{R}$, معادله (۳۴) را به یک معادله بدون بعد تبدیل می‌کنیم. به این ترتیب معادله غیرخطی

$$-h''(x) + ah(x) = \frac{b}{\sqrt{1 + (h'(x))^2}}, \quad h(1) = 0, \quad h'(0) = 0, \quad (35)$$

برای شکل قرینه حاصل می‌شود که در آن $a := \frac{KR^2}{T}$, $b := \frac{PR}{T}$ و همانطور که مشخص است a, b ثابت‌های مثبت هستند.

۳ روش هم مکانی تکه‌ای خطی

در ابتدا مساله مقدار مرزی (۳۵) که حاصل از مدل بندی قرینه می‌باشد را به معادله انتگرال تبدیل می‌کنیم. برای این منظور تعریف می‌کنیم $u(x) := h''(x)$. انتگرال‌گیری از طرفین این رابطه و استفاده از شرط مرزی $h'(0) = 0$ ، نتیجه می‌دهد

$$h'(x) = \int_0^x u(t)dt. \quad (36)$$

با انتگرال‌گیری مجدد داریم

$$h(x) - h(0) = \int_0^x h'(t)dt = \int_0^x \int_0^t u(x_1)dx_1dt = \int_0^x (x-t)u(t)dt,$$

در نظر بگیرید $\alpha := h(0)$ ، در این صورت از شرط مرزی $h(1) = 0$ و رابطه قبل داریم

$$\alpha = - \int_0^1 (1-t)u(t)dt.$$

با جایگذاری روابط بالا، مساله مقدار مرزی (۳۵) به معادله انتگرال

$$-u(x) + a \left(- \int_0^1 (1-t)u(t)dt + \int_0^x (x-t)u(t)dt \right) = \frac{b}{\sqrt{1 + \left(\int_0^x u(t)dt \right)^2}} \quad (37)$$

تبدیل می‌شود که می‌توان آن را به صورت

$$-u(x) + a \int_x^1 (t-1)u(t)dt + a \int_0^x (x-1)u(t)dt = \frac{b}{\sqrt{1 + \left(\int_0^x u(t)dt \right)^2}}, \quad (38)$$

نوشت.

۴ روش عددی

در این بخش از روش هم مکانی تکه‌ای خطی برای حل معادله انتگرال (۳۳) استفاده خواهیم کرد. برای این منظور شبکه یکنواخت از نقاط را بر بازه $[0, 1]$ به صورت $x_j = jh$, $j = 0, 1, \dots, n$ با طول گام $h = \frac{1}{n}$ در نظر می‌گیریم. فرض کنید X_n زیر



فضایی از مجموعه همه توابع تکه‌ای خطی روی $[0, 1]$ در نقاط گره‌ای $\{x_0, \dots, x_n\}$ باشد به این ترتیب عملگر تصویر درونیاب $P_n : X = C(D) \rightarrow X_n$ به صورت $P_n u(x) = \sum_{j=0}^n u(x_j) l_j(x)$ تعریف می‌شود که در آن توابع کلاهی هستند. فرض کنید $u(x_i) \approx u_i$ در این صورت با در نظر گرفتن شرط هم‌مکانی در نقاط شبکه، برای معادله (۳۳) می‌توان نوشت

$$-\sum_{i=0}^n u_i l_i(x_j) + a \int_0^{x_j} (x_j - 1) \sum_{i=0}^n u_i l_i(t) dt + a \int_{x_j}^1 (t - 1) \sum_{i=0}^n u_i l_i(t) dt - \frac{b}{\sqrt{1 + \left(\int_0^{x_j} \sum_{i=0}^n u_i l_i(t) dt \right)^2}} = 0,$$

از خواص توابع کلاهی و با شکستن انتگرال روی زیربازه‌ها داریم

$$-u_j + a(x_j - 1) \sum_{k=0}^{j-1} \int_{x_k}^{x_{k+1}} \sum_{i=0}^n u_i l_i(t) dt + a \sum_{k=j}^{n-1} \int_{x_k}^{x_{k+1}} (t - 1) \sum_{i=0}^n u_i l_i(t) dt - \frac{b}{\sqrt{1 + \left[\sum_{k=0}^{j-1} \int_{x_k}^{x_{k+1}} \sum_{i=0}^n u_i l_i(t) dt \right]^2}} = 0.$$

از آنجایی که برای $x_k \leq t \leq x_{k+1}$ تنها مقادیر $l_k(t)$ و $l_{k+1}(t)$ غیر صفر هستند. می‌توان رابطه بالا را به صورت زیر خلاصه نمود

$$-u_j + a(x_j - 1) \sum_{k=0}^{j-1} \int_{x_k}^{x_{k+1}} (u_k l_k(t) + u_{k+1} l_{k+1}(t)) dt + a \sum_{k=j}^{n-1} \int_{x_k}^{x_{k+1}} (t - 1) (u_k l_k(t) + u_{k+1} l_{k+1}(t)) dt - b / \left[1 + \left(\sum_{k=0}^{j-1} \int_{x_k}^{x_{k+1}} u_k l_k(t) + u_{k+1} l_{k+1}(t) \right)^2 \right]^{\frac{1}{2}} = 0, \quad j = 0, 1, \dots, n.$$

در ادامه با تعریف

$$\begin{aligned} \alpha_k &:= \int_{x_k}^{x_{k+1}} l_k(t) dt = \int_{x_k}^{x_{k+1}} l_{k+1}(t) dt = \frac{1}{h} \left[\frac{x_{k+1}^2}{2} - x_k x_{k+1} + \frac{x_k^2}{2} \right], \\ \beta_k &:= \int_{x_k}^{x_{k+1}} (t - 1) l_k(t) dt = \frac{1}{h} \left[\frac{x_{k+1}^3}{3} - \frac{x_{k+1}^2 x_k}{2} - \frac{x_{k+1}^3}{3} + x_k x_{k+1} + \frac{x_k^3}{6} - \frac{x_k^2}{2} \right], \\ \theta_k &:= \int_{x_k}^{x_{k+1}} (t - 1) l_{k+1}(t) dt = \frac{1}{h} \left[\frac{x_{k+1}^3}{6} - \frac{x_{k+1}^2}{2} - \frac{x_k^3}{6} x_{k+1} - \frac{x_k^2}{2} + x_k x_{k+1} + \frac{x_k^3}{3} \right], \end{aligned}$$

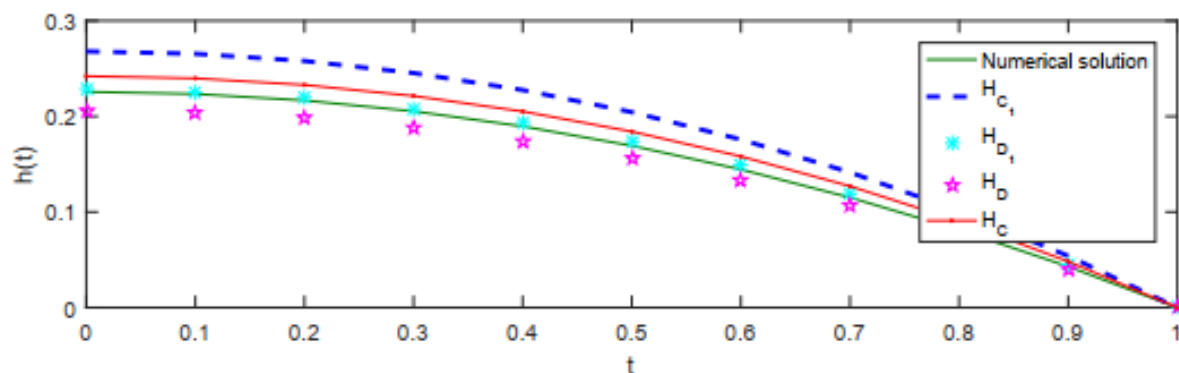
داریم

$$-u_j + a(x_j - 1) \sum_{k=0}^{j-1} \alpha_k [u_k + u_{k+1}] + a \sum_{k=j}^{n-1} [u_k \theta_k + u_{k+1} \beta_k] - b \left[1 + \left(\sum_{k=0}^{j-1} \alpha_k u_k + \beta_k u_{k+1} \right)^2 \right]^{\frac{1}{2}} = 0, \quad j = 0, 1, \dots, n$$

رابطه بالا به ازای $j = 0, 1, \dots, n$ دستگاه معادلات غیرخطی از تابع مجهول $u(t)$ را در نقاط شبکه تشکیل می‌دهد. با حل این دستگاه از روش‌های تکراری مانند نیوتون، مقادیر $u(t_0), u(t_1), \dots, u(t_n)$ حاصل می‌شود.

۵ نتایج عددی

شکل ۲ نتایج عددی حاصل از روش هم مکانی معرفی شده را برای مقادیر پارامترهای $a = 0.216$ و $b = 0.5$ نشان می‌دهد. بعلاوه در این شکل کران‌های بدست آمده در [۴] نیز نشان داده شده‌اند که در آن $D = \frac{r}{a} \left(\frac{\sqrt{a^2 b^2 - 2ab^2 + 1}}{ab} - \frac{1}{ab} + b \right)$, $C = \frac{1 - \sqrt{1 - 2ab^2 + a^2 b^2}}{ab}$ و $D_1 = b\sqrt{1 - b^2}$, $C_1 = b$ تعریف شده و ثابت شده است که $H_{D_1}(t) \leq h(t) \leq H_C(t)$, $H_{D_1}(t) \leq h(t) \leq H_{C_1}(t)$ همانطور که از شکل ۲ مشخص است، تقریب‌های به دست آمده از روش هم مکانی در محدوده کران‌های محاسبه شده توسط نویسندگان [۴] قرار دارند.



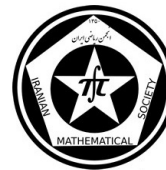
شکل ۲: نتایج عددی



- [1] E. Ahmed, *Finite element modeling of corneal biomechanical behavior*, J. Refractive Surgery 26 (2010) 289–300.
- [2] K. Anderson, A. El-Sheikh, T. Newson, *Application of structural analysis to the mechanical behaviour of the cornea*, J. Royal Society, 1 (2004) 3–15.
- [3] W. Okrański, L. Płociniczak, *A nonlinear mathematical model of the corneal shape*, Nonlin. Anal. RWA 13 (2012), 1498–1505.
- [4] W. Okrański, L. Płociniczak, *Solution estimates for a system of nonlinear integral equations arising in optometry, journal of integral equations and applications*, 1 (2018), 167-179.
- [5] L. Płociniczak, W. Okrański, et al., *On a nonlinear boundary value problem modeling corneal shape*, J. Math. Anal. Appl. 414 (2014), 461–471.

پست الکترونیکی: s.majidifar74@gmail.com

پست الکترونیکی: katani@yu.ac.ir



برآورد نوسانات تصادفی بازارهای مالی با استفاده از مدل ارنشتین آهلنبرگ با نویز لوی

مینا محمدی[†] و پریسا نباتی[‡]

کارشناسی ارشد رشته ریاضی کاربردی، دانشگاه صنعتی ارومیه، ایران[†]

استادیار گروه ریاضی کاربردی، دانشگاه صنعتی ارومیه، ایران[‡]

چکیده

پیش بینی بازارهای مالی همواره برای فعالین اقتصادی حائز اهمیت بوده است. هدف اصلی این مقاله ارائه یک مدل توسعه یافته جدید برای مدل سازی بازارهای مالی با استفاده از فرایند ارنشتین آهلنبرگ ترکیبی با نویز لوی می باشد. مدل های آماری زیادی برای پیش بینی سری های زمانی وجود دارد. در این مقاله یک روش برای برآورد پارامترهای فرایند تصادفی بررسی می شود بطوریکه سری زمانی مالی توسط مدل ارنشتین آهلنبرگ مدل سازی می شود. در سری زمانی مالی ناپایداری تغییرات سود و یا بازده نقش مهمی در تصمیم گیری اقتصادی دارد که در اینجا با استفاده از داده های قیمت بسته شدن روزانه بازار سهام سیمان ارومیه، یک تکنیک برای تخمین سری زمانی مالی بیان شده است. نتایج عددی نشان می دهد که نوسان پیش بینی شده این شرکت نزدیک به نوسان شبیه سازی شده می باشد که در آن دینامیک نوسان از یک مدل خودهمبسته پیروی می کند. مزیت این روش این است که برآوردهای به دست آمده در اطراف مقدار واقعی پایدار هستند، لذا الگوریتم تخمین برای مجموعه داده های بزرگ امکان پذیر است و از خصوصیات همگرایی خوبی برخوردار می باشد. واژه های کلیدی: سری زمانی مالی، مدل ارنشتین آهلنبرگ، نوسانات تصادفی.

طبقه بندی موضوعی [۲۰۱۰]: 62M04, 62G20, 62F9

۱ مقدمه

بازار مالی برای افراد همیشه به عنوان ابزاری برای افزایش ثروت جذاب بوده و کسانی که پیش بینی های دقیقی می کنند از آن سود می برند. پیش بینی مدل های خطی برای محاسبه بسیار ساده است، با این حال، آن ها تقریبی از روابط تحت اختیار در داده ها را نشان می دهند که منجر به پیش بینی های ضعیف می شود. یکی از مشهورترین فرایندهایی که اخیراً از آن در مدل های مالی بسیار استفاده شده است فرایند لوی می باشد که دارای نمونه های مستقل و مانا است. مدل های تصادفی استخراج شده از فرایند لوی توسط بارندرف و شفارد (۲۰۰۰) ارائه شده است. در اینجا ما یک ترکیب از فرایند لوی و فرایند ارنشتین آهلنبرگ را ارائه می دهیم. فرایند ارنشتین آهلنبرگ، یک فرایند



گاوس-مارکف (فرایندی با مشخصات گاوسی و مارکوف) ایستا بوده و تنها فرایند غیر جزئی است که سه شرط فرایندهای گاوس-مارکوف را در نظر می‌گیرد و تبدیلات خطی فضا و زمان را ممکن می‌سازد. در طول زمان این فرایند به میانگین طولانی مدت خود میل می‌کند لذا به این فرایندها، فرایند بازگشت به میانگین می‌گویند. با استفاده از قیمت بسته شده بازارهای سهام (که در آن داده‌های واقعی بازار از سایت www.fipiran.ir گرفته می‌شود) نتیجه می‌گیریم مدل تصادفی ارنشتین آهلنبرگ با پارامترهای وابسته به زمان به طور قابل توجهی عملکرد پیش‌بینی قیمت سهام را بهبود می‌بخشد. ما از یک معادله دیفرانسیل تصادفی که ترکیبی از دو فرایند مستقل ارنشتین آهلنبرگ است برای شبیه‌سازی داده‌های سری زمانی استفاده می‌کنیم. در ادامه نتایج عددی و جدول ارائه می‌شود [۳۱].

۲ مدل‌های نوسان تصادفی

مدل‌های نوسان تصادفی زمانی که واریانس فرایند تصادفی خود به صورت تصادفی توزیع شده است بسیار کارآمد هستند. آن‌ها در ناحیه‌ای از ریاضیات مالی برای ارزیابی اختیارات، بازده و قیمت‌های بسته شده بازار سهام مورد استفاده قرار می‌گیرند. تفاوت اصلی میان ARCH/GARCH و مدل‌های نوسان تصادفی این است که مدل‌های ARCH/GARCH نمی‌توانند نویز تصادفی مربوط به ساختار نوسان را در نظر بگیرند. به عنوان مثال مدل $GARCH(1,1)$ برای قیمت سهام S_t می‌تواند به صورت حاصل ضرب دو فرایند بیان شود:

$$S_t = \sigma_t \epsilon_t \quad (2.1)$$

که در آن σ_t نوسان قیمت‌ها و $\{\epsilon_t\}_t \in \mathbb{N}$ دنباله‌ای از نویز سفید گاوسی، مستقل از $\{S_t\}_t \in \mathbb{N}$ و σ_t است. معادله (۲.۱) می‌تواند به صورت (۲.۲) نوشته شود که در آن معادله حالت، یک فرمول بازگشتی برای حالت σ_t است.

$$\sigma_t = \psi + \psi S_{t-} + v \sigma_{t-} \quad (2.2)$$

به طوری که $\psi \geq \psi + \psi < \psi$. همچنین برای هر مقدار $\sigma_t > \sigma_t$ با گرفتن لگاریتم از مربع قیمت‌های بسته شده S_t به دست می‌آید:

$$\log S_t = \log \sigma_t + \log \epsilon_t. \quad (2.3)$$

آن گاه:

$$y_t = h_t + \log \epsilon_t. \quad (2.4)$$

که در آن $S_t = \log S_t$ و $y_t = \log \sigma_t$ و $h_t = \log \sigma_t$ معادله (۲.۴) معادله مشاهدات و واریانس تصادفی h_t فرایند حالت غیر مشاهده است در نظر گرفته می‌شود و اتورگرسیو آن به صورت زیر می‌باشد

$$h_t = \phi + \phi h_{t-} + \omega_t \quad (2.5)$$



که در آن ω_t نویز سفید گاوسی با واریانس σ_ω است. معادله (۲۴) و (۲۵) تشکیل دهنده مدل نوسان تصادفی توسط تیلور می باشد. سری زمانی مالی در مطالعه‌ی ما تماماً گاوسی نیست، زیرا گاهی اوقات سری دارای چوله و کشیدگی می باشد. بنابراین، معمولاً فرض نرمال بودن ARCH را روی ϵ_t نگه می داریم. هدف ما تخمین پارامترهای $\phi, \phi, \sigma_\omega$ در ϵ_t و سپس پیش بینی مشاهدات بعدی y_{n+m} از n مشاهدات قبلی است.

۳ تخمین پارامترها

در این بخش تخمین پارامترهای توزیع $\Gamma(a, b)$ و λ را از معادله (۳۱) برآورد می کنیم.

$$s_t = wse^{-\lambda t} + \int^t we^{-\lambda(t-q)} dz(\lambda q) + wse^{-\lambda t} + \int^t we^{-\lambda(t-q)} dz(\lambda q). \quad (31)$$

قبل از تخمین، یک اندازه لوی را بیان می کنیم. اگر F_z اندازه لوی باشد فرض می کنیم که یک ثابت مانند $M >$ وجود داشته باشد بطوریکه

$$\int_{|x|>} e^{vx} F_z(dx) < \infty, \quad \text{for all } |v| \leq M. \quad (32)$$

گزاره زیر گشتاور نظری $Z()$ را با گشتاور نظری توزیع مانا $\{S_t\}_{t \geq}$ مرتبط می کند.

گزاره ۳۱. فرض کنید $\{Z_t\}_{t \geq}$ یک فرایند لوی باشد به طوری

$$Var[Z()] = \sigma < \infty, \quad \mathbb{E}[Z()] = \mu < \infty.$$

گیریم M یک ثابت بزرگ باشد که در معادله (۳۲) صدق می کند و $\lambda, \lambda >$ آن گاه درستی عبارتهای زیر را داریم [۴]:

$$Var[S] = \sigma \quad (\text{ب}) \quad \mathbb{E}[S] = \mu \quad (\text{الف})$$

از گزاره ۳۱ پارامترهای a و b مرتبط با μ و σ بصورت زیر می باشند:

$$a = \frac{\mu}{\sigma}, \quad b = \frac{\mu}{\sigma}. \quad (33)$$

۴ تخمین پارامتر شدت

به منظور برآورد پارامتر شدت λ با تعریف تابع خودهمبستگی برای فرایند داده شده در معادله (۳۱) که به فرم زیر است شروع می کنیم.

$$\rho(q) = we^{-\lambda|q|} + we^{-\lambda|q|}. \quad (41)$$

از تابع خود همبسته بیان شده در (۴۱) می توانیم پارامتر λ را تخمین بزنیم. فرض کنید در معادله (۴۱) داشته باشیم $\lambda = \lambda$ خواهیم داشت:

$$\rho(q) = we^{-\lambda|q|} + we^{-\lambda|q|} = e^{-\lambda|q|}(w + w) \quad (42)$$

$$\lambda = -\frac{\log \rho(q - \log(w + w))}{q}.$$

بدون کاستن از کلیت با فرض $q =$ تخمین پارامتر λ به صورت زیر است:

$$\hat{\lambda} = -\log \rho() - \log(w + w). \quad (43)$$

۵ نتایج عددی و نتیجه گیری

در این مطالعه قیمت‌های بسته شدن روزانه شاخص سیمان ارومیه از سال‌های ۱۳۹۴-۱۳۹۷ در نظر گرفته شده است. از میان ارقام مشاهده می‌کنیم که نوسانات داده‌ها ثابت نیست و با گذشت زمان تغییر می‌کند، که با استفاده از نرم افزار متلب به شبیه‌سازی داده‌ها می‌پردازیم.

جدول ۱. نتایج شبیه‌سازی شاخص سیمان ارومیه

RMSE	w_1	w_2	λ_1	λ_2	بورس سهام
۵/۷۳۵۱	۰/۷۷۱۳	۰/۲۲۸۶	۱/۲۳۲۸	۰/۰۰۹۹	سیمان ارومیه

از نتایج به دست آمده مشاهده می‌شود تخمین‌های مدل نوسانات تصادفی روی مقادیر واقعی پایدار بوده و از خصوصیات همگرایی خوبی برخوردار است و خطای کم نشان می‌دهد الگوریتم تخمین مورد استفاده برای داده‌های بزرگ بسیار دقیق و امکان پذیر می‌باشد. حتی اگر نویز واقعی مشاهده شده دقیقاً توزیع ترکیبی نرمال نباشد. نمودار داده‌های شبیه‌سازی شده بسیار متناسب با سری زمانی اصلی است. همچنین داده‌های شبیه‌سازی شده رفتار حافظه طولانی مدت را نشان می‌دهند که پیش‌بینی را تسهیل می‌کند. می‌توان لگاریتم نوسان‌های تصادفی شبیه‌سازی شده و پیش‌بینی شده یک گام رویه جلو، برای شاخص‌ها را محاسبه و ترسیم کرد. نتایج عددی نشان دادند برآوردهای بدست آمده در اطراف مقدار واقعی پایدار هستند و همچنین خطای کم نشان می‌دهد که روش تخمین دقیق است و می‌توان از آن برای پیش‌بینی قیمت سهام‌های آتی، تجزیه و تحلیل بازار، طراحی نمونه کارها در سایر رشته‌ها چون ژئوفیزیک، پزشکی و علوم اجتماعی استفاده نمود [۶۵].

جدول ۲. نتایج آماری شاخص سیمان ارومیه

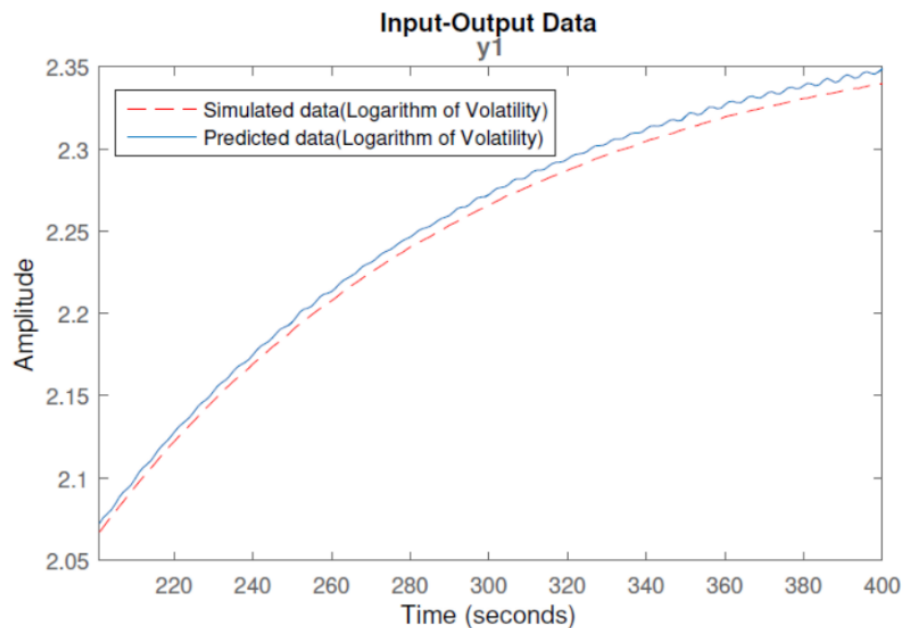


خطای استاندارد برآورد		
ϕ_0	۰/۸۱۵۷	۰/۰۴۴۳۷
ϕ_1	۰/۹۲۵۹	۰/۰۰۱۴
σ_0	۷/۰۲۷۷	۰/۰۰۰۷۱
σ_1	۰/۳۳۵۹	۰/۰۰۰۱
σ_w	۰/۰۰۳۸	۸/۳۴۸۷
ψ	۱/۵۷۲۷	۰/۰۶۰۰
μ_1	-۱/۲۱۰۸	۰/۰۰۰۸

شکل ۱. مقایسه داده‌های شبیه‌سازی شده و داده‌های واقعی ارزش شاخص سیمان ارومیه از سال ۱۳۹۴-۱۳۹۷



شکل ۲. مقایسه لگاریتم نوسان تصادفی شبیه‌سازی شده و پیش‌بینی شده یک گام رو به جلو برای شاخص سیمان ارومیه.

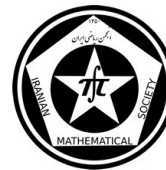


مراجع

1. J.L. Doob, The Brownian Movement and Stochastic Equations, Annals of Mathematics. (1942), 43(2), pp. 351-369.
2. <http://www.fipiran.com/DataService/TradeIndex>.
3. C.M. Mariani, O. K. Tweneboah, Estimation of stochastic volatility by using Ornstein–Uhlenbeck type models, Physica A: Statistical Mechanics and its Applications, (2012), 421, pp.167-176.
4. K. Spiliopoulos, Method of Moments Estimation of Ornstein-Uhlenbeck Processes Driven by General Levy Process, ANNALES de l’I.S.U.P., (2009), 53, pp. 3-19.
5. J. Li, L. Liu, J. W. Wang, The financial time series forecasting based on proposed ARIMAGRNN model, conf. on machine learning and cybernetics, (2005).
6. G. Zhang, Time series forecasting using a hybrid ARIMA and neural network model, Neurocomputing, (2004), 50, pp. 159-175.

پست الکترونیکی: minamohammadi118@gmail.com

پست الکترونیکی: p.nabati@uut.ac.ir



تحقق انتهای طیف‌ها به وسیله‌ی ماتریس‌های سه‌قطری حقیقی متقارن

محمد جواد فاضلی[†] و مهدی پناهی[‡]

گروه ریاضی، دانشکده علوم ریاضی و آمار، دانشگاه بیرجند، بیرجند، ایران

چکیده

مسئله‌ی ساختن یک ماتریس سه‌قطری حقیقی متقارن A ، از روی نوع خاصی از این اطلاعات طیفی مشخص که، $\lambda^{(j)}$ یک مقدارویژه‌ی زیر ماتریس اصلی پیشرو $j \times j$ ، A_j ، $j = 1, 2, \dots, n$ ، از A بوده و $(\lambda^{(n)}, x)$ یک جفت‌ویژه‌ی A باشد، را در نظر می‌گیریم. شرایط کافی برای وجود ماتریس A را ارائه می‌دهیم. همچنین شرایط کافی برای این که ماتریس ساخته شده، نامنفی باشد را بیان می‌کنیم. نتایج ما ساختاری هستند و یک روش الگوریتمی سودمند برای ساخت چنین ماتریسی را فراهم می‌کنند. واژه‌های کلیدی: ماتریس‌های سه‌قطری حقیقی متقارن، مقدارویژه. طبقه‌بندی موضوعی [۲۰۱۰]: 65F15, 65F18, 15A18.

۱ پیش‌گفتار

فرض کنید $A = [a_{ij}]$ یک ماتریس مربعی مرتبه‌ی n باشد. ترانهادی A را با $A^T = [a_{ji}]$ ، مزدوج A را با $\bar{A} = [\bar{a}_{ij}]$ و ترانهادی هرمیتی^۱ A را با $A^H = \bar{A}^T = [\bar{a}_{ji}]$ نشان می‌دهیم. A هرمیتی نامیده می‌شود هرگاه $A = A^H$. واضح است که اگر A حقیقی باشد آن‌گاه A هرمیتی است هرگاه $A = A^T$. A سه‌قطری نامیده می‌شود هرگاه $a_{ij} = 0$ برای $|i - j| > 1$.

^۱Hermitian

ماتریس سه قطری^۲ حقیقی متقارن به شکل

$$A = \begin{bmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & \ddots & \\ & & \ddots & \ddots & b_{n-1} \\ & & & b_{n-1} & a_n \end{bmatrix}, \quad (1,1)$$

را در نظر می‌گیریم. ساخت یک ماتریس خاص با خصوصیات طیفی^۳ معین، که این داده‌های طیفی ممکن است شامل اطلاعات کامل و یا جزئی از مقادیر ویژه یا بردارهای ویژه باشند، به مسأله‌ی مقدار ویژهی معکوس موسوم است. این مسائل کاربردهای زیادی در نظریه‌ی کنترل، ژئوفیزیک، مکانیک، مسأله‌ی اشتورم - لیوویل^۴ و ... دارند ([۱] و [۳]). در این مقاله یک مسأله‌ی مقدار ویژهی معکوس را مورد مطالعه قرار می‌دهیم. این مسأله در مورد ساختن یک ماتریس $n \times n$ سه قطری حقیقی متقارن خاص به شکل (۱,۱) می‌باشد به نحوی که در شرایط طیفی معینی صدق کند. n عدد حقیقی $\lambda^{(j)}$ ، $j = 1, 2, \dots, n$ ، و یک بردار حقیقی $x = (x_1, x_2, \dots, x_n)^T$ داده شده‌اند. شرایط لازم و کافی برای وجود ماتریس $n \times n$ سه قطری حقیقی متقارن A به شکل (۱,۱) را بیابید به طوری که $\lambda^{(j)}$ یک مقدار ویژهی زیرماتریس اصلی پیشرو $j \times j$ ، A_j برای $j = 1, 2, \dots, n$ ، از A بوده و $(\lambda^{(n)}, x)$ یک جفت ویژهی A باشد. در سراسر این مقاله فرض کنید A_j زیرماتریس اصلی پیشرو $j \times j$ از A ، $P_j(\lambda)$ چندجمله‌ای مشخصه‌ی A_j ، $\lambda^{(j)}$ یک مقدار ویژهی A_j و $\sigma(A_j)$ نشان‌دهنده‌ی طیف A_j است.

۲. لم‌ها و قضایای پیش‌نیاز

قضیه‌ی زیر اطلاعاتی راجع به مقادیر ویژه‌ی یک ماتریس هرمیتی بیان می‌کند.

قضیه ۱. مقادیر ویژه‌ی یک ماتریس هرمیتی، اعداد حقیقی هستند.

□

اثبات. به [۲] رجوع شود.

از قضیه‌ی فوق به راحتی نتیجه‌ی زیر به دست می‌آید.

نتیجه ۲. مقادیر ویژه‌ی ماتریس‌های حقیقی متقارن، اعداد حقیقی هستند.

لم زیر که یک رابطه‌ی بازگشتی برای محاسبه‌ی چندجمله‌ای مشخصه‌ی زیرماتریس اصلی پیشرو $j \times j$ ، A_j از A ، $j = 1, 2, \dots, n$ ، را ارائه می‌کند، با محاسبه‌ی مستقیم اثبات می‌شود ([۵]).

^۲Tridiagonal matrix

^۳Spectrum

^۴Sturm- Liouville



لم ۳. فرض کنید A یک ماتریس سه قطری حقیقی متقارن $n \times n$ به شکل $(1, 1)$ و $P_j(\lambda)$ چندجمله‌ای مشخصه‌ی زیرماتریس اصلی پیشرو $j \times j$ ، A_j ، برای $j = 1, 2, \dots, n$ ، از A باشد. در این صورت دنباله‌ی $\{P_j(\lambda)\}_{j=1}^n$ ، در رابطه‌ی بازگشتی

$$P_j(\lambda) = (\lambda - a_j)P_{j-1}(\lambda) - b_{j-1}^2 P_{j-2}(\lambda), \quad j = 1, 2, \dots, n,$$

صدق می‌کند که در آن

$$P_0(\lambda) = P_{-1}(\lambda) = 1, \quad b_0 = 0.$$

تعریف ۴. چندجمله‌ای تکین^۵، چندجمله‌ای یک متغیره است، به طوری که ضریب پیشرو (ضریب غیر صفر بزرگترین درجه) آن مساوی ۱ باشد. بنابراین یک چندجمله‌ای تکین درجه‌ی n به شکل

$$x^n + c_{n-1}x^{n-1} + \dots + c_2x^2 + c_1x + c_0,$$

می‌باشد.

لم زیر که در مرجع [۴] بیان شده است در اثبات نتایج اصلی این مقاله کاربرد زیادی دارد.

لم ۵. فرض کنید $P(\lambda)$ یک چندجمله‌ای تکین درجه n با صفرهای حقیقی باشد. اگر λ_1 و λ_n به ترتیب کوچکترین و بزرگترین صفر $P(\lambda)$ باشند، آن‌گاه:

$$1- \text{اگر } \mu < \lambda_1, \text{ آن‌گاه } (-1)^n P(\mu) > 0,$$

$$2- \text{اگر } \mu > \lambda_n, \text{ آن‌گاه } P(\mu) > 0.$$

۳ نتایج اصلی

اکنون نتایج اصلی این مقاله را برای حل مسأله ۱، بیان می‌کنیم.

قضیه ۶. n عدد حقیقی $\lambda^{(1)} < \lambda^{(2)} < \dots < \lambda^{(n)}$ و یک بردار حقیقی $x = (x_1, x_2, \dots, x_n)^T$ داده شده‌اند. در این صورت ماتریس $n \times n$ سه قطری حقیقی متقارن A به شکل $(1, 1)$ با b_i های مثبت موجود است به طوری که $\lambda^{(j)}$ یک مقدارویژه‌ی ماکسیمال^۶ (بزرگترین مقدارویژه از نظر قدرمطلق) زیرماتریس اصلی پیشرو $j \times j$ ، A_j ، برای $j = 1, 2, \dots, n$ ، از A بوده و $(\lambda^{(n)}, x)$ یک جفت‌ویژه‌ی A است اگر شرایط

$$x_i x_{i+1} > 0, \quad i = 1, 2, \dots, n-1, \quad (3, 1)$$

^۵Monic Polynomial

^۶Maximal



و

$$\lambda^{(n)} > a_j + b_{j-1} \frac{x_{j-1}}{x_j}, \quad j = 1, 2, \dots, n-1, \quad (3,2)$$

برقرار باشند.

اثبات. اثبات وجود ماتریس سه قطری حقیقی متقارن $A, n \times n$ به شکل $(1,1)$ با خواص طیفی مطلوب، معادل با این است که نشان دهیم سیستم معادلات

$$P_j(\lambda^{(j)}) = 0, \quad j = 1, 2, \dots, n, \quad (3,3)$$

$$Ax = \lambda^{(n)}x, \quad (3,4)$$

جواب‌های حقیقی a_j و $b_{j-1} > 0$ دارد. رابطه $(3,4)$ را به صورت

$$\left. \begin{aligned} a_1 x_1 + b_1 x_2 &= \lambda^{(n)} x_1 \\ b_1 x_1 + a_2 x_2 + b_2 x_3 &= \lambda^{(n)} x_2 \\ b_2 x_2 + a_3 x_3 + b_3 x_4 &= \lambda^{(n)} x_3 \\ \vdots & \\ b_{j-1} x_{j-1} + a_j x_j + b_j x_{j+1} &= \lambda^{(n)} x_j \\ \vdots & \\ b_{n-2} x_{n-2} + a_{n-1} x_{n-1} + b_{n-1} x_n &= \lambda^{(n)} x_{n-1} \\ b_{n-1} x_{n-1} + a_n x_n &= \lambda^{(n)} x_n \end{aligned} \right\}, \quad (3,5)$$

نیز می‌توان نوشت. از $(3,1)$ ، $(3,3)$ و $(3,5)$ ، واضح است که $a_1 = \lambda^{(1)}$ ،

$$b_1 = (\lambda^{(n)} - a_1) \frac{x_1}{x_2} > 0,$$

و

$$b_j = (\lambda^{(n)} - a_j) \frac{x_j}{x_{j+1}} - b_{j-1} \frac{x_{j-1}}{x_{j+1}}, \quad j = 2, 3, \dots, n-1. \quad (3,6)$$

از $(3,2)$ ، داریم

$$(\lambda^{(n)} - a_j) \frac{x_{j+1}}{x_j} > b_{j-1} \frac{x_{j-1}}{x_j},$$

$$(\lambda^{(n)} - a_j) \frac{x_j}{x_{j+1}} > b_{j-1} \frac{x_{j-1}}{x_{j+1}}.$$



بنابراین $b_j > 0$, $j = 2, 3, \dots, n-1$, همچنین از (۳,۳)، داریم

$$P_j(\lambda^{(j)}) = (\lambda^{(j)} - a_j)P_{j-1}(\lambda^{(j)}) - b_{j-1}^* P_{j-2}(\lambda^{(j)}) = 0, \quad j = 2, 3, \dots, n,$$

و

$$a_j = \lambda^{(j)} - b_{j-1}^* \frac{P_{j-2}(\lambda^{(j)})}{P_{j-1}(\lambda^{(j)})}, \quad j = 2, 3, \dots, n.$$

از طرفی از لم (۵)، نتیجه می‌شود که $P_{j-1}(\lambda^{(j)}) > 0$ برای نشان دادن برقراری آخرین معادله در (۳,۵)، ابتدا نشان می‌دهیم که

$$b_j = \frac{x_j}{x_{j+1}} \frac{P_j(\lambda^{(n)})}{P_{j-1}(\lambda^{(n)})}, \quad j = 1, 2, \dots, n-1, \quad (3,7)$$

که از لم (۵)، $P_{j-1}(\lambda^{(n)}) > 0$

برای $j = 1$ داریم

$$b_1 = \frac{x_1}{x_2} (\lambda^{(n)} - a_1) = \frac{x_1}{x_2} \frac{P_1(\lambda^{(n)})}{P_0(\lambda^{(n)})}.$$

حال فرض کنید که (۳,۷)، برای $j = k$ صحیح باشد، یعنی

$$b_k = \frac{x_k}{x_{k+1}} \frac{P_k(\lambda^{(n)})}{P_{k-1}(\lambda^{(n)})}.$$

از (۳,۶)، داریم

$$\begin{aligned} b_{k+1} &= (\lambda^{(n)} - a_{k+1}) \frac{x_{k+1}}{x_{k+2}} - b_k \frac{x_k}{x_{k+2}} \\ &= (\lambda^{(n)} - a_{k+1}) \frac{x_{k+1}}{x_{k+2}} - b_k \frac{x_k}{x_{k+2}} \left[\frac{x_k}{x_{k+1}} \frac{P_k(\lambda^{(n)})}{P_{k-1}(\lambda^{(n)})} \frac{x_{k+1}}{x_k} \frac{P_{k-1}(\lambda^{(n)})}{P_k(\lambda^{(n)})} \right] \\ &= (\lambda^{(n)} - a_{k+1}) \frac{x_{k+1}}{x_{k+2}} - b_k \frac{x_k}{x_{k+2}} \left[b_k \frac{x_{k+1}}{x_k} \frac{P_{k-1}(\lambda^{(n)})}{P_k(\lambda^{(n)})} \right] \\ &= \frac{x_{k+1}}{x_{k+2}} \left[(\lambda^{(n)} - a_{k+1}) - b_k^* \frac{P_{k-1}(\lambda^{(n)})}{P_k(\lambda^{(n)})} \right] \\ &= \frac{x_{k+1}}{x_{k+2}} \left[\frac{(\lambda^{(n)} - a_{k+1}) P_k(\lambda^{(n)}) - b_k^* P_{k-1}(\lambda^{(n)})}{P_k(\lambda^{(n)})} \right] \\ &= \frac{x_{k+1}}{x_{k+2}} \frac{P_{k+1}(\lambda^{(n)})}{P_k(\lambda^{(n)})}. \end{aligned}$$

بنابراین (۳,۷)، برقرار است و

$$x_{n-1} = b_{n-1}^* \frac{P_{n-2}(\lambda^{(n)})}{P_{n-1}(\lambda^{(n)})} x_n.$$



حال داریم

$$\begin{aligned} b_{n-1}x_{n-1} + a_n x_n &= b_{n-1}x_{n-1} + [(\lambda^{(n)}) - b_{n-1}^* \frac{P_{n-2}(\lambda^{(n)})}{P_{n-1}(\lambda^{(n)})}]x_n \\ &= b_{n-1}x_{n-1} + \lambda^{(n)}x_n - b_{n-1}[b_{n-1}^* \frac{P_{n-2}(\lambda^{(n)})}{P_{n-1}(\lambda^{(n)})}]x_n \\ &= b_{n-1}x_{n-1} + \lambda^{(n)}x_n - b_{n-1}x_{n-1} = \lambda^{(n)}x_n, \end{aligned}$$

□ لذا آخرین معادله (۳,۵)، نیز برقرار می‌باشد و اثبات کامل می‌شود.

نکته ۷. نتایج قضیه (۶)، برقرار است هرگاه $\lambda^{(j)}$ را مقدارویژه‌ی مینیمال^۷ (کوچکترین مقدارویژه از نظر قدرمطلق) A_j در نظر بگیریم، در این حالت باید داشته باشیم

$$\lambda^{(n)} < \dots < \lambda^{(2)} < \lambda^{(1)},$$

و شرط (۳,۱) به صورت $x_i x_{i+1} < 0$ ، $i = 1, 2, \dots, n-1$ ، تبدیل می‌شود. اثبات این حالت دقیقاً مشابه اثبات قضیه (۶)، می‌باشد.

نتیجه ۸. n عدد حقیقی $\lambda^{(1)} < \lambda^{(2)} < \dots < \lambda^{(n)}$ و یک بردار حقیقی $x = (x_1, x_2, \dots, x_n)^T$ داده شده‌اند. اگر شرایط

$$x_i x_{i+1} > 0, \quad i = 1, 2, \dots, n-1, \quad (3,8)$$

$$\lambda^{(n)} > a_j + b_{j-1} \frac{x_{j-1}}{x_j}, \quad j = 2, 3, \dots, n-1, \quad (3,9)$$

$$\lambda^{(1)} \geq 0, \quad \lambda^{(j)} \geq b_{j-1}^* \frac{P_{j-2}(\lambda^{(j)})}{P_{j-1}(\lambda^{(j)})}, \quad j = 1, 2, \dots, n, \quad (3,10)$$

برقرار باشند، آنگاه ماتریس سه‌قطری حقیقی متقارن $n \times n$ نامنفی A به شکل (۱,۱) موجود است به طوری که $\lambda^{(j)}$ یک مقدارویژه‌ی ماکسیمال زیرماتریس اصلی پیشرو $j \times j$ ، A_j برای $j = 1, 2, \dots, n$ ، از A بوده و $(\lambda^{(n)}, x)$ یک جفت‌ویژه‌ی A می‌باشد.

اثبات. از قضیه (۶)، شرایط (۳,۸) و (۳,۹) وجود ماتریس سه‌قطری حقیقی متقارن $n \times n$ ، A به شکل (۱,۱) با b_i های مثبت را تضمین می‌کند به طوری که $\lambda^{(j)}$ مقدارویژه‌ی ماکسیمال A_j و $(\lambda^{(n)}, x)$ جفت‌ویژه‌ی A می‌باشد. نشان دادن نامنفی بودن درایه‌های قطر اصلی $(a_j)_{j=1,2,\dots,n}$ باقی می‌ماند. از (۳,۱۰)، داریم $a_1 = \lambda^{(1)} \geq 0$ و

$$a_j = \lambda^{(j)} - b_{j-1}^* \frac{P_{j-2}(\lambda^{(j)})}{P_{j-1}(\lambda^{(j)})} \geq 0, \quad j = 2, 3, \dots, n.$$

□

^۷Minimal



مثال ۹. اعداد حقیقی

$$\begin{array}{ccccccc} \lambda^{(1)} & \lambda^{(2)} & \lambda^{(3)} & \lambda^{(4)} & \lambda^{(5)} & \lambda^{(6)} & \lambda^{(7)} \\ 0/6829 & 3/4716 & 4/4172 & 4/4194 & 4/4195 & 4/4198 & 4/4238, \end{array}$$

و بردار حقیقی

$$x = (0/2275, 0/5257, 0/4566, 0/532, 0/1407, 0/1829, 0/6383)^T,$$

در شرایط نتیجه (۸)، صدق می‌کنند. طبق راهکارهای ارائه شده در اثبات قضیه (۶)، با استفاده از این داده‌ها ماتریس سه‌قطری حقیقی متقارن نامنفی

$$A = \begin{bmatrix} 0/6829 & 1/6186 & 0 & 0 & 0 & 0 & 0 \\ 1/6186 & 2/5321 & 1/3716 & 0 & 0 & 0 & 0 \\ 0 & 1/3716 & 2/8277 & 0/1452 & 0 & 0 & 0 \\ 0 & 0 & 0/1452 & 0/6665 & 0/9498 & 0 & 0 \\ 0 & 0 & 0 & 0/9498 & 0/3571 & 2/8518 & 0 \\ 0 & 0 & 0 & 0 & 2/8518 & 1/0901 & 0/3266 \\ 0 & 0 & 0 & 0 & 0 & 0/3266 & 4/3302 \end{bmatrix},$$

حاصل می‌شود که طیف ماتریس‌های A_j عبارتند از:

$$\sigma(A_1) = \{6829/0\},$$

$$\sigma(A_2) = \{-0/2566, 4716/3\},$$

$$\sigma(A_3) = \{-0/4208, 2/0463, 4172/4\},$$

$$\sigma(A_4) = \{-0/4219, 0/6569, 2/0548, 4194/4\},$$

$$\sigma(A_5) = \{-0/4642, -0/4101, 1/4613, 2/0599, 4195/4\},$$

$$\sigma(A_6) = \{-2/3258, -0/4217, 0/6992, 2/0540, 3/7309, 4198/4\},$$

$$\sigma(A_7) = \{-2/3320, -0/4217, 0/6964, 2/0540, 3/6512, 4/4150, 4238/4\},$$

و

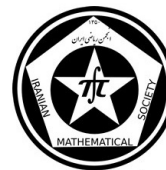
$$Ax = (4238/4)x.$$



- [1] M.T. Chu and G.H. Golub, *Inverse Eigenvalue Problems: Theory, Algorithms and Application*, Numerical Mathematics and Scientific Computation. Oxford University Press, New York, 2005.
- [2] K. Hoffman and R. Kunze, *Linear Algebra*, 2nd ed. Prentice-Hall, 1971.
- [3] X. Huang, X. Hu and L. Zhang, *Physical parameters reconstruction of a fixed-fixed mass-spring system from its characteristic data*, J. Comput. Appl. Math. 206 (2007), no. 2, 645–655.
- [4] X. Pickman, J. Egana and R. L. Soto, *Extremal inverse eigenvalue problem for bordered diagonal matrices*. Linear Algebra and its Applications, 427 (2007), no. 2-3, 256–271.
- [5] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*, Applied Mathematics, 12. Springer-Verlag, New York, 2002.

پست الکترونیکی: mohammadjavad5799@yahoo.com

پست الکترونیکی: mpanahi@birjand.ac.ir



تصحیح بهینه دستگاه نشدنی از معادلات قدرمطلق کلی با استفاده از مینیمم تغییرات در

ماتریس‌های ضرایب و بردار سمت راست

فخرالدین هاشمی[†]، سعید کتابچی[‡]

[†] دانشکده ریاضی، دانشگاه گیلان، رشت، ایرا

[‡] دانشکده ریاضی، دانشگاه گیلان، رشت، ایرا

چکیده

در این مقاله، تصحیح بهینه دستگاه نشدنی از معادلات قدرمطلق کلی، با استفاده از مینیمم نرم تغییرات در ماتریس‌های ضرایب و بردار سمت راست بررسی می‌شود و در ادامه، هموارسازی مساله کسری ناهموار و نامحدب بدست آمده در نظر گرفته می‌شود. سپس، برای حل مساله کسری هموار شده و فرمول نویسی شده به صورت تفاضل محدب، الگوریتم ترکیبی نیوتن هموار ساز-تفاضل محدب را به کار می‌گیریم. در پایان، برای دستگاه معادلات قدر مطلق نشدنی تولید شده به طور تصادفی، توابع هموار ساز برای هموارسازی مساله کسری ناهموار را به کمک روش دولان-موری مورد مقایسه قرار می‌دهیم. واژه‌های کلیدی: برنامه‌ریزی تفاضل محدب، دستگاه معادلات قدر مطلق کلی، منظم‌سازی، هموار ساز. طبقه‌بندی موضوعی [۲۰۱۰]: 65D10, 34M03, 90C25.

۱ پیش‌گفتار

دستگاه معادلات قدر مطلق کلی نشدنی به صورت

$$Ax + B|x| = b, \quad (1.1)$$

را در نظر می‌گیریم که در آن $A, B \in \mathbb{R}^{n \times n}$ و $b \in \mathbb{R}^n$ داده شده‌اند و $\|\cdot\|$ قدر مطلق مولفه به مولفه را نمایش می‌دهد. نشدنی بودن این دستگاه معادلات به خاطر دلایلی همچون خطا در داده‌ها، خطا در فرمول و غیره می‌باشد (برای جزئیات بیشتر [۱] را ببینید). همان‌طور که در مقالات [۱، ۳، ۸، ۹] بیان شده، دوباره مدل‌سازی و یافتن خطا در فرمول اقتصادی نیست. از این‌رو، تصحیح بهینه این دستگاه با



استفاده از مینیمم تغییرات در داده‌ها در نظر گرفته می‌شود که مساله کسری زیر حاصل شده است:

$$\min_{x \in \mathbb{R}^n} \frac{\|Ax + B|x| - b\|^2}{1 + \|x\|^2}. \quad (1'2)$$

در مساله (1'2) با در نظر گرفتن حالت‌های خاص ماتریس $B = -I$ ، $B = 0$ ، به ترتیب تصحیح بهینه دستگاه معادلات قدر مطلق خاص نشدنی و دستگاه معادلات خطی نشدنی در مقالات [۱، ۲، ۸، ۹] انجام شده‌اند. در این مقاله، با فرض $B \neq 0$ ، $-I$ ، تصحیح بهینه دستگاه (1'1) را با اعمال مینیمم نرم تغییرات در ماتریس‌های ضرایب و بردار سمت راست انجام داده‌ایم که مساله کسری

$$\min_{x \in \mathbb{R}^n} \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2}, \quad (1'3)$$

بدست آمده است. با ذکر یک مثال نشان داده می‌شود که مساله (1'3) دارای یک جواب با اندازه نرم ۲، بسیار بزرگ می‌باشد که در عمل قابل استفاده نمی‌باشد. بنابه دلایل ذکر شده در مقالات [۶، ۷]، برای کنترل جواب روش منظم‌سازی دقیق (ℓ_p -norm, $0 < p < 1$) را با شکل کلی

$$\min_{x \in \mathbb{R}^n} \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2} + \lambda \|x\|_p^p, \quad (1'4)$$

در نظر می‌گیریم که در آن $\|x\|_p^p = \sum_{i=1}^n |x_i|^p$.

مساله (1'3) یک مساله بهینه‌سازی غیرلیپشیتس است که تابع هدف آن نامحدوب و ناهموار است. برای حل این مساله، با استفاده از توابع هموار معرفی شده در [۱۰]، ابتدا این مساله را هموار کرده و سپس با اعمال الگوریتم‌های بیان شده در [۶] آن را حل می‌کنیم. در پایان، با استفاده از روش دولان-موری در [۵] این توابع هموارساز مورد مقایسه قرار داده می‌شوند و بهترین تابع هموارساز مشخص می‌شود. این مقاله، به سه بخش تقسیم می‌شود: بخش ۲ تصحیح بهینه دستگاه معادلات قدر مطلق کلی را تشریح می‌کند. بخش ۳ روش منظم‌سازی (ℓ_p -norm, $0 < p < 1$) و هموارسازی مساله را توصیف می‌کند. بخش پایانی نتایج عددی گزارش می‌شوند.

۲ تصحیح بهینه دستگاه نشدنی از معادلات قدر مطلق کلی (GAVEs)

در این حالات، برای تصحیح GAVEs، تغییراتی را در ماتریس‌های ضرایب و بردار سمت راست ایجاد می‌کنیم. سپس، مساله مینیمم‌سازی

$$\min_x \min_{E_1, E_2, r} (\|E_1\|^2 + \|E_2\|^2 + \|r\|^2) \\ s.t. \quad (A + E_1)x + (B + E_2)|x| = b + r, \quad (2'1)$$

را در نظر می‌گیریم که در آن $E_1, E_2 \in \mathbb{R}^n$ ماتریس‌های آشوب هستند و $r \in \mathbb{R}^n$ یک بردار آشوب است. به منظور ساده کردن مساله (2'1)، مساله مینیمم‌سازی داخلی زیر را در نظر می‌گیریم:

$$\min_{E_1, E_2, r} (\|E_1\|^2 + \|E_2\|^2 + \|r\|^2) \\ s.t. \quad (A + E_1)x + (B + E_2)|x| = b + r, \quad (2'2)$$



که یک مساله مقید محدب است.

قضیه ۱. فرض کنید که $(E_{\lambda}^*, E_{\tau}^*, r^*)$ یک جواب بهینه برای مساله (۲۲) باشد. در این صورت، داریم

$$r^* = \frac{Ax^* + B|x^*| - b}{1 + 2\|x^*\|^2}, \quad E_{\lambda}^* = \frac{Ax^* + B|x^*| - b}{1 + 2\|x^*\|^2}(x^*)^{\top},$$

$$E_{\tau}^* = \frac{Ax^* + B|x^*| - b}{1 + 2\|x^*\|^2}(|x^*|)^{\top},$$

که در آن x^* یک جواب بهینه مساله $\min_{x \in \mathbb{R}^n} \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2}$ است.

اثبات. تابع لاگرانژ مساله (۲۲) به صورت

$$L(E_{\lambda}, E_{\tau}, r, \lambda) = \|E_{\lambda}\|^2 + \|E_{\tau}\|^2 + \|r\|^2 - \lambda^{\top}((A + E_{\lambda})x + (B + E_{\tau})|x| - (b + r)),$$

داده می‌شود. چون مساله (۲۲) محدب است، بنابراین شرایط کاروش-کاهن-تاکر (KKT)* لازم و کافی هستند، و هر $(E_{\lambda}, E_{\tau}, r)$ صادق در شرایط (KKT) یک مینیمم‌کننده سراسری است. داریم:

$$\frac{\partial L}{\partial E_{\lambda}} = 2E_{\lambda} - \lambda x^{\top} = 0. \quad (23)$$

$$\frac{\partial L}{\partial E_{\tau}} = 2E_{\tau} - \lambda |x|^{\top} = 0. \quad (24)$$

$$\frac{\partial L}{\partial r} = 2r + \lambda = 0. \quad (25)$$

$$\frac{\partial L}{\partial \lambda} = (A + E_{\lambda})x + (B + E_{\tau})|x| - (b + r) = 0. \quad (26)$$

از معادلات (۲۳)-(۲۶)، داریم که $2E_{\lambda} = \lambda x^{\top}$ ، $2E_{\tau} = \lambda |x|^{\top}$ و $\lambda = -2r$. بنابراین، بدست می‌آوریم $E_{\lambda} = -rx^{\top}$ ، $E_{\tau} = -r|x|^{\top}$ و $r = (A + E_{\lambda})x + (B + E_{\tau})|x| - b$. با ترکیب کردن این عبارات، می‌یابیم که $r + 2r\|x\|^2 = Ax + B|x| - b$ و نتیجه می‌گیریم که

$$r = \frac{Ax + B|x| - b}{1 + 2\|x\|^2}, \quad E_{\lambda} = \frac{Ax + B|x| - b}{1 + 2\|x\|^2}(x)^{\top},$$

$$E_{\tau} = \frac{Ax + B|x| - b}{1 + 2\|x\|^2}(|x|)^{\top}. \quad (27)$$

از (۲۷)، بدست می‌آوریم $\|E_{\lambda}\|^2 + \|E_{\tau}\|^2 + \|r\|^2 = \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2}$ ، و مقدار تابع هدف در (۲۲) در جواب بهینه برابر است با

$$\|E_{\lambda}^*\|^2 + \|E_{\tau}^*\|^2 + \|r^*\|^2 = \min_{x \in \mathbb{R}^n} \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2}.$$

□

این اثبات را کامل می‌کند.

*Karush-Kuhn-Tucker



۳ روش منظم سازی و تقریب های هموار ساز

حال با استفاده از مثال زیر نشان می دهیم، مسأله (۱۰۳) دارای جوابی با نرم بسیار بزرگ می باشد که قابل استفاده عملی نیست.

مثال ۲. دستگاه نشدنی $Ax + B|x| = b$ تولید شده به طور تصادفی، به صورت زیر را در نظر می گیریم:

$$A = \begin{bmatrix} -۱/۵۸ & ۰ \\ ۰/۹۹ & ۰/۹۷ \end{bmatrix}, \quad B = \begin{bmatrix} -۱/۵۹ & ۰ \\ ۰ & ۰/۹۷ \end{bmatrix}, \quad b = \begin{bmatrix} ۰/۹۹ \\ -۰/۰۱ \end{bmatrix}.$$

حال، مسأله کسری حاصل شده از تصحیح بهینه این دستگاه نشدنی را با استفاده از روش های بهینه سازی در MATLAB، حل کردیم که جواب بهینه x^* با $۳/۳۹e + ۶۳$ بدست آمده است. بنابراین، همانطور که مشاهده می شود جواب حاصل شده از لحاظ اندازه نرم ۲، بسیار بزرگ می باشد.

به منظور کنترل کردن جواب بهینه (۱۰۳)، روش منظم سازی تیخانوف را به صورت ذیل برای آن در نظر می گیریم:

$$\min_{x \in \mathbb{R}^n} \omega(x) = \frac{\|Ax + B|x| - b\|^2}{1 + 2\|x\|^2} + \lambda\|x\|^2. \quad (۳۰۱)$$

حال، با استفاده از قضیه دینکل باخ در [۴]، مسأله (۳۰۱) را به صورت یک معادله تک متغیره فرمول نویسی مجدد می کنیم که به صورت زیر است:

$$\varphi(t) = \min_{x \in \mathbb{R}^n} \left\{ \|Ax + B|x| - b\|^2 + \lambda\|x\|^2(1 + \|x\|^2) - t(1 + 2\|x\|^2) \right\} = 0. \quad (۳۰۲)$$

به سادگی ثابت می شود که ریشه معادله (۳۰۲) به ترتیب با مقدار بهینه سراسری مسأله (۳۰۱) معادل است (برای توضیحات بیشتر [۴] را ببینید). قضیه و نتیجه بیان شده در [۶] به برخی از ویژگی های $\varphi(t)$ اشاره می کند و نشان می دهد که این تابع ریشه یکتا در بازه $[0, \|b\|^2]$ دارد. از طرف دیگر، بیان می کند که تابع $\Theta(t) = -\varphi(t)$ زیردیفرانسیل پذیر هست و $1 + 2\|x_t\|^2$ زیرگرادیان آن در نقطه t است که در آن x_t جواب مسأله زیر است:

$$\min_{x \in \mathbb{R}^n} \left\{ \|Ax + B|x| - b\|^2 + \lambda\|x\|^2(1 + \|x\|^2) - t(1 + 2\|x\|^2) \right\}. \quad (۳۰۳)$$

توجه می کنیم که t^* ریشه ی معادله Θ هست اگر و تنها اگر $t^* = \min_{x \in \mathbb{R}^n} \omega(x)$ از این رو، برای پیدا کردن t^* ، روش نیوتن تعمیم یافته استفاده می شود. برای اجرا این روش نیاز به محاسبه مقادیر زیرگرادیان $1 + 2\|x_t\|^2$ داریم. از طرف دیگر، برای محاسبه زیرگرادیان ذکر شده، جوابی برای مسأله داخلی ناهموار و نامحدب از معادله (۳۰۲) لازم می شود، یعنی، مسأله (۳۰۳). با توجه به ناهموار و نامحدب بودن تابع هدف مسأله (۳۰۳)، ابتدا با استفاده از تقریب های هموار که در [۱۰] معرفی می شوند این تابع هدف را هموار کرده و سپس آن را به صورت تفاضل محدب بازنویسی می کنیم. براساس توابع هموار ساز ϕ_i برای $i = 1, 2, 3, 4$ ، مسأله (۳۰۳)، به صورت زیر هموار می شود:

$$\min_{x \in \mathbb{R}^n} \left\{ \|Ax + B\Phi_i(x) - b\|^2 + \lambda\|x\|^2(1 + 2\|x\|^2) - t(1 + 2\|x\|^2) \right\}, \quad (۳۰۴)$$



که در آن، $\Phi_i(x) = (\phi_i(x_1, \mu), \phi_i(x_2, \mu), \dots, \phi_i(x_n, \mu))^T$ همانطور که می‌دانید، مساله (۳۴) می‌تواند به صورت مساله برنامه‌ریزی DC فرمول‌بندی مجدد شود که به صورت

$$\min_{x \in \mathbb{R}^n} f(x) = g(x) - h(x), \quad (35)$$

است، که در آن g, h توابع محدب تعریف شده به صورت زیر هستند:

$$g(x) = \|Ax + B\Phi_i(x) - b\|^2 + \beta\|x\|^2 + \lambda\|x\|^2(1 + 2\|x\|^2),$$

$$h(x) = t(1 + 2\|x\|^2) + \beta\|x\|^2.$$

برای حل مساله (۳۵) با انتخاب کردن پارامتر ثابت $\mu_0 > 0$ و بردار اولیه $x_0 \in \text{dom } g$ در هر تکرار، مساله ذیل را با استفاده از الگوریتم معرفی شده در [۶]، به منظور به دست آوردن یک جواب موضعی برای مساله (۳۵) حل می‌کنیم.

$$\min_{x \in \mathbb{R}^n} \bar{f}(x) = g(x) - (\nabla h(x_0))^T(x - x_0). \quad (36)$$

تابع هدف مساله (۳۶) محدب و به طور پیوسته دیفرانسیل پذیر هست، و گرادیان آن پیوسته لیپشیتس است. بنابراین، همین تعمیم یافته آن موجود است. همانطور که قبلاً ذکر شد، برای پیدا کردن ریشه معادله (۳۲)، یعنی t^* ، ما باید مساله (۳۳) را حل کنیم که الگوریتم ترکیبی، روش نیوتن هموارساز-تفاضل محدب (DC-Newton method) معرفی شده در [۶] را برای حل این مساله استفاده کرده‌ایم. حال، با در نظر گرفتن نتایج این الگوریتم، برای بدست آوردن t^* الگوریتم نیوتن تعمیم‌یافته (Generalized Newton method) در [۶] را به کار گرفته‌ایم.

۴ نتایج عددی

در این بخش، نتایج عددی بدست آمده برای تصحیح بهینه دستگاه نشدنی (۱۱) گزارش می‌شوند. این دستگاه نشدنی با استفاده از کد متلب ذکر شده در زیر به طور تصادفی تولید می‌شود.

کد متلب: یک دستگاه نشدنی تولید شده به طور تصادفی از معادلات به صورت $Ax + B|x| = b$

```
%Generates random inconsistent system Ax+B|x|=b%
```

```
n=input('n: ');
```

```
u=10*(rand(n,1)-rand(n,1));
```

```
u=u/norm(u);
```

```
k=null(u');
```

```
k=[k,zeros(n,1)];
```



$B = \text{input}('Enter the matrix B: ');$

$A = k + B;$

$b = u;$

تصحیح بهینه را با استفاده از توابع هموارساز ϕ_i ، $i = 1, 2, 3, 4$ و $n = 10, 50, 100, \dots, 5000$ برای حل مساله (۳۲) اجرا می‌کنیم. هر دستگاه نشدنی، با ابعاد یکسان برای آزمودن، ۱۰ بار به طور تصادفی تولید می‌شوند. آنگاه، متوسط نتایج عددی برای برخی ابعاد در جدول ۱ برای حل مساله (۳۲) ارائه می‌شوند. در جدول ۱ مقادیر $Mtime$, Me , $Merror$ ، به صورت زیر محاسبه می‌شوند:

$$Mtime = \frac{1}{10} \left(\sum_{l=1}^{10} (time)_l \right), Me = \frac{1}{10} \left(\sum_{l=1}^{10} (\| (A + E_l^*) x^* + (B + E_l^*) |x^*| - (b + r^*) \|)_l \right),$$

$$Merror = \frac{1}{10} \left(\sum_{l=1}^{10} (|t_{k+1} - t_k|)_l \right).$$

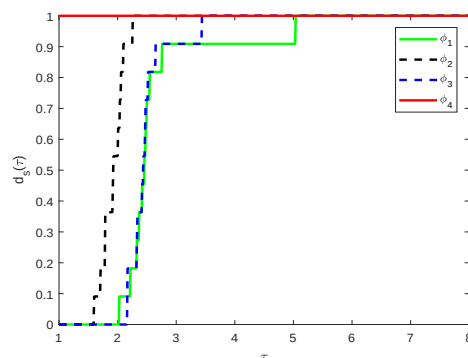
در فرمول‌های بالا، **time** زمان اجرا الگوریتم نیوتن تعمیم‌یافته را نمایش می‌دهد؛ x^* جوابی برای مساله (۳۱) هست. در ادامه، E_1^*, E_4^* و r^* مقادیری هستند که در قضیه ۱ نحوه محاسبه آن‌ها بیان شده است.



جدول ۱: نتایج عددی برای حل مساله (۳۰۲).

n	ϕ_1			ϕ_2			ϕ_3			ϕ_4		
	Me	$Merror$	$Mtime$	Me	$Merror$	$Mtime$	Me	$Merror$	$Mtime$	Me	$Merror$	$Mtime$
100	8.50e-17	2.09e-11	0.161	9.93e-17	5.17e-11	0.051	8.40e-17	8.11e-11	0.110	8.00e-17	1.26e-10	0.032
500	1.04e-16	6.31e-11	1.970	8.43e-17	2.45e-10	1.618	9.72e-17	2.71e-10	1.892	8.50e-17	9.99e-11	0.716
1000	7.79e-17	1.74e-10	10.514	9.50e-17	2.89e-10	8.561	8.66e-17	4.48e-10	9.829	9.49e-17	8.10e-11	4.211
1500	1.06e-16	1.39e-10	31.043	8.65e-17	4.81e-10	25.543	8.94e-17	3.44e-10	30.141	9.83e-17	1.81e-10	12.218
2000	8.50e-17	2.29e-10	66.434	9.62e-17	3.52e-10	53.737	8.90e-17	5.22e-10	65.401	8.82e-17	6.41e-11	28.090
2500	8.36e-17	2.65e-10	121.671	9.10e-17	4.16e-10	100.975	9.51e-17	4.22e-10	118.511	8.27e-17	9.92e-11	49.078
3000	9.02e-17	2.63e-10	195.197	9.73e-17	4.17e-10	161.878	9.44e-17	3.22e-10	196.227	9.27e-17	1.36e-10	80.695
3500	9.37e-17	2.92e-10	296.091	9.21e-17	5.71e-10	231.739	8.66e-17	5.88e-10	298.091	1.02e-16	2.08e-10	120.380
4000	8.77e-17	3.39e-10	442.878	9.23e-17	6.65e-10	341.281	8.77e-17	6.44e-10	479.350	8.73e-17	3.38e-09	190.894
4500	8.81e-17	3.70e-10	592.973	8.05e-17	7.32e-10	479.176	9.24e-17	7.32e-10	581.606	9.24e-17	6.86e-11	268.264
5000	8.75e-17	4.09e-10	794.597	8.57e-17	8.12e-10	671.141	9.44e-17	7.99e-10	851.873	9.51e-17	7.72e-11	392.720

در جدول ۱، به ترتیب مقادیر برای ستون‌های مرتبط با Me خطای تصحیح بهینه برای دستگاه نشدنی تولید شده، $Merror$ ، خطا همگرایی الگوریتم نیوتن تعمیم‌یافته، را با دقت بالایی مشخص می‌کنند. همچنین، می‌تواند نتیجه شود که مقادیر عددی ستون‌های ذکر شده تقریباً یکسان هستند. ولی اختلافات فاحشی در ستون‌های مرتبط با $Mtime$ مشاهده می‌شوند. این اختلافات با استفاده از روش دولان-موری مقایسه می‌شوند. در این روش، هر تابع هموارساز ϕ_i همراه با الگوریتم نیوتن تعمیم‌یافته به‌عنوان یک حل‌کننده و دستگاه‌های نشدنی با ابعاد متفاوت به‌عنوان مسائل تست از مجموعه P در نظر می‌گیریم که مجموعه مسائل تست به‌طور تصادفی با استفاده از کد متلب ذکر شده تولید می‌شوند. نتایج مقایسه‌ها در شکل ۱ نشان داده می‌شوند. شکل ۱ نشان می‌دهد که زمان محاسبه الگوریتم نیوتن تعمیم‌یافته



شکل ۱: مقایسه عملکرد زمان محاسبه شده برای توابع هموارساز با استفاده از الگوریتم نیوتن تعمیم‌یافته برای حل مساله (۳۰۲) با اعمال کد متلب ذکر شده

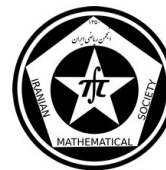
اشاره شده برای تابع هموارساز ϕ_4 بهترین عملکرد دارد و ϕ_1 یک عملکرد ضعیف‌تری نسبت به ϕ_2 و ϕ_3 دارد.



- [1] P. Amarala, J. Júdice, H. D. Serali, *A reformulation-linearization-convexification algorithm for optimal correction of an inconsistent system of linear constraints*, Computers and Operations Research 35, 1494–1509, (2008).
- [2] M. Behboodi-Kahoo, S. Ketabchi, *Parametric approach for correcting inconsistent linear equality system*, Optimization Methods and Software 29(6), 1317–1322, (2014).
- [3] Y. Censor, A. Ben-Israel, Y. Xiao, and J. M. Galvin, *On linear infeasibility arising in intensity-modulated radiation therapy invers planning*, Linear Algebra Appl 428, 1406–1420, (2008).
- [4] W. Dinkelbach, *On nonlinear fractional programming*, Managment Science 131, 492–498, (1967).
- [5] E. D. Dolan, J. J. Moré, *Benchmarking optimization software with performance profiles*, Mathematical Programming 91 (2), 201–213, (2002).
- [6] F. Hashemi, S. Ketabchi, *Numerical comparisons of smoothing functions for optimal correction of an infeasible system of absolute value equations*, Numer. Algebra Control Optim. (2019), 95-118.
- [7] F. Hashemi, S. Ketabchi, *Optimal correction of infeasible equations system as using norm regularization*, Bol. Soc. Paran. Mat. (2019), to appear.
- [8] S. Ketabchi, H. Moosaei, S. Fallahi, *Optimal error correction of the absolute value equation using a gentic algorithm*, Mathematical and Computer Modelling 57, 2339–2342, (2013).
- [9] H. Moosaei, S. Ketabchi, P. M. Pardalos, *Tikhonov regularization for infeasible absolute value equations*, Optimization 65, 1531–1537, (2016).
- [10] B. Saheya, C.-H. Yu and J.-S. Chen, *Numerical comparisons based on four smoothing functions for absolute value equation*, Journal of Applied Mathematics and Computing, 56, 131–149, (2018).

