



*Iranian Journal of
Numerical Analysis and Optimization*

Volume 10 , Number 2

Summer 2020

Serial Number: 18

Ferdowsi University of Mashhad, Iran

In the Name of God

Iranian Journal of Numerical Analysis and Optimization (IJNAO)

This journal is authorized under the registration No. 174/853 dated 1386/2/26, by the Ministry of Culture and Islamic Guidance.

Volume 10, Number 2, Summer 2020

ISSN-Print: 2423-6977, **ISSN-Online:** 2423-6969

Publisher: Faculty of Mathematical Sciences, Ferdowsi University of Mashhad

Published by: Ferdowsi University of Mashhad Press

Printing Method: Electronic

Address: Iranian Journal of Numerical Analysis and Optimization

Faculty of Mathematical Sciences, Ferdowsi University of Mashhad

P.O. Box 1159, Mashhad 91775, Iran.

Tel. : +98-51-38806222 , **Fax:** +98-51-38807358

E-mail: ijnao@um.ac.ir

Website: <http://ijnao.um.ac.ir>

This journal is indexed by:

- Scopus database of Elsevier (from 31/07/2020)
- Zentralblatt
- ISC
- SID
- Civilica
- Magiran
- DOAJ
- OAJI
- CiteFactor
- AcademicKeys
- The Journal granted the Scientific-Research degree by the Iranian Ministry of Science, Research, and Technology.

Iranian Journal of Numerical Analysis and Optimization

Volume 10, Number 2, Summer 2020

Ferdowsi University of Mashhad - Iran

©2020 All rights reserved. Iranian Journal of Numerical Analysis and Optimization

Iranian Journal of Numerical Analysis and Optimization

Director

M. H. Farahi

Editor-in-Chief

Ali R. Soheili

Managing Editor

M. Gachpazan

EDITORIAL BOARD

Abbasbandi, S.*

(Numerical Analysis)

Department of Applied Mathematics,

Faculty of Science,

Imam Khomeini International University,

Ghazvin.

e-mail: abbasbandy@ikiu.ac.ir

Area, I.*

(Numerical Analysis)

Departamento de Matemática Aplicada,

Universidade de Vigo.

e-mail: area@uvigo.es

Babolian, E.*

(Numerical Analysis)

Department of Mathematics,

Faculty of Mathematical Sciences

and Computer,

Kharazmi University, Karaj.

e-mail: babolian@khu.ac.ir

Effati, S.*

(Optimal Control & Optimization)

Department of Applied Mathematics,

Faculty of Mathematical Sciences,

Ferdowsi University of Mashhad, Mashhad.

e-mail: s-effati@um.ac.ir

Emrouznejad, A.*

(Operations Research)

Aston Business School,

Aston University, Birmingham, UK.

e-mail: a.emrouznejad@aston.ac.uk

Farahi, M. H.*

(Optimal Control & Optimization)

Department of Applied Mathematics,

Faculty of Mathematical Sciences,

Ferdowsi University of Mashhad, Mashhad.

e-mail: farahi@um.ac.ir

Gachpazan, M.**

(Numerical Analysis)

Department of Applied Mathematics,

Faculty of Mathematical Sciences,

Ferdowsi University of Mashhad, Mashhad.

e-mail: gachpazan@um.ac.ir

Ghanbari, R.**

(Operations Research)

Department of Applied Mathematics,
Faculty of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad.
e-mail: rghanbari@um.ac.ir

Hadizadeh Yazdi, M.**
(Numerical Analysis)
Department of Mathematics,
Faculty of Science
Khaje-Nassir-Toosi University of
Technology, Tehran
e-mail: hadizadeh@kntu.ac.ir

Hojjati, GH. R.*
(Numerical Analysis)
Department of Applied Mathematics,
Faculty of Mathematical Sciences
University of Tabriz, Tabriz
e-mail: ghojjati@tabrizu.ac.ir

Khojasteh Salkuyeh, D.*
(Numerical Analysis)
Department of Applied Mathematics,
Faculty of Mathematical Sciences,
University of Guilan, Rasht.
khojasteh@guilan.ac.ir

Lohmander, P.*
(Optimization)
Department of Forest Economics,
Swedish University of Agricultural Sciences,
Sweden
e-mail: Peter@Lohmander.com

Mahdavi-Amiri, N.*
(Optimization)
Department of Mathematical Sciences,
Faculty of Mathematics
Sharif University of Technology, Tehran.
e-mail: nezamm@sina.sharif.edu

Soheili, Ali R.*
(Numerical Analysis)
Department of Applied Mathematics,
Faculty of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad.
e-mail: soheili@um.ac.ir

Toutounian, F.*
(Numerical Analysis)
Department of Applied Mathematics,
Faculty of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad.
e-mail: toutouni@um.ac.ir

Kamyad, A.V.*
(Optimal Control & Optimization)
Department of Applied Mathematics,
Faculty of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad.
e-mail: vahidian@um.ac.ir

Vasagh, Z.
(English Text Editor)
Department of Applied Mathematics,
Faculty of Mathematical Sciences,
Ferdowsi University of Mashhad, Mashhad.

This journal is published under the auspices of Ferdowsi University of Mashhad

* Full Professor

** Associate Professor

We would like to acknowledge the help of Miss Narjes khatoon Zohorian in the preparation of this issue.

Letter from the Editor in Chief

I would like to welcome you to the Iranian Journal of Numerical Analysis and Optimization (IJNAO). This journal is published two issues per year and supported by the Faculty of Mathematical Sciences at the Ferdowsi University of Mashhad. Faculty of Mathematical Sciences with three centers of excellence and three research centers is well-known in mathematical communities in Iran.

The main aim of the journal is to facilitate discussions and collaborations between specialists in applied mathematics, especially in the fields of numerical analysis and optimization, in the region and worldwide.

Our vision is that scholars from different applied mathematical research disciplines, pool their insight, knowledge and efforts by communicating via this international journal.

In order to assure high quality of the journal, each article is reviewed by subject-qualified referees.

Our expectations for IJNAO are as high as any well-known applied mathematical journal in the world. We trust that by publishing quality research and creative work, the possibility of more collaborations between researchers would be provided. We invite all applied mathematicians especially in the fields of numerical analysis and optimization to join us by submitting their original work to the Iranian Journal of Numerical Analysis and Optimization.

Ali R. Soheili

Contents

2D-fractional Muntz–Legendre polynomials for solving the fractional partial differential equations	1
E. Hengamian Asl, J. Saberi-Nadjafi and M. Gachpazan	
Solving biobjective network flow problem associated with minimum cost-time loading	33
O. Baghani and S. Ghafoori	
A noniterative domain decomposition method for the forward-backward heat equation	49
S. Banei and K. Shanazari	
Comparison of homotopy perturbation transform method and fractional Adams–Bashforth method for the Caputo–Prabhakar nonlinear fractional differential equations	63
M.H. Derakhshan and A. Aminataei	
A four directions variational method for solving image processing problems	87
A.R. Hosseini and E.E. Esfahani	
Maximum probability O-D matrix estimation in large-sized networks	105
M. Abareschi	
Numerical methods for solving nonlinear Volterra integro-differential	131
S. Fazeli	
The most probable allocation solution for the p-median problem	155
M. Zaferanieh	

Numerical solution of the Allen-Cahn equation by using "shifted" surface spline radial basis functions	177
C.G Keshavarzi and F. Ghoreishi	
On the numerical solution of optimal control problems via Bell polynomials basis	197
M. R. Dadashi, A. R. Haghighi, F. Soltanian and A. Yari	
An approximate method based on Bernstein polynomials for solving fractional PDEs with proportional delays	223
A. Ketabdari, M.H. Farahi and S. Effati	
Nonlinear vibrations of functionally graded cylindrical shell by using numerical analysis in the wave propagation method	241
M.R. Assari, A. Kavooosi Nejad and Sh. Amirshirzad	



2D-fractional Muntz–Legendre polynomials for solving the fractional partial differential equations

E. Hengamian Asl, J. Saberi-Nadjafi* and M. Gachpazan

Abstract

We present a numerical method for solving linear and nonlinear fractional partial differential equations (FPDEs) with variable coefficients. The main aim of the proposed method is to introduce an orthogonal basis of two-dimensional fractional Muntz–Legendre polynomials. By using these polynomials, we approximate the unknown functions. Furthermore, an operational matrix of fractional derivative in the Caputo sense is provided for computing the fractional derivatives. The proposed approximation together with the Tau method reduces the solution of the FPDEs to the solution of a system of algebraic equations. Finally, to show the validity and accuracy of the presented method, we give some numerical examples.

AMS(2010): 26A33, 65M60.

Keywords: Two-dimensional fractional Muntz–Legendre polynomials (2D-FMLPs); Fractional partial differential equations (FPDEs); Operational matrix; Caputo fractional derivative.

*Corresponding author

Received 1 October 2019; accepted 7 April 2020

Elias Hengamian Asl

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: hengamianasl61@gmail.com

Jafar Saberi-Nadjafi

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: najafi141@gmail.com

Mortaza Gachpazan

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: gachpazan@um.ac.ir

1 Introduction

It is well known that fractional ordinary differential equations and fractional partial differential equations are important parts of the numerical analysis, statistical mechanics, and physics in recent years (see [5, 11]).

Recently, several numerical techniques have been proposed by researchers for solving some classes of the fractional ordinary differential equations and the fractional partial differential equations (FPDEs). For example, Nagy [9] used the Sinc-Chebyshev collocation method for fractional nonlinear Klein-Gordon equations. Yin et al. [18] applied two-dimensional fractional-order Legendre functions for solving the numerical solution of the fractional partial differential equations. Chen, Sun, and Liu [3] used a generalized fractional-order Legendre function for solving FPDEs with variable coefficients. For more numerical methods, the interested reader may refer to [6, 4, 2, 8, 1, 13].

In the present paper, we apply two-dimensional fractional Muntz-Legendre polynomials (2D-FMLPs) to find a numerical solution of linear and nonlinear FPDEs with variable coefficients.

This article is organized as follows. Review of Caputo fractional derivative is briefly provided in Section 2, and also we review one-dimensional fractional Muntz-Legendre polynomials in this section. In Section 3, two-dimensional FMLPs are introduced. In Section 4, we introduce an operational matrix of fractional derivative in the Caputo sense. In Section 5, we approximate the solution of FPDEs, using the Tau method based on 2D-FMLPs. Section 6 presents some examples of FPDEs to show the efficiency and the accuracy of the proposed method. Finally, a conclusion is expressed in Section 7.

2 Preliminaries

In this section, we review definitions and preliminary qualities of the fractional calculus, which are used in this paper.

Definition 1. The Caputo fractional derivative of order α is defined as

$$D^\alpha u(t) = \begin{cases} \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{u^{(n)}(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau, & n-1 < \alpha < n, \quad n \in \mathbb{N}, \\ \frac{d^n u(t)}{dt^n}, & \alpha = n, \quad t > 0. \end{cases}$$

It can easily be shown that

$$D^\alpha C = 0,$$

when C is a constant and

$$D^\alpha t^v = \begin{cases} 0, & v \in \mathbb{N}_0 \text{ and } v < [\alpha], \\ \frac{\Gamma(v+1)}{\Gamma(v+1-\alpha)} t^{v-\alpha}, & v \in \mathbb{N}_0 \text{ and } v \geq [\alpha] \text{ or } v \notin \mathbb{N}_0 \text{ and } v > [\alpha], \end{cases} \quad (1)$$

where $[\alpha]$ is the integer part of the positive real number α and $\mathbb{N}_0 = \{0, 1, 2, \dots\}$.

Moreover, we have

$$D^\alpha \left(\sum_{i=1}^n a_i u_i(t) \right) = \sum_{i=1}^n a_i D^\alpha u_i(t),$$

where $a_i, i = 1, \dots, n$ are constants.

The fraction Muntz–Legendre polynomials $L_i(t; \alpha)$ on the interval $[0, 1]$ are given by the following formula [4]:

$$L_i(t; \alpha) = \sum_{k=0}^i C_{i,k} t^{k\alpha}, \quad C_{i,k} = \frac{(-1)^{i-k}}{\alpha^i k! (i-k)!} \prod_{v=0}^{i-1} ((k+v)\alpha + 1). \quad (2)$$

Remark 1. According to (2), the analytic form of $L_i(t; \alpha)$ can be written as follows:

$$L_i(t; \alpha) = \sum_{k=0}^i b_{k,i} t^{k\alpha}, \quad (3)$$

where

$$b_{k,i} = \frac{(-1)^{i-k} \Gamma(\frac{1}{\alpha} + k + i)}{k! (i-k)! \Gamma(\frac{1}{\alpha} + k)}. \quad (4)$$

Also, we have

$$L_i(t; \alpha) = P_i^{(0, \frac{1}{\alpha}-1)}(2t^\alpha - 1), \quad \alpha > 0, \quad (5)$$

where $P_i^{(\alpha, \beta)}$ are the Jacobi polynomial with parameters $\alpha, \beta > -1$; see [4, 16]. Using (5) and the recurrence relation between the Jacobi polynomials [16], we can obtain the following recurrence formula:

$$L_{i+1}(t; \alpha) = a_i^\alpha L_i(t; \alpha) - b_i^\alpha L_{i-1}(t; \alpha), \quad i = 1, 2, \dots,$$

where

$$\begin{aligned} a_i^\alpha &= \frac{(2i + \frac{1}{\alpha})[(2i + \frac{1}{\alpha} - 1)(2i + \frac{1}{\alpha} + 1)(2t^\alpha - 1) - (\frac{1}{\alpha} - 1)^2]}{2(i+1)(i + \frac{1}{\alpha})(2i + \frac{1}{\alpha} - 1)}, \\ b_i^\alpha &= \frac{i(i + \frac{1}{\alpha} - 1)(2i + \frac{1}{\alpha} + 1)}{(i+1)(i + \frac{1}{\alpha})(2i + \frac{1}{\alpha} - 1)}, \\ L_0(t; \alpha) &= 1, \quad L_1(t; \alpha) = (\frac{1}{\alpha} + 1)t^\alpha - \frac{1}{\alpha}. \end{aligned}$$

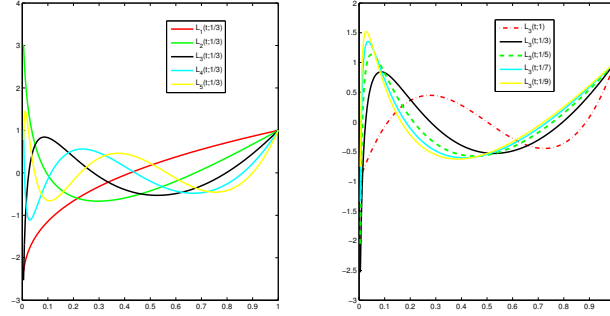


Figure 1: The graphs of $L_i(t; \alpha)$ for various values of i and α

Note that $L_i(1; \alpha) = 1$. Also, for $\alpha = 1$, we obtain shift Legendre polynomials [15] as follows:

$$L_i(t; 1) = P_i^{(0,0)}(2t - 1), \quad t \in [0, 1].$$

For $\alpha \neq 1$, we have

$$L_i(t; \alpha) = P_i^{(0, \frac{1}{\alpha}-1)}(2t^\alpha - 1) \neq P_i^{(0,0)}(2t^\alpha - 1),$$

where $P_i^{(0,0)}(2t^\alpha - 1)$ are fractional shift Legendre polynomials; see [6].

Remark 2. The FMLPs are orthogonal on the interval $[0, 1]$ with the orthogonality relation:

$$\int_0^1 L_i(t; \alpha) L_j(t; \alpha) dt = \frac{1}{2i\alpha + 1} \delta_{ij}, \quad (6)$$

where δ_{ij} is the Kronecker function.

The graphs of FMLPs are shown in Figure 1.

Definition 2. An arbitrary function $u(t)$ that is integrable in $[0, 1]$, can be expanded as follows:

$$u(t) = \sum_{i=0}^{\infty} a_i L_i(t; \alpha), \quad (7)$$

where

$$a_i = (2i\alpha + 1) \int_0^1 u(t) L_i(t; \alpha) dt, \quad i = 0, 1, \dots$$

In practice, only the first $(m + 1)$ -terms of FMLPs are considered. Then it can be written by the infinite series of (7) as

$$u(t) \simeq u_m(t) = \sum_{i=0}^m a_i L_i(t; \alpha) = A^T \phi(t; \alpha), \quad (8)$$

where

$$A = [a_0, a_1, \dots, a_m]^T, \\ \phi(t; \alpha) = [L_0(t; \alpha), L_1(t; \alpha), \dots, L_m(t; \alpha)]^T.$$

The following theorem shows that the approximation FMLPs converges to $u(t)$.

Theorem 1.[Error bound] Suppose $D^{k\alpha}u(t) \in C[0, 1]$ for $k = 0, 1, \dots, m$. If $u_m(t)$ in (8) is the best approximation to $u(t)$ from $M_{m,\alpha} = \text{span}\{L_0(t; \alpha), L_1(t; \alpha), \dots, L_m(t; \alpha)\}$, then

$$\|u(t) - u_m(t)\|_\omega \leq \frac{M_\alpha}{\Gamma(m\alpha + 1)\sqrt{2m\alpha + 1}},$$

where $M_\alpha \geq |D^{m\alpha}u(t)|$, $t \in [0, 1]$.

Proof. By applying generalized Taylor's formula (see [3, 10]), we have

$$u(t) = \sum_{k=0}^{m-1} \frac{t^{k\alpha}}{\Gamma(k\alpha + 1)} D^{k\alpha}u(0^+) + \frac{t^{m\alpha}}{\Gamma(m\alpha + 1)} D^{m\alpha}u(\xi), \quad 0 < \xi < t, \quad t \in [0, 1].$$

Also,

$$|u(t) - \sum_{k=0}^{m-1} \frac{t^{k\alpha}}{\Gamma(k\alpha + 1)} D^{k\alpha}u(0^+)| \leq \frac{M_\alpha t^{m\alpha}}{\Gamma(m\alpha + 1)}.$$

On the other hand, $u_m(t) = A^T \phi(t; \alpha)$ is the best approximation to $u(t)$, and $\sum_{k=0}^{m-1} \frac{t^{k\alpha}}{\Gamma(k\alpha + 1)} D^{k\alpha}u(0^+) \in M_{m,\alpha}$. So, it can be written as

$$\begin{aligned} \|u(t) - u_m(t)\|_\omega^2 &\leq \|u(t) - \sum_{k=0}^{m-1} \frac{t^{k\alpha}}{\Gamma(k\alpha + 1)} D^{k\alpha}u(0^+)\|_\omega^2 \\ &\leq \frac{M_\alpha^2}{\Gamma(m\alpha + 1)^2} \int_0^1 t^{2m\alpha} dt = \frac{M_\alpha^2}{\Gamma(m\alpha + 1)^2(2m\alpha + 1)}. \end{aligned}$$

Therefore,

$$\|u(t) - u_m(t)\|_\omega \leq \frac{M_\alpha}{\Gamma(m\alpha + 1)\sqrt{2m\alpha + 1}}.$$

Hence, the proof is complete. $\square$

3 Two-dimensional fractional Muntz–Legendre polynomials

In this section, we define the fractional Muntz–Legendre polynomials in the domain $\Omega = [0, 1] \times [0, 1]$. To this end, we can define these polynomials as follows.

Definition 3. Let $\{L_n(t; \alpha)\}_{n=0}^{\infty}$ be the one-dimensional fractional Muntz–Legendre polynomials on $[0, 1]$. We call $\{L_i(x; \alpha)L_j(t; \beta)\}_{i,j=0}^{\infty}$ the 2D-FMLPs (two-dimensional fractional Muntz–Legendre polynomials) on $\Omega = [0, 1] \times [0, 1]$.

Theorem 2. 2D-FMLPs are orthogonal on Ω .

Proof. For $i \neq p$ and $j \neq q$, we have

$$\begin{aligned} & \int_0^1 \int_0^1 L_i(x; \alpha) L_j(t; \beta) L_p(x; \alpha) L_q(t; \beta) dx dt \\ &= \left(\int_0^1 L_i(x; \alpha) L_p(x; \alpha) dx \right) \left(\int_0^1 L_j(t; \beta) L_q(t; \beta) dt \right) = 0, \end{aligned}$$

and for $i = p$ and $j = q$, we have

$$\begin{aligned} \int_0^1 \int_0^1 [L_i(x; \alpha)]^2 [L_j(t; \beta)]^2 dx dt &= \left(\int_0^1 [L_i(x; \alpha)]^2 dx \right) \left(\int_0^1 [L_j(t; \beta)]^2 dt \right) \\ &= \frac{1}{2i\alpha + 1} \cdot \frac{1}{2j\beta + 1}. \end{aligned}$$

□

Definition 4. An arbitrary function $u(x, t)$ that is integrable in $\Omega = [0, 1] \times [0, 1]$ can be expanded as follows:

$$u(x, t) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} L_i(x; \alpha) L_j(t; \beta), \quad (9)$$

where

$$a_{ij} = (2i\alpha + 1)(2j\beta + 1) \int_0^1 \int_0^1 u(x, t) L_i(x; \alpha) L_j(t; \beta) dx dt.$$

In practice, only the first $(m+1)(n+1)$ -terms of 2D-FMLPs are considered. Then it can be written by the infinite series of (9) as

$$u(x, t) \simeq u_{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n a_{ij} L_i(x; \alpha) L_j(t; \beta) = \phi^T(x; \alpha) A \phi(t; \beta), \quad (10)$$

where

$$\begin{aligned}\phi(x; \alpha) &= [L_0(x; \alpha), L_1(x; \alpha), \dots, L_m(x; \alpha)]^T, \\ \phi(t; \beta) &= [L_0(t; \beta), L_1(t; \beta), \dots, L_n(t; \beta)]^T,\end{aligned}$$

and A is an $(m+1) \times (n+1)$ matrix as follows:

$$A = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{0n} \\ a_{10} & a_{11} & \cdots & a_{1n} \\ \vdots & \vdots & & \vdots \\ a_{m0} & a_{m1} & \cdots & a_{mn} \end{bmatrix}.$$

To calculate the coefficients a_{ij} in (10), we use the Tau method, which is discussed in Section 6.

Remark 3. The product matrix of FMLPs can be obtained as follows:

$$\phi(x; \alpha) \phi^T(x; \alpha) A \simeq \tilde{A} \phi(x; \alpha),$$

where $\tilde{A}$ is an $(m+1) \times (m+1)$ product matrix for the matrix A . The elements of $\tilde{A}$ are as follows,

$$\tilde{A}_{i,j} = (2j\alpha + 1) \sum_{k=0}^m a_k g_{ijk}, \quad i, j = 0, \dots, m,$$

where

$$g_{ijk} = \int_0^1 L_i^\alpha(x) L_j^\alpha(x) L_k^\alpha(x) dx.$$

The following theorems state the existence of uniqueness and convergence analyses.

Theorem 3. If the double series

$$\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} L_i(x; \alpha) L_j(t; \beta)$$

converges uniformly to $u(x, t)$ on the $\Omega = [0, 1] \times [0, 1]$, then

$$a_{ij} = (2i\alpha + 1)(2j\beta + 1) \int_0^1 \int_0^1 u(x, t) L_i(x; \alpha) L_j(t; \beta) dx dt.$$

Proof. Set

$$u(x, t) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} L_i(x; \alpha) L_j(t; \beta),$$

and let m, n be fixed. We can write

$$\begin{aligned} & \int_0^1 \int_0^1 u(x, t) L_m(x; \alpha) L_n(t; \beta) dx dt \\ &= \int_0^1 \int_0^1 \left(\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} L_i(x; \alpha) L_j(t; \beta) \right) L_m(x; \alpha) L_n(t; \beta) dx dt \\ &= \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} \left(\int_0^1 L_i(x; \alpha) L_m(x; \alpha) dx \right) \left(\int_0^1 L_j(t; \beta) L_n(t; \beta) dt \right) \\ &= a_{ij} \left(\int_0^1 L_i^2(x; \alpha) dx \right) \left(\int_0^1 L_j^2(t; \beta) dt \right) \\ &= a_{ij} \left(\frac{1}{2i\alpha + 1} \right) \left(\frac{1}{2j\beta + 1} \right). \end{aligned}$$

□

Lemma 1. If $u(x, t)$ is a continuous function on $[0, 1] \times [0, 1]$ and the double series

$$\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{ij} L_i(x; \alpha) L_j(t; \beta) \quad (11)$$

converges uniformly to $u(x, t)$, then the series (11) is the FMLPs expansion of $u(x, t)$.

Proof. The proof of this lemma is similar to that of [8, Lemma 1]. □

Lemma 2. If two continuous functions defined on $[0, 1] \times [0, 1]$ have the identical FMLPs expansions, then these two functions are identical.

Proof. The lemma is proved similar to Lemma 2 in [8]. □

Theorem 4. If the 2D-FMLPs expansion of a continuous function $u(x, t)$ converges uniformly, then every 2D-FMLPs expansion converges to the function $u(x, t)$.

Proof. It is a result of Lemmas 1 and 2. □

4 FMLPs operational matrix of fractional derivatives

In this section, we introduce an operational matrix of fractional derivative. First, we present the following lemma.

Lemma 3. Let $L_i(t; \alpha)$ be an FMLP. Then the Caputo fractional derivative of $L_i(t; \alpha)$ of order $\gamma > 0$ can be obtained by the following relation:

$$D^\gamma L_i(t; \alpha) = \sum_{k=0}^i b'_{k,i} \frac{\Gamma(k\alpha + 1)}{\Gamma(k\alpha - \gamma + 1)} t^{k\alpha - \gamma},$$

where

$$b'_{k,i} = \begin{cases} 0, & k\alpha \in \mathbb{N}_0 \text{ and } k\alpha < \gamma, \\ b_{k,i}, & k\alpha \in \mathbb{N}_0 \text{ and } k\alpha \geq \gamma \text{ or } k\alpha \notin \mathbb{N}_0 \text{ and } k\alpha \geq [\gamma]. \end{cases}$$

and $b_{k,i}$ is given in (4).

Proof. By using relations (1) and (3), the proof of the lemma is clear. $\square$

Now, the next theorem gives the operational matrix of fractional derivative of FMLPs.

Theorem 5. If $\phi(x; \alpha)$ is a fractional Muntz–Legendre vector given in (10), then

$$D^\gamma \phi(x; \alpha) = D^{(\gamma)} \phi(x; \alpha),$$

where $D^{(\gamma)}$ is the $(m+1) \times (n+1)$ operational matrix of fractional derivative of order $\gamma > 0$. The elements of $D^{(\gamma)}$ are defined as follows:

$$D_{ij}^{(\gamma)} = (2j\alpha + 1) \sum_{k=0}^i \sum_{s=0}^j b_{s,j} b'_{k,i} \frac{\Gamma(k\alpha + 1)}{\Gamma(k\alpha - \gamma + 1)} \cdot \frac{1}{(k+s)\alpha - \gamma + 1},$$

where

$$b'_{k,i} = \begin{cases} 0, & k\alpha \in \mathbb{N}_0 \text{ and } k\alpha < \gamma, \\ b_{k,i}, & k\alpha \in \mathbb{N}_0 \text{ and } k\alpha \geq \gamma \text{ or } k\alpha \notin \mathbb{N}_0 \text{ and } k\alpha \geq [\gamma]. \end{cases}$$

Proof. According to Lemma 3, we have

$$D^\gamma L_i(t; \alpha) = \sum_{k=0}^i b'_{k,i} \frac{\Gamma(k\alpha + 1)}{\Gamma(k\alpha + 1 - \gamma)} t^{k\alpha - \gamma}. \quad (12)$$

On the other hand, the approximation of $t^{k\alpha - \gamma}$ by using (8) gives the following relation:

$$t^{k\alpha - \gamma} \simeq \sum_{j=0}^m a_{k,j} L_j(t; \alpha), \quad (13)$$

where

$$\begin{aligned}
a_{k,j} &= (2j\alpha + 1) \int_0^1 t^{k\alpha-\gamma} L_j(x; \alpha) dt \\
&= (2j\alpha + 1) \sum_{s=0}^j b_{s,j} \int_0^1 t^{k\alpha-\gamma+s\alpha} dt \\
&= (2j\alpha + 1) \sum_{s=0}^j b_{s,j} \frac{1}{k\alpha - \gamma + s\alpha + 1}.
\end{aligned} \tag{14}$$

Now, according to Equations (12)–(14), we can write

$$D^\gamma L_i(t; \alpha) = (2j\alpha + 1) \sum_{j=0}^m \sum_{k=0}^i \sum_{s=0}^j b_{s,j} b'_{k,i} \frac{\Gamma(k\alpha + 1)}{\Gamma(k\alpha + 1 - \gamma)} \cdot \frac{1}{(k\alpha - \gamma + s\alpha + 1)} L_j(t; \alpha).$$

Thus,

$$D_{ij}^\gamma = (2j\alpha + 1) \sum_{k=0}^i \sum_{s=0}^j b_{s,j} b'_{k,i} \frac{\Gamma(k\alpha + 1)}{\Gamma(k\alpha - \gamma + 1)} \cdot \frac{1}{(k + s)\alpha - \gamma + 1},$$

and this completes the proof. $\square$

5 Approximate solution to the FPDEs

In this section, we approximate the solution of the linear and nonlinear FPDEs by applying the operational matrix of fractional derivatives of 2D-FMLPs.