پست الکترونیکی: fhashemi@phd.guilan.ac.ir

پست الکترونیکی: sketabchi@guilan.ac.ir



تقریب جدید برای معادله‌ی برگرز کسری زمانی با مشتق کاپاتو

راضیه غلامپور[†] و علیمردان شاهرضایی[‡]

دانشجوی دکتری ریاضی دانشگاه الزهراء(س)[†]

دانشیار دانشگاه الزهراء(س)[‡]

چکیده

معادلات دیفرانسیل جزئی با مشتق کسری کاپاتو در بسیاری از زمینه‌های مهندسی کاربرد دارد. معادله‌ی برگرز کسری یک دسته از همین معادلات است. در این مقاله معادله‌ی برگرز کسری زمانی مرتبه اول با مشتق کاپاتو مورد بررسی قرار گرفته است. این مقاله یک تقریب جدید برای حل معادله‌ی برگرز کسری زمانی با استفاده از روش تبدیل لاپلاس ارائه می‌دهد، در ادامه با استفاده از روش تفاضلات متناهی (صریح و ضمنی)، با ارائه مثالهای عددی مسأله مورد ارزیابی قرار می‌گیرد. واژه‌های کلیدی: تبدیل لاپلاس، روش کرانک-نیکلسون، مشتق کاپاتو، معادله‌ی برگرز کسری زمانی. طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

معادلات دیفرانسیل کسری، تعمیم معادلات دیفرانسیل از هر مرتبه‌ی دلخواه است. به دلیل توانایی معادلات دیفرانسیل کسری در مدل‌سازی پدیده‌های پیچیده‌ی فیزیکی مطالعات بسیاری بر روی آنها انجام گرفته است [۱، ۲]. در سال‌های اخیر، توسعه سریع در نظریه علمی محاسبات کسری در شگردها و علوم کاربردی مشاهده شده است. معادلات دیفرانسیل جزئی کسری در زمینه‌های مختلف مانند دینامیک مایع، جریان گرما، مالی، هیدرولوژی و غیره کاربرد فراوان دارند [۳، ۴]. راه حل دقیق معادلات دیفرانسیل کسری به راحتی قابل دستیابی نیست، بنابراین نیاز به روش‌های عددی برای حل معادلات دیفرانسیل مرتبه کسری داریم. یکی از این معادلات دیفرانسیل کسری معادله‌ی برگرز است. این معادله ویژگی‌های غیرخطی معادلات حاکم در بسیاری از مسائل انتقالی عملی مانند انتقال تلاطم دارد و کاربرد گسترده‌ای نیز در بسیاری از پدیده‌های فیزیکی دارد از جمله نظریه موج ضربه‌ای، فرآیندهای موج در محیط ترموالاستیک، انتقال و پخش آلاینده‌ها در رودخانه و انتقال رسوب [۵]. همچنین این معادلات در مدل رسوب سازی، روش شبکه بولتزمن، حرکت دورانی غلظت مقادیر مقیاس شده دو نوع ذره در تعلیق‌های مایع و کلوئیدهای تحت گرانش به کار می‌رود [۶]. در شکل کلی مسأله‌ی برگرز کسری زمانی با مشتق کاپاتو به ازاء هر مقدار داده شده و مفروض ضریب چسبندگی ν به صورت زیر است:

$${}^c D_t^\alpha u(x, t) + u \frac{\partial u(x, t)}{\partial x} = \nu \frac{\partial^2 u(x, t)}{\partial x^2} \quad ; \quad 0 < x < L, t > 0, 0 < \alpha < 1, \quad (1.1)$$

$$u(x, 0) = f(x) \quad ; \quad 0 < x < L, \quad (1.2)$$

$$u(0, t) = \phi(t), u(L, t) = \psi(t) \quad ; \quad t > 0. \quad (1.3)$$

که در آن توابع ϕ, ψ و f به طور تکه‌ای پیوسته بردارنده هایشان و اعداد L, ν, α معلوم و تابع u مجهول است. در معادله اخیر نماد ${}^c D_t^\alpha$ عملگر مشتق کاپاتو نسبت به t در بازه $[0, t]$ است که به صورت زیر تعریف می‌گردد [۳]:

$${}^c D_t^\alpha u(x, t) = \frac{1}{\Gamma(1-\alpha)} \int_0^t \frac{\partial u(x, s)}{\partial s} \frac{1}{(t-s)^\alpha} ds, \quad (1.4)$$

که در آن $\Gamma(\cdot)$ تابع گاما است. ارائه این مقاله به صورت زیر سازمان‌دهی شده است: در بخش دوم از این مقاله با استفاده از تبدیل لاپلاس، تقریبی برای مشتق کسری معرفی می‌کنیم و پایداری تقریب را مورد بررسی قرار می‌دهیم. در بخش سوم، مسالهی حاصل را با استفاده از روش‌های صریح و ضمنی گسسته سازی می‌کنیم. در بخش چهارم به نتایج عددی حاصل از روش‌های مذکور و ارائه مثال می‌پردازیم. در بخش پنجم نتیجه‌گیری به همراه ارائه پیشنهاداتی برای ادامه کار اشاره می‌گردد.

۲ تقریب معادله برگرز کسری زمانی

تبدیل لاپلاس مشتق کسری کاپاتو را به صورت زیر می‌نویسیم [۷]:

$$L\{{}^c D_t^\alpha u(x, t)\} = s^\alpha \hat{u}(x, s) - s^{\alpha-1} \hat{u}(x, 0) = s^\alpha [\hat{u}(x, s) - s^{-1} \hat{u}(x, 0)] \quad (2.1)$$

که در آن $\hat{u}(x, s)$ تبدیل لاپلاس $u(x, t)$ است. از آن جایی که $0 < \alpha < 1$ است، جمله s^α را با استفاده از درونیابی لاگرانژ به صورت زیر خطی سازی می‌کنیم:

$$s^\alpha \approx \alpha s^1 + (1-\alpha) s^0 = \alpha s + (1-\alpha). \quad (2.2)$$

با جایگذاری رابطه (۶) در (۵) و سپس با اعمال تبدیل لاپلاس معکوس، به رابطه زیر می‌رسیم [۷]:

$${}^c D_t^\alpha u(x, t) \approx \alpha \frac{\partial u(x, t)}{\partial t} + (1-\alpha) [u(x, t) - u(x, 0)] \quad (2.3)$$



با جایگذاری رابطه (۷) در (۱) داریم:

$$\alpha \frac{\partial u(x, t)}{\partial t} + (1 - \alpha) [u(x, t) - f(x)] + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}; \quad 0 < x < L, t > 0, \quad (2.4)$$

با توجه به موارد ذکر شده، مساله (۱) تا (۳) به صورت زیر باز نویسی میگردد:

$$\alpha \frac{\partial u(x, t)}{\partial t} + (1 - \alpha) [u(x, t) - f(x)] + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}; \quad 0 < x < L, t > 0, 0 < \alpha < 1, \quad (2.5)$$

$$u(x, 0) = f(x) \quad ; \quad 0 < x < L, \quad (2.6)$$

$$u(0, t) = \phi(t), u(L, t) = \psi(t) \quad ; \quad t > 0 \quad (2.7)$$

۳ پایداری مساله برگرز کسری با مشتق کاپاتو

مساله ی برگرز کسری زمانی (۹) تا (۱۱) را در نظر می گیریم. نشان می دهیم مساله خوش خیم و جواب آن یکتاست و پایدار عددی می باشد. توابع ψ, ϕ, f و منبع گرمایی پیوسته h و همچنین مقادیر ثابت ν, α, L معلوم و تابع u مجهول است. با ضرب کردن رابطه (۹) در $u(x, t)$ و سپس انتگرالگیری نسبت به x در فاصله $[0, L]$ و با انجام محاسبات و استفاده از شرایط کرانه ای دیریکله همگن و نامساوی کوشی - شوارتز، و ساده سازی به رابطه زیر می رسیم:

$$\begin{aligned} & \frac{\alpha}{2} \frac{d}{dt} \int_0^L u^2(x, t) dx + (1 - \alpha) \int_0^L u^2(x, t) dx + \nu \int_0^L \left(\frac{\partial u(x, t)}{\partial x} \right)^2 dx \\ &= (1 - \alpha) \int_0^L f(x) u(x, t) dx + \int_0^L h(x, t) u(x, t) dx \\ &\leq \frac{1 - \alpha}{2} \left[\int_0^L u^2(x, t) dx + \int_0^L f^2(x) dx \right] + \frac{1 - \alpha}{2} \int_0^L u^2(x, t) dx + \frac{1}{2(1 - \alpha)} \int_0^L h^2(x, t) dx, \end{aligned} \quad (3.1)$$

رابطه (۱۲) پس از ساده سازی به صورت زیر تبدیل می گردد:

$$\frac{\alpha}{2} \frac{d}{dt} \int_0^L u^2(x, t) dx + \nu \int_0^L \left(\frac{\partial u(x, t)}{\partial x} \right)^2 dx \leq \frac{1 - \alpha}{2} \int_0^L f^2(x) dx + \frac{1}{2(1 - \alpha)} \int_0^L h^2(x, t) dx. \quad (3.2)$$

با توجه به رابطه (۱۳)، اگر توابع h و f به سمت صفر میل کنند و با توجه به اینکه ν نیز نامنفی می باشد، سمت چپ رابطه نیز به صفر میل می کند و این نشان می دهد جواب یکتا و پایدار است لذا مساله خوش تعریف است.

۴ روش‌های گسسته سازی

روش‌های گسسته سازی که در این مقاله مورد بررسی قرار خواهیم داد : روش صریح زمان پیشرو-مکان مرکزی و روش ضمنی کرانک-نیکلسون است.

روش زمان پیشرو-مکان مرکزی از مرتبه اول صریح و تک گامی و خطای برشی آن از مرتبه $O(\Delta t, \Delta x^2)$ است. گسسته سازی معادله‌ی برگرز با این روش زمانی که بسیار کوچک باشد، کارآمد نمی‌باشد. این روش برای مساله‌ی برگرز کسری زمانی (۱۱) - (۹) به صورت زیر است:

۱.۴ روش صریح

گسسته سازی مساله (۹) مساله زیر را پدید می‌آورد که آن را حل می‌کنیم [۸].

$$\alpha \frac{u_j^{n+1} - u_j^n}{\Delta t} + (1 - \alpha) u_j^n = (1 - \alpha) f_j + \nu \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2} - \quad (4.1)$$

$$\frac{u_j^n (u_{j+1}^n - u_{j-1}^n)}{2\Delta x} + h_j^n; 1 \leq j \leq M - 1, n \geq 0.$$

$$u_j^0 = f(x_j); 1 \leq j \leq M, \quad (4.2)$$

$$u_0^n = \phi(t_n), \quad u_M^n = \psi(t_n); \quad (4.3)$$

که در آن $h_j^n \approx h(x_j, t_n)$ و $u_j^n \approx u(x_j, t_n)$ است.

۲.۴ روش کرانک - نیکلسون

در این روش u_t با روش تفاضلی پیشرو، u_x و u_{xx} با میانگین تقریب تفاضلی مرکزی در گامهای زمانی n و $n+1$ جایگزین می‌شوند. روش کرانک - نیکلسون دارای دقتی از مرتبه $O(\Delta t^2, \Delta x^2)$ و بدون هیچ قید و شرطی پایدار است.

گسسته سازی مساله (۱۱) - (۹) به روش کرانک-نیکلسون به صورت زیر است:

$$\alpha \frac{u_j^{n+1} - u_j^n}{\Delta t} + (1 - \alpha) \frac{u_j^{n+1} + u_j^n}{2} = (1 - \alpha) f_j + \nu \delta_x^2 \frac{u_j^{n+1} + u_j^n}{2} - \gamma_x^1 \frac{F_j^{n+1} + F_j^n}{2} \quad (4.4)$$

$$+ h_j^{n+1}; \quad 1 \leq j \leq M - 1, n \geq 0.$$

که در آن داریم:

$$\delta_x^2 u_j^n = \frac{u_{j-1}^n - 2u_j^n + u_{j+1}^n}{\Delta x^2}, \quad (4.5)$$

$$\gamma_x^1 = \frac{F_{j+1}^n - F_{j-1}^n}{2\Delta x}, \quad (4.6)$$

$$F = \frac{u^2}{\gamma}, \quad (4.7)$$

لازم به ذکر است که گسسته سازیهای بالا در فاصله $[0, L]$ با توجه به روابط زیر صورت گرفته است:

$$x_j = j\Delta x, j = 0, 1, \dots, M \quad \Delta x = \frac{L}{M} \quad t_n = n\Delta t$$

با توجه به [6] و با طی مراحل و ساده سازی، به رابطه زیر می‌رسیم:

$$\begin{aligned} & \alpha u_j^{n+1} + (1 - \alpha) \frac{\Delta t}{\gamma} u_j^{n+1} + \frac{\Delta t}{\gamma} [\gamma_x^1 (u_j^n u_j^{n+1}) - \nu \delta_x^2 (u_j^n)] \\ & = (\alpha - 1) \frac{\Delta t}{\gamma} u_j^n + (1 - \alpha) \Delta t f_j + \nu \frac{\Delta t}{\gamma} \delta_x^2 (u_j^n) + \alpha u_j^n + (\Delta t) h_j^{n+1/2}, \end{aligned} \quad (4.8)$$

ماتریس ضرایب رابطه (۲۱) برای جملات u_{j+1}^{n+1} ، u_j^{n+1} و u_{j-1}^{n+1} یک ماتریس سه قطری است. در مساله حاضر روش حذفی گاوس را به راحتی می‌توان با استفاده از صفرهای موجود در ماتریس سه قطری استفاده کرد. این روش به وسیله توماس اصلاح شده است. روش‌های ضمنی به طور کلی به زمان محاسباتی بیشتری در هر گام زمانی نیاز دارند و از آنجایی که این روش‌ها بدون قید و شرط پایدارند، می‌توان گام‌های زمانی بزرگتری انتخاب کرد. البته با برداشتن یک گام زمانی خیلی بزرگ ممکن است جواب‌های بی‌معنی ایجاد شود زیرا یک گام زمانی بزرگ، خطاهای برشی را ایجاد می‌کند. [۸]

۵ نتایج عددی

نمودارهای موجود در ادامه ارزیابی جواب مساله برگز کسری زمانی زیر را با استفاده از دو روش صریح و کرانک - نیکلسون با مجموعه شرایط مختلف می‌باشد.

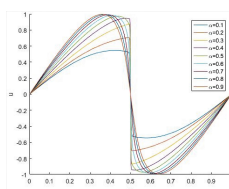
مثال ۱. مساله برگز کسری زمانی زیر که فاقد منبع h می‌باشد به صورت زیر مورد ارزیابی قرار می‌دهیم [۸]:

$${}^c D_t^\alpha u(x, t) + u \frac{\partial u(x, t)}{\partial x} = \nu \frac{\partial^2 u(x, t)}{\partial x^2} \quad ; 0 < x < 1, t > 0, 0 < \alpha < 1, \quad (5.1)$$

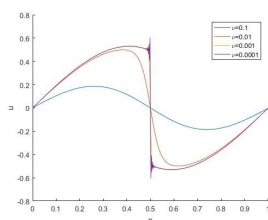
$$u(x, 0) = \sin(2\pi x) \quad ; 0 < x < 1, \quad (5.2)$$

$$u(0, t) = 0, u(1, t) = 0 \quad ; t > 0 \quad (5.3)$$

نمودارها با توجه به مقادیر پارامترهای $\Delta t = 0.01$ و $t = 0.5$ ، $n = 500$ به ازای مقادیر مختلف α و ν رسم شده است.



شکل ۱: حل عددی به ازای مقادیر متفاوت α



شکل ۲: حل عددی به ازای مقادیر متفاوت ضریب چسبندگی ν

نتایج حاصل از گسسته سازی مساله ی برگرز کسری زمانی با وجود منبع h توسط روش کرانک-نیکسون در سه مثال زیر مورد بررسی قرار می دهیم. با توجه به جواب های تحلیلی در هر مثال قادریم میزان خطا را محاسبه کنیم. در اینجا از دو نوع خطای گسسته سازی برای برآورد میزان کارایی روش یعنی نرم دو و نرم بینهایت استفاده شده است:

$$\|E\|_{\infty} = \max_{1 \leq i \leq n_{ip}} |u_i - \bar{u}_i|, \quad (5.4)$$

و

$$\|E\|_2 = \sqrt{\sum_{i=1}^{n_{ip}} (u_i - \bar{u}_i)^2} \quad (5.5)$$

که در آن n_{ip} تعداد نقاط، u_i جواب تحلیلی و $\bar{u}_i$ جواب عددی می باشد.

مثال ۲. برای مساله برگرز کسری زمانی [۹]

$${}^c D_t^\alpha u(x, t) + u \frac{\partial u(x, t)}{\partial x} = \nu \frac{\partial^2 u(x, t)}{\partial x^2} + h(x, t) \quad ; 0 < x < 1, t > 0 \quad (5.6)$$

$$u(x, 0) = 0 \quad ; 0 < x < 1, \quad (5.7)$$

$$u(0, t) = t^2, u(1, t) = e^{\alpha t^2} \quad ; t > 0 \quad (5.8)$$

که در آن

$$h(x, t) = \frac{2t^{2-\alpha} e^{\alpha x}}{\Gamma(3-\alpha)} + \alpha t^2 e^{2\alpha x} - \nu \alpha^2 t^2 e^x$$



می‌باشد و جواب تحلیلی به صورت

$$u(x, t) = t^\nu e^{\alpha x}$$

است. مقادیر نرم بی‌نهایت و نرم ۲ خطا برای مثال (۵-۲) در جدول ۱ و جدول ۲ با توجه به مقادیر داده‌ها آمده است.

جدول ۱: مقادیر نرم بینهایت خطا برای داده‌های متفاوت مثال ۲

$\ E\ _\infty$	$\alpha = 0.1$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$
$\nu = 1$	$6.999e^{-3}$	$6.326e^{-3}$	$6.805e^{-3}$	$7.590e^{-3}$	$8.379e^{-3}$	$9.233e^{-3}$
$\nu = 10^{-1}$	$5.975e^{-3}$	$6.191e^{-3}$	$6.640e^{-3}$	$7.123e^{-3}$	$7.569e^{-3}$	$7.777e^{-3}$
$\nu = 10^{-2}$	$5.605e^{-3}$	$5.554e^{-3}$	$5.325e^{-3}$	$5.155e^{-3}$	$4.830e^{-3}$	$3.917e^{-3}$
$\nu = 10^{-3}$	$4.421e^{-3}$	$3.065e^{-3}$	$1.922e^{-3}$	$1.575e^{-3}$	$1.289e^{-3}$	$2.937e^{-3}$
$\nu = 10^{-4}$	$1.573e^{-3}$	$1.154e^{-3}$	$1.786e^{-3}$	$1.592e^{-3}$	$1.579e^{-3}$	$2.981e^{-3}$
$\nu = 10^{-5}$	$1.425e^{-3}$	$1.221e^{-3}$	$1.793e^{-3}$	$1.598e^{-3}$	$1.761e^{-3}$	$3.013e^{-3}$

جدول ۲: مقادیر نرم ۲ خطا برای داده‌های متفاوت مثال ۲

$\ E\ _2$	$\alpha = 0.1$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$
$\nu = 1$	$2.816e^{-2}$	$2.889e^{-2}$	$2.995e^{-2}$	$2.905e^{-2}$	$2.486e^{-2}$	$2.314e^{-2}$
$\nu = 10^{-1}$	$1.460e^{-2}$	$2.095e^{-2}$	$2.404e^{-2}$	$2.158e^{-2}$	$1.466e^{-2}$	$2.485e^{-2}$
$\nu = 10^{-2}$	$1.633e^{-2}$	$1.695e^{-2}$	$2.113e^{-2}$	$1.766e^{-2}$	$7.544e^{-3}$	$2.537e^{-2}$
$\nu = 10^{-3}$	$1.750e^{-2}$	$1.441e^{-2}$	$1.961e^{-2}$	$1.607e^{-2}$	$3.833e^{-3}$	$2.579e^{-2}$
$\nu = 10^{-4}$	$1.729e^{-2}$	$1.381e^{-2}$	$1.964e^{-2}$	$1.613e^{-2}$	$3.910e^{-3}$	$2.597e^{-2}$
$\nu = 10^{-5}$	$1.739e^{-2}$	$1.398e^{-2}$	$1.973e^{-2}$	$1.619e^{-2}$	$3.990e^{-3}$	$2.600e^{-2}$

مثال ۳. برای مساله برگرزکسری زمانی زیر [۹]:

$${}^c D_t^\alpha u(x, t) + u \frac{\partial u(x, t)}{\partial x} = \nu \frac{\partial^2 u(x, t)}{\partial x^2} + h(x, t) \quad ; 0 < x < 1, t > 0 \quad (5.9)$$

$$u(x, 0) = 0 \quad ; 0 < x < 1, \quad (5.10)$$

$$u(0, t) = t^\nu, u(1, t) = -t^\nu \quad ; t > 0 \quad (5.11)$$

که در آن

$$h(x, t) = \frac{\nu t^{\nu-\alpha} \cos(\pi x)}{\Gamma(3-\alpha)} - \pi t^\nu \cos(\pi x) \sin(\pi x) + \nu t^\nu \pi^\nu \cos(\pi x)$$



می باشد و جواب تحلیلی به صورت

$$u(x, t) = t^\nu \cos(\pi x)$$

است. مقادیر نرم بی نهایت و نرم ۲ خطا برای مثال (۵-۳) در جدول ۳ و جدول ۴ با توجه به مقادیر داده ها آمده است.

جدول ۳: مقادیر نرم بینهایت خطا برای داده های متفاوت مثال ۳

$\ E\ _\infty$	$\alpha = 0.1$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$
$\nu = 1$	$5.914e^{-3}$	$5.904e^{-3}$	$5.873e^{-3}$	$5.822e^{-3}$	$5.737e^{-3}$	$5.592e^{-3}$
$\nu = 10^{-1}$	$5.803e^{-3}$	$5.726e^{-3}$	$5.471e^{-3}$	$5.153e^{-3}$	$4.746e^{-3}$	$4.179e^{-3}$
$\nu = 10^{-2}$	$5.251e^{-3}$	$4.611e^{-3}$	$3.370e^{-3}$	$2.848e^{-3}$	$2.479e^{-3}$	$1.684e^{-3}$
$\nu = 10^{-3}$	$3.424e^{-3}$	$2.748e^{-3}$	$3.982e^{-3}$	$4.409e^{-3}$	$4.445e^{-3}$	$4.208e^{-3}$
$\nu = 10^{-4}$	$1.348e^{-3}$	$4.596e^{-3}$	$5.369e^{-3}$	$5.415e^{-3}$	$5.268e^{-3}$	$4.947e^{-3}$
$\nu = 10^{-5}$	$1.449e^{-3}$	$5.137e^{-3}$	$5.581e^{-3}$	$5.551e^{-3}$	$5.373e^{-3}$	$5.038e^{-3}$

جدول ۴: مقادیر نرم ۲ خطا برای داده های متفاوت مثال ۳

$\ E\ _2$	$\alpha = 0.1$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$
$\nu = 1$	$1.208e^{-2}$	$1.214e^{-2}$	$1.215e^{-2}$	$1.161e^{-2}$	$1.033e^{-2}$	$8.784e^{-3}$
$\nu = 10^{-1}$	$9.939e^{-3}$	$1.124e^{-2}$	$1.173e^{-2}$	$1.004e^{-2}$	$7.070e^{-3}$	$8.386e^{-3}$
$\nu = 10^{-2}$	$8.591e^{-3}$	$9.632e^{-3}$	$1.050e^{-2}$	$8.121e^{-3}$	$4.009e^{-3}$	$8.497e^{-3}$
$\nu = 10^{-3}$	$9.334e^{-3}$	$8.289e^{-3}$	$1.037e^{-2}$	$8.348e^{-3}$	$4.960e^{-3}$	$9.634e^{-3}$
$\nu = 10^{-4}$	$9.305e^{-3}$	$8.787e^{-3}$	$1.086e^{-2}$	$8.851e^{-3}$	$5.705e^{-3}$	$1.011e^{-2}$
$\nu = 10^{-5}$	$9.528e^{-3}$	$9.075e^{-3}$	$1.097e^{-2}$	$8.942e^{-3}$	$5.818e^{-3}$	$1.018e^{-2}$

نتیجه گیری و پیشنهادات

در این مقاله بر روی مساله برگرز کسری زمانی برای مقادیر $0 < \alpha < 1$ متمرکز شدیم و تقریب جدیدی برای مساله معرفی کردیم که با استفاده از روش کرانک- نیکلسون برای اولین بار، مساله مذکور مورد بررسی و ارزیابی قرار گرفت و مشاهده کردیم که برای مقدار ثابت N ، مقادیر خطا کاهش می یابد. یک ویژگی مهم روش های تفاضل متناهی که برای حل معادله برگرز به کار می رود این است که آیا این



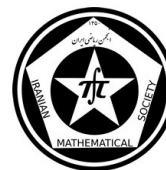
روش‌ها نوساناتی در حل ایجاد می‌کنند یا خیر. روشن است که ما نمی‌خواهیم چنین نوساناتی در جواب‌ها داشته باشیم. پس لازم است در جستجوی روش‌های دیگر باشیم. علاوه بر این در ادامه تحقیقات می‌توان مساله برگرز زمان کسری را برای مقادیر $\alpha > 1$ را نیز مورد بررسی و ارزیابی قرار داد.

مراجع

- [1] K. Miller and B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations*, Wiley, New York (1993).
- [2] I. Podlubny, *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*, Academic press, 1999.
- [3] E. Sousa, *Finite difference approximates for a fractional advection diffusion problem*, J. Comput. Phys, 2009.
- [4] M. . Meerschaert and C. Tadjeran, *Finite difference approximations for fractional advection-dispersion flow equations*, J. Comput. Appl. Math, **172** (2004), 65-77.
- [5] J. Beck and K. Khanin, *Burgers turbulence*, *Physics Reports*, BIT, **447** (2007), 1-66.
- [6] R.C. Mittal and G. Arora, *Numerical solution of the coupled viscous Burgers' equation*, *Communications in Nonlinear Science and Numerical Simulation*, BIT, **16** (2011), 1304-1313.
- [7] J. Ren and Z. zhongSun, *New approximations for solving the Caputo-type fractional partial differential equations*, (2016), 2625-2636.
- [8] A.H. Tabatabaei ,E. Shakour and M. Dehghan, *Some implicit methods for the numerical solution of Burgers' equation*, *Applied Mathematics and Computation*, (2007), 560-570.
- [9] A.Esen ,O , Tabsozan , *numerical solution of time fractional Burgers equation*, *Mathematica* , (2015), 167-185.

پست الکترونیکی: r.gholampour@alzahra.ac.ir

پست الکترونیکی: ashahrezaee@alzahra.ac.ir



توسعه روش های کانگ-تراب دو گامی بهین بدون حافظه به حالت آدابتیو با حافظه

محمد جواد لاله چینی[†]، ولی ترکاشوند[‡]

هیئت علمی دانشگاه فرهنگیان

چکیده

روش کانگ-تراب دوگامی بهین بدون حافظه برای حل معادلات غیر خطی به طور معمول به کار می رود. هدف اصلی این مقاله توسعه روش های کانگ-تراب دو گامی بهین بدون حافظه به حالت آدابتیو با حافظه است. به همین منظور در هر تکرار از تمام اطلاعات قبلی طوری استفاده می شود که بدون هیچ فراخوانی اضافی مرتبه همگرایی از ۴ به ۸ افزایش یابد و تا کنون چنین دست آوردی صورت نپذیرفته است. لذا این روش را یک روش آدابتیو یا تطبیقی گوئیم. به منظور مطالعه پایداری روش، از شیوه های به روز مانند استفاده از پویا نمایی، دینامیکی و فرکتالی استفاده می کنیم. در پایان علاوه بر پیاده سازی عددی روش گفته شده، به مقایسه آن با روش های با حافظه قبلی که در کل چنین ویژگی خاصی را تا به حال نداشته اند می پردازیم. واژه های کلیدی: معادله غیر خطی، روش با حافظه، روش تطبیقی، ضریب کارایی، حوزه جذب. طبقه بندی موضوعی [۲۰۱۰]: 65H05; 49M15; 65J15.

۱ پیش گفتار

در رشته های مهندسی و علوم کاربردی نیاز داریم جواب معادله غیر خطی $f(x) = 0$ را بیابیم. روش کانگ-تراب دوگامی نمونه ای از روش ها برای این منظور است [۲، ۴]. در این مقاله، یک روش تطبیقی با حافظه بر اساس روش دوگامی و بهین بدون حافظه کانگ و تراب ایجاد می کنیم [۱۰]. این عمل شامل تجزیه و تحلیل همگرایی همراه با پیاده سازی عددی است. روش تطبیقی پیشنهادی با حافظه، می تواند شاخص کارایی روش بهینه بدون حافظه را تا حد امکان بالا ببرد. در حالی که تمام اطلاعات موجود در فراخوانی تابع در تکرار قبلی برای این منظور مجدداً استفاده می شود. همچنین، در مورد پایداری عددی از نقطه نظر حوزه جذب نیز بحث می کنیم. در حالی که روش های بدون حافظه از اطلاعات تکرار کنونی استفاده می کنند، روش های یاد شده، بر استفاده از اطلاعات نه تنها تکرار فعلی، بلکه بر اطلاعات تکرارهای قبلی نیز تمرکز می کنند. از این رو، آنها از کارایی بالاتری برخوردار هستند. این مزیت روش با حافظه، بسیاری از



محققین را مجبور به توجه به چنین روش‌هایی کرده است [۸]. بسیاری از محققان روش‌های بسیار کارآمد با حافظه‌ای را توسعه داده‌اند که در آن تنها از اطلاعات دو تکرار گذشته استفاده می‌کنند. با این وجود، به اعتقاد ما، برای افزایش مرتبه همگرایی و همچنین شاخص کارایی یک روش بدون حافظه، هنوز امکان استفاده از اطلاعات موجود از فراخوانی‌های تابع از همه تکرارهای قبلی و نه تنها دو تکرار آخر وجود دارد. بنابراین، هدف اصلی ما توسعه یک روش سازگار با حافظه است، یعنی روشی که اطلاعات را از همه تکرارهای قبلی و نه تنها دو تکرار قبلی به کار می‌گیرد. در واقع، ما سعی می‌کنیم روش مطلوب دو مرحله‌ای بدون حافظه‌ای که توسط کانگ و تراب (۲۰۱) به کار گرفته شده است تا روش سازگار با حافظه را توسعه دهیم. این ایده ما را قادر می‌سازد تا تمام اطلاعات قبلی را دوباره استفاده کنیم و شاخص کارایی روش بهین بدون حافظه بسیار افزایش می‌یابد.

۲ توسعه حالت تطبیقی روش کانگ-تراب دوگامی با حافظه

این بخش به توسعه نوع جدیدی از روش‌های با حافظه می‌پردازد. روش‌های تطبیقی با حافظه تاکنون مورد بررسی قرار نگرفته است، بنابراین این موضوع جذابیت لازم را دارد. اصطلاح "روش تطبیقی با حافظه" به معنای استفاده مجدد از تمام داده‌های موجود برای افزایش مرتبه همگرایی یک روش بدون حافظه است. تأکید می‌کنیم که روش‌های توسعه یافته با حافظه در مباحث عددی فقط از اطلاعات دو تکرار گذشته استفاده می‌کنند. با این حال، تمام داده‌های موجود ممکن است برای استفاده مجدد در تکرار حاضر به کار آید. این یک اصل اساسی در ساخت روش تطبیقی با حافظه است. برای این منظور، ما در مرحله اول روش بدون مشتق دوگامی معرفی شده توسط کانگ-تراب را در نظر می‌گیریم.

$$\begin{cases} y_k = x_k - \frac{f(x_k)}{f[x_k, w_k]}, w_k = x_k + \gamma f(x_k) \\ x_{k+1} = y_k - \frac{f(w_k)}{f(w_k) - f(y_k)} \frac{f(y_k)}{f[x_k, y_k]}, \quad k = 0, 1, 2, \dots \end{cases} \quad (2.1)$$

که x_0 و γ به طور مناسب داده می‌شوند. این یک روش بهین بدون حافظه است زیرا از سه فراخوانی تابع در هر تکرار استفاده می‌کند و دارای مرتبه همگرایی چهار است. به معادلات خطای زیر نیاز داریم:

$$e_{k,w} = (1 + \gamma f'(\alpha))e_k + O(e_k^2), \quad (2.2)$$

$$e_{k,y} = c_2(1 + \gamma f'(\alpha))e_k^2 + O(e_k^3), \quad (2.3)$$

و

$$e_{k+1} = c_2(c_2^2 - c_2)(1 + \gamma f'(\alpha))e_k^4 + O(e_k^5) \quad (2.4)$$

که $e_k = x_k - \alpha$, $e_{k,w} = w_k - \alpha$, $e_{k,y} = y_k - \alpha$ و $c_k = \frac{f^{(k)}(\alpha)}{k!f'(\alpha)}$ در ادامه به دنبال توصیف ایده توسعه روش تطبیقی با حافظه بدست آمده از روش بهین بدون حافظه (۲۰۱) هستیم. اجازه دهید به معادلات خطا (۲۰۲)، (۲۰۳) و (۲۰۴) نگاه



کنیم. به وضوح دیده می شود که همه آنها شامل عامل اساسی $1 + \gamma f'(\alpha)$ می باشد. اگر $0 = 1 + \gamma f'(\alpha)$ ، ضریب e_k^4 در (۲۰۴) از بین می رود، بنابراین ما می توانیم مرتبه همگرایی را افزایش دهیم. از سوی دیگر، چون α ناشناخته است، پس این غیرممکن است. اما α می تواند به طور مؤثری تقریب شود، زیرا حد دنباله $\{x_n\}$ می باشد. جزئیات کامل این روش را در اینجا ارائه می دهیم. ما اشاره کردیم که α نامشخص است، پس محاسبه $f'(\alpha)$ غیرممکن است. شایان ذکر است که حتی اگر فرض کنیم α شناخته شده است، محاسبه $f'(\alpha)$ یک سیاست خوب نیست، زیرا فراخوانی تابع را افزایش می دهد؛ از این رو، بهینگی روش از بین می رود. در عوض، پیشنهاد می شود که $f'(\alpha)$ با استفاده از مفهوم درونیابی تقریب می شود. برای ایجاد تصویر واضحی از این موضوع، آن را به طور خلاصه بررسی می کنیم. این باعث انعطاف پذیری ما در توسعه روش تطبیقی با حافظه می شود. بنابراین، فرض کنیم که اطلاعاتی از تکرار فعلی و قبلی داشته باشیم، $f(x_k)$ ، $f(x_{k-1})$ ، $f(w_{k-1})$ و $f(y_{k-1})$. بنابراین، $f(t)$ را می توان به صورت زیر در نظر گرفت:

$$f(t) = N_{\gamma}(t; x_k, y_{k-1}, w_{k-1}, x_{k-1}) + E(t, x_k, y_{k-1}, w_{k-1}, x_{k-1}), \quad (205)$$

که $E(t)$ و $N_{\gamma}(t)$ چند جمله ای درونیاب نیوتن و معادله خطای آن می باشد. از این رو

$$f'(t) = N'_{\gamma}(t) + E'(t). \quad (206)$$

در نتیجه برای تقریب $f'(\alpha)$ در $0 = 1 + \gamma f'(\alpha)$ ، در نظر می گیریم:

$$\gamma = -\frac{1}{f'(\alpha)} \simeq -\frac{1}{N'_{\gamma}(x_k)} = \gamma_k. \quad (207)$$

اگر در نظر بگیریم تقریب $\gamma_k = -\frac{1}{N'_{\gamma}(x_k)}$ در هر تکرار در (۲۰۱)، سپس بدنبال آن روش باحافظه زیر را داریم:

$$\begin{cases} \gamma_k = -\frac{1}{N'_{\gamma}(x_k)} \\ w_k = x_k + \gamma_k f(x_k) & k = 0, 1, 2, \dots, \\ y_k = x_k - \frac{f(x_k)}{f[x_k, w_k]}, \\ x_{k+1} = y_k - \frac{f(w_k)}{f(w_k) - f(y_k)} \frac{f(y_k)}{f[x_k, y_k]}, \end{cases} \quad (208)$$

که γ_0 و x_0 به شکل مناسبی داده شده اند. گرچه روش (۲۰۸) به خوبی مورد بررسی قرار گرفته است [۸]، با این وجود، بازنگری خطای آن را دوباره با روشی متفاوت اما ساده تر بازبینی می کنیم. به لم زیر نیاز داریم:

لم ۰۱. اگر $\gamma_k = -1/N'_{\gamma}(x_k)$ ، سپس

$$1 + \gamma_k f'(\alpha) \sim e_{k-1} e_{k-1, w} e_{k-1, y}. \quad (209)$$



اثبات. از (۲۰۵)، داریم:

$$\begin{aligned} N_4^l(x_k) &= f'(x_k) - \frac{f^{(4)}(\xi)}{4!}(x_k - x_{k-1})(x_k - y_{k-1})(x_k - w_{k-1}) \\ &\sim f'(\alpha)(1 + c_4 e_{k-1} e_{k-1,w} e_{k-1,y}). \end{aligned} \quad (206)$$

از این رو

$$1 + \gamma_k f'(\alpha) = 1 - \frac{f'(\alpha)}{N_4^l(x_k)} \sim e_{k-1} e_{k-1,w} e_{k-1,y}. \quad (207)$$

□

قضیه بعدی دستورالعمل همگرایی روش با حافظه را نشان می دهد (۲۰۸).

قضیه ۲. اگر α یک ریشه ساده معادله $f(x) = 0$ و x_0 به اندازه کافی به α نزدیک باشد، آنگاه روش (۲۰۸) همگرایی مرتبه r خواهد داشت.

اثبات. فرض کنید $\{x_k\}, \{w_k\}, \{y_k\}$ به ترتیب دارای مرتبه‌های همگرایی r, p و q باشند. سپس

$$e_{k+1} \sim e_k^r \sim (e_{k-1}^r)^r \sim e_{k-1}^{r^2}, \quad (209)$$

$$e_{k,w} \sim e_k^p \sim (e_{k-1}^p)^r \sim e_{k-1}^{pr}, \quad (210)$$

و

$$e_{k,y} \sim e_k^q \sim (e_{k-1}^q)^r \sim e_{k-1}^{qr}. \quad (211)$$

همچنین، توسط لم (۱)، داریم:

$$1 + \gamma_k f'(\alpha) \sim e_{k-1}^{p+q+r}. \quad (212)$$

با جایگذاری (۲۱۵) در رابطه‌های (۲۰۲)، (۲۰۳) و (۲۰۴)، داریم:

$$e_{k,w} \sim e_{k-1}^{1+p+q+r}, \quad (213)$$

$$e_{k,y} \sim e_{k-1}^{1+p+q+2r}, \quad (214)$$

و

$$e_{k+1} \sim e_{k-1}^{2+2p+2q+4r}. \quad (215)$$



برای رسیدن به نتیجه، سمت راست رابطه‌های $(2012)-(2018)$ ، $(2013)-(2016)$ و $(2014)-(2017)$ با هم مقایسه می‌کنیم. بنابر این:

$$\begin{cases} 2 + 2p + 2q + 4r = r^2 \\ 1 + p + q + r = pr \\ 1 + p + q + 2r = qr. \end{cases} \quad (2019)$$

با حل دستگاه معادلات خطی (2019) جواب به صورت $p = 2$ ، $q = 3$ و $r = 6$ بدست می‌آید.

□

تا کنون، ما با استفاده مجدد از دو تکرار گذشته بدون آنکه توابع جدیدی را محاسبه کنیم، مرتبه همگرایی را از 4 به 6 تبدیل کرده‌ایم. به عبارت دیگر، با همان روش بهین اولیه بدون حافظه (201)، ما قادر به بهبود شاخص کارایی هستیم. شاخص کارایی روش (201) برابر با $E(1) = 4\frac{1}{2} \simeq 1,5874$ است و شاخص کارایی روش (208) برابر $E(8) = 6\frac{1}{2} \simeq 1,81712$ می‌باشد. در حال حاضر، هدف ما در نظر گرفتن اطلاعات موجود در سه تکرار برای تثبیت خود شتاب دهنده γ_k است. برای این منظور، ما $f(t)$ را با استفاده از چند جمله‌ای درونیاب نیوتن در نقاط $x_k, y_{k-1}, w_{k-1}, x_{k-1}, y_{k-2}, w_{k-2}, x_{k-2}$ تقریب می‌کنیم.

$$f(t) = N_{\varepsilon}(t) + E(t). \quad (2020)$$

از این رو

$$\gamma = -\frac{1}{f'(\alpha)} \simeq -\frac{1}{N'_{\varepsilon}(x_k)} = \gamma_k. \quad (2021)$$

روش جدید با حافظه زیر را پیشنهاد می‌کنیم

$$\begin{cases} \gamma_k = -\frac{1}{N'_{\varepsilon}(x_k)} \\ w_k = x_k + \gamma_k f(x_k) \\ y_k = x_k - \frac{f(x_k)}{f[x_k, w_k]}, \quad k = 0, 1, 2, \dots, \\ x_{k+1} = y_k - \frac{f(w_k)}{f(w_k) - f(y_k)} \frac{f(y_k)}{f[x_k, y_k]} \end{cases} \quad (2022)$$

که γ_0 و x_0 به شکل مناسبی در نظر می‌گیریم. برای بحث در مورد تجزیه و تحلیل همگرایی، به لم زیر نیاز داریم.



لم ۳. اگر $\gamma_k = -1/N'_\varepsilon(x_k)$ ، سپس

$$\begin{aligned} 1 + \gamma_k f'(\alpha) &\sim e_{k-1} e_{k-2} e_{k-1,w} e_{k-2,w} e_{k-1,y} e_{k-2,y} \\ &= \prod_{s=1}^2 e_{k-s} e_{k-s,w} e_{k-s,y}. \end{aligned} \quad (2.23)$$

اثبات. با مشتق گيري از (۲.۲۰) و قراردادن $t = x_k$ ، داریم

$$\begin{aligned} N'_\varepsilon(x_k) &= f'(x_k) - \frac{f^{(\vee)}(\xi)}{\vee!} \prod_{s=1}^2 (x_k - x_{k-s})(x_k - y_{k-s})(x_k - w_{k-s}) \\ &\sim f'(\alpha) (1 + c_\vee \prod_{s=1}^2 e_{k-s} e_{k-s,w} e_{k-s,y}). \end{aligned} \quad (2.24)$$

در نتیجه،

$$\begin{aligned} 1 + \gamma_k f'(\alpha) &= 1 - \frac{f'(\alpha)}{N'_\varepsilon(x_k)} = 1 - \frac{1}{1 + c_\vee \prod_{s=1}^2 e_{k-s} e_{k-s,w} e_{k-s,y}} \\ &\sim \prod_{s=1}^2 e_{k-s} e_{k-s,w} e_{k-s,y}. \end{aligned} \quad (2.25)$$

□

قضيه بعدي دستور مرتبه همگرایی روش جديد با حافظه را نشان مي دهد.

قضيه ۴. اگر α ریشه ساده معادله $f(x) = 0$ و x_0 مقدار اوليه که به اندازه کافي به α نزديک است باشد، آنگاه دنباله توليد شده $\{x_n\}$ توسط روش (۲.۲۲) دارای مرتبه همگرایی ۶.۳۱ است.

اثبات. فرض کنید $\{x_k\}, \{w_k\}, \{y_k\}$ به ترتيب دارای مرتبه هاي همگرایی r, p و q باشند. سپس

$$e_{k+1} \sim e_{k-2}^{r_2}, \quad (2.26)$$

$$e_{k-1,w} \sim e_{k-2}^{pr}, \quad (2.27)$$

و

$$e_{k-1,y} \sim e_{k-2}^{qr}. \quad (2.28)$$

همچنين، توسط لم (۳)، داریم:

$$\begin{aligned} 1 + \gamma_k f'(\alpha) &\sim \prod_{s=1}^2 e_{k-s} e_{k-s,w} e_{k-s,y} = e_{k-1} e_{k-2} e_{k-1,w} e_{k-2,w} e_{k-1,y} e_{k-2,y} \\ &= e_{k-2}^r e_{k-2}^{pr} e_{k-2}^p e_{k-2}^{qr} e_{k-2}^q e_{k-2}^q = e_{k-2}^{1+r} e_{k-2}^{p(1+r)} e_{k-2}^{q(1+r)} = e_{k-2}^{(1+p+q)(1+r)}, \end{aligned} \quad (2.29)$$



یا

$$1 + \gamma_k f'(\alpha) \sim e_{k-2}^{(1+p+q)(1+r)}. \quad (2.30)$$

با جایگذاری (2.30) در رابطه‌های (2.2)، (2.3) و (2.4)، داریم:

$$e_{k+1} \sim e_{k-2}^{2(p+q+1)(r+1)+4r^2}, \quad (2.31)$$

$$e_{k,w} \sim e_{k-2}^{(p+q+1)(r+1)+r^2}, \quad (2.32)$$

و

$$e_{k,y} \sim e_{k-2}^{(1+p+q)(1+r)+2r^2}. \quad (2.33)$$

برای رسیدن به نتیجه، سمت راست رابطه‌های (2.32)–(2.27)، (2.31)–(2.26) و (2.33)–(2.28) با هم مقایسه می‌کنیم. بنابر این:

$$\begin{cases} (p+q+1)(r+1) + r^2 = pr^2 \\ (p+q+1)(r+1) + 2r^2 = qr^2 \\ 2(p+q+1)(r+1) + 4r^2 = r^3. \end{cases} \quad (2.34)$$

با حل دستگاه معادلات خطی (2.34) بوسیله دستور Solve در نرم افزار ممتیکا جواب به صورت $q = p = \frac{1}{3}(1 + \sqrt{11})$ و $r = 3 + \sqrt{11} \simeq 6.31$ بدست می‌آید. □

روش‌های با حافظه (2.8) و (2.22) از اطلاعات موجود به ترتیب دو تکرار و 3 تکرار استفاده می‌کنند. لازم به ذکر است روش (2.8) در [8] مورد بررسی قرار گرفته است، در حالی که روش (2.22) جدید است، که در آن از تمام اطلاعات موجود از تکرار اول تا تکرار فعلی استفاده می‌شود.

فرض کنیم که ما در تکرار k ام هستیم. در هر تکرار سه فراخوانی تابع در دست است؛ بنابراین، تابع $f(t)$ را در نقاط زیر محاسبه می‌کنیم $\bigcup_{s=0}^{k-1} \{x_s, w_s, y_s\} \cup \{x_k\}$. ما $3k+1$ نقطه داریم، بنابراین با درونیایی نیوتن به یک چند جمله‌ای درجه $3k$ می‌رسیم. در حال حاضر، مشابه دو روش (2.8) و (2.22) می‌توانیم چند جمله‌ای درونیاب نیوتن زیر را برای تقریب پارامتر شتاب دهنده γ_k به کار ببریم.

$$f(t) - N_{3k}(t) = E_{3k+1}(t), \quad N_{3k}(t) = N_{3k}(t; x_k, y_{k-1}, w_{k-1}, x_{k-1}, \dots, y_0, w_0, x_0). \quad (2.35)$$



اگر فرض کنیم $\gamma_k = -\frac{1}{N'_{\mathfrak{r}_k}(x_k)}$ ، روش آدپتو با حافظه زیر بدست می آید.

$$\begin{cases} \gamma_k = -\frac{1}{N'_{\mathfrak{r}_k}(x_k)} \\ w_k = x_k + \gamma_k f(x_k), \\ y_k = x_k - \frac{f(x_k)}{f[x_k, w_k]}, \quad k = 0, 1, 2, \dots, \\ x_{k+1} = y_k - \frac{f(w_k)}{f(w_k) - f(y_k)} \frac{f(y_k)}{f[x_k, y_k]}, \end{cases} \quad (2.36)$$

که x_0 و γ_0 به شکل مناسبی در نظر می گیریم. برای بحث و آنالیز خطای این روش به لم زیر نیاز داریم.

لم ۵. اگر $\gamma_k = -1/N'_{\mathfrak{r}_k}(x_k)$ ، سپس

$$1 + \gamma_k f'(\alpha) \sim \prod_{s=1}^k e_{k-s} e_{k-s, w} e_{k-s, y}. \quad (2.37)$$

اثبات. با مشتق گیری از (۲۳۵) و قرار دادن $t = x_k$ داریم:

$$\begin{aligned} N'_{\mathfrak{r}_k}(x_k) &= f'(x_k) - \frac{f^{(\mathfrak{r}_k+1)}(\xi)}{(\mathfrak{r}_k+1)!} \prod_{s=1}^k (x_k - x_{k-s})(x_k - y_{k-s})(x_k - w_{k-s}) \\ &\sim f'(\alpha) (1 + c_{\mathfrak{r}_k-1} \prod_{s=1}^{\mathfrak{r}_k} e_{k-s} e_{k-s, w} e_{k-s, y}). \end{aligned} \quad (2.38)$$

بنابراین

$$\begin{aligned} 1 + \gamma_k f'(\alpha) &= 1 - \frac{f'(\alpha)}{N'_{\mathfrak{r}_k}(x_k)} = 1 - \frac{1}{1 + c_{\mathfrak{r}_k+1} \prod_{s=1}^k e_{k-s} e_{k-s, w} e_{k-s, y}} \\ &\sim \prod_{s=1}^k e_{k-s} e_{k-s, w} e_{k-s, y}. \end{aligned} \quad (2.39)$$

□

با قضیه زیر به تجزیه و تحلیل همگرایی روش تطبیقی با حافظه (۲۳۶) می پردازیم.

قضیه ۶. اگر α ریشه ساده معادله $f(x) = 0$ و x_0 مقدار اولیه که به اندازه کافی به α نزدیک است باشد، آنگاه مرتبه همگرایی روش (۲۳۶) از دستگاه زیر بدست می آید.

$$\begin{cases} (p+q+1) \sum_{s=0}^{k-1} r^s + r^k = r^k p, \\ (p+q+1) \sum_{s=0}^{k-1} r^s + \mathfrak{r} r^k = r^k q, \\ (p+q+1) \sum_{s=0}^{k-1} r^s + \mathfrak{r} r^k = r^{k+1}, \end{cases} \quad (2.40)$$



که r, p و q به ترتیب مرتبه همگرایی دنباله‌های $\{x_k\}$ ، $\{w_k\}$ و $\{y_k\}$ می‌باشند.

اثبات. معادلات زیر را داریم

$$e_1 \sim e_{\circ}^r, e_2 \sim e_{\circ}^{r^2}, \dots, e_{k+1} \sim e_{\circ}^{r^{k+1}}, \quad (2.41)$$

$$e_{\circ, w} \sim e_{\circ}^p, e_{1, w} \sim e_{\circ}^{rp}, \dots, e_{k, w} \sim e_{\circ}^{r^k p}, \quad (2.42)$$

و

$$e_{\circ, y} \sim e_{\circ}^q, e_{1, y} \sim e_{\circ}^{rq}, \dots, e_{k, y} \sim e_{\circ}^{r^k q}. \quad (2.43)$$

با به کارگیری روابط (2.41)-(2.42) در لم 5، داریم

$$\begin{aligned} 1 + \gamma_k f'(\alpha) &\sim e_{\circ}^{r^{k-1}} \dots e_{\circ}^r e_{\circ}^{r^{k-1} p} \dots e_{\circ}^{rp} e_{\circ}^{r^{k-1} q} \dots e_{\circ}^{rq} e_{\circ}^q \\ &\sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s}. \end{aligned} \quad (2.44)$$

با در نظر گرفتن این نتایج در معادلات خطای (2.2)-(2.4)، داریم

$$\begin{aligned} e_{k, w} &\sim (1 + \gamma_k f'(\alpha)) e_k \sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s} e_{\circ}^{r^k} \\ &\sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s + r^k}, \end{aligned} \quad (2.45)$$

$$\begin{aligned} e_{k, y} &\sim (1 + \gamma_k f'(\alpha)) e_k^y \sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s} e_{\circ}^{r^k} \\ &\sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s + r^k}, \end{aligned} \quad (2.46)$$

و

$$\begin{aligned} e_{k+1} &\sim (1 + \gamma_k f'(\alpha))^2 e_k^r \sim e_{\circ}^{2(p+q+1) \sum_{s=0}^{k-1} r^s} e_{\circ}^{r^k} \\ &\sim e_{\circ}^{2(p+q+1) \sum_{s=0}^{k-1} r^s + r^k}. \end{aligned} \quad (2.47)$$

برای رسیدن به نتیجه، سمت راست رابطه‌های (2.46)-(2.47)، (2.41)-(2.47) و (2.43)-(2.45) با هم مقایسه می‌کنیم. بنابر این:

$$\begin{cases} e_{\circ}^{r^k p} \sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s + r^k} \\ e_{\circ}^{r^k q} \sim e_{\circ}^{(p+q+1) \sum_{s=0}^{k-1} r^s + r^k} \\ e_{\circ}^{r^{k+1}} \sim e_{\circ}^{2(p+q+1) \sum_{s=0}^{k-1} r^s + r^k} \end{cases} \quad (2.48)$$

$$\left\{ \begin{array}{l} (p+q+1) \sum_{s=0}^{k-1} r^s + r^k = r^k p \\ (p+q+1) \sum_{s=0}^{k-1} r^s + 2r^k = r^k q \\ 2(p+q+1) \sum_{s=0}^{k-1} r^s + 4r^k = r^{k+1}. \end{array} \right. \quad (2.49)$$

□

که اثبات کامل است.

۳ بررسی نتایج عددی

در این بخش، ما سعی می‌کنیم روش تطبیقی با حافظه (۲۳) را اجرا کنیم تا ویژگی عملی آن را نشان دهیم. برای این منظور، در میان بسیاری از توابعی که بررسی شده‌اند، سه تابع در نظر گرفته و گزارش شده است. جدول (۱) توابع مورد آزمایش، و جدول (۲) نتایج عددی را نشان می‌دهد. روش بهین اولیه بدون حافظه (۲۱) دارای مرتبه همگرایی چهار است. با این حال، روش توسعه یافته و با حافظه (۲۳۶)، مرتبه همگرایی ۶، ۶۳۱ و ۶۳۶ به ترتیب برای k مساوی ۱، ۲ و ۳ دارد. نگاه دقیقتری به آن‌ها می‌اندازیم. همانطور که در ردیف اول جدول (۲) دیده می‌شود، شاخص‌های خطا با توجه به ساختار روش تطبیقی با حافظه (۲۳۶)، کاهش می‌یابد. در تکرار اول، مقدار خطا باتوان ۶-، مربوط به $k=1$ است، در حالی که برای تکرار دوم، با توان ۳۹- است. آخرین ستون COC ، تقریبی را نشان می‌دهد که مطابق با هدف مورد نظر در قضیه (۶) است. به طور مشابه، ردیف بعدی نتیجه گزارش شده مطابق با نظریه داده شده در قضیه (۶) است. آخرین مثال نشان می‌دهد که ممکن است که برای برخی از توابع مورد آزمایش، نتایج مورد نظر به خوبی پشتیبانی نشود. با این حال، مقدار COC که در حد ۶۲ بدست آمده، قابل قبول است. باید توجه داشت که اگر داده‌های اولیه را تغییر دهیم، احتمالاً این مشکل نیز حل می‌شود. در واقع، در اجرا، برای دستیابی به یک عملکرد خوب، نه فقط به کارکرد مستقیم، بلکه باید به تمام جوانب الگوریتم‌های ارائه شده آن توجه کافی داشته باشیم.

جدول ۱: توابع مورد آزمایش برای $\gamma = 0.1, \lambda = 0.1$

α	x_0	توابع
-۱.۶۵	-۱.۰۰	$f_1(t) = e^{(t^2-t)} - \cos(t^2 - 1) + t^3 + 1$
۰.۳۰	۰.۰۰	$f_2(t) = e^t \sin(t) + \log(t^4 - 3t + 1)$
۲.۰۰	۱.۰۰	$f_3(t) = \frac{1}{t^4} - t^2 - \frac{1}{t} + 1$



جدول ۲: نتایج رابطه (۲۰۳۶) برای توابع جدول ۱

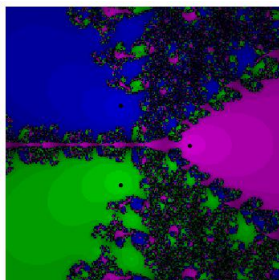
تابع	$ x_1 - \alpha $	$ x_2 - \alpha $	$ x_3 - \alpha $	COC
f_1	$0.6588(-1)$	$0.4012(-6)$	$0.4181(-39)$	۶۰۳۲۳۹
f_2	$0.1157(-1)$	$0.1492(-8)$	$0.8962(-53)$	۶۰۴۱۸۵
f_3	$0.9741(-5)$	$0.1095(-24)$	$0.9272(-149)$	۶۰۲۱۹۵

۴ رفتار دینامیکی

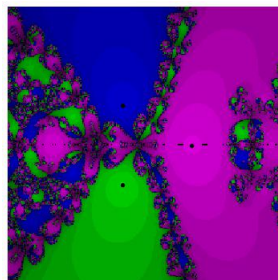
شکل‌های (۱) تا (۳) حوزه‌های جذب را برای هر دو روش (۲۰۱) و (۲۰۳۶) نشان می‌دهند. هدف از این تصاویر تعیین این است که کدام یک از روش‌های ذکر شده از نقطه نظر پایداری عددی بهتر است. تابع $p(z) = z^3 - 1$ برای آزمون پایداری و مرتبه همگرایی در نظر گرفته شده است. برای نشان دادن پایداری روش‌های (۲۰۱) و (۲۰۳۶)، خواص دینامیکی را بررسی کرده و بر روی صفحات دینامیکی مربوط به روش‌های تکراری تمرکز می‌کنیم. از نرم افزار متمتیکا برای انجام تحلیل مورد بحث استفاده می‌کنیم. که در آن یک صفحه گرافیکی در یک مستطیل با ابعاد $[-3, 3] \times [-3, 3]$ همراه با وضوح ۷۲ پیکسل نمایش داده می‌شود. فضاهای دینامیکی حوزه جذب برای چند جمله‌ای $p(z) = z^3 - 1$ را نشان می‌دهد. مجموعه جواب $p(z)$ برابر $\{1, \frac{1}{2}(-1 + i\sqrt{3}), \frac{1}{2}(-1 - i\sqrt{3})\}$ می‌باشد که به هر جواب یک رنگ اختصاص داده شده است. مناطق تیره، بیشترین شدت ناپایداری را دارند. در شکل‌های (۱) و (۳) پایداری روش تک پارامتری (۲۰۱) برای $\gamma = -0.1, 1$ نشان داده شده که در آن از چند جمله‌ای $p(z)$ استفاده شده است. همچنین در شکل (۳) پایداری روش برای $\gamma = -0.3332$ نشان داده شده که در آن باز هم چند جمله‌ای $p(z)$ به کار رفته است. همانطور که مشاهده می‌شود، در نگاه اول به نظر می‌رسد روش بهین بدون حافظه (۲۰۱)، از روش تطبیقی با حافظه (۲۰۳۶) پایدارتر است. در واقع، شکل (۱) با در نظر گرفتن پارامتر اولیه $\gamma = -0.1$ در روش بهین بدون حافظه (۲۰۱) را نشان می‌دهد. در حالی که شکل (۲) با در نظر گرفتن پارامتر تقریبی $\gamma = -0.3332$ که در رابطه (۲۰۱) محاسبه شده، می‌باشد. اگر ما پارامتر ثابت روش بهین بدون حافظه (۲۰۱) را به $\gamma = 1$ تغییر دهیم، شکل (۳) تولید می‌شود که حیرت انگیز است. برای نتیجه‌گیری می‌بینیم ثبات پایداری روش بهین بدون حافظه (۲۰۱) برای پارامترهای مختلف متفاوت است. برای بعضی از پارامترها پایداری روش تطبیقی با حافظه (۲۰۳۶) بیشتر است و برای برخی دیگر نیست. بنابراین، ما نمی‌توانیم بگوییم که کدام روش عموماً پایدارتر است. برای مطالعه برخی از جنبه‌های دیگر پایداری، می‌توان به [۱۱، ۱۳] مراجعه کرد.

۵ نتیجه‌گیری

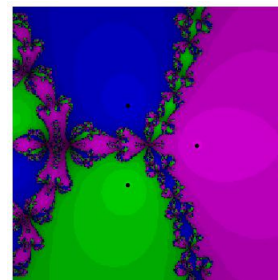
در این کار، ما برای اولین بار تلاش کردیم یک روش تطبیقی با حافظه بسازیم. در مقایسه با روش‌های قبلی با حافظه، این روش از تمام اطلاعات موجود از تکرار اول تا تکرار فعلی استفاده می‌کند، بنابراین شاخص کارایی را تا آنجا که ممکن است افزایش می‌دهد. لازم به ذکر است که ما فقط یک پارامتر شتاب دهنده را در نظر گرفتیم، در حالیکه روش‌های قبلی در این زمینه از یک، دو یا سه پارامتر استفاده می‌کنند.



شکل ۳: صفحه دینامیکی حوزه جذب
رابطه (۲۰۱) برای $\gamma = 1$.



شکل ۲: صفحه دینامیکی حوزه جذب
رابطه (۲۰۳۶) برای $\gamma = -0.3332$.



شکل ۱: صفحه دینامیکی حوزه جذب
رابطه (۲۰۱) برای $\gamma = -0.1$.

که در آن افزایش چنین شتاب دهنده‌هایی ممکن است موجب پیچیدگی و ناپایداری عددی شود. باز هم تاکید می‌کنیم که اگرچه ممکن است تمام اطلاعات را از تکرار اول تا تکرارهای کنونی حداقل از لحاظ تئوری مورد استفاده مجدد قرار گیرد. اما در این مطالعه موردی، تنها از اطلاعات سه تکرار اخیر استفاده کردیم و مشاهده شد تأثیر زیادی در افزایش شاخص کارایی دارد. همچنین، اشاره کردیم که از نقطه نظر حوزه جذب، در مورد ثبات عددی روش پیشنهادی با حافظه، نمی‌توانیم به نتیجه برسیم.

مراجع

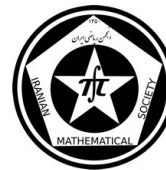
- [1] J. Džunić and M.S. Petković, A cubically convergent steffensen-like method for solving nonlinear equations, Applied Mathematics Letters 25 (2012), 1881-1886.
- [2] J.F. Traub, Iterative methods for the solution of equations, Prentice-Hall, Englewood Cliffs, New Jersey, 1964.
- [3] J.F. Stephenson, Remarks on iteration, Skand. Aktuarietidskr. 16 (1933) 64-72.
- [4] J.F. Traub, Iterative Methods for the Solution of Equations, Prentice Hall, New York, 1964.
- [5] Jovana Džunić, On efficient two-parameter methods for solving nonlinear equations, Numer Algor (2013) 63:549-569 DOI 10.1007/s11075-012-9641-3
- [6] M. Salimi, T. Lotfi, S. Sharifi, S. Siegmund, Optimal Newton-Secant like methods without memory for solving nonlinear equations with its dynamics, International Journal of Computer Mathematics 94, (2017) 1759-1777.



- [7] M. Zaka Ullah , S. Kosari , F. Soleymani , F. Khaksar Haghani , A. S. Al-Fhaid, A super-fast tri-parametric iterative .method with memory, *Applied Mathematics and Computation* 289 (2016) 486-491.
- [8] M.S. Petkovic, B. Neta , L.D. Petkovic , J. Dzunic, Multipoint methods for solving nonlinear equations: a survey, *Appl. Math. Comput.* 226 (2014) 635-660.
- [9] M. J. Lalehchini, Taher Lotfi, Katayoun Mahdiani, Adaptive Steffensen-like methods with memory for solving nonlinear equations with the highest possible efficiency indices, Accepted in *International Journal of Industrial Mathematics (IJIM)* 11 (2019) 337-345.
- [10] M. J. Lalehchini, Taher Lotfi, Katayoun Mahdiani, On Developing an Adaptive Free-Derivative Kung and Traub's Method with Memory, Accepted in *Journal of Mathematical Extension (JME)* 14 (2020).
- [11] P. Bakhtiari, A. Cordero, T. Lotfi, K. Mahdiani, J.R. Torregrosa, Widening basins of attraction of optimal iterative methods, *Nonlinear Dynamics* 87 (2), 2017 913-938.
- [12] S. Artidiello, A. Cordero, J. Torregrosa, M. Vassileva, Two weighted eight-order classes of iterative root-finding methods, *International Journal of Computer Mathematics*, 92, (2015) 1790-1805.
- [13] S. Amat, S. Busquier, C. Bermudez, A.A. Magrenan, On the election of the damped parameter of a two-step relaxed Newton-type method, *Nonlinear Dynamics* 84 (1), 2016 9-18.

پست الکترونیکی: lalehchini@gmail.com

پست الکترونیکی: torkashvand1978@gmail.com



حرارت درمانی تومور سرطانی: رویکرد مساله معکوس

ندا بیابان نورد[†] محمد دهقان نیری، محمد علی نژاد مفرد و[‡]

دانش آموخته کارشناسی ارشد ریاضی کاربردی دانشگاه بجنورد[†]

استادیار ریاضی کاربردی دانشگاه بجنورد[‡]

چکیده

مطالعه انتقال حرارت در بافت های زنده بسیار حائز اهمیت است. برخی از بیماری ها از جمله سرطان با حرارت دادن به تومور سرطانی درمان می شوند. این روش درمانی را حرارت درمانی یا هایپرترمی می نامند. تولید حرارت در بافت به وسیله مولدهای پرتو سونوگرافی، میکروویو و غیره صورت می گیرد. در این روش سعی می شود در عین اینکه بافت سرطانی تخریب شده و از بین برود، کمترین آسیب به بافت های سالم اطراف آن وارد گردد. در این پژوهش مساله تقریب مولد گرما به همراه توزیع دمایی درون تومور به عنوان یک مساله معکوس در نظر گرفته می شود. سپس با تبدیل این مساله به یک مساله بهینه سازی با استفاده از روش گرادیان مزدوج تابع مجهول تقریب زده می شود. داده های اضافی مورد نیاز از حل مساله کنترل بهینه متناظر در پایان حرارت دهی در نقاط خاصی از تومور به دست آمده اند. نتایج نشان می دهند که رویکرد مساله معکوس برای حل عددی مساله می تواند تابع کنترل بهینه را به خوبی تقریب بزند. واژه های کلیدی: هایپرترمی، مساله معکوس، حل عددی معادلات دیفرانسیل با مشتقات جزئی، روش گرادیان مزدوج. طبقه بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش گفتار

سرطان بیماری کشنده ای است که از زمان های قدیم انسان ها با آن درگیر بوده اند و به معنی رشد غیر معمول سلول های بدن است. راه های مختلفی برای درمان سرطان وجود دارند که عبارتند از: جراحی، شیمی درمانی، پرتو درمانی، حرارت درمانی و غیره. رویکرد حرارت درمانی یا هایپرترمی به این صورت است که با افزایش کنترل شده دما به بیش از ۴۰ درجه سانتی گراد باعث تخریب تومور سرطانی می شود. این درمان یک درمان غیر تهاجمی سرطان است و هدف از گرما دادن، لخته شدن خون و بافت مردگی تومور است.

۲ مدل‌بندی ریاضی

معادله حاکم، معادله انتقال حرارت زیستی پنس با شرط اولیه به صورت زیر است:

$$\begin{cases} \frac{\partial T(x, t)}{\partial t} = \frac{k}{\rho c} \frac{\partial^2 T(x, t)}{\partial x^2} + \frac{w_b \rho_b c_b}{\rho c} (T_a - T(x, t)) + \frac{Q_m}{\rho c} + \frac{Q_r(t)}{\rho c}, & 0 \leq x \leq L, 0 \leq t \leq t_f \\ T(x, 0) = T_0(x), & 0 \leq x \leq L \end{cases} \quad (2.1)$$

که در آن T معرف دما، t زمان، x مکان، k ضریب هدایت گرمایی، ρ چگالی بافت، c گرمای ویژه بافت، w_b پرفیوژن خون، ρ_b چگالی خون، c_b گرمای ویژه خون، T_a دمای شریانی، Q_m تولید حرارت متابولیک (حرارت حاصل از سوخت و ساز بدن)، $Q_r(t)$ مولد گرمای خارج از بافت، L طول بافت، t_f مدت زمان فرآیند و T_0 دمای اولیه است. توزیع اولیه دما در بافت، $T_0(x)$ ، متناظر با جواب حالت پایدار* معادله پنس با فرض $Q_r = 0$ و تحت شرایط مرزی زیر است

$$x = 0 : -k \frac{dT_0}{dx} = h [T_\infty - T_0],$$

$$x = L : T_0 = T_a,$$

که در آن h ضریب انتقال حرارت جابجایی و T_∞ دمای هوای اطراف آن است.

۳ مساله معکوس

در مساله مستقیم همه پارامترها و توابع معلوم هستند و هدف از حل مساله، تعیین توزیع دمای مجهول T در بافت تومور است [۱]. حال اگر مولد گرمای $Q_r(t)$ را نیز مجهول در نظر بگیریم، با مساله‌ای معکوس مواجهیم. داده‌های اضافی برای حل مساله معکوس از داده‌های دمایی حاصل از حل مساله مستقیم به ازای $Q_r(t)$ بهینه (حاصل از حل مساله کنترل بهینه متناظر [۳]) در پایان فرآیند درمان (t_f) به دست می‌آیند. برای یافتن تابع مولد حرارتی مجهول کافی است که حداقل مربعات خطا بین داده‌های اضافی و داده‌های دمایی حاصل از حل مساله معکوس در زمان t_f را کمینه کنیم یعنی:

$$\min s_k = \sum_{i=1}^5 (T_m(x_i) - \bar{T}_m(x_i))^2 \quad (3.1)$$

که در آن $T_m(x_i)$ و $\bar{T}_m(x_i)$ به ترتیب دماهای محاسباتی و داده‌های اضافی در (x_i, t_f) در تکرار k ام هستند. این کار به کمک الگوریتم گرادینان مزدوج انجام می‌پذیرد.

*Stable state

نخست تابع مجهول $Q_r(t)$ را به صورت‌های پارامتری زیر در نظر می‌گیریم:

$$Q_r(t) = at + b, \quad (3.2)$$

در این حالت یافتن تابع مجهول $Q_r(t)$ به یافتن بردار مجهول دو عضوی $p = [a, b]^T$ تقلیل می‌یابد. صورت دیگر، تابعی دوضابطه‌ای است:

$$Q_r(t) = \begin{cases} a, & 0 \leq t \leq \tau \\ b, & \tau < t \leq t_f \end{cases} \quad (3.3)$$

و از این رو در این حالت بردار مجهول برابر $p = [a, \tau, b]^T$ و سه عضوی خواهد بود. در ادامه به ازای هر از این صورت‌ها برای تابع مجهول، بردار مجهول را محاسبه می‌کنیم.

۱۰۳ روش گرادیان مزدوج برای تخمین پارامتر

برای تخمین پارامترهای مجهول در مساله معکوس از روش گرادیان مزدوج استفاده می‌کنیم. این روش، یک روش تکراری برای حل (۳۰۱) است [۲]:

$$p^{k+1} = p^k - \mu^k d^k, \quad k = 0, 1, 2, \dots, \quad (3.4)$$

که در آن μ^k اندازه طول گام، d^k جهت کاهش و بالانویس k تعداد تکرار است. جهت کاهش از رابطه زیر به دست می‌آید

$$d^k = \nabla s(p^k) + v^k d^{k-1}, \quad k = 0, 1, 2, \dots, \quad (3.5)$$

که در آن $\nabla s(p^k)$ جهت گرادیان است:

$$\nabla s(p^k) = -\nabla(J^k)[T_m(x_i) - \bar{T}_m(x_i)], \quad (3.6)$$

J^k ماتریس حساسیت است که به صورت زیر تعریف می‌شود:

$$J^k = \left[\frac{\partial \bar{T}^T(p^k)}{\partial p^k} \right]^T, \quad (3.7)$$

و ضریب v^k از رابطه زیر به دست می‌آید:

$$v^k = \frac{\sum_{j=1}^{\Psi} [\nabla s(p^k)]_j [\nabla s(p^k) - \nabla s(p^{k-1})]_j}{\sum_{j=1}^{\Psi} [\nabla s(p^{k-1})]_j^2}, \quad k = 0, 1, 2, \dots, \quad (3.8)$$



که $[\nabla s(p^k)]_j$ جهت شیب در تکرار k ام است. ماتریس حساسیت نقش مهمی را در مساله تخمین پارامتر دارد. سرانجام اندازه طول گام μ^k با کمینه سازی $s(p^{k+1})$ به دست می آید:

$$\mu^k = \frac{[J^k d^k]^T [\bar{T}_m(p^k) - T_m(p^k)]}{[J^k d^k]^T [J^k d^k]}. \quad (39)$$

معیار توقف روش تکراری عبارت است از

$$s(p) \leq \epsilon.$$

این روش برای حل انواع مسائل معکوس شامل تخمین پارامترهای مجهول استفاده می شود [5].

۴ نتایج اصلی

مساله معکوس درمان تومور سرطانی را با پارامترهای جدول ۱ در نظر بگیرید. با پیاده سازی الگوریتم حل عددی [4] این مساله در نسخه

جدول ۱: مقادیر پارامترهای مساله [3]

واحد	مقدار	پارامتر	واحد	مقدار	پارامتر
$J/kg^\circ C$	377°	c	kg/m^3	۹۹۸	ρ
$J/kg^\circ C$	377°	c_b	kg/m^3	۹۹۸	ρ_b
$W/m^2 \cdot ^\circ C$	۶	h	$\frac{ml \text{ of blood}}{m^2 \text{ of tissue.s}}$	۱۰۰۰	w_b
$W/m^\circ C$	۰.۵	k	s	۶۰۰	t_f
m	۰.۰۱	L	$^\circ C$	۴۳	T_r
W/m^3	۳۳۸۰۰	Q_m	W/m^3	6.6×10^7	Q_{total}
$^\circ C$	۲۵	T_∞	W/m^3	۲۲۰۰۰۰	Q_{max}
$^\circ C$	۳۷	T_a	W/m^3	۰	Q_{min}

۱۰ از نرم افزار متنتیکا، مقادیر پارامترهای مجهول در جداول ۱ و ۲ آمده است. با مقایسه تابع مولد گرمای حاصل از رویکرد مساله معکوس

جدول ۲: تخمین پارامترهای تابع مجهول (۳۰۲) به ازای $\epsilon = 10^{-3}$

s_k	b^k	a^k	k	$[a^\circ, b^\circ]^T$
۰/۰۰۰۰۲۵۵۱۱۸	۴۲۵۵۷/۵	۳۱۴/۷۳۲	۶	$[0, 0]^T$



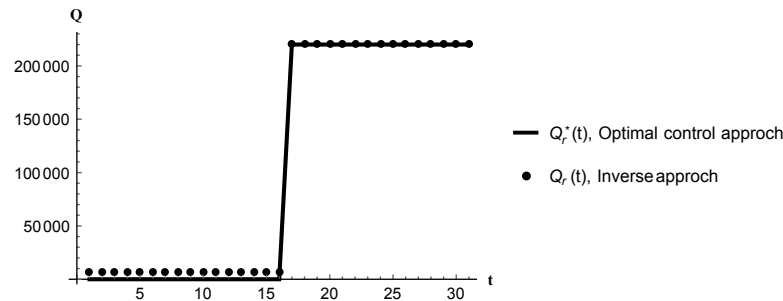
جدول ۳: تخمین پارامترهای تابع مجهول (۳۰۳) به ازای $\epsilon = 10^{-8}$

$[a^\circ, \tau^\circ, b^\circ]^T$	k	a^k	τ^k	b^k	s_k
$[0, 100, 0]^T$	۴۶	۶۷۷۴۶۷	۳۰۵/۱۷۵	۲۲۰۰۱۵	$5/49968 \times 10^{-9}$

با تابع مولد حاصل از رویکرد کنترل بهینه یعنی [۱]

$$Q_r^*(t) = \begin{cases} 0, & 0 \leq t \leq 300 \\ 220000, & 300 < t \leq t_f \end{cases} \quad (41)$$

واضح است که رویکرد مساله معکوس با روش حل پیشنهادی قادر است که تقریب بسیار خوبی برای تابع مجهول ارائه دهد. نمودار صورت دوضابطه‌ای $Q_r(t)$ حاصل از دو رویکرد مساله معکوس و رویکرد کنترل بهینه در شکل زیر رسم شده است.



شکل ۱: نمودار تابع (۳۰۳) با دو رویکرد مساله معکوس و رویکرد کنترل بهینه

مراجع

- [۱] ن. بیابان‌نورد، مسائل بهینه‌سازی برای معادله انتقال حرارت زیستی، پایان‌نامه کارشناسی ارشد، دانشگاه بجنورد، (۱۳۹۷).
- [2] A. Shidfar, B. Jazbi, M. Alinejadmoftad, *Inverse estimation of the pulse parameters of a time-varying laser pulse to obtain desired temperature at the material surface*, Optics & Laser Technology, **44** (2012), 1675–1680.
- [3] A.V. Kuznetsov, *Optimization problems for bioheat equation*, International Communications in Heat and Mass Transfer, **33** (2006), 537–543.

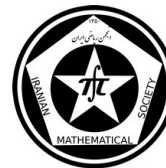


-
- [4] E. Kreyszig, *Advanced Engineering Mathematics*, Wiley, New York, 2011.
- [5] M.N. Ozisik, H.R.B. Orlande, *Inverse Heat transfer: A Fundamentals and Applications*, Taylor & Francis, New York, 2000.

پست الکترونیکی: biabannavardneda@gmail.com

پست الکترونیکی: m.alinejad@ub.ac.ir

پست الکترونیکی: m.dehghan@ub.ac.ir



حل دستگاه معادلات انتگرال آبل با استفاده از چندجمله‌ای های ژاکوبی انتقال یافته

نادیا دولت آبادی[†] و محبوبه زارعی[‡]

[†] دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی شاهرود، سمنان، ایران

[‡] دانشکده ریاضی و علوم کامپیوتر، دانشگاه صنعتی شاهرود، سمنان، ایران

چکیده

در این مقاله، برای حل معادلات انتگرال غیر خطی آبل با کمک چندجمله‌ای های ژاکوبی انتقال یافته و ماتریس عملیاتی حاصل ضرب، قسمت انتگرالی معادله را ساده نموده، سپس باروش هم محلی آن را به دستگاه معادلات جبری تبدیل می‌کنیم. با حل دستگاه، جواب تقریبی دستگاه معادلات انتگرال آبل را به دست آوریم. این روش حجم محاسبات را کم می‌کند که در مثال عددی صحت و کارایی آن را نشان داده ایم.

واژه‌های کلیدی: دستگاه معادله انتگرال آبل، چندجمله‌ای ژاکوبی انتقال یافته، روش هم محلی.

طبقه‌بندی موضوعی [۲۰۱۰]: 97N40, 97R20.

۱ مقدمه

معادله آبل یک نوع معادله انتگرال منفرد است که در فیزیک پلاسما، مکانیک جامدات و غیره کاربرد دارد [۱]. در این مقاله، دستگاه معادلات انتگرال آبل

$$\begin{cases} a_1(x) \int_0^x \frac{v^{r_1}(t)u^{s_1}(t)}{(x-t)^{\eta_1}} dt + a_2(x) \int_0^x \frac{v^{r_2}(t)u^{s_2}(t)}{(x-t)^{\eta_2}} dt = g_1(x), \\ a_3(x) \int_0^x \frac{v^{r_3}(t)u^{s_3}(t)}{(x-t)^{\eta_3}} dt + a_4(x) \int_0^x \frac{v^{r_4}(t)u^{s_4}(t)}{(x-t)^{\eta_4}} dt = g_2(x), \end{cases} \quad (1.1)$$

با توزیع درجه حرارت در امتداد سطحی که انتقال گرما را به تعادل می‌رساند برای $i = 1, \dots, 4$ ، $0 < \eta_i < 1$ ، که s_i و r_i اعداد حسابی، $a_i(x)$ ها چندجمله‌ای، $g_1(x)$ و $g_2(x)$ توابعی پیوسته و معلوم و $u(t)$ و $v(t)$ توابع مجهول هستند، بررسی می‌شود.

۲ چندجمله‌ای ژاکوبی انتقال یافته

فرم بازگشتی چندجمله‌ای ژاکوبی انتقال یافته در بازه $[0, 1]$ به ازای $i = 0, 1, 2, \dots$ و $\alpha, \beta > -1$ به صورت

$$P_0^{(\alpha, \beta)}(x) = 1, \quad P_1^{(\alpha, \beta)}(x) = \frac{(\alpha + \beta + 2)(2x - 1)}{2} + \frac{\alpha - \beta}{2},$$

$$P_{i+1}^{(\alpha, \beta)}(x) = A(\alpha, \beta, i)P_i^{(\alpha, \beta)}(x) + B(\alpha, \beta, i)(2x - 1)P_i^{(\alpha, \beta)}(x) - D(\alpha, \beta, i)P_{i-1}^{(\alpha, \beta)}(x)$$

که در آن

$$A(\alpha, \beta, i) = \frac{(2i + \alpha + \beta + 1)(\alpha^2 - \beta^2)}{2(i + 1)(i + \alpha + \beta + 1)(2i + \alpha + \beta)},$$

$$B(\alpha, \beta, i) = \frac{(2i + \alpha + \beta + 2)(2i + \alpha + \beta + 1)}{2(i + 1)(i + \alpha + \beta + 1)},$$

$$D(\alpha, \beta, i) = \frac{(i + \alpha)(i + \beta)(2i + \alpha + \beta + 2)}{(i + 1)(i + \alpha + \beta + 1)(2i + \alpha + \beta)}.$$

می‌باشد و فرم صریح آن که نسبت به تابع وزن $w^{(\alpha, \beta)}(x)$ برای $\alpha, \beta > -1$ در بازه $[0, 1]$ متعامدند، به شرح زیر است:

$$P_i^{(\alpha, \beta)}(x) = \sum_{k=0}^i \gamma_k^{(i)} x^k = \sum_{k=0}^i \frac{(-1)^{i-k} \Gamma(i + \beta + 1) \Gamma(i + k + \alpha + \beta + 1)}{\Gamma(k + \beta + 1) \Gamma(i + \alpha + \beta + 1) (i - k)! k!} x^k, \quad i = 0, 1, \dots \quad (2.1)$$

تابع وزن انتقال یافته و شرط عمود بودن در بازه $[0, 1]$ به ترتیب به صورت

$$w^{(\alpha, \beta)}(x) = (1 - x)^\alpha x^\beta,$$

$$\int_0^1 P_i^{(\alpha, \beta)}(x) P_j^{(\alpha, \beta)}(x) w^{(\alpha, \beta)}(x) dx = \theta_i \delta_{ij},$$

هستند که δ_{ij} دلتای کرونکر است و $\theta_i = \frac{\Gamma(i + \alpha + 1) \Gamma(i + \beta + 1)}{(2i + \alpha + \beta + 1) i! \Gamma(i + \alpha + \beta + 1)}$ هر تابع $u(x)$ در فضای $L^2[0, 1]$ را می‌توان برحسب چندجمله‌ای‌های ژاکوبی به صورت

$$u(x) \simeq u_N(x) = \sum_{j=0}^N c_j P_j^{(\alpha, \beta)}(x) = \phi^T(x) C = C^T \phi(x), \quad (2.2)$$

تقریب زد که c_j بردار ضرایب C و بردار ژاکوبی $\phi(x)$ به صورت

$$c_j = \frac{1}{\theta_j} \int_0^1 u(x) P_j^{(\alpha, \beta)}(x) w^{(\alpha, \beta)}(x) dx, \quad j = 0, 1, \dots, N$$

$$C = [c_0, c_1, \dots, c_N]^T, \quad \phi(x) = [P_0^{(\alpha, \beta)}(x), P_1^{(\alpha, \beta)}(x), \dots, P_N^{(\alpha, \beta)}(x)]^T \quad (2.3)$$

تعریف می‌شوند. [۲] همچنین، ماتریس عملیاتی حاصل ضرب متناظر با چندجمله‌ای ژاکوبی انتقال یافته به صورت

$$\phi(x) \phi^T(x) U \simeq \tilde{U} \phi(x), \quad (2.4)$$

می‌باشد که U بردار ستونی $(N + 1)$ تایی دلخواه و $\phi(x)$ بردار ژاکوبی معرفی شده در (۲.۳) است. [۳]

۳ آنالیز روش

با استفاده از قضیه ۱ و ۲ روش حل را بیان می‌کنیم. [۴]

قضیه ۱. فرض کنید $\phi(x)$ بردار ژاکوبی (۲۳) و $0 < \eta < 1$ باشد، بنابراین

$$\int_0^x \frac{\phi(t)}{(x-t)^\eta} dt \simeq \mathbf{B}^{(\eta)} \phi(x),$$

است که در آن $\mathbf{B}^{(\eta)}$ یک ماتریس $(N+1) \times (N+1)$ است و عناصر آن از رابطه

$$B_{ij}^{(\eta)} = b(i, j) = \sum_{k=0}^i \frac{\gamma_k^{(i)} \Gamma(k+1) \Gamma(1-\eta)}{\Gamma(k-\eta+2)} a_j^{(k-\eta+1)}, \quad i, j = 0, 1, \dots, N$$

محاسبه می‌شود که $a_j^{(n)}$ ضرایب ژاکوبی در بسط تابع x^n و $\gamma_k^{(i)}$ در فرم صریح چندجمله‌ای ژاکوبی انتقال یافته و رابطه (۲۱) معرفی شد.

قضیه ۲. فرض کنید $u(x) \simeq U^T \phi(x)$ و $v(x) \simeq V^T \phi(x)$ تقریب‌های ژاکوبی معرفی شده در (۲۲) و $\tilde{U}$ و $\tilde{V}$ ماتریس‌های عملیاتی حاصل ضرب تعریف شده در (۲۴) باشند، $B_{s-1} = (\tilde{U}^T)^{s-1} U$ قرار می‌دهیم، بنابراین داریم:

$$v^r(x) u^s(x) \simeq V^T (\tilde{V})^{r-1} \tilde{B}_{s-1} \phi(x), \quad r, s = 0, 1, \dots$$

اکنون با استفاده از قضیه ۱، ۲ و روش هم محلی انتگرال زیر را ساده می‌کنیم:

$$\begin{aligned} \int_0^x \frac{v^r(t) u^s(t)}{(x-t)^\eta} dt &= \int_0^x \frac{V^T (\tilde{V})^{r-1} \tilde{B}_{s-1} \phi(t)}{(x-t)^\eta} dt \\ &= V^T (\tilde{V})^{r-1} \tilde{B}_{s-1} \int_0^x \frac{\phi(t)}{(x-t)^\eta} dt = V^T (\tilde{V})^{r-1} \tilde{B}_{s-1} \mathbf{B}^{(\eta)} \phi(x), \end{aligned} \quad (31)$$

که بردار B_{s-1} از قضیه ۲ و $\tilde{B}_{s-1}$ ماتریس عملیاتی حاصل از این بردار است. حال با استفاده از (۳۱)، دستگاه (۱۱) را به صورت

$$\begin{cases} a_1(x) V^T (\tilde{V})^{r_1-1} \tilde{B}_{s_1-1} \mathbf{B}^{(\eta_1)} \phi(x) + a_2(x) V^T (\tilde{V})^{r_2-1} \tilde{B}_{s_2-1} \mathbf{B}^{(\eta_2)} \phi(x) - g_1(x) \approx 0, \\ a_3(x) V^T (\tilde{V})^{r_3-1} \tilde{B}_{s_3-1} \mathbf{B}^{(\eta_3)} \phi(x) + a_4(x) V^T (\tilde{V})^{r_4-1} \tilde{B}_{s_4-1} \mathbf{B}^{(\eta_4)} \phi(x) - g_2(x) \approx 0, \end{cases} \quad (32)$$

بازنویسی کرده که در آن $B_{s_i-1} = (\tilde{U}^T)^{s_i-1} U$. با جایگذاری $(N+1)$ ریشه‌ی چندجمله‌ای ژاکوبی انتقال یافته درجه $(N+1)$ ، $P_{N+1}^{(\alpha, \beta)}(x)$ در (۳۲) به دستگاه معادلات جبری غیرخطی شامل $2(N+1)$ معادله و $2(N+1)$ مجهول می‌رسیم که می‌توان آن را با استفاده از روش نیوتن حل کرد. پس از حل دستگاه و تعیین بردارهای ضرایب U و V ، جواب تقریبی دستگاه معادله انتگرال (۱۱) از

$$u(x) \simeq U^T \phi(x), \quad v(x) \simeq V^T \phi(x)$$

به دست می‌آید. لازم به ذکر است که دقت این روش به مقدار پارامترهای دلخواه α و β چندان بستگی ندارد، اما بهتر است که مقادیرشان را یکسان انتخاب کنیم تا عبارت A در رابطه بازگشتی صفر گردد و حجم محاسبات کاهش یابد.

۴ مثال عددی

در این بخش، صحت و کارایی روش پیشنهادی را با ارائه چند مثال عددی بررسی می‌کنیم. مثال‌ها با برنامه *Matlab* حل شده است.

مثال ۳. دستگاه معادلات انتگرال آبل خطی

$$\begin{cases} (x^2 - 1) \int_0^x \frac{u(t)}{(x-t)^{\frac{1}{2}}} dt + x \int_0^x \frac{v(t)}{(x-t)^{\frac{1}{2}}} dt = \frac{16}{15}x^{\frac{3}{2}} - \frac{16}{15}x^{\frac{5}{2}} + \frac{27}{40}x^{\frac{11}{2}} + \frac{223}{440}x^{\frac{13}{2}}, \\ x^3 \int_0^x \frac{u(t)}{(x-t)^{\frac{1}{2}}} dt + (1-x) \int_0^x \frac{v(t)}{(x-t)^{\frac{1}{2}}} dt = \frac{128}{231}x^{\frac{7}{2}} + \frac{125}{252}x^{\frac{9}{2}} - \frac{125}{1197}x^{\frac{13}{2}} - \frac{625}{1596}x^{\frac{15}{2}}. \end{cases}$$

برای $0 \leq x \leq 1$ با نقطه $0, 0, \dots, 0]_{N+1}$ در نظر بگیرید. جواب‌های دقیق این دستگاه $u(x) = x^2 + x^3$ و $v(x)$ هستند. طبق روش حل ارائه شده، با در نظر گرفتن بسط زاکوبی توابع مجهول $u(x)$ و $v(x)$ ، دستگاه جبری خطی زیر به دست می‌آید:

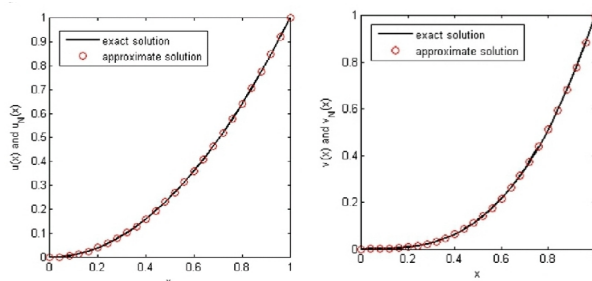
$$\begin{cases} (x^2 - 1) \mathbf{B}^{(\frac{1}{2})} \phi(x) U + x \mathbf{B}^{(\frac{1}{2})} \phi(x) V - g_1(x) \approx 0, \\ x^3 \mathbf{B}^{(\frac{1}{2})} \phi(x) U + (1-x) \mathbf{B}^{(\frac{1}{2})} \phi(x) V - g_2(x) \approx 0, \end{cases}$$

جدول ۱، ماکزیمم خطای مطلق $u(x)$ و $v(x)$ را به ازای مقادیر مختلف N و برای $\alpha = \beta = 3$ گزارش می‌کند و گویای آن است که با افزایش N خطا کاهش می‌یابد. شکل ۱، جواب دقیق و تقریبی را برای $\alpha = \beta = \frac{1}{2}$ و $N = 20$ بر روی نمودار مقایسه می‌کند.

جدول ۱: ماکزیمم خطای مطلق مثال ۳ با $\alpha = \beta = 3$

N	۵	۱۰	۱۵	۲۰	۲۵
خطای $u(x)$	5.8×10^{-5}	7.3×10^{-6}	2.2×10^{-6}	8.4×10^{-7}	8.4×10^{-8}
خطای $v(x)$	1.38×10^{-5}	7.3×10^{-6}	8.3×10^{-6}	2.6×10^{-7}	3.4×10^{-8}

شکل ۱: مقایسه جواب دقیق و تقریبی مثال ۳ برای $\alpha = \beta = \frac{1}{2}$ و $N = 20$





مثال ۴. دستگاه معادلات انتگرال آبل غیرخطی

$$\begin{cases} u(x) - 2v(x) + \int_0^x \frac{u(t)v(t)}{(x-t)^{\frac{1}{5}}} dt = x^2 - 2x^3 + \frac{390625}{1573656}x^{\frac{27}{5}} + \frac{3125}{9576}x^{\frac{22}{5}}, \\ u(x) - v(x) - \int_0^x \frac{u(t)v(t)}{(x-t)^{\frac{1}{7}}} dt = x^3 - x^2 - \frac{2187}{5236}x^{\frac{17}{7}}. \end{cases}$$

برای $0 \leq x \leq 1$ و نقطه آغازین $[0, 0, \dots, 0]_{N+1}$ در نظر بگیرید. جواب‌های دقیق این دستگاه $u(x) = x^2$ و $v(x) = x^3$ هستند. بنابر روش پیشنهادی، دستگاه (۳۰۲) در این مثال به صورت

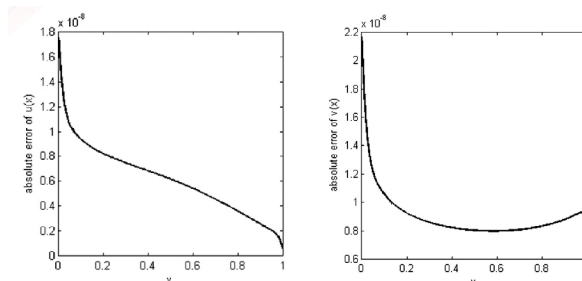
$$\begin{cases} \phi^T(x)U - 2\phi^T(x)V + U^T \tilde{U} \mathbf{B}^{(\frac{1}{5})} \phi(x) + V^T \tilde{V} \mathbf{B}^{(\frac{1}{5})} \phi(x) - g_1(x) \approx 0, \\ \phi^T(x)V - \phi^T(x)U - U^T \tilde{V} \mathbf{B}^{(\frac{1}{7})} \phi(x) - g_2(x) \approx 0. \end{cases}$$

می‌باشد. این مثال با $N = 20$ و استفاده از ریشه‌های $P_{\frac{1}{2}}^{(\alpha, \beta)}(x)$ به عنوان نقاط هم‌محلی، برای مقادیر مختلف α و β حل شده و در جدول ۲ مقدار ماکزیمم خطاهای مطلق گزارش شده است. نمودارهای شکل ۲، خطای مطلق $u(x)$ و $v(x)$ را برای $\alpha = \beta = 4$ و $N = 20$ روی بازه $[0, 1]$ نشان می‌دهد.

جدول ۲: ماکزیمم خطای مطلق مثال ۴ با $N = 20$

(α, β)	خطای $u(x)$	خطای $v(x)$	(α, β)	خطای $u(x)$	خطای $v(x)$
$(0, 0)$	1.9×10^{-9}	1.2×10^{-10}	$(\frac{1}{4}, \frac{1}{4})$	2.5×10^{-10}	1.9×10^{-10}
$(\frac{1}{4}, \frac{1}{4})$	2.4×10^{-10}	1.8×10^{-10}	$(1, 1)$	3.2×10^{-10}	2.5×10^{-10}
$(-\frac{1}{4}, -\frac{1}{4})$	1.9×10^{-9}	1.4×10^{-9}	$(-0.9, -0.9)$	2.6×10^{-9}	6.7×10^{-9}

شکل ۲: تابع خطای مطلق مثال ۴ برای $\alpha = \beta = 4$ و $N = 20$





۵ نتیجه

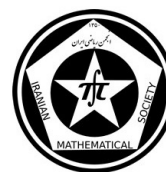
در این مقاله روشی برای حل دستگاه معادلات انتگرال غیرخطی آبل با استفاده از چند جمله ای ژاکوبی انتقال یافته که حالت کلی تر چند جمله ای های لژاندر و چیشف هستند، ارائه گردید. با توجه به شکل های ۱ و ۲ ملاحظه می شود که در بیستمین تکرار، میزان خطا روند کاهشی دارد. در مطالعات آینده می توان تعمیمی از روش برای مسائل آبل را به صورت کلی تر مورد توجه قرارداد.

مراجع

- [1] O. Knill , R. Dgani , M. Vogel , *A new approach to Abels integral operator and its application to stellar winds*, Astronom. Astrophys. 274 (3) (1993) 1002–1008.
- [2] A. Borhanifar , K.H. Sadri , *A new operational approach for numerical solution of generalized functional integro-differential equations*, Comput. Appl. Math. 279 (2015) 80–96.
- [3] K. Sadri, A. Amini, C. Cheng. *A new operational method to solve Abel's and generalized Abel's integral equations*. Appl. Math. Comput. 317 (2018) 49–67.
- [4] J. Biazar K. Sadri *Solution of weakly singular fractional integro-differential equations by using a new operational approach* J. Comput. Appl. Math. 352 (2019) 453–477.

پست الکترونیکی: dwltabadya@gmail.com

پست الکترونیکی: mahbobezarei96@gmail.com



حل عددی معادلات آب کم عمق تحت یک جریان برشی خطی و ساحل شیب دار

سیده نسیم مداح شریعتی^۱، محمدرضا یاقوتی^۲ و میراحمد لشته نشایی^۳

دانشجوی دکتری ریاضی کاربردی دانشگاه گیلان^۱

هیأت علمی دانشکده علوم ریاضی دانشگاه گیلان^۲

هیأت علمی دانشکده فنی دانشگاه گیلان^۳

چکیده

جریان‌ها در بستر دریا یا رودخانه برش ایجاد می‌کنند. در هر ناحیه‌ای که باد می‌وزد رانش سطحی آب وجود دارد. بسیاری از نتایج کلاسیک نظریه‌ی امواج آب سطحی با چشم‌پوشی از اثر چنین برشی به دست آمده است، اما روشن است که برش اثر بارزی بر ویژگی‌های امواج سطحی دارد. امواج منتشرشده در آب‌های کم عمق امواج بلند و یا امواج آب کم عمق نامیده می‌شوند. معادلات مربوط به این امواج تحت یک جریان برشی به سمت ساحل شیب‌دار به صورت عددی حل می‌شوند و سرعت افقی ذرات سیال و ارتفاع سطح آزاد در یک بازه‌ی زمانی معین به دست می‌آیند. حل دقیق معادلات با توجه به غیرخطی بودن آن‌ها آسان نیست. در این مقاله یک طرح تفاضلات متناهی معرفی می‌شود. تحت شرایط اولیه و مرزی معین این طرح روی دستگاه معادلات اجرا می‌شود و ارتفاع سطح آزاد در قالب نمودار نشان داده می‌شود. نتایج در مقایسه با روش‌های اویلر و آدامز-بشفرت، کارایی و پایداری طرح را نشان می‌دهند.

واژه‌های کلیدی: تفاضلات متناهی، روش اویلر، روش‌های عددی، معادلات آب کم عمق.

طبقه‌بندی موضوعی [۲۰۱۰]: 65H10, 65M06, 65N06.

۱ پیش‌گفتار

حرکت امواج در آب، یکی از رایج‌ترین پدیده‌هایی است که در طبیعت مشاهده می‌شود. بیشتر از ۱۵۰ سال، موضوع امواج آب موضوعی جالب برای بسیاری از پژوهشگران بوده است. خط ساحلی به عنوان مرز بین سطح خشکی و سطح توده‌ی آب نظیر اقیانوس، دریا یا دریاچه در نظر گرفته می‌شود. مناطق ساحلی، شامل خطوط ساحلی بوده که به اندازه‌ی کافی به طرف داخل خشکی و طرف دریا گسترده شده، جایی که فرآیندهای مهم دریایی در آن‌جا فعال می‌باشند. اکثر امواج آشکار و غالب دارای منشاء بادی هستند و باعث اکثر تغییرات قابل توجه در سواحل می‌باشند [۱]. این امواج جزء مهمترین امواج گرانشی* سطحی محسوب می‌شوند. بسیاری از قوانین حاکم بر پدیده‌ها در

*Gravity Waves



جهان هستی، عبارات یا روابطی، شامل تغییرات در پدیده‌هایی است که اتفاق می‌افتند که به زبان ریاضی روابط همان معادلات و تغییرات، مشتقات می‌باشند. بنابراین برای فهم مسائلی از قبیل حرکت سیالات، جریان در مدارهای الکتریکی، اتلاف گرما در اجسام جامد، انتشار و ردیابی امواج زمین‌لرزه‌ای یا افزایش یا کاهش جمعیت در میان بسیاری از مسائل دیگر، دانش معادلات دیفرانسیل ضروری است. یک معادله‌ی دیفرانسیل که یک فرآیند فیزیکی را توصیف می‌کند مدل ریاضی فرآیند نامیده می‌شود. مدل‌های ریاضی در طیف وسیعی از رشته‌ها از علوم و فن‌آوری تا جامعه‌شناسی و تجارت تشریح می‌کنند که چگونه کمیت‌ها تغییر می‌کنند. معادلات دیفرانسیل جزئی غیرخطی مرتبه‌ی اول، امواج غیرخطی را مدل‌بندی می‌کنند و در دینامیک گازها، امواج آب، واکنش‌های شیمیایی، انتقال آلاینده‌ها و سیستم‌های زیست‌محیطی رخ می‌دهند. بسیاری از معادلات دیفرانسیل جزئی مرتبط با فیزیک بر اساس قانون‌های بقاء[†] استنتاج می‌شوند. به عنوان مثال $u(x, t)$ به عنوان غلظت یا تابع چگالی بعضی از مواد یا محصولات شیمیایی رقیق در یک مایع در نظر گرفته می‌شود. معادلات اصلی، در شکلی مشابه در کاربردهای دیگر نیز به دست می‌آیند. هرچند در بعضی از موارد می‌توان جواب‌های دقیق معادلات دیفرانسیل جزئی را به دست آورد اما در حالت کلی این امر ممکن نیست. در این موارد نیازمند روش‌هایی برای به دست آوردن جواب‌های تقریبی برای معادله هستیم. به طور کلی روش‌های عددی بهترین ابزار محاسبات برای تعیین خصوصیات امواج ناشی از باد محسوب می‌شوند. در سال‌های اخیر پیشرفت‌های بسیاری در زمینه‌ی مدل‌سازی عددی هیدرودینامیکی رخ داده است. یک مدل دینامیکی آب، معادلات آب کم‌عمق[‡] است. معادلات آب کم‌عمق، تقریباً کاملاً شناخته‌شده‌ای می‌باشند که زمینه‌ساز شبیه‌سازی عددی مسائل مختلف در اقتصاد، جریان‌های آب در رودخانه‌ها، مخازن و مناطق ساحلی در دریاها و اقیانوس‌ها، حرکت‌های یخچال‌های طبیعی، سونامی و مسائل گردش‌های جوی است. در مقایسه با معادلات آب کم‌عمق سه‌بعدی، معادلات یک‌بعدی هزینه‌ی کمتری در شبیه‌سازی‌های وابسته به زمان و محاسبات کمتر پیچیده‌ای دارند [۱۰]. به دلیل کاربرد آن‌ها در زندگی واقعی، نیاز به حل تقریبی مناسب از معادلات آب کم‌عمق، احساس می‌شود زیرا به دست آوردن جواب تحلیلی این معادلات بسیار سخت است. بنابراین می‌توان یک روش غیرتحلیلی را برای حل آن‌ها به کار برد. در هیدرولیک، مهندسی ساحل، اقیانوس و محیط اغلب با موقعیت‌هایی روبرو می‌شویم که مقیاس طولی مسأله نسبت به مقیاس عمودی نسبتاً زیاد است. برای به دست آوردن معادلات آب کم‌عمق در حالت بدون بُعد نیز سیال را در یک کانال با عرض واحد در نظر می‌گیریم و فرض می‌کنیم سرعت عمودی سیال قابل چشم‌پوشی است و سرعت افقی $u(x, t)$ که واحد آن متر بر ثانیه است، در کل سطح مقطع کانال تقریباً ثابت است. این فرض، زمانی که امواج با دامنه‌ی کوچک[§] در سیالی که نسبت به طول موج کم‌عمق است نیز درست است [۷]. بسیاری از نتایج نظریه‌ی امواج آب سطحی با صرف‌نظر از اثر برش به دست آمده است. تنش‌های برشی روی سطح عمل می‌کنند و همیشه در یک سیال جاری واقعی وجود دارند. اندرکنش امواج و جریان‌ها و همین‌طور جریان‌های کرانه‌ای در منطقه‌ی شکست امواج که از وزش امواج به صورت مایل به سواحل تولید می‌شوند از جمله عوامل تأثیرگذار بر پدیده‌های خیزآب (بالارفتگی متوسط سطح آب) و فروآب (پایین رفتگی متوسط سطح آب) می‌باشند [۱]. در این مقاله حالتی مطالعه می‌شود که جریان برشی[¶] زمینه با ساحل شیب‌دار اثر متقابل دارد. فرض بر این است که کف دریا به صورت $h(x) = \alpha x$ باشد و یک جریان برشی زمینه $U(x, z)$ اعمال شود. چنین وضعیتی ممکن است زمانی به وجود آید که باد ساحلی (که از اقیانوس به سوی زمین می‌وزد) یک جریان سطحی به سمت ساحل ایجاد کند [۹]. بسته به شدت نیروی باد و

[†] Conservation Laws

[‡]Shallow Water Equations

[§]Small Amplitude

[¶]Shear Flow



هندسه ی ساحل، جریان بازگشتی حاصل ممکن است به شکل جریان زیر آب باشد و بقای جرم ایجاب می کند که $\int_{-h}^{\circ} U(x, z) dz$ برابر با صفر باشد. این وضعیت ممکن است توسط جریان برشی خطی اعمال شده ی $U(x, z) = \Gamma_1(\frac{h}{\gamma} + z)$ توصیف شود که دارای ویژگی $\int_{-h}^{\circ} U(x, z) dz = 0$ می باشد. z فاصله ی نقطه ی مورد نظر از سطح آزاد در جهت محور عرضی بر حسب متر، h فاصله ی بین سطح آزاد و کف شیب دار بر حسب متر و Γ_1 چرخش را نشان می دهد و واحد آن $\frac{1}{s}$ است. از نظر دانشمندان به ویژه ریاضی دان ها و فیزیک دان ها و مهندسين علوم دریایی حل دقیق معادلات آب کم عمق در این حالت از اهمیت ویژه ای برخوردار است، هرچند در حالت کلی این مسأله با پیچیدگی فراوان است و به ندرت یک فرمول تحلیلی برای جواب پیدا می شود. با این حال از سال ۱۹۵۸ با کار Carrier و Greenspan [۴] امکان حل دقیق معادلات آب کم عمق در حالت عمق غیر ثابت تحت فرضیات خاص فراهم شد. در سال ۲۰۰۳ داده های اولیه ی کلی تر در کار Carrier و همکاران [۵] بررسی شد و ویژگی های فیزیکی نظیر جرم و جریان های حرکت مربوط به پیشروی آب در ساحل محتمل با سونامی صرف نظر شد. در سال ۲۰۰۹ برخی از کلیات درباره ی شکل نیم رخ ساحل در کار Didenkulova و همکاران [۶] بیان شد که یک توپوگرافی بستر محدب $h(x) = x^{\frac{2}{3}}$ از نوع [‡] بررسی شد. در سال ۲۰۱۰ در کار Antuono و Brocchini [۲] از تکنیک های تحلیلی برای برآورد تابع ژاکوبین ^{**} مربوط به بستر دلخواه استفاده شد و نشان داده شد که حداقل در نظریه، محدودیت ساحل هموار ^{††} یا محدب لازم نیست. برخی از مطالعات اخیر اهمیت جریان برشی نزدیک ساحل را از دیدگاه های مختلف نشان داده اند. به ویژه در [۹] بحث شد که چنین جریانی می تواند به وسیله ی باد روی ساحل ایجاد شود و اثر قابل توجهی بر میانگین سطح آب ^{‡‡} در ساحل دارد و در نتیجه ممکن است در شرایط مختلف موجب افزایش پیشروی موج در ساحل شود. در سال ۲۰۱۷ و Kalisch و Bjørnstad [۳] با استفاده از ایده ی تبدیل نقش های متغیرهای مستقل و وابسته و کاربرد تبدیل هودوگراف، معادلات آب کم عمق را در حالت همگن و با چشم پوشی از جملات سمت راست معادله در حضور یک جریان برشی زمینه و در نظر گرفتن یک ساحل شیب دار به طور دقیق حل کردند. در حالت کلی و غیرهمگن و در نظر گرفتن جملات سمت راست معادله این یک مسأله ی مشکل است و به دست آوردن یک جواب تحلیلی به راحتی امکان پذیر نیست. در این حالت روش های عددی می توانند مفید باشند. در این مقاله روشی عددی برای حل معادلات آب کم عمق در [۳] در حضور یک جریان برشی زمینه و ساحل شیب دار در حالت غیرهمگن به کار برده می شود. ابتدا مدل ریاضی معادلات آب کم عمق معرفی می شود و سپس یک طرح عددی و روش های اویلر و آدامز-بشفرت برای حل دستگاه معادلات آب کم عمق معرفی شده در [۳] تشریح می شود. سپس طرح عددی و روش های اویلر و آدامز-بشفرت بر روی معادلات پیاده سازی و نتایج نشان داده می شوند. سرانجام با بحث روی نتایج و یافته ها، مزیت های طرح عددی معرفی شده بیان می شود. یکی از مهمترین رده بندی های سطح امواج، بر مبنای عمق نسبی $\frac{d}{L}$ می باشد که d ژرفای آب و L طول موج هر دو بر حسب متر هستند. عمق نسبی وقتی موج به ساحل نزدیک شود، کاهش خواهد داشت [۱]. نسبت $\frac{d}{L} = 0.5$ را می توان به عنوان مرزی مناسب برای جدا کردن امواج آب عمیق از امواج آب کم عمق در نظر

[‡]Convex Bed

^{**}Jacobian Function

^{††}Planar Beach

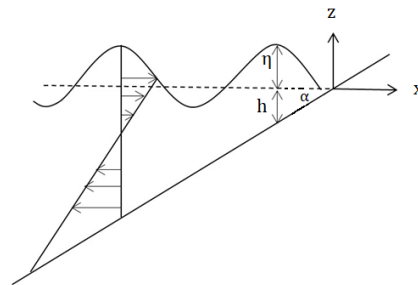
^{‡‡}Mean-Water Level

Hodograph Transform

Relative Depth

Water Depth

Wave Height



شکل ۱: امواج بلند پیش‌رونده تحت جریان بازگشتی زیر آب و ساحل شیب‌دار با معادله‌ی $h(x) = \alpha x$ [۳]

گرفت. امواج با طول کافی که معادله‌ی $c = \sqrt{gd}$ در مورد آن‌ها قابل کاربرد باشد، امواج بلند نامیده می‌شوند. در این رابطه c سرعت موج بر حسب متر بر ثانیه و g شتاب گرانش بر حسب متر بر مجذور ثانیه است. رابطه‌ی مذکور به لاگرانژ نسبت داده می‌شود. برای امواج بلند، زاویه‌های کوچک α و دامنه‌های نسبتاً کوچک، یک مجموعه از معادلات آب کم‌عمق (شکل ۱) از اصول اولیه به صورت زیر به دست می‌آید [۳]

$$\eta_t + \left(\frac{\Gamma_1}{\gamma} \eta(\eta - \alpha x) + u(\eta - \alpha x) \right)_x = 0 \quad (1'1)$$

$$u_t + \left(\frac{\alpha \Gamma_1}{\gamma} x \left(u + \frac{\alpha \Gamma_1}{\gamma} x \right) + \frac{u^2}{\gamma} + g\eta \right)_x = 0. \quad (1'2)$$

به طور کلی موج پیش‌رونده (رونده) را می‌توان با متغیرهای x (فضایی) و t (زمانی) یا به وسیله‌ی ترکیب آن‌ها یعنی فاز بیان کرد [۱]. $\eta(x, t)$ انحراف سطح آزاد و $u(x, t)$ سرعت افقی ذرات سیال را نشان می‌دهد. ضریب $\alpha \Gamma_1$ شدت تعامل بین کف شیب‌دار و جریان برشی را نشان می‌دهد [۳].

ایده‌ی اساسی در مقاله‌ی مذکور این است که از تفاضلات متناهی برای تقریب مشتقات در معادله‌ی دیفرانسیل جزئی استفاده کنیم. اما این روش در دامنه‌های نامنظم کاملاً پیچیده می‌شود. همین‌طور تحلیل‌های ریاضی از قبیل پایداری و همگرایی برای معادلات دیفرانسیل جزئی غیرخطی و یا با ضرایب متغیر آسان نیست [۸]. در هندسه‌های نامنظم می‌توان با استفاده از نگاشت‌های خطی هندسه‌ی فیزیکی نامنظم را به هندسه‌ی محاسباتی منظم تبدیل نمود. در این مقاله یک طرح تفاضل متناهی بررسی می‌شود و جواب‌های تقریبی مسأله در مجموعه‌ی گسسته‌ای از نقاط شبکه‌ی x_i در زمان‌های مختلف t به دست می‌آیند.



۲ طرح‌های عددی

اکنون به بیان طرح‌های عددی می‌پردازیم. در تمام این بخش طرح‌های مطرح شده در نقاط شبکه‌ای (x_i, t^n) نوشته می‌شوند. با مشتق‌گیری از معادلات (۱۱) و (۱۲) داریم

$$\eta_t + \frac{\Gamma_1}{\gamma} (\eta_x(\eta - \alpha x) + \eta(\eta_x - \alpha)) + u_x(\eta - \alpha x) + u(\eta_x - \alpha) = 0 \quad (2.1)$$

$$u_t + \frac{\alpha \Gamma_1}{\gamma} \left(u + \frac{\alpha \Gamma_1}{\gamma} x + x \left(u_x + \frac{\alpha \Gamma_1}{\gamma} \right) \right) + uu_x + g\eta_x = 0. \quad (2.2)$$

در طرح‌های زیر، شرایط اولیه مقادیر u و η در لحظه‌ی آغازین معلوم است. در تمامی گام‌های زمانی، مقادیر u و η در نقاط انتهایی دامنه با استفاده از شرایط مرزی معلوم است و برای $i = 2, \dots, nx - 1$ که nx تعداد مؤلفه‌های گسسته‌سازی و $1 - nx$ تعداد زیربازه‌ها است، محاسبه می‌شوند. برای تمامی طرح‌ها، شرایط مرزی زیر در نظر گرفته می‌شود.

$$u_1^{n+1} = 0, \quad u_{nx}^{n+1} = 0, \quad \eta_1^{n+1} = \eta_2^{n+1}, \quad \eta_{nx}^{n+1} = \eta_{nx-1}^n.$$

طرح عددی براساس Upwind. از معادله‌ی (۱۱) داریم

$$\eta_t + \left(\frac{\Gamma_1}{\gamma} \eta H + uH \right)_x = 0,$$

که $H(x, t) = \eta(x, t) - h(x)$ و $h(x) = \alpha x$ به طوری که $h(x)$ کف دریا است. در این طرح برای η_t عملگر تفاضل پیشرو و برای $\left(\frac{\Gamma_1}{\gamma} \eta H + uH \right)_x$ طرح upwind به شرح زیر به کار می‌رود.

$$\frac{\eta_i^{n+1} - \eta_i^n}{\Delta t} + \frac{\Gamma_1}{\gamma} \frac{\eta H_p - \eta H_m}{\Delta x} + \frac{u H_p - u H_m}{\Delta x} = 0,$$

و از آنجا

$$\eta_i^{n+1} = \eta_i^n - r \frac{\Gamma_1}{\gamma} (\eta H_p - \eta H_m) - r(u H_p - u H_m),$$

که $r = \frac{\Delta t}{\Delta x}$ و

$$\eta H_p = H_p \eta_i + H_m \eta_{i+1}, \quad u H_p = H_p u_i + H_m u_{i+1}, \quad H_p = \frac{H_i + |H_i|}{\gamma}, \quad H_m = \frac{H_i - |H_i|}{\gamma}$$

$$\eta H_m = H_p \eta_{i-1} + H_m \eta_i, \quad u H_m = H_p u_{i-1} + H_m u_i, \quad H_p = \frac{H_{i-1} + |H_{i-1}|}{\gamma}, \quad H_m = \frac{H_{i-1} - |H_{i-1}|}{\gamma}.$$

اکنون برای u_t و $\left(x \left(u + \frac{\alpha \Gamma_1}{\gamma} x \right) \right)_x$ و $(g\eta)_x$ عملگر تفاضل پیشرو در رابطه‌ی (۱۲) و برای $\left(\frac{u}{\gamma} \right)_x$ طرح upwind به شرح زیر به کار می‌رود. بنابراین خواهیم داشت

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{\alpha \Gamma_1}{\gamma} \frac{\left(x_{i+1} \left(u_{i+1}^n + \frac{\alpha \Gamma_1}{\gamma} x_{i+1} \right) - x_i \left(u_i^n + \frac{\alpha \Gamma_1}{\gamma} x_i \right) \right)}{\Delta x} + \frac{u u_p - u u_m}{\gamma \Delta x} + g \frac{\eta_{i+1}^n - \eta_i^n}{\Delta x} = 0,$$



و از آنجا

$$u_i^{n+1} = u_i^n - r \frac{\alpha \Gamma_{\lambda}}{\gamma} \left(x_{i+1} \left(u_{i+1}^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_{i+1} \right) - x_i \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) \right) - \frac{r}{\gamma} (u u_p - u u_m) - r g (\eta_{i+1}^n - \eta_i^n),$$

که $r = \frac{\Delta t}{\Delta x}$ و

$$u u_p = u_p u_i + u_m u_{i+1}, \quad u_p = \frac{u_i + |u_i|}{\gamma}, \quad u_m = \frac{u_i - |u_i|}{\gamma}$$

$$u u_m = u_p u_{i-1} + u_m u_i, \quad u_p = \frac{u_{i-1} + |u_{i-1}|}{\gamma}, \quad u_m = \frac{u_{i-1} - |u_{i-1}|}{\gamma}.$$

در نهایت دستگاهی از معادلات به شکل زیر خواهیم داشت

$$\eta_i^{n+1} = \eta_i^n - r \frac{\Gamma_{\lambda}}{\gamma} (\eta H_p - \eta H_m) - r (u H_p - u H_m)$$

$$u_i^{n+1} = u_i^n - r \frac{\alpha \Gamma_{\lambda}}{\gamma} \left(x_{i+1} \left(u_{i+1}^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_{i+1} \right) - x_i \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) \right) - \frac{r}{\gamma} (u u_p - u u_m) - r g (\eta_{i+1}^n - \eta_i^n).$$

روش اولر. از معادلات (۲۰۱) و (۲۰۲) داریم

$$\eta_t + \eta_x \left(\Gamma_{\lambda} \eta - \frac{\alpha \Gamma_{\lambda}}{\gamma} x + u \right) - \frac{\alpha \Gamma_{\lambda}}{\gamma} \eta + u_x (\eta - \alpha x) - \alpha u = 0.$$

$$u_t + \frac{\alpha \Gamma_{\lambda}}{\gamma} \left(u + \frac{\alpha \Gamma_{\lambda}}{\gamma} x \right) + u_x \left(u + \frac{\alpha \Gamma_{\lambda}}{\gamma} x \right) + g \eta_x = 0.$$

با به کارگیری عملگر تفاضل پیشرو برای η_t و u_t و عملگر تفاضل مرکزی برای η_x و u_x با طول گام های به ترتیب Δt و Δx داریم

$$\frac{\eta_i^{n+1} - \eta_i^n}{\Delta t} + \frac{\eta_{i+\frac{1}{\gamma}}^n - \eta_{i-\frac{1}{\gamma}}^n}{\Delta x} \left(\Gamma_{\lambda} \eta_i^n - \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i + u_i^n \right) - \frac{\alpha \Gamma_{\lambda}}{\gamma} \eta_i^n + \frac{u_{i+\frac{1}{\gamma}}^n - u_{i-\frac{1}{\gamma}}^n}{\Delta x} (\eta_i^n - \alpha x_i) - \alpha u_i^n = 0.$$

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + \frac{\alpha \Gamma_{\lambda}}{\gamma} \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) + \frac{u_{i+\frac{1}{\gamma}}^n - u_{i-\frac{1}{\gamma}}^n}{\Delta x} \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) + g \frac{\eta_{i+\frac{1}{\gamma}}^n - \eta_{i-\frac{1}{\gamma}}^n}{\Delta x} = 0.$$

پس از محاسباتی کوتاه خواهیم داشت

$$\eta_i^{n+1} - \eta_i^n + r \left(\eta_{i+\frac{1}{\gamma}}^n - \eta_{i-\frac{1}{\gamma}}^n \right) \left(\Gamma_{\lambda} \eta_i^n - \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i + u_i^n \right) - \Delta t \frac{\alpha \Gamma_{\lambda}}{\gamma} \eta_i^n \quad (203)$$

$$+ r \left(u_{i+\frac{1}{\gamma}}^n - u_{i-\frac{1}{\gamma}}^n \right) (\eta_i^n - \alpha x_i) - \alpha \Delta t u_i^n = 0. \quad (204)$$

$$u_i^{n+1} - u_i^n + r \Delta x \frac{\alpha \Gamma_{\lambda}}{\gamma} \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) + r \left(u_{i+\frac{1}{\gamma}}^n - u_{i-\frac{1}{\gamma}}^n \right) \left(u_i^n + \frac{\alpha \Gamma_{\lambda}}{\gamma} x_i \right) + r g \left(\eta_{i+\frac{1}{\gamma}}^n - \eta_{i-\frac{1}{\gamma}}^n \right) = 0,$$

که مقادیر متغیرها در نقاط فضایی میان شبکه با میانگین حسابی مقادیر در نقاط همسایه محاسبه می شود. به عبارت دیگر

$$\eta_{i+\frac{1}{\gamma}} = \frac{\eta_i + \eta_{i+1}}{\gamma}, \quad \eta_{i-\frac{1}{\gamma}} = \frac{\eta_i + \eta_{i-1}}{\gamma},$$

و به طور مشابه برای u . سرانجام خواهیم داشت

$$\eta_i^{n+1} = \eta_i^n - \frac{r}{\gamma} (\eta_{i+1}^n - \eta_{i-1}^n) \left(\Gamma_1 \eta_i^n - \frac{\alpha \Gamma_1}{\gamma} x_i + u_i^n \right) + \alpha \Delta t \left(\frac{\Gamma_1}{\gamma} \eta_i^n + u_i^n \right) \quad (2.5)$$

$$- \frac{r}{\gamma} (u_{i+1}^n - u_{i-1}^n) (\eta_i^n - \alpha x_i) \quad (2.6)$$

$$u_i^{n+1} = u_i^n - r \Delta x \frac{\alpha \Gamma_1}{\gamma} \left(u_i^n + \frac{\alpha \Gamma_1}{\gamma} x_i \right) - \frac{r}{\gamma} (u_{i+1}^n - u_{i-1}^n) \left(u_i^n + \frac{\alpha \Gamma_1}{\gamma} x_i \right) - \frac{r g}{\gamma} (\eta_{i+1}^n - \eta_{i-1}^n).$$

روش آدامز-بشفرث. از معادلات (۱۱) و (۱۲) داریم

$$u_t = - \left(\frac{\alpha \Gamma_1}{\gamma} x \left(u + \frac{\alpha \Gamma_1}{\gamma} x \right) + \frac{u^2}{\gamma} + g \eta \right)_x$$

$$\eta_t = - \left(\frac{\Gamma_1}{\gamma} \eta (\eta - \alpha x) + u (\eta - \alpha x) \right)_x.$$

با استفاده از طرح دو گامی آدامز-بشفرث و به کارگیری عملگر تفاضل مرکزی برای مشتقات نسبت به x در معادلات فوق خواهیم داشت

$$\eta_i^{n+2} - \eta_i^{n+1} = - \frac{1}{\gamma} \frac{\Delta t}{2 \Delta x} (f_{i+1} - f_{i-1})$$

$$u_i^{n+2} - u_i^{n+1} = - \frac{1}{\gamma} \frac{\Delta t}{2 \Delta x} (g_{i+1} - g_{i-1}),$$

که در آن

$$f_i = 3 \left(\frac{\Gamma_1}{\gamma} \eta_i^{n+1} (\eta_i^{n+1} - \alpha x_i) + u_i^{n+1} (\eta_i^{n+1} - \alpha x_i) \right) - \left(\frac{\Gamma_1}{\gamma} \eta_i^n (\eta_i^n - \alpha x_i) + u_i^n (\eta_i^n - \alpha x_i) \right)$$

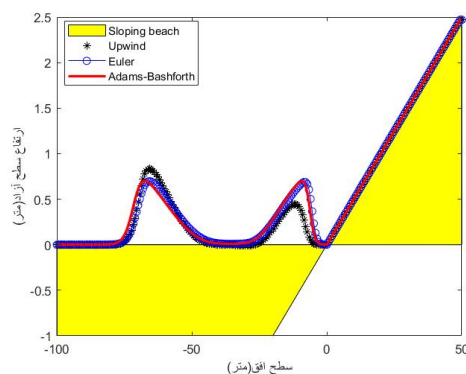
$$g_i = 3 \left(\frac{\alpha \Gamma_1}{\gamma} x_i (u_i^{n+1} + \frac{\alpha \Gamma_1}{\gamma} x_i) + \frac{(u_i^{n+1})^2}{\gamma} + g \eta_i^{n+1} \right) - \left(\frac{\alpha \Gamma_1}{\gamma} x_i (u_i^n + \frac{\alpha \Gamma_1}{\gamma} x_i) + \frac{(u_i^n)^2}{\gamma} + g \eta_i^n \right).$$

برای استفاده از این طرح دو گامی ابتدا با به کارگیری یک طرح عددی با استفاده از شرایط اولیه (گام اول)، گام دوم را به دست می آوریم و سپس با طرح دو گامی ادامه می دهیم.

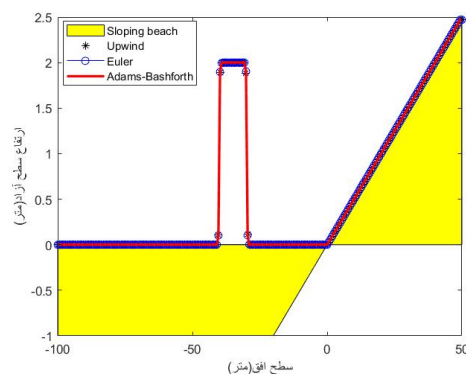
۳ یافته ها

در این بخش نتایج عددی اعمال شده بر روی معادلات بیان می شوند. برنامه های مربوط به طرح عددی معرفی شده و روش اویلر و روش آدامز-بشفرث در $MATLAB R2016a$ تحت سیستم عامل $Windows 10$ نوشته شدند. تمامی محاسبات با یک سیستم $Intel CORE i7 CPU$ انجام شد. برای هر سه طرح از داده های یکسان $x = [-100, 50]$ ، $\alpha = 0.5$ ، $m x = 250$ ، $\Gamma_1 = 0.1$ و $\Delta t = 0.1$ استفاده شده است که البته داده های فوق قابل تغییر می باشند و در این قسمت ما به ذکر نتایج عددی حاصل در یک حالت بسنده می کنیم. با توجه به این که شیب سواحل ماسه ای معمولاً در محدوده ی مشخصی تغییر می نمایند مدل های حاضر برای شیب های نسبتاً ملایم کارایی داشته و در شیب های تند که ساحل معمولاً از نوع صخره ای می باشد نتایج مدل ها چندان دقیق نخواهد بود.

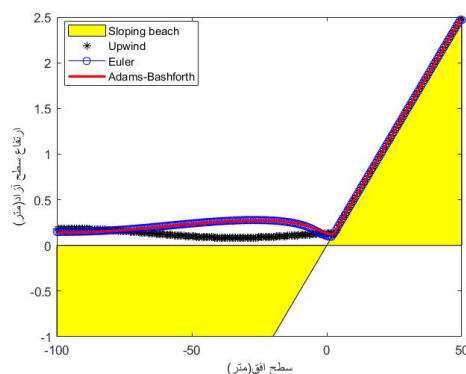
لازم به ذکر است که مدل‌های استفاده شده در این مقاله برای مشخصات امواج و اعماق مختلف آب کارا می‌باشد و از نتایج آن می‌توان به صورت کاربردی در سواحل ماسه‌ای استفاده نمود. نمودارهای مربوط در ۵ زمان مختلف رسم شده‌اند. در شکل‌های ۲ و ۳ و جدول ۱ همان‌طور که ملاحظه می‌شود، طرح upwind در زمان‌های یکسان، با دو طرح دیگر مقایسه شده است.



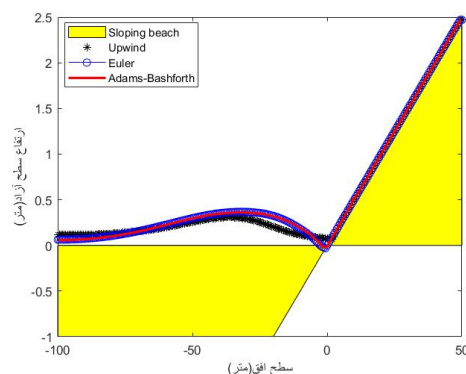
(ب) ارتفاع سطح آزاد (بر حسب متر) در ۵ ثانیه



(آ) ارتفاع سطح آزاد (بر حسب متر) در ۰/۱ ثانیه



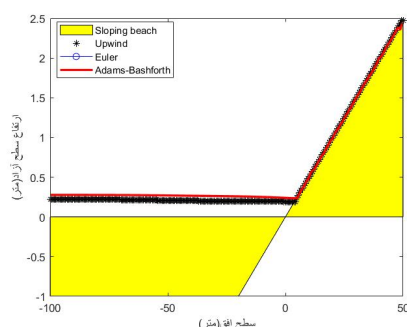
(د) ارتفاع سطح آزاد (بر حسب متر) در ۱۰۰ ثانیه



(ج) ارتفاع سطح آزاد (بر حسب متر) در ۵۰ ثانیه

شکل ۲: ارتفاع سطح آزاد (بر حسب متر) بر اساس دو روش اویلر و آدامز-بشفرت و طرح عددی در چند زمان مختلف بر حسب ثانیه

در هر زمان، مشاهده شد که طرح upwind از دو روش دیگر سریع‌تر به حالت پایداری میل می‌کند. علاوه بر سرعت میل به حالت پایداری، در طرح upwind این موضوع نیز چشم‌گیر است که طرح مذکور، در کل بازه زمانی در نظر گرفته شده، دارای نوسانات کمتری نسبت به دو روش دیگر است. قابل ذکر است که روش اویلر برای مساله فوق، بعد از گذشت زمان تقریباً ۲۳۰ ثانیه به دلیل خطای انباشته‌گی در



شکل ۳: ارتفاع سطح آزاد (بر حسب متر) بر اساس دو روش اویلر و آدامز-بشفرت و طرح عددی در ۱۰۰۰ ثانیه

جدول ۱: مقایسه‌ی دو روش اویلر و آدامز-بشفرت و طرح عددی

$\eta(x, t)$	روش اویلر	روش آدامز-بشفرت	طرح عددی	$\eta(x, t)$	روش اویلر	روش آدامز-بشفرت	طرح عددی
$\eta(-100, 0.01)$	۰	۰	۰	$\eta(-100, 5)$	$-1.516e(-0.5)$	$-1.575e(-0.5)$	$-1.758e(-0.5)$
$\eta(-50, 0.01)$	۰	۰	۰	$\eta(-50, 5)$	۰.۱۵۴۴	۰.۱۲۶۸	۰.۱۱۸۴
$\eta(0, 0.01)$	$-6.125e(-0.7)$	۰	$2.453e(-0.6)$	$\eta(0, 5)$	-0.0003074	$3.919e(-0.5)$	0.0004498
$\eta(50, 0.01)$	۲.۴۶۹۸۸	۲.۴۶۹۸۸	۲.۴۶۹۸۸	$\eta(50, 5)$	۲.۴۶۹۸۹	۲.۴۷۱۳۸	۲.۴۶۹۸۸
$\eta(-100, 50)$	۰.۰۶۳۴۷	۰.۰۵۸۲۶	۰.۱۱۲۲	$\eta(-100, 100)$	۰.۱۴۷۸۷	۰.۱۴۰۵۶	۰.۱۸۴۳
$\eta(-50, 50)$	۰.۲۹۸۴	۰.۲۹۵۵	۰.۲۵۷۶	$\eta(-50, 100)$	۰.۲۴۰۴	۰.۲۳۳۶	۰.۰۹۸۲۸
$\eta(0, 50)$	-0.0001209	-0.01723	۰.۰۷۷۷۲	$\eta(0, 100)$	۰.۱۰۰۳	۰.۱۳۱۵	۰.۱۳۰۶
$\eta(50, 50)$	۲.۴۶۹۸۹	۲.۴۶۶۵۴	۲.۴۶۹۸۸	$\eta(50, 100)$	۲.۴۶۹۸۹	۲.۴۶۴۹۸	۲.۴۶۹۸۸

محاسبات عددی، به حالت ناپایدار رسید که با کاهش طول گام مکانی و زمانی، نتیجه‌ای مثبت در روند اویلر حاصل نگردید در حالی که طرح عددی مطرح شده و روش آدامز-بشفرت، با همان طول گام‌های مکانی و زمانی، بعد از گذشت ۱۰۰۰ ثانیه، همچنان پایدار ماند که این مزیت طرح عددی ارائه شده را نشان می‌دهد با این تفاوت که تمایل به حالت پایدار در طرح upwind سریع‌تر از روش آدامز-بشفرت بوده است.

۴ نتیجه‌گیری

در این مقاله مدل آب کم عمق تحت جریان‌های برشی خطی روی یک کف شیب‌دار به کار برده شد. این معادلات قبلاً با تبدیل هودوگراف به طور دقیق در حالت همگن حل شدند اما برای حالت ناهمگن بررسی نشدند که در این مقاله، در حالت ناهمگن نیز موضوع مربوطه از لحاظ عددی مورد ارزیابی قرار گرفت. نتایج عددی طرح معرفی شده بر اساس روش تفاضلات متناهی روی معادلات مذکور در مقایسه با نتایج عددی روش اویلر و روش آدامز-بشفرت قابل قبول می‌باشند. لازم به ذکر است که با توجه به محدودیت‌های موجود در اندازه‌گیری



نوسانات سطح آب در سواحل ماسه‌ای به طور دقیق، امکان مقایسه‌ی نتایج حاصله از مقاله‌ی حاضر با داده‌های تجربی در این حالت وجود نداشته همچنین شیب ساحل در نظر گرفته شده متناظر با سواحل ماسه‌ای موجود در کشور نبوده است. در مقاله حاضر از هندسه‌ی موجود در مرجع [۳] استفاده شده که چنین شیبی در سواحل ماسه‌ای کشور وجود ندارد. در صورت وجود داده‌های اندازه‌گیری شده در سواحل کشور می‌توان نتایج تحقیق حاضر را به درستی اندازه‌گیری نمود و دقت مدل‌ها را افزایش داد. بنابراین از این طرح تفاضلاتی می‌توان برای پیش‌بینی امواج در معادلاتی شامل جملات غیرخطی مشابه معادلات مطرح شده در این مقاله، استفاده نمود.

مراجع

- [۱] خ. برگی، اصول مهندسی دریا، انتشارات دانشگاه تهران، ۱۳۹۶.
- [2] M. Antuono, M. Brocchini, *Analysis of the nonlinear shallow water equations over nonplanar topography*, Studies in Applied Mathematics, **124** (2010), 85–103.
- [3] M. Bjørnstad, H. Kalisch, *Shallow water dynamics on linear shear flows and plane beaches*, Physics of Fluids, **29** (2017), 073602.
- [4] G. F. Carrier, H. P. Greenspan, *Water waves of finite amplitude on a sloping beach*, Journal of Fluid Mechanics, **4** (1958), 97–109.
- [5] G. F. Carrier, T. T. Wu, H. Yeh, *Tsunami run-up and draw-down on a plane beach*, Journal of Fluid Mechanics, **475** (2003), 79–99.
- [6] I. Didenkulova, E. Pelinovsky, T. Soomere, *Long surface wave dynamics along a convex bottom*, Journal of Geophysical Research: Oceans, **114** (2009), 394–399.
- [7] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations*, Siam, 2007.
- [8] J. Li, Y. T. Chen, *Computational Partial Differential Equations Using MATLAB*, Taylor and Francis Group, 2008.
- [9] T. L. Walton, R. G. Dean, *Landward limit of wind setup on beaches*, Ocean Engineering, **36** (2009), 763–766.

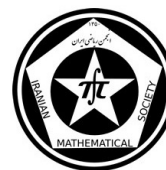


- [10] P. Welahettige, K. Vaagsaether, B. Lie, *A solution method for one-dimensional shallow water equations using flux limiter centered scheme for open Venturi channels*, Journal of Computational Multiphase Flows, **0** (2018), 1–11.

پست الکترونیکی: nasim.maddah62@gmail.com

پست الکترونیکی: yaghouti@guilan.ac.ir

پست الکترونیکی: maln@guilan.ac.ir



حل عددی معادلات دیفرانسیل کسری مرتبه متغیر با استفاده از تقریب ریتز و روش هم محلی

سمیه شیخی[†]، ماشاله متین فر[‡] و محمد عرب فیروزجایی*

[†] دانشگاه مازندران، دانشکده علوم ریاضی، بابلسر، ایران

[‡] دانشگاه مازندران، دانشکده علوم ریاضی، بابلسر، ایران

* دانشگاه علم و صنعت مازندران، دانشکده علوم ریاضی، بهشهر، ایران

چکیده

در این مقاله سعی بر آن است که با استفاده از تقریب ریتز و روش هم محلی به حل دسته‌ای از معادلات دیفرانسیل کسری مرتبه متغیر بپردازیم. ابتدا تابع مجهول به کمک تقریب ریتز و چند جمله‌ایهای لژاندر انتقالی به عنوان توابع پایه، تقریب زده می‌شود. سپس با استفاده از روش هم محلی و بکارگیری ریشه‌های چند جمله‌ایهای لژاندر به عنوان نقاط هم محلی، به دستگاہی از معادلات جبری می‌رسیم، که با نرم‌افزار Mathematica 10 و روش‌های معمول به حل این دستگاہ می‌پردازیم. در انتها با ارائه یک مثال عددی دقت و کارایی روش مدنظر نشان داده می‌شود.

واژه‌های کلیدی: تابع برآورده، تقریب ریتز، چند جمله‌ای لژاندر انتقالی، مشتق کاپوتو مرتبه‌ی متغیر.

طبقه‌بندی موضوعی [۲۰۱۰]: 34A08, 65L60.

۱ پیش‌گفتار

معادلات دیفرانسیل مرتبه‌ی کسری که تعمیمی از معادلات دیفرانسیل مرتبه‌ی صحیح می‌باشد، در دهه‌های اخیر مورد توجه دانشمندان و مهندسان قرار گرفته است. علت علاقه به مطالعه معادلات دیفرانسیل مرتبه کسری در این واقعیت نهفته است که مدل‌های مرتبه کسری از مدل‌های مرتبه صحیح دقیق‌ترند، از طرفی یکی از خصوصیات مهم و پرکاربرد مشتقات کسری توضیح خواص ارثی و حافظه مواد مختلف و روند مربوط به وجود مدت حافظه در مدل ارائه شده می‌باشد. معادلات کسری مرتبه ثابت در سیستم‌های پیچیده‌ای مانند پدیده‌های پخش زبستی [۲]، غلظت مواد در محیط‌های ناهمگن [۳]، قابلیت خوبی در توصیف آنها نداشته و این بدان معناست که مشتق کسری ممکن است تابعی از زمان یا مکان یا پارامترهای سیستم باشد. این عملگر کسری با عنوان مشتق کسری مرتبه متغیر نام‌گذاری شده است که اولین بار توسط سامکو و رز^۱ معرفی شده‌اند. [۸]

یکی از دغدغه‌های مهم در معادلات دیفرانسیل کسری به دست آوردن جواب برای آنها می‌باشد، که در اکثر موارد به دست آوردن جواب تحلیلی پیچیده یا غیرممکن است، لذا روش‌های عددی در حل این دسته از معادلات از اهمیت بسزایی برخوردار است. از جمله روش‌های

^۱Samko and Ross



عددی بکار گرفته شده می‌توان از روش هم‌محلی طیفی [۱۰]، روش عناصر متناهی نام برد [۹]. در سال ۱۹۰۸ ریتز روشی عددی برای حل معادلات دیفرانسیل با شرایط مرزی و اولیه ارائه داد. این روش در حل مسائل فیزیکی نوسانی و پدیده‌های تکرارشونده کاربرد فراوان دارد. در [۳] با استفاده از روش ریتز و ماتریس شبه عملیاتی به حل معادلات دیفرانسیل مرتبه متغیر کسری پرداخته شده است، در [۷] با استفاده از تقریب ریتز به حل مسائل معکوس پرداخته شده است. در [۶] با تقریب ریتز و روش مینیم سازی معادلات دیفرانسیل جزئی کسری حل شده است. در این مقاله با استفاده از روش تقریب ریتز، به حل دسته‌ای از معادلات دیفرانسیل کسری مرتبه متغیر که به صورت زیر است می‌پردازیم.

$$\begin{cases} D^{\alpha(x)}u(x) + a(x)u'(x) + b(x)u(x) + c(x)u(\tau(x)) = f(x), & x \in [0, 1], \\ u(0) = \lambda_0, & u(1) = \lambda_1, \end{cases} \quad (1.1)$$

که در آن $0 \leq \tau(x) \leq 1$ ، $\alpha(x), \tau(x), f(x) \in C[0, 1]$ ، $a(x), b(x), c(x) \in C^r[0, 1]$ و $D^{\alpha(x)}$ نشان‌دهنده مشتق کاپوتو^۲ با $n-1 \leq \alpha(x) < n$ می‌باشد.

۲ تعاریف و پیش‌نیازها

در این بخش بعضی از تعاریف مورد نیاز حساب کسری و چندجمله‌ای‌های متعامد لژاندر انتقال یافته و تابع برآورده^۳ را بیان می‌کنیم. تعاریف مختلفی برای مشتق کسری وجود دارد که از جمله مهم‌ترین آن‌ها می‌توان از مشتق ریمان-لیوویل^۴ و کاپوتو نام برد، که ما در این جا بنا بر ضرورت، به تعریف مشتق کسری مرتبه متغیر کاپوتو می‌پردازیم.

تعریف ۰.۱. مشتق کسری کاپوتو مرتبه متغیر برای تابع پیوسته^۵ $u(x) : [a, \infty) \rightarrow \mathbb{R}$ که در آن $n = [\max_{0 \leq x \leq 1} \alpha(x)] + 1$ است، به صورت زیر تعریف می‌شود

$${}_C D^{\alpha(x)}u(x) = \begin{cases} \frac{1}{\Gamma(n-\alpha(x))} \int_0^x (x-t)^{n-\alpha(x)-1} \frac{d^n u(t)}{dt^n} dt, & n-1 < \alpha(x) < n, \\ \frac{d^n u(x)}{dx^n}, & \alpha(x) = n, \end{cases} \quad (2.1)$$

همچنین $\Gamma()$ ، نشان‌دهنده تابع گاما است. اگر $\alpha(x)$ ، یک مقدار صحیح باشد آنگاه مشتق کسری همانند مشتق مرتبه صحیح عمل می‌کند.

تعریف ۰.۲. عملگر کسری کاپوتو دارای خاصیت خطی می‌باشد

$$D^{\alpha(x)}(\lambda f(x) + \mu g(x)) = \lambda D^{\alpha(x)}f(x) + \mu D^{\alpha(x)}g(x),$$

^۲Caputo

^۳Satisfier function

^۴Riemann-Liouville

که λ و μ مقادیر صحیح هستند، هم‌چنین این مشتق دارای خواص زیر نیز می‌باشد

$$D^{\alpha(x)} C = \begin{cases} C \text{ is a constant,} \\ 0, & k = 0, \dots, n-1, \\ \frac{\Gamma(k+1)}{\Gamma(k+1-\alpha(x))} x^{k-\alpha(x)}, & k = n, n+1, \dots \end{cases}$$

در ادامه به تعریف چندجمله‌ای‌های متعامد لژاندر انتقال یافته می‌پردازیم.

تعریف ۳. چندجمله‌ای‌های لژاندر $L_n(x)$ چندجمله‌ای‌های متعامد در بازه $[-1, 1]$ می‌باشند که به صورت زیر تعریف می‌شوند

$$L_n(x) = \frac{1}{(2^n n!)} \cdot \frac{d^n (x^2 - 1)^n}{dx^n}, \quad n = 0, 1, 2, \dots,$$

که مقادیر اولیه آن

$$L_0(x) = 1, \quad L_1(x) = x,$$

و تابع وزن $\omega(x) = 1$ می‌باشد. حال اگر بازه‌ی $[-1, 1]$ به هر بازه‌ی دلخواه $[a, b]$ تغییر یابد، با استفاده از تغییر متغیر $x = \frac{2t-(a+b)}{b-a}$ می‌توان چندجمله‌ای‌های لژاندر انتقال یافته $L_n^*(t)$ را به دست آورد.

این چندجمله‌ای‌ها در بازه $[a, b]$ ، متعامد بوده و ضرب داخلی آن‌ها به صورت زیر تعریف می‌شود

$$\langle L^*(t), L_m^*(t) \rangle = \int_a^b \omega(t) L_n^*(t) L_m^*(t) dt = \frac{b-a}{2n+1} \delta_{mn},$$

که δ دلتای کرونگر می‌باشد.

۳ شرح روش تقریب ریتز

روش تقریب ریتز روشی کارآمد و ساده در حل معادلات دیفرانسیل مقدار مرزی و اولیه می‌باشد. از این تقریب به همراه چندجمله‌ای‌های لژاندر انتقالی و روش هم‌محلی استفاده می‌کنیم. علاقه‌مندان برای مطالعه‌ی بیشتر می‌توانند به [۱] مراجعه کنند. در این تقریب تابع مجهول $u(x)$ در معادله‌ی (۱۱) بصورت زیر در نظر گرفته می‌شود:

$$u(x) \cong u_k(x) = \sum_{n=0}^k c_n \phi_n(x) + w(x). \quad (31)$$

در معادله‌ی فوق $\phi_n(x)$ باید بگونه‌ای انتخاب شود که در فرم همگن مساله‌ی مقدار مرزی (۱۱) صدق کند [۵]. در اینجا

$$\phi_n(x) = x(x-1)L_n^*(x)$$

در نظر گرفته می‌شود. تابع $w(x)$ را تابع برآورنده گویند و این تابع باید در شرایط مرزی مساله‌ی (۱۱) صدق کند یعنی:

$$w(0) = u(0), \quad w(1) = u(1).$$



انتخاب تابع برآورنده نقش بسزایی در میزان دقت و حجم محاسبات دارد. تجربه نشان داده که چون در ساخت تابع برآورنده از اطلاعات مساله استفاده می‌شود این تابع تقریباً نزدیک جواب دقیق مساله بدست می‌آید. در ساده‌ترین حالت می‌توان تابع برآورنده را با استفاده از شرایط مرزی مساله و روش درونیابی محاسبه کرد. با جایگذاری رابطه‌ی (۳۱) در معادله‌ی (۱۱) مساله به حالت گسسته‌ی زیر تبدیل می‌شود:

$${}^c D^{\alpha(x)} u_k(x) + a(x)u'_k(x) + b(x)u_k(x) + c(x)u_k(\tau(x)) = f(x). \quad (32)$$

حال با استفاده از روش هم‌محلی و ریشه‌های چندجمله‌ای لژاندر به عنوان نقاط هم‌محلی به حل معادله‌ی (۳۲) می‌پردازیم تا ضرایب مجهول c_n را بدست آوریم. محاسبات به کمک نرم‌افزار Mathematica10 انجام شده است.