5.1 Numerical solution of linear FPDEs with variable coefficients

To study the linear FPDEs with variable coefficients, the following form is considered:

$$a(x) \frac{\partial^\alpha u(x, t)}{\partial x^\alpha} + b(t) \frac{\partial^\beta u(x, t)}{\partial t^\beta} + c(x) \frac{\partial u(x, t)}{\partial x} + d(x) u(x, t) = f(x, t), \tag{15}$$

$$u(0, t) = g_0(t), \quad u(x, 0) = h_0(x), \tag{16}$$

where $(x, t) \in \Omega = (0, 1] \times (0, 1]$, $0 < \alpha, \beta \leq 1$ and α, β are the order of the fractional derivatives in the Caputo sense. In (15) and (16), the continuous functions a , b , c , d , g_0 , h_0 , and f are known and the function $u(x, t)$ is unknown.

To solve problem (15)–(16), we approximate the known functions, the unknown function $u(x, t)$, and the differentials $\frac{\partial u(x, t)}{\partial x}$, $\frac{\partial^\alpha u(x, t)}{\partial x^\alpha}$, $\frac{\partial^\beta u(x, t)}{\partial t^\beta}$ by applying (10) and Theorem 5 as follows:

$$u(x, t) \simeq \phi^T(x; \alpha)U\phi(t; \beta), \quad f(x, t) \simeq \phi^T(x; \alpha)F\phi(t; \beta), \quad (17)$$

$$a(x) \simeq A^T\phi(x; \alpha), \quad b(t) \simeq \phi^T(t; \beta)B, \quad (18)$$

$$c(x) \simeq C^T\phi(x; \alpha), \quad d(x) \simeq E^T\phi(x; \alpha), \quad (19)$$

$$\frac{\partial u(x, t)}{\partial x} \simeq \phi^T(x; \alpha)(D^{(1)})^T U\phi(t; \beta), \quad (20)$$

$$\frac{\partial^\alpha u(x, t)}{\partial x^\alpha} \simeq \phi^T(x; \alpha)(D^{(\alpha)})^T U\phi(t; \beta), \quad (21)$$

$$\frac{\partial^\beta u(x, t)}{\partial t^\beta} \simeq \phi^T(x; \alpha)UD^{(\beta)}\phi(t; \beta). \quad (22)$$

Substituting (17)–(22) in (15), we have

$$\begin{aligned} & \phi^T(x; \alpha)\tilde{A}^T(D^\alpha)^T U\phi(t; \beta) + \phi^T(x; \alpha)UD^{(\beta)}\tilde{B}\phi(t; \beta) \\ & + \phi^T(x; \alpha)\tilde{C}^T(D^{(1)})^T U\phi(t; \beta) + \phi^T(x; \alpha)\tilde{E}^T U\phi(t; \beta) \simeq \phi^T(x; \alpha)F\phi(t; \beta) \end{aligned}$$

or

$$\phi^T(x; \alpha)[\tilde{A}^T(D^\alpha)^T U + UD^{(\beta)}\tilde{B} + \tilde{C}^T(D^{(1)})^T U + \tilde{E}^T U]\phi(t; \beta) \quad (23)$$

$$\simeq \phi^T(x; \alpha)F\phi(t; \beta). \quad (24)$$

In addition, for the initial and boundary conditions (16), we have

$$u(0, t) \simeq \phi^T(0; \alpha)U\phi(t; \beta) \simeq G_0^T\phi(t; \beta), \quad g_0(t) \simeq G_0^T\phi(t; \beta), \quad (25)$$

$$u(x, 0) \simeq \phi^T(x; \alpha)U\phi(0; \beta) \simeq \phi^T(x; \alpha)H_0, \quad h_0(x) \simeq \phi^T(x; \alpha)H_0. \quad (26)$$

Now, by using the Tau method in [14], we could generate $m \times n$ equations from (24) as follows:

$$\begin{aligned} & \int_0^1 \int_0^1 L_i^\alpha(x)\phi^T(x; \alpha)[\tilde{A}^T(D^\alpha)^T U + UD^{(\beta)}\tilde{B} + \tilde{C}^T(D^{(1)})^T U + \tilde{E}^T U]\phi(t; \beta)L_j^\beta(t)dxdt \\ & \simeq \int_0^1 \int_0^1 L_i^\alpha(x)\phi^T(x; \alpha)F\phi(t; \beta)L_j^\beta(t)dxdt. \end{aligned} \quad (27)$$

By using the orthogonality property (6), we have

$$\int_0^1 \int_0^1 L_i^\alpha(x) \phi^T(x; \alpha) \phi(t; \beta) L_j^\beta(t) dx dt = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ \frac{1}{(2i\alpha+1)(2j\beta+1)} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix},$$

for $i = 0, 1, \dots, m-1$ and $j = 0, 1, \dots, n-1$. Therefore, we rewrite (27) as follows:

$$\tilde{A}^T (D^\alpha)^T U + U D^{(\beta)} \tilde{B} + \tilde{C}^T (D^{(1)})^T U + \tilde{E}^T U \simeq F. \quad (28)$$

Also, we generate $2n + m + 1$ equations from conditions (25) and (26) as follows:

$$\phi^T(0; \alpha) U \simeq G_0^T, \quad j = 0, 1, \dots, n, \quad (29)$$

$$U \phi(0; \beta) \simeq H_0, \quad i = 0, 1, \dots, m-2. \quad (30)$$

Now, we have a system of the $mn + m + n + 1$ algebraic equations of Equations (28)–(30) with the $mn + m + n + 1$ elements of the unknown matrix U . After solving this matrix system, by using the fixed point method, we can find the approximate solution $u_{m,n}(x, t) = \phi^T(x; \alpha) U \phi(t; \beta)$.

5.2 Numerical solution of nonlinear FPDEs

To study the various types of nonlinear FPDEs, the following forms are considered:

(a) Time-fractional convection-diffusion equation:

A class of time-fractional convection-diffusion equation may be written as follows:

$$\frac{\partial^\alpha u}{\partial t^\alpha} = u_{xx} - cu_x + N(u) + f(x, t), \quad (31)$$

$$u(0, t) = g_0(t), \quad u(1, t) = g_1(t), \quad u(x, 0) = h_0(x), \quad 0 < x < 1, t > 0, \quad (32)$$

where $N(u)$ is a nonlinear operator. In this paper, we choose one or more terms of the nonlinear operator $N(u)$ in the form of u^k , $k \in \mathbb{N}$, uu_x , or uu_{xx} .

(b) Time- fractional-order Klein–Gordon equation:

A class of time-fractional order Klein–Gordon equation reads as follows:

$$\begin{aligned} \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + \frac{\partial^2 u(x, t)}{\partial x^2} + au(x, t) + bu^2(x, t) + cu^3(x, t) &= f(x, t), \\ u(0, t) &= g_0(t), \quad u(1, t) = g_1(t), \quad u(x, 0) = h_0(x). \end{aligned}$$

(c) Time- fractional-order gas dynamics equation:

A class of time-fractional order gas dynamics equation reads as follows:

$$\begin{aligned} \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + u(x, t) \frac{\partial u(x, t)}{\partial x} - u(x, t)(1 - u(x, t)) &= f(x, t), \\ u(0, t) &= g_0(t), \quad u(x, 0) = h_0(x). \end{aligned}$$

To solve problem **(a)**, **(b)**, or **(c)**, we approximate the unknown function $u(x, t)$ as follows:

$$\begin{aligned} u(x, t) &\simeq u_{m,n}(x, t) = \sum_{i=0}^m \sum_{j=0}^n u_{ij} L_i(x) L_j(t; \alpha) \\ &= U^T \Upsilon(x, t) = U^T (\phi(x) \otimes \phi(t; \alpha)), \end{aligned}$$

where $\otimes$ denotes the Kronecker product and $L_i(x) = L_i(x; 1)$ are the Muntz–Legendre polynomials with $\alpha = 1$. Also $\phi(x)$, $\phi(t; \alpha)$, U , and $\Upsilon(x, t)$ read as

$$\begin{aligned} \phi(x) &= \phi(x; 1) = [L_0(x), L_1(x), \dots, L_m(x)]^T, \\ \phi(t; \alpha) &= [L_0(t; \alpha), L_1(t; \alpha), \dots, L_n(t; \alpha)]^T, \end{aligned}$$

$$U = \begin{bmatrix} u_{00} \\ u_{01} \\ \vdots \\ u_{0n} \\ u_{10} \\ \vdots \\ u_{i,j-1} \\ u_{i,j} \\ u_{i,j+1} \\ \vdots \\ u_{m,n-1} \\ u_{mn} \end{bmatrix}, \quad \Upsilon(x, t) = \begin{bmatrix} L_0(x)L_0(t; \alpha) \\ L_0(x)L_1(t; \alpha) \\ \vdots \\ L_0(x)L_n(t; \alpha) \\ L_1(x)L_0(t; \alpha) \\ \vdots \\ L_i(x)L_{j-1}(t; \alpha) \\ L_i(x)L_j(t; \alpha) \\ L_i(x)L_{j+1}(t; \alpha) \\ \vdots \\ L_m(x)L_{n-1}(t; \alpha) \\ L_m(x)L_n(t; \alpha) \end{bmatrix}.$$

Furthermore, the first and second order derivative operator matrices of the vector $\Upsilon(x, t)$ can be obtained by

$$\begin{aligned}\frac{\partial}{\partial x}\Upsilon(x, t) &\simeq \mathbf{D}_x\Upsilon(x, t), & \mathbf{D}_x &= D_x^{(1)} \otimes I_{n+1}, \\ \frac{\partial^2}{\partial x^2}\Upsilon(x, t) &\simeq \mathbf{D}_x^2\Upsilon(x, t), & \mathbf{D}_x^2 &= D_x^{(2)} \otimes I_{n+1},\end{aligned}$$

where $D^{(1)}$ and $D^{(2)}$ are the first and second order derivative matrices of the vector $\phi(x)$, respectively.

The Caputo fractional derivatives operator of order $\alpha > 0$ of the vector $\Upsilon(x, t)$ can be obtained by

$$D_t^\alpha \Upsilon(x, t) \simeq \mathbf{D}_t^\alpha \Upsilon(x, t), \quad \mathbf{D}_t^\alpha = I_{m+1} \otimes D_t^{(\alpha)},$$

where $D_t^{(\alpha)}$ is the Caputo fractional derivatives matrix of vector $\phi(t; \alpha)$, which is obtained in Theorem 5.

Remark 4. The product matrix of FMLPs vectors can be obtained as follows:

$$\Upsilon(x, t)\Upsilon^T(x, t)U \simeq \tilde{U}\Upsilon(x, t),$$

where $\tilde{U}$ is an $(m+1)(n+1) \times (m+1)(n+1)$ product matrix for the vector U as follows:

$$\tilde{U} = [\tilde{U}^{0,0}, \tilde{U}^{0,1}, \dots, \tilde{U}^{0,n}, \tilde{U}^{1,0}, \dots, \tilde{U}^{1,n}, \dots, \tilde{U}^{m,0}, \dots, \tilde{U}^{m,n}]^T,$$

where $\tilde{U}^{p,q}$, $p = 0, 1, \dots, m$, $q = 0, 1, \dots, n$, is an $1 \times (m+1)(n+1)$ matrix as

$$\tilde{U}^{p,q} = [\tilde{U}_{0,0}^{p,q}, \tilde{U}_{0,1}^{p,q}, \dots, \tilde{U}_{0,n}^{p,q}, \tilde{U}_{1,0}^{p,q}, \dots, \tilde{U}_{1,n}^{p,q}, \dots, \tilde{U}_{m,0}^{p,q}, \dots, \tilde{U}_{m,n}^{p,q}],$$

and each element of $\tilde{U}^{p,q}$ is obtained as

$$\tilde{U}_{k,l}^{p,q} = (2k+1)(2l+1) \sum_{i=0}^m \sum_{j=0}^n u_{ij} g_{ijpqkl}, \quad p, k = 0, 1, \dots, m, \quad q, l = 0, 1, \dots, n,$$

where g_{ijpqkl} is given by

$$g_{ijpqkl} = \int_0^1 \int_0^1 L_i(x) L_j(t; \alpha) L_p(x) L_q(t; \alpha) L_k(x) L_l(t; \alpha) dx dt.$$

Remark 5. Assume that $f(x, t) \simeq F^T \Upsilon(x, t)$ and $g(x, t) \simeq G^T \Upsilon(x, t)$, where F and G are an $(m+1)(n+1)$ -vector as

$$\begin{aligned}F &= [f_{0,0}, f_{0,1}, \dots, f_{0,n}, f_{1,0}, \dots, f_{i,j-1}, f_{i,j}, f_{i,j+1}, \dots, f_{m,n-1}, f_{m,n}]^T, \\ G &= [g_{0,0}, g_{0,1}, \dots, g_{0,n}, g_{1,0}, \dots, g_{i,j-1}, g_{i,j}, g_{i,j+1}, \dots, g_{m,n-1}, g_{m,n}]^T,\end{aligned}$$

where, from Definition 4, each element of F and G is obtained as

$$\begin{aligned} f_{i,j} &= (2i+1)(2j\alpha+1) \int_0^1 \int_0^1 f(x,t) L_i(x) L_j(t; \alpha) dx dt, \\ g_{i,j} &= (2i+1)(2j\alpha+1) \int_0^1 \int_0^1 g(x,t) L_i(x) L_j(t; \alpha) dx dt. \end{aligned}$$

We can write

$$f(x,t)g(x,t) \simeq A^T \Upsilon(x,t) \Upsilon^T(x,t) B \simeq A^T \tilde{B} \Upsilon(x,t) \simeq B^T \tilde{A} \Upsilon(x,t).$$

Lemma 4. The nonlinear operator $N(u(x,t)) = u^k(x,t)$, where $u(x,t) \simeq U^T \Upsilon(x,t)$, can be approximated by

$$u^k(x,t) \simeq U^T \tilde{U}^{k-1} \Upsilon(x,t).$$

Proof. From Remark 5, we have

$$\begin{aligned} u^2(x,t) &= u(x,t)u(x,t) \simeq U^T \tilde{U} \Upsilon(x,t), \\ u^k(x,t) &= u^2(x,t)u^{k-2}(x,t) \simeq U^T \tilde{U} \Upsilon(x,t)u(x,t)u^{k-3}(x,t) \\ &\simeq U^T \tilde{U} \Upsilon(x,t) \Upsilon^T(x,t) U u^{k-3}(x,t) \\ &\simeq U^T \tilde{U}^2 \Upsilon(x,t)u^{k-3}(x,t) \simeq \dots \simeq U^T \tilde{U}^{k-1} \Upsilon(x,t). \end{aligned}$$

□

Remark 6. The nonlinear operator $N_1(u, u_x) = uu_x$ and $N_2(u, u_{xx}) = uu_{xx}$, can be approximated by

$$\begin{aligned} N_1(u, u_x) &= uu_x \simeq U^T \mathbf{D}_x \tilde{U} \Upsilon(x,t), \\ N_2(u, u_{xx}) &= uu_{xx} \simeq U^T \mathbf{D}_x^2 \tilde{U} \Upsilon(x,t). \end{aligned}$$

Now, to solve one of the problems (a), (b), or (c), for instance, problem (a), with the nonlinear operator $N(u) = uu_x + uu_{xx} + u^2$, by applying the above approximations, we rewrite (31) as

$$[U^T (\mathbf{D}_t^\alpha - \mathbf{D}_x^2 + c\mathbf{D}_x) - \tilde{N}] \Upsilon(x,t) = F^T \Upsilon(x,t), \quad (33)$$

where

$$N(u) \simeq \tilde{N} \Upsilon(x,t) = U^T [\mathbf{D}_x \tilde{U} + \mathbf{D}_x^2 \tilde{U} + \tilde{U}] \Upsilon(x,t), \quad f(x,t) \simeq F^T \Upsilon(x,t).$$

As in the Tau method in [14], we should generate $(m-1)n$ equations from (33) as follows:

$$[U^T (\mathbf{D}_t^\alpha - \mathbf{D}_x^2 + c\mathbf{D}_x) - \tilde{N}] \int_0^1 \int_0^1 \Upsilon(x,t) L_i(x) L_j(t; \alpha) dx dt \quad (34)$$

$$= F^T \int_0^1 \int_0^1 \Upsilon(x, t) L_i(x) L_j(t; \alpha) dx dt,$$

for $i = 0, 1, \dots, m-2$ and $j = 0, 1, \dots, n-1$. On the other hand, by using the orthogonality property (6), for $i = 0, 1, \dots, m-2$ and $j = 0, 1, \dots, n-1$, we have

$$\int_0^1 \int_0^1 \Upsilon(x, t) L_i(x) L_j(t; \alpha) dx dt = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ \frac{1}{(2i+1)(2j\alpha+1)} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}.$$

Therefore, we rewrite $(m-1)n$ equations (34) as follows:

$$[U^T (\mathbf{D}_t^\alpha - \mathbf{D}_x^2 + c\mathbf{D}_x) - \tilde{N}] = F^T. \quad (35)$$

Also, we generate $2n + m + 1$ equations from conditions (32) as follows:

$$U^T (\phi(0) \otimes I_{n+1}) = G_0^T, \quad g_0(t) = G_0^T \phi(t; \alpha), \quad j = 0, 1, \dots, n, \quad (36)$$

$$U^T (\phi(1) \otimes I_{n+1}) = G_1^T, \quad g_1(t) = G_1^T \phi(t; \alpha), \quad j = 0, 1, \dots, n, \quad (37)$$

$$U^T (I_{m+1} \otimes \phi(0; \alpha)) = H_0^T, \quad h_0(x) = H_0^T \phi(x), \quad i = 0, 1, \dots, m-2. \quad (38)$$

Now, we have a system of the $mn + m + n + 1$ nonlinear algebraic equations of (35)–(38) with the $mn + m + n + 1$ elements of the unknown matrix U . After solving this linear system, by using the fixed point method, we can find the approximate solution $u_{m,n}(x, t) = U^T \Upsilon(x, t)$.

6 Numerical illustration

In this section, we present some examples of linear and nonlinear of FPDEs to show the efficiency of the proposed method. The results will be compared with the exact solutions and other methods. The accuracy of the presented

method is estimated by the absolute error $E_{m,n}$ and the maximum absolute error $\epsilon_{m,n}$, which are given as follows:

$$(E_{m,n})_{i,j} = |u(\theta_i, \beta_j) - u_{m,n}(\theta_i, \beta_j)|,$$

$$\epsilon_{m,n} = \max\{|u(\theta_i, \beta_j) - u_{m,n}(\theta_i, \beta_j)|\},$$

where θ_i and β_j are defined as,

$$(\theta_i, \beta_j) = (x_i^{\frac{1}{\alpha}}, t_j^{\frac{1}{\beta}}), \quad i = 0, 1, \dots, m, \quad j = 0, 1, \dots, n,$$

where x_i and t_j are Chebyshev–Gauss–Lobatto points with the following relations:

$$x_i = \frac{1}{2} - \frac{1}{2} \cos \frac{\pi i}{m}, \quad t_j = \frac{1}{2} - \frac{1}{2} \cos \frac{\pi j}{n}.$$

Example 1. Consider the linear FPDE with variable coefficients

$$x \frac{\partial^\alpha u(x, t)}{\partial x^\alpha} + \frac{\partial^\beta u(x, t)}{\partial t^\beta} = \frac{1}{\Gamma(2-\alpha)} x^{2-\alpha} t + \frac{1}{\Gamma(2-\beta)} x t^{1-\beta}, \quad (39)$$

$$u(0, t) = u(x, 0) = 0, \quad (x, t) \in (0, 1] \times (0, 1], \quad 0 < \alpha, \beta \leq 1. \quad (40)$$

The exact solution is $u(x, t) = xt$. Applying the presented method to solve (39) and (40) with $m = n = 2$, for $\alpha = \frac{1}{2}$ and $\beta = \frac{1}{2}$, we get

$$U = \begin{bmatrix} 2.5000e-01 & 2.0000e-01 & 5.0000e-02 \\ 2.0000e-01 & 1.6000e-01 & 4.0000e-02 \\ 5.0000e-02 & 4.0000e-02 & 1.0000e-02 \end{bmatrix} \simeq \begin{bmatrix} \frac{1}{4} & \frac{1}{5} & \frac{1}{20} \\ \frac{1}{5} & \frac{4}{25} & \frac{1}{25} \\ \frac{1}{20} & \frac{1}{25} & \frac{1}{100} \end{bmatrix}.$$

Therefore, the approximation solution is

$$\begin{aligned} u_{2,2}(x, t) &= \phi^T(x; \frac{1}{2}) U \phi(t; \frac{1}{2}) \\ &= [1 \ 3\sqrt{x} - 2 \ 10x - 12\sqrt{x} + 3] \begin{bmatrix} \frac{1}{4} & \frac{1}{5} & \frac{1}{20} \\ \frac{1}{5} & \frac{4}{25} & \frac{1}{25} \\ \frac{1}{20} & \frac{1}{25} & \frac{1}{100} \end{bmatrix} \begin{bmatrix} 1 \\ 3\sqrt{t} - 2 \\ 10t - 12\sqrt{t} + 3 \end{bmatrix} = xt, \end{aligned}$$

which is actually the exact solution. The results are shown in Tables 1 and 2. Table 1 shows the comparison of the presented method with other methods in [3, 14]. From Table 1, it is seen that we obtained a lesser error. Table 2 shows the maximum absolute error $\epsilon_{m,n}$ for various values of m, n, α , and β . It can be seen that the approximate solution is obtained from the proposed method is in a good agreement with the exact solution.

Example 2. Consider the linear FPDE with variable coefficients

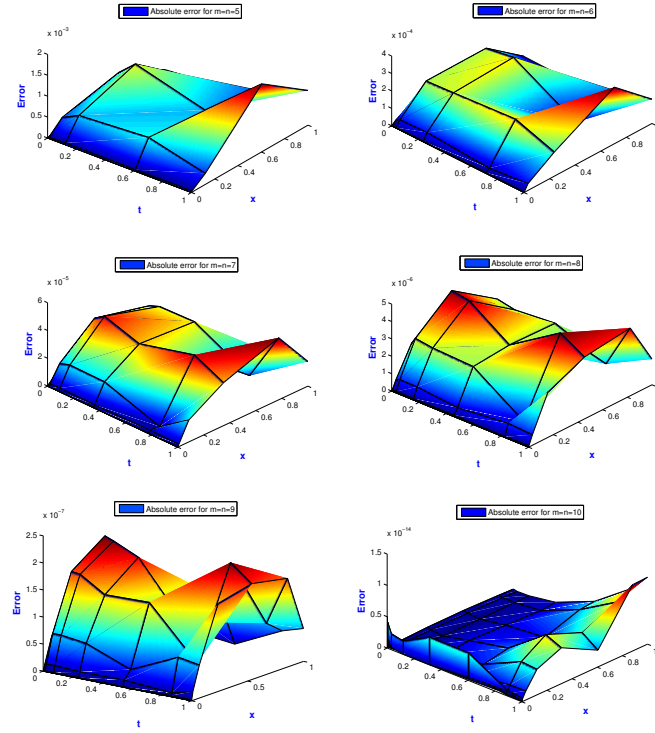


Figure 2: The absolute error function with $\alpha = \beta = \frac{1}{5}$ and $m = n = 5, 6, 7, 8, 9, 10$ for Example 2

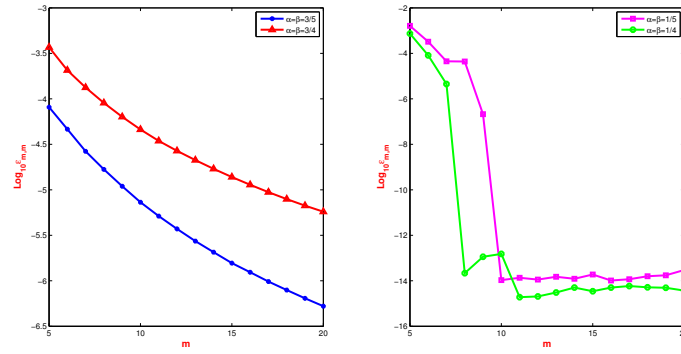


Figure 3: The logarithmic graphs of $\epsilon_{m,m}$ for Example 2

Table 1: The absolute error for example 1

(x, t)	the presented method	method in [3]	method in [14]
	$\alpha = \beta = \frac{1}{2}$ $m = n = 2$	$\alpha = \beta = \frac{1}{2}$ $m = n = 2$	$\alpha = \beta = \frac{1}{2}$ $\alpha = \beta = 1$
(0.1, 0.2)	$3.4694e - 18$	$3.47e - 17$	$1.20e - 04$
(0.5, 0.2)	0	0	$7.61e - 03$
(1.1, 0.2)	$2.7756e - 17$	$5.55e - 17$	$2.11e - 02$
(1.5, 0.2)	$5.5511e - 17$	$1.39e - 16$	$3.27e - 02$
(1.9, 0.2)	0	$1.67e - 16$	$4.65e - 02$

Table 2: Maximum absolute error for Example 1

$\varepsilon_{m,n}$	$\alpha = \beta = \frac{1}{5}$	$\alpha = \beta = \frac{1}{3}$	$\alpha = \beta = 1$
$\varepsilon_{5,5}$	$4.3696e - 14$	$3.8858e - 15$	$3.1086e - 15$
$\varepsilon_{7,7}$	$7.1054e - 15$	$1.4433e - 15$	$1.1102e - 15$

$$\sin x \frac{\partial^\alpha u(x, t)}{\partial x^\alpha} + (1 + \cos t) \frac{\partial^\beta u(x, t)}{\partial t^\beta} + \sqrt{x} u(x, t) = g(x, t),$$

$$u(0, t) = u(x, 0) = 0, \quad (x, t) \in (0, 1] \times (0, 1], \quad 0 < \alpha, \beta < 1,$$

where

$$g(x, t) = t^2 \sin x \left(\frac{x^{1-\alpha}}{\Gamma(2-\alpha)} - \frac{2x^{2-\alpha}}{\Gamma(3-\alpha)} \right) + (1-x) \left(x(1+\cos x) \frac{2t^{2-\beta}}{\Gamma(3-\beta)} + x^{\frac{3}{2}} t^2 \right).$$

The exact solution is $u(x, t) = t^2(x - x^2)$. The results are shown in Table 3 and Figures 2 and 3. Figure 2 shows the absolute error function $E_{m,n}$ for $\alpha = \beta = \frac{1}{5}$ and various values of m and n . Figure 3 shows the logarithmic graphs of $\varepsilon_{m,n}$ ($\log_{10} \varepsilon_{m,n}$) for various values of α . It is seen that the absolute error converges to zero by increasing m and n ($m = n$).