۴ نتایج عددی

روش بیان شده را با حل یک مثال به کار می‌گیریم. نمودار و جدول مربوطه نشان دهنده‌ی کارایی و دقت روش بیان شده می‌باشد. خطای مطلق به صورت زیر در محاسبات در نظر گرفته شده است

$$E_k = |u(x) - u_k(x)|.$$

مثال ۴. معادله دیفرانسیل کسری مرتبه متغیر زیر را در نظر بگیرید

$$\begin{cases} {}^C D^{\alpha(x)} u(x) + 3u'(x) - u(x) = f(x), & x \in [0, 1], \\ u(0) = 1, \end{cases} \quad (41)$$

که $\alpha(x) = \frac{1 + \cos^2(x)}{4}$ و تابع $f(x)$ برابر با مقدار حاصل از جایگذاری جواب دقیق $u(x)$ در معادله‌ی (۴۱) می‌باشد. جواب دقیق مساله بصورت $u(x) = e^x$ می‌باشد. با توجه به شرایط مرزی مساله تابع برآورنده آن را می‌توان بصورت زیر در نظر گرفت:

$$w(x) = e^x,$$

و مقادیر c_k ها، به صورت زیر به دست می‌آیند

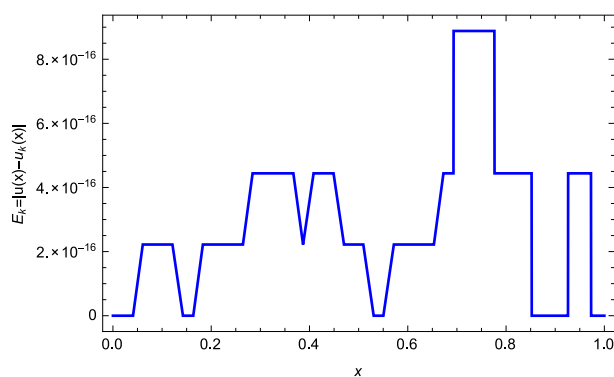
$$\begin{aligned} c_0 &= -9.83781 \times 10^{-16}, & c_1 &= -2.2457 \times 10^{-15}, & c_2 &= -2.81895 \times 10^{-15}, \\ c_3 &= -6.35467 \times 10^{-15}, & c_4 &= -3.1376 \times 10^{-15}. \end{aligned}$$

در جدول ۱ خطای مطلق روش بیان شده و روش ارائه شده در [۳] مقایسه شده است. هم‌چنین در نمودار ۱ خطای مطلق و در نمودار ۲ نمودار جواب تقریبی و دقیق رسم شده است.

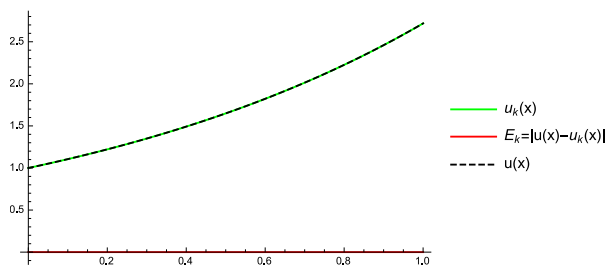


جدول ۱: مقایسه‌ی خطای مطلق مثال ۳.۲.

x	خطای مطلق در [۳] برای $n = 11$	خطای مطلق روش برای $n = 5$
۰/۱	$7,0166 \times 10^{-14}$	$6,79951 \times 10^{-17}$
۰/۳	$7,5495 \times 10^{-14}$	$3,76757 \times 10^{-16}$
۰/۵	$7,2075 \times 10^{-13}$	$1,87727 \times 10^{-16}$
۰/۷	$9,4946 \times 10^{-13}$	$4,20308 \times 10^{-16}$
۰/۹	$8,2162 \times 10^{-12}$	$3,46894 \times 10^{-16}$



شکل ۱: خطای مطلق تقریب ریتز برای مثال ۳.۲.



شکل ۲: نمودار تابع جواب تقریب و جواب دقیق مثال ۳.۲.

مراجع

- [1] A. Lotfi, S. A. Yousefi, *A numerical technique for solving a class of fractional variational problems*, J. Computat. Appl. Math. **237** (2013) 633-643 .
- [2] E.N. de Azevedo, P.L. de Sousa, R.E. de Souza, M. Engelsberg, M.d.N.d.N. Miranda and

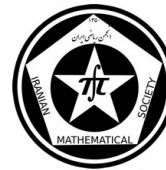


- M.A. Silva, *Concentration dependent diffusivity and anomalous diffusion: A magnetic resonance imaging study of water ingress in porous zeolite*, Phys. Rev. **73** (2006) , 201–204.
- [3] F. S. Yousefi, Y. Ordokhani, S. Yousefi *Numerical solution of variable order fractional differential equations by using shifted Legendre cardinal functions and Ritz method*. Engineering with Computers, (2020), 1-8.
- [4] K. Atkinson, W. Han, D.E. Stewart, *Numerical solution of ordinary differential equations*. John Wiley and Sons, 2011.
- [5] L. P. Lebedev, M. j. Cloud, *The calculus of variations and functional analysis: With optimal control and applications in mechanics*, World Scientific, **12** (2003).
- [6] M. A. Firoozjaee, S. A. Yousefi, *A numerical approach for fractional partial differential equations by using Ritz approximation*, Applied Mathematics and Computation, **338** (2018), 711-721.
- [7] M. N. Khorshidi, S. A. Yousefi, M. A. Firoozjaee, *Determination of an unknown source term for an inverse source problem of the time-fractional equation*. Asian-European Journal of Mathematics, **10** (2017), 1750031.
- [8] S.G. Samko and B. Ross, *Integration and differentiation to a variable fractional order*, Integral Transf. Special Funct., **1** (1993), 277–300.
- [9] R. Lin, F. Liu, V. Anh and I. Turner, *Stability and convergence of a new explicit finite-difference approximation for the variable-order nonlinear fractional diffusion equation*, Appl. Math. Comput. **212** (2009), 435–445.
- [10] W.Y. Tian, W. Deng and Y. Wu, *Polynomial spectral collocation method for space fractional advection-diffusion equation*, Numer. Methods Partial Differential Equations, **30** (2014), 514–535.

پست الکترونیکی: smy.sheikhi93@gmail.com

پست الکترونیکی: m.matinfar@umz.ac.ir

پست الکترونیکی: m_firoozjaee@mazust.ac.ir



حل عددی معادله تلگراف کسری با روش اجزای حجمی محدود

اله‌بخش یزدانی چراتی[†]، محمود بدر و زهره عظیمی[‡]

دانشکده علوم ریاضی، دانشگاه مازندران، بابلسر، ایران

چکیده

در این مقاله، روش اجزای حجمی محدود برای حل معادله تلگراف با مشتق کسری-زمانی را ارائه می‌دهیم. سپس نشان می‌دهیم که روش، پایدار نامشروط است و با حل مثال عددی، دقت روش مورد نظر را نیز بررسی می‌کنیم. واژه‌های کلیدی: مشتق کسری کاپاتو، معادله تلگراف کسری، روش اجزای حجمی محدود، پایداری. طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

علم ریاضی این امکان را برای بشر فراهم آورده که با مدل‌سازی اکثر پدیده‌ها، طبیعت را تحت کنترل خود داشته باشد [۲]. پدیده‌های طبیعی زیادی به صورت یک معادله دیفرانسیل کسری مدل‌بندی می‌شوند. حسابان کسری موضوعی است که بیش از ۳۰۰ سال قدمت دارد. ایده‌ی حسابان کسری به زمان حسابان معمولی برمی‌گردد. نخستین بار در سال ۱۶۹۵ این موضوع را لایبنیتس و هوییتال مطرح کردند [۳]. از معادلات دیفرانسیل کسری می‌توان به معادلات تلگراف، انتشار و شرودینگر اشاره کرد. معادلات تلگراف، یک جفت معادلات دیفرانسیل خطی است که ولتاژ و جریان در خطوط انتقال الکتریکی را با مکان و زمان توصیف می‌کند. این معادلات در مقاله‌ای از ارسینگر و ژایو حل شد و سپس راه‌های مختلف برای حل آن ارائه شد [۴]. حل این معادلات و یافتن جواب دقیق، اهمیت زیادی در علوم مختلف دارد. روش تفاضلات متناهی، روش حجم محدود، روش اجزای محدود و روش اجزای حجمی محدود از جمله روش‌های پرکاربرد برای حل اینگونه مسائل هستند. در این مقاله، یک معادله تلگراف کسری را با روش اجزای حجمی محدود حل خواهیم کرد.

۲ روش اجزای حجمی محدود برای حل معادلات تلگراف کسری

معادله تلگراف کسری مرتبه α ($1 < \alpha \leq 2$) با شرایط مرزی و اولیه به صورت زیر را در نظر می‌گیریم:

$$\frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + \mu_1 \frac{\partial^{\alpha-1} u(x, t)}{\partial t^{\alpha-1}} + \mu_2 u(x, t) = \mu_3 \frac{\partial^2 u(x, t)}{\partial x^2} + f(x, t), \quad a \leq x \leq b, \quad t \geq 0, \quad (2.1)$$

$$u(a, t) = h_1(t), \quad u(b, t) = h_2(t), \quad t \geq 0,$$

$$u(x, 0) = g_1(x), \quad u_t(x, 0) = g_2(x), \quad a \leq x \leq b.$$

که در آن $h_2(x), h_1(x), g_2(x), g_1(x)$ توابعی معلوم هستند و μ_1, μ_2, μ_3 ضرایب ثابت داده شده می‌باشند. مشتق کسری به صورت زیر تعریف می‌شود:

$$\frac{\partial^\alpha u(x, t)}{\partial t^\alpha} = \begin{cases} \frac{1}{\Gamma(2-\alpha)} \int_0^t (t-\xi)^{1-\alpha} \frac{\partial^2 u(x, \xi)}{\partial \xi^2} d\xi, & 1 < \alpha < 2, \\ \frac{\partial^2 u(x, t)}{\partial t^2}, & \alpha = 2. \end{cases}$$

با توجه به اینکه $0 < \alpha - 1 \leq 1$ داریم:

$$\frac{\partial^{\alpha-1} u(x, t)}{\partial t^{\alpha-1}} = \begin{cases} \frac{1}{\Gamma(2-\alpha)} \int_0^t (t-\xi)^{1-\alpha} \frac{\partial u(x, \xi)}{\partial \xi} d\xi, & 1 < \alpha < 2, \\ \frac{\partial u(x, t)}{\partial t}, & \alpha = 2. \end{cases}$$

فرض می‌کنیم $\Omega = (a, b)$. یک افراز یکنواخت $\xi_i = [x_{i-1}, x_i]$ را روی دامنه Ω با نقاط گرهی $x_i = ih$ ($i = 0, 1, 2, \dots, M$) معرفی می‌کنیم. قرار می‌دهیم $h = x_i - x_{i-1}$. همچنین فرض می‌کنیم $t_k = k\tau$ ($k = 0, 1, 2, \dots, n$). سپس فضای $S_h \subset H_0^1(\Omega)$ را به شکل زیر تعریف می‌کنیم:

$$S_h = \{v_h \in C(\Omega) : v_h|_{\xi_i} \text{ و خطی می‌باشد} \text{ و } v_h|_{\partial\Omega} = 0\}.$$

مجموعه S_h یک پایه برای S_h است که در آن

$$\varphi_i(x) = \begin{cases} \frac{x-x_{i-1}}{h}, & x_{i-1} \leq x \leq x_i, \\ \frac{x_{i+1}-x}{h}, & x_i \leq x \leq x_{i+1}. \end{cases}$$

S_h بعنوان فضای جواب روش اجزای حجمی محدود شناخته می‌شود. برای هر $u_h(x, t) \in S_h$ داریم:

$$u_h(x, t) = \sum_{i=0}^M \delta_i(t) \varphi_i(x), \quad t > 0,$$

که $\delta_i(t)$ ضرایب وابسته به زمان و مجهول هستند و در این مقاله با روش اجزای حجمی محدود محاسبه می‌شوند. برای هر $x_i \in \Omega$ $V_i = [x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ را بعنوان حجم‌های کنترلی در نظر می‌گیریم، پس از نوشتن فرم تغییراتی معادله (۲.۱) و انتگرال‌گیری روی



با در نظر گرفتن $\xi_m^k = v^k e^{im\varphi}$ که در آن $i := \sqrt{-1}$ است، به رابطۀ زیر می‌رسیم:

$$\begin{aligned} & v^{k+1} \{ (\mathfrak{z}r + p + w - z)e^{-i\varphi} + (\mathfrak{z}r + \mathfrak{e}p + \mathfrak{e}w + \mathfrak{z}z) + (\mathfrak{z}r + p + w - z)e^{i\varphi} \} \\ &= \sum_{s=0}^k b_s \left[(b_s - b_{s+1})v^{k-s} \right] \times \left[(\mathfrak{z}r + p)e^{-i\varphi} + (\mathfrak{z}r + \mathfrak{e}p) + (\mathfrak{z}r + p)e^{i\varphi} \right] \\ &+ b_k \left[(\mathfrak{z}r + p)e^{-i\varphi} + (\mathfrak{z}r + \mathfrak{e}p) + (\mathfrak{z}r + p)e^{i\varphi} \right], \end{aligned}$$

یا

$$v^{k+1} = R \{ b_k v^\circ + \sum_{s=0}^{k-1} (b_s - b_{s+1}) v^{k-s} \}, \quad (3.1)$$

که در آن

$$R = \frac{\mathfrak{z}r + \mathfrak{e}p + \mathfrak{z}(\mathfrak{z}r + p) \cos(\varphi)}{\mathfrak{z}r + \mathfrak{e}p + \mathfrak{e}w + \mathfrak{z}z + \mathfrak{z}(\mathfrak{z}r + p + w - z) \cos(\varphi)}.$$

به سادگی رابطۀ $|R|^2 \leq 1$ برقرار است. حال برای بررسی شرط $|v^k| \leq 1$ با استفاده از استقرای ریاضی و با شروع از $k = 0$ در دستور (3.1) رابطۀ

$$v^1 = R b_0 v^\circ,$$

را خواهیم داشت. در این صورت

$$|v^1| = |R| |b_0| |v^\circ| \leq |v^\circ|.$$

فرض کنیم $|v^j| \leq |v^\circ|$ ، $(j = 1, 2, \dots, k)$. بنابر این با توجه به رابطۀ (3.1) داریم

$$|v^{k+1}| \leq b_k |v^\circ| + \sum_{s=0}^{k-1} (b_s - b_{s+1}) |v^{k-s}| \leq b_k |v^\circ| + \sum_{s=0}^{k-1} (b_s - b_{s+1}) |v^\circ| = |v^\circ| (b_k + \sum_{s=0}^{k-1} (b_s - b_{s+1})) = |v^\circ|.$$

لذا روش مورد استفاده پایدار نامشروط خواهد بود و اثبات قضیه کامل می‌شود.

□

قضیه ۲. فرض کنید u و u_h به ترتیب جواب دقیق و جواب تقریبی معادله (2.1) باشد. مرتبه همگرایی روش پیشنهادی (2.3) روی معادله (2.1) برای زمان از مرتبه $3 - \alpha$ و برای مکان از مرتبه ۲ است.

۴ مثال عددی

در این بخش، با استفاده از مثال عددی، دقت روش اجزای حجمی محدود برای حل معادلات خطی تلگراف را مورد بررسی قرار می‌دهیم.



مثال ۳. معادله تلگراف کسری (۲۰۱) را در بازه $[0, 1]$ در نظر بگیرید. و $\mu_1, \mu_2, \mu_3 = 1$ و $f(x, t)$ هنگامی که جواب دقیق $u(x, t) = t^2 x^2 (1-x)^2$ است به صورت زیر بدست می آید:

$$f(x, t) = \frac{t^{2-\alpha}}{\Gamma(3-\alpha)} x^2 (1-x)^2 + \frac{t^{3-\alpha}}{\Gamma(4-\alpha)} x^2 (1-x)^2 + t^2 x^2 (1-x)^2 - 12t^2 x^2 + 12t^2 x - 2t^2,$$

در جدول ۱، مقادیر جواب عددی در زمان t_1 نمایش داده شده اند.

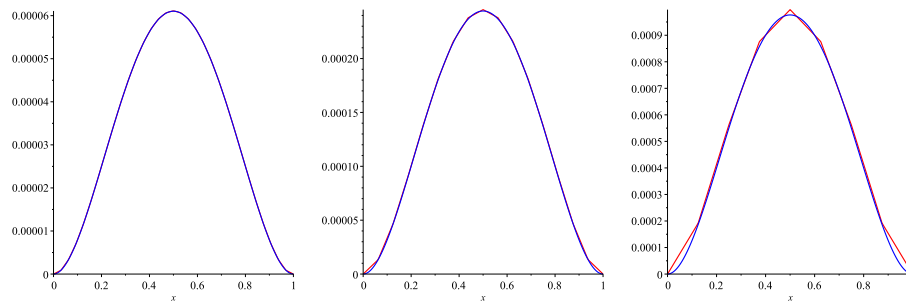
جدول ۱: خطاهای محاسبه شده از جوابهای تقریبی برای $h = 0.001$, $\tau_i = \frac{1}{\tau_i}$ در زمان t_1 .

مرتبۀ همگرایی	نرم - ∞	نرم - ۲	τ	i	α
-	$0.2393e-4$	$0.1655e-4$	0.5	۱	۱.۰۵
۱.۸۳	$0.4856e-5$	$0.3341e-5$	0.25	۲	
۱.۹۳	$8.8820e-7$	$6.0581e-7$	0.125	۳	
۱.۹۹	$9.3583e-8$	$9.3583e-8$	0.0625	۴	
-	$0.1236e-2$	$0.8504e-3$	0.5	۱	۱.۹۵
0.95	$0.2792e-3$	$0.1884e-3$	0.25	۲	
۱.۰۳	$0.4618e-4$	$0.3027e-4$	0.125	۳	
۱.۰۶	$0.6464e-5$	$0.4157e-5$	0.0625	۴	

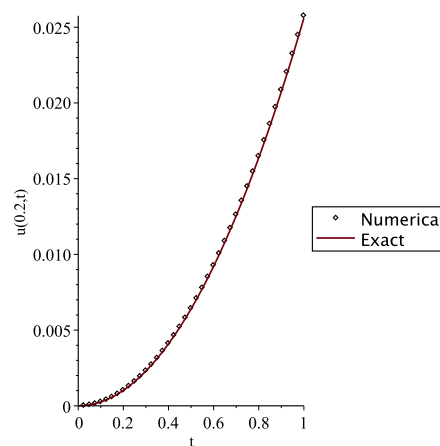
جدول ۲: خطاهای محاسبه شده از جوابهای تقریبی برای $\tau = 0.001$, $h_i = \frac{1}{\tau_i}$ در زمان t_1 .

مرتبۀ همگرایی	نرم - ∞	نرم - ۲	h	i	α
-	$8.0726e-9$	$5.7082e-9$	0.5	۱	۱.۰۵
۱.۴۸	$5.7747e-9$	$2.9002e-9$	0.25	۲	
۱.۶۵	$1.30213e-9$	$9.5209e-10$	0.125	۳	
۱.۷۸	$4.4883e-10$	$2.5374e-10$	0.0625	۴	
-	$8.0181e-9$	$5.6696e-9$	0.5	۱	۱.۹۵
۱.۴۴	$5.8977e-9$	$2.9669e-9$	0.25	۲	
۱.۵۳	$1.3318e-9$	$1.0002e-9$	0.125	۳	
۱.۸۹	$5.3210e-10$	$2.8270e-10$	0.0625	۴	

شکل ۱: شکل‌ها به ترتیب از راست به چپ برای $M = n = 8$, $M = n = 16$ و $M = n = 32$ است.



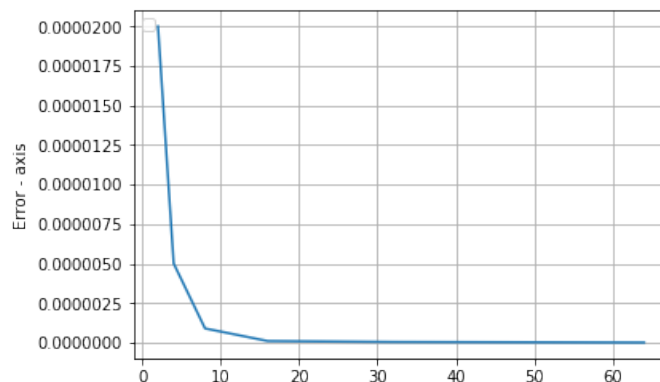
شکل ۲: جواب عددی و دقیق مثال (۳)



در شکل ۱، جواب تقریبی و دقیق مثال (۳) برای $\alpha = 1/5$ با بکارگیری روش عددی اجزای حجمی محدود (۲۰۲) در زمان و مکان‌های مختلف نشان داده شده است. همان‌طور که در شکل مشخص است، با کم شدن طول گام مکانی h و طول گام زمانی τ ، خطای روش پیشنهادی از مرتبه $O(\tau^{3-\alpha} + h^2)$ می‌باشد. شکل ۲، جواب‌های عددی و دقیق مثال (۳) برای $x = 0.2$ و همچنین $\alpha = 1/5$ و $n = 40$ و $M = 20$ است. همچنین در جدول ۲، $N = 1000$ را در نظر می‌گیریم و به ترتیب برای $h = 2, 4, 8, 16$ نرم ۲- و نرم ∞ را بدست آورده‌ایم. نتایج بدست آمده در جدول ۱ و شکل ۱ و ۲، دقت و کارایی روش مورد استفاده را تایید می‌کند. اعداد بدست آمده برای جوابهای دقیق و تقریبی را به وسیله نرم‌افزار میپل را محاسبه کرده‌ایم.



شکل ۳: خطای مربوط به مثال فوق برای $\alpha = 1/5$, $h = 10^{-5}$ و $N = 16$



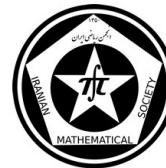
مراجع

- [1] M. Badr, A. Yazdani and H. Jafari, *Stability of a finite volume element method for the time-fractional advection-diffusion equation*, Numerical Methods for Partial Differential Equations, 34(5). (2018), 1459–1471.
- [2] F.R. Giordano, M.D. Weir and W.P. Fox, *A First Course in Mathematical Modelling*, by Books/Cole Publishing Company, 2nd, 1997.
- [3] I. Podlubny, *Fractional differential equations*, San Diego. CA: Academic Press, 1999.
- [4] F. Liu, P. Zhuang, V. Anh and I. Turner, *A fractional-order implicit difference approximation for the space time fractional diffusion equation*, ANZIAM. (2006), 47:48–68.

پست الکترونیکی: yazdani@umz.ac.ir

پست الکترونیکی: m.bard@stu.umz.ac.ir

پست الکترونیکی: azimi.umz@gmail.com



حل مدل بازه‌ای بلک-شولز با استفاده از روش گسسته‌سازی

دکتر سمیه زنگویی‌زاده[†] و محدثه سروری[‡]

کرمان، دانشگاه شهید باهنر کرمان، دانشکده ریاضی و کامپیوتر، بخش ریاضی کاربردی

چکیده

مدل بلک-شولز یک ابزار قوی برای ارزش‌گذاری اختیار معامله‌ها است. در این مقاله، به منظور مطابقت بیشتر این مدل با واقعیت‌های بازار، پارامترهای ثابت موجود در مدل، نرخ بهره بدون ریسک و نوسان‌پذیری قیمت سهام به صورت بازه‌ای در نظر گرفته شده است و به حل مدل بازه‌ای به دست آمده، با استفاده از روش گسسته‌سازی پرداخته‌ایم. در پایان یک مثال عددی ارائه می‌شود که تا حدودی نشان‌دهنده کارایی این روش می‌باشد.

واژه‌های کلیدی: مدل بلک-شولز، روش گسسته‌سازی، عدم قطعیت، آنالیز بازه‌ای.

طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

مدل بلک-شولز یک ابزار قوی برای ارزش‌گذاری اختیار معامله‌ها علی‌الخصوص اختیار معامله اروپایی است. اختیار معامله توافقی است بین واگذارکننده و دارنده‌ی اختیار حق خرید و یا فروش دارایی تعهد شده؛ بدین صورت که دارنده‌ی اختیار، حق خرید یا فروش دارایی خاصی را با قیمت مشخصی (قیمت توافقی) و در تاریخ معینی در زمان آینده (تاریخ سررسید) خواهد داشت. دارنده‌ی اختیار معامله در قبال این اختیار باید بهایی بپردازد، که به آن قیمت اختیار گویند [۲].

رویکرد تحلیلی و عددی دو راهکار برای حل این مدل‌ها است. اقلیاردی^{*} و همکاران در سال ۲۰۱۳ میلادی در مقاله خود [۳] با استفاده از تبدیلات ملین، معادله بلک-شولز را به صورت تحلیلی حل نمودند. یرموکانوا[†] و همکاران در سال ۲۰۱۶ میلادی در مقاله خود [۷] با تبدیل مدل بلک-شولز به معادله دیفرانسیل معمولی و همچنین با استفاده از روش جداسازی متغیرها به حل مدل بلک-شولز پرداختند. نورول انور[‡] و همکاران در مقاله خود [۶] در سال ۲۰۱۸ میلادی برای حل عددی مدل بلک-شولز از طرح تفاضلات متناهی

^{*} Agliardi

[†] Yermukanova

[‡] Nurul Anwar

استفاده کردند و پایداری طرح را نیز مورد بررسی قرار دادند.

از آنجایی که در دنیای واقعی پارامترهای مسائل مقدار دقیق نیستند، عدم قطعیت در پارامترهای یک مسأله نمود پیدا می‌کند. یک راه برای نمایش این عدم قطعیت استفاده از بازه‌های حقیقی می‌باشد [۵]. در این مقاله قصد داریم مدل بلک-شولز به صورت

$$\frac{\partial V}{\partial t} = rs(t) \frac{\partial V}{\partial s} + \frac{1}{2} \sigma^2 s^2(t) \frac{\partial^2 V}{\partial s^2} + rV(s, t)$$

را با در نظر گرفتن عدم قطعیت در پارامترهای r و σ بیان کرده و روشی برای حل آن ارائه دهیم. و در پایان یک مثال عددی آورده‌ایم.

۲ مفاهیم اولیه

تعریف ۱. مجموعه‌ی بازه‌ها روی محور اعداد حقیقی به صورت $\mathbb{IR} = \{x = [\underline{x}, \bar{x}] | \underline{x}, \bar{x} \in \mathbb{R}\}$ تعریف می‌شود. که در آن $\underline{x}$ نقطه‌ی ابتدا و $\bar{x}$ نقطه‌ی انتهای بازه نامیده می‌شود [۵].

فرض کنید I خانواده تمام بازه‌های غیرتهی، فشرده و محدب از خط حقیقی $\mathbb{R}$ باشد. جمع و ضرب اسکالر در I ، برای هر دو بازه دلخواه $a = [\underline{a}, \bar{a}]$, $b = [\underline{b}, \bar{b}]$ و $\lambda \geq 0$ به صورت زیر تعریف می‌شود:

$$a + b = [\underline{a} + \underline{b}, \bar{a} + \bar{b}], \quad \lambda a = [\lambda \underline{a}, \lambda \bar{a}], \quad (-\lambda) a = [-\lambda \bar{a}, -\lambda \underline{a}].$$

تعریف ۲. متر هاسدورف H ، برای هر دو بازه دلخواه a و b در فضای I به صورت زیر تعریف می‌شود:

$$H(a, b) = \max \{|\underline{a} - \underline{b}|, |\bar{a} - \bar{b}|\}.$$

تعریف ۳. فرض کنید a, b دو بازه دلخواه در I باشند. اگر بازه $c \in I$ وجود داشته باشد بطوریکه $a = b + c$. c را تفاضل هاکوهارای a و b می‌نامیم و به صورت $a \ominus b$ نشان می‌دهیم [۵].

تعریف ۴. تابع بازه‌ای مقدار $f: [\alpha, \beta] \rightarrow I$ را در نظر بگیرید. تابع f را در نقطه‌ی $x \in (\alpha, \beta)$ ، مشتق‌پذیر هاکوهارا نامیم اگر حد
$$f'(x) = \lim_{h \rightarrow 0} \frac{f(x+h) \ominus f(x)}{h}$$
 موجود و $f'(x) \in I$ باشد [۴].

در نتیجه، دو نوع تعریف برای مشتق‌پذیری تابع f در نقطه‌ی $x \in (\alpha, \beta)$ بوجود می‌آید، که به صورت زیر ارائه می‌شود.

تعریف ۵. فرض کنید تابع بازه‌ای مقدار $f: [\alpha, \beta] \rightarrow I$ در نقطه‌ی $x \in (\alpha, \beta)$ مشتق‌پذیر هاکوهارا باشد. در اینصورت می‌گوییم، f ، i -مشتق‌پذیر در نقطه‌ی $x \in (\alpha, \beta)$ است اگر،

$$f'_i = \left[(f')'(x), (\bar{f}')'(x) \right]$$

و f ، ii -مشتق‌پذیر در نقطه‌ی $x \in (\alpha, \beta)$ است اگر،

$$f'_{ii} = \left[(\bar{f}')'(x), (f')'(x) \right]$$

باشد [۴].

۳ تعمیم مدل بلک-شولز به مدل بازه‌ای

معادله دیفرانسیل جزئی مرتبه دوم زیر به مدل بلک-شولز معروف است، که در آن $V = V(s, t)$ قیمت اختیار معامله، $s(t)$ قیمت سهام، r نرخ بهره بدون ریسک و σ نوسان‌پذیری قیمت سهام تعریف می‌شود.

$$\frac{\partial V}{\partial t} = rs(t) \frac{\partial V}{\partial s} + \frac{1}{2} \sigma^2 s^2(t) \frac{\partial^2 V}{\partial s^2} + rV(s, t). \quad (3.1)$$

یک معادله‌ی دیفرانسیل در حالت کلی نیاز به شرایط مرزی و نهایی دارد که توسط آن بتوان به یک جواب منحصر به فرد معادله دیفرانسیل دست یافت. حال شرایط مرزی و نهایی را برای اختیار معامله خرید و فروش اروپایی بیان می‌کنیم.

شرایط مرزی و نهایی اختیار معامله خرید

$$\begin{aligned} V(\circ, t) &= \circ, & \text{for } \circ \leq t \leq T, \\ V(s, t) &= s - Ke^{-r(T-t)}, & \text{as } s \rightarrow \infty, \\ V(s, T) &= (s - K)^+, & \text{for } t = T. \end{aligned}$$

شرایط مرزی و نهایی اختیار معامله فروش

$$\begin{aligned} V(\circ, t) &= Ke^{-r(T-t)}, & \text{for } \circ \leq t \leq T, \\ V(s, t) &= \circ, & \text{as } s \rightarrow \infty, \\ V(s, T) &= (K - s)^+, & \text{for } t = T. \end{aligned}$$

در شرایط فوق K نشان‌دهنده قیمت توافقی است و منظور از $s \rightarrow \infty$ زمانی است که قیمت دارایی به بی‌نهایت میل می‌کند، البته قیمت دارایی تا حد خاصی بالا می‌رود که این حد را در این مقاله با s_{max} نشان داده‌ایم. همچنین منظور از $V(\circ, t)$ ارزش اختیار معامله در زمانی است که قیمت دارایی صفر باشد و ارزش اختیار معامله در زمان سررسید را با $V(s, T)$ نشان می‌دهیم [۲].
مدل ریاضی ارائه شده برای قیمت‌گذاری اختیار معامله خرید و فروش (۳.۱)، را به مدل بازه‌ای زیر تعمیم می‌دهیم:

$$\frac{\partial \mathbf{V}}{\partial t} = -\mathbf{r}s(t) \frac{\partial \mathbf{V}}{\partial s} - \frac{1}{2} \sigma^2 s^2(t) \frac{\partial^2 \mathbf{V}}{\partial s^2} + \mathbf{rV}. \quad (3.2)$$

که در رابطه‌ی فوق،

$\mathbf{r}$ ، نرخ بهره بدون ریسک تحت بازه $[T, \bar{T}]$ با این شرط که، $\bar{T} > \circ$ ،

σ ، نوسان‌پذیری قیمت سهام تحت بازه $[\underline{s}, \bar{\sigma}]$ با این شرط که، $\bar{\sigma} > \circ$ ،

$\mathbf{V}$ ، ارزش اختیار معامله تحت قیمت سهام، نوسان‌پذیری قیمت سهام و نرخ بهره بدون ریسک است، به طوریکه $\underline{V} \geq \circ$. توجه. برای نشان دادن عدم قطعیت در پارامترها و در نظر گرفتن آن‌ها به صورت بازه از حروف پر رنگ استفاده کرده‌ایم.
در ادامه عملگر d را به منظور تفاضل دو عدد بازه‌ای به صورت زیر تعریف می‌کنیم:

$$d(\mathbf{u}, \mathbf{v}) = (\underline{u} - \bar{v}, \bar{u} - \underline{v}).$$

تابع بازه‌ای خطای نقطه‌ای برای معادله بلک-شولز بازه‌ای (۳۲) را به صورت زیر تعریف می‌کنیم:

$$\mathbf{R}(s, t) = [\underline{R}(s, t), \bar{R}(s, t)],$$

که برای اختیار معامله خرید به صورت،

$$\underline{R}(s, t) = \frac{\partial V}{\partial t} + \underline{r}s(t)\frac{\partial V}{\partial s} + \frac{1}{2}\underline{\sigma}^2 s^2(t)\frac{\partial^2 V}{\partial s^2} - \bar{r}\bar{V}, \quad (33)$$

$$\bar{R}(s, t) = \frac{\partial \bar{V}}{\partial t} + \bar{r}s(t)\frac{\partial \bar{V}}{\partial s} + \frac{1}{2}\bar{\sigma}^2 s^2(t)\frac{\partial^2 \bar{V}}{\partial s^2} - \underline{r}\underline{V}. \quad (34)$$

و برای اختیار معامله فروش به صورت،

$$\underline{R}(s, t) = \frac{\partial V}{\partial t} + \bar{r}s(t)\frac{\partial V}{\partial s} + \frac{1}{2}\underline{\sigma}^2 s^2(t)\frac{\partial^2 V}{\partial s^2} - \bar{r}\bar{V}, \quad (35)$$

$$\bar{R}(s, t) = \frac{\partial \bar{V}}{\partial t} + \underline{r}s(t)\frac{\partial \bar{V}}{\partial s} + \frac{1}{2}\bar{\sigma}^2 s^2(t)\frac{\partial^2 \bar{V}}{\partial s^2} - \underline{r}\underline{V}. \quad (36)$$

می‌باشد. بدیهی است که $\mathbf{R}(s, t)$ یک تابع بازه‌ای است.

نکته ۶. واضح است که در اختیار معامله خرید و فروش $\underline{R}(s, t) \leq \bar{R}(s, t)$ و $[\underline{R}(s, t), \bar{R}(s, t)]$ بازه بسته و کراندار می‌باشد.

تابع بازه‌ای باقیمانده کل متناظر با معادله بازه‌ای (۳۲) را به صورت زیر تعریف می‌کنیم:

$$\mathbf{TR} = [\underline{TR}, \bar{TR}],$$

که در آن،

$$\underline{TR} = \int_0^{s_{max}} \int_0^T \underline{R}(s, t) dt ds, \quad \bar{TR} = \int_0^{s_{max}} \int_0^T \bar{R}(s, t) dt ds. \quad (37)$$

در نهایت برای به دست آوردن جواب تقریبی معادله بازه‌ای (۳۲) و مقایسه آن با جواب دقیق، تابع باقی‌مانده تجمعی به صورت زیر

تعریف می‌شود:

$$TR = \underline{TR} + \bar{TR}.$$

نکته ۷. با توجه به اینکه با افزایش قیمت سهام، قیمت اختیار معامله فروش کاهش می‌یابد و تقریباً نمودار مربوط به آن به سمت بالا می‌باشد،

می‌توان نتیجه گرفت: $\frac{\partial V}{\partial s} \leq 0$ و $\frac{\partial \bar{V}}{\partial s} \geq 0$.

در ادامه با بیان دو قضیه، مسأله حل دستگاه معادله بازه‌ای (۳۲) به حل مسأله بهینه‌سازی تبدیل می‌گردد [۱].



قضیه ۸. اگر تابع بازه‌ای i -مشتق‌پذیر $V(s, t)$ ، جواب دستگاه معادله بازه‌ای (۳۲) باشد، آنگاه

$$\underline{TR} + \overline{TR} = 0.$$

اثبات. فرض می‌کنیم $V(s, t)$ جواب معادله بازه‌ای (۳۲) برای اختیار معامله خرید باشد در نتیجه با در نظر گرفتن اعمال بازه‌ای داریم:

$$\begin{aligned} \left[\frac{\partial V}{\partial t}, \frac{\partial \bar{V}}{\partial t} \right] &= -[\underline{r}, \bar{r}]s(t) \left[\frac{\partial V}{\partial s}, \frac{\partial \bar{V}}{\partial s} \right] - \frac{1}{\gamma} [\underline{\sigma}^\gamma, \bar{\sigma}^\gamma] s^\gamma(t) \left[\frac{\partial^\gamma V}{\partial s^\gamma}, \frac{\partial^\gamma \bar{V}}{\partial s^\gamma} \right] + [\underline{r}, \bar{r}][V(s, t), \bar{V}(s, t)] \\ &= [-\bar{r}, -\underline{r}] \left[s(t) \frac{\partial V}{\partial s}, s(t) \frac{\partial \bar{V}}{\partial s} \right] + \left[-\frac{1}{\gamma} \bar{\sigma}^\gamma, -\frac{1}{\gamma} \underline{\sigma}^\gamma \right] \left[s^\gamma(t) \frac{\partial^\gamma V}{\partial s^\gamma}, s^\gamma(t) \frac{\partial^\gamma \bar{V}}{\partial s^\gamma} \right] + [\underline{r} V(s, t), \bar{r} \bar{V}(s, t)] \\ &= \left[-\bar{r} s(t) \frac{\partial \bar{V}}{\partial s}, -\underline{r} s(t) \frac{\partial V}{\partial s} \right] + \left[-\frac{1}{\gamma} \bar{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma \bar{V}}{\partial s^\gamma}, -\frac{1}{\gamma} \underline{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma V}{\partial s^\gamma} \right] + [\underline{r} V(s, t), \bar{r} \bar{V}(s, t)] \\ &= \left[-\bar{r} s(t) \frac{\partial \bar{V}}{\partial s} - \frac{1}{\gamma} \bar{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma \bar{V}}{\partial s^\gamma} + \underline{r} V(s, t), -\underline{r} s(t) \frac{\partial V}{\partial s} - \frac{1}{\gamma} \underline{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma V}{\partial s^\gamma} + \bar{r} \bar{V}(s, t) \right]. \end{aligned}$$

بنابراین،

$$\begin{aligned} \frac{\partial V}{\partial t} &= -\bar{r} s(t) \frac{\partial \bar{V}}{\partial s} - \frac{1}{\gamma} \bar{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma \bar{V}}{\partial s^\gamma} + \underline{r} V(s, t), \\ \frac{\partial \bar{V}}{\partial t} &= -\underline{r} s(t) \frac{\partial V}{\partial s} - \frac{1}{\gamma} \underline{\sigma}^\gamma s^\gamma(t) \frac{\partial^\gamma V}{\partial s^\gamma} + \bar{r} \bar{V}(s, t). \end{aligned}$$

با جایگذاری روابط فوق در تساوی (۳۳) و (۳۴) به دست می‌آید:

$$\begin{aligned} \underline{R}(s, t) &= \frac{\partial V}{\partial t} - \frac{\partial \bar{V}}{\partial t}, \\ \bar{R}(s, t) &= \frac{\partial \bar{V}}{\partial t} - \frac{\partial V}{\partial t}. \end{aligned}$$

با جایگذاری تساوی به دست آمده در رابطه فوق، در تعریف تابع بازه‌ای باقیمانده کل (۳۷) داریم:

$$\begin{aligned} \underline{TR} + \overline{TR} &= \int_0^{s_{max}} \int_0^T \underline{R}(s, t) dt ds + \int_0^{s_{max}} \int_0^T \bar{R}(s, t) dt ds \\ &= \int_0^{s_{max}} \int_0^T (\underline{R}(s, t) + \bar{R}(s, t)) dt ds \\ &= \int_0^{s_{max}} \int_0^T \left(\frac{\partial V}{\partial t} - \frac{\partial \bar{V}}{\partial t} + \frac{\partial \bar{V}}{\partial t} - \frac{\partial V}{\partial t} \right) dt ds \\ &= 0, \end{aligned}$$

بنابراین می‌توان نتیجه گرفت:

$$\underline{TR} + \overline{TR} = 0.$$

□

اثبات برقراری این قضیه برای اختیار معامله فروش، با توجه به نکته (۷) مشابه اختیار معامله خرید است.



قضیه ۹. شرط لازم و کافی برای اینکه تابع بازه‌ای i -مشتق‌پذیر $V(s, t)$ جواب دستگاه معادله بازه‌ای (۳۰۲) باشد، این است که در شرط زیر صدق کند:

$$d\left(\frac{\partial V}{\partial t}, R(s, t)\right) = d\left(-rs(t)\frac{\partial V}{\partial s} - \frac{1}{\gamma}\sigma^2 s^2(t)\frac{\partial^2 V}{\partial s^2} + rV, R(s, t)\right). \quad (308)$$

بنابراین با توجه به قضیه (۸) حل معادله بازه‌ای (۳۰۲) به حل معادله $TR = 0$ تبدیل می‌شود. از این رو می‌توان مسأله بهینه‌سازی،

$$\text{Minimize } TR = \int_0^{s_{max}} \int_0^T \underline{R}(s, t) dt ds + \int_0^{s_{max}} \int_0^T \overline{R}(s, t) dt ds. \quad (309)$$

را در نظر گرفت که قیود این مسأله، همان شرایط مرزی و نهایی معادله بلک-شولز می‌باشد.

برای حل معادله (۳۰۹)، ابتدا بازه $[0, s_{max}]$ را به m قسمت و بازه $[0, T]$ را به n قسمت مساوی افزایش می‌کنیم، که در آن m و n اعداد طبیعی ثابت و دلخواه می‌باشند.

بنابراین تابع خطای کل را می‌توان به صورت زیر نوشت:

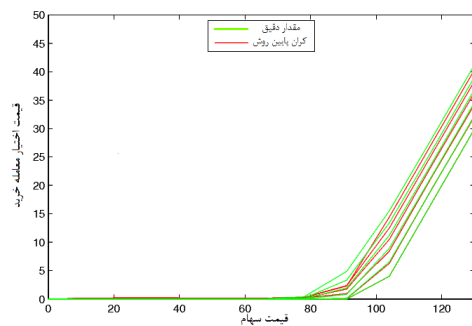
$$\text{Minimize } TR = \sum_{i=0}^m \sum_{j=0}^n \int_{s_i}^{s_{i+1}} \int_{t_j}^{t_{j+1}} (\underline{R}(s, t) + \overline{R}(s, t)) dt ds,$$

از قاعده دوزنقه‌ای برای تقریب انتگرال‌های موجود در عبارت فوق استفاده می‌کنیم، بنابراین داریم:

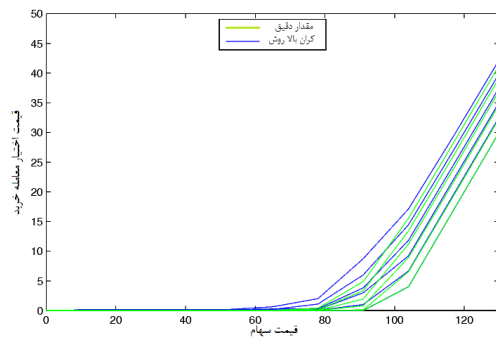
$$\begin{aligned} \text{Minimize } TR = \frac{\Delta s \Delta t}{4} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} & (\underline{R}(s_{i+1}, t_{j+1}) + \underline{R}(s_{i+1}, t_j) + \underline{R}(s_i, t_{j+1}) + \underline{R}(s_i, t_j) + \\ & \overline{R}(s_{i+1}, t_{j+1}) + \overline{R}(s_{i+1}, t_j) + \overline{R}(s_i, t_{j+1}) + \overline{R}(s_i, t_j)). \end{aligned}$$

برای محاسبه‌ی تابع خطای نقطه‌ای در گره‌های مختلف از روش تفاضلات متناهی برای تقریب مشتقات جزئی موجود در تابع خطا، استفاده می‌کنیم، به عنوان مثال $\underline{R}(s_i, t_j)$ و $\overline{R}(s_i, t_j)$ را به صورت زیر در نظر می‌گیریم و در نهایت با استفاده از نرم‌افزار متلب به حل این مسأله بهینه‌سازی مقید می‌پردازیم. در ادامه برای درک بهتر مثالی عددی ارائه خواهیم کرد.

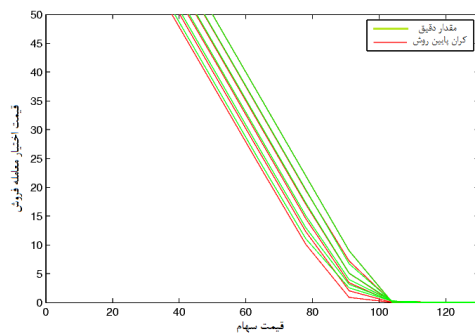
مثال ۱۰. پارامترهای ثابت ورودی معادله بازه‌ای به صورت، $K = 100$ ، $s_{max} = 130$ ، $T = 1$ ، بازه نرخ بهره بدون ریسک را برابر $r = [0.11, 0.13]$ و بازه نوسان‌پذیری قیمت سهام را برابر $\sigma = [0, 0.2]$ در نظر می‌گیریم. و برای مشاهده صحت روش، با در نظر گرفتن پارامترهای این مثال و یک مقدار ثابت بین، بازه نرخ بهره بدون ریسک $r = 0.12$ و نوسان‌پذیری قیمت سهام $\sigma = 0.1$ ، یک جواب دقیق با استفاده از کد $blsprice$ متلب به دست آورده و جواب حاصل را برای قیمت اختیار معامله خرید و فروش در نمودارهای مربوط به کران پایین و بالای به دست آمده از روش گسسته‌سازی نشان داده‌ایم. خطوط سبز رنگ نشان دهنده‌ی جواب دقیق کد $blsprice$ متلب و خطوط قرمز و آبی رنگ مربوط به کران پایین و بالای به دست آمده از روش گسسته‌سازی است. با توجه به نمودارهای زیر مشاهده می‌شود، جواب دقیق به دست آمده از کد $blsprice$ در هر نقطه افزایش بین کران پایین و کران بالا قرار گرفته است.



شکل ۱: نمودار مربوط به مقدار دقیق و کران پایین روش گسسته سازی برای اختیار معامله خرید اروپایی با $m = 10$ و $n = 5$

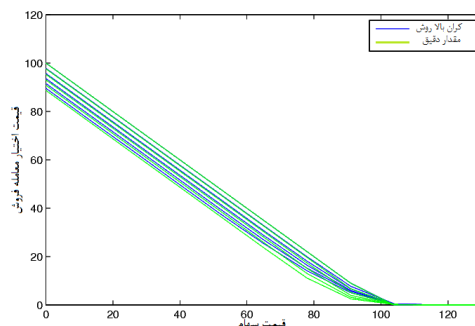


شکل ۲: نمودار مربوط به مقدار دقیق و کران بالای روش گسسته سازی برای اختیار معامله خرید اروپایی با $m = 10$ و $n = 5$



شکل ۳: نمودار مربوط به مقدار دقیق و کران پایین روش گسسته سازی برای اختیار معامله فروش اروپایی با $m = 10$ و $n = 5$

*



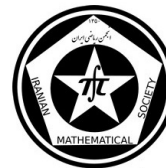
شکل ۴: نمودار مربوط به مقدار دقیق و کران بالای روش گسسته سازی برای اختیار معامله فروش اروپایی با $m = 10$ و $n = 5$

مراجع

- [۱] م. عزیزیان، بهینه سازی برخی مدل های دینامیکی کنترل تومور با رویکرد مدل های دینامیکی فازی، رساله دکتری، دانشکده ریاضی و رایانه، دانشگاه شهید باهنر کرمان، ۱۳۹۷.
- [۲] ج. هال، مبانی مهندسی مالی و مدیریت ریسک، ترجمه سجاد صبیح، علی صالح آبادی، تهران، گروه رایانه تدبیر پرداز، ۱۳۸۴.
- [3] R. Agliardi, P. Popivanov, A. Slovoa, *On Nonlinear Black-Scholes Equations*, Nonl Analysis and Differential Equations, 2013, 1: 75-81.
- [4] T. Allahviranloo, S. Gholami, *Note on "Generalized Hukuhara differentiability of interval-valued functions and interval differential equations"*, Journal of Fuzzy Set Valued Analysis, 2012.
- [5] RE. Moore, RB. Kearfott, MJ. Cloud, *Introduction to interval analysis*, 2009, Siam.
- [6] Md. Nurul Anwar, Andallah, Laek Sazzad, *A study on numerical solution of black-scholes model*, Journal of Mathematical Finance, Scientific Research Publishing, Scientific Research Publishing, 2018, 8: 372-381.
- [7] B.Yermukanova, L.Zhexembay, N.Karjanto, *On a method of solving the Black-Scholes Equation*, 2015, arXiv preprint arXiv:1504.03074.

پست الکترونیکی: zangoei@uk.ac.ir

پست الکترونیکی: m.sarvari94@gmail.com



حل مسأله میانگین-واریانس مارکوتیز با روش‌های بهینه‌سازی درجه دوم

فرزاد راهپیمایی[†]

عضو هیأت علمی گروه ریاضی، دانشگاه پیام نور، صندوق پستی ۱۹۳۹۶-۳۶۹۷، تهران، ایران[†]

چکیده

مدل‌سازی بسیاری از کاربردها در ریاضیات مالی به مسأله بهینه‌سازی تبدیل می‌شود. در این مقاله مسأله میانگین-واریانس مارکوتیز را به یک مسأله بهینه‌سازی درجه دوم تبدیل می‌کنیم و سپس آن را با روش‌های موجود حل می‌کنیم. با استفاده از یک مثال عددی با داده‌های واقعی مدل‌سازی مسأله میانگین-واریانس مارکوتیز و حل مسأله بهینه‌سازی درجه دوم را بیان می‌کنیم. واژه‌های کلیدی: بهینه‌سازی درجه دوم، روش‌های نقطه درونی، میانگین-واریانس مارکوتیز. طبقه‌بندی موضوعی [۲۰۱۰]: 90C20, 90C25, 15A23.

۱ مقدمه

تئوری احتمال، بهینه‌سازی و رابطه بین آن‌ها از موضوعات اساسی در ریاضیات مالی هستند. به‌طور کلی بهینه‌سازی به‌دست آوردن ماکسیمم یا مینیمم مقدار یک تابع است که گاهی اوقات تحت قیود خاصی می‌باشند. امروزه بهینه‌سازی نقش مهمی در ریاضی مالی ایفا می‌کند. یکی از اولین کاربردهای بهینه‌سازی در ریاضیات مالی ایجاد تعادل بین ریسک بازده و سید سرمایه‌گذاری است [۳]. معمولاً مسائل مختلفی که در ریاضیات مالی مطرح می‌شوند بعد از مدل‌سازی به یک مسأله بهینه‌سازی تبدیل می‌شوند که بستگی به نوع مدل حاصل می‌توان آن را با روش‌های بهینه‌سازی متناسب با آن حل کرد. مدل‌سازی مسائلی از قبیل تخصیص دارایی، مدیریت ریسک، قیمت‌گذاری و ... یک مسأله بهینه‌سازی می‌باشند. مسائل بهینه‌سازی حاصل از نوع مسأله برنامه‌ریزی خطی، درجه دوم، برنامه‌ریزی اعداد صحیح، برنامه‌ریزی پویا، برنامه‌ریزی غیرخطی می‌باشند. که برای حل این مسائل روش‌های بهینه‌سازی مختلفی وجود دارد [۲].



۲ برنامه‌ریزی درجه دوم

یک مسأله برنامه‌ریزی درجه دوم (QP) به صورت

$$\begin{cases} \min_{x \in \mathbb{R}^n} & x^T Q x + c^T x \\ & Ax = b, \\ & x \geq 0, \end{cases} \quad (2.1)$$

می‌باشد که در آن $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ و $Q \in \mathbb{R}^{n \times n}$ معلوم و $x \in \mathbb{R}^n$ مجهول است. در (۲.۱) تابع هدف درجه دوم و قیدها خطی هستند. اگر ماتریس Q معین مثبت باشد آن‌گاه مسأله برنامه‌ریزی درجه دوم (۲.۱) دارای جواب منحصری‌فرد است. با استفاده از شرایط کرون-کیون-تاکر (KKT) شرایط لازم بهینگی مسأله (۲.۱) به دست می‌آید.

قضیه ۱. فرض کنید x^* یک جواب بهینه موضعی برای مسأله برنامه‌ریزی درجه دوم باشد که در شرایط

$$Ax^* = b, \quad x^* \geq 0,$$

صدق می‌کند. همچنین Q یک ماتریس معین مثبت باشد. در این صورت بردارهای y^* و s^* وجود دارند به طوری که

$$A^T y^* - Q x^* + s^* = c,$$

$$s^* \geq 0,$$

$$x_i^* s_i^* = 0, \quad i = 1, \dots, n.$$

به علاوه x^* جواب بهینه سراسری برای مسأله QP است.

برای حل مسأله بهینه‌سازی QP روش‌های مختلفی از جمله روش‌های نقطه درونی، مسیر مرکزی، روش جهت‌های نیوتن و ... وجود دارد.

۳ مسأله میانگین-واریانس

در این بخش یک مسأله مهم در ریاضیات مالی را مدل‌سازی می‌کنیم که مدل حاصل یک برنامه‌ریزی درجه دوم است. مسأله QP حاصل را می‌توان با روش‌های مختلف حل کرد.

نظریه میانگین-واریانس مارکوویتز* یک ماکسیم مناسب برای انتخاب پرتفوی و اوراق بهادار ارائه می‌دهد که به بازده مورد انتظار دستیابی پیدا کند. فرض کنید $S_1, S_2, \dots, S_n$ ($n \geq 1$) متغیرهای تصادفی و μ_i و σ_i به ترتیب بازده مورد انتظار و انحراف معیار بازگشت

*Markowitz



دارایی S_i باشند. به علاوه ضریب همبستگی بازده از دارایی‌های S_i و S_j است. بردار

$$\mu = [\mu \quad \mu \quad \cdots \quad \mu_n]^T,$$

و ماتریس

$$\Sigma = \begin{cases} \sigma_i, & i = j, \\ \rho_{ij}\sigma_i\sigma_j, & i \neq j, \end{cases}$$

را در نظر بگیرید. اگر x_i کل مبلغ سرمایه‌گذاری شده از دارایی S_i باشد آنگاه

$$E(x) = x\mu + \cdots + x_n\mu_n,$$

و

$$Var(x) = \sum_{i,j} \rho_{ij}\sigma_i\sigma_j x_i x_j = x^T \Sigma x,$$

که $\rho_{ii} = 1$ است. چون واریانس همواره نامنفی است لذا $x^T \Sigma x \geq 0$ می‌باشد و Σ معین مثبت است. مجموعه اوراق بهادار قابل قبول یک مجموعه نا تهی به صورت زیر است

$$\mathcal{X} = \{x \mid Ax = b, Cx \geq d\},$$

که در آن $A \in \mathbb{R}^{m \times n}$, $C \in \mathbb{R}^{p \times n}$, $b \in \mathbb{R}^m$ و $d \in \mathbb{R}^p$ است. یکی از قیدهای مجموعه $\mathcal{X}$ به صورت

$$\sum_{i=1}^n x_i = R,$$

می‌باشد که آن را قید پرتفوی خطی می‌نامیم.

مدل‌سازی بهینه‌سازی میانگین-واریانس (WVO) یک مسأله بهینه‌سازی درجه دوم به صورت زیر ایجاد می‌کند

$$\begin{cases} \min_{x \in \mathbb{R}^n} & x^T \Sigma x \\ & \mu^T x \geq R, \\ & Ax = b. \end{cases} \quad (3.1)$$

قید اول در (3.1) نشان می‌دهد که بازده مورد انتظار از R کمتر نیست. مسأله بهینه‌سازی WVO محدب است و شرایط لازم و کافی برای بهینگی را دارد. با استفاده از شرایط KKT، x_R جواب بهینه است اگر و تنها اگر



$$\begin{cases} \Sigma x_R - \lambda_R \mu - A^T \gamma_E - C^T \gamma_I = , \\ \mu^T x_R \geq R, \quad Ax_R = b, \quad Cx_R \geq d, \\ \lambda_R \geq , \quad \lambda_R (\mu^T x_R - R) = , \\ \gamma_I \geq , \quad \gamma_I^T (Cx_R - d) = . \end{cases} \quad (3.2)$$

مسأله بهینه‌سازی WVO را می‌توان به صورت

$$\begin{cases} \max_{x \in \mathbb{R}^n} \mu^T x \\ x^T \Sigma x \leq \sigma, \\ Ax = b, \\ Cx = d, \end{cases} \quad (3.3)$$

یا

$$\begin{cases} \max_{x \in \mathbb{R}^n} \mu^T x - \frac{\delta}{2} x^T \Sigma x \\ Ax = b, \\ Cx \geq d, \end{cases} \quad (3.4)$$

بیان کرد.

۴ نتایج عددی

در این بخش مدل WVO را برای ساختن پورتفوی طولانی مدت سهام، اوراق قرضه و پول نقد برای یک مورد خاص در بازه زمانی مشخص را بررسی می‌کنیم [۱]. از داده‌های مربوط به سال‌های ۱۹۶۰ تا ۲۰۰۳ استفاده خواهیم کرد که اطلاعات مربوط به آن‌ها در جدول ۱ ارائه شده است.

فرض کنید

$$I_{i,t}, \quad i = , , , \quad t = , , \dots , T.$$

بازده کل باشد که به‌ازای $t = T$ بازده در سال ۱۹۶۰ و به‌ازای $t = T$ بازده در سال ۲۰۰۳ به‌دست می‌آیند. برای تبدیل بازده به نرخ بازده از فرمول زیر استفاده می‌کنیم

$$r_{i,t} = \frac{I_{i,t} - I_{i,t-}}{I_{i,t-}}. \quad (4.1)$$



جدول ۱: داده‌های مربوط به سهام، اوراق قرضه و پول نقد بین سال‌های ۱۹۶۰ تا ۲۰۰۳

سال	سهام	اوراق قرضه	پول نقد	سال	سهام	اوراق قرضه	پول نقد
۱۹۶۰	۲۰'۲۵۵۳	۲۶۲'۹۳۵	۱۰۰'۰۰۰	۱۹۸۲	۱۱۵'۳۰۸	۷۷۷'۳۳۲	۴۴۰'۶۸
۱۹۶۱	۲۵'۶۸۶۰	۲۶۸'۷۳۰	۱۰۲'۳۳	۱۹۸۳	۱۴۱'۳۱۶	۷۸۷'۳۵۷	۴۸۲'۴۲
۱۹۶۲	۲۳'۴۲۹۷	۲۸۴'۰۹۰	۱۰۵'۳۳	۱۹۸۴	۱۵۰'۱۸۱	۹۰۷'۷۱۲	۵۲۲'۸۴
۱۹۶۳	۲۸'۷۴۶۳	۲۸۹'۱۶۲	۱۰۸'۸۹	۱۹۸۵	۱۹۷'۸۲۹	۱۲۰۰'۶۳	۵۶۶'۰۸
۱۹۶۴	۳۳'۴۴۸۴	۲۹۹'۸۹۴	۱۱۳'۰۸	۱۹۸۶	۲۳۴'۷۵۵	۱۴۶۹'۴۵	۶۰۵'۲۰
۱۹۶۵	۳۷'۵۸۱۳	۳۰۲'۶۹۵	۱۱۷'۹۷	۱۹۸۷	۲۴۷'۰۸۰	۱۴۲۴'۹۱	۶۴۶'۱۷
۱۹۶۶	۳۳'۷۸۳۹	۳۱۸'۱۹۷	۱۲۴'۳۴	۱۹۸۸	۲۸۸'۱۱۶	۱۵۲۲'۴۰	۷۰۲'۷۷
۱۹۶۷	۴۱'۸۷۲۵	۳۰۹'۱۰۳	۱۲۹'۹۴	۱۹۸۹	۳۷۹'۴۰۹	۱۸۰۴'۶۳	۷۶۲'۱۶
۱۹۶۸	۴۶'۴۷۹۵	۳۱۶'۰۵۱	۱۳۷'۷۷	۱۹۹۰	۳۶۷'۶۳۶	۱۹۴۴'۲۵	۸۱۷'۸۷
۱۹۶۹	۴۲'۵۴۴۸	۲۹۸'۲۴۹	۱۵۰'۱۲	۱۹۹۱	۴۷۹'۶۳۳	۲۳۲۰'۶۴	۸۵۴'۱۰
۱۹۷۰	۴۴'۲۲۱۲	۳۵۴'۶۷۱	۱۵۷'۴۸	۱۹۹۲	۵۱۶'۱۷۸	۲۴۹۰'۹۷	۸۷۹'۰۴
۱۹۷۱	۵۰'۵۴۵۱	۳۹۴'۵۳۲	۱۶۴'۰۰	۱۹۹۳	۵۶۸'۲۰۲	۲۸۱۶'۴۰	۹۰۵'۰۶
۱۹۷۲	۶۰'۱۴۶۱	۴۰۳'۹۴۲	۱۷۲'۷۴	۱۹۹۴	۵۷۵'۷۰۵	۲۶۱۰'۱۲	۹۵۴'۳۹
۱۹۷۳	۵۱'۳۱۱۴	۴۱۷'۲۵۲	۱۸۹'۹۳	۱۹۹۵	۷۲۹'۰۴۲	۳۲۸۷'۲۷	۱۰۰۷'۸۴
۱۹۷۴	۳۷'۷۳۰۶	۴۳۳'۹۲۷	۲۰۶'۱۳	۱۹۹۶	۹۷۳'۸۹۷	۳۲۹۱'۵۸	۱۰۶۱'۱۵
۱۹۷۵	۵۱'۷۷۷۲	۴۵۷'۸۸۵	۲۱۶'۸۵	۱۹۹۷	۱۲۹۸'۸۲	۳۶۸۷'۳۳	۱۱۱۹'۵۱
۱۹۷۶	۶۴'۱۶۵۹	۵۲۹'۱۴۱	۲۲۶'۹۳	۱۹۹۸	۱۶۷۰'۰۱	۴۲۲۰'۲۴	۱۱۷۱'۹۱
۱۹۷۷	۵۹'۵۷۳۹	۵۳۱'۱۴۴	۲۴۱'۸۲	۱۹۹۹	۲۰۲۱'۴۰	۳۹۰۳'۳۲	۱۲۳۴'۰۲
۱۹۷۸	۶۳'۴۸۸۴	۵۲۴'۴۳۵	۲۶۶'۰۷	۲۰۰۰	۱۸۳۷'۳۶	۴۵۷۵'۳۳	۱۳۱۳'۰۰
۱۹۷۹	۷۵'۳۰۳۲	۵۳۱'۰۴۰	۳۰۲'۷۴	۲۰۰۱	۱۶۱۸'۹۸	۴۸۲۷'۲۶	۱۳۳۶'۸۹
۱۹۸۰	۹۹'۷۷۹۵	۵۱۷'۸۶۰	۳۵۹'۹۶	۲۰۰۲	۱۲۶۱'۱۸	۵۵۵۸'۴۰	۱۳۵۳'۴۷
۱۹۸۱	۹۴'۸۶۷۱	۵۳۸'۷۶۹	۴۰۴'۴۸	۲۰۰۳	۱۶۲۲'۹۴	۵۵۸۸'۱۹	۱۳۶۶'۷۳

با استفاده از (۴۰۱) داده‌های مربوط به جدول ۱ را به داده‌های جدول ۲ تبدیل می‌کنیم.



جدول ۲: داده‌های مربوط به سهام، اوراق قرضه و پول نقد بین سال‌های ۱۹۶۰ تا ۲۰۰۳ براساس نرخ بازده

سال	سهام	اوراق قرضه	پول نقد	سال	سهام	اوراق قرضه	پول نقد
۱۹۶۱	۲۶'۸۱	۲'۲۰	۲'۳۳	۱۹۸۳	۲۲'۵۶	۱'۲۹	۹'۴۷
۱۹۶۲	-۸'۷۸	۵'۷۲	۲'۹۳	۱۹۸۴	۶'۲۷	۱۵'۲۹	۸'۳۷
۱۹۶۳	۲۲'۶۹	۱'۷۹	۳'۳۸	۱۹۸۵	۳۱'۱۷	۳۲'۲۷	۸'۲۷
۱۹۶۴	۱۶'۳۶	۳'۷۱	۳'۸۵	۱۹۸۶	۱۸'۶۷	۲۲'۳۹	۶'۹۱
۱۹۶۵	۱۲'۳۶	۰'۹۳	۴'۳۲	۱۹۸۷	۵'۲۵	-۳'۰۳	۶'۷۷
۱۹۶۶	-۱۰'۱۰	۵'۱۲	۵'۴۰	۱۹۸۸	۱۶'۶۱	۶'۸۴	۸'۷۶
۱۹۶۷	۲۳'۹۴	-۲'۸۶	۴'۵۱	۱۹۸۹	۳۱'۶۹	۱۸'۵۴	۸'۴۵
۱۹۶۸	۱۱'۰۰	۲'۲۵	۶'۰۲	۱۹۹۰	-۳'۱۰	۷'۷۴	۷'۳۱
۱۹۶۹	-۸'۴۷	-۵'۶۳	۸'۹۷	۱۹۹۱	۳۰'۴۶	۱۹'۳۶	۴'۴۳
۱۹۷۰	۳'۹۴	۱۸'۹۲	۴'۹۰	۱۹۹۲	۷'۶۲	۷'۳۴	۲'۹۲
۱۹۷۱	۱۴'۳۰	۱۱'۲۴	۴'۱۴	۱۹۹۳	۱۰'۰۸	۱۳'۰۶	۲'۹۶
۱۹۷۲	۱۸'۹۹	۲'۳۹	۵'۳۳	۱۹۹۴	۱'۳۲	-۷'۳۲	۵'۴۵
۱۹۷۳	-۱۴'۶۹	۳'۲۹	۹'۹۵	۱۹۹۵	۳۷'۵۸	۲۵'۹۴	۵'۶۰
۱۹۷۴	-۲۶'۴۷	۴'۰۰	۸'۵۳	۱۹۹۶	۲۲'۹۶	۰'۱۳	۵'۲۹
۱۹۷۵	۳۷'۲۳	۵'۵۲	۵'۲۰	۱۹۹۷	۳۳'۳۶	۱۲'۰۲	۵'۵۰
۱۹۷۶	۲۳'۹۳	۱۵'۵۶	۴'۶۵	۱۹۹۸	۲۸'۵۸	۱۴'۴۵	۴'۶۸
۱۹۷۷	-۷'۱۶	۰'۳۸	۶'۵۶	۱۹۹۹	۲۱'۰۴	-۷'۵۱	۵'۳۰
۱۹۷۸	۶'۵۷	-۱'۲۶	۱۰'۰۳	۲۰۰۰	-۹'۱۰	۱۷'۲۲	۶'۴۰
۱۹۷۹	۱۸'۶۱	-۱'۲۶	۱۳'۷۸	۲۰۰۱	-۱۱'۸۹	۵'۵۱	۱'۸۲
۱۹۸۰	۳۲'۵۰	-۲'۴۸	۱۸'۹۰	۲۰۰۲	-۲۲'۱۰	۱۵'۱۵	۱'۲۴
۱۹۸۱	-۴'۹۲	۴'۰۴	۱۲'۳۷	۲۰۰۳	۲۸'۶۸	۰'۵۴	۰'۹۸
۱۹۸۲	۲۱'۵۵	۴۴'۲۸	۸'۹۵				

اکنون فرض کنیم R_i نرخ بازگشت تصادفی برای دارایی S_i باشد. میانگین حسابی نرخ بازگشت از رابطه زیر محاسبه می‌شود

$$\bar{r}_i = \frac{1}{T} \sum_{t=1}^T r_{i,t}.$$



میانگین حسابی نرخ بازگشت برای سهام، اوراق قرضه و پول نقد در جدول ۳ آمده است.

جدول ۳: میانگین حسابی نرخ بازگشت برای سهام، اوراق قرضه و پول نقد

پول نقد	اوراق قرضه	سهام	
۶۰۳۲	۷۰۸۵	۱۲۰۰۶	$\bar{r}_i$

به دلیل چندگانه بودن نرخ بازگشت در طول زمان به جای میانگین حسابی از میانگین هندسی زیر استفاده می‌کنیم

$$\mu_i = \left(\Pi_{t=1}^T (1 + r_{i,t}) \right)^{\frac{1}{T}} - 1.$$

مقادیر میانگین هندسی در جدول ۴ محاسبه شده است.

جدول ۴: میانگین هندسی نرخ بازگشت برای سهام، اوراق قرضه و پول نقد

پول نقد	اوراق قرضه	سهام	
۶۰۲۷	۷۰۳۷	۱۰۰۷۳	μ_i

درایه‌های ماتریس کوواریانس از رابطه زیر محاسبه می‌شود

$$Cov(R_i, R_j) = \frac{1}{T} \sum_{t=1}^T (r_{i,t} - \bar{r}_i)(r_{j,t} - \bar{r}_j).$$

جدول ۵: ماتریس کوواریانس

پول نقد	اوراق قرضه	سهام	کوواریانس
۰۰۰۰۰۲۱	۰۰۰۰۳۸۷	۰۰۰۲۷۷۸	سهام
-۰۰۰۰۰۳۰	۰۰۰۱۱۱۲	۰۰۰۰۳۸۷	اوراق قرضه
۰۰۰۰۱۱۵	-۰۰۰۰۰۲۰	۰۰۰۰۰۲۱	پول نقد

نوسان‌های نرخ بازگشت از رابطه $\sigma_i = \sqrt{Cov(R_i, R_i)}$ به دست می‌آید که در جدول ۶ ارائه شده است.

درایه‌های ماتریس همبستگی نیز از رابطه زیر پیدا می‌شود که نتایج حاصل در جدول ۷ بیان شده است.



جدول ۶: نوسان‌های نرخ بازگشت

	سهام	اوراق قرضه	پول نقد
نوسان	۱۶۰۶۷	۱۰۰۵۵	۳۰۴۰

جدول ۷: ماتریس همبستگی

همبستگی	سهام	اوراق قرضه	پول نقد
سهام	۱	۰.۲۱۹۹	۰.۰۳۶۶
اوراق قرضه	۰.۲۱۹۹	۱	-۰.۰۵۴۵
پول نقد	۰.۰۳۶۶	-۰.۰۵۴۵	۱

$$\rho_{ij} = \frac{Cov(R_i, R_j)}{\sigma_i \sigma_j}.$$

بنابراین مسأله بهینه‌سازی درجه دوم WVO برای این مثال به صورت زیر حاصل می‌شود

$$\left\{ \begin{array}{l} \min_{x \in \mathbb{R}^n} x_S + x_S x_B + x_S x_M + x_B \\ -x_B x_M + x_M \\ x_S + x_B + x_M \leq R, \\ x_S + x_B + x_M = , \\ x_S, x_B, x_M \geq , \end{array} \right. \quad (۴۰۲)$$

این مسأله درجه دوم را می‌توان با روش‌های مختلف حل و جواب بهینه مسأله QP را به دست آورد. جواب‌های بهینه برای سهام، اوراق قرضه و پول نقد به ازای مقادیر مختلف R در جدول ۸ آمده است.

مراجع

- [1] G. Cornuejols, R. Tutuncu, *Optimization methods in finance*, Carnegie Mellon University, Pittsburgh, PA 15213 USA, (2005).



جدول ۸: جواب‌های بهینه برای سهام، اوراق قرضه و پول نقد به‌ازای مقادیر مختلف R

R	پول نقد	اوراق قرضه	سهام	واریانس
۰.۰۶۵	۰.۸۷	۰.۱۰	۰.۰۳	۰.۰۰۱۰
۰.۰۷۰	۰.۷۵	۰.۱۲	۰.۱۳	۰.۰۰۱۴
۰.۰۷۵	۰.۶۲	۰.۱۴	۰.۲۴	۰.۰۰۲۶
۰.۰۸۰	۰.۴۹	۰.۱۶	۰.۳۵	۰.۰۰۴۴
۰.۰۸۵	۰.۳۷	۰.۱۸	۰.۴۵	۰.۰۰۷۰
۰.۰۹۰	۰.۲۴	۰.۲۰	۰.۵۶	۰.۰۱۰۲
۰.۰۹۵	۰.۱۱	۰.۲۲	۰.۶۷	۰.۰۱۴۲
۰.۱۰۰	۰	۰.۲۲	۰.۷۸	۰.۰۱۸۹
۰.۱۰۵	۰	۰.۰۷	۰.۹۳	۰.۰۲۴۶

[2] J. Nocedal, S. J. Wright, *Numerical Optimization*, Springer, NewYork, (2006).

[3] A. D. Sezer, G. W. Weber, *Optimization methods in mathematical finance*, Optimization, **62(11)** (2003), 139–1402.