Table 3: Maximum absolute error for Example 2

$\varepsilon_{m,n}$	$\alpha = \beta = \frac{1}{5}$	$\alpha = \beta = \frac{1}{4}$	$\alpha = \beta = \frac{3}{5}$	$\alpha = \beta = \frac{3}{4}$
$\varepsilon_{5,5}$	$1.6256e - 03$	$7.4183e - 04$	$8.1055e - 05$	$3.6924e - 04$
$\varepsilon_{6,6}$	$3.2465e - 04$	$8.1570e - 05$	$4.6443e - 05$	$2.0615e - 04$
$\varepsilon_{7,7}$	$4.4705e - 05$	$4.4833e - 06$	$2.6524e - 05$	$1.3334e - 04$
$\varepsilon_{8,8}$	$4.3807e - 05$	$2.1630e - 14$	$1.6770e - 05$	$9.0206e - 05$
$\varepsilon_{9,9}$	$2.1148e - 07$	$1.1367e - 13$	$1.0939e - 05$	$6.3418e - 05$
$\varepsilon_{10,10}$	$1.0694e - 14$	$1.5205e - 13$	$7.3022e - 06$	$4.6034e - 05$

Example 3. Consider the following time-fractional order Klein–Gordon equation

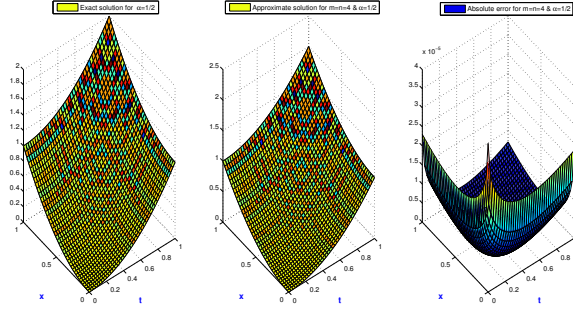


Figure 4: Comparison between the exact solution and the approximate solution with absolute error function $E_{4,4}(x, t)$ for $\alpha = 1/2$ for Example 3

$$\frac{\partial^\alpha u(x, t)}{\partial t^\alpha} - \frac{\partial^2 u(x, t)}{\partial x^2} - u(x, t) = \frac{2t^{2-\alpha}}{\Gamma(3-\alpha)} - x^2 - t^2 - 2,$$

$$u(0, t) = t^2, u(x, 0) = x^2, u(1, t) = 1 + t^2, \quad (x, t) \in (0, 1) \times (0, 1], \quad 0 < \alpha < 1.$$

The exact solution is $u(x, t) = x^2 + t^2$. Applying the presented method to solve this problem for $\alpha = \frac{1}{2}$ with $m = 4, n = 4$, we get

$$U = \begin{bmatrix} 6.6667e-01 & 3.8095e-01 & 2.1429e-01 & 6.3492e-02 & 7.9365e-03 \\ 3.8095e-01 & 1.3292e-13 & -9.7353e-14 & 3.7575e-14 & 5.2979e-15 \\ 2.1429e-01 & 1.0596e-14 & -7.8481e-15 & -2.3027e-14 & -3.2506e-15 \\ 6.3492e-02 & -3.3587e-14 & 2.6891e-14 & -1.3973e-14 & -1.8461e-15 \\ 7.9365e-03 & -1.0251e-14 & 8.1934e-15 & 1.3784e-15 & 5.2590e-17 \end{bmatrix}.$$

Therefore, the approximation solution is obtained as

$$u_{4,4}(x, t) = \phi^T(x; \frac{1}{2}) U \phi(t; \frac{1}{2})$$

$$= \begin{bmatrix} 1 \\ 3x^{\frac{1}{2}} - 2 \\ 10x - 12x^{\frac{1}{2}} + 3 \\ 30x^{\frac{1}{2}} + 35x^{\frac{3}{2}} - 60x - 4 \\ 210x - 280x^{\frac{3}{2}} - 60x^{\frac{1}{2}} + 126x^2 + 5 \end{bmatrix}^T \times U \times \begin{bmatrix} 1 \\ 3t^{\frac{1}{2}} - 2 \\ 10t - 12t^{\frac{1}{2}} + 3 \\ 30t^{\frac{1}{2}} + 35t^{\frac{3}{2}} - 60t - 4 \\ 210t - 280t^{\frac{3}{2}} - 60t^{\frac{1}{2}} + 126t^2 + 5 \end{bmatrix}$$

$$= 0.000045x - 0.00006x^{\frac{1}{2}} - (3t^{\frac{1}{2}} - 2)((1.2094e - 13)x^{\frac{1}{2}} - (1.6947e - 12)x^{\frac{3}{2}} \\ + (3.1530e - 14)x + (1.2916e - 12)x^2 - 3.8095e - 01) + x^2 + ((8.9304e - 14)x \\ - (7.9339e - 14)x^{\frac{3}{2}} - (3.6375e - 15)x^{\frac{1}{2}} + (6.6263e - 15)x^2 + 7.9365e - 03)(210t \\ - 280t^{\frac{3}{2}} - 60t^{\frac{1}{2}} + 126t^2 + 5) + (30t^{\frac{1}{2}} + 35t^{\frac{3}{2}} - 60t - 4)((8.9757e - 13)x \\ - (8.7501e - 13)x^{\frac{3}{2}} - (1.1285e - 13)x^{\frac{1}{2}} + (1.7368e - 13)x^2 + 6.3492e - 02) \\ + (10t - 12t^{\frac{1}{2}} + 3)((1.1724e - 13)x^{\frac{1}{2}} - (1.3530e - 12)x^{\frac{3}{2}} \\ + (2.8673e - 14)x + (1.0324e - 12)x^2 + 2.1429e - 01) + 0.33335.$$

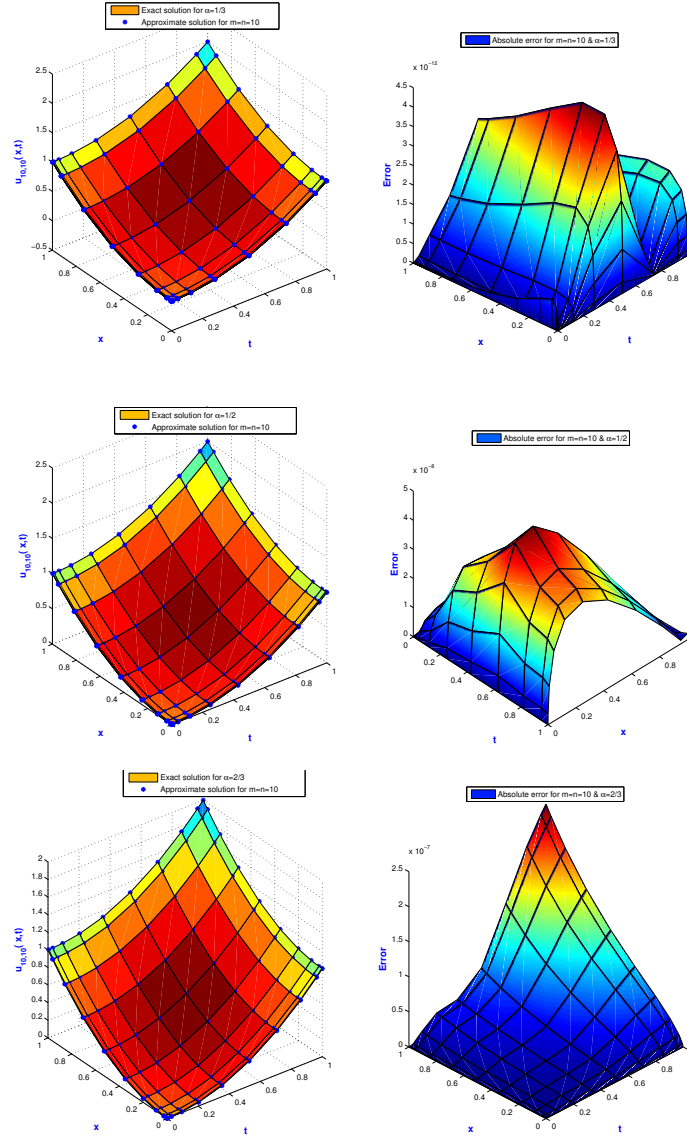


Figure 5: Comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for $m = n = 10$ and $\alpha = \frac{1}{3}, \alpha = \frac{1}{2}, \alpha = \frac{2}{3}$ for Example 3

Figure 4 shows the comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}(x, t) = |u(x, t) - u_{m,n}(x, t)|$ for $m = 4, n = 4$, and $\alpha = 1/2$. Moreover, Figure 5 shows the comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for $m = 10, n = 10$ and $\alpha = 1/3, \alpha = 1/2, \alpha = 2/3$. Also, It can be seen that there is a high level of accuracy.

Example 4. Consider the nonlinear FPDE

$$\begin{aligned} \frac{\partial^\alpha u(x, t)}{\partial t^\alpha} + u^2(x, t) + u(x, t) &= g(x, t), \quad 0 < \alpha \leq 1, \\ u(0, t) &= E_\alpha(-t^\alpha), \quad u(x, 0) = e^x, \quad (x, t) \in (0, 1] \times (0, 1], \end{aligned}$$

where

$$g(x, t) = e^{2x}(E_\alpha(-t^\alpha))^2.$$

The exact solution is $u(x, t) = e^x E_\alpha(-t^\alpha)$, where $E_\alpha(-t^\alpha)$ is the Mittag-Leffler function defined in [12].

The numerical results are shown in Figures 6 and 7. Figure 6 shows the comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for various values of m, n , and α . Also, Figure 7 shows the logarithmic graphs of $\varepsilon_{m,n}$ ($\log_{10} \varepsilon_{m,n}$) for various values of α . It is seen that the absolute error converges to zero by increasing m and n .

Table 4: Maximum absolute error for various values of m, n and α for Example 5

$\varepsilon_{m,n}$	$\alpha = 0.2$	$\alpha = 0.7$	$\alpha = 0.8$
$\varepsilon_{3,3}$	$1.3809e + 00$	$2.5202e - 02$	$1.5926e - 02$
$\varepsilon_{4,4}$	$6.7523e - 01$	$2.2084e - 03$	$1.1443e - 03$
$\varepsilon_{5,5}$	$1.5944e - 01$	$1.8012e - 04$	$7.4059e - 05$
$\varepsilon_{6,6}$	$5.5797e - 02$	$1.3632e - 05$	$4.3405e - 06$
$\varepsilon_{7,7}$	$1.6813e - 02$	$9.6897e - 07$	$2.3238e - 07$
$\varepsilon_{8,8}$	$4.9588e - 03$	$6.9251e - 08$	$1.1452e - 08$
$\varepsilon_{9,9}$	$1.5654e - 03$	$4.4995e - 09$	$5.2297e - 10$
$\varepsilon_{10,10}$	$4.5928e - 04$	$2.7060e - 10$	$2.2264e - 11$
$\varepsilon_{11,11}$	$1.3696e - 04$	$1.5228e - 11$	$8.9617e - 13$
$\varepsilon_{12,12}$	$4.0219e - 05$	$8.0180e - 13$	$1.5720e - 13$
$\varepsilon_{13,13}$	$1.1339e - 05$	$1.5720e - 13$	$1.7497e - 13$
$\varepsilon_{14,14}$	$3.2635e - 06$	$1.1945e - 13$	$4.4941e - 13$
$\varepsilon_{15,15}$	$9.0875e - 07$	$2.3048e - 12$	$8.7752e - 13$

Example 5. Consider the nonlinear time-fractional gas dynamics equation [17]

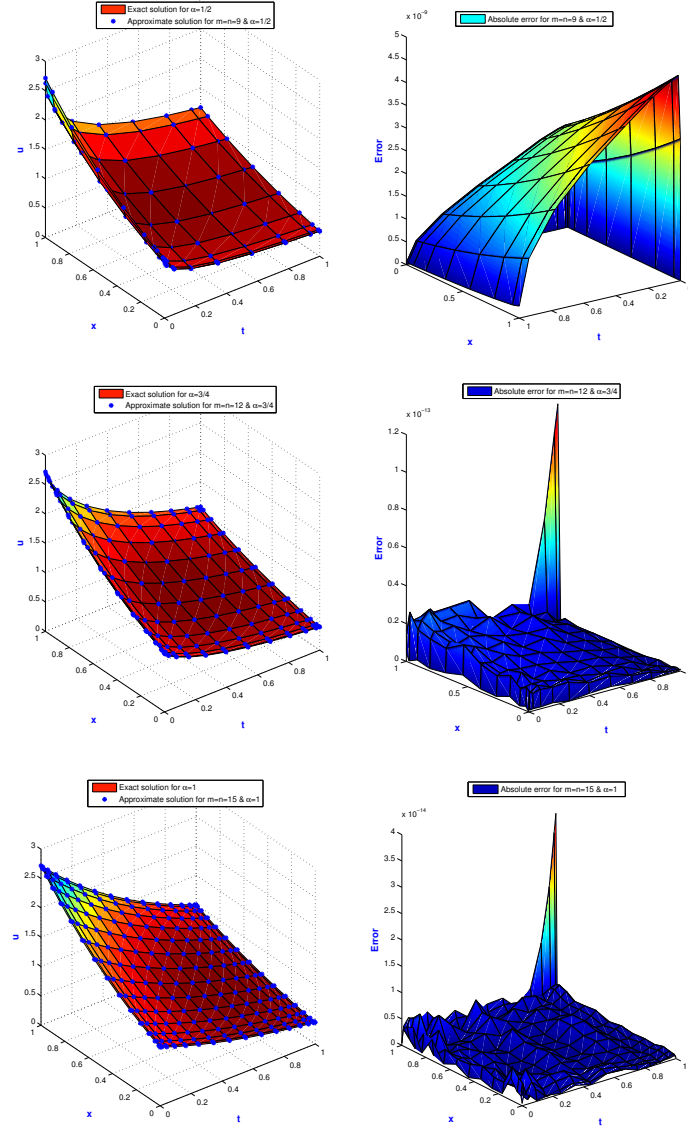
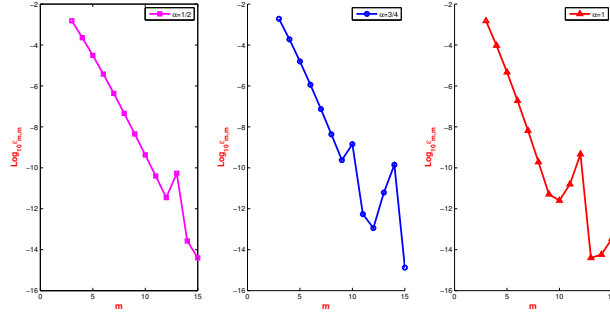


Figure 6: Comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for various values of m, n and α for Example 4

Figure 7: The logarithmic graphs of $\epsilon_{m,m}$ for Example 4

$$\begin{aligned} \frac{\partial^\alpha u(x,t)}{\partial t^\alpha} + u(x,t) \frac{\partial u(x,t)}{\partial x} - u(1-u) &= 0, \quad 0 < \alpha \leq 1, \\ u(0,t) &= E_\alpha(t^\alpha), \quad u(x,0) = e^{-x}, \quad (x,t) \in (0,1] \times (0,1]. \end{aligned}$$

The exact solution is $u(x,t) = e^{-x} E_\alpha(t^\alpha)$.

Figure 8 shows the approximate solutions obtained by the proposed method for $t = 1$ and various values of α . Similar results were obtained by Tamsir and Srivastava [17], by employing the fractional reduced differential transform method. For $\alpha = 1$, the exact solution is $u(x,t) = e^{t-x}$; see [7]. Figure 9 shows the comparison between the exact solution and the approximate solution with the absolute error function for $\alpha = 1$. We obtain a high level of accuracy, similar results in [17, 7], and we see that these results confirm the theoretical results. Also Figures 10, 11, and 12 show the comparison between the exact solution and the approximate solution with the absolute error function for various values of m, n , and $\alpha = 0.2, 0.7, 0.8$. In addition, Table 4 shows the maximum absolute error $\epsilon_{m,n}$ for various values of m, n , and $\alpha = 0.2, 0.7, 0.8$. Figure 13 shows the logarithmic graphs of $\epsilon_{m,n}$ ($\text{Log}_{10}\epsilon_{m,n}$). It is seen that the absolute error converges to zero by increasing m and n .

Example 6. Consider the inhomogeneous fractional convection-diffusion equation

$$\begin{aligned} \frac{\partial^\alpha u(x,t)}{\partial t^\alpha} - \frac{\partial^2 u(x,t)}{\partial x^2} + \frac{\partial u(x,t)}{\partial x} + u^2(x,t) + u(x,t) &= g(x,t), \quad 0 < \alpha \leq 1, \\ u(0,t) &= t^\alpha, \quad u(x,0) = x^2, \quad u(1,t) = 1 + t^\alpha, \quad (x,t) \in (0,1) \times (0,1], \end{aligned}$$

where

$$g(x,t) = \Gamma(1+\alpha) + (x^2 + t^\alpha)^2 + t^\alpha + x^2 + 2x - 2.$$

The exact solution is $u(x,t) = x^2 + t^\alpha$.
The numerical results are shown in Figure 14.

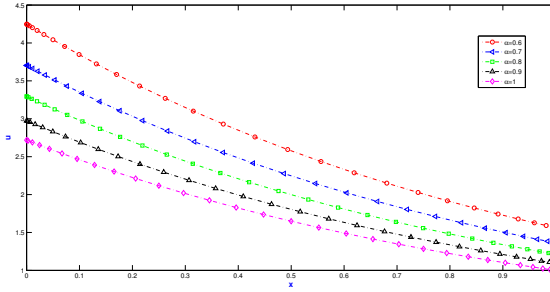


Figure 8: The approximate solutions obtained by proposed method for $t = 1$ and various values of α for Example 5

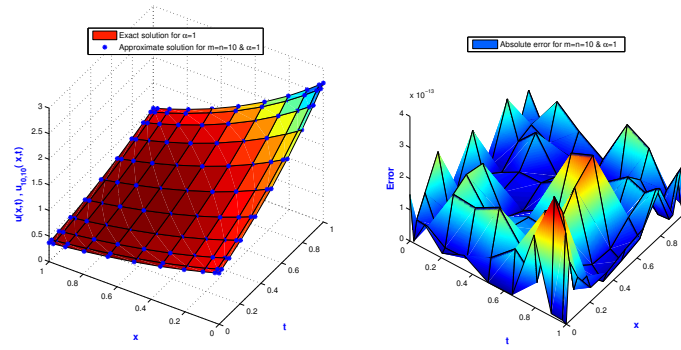


Figure 9: Comparison between the exact solution and the approximate solution with absolute error function $E_{10,10}$ for $\alpha = 1$ for Example 5

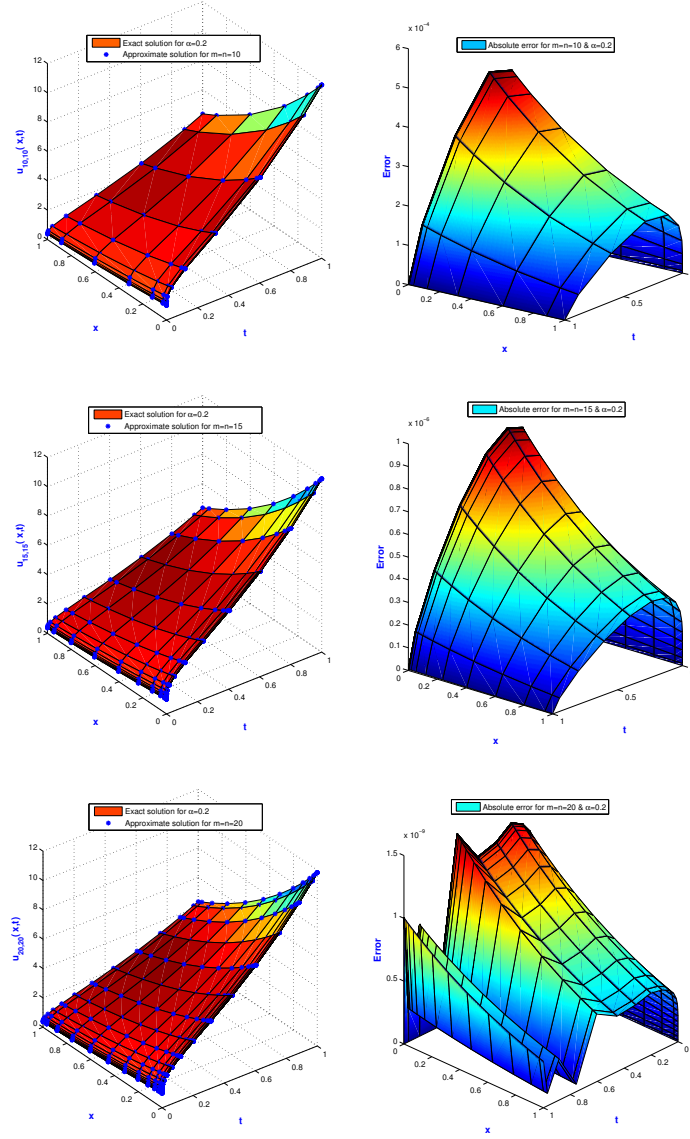


Figure 10: Comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for $m = n = 10, 15, 20$, and $\alpha = 0.2$ for Example 5

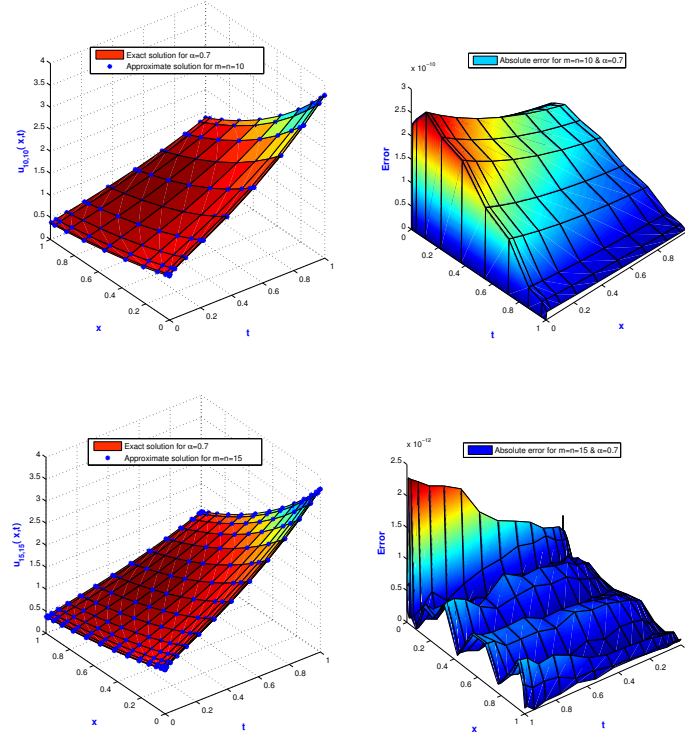


Figure 11: Comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for $m = n = 10, 15$, and $\alpha = 0.7$ for Example 5

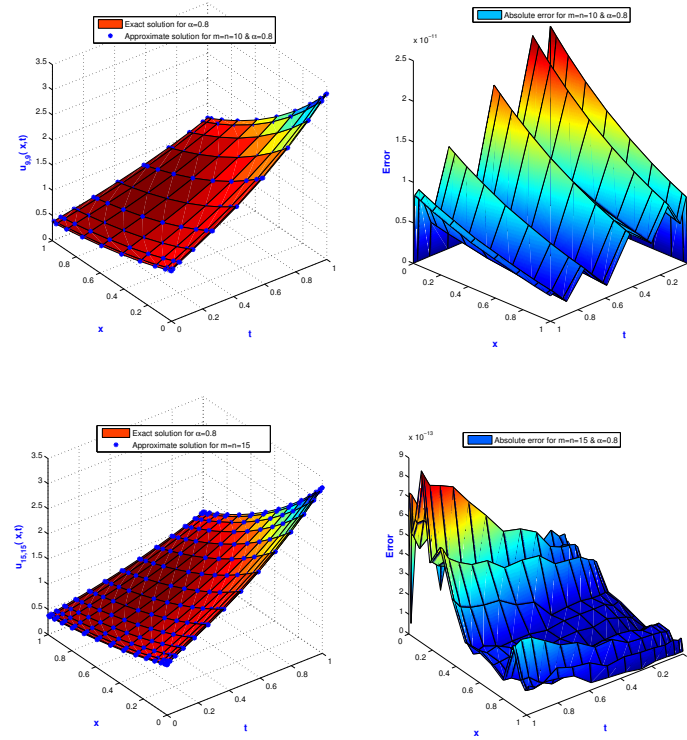


Figure 12: Comparison between the exact solution and the approximate solution with absolute error function $E_{m,n}$ for $m = n = 10, 15$, and $\alpha = 0.8$ for Example 5

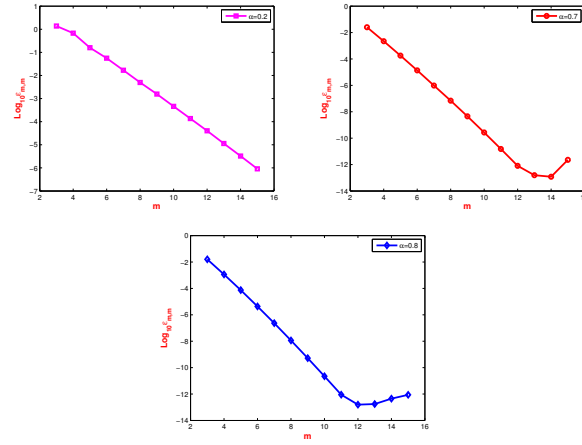


Figure 13: The logarithmic graphs of $\epsilon_{m,n}$ for $\alpha = 0.2, 0.7$ and $\alpha = 0.8$ for Example 5

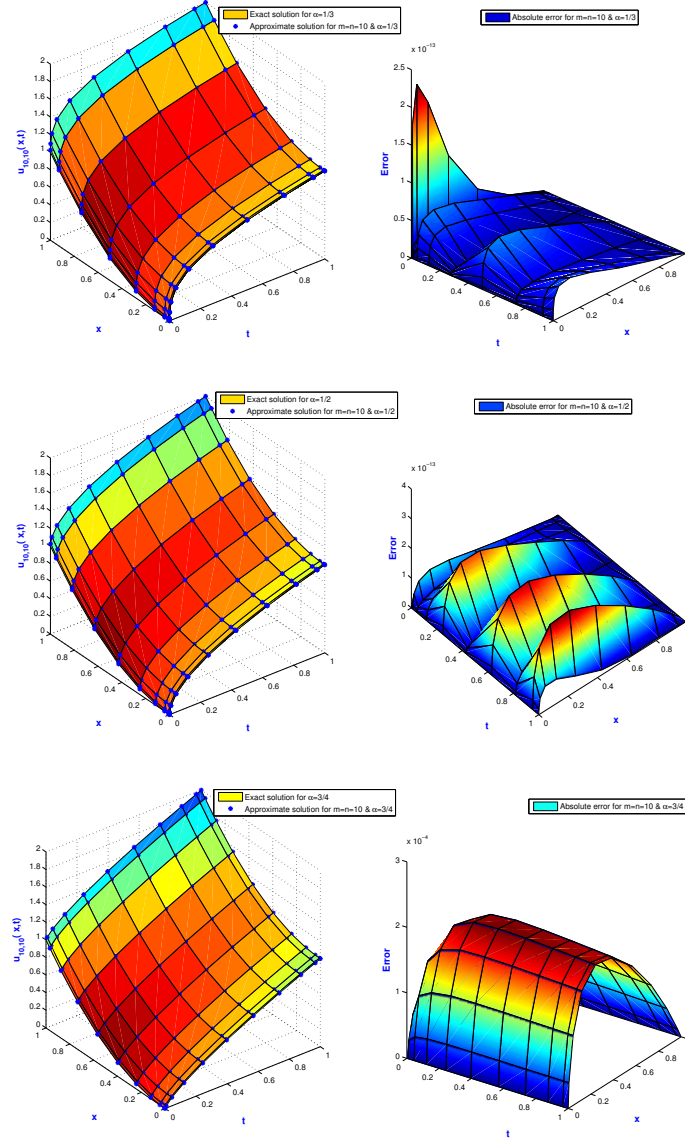


Figure 14: Comparison between the exact solution and the approximate solution with absolute error function $E_{10,10}$ at $\alpha = \frac{1}{3}, \frac{1}{2}$ and $\frac{3}{4}$ for Example 6

7 Conclusion

In this paper, we applied a basis of 2D-FMLPs to obtain the numerical solution of linear and nonlinear FPDEs with variable coefficients. To get the unknown coefficients 2D-FMLPs, we used the operational matrix of fractional derivatives of FMLPs together with the Tau method. The results of the numerical examples and the comparison with other methods showed the efficiency and accuracy of the proposed method.

Acknowledgements

Authors are grateful to there anonymous referees and editor for their constructive comments.

References

1. Abbasbandy, S. Kazem, S. Alhuthali, M.S. and Alsulami, H.H. *Application of the operational matrix of fractional-order Legendre functions for solving the time-fractional convection-diffusion equation*. Appl. Math. Comput. 266 (2015), 31–40.
2. Ahmadi Darani, M. and Nasiri, M. *A fractional type of the Chebyshev polynomials for approximation of solution of linear fractional differential equations*. Computational Methods for Differential Equations, 1(2) (2013), 96–107.
3. Chen, Y. Sun, Y. and Liu, L. *Numerical solution of fractional partial differential equations with variable coefficients using generalized fractional-order Legendre functions*. Appl. Math. Comput. 244 (2014), 847–858.
4. Esmaeili, S. Shamsi, M. and Luchko, Y. *Numerical solution of fractional differential equations with a collocation method based on Mntz polynomials*. Comput. Math. Appl. 62(3) (2011), 918–929.
5. Hilfer, R. *Applications of Fractional calculus in Physics*, World Scientific Publishing Company, Singapore, 2000.
6. Kazem, S. Abbasbandy, S. and Kumar, S. *Fractional-order Legendre functions for solving fractional-order differential equations*. Appl. Math. Model. 37 (2013) (7), 5498–5510.
7. Kumar, S. and Rashidi, M.M. *New analytical method for gas dynamics equation arising in shock fronts*, Comput. Phys. Commun. 185 (7) (2014) 1947–1954.

8. Liu, N. and Lin, E.B. *Legendre wavelet method for numerical solutions of partial differential equations*. Numer. Methods Partial Differential Equations, 26(1) (2010), 81–94.
9. Nagy, A.M. *Numerical solution of time fractional nonlinear Klein–Gordon equation using Sinc-Chebyshev collocation method*, Appl. Math. Comput. , 310 (2017) 139–148.
10. Odibat, Z. and Momani, S. *An algorithm for the numerical solution of differential equations of fractional order*, J. Appl. Math. inform. 26 (2008) 15–27.
11. Podlubny, I. *Fractional Differential Equations*, Academic Press, San Diego, 1999.
12. Podlubny, I. and Kacenak, M. *MATLAB implementation of the Mittag-Leffler function*, 2005–2009. <http://www.mathworks.com>.
13. Saadatmandi, A. and Dehghan, M. *A new operational matrix for solving fractional-order differential equations*. Comput. Math. Appl. 59(3) (2010), 1326–1336.
14. Saadatmandi, A. and Dehghan, M. *A tau approach for solution of the space fractional diffusion equation*. Comput. Math. Appl. 62(3) (2011), 1135–1142.
15. Shen, J. and Tang, T. *High order numerical methods and algorithms*. Abstract and Applied Analysis, Chinese Science Press, 2005.
16. Szeg, G. *Orthogonal Polynomials*, 4th edn (Providence, RI: American Mathematical Society), 1975.
17. Tamsir, M. and Srivastava, V.K. *Revisiting the analytical solution of gas dynamics equation*, Alex. Eng. J. 55 (2016), 867–874.
18. Yin, F. Song, J. Wu Y. and Zhang, L. *Numerical solution of the fractional partial differential equations by the two-dimensional fractional-order Legendre functions*, Abstr. Appl. Anal. 2013, Art. ID 562140, 13 pp.



Solving biobjective network flow problem associated with minimum cost-time loading[†]

O. Baghani* and S. Ghafoori

Abstract

We apply a primal-dual simplex algorithm for solving the biobjective minimum cost-time network flow problem such that the total shipping cost and the total shipping fixed time are considered as the first and second objective functions, respectively. To convert the proposed model into a single-objective parametric one, the weighted sum scalarization technique is commonly used. This problem is a mixed-integer programming, which the decision variables are directly dependent together. Generally, the previous works have considered the linear biobjective problem with the traditional network flow constraints, while in this paper, corresponding to each flow variable, a binary variable is defined. These zero-one variables are utilized to describe a fixed shipping time for positive flows. The proposed method is successful in finding all supported efficient solutions of a real numerical example.

AMS(2010): 90C29, 90C35.

Keywords: Biobjective network flow; Minimum cost-time; Primal-dual algorithm; Fixed time.

*Corresponding author

Received 1 January 2020; revised 16 February 2020 ; accepted 15 May 2020

Omid Baghani

Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Hakim Sabzevari University, P.O. Box 397, Sabzevar, Iran.

e-mail: omid.baghani@gmail.com, o.baghani@hsu.ac.ir

Saideh Ghafoori

Department of Mathematics, Faculty of Mathematics, Statistics and Computer Science, Semnan University, P.O. Box 363, Semnan, Iran.

e-mail: saeide.gh.70@gmail.com, S.ghafoori@semnam.ac.ir

[†] This article was suggested by the Scientific Committee of the Third Iranian Seminar on Control and Optimization for publication in IJNAO, which was accepted after independent review.

1 Introduction

In multi-objective network flow optimization, several objective functions, which generally are time, cost, risk, environmental concerns, and so on, have to be optimized such that the flow conservation and capacity constraints are satisfied. In actual problems, since the objectives are frequently conflicting, finding a feasible solution optimizing all objectives simultaneously is difficult or in most cases is impossible. Therefore, the aim of multi-objective optimization is to compute nondominated points rather than optimal values. A perfect review of the multi-objective cost flow problem is done in [14, 25].

The minimum cost-time network flow problem (MCTNFP) is a well-known and valuable topic in the scientific researches, which can be classified in a subset of the fixed charge network flow problems (FCNFPs). The first objective function considers the total shipping cost of flow, and the second objective function measures the total fixed shipping time of flow. Corresponding to any flow variables, there is a zero-one decision variable that is utilized to compute the total fixed shipping time on each directed arc. The weighted sum scalarization technique is used to convert the reformulated model into a parametric single objective optimization. Eusébio, Figueria, and Ehrgott [11] presented a modified primal-dual simplex algorithm based on the work done in [8] to find nondominated extreme points of the linear biobjective network flow problems. The main idea in both algorithms is that the scalarization parameter $\lambda \in (0, 1)$ is broken into subintervals such that each nondominated extreme point is associated with one and only one set in the partition. According to these papers, the primal-dual approach is considered for computing all efficient solutions of MCTNFP (see also [22, 4]).

FCNFP is a classical NP-hard combinatorial problem, (see [13]). This problem consists of two costs corresponding to each incident arc in the network: A fixed cost for the use of the arc and a variable cost proportional to the number of units sending along the arc (see [7]). The end of the FCNFP is to choose some arcs and assign feasible flows to them in order to transfer commodities from an origin to a destination at a minimal total cost. Some important traditional problems such as transportation problem, facility location problem, and network design problem can be formulated as an FCNFP. Many exact and heuristic techniques have been applied for solving the FCNFP. Analytic procedures commonly utilize branch and bound method to search an exact solution of the FCNFP (see [21]). Heuristic and evolutionary techniques have been developed to find a near-optimal solution to the under regarded problem (see [19, 20]).

Raith and Ehrgott [24] presented an algorithm to compute a complete set of efficient solutions for the biobjective integer minimum cost flow (BOIMCF) problem based on a two-phase method (see [23]). Using the mean of weighted sum technique, Abo-Sinna and Rizk-Allah [2] introduced a new gradient-based neural network approach for solving biobjective optimization, which is stable in the sense of Lyapunov. According to Benson's scalarization [6]

and min-max methods, Mohammadi, PourKarimi, and Pedram [17] studied the scientific workflow scheduling problem in a multi-cloud environment with cost and makespan minimization objectives. Eusébio and Figueria [10] solved a sequence of ϵ -constraint problem to find all integer nondominated extreme points of small or medium size of BOIMCF problem by a branch-and-bound method. Ehrgott, Shao, and Schöbel [9] implemented a method for approximating the nondominated set of a multi-objective nonlinear convex programming problem. Abareshi and Zaferanieh [1] introduced a bilevel model to solve the capacitated p -median facility location problem with the most likely allocation solution. Applying Lagrangian dual theory, the proposed bilevel problem is reduced to a new one-level nonlinear mixed-integer problem whose solution is obtained by comparing two mixed-integer linear problems.

In this study, we introduce a strategy for formulating and solving biobjective minimum cost-time network flow problems (BOMCTNFPs), which in addition to the flow variables, some corresponding zero-one decision variables are used in their construction. Keshavarz and Toloo [16] considered the weighted sum scalarization approach to convert their proposed model to a corresponding parametric mixed-integer minimum cost flow problem with single objective function and finally reported the supported efficient solutions for only two values of the scalarization parameter $\lambda = 0.05, 0.5$. The main difference of our work with [16] is that, by utilizing the primal-dual technique, all of the nondominated extreme points and the supported efficient solutions of the proposed problem are computed.

The idea of the mathematical formulation of BOMCTNFP is taken from Hochbaum and Segev [15]. They offered an augmented formulation of the fixed charge problem and proved that an optimal solution of the augmented problem is an optimal solution of the fixed charge problem. The numerical procedure that they have been implemented is based on Lagrangian relaxation, which behaves well empirically.

2 Mathematical formulation of the BOMCTNFP

Ehrgott and Puerto [8] suggested the primal-dual algorithm for solving a general form of the multi-objective programming problem. This algorithm solves the problem by constructing a partition of the set $(0, 1)$ such that each subinterval of this partition is attributed to only one efficient solution. A modified primal-dual approach for solving linear biobjective minimum cost network flow problem was developed in [11], which uses reduced cost information to avoid redundancy. Here, we apply a similar idea to find nondominated points for solving a mixed BOMCTNFP.

Suppose that $G = (\mathcal{V}, \mathcal{A})$ is a directed and connected network in which $\mathcal{V}$ is a finite set of nodes with $|\mathcal{V}| = m$, and $\mathcal{A}$ is a collection of directed pairs of

elements of $\mathcal{V}$ called arcs, with $|\mathcal{A}| = n$. Corresponding to any arc $(i, j) \in \mathcal{A}$, a triple label as (c_{ij}, t_{ij}, u_{ij}) is assigned, where the first, second, and third components are shipping cost, shipping fixed time, and capacity, respectively. Associated with every node $i \in \mathcal{V}$, there is a value b_i , which shows the supply (if $b_i > 0$) or demand (if $b_i < 0$). If $b_i = 0$ for some $i \in \mathcal{V}$, then the node i is called a transshipment node. The interested reader can consult [5, 3].

Consider the following BOMCTNFP with mixed continuous-binary variables:

$$\begin{aligned}
\min cx &= \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}, \\
\min ty &= \sum_{(i,j) \in \mathcal{A}} t_{ij} y_{ij}, \\
\text{s.t.} \quad & \sum_{(k,j) \in \mathcal{A}} x_{kj} - \sum_{(i,k) \in \mathcal{A}} x_{ik} = b_k \quad \text{for all } k \in \mathcal{V}, \\
& 0 \leq x_{ij} \leq u_{ij} \quad \text{for all } (i, j) \in \mathcal{A}, \\
& y_{ij} = \begin{cases} 1 & x_{ij} > 0, \\ 0 & x_{ij} = 0, \end{cases} \\
& y_{ij} \in \{0, 1\} \quad \text{for all } (i, j) \in \mathcal{A},
\end{aligned} \tag{1}$$

where the binary variable y_{ij} is an auxiliary variable to consider the fixed time t_{ij} for the positive flow x_{ij} on arc (i, j) . If we assume that the values of the flow variables are real, then the model (1) is a biobjective mixed-integer problem, and if the value of the flow variables is integer, then this model is a biobjective integer problem. We refer to the first case with minimum cost-time continuous flow (MCTCF) problem, and to the second one with minimum cost-time integer flow (MCTIF) problem.

The efficient solution is an important concept in the problem of multi-objective optimization. In fact, an efficient solution to a multi-objective problem is the response, which cannot improve some of the objectives without worsening other objectives. Assume that X is a set of feasible solutions to problem (1), providing the constraint $y_{ij} \in \{0, 1\}$, $(i, j) \in \mathcal{A}$ is replaced with the continuous constraint $y_{ij} \in [0, 1]$, $(i, j) \in \mathcal{A}$. The set X is called the decision space. We denote the feasible set in the objective space as $Z(x, y) = \{(cx, ty) | (x, y) \in X\}$.

Definition 1. [12] Let $(x, y), (x', y') \in X$. If $cx \leq cx', ty \leq ty'$ and $(cx, ty) \neq (cx', ty')$, then it is called (x, y) dominates (x', y') in the decision space, and equivalently (cx, ty) dominates (cx', ty') in the objective space. This notion is denoted by $Z(x, y) \preceq Z(x', y')$.

Definition 2. [18] The feasible solution $(x, y) \in X$ is an efficient solution, if there is no other feasible solution like $(x', y') \in X$ such that

$Z(x', y') \prec Z(x, y)$. If (x, y) is an efficient solution in the decision space X , then its image in the objective space is non-dominated, that is, $Z(x, y)$ is named a nondominated point. The set of all efficient solution is denoted by X_E , and the set of all nondominated points is denoted by Z_E .

Definition 3. [12] The efficient solution $(x^*, y^*) \in X_E$ is a supported efficient solution, whenever it is an optimal solution to the following weighted objective problem:

$$\min\{\lambda_1 cx + \lambda_2 ty : (x, y) \in X\}, \quad (2)$$

where $(\lambda_1, \lambda_2) \in \Lambda_0$ and $\Lambda_0 = \{(\lambda_1, \lambda_2) : \lambda_1 > 0, \lambda_2 > 0, \lambda_1 + \lambda_2 = 1\}$.

Remark 1. [8] If (x^*, y^*) is a supported efficient solution, then $Z(x^*, y^*)$ is also called a supported nondominated point. We use X_{sE} and Z_{sN} to denote sets of supported efficient solutions and supported nondominated points, respectively. The efficient solution (x^*, y^*) is a nonsupported efficient solution, if there exist some nonpositive values of λ_1 or λ_2 such that (x^*, y^*) is an optimal solution of problem (2).

3 An augmented formulation of the BOMCTNFP

In this section, we look for the supported and nonsupported efficient solutions that are the main challenges in solving mixed BOMCTNFPs. In addition, we show the relation between x_{ij} and y_{ij} , which is an additional challenge. We solve this problem with an alternative linear constraint, which is expressed in many articles in the field of fixed-charge problem (see [15, 7, 20]). First, we reformulate model (1) by replacing the third constraint with $x_{i,j} \leq u_{i,j}y_{i,j}$, as follows:

$$\begin{aligned} \min cx &= \sum_{(i,j) \in \mathcal{A}} c_{ij}x_{ij}, \\ \min ty &= \sum_{(i,j) \in \mathcal{A}} t_{ij}y_{ij}, \\ \text{s.t.} \quad &\sum_{(k,j) \in \mathcal{A}^= } x_{kj} - \sum_{(i,k) \in \mathcal{A}} x_{ik} = b_k \quad \text{for all } k \in \mathcal{V}, \\ &x_{ij} \leq u_{ij}y_{ij} \quad \text{for all } (i,j) \in \mathcal{A}, \\ &x_{ij} \geq 0, y_{ij} \in \{0, 1\} \quad \text{for all } (i,j) \in \mathcal{A}. \end{aligned} \quad (3)$$

Hochbaum and Segev [15] proved a theorem that states that an efficient solution to (3) is an efficient solution to (1). On the other hand, a fundamental theorem in multi-objective linear problem assures us that any efficient solu-

tion can be characterized as an optimal solution of the weighted sum single-objective problem (2). A proof for this fact can be found in [26]. Therefore, problem (3) can be reformulated as the following parametric problem, which provides a set of supported efficient solutions of the above problem:

$$\begin{aligned}
 & \min \lambda cx + (1 - \lambda)ty = \lambda \sum_{(i,j) \in \mathcal{A}} c_{ij}x_{ij} + (1 - \lambda) \sum_{(i,j) \in \mathcal{A}} t_{ij}y_{ij}, \\
 \text{s.t. } & \sum_{(k,j) \in \mathcal{A}} x_{kj} - \sum_{(i,k) \in \mathcal{A}} x_{ik} = b_k \quad \text{for all } k \in \mathcal{V}, \\
 & x_{ij} \leq u_{ij}y_{ij} \quad \text{for all } (i,j) \in \mathcal{A}, \\
 & x_{ij} \geq 0, y_{ij} \in \{0, 1\} \quad \text{for all } (i,j) \in \mathcal{A},
 \end{aligned} \tag{4}$$

for all $\lambda \in (0, 1)$.

4 Primal-dual algorithm for solving mixed BOMCTNFP

In the following, we use a primal-dual algorithm for the weighted sum single-objective FCNFP (4) and evaluate the efficiency of the algorithm. Since the primal-dual algorithm is based on the duality theory of problem (4), the relaxation relation $y_{ij} \in [0, 1]$ is used instead of $y_{ij} \in \{0, 1\}$. Hence, the dual problem of (4) is cast as follows:

$$\begin{aligned}
 & \max \quad \sum_{k \in \mathcal{V}} b_k \pi_k - \sum_{(i,j) \in \mathcal{A}} \delta_{ij}, \\
 \text{s.t. } & \pi_i - \pi_j - \mu_{ij} \leq c_{ij}(\lambda) \quad \text{for all } (i,j) \in \mathcal{A}, \\
 & \mu_{ij} u_{ij} - \delta_{ij} \leq t_{ij}(\lambda) \quad \text{for all } (i,j) \in \mathcal{A}, \\
 & \mu_{ij} \geq 0, \delta_{ij} \geq 0 \quad \text{for all } (i,j) \in \mathcal{A},
 \end{aligned} \tag{5}$$

where $c_{ij}(\lambda) = \lambda c_{ij}$, $t_{ij}(\lambda) = (1 - \lambda)t_{ij}$, for some $\lambda \in (0, 1)$, $\pi = (\pi_1, \dots, \pi_m)$ is the vector of dual variables corresponding to conservation constraint $\sum_{(k,j) \in \mathcal{A}} x_{kj} - \sum_{(i,k) \in \mathcal{A}} x_{ik} = b_k$, $\mu = (\mu_{i,j})_{(i,j) \in \mathcal{A}}$ is the vector of dual variables corresponding to the constraint $x_{ij} \leq u_{ij}y_{ij}$, and $\delta = (\delta_{i,j})_{(i,j) \in \mathcal{A}}$ is the vector of dual variables associated with the constraint $y_{ij} \leq 1$, that is,

$$\begin{aligned}
 \sum_{(k,j) \in \mathcal{A}} x_{kj} - \sum_{(i,k) \in \mathcal{A}} x_{ik} = b_k : & \quad \pi_k \quad \text{for all } k \in \mathcal{V}, \\
 x_{ij} \leq u_{ij}y_{ij} : & \quad \mu_{i,j} \quad \text{for all } (i,j) \in \mathcal{A}, \\
 y_{ij} \leq 1 : & \quad \delta_{i,j} \quad \text{for all } (i,j) \in \mathcal{A}.
 \end{aligned} \tag{6}$$

Since the right side of the constraints in (5) depends on λ , the feasible solutions of this problem inevitably depends on λ . Suppose that (x_{ij}^*, y_{ij}^*) and $(\pi(\lambda)^*, \mu(\lambda)^*, \delta(\lambda)^*)$ are the feasible solutions of the primal problem (4) and the dual problem (5), respectively. By using the complementary slackness conditions, (x^*, y^*) and $(\pi^*, \mu^*, \delta^*)(\lambda)$ are optimized for their corresponding problems if and only if

$$\begin{aligned} i) \quad & \left(-\pi_i^* + \pi_j^* + \mu_{ij}^* + c_{ij}(\lambda) \right) x_{ij}^* = 0 \quad \text{for all } i, j \in \mathcal{V} \text{ \& for all } (i, j) \in \mathcal{A}, \\ ii) \quad & \left(-\mu_{ij}^* u_{ij} + \delta_{ij}^* - t_{ij}(\lambda) \right) y_{ij}^* = 0 \quad \text{for all } (i, j) \in \mathcal{A}, \\ iii) \quad & \left(-x_{ij}^* + u_{ij} y_{ij}^* \right) \mu_{ij}^* = 0 \quad \text{for all } (i, j) \in \mathcal{A}, \\ iv) \quad & (1 - y_{ij}^*) \delta_{ij}^* = 0 \quad \text{for all } (i, j) \in \mathcal{A}. \end{aligned}$$

If $0 < x_{ij}^* < u_{ij}$, then $\mu_{ij}^* = 0$ and $-\pi_i^* + \pi_j^* + c_{ij}(\lambda) = 0$. Also, if $c_{ij}(\lambda) - \pi_i^* + \pi_j^* > 0$, then $x_{ij}^* = 0$ and $y_{ij}^* = 0$. If $c_{ij}(\lambda) - \pi_i^* + \pi_j^* < 0$, then consequently $\mu_{ij}^* > 0$ and $x_{ij}^* = y_{ij}^* u_{ij}$. Two cases may be occurred:

a) If $t_{ij}(\lambda) \geq \mu_{ij}^* u_{ij}$, then $\delta_{ij}^* = 0$; hence $y_{ij}^* = 0$ and $x_{ij}^* = 0$.

b) If $t_{ij}(\lambda) < \mu_{ij}^* u_{ij}$, then $\delta_{ij}^* > 0$; hence $y_{ij}^* = 1$ and $x_{ij}^* = u_{ij}$.

Let $(\pi, \mu, \delta)(\lambda)$ be a feasible solution to (5) and let Λ_q , $q = 1, 2, \dots, r$ be a partition of the interval $\Lambda_0 = (0, 1)$. Also, let $\mathcal{A}_q^- \subset \mathcal{A}$ be a set of arcs (i, j) such that for all $\lambda \in \Lambda_q$, it satisfies $\pi_i(\lambda) - \pi_j(\lambda) = c_{ij}(\lambda)$, that is,

$$\mathcal{A}_q^- = \{(i, j) \in \mathcal{A} : \pi_i(\lambda) - \pi_j(\lambda) = c_{ij}(\lambda), \lambda \in \Lambda_q\}.$$

For the arcs not in the set $\mathcal{A}_q^-$, assuming $\lambda \in \Lambda_q$, if $\pi_i - \pi_j < c_{ij}(\lambda)$, then $x_{ij} = 0$ and $y_{ij} = 0$, and if $\pi_i - \pi_j > c_{ij}(\lambda)$, then $x_{ij} = u_{ij}$ and $y_{ij} = 1$ or $x_{ij} = 0$ and $y_{ij} = 0$. We define sets L_q and U_q for $q = 1, 2, \dots, r$, as follows:

$$L_q = \{(i, j) \in \mathcal{A} : \pi_i - \pi_j < c_{ij}(\lambda), \lambda \in \Lambda_q\},$$

$$U_q = \{(i, j) \in \mathcal{A} : \pi_i - \pi_j > c_{ij}(\lambda), \lambda \in \Lambda_q\}.$$

For each interval Λ_q , we now define the restricted primal problem as follows. This problem tries to find a feasible solution for the problem of the parametric single-objective minimum cost-time network flow (4) using the arcs in the set $\mathcal{A}_q^-$, only. Consider the following restricted primal (RP($\mathcal{A}_q^-$)) problem:

$$\begin{aligned}
(\text{RP}(\mathcal{A}_q^=)) \min z &= \sum_{k \in \mathcal{V} \setminus \{1\}} s_k, \\
\text{s.t. } \sum_{(1,j) \in \mathcal{A}_q^=} x_{1j} - \sum_{(i,1) \in \mathcal{A}_q^=} x_{i1} - s_1 + \sum_{k=2}^m (-1)^{t_k} s_k &= b'_1, \\
\sum_{(k,j) \in \mathcal{A}_q^=} x_{kj} - \sum_{(i,k) \in \mathcal{A}_q^=} x_{ik} - (-1)^{t_k} s_k &= b'_k \quad \forall k \in \mathcal{V} \setminus \{1\}, \\
x_{ij} &\leq u_{ij} y_{ij} \quad \forall (i,j) \in \mathcal{A}_q^=, \\
x_{ij} &\geq 0, \quad y_{ij} \in \{0, 1\} \quad \forall (i,j) \in \mathcal{A}_q^=, \\
s_k &\geq 0 \quad \forall k \in \mathcal{V},
\end{aligned} \tag{7}$$

where

$$t_k = \begin{cases} 1 & b'_k \geq 0, \\ 0 & b'_k < 0, \end{cases}$$

in which for each $k \in \mathcal{V}$, b'_k is defined as

$$b'_k = b_k + \sum_{(i,k) \in U_q} x_{ik} - \sum_{(k,i) \in U_q} x_{ki}.$$

Problem (7) has a feasible solution as follows:

$$x_{ij} = 0, \quad y_{ij} = 0, \quad s_1 = 0, \quad s_k = |b'_k| \quad (k \in \mathcal{V} \setminus \{1\}).$$

Note that $\text{RP}(\mathcal{A}_q^=)$ does not depend on λ . Assume that $(\hat{x}, \hat{y}, \hat{s})$ is the optimal solution of problem (7). If the optimal value of the problem $\text{RP}(\mathcal{A}_q^=)$, $\hat{z}$, is zero, then the optimal solutions for problem (4) are as $x_{ij}^* = \hat{x}_{ij}$ and $y_{ij}^* = \hat{y}_{ij}$ for all $(i,j) \in \mathcal{A}_q^=$, and $x_{ij}^* = 0$ and $y_{ij}^* = 0$ or $x_{ij}^* = u_{ij}$ and $y_{ij}^* = 1$ for remaining arcs in $\mathcal{A}$. Now if $\hat{z} > 0$, then the given solution is not a feasible solution to the primal problem (4). At this time, either a new feasible dual solution should be found that improves the objective function (7) or it can be concluded that the primal problem is infeasible. Since $(\hat{x}, \hat{y}, \hat{s})$ is an optimal solution to problem (7) with the arcs in $\mathcal{A}_q^=$, so a new arc in $\mathcal{A} \setminus \mathcal{A}_q^=$ is needed to reduce the value of the objective function in problem (7). We now consider the dual of the $\text{RP}(\mathcal{A}_q^=)$ as follows:

$$\begin{aligned}
& (\text{DRP}(\mathcal{A}_q^-)) \max \sum_{k \in \mathcal{V}} b'_k \pi_k - \sum_{(i,j) \in \mathcal{A}_q^-} \delta_{ij}, \\
& \text{s.t.} \quad \pi_i - \pi_j - \mu_{ij} \leq 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-, \\
& \quad \mu_{ij} u_{ij} - \delta_{ij} \leq 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-, \\
& \quad \pi_1 \geq 0 \\
& \quad (-1)^{t_k} \pi_1 - (-1)^{t_k} \pi_k \leq 1 \quad \text{for all } k \in \mathcal{V} \setminus \{1\}, \\
& \quad \mu_{ij} \geq 0, \delta_{ij} \geq 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-.
\end{aligned} \tag{8}$$

Suppose that $(\hat{\pi}, \hat{\mu}, \hat{\delta})$ is an optimal solution for $\text{DRP}(\mathcal{A}_q^-)$. According to the complementary conditions, we obtain

$$\begin{aligned}
i) \quad & (\hat{\pi}_i - \hat{\pi}_j - \hat{\mu}_{ij}) \hat{x}_{ij} = 0 \quad \text{for all } i, j \in \mathcal{V} \text{ \& for all } (i,j) \in \mathcal{A}_q^-, \\
ii) \quad & (-\hat{\mu}_{ij} u_{ij} - \hat{\delta}_{ij}) \hat{y}_{ij} = 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-, \\
iii) \quad & (-\hat{x}_{ij} + u_{ij} \hat{y}_{ij}) \hat{\mu}_{ij} = 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-, \\
iv) \quad & (1 - \hat{y}_{ij}) \hat{\delta}_{ij} = 0 \quad \text{for all } (i,j) \in \mathcal{A}_q^-.
\end{aligned}$$

If $0 < \hat{x}_{ij} < u_{ij}$, then $\hat{\mu}_{ij} = 0$ and $\hat{\pi}_i - \hat{\pi}_j = 0$. In addition, if $\hat{\pi}_i - \hat{\pi}_j < 0$, then $\hat{x}_{ij} = 0$ and $\hat{y}_{ij} = 0$. If $\hat{\pi}_i + \hat{\pi}_j > 0$, then $\hat{\mu}_{ij} > 0$ and $\hat{x}_{ij} = \hat{y}_{ij} u_{ij}$. From the dual constraint $\hat{\mu}_{ij} u_{ij} - \hat{\delta}_{ij} \leq 0$, it can be deduced that $\hat{\delta}_{ij} > 0$, therefore $\hat{y}_{ij} = 1$ and so $\hat{x}_{ij} = 0$.

We now define the following sets:

$$\begin{aligned}
\mathcal{A}_q^{==} &= \{(i,j) \in \mathcal{A}_q^- : \hat{\pi}_i - \hat{\pi}_j = 0\}, \\
L_q^- &= \{(i,j) \in \mathcal{A}_q^- : \hat{\pi}_i - \hat{\pi}_j < 0\} = \{(i,j) \in \mathcal{A}_q^- \setminus \mathcal{A}_q^{==} : \hat{x}_{ij} = 0, \hat{y}_{ij} = 0\}, \\
U_q^- &= \{(i,j) \in \mathcal{A}_q^- : \hat{\pi}_i - \hat{\pi}_j > 0\} = \{(i,j) \in \mathcal{A}_q^- \setminus \mathcal{A}_q^{==} : \hat{x}_{ij} = u_{ij}, \hat{y}_{ij} = 1\}.
\end{aligned}$$

Now Suppose that $(\hat{\pi}, \hat{\mu}, \hat{\delta})$ is a new solution to the dual problem (5), which is defined as follows:

$$\begin{aligned}
\hat{\pi}(\lambda) &= \pi + \theta(\lambda) \hat{\pi}, \\
\hat{\mu}(\lambda) &= \mu + \theta(\lambda) \hat{\mu}, \\
\hat{\delta}(\lambda) &= \delta + \theta(\lambda) \hat{\delta},
\end{aligned} \tag{9}$$

where (π, μ, δ) is a dual feasible solution for the original dual problem (5), $(\hat{\pi}, \hat{\mu}, \hat{\delta})$ is a dual feasible solution for $\text{DRP}(\mathcal{A}_q^-)$, $q = 1, \dots, r$, and $\theta(\lambda) > 0$. For $(i,j) \in \mathcal{A}_q^-$, the new solution $(\hat{\pi}, \hat{\mu}, \hat{\delta})$ is a dual feasible solution for (5), because

$$\begin{aligned}
\hat{\pi}_i - \hat{\pi}_j - \hat{\mu}_{ij} - c_{ij} &= \left(\pi_i + \theta(\lambda) \hat{\pi}_i \right) - \left(\pi_j + \theta(\lambda) \hat{\pi}_j \right) - \left(\mu_{ij} + \theta(\lambda) \hat{\mu}_{ij} \right) - c_{ij}(\lambda) \\
&= \left(\pi_i - \pi_j - \mu_{ij} - c_{ij}(\lambda) \right) + \theta(\lambda) \left(\pi_i - \pi_j - \mu_{ij} \right) \\
&\leq 0.
\end{aligned}$$

For $(i, j) \notin \mathcal{A}_q^=$, the dual feasibility depends on the parameter $\theta(\lambda)$. The new arc $(i, j) \in \mathcal{A} \setminus \mathcal{A}_q^=$, which will be added to the current set $\mathcal{A}_q^=$, must reduce the value of objective function $\text{RP}(\mathcal{A}_q^=)$. To this end, any arc $(i, j) \in \mathcal{A} \setminus \mathcal{A}_q^=$ satisfying either $(i, j) \in L_q$ (i.e., $\pi_i - \pi_j < c_{ij}(\lambda)$) and $\hat{\pi}_i - \hat{\pi}_j > 0$ or $(i, j) \in U_q$ (i.e., $\pi_i - \pi_j > c_{ij}(\lambda)$) and $\hat{\pi}_i - \hat{\pi}_j < 0$ can be included to $\mathcal{A}_q^=$ provided that the dual feasibility for problem (5) is valid. Thus for keeping this condition, $\theta(\lambda)$ is chosen as

$$\theta(\lambda) = \min \left\{ \frac{c_{ij}(\lambda) - \pi_i + \pi_j + \mu_{ij}}{\hat{\pi}_i - \hat{\pi}_j - \hat{\mu}_{ij}} : \left((i, j) \in L_q, \hat{\pi}_i - \hat{\pi}_j > 0 \right), \right. \\
\left. \left((i, j) \in U_q, \hat{\pi}_i - \hat{\pi}_j < 0 \right) \right\}. \quad (10)$$

This process continues until $\hat{z} = 0$. At this time, we will have an optimal solution for the primal problem (4). If the condition $\hat{z} = 0$ is not satisfied, then an arc such as $(i, j) \in L_q$ such that $\hat{\pi}_i - \hat{\pi}_j > 0$ or $(i, j) \in U_q$ so that $\hat{\pi}_i - \hat{\pi}_j < 0$ cannot be found. Therefore, the starting problem (4) is infeasible.

The primal-dual algorithm for solving the BOMCTNFP is summarized in Algorithm 1.

5 A biobjective network flow example

Consider the following cities network. where 9 cities of Iran with 17 arcs are shown in Figure 1 (each city is a node). For each arc (i, j) , there is a triple (c_{ij}, t_{ij}, u_{ij}) specifying the cost (million rials), time (hours) and capacity (million tons), respectively. Also, for each city i , there is a scalar b_i that indicates the value of supply or demand (million tons). As can be seen from Figure 1, three cities produce a special commodity, that is, Mashhad, Isfahan, and Ahwaz, and the rest cities are consumers. We trade 13.5 million tons of all network goods, and supply all demands with minimum cost at the minimum time. For simplicity, each city of the network is labeled with the numbers 1 to 9, as Table 1.