پست الکترونیکی: rahpeyma_83@yahoo.com



حل و استخراج معادلات دیفرانسیل مشتقات جزئی داده‌محور از دیدگاه یادگیری ماشین

مهدی جلیلی[†] و رضوان صالحی[‡]

دانشگاه تربیت مدرس[†]

دانشگاه تربیت مدرس[‡]

چکیده

در این مقاله با استفاده از روش‌های بدون شبکه و تطبیق آن روش‌ها با فرآیندهای یادگیری ماشین، به شناسایی و حل سیستم دینامیکی و رفتار حالات معادلات بر اساس داده‌های دنیای واقعی پرداخته شده‌است. با توجه به دو مثال خطی و غیرخطی حل شده نتایج قابل قبولی بدست آوردیم. ابتدا از روی نمونه‌ها به شناسایی و تقریب پارامترهای حالت معادلات پرداختیم سپس با استفاده از روش بدون شبکه کارآمد جواب معادلات را تقریب زدیم. واژه‌های کلیدی: یادگیری ماشین - روش‌های بدون شبکه - معادلات دیفرانسیلی با مشتقات جزئی - فرآیندهای گاوسی. طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

۲ مقدمه

در دهه‌ی اخیر انگیزه برای استفاده از تکنولوژی‌های کامپیوتری و هوش مصنوعی در اکثر زمینه‌های صنعت، علوم مهندسی، و پزشکی به صورت جدی افزایش یافته است. مسبب این عمل افزایش قدرت محاسباتی سیستم‌ها و افزایش روزمره منابع اطلاعاتی و مهم‌تر از آن علم هوش مصنوعی^۱ است که با جایگزین کردن عملکرد مغز انسان با یک مرکز محاسبه‌کننده سعی در کم کردن خطا و سرعت بخشیدن به فرآیند محاسبه دارد.

^۱Artificial Intelligence



یکی از شاخه‌های مهم و پرکاربرد هوش مصنوعی، یادگیری ماشین نظارت‌شده^۲ می‌باشد که سرعت خارق العاده‌ای در شبیه‌سازی روش محاسباتی مغز انسان دارد. این شاخه ترکیب انعطاف‌پذیری از روش‌های عددی با رویکرد بهینه‌سازی با تحلیل آماری است که برای اکثر مسائلی که با داده‌های معلوم سروکار دارد، بهترین گزینه می‌باشد.

در این مقاله ما از روش‌های بدون شبکه^۳ که در دهه‌ی اخیر جایگزین اکثر روش‌های وابسته به شبکه شده و طیف گسترده‌ای از روش‌های کارآمد برای مسائلی همچون درونیابی چندبعدی، تقریب و حل معادلات دیفرانسیلی (معمولی، جزئی و تصادفی) ارائه داده، استفاده کردیم.

یکی از بزرگترین چالش‌ها برای دانشمندان و محققان در زمینه‌های علوم و مهندسی شناسایی حالت کلی یا رفتار پدیده‌های طبیعی در زندگی روزمره و در آزمایشگاه بر اساس داده‌های ارزشمندی که از آن پدیده بدست آورده شده، می‌باشد. روش‌های متعددی برای انجام این کار ارائه شده که می‌توان به مقالات شناسایی سیستم داده‌محور [۱]، رگرسیون غیرخطی [۲] و آنالیز طیفی لاپلاسین غیرخطی [۳] اشاره کرد.

در اکثر روش‌های داده‌محور یادگیری ماشین، اگر همه‌ی داده‌ها مورد استفاده قرار گیرد نتایج بهتری به دست می‌آید ولی مشکلاتی مانند هزینه محاسباتی اعم از زمان محاسبه طولانی، سیستم محاسباتی قوی و کمبود داده‌ی کافی رخ می‌دهد. لذا روش‌هایی که با استفاده از داده‌ی کمتر، نتایج بهتری به دست می‌آورند، با ارزش‌اند. در این مقاله سعی کردیم قالبی معرفی کنیم که با استفاده از داده‌ی کم معادله را شناسایی کرده و تقریبی برای آن ارائه دهد.

۳ مسئله اصلی

فرض کنیم فرم کلی یک معادله دیفرانسیلی با مشتقات جزئی غیرخطی در فضای n - بعدی به صورت زیر باشد:

$$\begin{cases} \frac{\partial u}{\partial t} + A_{x;\theta} u = 0, & \text{in } \Omega \times [0, T], \\ B_x u = 0, & \text{on } \partial\Omega \times [0, T], \end{cases} \quad (3,1)$$

که در آن $u(x, t)$ جواب معادله، $A_{x;\theta}$ عملگر غیرخطی داخل ناحیه وابسته به پارامترهای $\theta = (\theta_1, \dots, \theta_m)$ ، پارامترهای مجهول معادله، B_x عملگر مرزی مسئله و

$$x = (x_1, \dots, x_n) \in \Omega,$$

نقاط تشکیل دهنده ناحیه است.



برای مثال اگر معادله‌ی زیر را در نظر بگیریم:

$$\begin{cases} \frac{\partial u}{\partial t} + \theta_1 \Delta u - \theta_2 \nabla \cdot u = 0, & \text{in } [0, 1]^2 \times [0, T], \\ u = 0, & \text{on } \partial[0, 1]^2 \times [0, T], \end{cases}$$

آنگاه داریم $\theta = (\theta_1, \theta_2)$ و $\mathcal{B}_x = \mathcal{I}$ ، $\mathcal{A}_{x;\theta} = \theta_1 \Delta - \theta_2 \nabla \cdot [\cdot]$

قضیه ۱. فرض کنیم معادله (۳، ۱) به ازای پارامترها و عملگرهای مناسب موجود باشد. آنگاه جواب یکتای $u \in \mathcal{H}^m(\mathcal{D})$ تقریباً همه‌جا^۴ موجود است [۴]، که در آن $\mathcal{H}^m(\mathcal{D})$ فضای سوپولوف کلاسیک (L_2) از مرتبه $m > \mathcal{O}(\mathcal{A}_{x;\theta}) + (\text{card}\{\Omega\})/2$ است و منظور از $\mathcal{O}(\mathcal{A}_{x;\theta})$ مرتبه‌ی عملگر است.

۴ روش هم‌مکانی مبتنی بر هسته بازتولیدکننده برای فرآیند گاوسی

فرض کنید فضای تابعی $\mathcal{H}_K(\Omega)$ فضای هیلبرت هسته بازتولیدکننده با نرم $\|\cdot\|_{K,\Omega}$ باشد و هسته‌ی بازتولیدکننده‌اش $K \in C(\bar{\Omega}, \bar{\Omega})$ متقارن و معین مثبت باشد. برای داده‌های معلوم $\{y_i\}_{i=1}^N \subset \mathbb{R}$ در نقاط هم‌مکانی $X_\Omega := \{\mathbf{x}_i\}_{i=1}^N \subset \Omega$ تابع $u \in \mathcal{H}_K(\Omega)$ موجود است که در شرط

$$y_i = u(\mathbf{x}_i), \quad \mathbf{x}_i = (x_{1,i}, \dots, x_{d,i}) \in \Omega \subset \mathbb{R}^d, \quad i = 1, \dots, N.$$

صدق می‌کند. هدف پیدا کردن تابع تقریب بهینه در $\mathcal{H}_K(\Omega)$ است که همه نقاط را درونیایی می‌کند. طبق خواص فضای $\mathcal{H}_K(\Omega)$ می‌توان نشان داد که $\mathcal{H}_K(\Omega) \equiv \mathcal{H}^m(\Omega)$.

تعریف ۲. فرآیند تصادفی $S : \Omega \times \Gamma \rightarrow \mathbb{R}$ فرآیند گاوسی با میانگین $\mu : \Omega \rightarrow \mathbb{R}$ و هسته کواریانس $\Phi : \Omega \times \Omega \rightarrow \mathbb{R}$ بروی فضای احتمال $(\Gamma, \mathcal{F}, \mathbb{P})$ نامیده می‌شود اگر برای هر مجموعه از نقاط دوبه‌دو متمایز $X_\Omega := (\mathbf{x}_1, \dots, \mathbf{x}_N) \subset \Omega$ بردار تصادفی $S := (S_{\mathbf{x}_1}, \dots, S_{\mathbf{x}_N})^T$ یک متغیر تصادفی چند-نرمال^۵ روی $(\Gamma, \mathcal{F}, \mathbb{P})$ با میانگین μ و ماتریس کواریانس Φ باشد. یعنی $S \sim \mathcal{N}(\mu, \Phi)$ که $\Phi = (\Phi(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1}^{N,N}$ و $\mu := (\mu(\mathbf{x}_1), \dots, \mu(\mathbf{x}_N))^T$.

۵ تقابل دو دیدگاه تقریب و آماری

برای به‌دست آوردن بهترین تقریب در دیدگاه هم‌مکانی از درونیایی مبتنی بر هسته، یک مسئله بهینه‌سازی به صورت زیر حل می‌شود:

$$\hat{u}_K = \underset{u \in \mathcal{H}_K(\Omega)}{\operatorname{argmin}} \{ \|u\|_{K,\Omega} \text{ s.t. } u(\mathbf{x}_i) = y_i, i = 1, \dots, N \}.$$

Almost Surely^{*}
Multi-Normal^۵



در این دیدگاه، جواب هم‌مکانی $\hat{u}_K(x)$ ترکیب خطی از انتقال‌های هسته‌ی بازتولیدکننده K است. یعنی

$$\hat{u}_K(x) = \sum_{j=1}^N c_j K(x, x_j), \quad x \in \Omega, \quad (5,1)$$

که ضرایب مجهول c با حل دستگاه

$$\mathcal{K}c = (K(x_i, x_j))_{i,j=1}^{N,N} c = y \quad (5,2)$$

که

$$c := (c_1, \dots, c_N)^T, \quad y := (y_1, \dots, y_N)^T$$

به‌دست می‌آیند.

در دیدگاه آماری برای تقریب توابع از روش کرایجینگ^۶ استفاده می‌شود که بدین صورت تعریف می‌شود که، فرض کنید S یک فرآیند گاوسی با میانگین صفر و هسته‌ی کواریانس K بروی فضای احتمال $(\Gamma, \mathcal{F}, \mathbb{P})$ باشد. کرایجینگ براساس این فرض مدلسازی شده است که u یک تحقق^۷ از میدان گاوسی S می‌باشد. مقادیر $y_1, \dots, y_N$ تحقق متغیرهای تصادفی $S_{x_1}, \dots, S_{x_N}$ هستند. تقریب (پیش‌بینی‌کننده) بدون بایاس^۸ بهینه‌ی S_x مبتنی بر S معادل است با

$$\hat{U}_x := \sum_{j=1}^N c_j(x) S_{x_j} = \underset{U \in \text{span}\{S_{x_k}\}_{k=1}^N}{\operatorname{argmin}} \mathbb{E}|U - S_x|^2,$$

که ضرایب $c(x) := (c_1(x), \dots, c_N(x))^T$ توسط

$$c(x) = \mathcal{K}^{-1}k(x),$$

به‌دست می‌آیند، که در آن $k(x) := (K(x, x_1), \dots, K(x, x_N))^T$ می‌باشد. از طرفی داریم:

$$\mathbb{E}(\hat{U}_x | S_{x_1} = y_1, \dots, S_{x_N} = y_N) = \hat{u}_K(x).$$

۶ روش محاسبه

برای حل و شناسایی (۳، ۱) از میدان گاوسی معرفی شده و روش کرایجینگ (رگرسیون فرآیند گاوسی) [۵] استفاده کردیم.

برای اینکه داده‌های دنیای واقعی داشته باشیم نوفه هم وارد محاسبات می‌شود لذا طبق توزیع نرمال داریم:

$$\begin{bmatrix} y \\ \tilde{y} \end{bmatrix} \sim \mathcal{N} \left(\circ, \begin{bmatrix} K(x, x) + \sigma_n^2 I & K(x, x_*) \\ K(x_*, x) & K(x_*, x_*) \end{bmatrix} \right) \quad (6,1)$$

Kriging^۶

Realization^۷

Unbiased Predictor^۸

با استفاده از روابط توزیع شرطی داریم:

$$\tilde{y}|x, y, x_* \sim \mathcal{N}(\mu(\tilde{y}), \text{cov}(\tilde{y})), \quad \text{که} \quad (6,2)$$

$$\mu(\tilde{y}) := \mathbb{E}[\tilde{y}|x, y, x_*] = K(x_*, x)[K(x, x) + \sigma_n^2 I]^{-1} y, \quad (6,3)$$

$$\text{cov}(\tilde{y}) = K(x_*, x_*) - K(x_*, x)[K(x, x) + \sigma_n^2 I]^{-1} K(x, x_*), \quad (6,4)$$

که $\tilde{y}$ پیش‌بینی (تقریب) است.

پس برای حل و شناسایی معادله (3,1) ابتدا زمان را براساس روش‌های تفاضل متناهی گسسته‌سازی می‌کنیم سپس با تنظیم پیشین

۹

$$u^{m-1} \sim \mathcal{N}(\circ, \Sigma^{m-1, m-1}),$$

و محاسبه

$$\begin{bmatrix} u^{m-1} \\ u^m \end{bmatrix} \sim \mathcal{N}\left(\circ, \begin{bmatrix} \Sigma^{m-1, m-1} & \Sigma^{m-1, m} \\ \Sigma^{m, m-1} & \Sigma^{m, m} \end{bmatrix}\right) \quad (6,5)$$

که توابع کواریانس برای معادله (3,1) به صورت زیر محاسبه می‌شوند:

$$\Sigma^{m-1, m-1} := \mathcal{K}|_{t_{m-1}, t_{m-1}},$$

$$\Sigma^{m, m-1} := \mathcal{A}_{x_*} \mathcal{B}_x \Sigma^{m-1, m-1},$$

$$\Sigma^{m-1, m} := \mathcal{A}_x \mathcal{B}_{x_*} \Sigma^{m-1, m-1},$$

$$\Sigma^{m, m} := \mathcal{A}_x \mathcal{A}_{x_*} \Sigma^{m-1, m-1},$$

که در آن $t_{m-1} = (m-1) * \Delta t$. حال طبق قانون بیزی در احتمال می‌توانیم پسین $^{\circ}$ را برحسب پیشین محاسبه شده به صورت

$$p(u^m | \theta, \Omega) = \frac{p(\theta | u^{m-1}, \Omega) p(u^{m-1} | \Omega)}{p(\theta)}$$

با اجرای همین روش برای پیدا کردن ضرایب گام اول محاسبه انجام می‌شود. سپس با محاسبه احتمال شرطی $p(u^m | \theta, \Omega)$ جواب، تقریب به دست می‌آید.

Prior^۹
Posterior^{۱۰}



حال برای کمینه کردن خطای محاسبه از تابع چگالی توزیع نرمال بهره می‌گیریم و برای داشتن بخش‌های موثر از کل تابع چگالی لگاریتم می‌گیریم:

$$\log p(u|\theta, \Omega) = \underbrace{-\frac{1}{2}u^T(Q + \sigma_n^2 I)^{-1}u}_1 - \underbrace{\frac{1}{2}\log|Q + \sigma_n^2 I|}_2 - \underbrace{\frac{N}{2}\log(2\pi)}_3,$$

که در آن $u = \begin{bmatrix} u^{m-1} \\ u^m \end{bmatrix}$ و Q ماتریس کواریانس معرفی شده در (۶،۵) است و قسمت ۱ عبارت فوق برازش^{۱۱} داده‌ها، قسمت ۲ منظم‌سازی^{۱۲} و قسمت ۳ نرمال‌سازی^{۱۳} را انجام می‌دهد.

۷ نتایج عددی

برای آزمایش روش معرفی شده از دو معادله‌ی خطی و غیرخطی استفاده کردیم. برای ساخت توابع کواریانس همه‌ی توابع پایه‌ای شعاعی قابل اعمال هستند. ما در این کار از تابع

$$\Phi(x, x_*) = \exp\left(-\frac{(x - x_*)^2}{2\rho}\right)$$

استفاده کردیم که ρ پارامتر کنترل مقیاس می‌باشد.

در همه‌ی مثال‌های عددی، ابتدا ضرایب مجهول مسئله را محاسبه کردیم. سپس با استفاده از همان ضرایب تقریب جواب مسئله را با محاسبه احتمال شرطی تعریف شده به‌دست می‌آوریم. با مقایسه جواب تقریبی با داده‌ی اصلی خطای محاسبات را نشان داده‌ایم.

مثال ۳. معادله‌ی خطی ۱-بعدی انتقال-انتشار^{۱۴}

$$\frac{\partial u}{\partial t} + \gamma \frac{\partial u}{\partial x} = k \frac{\partial^2 u}{\partial x^2}, \quad \text{in } [0, 1] \times [0, 1],$$

را در نظر بگیرید. جواب دقیق عبارت‌است از

$$u(x, t) = \frac{0.25}{\sqrt{0.00625 + 0.02t}} \exp\left(-\frac{(x + 0.5 - t)^2}{0.0125 + 0.04t}\right),$$

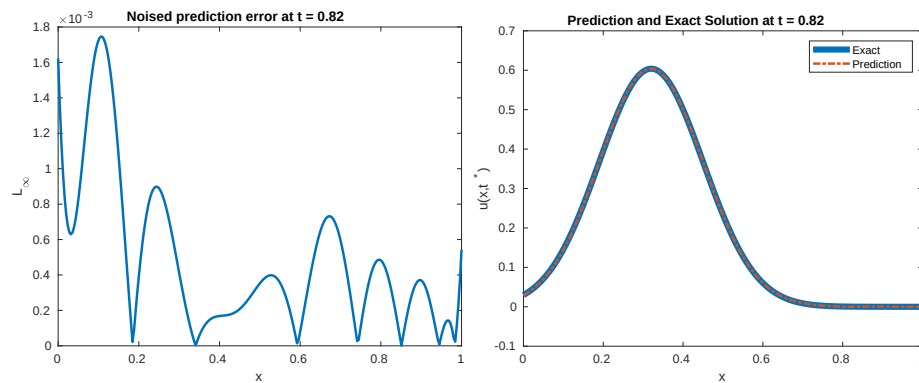
شرایط مرزی مسئله دیریکله در نظر گرفته شده است. برای ساخت داده‌ی مورد نیاز با استفاده از جواب دقیق، پارامترهای انتقال و انتشار $k = 0.1$ و $\gamma = 1$ به ترتیب در نظر می‌گیریم.

^{۱۱}Fitting

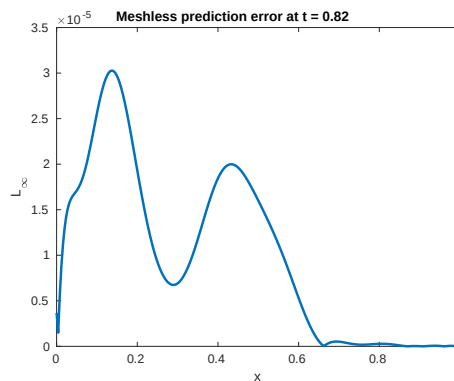
^{۱۲}Regularization

^{۱۳}Normalizing

^{۱۴}Advection-Diffusion



شکل ۱: (راست) مقایسه تقریب به دست آمده با نوفه ۱ درصد و جواب دقیق در $t = ۰.۸۲$ (چپ) خطای متناظر با تقریب بدست آمده



شکل ۲: خطای تقریب بدون نوفه (جواب دیدگاه بدون شبکه) در $t = ۰.۸۲$

در شکل ۱ خطای تقریب مدل براساس نوفه‌ی وارد شده را نمایش دادیم. در واقع پس از محاسبه پارامترهای مسئله که در جدول؟؟ ارائه شده و محاسبه جواب تقریبی براساس پارامترهای نوفه‌دار، خطای L_∞ را بین داده‌ی دقیق و تقریب را نمایش داده‌ایم. برای دیدگاه بدون شبکه هم همان تقریب را بدون نوفه محاسبه کردیم و خطای محاسبه را در شکل ۲ مصور کرده‌ایم.

	$\gamma = ۱$	$k = ۰.۰۱$
بدون نوفه	۱/۰۰۷۳	۰/۰۰۵۲
نوفه ۱ درصد	۰/۹۹۷۶	۰/۰۰۳۹

جدول ۱: پارامترهای بدست آمده

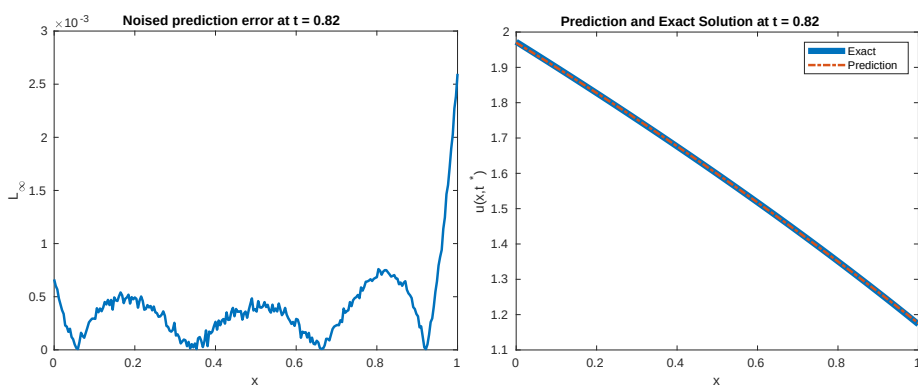
مثال ۴. حال، معادله ی ۱-بعدی غیرخطی هاری-دیم^{۱۵}

$$\frac{\partial u}{\partial t} = hu^3 \frac{\partial^3 u}{\partial x^3}, \quad \text{in } [0, 1] \times [0, 1],$$

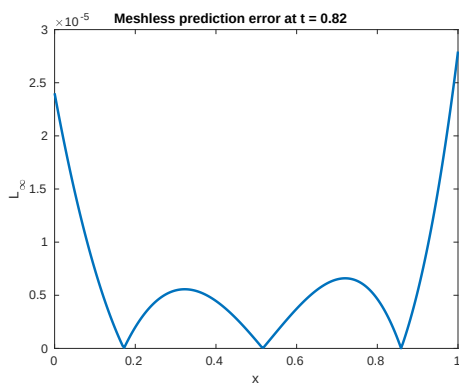
را در نظر بگیرید. جواب دقیق این معادله عبارت است از

$$u(x, t) = \left(4 - \frac{3}{4}(x+t)\right)^{\frac{4}{3}},$$

همانند مثال خطی، برای ساختن داده ی دقیق پارامتر مجهول را $h = 1$ در نظر می گیریم.



شکل ۳: مقایسه تقریب به دست آمده با نوفه ۱ درصد و جواب دقیق در $t = 0.82$



شکل ۴: خطای تقریب بدون نوفه (جواب دیدگاه بدون شبکه) در $t = 0.82$



$h = 1$	
بدون نوفه	۱/۰۵۹۱
نوفه ۱ درصد	۱/۰۷۰۲

جدول ۲: پارامترهای بدست آمده

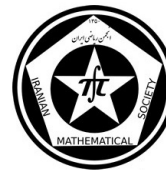
مراجع

- [1] J.P. Crutchfield, B.S. McNamara, Equations of motion from a data series, Complex Syst. 1 (1987).
- [2] H.U. Voss, P. Kolodner, M. Abel, J. Kurths, Amplitude equations from spatiotemporal binary-fluid convection data, Phys. Rev. Lett. 83 (1999) 3422-3425.
- [3] D. Giannakis, A.J. Majda, Nonlinear laplacian spectral analysis for time series with intermittency and low-frequency variability, Proc. Natl. Acad. Sci. 109 (2012) 2222-2227.
- [4] P.L. Chow: Stochastic Partial Differential Equations, Chapman Hall/CRC, Taylor Francis, Boca Raton (2007)
- [5] C.E. Rasmussen, C.K. I. Williams, Gaussian Processes for Machine Learning Adaptive Computation and Machine Learning, The MIT Press, (2005)
- [6] I. Cialenco, G. E. Fasshauer, Q. Ye, Approximation of stochastic partial differential equations by a kernel-based collocation method, International Journal of Computer Mathematics, (2012), 2543-2561
- [7] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
- [8] M. Raissi, P. Perdikaris, G.E. Karniadakis, Numerical gaussian processes for time-dependent and non-linear partial differential equations, arXiv preprint, arXiv:1703.10230, 2017.
- [9] G.E. Fasshauer, Q. Ye, Kernel-based collocation methods versus Galerkin finite element methods for approximating elliptic stochastic partial differential equations, Meshfree Methods for Partial Differential Equations VI, Springer (2013) 155-170.



پست الکترونیکی: dan7h34u@gmail.com

پست الکترونیکی: r.salehi@modares.ac.ir



روش بلوکی چندگامی برای حل عددی معادلات دیفرانسیل تأخیری درجه دوم با شرایط مرزی

محمود دادخواه[†]

دانشکده علوم پایه - گروه ریاضی، دانشگاه پیام نور تهران، ایران[†]

چکیده

در این مقاله، یک روش بلوکی چندگامی مرتبه ۵ برای حل عددی معادلات دیفرانسیل تأخیری درجه دوم با شرایط مرزی و با طول گام ثابت، ارائه شده است. این روش، در هر گام، جوابهای عددی را بطور همزمان برای دو نقطه محاسبه می نماید. تأخیرهای در نظر گرفته شده شامل سه نوع ثابت، پانتوگراف و وابسته به زمان هستند. تجزیه و تحلیل مدل بر اساس مرتبه مدل، سازگاری و همگرایی روش بیان شده، بطور کامل مورد بررسی قرار گرفته است. با ارائه چند مثال، کارایی و دقت روش بیان شده در مقایسه با روشهای موجود مورد ارزیابی قرار گرفته است.

واژه‌های کلیدی: معادلات دیفرانسیل تأخیری، روش بلوکی، مسائل مقدار مرزی، روشهای چندگامی خطی.

طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

تاخیر غالباً در دستگاههای بیولوژیکی، شیمیایی، فیزیکی، حمل و نقل، ارتباطات و ... رخ می دهد. لذا دستگاههای تأخیر زمانی، کلاس مهمی از دستگاههای دینامیکی هستند که بررسی جوابهای تحلیلی و بخصوص روشهای عددی حل این نوع معادلات در سالهای اخیر، مورد توجه ویژه ی بسیاری از پژوهشگران بوده است. [۸]

در این مقاله فرم کلی یک معادله دیفرانسیل تأخیری مرتبه دوم را بصورت زیر در نظر می گیریم:

$$x'' = f(t, x(t), x'(t), x(t - \tau), x'(t - \tau)), \quad t \in [a, b] \quad (1.1)$$

$$x(t) = \theta(t), \quad t \in [a - \tau, a], \quad \tau \in \mathbb{R}^+ \quad (1.2)$$

شرایط مرزی را نیز به شکل زیر فرض می کنیم:

$$x(a) = \alpha, \quad x(b) = \beta \quad (13)$$

که $\alpha = \theta(a)$ و تأخیر τ ، مقداری مثبت برای تأخیرهای ثابت $(\tau = mh, m = 1, 2, 3, \dots)$ و یا تابعی از t ($\tau = \tau(t)$) برای تأخیرهای پانتوگراف یا وابسته زمانی خواهد بود. همچنین $\theta(t)$ تابع اولیه و هموار داده شده است. توجه داریم که شکل کلی معادلات پانتوگراف را می توان به فرم زیر بیان کرد:

$$x'' = f(t, x(t), x'(t), x(qt), x'(qt)), \quad t \in [a, b], \quad 0 < q < 1. \quad (14)$$

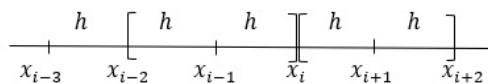
مطالعه بر روی حل عددی معادلات دیفرانسیل تأخیری درجه دوم با تأخیرهای ثابت و با شرایط مرزی، نخستین بار در [۱۱] و به کمک روش اولر و تکنیک پرتابی انجام شد. بعد از آن روشهای عددی متنوع دیگری در همین حوزه مورد بررسی قرار گرفت که می توان به روش تفاضلات متناهی، روش اسپلاین مکعبی، روش تفاضل نقطه میانی، روش هم مکانی و روش برونابی ریچاردسون اشاره کرد [۱۳، ۱۲، ۵، ۲، ۱]. با وجود این، تنها در چند مطالعه، روشهای عددی حل عددی معادلات دیفرانسیل تأخیری درجه دوم با تأخیرهای غیر ثابت و شرایط مرزی مورد بررسی قرار گرفته اند. مثلاً در [۱] روشی همگرا از مرتبه سه با اعمال برونابی ریچاردسون بدست آمده است. در [۴] به کمک روش انتگرال ضربی اتکینسون، مسایل تأخیری وابسته به زمان حل شده است. در [۳] به کمک ترکیبی از دنباله پیکارد، قانون دوزنقه ای و درونابی اسپلاین، روشی برای حل مسایل تأخیری پنتوگراف ارایه شده است و در [۱۶]، به کمک روشهای تحلیلی، تجزیه آدومیان و حساب تغییرات، مسایل گوناگون خطی و غیر خطی پنتوگراف مورد بررسی و تحلیل قرار گرفته اند. همچنین روشهای مستقیم پیشنهادی در [۱۴] و [۱۷]، با استفاده از تکنیک تصحیح کننده ی پیشگو برای محاسبه جواب تقریبی یک نقطه ای در هر تکرار، اجرا شده اند. استفاده از تکنیک پرتابی برای حل معادلات دیفرانسیل تأخیری در [۱۰] به تفصیل مورد بررسی قرار گرفته است.

در این مقاله، روش مستقیم بلوکی چندگامی به فرم بلوک تصحیح کننده ی پیشگو برای حل عددی معادلات دیفرانسیل تأخیری درجه دوم با شرایط مرزی ارایه می شود. تکنیک پرتابی برای محاسبه مقدار اولیه ی ابتدایی و روشی شبیه نیوتن [۶] برای یافتن مقادیر اولیه بعدی مورد استفاده قرار خواهد گرفت.

۲ بیان روش

در این بخش، روش مورد اشاره در مقاله را به تفصیل بررسی می کنیم. بازه ی $[a, b]$ را به بلوکهایی با طول گام $h = x_{i+1} - x_i$ و به شرح شکل ۱، افراز می کنیم. اکنون معادله دیفرانسیل معمولی $x'' = f(t, x, x')$ را در نظر گرفته و از آن دو بار انتگرال می گیریم تا فرمولهای مشتق برای نقاط x_{i+1} و x_{i+2} به شرح زیر حاصل شوند:

$$\text{نقطه اول } (x_{i+1})$$



شکل ۱: روش بلوکی دو نقطه ای

انتگرال گیری اول

$$\begin{aligned} \int_{t_i}^{t_{i+1}} x'' dt &= \int_{t_i}^{t_{i+1}} f(t, x, x') dt \\ x'_{i+1} &= x'_i + \int_{t_i}^{t_{i+1}} f(t, x, x') dt \end{aligned} \quad (2.1)$$

انتگرال گیری دوم

$$\begin{aligned} \int_{t_i}^{t_{i+1}} \int_{t_i}^t x'' dt dt &= \int_{t_i}^{t_{i+1}} \int_{t_i}^t f(t, x, x') dt dt \\ x_{i+1} &= x_i + h x'_i + \int_{t_i}^{t_{i+1}} (t_{i+1} - t) f(t, x, x') dt \end{aligned} \quad (2.2)$$

نقطه دوم (x_{i+2})

انتگرال گیری اول

$$\begin{aligned} \int_{t_i}^{t_{i+2}} x'' dt &= \int_{t_i}^{t_{i+2}} f(t, x, x') dt \\ x'_{i+2} &= x'_i + \int_{t_i}^{t_{i+2}} f(t, x, x') dt \end{aligned} \quad (2.3)$$

انتگرال گیری دوم

$$\begin{aligned} \int_{t_i}^{t_{i+2}} \int_{t_i}^t x'' dt dt &= \int_{t_i}^{t_{i+2}} \int_{t_i}^t f(t, x, x') dt dt \\ x_{i+2} &= x_i + 2h x'_i + \int_{t_i}^{t_{i+2}} (t_{i+2} - t) f(t, x, x') dt \end{aligned} \quad (2.4)$$

حال برای تقریب $f(t, x, x')$ در معادلات (۲/۱) تا (۲/۴) از درونیایی لاگرانژ به فرم زیر استفاده می کنیم [۱۰]:

$$L_r(t) = \sum_{j=0}^r \left(\prod_{n=0, n \neq j}^r \frac{(t - t_{i+p-n})}{(t_{i+p-j} - t_{i+p-n})} \right) f_{i+p-j}, \quad p = 0, 1, 2.$$

بنابراین روش بلوکی چند گامی مفروض، شامل روشهای بلوکی تصحیح کننده و پیشگو خواهد بود که برای فرمول پیشگو، $p = 0$ را برای هردو نقطه در نظر گرفته و درجه چند جمله ایهای لاگرانژ را $r = 3$ اختیار می کنیم که در نتیجه نقاط درگیر در درونیایی پیشگو برای هر دو نقطه



ی مفروض x_{i+1}, x_{i+2} عبارتند از $\{(t_i, f_i), (t_{i-1}, f_{i-1}), (t_{i-2}, f_{i-2}), (t_{i-3}, f_{i-3})\}$ و برای فرمول تصحیح کننده، $p = 1$ را برای نقطه اول و $p = 2$ را برای نقطه دوم در نظر گرفته و درجه چند جمله ایهای لاگرانژ را $r = 4$ اختیاری کنیم که در نتیجه نقاط درگیر در درونبایی تصحیح کننده برای نقطه ی x_{i+1} عبارتند از $\{(t_{i+1}, f_{i+1}), (t_i, f_i), (t_{i-1}, f_{i-1}), (t_{i-2}, f_{i-2}), (t_{i-3}, f_{i-3})\}$ و برای نقطه ی x_{i+2} عبارتند از $\{(t_{i+2}, f_{i+2}), (t_{i+1}, f_{i+1}), (t_i, f_i), (t_{i-1}, f_{i-1}), (t_{i-2}, f_{i-2})\}$.
با جایگذاری $t = t_{i+2} + sh$ در انتگرالهای فوق و ساده کردن انتگرالها، می توان روابط تصحیح کننده زیر را برای نقاط بیان شده بدست آورد:

نقطه اول (x_{i+1})

$$\begin{aligned} x'_{i+1} &= x'_i + \frac{h}{720} (251f_{i+1} + 646f_i - 264f_{i-1} + 106f_{i-2} - 19f_{i-3}) \\ x_{i+1} &= x_i + hx'_i + \frac{h^2}{1440} (135f_{i+1} + 752f_i - 246f_{i-1} + 96f_{i-2} - 17f_{i-3}) \end{aligned} \quad (2'5)$$

نقطه دوم (x_{i+2})

$$\begin{aligned} x'_{i+2} &= x'_i + \frac{h}{90} (29f_{i+2} + 124f_{i+1} + 24f_i + 4f_{i-1} - f_{i-2}) \\ x_{i+2} &= x_i + 2hx'_i + \frac{h^2}{90} (5f_{i+2} + 104f_{i+1} + 78f_i - 8f_{i-1} + f_{i-2}) \end{aligned} \quad (2'6)$$

چون در روش بیان شده در این مقاله، از تکنیک تصحیح کننده ی پیشگو استفاده خواهیم کرد، برای بخش پیشگو، با توجه به اینکه تصحیح کننده یک مرتبه بالاتر از آن قرار داده می شود، مشابه روند قبلی، می توان روابط پیشگو را به شرح زیر برای نقاط بیان شده بدست آورد:

نقطه اول (x_{i+1})

$$\begin{aligned} x'_{i+1} &= x'_i + \frac{h}{24} (55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}) \\ x_{i+1} &= x_i + hx'_i + \frac{h^2}{360} (323f_i - 264f_{i-1} + 159f_{i-2} - 38f_{i-3}) \end{aligned}$$

نقطه دوم (x_{i+2})

$$\begin{aligned} x'_{i+2} &= x'_i + \frac{h}{3} (27f_i - 44f_{i-1} + 31f_{i-2} - 8f_{i-3}) \\ x_{i+2} &= x_i + 2hx'_i + \frac{h^2}{45} (272f_i - 366f_{i-1} + 246f_{i-2} - 62f_{i-3}) \end{aligned}$$



۳ آنالیز روش

در این بخش، مرتبه روش بیان شده و پایداری آن را مورد بررسی قرار خواهیم داد. اگر روش بیان شده در بخش قبلی را به فرم ماتریسی در آوریم، می توان نوشت:

$$aX_m = hbX'_m + h^{\vee}cF_m$$

که در آن داریم:

$$X_m = \begin{bmatrix} x_{i-2} \\ x_{i-1} \\ x_i \\ x_{i+1} \\ x_{i+2} \end{bmatrix}, X'_m = \begin{bmatrix} x'_{i-2} \\ x'_{i-1} \\ x'_i \\ x'_{i+1} \\ x'_{i+2} \end{bmatrix}, F_m = \begin{bmatrix} f_{i-3} \\ f_{i-2} \\ f_{i-1} \\ f_i \\ f_{i+1} \\ f_{i+2} \end{bmatrix}$$

عملگر تفاضلی خطی L برای این رابطه را بصورت زیر در نظر می گیریم [۷]:

$$L[x(t); h] = \sum_{j=0}^k [a_j x(t+jh) - hb_j x'(t+jh) - h^{\vee}c_j x''(t+jh)] \quad (3.1)$$

اگر جملات مجموع بالا را برحسب بسط سری تیلور بازنویسی کنیم، به رابطه ی زیر می رسیم:

$$L[x(t); h] = d_0 x(t) + d_1 h x'(t) + d_2 h^{\vee} x''(t) + \dots + d_r h^r x^{(r)}(t) + \dots,$$

که در آن داریم:

$$\begin{aligned} d_0 &= a_0 + a_1 + \dots + a_s \\ d_1 &= a_1 + 2a_2 + 3a_3 + \dots + sa_s - (b_0 + b_1 + b_2 + \dots + b_s) \\ d_r &= \frac{1}{r!} (a_1 + 2^r a_2 + 3^r a_3 + \dots + s^r a_s) - \frac{1}{(r-1)!} (b_1 + 2^{r-1} b_2 + 3^{r-1} b_3 + \dots + s^{r-1} b_s) \\ &\quad - \frac{1}{(r-2)!} (c_1 + 2^{r-2} c_2 + 3^{r-2} c_3 + \dots + s^{r-2} c_s), \quad r = 2, 3, \dots \end{aligned} \quad (3.2)$$

تعریف ۱. مرتبه عملگر تفاضلی خطی $L[x(t); h]$ ، عدد منحصر بفرد r است بطوریکه $d_{r+2} \neq 0$ [۷]

اگر فرمولهای (۲۵) و (۲۶) را بفرم ساختار ماتریسی بیان شده بنویسیم و معادلات متناظر (۳۲) را برای آنها حل کنیم، داریم:
 $d_0 = d_1 = d_2 = \dots = d_6 = [0 \ 0 \ 0 \ 0]^T \neq 0$ در حالیکه $d_7 = [-\frac{3}{16} \ -\frac{41}{504} \ -\frac{1}{90} \ -\frac{1}{315}]^T \neq 0$ یعنی طبق تعریف بیان شده، مرتبه ی روش ۵ خواهد بود.

تعریف ۲. روش چندگامی خطی را صفر- پایدار گوئیم هرگاه اولین چندجمله ای مشخصه یعنی $\rho(R) = \det[\sum_{i=0}^p A_{(i)} R^{p-i}] = 0$ دارای ریشه های R_j ای باشد که $|R_j| \leq 1$ و آن را سازگار گوئیم هرگاه مرتبه ی آن $r \geq 1$ باشد. [۷]

فرمولهای (۲۵) و (۲۶) را بصورت زیر بازنویسی می کنیم:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x'_{i+1} \\ x_{i+1} \\ x'_{i+2} \\ x_{i+2} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x'_{i-1} \\ x_{i-1} \\ x'_i \\ x_i \end{pmatrix} + h \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \end{pmatrix} \begin{pmatrix} x'_{i-1} \\ x_{i-1} \\ x'_i \\ x_i \end{pmatrix} \\ + h \begin{pmatrix} 0 & 0 & -\frac{19}{720} & \frac{53}{360} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{90} \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} f_{i-5} \\ f_{i-4} \\ f_{i-3} \\ f_{i-2} \end{pmatrix} + h \begin{pmatrix} -\frac{11}{30} & \frac{223}{360} & \frac{251}{720} & 0 \\ 0 & 0 & 0 & 0 \\ \frac{2}{45} & \frac{4}{15} & \frac{62}{45} & \frac{29}{90} \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} f_{i-1} \\ f_i \\ f_{i+1} \\ f_{i+2} \end{pmatrix} \\ + h^2 \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{17}{1440} & \frac{1}{15} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{90} \end{pmatrix} \begin{pmatrix} f_{i-5} \\ f_{i-4} \\ f_{i-3} \\ f_{i-2} \end{pmatrix} + h^2 \begin{pmatrix} 0 & 0 & 0 & 0 \\ -\frac{41}{720} & \frac{47}{90} & \frac{3}{32} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{4}{45} & \frac{13}{15} & \frac{52}{45} & \frac{1}{18} \end{pmatrix} \begin{pmatrix} f_{i-1} \\ f_i \\ f_{i+1} \\ f_{i+2} \end{pmatrix}$$

بنابراین اولین چندجمله ای مشخصه مدل بیان شده عبارتست از

$$\rho(R) = \det[\sum_{i=0}^p A_{(i)} R^{p-i}] = \det[A_0 R^1 - A_1] = 0$$

که

$$A_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \rho(R) = \begin{vmatrix} R & 0 & -1 & 0 \\ 0 & R & 0 & -1 \\ 0 & 0 & R-1 & 0 \\ 0 & 0 & 0 & R-1 \end{vmatrix}$$

و ریشه های دترمینان فوق برابرند با $1, 1, 0, 0$ و چون $|R_j| \leq 1$ ، بنابراین روش مورد بحث، صفرپایدار هم هست.

تعریف ۳. شرط لازم و کافی برای همگرایی روش چند گامی خطی، آن است که روش سازگار و صفرپایدار باشد. [۹]

با توجه به اینکه روش بیان شده، سازگار و صفرپایدار بود، بنابراین همگرا هم هست.



جدول ۱: مقایسه نتایج حاصل از اجرای روش مقاله با روشهای دیگر با طول گام $h = \frac{\pi}{400}$

روش مقاله	روش [۱۵]	روش [۳]	t
0.0000×10^0	0.0000×10^0	0.0000×10^0	۰/۰
5.6732×10^{-9}	1.6360×10^{-7}	2.4115×10^{-7}	۰/۰۸
5.2948×10^{-9}	1.5256×10^{-7}	4.3261×10^{-7}	۰/۱۶
4.8466×10^{-9}	1.3953×10^{-7}	5.7246×10^{-7}	۰/۲۴
4.3312×10^{-9}	1.2460×10^{-7}	6.5906×10^{-7}	۰/۳۱
3.7513×10^{-9}	1.0784×10^{-7}	6.9097×10^{-7}	۰/۳۹
3.1100×10^{-9}	8.9341×10^{-8}	6.6707×10^{-7}	۰/۴۷
2.4105×10^{-9}	6.9204×10^{-8}	5.8648×10^{-7}	۰/۵۵
1.6565×10^{-9}	4.7529×10^{-8}	4.4860×10^{-7}	۰/۶۳
8.5168×10^{-10}	2.4422×10^{-8}	2.5312×10^{-7}	۰/۷۱
4.4409×10^{-16}	0.0000×10^0	0.0000×10^0	۰/۷۹
5.9510×10^{-9}	1.6926×10^{-7}	6.9097×10^{-7}	حداکثر خطای مطلق
۰/۰۰۰۶۶۷	۰/۰۰۱۳۳۳	—	زمان پردازش سیستم

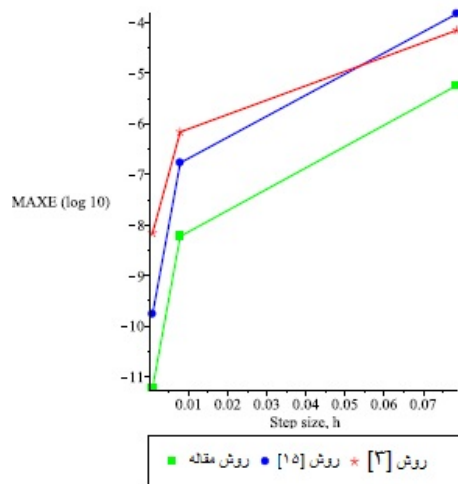
۴ مثال عددی

معادله دیفرانسیل پنتوگراف زیر را در نظر بگیرید [۳، ۱۵]

$$x''(t) = 1 + 2\left(1 + \frac{t^2}{8}\right)\cos\left(\frac{t}{4}\right) - 2\cos\left(\frac{t}{4}\right)x\left(\frac{t}{4}\right), \quad 0 \leq t \leq \frac{\pi}{4}$$

$$x(0) = 1, \quad x\left(\frac{\pi}{4}\right) = 1 + \frac{\sqrt{2}}{2} + \frac{\pi^2}{32}$$

جواب واقعی این مسئله برابر $x(t) = 1 + \frac{t^2}{4} + \sin(t)$ است. در جدول ۱، نتایج حاصل از به کارگیری روش مورد اشاره در این مقاله را نسبت به روشهای بیان شده در مراجع [۳] و [۱۵] نشان داده ایم. همچنین در شکل ۲، عملکرد روش مورد بررسی در این مقاله را در مقایسه با روشهای دیگر، وقتی طول گام h کوچکتر می شود، نشان داده ایم. همانطور که دیده می شود، خطای مطلق در روش جدید، در مقایسه با روشهای مورد بررسی دیگر، در هرگام کمتر می باشد که کارایی و دقت روش جدید را در مقایسه با روشهای دیگر نشان می دهد.



شکل ۲: نمودار لگاریتمی حداکثر خطای مطلق در مقابل طول گام h در روشهای مورد بررسی

۵ نتیجه گیری

در این مقاله، یک روش بلوکی چندگامی مرتبه ۵ برای حل عددی معادلات دیفرانسیل تأخیری درجه دوم با شرایط مرزی و با طول گام ثابت، ارایه گردید. این روش، در هر گام، جوابهای عددی را بطور همزمان برای دو نقطه محاسبه می کرد. روش همگرا، سازگار و صفرپایدار بود و دقت خوبی در مقایسه با روشهایی داشت که در همین حیطه و قبلاً بیان شده بودند. این روش همچنین برای تأخیرهای از نوع ثابت، پانتوگراف و وابسته به زمان قابل اجراست.

مراجع

- [1] V.L. Bakke; Z. Jackiewicz, *The numerical solution of boundary value problems for differential equations with state dependent deviating arguments* Apl. Mat. **34**(1989), 1–17.
- [2] A. Bellen; M. Zennaro, *A collocation method for boundary value problems of differential equations with functional arguments*, Computing. **32**(1984), 307–318.
- [3] A.M. Bica, *The numerical method of successive interpolations for two-point boundary value problems with deviating argument*, Comput. Math. Appl. **62**(2011), 3829–3843.

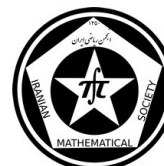


- [4] B. Cahlon; L.J. Nachman, *Numerical methods for discontinuous linear boundary value problems with deviation arguments*, J. Math. Anal. Appl. **154**(1991), 529–542.
- [5] C.W. Cryer, *The numerical solution of boundary value problems for second order functional differential equations by finite differences*, Numer. Math. **20**(1973), 288–299.
- [6] L. Fang; L. Sun; G. He, *An efficient Newton-type method with fifth-order convergence for solving nonlinear equations*. Comput. Appl. Math, **27**(2008), 269–274.
- [7] S. Ola Fatunla, *Block methods for second order ODEs*. Int. J. Comput. Math. **41**(1991), 55–63.
- [8] Y. Kuang, *Delay Differential Equations: With Applications in Population Dynamics*, Academic Press: Cambridge, MA, USA, 1993.
- [9] J.D. Lambert, *Computational Methods in Ordinary Differential Equations*, Wiley: Hoboken, NJ, USA, 1973.
- [10] N. Mohd Nasir; Z. Abdul Majid; F. Ismail; N. Bachok, *Direct Integration of Boundary Value Problems Using the Block Method via the Shooting Technique Combined with Steffensen's Strategy*, Mathematics **7**(2019), 1075.
- [11] K.D. Nevers; K. Schmitt, *An application of the shooting method to boundary value problems for second order delay equations*. J. Math. Anal. Appl. **36**(1971), 588–597.
- [12] G. Reddien, *Difference approximations of boundary value problems for functional differential equations*, J. Math. Anal. Appl. **63**(1978), 678–686.
- [13] M. Sakai, *Numerical solution of boundary value problems for second order functional differential equations by the use of cubic splines*, Mem. Fac. Sci. Kyushu Univ. Ser. Math. **29**(1975), 113–122.
- [14] H.Y. Seong; Z. Abdul Majid, *Direct method for solving second order delay differential equations*, In Proceedings of the AIP Conference, Sozopol, Bulgaria, **1522**(2013), 676–680.
- [15] H.Y. Seong; Z.A. Majid, *Solving second order delay differential equations using direct two-point block method*, Ain Shams Eng. J. **8**(2017), 59–66.



- [16] A.M. Wazwaz; M.A.Z. Raja; M.I. Syam, *Reliable treatment for solving boundary value problems of pantograph delay differential equation*, Rom. Rep. Phys. **69**(2017), 102.
- [17] H. Yann Seong; Z. Abdul Majid; F. Ismail, *Solving second-order delay differential equations by direct Adams-Moulton method*, Math. Probl. Eng.(2013).

پست الکترونیکی: m_dadkhah@pnu.ac.ir



روش تقریب‌های پیشرو چندجمله‌ای‌های چیشف نوع اول برای حل عددی معادلات دیفرانسیل

معمولی مرتبه دوم

محمد تفکری بافقی[†]، قاسم برید لقمانی و محمد حیدری[‡]

بخش ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه یزد، یزد، ایران[†]

چکیده

در این مقاله یک روش کارا برای حل مسائل مقدار اولیه مرتبه‌ی دوم ارائه شده است. روش پیشنهادی یک روش تکراری مبتنی بر روش تقریب‌های متوالی به همراه چندجمله‌ای‌های چیشف نوع اول است. با انتقال مسئله به بازه‌ی $[-1, 1]$ و دنباله‌ی روش تقریب‌های متوالی، در هر مرحله جواب تقریبی به صورت یک سری چندجمله‌ای به دست می‌آید. در پایان کارایی روش پیشنهادی با یک مثال شرح داده شده است.

واژه‌های کلیدی: معادلات دیفرانسیل معمولی، مسائل مقدار اولیه مرتبه دوم، روش تقریب‌های متوالی، چندجمله‌ای‌های چیشف. طبقه‌بندی موضوعی [۲۰۱۰]: 34B05, 47J25.

۱ مقدمه

یکی از روش‌های علمی برای تجزیه و تحلیل پدیده‌های طبیعی، ساخت و ارائه‌ی مدل‌های ریاضی برای پدیده‌هاست. این مدل‌ها در واقع بیان‌گر واقعیت‌های حاکم بر مسائل، به‌ویژه بیان روابط میان داده‌ها و مجهول‌های مسئله با استفاده از معادلات ریاضی است. بسیاری از مسائل آکوستیک، الکتریسته و مغناطیس، مکانیک سماوی، مکانیک کوانتومی، مکانیک سیالات، زیست-شناسی، واکنش‌های شیمیایی گازها، داروسازی و حتی اقتصاد را می‌توان به صورت معادلات دیفرانسیل مدل‌سازی کرد. در حالت کلی، به‌ویژه زمانی که مسئله‌ی مورد بحث غیرخطی باشد، روش‌های تحلیلی برای حل آن موجود نیست یا محاسبه‌ی جواب دقیق، دشوار و پرهزینه است؛ بنابراین توسعه‌ی الگوریتم‌های اساساً عددی و نیمه‌تحلیلی کارا برای این که جواب را با دقت مناسب محاسبه کرده و نیز از نظر حجم محاسبات، زمان انجام و حجم حافظه‌ی اشغال‌شده، مقرون به صرفه باشند، موضوع مورد علاقه‌ی بسیاری از مهندسان و ریاضیدانان است [۱-۵].



۲ چندجمله‌ای‌های چیشف و ویژگی‌های آن

چندجمله‌ای‌های چیشف، مجموعه‌ای از چندجمله‌ای‌های متعامد* هستند که کاربرد بسیاری در تقریب توابع دارند. این چندجمله‌ای‌ها روی بازه‌ی $[-1, 1]$ نسبت به تابع وزن $\omega(x) = \frac{1}{\sqrt{1-x^2}}$ متعامدند. چندجمله‌ای‌های چیشف، توابع ویژه‌ی مسئله‌ی اشتورم-لیوویل تکین[†]

$$\sqrt{1-\tau^2} \left(\sqrt{1-\tau^2} T_n'(\tau) \right)' + n^2 T_n(\tau) = 0, \quad (2.1)$$

هستند و این چندجمله‌ای‌ها را می‌توان از رابطه‌ی بازگشتی

$$T_0(\tau) = 1, \quad T_1(\tau) = \tau, \quad (2.2)$$

$$T_{n+1}(\tau) = 2\tau T_n(\tau) - T_{n-1}(\tau), \quad n \geq 1, \quad -1 \leq \tau \leq 1,$$

به‌دست آورد [۶].

نتیجه ۱. [۶] چندجمله‌ای‌های چیشف در رابطه‌های زیر صدق می‌کنند:

$$T_n(-\tau) = (-1)^n T_n(\tau), \quad n \geq 0, \quad (2.3)$$

$$T_n(\pm 1) = (\pm 1)^n, \quad n \geq 0, \quad (2.4)$$

$$|T_n(\tau)| \leq 1, \quad n \geq 0, \quad (2.5)$$

$$T_m(\tau) T_n(\tau) = \frac{1}{2} (T_{m+n}(\tau) + T_{m-n}(\tau)), \quad m \geq n \geq 0, \quad (2.6)$$

$$\int_{-1}^1 T_m(\tau) T_n(\tau) \frac{1}{\sqrt{1-\tau^2}} d\tau = \frac{c_n \pi}{2} \delta_{mn}, \quad c_0 = 2, \quad c_n = 1, \quad n \geq 1, \quad (2.7)$$

$$\int T_n(\tau) d\tau = \frac{1}{2} \left(\frac{1}{n+1} T_{n+1}(\tau) - \frac{1}{n-1} T_{n-1}(\tau) \right) + c, \quad n \geq 1. \quad (2.8)$$

۳ تقریب گسسته توابع با استفاده از چندجمله‌ای‌های چیشف نوع اول

فرض کنید $f(\tau) \in L_\omega^1[-1, 1]$. در این صورت $f(\tau)$ را می‌توان روی بازه‌ی $[-1, 1]$ به‌صورت

$$f(\tau) \simeq \sum_{k=0}^N a_k T_k(\tau), \quad (3.1)$$

تقریب زد، که در آن ضرایب از رابطه‌ی

$$a_k = \frac{2}{N} \sum_{j=0}^N f(\tau_j) T_k(\tau_j), \quad k = 0, 1, \dots, N, \quad (3.2)$$

*Orthogonal polynomials

[†]Singular Sturm-Liouville problem



به دست می آیند و

$$\tau_j = \cos\left(\frac{j\pi}{N}\right), \quad j = 0, 1, \dots, N. \quad (3.3)$$

علامت‌های ' و '' به ترتیب به این معناست که جمله‌ی اول و جملات اول و آخر مجموع، نصف می‌شوند [۶].

۴ روش تقریب‌های پیشرو چند جمله‌ای‌های چیشف نوع اول

معادله‌ی دیفرانسیل معمولی غیرخطی مرتبه‌ی اول

$$y'(t) = f(t, y(t)), \quad a \leq t \leq b, \quad (4.1)$$

را با شرط اولیه‌ی

$$y(a) = y_a, \quad (4.2)$$

در نظر بگیرید که در آن t متغیر مستقل است. نخست با تعویض متغیر

$$t = \frac{b-a}{2}\tau + \frac{b+a}{2}, \quad -1 \leq \tau \leq 1, \quad (4.3)$$

مسئله را از بازه‌ی $[a, b]$ به بازه‌ی $[-1, 1]$ انتقال می‌دهیم. لذا با در نظر گرفتن $x(\tau) = y\left(\frac{b-a}{2}\tau + \frac{b+a}{2}\right)$ خواهیم داشت:

$$\frac{dx(\tau)}{d\tau} = \frac{b-a}{2} f\left(\frac{b-a}{2}\tau + \frac{b+a}{2}, x(\tau)\right) \equiv g(\tau, x(\tau)), \quad -1 \leq \tau \leq 1, \quad (4.4)$$

$$x(-1) = y_a. \quad (4.5)$$

با انتگرال‌گیری از طرفین (۴.۴)، این معادله‌ی دیفرانسیل را می‌توان به صورت معادله‌ی انتگرال

$$x(\tau) = y_a + \int_{-1}^{\tau} g(s, x(s)) ds, \quad (4.6)$$

بازنویسی کرد و دنباله‌ی تقریب‌های متوالی را به صورت

$$x_i(\tau) = y_a + \int_{-1}^{\tau} g(s, x_{i-1}(s)) ds, \quad (4.7)$$

در نظر گرفت. دنباله‌ی $\{x_i(\tau)\}_{i=0}^{\infty}$ تحت شرایط مناسب روی g به جواب معادله‌ی دیفرانسیل (۴.۴) همگراست [۷].

برای حل مسئله ی (۴۰۷) با روش پیشنهادی، نخست تقریب انتگرال ده[‡] $g(s, x_{i-1}(s))$ را با استفاده از $\{T_r(s)\}_{r=0}^N$ به دست می آوریم. برای شروع تکرار، حدس اولیه ی $x_0(s)$ لازم است. با فرض این که جواب تقریبی در $(i-1)$ -امین تکرار با $x_{i-1}(\tau)$ نشان داده شود، داریم:

$$g(s, x_{i-1}(s)) \simeq \sum_{r=0}^N G_r^{i-1} T_r(s), \quad (408)$$

که در آن:

$$G_r^{i-1} = \frac{1}{N} \sum_{j=0}^N g(\tau_j, x_{i-1}(\tau_j)) T_r(\tau_j), \quad r = 0, 1, \dots, N, \quad (409)$$

و نقاط $\{\tau_j\}_{j=0}^N$ در (۳۰۳) تعریف شده اند. با جای گذاری (۴۰۸) در (۴۰۷) و استفاده از فرمول انتگرال گیری (۲۰۸)، داریم:

$$\begin{aligned} x_i(\tau) &\simeq y_a + \int_{-1}^{\tau} \sum_{r=0}^N G_r^{i-1} T_r(s) ds \\ &= y_a + \int_{-1}^{\tau} \left(\frac{1}{N} G_0^{i-1} T_0(s) + G_1^{i-1} T_1(s) + \dots + G_N^{i-1} T_N(s) \right) ds \\ &= c + \frac{1}{N} G_0^{i-1} T_1(\tau) + \frac{1}{N} G_1^{i-1} T_2(\tau) + \frac{1}{N} G_2^{i-1} \left(\frac{1}{N} T_3(\tau) - T_1(\tau) \right) \\ &\quad + \dots + \frac{1}{N} G_N^{i-1} \left(\frac{1}{N+1} T_{N+1}(\tau) - \frac{1}{N-1} T_{N-1}(\tau) \right), \end{aligned} \quad (410)$$

که در آن c یک عدد ثابت است و از شرط اولیه ی (۴۰۵) به دست می آید. با فرض این که جواب تقریبی در i -امین تکرار به صورت

$$x_i(\tau) = \sum_{k=0}^N a_k^i T_k(\tau), \quad (411)$$

در نظر گرفته شود، با جای گذاری در (۴۰۹)، بعد از ساده سازی سمت راست، ضرایب $\{a_k^i\}_{k=0}^N$ بلافاصله از برابر هم قرار دادن ضرایب چندجمله ای های چپیش در طرفین (۴۰۹)، به صورت

$$a_r^i = \frac{1}{N} (G_{r-1}^{i-1} - G_{r+1}^{i-1}), \quad r = 1, 2, \dots, N-1, \quad (412)$$

$$a_N^i = \frac{1}{N} G_{N-1}^{i-1}, \quad (413)$$

به دست می آیند. حال با توجه به (۴۰۹) و شرط اولیه ی (۴۰۵) داریم:

$$a_0^i = y_a + \sum_{k=1}^N (-1)^{k+1} a_k^i. \quad (414)$$

[‡]Integrand



ضرایب به دست آمده، جواب تقریبی تکرار i -ام را برای استفاده در (۴۰۷) به دست می دهند. بدین صورت روش تا زمانی که شرط توقف معینی برآورده گردد، ادامه می یابد. معمولاً از معیار توقف

$$\|x_i(\tau) - x_{i-1}(\tau)\| < \epsilon, \quad (405)$$

استفاده می شود که در آن ϵ دقت داده شده است.

نکته ۲. مسئله مقدار اولیه مرتبه دوم

$$y''(t) = f(t, y(t), y'(t)), \quad a \leq t \leq b, \quad (406)$$

را با شرایط اولیه

$$y(a) = y_a, \quad y'(a) = y'_a, \quad (407)$$

در نظر بگیرید. باتوجه به روند بخش ۴، می توان مسائل مقدار اولیه مرتبه دوم را حل کرد و به مرتبه های بالاتر نیز تعمیم داد؛ بدین صورت که با در نظر گرفتن $Y_1(t) = y(t)$ و $Y_2(t) = y'(t)$ ، می توان (۴۰۶) را به یک دستگاه معادلات دیفرانسیل مرتبه اول به صورت

$$Y_1'(t) = Y_2(t), \quad Y_1(a) = y_a, \quad (408)$$

$$Y_2'(t) = f(t, Y_1(t), Y_2(t)), \quad Y_2(a) = y'_a, \quad (409)$$

تبدیل کرد. حال باتوجه به روند بخش ۴، نخست (۴۰۹) را با روش پیشنهادی حل نموده و $Y_{2,i}(t)$ را به دست می آوریم. سپس با جای گذاری $Y_{2,i}(t)$ در (۴۰۸)، $Y_{1,i}(t)$ یا همان $y_i(t)$ را با روش پیشنهادی به دست می آوریم.

نکته ۳. روش پیشنهادی اساساً یک روش تکراری است، که می توان با تکرار بیشتر الگوریتم، جواب دقیق تری به دست آورد. در روند اجرای الگوریتم نیازی به مشتق گیری یا انتگرال گیری نیست. نیاز به حل دستگاه خطی یا غیرخطی و نیز محاسبه ی وارون ماتریس ندارد؛ پس در ابعاد بالا پرهزینه نیست.

۵ نتایج عددی

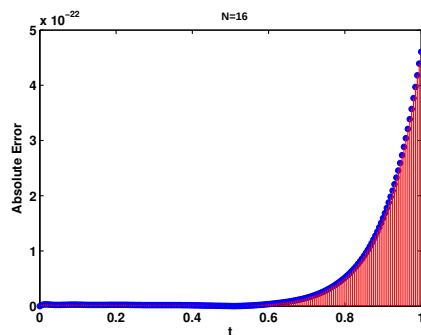
در این بخش با یک مثال، کارایی روش پیشنهادی را نشان می دهیم.

مثال ۴. مسئله مقدار اولیه زیر را در نظر بگیرید [۹، ۸]:

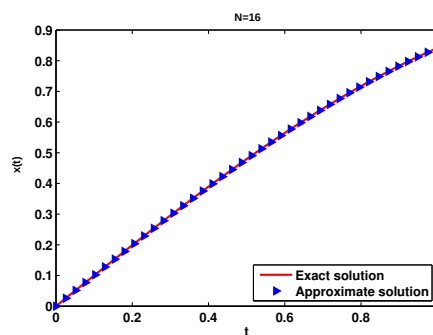
$$\begin{cases} x''(t) + x'(t) + x(t) + x^2(t)x'(t) = 2 \cos(t) - \cos^3(t), & t \in [0, 1], \\ x(0) = 0, & x'(0) = 1. \end{cases} \quad (51)$$



که جواب دقیق آن به صورت $x(t) = \sin(t)$ است. نمودار جواب‌های تقریبی و دقیق در شکل ۱ رسم شده‌اند. از جدول ۱، مشاهده می‌شود با افزایش N ، دقت روش افزایش می‌یابد. برای مقایسه‌ی بهتر، خطای مطلق نقطه به نقطه‌ی حاصل از روش پیشنهادی به‌ازای $N = 10$ ، $\varepsilon = 10^{-15}$ با خطای مطلق روش‌های تکرار وردشی [۸]، اختلال هموتوبی [۸] و روش هیبرید تک‌گامی بلوکی [۹] در جدول ۲ و شکل ۲ مقایسه شده است. ملاحظه می‌شود جواب تقریبی حاصل از روش پیشنهادی از سایر روش‌های تقریبی ذکر شده، به جواب دقیق نزدیک‌تر است. شکل ۳ نمودار لگاریتم ماکزیمم خطای مطلق روش را به‌ازای مقادیر مختلف N نشان می‌دهد. همان‌گونه که انتظار داشتیم، خطا با افزایش N به‌صورت نمایی کاهش می‌یابد.



(ب)



(T)

شکل ۱: (T) مقایسه‌ی جواب‌های دقیق و تقریبی، (ب) خطای مطلق روش پیشنهادی به‌ازای $N = 16$ و معیار توقف $\varepsilon = 10^{-20}$.

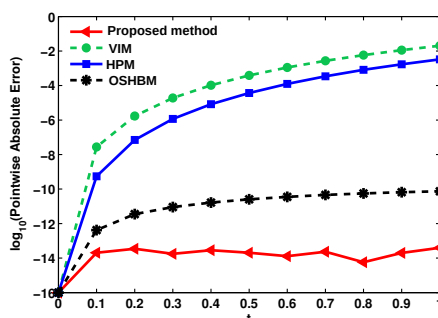
جدول ۱: ماکزیمم خطای مطلق (L_∞) و تعداد تکرارهای (IN) روش پیشنهادی با معیار توقف $\varepsilon = 10^{-20}$.

N	زمان محاسباتی	IN	L_∞
۴	۰/۲۰۴	۳۶	$2/16e - 05$
۸	۰/۲۰۳	۲۵	$2/69e - 11$
۱۲	۰/۲۸۱	۲۳	$6/06e - 18$
۱۶	۰/۵۹۴	۲۲	$4/68e - 23$

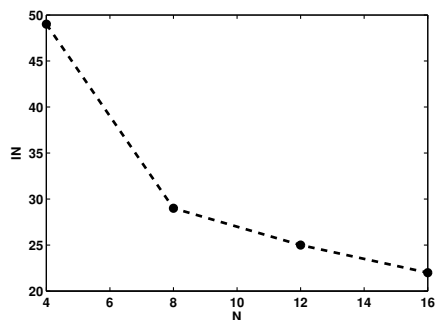


جدول ۲: مقایسه‌ی خطای مطلق نقطه به نقطه‌ی روش پیشنهادی به‌ازای $N = 10$ ، $\varepsilon = 10^{-15}$ با روش‌های مختلف.

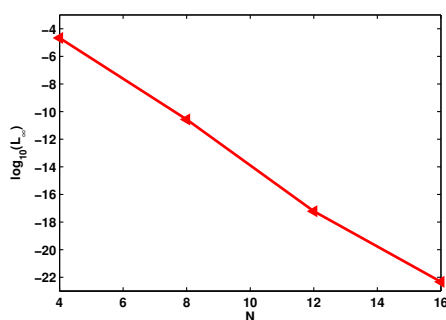
t	روش پیشنهادی	OSHBM [۹]	HPM [۸]	VIM [۸]
۰/۰	۰	۰	۰	۰
۰/۱	$2/62645218e-15$	$4/16719627e-13$	$5/50000000e-10$	$2/73500000e-08$
۰/۲	$1/64378234e-14$	$3/54860749e-12$	$7/04000000e-08$	$1/71600000e-06$
۰/۳	$1/76111984e-15$	$9/04722120e-12$	$1/15750000e-06$	$1/90997000e-05$
۰/۴	$1/61807547e-14$	$1/65024104e-11$	$8/24130000e-06$	$1/04508100e-04$
۰/۵	$6/62017234e-15$	$2/54436093e-11$	$3/71214000e-05$	$3/868817900e-04$
۰/۶	$2/14108525e-15$	$3/52559007e-11$	$1/24934600e-04$	$1/11718070e-03$
۰/۷	$1/55330026e-14$	$4/57082897e-11$	$3/43244100e-04$	$2/71479400e-03$
۰/۸	$3/20965167e-15$	$5/59198114e-11$	$8/11615800e-04$	$5/80911310e-03$
۰/۹	$1/10091223e-14$	$6/57446428e-11$	$1/70896240e-03$	$1/12705564e-02$
۱/۰	$1/32179438e-14$	$7/46029138e-11$	$3/27986730e-03$	$2/02266500e-02$



شکل ۲: نمودار خطای مطلق نقطه به نقطه‌ی روش پیشنهادی با معیار توقف $\varepsilon = 10^{-20}$ در مقایسه با سایر روش‌ها.



(ب)



(ت)

شکل ۳: (ت) نمودار لگاریتم خطای L_{∞} ، (ب) لگاریتم تعداد تکرارهای روش با معیار توقف $\varepsilon = 10^{-20}$.

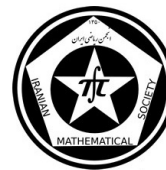


- [1] F. Costabile, A. Napoli, *A class of collocation methods for numerical integration of initial value problems*, Comput. Math. App. 62 (2011) 3221-3235.
- [2] E.H. Doha, A.H. Bhrawy, D. Baleanu, R.M. Hafez, *Efficient Jacobi-Gauss Collocation Method for Solving Initial Value Problems of Bratu Type*, Comp. Math. Math. Phys. 53(9) (2013) 1292-1302.
- [3] M. Heydari, G.B. Loghmani, A.M. Wazwaz, *A numerical approach for a class of astrophysics equations using piecewise spectral-variational iteration method*, Int. J. Numer. Methods Heat. Fluid. Flow. 27(2) (2017) 358-378.
- [4] M.Y. Ongun, *Laplace Adomian decomposition method for solving a model for HIV Infection of CD4+ T cells*, Math. Comput. Model. 53 (2011) 597-603.
- [5] R. Rach, A.M. Wazwaz, J.S. Duan, *A reliable modification of the Adomian decomposition method for higher-order nonlinear differential equations*, Kybernetes 42(2) (2013) 282-308.
- [6] J. Shen, T. Tang, L.L. Wang, *Spectral Methods: Algorithms, Analysis and Applications*, Springer-Verlag, Berlin, Heidelberg, 2011.
- [7] A.M. Wazwaz, *Linear and Nonlinear Integral Equations: Methods and Applications*, Springer Science & Business Media, Beijing, 2011.
- [8] A. Barari, M. Omidvar, A.R. Ghotbi, D.D. Ganji, *Application of homotopy perturbation method and variational iteration method to nonlinear oscillator differential equations*, Acta Appl. Math. 104(2) (2008) 161-171.
- [9] E.O. Adeyefa, *Orthogonal-based hybrid block method for solving general second order initial value problems*, Int. J. Pure Appl. Math. 37 (2017) 659-672.

پست الکترونیکی: m.tafakkori@stu.yazd.ac.ir, mtafakkori@gmail.com

پست الکترونیکی: loghmani@yazd.ac.ir

پست الکترونیکی: m.heydari@yazd.ac.ir



روش هم‌محلی بر اساس پایه‌های چند ترازی برای حل دستگاه‌های مرکب معادلات انتگرال

ولترا-فردهلم نوع اول و دوم

منصور حسینی آذر[†]، سعید پیش‌بین[‡]

دانشکده ریاضی، دانشگاه ارومیه، ارومیه، ایران[†]

دانشکده ریاضی، دانشگاه ارومیه، ارومیه، ایران[‡]

چکیده

در این مقاله، حل دستگاه‌های مرکب معادلات انتگرال ولترا-فردهلم نوع اول و دوم بر اساس روش هم‌محلی چند ترازی بررسی می‌شود. این نوع معادلات جز مسائل بدووضع محسوب می‌شوند، به همین دلیل با به کار بردن روش منظم سازی مناسب ابتدا این معادلات را به معادلات خوش‌وضع تبدیل کرده و سپس روش هم‌محلی بر اساس پایه‌های چند ترازی را برای حل مسئله‌ی خوش‌وضع پیاده می‌کنیم. در ادامه پارامتر منظم سازی را تعیین نموده و با مثال‌های عددی متنوع کارایی روش را بررسی می‌کنیم. واژه‌های کلیدی: دستگاه‌های مرکب معادلات انتگرال ولترا-فردهلم نوع اول و دوم، منظم سازی، روش هم‌محلی چند ترازی. طبقه‌بندی موضوعی [۲۰۱۰]: 65R20, 34A04.