Algorithm 1 Primal-dual biobjective minimum cost-time network flow algorithm

-
- 1: **Set** $\Lambda_0 := (0, 1)$, and **let** $(\pi^0, \mu^0, \delta^0)(\lambda) = (0, 0, 0)$, $\lambda \in \Lambda_0$ be an initial feasible solution of (5);
 - 2: **Set** $\mathcal{C} := \{l_0\}$, where $l_0 := \left(\Lambda_0, \mathcal{A}_0^- := \emptyset, L_0 := \mathcal{A}, U_0 := \emptyset, (\pi^0, \mu^0, \delta^0)(\lambda) \right)$;
 - 3: **While** $\mathcal{C} \neq \emptyset$ **do**
 - 4: **Select** $l_q \in \mathcal{C}$ and **solve** $\text{RP}(\mathcal{A}_q^-)$ with l_q ;
 - 5: (a) **If** the optimal value is 0, then any optimal solution is an efficient solution to (3). **Set** $\mathcal{C} := \mathcal{C} \setminus \{l_q\}$.
 - 6: (b) **Else** solve $\text{DRP}(\mathcal{A}_q^-)$. Let $(\hat{\pi}, \hat{\mu}, \hat{\delta})(\lambda)$, $\lambda \in \Lambda_q$ be an optimal solution for $\text{DRP}(\mathcal{A}_q^-)$;
 - 7: i. **If** there is no arc (i, j) such that $\pi_i - \pi_j < c_{ij}(\lambda)$ and $\hat{\pi}_i - \hat{\pi}_j > 0$ or $\pi_i - \pi_j > c_{ij}(\lambda)$ and $\hat{\pi}_i - \hat{\pi}_j < 0$, then (3) is infeasible and **STOP**.
 - 8: ii. **Else** set $\Lambda_{q,q'}$, $q' = 1 \dots, r'$ of Λ_q employing (10) and compute the new feasible solution $(\pi^{qq'}, \mu^{qq'}, \delta^{qq'})(\lambda)$, $q' = 1, \dots, r'$ according to (10).
 Put $\mathcal{C} := \mathcal{C} \setminus \{l_q\} \cup \{l_{qq'}\}$.
 - 9: **End while**
-

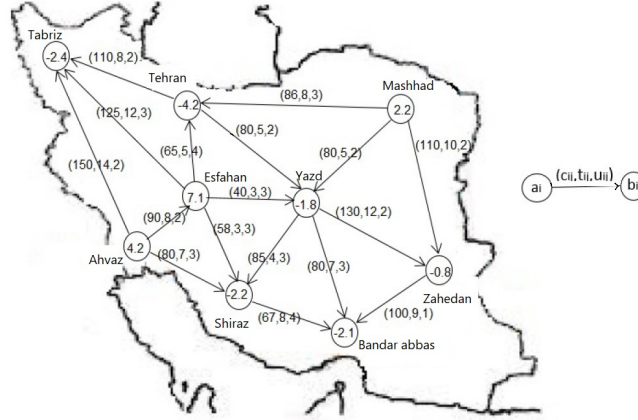


Figure 1: Graph of 9 cities of Iran with 17 arcs

Keshavarz and Toloo [16] computed the supported efficient solutions and supported nondominated points for only two values of the scalarization parameter $\lambda = 0.05, 0.5$, which is reported in Table 2. Utilizing the primal-dual technique proposed in this paper, we have obtained all the supported efficient solutions and nondominated points of the problem that are introduced in Ta-

Table 1: Labelling the cities of the network

City	Tabriz	Tehran	Esfahan	Ahvaz	Yazd	Shiraz	Mashhad	Zahedan	Bandar abbas
label	1	2	3	4	5	6	7	8	9

Table 2: Efficient solutions and nondominated points proposed in [16]

λ	Supported efficient solution	Supported nondominated point
0.05	$x_{32}^* = 3.2, x_{36}^* = 2.1, x_{35}^* = 1.8$	$(cx, ty) = (1270.9, 66)$
	$x_{72}^* = 1.4, x_{78}^* = 0.8, x_{21}^* = 0.4$	
	$x_{46}^* = 2.2, x_{41}^* = 2.0, x_{69}^* = 2.1$	
0.50	$x_{32}^* = 2.8, x_{36}^* = 0.1, x_{31}^* = 1.2$	$(cx, ty) = (1242.5, 77)$
	$x_{35}^* = 3.0, x_{72}^* = 1.4, x_{78}^* = 0.8$	
	$x_{46}^* = 3.0, x_{41}^* = 1.2, x_{69}^* = 0.9$	
	$x_{59}^* = 1.2$	

ble 3. As can be seen, some new supported efficient solutions are computed that such solutions were not investigated earlier in [16]. These new efficient solutions supply a decision maker with more alternatives, who prefers less to more in each objective.

Table 3: Efficient solutions and nondominated points computed by our method for the network of Figure 1

Supported efficient solutions		Supported nondominated points
$x_{i,j}$	$y_{i,j}$	
$x_{32}^* = 3.2, x_{36}^* = 2.1, x_{35}^* = 1.8$ $x_{72}^* = 1.4, x_{78}^* = 0.8, x_{21}^* = 0.4$ $x_{46}^* = 2.2, x_{41}^* = 2.0, x_{69}^* = 2.1$	$y_{32}^* = 1, y_{36}^* = 1, y_{35}^* = 1$ $y_{72}^* = 1, y_{78}^* = 1, y_{21}^* = 1$ $y_{46}^* = 1, y_{41}^* = 1, y_{69}^* = 1$	$(cx, ty) = (1270.9, 66)$
$x_{25}^* = x_{31}^* = x_{43}^* = x_{56}^* = 0$ $x_{58}^* = x_{59}^* = x_{75}^* = x_{89}^* = 0$	$y_{25}^* = y_{31}^* = y_{43}^* = y_{56}^* = 0$ $y_{58}^* = y_{59}^* = y_{75}^* = y_{89}^* = 0$	
$x_{32}^* = 2.8, x_{36}^* = 0.1, x_{31}^* = 1.2$ $x_{35}^* = 3.0, x_{72}^* = 1.4, x_{78}^* = 0.8$ $x_{46}^* = 3.0, x_{41}^* = 1.2, x_{69}^* = 0.9$ $x_{59}^* = 1.2$	$y_{32}^* = 1, y_{36}^* = 1, y_{31}^* = 1$ $y_{35}^* = 1, y_{72}^* = 1, y_{78}^* = 1$ $y_{46}^* = 1, y_{41}^* = 1, y_{69}^* = 1$ $y_{59}^* = 1$	$(cx, ty) = (1242.5, 77)$
$x_{21}^* = x_{25}^* = x_{43}^* = x_{56}^* = 0$ $x_{58}^* = x_{75}^* = x_{89}^* = 0$	$y_{21}^* = y_{25}^* = y_{43}^* = y_{56}^* = 0$ $y_{58}^* = y_{75}^* = y_{89}^* = 0$	
$x_{31}^* = 0.4, x_{32}^* = 2.8, x_{35}^* = 2.9$ $x_{36}^* = 3.0, x_{41}^* = 2.0, x_{43}^* = 2.0$ $x_{46}^* = 0.2, x_{59}^* = 1.1, x_{69}^* = 1.0$ $x_{72}^* = 1.4, x_{78}^* = 0.8$	$y_{31}^* = 1, y_{32}^* = 1, y_{35}^* = 1$ $y_{36}^* = 1, y_{41}^* = 1, y_{43}^* = 1$ $y_{46}^* = 1, y_{59}^* = 1, y_{69}^* = 1$ $y_{72}^* = 1, y_{78}^* = 1$	$(cx, ty) = (1381.4, 85)$
$x_{21}^* = x_{25}^* = x_{56}^* = 0$ $x_{58}^* = x_{75}^* = x_{89}^* = 0$	$y_{21}^* = y_{25}^* = y_{56}^* = 0$ $y_{58}^* = y_{75}^* = y_{89}^* = 0$	
$x_{31}^* = 0.4, x_{32}^* = 2.8, x_{35}^* = 1.8$ $x_{36}^* = 2.1, x_{41}^* = 2.0, x_{46}^* = 2.2$ $x_{69}^* = 2.1, x_{72}^* = 1.4, x_{78}^* = 0.8$	$y_{31}^* = 1, y_{32}^* = 1, y_{35}^* = 1$ $y_{36}^* = 1, y_{41}^* = 1, y_{46}^* = 1$ $y_{69}^* = 1, y_{72}^* = 1, y_{78}^* = 1$	$(cx, ty) = (1250.9, 70)$
$x_{21}^* = x_{25}^* = x_{43}^* = x_{56}^* = 0$ $x_{58}^* = x_{59}^* = x_{75}^* = x_{89}^* = 0$	$y_{21}^* = y_{25}^* = y_{43}^* = y_{56}^* = 0$ $y_{58}^* = y_{59}^* = y_{75}^* = y_{89}^* = 0$	
$x_{31}^* = 0.4, x_{32}^* = 2.8, x_{35}^* = 3.0$ $x_{36}^* = 0.9, x_{41}^* = 2.0, x_{46}^* = 2.2$ $x_{59}^* = 1.2, x_{69}^* = 0.9, x_{72}^* = 1.4$	$y_{31}^* = 1, y_{32}^* = 1, y_{35}^* = 1$ $y_{36}^* = 1, y_{41}^* = 1, y_{46}^* = 1$ $y_{59}^* = 1, y_{69}^* = 1, y_{72}^* = 1$	$(cx, ty) = (1156.9, 67)$
$x_{21}^* = x_{25}^* = x_{43}^* = x_{56}^* = 0$ $x_{58}^* = x_{75}^* = x_{78}^* = x_{89}^* = 0$	$y_{21}^* = y_{25}^* = y_{43}^* = y_{56}^* = 0$ $y_{58}^* = y_{75}^* = y_{78}^* = y_{89}^* = 0$	
$x_{31}^* = 1.2, x_{32}^* = 2.8, x_{35}^* = 1.8$ $x_{36}^* = 1.3, x_{41}^* = 1.2, x_{46}^* = 3.0$ $x_{69}^* = 2.1, x_{72}^* = 1.4, x_{78}^* = 0.8$	$y_{31}^* = 1, y_{32}^* = 1, y_{35}^* = 1$ $y_{36}^* = 1, y_{41}^* = 1, y_{46}^* = 1$ $y_{69}^* = 1, y_{72}^* = 1, y_{78}^* = 1$	$(cx, ty) = (1248.5, 70)$
$x_{21}^* = x_{25}^* = x_{43}^* = x_{56}^* = 0$ $x_{58}^* = x_{59}^* = x_{75}^* = x_{89}^* = 0$	$y_{21}^* = y_{25}^* = y_{43}^* = y_{56}^* = 0$ $y_{58}^* = y_{59}^* = y_{75}^* = y_{89}^* = 0$	

6 Computational experiments

The primal-dual approach for solving five instances of the BOMCTNFP has been implemented using MATLAB R2017a software, on a laptop equipped with Intel Core i5-8250U 1.80GHz processor and 8 GB of RAM, to find all

their efficient extreme points. Table 4 shows that when the number of nodes and arcs grows as compared to the number of supported efficient solutions, the iterations and computational cost of the algorithm extremely increase.

Table 4: Numerical results by Algorithm 1 for several instances of BOMCTNFP (nse stands for the number of supported efficient solutions)

Instances	The parameters of each instance							nse	Iterations	CPU time (sec.)
	Nodes	Arcs	Sources	Sinks	max $c_{i,j}$	max $t_{i,j}$	$u_{i,j}$			
1	4	6	2	2	4	2	1 – 2	1	2	0.02
2	5	8	3	2	7	3	1 – 3	2	5	0.98
3	8	14	3	5	26	3	1 – 3	3	43	1.24
4	9	16	5	4	34	4	1 – 4	3	137	2.78
5	14	29	10	4	62	14	1 – 5	6	198	4.53

7 Conclusion

The purpose of this paper was to solve the mixed biobjective problem of minimum cost-time network flow. To this end, we first formulated the mixed biobjective problem and proposed the primal-dual algorithm for solving it. In this article, an example with 9 city nodes and 17 arcs was solved. We found 6 supported efficient solutions in the decision space. In comparison with the method presented in [16], we could find some new supported efficient solutions to provide a decision maker with more alternatives.

Acknowledgements

The authors would like to thank the referees (especially the second reviewer) for constructive comments and suggestions, which improved the quality of the first version of this paper.

References

1. M. Abareshi, M. Zaferanieh, *A bilevel capacitated P-median facility location problem with the most likely allocation solution*, Trans. Res. B-Meth. 123 (2019) 1–20.
2. Abo-Sinna, M.A. and Rizk-Allah, R.M. *Decomposition of parametric space for biobjective optimization problem using neural network approach*, Opsearch, 55(2) (2018) 502–531.

3. Ahuja, R., Magnanti, T.L. and Orlin, J. *Network flows: Theory, algorithms, and applications*. Prentice Hall, Englewood Cliffs, 1993.
4. Arbel, A. *Multi-objective interior primal-dual linear programming algorithm*, Comput. Oper. Res. 21 (1994) 433–445.
5. Bazaraa, M.S., Jarvis, J.J. and Sherali, H.D. *Linear programming and network flows*, third edition. Wiley-Interscience [John Wiley & Sons], Hoboken, NJ, 2005.
6. Benson, H.P. and Sun, E. *Outcome space partition of the weight set in multi-objective linear programming*, J. Optimiz. Theory App. 105 (2000) 17–36.
7. Calvete, H.I., Galéb, C., Iranzo, J.A. and Toth, P. *A matheuristic for the two-stage fixed-charge transportation problem*, Comput. Oper. Res. 95 (2018) 113–122.
8. Ehrgott, M. and Puerto, J. *Primal-Dual simplex method for multi-objective linear programming*, J. Optimiz. Theory App. 134 (2007) 483–497.
9. Ehrgott, M., Shao, L. and Schöbel, A. *An approximation algorithm for convex multi-objective programming problems*, J. Global Optim. 50 (2011) 397–416.
10. Eusébio, A. and Figueira, J.R. *Finding nondominated solutions in biobjective integer network flow problems*, Comput. Oper. Res. 36 (2009) 2554–2564.
11. Eusébio, A., Figueira, J.R. and Ehrgott, M. *A primal-dual simplex algorithm for bi-objective network flow problems*, 4OR J. Oper. Res. 7 (2009) 255–273.
12. Eusébio, A., Figueira, J.R. and Ehrgott, M. *On finding representative nondominated points for biobjective integer network flow problems*, Comput. Oper. Res. 48 (2014) 1–10.
13. Guisewite, G. and Pardalos, P. *Minimum concave-cost network flow problems: Applications, complexity, and algorithms*, Ann. Oper. Res. 25 (1990) 75–99.
14. Hamacher, H.W., Pedersen, C.R. and Ruzika, S. *Multiple objective minimum cost flow problems: a review*, European J. Oper. Res. 176 (2007) 1404–1422.
15. Hochbaum, D.S. and Segev, A. *Analysis of a flow problem with fixed charges*, Networks, 19 (1989) 291–312.
16. Keshavarz, E. and Toloo, M. *A biobjective minimum cost-time network flow problem*, Procedia Econ. Finance. 23 (2015) 3–8.

17. Mohammadi, S., PourKarimi, L. and Pedram, H. *Integer linear programming based multi-objective scheduling for scientific workflows in multi-cloud environments*, J. Supercomput. 75 (2019) 6683–6709.
18. Moradi, S., Raith, A. and Ehrgott, M. *A biobjective column generation algorithm for the multi-commodity minimum cost flow problem*, European J. Oper. Res. 244 (2015) 369–378.
19. Munguía, L., Ahmed, S., Bader, D.A., Nemhauser, G.L., Goel, V. and Shao, Y. *A parallel local search framework for the fixed charge multicommodity network flow problem*, Comput. Oper. Res. 77 (2017) 44–57.
20. Nicholson, C.D. and Zhang, W. *Optimal network flow: A predictive analytics perspective on the fixed-charge network flow problem*, Comput. Ind. Eng. 99 (2016) 260–268.
21. Ortega, F. and Wolsey, L. *A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem*, Networks, 41 (2003) 143–158.
22. Przybylski, A. *Application of primal-dual simplex method for MOLP to the MO assignment problem*, Technical Report, University of Nantes, Nantes, France, 2005.
23. Przybylski, A., Gandibleux, X. and Ehrgott, M. *The two-phase method for multi-objective combinatorial optimization problems*, In A. R. Mahjoub (Ed.), Progress in combinatorial optimization (pp. 559–596), London, ISTEWiley, 2011.
24. Raith, A. and Ehrgott, M. *A two-phase algorithm for the biobjective integer minimum cost flow problem*, Comput. Oper. Res. 36 (2009) 1945–1954.
25. Sedeño-Noda, A. and Gonzalez-Martin, C. *The biobjective minimum cost flow problem*, European J. Oper. Res. 124 (2000) 591–600.
26. Steuer, R.E. *Multiple Criteria Optimization: Theory, Computation, and Application*, Wiley, New York, 1986.



A noniterative domain decomposition method for the forward-backward heat equation

S. Banei and K. Shanazari*

Abstract

A nonoverlapping domain decomposition technique applied to a finite difference method is presented for the numerical solution of the forward-backward heat equation in the case of one-dimension. While the previous attempts in dealing with this problem have been based on an iterative domain decomposition scheme, the current work avoids iterations. Also a physical matching condition is suggested to avoid difficulties caused by the interface boundary nodes. Furthermore, we obtain a square system of equations. In addition, the convergence and stability of the proposed method are investigated. Some numerical experiments are given to show the effectiveness of the proposed method.

AMS(2010): Primary 65M06; Secondary 35K20.

Keywords: Forward-backward heat equation; Nonoverlapping domain decomposition; Finite difference; Noniterative method.

1 Introduction

The analysis and numerical solution of the forward-backward heat equation (FBHE) has been under consideration for more than three decades; see [1, 8, 12, 20]. This problem appears in many applications such as boundary layer problems in fluid dynamics and steady state computation, plasma

*Corresponding author

Received 27 August 2019; revised 7 April 2020; accepted 24 May 2020

Siamak Banei

Department of Mathematics, Faculty of Sciences, University of Kurdistan, Sanandaj, Iran.
e-mail: sbaneh@gmail.com

Kamal Shanazari

Department of Mathematics, Faculty of Sciences, University of Kurdistan, Sanandaj, Iran.
e-mail: k.shanazari@uok.ac.ir

physics, and studies of the propagation of an electron beam through the solar corona (see [7] and the references therein). It also arises in computational fluid dynamics and randomly accelerated particle problem, and some more applications are mentioned in [1]. The forward-backward problem has been solved by various methods such as finite difference method (FDM) [7, 6, 13, 23], transformation to a system of first order differential equations [17], least square approach [3], Galerkin finite element [2, 10, 9, 16], and radial basis function meshless method [4].

The application of the domain decomposition method (DDM) together with FDM for this problem in the case of one-dimension is widely seen in the literature (see, for example, [7, 22]). This method was also applied to the two-dimensional FBHE, and convergence of the iterative method was considered in [23].

In addition, the theoretical aspects of the underlying problem have been studied by some authors [5, 14, 19]. Daoud [7] analyzed the convergence of the overlapping Schwarz waveform relaxation method for solving the FBHE.

Kuznetsov [15] proved the uniqueness of the entropy solution for a nonlinear forward-backward parabolic equation with Dirichlet boundary condition and showed that initial and final conditions must be formulated in the form of inequalities. Paronetto [20] considered the existence and uniqueness of the solution for elliptic problems with a small parameter and proved that the solution converges to the solution of a mixed type equation, elliptic-parabolic, parabolic both forward and backward. In the above-mentioned methods, the domain is usually divided into two subdomains, each of which is associated with either a standard forward or backward problem. A numerical method is then applied to each subproblem. A primary approximate solution is assumed on the interior boundary followed by solving each subproblem and updating the interface boundary solution iteratively. While the previous DDM-FDM has been applied based on an iterative scheme, the current work is based on applying a nonoverlapping DDM and employing an FDM without using iterations. Some advantages of the proposed method read as follows. Firstly, an initial approximate solution on the interface boundary, which may affect the convergence, is not required. Secondly, the iteration, which may considerably increase the computational costs, is avoided. In the suggested method, the domain is divided into two nonoverlapping subdomains, and a forward or a backward finite difference formulation is applied to each subproblem according to the sort of the initial conditions. The local algebraic equations produced by the subproblems are then assembled in a global system via employing a physical matching condition on the interface and obtaining a square system of equations by removing the virtual boundary nodes. This paper is organized as follows: The forward-backward equation is introduced in Section 2. The DDM will be applied to the underlying problem in Section 3. The convergence and stability of the proposed method will be discussed in Section 4. Finally some numerical results will be presented in Section 5.

2 The forward-backward equation

A boundary value problem of a forward-backward parabolic equation in one-dimension is introduced as

$$\begin{aligned} xu_t - u_{xx} &= f(x, t), & (x, t) \in \Omega = (-1, 1) \times (0, 1), \\ u(-1, t) &= g_{-1}(t), & \text{for all } t \in [0, 1], \\ u(1, t) &= g_1(t), & \text{for all } t \in [0, 1], \\ u(x, 0) &= u_0(x), & \text{for all } x \in [0, 1], \\ u(x, 1) &= u_1(x), & \text{for all } x \in [-1, 0], \end{aligned} \quad (1)$$

where $u_0(x)$, $u_1(x)$, $g_{-1}(t)$, and $g_1(t)$ are known functions with $u_0(1) = g_1(0)$ and $u_1(-1) = g_{-1}(1)$.

The existence, uniqueness, and stability of the above forward-backward problem have been considered by some authors [1, 2, 3, 18]. Authors of [1] used the energy method to prove the existence and uniqueness of a weak solution to FBHE on some special spaces equipped with some suitable norms. Lu and Wen [18] studied the existence and uniqueness of a weak solution to (1) on a certain Hilbert space. Aziz and Liu [3] reduced (1) to a first-order symmetric-positive system of differential equations and proved existence and uniqueness to the solution of the equivalent problem using the theory of symmetric-positive systems previously presented by Friedrichs [11] for a strong solution, under certain smoothness assumptions.

In order to apply the FDM, we produce a set of grids as follows: Let $h = \frac{1}{M}$, let $x_i = ih$ for $i = 0, \pm 1, \dots, \pm(M-1)$, let $\tau = \frac{1}{N}$, and let $t_j = j\tau$ for $j = 0, \dots, N$.

3 Domain decomposition method

In this section, the DDM is applied to the forward backward problem in two steps. First, the domain is divided into two subdomains and the algebraic equations are formed for each subregion. Then, the local formulations are assembled to obtain the global system of equations.

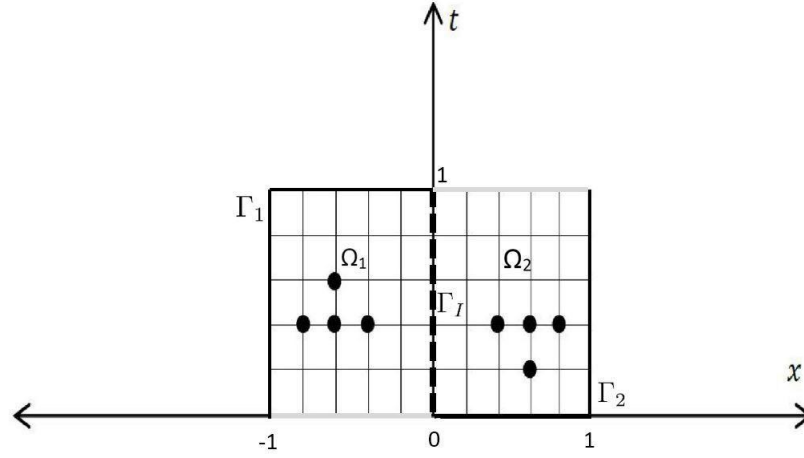


Figure 1: The domain partitioning and the pattern of the finite difference schemes.

3.1 Domain partitioning

Let the domain Ω be divided into two nonoverlapping subregions, $\Omega_1 = (-1, 0) \times (0, 1)$ and $\Omega_2 = (0, 1) \times (0, 1)$ with the real boundaries Γ_1 and Γ_2 and an interface boundary Γ_I in between (see Figure 1). In order to deal with this problem, we apply a forward and a backward difference scheme to the domains Ω_1 and Ω_2 , respectively. Employing the first order forward formulation for time derivative and the second order central difference scheme for the spatial derivative at the mesh points in Ω_1 , we obtain

$$\begin{aligned} ih \frac{u_{i,j+1} - u_{i,j}}{\tau} &= \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + f_{i,j}, \\ 0 \leq j \leq N-1, \quad -M+1 \leq i \leq -1, \\ u_{i,N} &= u_1(ih), \quad -M+1 \leq i \leq 0, \\ u_{-M,j} &= g_{-1}(j\tau), \quad 0 \leq j \leq N. \end{aligned} \quad (2)$$

where $u_{i,j}$ denotes the approximation of $u(ih, j\tau)$ and $f_{i,j} = f(ih, j\tau)$.

Using the first order backward formulation and the second order central difference scheme for the mesh points in Ω_2 , the following equations are achieved:

$$\begin{aligned} ih \frac{u_{i,j+1} - u_{i,j}}{\tau} &= \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} + f_{i,j+1}, \\ 0 \leq j \leq N-1, \quad 1 \leq i \leq M-1, \\ u_{i,0} &= u_0(ih), \quad 0 \leq i \leq M-1, \\ u_{M,j} &= g_1(j\tau), \quad 0 \leq j \leq N, \end{aligned} \quad (3)$$

The pattern of the finite difference schemes used in (2) and (3) is shown in Figure 1.

Note that the above equations produce two separate underdetermined systems of equations, as the boundary conditions on the interface Γ_I are not prescribed. In the next subsection, we propose a method in which the local equations are assembled in a global system with a square coefficient matrix.

3.2 Global formulation

We now globalize the problem by assembling the local equations constructed by the subproblems. As mentioned above, the systems obtained by equations (2) and (3) are underdetermined as the number of unknowns is equal to $2[N(M-2) + N-1]$, whereas there are only $2N(M-2)$ equations, taking into account the boundary conditions imposed on the interface boundary have different values for two subproblems.

Let $u_{0,j}^{(k)}$, $k = 1, 2$, be the solutions of the subproblems associated with Ω_k at $(0, j\tau)$, $j = 1, \dots, N$. In order to achieve a system with a square coefficient matrix, we use a sort of physical matching condition on the interface, that is, $u_{0,j} = u_{0,j}^{(1)} = u_{0,j}^{(2)}$, which guarantees a continuous solution on the virtual boundary. We now make a square system by eliminating the nodal values located on the interface. To do so, we utilize the central difference scheme for spatial derivative at the mesh points on Γ_I in order to make interaction between two subproblems, that is,

$$-\left(\frac{u_{1,j} - 2u_{0,j} + u_{-1,j}}{h^2}\right) = f_{0,j}, \quad 1 \leq j \leq N-1. \quad (4)$$

Note that the first term of the governing equation in (1) vanishes since on Γ_I , $x = 0$ holds. By combining this equation with (2) and (3) in the case of $i = 1$ and $i = -1$, we can eliminate the nodal values on the line $x = 0$ from the eventual system of equations.

To this end, we consider the first equation of (3) in the case of $i = 1$, that is,

$$h \frac{u_{1,j+1} - u_{1,j}}{\tau} = \frac{u_{2,j+1} - 2u_{1,j+1} + u_{0,j+1}^{(2)}}{h^2} + f_{1,j+1}, \quad 0 \leq j \leq N-1, \quad (5)$$

where the superscript indicates the subdomain considered.

Replacing j by $j+1$ in (4) and solving it for $u_{0,j+1} = u_{0,j+1}^{(1)}$, we obtain

$$u_{0,j+1}^{(1)} = \frac{h^2}{2} f_{0,j+1} + \frac{1}{2} u_{1,j+1} + \frac{1}{2} u_{-1,j+1}, \quad 0 \leq j \leq N-1. \quad (6)$$

Substituting (6) into (5), we find that

$$h \frac{u_{1,j+1} - u_{1,j}}{\tau} = \frac{u_{2,j+1} - \frac{3}{2}u_{1,j+1} + \frac{1}{2}u_{-1,j+1}}{h^2} + \frac{3}{2}f_{0,j+1}, \quad (7)$$

$$j = 0, \dots, N-1.$$

Repeating a similar process for the line $i = -1$ in the first equation of (2), we achieve a similar equation without presence of the term $u_{0,j+1}$ as follows:

$$-h \frac{u_{-1,j+1} - u_{-1,j}}{\tau} = \frac{u_{-2,j} - \frac{3}{2}u_{-1,j} + \frac{1}{2}u_{1,j}}{h^2} + \frac{3}{2}f_{0,j}, \quad j = 1, \dots, N. \quad (8)$$

Assembling equations (2) and (3) with the revised equations (7) and (8), a linear system of size $2N(M-2)$ is obtained as $PU = F$, where $P =$

$$\begin{bmatrix} P_{LL} & P_{LR} \\ P_{RL} & P_{RR} \end{bmatrix}, \quad U = \begin{bmatrix} U_L \\ U_R \end{bmatrix}, \quad F = \begin{bmatrix} F_L \\ F_R \end{bmatrix},$$

$$P_{LL} = \begin{bmatrix} A_L & & & \\ B_L & \ddots & & \\ & \ddots & A_L & \\ & & B_L & A_{L'} \end{bmatrix}, \quad P_{RR} = \begin{bmatrix} A_R & & & \\ B_R & \ddots & & \\ & \ddots & A_R & \\ & & B_R & A_{R'} \end{bmatrix},$$

$$P_{LR} = \begin{bmatrix} & C_L & 0 \\ & \ddots & \ddots \\ C_L & 0 & \\ 0 & & \end{bmatrix}, \quad P_{RL} = \begin{bmatrix} & C_R & 0 \\ & \ddots & \ddots \\ C_R & 0 & \\ 0 & & \end{bmatrix},$$

$$A_L = \begin{bmatrix} 2r - (-M+1)h & -r & & & \\ -r & 2r - (-M+2)h & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 2r - (-2)h & -r \\ & & & -r & \frac{3}{2}r - (-h) \end{bmatrix},$$

$$A_{L'} = \begin{bmatrix} 2r - (-M+1)h & -r & & & \\ -r & 2r - (-M+2)h & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 2r - (-h) & \end{bmatrix},$$

$$C_L = \begin{bmatrix} & 0 \\ & \ddots \\ -\frac{r}{2} & \end{bmatrix} \quad ; \quad B_L = \begin{bmatrix} (-M+1)h & & \\ & \ddots & \\ & & -h \end{bmatrix},$$

$$\begin{aligned}
A_R &= \begin{bmatrix} \frac{3}{2}r + h & -r & & & \\ -r & 2r + 2h & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 2r + (M-2)h & -r \\ & & & -r & 2r + (M-1)h \end{bmatrix}, \\
A_{R'} &= \begin{bmatrix} 2r + h & -r & & & \\ -r & 2r + 2h & -r & & \\ & \ddots & \ddots & \ddots & \\ & & -r & 2r + (M-2)h & -r \\ & & & -r & 2r + (M-1)h \end{bmatrix}, \\
C_R &= \begin{bmatrix} & -\frac{r}{2} \\ \ddots & \\ 0 & \end{bmatrix} \quad ; \quad B_R = \begin{bmatrix} -h & & \\ & \ddots & \\ & & (-M+1)h \end{bmatrix}, \\
\text{where } r &= \frac{\tau}{h^2}.
\end{aligned}$$

The indices L and R denote the portions of the matrices associated with the left and right subregions, respectively. For instance, P_{LR} represents the portions associated with coupling the nodal values of subproblem 1 (left side) to the nodal values of subproblem 2 (right side).

Having solved the above linear system and obtained the interior solution of the subproblems, the interface boundary solution can be found by using (6).

4 Convergence

In order to show the convergence of the proposed method, we use Lax's equivalence theorem [21]. To do so, we need to seek the stability condition and consistency of this method. We first check the stability.

Theorem 1. The finite difference equations (2)–(4) are unconditionally stable.

Proof. Substituting $u_{p,q} = e^{\beta p h} \xi^q$ into the difference equation (2), we find that

$$ph(\xi - 1) = r\xi(2\cos(\beta h) - 2) = r\xi(-4r\sin^2(\frac{\beta h}{2})).$$

Since $ph < 1$, we can write

$$\xi(4r\sin^2(\frac{\beta h}{2}) + 1) > 1.$$

Using the stability condition, $|\xi| \leq 1$, we have

$$\frac{1}{4r \sin^2(\frac{\beta h}{2}) + 1} < |\xi| \leq 1.$$

Clearly $0 \leq \xi \leq 1$ for all $r > 0$ and all β . Therefore, the equations are unconditionally stable. For equation (3), the same result can be obtained. $\square$

The consistency of the method is considered in the following theorem.

Theorem 2. The finite difference equations (2)–(4) are consistent and they have the local truncation error of $O(hk) + O(h^2)$.

Proof. Let $F_{ij}(u) = 0$ represent the difference equation approximating the partial differential equation (1) at the (i, j) th mesh point, with exact solution u , and let U be the exact solution of equation (1). Then we have the local truncation error T_{ij} at the mesh point $(ih, j\tau)$ as follows:

$$\begin{aligned} T_{ij} &= F_{ij}(U) \\ &= ih \frac{U_{i,j+1} - U_{i,j}}{k} - \frac{U_{i-1,j+1} - 2U_{i,j+1} + U_{i+1,j+1}}{h^2} - f(i, j+1). \end{aligned}$$

By Taylor's expansion, we have

$$\begin{aligned} T_{ij} &= ih \left(\frac{\partial U}{\partial t} \right)_{i,j+1} - \left(\frac{\partial^2 U}{\partial x^2} \right)_{i,j+1} - f(i, j+1) \\ &\quad + ih \frac{hk}{2} \left(\frac{\partial^2 U}{\partial t^2} + \frac{k}{3} \frac{\partial^3 U}{\partial t^3} + \dots \right) - \frac{h^2}{12} \left(\frac{\partial^4 U}{\partial x^4} - \frac{h^2}{20} \frac{\partial^6 U}{\partial x^6} + \dots \right). \end{aligned}$$

Hence

$$T_{ij} = O(hk) + O(h^2).$$

$\square$

4.1 Higher order finite difference schemes

The accuracy of the solution can be improved by employing higher order finite difference approximations. To do so, we first apply the previous finite difference formulation to the nodes located on the lines $x = ih$ for $i = M - 1$ and $i = -M + 1$ as follows:

$$ih \frac{u_{i,j+1} - u_{i,j}}{\tau} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + f_{i,j}.$$

Now there is sufficient number of nodal values to use the following scheme for the other nodes:

$$\begin{aligned}
& -ih \frac{u_{i,j+2} - 4u_{i,j+1} + 3u_{i,j}}{2\tau} \\
& = \frac{-u_{i+2,j} + 16u_{i-1,j} - 30u_{i,j} + 16u_{i,j-1} - u_{i,j-2}}{12h^2} + f_{i,j}. \quad (9)
\end{aligned}$$

Applying this formula improves the accuracy of the solution, taking into account that the order of the scheme (9) is $O(hk^2) + O(h^4)$. As will be observed, the numerical errors are remarkably declined in the case of using this formula.

5 Numerical results

We now present some numerical results in order to show the effectiveness of the proposed method.

Consider equation (1) with $u_0(x) = 0$, $u_1(x) = 0$, $g_{-1}(t) = 0$, $g_1(t) = 0$, and

$$f(x, t) = \begin{cases} 2x(x^2 - 1)t[(t - 1)^2 - 4x^2 + t(t - 1)] - 2t^2[(t - 1)^2 - 24x^2 + 4], & x \geq 0, \quad t \in [0, 1], \\ 2x(x^2 - 1)(t - 1)(2t^2 - t - 4x^2) - 2(t - 1)^2(t^2 - 24x^2 + 4), & x \leq 0, \quad t \in [0, 1], \end{cases}$$

The exact solution of the above problem is given by

$$u(x, t) = \begin{cases} (x^2 - 1)t^2[(t - 1)^2 - 4x^2], & x \geq 0, \quad t \in [0, 1], \\ (x^2 - 1)(t^2 - t - 4x^2)(t - 1)^2, & x \leq 0, \quad t \in [0, 1]. \end{cases}$$

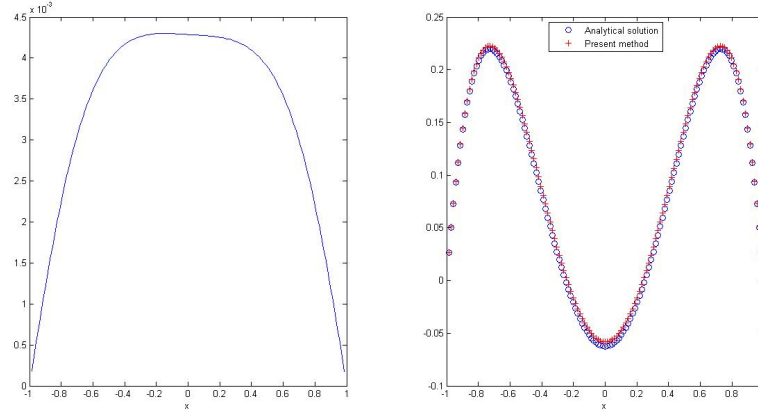
To measure the accuracy of the results, the maximum error (Max error) and the root-mean-square error (RMSE) are employed, that is,

$$Max \text{ error} = \max_{j=1}^N |\hat{u}_j - u_j|, \quad RMSE = \sqrt{\frac{1}{N} \sum_{j=1}^N (\hat{u}_j - u_j)^2},$$

where N is the number of nodes and u_j and $\hat{u}_j$ denote the exact and approximate solutions at the j th node. The numerical solution of the above equation was obtained in two cases: Scheme (I), the FDM proposed in (2) and (3), and scheme (II), the FDM suggested in (9). The numerical errors are listed in Tables 1 and 2, respectively, for the cases (I) and (II). The agreement of the approximate and the exact solutions together with the configuration of the error functions is shown in Figures 2 and 3, respectively, for the cases (I) and (II). The numerical errors demonstrate reasonable accuracy in both cases, especially in the case of using the scheme (II).

Table 1: Numerical errors produced by the new method using scheme (I).

M	N	Max error	RMSE	M	N	Max error	RMSE
10	10	3.37E-2	8.40E-3	30	15	9.81E-3	4.80E-3
20	10	1.43E-2	6.89E-3	40	20	7.40E-3	3.70E-3
30	10	1.51E-2	7.68E-3	50	25	6.01E-3	2.95E-3
40	10	1.54E-2	8.02E-3	60	30	4.91E-3	2.50E-3
50	10	1.56E-2	8.20E-3	70	35	4.27E-3	2.22E-3

Figure 2: Comparing the exact solution and the numerical solution by scheme (I) with $M = 70$ and $N = 35$ and configuration of the error function.

In order to show the efficiency of the new method, we compare the numerical results with those presented in [13]. The errors obtained by scheme (II) and those presented in [13] are listed in Table 3 using the same mesh size and time steps. It is observed that although in some cases our method produces more accurate solutions, overall, the difference between the errors is not significant. However, the new method has some advantages over the previous ones, that is, iterative methods. In the previous works, an initial solution should be carefully selected, otherwise the convergence of the iterative method is not guaranteed. In addition, as seen in Table 3, in the previous work, the solution is obtained at the expense of performing some iterations, which considerably increase the computational costs. Moreover, a parameter is needed in the iterative method (see [13]). Choosing this parameter is also important in order to achieve convergence.

It should be noted, in the suggested method, a global system of equations is solved rather than the local ones. Although, apparently, this may increase the computational costs, the resulting system has a sparse coefficient matrix,

Table 2: Numerical errors produced by the new method using scheme (II).

M	N	Max error	RMSE	M	N	Max error	RMSE
10	10	3.10E-2	8.53E-3	30	15	3.85E-3	9.12E-4
20	10	7.48E-3	2.01E-3	40	20	2.36E-3	5.18E-4
30	10	3.50E-3	1.18E-3	50	25	1.62E-3	3.33E-4
40	10	2.86E-3	1.10E-3	60	30	1.18E-3	2.32E-4
50	10	2.61E-3	1.00E-3	70	35	8.75E-4	1.71E-4

which can be treated by the efficient solvers. Moreover, the iterations required in the case of using local matrices, are avoided.

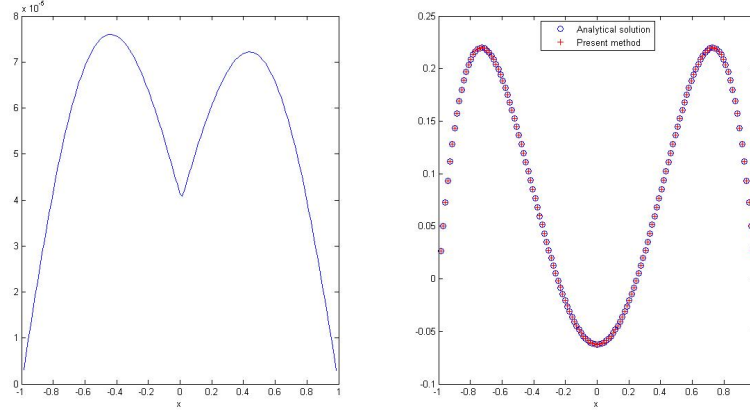


Figure 3: Comparing the exact solution and numerical solution by scheme (II) with $M = 70$, $N = 35$ and configuration of the error function.

6 Conclusions

A nonoverlapping DDM was applied to the FBHE in a one-dimensional case. The work was based on partitioning the spatial domain and considering each part as an independent forward or a backward subproblem. A forward and a backward finite difference schemes were employed for local problems followed by assembling the equations in a global one. The under determined system of equations, arising from the lack of the interface boundary conditions, was treated by a physical matching condition and a square system was obtained.

Table 3: Comparing the error values (Max error) of scheme (II) with the previous results produced by the iterative method with K denoting the number of iterations.

M	N	K	Iterative method	New method
4	4	24	1.36E-2	7.56E-2
8	16	38	3.68E-3	1.02E-2
16	64	63	9.22E-3	1.10E-3
32	256	100	2.42E-4	2.56E-4

Dealing with a single global system of equations, while avoiding iterations used in the previous attempts, led to increasing the computational efficiency and keeping a reasonable accuracy for the solution. The proposed method performed well in the one-dimensional case. The application of the present method to the case of two-dimensional is also under consideration.

References

1. Aziz, A.K., French, D.A., Jensen, S. and Kollegg, R.B. *Origins, analysis, numerical analysis, and numerical approximation of a forward-backward parabolic problem*, M2AN Math. Model. Numer. Anal. 33 (1999), 895–922.
2. Aziz, A.K. and Liu, J.L. *A Galerkin method for the forward-backward heat equation*, Math. Comp. 56 (1991), 35–44.
3. Aziz, A.K. and Liu, J.L. *A weighted least squares method for the backward-forward heat equation*, SIAM J. Numer. Anal. 28 (1991), 156–167.
4. Banei, S. and Shanazari, K. *A meshfree method with a non-overlapping domain decomposition method based on TPS for solving the forward-backward heat equation in two-dimension*, Numer. Algorithms. (forthcoming).
5. Boulton, L., Marletta, M. and Rule, D. *On the stability of a forward-backward heat equation*, Integral Equations Operator Theory 73(2) (2012), 195–216.
6. Cheng, X.L. and Sun, J. *Iterative methods for the forward-backward heat equation*, J. Comput. Math. 23 (2005), 419–424.
7. Daoud, D. S. *Overlapping Schwarz waveform relaxation method for the solution of the forward-backward heat equation*, J. Comput. Appl. Math. 208(2) (2007), 380–390.
8. Dawson, C.N., Du, Q. and Dupont, T.F. *A Finite difference domain decomposition algorithm for numerical solution of the heat equation*, Math. Comp. 57 (1991), 63–71.

9. French, D.A. *Discontinuous Galerkin finite element methods for a forward-backward heat equation*, Appl. Numer. Math. 28 (1998), 37–44.
10. French, D.A. *Continuous Galerkin finite element methods for a forward-backward heat equation*, Numer. Methods Partial Differential Equations 15 (1999), 257–265.
11. Friedrichs, K.O. *Symmetric positive differential equations*, Comm. Pure Appl. Math. 11 (1958), 333–418.
12. Goldstein, J.A. and Mazumdar, T. *A heat equation in which the diffusion coefficient changes sign*, J. Math. Anal. Appl. 103 (1984), 533–564.
13. Han H.D. and Yin, D.S. *A non-overlap domain decomposition method for the forward-backward heat equation*, J. Comput. Appl. Math. 159 (2003), 35–44.
14. Kuznetsov, I.V. *Traces of entropy solutions to second order forward-backward parabolic equations*, J. Math. Sci. (N.Y.) 211(6) (2015), 767–788.
15. Kuznetsov, I.V. *Kinetic formulation of forward-backward parabolic equations*, Sib. elektron. Mat. Izv. 13 (2016), 930–949.
16. Lu, H. *Galerkin and weighted Galerkin methods for a forward-backward heat equation*, Numer. Math. 75 (1997), 338–356.
17. Lu, H. and Maubach, J. *A finite element method and variable transformations for a forward-backward heat equation*, Numer. Math. 81 (1998), 249–272.
18. Lu, H. Wen, Z-Y. *Solution of a forward-backward heat equation. Technical report, 9439*, Department of Mathematics, University of Nijmegen, The Netherlands, 1994.
19. Paronetto, F. *A remark on forward-backward parabolic equations*, Appl. Anal. 98(6) (2017), 1042–1051.
20. Paronetto, F. *Elliptic approximation of forward-backward parabolic equations*, Commun. Pur. Appl. Anal. 19(2) (2020), 1017–1036.
21. Smith, G.D. *Numerical solution of partial differential equation*, Oxford University Press, 3rd ed. 1985.
22. Sun, J. *Numerical schemes for the forward-backward heat equation*, Int. J. Comput. Math. 87(3) (2010), 552–564.
23. Sun, J. and Cheng, X.L. *Iterative methods for a forward-backward heat equation in two-dimension*, Appl. Math. J. Chinese Univ. Ser. B 25(1) (2010), 101–111.
24. Vanaja, V. and Kellogg, R.B. *Iterative methods for a forward-backward heat equation*, SIAM J. Numer. Anal. 27 (1990), 622–635.



Comparison of homotopy perturbation transform method and fractional Adams–Bashforth method for the Caputo–Prabhakar nonlinear fractional differential equations

M.H. Derakhshan* and A. Aminataei

Abstract

We study two numerical techniques based on the homotopy perturbation transform method (HPTM) and the fractional Adams–Bashforth method (FABM) for solving a class of nonlinear time-fractional differential equations involving the Caputo–Prabhakar fractional derivatives. In this manuscript, the convergence for numerical solutions obtained using HPTM and the convergence and stability for numerical solutions obtained using FABM are investigated. We compare the solutions obtained by the HPTM and the FABM for some nonlinear time-fractional differential equations. Moreover, some numerical examples are demonstrated in order to show the validity and reliability of the suggested methods.

AMS(2010): 65D15; 65D25.

Keywords: Nonlinear time-fractional differential equations; Fractional Homotopy perturbation transform method; Fractional Adams–Bashforth method; Caputo–Prabhakar fractional derivative.

*Corresponding author

Received 2 March 2020; revised 15 May 2020; accepted 31 May 2020

Mohammadhossein Derakhshan

Faculty of Mathematics, K. N. Toosi University of Technology, P. O. Box: 16765-3381, Tehran, Iran. e-mail: M.h.derakhshan.20@gmail.com

Azim Aminataei

Faculty of Mathematics, K. N. Toosi University of Technology, P. O. Box: 16765-3381, Tehran, Iran. e-mail: ataei@kntu.ac.ir

1 Introduction

In recent years, fractional calculus, which is a generalization of the integer-order calculus or integration of integer order to arbitrary order (real or complex order), plays an important, fundamental and significant role in mathematical modeling and solving the models obtained in different fields of mathematics, sciences, physics, finance and artificial intelligence, and engineering; see [11, 39, 3, 10, 31, 38, 2]. Nowadays, fractional differential equations have been applied extensively as an important and crucial tool to describe many different types of complex mechanical, physical behaviors, sciences, engineering, anomalous diffusion, vibration, control, viscoelasticity, electrochemistry, and others, since most of the models in different research areas and engineering applications are nonlinear and obtaining their solutions has an important role.

In this manuscript, we consider a numerical method based on the homotopy perturbation transform method (HPTM), which is a combination of homotopy analysis method and Laplace transform scheme [29, 40] for solving the following nonlinear time-fractional partial differential equation (FPDE) with $m - 1 < \mu \leq m$:

$$\begin{aligned} {}^C\mathbb{D}_{\rho,\mu,\omega,0+}^\gamma u(t) + \mathbf{A}u(t) + \mathbf{N}u(t) &= h(t), \\ u^{(k)}(0) &= u_0^k, \quad k = 0, 1, 2, \dots, m-1, \quad m \in \mathbb{N}, \end{aligned} \quad (1)$$

where $\mathbf{A}u(t)$ and $\mathbf{N}u(t)$ are linear and nonlinear operators applied to $u(t)$, respectively, and $h(t)$ is a continuous function. Also ${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^\gamma$ denotes the Caputo–Prabhakar fractional derivative defined by Garra et al. (see [11])

$$\begin{aligned} {}^C\mathbb{D}_{\rho,\mu,\omega,0+}^\gamma u(t) &= \int_0^t (t-\tau)^{-\mu} E_{\rho,1-\mu}^{-\gamma}(\omega(t-\tau)^\rho) \dot{u}(\tau) d\tau \\ &= \mathbf{E}_{\rho,1-\mu,\omega,0+}^{-\gamma} \frac{d}{dt} u(t), \quad 0 < \mu \leq 1, \end{aligned} \quad (2)$$

where $\mathbf{E}$ is the Prabhakar. It is obtained by modifying the Riemann–Liouville integral operator [31] by extending its kernel with a nonsingular three-parameter Mittag–Leffler function (ML), which is known as the Prabhakar function and its integral is (see [11])

$$(\mathbf{E}_{\rho,\mu,\omega,0+}^\gamma u)(t) = \int_0^t (t-\tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t-\tau)^\rho) u(\tau) d\tau,$$

where $E_{\rho,\mu}^\gamma$ is the Prabhakar function and its function is (see [11])

$$E_{\rho,\mu}^\gamma(t) = \sum_{n=0}^{\infty} \frac{(\gamma)_n}{n! \Gamma(\rho n + \mu)} t^n, \quad \Re(\rho), \Re(\mu) > 0, \gamma > 0, \quad t \in \mathbb{C}, \quad (3)$$

and $(\gamma)_n$ is the Pochhammer symbol, which is formulated as [15]

$$(\gamma)_0 = 1, \quad (\gamma)_n = \gamma(\gamma + 1) \dots (\gamma + n - 1), \quad n \in \mathbb{N}.$$

Most of our interest in studying ML functions is related to their importance in fractional calculus which arise in modeling a complex susceptibility in the response of disordered materials and heterogeneous systems [26], the response in anomalous dielectrics of Havriliak–Negami type [12, 37], in fractional viscoelasticity [14], in the discussion of stochastic processes [9], in probability theory [16], in the description of dynamical models of spherical stellar systems [1], and fractional or integral differential equations [25, 6, 5, 7, 8, 4, 18]. In this manuscript, we consider the following nonlinear time-fractional order differential equations with initial conditions:

$$\begin{aligned} {}^C\mathbb{D}_{\rho, \mu, \omega, 0^+}^\gamma u(t) &= f(t), \\ u^{(k)}(0) &= u_0^k, \quad k = 0, 1, 2, \dots, m-1, \quad m \in \mathbb{N}, \end{aligned} \quad (4)$$

and in order to solve the nonlinear FPDE (4) with $m-1 < \mu \leq m$, we apply a numerical technique based on the fractional Adams–Bashforth method (FABM) to obtain the numerical solution of (4). The present work aim is to compare the numerical solutions that are obtained by applying the FABM and HPTM to solve nonlinear time-fractional order differential equations. Several numerical-analytical methods have been applied to solve linear and nonlinear differential equations of fractional derivatives in order to obtain exact solutions and numerical solutions; for instance, the Hermite collocation method [30], optimal homotopy asymptotic method [17], q -homotopy analysis transform technique [36], homotopy analysis Laplace transform method [23, 24], the homotopy analysis Sumudu transform method for solving the linear and nonlinear Fokker-Planck equations [33], and other methods [19, 20, 28, 27, 34, 21] were suggested.

This paper is organized as follows. Some necessary definitions and mathematical preliminaries of the fractional calculus are introduced in Section 2. In Section 3, we introduce a numerical method based on the HPTM. In Section 4, the FABM for time-fractional linear and nonlinear differential equations is introduced to obtain the numerical solution. In Section 5, the proposed methods are applied to some nonlinear time-fractional differential equations to verify the validity and applicability of the proposed methods. Finally, a brief conclusion is added at the end of the manuscript.

2 Preliminaries

In this section, we study some important and basic properties of fractional calculus theory such as Laplace transform, definitions, and lemmas, which

are applied in the next sections.

Definition 1. [see [22, 31]] Let $0 < \alpha \leq 1$, let $f \in L^1[a, b]$, and let $0 < t < b \leq \infty$. Then the left-sided and the right-sided Riemann–Liouville fractional integrals and derivatives of order α are defined as follows,

$$\begin{aligned} I_{a+}^{\alpha} f(t) &= \frac{1}{\Gamma(\alpha)} \int_a^t f(\tau)(t-\tau)^{\alpha-1} d\tau, \\ I_{b-}^{\alpha} f(t) &= \frac{1}{\Gamma(\alpha)} \int_t^b f(\tau)(\tau-t)^{\alpha-1} d\tau, \\ D_{a+}^{\alpha} f(t) &= \frac{1}{\Gamma(\alpha)} \frac{d}{dt} \int_a^t f(\tau)(t-\tau)^{-\alpha} d\tau, \\ D_{b-}^{\alpha} f(t) &= -\frac{1}{\Gamma(\alpha)} \frac{d}{dt} \int_t^b f(\tau)(\tau-t)^{-\alpha} d\tau, \end{aligned}$$

where $\Gamma(\cdot)$ is the Gamma function. Also, the left-sided and the right-sided Caputo fractional derivatives of order α are given by

$$\begin{aligned} {}^C D_{a+}^{\alpha} f(t) &= I_{a+}^{1-\alpha} \frac{d}{dt} f(t) = \frac{1}{\Gamma(1-\alpha)} \int_a^t (t-\tau)^{-\alpha} \frac{d}{d\tau} f(\tau) d\tau, \\ {}^C D_{b-}^{\alpha} f(t) &= -I_{b-}^{1-\alpha} \frac{d}{dt} f(t) = -\frac{1}{\Gamma(1-\alpha)} \int_t^b (\tau-t)^{-\alpha} \frac{d}{d\tau} f(\tau) d\tau. \end{aligned}$$

Lemma 1. [4] Let $\rho, \mu, \omega, \gamma \in \mathbb{C}$, let $\Re(\mu) > 0$, and let $\Re(\rho) > 0$. Then the Laplace transform of (2) for $m-1 < \mu \leq m$ is given,

$$\mathcal{L}\left({}^C \mathbb{D}_{\rho, \mu, \omega, 0+}^{\gamma} u(t); s\right) = s^{\mu}(1-\omega s^{-\rho})^{\gamma} U(s) - \sum_{k=0}^{m-1} s^{\mu-k-1} (1-\omega s^{-\rho})^{\gamma} u^{(k)}(0^{+}), \quad (5)$$

where $U(s) = \int_0^{\infty} e^{-st} u(t) dt$. Also, the Laplace transformation of the Prabhakar function (3) is denoted by (see [11, 15])

$$\mathcal{L}\left(t^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega t^{\rho}); s\right) = s^{-\mu} (1-\omega s^{-\rho})^{-\gamma}.$$

Lemma 2. [22] Let $\rho, \mu, \omega, \gamma \in \mathbb{C}$ and let $\Re(\mu) > 0, \Re(\rho) > 0$. Then,

$$\int_0^t (t-y)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t-y)^{\rho}) y^{\nu-1} dy = \Gamma(\nu) t^{\mu+\nu-1} E_{\rho, \mu+\nu}^{\gamma}(\omega t^{\rho}). \quad (6)$$

Lemma 3. [22] Let $\rho, \mu, \nu, \omega, \sigma, \gamma \in \mathbb{C}$ and let $\Re(\mu) > 0, \Re(\rho) > 0, \Re(\nu) > 0$. Then the following relation is hold for any summable function $u \in L(a, b)$,

$$\mathbf{E}_{\rho,\mu,\omega,0+}^{\gamma} \mathbf{E}_{\rho,\nu,\omega,0+}^{\sigma} u = \mathbf{E}_{\rho,\mu+\nu,\omega,0+}^{\gamma+\sigma} u. \quad (7)$$

Also, by substituting $\sigma = -\gamma$ in (7), the following relation is obtained,

$$\mathbf{E}_{\rho,\mu,\omega,0+}^{\gamma} \mathbf{E}_{\rho,\nu,\omega,0+}^{-\gamma} u = I_{0+}^{\mu+\nu} u. \quad (8)$$

3 HPTM to solve the nonlinear fractional order differential equations

In this section, we obtain a combined numerical method based on the homotopy analysis method and the Laplace transform method for solving the nonlinear FPDE in Caputo–Prabhakar fractional derivatives sense. Using formula (5) for $m = 1$ and applying the Laplace transform on both sides of equation (1), we get

$$\begin{aligned} \mathcal{L}\left({}^C\mathbb{D}_{\rho,\mu,\omega,0+}^{\gamma} u(t) + \mathbf{A}u(t) + \mathbf{N}u(t) - h(t); s\right) &= 0, \\ s^{\mu}(1 - \omega s^{-\rho})^{\gamma} U(s) - s^{\mu-1}(1 - \omega s^{-\rho})^{\gamma} u(0^+) + \mathcal{L}\left(\mathbf{A}u(t) + \mathbf{N}u(t); s\right) &= H(s), \\ U(s) = s^{-\mu}(1 - \omega s^{-\rho})^{-\gamma} H(s) + \frac{1}{s} u(0) - (s^{-\mu}(1 - \omega s^{-\rho})^{-\gamma}) & \\ \times \left[\mathcal{L}\left(\mathbf{A}u(t) + \mathbf{N}u(t); s\right)\right]. & \end{aligned} \quad (9)$$

By applying the Laplace inverse transform on (9), we obtain

$$\begin{aligned} u(t) = \mathcal{L}^{-1}\left[s^{-\mu}(1 - \omega s^{-\rho})^{-\gamma} H(s) + \frac{1}{s} u(0); t\right] - \mathcal{L}^{-1}\left[(s^{-\mu}(1 - \omega s^{-\rho})^{-\gamma}) \right. & \\ \left. \times \left[\mathcal{L}\left(\mathbf{A}u(t) + \mathbf{N}u(t); s\right)\right]; t\right]. & \end{aligned} \quad (10)$$

To obtain the solution of (10), we use the HPTM. Suppose that the solution of (1) is $u(t)$, which can be expressed as the following infinite series:

$$u(t) = \sum_{n=0}^{\infty} p^n u_n(t), \quad (11)$$

where $u_n(t)$ for $n = 0, 1, 2, \dots$ are known functions. Also, the nonlinear term of (1) is $\mathbf{N}u(t)$, which can be presented as the following infinite series:

$$\mathbf{N}u(t) = \sum_{n=0}^{\infty} p^n \mathbf{H}_n(u(t)), \quad (12)$$

where $\mathbf{H}_n(u(t)) = \left\{ \frac{1}{n!} \frac{\partial^n}{\partial p^n} \left[N \left(\sum_{i=0}^{\infty} p^i u_i \right) \right]_{p=0} \right\}$ are polynomials of the Adomian polynomials type; see [13]. Now, substituting (11) and (12) into (10), we conclude that

$$\begin{aligned} \sum_{n=0}^{\infty} p^n u_n(t) = & H(t) - \mathcal{L}^{-1} \left[(s^{-\mu} (1 - \omega s^{-\rho})^{-\gamma}) \right. \\ & \times \left. \left[\mathcal{L} \left(\mathbf{A} \sum_{n=0}^{\infty} p^n u_n(t) + \mathbf{N} \sum_{n=0}^{\infty} p^n u_n(t); s \right) \right]; t \right], \end{aligned} \quad (13)$$

where $H(t) = \mathcal{L}^{-1} \left[s^{-\mu} (1 - \omega s^{-\rho})^{-\gamma} H(s) + \frac{1}{s} u(0); t \right]$. By equaling the coefficients on powers of p on the both sides of relation (13), we obtain the series solution as follows:

$$\begin{aligned} p^0 : u_0(t) &= H(t), \\ p^1 : u_1(t) &= -\mathcal{L}^{-1} \left[(s^{-\mu} (1 - \omega s^{-\rho})^{-\gamma}) \times \left[\mathcal{L} \left(\mathbf{A} u_0(t) + \mathbf{H}_0(u(t)); s \right) \right]; t \right], \\ p^2 : u_2(t) &= -\mathcal{L}^{-1} \left[(s^{-\mu} (1 - \omega s^{-\rho})^{-\gamma}) \times \left[\mathcal{L} \left(\mathbf{A} u_1(t) + \mathbf{H}_1(u(t)); s \right) \right]; t \right], \\ &\vdots \\ p^{n+1} : u_{n+1}(t) &= -\mathcal{L}^{-1} \left[(s^{-\mu} (1 - \omega s^{-\rho})^{-\gamma}) \times \left[\mathcal{L} \left(\mathbf{A} u_n(t) + \mathbf{H}_n(u(t)); s \right) \right]; t \right]. \end{aligned}$$

So, the solution of (1) by using the HPTM is obtained as follows:

$$u(t) = \lim_{p \rightarrow 1} \lim_{n \rightarrow \infty} \sum_{k=0}^n p^k u_k(t) = \lim_{n \rightarrow \infty} \sum_{k=0}^n u_k(t).$$

Theorem 1. Let $S_n = \sum_{k=0}^n u_k(t)$ and let $u(t) = \sum_{k=0}^{\infty} u_k(t)$ such that $u(t), u_k(t) \in C([0, 1])$ and for any $n = 1, 2, 3, \dots$, $\|u_{n+1}(t)\| \leq \lambda \|u_n(t)\|$, $0 < \lambda < 1$. Then the solution of (1) using the HPTM is converges.