۱ پیش‌گفتار

در این مقاله دستگاه مرکب معادلات انتگرال ولترا-فردهلم (VFIEs)

$$F(t, x) = AU(t, x) + \int_0^t \int_{\Omega} K(t, \eta, x, s)U(\eta, s)dsd\eta, \quad (1.1)$$

که در آن:

$$F(t, x) = [f(t, x), g(t, x)] \quad , \quad U(t, x) = [u_1(t, x), u_2(t, x)],$$

و

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \quad K(t, \eta, x, s) = \begin{bmatrix} k_{11}(t, \eta, x, s) & k_{12}(t, \eta, x, s) \\ k_{21}(t, \eta, x, s) & k_{22}(t, \eta, x, s) \end{bmatrix}.$$

را در نظر می‌گیریم که Ω یک ناحیه ی کراندار روی R^d ، $d = (1, 2, 3)$ با مرز هموار $\partial\Omega$ می‌باشد [۵]. سیستم‌های مرکب معادلات انتگرال ولترا-فردهلم (VFIEs) به عنوان مسائل بدووضع در نظر گرفته می‌شوند. با وجود این ما ابتدا روش منظم سازی را که به صورت قابل توجهی در حل معادلات نوع اول مفید خواهد بود اعمال می‌کنیم. این روش سیستم ترکیبی را به سیستم معادلات انتگرال نوع دوم تبدیل می‌کند. روش‌های متفاوتی برای حل معادلات انتگرال ولترا-فردهلم نوع دوم پیشنهاد شده‌اند، در این مقاله روش هم‌محلی بر اساس پایه‌های چند ترازی ارائه شده است. ما روش هم‌محلی چند ترازی را بر روی سیستم معادلات منظم سازی شده اعمال می‌کنیم و معادلات را بر اساس پایه های چند ترازی فرمول بندی می‌کنیم. با این کار، سیستم معادلات پیوسته را به سیستم گسسته تبدیل کرده و از حل دستگاه گسسته جواب تقریبی معادلات را بدست می‌آوریم.

۲ منظم سازی تیخونوف

عملگر انتگرالی خطی تعریف شده در سیستم (۱'۱) در حالت کلی دارای معکوس پیوسته نمی‌باشد، بنابراین بدست آوردن یک جواب مناسب با روش‌های گسسته سازی کلاسیک دشوار است. به همین دلیل تکنیک‌های منظم سازی را می‌توان به جای تبدیل معادلات انتگرالی از قبیل معادله ی دوم در سیستم (۱'۱) به معادله ی انتگرال نوع دوم بکار برد. به طور دقیق تر معادلات انتگرالی زیر را در نظر می‌گیریم

$$\begin{cases} f(t, x) = u_1(t, x) + \int_0^t \int_{\Omega} k_{11}(t, \eta, x, s) u_1(\eta, s) ds d\eta \\ \quad + \int_0^t \int_{\Omega} k_{12}(t, \eta, x, s) u_{2,\epsilon}(\eta, s) ds d\eta \\ g(t, x) = \epsilon u_{2,\epsilon}(t, x) + \int_0^t \int_{\Omega} k_{21}(t, \eta, x, s) u_1(\eta, s) ds d\eta \\ \quad + \int_0^t \int_{\Omega} k_{22}(t, \eta, x, s) u_{2,\epsilon}(\eta, s) ds d\eta, \end{cases} \quad (2'1)$$

به طوریکه ϵ عدد ثابت مثبتی است.

می‌توان ثابت کرد که جواب $u_{2,\epsilon}(x, t)$ از معادله ی (۲'۱) تحت شرایطی وقتی $\epsilon \rightarrow 0$ به جواب $u_2(x, t)$ از معادله ی (۱'۱) همگرا است. برای این منظور شکل ماتریسی معادلات (۲'۱) را به صورت زیر در نظر می‌گیریم

$$F(t, x) = EU_{\epsilon}(t, x) + \int_0^t \int_{\Omega} K(t, \eta, x, s) U_{\epsilon}(\eta, s) ds d\eta, \quad (2'2)$$

به طوریکه

$$E = \begin{bmatrix} 1 & 0 \\ 0 & \epsilon \end{bmatrix}, \quad U_{\epsilon}(x, t) = [u_1(x, t), u_{2,\epsilon}(x, t)]^T.$$

تعریف ۱. [۵] یک اپراتور خود الحاق $k : H \rightarrow H$ که H یک فضای هیلبرت حقیقی است، coercive نامیده می‌شود اگر یک ثابت $c > 0$ وجود داشته باشد به‌طوری‌که

$$\langle kx, x \rangle \geq c \|x\|^2, \quad \forall x \in H.$$



لم ۲. [۵] فرض کنید که اپراتور انتگرال سیستم (۲۲) در فضای هیلبرت پیوسته و *coercive* باشد که U و U_ϵ تعریف شده اند، بنابراین:

$$1 - \|U_\epsilon\| \text{ مستقل از } \epsilon \text{ کراندار است}$$

$$2 - \|U_\epsilon - U\| \text{ به صفر میل میکند هرگاه } \epsilon \rightarrow 0$$

قضیه ۳. [۵] فرض کنید که

$$1 - F \in C(I \times \Omega)$$

$$2 - D = (t, \eta), 0 \leq \eta \leq t \leq T \text{ که } K \in C(D \times \Omega^2)$$

پس سیستم مرکب (۲۲) دارای یک جواب منحصر بفرد $U_\epsilon \in C(I \times \Omega)$ است.

در ادامه روش منظم سازی تیخونوف را بررسی می کنیم.

یک روش متداول برای حل سیستم های خطی $Kx = y$ که در آن $K : X \rightarrow Y$ یک اپراتور خطی و کراندار است، تعیین بهترین تقریب به این مفهوم است که سعی می شود با توجه به $x \in X$ برای برخی از نرم ها در Y خطای $\|Kx - y\|$ مینیمم شود. با توجه به عملگر خطی و کراندار $K : X \rightarrow Y$ و $x_\epsilon, y \in Y$ را انتخاب می کنیم که تابع تیخونوف زیر را مینیموم کند

$$J_\epsilon(x) = \|Kx - y\|^2 + \epsilon \|x\|^2, \quad \forall x \in X.$$

برای این منظور قضیه زیر را در نظر می گیریم:

قضیه ۴. فرض کنید $K : X \rightarrow Y$ یک اپراتور خطی و کراندار بین فضاهای هیلبرت X و Y باشد و $\epsilon > 0$ پس تابع تیخونوف J_ϵ یک مینیموم منحصر بفرد $x_\epsilon \in X$ دارد این مینیمم x_ϵ جواب منحصر بفرد معادله ی نرمال زیر است:

$$\epsilon x_\epsilon + K^* K x_\epsilon = K^* y. \quad (2.3)$$

به طوریکه $K^* : Y \rightarrow X$ اپراتور الحاقی K است. [۴]

با استفاده از قضیه ی ۴ سیستم (۱۱) با $I = \Omega = [0, 1]$ به شکل نرمال زیر می نویسیم:

$$\left\{ \begin{aligned} f(p, q) &= u_1(p, q) + \int_0^p \int_0^1 k_{11}(p, \eta, q, s) u_1(\eta, s) ds d\eta \\ &\quad + \int_0^p \int_0^1 k_{12}(p, \eta, q, s) u_{2,\epsilon}(\eta, s) ds d\eta \\ \int_p^1 \int_0^1 k_{22}(p, t, q, x) g(t, x) dx dt &= \epsilon u_{2,\epsilon}(p, q) \\ &\quad + \int_p^1 \int_0^1 \int_0^t \int_0^1 k_{22}(p, t, q, x) k_{21}(t, \eta, x, s) u_1(\eta, s) ds d\eta dx dt \\ &\quad + \int_p^1 \int_0^1 \int_0^t \int_0^1 k_{22}(p, t, q, x) k_{22}(t, \eta, x, s) u_{2,\epsilon}(\eta, s) ds d\eta dx dt. \end{aligned} \right. \quad (2.4)$$

در ادامه با استفاده از روش هم محلی بر اساس پایه های چند ترازى جواب سیستم (۲۴) را تقریب می زنیم.



۳ روش هم محلی بر اساس پایه های چند ترازی

یک دنباله از افزایهای دامنه‌ی Ω که هر قسمت در دنباله از اصلاح (پالایش) افزاز قبل از آن بدست می‌آید را چند ترازی می‌نامند. فضای جواب معادله‌ی انتگرالی، چند جمله‌ای‌های تکه‌ای روی افزایهای چند ترازی خواهد بود. ابتدا نحوه تقسیم‌بندی‌های چند ترازی یک مجموعه‌ی ثابت Ω در فضای اقلیدسی d -بعدی R^d را بررسی می‌کنیم. برای شروع یک عدد صحیح مثبت μ و یک خانواده $\Phi = \{\phi_e : e \in Z_\mu\}$ از نگاشت‌های تطبیق‌پذیر روی R^d را در نظر می‌گیریم. یک مجموعه ثابت منحصر بفرد Ω از R^d همراه با خانواده‌ی نگاشت‌های Φ وجود دارد که

$$\Phi(\Omega) = \Omega, \quad (3.1)$$

به‌طوریکه

$$\Phi(\Omega) := \bigcup_{e \in Z_\mu} \Phi_e(\Omega).$$

که $Z_\mu = \{0, 1, \dots, \mu-1\}$ ما همچنین محدودیت‌های زیر را بر روی نگاشت Ω اعمال می‌کنیم:

۱. برای هر $e \in Z_\mu$ نگاشت ϕ_e بر روی Ω پیوسته است.

۲. مجموعه‌ی Ω تهی نیست و

$$(\phi_e(\Omega) \cap \phi_{e'}(\Omega)) = \emptyset, \quad e, e' \in Z_\mu, e \neq e'.$$

ما از Φ برای بدست آوردن یک دنباله از تقسیم‌بندی‌های چند ترازی $\{\Omega_n : n \in N_0\}$ استفاده می‌کنیم. برای هر $e = (e_j : j \in Z_n)$ نگاشت:

$$\phi_e = \phi_{e_n} \circ \phi_{e_{n-1}} \circ \dots \circ \phi_{e_1},$$

و عدد

$$\mu(e) = \mu^{n-1}e_0 + \dots + \mu e_{n-2} + e_{n-1}.$$

را تعریف می‌کنیم. حال هر $i \in Z_{\mu^n}$ را می‌توانیم به صورت منحصر بفرد $i = \mu(e)$ برای $e \in Z_{\mu^n}$ بنویسیم. از معادله‌ی (۳.۱) و شرایط (۱) و (۲)، داریم

$$\Omega_n = \{\Omega_{n,e} : \Omega_{n,e} = \phi_e(\Omega), e \in Z_{\mu^n}\}.$$

چند نمونه از تقسیم بندی های مهم وقتی که $d = 1$ و $\Omega = [0, 1]$ ، به صورت زیر می باشد [۳]

$$\begin{aligned} a) \quad k = 2, \quad \mu = 2, \quad \phi_e(t) = \frac{t+e}{2}, i \in \Omega, \quad e = 0, 1, \quad t_i = \frac{i+1}{3}, \quad i = 0, 1 \\ b) \quad k = 3, \quad \mu = 2, \quad \phi_e(t) = \frac{t+e}{2}, i \in \Omega, \quad e = 0, 1, \quad t_i = \frac{2^i}{3}, \quad i = 0, 1, 2 \\ c) \quad k = 3, \quad \mu = 3, \quad \phi_e(t) = \frac{t+e}{3}, i \in \Omega, \quad e = 0, 1, 2, \quad t_i = \frac{i+1}{4}, \quad i = 0, 1, 2 \end{aligned}$$

در ادامه به بررسی روش هم محلی بر اساس پایه های چند تراز می پردازیم.

فرض کنیم $N = 1, 2, \dots$ و $N_0 = N \cup \{0\}$ به دنبال زیر فضاهای (چند تراز) $\{X_n : n \in N_0\}$ که فضاهای چند جمله ای های تکه ای از درجه حداکثر $l - 1$ می باشند، هستیم. این زیر فضاها تجزیه چند تراز به صورت زیر دارند

$$X_n = W_0 \oplus^\perp W_1 \oplus^\perp \dots \oplus^\perp W_n, \quad (3.2)$$

به طوریکه $W_0 := X_0$. توجه کنید که $A \oplus^\perp B$ یعنی حاصل جمع مستقیم دو فضای A و B به طوریکه $A \perp B$. زمانی که W_1 داده شده است، فضاهای W_i می توانند به صورت بازگشتی ساخته شوند. ساخت چنین فضاهایی را می توان در [۱] و [۲] و [۶] دید. برای هر $i \in N_0$ ما از $w(i)$ برای نشان دادن ابعاد فضای W_i استفاده می کنیم. فرض کنیم W_i به طوریکه :

$$W_i = \text{span}\{w_{ij} : j \in Z_{w(i)}\},$$

به طوریکه $Z_n := \{0, 1, \dots, n-1\}$ برای هر $n \in N$ تعریف می کنیم $U_n := \{(i, j) : i \in Z_{n+1}, j \in Z_{w(i)}\}$ بنابراین از (۳.۲) داریم:

$$W_i = \text{span}\{w_{ij} : (i, j) \in U_n\}$$

برای هر $(i, j) \in U_n$ ، S_{ij} را بعنوان تکیه گاه توابع پایه ای w_{ij} تعریف می کنیم. همچنین $d(A)$ را قطر مجموعه A در نظر می گیریم که $A \subset \Omega$ برای هر $i \in Z_{n+1}$ ، $s(n)$ را به صورت $s(n) = \dim X_n$ تعریف کرده و قرار می دهیم $h_i = \max\{d(S_{i,j})\}$ به طوریکه $j \in Z_{w(i)}$. همچنین به ویژگی های مربوط به فضاها و پایه ها بصورت $s(n) \sim \mu^n$ ، $w(i) \sim \mu^i$ و $h_i \sim \mu^{i/d}$ به طوریکه $\mu > 1$ یک عدد صحیح است، نیاز داریم. همچنین یک ثابت مثبت c وجود دارد که به ازای همه $(i, j) \in U_n$ داریم $\|w\|_\infty \leq c$ برای جزئیات بیشتر در مورد این ویژگی ها برای فضاها و پایه های ساخته شده در آنها به مرجع [۱] رجوع شود.

همراه هر تابع پایه ای w_{ij} یک تابع هم محلی l_{ij} وجود دارد که یک مجموع از نقاط تابع هم محلی در تعداد ثابتی از نقاط در S_{ij} است. فرض کنیم $\langle l_{ij}, w \rangle$ نشان دهنده عملکرد تابع خطی l_{ij} در تابع w است. برای هر $(i, j) \in U_n$ با $i > 0$ داریم

$$\langle l_{ij}, q \rangle = 0, \quad \forall q \in \Pi_{l-1}.$$

به طوریکه Π_{l-1} فضای چند جمله ای های حداکثر از درجه $l - 1$ ، همچنین توابع پایه ای و توابع هم محلی نظیر آنها دارای ویژگی نیمه متعامد هستند یعنی

$$\langle l_{i'j'}, w_{ij} \rangle = \delta_{i'i} \delta_{j'j}, \quad (i, j), (i', j') \in U, \quad i' \leq i.$$

حال برای هر W_i ، فضای توابع هم‌محلی را به صورت زیر داریم

$$L_n = V_0 \oplus V_1 \oplus \dots \oplus V_n = \text{span}\{l_{ij} : (i, j) \in U_n\}. \quad (3.3)$$

اکنون روش هم‌محلی را برای دستگاه (2.4) فرمول بندی می‌کنیم. ما جواب $u(t, x)$ را به صورت زیر می‌نویسیم

$$u(t, x) = \sum_{(i,j)} u_{i,j} v_{i',j'} w_{i,j}(t) w_{i',j'}(x), \quad (i, j), (i', j') \in U_n. \quad (3.4)$$

بنابر (3.4) برای u_1 و $u_{2,\epsilon}$ از دستگاه (2.4) داریم:

$$u_1(t, x) = \sum_{(i,j)} \sum_{(i',j')} u_{i,j} v_{i',j'} w_{i,j}(t) w_{i',j'}(x), \quad (i, j), (i', j') \in U_n, \quad (3.5)$$

$$u_{2,\epsilon}(t, x) = \sum_{(i,j)} \sum_{(i',j')} u'_{i,j} v'_{i',j'} w_{i,j}(t) w_{i',j'}(x), \quad (i, j), (i', j') \in U_n.$$

با جایگذاری (3.5) در سیستم (2.4) داریم

$$\left\{ \begin{aligned} f(l_{rm}, l_{r'm'}) &= \sum_{(i,j)} \sum_{(i',j')} u_{i,j} v_{i',j'} w_{i,j}(l_{rm}) w_{i',j'}(l_{r'm'}) \\ &+ \sum_{(i,j)} \sum_{(i',j')} \int_0^{l_{rm}} \int_0^1 k_{11}(l_{rm}, \eta, l_{r'm'}, s) u_{i,j} v_{i',j'} w_{i,j}(\eta) w_{i',j'}(s) ds d\eta \\ &+ \sum_{(i,j)} \sum_{(i',j')} \int_0^{l_{rm}} \int_0^1 k_{12}(l_{rm}, \eta, l_{r'm'}, s) u'_{i,j} v'_{i',j'} w_{i,j}(\eta) w_{i',j'}(s) ds d\eta \\ f(l_{rm}, l_{r'm'}) &= \int_0^{l_{rm}} \int_0^1 k_{22}(l_{rm}, t, l_{r'm'}, x) g(t, x) dx dt = \epsilon \sum_{(i,j)} \sum_{(i',j')} u'_{i,j} v'_{i',j'} w_{i,j}(l_{rm}) w_{i',j'}(l_{r'm'}) \\ &+ \sum_{(i,j)} \sum_{(i',j')} \int_0^{l_{rm}} \int_0^1 \int_0^t \int_0^1 k_{21}(l_{rm}, t, l_{r'm'}, x) k_{21}(t, \eta, x, s) u_{i,j} v_{i',j'} w_{i,j}(\eta) w_{i',j'}(s) ds d\eta dx dt \\ &+ \sum_{(i,j)} \sum_{(i',j')} \int_0^{l_{rm}} \int_0^1 \int_0^t \int_0^1 k_{22}(l_{rm}, t, l_{r'm'}, x) k_{22}(t, \eta, x, s) u'_{i,j} v'_{i',j'} w_{i,j}(\eta) w_{i',j'}(s) ds d\eta dx dt, \end{aligned} \right.$$

به‌طوری‌که $(r, m), (r', m') \in U_n$ و $\langle \delta_t, f \rangle = f(t)$ از حل دستگاه فوق چهار بردار $u_{i,j}$ و $v_{i',j'}$ و $u'_{i,j}$ و $v'_{i',j'}$ به دست می‌آید

که با جایگذاری در (3.5) به ترتیب جواب تقریبی u_1 و $u_{2,\epsilon}$ حاصل می‌شود.

مراجع

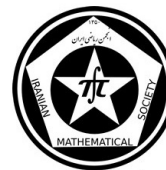
- [1] Z. Chen, S. Ding, H. Yang, *Multilevel augmentation algorithms based on fast collocation methods for solving ill-posed integral equations*, Computers Mathematics with Applications 62.4 (2011), 2071–2082.



- [2] Z. Chen, C.A. Micchelli, Y. Xu, *Fast collocation methods for second kind integral equations*, SIAM J. Numer. Anal. 40 (2002), 344–375.
- [3] X. Chen, Z. Chen, B. Wu, *Multilevel augmentation methods with matrix compression for solving reformulated Hammerstein equations*, The Journal of Integral Equations and Applications (2012), 513–544.
- [4] C.W. Groetsch, *The Theory of Tikhonov Regularization for Fredholm Equations*, 104p, Boston Pitman Publication, (1994)
- [5] S. Pishbin, J. Shokri, *Regularization technique and numerical analysis of the mixed system of first and second-kind Volterra–Fredholm integral equations*, Iranian Journal of Numerical Analysis and Optimization Vol. 9, No. 1, (2019), pp 127–150
- [6] Y. Wang, Y. Xu, *A fast wavelet collocation method for integral equations on polygons*, J. Integral Equations Appl, 17(2005), 277–330.

پست الکترونیکی: st_m.hoseiniazar@urmia.ac.ir

پست الکترونیکی: s.pishbin@urmia.ac.ir



روش‌های هم‌محلی چندگامی برای حل معادلات انتگرال ولترای اتو-کانولوشن نوع دوم

اعظم عبادی [†] ، پرویز دارانیا و سعید پیش بین [‡]

[†] دانشکده علوم، گروه ریاضی، دانشگاه ارومیه، ارومیه، ایران

[‡] دانشکده علوم، گروه ریاضی، دانشگاه ارومیه، ارومیه، ایران

چکیده

در این مقاله ابتدا معادله انتگرال ولترای اتو-کانولوشن تعمیم یافته را معرفی می‌کنیم. در ادامه روش هم‌محلی چندگامی را به عنوان یک روش عددی برای حل این نوع معادله در نظر گرفته و همگرایی روش را به صورت عددی تحلیل می‌کنیم. در روش عددی مورد نظر جواب تقریبی معادله به $r - 1$ گام قبل از خود وابسته بوده بطوریکه این $r - 1$ گام را در صورت نبود مقدار جواب با استفاده از روش‌های تک‌گامی هم‌مرتبه با روش چندگامی محاسبه می‌کنیم. همچنین روش هم‌محلی مورد نظر، به صورت تکه‌ای جواب تقریبی را در هر زیر بازه بر اساس نقاط هم‌محلی مشخص می‌کند بطوریکه انتخاب مناسب نقاط هم‌محلی می‌تواند خطا را کاهش و دقت روش را افزایش دهد. در پایان با ارائه نتایج عددی صحت و اعتبار این روش را تصدیق می‌نماییم. واژه‌های کلیدی: آنالیز همگرایی، روش‌های هم‌محلی چندگامی، معادلات انتگرال ولترای اتو-کانولوشن. طبقه‌بندی موضوعی [۲۰۱۰]: 65R20, 65Q20, 45D05.

۱ پیش‌گفتار

معادله انتگرال ولترای اتو-کانولوشن معمولی

$$u(t) := g(t) + \lambda \int_0^t u(t-s)u(s)ds, \quad t \in I := [0, T], \lambda \in R \quad (1.1)$$

به صورت گسترده توسط افراد مختلف مطالعه شده است (به عنوان نمونه به مراجع [۱، ۲، ۷] مراجعه شود). اخیراً معادلات غیراستاندارد ولترا مورد توجه قرار گرفته است بطوریکه گوان و همکارانش [۴] روش هم‌محلی را برای معادله

$$u(t) := g(t) + \int_0^t k(t, s, u(t), u(s))ds, \quad t \in I := [0, T]$$



بررسی کرده‌اند. در این مقاله به عنوان نوع خاصی از معادلات غیراستاندارد غیرخطی ولترا، معادله انتگرال اتو-کانولوشن تعمیم یافته بفرم:

$$u(t) := g(t) + \int_0^t k(t, s)u(t-s)u(s)ds, \quad t \in I := [0, T] \quad (1'2)$$

را در نظر می‌گیریم که در آن $D := \{(t, s) | 0 \leq s \leq t \leq T\}$ و $k \in C(D)$ و $g \in C(I)$ توابعی معلوم و $u(t)$ تابع مجهول می‌باشد. معادلات انتگرال ولترای اتو-کانولوشن نقش مهمی را در توصیف بسیاری از پدیده‌های فیزیکی بازی می‌کنند، از جمله کاربردها می‌توان به مشخص کردن حافظه هسته در تئوری ویسکوالاستیک [۵] و محاسبه برخی توابع خاص مانند توابع میتاگ-لفلر [۶] اشاره کرد. زانگ و همکارانش [۸] قضیه وجود و یکتایی مربوط به معادله (۱'۲) را بررسی کرده و درجه همواری جواب معادله را بر اساس همواری توابع معلوم مشخص نمودند. همچنین روش هم محلی براساس چندجمله‌ایهای تکه‌ای را به عنوان یک روش تک‌گامی برای حل تقریبی معادله (۱'۲) در نظر گرفته و همگرایی و ابرهمگرایی روش را به طور کامل بررسی نمودند. در این مقاله روش هم محلی چندگامی را برای حل این نوع معادله در نظر گرفته و با انتخاب مناسب نقاط هم محلی تضمینی برای همگرایی روش و افزایش دقت آن را داریم.

۲ روش هم محلی چندگامی

در این بخش ابتدا قضیه وجود و یکتایی معادله (۱'۲) را بیان نموده و در ادامه درباره روش هم محلی چندگامی بحث خواهیم کرد.

قضیه ۱. [۸] فرض کنید توابع معلوم $g \in C^m(I)$, $k \in C^m(D)$ در معادله (۱'۲) برای $m \geq 1$ صدق کنند. آنگاه معادله انتگرال اتو-کانولوشن نوع دوم (۱'۲) یک جواب یکتای $u \in C^m(I)$ دارد.

حال می‌خواهیم در ادامه روش عددی هم محلی چندگامی را برای معادله (۱'۲) در نظر بگیریم. برای این منظور، فرض کنید

$$I_h := \{t_n := nh, n = 0, 1, \dots, N (t_n := T)\}, \quad (2'1)$$

یک افراز یکنواخت بر بازه‌ی داده شده I را نمایش دهد. همچنین برای $n = 0, 1, \dots, N-1$ نمادهای

$$\sigma_n := (t_n, t_{n+1}), \quad h_n := t_{n+1} - t_n, \quad h = \max\{h_n : 0 \leq n \leq N-1\}. \quad (2'2)$$

را در نظر گرفته و برای نقاط هم محلی از پیش تعیین شده

$$X_h := \{t_{n,i} = t_n + c_i h_n : 0 \leq c_1 < c_2 < \dots < c_m \leq 1 (n = 0, 1, \dots, N-1)\},$$

جواب تقریبی u_h توسط معادله‌ی هم محلی

$$u_h(t) := g(t) + \int_0^t k(t, s)u_h(t-s)u_h(s)ds, \quad t \in X_h, \quad (2'3)$$



برای (۱۲) بدست می آید. با جایگذاری نقاط هم محلی $t_{n,i} = t_n + c_i h_n \in [t_n, t_{n+1}]$ بجای t در معادله (۲۳) خواهیم داشت:

$$\begin{aligned} u_h^n(t_{n,i}) &= g(t_{n,i}) + \int_0^{t_{n,i}} K(t_{n,i}, s) u_h(t_{n,i} - s) u_h(s) ds \\ &= g(t_{n,i}) + \int_{t_n}^{t_{n,i}} K(t_{n,i}, s) u_h^n(t_{n,i} - s) u_h^n(s) ds \\ &\quad + \sum_{k=0}^{n-1} \int_{t_k}^{t_{k,i}} K(t_{n,i}, s) u_h^{n-k}(t_{n,i} - s) u_h^k(s) ds \\ &\quad + \sum_{k=0}^{n-1} \int_{t_{k,i}}^{t_{k+1}} K(t_{n,i}, s) u_h^{n-k-1}(t_{n,i} - s) u_h^k(s) ds. \end{aligned} \quad (24)$$

در روش هم محلی چندگامی که به $r-1$ گام اولیه وابسته می باشند، معادله را به بازه $[t_n, t_{n+1}]$ محدود می کنیم، لذا داریم:

$$u_n(t_n + sh) = \sum_{k=0}^{r-1} \varphi_k(s) y_{n-k} + \sum_{j=1}^m \psi_j(s) U_{n,j}, \quad s \in [0, 1], \quad n = r, \dots, N-1. \quad (25)$$

که در آن

$$U_{n,j} := u_n(t_{n,j}), \quad (26)$$

و $\varphi_k(s)$ و $\psi_j(s)$ چند جمله ای از درجه $m+r-1$ می باشند، که با اعمال شرایط درونیاب در نقاط t_{n-k} یعنی $y_{n-k} := u_n(t_{n-k})$ تعیین می شوند. مقادیر اولیه $y_1, \dots, y_{r-1}$ مورد نیاز در (۲۵) ممکن است با یک روش تک گامی هم مرتبه با روش چندگامی حاصل شوند.

شرایط درونیاب در نقاط t_{n-k} ، $k = 0, 1, \dots, r-1$ با شرط (۲۶) دستگاه خطی زیر را بوجود می آورد.

$$\begin{aligned} \varphi_l(-k) &= \delta_{lk}, \quad \varphi_l(c_j) = 0, \quad l, k = 0, \dots, r-1 \\ \psi_i(-k) &= 0, \quad \psi_i(c_j) = \delta_{ij}, \quad i, j = 1, \dots, m. \end{aligned} \quad (27)$$

رابطه (۲۷) یک دستگاه خطی متشکل از $(r+m)^2$ معادله که در آن $(r+m)^2$ مجهول، ضربایی از چند جمله ای های $\varphi_k(s)$ و $\psi_j(s)$ است، می باشد. و $m, \dots, j = 1, c_j$ پارامترهای ثابت فرض می شوند.



نکته ۲. فرض کنید $c_i \neq c_j$ و $c_1 \neq 0$ باشد. پس جواب یکتای دستگاه (۲۰۷) به فرم

$$\begin{aligned}\varphi_k(s) &= \prod_{i=1}^m \frac{s - c_i}{-k - c_i} \cdot \prod_{i=0, i \neq k}^{r-1} \frac{s + i}{-k + i}, \\ \psi_j(s) &= \prod_{i=0}^{r-1} \frac{s + i}{c_j + i} \cdot \prod_{i=1, i \neq j}^m \frac{s - c_i}{c_j - c_i}.\end{aligned}\quad (208)$$

فرض می شود.

نکته ۳. واضح است معادله (۲۰۴) فقط برای $n = 0$ غیرخطی بوده و برای $n = 1, \dots, N-1$ یک معادله خطی می باشد. از طرفی دیگر در روش چندگامی چون $n = r \dots N-1$ می باشد در نتیجه معادله مورد نظر به صورت خطی می باشد در حالی که در روش تک گامی معادله به دو صورت خطی و غیرخطی بررسی می شود.

$$\begin{aligned}U_{n,i} &= g(t_{n,i}) + \int_{t_n}^{t_{n,i}} K(t_{n,i}, s) u_h^n(t_{n,i} - s) u_h^n(s) ds \\ &\quad + \sum_{l=0}^{n-1} \int_{t_l}^{t_{l,i}} K(t_{n,i}, s) u_h^{n-l}(t_{n,i} - s) u_h^l(s) ds \\ &\quad + \sum_{l=0}^{n-1} \int_{t_{l,i}}^{t_{l+1}} K(t_{n,i}, s) u_h^{n-l-1}(t_{n,i} - s) u_h^l(s) ds \\ &= g(t_{n,i}) + h \int_0^{c_i} K(t_{n,i}, t_n + sh) u_h^n((c_i - s)h) u_h^n(t_n + sh) ds \\ &\quad + h \int_0^{c_i} K(t_{n,i}, sh) u_h^n(t_n + (c_i - s)h) u_h^n(sh) ds \\ &\quad + h \sum_{l=1}^{n-1} \int_0^{c_i} K(t_{n,i}, t_l + sh) u_h^{n-l}(t_{n-l} + (c_i - s)h) u_h^l(t_l + sh) ds \\ &\quad + h \sum_{l=0}^{n-1} \int_{c_i}^1 K(t_{n,i}, t_l + sh) u_h^{n-l-1}(t_{n-l} + (c_i - s)h) u_h^l(t_l + sh) ds\end{aligned}\quad (209)$$



با جایگذاری رابطه (۲۰۵) در رابطه (۲۰۹)، دستگاه خطی زیر

$$\begin{aligned}
 [I - h\bar{B}_n]U_n &= G_n + h\tilde{B}_n Y_n + h \sum_{l=\imath}^{r-\imath} [\tilde{B}_n^{(l)} Y_{n-l} + \bar{B}_n^{(l)} U_{n-l}] \\
 &+ h \sum_{l=r-\imath}^{n-r+\imath} [(\tilde{C}_n^{(l)} + \bar{C}_n^{(l)}) Y_{n-l} + (\tilde{D}_n^{(l)} + \bar{D}_n^{(l)}) U_{n-l}] + h \sum_{l=n-r+\imath}^{n-\imath} [\tilde{E}_n^{(l)} Y_l + \bar{E}_n^{(l)} U_l] \\
 &+ h \sum_{l=\circ}^{r-\imath} [(\tilde{B}_n^{(l)}) Y_{n-l-\imath} + (\bar{B}_n^{(l)}) U_{n-l-\imath}] + h \sum_{l=r-\imath}^{n-r} [((\tilde{C}_n^{(l)}) + (\bar{C}_n^{(l)})) Y_{n-l-\imath} + ((\tilde{D}_n^{(l)}) + (\bar{D}_n^{(l)})) U_{n-l-\imath}] \\
 &+ \sum_{l=n-r+\imath}^{n-\imath} [(\tilde{E}_n^{(l)}) Y_l + (\bar{E}_n^{(l)}) U_l]
 \end{aligned} \tag{۲۰۱۰}$$

حاصل می‌شود که در آن

$$Y_l := (y_l, \dots, y_{l-r+\imath})^T, U_l := (U_{l,\imath}, \dots, U_{l,m})^T, G_n := (g(t_{n,\imath}), \dots, g(t_{n,m}))^T,$$

و

$$(\tilde{B}_n)_{i,k} := \int_{\circ}^{c_i} [K(t_{n,i}, t_n + sh) u_h^{\circ}((c_i - s)h)(\varphi_k(s)) + K(t_{n,i}, sh) \varphi_k(c_i - s) u_h^{\circ}(sh)] ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

$$(\bar{B}_n)_{i,j} := \int_{\circ}^{c_i} [K(t_{n,i}, t_n + sh) u_h^{\circ}((c_i - s)h)(\psi_j(s)) + K(t_{n,i}, sh)(\psi_j(c_i - s)) u_h^{\circ}(sh)] ds,$$

$$(i, j = \imath, \dots, m),$$

$$(\tilde{B}_n^{(l)})_{i,k} := \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\varphi_k(c_i - s)) u_h^l(t_l + sh) ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

$$(\bar{B}_n^{(l)})_{i,j} := \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\psi_j(c_i - s)) u_h^l(t_l + sh) ds,$$

$$(i, j = \imath, \dots, m),$$



$$(\tilde{C}_n^{(l)})_{i,k} := \sum_{p=\circ}^{r-\imath} \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\varphi_k(c_i - s))\varphi_p(s)y_{l-p}ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

$$(\bar{C}_n^{(l)})_{i,k} := \sum_{j=\imath}^m \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\varphi_k(c_i - s))\psi_j(s)U_{l,j}ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

$$(\bar{D}_n^{(l)})_{i,j} := \sum_{k=\circ}^{r-\imath} \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\psi_j(c_i - s))(\varphi_k(s))y_{l-k}ds,$$

$$(i, j = \imath, \dots, m),$$

$$(\bar{D}_n^{(l)})_{i,j} := \sum_{q=\imath}^m \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(\psi_j(c_i - s))(\psi_q(s)U_{l,q})ds,$$

$$(i, j = \imath, \dots, m),$$

$$(\tilde{E}_n^{(l)})_{i,k} := \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(u_h^{n-l}(t_{n-l} + (c_i - s)h))(\varphi_k(s))ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

$$(\bar{E}_n^{(l)})_{i,j} := \int_{\circ}^{c_i} K(t_{n,i}, t_l + sh)(u_h^{n-l}(t_{n-l} + (c_i - s)h))(\psi_j(s))ds,$$

$$(i, j = \imath, \dots, m),$$

همچنین

$$(\underline{\tilde{B}_n^{(l)}})_{i,k} := \int_{c_i}^{\imath} K(t_{n,i}, t_l + sh)(\varphi_k(\imath + c_i - s))u_h^l(t_l + sh)ds,$$

$$(i = \imath, \dots, m, \quad k = \circ, \dots, r - \imath),$$

و به همین ترتیب ماتریس‌های $(\underline{\tilde{B}_n^{(l)}})_{i,j}, (\underline{\tilde{C}_n^{(l)}})_{i,k}, \dots, (\underline{\bar{E}_n^{(l)}})_{i,j}$ بطور مشابه تعریف می‌شود. از حل دستگاه خطی $(\mathbf{2}^{\circ}\mathbf{1}^{\circ})$ بردار مجهول U_n حاصل می‌شود که با جایگذاری آن در معادله $(\mathbf{2}^{\circ}\mathbf{5})$ جواب تقریبی معادله اتو-کانولوشن نوع دوم حاصل می‌شود.

۳ نتایج عددی

در این بخش از مقاله، معادله (۱۰۲) را با استفاده از روش هم‌محلی چندگامی حل می‌کنیم. همچنین دقت و مرتبه همگرایی این روش را با روش هم‌محلی پیشنهاد شده در مقاله [۸] مقایسه می‌کنیم. تمامی محاسبات با استفاده از نرم‌افزار ممتیکا انجام شده است.

مثال ۴. معادله انتگرال اتو-کانولوشن نوع دوم بفرم:

$$u(t) = \beta t e^{(-\gamma t)} - \beta^\gamma e^{(-\gamma t)}(t + t \cos t - \gamma \sin t) - \int_0^t k(t, s) u(t-s) u(s) ds, \quad t \in [0, 1], \quad (3.1)$$

را با $k(t, s) = \cos(t-s)$ در نظر بگیرید. مقادیر اولیه u_1, u_2 جواب دقیق معادله، $u(t) = \beta t e^{-\gamma t}$ می‌باشد. فرض کنید $\beta = \gamma = 1$ باشد و همچنین نقاط هم‌محلی را $c_1 = 0.7$ و $c_2 = 1$ در نظر بگیرید. جدول ۱ مقدار ماکزیمم خطا و مرتبه همگرایی را برای N های مختلف برای $r = 3$ و $m = 2$ نشان می‌دهد. ملاحظه می‌شود با افزایش N مقدار خطا کاهش و دقت روش افزایش می‌یابد. همچنین ملاحظه می‌شود که روش هم‌محلی چندگامی از دقت بالاتری نسبت به روش هم‌محلی تک‌گامی برخوردار است.

جدول ۱: بررسی خطا و همگرایی جواب برای N مختلف

N	ماکزیمم خطا در روش هم‌محلی چندگامی	مرتبه همگرایی	ماکزیمم خطا [۸]	مرتبه همگرایی [۸]
۴	1.07×10^{-7}	۲.۸۸	2.31×10^{-3}	۱.۹۴
۸	1.45×10^{-8}	۴.۴۱	6.02×10^{-4}	۱.۹۷
۱۶	6.79×10^{-10}	۴.۷۷	1.52×10^{-4}	۱.۹۹

مراجع

- [1] A.L. Bukhgeim, Inverse problems of memory reconstruction, J. Inverse Ill-Posed Probl., 1 (1993), pp. 193–205.
- [2] P.L. Butzer, The application of Jan Mikusi'nski's operator calculus to integral equations of convolution type, Arch. Ration. Mech. Anal., 2 (1958), pp. 114–128.
- [3] Brunner, H.: Collocation Methods for Volterra Integral and Related Functional Differential Equations. Cambridge University Press, Cambridge (2004)
- [4] . Guan, Q., Zhang, R., Zou, Y.: Analysis of collocation solutions for nonstandard Volterra integral equations. IMA J. Numer. Anal. 32(4), 1755–1785 (2011)

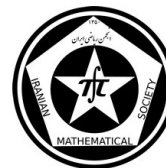


- [5] L. von Wolfersdorf, Einige Klassen quadratischer Integralgleichungen [Some classes of quadratic integral equations], Sitz.ber. S $\ddot{a}$ chs. Akad. Wiss. Leipz. Math.-Nat.wiss. Kl., 2000.
- [6] L. von Wolfersdorf, Autoconvolution equations and special functions, Integral Transforms Spec. Funct., 21 (2010), pp. 295–306.
- [7] L. von Wolfersdorf, Einige Klassen quadratischer Integralgleichungen [Some classes of quadratic integral equations], Sitz.ber. S $\ddot{a}$ chs. Akad. Wiss. Leipz. Math.-Nat.wiss. Kl., 2000
- [8] Zhang, R., Liang, H., Brunner, H.: Analysis of collocation methods for generalized auto-convolution Volterra integral equations. SIAM J. Numer. Anal. 54(2), 899–920 (2016)

پست الکترونیکی: azamebadi172@gmail.com

پست الکترونیکی: p.darania@urmia.ac.ir

پست الکترونیکی: s.pishbin@urmia.ac.ir



روش هم محلی بر اساس چند جمله ایهای تکه ای برای حل معادلات جبری انتگرال تاخیری

نیمه صریح

مریم شیخی[†] و سعید پیش بین[‡]

دانشکده علوم، گروه ریاضی، دانشگاه ارومیه، ارومیه، ایران[†]

دانشکده علوم، گروه ریاضی، دانشگاه ارومیه، ارومیه، ایران[‡]

چکیده

در این مقاله ابتدا معادله جبری انتگرال تاخیری نیمه صریح با تاخیر غیر صفر شونده را معرفی کرده و بر اساس مفهوم اندیس دیفرانسیلی قضیه وجود و یکتایی جواب را برای این دسته از معادلات مطرح می کنیم. جواب این نوع معادلات دارای ناپیوستگی های اولیه بوده که شناسایی این نقاط برای در نظر گرفتن یک روش عددی به شکل مناسب، ضروری است. در ادامه روش هم محلی بر اساس چند جمله ای های تکه ای با پایه لاگرانژ را برای حل این نوع معادله در نظر گرفته و در پایان با ارائه مثال های عددی صحت و اعتبار این روش را می سنجیم.

واژه های کلیدی: معادله جبری انتگرالی تاخیری، روش هم محلی، اندیس.

طبقه بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

دستگاه معادلات جبری انتگرال تاخیری نیمه صریح به فرم زیر می باشد

$$\left\{ \begin{array}{l} y(t) + \int_{t_0}^t k_{11}(t, s)y(s)ds + \int_{t_0}^{\theta(t)} \bar{k}_{11}(t, s)y(s)ds + \int_{t_0}^t k_{12}(t, s)z(s)ds \\ \quad + \int_{t_0}^{\theta(t)} \bar{k}_{12}(t, s)z(s)ds = f_1(t) \\ \\ \int_{t_0}^t k_{21}(t, s)y(s)ds + \int_{t_0}^{\theta(t)} \bar{k}_{21}(t, s)y(s)ds + \int_{t_0}^t k_{22}(t, s)z(s)ds \\ \quad + \int_{t_0}^{\theta(t)} \bar{k}_{22}(t, s)z(s)ds = f_2(t), \quad t \in (t_0, T], \end{array} \right. \quad (1'1)$$

و

$$y(t) = \phi_1(t), \quad z(t) = \phi_2(t), \quad t \in [\theta(t_0), t_0].$$

در این دستگاه تابع تاخیر θ در شرایط زیر صدق می کند:

$$\theta(t) = t - \tau(t), \quad \text{و } \tau(t) \geq \tau_0 > 0, \quad t \in I \quad (D1)$$

$$\theta \text{ بر } I \text{ اکیداً صعودی است} \quad (D2)$$

$$\tau(t) \in C^\nu(I), \quad \nu \geq 0 \quad (D3)$$

تابع تاخیر غیر صفر شونده $\theta(t)$ نقاط ناپیوستگی اولیه $\{\xi_\mu\}$ که از روابط بازگشتی زیر بدست می آیند را تعیین می کند.

$$\theta(\xi_\mu) = \xi_{\mu-1}, \quad \mu \geq 0, \quad \theta(t_0) = \xi_{-1}, \quad \xi_0 = t_0.$$

همچنین داریم

$$\xi_\mu - \xi_{\mu-1} = \tau(\xi_\mu) \geq \tau_0 > 0, \quad \mu \geq 0.$$

که این رابطه نشان میدهد نقاط ناپیوستگی اولیه در هر بازه کراندار متناهی هستند. معادلات انتگرال جبری تاخیری به عنوان کلاس خاصی از معادلات انتگرال ولترا تابعی نقش مهمی را در مدلسازی بسیاری از پدیده های فیزیکی و بیولوژیکی بازی می کنند (برای جزئیات بیشتر به مراجع [۱، ۲] مراجعه کنید) روش های عددی برای حل معادلات انتگرالی تاخیری توسط تعدادی از نویسندگان بررسی شده است [۲، ۳، ۴] اما بحث تحلیل عددی دستگاه (۱'۱) جدید بوده و نتایج کمی در این زمینه در دسترس می باشد. این مقاله به تحلیل عددی معادله جبری انتگرالی تاخیری نیمه صریح (۱'۱) پرداخته و مشکلات مربوط به ارائه یک روش عددی مناسب را با توجه به ساختار جواب دستگاه مشخص می کند.

۲ ساختار جواب معادلات جبری انتگرال تاخیری

در این بخش ابتدا ساختار جواب معادلات جبری انتگرال تاخیری را بررسی کرده و سپس قضیه وجود و یکتایی این دسته از معادلات را مطرح می کنیم. معادله دوم از دستگاه (۱'۱) را می توان به فرم موضعی زیر نوشت:

$$\int_{\xi_\mu}^t k_{\nu\nu}(t, s)z(s)ds = f_{\nu,\mu}(t), \quad t \in (\xi_\mu, \xi_{\mu+1}], \quad (2'1)$$

که

$$f_{\nu,\mu}(t) = f_\nu(t) - \int_{t_*}^{\xi_\mu} k_{\nu\nu}(t, s)z(s)ds - \int_{t_*}^{\theta(t)} \bar{k}_{\nu\nu}(t, s)z(s)ds - \int_{t_*}^t k_{\nu 1}(t, s)y(s)ds \\ - \int_{t_*}^{\theta(t)} \bar{k}_{\nu 1}(t, s)y(s)ds, \quad \mu \geq 1,$$

و برای هر $t \in (\xi_0, \xi_1]$ داریم:

$$f_{\nu,0}(t) = f_\nu(t) + \int_{\theta(t)}^{t_*} \bar{k}_{\nu\nu}(t, s)\phi_\nu(s)ds - \int_{t_*}^t k_{\nu 1}(t, s)y(s)ds + \int_{\theta(t)}^{t_*} \bar{k}_{\nu 1}(t, s)\phi_1(s)ds.$$

همچنین در $t_* = \xi_0$

$$f_{\nu,0}(t_*) = f_\nu(t_*) + \int_{\theta(t_*)}^{t_*} \bar{k}_{\nu\nu}(t, s)\phi_\nu(s)ds + \int_{\theta(t_*)}^{t_*} \bar{k}_{\nu 1}(t, s)\phi_1(s)ds \neq 0,$$

از قضیه ولترای کلاسیک [۲]: جواب (۲'۱) در $t = t_*^+$ بیگران است. جواب z از معادله (۲'۱) در نقطه $t_* = \xi_0$ کراندار خواهد بود اگر و تنها اگر رابطه زیر برقرار باشد:

$$\int_{\theta(t_*)}^{t_*} \bar{k}_{\nu\nu}(t, s)\phi_\nu(s)ds + \int_{\theta(t_*)}^{t_*} \bar{k}_{\nu 1}(t, s)\phi_1(s)ds = -f_\nu(t_*). \quad (2'2)$$

در معادله (۲'۱)، فرض کنید که $k_{\nu\nu}(t, t) \geq k_0 > 0$ برقرار باشد، پس داریم:

$$z(t) + k_{\nu\nu}^{-1}(t, t) \int_{\xi_\mu}^t \frac{\partial k_{\nu\nu}(t, s)}{\partial t} z(s)ds = k_{\nu\nu}^{-1}(t, t)f'_{\nu,\mu}(t), \quad (2'3)$$

مشاهده می کنیم که:

$$\lim_{t \rightarrow t_*^-} z(t) = \phi_\nu(t_*), \\ \lim_{t \rightarrow t_*^+} z(t) = k_{\nu\nu}^{-1}(t_*, t_*)f'_{\nu,0}(t_*),$$

بنابراین به طور کلی $z(t_*)^+ \neq \phi_\nu(t_*)$. با استدلال مشابه می توان نشان داد که نقطه ناپیوستگی z در $t = t_*$ منجر به این چنین ناپیوستگی در نقطه بعدی $\xi_1 = t_*$ می شود و نتیجه می گیریم که اگر ناپیوستگی نامتناهی در $t = t_*$ داشته باشد آنگاه ناپیوستگی در بقیه نقاط ξ_μ باقی می ماند. در ادامه اولین معادله از دستگاه (۱'۱) را در نظر گرفته و به طور مشابه ناپیوستگی اولیه مربوط به جواب $y(t)$ را مشخص می کنیم. حال به کمک استدلال بالا می توان قضیه وجود و یکتای جواب دستگاه (۱'۱) را مطرح کرد.



قضیه ۱. فرض کنید که تابع تاخیر θ در شرایط $D_3 - D_1$ صدق کند و همچنین شرایط زیر برقرار باشد:

$$\bar{k}_{11}(\cdot, \cdot), \bar{k}_{12}(\cdot, \cdot) \in C(D_\theta), k_{11}(\cdot, \cdot), k_{12}(\cdot, \cdot) \in C(D), \quad 1$$

$$\bar{k}_{21}(\cdot, \cdot), \bar{k}_{22}(\cdot, \cdot) \in C^1(D_\theta), k_{21}(\cdot, \cdot), k_{22}(\cdot, \cdot) \in C^1(D), \quad 2$$

$$|\det k_{22}(t, t)| \geq k_0 > 0, \quad t \in I, f_1 \in C(I), f_2 \in C^1(I), \quad 3$$

آنگاه برای هر تابع اولیه $\phi_1(t), \phi_2(t) \in C[\theta(t_0), t_0]$ دستگاه (۱۱) دارای جواب یکتای $(y, z)^T \in C(t_0, T]$ و $y \in C(t_0, T]$ و $z \in C(\xi_\mu, \xi_{\mu+1})$ ($\mu = 0, \dots, M$) اگر تابع ورودی به فرم زیر باشد جواب y در $t = t_0$ پیوسته است

$$f_1(t_0) + \int_{\theta(t_0)}^{t_0} \bar{k}_{11}(t_0, s) \phi_1(s) ds + \int_{\theta(t_0)}^{t_0} \bar{k}_{12}(t_0, s) \phi_2(s) ds = \phi_1(t_0),$$

و جواب z در $t = t_0$ کراندار باقی می ماند اگر و تنها اگر (۲۲) برقرار باشد.

۳ روش هم محلی تکه ای

در این بخش روش هم محلی بر اساس چند جمله ای تکه ای را برای حل تقریبی معادله (۱۱) در نظر می گیریم. فرض کنیم که T تعریف شده به صورت $I = [t_0, T]$ به گونه ای است که:

$$T = \xi_M \quad \text{و} \quad M \geq 1,$$

و

$$I_h := \bigcup_{\mu=0}^M I_h^{(\mu)}, \quad I_h^{(\mu)} := \{t_n^{(\mu)} : \xi_\mu = t_0^{(\mu)} < t_1^{(\mu)} < \dots < t_N^{(\mu)} = \xi_{\mu+1}\},$$

که $h_n^{(\mu)} = t_{n+1}^{(\mu)} - t_n^{(\mu)}, \mu = 0, \dots, M (M \geq 1)$ برای $[\theta(t_0), t_0]$ افراز را به فرم زیر در نظر می گیریم:

$$I_h^{(-1)} := \{t_n^{(-1)} : \theta(t_0) = t_0^{(-1)} < t_1^{(-1)} < \dots < t_N^{(-1)} = t_0\}$$

حال نقاط کالوکیشن را به صورت زیر در نظر می گیریم:

$$X_h := \bigcup_{\mu=0}^M X_h^{(\mu)},$$

که در آن داریم:

$$X_h^{(\mu)} = \{t_{n,i}^{(\mu)} = t_n^{(\mu)} + c_i h_n^{(\mu)} : 0 < c_1 < \dots < c_m \leq 1, (0 \leq n \leq N-1)\}.$$



معادله هم محلی از (۱۰۱) را به فرم زیر در نظر می گیریم:

$$\left\{ \begin{array}{l} u(t) = f(t) + \int_{t_*}^t k_{11}(t, s)u(s)ds + \int_{t_*}^t k_{12}(t, s)v(s)ds \\ \quad + \int_{t_*}^{\theta(t)} \hat{k}_{11}(t, s)u(s)ds + \int_{t_*}^{\theta(t)} \hat{k}_{12}(t, s)v(s)ds, \\ \\ \circ = g(t) + \int_{t_*}^t k_{21}(t, s)u(s)ds + \int_{t_*}^t k_{22}(t, s)v(s)ds \\ \quad + \int_{t_*}^{\theta(t)} \hat{k}_{21}(t, s)u(s)ds + \int_{t_*}^{\theta(t)} \hat{k}_{22}(t, s)v(s)ds, \quad t \in X_h, \end{array} \right. \quad (3.1)$$

که

$$u(t) = \phi(t), \quad v(t) = \varphi(t), \quad t \in [\theta(t_*), t_*],$$

و در هر زیر بازه $\sigma_n = (t_n^{(\mu)}, t_{n+1}^{(\mu)}]$ جواب های تقریبی هم محلی u, v به فرم زیر می باشد

$$u_n(t_n^{(\mu)} + \rho h_n^{(\mu)}) = \sum_{j=1}^m \psi_j(\rho) U_{n,j}^{(\mu)}, \quad (3.2)$$

$$v_n(t_n^{(\mu)} + \rho h_n^{(\mu)}) = \sum_{j=1}^m \psi_j(\rho) V_{n,j}^{(\mu)}, \quad \rho \in (0, 1], \quad (3.3)$$

که

$$\psi_j(\rho) = \prod_{\substack{i=1 \\ i \neq j}}^m \frac{\rho - c_i}{c_j - c_i},$$

و

$$U_{n,j}^{(\mu)} = u_n(t_{n,j}^{(\mu)}), \quad V_{n,j}^{(\mu)} = v_n(t_{n,j}^{(\mu)}), \quad n = 0, \dots, N-1.$$

فرض کنید

$$\begin{aligned} B_{npq}^{(l, \alpha, \beta)} &= \int_0^1 k_{pq}(t_{n,i}^{(\alpha)}, t_l^{(\beta)} + \rho h_l^{(\beta)}) \psi_j(\rho) d\rho, \\ D_{npq}^{(n, \alpha, \beta)} &= \int_0^{c_i} k_{pq}(t_{n,i}^{(\alpha)}, t_n^{(\beta)} + \rho h_n^{(\beta)}) \psi_j(\rho) d\rho, \\ \hat{B}_{npq}^{(l, \alpha, \beta)} &= \int_0^1 \hat{k}_{pq}(t_{n,i}^{(\alpha)}, t_l^{(\beta)} + \rho h_l^{(\beta)}) \psi_j(\rho) d\rho, \\ \hat{D}_{npq}^{(n, \alpha, \beta)} &= \int_0^{c_i} \hat{k}_{pq}(t_{n,i}^{(\alpha)}, t_n^{(\beta)} + \rho h_n^{(\beta)}) \psi_j(\rho) d\rho, \\ U_l^{(\alpha)} &= [U_{l,1}^{(\alpha)}, \dots, U_{l,m}^{(\alpha)}], \end{aligned}$$

$$V_l^{(\alpha)} = [V_{l,1}^{(\alpha)}, \dots, V_{l,m}^{(\alpha)}], \quad f_n^{(\mu)} = [f(t_{n,1}^{(\mu)}), \dots, f(t_{n,m}^{(\mu)})], \quad g_n^{(\mu)} = [g(t_{n,1}^{(\mu)}), \dots, g(t_{n,m}^{(\mu)})],$$



با جایگذاری $t_{n,i}^{(\mu)}$ و (3.2) و (3.3) در (3.1) ، یک دستگاه خطی برای بردارهای مجهول $U_n^{(\mu)}, V_n^{(\mu)}$ در دو حالت زیر بدست می آید.

حالت اول: اگر $t_{n,i}^{(\mu)} \leq t_0$ ، آنگاه:

$$\left\{ \begin{aligned}
 U_n^{(\mu)} &= f_n^{(\mu)} + \sum_{\eta=0}^{\mu-1} \sum_{l=0}^{N-1} h_l^{(\eta)} B_{n\backslash 1}^{(l,\eta+1,\eta)} U_l^{(\eta)} + B_{n\backslash 2}^{(l,\eta+1,\eta)} V_l^{(\eta)} \Big] \\
 &+ \sum_{l=0}^{n-1} h_l^{(\mu)} \left[B_{n\backslash 1}^{(l,\mu,\mu)} U_l^{(\mu)} + B_{n\backslash 2}^{(l,\mu,\mu)} V_l^{(\mu)} \right] \\
 &- h_n^{(-1)} \int_{c_i}^1 \hat{k}_{11}(t_{n,i}^{(\circ)}, t_n^{(-1)} + \rho h_n^{(-1)}) \phi(t_n^{(-1)} + \rho h_n^{(-1)}) d\rho \\
 &- h_n^{(-1)} \int_{c_i}^1 \hat{k}_{12}(t_{n,i}^{(\circ)}, t_n^{(-1)} + \rho h_n^{(-1)}) \varphi(t_n^{(-1)} + \rho h_n^{(-1)}) d\rho \\
 &- \sum_{l=n+1}^{N-1} h_l^{(-1)} \int_0^1 \hat{k}_{11}(t_{n,i}^{(\circ)}, t_l^{(-1)} + \rho h_l^{(-1)}) \phi(t_l^{(-1)} + \rho h_l^{(-1)}) d\rho \\
 &- \sum_{l=n+1}^{N-1} h_l^{(-1)} \int_0^1 \hat{k}_{12}(t_{n,i}^{(\circ)}, t_l^{(-1)} + \rho h_l^{(-1)}) \varphi(t_l^{(-1)} + \rho h_l^{(-1)}) d\rho, \\
 \\
 &= g_n^{(\mu)} + \sum_{\eta=0}^{\mu-1} \sum_{l=0}^{N-1} h_l^{(\eta)} \left[B_{n\backslash 1}^{(l,\eta+1,\eta)} U_l^{(\eta)} + B_{n\backslash 2}^{(l,\eta+1,\eta)} V_l^{(\eta)} \right] \\
 &+ \sum_{l=0}^{n-1} h_l^{(\mu)} \left[B_{n\backslash 1}^{(l,\mu,\mu)} U_l^{(\mu)} + B_{n\backslash 2}^{(l,\mu,\mu)} V_l^{(\mu)} \right] \\
 &+ h_n^{(\mu)} \left[D_{n\backslash 1}^{(n,\mu,\mu)} U_n^{(\mu)} + D_{n\backslash 2}^{(n,\mu,\mu)} V_n^{(\mu)} \right] \\
 &- h_n^{(-1)} \int_{c_i}^1 \hat{k}_{21}(t_{n,i}^{(\circ)}, t_n^{(-1)} + \rho h_n^{(-1)}) \phi(t_n^{(-1)} + \rho h_n^{(-1)}) d\rho \\
 &- h_n^{(-1)} \int_{c_i}^1 \hat{k}_{22}(t_{n,i}^{(\circ)}, t_n^{(-1)} + \rho h_n^{(-1)}) \varphi(t_n^{(-1)} + \rho h_n^{(-1)}) d\rho \\
 &- \sum_{l=n+1}^{N-1} h_l^{(-1)} \int_0^1 \hat{k}_{21}(t_{n,i}^{(\circ)}, t_l^{(-1)} + \rho h_l^{(-1)}) \phi(t_l^{(-1)} + \rho h_l^{(-1)}) d\rho \\
 &- \sum_{l=n+1}^{N-1} h_l^{(-1)} \int_0^1 \hat{k}_{22}(t_{n,i}^{(\circ)}, t_l^{(-1)} + \rho h_l^{(-1)}) \varphi(t_l^{(-1)} + \rho h_l^{(-1)}) d\rho.
 \end{aligned} \right. \quad (3.4)$$

حالت دوم: اگر $t > \theta(t_{n,i}^{(\mu)})$:

$$\left\{ \begin{aligned} U_n^{(\mu)} &= f_n^{(\mu)} + \sum_{\eta=0}^{\mu-1} \sum_{l=0}^{N-1} h_l^{(\eta)} \left[B_{n\backslash 1}^{(l,\eta+1,\eta)} U_l^{(\eta)} + B_{n\backslash 2}^{(l,\eta+1,\eta)} V_l^{(\eta)} \right] \\ &\quad + \sum_{l=0}^{n-1} h_l^{(\mu)} \left[B_{n\backslash 1}^{(l,\mu,\mu)} U_l^{(\mu)} + B_{n\backslash 2}^{(l,\mu,\mu)} V_l^{(\mu)} \right] \\ &\quad + h_n^{(\mu)} \left[D_{n\backslash 1}^{(n,\mu,\mu)} U_n^{(\mu)} + D_{n\backslash 2}^{(n,\mu,\mu)} V_n^{(\mu)} \right] \\ &\quad + \sum_{\eta=0}^{\mu-2} \sum_{l=0}^{N-1} h_l^{(\eta)} \left[\hat{B}_{n\backslash 1}^{(l,\eta+2,\eta)} U_l^{(\eta)} + \hat{B}_{n\backslash 2}^{(l,\eta+2,\eta)} V_l^{(\eta)} \right] \\ &\quad + \sum_{l=0}^{n-1} h_l^{(\mu)} \left[\hat{B}_{n\backslash 1}^{(l,\mu,\mu-1)} U_l^{(\mu)} + \hat{B}_{n\backslash 2}^{(l,\mu,\mu-1)} V_l^{(\mu)} \right] \\ &\quad + h_n^{(\mu)} \left[\hat{D}_{n\backslash 1}^{(n,\mu,\mu-1)} U_n^{(\mu)} + \hat{D}_{n\backslash 2}^{(n,\mu,\mu-1)} V_n^{(\mu)} \right] \\ &= g_n^{(\mu)} + \sum_{\eta=0}^{\mu-1} \sum_{l=0}^{N-1} h_l^{(\eta)} \left[B_{n\backslash 1}^{(l,\eta+1,\eta)} U_l^{(\eta)} + B_{n\backslash 2}^{(l,\eta+1,\eta)} V_l^{(\eta)} \right] \\ &\quad + \sum_{l=0}^{n-1} h_l^{(\mu)} \left[B_{n\backslash 1}^{(l,\mu,\mu)} U_l^{(\mu)} + B_{n\backslash 2}^{(l,\mu,\mu)} V_l^{(\mu)} \right] \\ &\quad + h_n^{(\mu)} \left[D_{n\backslash 1}^{(n,\mu,\mu)} U_n^{(\mu)} + D_{n\backslash 2}^{(n,\mu,\mu)} V_n^{(\mu)} \right] \\ &\quad + \sum_{\eta=0}^{\mu-2} \sum_{l=0}^{N-1} h_l^{(\eta)} \left[\hat{B}_{n\backslash 1}^{(l,\eta+2,\eta)} U_l^{(\eta)} + \hat{B}_{n\backslash 2}^{(l,\eta+2,\eta)} V_l^{(\eta)} \right] \\ &\quad + \sum_{l=0}^{n-1} h_l^{(\mu)} \left[\hat{B}_{n\backslash 1}^{(l,\mu,\mu-1)} U_l^{(\mu)} + \hat{B}_{n\backslash 2}^{(l,\mu,\mu-1)} V_l^{(\mu)} \right] \\ &\quad + h_n^{(\mu)} \left[\hat{D}_{n\backslash 1}^{(n,\mu,\mu-1)} U_n^{(\mu)} + \hat{D}_{n\backslash 2}^{(n,\mu,\mu-1)} V_n^{(\mu)} \right]. \end{aligned} \right. \quad (3'5)$$

انتگرال های دستگاه (3'4)، (3'5) به کمک روشهای انتگرال گیری با دقت مناسب تقریب زده می شوند و جواب های تقریبی (3'2) و (3'3) با حل دستگاه خطی (3'4) و (3'5) در هر زیر بازه σ_n ، $(n = 0, \dots, N-1)$ بدست می آیند.

۴ مثال های عددی

در این بخش یک مثال از دستگاه جبری انتگرال تاخیری نیمه صریح را در نظر می گیریم. تمامی محاسبات با استفاده از نرم افزار متمتیکا انجام شده است.

مثال ۲. در دستگاه معادلات جبری انتگرالی با تاخیر غیر صفر شونده به فرم زیر:

$$A(t)X(t) + \int_{t_*}^t K(t,s)X(s)ds + \int_{t_*}^{\theta(t)} \bar{K}(t,s)X(s)ds = f(t), \quad t \in (t_*, T], \quad (4'1)$$



با فرض

$$A(t) = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad K(t, s) = \begin{pmatrix} \exp(s-t) & (s+t) \\ \exp(s+t) & (s+t^2+1) \end{pmatrix} \quad \bar{K}(t, s) = \begin{pmatrix} t * \sin(s) & (t * s) \\ \sin(s) & (t * s + 3) \end{pmatrix}$$

مقادیر $f_1(t)$ و $f_2(t)$ را طوری می یابیم که $t \in [\theta(t_0), t_0]$ ، $X(t) = (y(t), z(t))^T = (\sin(t) + 1, \cos(t))^T$

$$t_0 = 0, \theta(t_0) = t - 1/2$$

باشد.

جدول ۱ مقدار ماکزیمم خطا را برای N مختلف و $m=2$ نشان می دهد به طوری که نقاط هم محلی $c_1=0/5$ ، $c_2=1$ هستند.

جدول ۱: بررسی خطا و همگرایی جواب برای N مختلف

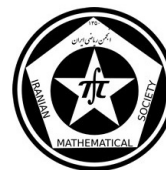
N	ماکزیمم خطا در روش هم محلی $y(t)$	ماکزیمم خطا در روش هم محلی $z(t)$
۴	2.44×10^{-4}	2.69×10^{-3}
۸	7.00×10^{-5}	7.088×10^{-4}
۱۶	1.87×10^{-5}	1.81×10^{-4}
۳۲	4.83×10^{-6}	4.61×10^{-5}

مراجع

- [1] J. Belair, Population models with state-dependent delays. In: Arino, O., Axelrod, D.E., Kimmel, M. (eds.) Mathematical Population Dynamics, 165-176. Marcel Dekker, New York (1991)
- [2] Brunner, H.: Collocation Methods for Volterra Integral and Related Functional Differential Equations. Cambridge University Press, Cambridge (2004)
- [3] M.V. Bulatov, M. N. Machkhina, V.N. Phat, Existence and uniqueness of solutions to nonlinear integral-algebraic equations with variable limits of integrations. Commun. Appl. Nonlinear Anal. 21, No. 1, 65-76 (2014)
- [4] M.V. Bulatov, M. N. Machkhina, On a class of integro-algebraic equations with variable integration limits. Zh. Sredn. Mat. Obshch. 12, No. 2, 40-45 (2010)

پست الکترونیکی: shnasheikhi@gmail.com

پست الکترونیکی: s.pishbin@urmia.ac.ir



کاربرد روش توابع پایه شعاعی برای حل مساله معکوس سهموی خطی

علی ذاکری^۱، امیرحسین صالحی شایگان^۲ و مسلم جمالپور مالک آبادی^۳

دانشیار، دانشکده ریاضی، دانشگاه صنعتی خواجه نصیرالدین طوسی، تهران^۱

استادیار، دانشکده علوم پایه، دانشگاه خاتم الانبیاء، تهران^۲

دانشجوی کارشناسی ارشد، دانشکده ریاضی، دانشگاه صنعتی خواجه نصیرالدین طوسی، تهران^۳

چکیده

در این مقاله با استفاده از روش‌های توابع پایه شعاعی و لوبنبرگ-مارکوارت جواب عددی پایدار برای مساله معکوس سهموی خطی به دست می‌آوریم. در پایان برای نشان دادن دقت روش پیشنهادی مثال عددی ارائه می‌کنیم. واژه‌های کلیدی: مساله معکوس؛ روش توابع پایه شعاعی؛ روش لوبنبرگ-مارکوارت؛ مساله سهموی خطی.

۱ پیش‌گفتار

۱.۱ مقدمه

یکی از روش‌هایی که برای تقریب توابع چندمتغیره به کار می‌رود و اخیراً مورد توجه قرار گرفته است درونیایی مبتنی بر توابع پایه شعاعی (RBF) می‌باشد. روش RBF به دلیل پیچیدگی محاسباتی کمتر در ابعاد بالاتر عدم نیاز به شبکه‌بندی دامنه و زمان اجرای کم در ابعاد بالاتر مورد استفاده قرار می‌گیرد. توابع پایه شعاعی انواع مختلفی دارد، ما در این مقاله تابع پایه شعاعی گاوسی را به صورت زیر در نظر می‌گیریم:

$$\varphi(r) = e^{-cr^2}$$

که c پارامتر شکل و به ازای $j = 0, 1, \dots, n$ ، $r = \|x - x_j\|$ محققان در [۳] و [۴] از توابع پایه شعاعی چندمربعی و معکوس چندمربعی برای تقریب استفاده کرده‌اند. در این مقاله به حل یک مساله سهموی معکوس خطی با استفاده از روش RBF می‌پردازیم. لذا این مساله به یک مساله تقریب پارامتر کاهش پیدا می‌کند از این رو از روش RBF برای حل یک مساله با پارامتر مجهول استفاده کرده



و تابعک هزینه را با کمک آن و داده‌های اندازه‌گیری شده می‌سازیم. با کمینه‌سازی تابعک هزینه به کمک روش‌های منظم‌سازی از قبیل روش لونبرگ-مارکوارت، پارامتر مجهول که همان جواب مساله معکوس است را تقریب می‌زنیم [۲]. بنابراین حل یک مساله معکوس از سه مرحله زیر تشکیل می‌شود:

الف) بکارگیری روش عددی (در این مقاله روش RBF) برای حل یک مساله با پارامتر مجهول

ب) ساخت تابعک هزینه

ج) استفاده از روش‌های منظم‌سازی به منظور کمینه‌کردن تابعک هزینه

۲.۱ حل مساله سهموی معکوس خطی

در این بخش با استفاده از روش توابع پایه شعاعی و روش منظم‌سازی لونبرگ-مارکوارت یک جواب عددی پایدار برای یک مساله معکوس سهموی خطی به دست می‌آوریم. برای این منظور، با ساخت تابعک هزینه و کمینه‌سازی آن، مساله یافتن یک تابع مجهول را به یک مساله تقریب پارامتر تبدیل می‌کنیم و به کمک روش منظم‌سازی لونبرگ-مارکوارت این پارامترهای مجهول را می‌یابیم [۲].

مساله سهموی معکوس خطی زیر را در نظر می‌گیریم. در این مساله که یک مساله پسرواست، هدف ما یافتن تابع اولیه

$$g = g(x), \quad (1.1)$$

است به طوری که $U = U(x, t)$ و $U(x, 0) = g(x)$ جواب مساله زیر است:

$$U_t - U_{xx} = F(x, t),$$

$$U(0, t) = 0, \quad 0 \leq t \leq T,$$

$$U(1, t) = 0, \quad 0 \leq t \leq T,$$

$$U(x, T) = \bar{g}(x), \quad 0 \leq x \leq 1. \quad (1.2)$$

در این مساله، توابع $F(x, t)$ و $\bar{g}(x)$ توابعی معلوم هستند ($\bar{g}(x)$ را داده اندازه‌گیری شده یا شرط اضافی می‌گویند) و توابع مجهول در مساله $U(x, t)$ و $g(x)$ می‌باشند. در صورتیکه $g = g(x)$ معلوم باشد، آنگاه مساله (۱.۲) یک مساله مستقیم می‌باشد. وجود و یکتایی این مساله مستقیم در [۲] بررسی شده است. برای پیدا کردن جفت توابع مجهول (U, g) با استفاده از داده‌های اندازه‌گیری شده (شرط اضافی)، تابعک هزینه را به صورت زیر تشکیل می‌دهیم:

$$S(g) := \sum_{i=1}^I (U(x_i^*, t; g) |_{t=T} - \bar{g}(x_i^*))^2, \quad (1.3)$$

که در آن $x_i^* = \frac{i}{I}$ است. با استفاده از این تابعک هزینه، مساله کمینه‌سازی زیر را معرفی می‌کنیم:

$$\inf_{g \in G} S(g), \quad (1.4)$$



که در آن G مجموعه جواب‌های قابل قبول است. برای یافتن کمینه تابع هزینه (۱۳)، ابتدا تابع مجهول $g(x)$ را به صورت زیر تقریب می‌زنیم:

$$g(x) \simeq \sum_{i=0}^n g_i \varphi(\|x - x_i\|),$$

که یک ترکیب خطی از توابع پایه شعاعی و g_i پارامترهای مجهول هستند و باید محاسبه شوند. با این کار مساله یافتن تابع g به مساله یافتن پارامترهای g_i تبدیل می‌شود. این ترکیب خطی را به عنوان شرط اولیه یک مساله مستقیم در نظر می‌گیریم و به کمک روش RBF مساله را همانطور که در بخش قبل بیان شد حل می‌کنیم. پس از یافتن جواب، $U(x, t; g)|_{t=T}$ را به کمک آن می‌یابیم و آن را در (۱۳) قرار می‌دهیم. از آنجاییکه مساله پسرو (۱۲) و در نتیجه یافتن پارامترهای مجهول g_i در مساله کمینه‌سازی (۱۴) در رده مسائل بدوضع هستند لذا باید از روش‌های منظم‌سازی به منظور غلبه بر بدوضعی کمک گرفت. در اینجا از روش منظم‌سازی لوبنرگ-مارکوارت استفاده می‌کنیم و در الگوریتم زیر به بیان این روش می‌پردازیم.

۳.۱ الگوریتم محاسباتی یافتن تابع مجهول $g(x)$

در این قسمت یک الگوریتم محاسباتی برای پیدا کردن $g(x)$ ارائه می‌کنیم. این الگوریتم به صورت زیر است:

مرحله ۱: فرض می‌کنیم

$$g(x) \simeq \sum_{i=0}^n g_i \varphi(\|x - x_i\|),$$

تقریبی از تابع مجهول باشد که در آن g_i پارامترهای مجهول باشند.

مرحله ۲: با استفاده از روش RBF یک جواب تقریبی برای $U(x, t; g)|_{t=T}$ به دست می‌آوریم. در این حالت وقتی مساله مستقیم غیرخطی را حل کنیم تابع U بر حسب پارامترهای مجهول g_i به دست خواهد آمد.

مرحله ۳: با کمینه‌سازی تابع هزینه زیر، $n+1$ پارامتر مجهول g_i را به دست می‌آوریم:

$$S(g) := \sum_{i=1}^I (U(x_i^*, t; g)|_{t=T} - \bar{g}(x_i^*))^2.$$

چون دستگاه معادلات جبری به دست آمده بدوضع می‌باشد، بنابراین از روش لوبنرگ-مارکوارت برای غلبه بر بدوضع مساله استفاده می‌کنیم.

مرحله ۴: منظم‌سازی لوبنرگ-مارکوارت

فرض کنید

$$U(g) = [U_1, U_2, \dots, U_I]^T,$$

$$\bar{g} = [\bar{g}_1, \bar{g}_2, \dots, \bar{g}_I]^T,$$

$$g = [g_0, g_1, \dots, g_n]^T,$$



که در آن $\bar{g}_i = \bar{g}(x_i^*)$ و $U_i = U(x_i^*, T, g)$ ، به ازای $i = 1, 2, \dots, I$ هستند. آنگاه فرم ماتریسی تابعک (۱۳) به صورت زیر است:

$$S(g) = [U(g) - \bar{g}]^T [U(g) - \bar{g}],$$

که در آن

$$[U(g) - \bar{g}] = [U_1 - \bar{g}_1, U_2 - \bar{g}_2, \dots, U_I - \bar{g}_I],$$

و T نشان‌دهنده ترانهاده و I تعداد کل اندازه‌گیری‌ها است. برای کمینه‌سازی نرم حداقل مربعات، مشتقات $S(g)$ نسبت به هر یک از پارامترهای مجهول $\{g_i\}_{i=1}^n$ را به صورت زیر برابر صفر قرار می‌دهیم:

$$\frac{\partial S(g)}{\partial g_1} = \frac{\partial S(g)}{\partial g_2} = \dots = \frac{\partial S(g)}{\partial g_n} = 0,$$

و یا فرم ماتریسی آن که به شکل زیر است:

$$\nabla S(g) = \left[\frac{\partial U^T(g)}{\partial g} \right] [U(g) - \bar{g}] = 0,$$

که

$$\frac{\partial U^T(g)}{\partial g} = \begin{bmatrix} \frac{\partial}{\partial g_1} \\ \frac{\partial}{\partial g_2} \\ \vdots \\ \frac{\partial}{\partial g_n} \end{bmatrix} [U_1, U_2, \dots, U_I].$$

از این رو ماتریس حساسیت به فرم زیر است:

$$J(g) = \left[\frac{\partial U^T(g)}{\partial g} \right] = \begin{bmatrix} \frac{\partial U_1}{\partial g_1} & \frac{\partial U_1}{\partial g_2} & \dots & \frac{\partial U_1}{\partial g_n} \\ \frac{\partial U_2}{\partial g_1} & \frac{\partial U_2}{\partial g_2} & \dots & \frac{\partial U_2}{\partial g_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial U_I}{\partial g_1} & \frac{\partial U_I}{\partial g_2} & \dots & \frac{\partial U_I}{\partial g_n} \end{bmatrix} \quad (15)$$

باید توجه کنیم که زمانیکه ضرایب حساسیت $\frac{\partial U_i}{\partial g_j}$ خیلی کوچک باشند داریم:

$$\left| (J(g))^T J(g) \right| \approx 0$$

مساله معکوس بدوضع می‌باشد. همچنین دترمینان $\left| (J(g))^T J(g) \right|$ نزدیک صفر است اگر هر ستون $J(g)$ بتواند بصورت ترکیب خطی از ستون‌های دیگر بیان شود.

اکنون الگوریتم محاسباتی برای روش منظم‌سازی لونیگ-مارکوارت بصورت زیر می‌باشد:



فرض کنید حدس اولیه برای بردار ضرایب مجهول g در دسترس باشد و آن را با $g^{(0)}$ نشان دهیم، آنگاه:

۱- μ_0 را پارامتر منظم سازی دلخواهی در نظر می گیریم (به عنوان مثال $\mu_0 = 0.001$) و قرار می دهیم $k = 0$.

۲- $U(g^{(0)})$ و $S(g^{(0)})$ را محاسبه می کنیم.

۳- ماتریس حساسیت J^k تعریف شده در (۱۵) را محاسبه کرده و سپس قرار می دهیم:

$$\Omega^{(k)} = \text{diag}[(J^k)^T J^k]$$

۴- برای به دست آوردن $\Delta g^k = g^{k+1} - g^k$ دستگاه خطی از معادلات جبری زیر را حل می کنیم:

$$[(J^k)^T J^k + \mu^k \Omega^{(k)}] \Delta g^k = (J^k)^T [\bar{g} - U(g^{(k)})]$$

۵- $g^{k+1} = \Delta g^k + g^k$ را محاسبه می کنیم.

۶- اگر $S(g^{k+1}) \geq S(g^k)$ باشد μ^k را با $\mu^k \times 10$ جایگزین می کنیم و به مرحله ۴ باز می گردیم.

۷- اگر $S(g^{k+1}) < S(g^k)$ باشد μ^k را با $\mu^k \times 0.8$ جایگزین می کنیم.

۸- فرض کنیم tol داده شده باشد. اگر $\|g^{k+1} - g^k\| < tol$ آنگاه g^{k+1} را به عنوان یک تقریب در نظر می گیریم. در غیر

این صورت k را با $k + 1$ جایگزین کرده و به مرحله ۳ باز می گردیم.

۲ نتایج اصلی

در این قسمت با ارائه یک مثال دقت روش ارائه شده را نشان می دهیم.

مثال ۱. مساله پسر خطی زیر را در نظر بگیرید:

$$U_t - U_{xx} = F(x, t),$$

$$U(0, t) = 0, \quad 0 \leq t \leq 1$$

$$U(1, t) = 0, \quad 0 \leq t \leq 1$$

$$U(x, 1) = x^5(1-x), \quad 0 \leq x \leq 1. \quad (21)$$

که در آن جواب دقیق مساله به صورت $U(x, t) = tx^5(1-x)$ می باشد. در این مساله علاوه بر $U(x, t)$ ، شرط اولیه $g(x) = U(x, 0)$ نیز مجهول می باشد. با استفاده از الگوریتم ارائه شده به ازای $I = 10$ و $m = n = 4$ و $m = n = 6$ خطای مطلق روش پیشنهاد شده با تابع پایه شعاعی گاوسی و پارامتر شکل $c = 1$ در بازه $[a, b] = [0, 1]$ و $T = 1$ در جدول های ۱ و ۲ و در شکل های ۱ و ۲ ارائه شده است. با توجه به نتایج به دست آمده مشاهده می شود که با افزایش تعداد نقاط خطای مطلق روش کمتر می شود.

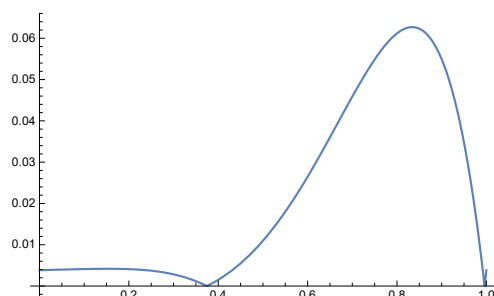


x_i	۰	۰/۲۵	۰/۵	۰/۷۵	۱
$ g(x_i) - \bar{g}(x_i) $	۰/۰۰۳۸۳۳۱۵	۰/۰۰۳۷۱۶۲۸	۰/۰۱۰۹۴۹۵	۰/۰۵۴۸۷۴۹	۰/۰۰۳۸۳۷۵۳

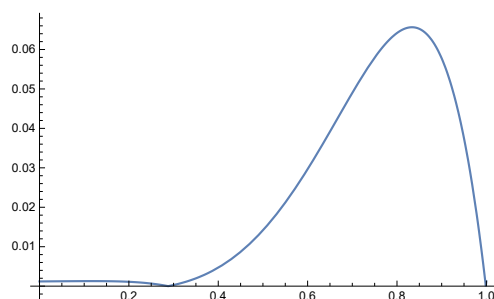
جدول ۱: خطای مطلق حاصل از حل مساله سهموی معکوس خطی با کمک الگوریتم ارائه شده به ازای $m = ۴, n = ۴$

x_i	۰	۰/۱۶۶۶۶۷	۰/۳۳۳۳۳۳	۰/۵	۰/۶۶۶۶۶۷	۰/۸۳۳۳۳۳	۱
$ g(x_i) - \bar{g}(x_i) $	۰/۰۰۱۱۹۱۵۳	۰/۰۰۱۲۲۶۹۶	۰/۰۰۱۳۱۵۷۸	۰/۰۰۱۴۱۶۴۸	۰/۰۰۴۲۴۶۸۱	۰/۰۰۶۵۶۴۵۳	۰/۰۰۱۱۹۱۷۳

جدول ۲: خطای مطلق حاصل از حل مساله سهموی معکوس خطی با کمک الگوریتم ارائه شده به ازای $m = ۶, n = ۶$



شکل ۱: نمودار خطای مطلق حاصل از حل مساله سهموی معکوس خطی با کمک الگوریتم ارائه شده به ازای $m = ۴, n = ۴$



شکل ۲: نمودار خطای مطلق حاصل از حل مساله سهموی معکوس خطی با کمک الگوریتم ارائه شده به ازای $m = ۶, n = ۶$

مراجع

- [1] Fornberg, B. and Flyer, N. A primer on radial basis functions with applications to the geosciences. Society for Industrial and Applied Mathematics, 2015.
- [2] Zakeri, A. Shayegan, A. H. and Sakaki S. Application of sinc-Galerkin method for solving a



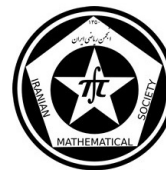
nonlinear inverse parabolic problem. Transactions of A. Razmadze Mathematical Institute 171, no. 3 (2017): 411–423.

- [3] Liu, C.S., Chen, W. and Fu, Z. A multiple-scale MQ-RBF for solving the inverse Cauchy problems in arbitrary plane domain. Engineering Analysis with Boundary Elements 68, (2016): 11–16.
- [4] Ostrowski, Z., Bialecki, R.A. and Kassab, A.J. Solving inverse heat conduction problems using trained POD-RBF network inverse method. Inverse Problems in Science and Engineering 16, no. 1 (2008): 39–54.

پست الکترونیکی: azakeri@kntu.ac.ir

پست الکترونیکی: ah.salehi@mail.kntu.ac.ir

پست الکترونیکی: m.jamalpour95@email.kntu.ac.ir



کاربرد روش‌های پروکسیمال در حل دستگاه معادلات قدر مطلق

سمیرا شهسواری[†]، سعید کتابچی[‡]

[†] دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

[‡] دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

چکیده

در این مقاله به بیان دو روش جدید، روش تفاضل محدب پروکسیمال با گام برونابی و روش زیرگرادیان پروکسیمال، برای حل دستگاه معادلات قدر مطلق می‌پردازیم و همگرایی آنها را به یک جواب از این دستگاه ثابت می‌کنیم. برای نشان دادن کارایی روش‌های پیشنهادی، مسائل دستگاه معادلات قدر مطلق را در ابعاد بالا تولید کرده و حل می‌نماییم. نتایج محاسباتی نشان می‌دهد، روش‌های پیشنهادی از دقت و سرعت بالایی برخوردار هستند.

واژه‌های کلیدی: الگوریتم تفاضل محدب پروکسیمال، الگوریتم زیرگرادیان پروکسیمال، دستگاه معادلات قدر مطلق. طبقه‌بندی موضوعی [۲۰۱۰]: 90C26, 90C25, 65D10.

۱ پیش‌گفتار

در سالهای اخیر حل دستگاه معادلات قدر مطلق*

$$Ax - |x| = b, \quad (1.1)$$

که در آن $A \in \mathbb{R}^{n \times n}$ ، $b \in \mathbb{R}^n$ و $|x|$ نشان دهنده قدر مطلق بردار x است، مورد توجه بسیاری از محققین قرار گرفته است و الگوریتم‌های متفاوتی برای حل آن ارائه شده است [۱، ۲، ۴، ۵، ۶، ۷]. موسایی و کتابچی در [۷]، روش‌های نیوتن تعمیم یافته با قاعده آرمیثو (NMA) و آشوب هوموتوپیی را برای حل این سیستم ارائه دادند. آنها در روش نیوتن تعمیم یافته با قاعده آرمیثو، مسئله بهینه سازی نامقید

$$\min_{x \in \mathbb{R}^n} \frac{1}{2} \|Ax - |x| - b\|^2 \quad (1.2)$$

*Absolute value equations (AVE)



را که معادل با سیستم (۱۱) است، را در نظر گرفتند. این مسئله، یک مسئله نامحدب و ناهموار و از پایین کراندار است. الگوریتم نیوتن تعمیم یافته با قاعده آرمیثو، بدون توجه به خاصیت نامحدب مسئله (۱۲)، برای حل این مسئله ارائه شد. بنابراین در این مقاله، خاصیت نامحدب مسئله (۱۲) را در نظر می‌گیریم و تابع هدف این مسئله را به دو صورت، نخست به صورت مجموع یک تابع محدب هموار با گرادیان پیوسته لیپ‌شیتز با ضریب $L > 0$ و یک تابع محدب محض و یک تابع مقعر پیوسته و سپس به صورت مجموع یک تابع محدب و یک تابع دیفرانسیل‌ناپذیر، بازنویسی می‌کنیم. از این رو، ما دو مسئله جدید برای حل سیستم $AVE(11)$ بدست می‌آوریم و سپس این دو مسئله را با دو نوع از روشهای پروکسیمال که عبارتند از روش تفاضل محدب پروکسیمال با گام برونابی[†] و روش زیرگرادیان پروکسیمال[‡] حل می‌کنیم.

در این مقاله، $\|\cdot\|_\infty$ و $\|\cdot\|$ ، در فضای اقلیدسی $\mathbb{R}^n$ به ترتیب برای نمایش نرم ۲ و نرم بی‌نهایت به کار می‌روند. x^t و x^k به ترتیب نشان دهنده ترانهاده بردار $x \in \mathbb{R}^n$ و ضرب داخلی می‌باشند. $(x)_+$ و $sign(x)$ بردارهایی در $\mathbb{R}^n$ هستند که برای $i = 1, \dots, n$ ، $(x)_+ = \max(0, x_i)$ و $(sign(x))_i = \begin{cases} 1 & \text{if } x_i > 0 \\ -1 & \text{if } x_i < 0 \\ 0 & \text{if } x_i = 0 \end{cases}$ است. اگر $x_i < 0$ یا $x_i > 0$ و در غیر این صورت $(sign(x))_i = 0$ است. حال فرض کنید f تابعی در $\mathbb{R}^n$ باشد، آنگاه دامنه آن با $dom(f)$ نشان داده می‌شود و اگر f تابعی دیفرانسیل‌پذیر در نقطه $x \in \mathbb{R}^n$ باشد، $\nabla f(x)$ نشان دهنده گرادیان این تابع در نقطه x است و در غیر این صورت، گرادیان تعمیم یافته f با $\partial f(x)$ نشان داده می‌شود و به صورت زیر تعریف می‌شود:

$$\partial f(x) = \{\xi \in \mathbb{R}^n : f(y) \geq f(x) + \xi^t(y - x), \forall y \in dom(f)\}.$$

برای $\sigma > 0$ پروکسیمال تابع f در نقطه $y \in \mathbb{R}^n$ به صورت

$$prox_{\sigma f}(y) = \arg \min_{x \in \mathbb{R}^n} f(x) + \frac{\sigma}{2} \|x - y\|^2,$$

تعریف می‌شود.

۲ نتایج اصلی

الگوریتم‌های پروکسیمال ابزار مفیدی برای حل مسائل ناهموار، نامحدب و مقید و مقیاس بزرگ می‌باشند. عملیات اصلی این روش‌ها، ارزیابی عملگر پروکسیمال[§] یک تابع در هر تکرار است، که خود این عملگر درگیر حل یک مسئله بهینه‌سازی محدب می‌شود. این زیر مسئله که عموماً مسئله تصویر یک نقطه روی یک مجموعه محدب است، اغلب بدست آوردن یک جواب تقریبی برای آن کافی است یا می‌توان آن را با روش‌های خاص ساده یا استاندارد خیلی ساده حل کرد. در این بخش، در قالب دو زیربخش دو نوع از الگوریتم‌های پروکسیمال، روش تفاضل محدب پروکسیمال با گام برونابی و روش زیرگرادیان پروکسیمال، را برای حل دستگاه معادلات قدر مطلق ارائه می‌دهیم.

[†]Proximal difference-of-convex algorithm with extrapolation

[‡]Proximal subgradient method

[§]proximal operator

۱۰۲ روش تفاضل محدب پروکسیمال با گام برونابی

روش تفاضل محدب با گام برونابی که به اختصار $PDCA_e$ نامیده می‌شود، توسط ون و همکارانش در [۱۰] برای حل مسائل نامحدب نامقید

$$\min_{x \in \mathbb{R}^n} f(x) + p(x) - g(x), \quad (2'1)$$

معرفی شد، که در آن f تابع محدب هموار با گرادیان پیوسته لیپ‌شیتزی با ضریب $L > 0$ و p تابع محدب محض بسته و g تابع محدب پیوسته است.

در اینجا، ابتدا نشان می‌دهیم که حل دستگاه (۱'۱) معادل با حل مسئله بهینه سازی نامقید (۲'۱) است و سپس روش $PDCA_e$ را برای حل آن بکار می‌گیریم.

قضیه ۱. فرض کنید که $S = Ax - b$ و

$$\phi_1(x_i, \mu) = \begin{cases} -x_i, & \text{if } x_i < -\frac{\mu}{\sqrt{L}}, \\ \frac{x_i^2}{\mu} + \frac{\mu}{\sqrt{L}}, & \text{if } -\frac{\mu}{\sqrt{L}} \leq x_i \leq \frac{\mu}{\sqrt{L}}, \\ x_i, & \text{if } x_i > \frac{\mu}{\sqrt{L}}. \end{cases} \quad \phi_2(x_i, \mu) = \begin{cases} -x_i - \frac{\mu}{\sqrt{L}}, & \text{if } x_i < -\mu, \\ \frac{x_i^2}{\sqrt{L}\mu} & \text{if } -\mu \leq x_i \leq \mu, \\ x_i - \frac{\mu}{\sqrt{L}}, & \text{if } x_i > \mu. \end{cases}$$

برای $\mu > 0$ ، توابع هموار ساز $|x_i|$ ها به ازای $i = 1, \dots, n$ باشند. آنگاه مسئله (۱'۲) معادل با مسئله بهینه سازی تفاضل محدب هموار

$$\min_{x \in \mathbb{R}^n} \bar{F}(x) = f(x) + p(x) - g(x), \quad (2'2)$$

است، که

$$\Phi_i(x) = (\phi_i(x_1, \mu), \phi_i(x_2, \mu), \dots, \phi_i(x_n, \mu))^t \text{ و } i = 1, 2$$

$$f(x) = \frac{1}{\sqrt{L}}(\Phi_i(-S) - S)^t \Phi_i(x), \quad p(x) = \frac{1}{\sqrt{L}}\|Ax - b\|^2 + \frac{1}{\sqrt{L}}\|x\|^2, \quad g(x) = -\frac{1}{\sqrt{L}}(\Phi_i(S) + S)^t \Phi_i(x). \quad (2'3)$$

که در آن f تابع محدب با گرادیان پیوسته لیپ‌شیتزی با ضریب $L = 2\mu(\|A\| + \|b\|)$ و p تابع محدب محض و g تابع محدب پیوسته هستند.

□

اثبات. واضح است.

حال برای حل مسئله (۲'۲) با الگوریتم $PDCA_e$ ، در هر مرحله نقطه جدید به صورت

$$\begin{aligned} x_{k+1} &= \text{prox}_{Lp}(y_k - \frac{1}{L}(\nabla f(y_k) - \nabla g(x_k))) \\ &= \arg \min_{x \in \mathbb{R}^n} \frac{1}{\sqrt{L}}\|Ax - b\|^2 + \frac{1}{\sqrt{L}}\|x\|^2 + (\nabla f(y_k) - \nabla g(x_k))^t x + \frac{L}{\sqrt{L}}\|x - y_k\|^2, \end{aligned} \quad (2'4)$$



محاسبه می‌شود، که $y_k = x_k + \beta_k(x_k - x_{k-1})$ و β_k پارامتر برونیایی است (برای دیدن جزئیات بیشتر به [۱۰] رجوع کنید). از این رو، الگوریتم $PDCA_e$ برای حل سیستم (۱۱) عبارت است از:

الگوریتم ۱. الگوریتم $PDCA_e$ برای حل AVE .

۱. پارامترهای $\mu_0, \eta_1, \eta_2, \epsilon > 0$ و $\{\beta_k\} \in (0, 1]$ که $\sup_k \beta_k < 1$ و نقطه $x_0 \in \text{dom}(p)$ را انتخاب کن. قرار بده $x_{-1} = x_0, k = 0$.

۲. مادامی که $\|F(x_k)\|_\infty \geq \epsilon$ انجام بده:

- x_{k+1} را از زیر مسئله محدب قوی (۲۴) محاسبه کن.

- پارامتر هموار ساز μ_k را به روز کن:

اگر $\|\Psi(x_{k+1})\|_\infty > \eta_1 \mu_k$ ، قرار بده $\mu_{k+1} = \mu_k$

در غیر اینصورت قرار بده $\mu_{k+1} = \eta_2 \mu_k$.

- اگر $x_{k+1} = x_k$ ، آنگاه روند تمام و x_{k+1} جواب است. در غیر اینصورت قرار بده $k = k + 1$.

گزاره ۲. فرض $\{x_k\}$ دنباله تولید شده توسط الگوریتم ۱ باشد. آنگاه هر نقطه انباشتگی این دنباله یک نقطه ایستای مسئله (۱۲) است.

اثبات. به [۱۰] مراجعه شود. □

۲.۲ روش زیرگرادیان پروکسیمال

روش زیرگرادیان پروکسیمال (PSM) برای حل مسئله بهینه‌سازی

$$\min_{x \in \mathbb{R}^n} F(x) = f(x) + g(x) \quad (25)$$

در [۹] و [۳] معرفی شد، که در آن g و f به ترتیب توابع محدب محض و پیوسته لیپ‌شیتز هستند.

قضیه ۳. فرض کنید $\gamma > 0$ و $\frac{\gamma}{4}\|x\|^2 + (b)_+^t|x| - \frac{\gamma}{4}\|x\|^2, g(x) = \frac{1}{4}\|Ax - |x|\|^2 - b^t Ax + (b)_+^t|x| + \frac{\gamma}{4}\|x\|^2$ فرض کنید $f(x) = -(-b)_+^t|x| - \frac{\gamma}{4}\|x\|^2$. آنگاه برای $\gamma > 0$ به اندازه کافی بزرگ، g یک تابع محدب محض است و مسئله (۱۲) معادل است با مسئله بهینه‌سازی

$$\min_{x \in \mathbb{R}^n} F(x) = f(x) + g(x). \quad (26)$$



اثبات. واضح است که $k_1(x) = \frac{1}{\gamma} \|Ax\|^2 + \frac{1}{\gamma} \|x\|^2 - b^t Ax + (b)_+^t |x|$ یک تابع محدب است. بنابراین برای اثبات تحدب g کفیت ثابت کنیم برای γ به اندازه کافی بزرگ، تابع $k_2(x) = -x^t A|x| + \frac{\gamma}{2} \|x\|^2$ محدب است. از این رو،

$$k_2(x) = x^t \left(\frac{\gamma}{2} I - \frac{1}{2} (AD_x + D_x A^t) \right) x \sum_{i=1}^n \left(\left(\frac{\gamma}{2} - A_{ii} \text{sign}(x_i) \right) x_i^2 \right) - \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n (A_{ij} \text{sign}(x_i) + A_{ji} \text{sign}(x_j)) x_i x_j,$$

که $\text{sign}(x_i) = 0, \pm 1$ چون $i, j = 1, \dots, n$ درایه‌های ماتریس A می‌باشند. تمام i ها، بنابراین $A_{ij} \text{sign}(x_i) \in \mathbb{R}$ ، $i, j = 1, \dots, n$ ، از این رو، وجود دارد $\alpha_i, \lambda_i \in \mathbb{R}$ ، $i = 1, \dots, n$ ، به طوری که $k_2(x) = \sum_{i=1}^n \left(\left(\frac{\gamma}{2} - \alpha_i \right) x_i^2 + \lambda_i x_i \right)$ ، بنابراین، برای γ به اندازه کافی بزرگ $k_2(x)$ محدب است. $\square$

بر اساس روش زیرگردان پروکسیمال در هر تکرار، نقطه جدید به صورت زیر محاسبه می‌شود:

$$\begin{aligned} x_{k+1} &= \text{prox}_{\sigma g} \left(x_k - \frac{1}{\sigma} \xi_k \right) \\ &= \arg \min_{x \in \mathbb{R}^n} \frac{1}{\gamma} \|Ax - |x|\|^2 - b^t Ax + (b)_+^t |x| + \frac{\gamma}{2} \|x\|^2 + \xi_k^t x + \frac{\sigma}{2} \|x - x_k\|^2, \end{aligned} \quad (2.7)$$

که $\xi_k = -D_k(-b)_+ - \gamma x_k$ زیرگردان f در x_k است و $D_k = \text{diag}(\text{sign}(x_k))$. پس، روش PSM برای حل سیستم (1.1) :

الگوریتم ۲. الگوریتم PSM برای حل AVE.

۱. پارامتر $\epsilon > 0$ و نقطه شروع $x_0 \in \text{dom}(g)$ را انتخاب کن و قرار بده $k = 0$.

۲. مادامی که $\|c_k\|_\infty \geq \epsilon$ انجام بده:

- y_k را از زیر مسئله محدب قوی (2.7) محاسبه کن.

- قرار بده $c_k = y_k - x_k$ و $x_{k+1} = y_k$ ، $k = k + 1$.

قضیه ۴. فرض کن $\{x_k\}$ دنباله تولید شده توسط الگوریتم ۲ باشد. آنگاه به ازای هر $k \geq 0$ ، $F(x_{k+1}) \leq F(x_k)$ و $\{x_k\}$ یک دنباله کراندار است و هر نقطه انباشتی این دنباله یک نقطه ایستای مسئله (1.2) است.

اثبات. چون $x_{k+1} = \text{prox}_{\sigma g} \left(x_k - \frac{1}{\sigma} \xi_k \right)$ ،

$$\begin{aligned} F(x_{k+1}) - \frac{\gamma}{2} \|x_k\|^2 - \frac{1}{2} \|b\|^2 &\leq \frac{1}{2} \|Ax_{k+1} - |x_{k+1}|\|^2 - b^t Ax_{k+1} + (b)_+^t |x_{k+1}| \\ &+ \frac{\gamma}{2} \|x_{k+1}\|^2 - (-b)_+^t D_k x_{k+1} - \gamma x_k^t x_{k+1} + \frac{\sigma}{2} \|x_{k+1} - x_k\|^2 \leq F(x_k) - \frac{\gamma}{2} \|x_k\|^2 - \frac{1}{2} \|b\|^2. \end{aligned}$$

$$F(x_{k+1}) \leq F(x_{k+1}) + \frac{\gamma}{2} \|x_{k+1} - x_k\|^2 \leq F(x_k). \quad (2.8)$$

برای اثبات قسمت دوم، با توجه به نامساوی (2.8) واضح است که $\{x_k\}$ کراندار است. از طرف دیگر، اگر $x_k \rightarrow \bar{x}$ آنگاه $D_K \rightarrow \bar{D}$ که $D_k = \text{diag}(\text{sign}(x_k))$ و $\bar{D} = \text{diag}(\text{sign}(\bar{x}))$. چون $x_{k+1} = \text{prox}_{\sigma g}(x_k - \frac{1}{\sigma} \xi_k)$ بنابراین

$$\begin{aligned} \circ \in \partial g(x_{k+1}) + \sigma(x_{k+1} - (x_k - \frac{1}{\sigma} \xi_k)) &\implies \circ = (A - D_{k+1})^t (A - D_{k+1}) x_{k+1} - A^t b + D_{k+1} (b)_+ \\ &+ \gamma x_{k+1} + \sigma(x_{k+1} - (x_k - \frac{1}{\sigma} (-D_k (-b)_+ - \gamma x_k))). \end{aligned}$$

حال اگر $k \rightarrow +\infty$ آنگاه

$$\begin{aligned} \circ &= (A - \bar{D})^t (A - \bar{D}) \bar{x} - A^t b + \bar{D} (b)_+ - \bar{D} (-b)_+ \implies \circ = (A - \bar{D})^t (A - \bar{D}) \bar{x} - A^t b + \bar{D} b \\ &\implies \circ = (A - \bar{D})^t ((A - \bar{D}) \bar{x} - b) \implies \circ \in \partial F(\bar{x}). \end{aligned}$$

□

۳ دست آوردهای پژوهش

در این بخش، برای نشان دادن کارایی و درستی روش‌های ارائه شده، مثال‌هایی که به طور تصادفی توسط یک کامپیوتر شخصی ($\text{CORE } i5, \text{ CPU } 2.50 \text{ GHz}, 4 \text{ GB memory}, \text{ MATLAB } 2013$) تولید شده‌اند را حل می‌کنیم و سپس نتایج عددی به دست آمده را از نقطه نظر دقت و زمان محاسبه با روش NMA مقایسه می‌کنیم. برای تولید مثال‌های تصادفی از کد متلب زیر استفاده شده است:

```
n=input('Entern : ');
d=input('Enterdin(°, 1] : ');
A=100*(sprandsym(n,d));
A=(A-0.5*spones(A));
x=0.1*(rand(n,1)-rand(n,1));
b=A*x-abs(x);
```

در هر بعد هر مسئله 10 بار تولید می‌شود و میانگین نتایج عددی در قالب جدول 1 ارائه می‌شود. در این جدول n نشان دهنده اندازه ماتریس A است و k و $Time(s)$ به ترتیب میانگین تعداد تکرارها و میانگین زمان برحسب ثانیه را نشان می‌دهند. همچنین



ثانیه باشد و یا اگر $f^* > 10^{-6}$ آنگاه در جدول به جای k ، f^* و $Time(s)$ علامت ” - “ قرار می‌دهیم.

جدول ۱: نتایج عددی

n	PSM			$PDCA_e(\phi_1)$			$PDCA_e(\phi_2)$			NMA		
	k	f^*	$Time(s)$	k	f^*	$Time(s)$	k	f^*	$Time(s)$	k	f^*	$Time(s)$
۱۰۰	۳	$1.3014e-13$	۰/۰۰	۶	$3.1549e-09$	۰/۰۱	۶	$6.7541e-09$	۰/۰۱	۳	$3.3234e-14$	۰/۰۰
۲۰۰	۴	$4.8438e-13$	۰/۰۰	۱۵	$6.1291e-09$	۰/۱۴	۱۵	$4.2898e-08$	۰/۱۴	۳	$5.5826e-13$	۰/۰۲
۳۰۰	۴	$5.7916e-13$	۰/۰۱	۱۱	$3.1076e-09$	۰/۳۳	۱۱	$4.9451e-08$	۰/۳۵	۳	$1.5016e-13$	۰/۰۷
۴۰۰	۴	$2.1847e-12$	۰/۰۳	۱۵	$5.3480e-09$	۱/۰۶	۱۵	$4.9031e-08$	۱/۱۳	۳	$8.9333e-14$	۰/۱۵
۵۰۰	۳	$1.4051e-12$	۰/۰۵	۲۱	$4.1335e-09$	۲/۹۹	۲۲	$4.9321e-08$	۳/۱۳	۳	$4.7710e-14$	۰/۳۱
۶۰۰	۴	$2.5018e-12$	۰/۰۹	۲۸	$5.1572e-09$	۶/۹۷	۳۴	$2.7704e-07$	۸/۶۴	۴	$7.6831e-14$	۰/۶۵
۷۰۰	۴	$4.9005e-12$	۰/۱۲	۴۸	$2.5569e-08$	۱۷/۹۷	۴۹	$5.0799e-07$	۱۸/۴۰	۴	$1.6743e-13$	۰/۸۸
۸۰۰	۳	$1.8771e-12$	۰/۱۸	۹	$1.7880e-08$	۵/۴۳	۱۰	$4.9939e-07$	۶/۵۷	۳	$7.4302e-13$	۱/۲۲
۹۰۰	۳	$2.3440e-12$	۰/۲۵	۱۰	$4.8424e-09$	۸/۴۹	۱۲	$5.0099e-07$	۱۰/۴۴	۳	$4.8301e-13$	۱/۸۰
۱۰۰۰	۴	$2.2558e-11$	۰/۳۵	۱۸۷	$1.2343e-08$	۲۱۱/۲۱	۲۳۵	$5.4557e-07$	۲۶۵/۱۲	۴	$7.5471e-13$	۲/۸۷
۱۲۰۰	۳	$8.5001e-12$	۰/۵۰	۶۰	$5.7357e-09$	۱۱۳/۳۴	۶۴	$5.0522e-07$	۱۲۳/۴۰	۳	$1.4848e-12$	۴/۴۶
۱۴۰۰	۴	$6.2183e-12$	۰/۷۵	۲۲	$1.7305e-08$	۶۵/۰۷	۲۷	$5.0418e-07$	۸۲/۳۸	۳	$4.7878e-13$	۶/۹۲
۱۶۰۰	۴	$1.2471e-11$	۱/۱۵	۵۶	$2.3000e-08$	۲۴۴/۲۴	۷۱	$5.2194e-07$	۳۱۳/۱۲	۳	$1.3944e-12$	۹/۹۳
۱۸۰۰	۴	$2.6274e-11$	۱/۶۴	۷۷	$1.8945e-08$	۴۷۰/۲۰	۹۸	$5.0848e-07$	۵۹۸/۷۰	۳	$3.8787e-12$	۱۴/۳۴
۲۰۰۰	۴	$7.3268e-11$	۲/۱۱	۲۹	$2.0366e-06$	۲۵۰/۲۱	۲۸	$6.7804e-06$	۲۵۴/۹۴	۳	$3.2740e-11$	۲۰/۲۴
۲۵۰۰	۴	$1.5833e-11$	۳/۷۰	۲۱	$3.5403e-06$	۳۶۰/۳۲	۲۱	$7.2616e-06$	۳۶۳/۴۷	۳	$6.4072e-13$	۳۷/۴۷
۳۰۰۰	۴	$1.9845e-11$	۶/۱۸	-	-	-	-	-	-	۳	$1.2198e-12$	۶۴/۵۹
۳۵۰۰	۴	$1.1487e-10$	۹/۷۲	-	-	-	-	-	-	۴	$1.4824e-12$	۱۱۴/۰۵
۴۰۰۰	۴	$4.8631e-11$	۱۳/۹۹	-	-	-	-	-	-	۴	$1.7890e-08$	۱۶۵/۸۴
۴۵۰۰	۴	$1.8675e-10$	۱۹/۱۴	-	-	-	-	-	-	۳	$3.4833e-09$	۲۲۳/۲۱
۵۰۰۰	۴	$7.5360e-11$	۲۵/۷۱	-	-	-	-	-	-	۳	$3.0396e-12$	۲۹۶/۱۷
۵۵۰۰	۴	$5.9422e-11$	۳۵/۵۴	-	-	-	-	-	-	۴	$2.7493e-8$	۴۱۶/۶۳
۶۰۰۰	۴	$8.8736e-11$	۴۱/۲۰	-	-	-	-	-	-	۳	$3.8974e-12$	۵۳۱/۲۰
۶۵۰۰	۳	$8.3807e-11$	۵۲/۳۸	-	-	-	-	-	-	-	-	-
۷۰۰۰	۴	$1.0264e-10$	۷۱/۸۵	-	-	-	-	-	-	-	-	-
۷۵۰۰	۴	$3.8926e-10$	۸۲/۲۵	-	-	-	-	-	-	-	-	-
۸۰۰۰	۴	$5.8126e-11$	۱۶۳/۷۷	-	-	-	-	-	-	-	-	-

بر اساس نتایج جدول ۱، روش‌های PSM و $PDCA_e$ ، روش‌های کارایی برای حل دستگاه معادلات قدر مطلق هستند. همانطور که مشاهده می‌شود روش PSM از نظر دقت به روش NMA نزدیک است، اما این روش از سرعت بسیار بالاتری برخوردار بوده و در ابعاد بزرگ

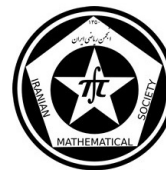


مراجع

- [1] V. Edalatpour, D. Hezari and D. K. Salkuyeh, *A generalization of the Gauss Seidel iteration method for solving absolute value equations*, Appl. Math. Comput. 293 (2017), 156–167.
- [2] C.-X. Li, *A modified generalized newton method for absolute value equations*, J. Optim. Theory Appl. 170 (2016), 1055–1059.
- [3] P. L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM Journal on Numerical Analysis. 16 (1979), 964–979.
- [4] O. L. Mangasarian, *A generalized Newton method for absolute value equations*, Optim. Lett. 3 (2009), 101–108.
- [5] O. L. Mangasarian, *Absolute value equation solution via concave minimization*, Optim. Lett. 1 (2007), 3–8.
- [6] O. L. Mangasarian, *Primal-dual bilinear programming solution of the absolute value equation*, Optim. Lett. 1 (2007), 3–8.
- [7] H. Moosaei, S. Ketabchi, M. A. Noor, J. Iqbal and V. Hooshyarbakhsh, *Some techniques for solving absolute value equations*, Appl. Math. Comput.. 268 (2015), 696–705.
- [8] G. Ning and Y. Zhou, *An improved differential evolution algorithm for solving absolute value equations*, High Performance Computing and Applications. (2016), 38–47.
- [9] G. B. Passty, *Ergodic convergence to a zero of the sum of monotone operators in Hilbert space*, J. Math. Anal. Appl. . 72 (1979), 383–390.
- [10] B. Wen, X. Chen and T. K. Pong, *A proximal difference-of-convex algorithm with extrapolation*, Comput. Optim. Appl. , 69 (2018), 297–324.

پست الکترونیکی: samira.tmoalem@gmail.com, s-shahsavari@phd.guilan.ac.ir

پست الکترونیکی: sketabchi@guilan.ac.ir



کاربردهایی از دترمینان

محمدظاهر کاظمی بانه[†] و صدیق حیدریان[‡]

[†] دانشگاه کردستان، سنندج، ایران

[‡] دانشگاه کردستان، سنندج، ایران

چکیده

در این مقاله، بعضی از کاربردهای دترمینان را در زمینه‌های تعیین تحدب و تقعر توابع، بخش‌پذیری بر اعداد اول، اثبات نابرابری‌ها، محاسبه‌ی خارج قسمت دو سری توانی و بحث در مورد علامت و ریشه‌های معادلات درجه‌ی سوم مورد مطالعه قرار می‌دهیم. واژه‌های کلیدی: توابع محدب، دترمینان، سری توانی، معادله‌ی درجه سه. طبقه‌بندی موضوعی [۲۰۱۰]: 65F40.

۱ پیش‌گفتار

دترمینان یک ماتریس، مانند مشتق یک تابع، در بسیاری از موارد، اطلاعات مفیدی راجع به خود ماتریس و خواص آن در اختیار ما قرار می‌دهد. مفهوم دترمینان در خیلی از کاربردهای جبرخطی، در هندسه و آنالیز نقش مهمی را ایفا می‌کند. از جمله‌ی این کاربردها می‌توان به کاربرد در اثبات نابرابری‌ها و اتحادهای جبری، تجزیه‌ی عبارت‌های جبری متقارن، همچنین بحث در مورد ریشه‌های معادلات درجه‌ی دوم و سوم در جبر مقدماتی، کاربرد در بخش‌پذیری بر اعداد اول و حل بعضی از مسئله‌های تئوری اعداد، کاربرد در به‌دست آوردن تاو یا کرل یک میدان برداری F در $\mathbb{R}^3$ و ارتباط سه عملگر تاو، گرادیان و دیورژانس و نیز تعیین بردار واحد قراردادی رویه‌ای هموار مانند φ ، و برهان قضیه‌ی استوکس و همچنین نوشتن معادله‌های قائم دوم و صفحه‌ی بوسان و صفحه‌ی راست‌گرد (رکتیفیان) در آنالیز برداری، کاربرد در حل بعضی از مسائل جبر مجرد و ... اشاره نمود.

بررسی عملکرد نگاشت‌های خطی در صفحه و فضای سه‌بعدی و n بعدی و تعیین حجم متوازی‌السطوح n بعدی توسط دترمینان صورت می‌گیرد. در میان روش‌های مختلفی که برای تعیین نوع مقاطع مخروطی در صفحه، و تعیین نوع رویه‌های درجه دوم در فضای سه‌بعدی وجود دارد، می‌توان گفت که استفاده از دترمینان، آسان‌ترین و زیباترین آن‌هاست. از دترمینان برای تعیین نوع تابع‌های محدب و مقعر یک متغیره و تعبیر هندسی آن‌ها استفاده می‌کنیم.

اهمیت دترمینان فقط در کاربردهای محاسباتی آن نیست، بلکه، دترمینان ما را قادر می‌سازد تا معکوس یک ماتریس و مجموع توان‌های طبیعی n عدد طبیعی متوالی ابتدا از یک و مشتق تابع‌های ضمنی و همچنین مشتق n -ام یک تابع مانند h که به صورت خارج قسمت دو



تابع دیگر مانند f, g است را به صورت فرمول‌های فشرده‌ای بیان کنیم. ژاکوبین یا سازه‌ی بزرگ‌کن مساحت در صفحه و سازه‌ی بزرگ‌کن حجم در فضای سه بعدی و یا همان فرمول تعویض متغیر در انتگرال‌های چندگانه به طور طبیعی در قالب دترمینان ظهور پیدا می‌کند. دترمینان به ما این توانایی را می‌دهد که، رونسکین تابع‌ها و معادله‌ی دیفرانسیل خانواده‌ای از منحنی‌ها و همچنین نوع نقاط ماکزیمم و مینیمم و یا نقاط زینی را در تابع‌های چندمتغیره تعیین نماییم. مقدارهای ویژه‌ی یک ماتریس همان ریشه‌های چندجمله‌ای مشخصه یا مفسر و یا چندجمله‌ی سرشت‌نمای آن ماتریس هستند، که به کمک دترمینان تعریف می‌شود و ضرایب چندجمله‌ای، همگی از دترمینان آن ماتریس و جمع مینورهای آن دترمینان ظاهر می‌گردند. در این مقاله، در حد توان با جمع‌آوری روش‌های وابسته به دترمینان و کاربردهای آن که معمولاً در شاخه‌های مختلف ریاضی و منابع گوناگون، به طور پراکنده مطرح شده‌اند، در این زمینه قدم برداشتیم.

۲ تعیین تحدب و تقعر توابع با استفاده از دترمینان

تعریف ۱. تابع f را در فاصله $[a, b]$ محدب می‌گوییم، اگر و تنها اگر برای هر دو عدد حقیقی a, b از این بازه و هر λ متعلق به بازه $(0, 1)$ ، داشته باشیم؛

$$f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b)$$

تعبیر هندسی این نابرابری این است که پاره‌خطی که نقطه‌های $(a, f(a))$ و $(b, f(b))$ را روی نمودار تابع f در بازه $[a, b]$ به هم وصل می‌کند، روی نمودار یا بالای آن قرار دارد. و این یعنی اینکه نمودار تابع f همانند شکل ۱ کمائی از آن را که بین نقطه‌های $A_1(x_1, y_1)$ و $A_2(x_2, y_2)$ واقع است، در نظر بگیریم، با توجه به اینکه $y_1 = f(x_1)$ و $y_2 = f(x_2)$ ، در این صورت، با فرض اینکه

$$x_1 < x < x_2, \quad \lambda = \frac{x_2 - x}{x_2 - x_1}, \quad 1 - \lambda = \frac{x - x_1}{x_2 - x_1}$$

در سمت چپ نابرابری

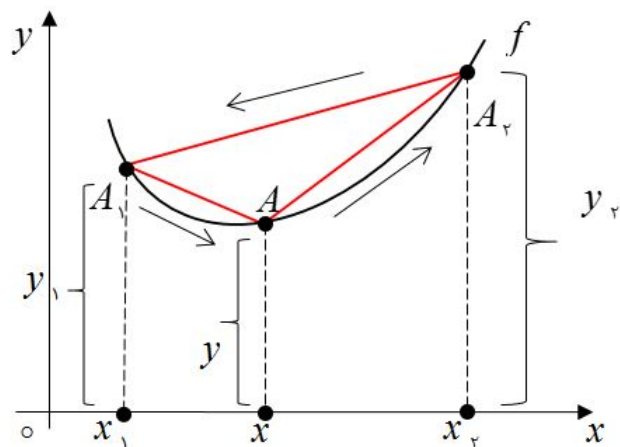
$$f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2)$$

عرض نقطه A از نمودار f روی کمان A_1A_2 به طول x را داریم و در سمت راست این نابرابری، عرض نقطه B از وتر A_1A_2 با همان طول x را خواهیم داشت، یعنی؛

$$y = \left(\frac{x_2 - x}{x_2 - x_1} \right) y_1 + \left(\frac{x - x_1}{x_2 - x_1} \right) y_2$$

پس تابع محدب به این ترتیب معلوم می‌شود که، همه‌ی نقطه‌های کمان دلخواه نمودار آن، زیر نقطه‌های نظیر وتر یا منطبق بر آنها باشند.

نکته ۲. در حالت تقعر تابع، به جای زیر باید بالا گفته شود.



شکل ۱: تعبیر هندسی تابع محدب

قضیه ۳. تابع f روی یک بازه محدب است اگر و فقط اگر به ازای هر سه عدد از این بازه مانند x_2, x, x_1 که در شرط $x_1 < x < x_2$ صدق کنند، داشته باشیم:

$$\begin{vmatrix} 1 & 1 & 1 \\ x_1 & x & x_2 \\ f(x_1) & f(x) & f(x_2) \end{vmatrix} \geq 0$$

در صورت تقعر تابع جهت نامساوی عوض می شود.

مثال ۴. برای هر سه عدد حقیقی a, b, c با شرط $0 < a < b < c$ ثابت کنید:

$$a^c \cdot c^b \cdot b^a \leq a^b \cdot b^c \cdot c^a.$$

□

اثبات. برای حل مسئله، تابع مقعر $f(x) = \ln x$ را روی $(0, +\infty)$ در نظر می گیریم.

مثال ۵. برای سه عدد حقیقی مثبت a, b, c با شرط $0 < a < b < c$ ثابت کنید:

$$\frac{a}{b} + \frac{b}{c} + \frac{c}{a} \geq \frac{b}{a} + \frac{c}{b} + \frac{a}{c}.$$



□

اثبات. برای حل مسئله، تابع محدب $f(x) = \frac{1}{x}$ را روی $(0, +\infty)$ در نظر می‌گیریم.

مثال ۶. برای اعداد حقیقی x, y, z با شرط $x < y < z$ ثابت کنید:

$$xy^2 + yz^2 + zx^2 \geq x^2y + y^2z + z^2x.$$

□

اثبات. برای این منظور تابع محدب $f(x) = x^2$ را در نظر می‌گیریم.

۳ اثبات نابرابری‌ها به وسیله‌ی دترمینان

می‌دانیم که برای هر سه عدد حقیقی a, b و c رابطه‌ی زیر همواره درست است:

$$(a-b)^2 + (a-c)^2 + (b-c)^2 \geq 0,$$

با توسیع و مرتب کردن آن داریم:

$$\begin{vmatrix} b & c \\ a & b \end{vmatrix} - \begin{vmatrix} a & c \\ c & b \end{vmatrix} + \begin{vmatrix} a & b \\ c & a \end{vmatrix} \geq 0.$$

لذا برای هر سه عدد حقیقی a, b و c دترمینان ماتریس 3×3 زیر همواره مقداری مثبت است.

$$\begin{vmatrix} 1 & 1 & 1 \\ a & b & c \\ c & a & b \end{vmatrix} \geq 0.$$

از این مطلب برای اثبات نامساوی‌های جبری استفاده می‌کنیم. در ضمن به خاطر سپردن این دترمینان خیلی ساده است.

مثال ۷. برای هر سه عدد حقیقی غیر صفر x, y, z نابرابری زیر را ثابت کنید:

$$\frac{1}{x^2} + \frac{1}{y^2} + \frac{1}{z^2} \geq \frac{x+y+z}{xyz}$$

□

اثبات. برای اثبات دترمینان همواره نامنفی زیر را در نظر می‌گیریم:

$$\begin{vmatrix} 1 & 1 & 1 \\ \frac{1}{x} & \frac{1}{y} & \frac{1}{z} \\ \frac{1}{z} & \frac{1}{x} & \frac{1}{y} \end{vmatrix} \geq 0$$

مثال ۸. برای هر سه عدد حقیقی غیر صفر x, y, z نابرابری زیر را ثابت کنید:

$$\frac{x^2}{y^2} + \frac{y^2}{z^2} + \frac{z^2}{x^2} \geq \frac{x}{z} + \frac{y}{x} + \frac{z}{y}.$$



□

$$\begin{vmatrix} 1 & 1 & 1 \\ \frac{x}{y} & \frac{y}{z} & \frac{z}{x} \\ \frac{z}{x} & \frac{x}{y} & \frac{y}{z} \end{vmatrix} \geq 0$$

اثبات. برای اثبات، دترمینان همواره نامنفی زیر را در نظر می‌گیریم؛

مثال ۹. برای هر سه عدد حقیقی x, y, z و نابرابری زیر را ثابت کنید؛

$$x^2 y^2 + x^2 z^2 + y^2 z^2 \geq xyz(x + y + z). \quad (3.1)$$

□

$$\begin{vmatrix} 1 & 1 & 1 \\ xy & yz & xz \\ xz & xy & yz \end{vmatrix} \geq 0$$

اثبات. برای اثبات، دترمینان همواره نامنفی زیر را در نظر می‌گیریم؛

مثال ۱۰. برای هر سه عدد حقیقی x, y, z و نابرابری زیر را ثابت کنید؛

$$x^4 + y^4 + z^4 \geq xyz(x + y + z).$$

□

$$\begin{vmatrix} 1 & 1 & 1 \\ x^2 & y^2 & z^2 \\ z^2 & x^2 & y^2 \end{vmatrix} \geq 0$$

اثبات. همانند مثال‌های قبل دترمینان نامنفی زیر را در نظر می‌گیریم؛

مثال ۱۱. برای هر سه عدد حقیقی و مثبت x, y, z و نابرابری زیر را ثابت کنید؛

$$\frac{1}{\sqrt{xy}} + \frac{1}{\sqrt{xz}} + \frac{1}{\sqrt{yz}} \leq \frac{1}{x} + \frac{1}{y} + \frac{1}{z}$$

□

$$\begin{vmatrix} 1 & 1 & 1 \\ \frac{1}{\sqrt{x}} & \frac{1}{\sqrt{y}} & \frac{1}{\sqrt{z}} \\ \frac{1}{\sqrt{z}} & \frac{1}{\sqrt{x}} & \frac{1}{\sqrt{y}} \end{vmatrix} \geq 0$$

اثبات. برای اثبات نابرابری، کافی است دترمینان همواره نامنفی زیر را بسط دهیم.

مثال ۱۲. نابرابری زیر را برای هر سه عدد حقیقی و مثبت x, y, z ثابت کنید؛

$$x + y + z \geq \sqrt{xy} + \sqrt{xz} + \sqrt{yz}$$

□

$$\begin{vmatrix} 1 & 1 & 1 \\ \sqrt{x} & \sqrt{y} & \sqrt{z} \\ \sqrt{z} & \sqrt{x} & \sqrt{y} \end{vmatrix} \geq 0$$

اثبات. برای اثبات نابرابری، کافی است دترمینان همواره نامنفی زیر را بسط دهیم.



۴ کاربرد دترمینان در بخش‌پذیری بر اعداد اول

برای وارد شدن به بحث می‌گوییم، به هر عدد مانند $\overline{Ab}$ ماتریسی سطری مانند $\begin{bmatrix} A & b \end{bmatrix}$ متناظر می‌شود، که کاملاً آن را مشخص می‌کند. در ضمن، در این عدد b یکان و A تعداد ده‌تایی‌های عدد $\overline{Ab}$ است. توجه شود که A ممکن است چند رقمی باشد.

مثال ۱۳. در عدد ۳۷۵۶ ماتریس سطری $\begin{bmatrix} ۳۷۵ & ۶ \end{bmatrix}$ را در نظر می‌گیریم، یعنی $b = ۶$ یکان عدد و $A = ۳۷۵$ تعداد ده‌تایی‌های عدد ۳۷۵۶ یعنی $\overline{Ab}$ است.

قضیه ۱۴. اگر $\overline{Ab}$ و $\overline{Cd}$ دو عدد بزرگتر از ۹ باشند، و $\overline{Ab}$ عددی اول باشد، همچنین داشته باشیم:

$$\begin{vmatrix} C & d \\ A & b \end{vmatrix} = k \cdot \overline{Ab}, \quad k \in \mathbb{Z}$$

در این صورت $\overline{Cd}$ بر $\overline{Ab}$ بخش‌پذیر است، یعنی $\overline{Ab} \mid \overline{Cd}$.

اکنون از این قضیه برای بخش‌پذیری بر اعداد اول بزرگتر از ۹ استفاده می‌کنیم.

مثال ۱۵. نشان دهید که ۳۵۰۳ بر عدد ۳۱ بخش‌پذیر است.

حل:

$$\begin{vmatrix} ۳۵۰ & ۳ \\ ۳ & ۱ \end{vmatrix} = ۳۵۰ - ۹ = ۳۴۱ \Rightarrow \begin{vmatrix} ۳۴ & ۱ \\ ۳ & ۱ \end{vmatrix} = ۳۴ - ۳ = ۳۱$$

چون $۳۱ \mid ۳۴۱$ پس $۳۱ \mid ۳۵۰۳$.



۵ دستور صریح محاسبه‌ی خارج قسمت دو سری توانی به کمک دترمینان

فرض کنیم که $A = \sum_{n=0}^{\infty} a_n x^n$ و $B = \sum_{n=0}^{\infty} b_n x^n$ و $b_0 \neq 0$ می‌خواهیم ضرایب خارج قسمت A بر B را تعیین کنیم، اگر $C = \sum_{n=0}^{\infty} c_n x^n$ خارج قسمت A بر B باشد، در آن صورت $A = B.C$ که اگر ضرایب جمله‌های متناظر را در دو طرف مساوی هم قرار دهیم، نتیجه می‌شود؛

$$a_n = \sum_{k=0}^n b_{n-k} c_k, \quad n = 0, 1, 2, 3, \dots$$

و از این دستگاه مثلثی زیر حاصل می‌شود:

$$a_0 = b_0 c_0$$

$$a_1 = b_1 c_0 + b_0 c_1$$

$$a_2 = b_2 c_0 + b_1 c_1 + b_0 c_2$$

$$a_3 = b_3 c_0 + b_2 c_1 + b_1 c_2 + b_0 c_3$$

$$a_4 = b_4 c_0 + b_3 c_1 + b_2 c_2 + b_1 c_3 + b_0 c_4$$

$$\vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots$$

$$a_n = b_n c_0 + b_{n-1} c_1 + b_{n-2} c_2 + \dots + b_1 c_{n-1} + b_0 c_n$$



این دستگاه برحسب $c_0, c_1, c_2, \dots, c_n$ یک دستگاه $(n+1)$ معادله‌ی $(n+1)$ متغیره است، و دترمینان ضرایب آن b_0^{n+1} است. پس بنا به دستور کرامر داریم:

$$c_n = \frac{\begin{vmatrix} b_0 & 0 & 0 & \dots & 0 & a_0 \\ b_1 & b_0 & 0 & \dots & 0 & a_1 \\ b_2 & b_1 & b_0 & \dots & 0 & a_2 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ b_{n-1} & b_{n-2} & b_{n-3} & \dots & b_0 & a_{n-1} \\ b_n & b_{n-1} & b_{n-2} & \dots & b_1 & a_n \end{vmatrix}}{\begin{vmatrix} b_0 & 0 & 0 & \dots & 0 & 0 \\ b_1 & b_0 & 0 & \dots & 0 & 0 \\ b_2 & b_1 & b_0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ b_{n-1} & b_{n-2} & b_{n-3} & \dots & b_0 & 0 \\ b_n & b_{n-1} & b_{n-2} & \dots & b_1 & b_0 \end{vmatrix}}$$

$$= \frac{1}{b_0^{n+1}} \begin{vmatrix} b_0 & 0 & 0 & \dots & 0 & a_0 \\ b_1 & b_0 & 0 & \dots & 0 & a_1 \\ b_2 & b_1 & b_0 & \dots & 0 & a_2 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ b_{n-1} & b_{n-2} & b_{n-3} & \dots & b_0 & a_{n-1} \\ b_n & b_{n-1} & b_{n-2} & \dots & b_1 & a_n \end{vmatrix}$$

۶ بحث در مورد تعداد و وضعیت ریشه‌های معادله‌ی درجه‌ی سوم به کمک دترمینان‌ها

هر معادله‌ی درجه‌ی سوم در حالت کلی به شکل $ax^3 + bx^2 + cx + d = 0$ می‌باشد که در آن، $a \neq 0$ است. چون a غیر صفر است پس با تبدیل x به عبارت $(x - \frac{b}{3a})$ معادله‌ی مورد نظر به صورت $x^3 + px + q = 0$ در می‌آید. اکنون اگر فرض کنیم که x_1 و x_2 و x_3 ریشه‌های این معادله باشند، بحث در مورد ریشه‌های آن به بحث در مورد مقدار دترمینان زیر منجر



می‌شود. لذا فرض می‌کنیم که

$$\delta = \begin{vmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ x_1^2 & x_2^2 & x_3^2 \end{vmatrix} = (x_1 - x_2)(x_2 - x_3)(x_3 - x_1)$$

زیرا این دترمینان، دترمینان واندرموند مرتبه‌ی ۳ است.

$$\Delta = \delta^2 = (x_1 - x_2)^2(x_2 - x_3)^2(x_3 - x_1)^2$$

با توجه به این‌که اگر $x_1 = x_2 = x_3$ یا در آن صورت دو ستون یا هر سه ستون دترمینان فوق صفر و لذا $\delta = 0$ ، در نتیجه $\Delta = 0$ است.

حال فرض کنیم که در حالت کلی ریشه‌های این معادله یعنی x_1 و x_2 و x_3 متمایز باشند. چون هر سه در معادله صدق می‌کنند لذا؛

$$x_1^2 + px_1 + q = 0, \quad x_2^2 + px_2 + q = 0, \quad x_3^2 + px_3 + q = 0$$

از تفاضل دویه‌دوی این برابری‌ها به دست می‌آوریم،

$$(x_1 - x_2)^2 = -(p + 3x_1x_2), \quad (x_2 - x_3)^2 = -(p + 3x_2x_3), \quad (x_3 - x_1)^2 = -(p + 3x_1x_3)$$

از ضرب این برابری‌ها در یک‌دیگر داریم؛

$$\Delta = \delta^2 = (x_1 - x_2)^2(x_2 - x_3)^2(x_3 - x_1)^2 \implies \Delta = \delta^2 = -(4p^3 + 27q^2)$$

همچنانکه قبلاً ثابت شد، حداقل دو ریشه‌ی معادله با هم برابرند، هرگاه $\Delta = 0$. حال سه ریشه‌ی معادله، حقیقی و متمایزند هرگاه $\delta = (x_1 - x_2)(x_2 - x_3)(x_3 - x_1)$ عددی حقیقی و مخالف صفر شود، و در این صورت $\Delta > 0$. چنانچه فرض کنیم که x_1 عددی حقیقی و ریشه‌های $x_2 = \alpha - \beta i$ و $x_3 = \alpha + \beta i$ مختلط و مزدوج هم باشند، در این صورت عبارت δ به صورت زیر در می‌آید؛

$$\delta = 2((x_1 - \alpha)^2 + \beta^2)\beta i$$

$$\Delta = \delta^2 = -(4p^3 + 27q^2) = -4((x_1 - \alpha)^2 + \beta^2)^2\beta^2$$

که می‌توان بحث را به صورت زیر جمع‌بندی نمود:

اگر $p < 0$ و $4p^3 + 27q^2 > 0$ معادله یک ریشه دارد. که در صورت مثبت بودن q ، ریشه، منفی و هنگامی که q منفی باشد، ریشه، مثبت است.

اگر $p < 0$ و $4p^3 + 27q^2 = 0$ معادله یک ریشه‌ی ساده و یک ریشه‌ی دوگانه دارد. که باز هم اگر $q > 0$ در آن صورت ریشه‌ی ساده منفی و ریشه‌ی دوگانه یا مضاعف، مثبت است، و چنانچه $q < 0$ ریشه‌ی ساده مثبت و ریشه‌ی دوگانه منفی خواهد بود.



اگر $p < 0$ و $4p^3 + 27q^2 < 0$ معادله سه ریشه دارد. که در صورت مثبت بودن q یک ریشه منفی و دو ریشه مثبت است. و اگر q برابر صفر باشد، دو ریشه قرینه و ریشه‌ی دیگر صفر است و وقتی که q منفی باشد، یک ریشه مثبت و دو ریشه‌ی دیگر منفی می‌شوند. اگر $P \geq 0$ باشد، آنگاه تابع $f(x) = x^3 + px + q$ را در نظر می‌گیریم که مشتق همواره مثبت دارد، زیرا $f'(x) = 3x^2 + P$ و لذا، تابع همواره صعودی است پس منحنی فقط در یک نقطه محور x ها را قطع می‌کند، بنابراین، معادله‌ی درجه‌ی سوم $x^3 + px + q = 0$ فقط یک ریشه دارد.

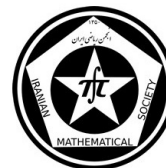
نکته ۱۶. کاربردهای زیادی از دترمینان در زمینه‌های مختلف ریاضی از جمله محاسبه‌ی مشتق n -م یک تابع کسری، بحثی درباره‌ی مقاطع مخروطی، اثبات بعضی اتحادها، تعیین معادله‌ی سرشتنمایی و مقادیر ویژه و حل بعضی مسائل جبر مجرد از نظریه‌ی مدول با دترمینان مورد بحث و بررسی قرار گرفت که خارج از توان این مقاله می‌باشد.

مراجع

- [۱] برنارد کلنن؛ «جبر خطی مقدماتی»، ترجمه‌ی دکتر علی اکبر عالم‌زاده و دکتر مسعود فرزانه، چاپ دوم ۱۳۷۰، انتشارات علوی.
- [۲] حیدریان، صدیق؛ «تعمیم قاعده‌ی ساروس و لاپلاس در بسط دترمینان‌ها و کاربردهای دترمینان»، پایان نامه ارشد ۱۳۹۵، دانشگاه کردستان.
- [۳] سیمور لیشوتس؛ «نظریه و مسائل جبر خطی»، ترجمه‌ی دکتر علی اکبر عالم‌زاده و دکتر مصطفی شاهزادگان، چاپ دوم ۱۳۷۶، موسسه‌ی نشر علوم نوین.
- [۴] شادمان، ارسلان و الهی، کاظم و دیگران؛ «ریاضیات جدید»، چاپ سوم ۱۳۷۷، انتشارات مبتکران.
- [۵] شهشانی، سیاوش و شفیع‌ی ده‌آباد، احمد؛ «زنده باد دترمینان»، مجله‌ی نشر ریاضی، مرکز نشر دانشگاهی، شماره ۲ سال هشتم، شماره‌ی پیاپی ۱۶، چاپ مهرماه ۱۳۷۶ ص ۴۵-۴۶.
- [۶] شهشانی، سیاوش؛ «حساب دیفرانسیل و انتگرال»، جلد دوم، چاپ دوم ۱۳۸۸، انتشارات فاطمی.
- [۷] قصیری، محمدنادر؛ «مروری بر جبر مجرد»، چاپ اول ۱۳۸۸، انتشارات دانشگاه کردستان.
- [8] Sheldon. Axler, *Down with determinants!*, Amer. Math. Monthly, (2) 102 (1995), 139-154.
- [9] K. Hoffman, R. Kunze, Linear Algebra, Prentice- Hull, Inc. 1971.
- [10] T.W. Hungerford, Algebra, Springer Verlag, New York, Inc. 1974.

پست الکترونیکی: zaherkazemi@uok.ac.ir

پست الکترونیکی: sedigh.haidarian@gmail.com



مدل سازی بیماری کرونا

ایمان توکلی[†]، سهراب عفتی[‡]

[†] دانشکده علوم ریاضی، دانشگاه فردوسی، مشهد، ایران

[‡] دانشکده علوم ریاضی، دانشگاه فردوسی، مشهد، ایران

چکیده

دینامیک زیستی به عنوان یک مبحث میان رشته‌ای بین علوم ریاضی و علوم زیستی سبب به وجود آمدن نتایج زیادی در زمینه‌های ریاضیات و زیست شناسی می گردد. در این مطالعه ما به کمک سیستم های دینامیکی و مدل سازی مبتنی بر عامل به پیش بینی رفتار ویروس کوید ۱۹ در جامعه می پردازیم. در این روش افراد جامعه را به چهار گروه مستعد، در معرض بیماری، بیمار و بهبود یافته تقسیم می کنیم. مدل $SEIRS$ با استفاده از تقسیم بندی جامعه ساخته خواهد شد و نتایج عددی آن را در دو نرم افزار متفاوت بررسی و مقایسه خواهیم کرد.

واژه‌های کلیدی: کوید ۱۹، سیستم های دینامیکی، مدل سازی مبتنی بر عامل.

طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

در دسامبر سال ۲۰۱۹، بسیاری از داده های جمع آوری شده پزشکی در کشور چین خبر از یک بیماری ذات الریه ویروسی ناشناخته می داد. علائم اولیه بیماری ویروسی جدید شامل تب، سرفه های خشک و مشکلات تنفسی بود. سرمنشا ابتدایی این بیماری مرموز مربوط به بازار غذاهای دریایی شهر ووهان چین بود. ابتدا در ۶ ژانویه سال ۲۰۲۰ تایید شد که ویروس جدیدی از خانواده کرونا ویروس ها کشف شده است که می تواند انسان را آلوده کند و در ۹ ژانویه، عامل این بیماری ویروسی nCoV-۲۰۱۹ نامیده شد. در تاریخ ۳۰ ژانویه، ویروس نو ظهور سبب راه اندازی ششمین اورژانس بهداشت عمومی در جهان شد و این کنگره ویروس جدید را نه تنها تهدیدی برای چین بلکه برای همه ی کشورها اعلام کرد. در ۱۱ فوریه ۲۰۲۰، سازمان بهداشت جهانی نام رسمی ویروس جدید خانواده کرونا ویروس را کوید ۱۹ انتخاب کرد. تحقیقات ژنومی انجام شده در بازار غذاهای دریایی ووهان، رد پای این ویروس را در خفاش و یا مدفوع آن در اطراف بازار نشان می دهد. طبق اطلاعات فعلی کسب شده، به نظر می رسد میزبان اصلی و طبیعی ویروس کوید ۱۹ خفاش است و سپس ویروس وارد



یک میزبان با اندازه متوسط که احتمالاً مورچه خوار هست شده است و سپس برای فروش در بازار ووهان از آن میزبان به انسان منتقل شده است. سپس روند انتقال انسان به انسان آغاز می شود و اپیدمی موجود شکل گرفته است. این موضوع که به طور خاص از مورچه خوار به عنوان میزبان اولیه ویروس کوید ۱۹ نام برده می شود به این دلیل است که ۷۰ درصد مورچه خواران از نظر تست ویروس کوید ۱۹ مثبت بوده اند. در ۲۰ فوریه سال ۲۰۲۰ میلادی برای اولین بار در ایران، وزارت بهداشت نتایج مثبت دو مورد احتمالی ویروس کرونا را در قم اعلام کرد. مرگ هر دو بیمار مرد توسط وزارت بهداشت یک ساعت بعد تأیید شد. در ۲۳ مارس ۲۰۲۰، تعداد قربانیان ویروس کرونا در ایران به ۱۸ نفر رسید. سازمان بهداشت جهانی در ۳۰ ژانویه اعلام وضعیت اضطراری بین المللی در ارتباط با شیوع کوید ۱۹ کرد [۱]. از آنجایی که ویروس کرونا یک ویروس نو ظهور می باشد، بنابراین برای سیستم ایمنی بدن نا آشنا بوده و مکانیزم موثری برای مقابله با این ویروس وجود ندارد. مطابق با یافته های مرکز کنترل و پیشگیری بیماری ها، تصور می شود تماس فرد به فرد راه اصلی انتقال ویروس کوید ۱۹ باشد. فرض کنید در یک اتوبوس یا اتاق برگزاری جلسه یا یک مهمانی خانوادگی کوچک در کنار یک فرد مبتلا به بیماری کرونا نشسته اید و فرد بیمار ناگهان عطسه یا سرفه می کند. اگر شخص مذکور دهان و بینی خود را نپوشانده باشد، قادر خواهد بود قطرات تنفسی را از طریق دهان یا بینی به سمت شما بفرستد. در این حالت قطراتی که بر روی شما می نشینند احتمالاً حاوی ویروس می باشند یا فرض کنید که با فردی آلوده به ویروس کوید ۱۹ ملاقات می کنید که بینی و دهان خود را با دست لمس کرده است، در این حالت زمانی که با این فرد دست می دهید، ویروس ها از دست وی به دست شما منتقل می شوند. در این وضعیت اگر بدون شستشو، با دست آلوده بینی یا دهان خود را لمس کنید ممکن است تصادفاً یک نقطه برای ورود ویروس به درون بدنتان فراهم نمایید [۲]. افراد می توانند برای کمک به محافظت از خود و دیگران در برابر این بیماری، یک سری از اقدامات محافظتی از جمله شستن مرتب دست ها، پوشیدن ماسک و یادگیری نحوه صحیح استفاده از آن و رعایت فاصله گذاری اجتماعی و قرنطینه خانگی در صورت امکان را انجام دهند تا از گسترش سرسام آور این بیماری در سطح جامعه جلوگیری کنند [۳، ۴، ۵].

۲ سیستم دینامیکی

در این بخش یک سیستم دینامیکی مناسب که بیانگر رفتار بیماری کرونا در جامعه انسانی است را به کمک متغیرهای جدول ۱ فرموله می کنیم. ابتدا تمام افراد جامعه در مقابل بیماری کرونا مستعدینی اند که توانایی مبتلا شدن به این بیماری را بالقوه دارا هستند. اگر افراد مستعد جامعه به دلیل عدم رعایت پروتکل های بهداشتی مخصوصاً رعایت فاصله گذاری اجتماعی در برخورد نزدیک (کمتر از ۲ متر) با یک فرد بیمار کرونایی قرار بگیرند آنگاه به دومین گروه یعنی افراد در معرض ویروس کرونا منتقل می شوند. در گام بعدی با توجه به قدرت سرایت و انتقال ویروس کرونا، سومین گروه یعنی افراد بیمار کرونایی شکل می گیرند که در حقیقت دچار این بیماری عفونی شده اند و ویروس از فرد بیمار به آنها منتقل می شود با روش های شایعی که در پیش گفتار مطرح شد. آخرین گروه در این چرخه مربوط به افراد بهبود یافته است که سلامتی آنها پس از دچار شدن به این بیماری عفونی بازبایی شده است. با توجه به گزارش موارد متعددی از افراد که پیش از یک مرتبه دچار بیماری کرونا شده اند لازم دانستیم که مفهومی تحت عنوان برگشت پذیری بیماری را در این مدل اضافه کنیم که در آن افراد گروه بهبود یافته با ضریب معینی، شانس برگشت به گروه مستعدین و نهایتاً بیمار شدن مجدد آنها وجود دارد. همچنین پارامترهای مورد نیاز برای انتقال بین گروه های تعریف شده و شکل دهی به دستگاه معادلات دیفرانسیل را در ادامه تعریف می کنیم.

جدول ۱: متغیرها

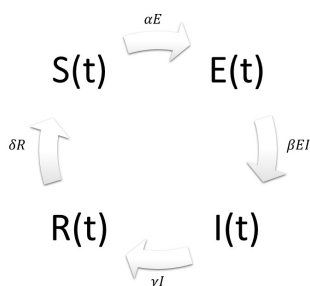
متغیر	تعریف	مقدار اولیه	منبع
$S(t)$	افراد مستعد جامعه	۷۷,۵۹۴,۷۷۲	[۳]
$E(t)$	افراد در معرض بیماری کرونا	۲,۸۱۸,۰۳۵	[۳]
$I(t)$	افراد بیمار کرونایی	۱,۶۹۸,۰۰۵	[۳]
$R(t)$	افراد بهبود یافته	۱,۴۴۹,۳۵۰	[۳]

پارامتر α برای نرخ انتقال افراد از گروه مستعدین به گروه در معرض بیماری، پارامتر β برای نرخ انتقال از گروه در معرض بیماری به گروه بیماران، پارامتر γ برای نرخ بهبودی گروه بیماران و در نهایت پارامتر δ برای نرخ برگشت پذیری بهبود یافتگان به گروه مستعدین با توجه به ماهیت بیماری کرونا که در آن امکان چند مرتبه بیمار شدن افراد وجود دارد. مقادیر عددی و تعاریف خلاصه شده پارامترها در جدول ۲ قابل مشاهده است همچنین فلوجارت متناظر با مدل ارائه شده با توجه به اطلاعات فوق در شکل ۱ به نمایش درآمده است.