Proof. To prove convergence, we need to show that the series S_n is a Cauchy sequence. We have

$$\begin{aligned} \|S_{n+1} - S_n\| &= \left\| \sum_{k=0}^{n+1} u_k(t) - \sum_{k=0}^n u_k(t) \right\| \\ &= \|u_{n+1}(t)\| \leq \lambda \|u_n(t)\| \leq \lambda^2 \|u_{n-1}(t)\| \leq \dots \\ &\leq \lambda^{n+1} \|u_0(t)\|. \end{aligned}$$

Then for any $m, n \in \mathbb{N}, n > m$, we obtain

$$\begin{aligned} \|S_n - S_m\| &= \|(S_n - S_{n-1}) + (S_{n-1} - S_{n-2}) + (S_{n-2} - S_{n-3}) + \dots + (S_{m+1} - S_m)\| \end{aligned}$$

$$\begin{aligned}
&\leq \|S_n - S_{n-1}\| + \|S_{n-1} - S_{n-2}\| + \|S_{n-2} - S_{n-3}\| + \|S_{m+1} - S_m\| \\
&\leq \lambda^n \|u_0(t)\| + \lambda^{n-1} \|u_0(t)\| + \dots + \lambda^{m+1} \|u_0(t)\| \\
&\leq \frac{1 - \lambda^{n-m}}{1 - \lambda} \lambda^{m+1} \|u_0(t)\| \leq \frac{\lambda^{m+1}}{1 - \lambda} \|u_0(t)\|, \quad 1 - \lambda^{n-m} < 1.
\end{aligned}$$

As $m, n \rightarrow \infty$ then $\|S_n - S_m\| \rightarrow 0$, since $u_0(t)$ is bounded and $0 < \lambda < 1$. So the series $S_n = \sum_{k=0}^n u_k(t)$ is a Cauchy sequence in the Banach space and therefore converges. $\square$

4 The Adams–Bashforth scheme to numerically solve nonlinear fractional order differential equations

In this section, we consider the fractional differential equation (4) with initial conditions and apply the Adams–Bashforth scheme in order to numerically solve this equation. Using relation (8) in Lemma 3 and applying the integral transform operator $\mathbf{E}_{\rho, \mu, \omega, 0+}^\gamma$ to both sides of the fractional differential equation (4), the solutions of this equation are obtained as follows:

$$\begin{aligned}
&\mathbf{E}_{\rho, \mu, \omega, 0+}^\gamma ({}^C \mathbb{D}_{\rho, \mu, \omega, 0+}^\gamma u(t)) = \mathbf{E}_{\rho, \mu, \omega, 0+}^\gamma f(t), \\
&\mathbf{E}_{\rho, \mu, \omega, 0+}^\gamma \left(\mathbf{E}_{\rho, 1-\mu, \omega, 0+}^{-\gamma} \frac{d}{dt} u(t) \right) = \mathbf{E}_{\rho, \mu, \omega, 0+}^\gamma f(t), \\
&u(t) - u(0) = \int_0^t (t - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t - \tau)^\rho) f(\tau) d\tau, \\
&u(t) = u(0) + \int_0^t (t - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t - \tau)^\rho) f(\tau) d\tau. \tag{14}
\end{aligned}$$

Now, we first put $t = t_{n+1}$, $n = 0, 1, 2, 3, \dots$ and then $t = t_n$, $n = 0, 1, 2, 3, \dots$ in relation (14). Now we obtain

$$u(t_{n+1}) = u(0) + \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) f(\tau) d\tau, \tag{15}$$

$$u(t_n) = u(0) + \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_n - \tau)^\rho) f(\tau) d\tau. \tag{16}$$

By subtracting two relations (15) and (16), we obtain

$$\begin{aligned}
u(t_{n+1}) - u(t_n) &= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) f(\tau) d\tau \\
&\quad - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_n - \tau)^\rho) f(\tau) d\tau. \tag{17}
\end{aligned}$$

To solve (17), we approximate the function $f(\tau)$ by the Lagrange interpolation as follows:

$$\begin{aligned} P(\tau) &\simeq \frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}), \\ P(\tau) &\simeq \frac{\tau - t_{n-1}}{h} f(t_n) - \frac{\tau - t_n}{h} f(t_{n-1}), \end{aligned} \quad (18)$$

$$t_j = jh, j = 0, 1, 2, 3, \dots, h = t_n - t_{n-1}.$$

Thus, by substituting (18) in (17), we obtain

$$\begin{aligned} u(t_{n+1}) &= u(t_n) \quad (19) \\ &+ \left[\frac{f(t_n)}{h} \times \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_{n-1}) d\tau \right. \\ &\quad \left. - \frac{f(t_{n-1})}{h} \times \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_n) d\tau \right] \\ &- \left[\frac{f(t_n)}{h} \times \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_{n-1}) d\tau \right. \\ &\quad \left. - \frac{f(t_{n-1})}{h} \times \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_n) d\tau \right], \\ &= u(t_n) + \left[\frac{f(t_n)}{h} A_{\mu,1} - \frac{f(t_{n-1})}{h} A_{\mu,2} \right] - \left[\frac{f(t_n)}{h} A_{\mu,3} - \frac{f(t_{n-1})}{h} A_{\mu,4} \right], \end{aligned} \quad (20)$$

where

$$A_{\mu,1} = \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_{n-1}) d\tau, \quad (21)$$

$$A_{\mu,2} = \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_n) d\tau, \quad (22)$$

$$A_{\mu,3} = \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_{n-1}) d\tau, \quad (23)$$

$$A_{\mu,4} = \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_n) d\tau. \quad (24)$$

To calculate relations (21)–(24), first we calculate the integral (21) as follows:

$$\begin{aligned}
A_{\mu,1} &= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_{n-1}) d\tau \\
&= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \tau d\tau \\
&\quad - t_{n-1} \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) d\tau. \tag{25}
\end{aligned}$$

By applying (6) in relation (25), we obtain

$$\begin{aligned}
A_{\mu,1} &= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_{n-1}) d\tau \\
&= \Gamma(2)(t_{n+1})^{\mu+1} E_{\rho,\mu+2}^\gamma(\omega(t_{n+1})^\rho) - t_{n-1} \times \Gamma(1)(t_{n+1})^\mu E_{\rho,\mu+1}^\gamma(\omega(t_{n+1})^\rho) \\
&= (n+1)^{\mu+1} h^{\mu+1} E_{\rho,\mu+2}^\gamma(\omega((n+1)h)^\rho) \\
&\quad - (n-1)h \times \left[(n+1)^\mu h^\mu E_{\rho,\mu+1}^\gamma(\omega((n+1)h)^\rho) \right] \\
&= (n+1)^\mu h^{\mu+1} \times \left[(n+1) E_{\rho,\mu+2}^\gamma(\omega((n+1)h)^\rho) \right. \\
&\quad \left. - (n-1) E_{\rho,\mu+1}^\gamma(\omega((n+1)h)^\rho) \right]. \tag{26}
\end{aligned}$$

Similarly, for the residual terms, we have

$$\begin{aligned}
A_{\mu,2} &= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho)(\tau - t_n) d\tau \\
&= (n+1)^\mu h^{\mu+1} \\
&\quad \times \left[(n+1) E_{\rho,\mu+2}^\gamma(\omega((n+1)h)^\rho) - n E_{\rho,\mu+1}^\gamma(\omega((n+1)h)^\rho) \right], \tag{27}
\end{aligned}$$

$$\begin{aligned}
A_{\mu,3} &= \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_{n-1}) d\tau \\
&= n^\mu h^{\mu+1} \times \left[n E_{\rho,\mu+2}^\gamma(\omega(nh)^\rho) - (n-1) E_{\rho,\mu+1}^\gamma(\omega(nh)^\rho) \right], \tag{28}
\end{aligned}$$

$$\begin{aligned}
A_{\mu,4} &= \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho)(\tau - t_n) d\tau \\
&= n^{\mu+1} h^{\mu+1} \times \left[E_{\rho,\mu+2}^\gamma(\omega(nh)^\rho) - E_{\rho,\mu+1}^\gamma(\omega(nh)^\rho) \right]. \tag{29}
\end{aligned}$$

Substituting relations (26)–(29) in (19), we obtain the approximation scheme to solve numerically (4).

Theorem 2. [Convergence] Let $u(t)$ be a solution of (4) and let $f(t)$ be continuous and bounded that is introduced in (4). Then the numerical solution of $u(t)$ is given by

$$\begin{aligned}
u(t_{n+1}) = & u(t_n) + \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) \right] d\tau \\
& - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) \right] d\tau \\
& + \mathbf{R}(t, \mu, n),
\end{aligned}$$

where $\mathbf{R}(t, \mu, n)$ is the error function and $\|\mathbf{R}(t, \mu, n)\|_\infty < M$.

Proof. Using (17) and (18), we have

$$\begin{aligned}
u(t_{n+1}) = & u(t_n) + \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) f(\tau) d\tau \\
& - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho) f(\tau) d\tau \\
= & u(t_n) \\
& + \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) + \frac{f^{(n+1)}(\tau)}{(n+1)!} \prod_{i=0}^n (\tau - t_i) \right] d\tau \\
& - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) + \frac{f^{(n)}(\tau)}{(n)!} \prod_{i=0}^{n-1} (\tau - t_i) \right] d\tau \\
= & u(t_n) + \mathbf{L}(t, \mu, n) + \mathbf{R}(t, \mu, n),
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{L}(t, \mu, n) = & \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) \right] d\tau \\
& - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho,\mu}^\gamma(\omega(t_n - \tau)^\rho) \\
& \left[\frac{\tau - t_{n-1}}{t_n - t_{n-1}} f(t_n) + \frac{\tau - t_n}{t_{n-1} - t_n} f(t_{n-1}) \right] d\tau,
\end{aligned}$$

$$\begin{aligned}
& \mathbf{R}(t, \mu, n) \\
&= \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_{n+1} - \tau)^{\rho}) \left(\frac{f^{(n+1)}(\tau)}{(n+1)!} \prod_{i=0}^n (\tau - t_i) \right) d\tau \\
&\quad - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_n - \tau)^{\rho}) \left(\frac{f^{(n)}(\tau)}{(n)!} \prod_{i=0}^{n-1} (\tau - t_i) \right) d\tau.
\end{aligned}$$

To establish convergence, we need to show that $\|\mathbf{R}(t, \mu, n)\|_{\infty} < M$, $M \in \mathbb{N}$. For this, we have

$$\begin{aligned}
& \|\mathbf{R}(t, \mu, n)\|_{\infty} \\
&= \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_{n+1} - \tau)^{\rho}) \left(\frac{f^{(n+1)}(\tau)}{(n+1)!} \prod_{i=0}^n (\tau - t_i) \right) d\tau \right. \\
&\quad \left. - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_n - \tau)^{\rho}) \left(\frac{f^{(n)}(\tau)}{(n)!} \prod_{i=0}^{n-1} (\tau - t_i) \right) d\tau \right\|_{\infty} \\
&< \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_{n+1} - \tau)^{\rho}) \left(\frac{f^{(n+1)}(\tau)}{(n+1)!} \prod_{i=0}^n (\tau - t_i) \right) d\tau \right\|_{\infty} \\
&\quad + \left\| \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_n - \tau)^{\rho}) \left(\frac{f^{(n)}(\tau)}{(n)!} \prod_{i=0}^{n-1} (\tau - t_i) \right) d\tau \right\|_{\infty} \\
&< \max_{t \in [0, t_{n+1}]} \frac{|f^{(n+1)}(t)|}{(n+1)!} \left\| \prod_{i=0}^n (t - t_i) \right\|_{\infty} \\
&\quad \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_{n+1} - \tau)^{\rho}) d\tau \\
&\quad + \max_{t \in [0, t_{n+1}]} \frac{|f^{(n)}(t)|}{(n)!} \left\| \prod_{i=0}^{n-1} (t - t_i) \right\|_{\infty} \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_n - \tau)^{\rho}) d\tau.
\end{aligned}$$

Using Lemma 2, we obtain

$$\begin{aligned}
\|\mathbf{R}(t, \mu, n)\|_{\infty} &< \max_{t \in [0, t_{n+1}]} \frac{|f^{(n+1)}(t)|}{(n+1)!} \left\| \prod_{i=0}^n (t - t_i) \right\|_{\infty} t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho}) \\
&\quad + \max_{t \in [0, t_{n+1}]} \frac{|f^{(n)}(t)|}{(n)!} \left\| \prod_{i=0}^{n-1} (t - t_i) \right\|_{\infty} t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho})
\end{aligned}$$

$$\begin{aligned}
&< \sup_{t \in [0, t_{n+1}]} \left\{ \max_{t \in [0, t_{n+1}]} \frac{|f^{(n+1)}(t)|}{(n+1)!}, \max_{t \in [0, t_{n+1}]} \frac{|f^{(n)}(t)|}{n!} \right\} \\
&\quad \times \left[\frac{n!h^{n+1}}{4} t_{n+1}^\mu E_{\rho, \mu+1}^\gamma(\omega t_{n+1}^\rho) + \frac{(n-1)!h^n}{4} t_n^\mu E_{\rho, \mu+1}^\gamma(\omega t_n^\rho) \right] \\
&= M.
\end{aligned}$$

□

Theorem 3.[Stability condition] If

$$\frac{M_1}{4} \frac{n!h^n}{n!} \left[\frac{(n+1)t_{n+1}^\mu E_{\rho, \mu+1}^\gamma(\omega t_{n+1}^\rho)}{h} + \frac{t_n^\mu E_{\rho, \mu+1}^\gamma(\omega t_n^\rho)}{h^2} \right] \rightarrow 0$$

as $n \rightarrow \infty$ and $h \rightarrow 0$, then, the FABM is stable.

Proof. Using (17), we have

$$\begin{aligned}
\|u(t_{n+1}) - u(t_n)\|_\infty &= \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) f(\tau) d\tau \right. \\
&\quad \left. - \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_n - \tau)^\rho) f(\tau) d\tau \right\|_\infty \\
&< \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) f(\tau) d\tau \right\|_\infty \\
&\quad + \left\| \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_n - \tau)^\rho) f(\tau) d\tau \right\|_\infty.
\end{aligned}$$

Hence

$$\begin{aligned}
&\|u(t_{n+1}) - u(t_n)\|_\infty \\
&< \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \left[\sum_{k=0}^n \prod_{k=0}^n \frac{(\tau - t_k) f(t_k)}{(-1)^k h} \right] d\tau \right\|_\infty \\
&\quad + \left\| \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_n - \tau)^\rho) \left[\sum_{k=0}^{n-1} \prod_{k=0}^{n-1} \frac{(\tau - t_k) f(t_k)}{(-1)^k h} \right] d\tau \right\|_\infty \\
&< \|\mathbf{P}(t, \mu, n)\|_\infty + \|\mathbf{R}_n^\mu(t)\|_\infty,
\end{aligned}$$

where $\|\mathbf{P}(t, \mu, n)\|_\infty$ and $\|\mathbf{R}_n^\mu(t)\|_\infty$ are calculated as follows:

$$\begin{aligned}
&\|\mathbf{P}(t, \mu, n)\|_\infty \\
&= \left\| \int_0^{t_{n+1}} (t_{n+1} - \tau)^{\mu-1} E_{\rho, \mu}^\gamma(\omega(t_{n+1} - \tau)^\rho) \left[\sum_{k=0}^n \prod_{k=0}^n \frac{(\tau - t_k) f(t_k)}{(-1)^k h} \right] d\tau \right\|_\infty
\end{aligned}$$

$$\begin{aligned}
&\leq \sum_{k=0}^n \frac{\|f(t_k)\|_{\infty} t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho})}{h} \prod_{k=0}^n |t - t_k| \\
&\leq \sum_{k=0}^n \frac{\|f(t_k)\|_{\infty} t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho}) n! h^n}{4h}
\end{aligned}$$

and

$$\begin{aligned}
&\|\mathbf{R}_n^{\mu}(t)\|_{\infty} \\
&= \left\| \int_0^{t_n} (t_n - \tau)^{\mu-1} E_{\rho, \mu}^{\gamma}(\omega(t_n - \tau)^{\rho}) \left[\sum_{k=0}^{n-1} \prod_{k=0}^{n-1} \frac{(\tau - t_k) f(t_k)}{(-1)^k h} \right] d\tau \right\|_{\infty} \\
&\leq \sum_{k=0}^{n-1} \frac{\|f(t_k)\|_{\infty} t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho})}{h} \prod_{k=0}^{n-1} |t - t_k| \\
&\leq \sum_{k=0}^{n-1} \frac{\|f(t_k)\|_{\infty} t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho}) (n-1)! h^{n-1}}{4h}.
\end{aligned}$$

Then

$$\begin{aligned}
&\|u(t_{n+1}) - u(t_n)\|_{\infty} \\
&< \sum_{k=0}^n \frac{\|f(t_k)\|_{\infty} t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho}) n! h^n}{4h} \\
&\quad + \sum_{k=0}^{n-1} \frac{\|f(t_k)\|_{\infty} t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho}) (n-1)! h^{n-1}}{4h} \\
&< \frac{M_1 n! h^n}{4} \left[\frac{(n+1) t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho})}{h} + \frac{t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho})}{h^2} \right],
\end{aligned}$$

where $M_1 = \max_{t \in [0, t_{n+1}]} |f(t)|$. Therefore,

$$\frac{M_1 n! h^n}{4} \left[\frac{(n+1) t_{n+1}^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_{n+1}^{\rho})}{h} + \frac{t_n^{\mu} E_{\rho, \mu+1}^{\gamma}(\omega t_n^{\rho})}{h^2} \right] \rightarrow 0$$

as $n \rightarrow \infty$, $h \rightarrow 0$ in which $h = \frac{1}{n+1}$ is considered. $\square$

5 Illustrative examples

In this section, we compare the obtained solution with the numerical simulations of the applied methods on five test problems in order to show the accuracy and efficiency of the proposed techniques.

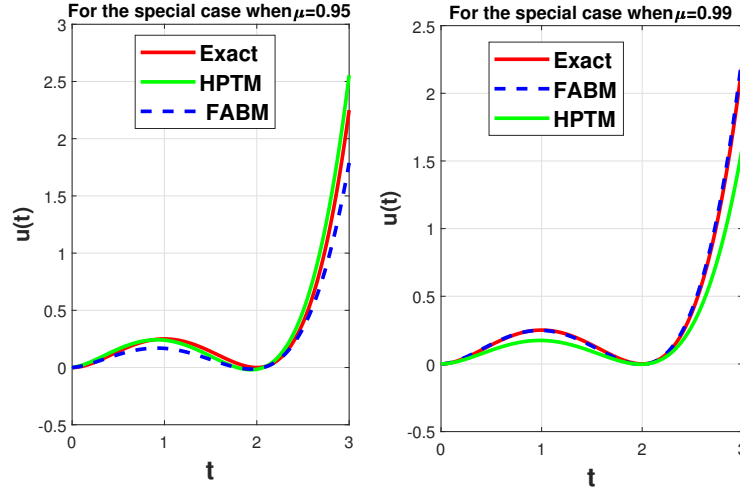


Figure 1: Comparison between numerical solution and the exact solution of Example 1 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

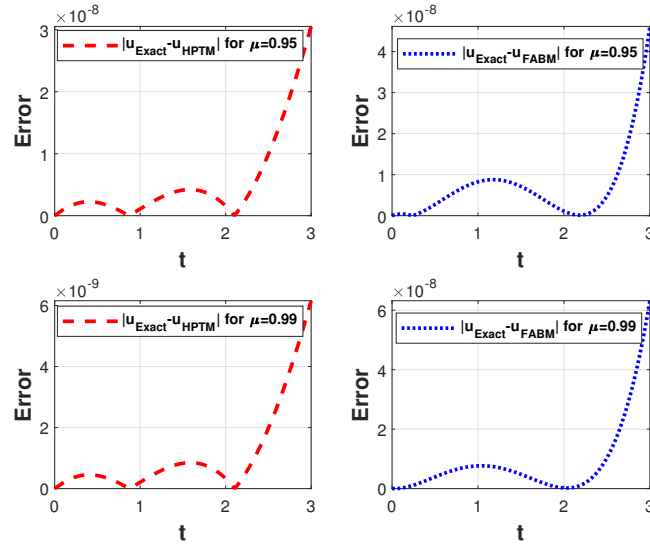


Figure 2: The value of absolute error of Example 1 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

Example 1. We consider the following nonlinear fractional order differential equation:

Table 1: The exact solution, the numerical solutions, and the absolute errors of Example 1 for different values of t when $\mu = 0.95$, $\rho = \omega = 1$.

t	$u_{Exact}(t)$	$u_{HPTM}(t)$	$u_{FABM}(t)$	$ u_{Exact}(t) - u_{HPTM}(t) $	$ u_{Exact}(t) - u_{FABM}(t) $
0	0	0	0	0	0
0.01	0.0001	0.0011	0.0008	0.001009960804482	0.000677271813138
0.02	0.0004	0.0024	0.0017	0.002022248070965	0.001297961649675
0.03	0.0009	0.0039	0.0027	0.003027289212479	0.001857141698735
0.04	0.0015	0.0056	0.0039	0.004020110685205	0.002353085479644
0.05	0.0024	0.0074	0.0052	0.004997320737974	0.002785155766582
0.06	0.0034	0.0093	0.0065	0.005956376904834	0.003153291833384
0.07	0.0046	0.0115	0.0080	0.006895275398590	0.003457792029013
0.08	0.0059	0.0137	0.0096	0.007812388169833	0.003699199718883
0.09	0.0074	0.0161	0.0113	0.008706366104405	0.003878235523084

Table 2: The exact solution, the numerical solutions, and the absolute errors of Example 1 for different values of t when $\mu = 0.99$, $\rho = \omega = 1$.

t	$u_{Exact}(t)$	$u_{HPTM}(t)$	$u_{FABM}(t)$	$ u_{Exact}(t) - u_{HPTM}(t) $	$ u_{Exact}(t) - u_{FABM}(t) $
0	0	0	0	0	0
0.01	0.0001	0.0003	0.0002	0.000201585080800	0.000111408806560
0.02	0.0004	0.0008	0.0006	0.000403349846034	0.000164732892224
0.03	0.0009	0.0015	0.0010	0.000603578912785	0.000160544488950
0.04	0.0015	0.0023	0.0016	0.000801354185865	0.000099955930106
0.05	0.0024	0.0034	0.0024	0.000996039906343	0.000015740815560
0.06	0.0034	0.0046	0.0032	0.001187154171698	0.000185164079811
0.07	0.0046	0.0059	0.0042	0.001374313737802	0.000406881133539
0.08	0.0059	0.0075	0.0052	0.001557204785139	0.000679428650402
0.09	0.0074	0.0091	0.0064	0.001735565417799	0.001001324957541

$${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^\gamma u(t) = f(t) - u^2(t),$$

subject to the initial condition

$$u(0) = 0$$

and

$$f(t) = (2t - 3t^2 - t^3) + \left[2t \left(1 - \mu + \frac{\mu t^\mu}{\Gamma(\mu + 2)} \right) - 3t^2 \left(2t \left(1 - \mu + \frac{2\mu t^\mu}{\Gamma(\mu + 3)} \right) + t^3 \left(1 - \mu + \frac{6\mu t^\mu}{\Gamma(\mu + 4)} \right) \right) \right]^2.$$

Under these conditions, the analytical solution is given by (see [41])

$$u(t) = \left[2t \left(1 - \mu + \frac{\mu t^\mu}{\Gamma(\mu + 2)} \right) - 3t^2 \left(2t \left(1 - \mu + \frac{2\mu t^\mu}{\Gamma(\mu + 3)} \right) + t^3 \left(1 - \mu + \frac{6\mu t^\mu}{\Gamma(\mu + 4)} \right) \right) \right]^2.$$

We apply the proposed schemes in this article and solve this problem. Figure 1 shows the analytical and numerical results of Example 1 obtained by applying the HPTM and the FABM together with the exact solution. Also the absolute error for $\mu = 0.95, 0.99$ with several values of t is shown in Tables 1 and 2, and for $\mu = 0.95, 0.99$ with several values of t , it is shown in Figure 2. The numerical results demonstrate that the FABM method is an efficient algorithm.

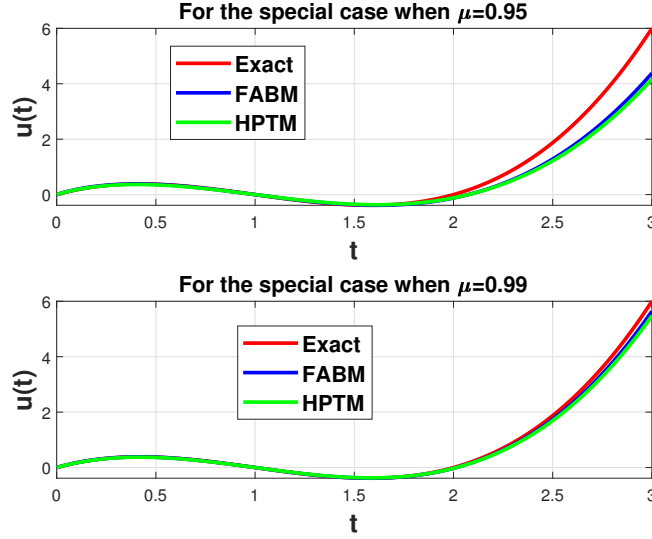


Figure 3: Comparison between numerical solution and the exact solution of Example 2 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

Example 2. We consider the following nonlinear fractional order differential equation:

$${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^{\gamma} u(t) = f(t) - u^2(t),$$

subject to the initial condition

$$u(0) = 0$$

and

$$f(t) = (2t - 3t^2 + t^3)^2 + \frac{2t^{1-\mu}}{\Gamma(2-\mu)} - \frac{6t^{2-\mu}}{\Gamma(3-\mu)} + \frac{6t^{3-\mu}}{\Gamma(4-\mu)}.$$

Under these conditions, the analytical solution is given by (see [41])

$$u(t) = 2t - 3t^2 + t^3.$$

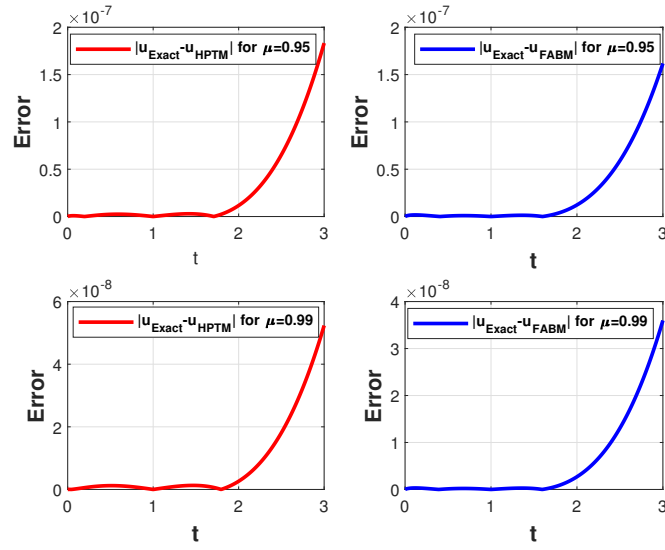


Figure 4: The value of absolute error of Example 2 when $\mu = 0.95, 0.99, \rho = \omega = 1$.