جدول ۲: پارامترها

پارامتر	تعریف	مقدار عددی	مراجع
α	نرخ انتقال به گروه در معرض خطر بیماری	۹۶ درصد	[۶]
β	نرخ انتقال به گروه بیماران	۷۱ درصد	[۶]
γ	نرخ بهبودی	۹۵ درصد	[۶]
δ	نرخ بازگشت پذیری مستعدین بیماری	۱۰ درصد	[۷]

شکل ۱: مدل SEIRS

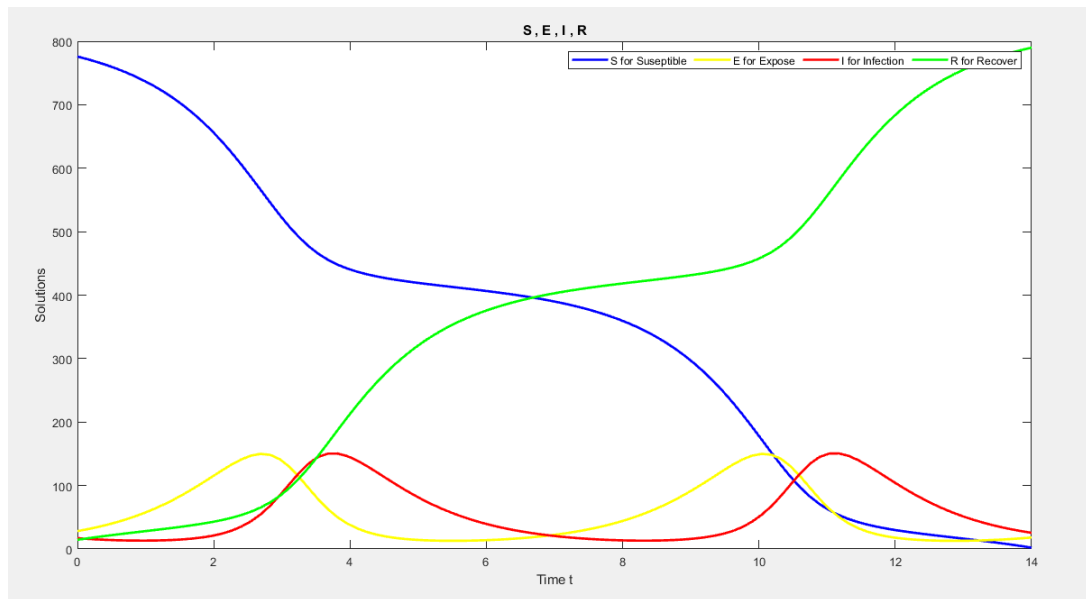


دستگاه معادلات دیفرانسیل متناظر با مدل رفتار بیماری کرونا در جامعه انسانی را به شکل زیر تعریف می کنیم:

$$\begin{cases} \dot{S}(t) = -\alpha E(t) + \delta R(t) \\ \dot{E}(t) = \alpha E(t) - \beta E(t)I(t) \\ \dot{I}(t) = \beta E(t)I(t) - \gamma I(t) \\ \dot{R}(t) = \gamma I(t) - \delta R(t) \end{cases}$$

در شکل ۲ به کمک روش رانگ کوتای دورمند - پرنس در نرم افزار متلب با توجه به مقادیر اولیه جدول ۱، پارامترهای جدول ۲ و در بازه زمانی دو هفته ای نتایج عددی به تصویر کشیده شده است (نمودار عمودی بر حسب ۱۰۰,۰۰۰ نفر).

شکل ۲: نتایج عددی



همانطور که در شکل ۲ مشاهده می کنید رفتار نمودار متناظر با گروه مستعدین در جامعه یک رفتار نزولی است و با گذشت زمان تعداد آنها کمتر و کمتر می شود. از طرفی با توجه به رفتار نمودار گروه مستعدین، جمعیت حاضر در این گروه با سرعت کمی به گروه افراد در معرض بیماری افزوده می شود و سبب صعود نمودار متناظر با آنها می شود. توجه داشته باشید که رفتار مشابهی برای گروه های افراد در معرض بیماری و گروه افراد بیمار رخ می دهد و سبب به وجود آمدن قله ای از بیماری می شود. پس از پشت سر گذاشتن این پیک به تعداد نفرات گروه بهبود یافتگان افزوده می شود و توانان از تعداد افراد گروه بیماران کاسته می شود تا یک پیک بیماری را پشت سر گذاریم. به شکل مشابه این اتفاقات یک مرتبه دیگر رخ می دهد و دو پیک بیماری در این بازه زمانی مشاهده می شود.



۳ مدل سازی مبتنی بر عامل

مدل های مبتنی بر عامل، با محوریت کنش و واکنش بین عوامل و محیط برای خلق بینش دینامیک درون سیستم معرفی می شوند. فاکتور های زیادی به عنوان اجزای اصلی یک مدل عامل محور معرفی می شوند. به عنوان مثال می توان به موارد زیر اشاره کرد: عامل، محیط، کنش و واکنش و حمل و نقل. با توجه به اهداف تحقیق ما، نوع بیماری، می تواند یکی دیگر از اجزای اصلی مدل سازی باشد.

۱.۳ عامل

هر عامل می تواند نماینده یک فرد، یک خانواده یا یک موسسه کوچک باشد. عوامل همیشه منحصر به فرد هستند و تصمیمات را براساس قوانینی که بر مدل حاکم می کنیم، می گیرند و امکان دارد این تصمیمات از هر عامل به عامل دیگر متفاوت باشد.

۲.۳ محیط

محیط در شبیه سازی مبتنی بر عامل، چشم اندازی است که عوامل در آن بر هم کنش دارند. محیط می تواند در نشان دادن پویایی عوامل موثر باشد. بسته به هدف، محیط می تواند یک جاده، یک انسان، یک جامعه یا حتی یک شهر باشد.

۳.۳ کنش و واکنش

کنش و واکنش یا تعاملات بین عوامل تعریف شده، شامل قواعد و قوانینی است که سیستم را شکل می دهد. قوانین می توانند بسیار ساده باشند مانند حرکت عامل در جهات مختلف تا قوانین پیچیده تری مانند یافتن یک عامل خاص.

۴.۳ حمل و نقل

مسئله حمل و نقل، به گونه ای در فاکتور محیط قرار دارد اما دلیل اصلی جداسازی آن به عنوان یک بخش جدا، اهمیت آن در مورد شیوع بیماری های همه گیر در وسایل حمل و نقل عمومی است.

۵.۳ نوع بیماری

فاکتور نوع بیماری که آن هم جز فاکتورهای خاص به شمار می آید و در تحقیق ما مورد استفاده قرار می گیرد، از اجزای مهم در مدل سازی عامل محور در شاخه مدل سازی بیماری ها خواهد بود.

مثال ۱. فرض کنید که در صدد رفع ترافیک یکی از تقاطع های شلوغ شهری باشیم. رفت و آمد خودروها در یک تقاطع شهری قابل مدلسازی و پیاده سازی به زبان سیستم های دینامیکی خواهد بود اما بیاپید به آن از منظر مدلسازی عامل محور نگاهی بیاندازیم. خودروها

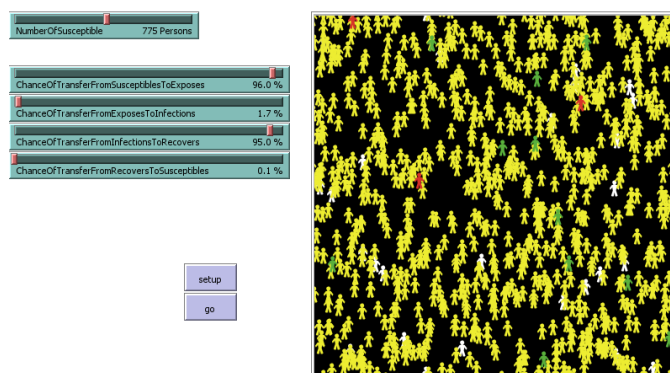
در مدل‌سازی عامل محور نقش عوامل مدل را دارند که در سطح تقاطع دائما در حال حرکت هستند. محیط تقاطع به عنوان محیط مدل‌سازی قلمداد می شود که می تواند به شکل خیابان، سه راه و چهار راه باشد. کشش و واکنش در این مثال می تواند چراغ راهنمایی و رانندگی، میزان فاصله ی بین خودروها، تعداد لاین هایی که خیابان دارد یا حتی میزان رعایت کردن قوانین راهنمایی و رانندگی باشد.

شکل ۳: محیط شبیه سازی ترافیک یک تقاطع



با توجه به ویژگی های بیان شده برای مدل سازی مبتنی بر عامل، جهت پیاده سازی آن، نیازمند ابزار و نرم افزارهای قدرتمند در بحث مدل سازی خواهیم بود. یکی از نرم افزارهای معروف و البته تازه متولد شده در این مورد، نرم افزار نت لوگو* است. به کمک این نرم افزار، توانایی پیاده سازی مدل های عامل محور به بهترین شکل ممکن مقدور خواهد بود. نت لوگو، یک محیط مدل سازی با قابلیت برنامه نویسی برای شبیه سازی پدیده های طبیعی است که یوری ویلنسکی، استاد دانشگاه نورث وسترن، در سال ۱۹۹۹ آن را نوشت و از آن زمان تاکنون در مرکز یادگیری متصل و مدل سازی مبتنی بر کامپیوتر در توسعه مداوم بوده است. نت لوگو خصوصا برای مدل سازی سیستم های پیچیده در طول زمان مناسب است مانند بحث مورد نظر ما یعنی شیوع یک بیماری همه گیر. مدل سازان و برنامه نویسان می توانند دستور العمل هایی را به صد ها یا هزاران عامل تعریف شده بدهند که همه آنها به صورت مستقل از هم دیگر عمل می کنند. این امر ما را قادر می سازد که ارتباطات بین سطح خرد عوامل و الگوهای سطح کلان ناشی از تعامل بین آنها را کاوش کنیم. همچنین نت لوگو این اجازه را به کاربران می دهد که رفتار مدل را تحت شرایط مختلف مورد بررسی قرار دهد [۸].

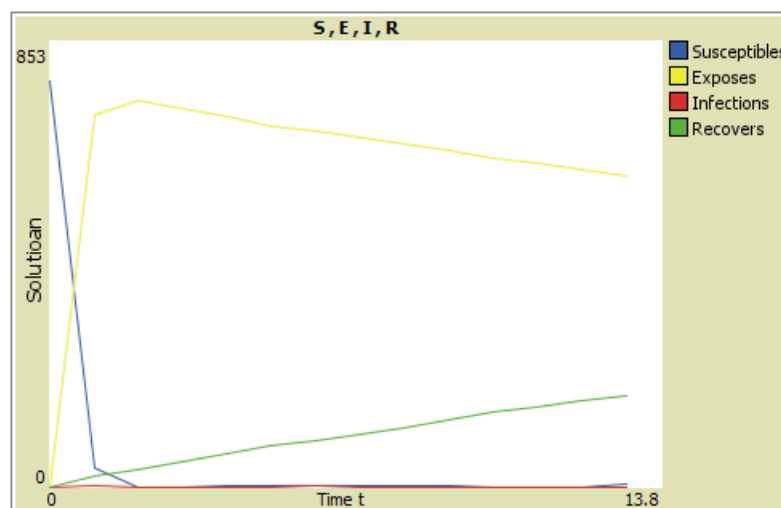
شکل ۴: محیط شبیه سازی شیوع بیماری کرونا در سطح جامعه



NetLogo®

با توجه به مطالب مطرح شده تا اینجا بحث، عوامل ما در شبیه سازی عامل محور، افراد جامعه هستند و محیط شبیه سازی، جامعه ای فرضی در ساده ترین شکل ممکن است. کنش و واکنش بین افراد در این مدل از میزان ضرائب تعیین شده مشخص می گردد و احتمال تبدیل شدن هر یک از عوامل از گروهی به گروهی دیگر با این ضرائب رابطه ای مستقیم دارد. شکل ۴ محیط شبیه سازی در نرم افزار نت لوگو را نمایش می دهد. در شکل ۵ به کمک نرم افزار نت لوگو با توجه به پارامترهای جدول ۱ و مجددا در بازه زمانی دو هفته ای نتایج عددی به تصویر کشیده شده است. همچنین لازم به ذکر است که شرایط اولیه را به شکل یکسان با شرایط اولیه دستگاه معادلات دیفرانسیل در نظر گرفتیم. در شروع اجرای برنامه، با توجه به احتمال بالای تبدیل افراد از گروه مستعد به گروه در معرض بیماری، جا به جایی سریعی بین این دو نمودار رخ می دهد و به یک باره تمام مستعدین جامعه ما به افراد در معرض بیماری تبدیل می شوند و پس از آن گروه افراد در معرض بیماری با شیب ملایمی کم می شوند و به دسته افراد بیمار منتقل می شوند و متعاقبا نمودار متناظر با افراد مستعد به صفر میل می کند. همچنین با توجه به ضریب بسیار کوچک تبدیل افراد از گروه در معرض به گروه بیمار، تغییرات متناظر با نمودار افراد بیمار با رفتار موجی شکل در نزدیکی محور افقی نوسان می کند. در نهایت نمودار متناظر با گروه بهبود یافته با شیب قابل قبولی در حال رشد است.

شکل ۵: نتایج عددی



۴ نتیجه گیری و مقایسه نتایج

بر خلاف انتظار با توجه به طرز نگاه متفاوت سیستم های دینامیکی و مدل های عامل محور به مسئله به سبب ماهیت متفاوت آنها اما رفتار کلی نمودارهای خروجی متناظر با دو روش در دو نرم افزار مطرح شده یکسان است. همانطور که مشاهده کردید جمعیت افراد مستعد روندی کاهشی گرفته و پس از گذشت زمانی به حالت ثابت می رسد. نمودار افراد در معرض بیماری هم در ابتدا با شیب تند حالت صعودی دارد و سپس حالت نزولی و کاهشی می گیرد. اما نمودار متناظر با افراد بیمار محل بحث است. به نظر می رسد تفاوت ماهیت دو روش سبب



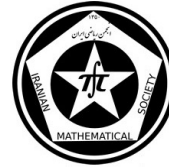
اختلاف نمودارهای این دو تصویر شده است. همانطور که در شکل ۲ مشاهده کردید، دو قله از بیماران در اواسط بازه زمانی ایجاد می شود که در خروجی شکل ۵ چنین چیزی مشاهده نمی شود. در انتها نمودار افراد بهبود یافته رفتار صعودی و مشابه ای در هر دو روش داشته است.

مراجع

- [1] Mohammadzadeh rostami F, Nasr Esfahani B N, Ahadi A M, Shalibeik S. Review of Novel Coronavirus, Severe Acute Respiratory Syndrome Corona virus 2, Iran J Med Microbiol. 2020.
- [2] Hatef Koze chi, How the coronavirus is transmitted, ways of transmission and infection of the coronavirus. <https://www.darmankade.com/blog/corona-transmission>. Iran. 2020.
- [3] Susanna Expositor, Nicola Principi, Chi Chi Leung, Giovanni Battista Migliori, Universal use of face masks for success against COVID-19: evidence and simplification for prevention policies, European Respiratory Journal. Jan 2020.
- [4] <https://www.medicalnewstoday.com/articles/coronavirus-prevention>. Iran. 2020.
- [5] Corona and Mask Disease: Tips for Families. What we need to know about the mask and how the family gets acquainted with it. <https://www.unicef.org/iran>. Iran. 2020.
- [6] <https://covid19.healthdata.org/iran-islamic-republic-of>
- [7] Sangeeta SahaT, G. P. Samanta. Modelling the role of optimal social distancing on disease prevalence of COVID-19 epidemic. Indian Institute of Engineering Science and Technology, Shibpur, Howrah 711103, India. 2020.
- [8] Haitang Hu, Peng Gong, Bing Xu. Spatially explicit agent-based modeling for schistosomiasis transmission: Human–environment interaction simulation and control strategy assessment, 2010. Page 58.

<https://www.amar.org.ir>

<https://www.worldometers.info>



مشخص سازی لایه های شبکه چشم انسان با استفاده از موجک در ماشین بردار پشتیبان

جمشید سعیدیان[†]، الیاس خلیلی پور[‡] و حسین عظیمی[†]

دانشکده علوم ریاضی و کامپیوتر، دانشگاه خوارزمی، تهران، ایران[†]

دپارتمان بیماریهای شبکه، بیمارستان فارابی، دانشگاه علوم پزشکی تهران، تهران، ایران[‡]

چکیده

در این مطالعه، به دنبال مشخص سازی خطوط مختلف در تصاویر *OCT* شبکه از چشم انسان هستیم، این خطوط بر اثر بیماری های متفاوت دچار تغییراتی می شوند که تحلیل این تغییرات برای چشم پزشکان بسیار مفید است. هدف ما یافتن خطوط پلکسی فرم درونی (*IPL*) و پلکسی فرم بیرونی (*OPL*) به صورت خودکار است. بدین منظور ابتدا با استفاده از درونیایی بی-اسپالین مکعبی به یافتن خطوط می پردازیم و سپس، به منظور بالا بردن دقت، از رگرسیون بردار پشتیبان با هسته هایی از نوع توابع پایه ای شعاعی و موجک استفاده می کنیم. روش خودکار پیشنهادی را روی تصاویر مختلف از بیماران بیمارستان فارابی اعمال می کنیم، نتایج به دست آمده در مقایسه با روش دستی که توسط متخصصین انجام می شود دارای دقتی قابل رقابت است و علاوه بر خودکار بودن، با معرفی معیارهای سنجش جدید، پیشنهاداتی را برای تحلیل دقیق تر در اختیار متخصصین می گذارد. واژه های کلیدی: شبکه، تصاویر *OCT*، رگرسیون بردار پشتیبان، توابع پایه ای شعاعی، موجک. طبقه بندی موضوعی [۲۰۱۰]: 65Z05, 65D12, 68T99.

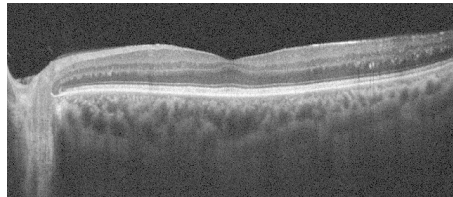
۱ مقدمه

علم ریاضی این امکان را برای بشر فراهم آورده که با مدل سازی اکثر پدیده ها، طبیعت را تحت کنترل خود داشته باشیم و بتوانیم رفتار پدیده ها را تحلیل کنیم. پدیده های طبیعی زیادی به صورت یک تابع رگرسیون مدل بندی می شوند. یکی از این پدیده ها، رفتار دو لایه ی پلکسی فرم بیرونی و پلکسی فرم درونی در شبکه چشم است که بر اثر شرایط متفاوت و بیماری های مختلف رفتار متفاوت و منحصر به فردی از خود نشان می دهند، مطالعه این خطوط به منظور تشخیص بیماری برای چشم پزشکان مورد اهمیت است. رگرسیون بردار پشتیبان مدلی است که به ما کمک می کند تا موفق به یافتن این خطوط با ساخت تابع رگرسیون شویم. نخستین بار در سال ۱۹۶۳ این موضوع را وپنیک در رساله دکتری خود بیان کرد [۲]. امروزه ماشین بردار پشتیبان برای حالات غیرخطی و همچنین ماشین بردار

پشتیبان کمترین مربعات برای مسائل رگرسیون و طبقه‌بندی تعمیم داده شده است. در این مطالعه، با استفاده از رگرسیون بردار پشتیبان و همچنین درونیایی بی-اسپلاین به یافتن خطوط OPL و IPL می‌پردازیم.

۲ پیش پردازش تصاویر

تصاویری که در ادامه از آن‌ها استفاده شده است، از شبکه چشم بیماران بیمارستان فارابی گرفته شده است و تصویر (۱) یکی از برش‌های تصویر OCT است و ما در این مطالعه، عموماً روی برش یازدهم این تصاویر کار می‌کنیم.



شکل ۱: تصویر OCT از سطح شبکه چشم

همانطور که مشاهده می‌کنیم این تصاویر به شدت دارای نویز هستند و خطوط مدنظر در برخی نواحی به سختی قابل تشخیص هستند، به همین جهت لازم است قبل از بررسی این تصاویر، پاره‌ای از عملیات پردازش تصویر روی آن‌ها اعمال شود. این عملیات شامل کاهش نویز تصاویر، چند مرحله فیلترینگ، استفاده از مسطح سازی هیستوگرام و اعمال عملگرهای مورفولوژیک بر روی تصاویر است [۴]. برای کاهش نویز تصاویر از تکنیک میانگین‌گیری غیر محلی کمک می‌گیریم. تصویر P را تصویر نویزی می‌نامیم که به صورت $P = I + P_0$ نشان می‌دهیم که در آن I تصویر اصلی و P_0 مقدار نویز اضافه شده به تصویر است. اگر تصویر نویز زدایی شده را با P' نشان دهیم، آن‌گاه مقدار پیکسل i ام در تصویر نویز زدایی شده، $P'(i)$ ، به صورت ذیل قابل محاسبه است:

$$P'(i) = \sum_j w(i, j)P(j), \quad (2,1)$$

که در آن $w(i, j)$ ها وزن‌های میانگین‌گیری هستند و به شباهت بین پیکسل‌های i و j بستگی دارند. لازم به ذکر است که این وزن‌ها باید در شرط

$$\begin{cases} 0 \leq w(i, j) \leq 1 \\ \sum_j w(i, j) = 1. \end{cases} \quad (2,2)$$

صدق کنند، شباهت بین دو پیکسل i و j بر مبنای یک تابع کاهشی برحسب فاصله وزنی اقلیدسی است که طبق رابطه‌ی

$$D^*(i, j) = \|P(N_i) - P(N_j)\|_{\Psi, a}^2 = \sum_{l \in nl} \{G_a(l) [P(N_i(l)) - P(N - j(l))]\}, \quad (2,3)$$

به دست می‌آید، که $P(N_i)$ و $P(N_j)$ ، بردارهای شدت تراز خاکستری و N_k یک همسایگی با ابعاد ثابت حول پیکسل k ام است، G_a هسته گاوسی و a انحراف معیار آن است. هسته گاوسی استفاده شده در رابطه (۲,۴) نشان داده شده است.

$$G_a(x, y) = \exp \left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2a} \right), \quad (2,4)$$

که در آن x_0 و y_0 مرکز هسته گاوسی هستند. بدین ترتیب وزن‌های مورد نیاز در رابطه (۲,۱) به صورت

$$w(i, j) = \frac{\exp \left(\frac{-D^*(i, j)}{h^2} \right)}{\sum_j \exp \left(\frac{-D^*(i, j)}{h^2} \right)}, \quad (2,5)$$

محاسبه می‌شوند. در این روش، برای هر پیکسل دلخواه مانند i با همسایگی N_i می‌توان پیکسل‌های j با همسایگی N_j یافت که همسایگی آن‌ها دارای شباهت زیادی به همسایگی پیکسل i ام باشد و این پیکسل‌ها مقادیر وزن بزرگتری را به خود اختصاص می‌دهند و در تصویر نویز زدایی روی پیکسل مدنظر تأثیر بیشتری می‌گذارند [۱].

حال تصویر نویز زدایی شده را با روش‌های مختلف فیلتر می‌کنیم. ابتدا از فیلتر میانگین استفاده می‌کنیم تا شدت نوسانات در پیکسل‌های همسایگی تصویر اصلی I را کاهش دهیم. برای اینکار از الگوریتم پنجره متحرک استفاده می‌شود و روی تمام پیکسل‌های تصویر اصلی قرار می‌گیرد و پیکسل جدید را از رابطه (۲,۶) به دست می‌آوریم.

$$J(x, y) = \sum_{i=-k}^k \sum_{j=-k}^k \frac{1}{(2k+1)^2} I(x+i, y+j) \quad (2,6)$$

سپس از فیلتر میانه بهره می‌بریم که الگوریتمی مشابه فیلتر میانگین دارد با این تفاوت که در روش فیلتر میانه، مقدار میانه اعداد قرار گرفته شده در پنجره به مرکز آن نسبت داده می‌شود.

فیلتر بعدی که از آن استفاده می‌کنیم فیلتر گاوسی است که یک نوع فیلتر میانگین وزندار است که مؤلفه‌های پنجره متحرک آن از تابع گاوس به صورت $G(x, y) = \frac{1}{\sqrt{\pi}\sigma^2} \exp \left(-\frac{x^2+y^2}{2\sigma^2} \right)$ که در آن σ یک عدد حقیقی مثبت است، اقتباس می‌گردد. در فیلتر گاوسی ماتریس پنجره $(2k+1) \times (2k+1)$ تایی به صورت رابطه (۲,۷) است.

$$H = \frac{1}{S} \begin{bmatrix} G(-k, -k) & \cdots & G(-, 0) & \cdots & G(-k, +k) \\ \vdots & & \vdots & & \vdots \\ G(0, -k) & \cdots & G(0, 0) & \cdots & G(0, +k) \\ \vdots & & \vdots & & \vdots \\ G(+k, -k) & \cdots & G(+k, 0) & \cdots & G(+k, +k) \end{bmatrix} \quad (2,7)$$



بدین ترتیب اگر H پنجره با بعد دلخواه n باشد، آنگاه مؤلفه سطر i ام و ستون j ام این پنجره به صورت ذیل قابل بیان است

$$h(i, j) = \frac{1}{S} G\left(i - \frac{n+1}{2}, j - \frac{n+1}{2}\right) \quad (2,8)$$

که در آن S پارامتر نرمال کننده است که باعث می شود مجموع مؤلفه های پنجره H برابر یک گردد:

$$S = \sum_{i=-k}^k \sum_{j=-k}^k G\left(i - \frac{n+1}{2}, j - \frac{n+1}{2}\right) \quad (2,9)$$

در ادامه روند پیش پردازش لازم است تا مسطح سازی هیستوگرام انجام شود. برای این منظور، با فرض اینکه L دقت شدت تصویر و N تعداد پیکسل های تصویر باشد، روی مجموعه $\{0, 1, \dots, L-1\}$ تابع چگالی احتمال و تابع توزیع تجمعی احتمال را برای تصویر به صورت ذیل تعریف می کنیم.

$$p(i) = \frac{n_i}{N} = \frac{\text{تعداد پیکسل هایی که شدت رنگ آن ها برابر } i \text{ است}}{\text{تعداد کل پیکسل ها}},$$

$$S(i) = \sum_{j=0}^i p(j) = \sum_{j=0}^i \frac{n_j}{N}.$$

حال تابع تبدیل شدت مناسب را نیز به فرم $(2,10)$ تعریف می کنیم:

$$g(i) = \text{Round}((L-1) \times S(i)) \quad (2,10)$$

در نهایت برای اعمال دو عملگر گسترش و فرسایش، تصاویر I_1 و I_2 را در نظر می گیریم:

$$I_1 = \{r \in E^2\}, \quad I_2 = \{s \in E^2\}.$$

عملگر گسترش طبق رابطه ذیل اعمال می شود:

$$I_1 \oplus I_2 = \{z : z = r + s \quad \forall r \in I_1 \quad \text{and} \quad s \in I_2\}, \quad (2,11)$$

و برای اعمال عملگر فرسایش نیز طبق رابطه $(2,12)$ عمل می کنیم.

$$I_1 \ominus I_2 = \{z : z + s \in I_1 \quad \forall s \in I_2\}. \quad (2,12)$$

حال نیاز داریم تا داده های آموزشی خود را بسازیم و با استفاده از رگرسیون بردار پشتیبان آن ها را برازش دهیم. برای این کار، بعد از پیش پردازش تصویر با استفاده از ویژگی پیکسل هایی که روی خطوط IPL و OPL هستند آن ها را به عنوان داده های آموزشی برچسب گذاری می کنیم.

۳ رگرسیون بردار پشتیبان

مدل ریاضیاتی رگرسیون بردار پشتیبان تلاش می‌کند یک ابرصفحه در فضای ویژگی را به گونه‌ای بر روی داده‌های آموزشی برازش دهد که فاصله تمام داده‌های آموزشی از این ابرصفحه کمتر از یک ابرپارامتر مانند ϵ باشد و درعین حال ابرصفحه تا حد ممکن مسطح باقی بماند. در رگرسیون بردار پشتیبان با حاشیه نرم، از آنجایی که وپنیک معتقد بود مقداری انحراف از ϵ قابل قبول است، از یک تابع زیان به صورت

$$|\xi|_\epsilon = \begin{cases} 0 & \text{if } |\xi| \geq \epsilon, \\ |\xi| - \epsilon & \text{otherwise.} \end{cases} \quad (3,1)$$

استفاده می‌شود تا خطاهایی را که در یک فاصله معین از مقادیر واقعی هستند نادیده بگیرد. این میزان خطا با وارد کردن متغیرهای ξ_k^+ و ξ_k^- و اعمال ضریب جریمه C در مسئله بهینه‌سازی لحاظ می‌شود که در نهایت به مسئله‌ی کمینه‌سازی ذیل می‌رسیم:

$$\begin{aligned} \min_{w,b} \quad & \frac{1}{2} w^T w + C \sum_{k=1}^n (\xi_k^+ + \xi_k^-) \\ \text{s.t.} \quad & y_k - w^T x_k - b \leq \epsilon - \xi_k^+, \quad k = 1, \dots, n, \\ & w^T x_k + b - y_k \leq \epsilon - \xi_k^-, \quad k = 1, \dots, n, \\ & \xi_k^+, \xi_k^- \geq 0, \quad k = 1, \dots, n. \end{aligned} \quad (3,2)$$

در این حالت با محاسبه تابع پیش‌بینی به رابطه (۳,۳) خواهیم رسید.

$$y(x) = \sum_{k=1}^n (\alpha_k^+ - \alpha_k^-) x_k x + b, \quad (3,3)$$

که مقدار b نیز از رابطه (۳,۴) بدست می‌آید.

$$b = \frac{1}{N_V} \sum_{v \in V} \left[y_v - \epsilon - \sum_{m=1}^n (\alpha_m^+ - \alpha_m^-) x_v \cdot x_m \right]. \quad (3,4)$$

برای مسائل غیرخطی نیز از ترفند هسته استفاده می‌شود که در این حالت الگوهای ورودی به فضایی با ابعاد بالاتر نگاشت می‌شوند به طوری که در فضای نگاشت شده می‌توان به صورت خطی رگرسیون‌گیری کرد. در این حالت با محاسبه تابع پیش‌بینی به رابطه (۳,۵) و مقدار b نیز از رابطه (۳,۶) بدست می‌آید.

$$y(x) = \sum_{k=1}^n (\alpha_k^+ - \alpha_k^-) K(x, x_k) + b, \quad (3,5)$$

$$b = \frac{1}{N_V} \sum_{v \in V} \left[y_v - \epsilon - \sum_{m=1}^n (\alpha_m^+ - \alpha_m^-) K(x_v, x_m) \right]. \quad (3,6)$$

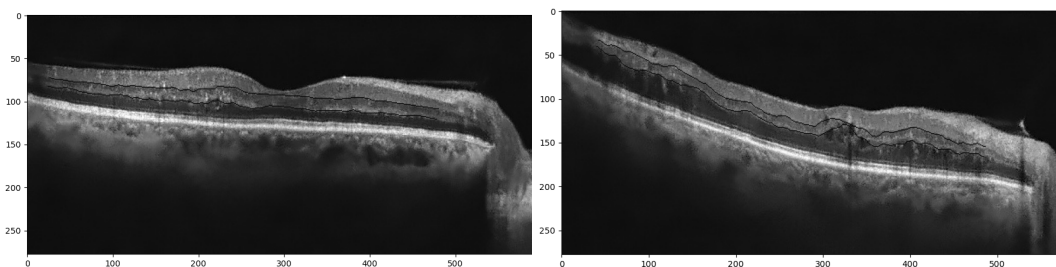
از توابع شناخته شده‌ای که می‌توان به عنوان هسته در ماشین بردار پشتیبان از آن‌ها بهره برد می‌توان به توابع خطی، توابع چندجمله‌ای و توابع سیگموئید نام برد. ما در این مطالعه از دو هسته گاوسی که به عنوان توابع پایه‌ای شعاعی شناخته می‌شوند و موجب استفاده کرده‌ایم. این دو هسته در روابط (3,7) و (3,8) نشان داده شده‌اند [3, 5].

$$K(x, t) = \exp \left(-\frac{\|x - t\|^2}{2\sigma^2} \right), \quad (3,7)$$

$$K(x, t) = \prod_{i=1}^m \cos \left(\sqrt{\lambda_i} \frac{x_i - t_i}{a} \right) \exp \left(-\frac{\|x - t\|^2}{2a^2} \right). \quad (3,8)$$

4 حل مسأله

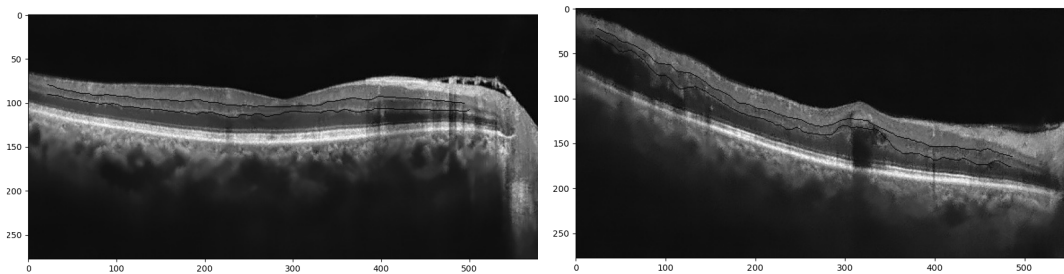
بعد از برچسب‌گذاری پیکسل‌های مورد نظر، با باینری کردن تصویر به روش آستانه‌گذاری نقاطی روی *OPL* و *IPL* باقی می‌ماند و برای مشخص‌سازی خطوط می‌توان به تکنیک‌های تقریب متوسل شد. یک روش معمول استفاده از درونیابی بی-اسپلاین درجه سوم است که به ما کمک می‌کند تا علاوه بر یافتن خطوط تقریب، نوسانات موجود در خطوط را نیز به خوبی نشان دهیم، به علاوه اگر داده‌ی پرتی هم وجود داشت خط حاصل شده انحراف زیادی پیدا نمی‌کند. شکل 2 نشان‌دهنده این موضوع است.



شکل 2: عملکرد روش درونیابی بی-اسپلاین درجه سوم برای یافتن خطوط

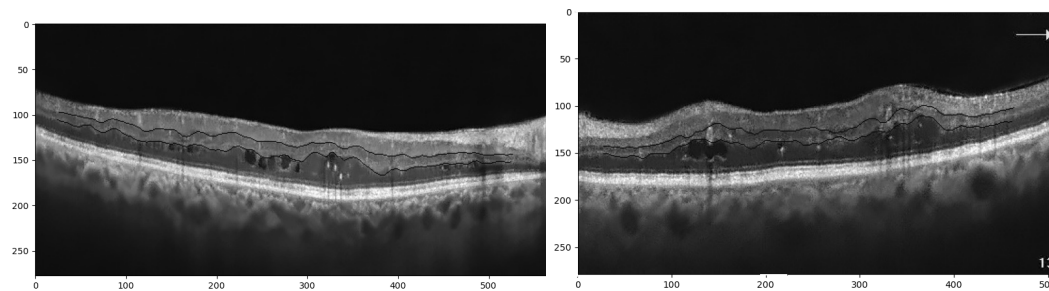
در گام بعدی از رگرسیون بردار پشتیبان استفاده می‌کنیم تا با بهره‌گیری از توابع پیش‌بینی رگرسیون به دنبال مشخص‌سازی خطوط باشیم. برای این کار ما از دو هسته معرفی شده در بخش قبل استفاده می‌کنیم و در نهایت نشان می‌دهیم عملکرد هسته موجکی به‌خاطر دارا بودن خواص توابع موجکی، عملکرد بهتری دارد. برای استفاده از توابع پایه‌ای شعاعی به عنوان هسته‌ی رگرسیون بردار پشتیبان، تابع دو ضابطه‌ای (3,7) را در معادله (3,6) قرار می‌دهیم و بعد از محاسبه مقدار b ، با جایگذاری (3,7) در معادله (3,5) و اعمال

مقدار محاسبه شده b داده‌ها را از فضای خام به فضای هیلبرت بازتولید شده نگاشت می‌کنیم و با اعطای مقدار مناسب ϵ و C عملیات رگرسیون‌گیری را انجام می‌دهیم و خطوط پلکسی فرم را با دقت بالاتری پیدا می‌کنیم. استفاده از این روش کمک می‌کند تا در تصاویری که نور کافی وجود نداشته با استفاده از تابع پیش‌بینی مقدار مناسب محاسبه شده و در آن نقاط قرار داده شود. در شکل ۳ عملکرد این روش را مشاهده می‌کنیم.



شکل ۳: عملکرد روش رگرسیون بردار پشتیبان با هسته RBF

در گام نهایی از $(3, 8)$ به عنوان هسته در رابطه $(3, 5)$ استفاده می‌کنیم با این تفاوت که استفاده از این هسته باعث می‌شود تا روش نسبت به نوسانات بزرگ عملکرد بهتری از خود نشان دهد و به همین دلیل در تصاویری که بر اثر عوامل مختلف دچار تغییرات ناگهانی شده‌اند به خوبی جواب می‌دهد. در شکل ۴ عملکرد این روش را روی تصاویر مشاهده می‌کنیم.



شکل ۴: عملکرد روش رگرسیون بردار پشتیبان با هسته موجک

۵ اعتبار سنجی و شاخص‌های مقایسه‌ای

برای اعتبار سنجی و بررسی دقت روش‌های معرفی شده در مشخص‌سازی خطوط OPL و IPL ، روش را روی ۱۵۰ نمونه از تصاویر شبکه چشم بیماران بیمارستان فارابی اعمال کردیم و نتیجه را با خطوط یافته شده در حالت دستی توسط متخصص شبکه مقایسه نمودیم. خوشبختانه نتایج هر سه رویکرد در بیش از ۹۰ درصد موارد با نظر متخصص همخوانی داشت و حتی این نسبت در روش موجک، برای هر دو خط OPL و IPL ، به همخوانی بالای ۹۷ درصد منتج گردید.



برای تحلیل بهتر این خطوط و بررسی دقیق‌تر تغییرات آنها در اثر بیماریهای مختلف، با استفاده از ویژگی‌هایی مانند طول منحنی و تغییر تقعر، از تعدادی شاخص مقایسه‌ای استفاده می‌کنیم [۶]. یکی از شاخص‌های کاربردی نسبت طول منحنی به طول خط مستقیم واصل بین ابتدا و انتها است. علاوه بر آن، به دلیل اهمیتی که نوسانات این خطوط دارند، مهمترین شاخصی که می‌تواند برای تحلیل تغییرات مفید باشد شاخص بینظمی است که به صورت ذیل تعریف می‌شود.

$$SI = \frac{\text{تعداد تغییر تقعر منحنی از ابتدا تا انتهای تعیین شده}}{\text{نسبت طول منحنی به طول خط مستقیم واصل بین ابتدا و انتهای منحنی}}$$

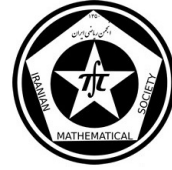
با محاسبه‌ی این شاخص‌ها و بررسی آنها در بیماریهای مختلف می‌توانیم به ارتباط بین تغییرات این خطوط و علائم بالینی بیمار پی ببریم. این مهم تا پیش از این به دلیل زمانبر بودن استخراج دستی خطوط و نبودن ابزار مناسب برای محاسبه ویژگی‌های گرافیکی امکان‌پذیر نبود.

- [1] A. Buades, B. Coll, & J.M. Morel, *A non-local algorithm for image denoising*. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) Vol. 2, pp. 60-65.
- [2] C. Cortes, & V. Vapnik, (1995). *Support-vector networks*. Machine learning, **20** (3) (1995), 273-297.
- [3] S. Han, C. Qubo, & H. Meng, *Parameter selection in SVM with RBF kernel function*. In World Automation Congress 2012 (pp. 1-4). IEEE.
- [4] J. C. Russ, *Image processing*, In Computer-assisted microscopy (pp. 33-69). Springer, Boston, MA, 1990.
- [5] L. Zhang, W. Zhou, & L. Jiao, *Wavelet support vector machine*. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), **34** (1) (2004), 34-39.
- [6] M. Zarei, E. KhaliliPour, N. Ebrahimiadib, & H. Riazi-Esfahani, *Quantitative Analysis of the Iris Surface Smoothness by Anterior Segment Optical Coherence Tomography in Fuchs Uveitis*. Ocular Immunology and Inflammation. (2020), 1-6.

پست الکترونیکی: j.saeidian@khu.ac.ir

پست الکترونیکی: ekhalilipour@gmail.com

پست الکترونیکی: hosseinazimi0213120@gmail.com



همگرایی سراسری روش گرادیان مزدوج PRP برای توابع غیرمحدب

سیدعلیرضا حسینی ده‌میری[†] و مریم کارگرفرد[‡]

دانشگاه ولیعصر (عج) رفسنجان - بخش ریاضی

چکیده

اکثر مسائل مهندسی امروزه در رده‌بندی مسائل بهینه‌سازی نامقید قرار می‌گیرند. مسائلی که هدف آنها یافتن نقاط بهینه‌ی یک تابع غیرخطی و معمولاً غیرمحدب است. الگوریتم‌های متعددی برای حل این گونه مسائل توسط محققین ارائه شده است که هر کدام با معایب و محاسن ویژه‌ای همراه هستند. در بین این روش‌ها، الگوریتم گرادیان مزدوج دارای محبوبیت بیشتری نسبت به بقیه می‌باشد. اگرچه عملکرد قابل قبول این الگوریتم، مخصوصاً در حل مسائل با ابعاد بالا، از مزایای بارز این الگوریتم می‌باشد، اما این روش نمی‌تواند همگرایی سراسری را برای توابع غیرمحدب تحت فرمول جستجوی خطی دقیق تضمین کند. این مقاله یک فرمول جستجوی خطی نادقیق جدید را برای روش گرادیان مزدوج پیشنهاد می‌دهد و همگرایی سراسری آن را برای توابع غیرمحدب اثبات می‌کند. واژه‌های کلیدی: بهینه‌سازی نامقید، روش شبه‌نیوتن، روش گرادیان مزدوج، همگرایی سراسری. طبقه‌بندی موضوعی [۲۰۱۰]: 90C26, 90C30, 90C06.

۱ پیش‌گفتار

مسأله‌ی بهینه‌سازی غیرخطی و نامقید زیر را در نظر بگیرید:

$$\min_{x \in \mathbb{R}^n} f(x), \quad (1.1)$$

که $f: \mathbb{R}^n \rightarrow \mathbb{R}$ دو بار به طور پیوسته مشتق‌پذیر است. همچنین فرض کنید مجموعه سطوح تراز $D = \{x \mid f(x) \leq f(x_0)\}$ به ازای هر x_0 کران‌دار باشد و علاوه بر آن تابع گرادیان f در شرط پیوستگی لیپ‌شیتز* صدق کند، این بدان معناست که ثابت $L > 0$ وجود

*Lipschitz continuity



داشته باشد به طوری که:

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y \in D. \quad (1.2)$$

برای حل مساله فوق می توان از روش های تکراری بر پایه ی گرادیان استفاده کرد. این روش ها از فرمول تکراری

$$x_{k+1} = x_k + \alpha_k d_k,$$

برای حل مساله (۱.۱) استفاده می کنند که در آن α_k یک طول گام و d_k یک جهت نزولی است. فرض کنید مقدار $g_k = \nabla f(x_k)$ و ماتریس B_k یک تقریب از ماتریس هسی تابع f در نقطه ی x_k باشد. در ادامه برای سادگی نوشتار از f_k و g_k به جای $f(x_k)$ و $g(x_k)$ استفاده شده است. به کمک نمادهای فوق معادله ی خط قاطع استاندارد را می توان به صورت زیر نوشت:

$$y_k = g_{k+1} - g_k = B_{k+1} s_k. \quad (1.3)$$

که در آن $s_k = x_{k+1} - x_k$. یک خانواده مشهور از روش های بر پایه ی گرادیان، خانواده ی روش های برویدن[†] است [۱، ۲]، که به صورت زیر تعریف می شوند:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k} + \mu w_k w_k^T, \quad (1.4)$$

$$w_k = (s_k^T B_k s_k)^{1/2} \left[\frac{y_k}{s_k^T y_k} - \frac{B_k s_k}{s_k^T B_k s_k} \right],$$

که در آن μ یک پارامتر سنجش می باشد. روش های $BFGS$ ، DFP و SR به ترتیب با جای گذاری $\mu = 0$ ، $\mu = 1$ و $\mu = 1 / \left(1 - \frac{s_k^T B_k s_k}{s_k^T y_k} \right)$ به دست می آیند.

بخش اصلی الگوریتم های مورد استفاده در بهینه سازی نامقید محاسبه ی جهت بهبود دهنده است، که با استفاده از مشتق یا مشتقات جزئی صورت می پذیرد. اخیراً ده میری یک فرمول جستجوی خطی نادقیق جدید برای یافتن طول گام α_k در روش $BFGS$ ارائه داده است [۳]، که سریع تر از فرمول جستجوی خطی یوآن-وی-لو[‡] می باشد [۵]. این جستجوی خطی نادقیق از فرمول های زیر برای پیدا کردن α_k استفاده می کند:

$$f_{k+1} \leq f_k + \delta \alpha_k g_k^T d_k - \frac{\delta \alpha_k^2}{2M} \|d_k\|^2, \quad (1.5)$$

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k - \frac{\delta \alpha_k}{M} \|d_k\|^2,$$

[†]Broyden

[‡]Yuan-Wei-Lu



که در آن $(\frac{1}{p}, \frac{1}{q})$ ، $\delta \in (0, 1)$ و M یک عدد مثبت خیلی بزرگ است. این فرمول جستجوی خطی را می‌توان در سایر الگوریتم‌های حل مسائل بهینه‌سازی نامقید، مانند روش گرادیان مزدوج (CG)[§] نیز به کار گرفت. در این مقاله سعی شده است که با کمک فرمول جستجوی خطی (۱۵)، بتوان همگرایی سراسری دسته‌ی خاصی از روش‌های گرادیان مزدوج، به نام روش PRP^* را، برای توابع غیرمحدب ثابت کرد. اگرچه الگوریتم گرادیان مزدوج PRP ، یکی از کاراترین الگوریتم‌ها در برخورد با مسائل بهینه‌سازی محدب با ابعاد بالا می‌باشد، اما در حل مسائل غیرمحدب همگرایی تضمین شده‌ای ندارد.

۲ اصلاح الگوریتم PRP برای رسیدن به همگرایی سراسری

همانگونه که در بخش قبل نیز اشاره شد، الگوریتم PRP برای توابع غیرمحدب، همگرایی تضمین شده‌ای ندارد. برای اطمینان از همگرایی این روش نیاز است که جهت انتخابی و طول گام حرکت به گونه‌ای انتخاب شود که بتواند کاهش اکید تابع هدف را در هر تکرار به دنبال داشته باشد. فرمول سه جمله‌ای مزدوج گرادیان به صورت زیر بیان می‌شود [۴]:

$$d_{k+1} = \begin{cases} -g_{k+1} + \beta_k^{PRP} d_k - \frac{d_k^T g_{k+1} y_k}{\|g_k\|^2}, & k \geq 1, \\ -g_{k+1}, & k = 0, \end{cases} \quad (21)$$

که در آن $\beta_k^{PRP} = \frac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2}$. در ادامه الگوریتم سه جمله‌ای اصلاح شده‌ی PRP برای حل مسأله‌ی نامقید (۱۱) را به طور کامل آورده، سپس به اثبات بعضی از ویژگی‌های این الگوریتم می‌پردازیم. این نتایج ما را به خوش تعریفی الگوریتم و همگرایی سراسری آن سوق خواهد داد.

[§]conjugate gradient

^{*}Polak-Ribière-Polyak



الگوریتم ۱

ورودی‌ها

۱. یک نقطه‌ی اولیه مانند $x_1 \in \mathbb{R}^n$ و ثابت‌های $\delta \in (0, \frac{1}{4})$, $\sigma \in (0, 1)$, $\varepsilon \in (0, 1)$ را انتخاب کنید. همچنین یک عدد دلخواه خیلی بزرگ مانند M را در نظر بگیرید.

۲. قرار دهید $k := 1$.

حلقه اصلی

مرحله ۱. اگر $\|g_k\| \leq \varepsilon$ متوقف شوید.

مرحله ۲. برای به دست آوردن جهت بهبود دهنده‌ی d_k ، معادله‌ی (۲۰۱) را حل کنید.

مرحله ۳. α_k را طوری پیدا کنید که در معادلات (۱۰۵) صدق کند و

$$\alpha_k < -\frac{M(1-\sigma)g_k^T d_k}{\delta \|d_k\|^2}. \quad (202)$$

مرحله ۴. قرار دهید $x_{k+1} := x_k + \alpha_k d_k$.

مرحله ۵. قرار دهید $k := k + 1$ و به مرحله ۱ بروید.

خروجی

x_k یک نقطه‌ی بحرانی مسأله‌ی (۱۰۱) است.



۳ نتایج اصلی

قضیه ۱. ([۳]) فرض کنید $f(x) \in C^2$ از پایین کران دار باشد. اگر $g_k^T d_k \leq 0$ آنگاه یک ثابت $0 < \alpha < \infty$ وجود دارد که در روابط (۱۵) و (۲۲) صدق می‌کند.

قضیه ۲. برای هر دو جمله‌ی متوالی از دنباله‌ی $\{g_k\}$ ، رابطه‌ی $g_{k+1}^T d_k > g_k^T d_k$ برقرار است.

اثبات. با توجه به رابطه‌های (۱۵) می‌توان نوشت:

$$\begin{aligned}(g_{k+1} - g_k)^T d_k &= g_{k+1}^T d_k - g_k^T d_k \\ &\geq \sigma g_k^T d_k - \frac{\delta \alpha_k}{M} \|d_k\|^2 - g_k^T d_k \\ &= -(\sigma - 1) g_k^T d_k - \frac{\delta \alpha_k}{M} \|d_k\|^2 > 0.\end{aligned}$$

□

که در آن، آخرین نامساوی از رابطه‌ی (۲۲) نتیجه شده است.

قضیه ۳. برای هر جمله از دنباله‌ی $\{g_k\}$ ، رابطه‌ی $g_{k+1}^T d_{k+1} = \|g_{k+1}\|^2$ برقرار است.

اثبات. با توجه به رابطه‌ی (۲۱) می‌توان نوشت:

$$\begin{aligned}g_{k+1}^T d_{k+1} &= -g_{k+1}^T g_{k+1} + g_{k+1}^T \left(\frac{g_{k+1}^T y_k}{\|g_k\|^2} \right) d_k - \frac{d_k^T g_{k+1} g_{k+1}^T y_k}{\|g_k\|^2} \\ &= -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^T y_k}{\|g_k\|^2} \right) g_{k+1}^T d_k - \frac{d_k^T g_{k+1} g_{k+1}^T y_k}{\|g_k\|^2} \\ &= -\|g_{k+1}\|^2.\end{aligned}$$

□



قضیه ۴. فرض کنید $\{g_k\}$ دنباله‌ی تولید شده توسط الگوریتم ۱ بوده و شرط $\|d_k\| \leq c\|g_k\|$ به ازای یک ثابت $c > 0$ برقرار باشد. در این صورت،

$$\lim_{k \rightarrow \infty} \|g_k\| = 0. \quad (3.1)$$

اثبات. با توجه به قضیه ۲، می‌توان نوشت:

$$g_{k+1}^T d_k - g_k^T d_k > 0$$

بنابراین $\varepsilon_0 \in \mathbb{R}^+$ وجود دارد که $g_{k+1}^T d_k - g_k^T d_k = \varepsilon_0 \|g_k\|$. حال به کمک رابطه‌ی (۱۲) خواهیم داشت:

$$\varepsilon_0 \|g_k\| = g_{k+1}^T d_k - g_k^T d_k \leq \|g_{k+1} - g_k\| \|d_k\| \leq \alpha_k L \|d_k\|^2.$$

بنابراین خواهیم داشت:

$$\alpha_k \geq \frac{\varepsilon_0 \|g_k\|}{L \|d_k\|^2} \geq \frac{\varepsilon_0}{L c^2 \|g_k\|}.$$

$$\begin{aligned} f_k - f_{k+1} &\geq -\delta \alpha_k g_k^T d_k + \frac{\delta \alpha_k^2}{2M} \|d_k\|^2 \\ &\geq -\delta \alpha_k g_k^T d_k \\ &= \delta \alpha_k \|g_k\|^2 \\ &\geq \frac{\delta \varepsilon_0}{L c^2} \|g_k\|. \end{aligned}$$

با جمع این روابط خواهیم داشت:

$$\sum_{k=0}^{\infty} \|g_k\| \leq \frac{L c^2}{\delta \varepsilon_0} (f_0 - f_{\infty}).$$

□

حال از آنجا که تابع $f(x)$ کراندار است، می‌توان نوشت: $\|g_k\| \rightarrow 0$ و اثبات تمام می‌شود.

۴ نتایج عددی

در این بخش الگوریتم پیشنهادی به کمک یک مثال عددی تشریح می‌شود.

مثال ۵. مسأله زیر را در نظر بگیرید:

$$\min f(x, y) = (y - x^2)^2 + (x - 1)^2.$$



نتایج حل این مسأله با استفاده از الگوریتم پیشنهادی با پارامترهای

$$\varepsilon = 0.01, \quad \delta = \frac{1}{3}, \quad \sigma = \frac{2}{3}, \quad x_1 = (0, 0)^T, \quad M = 10000,$$

در جدول زیر آورده شده است.

جدول ۱: نتایج عددی مثال ۵.

k	x_k	$f(x_k)$	$g(x_k)$
۱	$(0, 0)^T$	۱	$(-2, 0)^T$
۲	$(25, 0)^T$	۳۹۲۸۷۵	$(628, -125)^T$
۳	$(0.5167, 0.4960)^T$	۰.۷۵۸۰	$(-57000, 45804)^T$
۴	$(0.8242, 0.6124)^T$	۰.۰۷۵۶	$(18541, -13381)^T$
۵	$(0.8612, 0.7207)^T$	۰.۰۲۳۶	$(0.4446, -0.4193)^T$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
۱۵	$(0.9963, 0.9532)^T$	۰.۰۱۵۵	$(15633, -0.7883)^T$
۱۶	$(0.9998, 0.9991)^T$	2.5404×10^{-4}	$(0.0195, -0.0100)^T$
۱۷	$(0.9999, 0.9998)^T$	1.0000×10^{-4}	$(-1.9960 \times 10^{-4}, -2.0000 \times 10^{-7})^T$

۵ نتیجه‌گیری

در این مقاله برای رسیدن به همگرایی سراسری در روش PRP ، از یک فرمول جستجوی خطی نادقیق استفاده شد و ثابت گردید که استفاده از آن در الگوریتم PRP همگرایی سراسری را برای همه‌ی توابع اعم از محدب و غیرمحدب، به دنبال خواهد داشت. فرمول جستجوی نادقیق جدید نسبت به فرمول جستجوی خطی یوآن-وی-لو دارای سرعت اجرای بالاتری است. تعیین سرعت همگرایی این روش و همچنین اصلاح دوباره‌ی این فرمول برای رسیدن به سرعت همگرایی بالاتر، می‌تواند یک ایده مناسب برای پژوهش‌های آتی باشد.

مراجع

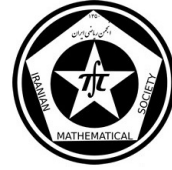
- [1] Charles G Broyden, *A class of methods for solving nonlinear simultaneous equations*, Mathematics of computation 19 (1965), 577–593.



- [2] Charles George Broyden, *The convergence of a class of double-rank minimization algorithms 1.* general considerations, IMA Journal of Applied Mathematics 6 (1970), 76–90.
- [3] Alireza Hosseini Dehmiry, The global convergence of the BFGS method under a modified Yuan-Wei-Lu line search technique, Numerical Algorithms 84 (2019), 781–793.
- [4] JK Liu, YX Zhao, and XL Wu, Some three-term conjugate gradient methods with the new direction structure, Applied Numerical Mathematics 150 (2020), 433–443.
- [5] Gonglin Yuan, Zhou Sheng, Bopeng Wang, Wujie Hu, and Chunnian Li, The global convergence of a modified BFGS method for nonconvex functions, Journal of Computational and Applied Mathematics 327 (2018), 274–294.

پست الکترونیکی: dehmiry@vru.ac.ir

پست الکترونیکی: kargarfard.maryam@gmail.com



یک روش تکراری موثر بدون معکوس برای محاسبه تجزیه قطبی ماتریس

خسرو سایوند[†] و راضیه عرفانی فر[‡]

[†] ksayehvand@malayeru.ac.ir

[‡] raziye.erfanifar@stu.malayeru.ac.ir

چکیده

در این مقاله، یک روش تکراری بدون معکوس برای محاسبه تجزیه قطبی یک ماتریس ارائه شده است. در هر تکرار از این روش فقط به چهار ضرب ماتریس در ماتریس احتیاج است. همگرایی روش از مرتبه چهار است. در نهایت، مقایسه عددی بین چندین روش برای محاسبه تجزیه قطبی انجام شده است.

واژه‌های کلیدی: تجزیه قطبی، روش تکراری عامل قطبی، مرتبه همگرایی.

طبقه‌بندی موضوعی [۲۰۱۰]: 13D45, 39B42.

۱ پیش‌گفتار

یکی از مهمترین کاربردی‌ترین تجزیه‌های ماتریسی، تجزیه قطبی ماتریس می‌باشد که از دیرباز راه‌های مختلف عددی برای آن ارائه شده است [۱، ۲، ۳، ۴].

فرض کنید A یک ماتریس دلخواه $m \times n$ باشد. تجزیه قطبی A

$$A = UH,$$

که $H \in \mathbb{C}^{n \times n}$ ماتریس نیمه معین مثبت هرمیتی است و $U \in \mathbb{C}^{m \times n}$ در یکی از شرایط زیر صدق می‌کند:

$$U^H U = I_n, \quad m \geq n,$$

$$U U^H = I_m, \quad m \leq n.$$



در اینجا I_k ماتریس همانی با مرتبه k را نشان می‌دهد. ماتریس H یک عامل هرمیتی نامیده می‌شود و U یک عامل واحد از تجزیه قطبی A است. اگر A حقیقی باشد، U و H نیز حقیقی است. بنابراین، H یک عامل متقارن و U یک عامل متعامد است. تجزیه قطبی برای مورد $m \geq n$ شناخته شده است. برای $m < n$ ، می‌توان تعریف زیر را نیز در نظر گرفت:

$$A = HU,$$

که در آن H نیمه معین مثبت و U دارای سطرهای متعامد است. در این مقاله، ما روی ماتریس‌های حقیقی متمرکز می‌شویم. بر طبق [۵] ثابت شده است که ماتریس H ریشه ماتریس نیمه معین مثبت $A^T A$ است،

$$H = (A^T A)^{\frac{1}{2}}.$$

این بدان معنی است که ضرب متقارن H همیشه منحصر به فرد است. از طرف دیگر، فاکتور متعامد U منحصر به فرد است اگر و فقط اگر ماتریس A دارای رتبه کامل باشد. توجه داشته باشید که در حقیقت، فقط به روش‌هایی برای یافتن ضرب متعامد U نیاز داریم زیرا ضرب متقارن H را می‌توان از فرمول زیر بدست آورد

$$H = \frac{1}{2}(U^T A + A^T U).$$

برای این منظور، نیازی به استفاده از فرمول $H = U^T A$ نیست زیرا به دلیل خطاهای گرد کردن، ماتریس محاسبه شده H نمی‌تواند متقارن باشد. برای رسیدن به مرحله همگرایی، یک ماتریس اولیه X_0 باید در روش نوعی ماتریس ثابت استفاده شود. همچنین تقریب اولیه را برای عامل واحد می‌توان به صورت زیر بیان کرد

$$X_0 = \frac{1}{\alpha} A, \quad \alpha = \frac{1}{\sqrt{\|A\|_1 \|A\|_\infty + 1}}. \quad (11)$$

اکنون برخی از مهمترین روش‌های تکراری برای محاسبه تجزیه قطبی ماتریس را به طور خلاصه مرور می‌کنیم. فرض کنید A یک ماتریس نامنفرد باشد، سپس روش کوواریک به شرح زیر است [۶]:

$$\begin{aligned} V_k &= X_k^T X_k \\ K_k &= (I - V_k)(I + V_k)^{-1}, \\ X_{k+1} &= X_k(I + K_k), \quad k \geq 0. \end{aligned} \quad (12)$$

روش همگرا مرتبه دو روش نیوتن شولتز:

$$\begin{aligned} V_k &= X_k^T X_k \\ S_k &= I - V_k, \\ X_{k+1} &= X_k(I + 0.5 S_k), \quad k \geq 0. \end{aligned} \quad (13)$$

هیقم [۷] با الهام از روش نیوتن برای محاسبه یک ریشه از یک عدد مثبت، روش

$$X_{k+1} = \frac{1}{2}(X_k + X_k^{-T}), \quad k \geq 0, \quad (104)$$

را برای محاسبه ضریب متعامد U در نظر گرفته است، که با مرتبه دوم به U همگراست. این اصلاح ساده، پیشنهاد شده توسط جندر [۸]،

$$\begin{aligned} V &= X_k^T X_k \\ X_{k+1} &= \frac{1}{2} X_k (I + V_k^{-1}), \quad k \geq 0. \end{aligned} \quad (105)$$

می باشد.

به جای روش نیوتن، می توانیم از روش هالی برای محاسبه ریشه یک عدد مثبت استفاده کنیم. در این حالت، برای محاسبه ضریب متعامد U ، روش زیر با همگرایی مکعبی بدست می آید:

$$\begin{aligned} V_k &= X_k^T X_k \\ X_{k+1} &= X_k (V_k + 3I)(3V_k + I)^{-1}, \quad k \geq 0. \end{aligned} \quad (106)$$

۲ یک کلاس روش های تکراری برای محاسبه فاکتور متعامد

با کمی توجه به روش های فوق می بینیم که اغلب آنها بصورت

$$X_{k+1} = X_k h(X_k^T X_k), \quad k \geq 0, \quad (201)$$

می باشند که در آن $h(t)$ تابعی حقیقی مناسب است. با استفاده از $L_k = X_k^T X_k$ ، $X_k = (X_k^T)^{-1} L_k$ و رابطه (۲۰۱) داریم:

$$X_{k+1} = (X_k^T)^{-1} L_k h(L_k).$$

اگر $L_k h(L_k) \rightarrow I$ وقتی $k \rightarrow \infty$ ، سپس $X_k \rightarrow X^{-T}$. یک شرط لازم برای برقراری $L_k h(L_k) \rightarrow I$ این است که h یک نقطه ثابت از تابع حقیقی است

$$g(x) = xh(x^T),$$

به عنوان مثال، $g(1) = 1$ از طرف دیگر، اگر

$$g^{(i)} = 0, \quad i = 1, \dots, p-1,$$

پس طرح تکراری، همگرایی مرتبه p دارد.

در راستای این هدف، ما تابعی حقیقی با ساختار زیر معرفی می کنیم:

$$g(x) = ax + bx^2 + cx^5 + dx^9 + ex^9,$$



سپس

$$g(1) = 1, \Rightarrow a + b + c + d + e = 1,$$

$$g'(1) = 0, \Rightarrow a + 3b + 5c + 7d + 9e = 0,$$

$$g''(1) = 0, \Rightarrow 6b + 20c + 42d + 72e = 0,$$

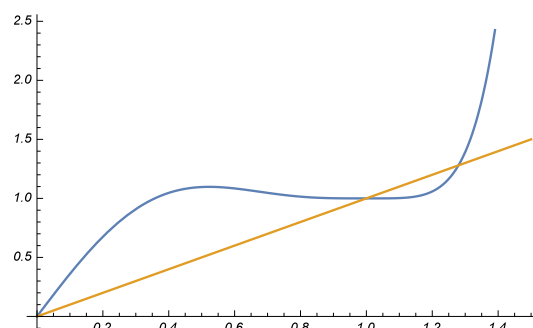
$$g'''(1) = 0, \Rightarrow 6b + 60c + 210d + 504e = 0,$$

بنابراین

$$a = \frac{1}{16}(35 + 16e), \quad b = -\frac{1}{16}(35 + 64e), \quad c = \frac{3}{16}(7 + 32e), \quad d = -\frac{1}{16}(5 + 64e),$$

در نهایت

$$g(x) = \frac{1}{16}(35 + 16e)x - \frac{1}{16}(35 + 64e)x^2 + \frac{3}{16}(7 + 32e)x^3 - \frac{1}{16}(5 + 64e)x^4 + ex^5, \quad (2.2)$$



شکل ۱: نمودار $y = x$ و تابع $g(x)$ برای $e = \frac{2}{3}$.

$$g(x) = x \Rightarrow x = 0, 1, 1 + \gamma,$$

$$g'(x) = 0 \Rightarrow x = 1, \frac{\sqrt{35 + 16e}}{12\sqrt{e}}.$$

که

$$\gamma = \left(\left(1 + \frac{5}{48e} - \frac{6}{(125 - 2160e + 27648e^2 + 48\sqrt{3}\sqrt{775}e^2 - 13824e^3 + 110592e^4)^{1/2}} \right. \right. \\ \left. \left. + \frac{25}{48e(125 - 2160e + 27648e^2 + 48\sqrt{3}\sqrt{775}e^2 - 13824e^3 + 110592e^4)^{1/2}} \right. \right. \\ \left. \left. + \frac{(125 - 2160e + 27648e^2 + 48\sqrt{3}\sqrt{775}e^2 - 13824e^3 + 110592e^4)^{1/2}}{48e} \right)^{1/2}$$



در ادامه، ما نشان خواهیم داد که $[0, 1 + \gamma]$ بازه مطلوب است. توجه داشته باشید که $x = 0$ و $1 + \gamma$ به $\bar{x} = \frac{\sqrt{35} + 16e}{12\sqrt{e}}$ ترتیب مینیمم کننده و ماکسیمم کننده $g(x)$ در بازه $[0, 1 + \gamma]$ هستند. حال می‌خواهیم بازه $[0, 1 + \gamma]$ توسط تابع $g(x)$ به خودش نگاشته شود، برای این امر لازم است

$$g\left(\frac{\sqrt{35} + 16e}{12\sqrt{e}}\right) < g(1 + \gamma) = 1 + \gamma.$$

در نتیجه بدست می‌آید $0.2 \leq e \leq 2$. بنابراین برای هر $e \in [0.2, 2]$ داریم

$$0 = g(0) \leq g(x) \leq g(1 + \gamma) = 1 + \gamma.$$

شکل ۱ نمودارهای خط $y = x$ و تابع (2.2) را برای $e = \frac{2}{3}$ نشان می‌دهد. با انتخاب $e = \frac{2}{3}$ داریم:

$$g(x) = \frac{59}{16}x - \frac{131}{16}x^2 + \frac{165}{16}x^3 - \frac{101}{16}x^4 + \frac{3}{4}x^5,$$

فرض کنید ماتریس غیرمنفرد $A \in \mathbb{C}^{n \times n}$ داده شود. با استفاده از تابع تکرار (2.2) ، روش تکراری مرتب شده برای یافتن X^{-T} را بدست می‌آوریم:

$$\begin{aligned} X_{k+1} = X_k \left\{ \frac{59}{16}I - \frac{131}{16}(X_k^T X_k) + \frac{165}{16}(X_k^T X_k)^2 \right. \\ \left. - \frac{101}{16}(X_k^T X_k)^3 + \frac{3}{4}(X_k^T X_k)^4 \right\}, \end{aligned} \quad (2.3)$$

در قضیه زیر، ثابت می‌کنیم که روش (2.3) همگرایی مرتبه چهار دارد.

قضیه ۱. فرض کنید $A \in \mathbb{C}^{n \times n}$ یک ماتریس دلخواه است. سپس، ماتریس تکرار $\{X_k\}_{k=0}^{\infty}$ (2.3) با استفاده از $X_0 = \frac{1}{\alpha}A$ به U همگرا می‌شود.

اثبات. فرض کنید $E_k = I - X_k$ سپس

$$\begin{aligned} E_{k+1} &= I - X_{k+1} \\ &= I - X_k \left\{ \frac{59}{16}I - \frac{131}{16}(X_k^T X_k) - \frac{165}{16}(X_k^T X_k)^2 \right. \\ &\quad \left. + \frac{101}{16}(X_k^T X_k)^3 - \frac{3}{4}(X_k^T X_k)^4 \right\} \\ &= (I - X_k)^4 \left(I - \frac{53}{16}(X_k^T X_k) - \frac{19}{4}(X_k^T X_k)^2 \right), \end{aligned}$$

□

در اینجا برهان تمام است.

روش (2.3) را بصورت زیر نیز می‌توان نوشت:

$$\begin{aligned} V_k &= X_k^T X_k, \quad W_k = V_k^T \\ X_{k+1} &= X_k \left(\frac{59}{16}I - \frac{131}{16}V_k + \frac{165}{16}W_k + W_k \left(\frac{3}{4}W - \frac{101}{16}V_k \right) \right), \quad k \geq 0. \end{aligned} \quad (2.4)$$



سوال ۲. ماتریس‌های مربعی زیر را با ابعاد $n = 10, 20$ و شرط توقف

$$\|X_{k+1} - X_k\|_1 < 10^{-10} \|X_{k+1}\|_1,$$

مورد آزمایش قرار می‌دهیم:

ماتریس A : ماتریس هیلبرت با درایه‌های $a_{ij} = \frac{1}{i+j-1}$.

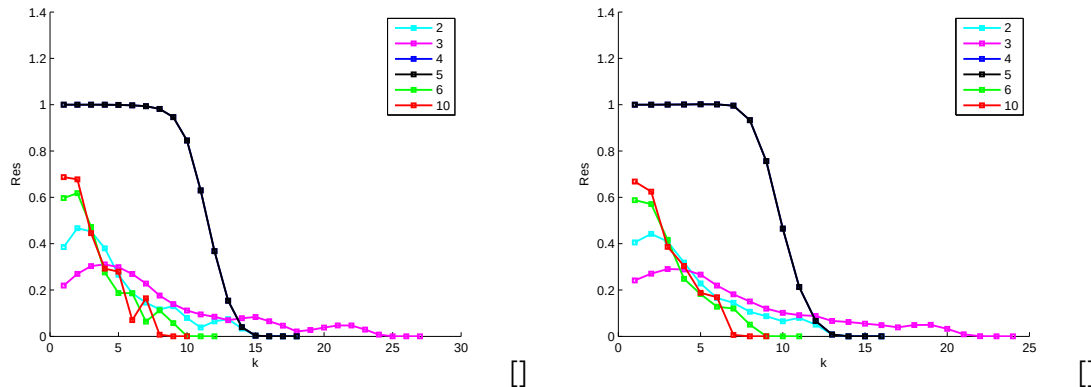
ماتریس B : ماتریس پاسکال با درایه‌های $b_{ij} = b_{i-1,j} + b_{i,j-1}$ و $b_{i1} = b_{1j} = 1$.

ماتریس C : ماتریس لاتکین با درایه‌های $c_{ij} = \frac{1}{i+j-1}$ و $c_{1j} = 1$.

جدول ۱: عنوان جدول

ماتریس	اندازه ماتریس	۱۲	۱۳	۱۴	۱۵	۱۶	۲۴
A		۴۹	۸۱	۴۹	-	۳۱	۲۶
B	۱۰	۳۷	۶۰	۳۷	-	۲۳	۲۰
C		۴۹	۸۲	۴۹	-	۳۱	۲۷
A		۶۳	۱۱۰	-	-	۴۲	۳۶
B	۲۰	۷۵	۱۲۳	-	-	۴۹	۴۱
C		۷۳	۱۱۰	-	-	۴۵	۳۷

جدول ارائه شده نتایج تعداد تکرار روش‌های ارائه شده در مقاله را نشان می‌دهد. علامت - نشان می‌دهد که روش ارائه شده برای این ماتریس واگرا است. در ادامه، دو ماتریس تصادفی با ابعاد بالا را مورد آزمایش قرار دادیم و شکل ۲ باقیمانده تکرارها را نمایش می‌دهد. با ملاحظه در جدول و شکل ارائه شده می‌بینیم که روش ارائه شده دارای کارایی خوبی می‌باشد.



شکل ۲: (آ) و (ب) به ترتیب نمایش باقیمانده دو ماتریس تصادفی با ابعاد $n = 50, 100$.

مراجع

- [1] E. S. Gawlik, M. Leok, *Iterative computing of the Frechet derivative of the polar decomposition*, SIAM J. Matrix Anal. Appl. 38, (2017) 1354-1379.
- [2] A. Cordero, J. R. Torregrosa, *A sixth-order iterative method for approximating the polar decomposition of an arbitrary matrix*, Comput. Appl. Math. 318, (2017) 591-598.
- [3] A. Kielbasinski, P. Zielinski, K. Zietak, *On iterative algorithms for the polar decomposition of a matrix*, Appl. Math. Comput. 270, (2015) 483-495.
- [4] R. Kumar, C. Quan, *Asymmetric multi-user optical cryptosystem based on polar decomposition and Shearlet transform*, Opt. Laser Eng. 120, (2019) 118-126.
- [5] R. A. Horn, C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1985.
- [6] Z. Kovarik, *Some iterative methods for improving orthogonality*, SIAM J. Numer. Anal. 7 (1970) 386-389.
- [7] N. J. Higham, *Computing the polar decomposition – with applications*, SIAM J. Sci. Stat. Comput. 7 (1986) 1160-1173.
- [8] W. Gander, *Algorithms for polar decomposition*, SIAM J. Sci. Stat. Comput. 11 (1990) 1102-1115.