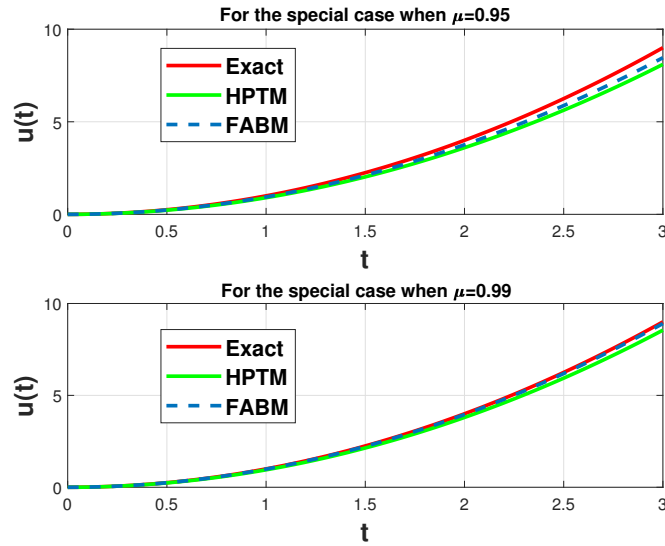


Figure 5: Comparison between numerical solution and the exact solution of Example 3 when $\mu = 0.95, 0.99, \rho = \omega = 1$.

We apply the proposed schemes in this article and solve this problem. Figure 3, shows the analytical and numerical results obtained by applying the HPTM

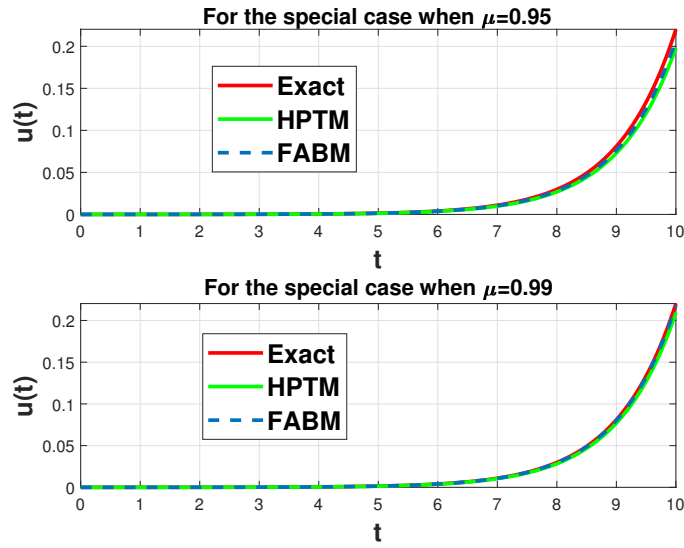


Figure 6: Comparison between numerical solution and the exact solution of Example 4 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

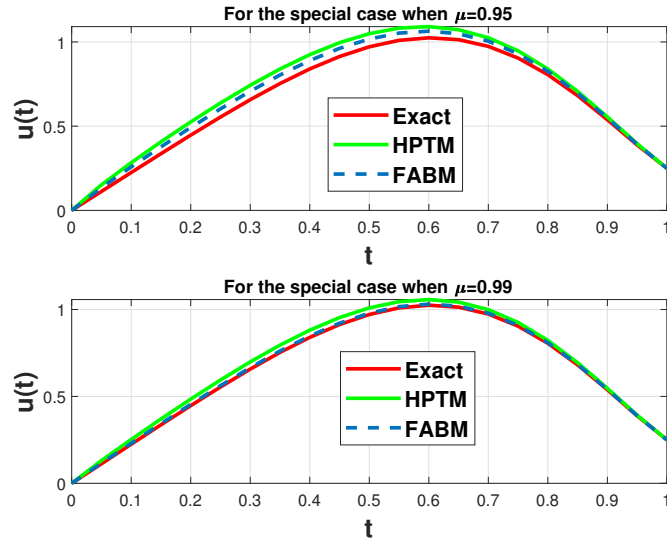


Figure 7: Comparison between numerical solution and the exact solution of Example 5 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

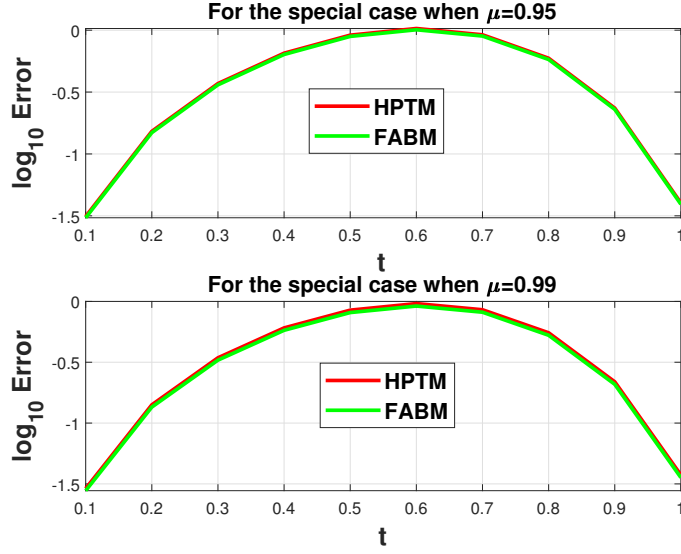


Figure 8: The value of absolute error of Example 5 when $\mu = 0.95, 0.99$, $\rho = \omega = 1$.

and the FABM together with the exact solution of Example 2. Also the absolute error for $\mu = 0.95, 0.99$ with several values of t is shown in Figure 4. The numerical results demonstrate that the FABM method is an efficient algorithm.

Example 3. We consider the following equation:

$${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^{\gamma}u(t) + u(t) = \frac{2t^{2-\mu}}{\Gamma(3-\mu)} + t^2, \quad u(0) = 0, \quad 0 < \mu \leq 1.$$

For this example, the exact solution $u(t) = t^2$ was presented in [32]. This example is solved by using numerical methods introduced in this article for different values of μ , and their diagrams are shown in Figure 5.

Example 4. We consider the following equation:

$${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^{\gamma}u(t) - \frac{1}{2}u(t) = \frac{1}{2}e^t u\left(\frac{t}{2}\right), \quad u(0) = 1, \quad 0 < \mu \leq 1, \quad 0 \leq t \leq 10.$$

The exact analytical solution of this example is $u(t) = e^t$, which was presented in [18]. This example is solved by using numerical methods introduced in this article for different values of μ , and their diagrams are shown in Figure 6.

Example 5. We consider the following nonlinear fractional initial value equation:

$${}^C\mathbb{D}_{\rho,\mu,\omega,0+}^{\gamma}u(t) + u^{\frac{3}{2}}(t) = f(t), \quad u(0) = 1, \quad 0 < \mu \leq 1, \quad 0 < t < 1,$$

where

$$f(t) = \frac{40320t^{8-\mu}}{\Gamma(9-\mu)} - 3\frac{\Gamma(5+\frac{\mu}{2})t^{4-\frac{\mu}{2}}}{\Gamma(5-\frac{\mu}{2})} + \frac{9\Gamma(\mu+1)}{4} + \left[\frac{3t^{\frac{\mu}{2}}}{2} - t^4\right]^3.$$

The exact solution of this example is $u(t) = t^8 - 3t^{4+\frac{\mu}{2}} + \frac{9t^\mu}{4}$, which was given in [35]. Diagrams of the numerical solutions for different values of μ are shown in Figure 7. We calculated the logarithm of the absolute error for different values of μ , and their diagrams are shown in Figure 8.

6 Conclusions

In this paper, the FABM and the HPTM were presented to study the initial value problem of the linear and nonlinear fractional differential equations involving the Caputo–Prabhakar fractional derivatives for numerically solving this type of equations. In addition, the two proposed methods were compared. Numerical examples and graphical representations were presented for testing the validity and accuracy of suggested methods. These examples showed that the FABM is a strong tool for obtaining the numerical solutions of linear and nonlinear fractional differential equations. Furthermore, the stability property of the FABM and the convergence analysis for both methods were considered. The numerical results were stable and converged well for different values of t for all examples.

References

1. An, J., Van Hese, E. and Baes, M. *Phase-space consistency of stellar dynamical models determined by separable augmented densities*, Mon. Not. R. Astron. Soc. 422(1) (2012), 652–664.
2. Atangana, A. and Owolabi, K.M. *New numerical approach for fractional differential equations*, Math. Model. Nat. Phenom, 13(1) (2018), Paper No. 3, 21 pp.
3. Balci, E., Öztürk, İ. and Kartal, S. *Dynamical behaviour of fractional order tumor model with Caputo and conformable fractional derivative*, Chaos Solitons Fractals, 123 (2019), 43–51.
4. Derakhshan, M.H., Ahmadi Darani, M., Ansari, A. and Khoshsiar Ghaziani, R. *On asymptotic stability of Prabhakar fractional differential systems*, Comput. Methods Differ. Equ. 4(4) (2016), 276–284.

5. Derakhshan, M.H. and Ansari, A. *On Hyers-Ulam stability of fractional differential equations with Prabhakar derivatives*, Anal. 38(1) (2018), 37–46.
6. Derakhshan, M.H. and Ansari, A. *Fractional Sturm-Liouville problems for Weber fractional derivatives*, Int. J. Comput. Math. 96(2) (2019), 217–237.
7. Derakhshan, M.H. and Ansari, A. *Numerical approximation to Prabhakar fractional Sturm-Liouville problem*, Comput. Appl. Math, 38(2) (2019), 71.
8. Derakhshan, M.H., Ansari, A. and Ahmadi Darani, M. *On asymptotic stability of Weber fractional differential systems*, Comput. Methods Differ. Equ. 6(1) (2018), 30–39.
9. D'Ovidio, M. and Polito, F. *Fractional diffusion-telegraph equations and their associated stochastic solutions*, Theory. Probab. Appl. 64(4) (2018), 552–574.
10. Dumitru, B., Kai, D. and Enrico, S. *Fractional calculus: Models and numerical methods*, World Scientific, 2012.
11. Garra, R., Gorenflo, R., Polito, F. and Tomovski, Ž. *Hilfer-Prabhakar derivatives and some applications*, Appl. math. comput. 242 (2014), 576–589.
12. Garrappa, R., Mainardi, F. and Guido, M. *Models of dielectric relaxation based on completely monotone functions*, Fract. Calc. Appl. Anal. 19(5) (2016), 1105–1160.
13. Ghorbani, A. *Beyond Adomian polynomials: He polynomials*, Chaos. Solitons. Fractals, 39(3) (2009), 1486–1492.
14. Giusti, A. and Colombaro, I. *Prabhakar-like fractional viscoelasticity*, Commun. Nonlinear. Sci. Numer. Simul. 56 (2018), 138–143.
15. Gorenflo, R., Kilbas, A.A., Mainardi, F. and Rogosin, S.V. *Mittag-Leffler functions, related topics and applications*, Springer Monographs in Mathematics. Springer, Heidelberg, 2014.
16. Górska, K., Horzela, A., Bratek, L., Penson, K.A. and Dattoli, G. *The probability density function for the Havriliak-Negami relaxation*, J. Phys. A, Math. Theor. 51(13) (2018) Article ID 135202, 15 pp.
17. Hamarsheh, M., Ismail, A.I. and Odibat, Z. *An analytic solution for fractional order Riccati equations by using optimal homotopy asymptotic method*, Appl. Math. Sci. 10(23) (2016), 1131–1150.

18. Heydari, M., Loghmani, G.B. and Hosseini, S.M. *Operational matrices of Chebyshev cardinal functions and their application for solving delay differential equations arising in electrodynamics with error estimation*, Appl. Math. Model, 37(14–15) (2013), 7789–7809.
19. Hou, T. and Leng, H. *Numerical analysis of a stabilized Crank-Nicolson/Adams-Bashforth finite difference scheme for Allen-Cahn equations*, Appl. Math. Lett. 102 (2020), 106–150.
20. Karaagac, B. *Two step Adams Bashforth method for time-fractional Tricomi equation with non-local and non-singular Kernel*, Chaos. Solitons. Fractals, 128 (2019), 234–241.
21. Karunakar, P. and Chakraverty, S. *Solutions of time-fractional third- and fifth-order Korteweg-de-Vries equations using homotopy perturbation transform method*, Eng. Comput. 36(7) (2019), 2309–2326.
22. Kilbas, A.A., Saigo, M. and Saxena, R.K. *Generalized Mittag-Leffler function and generalized fractional calculus operators*, Integral. Transform. Spec. Funct. 15(1) (2004), 31–49.
23. Kumar, S., Kumar, A. and Odibat, Z.M. *A nonlinear fractional model to describe the population dynamics of two interacting species*, Math. Methods. Appl. Sci. 40(11) (2017), 4134–4148.
24. Li, C., Kumar, A., Kumar, S. and Yang, X.J. *On the approximate solution of nonlinear time-fractional KdV equation via modified homotopy analysis Laplace transform method*, J. Nonlinear. Sci. Appl. 9 (2016), 5463–5470.
25. Liemert, A., Sandev, T. and Kantz, H. *Generalized Langevin equation with tempered memory kernel*, Physica. A. 466 (2017), 356–369.
26. Miskinis, P. *The Havriliak-Negami susceptibility as a nonlinear and non-local process*, Phys. Scr. 2009.
27. Owolabi, K.M. and Atangana, A. *Analysis and application of new fractional Adams-Bashforth scheme with Caputo-Fabrizio derivative*, Chaos. Solitons. Fractals, 105 (2017), 111–119.
28. Pan, Y., He, Y. and Mikkola, A. *Accurate real-time truck simulation via semirecursive formulation and Adams-Bashforth-Moulton algorithm*, Acta. Mech. Sin. 35 (2019), 641–652.
29. Pandey, R. K. and Mishra, H.K. *Homotopy analysis Sumudu transform method for time-fractional third order dispersive partial differential equation*, Adv. Comput. Math. 43(2) (2017), 365–383.
30. Pirim, N.A. and Ayaz, F. *A new technique for solving fractional order systems: Hermite collocation method* Appl. Math, 7(18)(2016), 2307–2323.

31. Podlubny, I. *Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications*, Elsevier, 1998.
32. Qureshi, S. and Kumar, P. *Using Shehu integral transform to solve fractional order Caputo type initial value problems*, J. Appl. Math. Comput. Mech. 18(2) (2019), 75–83.
33. Rathore, S., Kumar, D., Singh, J. and Gupta, S. *Homotopy analysis Sumudu transform method for nonlinear equations*, Int. J. Ind. Math. 4(4) (2012), 301–314.g
34. Ratti, I. *Comparative study of nonlinear partial differential equation using homotopy perturbation transform method (HPTM) using He's polynomial and mixture of Elzaki transform and partial differential transform method (PDTM)*, Int. J. Appl. Eng. Res. 13(18) (2018), 14037–14040.
35. Saadatmandi, A. and Dehghan, M. *A new operational matrix for solving fractional-order differential equations*, Comput. Math. Appl. 59(3) (2010), 1326–1336.
36. Singh, J., Kumar, D., Swroop, R. and Kumar, S. *An efficient computational approach for time-fractional Rosenau–Hyman equation*, Neural. Comput. Appl. 30(10) (2018), 3063–3070.
37. Stanislavsky, A. and Weron, K. *A typical case of the dielectric relaxation responses and its fractional kinetic equation*, Fract. Calc. Appl. Anal. 19(1) (2016), 212–228.
38. Subashini, R., Ravichandran, C., Jothimani, K. and Baskonus, H.M. *Existence results of Hilfer integro-differential equations with fractional order*, Discrete. Contin. Dyn. Syst. 13(3) (2020), 911–923.
39. Touchent, K.A., Hammouch, Z., Mekkaoui, T. and Belgacem, F. *Implementation and convergence analysis of homotopy perturbation coupled with sumudu transform to construct solutions of local-fractional pdes*, Fractal. Fract. 2(3) (2018), 22.
40. Vahidi, J. *The combined Laplace-homotopy analysis method for partial differential equations*, J. Math. Comput. Sci. 16(1) (2016), 88–102.
41. Yépez-Martínez, H., and Gómez-Aguilar, J.F. *Numerical and analytical solutions of nonlinear differential equations involving fractional operators with power and Mittag-Leffler kernel*, Math. Model. Nat. Phenom. 13(1) (2018), Paper No. 13, 17 pp.



A four directions variational method for solving image processing problems[†]

A.R. Hosseini* and E.E. Esfahani

Abstract

In this paper, based on a discrete total variation model, a modified discretization of total variation (TV) is introduced for image processing problems. Two optimization problems corresponding to compressed sensing magnetic resonance imaging (MRI) data reconstruction problem and image denoising are proposed. In the proposed method, instead of applying isotropic TV whose gradient field is a two directions vector, a four directions discretization with some modification is applied for the inverse problems. A dual formulation for the proposed TV is explained and an efficient primal-dual algorithm is employed to solve the problem. Some important image test problems in MRI and image denoising problems are considered in the numerical experiments. We compare our model with the state of the art methods.

AMS(2010): Primary 62H35; Secondary 68U10, 94A08.

Keywords: Total variation; Magnetic resonance imaging; Primal-dual optimization method; Regularization; Image denoising.

1 Introduction

There are different methods to solve image processing problems such as spatial filtering [11, 6], methods based on partial differential equations [21, 26],

*Corresponding author

Received 1 January 2020; revised 1 March 2020; accepted 31 May 2020

Alireza Hosseini

School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Iran. e-mail: hosseini.alireza@ut.ac.ir

Erfan Ebrahim Esfahani

School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, Iran. e-mail: ebrahim.esfahani@ut.ac.ir

[†] This article was suggested by the Scientific Committee of the Third Iranian Seminar on Control and Optimization for publication in IJNAO, which was accepted after independent review.

variational methods [8, 2, 10, 1, 5], and machine learning approach as well as the methods based on deep learning [17, 25]. Variational methods propose some different functionals as the tools for calculating accidental variations in a signal or function. The pioneer work in this area was proposed by Rudin, Osher, and Fatemi [24]. Recently, Condat [10] proposed an isotropic discrete total variation with high accuracy. Moreover, the total generalized variation as a generalization of total variation functional has been proposed in [5]. Consider the following optimization problem to solve a continuous version of mathematical image problems:

$$\min_s \frac{1}{2} \int_{\Omega} |G(s(t)) - g(t)|^2 dt + \lambda J(s),$$

where G is an appropriate function (for more details see [7]) and

$$J(s) = \sup \left\{ - \int_{\Omega} s \cdot \text{div} \phi dt : \phi \in C_c^1(\Omega, \mathbb{R}^N), |\phi(t)| \leq 1, \forall t \in \Omega \right\}. \quad (1)$$

Also J is the dual definition of total variation (TV) of the $L_c^1(\Omega)$ (locally Lebesgue integrable) function s . If $G = I$, then problem (1) is named denoising problem (for image g). A function s is said to have bounded variation whenever $J(s) < \infty$. The space $BV(\Omega)$ of functions with bounded variation is the set of functions $s \in L^1(\Omega)$ such that $J(s) < \infty$, endowed with the norm $\|s\|_{BV(\Omega)} = \|s\|_{L^1(\Omega)} + J(s)$. Obviously, for the smooth function $s \in C^1(\Omega)$ (or $s \in W^{1,1}(\Omega)$),

$$J(s) = \int_{\Omega} |\nabla s| dt$$

the minimization of $J(s)$ is equivalent to minimization of the sum of abrupt variations (the majority of the derivative) over its dimension. To solve real digital imaging problems, discretization of (1) is inevitable. There are different forms of discretization for TV functional in the papers. As the review of such literatures is long some, we refer the readers to [10, 13] and references therein.

Total generalized variation (TGV) is a cleverly defined functional for calculating a combinational value of function variation by means of derivatives up to a given order. TGV has some superiorities in comparison with the first-order TV, such as its ability to distinguish staircase artifacts, and as a result, more cleaner image can be reconstructed. Usually in applications, the second-order TGV is sufficient, because the higher-order will face to the complexity of calculation. The second-order TGV is defined by the following optimization problem (see [5]):

$$\text{TGV}_{\alpha}^2(u) = \sup \left\{ \int_{\Omega} u \cdot \text{div}^2 v dx : v \in C_c^2(\Omega, \text{Sym}^2(\mathbb{R}^d)), \|\text{div}^l v\|_{\infty} \leq \alpha_l, l = 0, 1 \right\}.$$

In [5], a discretization of the TGV was proposed. Based on discretization of TV in [13], in this paper, a new discretization of TV (1) is given; operators are changed and linear operator $\mathcal{O}_+$ is inserted to ensure the projection on the

vertices. Then the magnetic resonance imaging (MRI) data reconstruction and image denoising problems will be formulated based on the new proposed discrete TV and will be compared with some state of the art models. In the proposed model, instead of using two directions gradient field in isotropic TV, a four directions discretization model is proposed. Actually, we can take more information from structure of the images, and consequently a new function for measuring the majority of the noise and oscillation is proposed. Moreover, after some modification based on the Condat's model [10], a discretization of the total variation with better isotropy property will be defined. Numerical results show that the proposed method has better performance in comparison with isotropic TV, TGV, and some other famous models in terms of removing noise and compressed sensing MRI problems.

The paper is arranged as the following; in Section 2, the Chambolle–Pock algorithm is explained for solving optimization problems in image processing. In Section 3, a new discrete TV model is proposed. In Section 4, the problem of reconstruction of undersampled MRI data as well as the problem of image denoising are formulated. Some important test problems in medical MRI and denoising are solved numerically and compared with some state of the art, in Section 4. Finally some conclusions are given in Section 5.

2 A numerical algorithm for solving convex image problems

2.1 Algorithm description

Consider a finite-dimensional vector space X with an inner product $\langle \cdot, \cdot \rangle_X$ and Euclidean norm $\|\cdot\|_2 = \sqrt{\langle \cdot, \cdot \rangle_X}$. $C \subset X$ we define the *indicator* function of C as

$$\delta_C(x) = \begin{cases} 0, & x \in C, \\ \infty, & x \notin C. \end{cases}$$

For any function $F : X \rightarrow [-\infty, +\infty]$, the *convex conjugate* at $z \in X$ is defined by

$$F^*(z) = \max_x \langle x, z \rangle - F(x). \quad (2)$$

Moreover, for any $\theta > 0$ and $\bar{x} \in X$, the *proximal* (or *proximity*) mapping of F at $\bar{x}$ is defined by

$$\text{prox}_{\theta F}(\bar{x}) = \arg \min_x \frac{\|x - \bar{x}\|^2}{2\theta} + F(x).$$

Now, suppose that Y is another real vector space with inner product $\langle \cdot, \cdot \rangle_Y$ and the corresponding induced norm $\|\cdot\|_2 = \sqrt{\langle \cdot, \cdot \rangle_Y}$. Let $K : X \rightarrow Y$ be a

bounded linear operator with the operator norm

$$\|K\| := \max\{\|Kx\|_2 : x \in X \text{ and } \|x\|_2 \leq 1\} < \infty.$$

The adjoint of K is defined as an operator K^* for which the equality $\langle Kx, y \rangle_Y = \langle x, K^*y \rangle_X$ holds for all x and y . The general problem, we tackle, is

$$\min_x F(Kx) + G(x), \quad (3)$$

where $F : X \rightarrow [0, \infty)$ and $G : Y \rightarrow [0, \infty)$ are proper, closed, and convex. By virtue of (2), the convex minimization problem (3) can be reformulated as the saddle point convex-concave problem

$$\min_x \max_y \langle Kx, y \rangle_Y + G(x) - F^*(y), \quad (4)$$

which is also known as the primal-dual formulation of (3). Hereafter we assume that a solution to this problem always exists. The method we employ to solve (4), which lays the groundwork of what we shall propose in this paper, is the celebrated Chambolle–Pock algorithm, recalled below: The algorithm

Data: K, K^*, F^*, G

Result: x^k

initialization: Choose parameters $\sigma > 0$, $\tau > 0$, and initial estimates $(x^0, y^0) \in X \times Y$.

while *convergence criterion not met, for $k=0, 1, \dots$, do*

$$\begin{array}{l} 1- y^{k+1} = \text{prox}_{\sigma F^*}(y^k + \sigma K\tilde{x}^k), \\ 2- x^{k+1} = \text{prox}_{\tau G}(x^k - \tau K^*y^{k+1}), \\ 3- \tilde{x}^{k+1} = 2x^{k+1} - x^k. \end{array}$$

end

Algorithm 1: The Chambolle–Pock algorithm for solving problem (4)

is guaranteed to converge provided that $\sigma\tau < \frac{1}{\|K\|^2}$. In practical situations, where computing the exact value of $\|K\|$ is not easy, finding an upper bound $B \geq \|K\|^2$ and setting $\sigma = \tau = \frac{1}{\sqrt{B}}$ are sufficient for convergence. We chose this algorithm over other well-known first-order methods such as ADMM or FISTA because of its superior convergence rate [9], and the fact that it requires fewer parameters to be tuned.

2.2 Convergence analysis

Here, we express the convergence theorem of the Chambolle–Pock algorithm to solve problem (4). The convergence analysis was argued in [9]. The following theorem guarantees the convergence of Chambolle–Pock algorithm for

solving problem (4).

Theorem 1. [9] Assume X and Y are two finite-dimensional Hilbert spaces and that $\tau, \sigma > 0$. Moreover assume $\sigma\tau\|K\|^2 < 1$. Then there exists a solution $(\bar{x}, \bar{y}) \in X \times Y$ of problem (4) such that the sequence (x^n, y^n) in Algorithm 1 converges to $(\bar{x}, \bar{y})$.

Note that, in [9], it was shown that the rate of convergence is $\mathcal{O}(\frac{1}{n})$. Moreover, under some uniformly convexity assumptions, the rate of convergence can be improved to $\mathcal{O}(\frac{1}{e^N})$.

3 A Discretization for TV

Assume that $x \in \mathbb{R}^{N_1 \times N_2}$ is a gray scaled two-dimensional image and that $A = \{(n_1, n_2), n_1 = 1, \dots, N_1, n_2 = 1, \dots, N_2\}$ is a set of grids. Consider the following new discrete TV model, which is based on four directional gradient with imposed some operational constraints:

$$\text{TV}_{dis}(x) = \max_{u \in (\mathbb{R}^4)^{N_1 \times N_2}} \{ \langle \mathbb{D}x, u \rangle : |\mathcal{O}_{\uparrow}u(n_1, n_2)| \leq 1, |\mathcal{O}_{\leftrightarrow}u(n_1, n_2)| \leq 1, \\ |\mathcal{O}_{\bullet}u(n_1, n_2)| \leq 1, |\mathcal{O}_{+}u(n_1, n_2)| \leq 1, \forall (n_1, n_2) \in X \},$$

where

$$\begin{aligned} \mathbb{D} &= (D_1, D_2, D_3, D_4), \\ D_1x(n_1, n_2) &= x(n_1 + 1, n_2) - x(n_1, n_2), \\ D_2x(n_1, n_2) &= x(n_1, n_2 + 1) - x(n_1, n_2), \\ D_3x(n_1, n_2) &= x(n_1 + 1, n_2 + 1) - x(n_1, n_2), \\ D_4x(n_1, n_2) &= x(n_1 - 1, n_2 + 1) - x(n_1, n_2), \end{aligned}$$

$$|\mathcal{O}_{\star}u(n_1, n_2)| = \sqrt{\sum_{j=1}^4 [(\mathcal{O}_{\star}u)_j(n_1, n_2)]^2}, \quad (n_1, n_2) \in X, \star = \uparrow, \leftrightarrow, \bullet, +,$$

and

$$\begin{aligned} (\mathcal{O}_{\uparrow}u)_1(n_1, n_2) &= u_1(n_1, n_2), \\ (\mathcal{O}_{\uparrow}u)_2(n_1, n_2) &= \frac{1}{4}(u_2(n_1, n_2) + u_2(n_1, n_2 - 1) \\ &\quad + u_2(n_1 + 1, n_2) + u_2(n_1 + 1, n_2 - 1)), \\ (\mathcal{O}_{\uparrow}u)_3(n_1, n_2) &= \frac{1}{2}(u_3(n_1, n_2) + u_3(n_1, n_2 - 1)), \\ (\mathcal{O}_{\uparrow}u)_4(n_1, n_2) &= \frac{1}{4}(u_4(n_1, n_2) + u_4(n_1, n_2 - 1) \\ &\quad + u_4(n_1 + 1, n_2) + u_4(n_1 + 1, n_2 - 1)), \end{aligned}$$

$$\begin{aligned}
(\mathcal{O}_{\leftrightarrow u})_1(n_1, n_2) &= \frac{1}{4}(u_1(n_1, n_2) + u_1(n_1 - 1, n_2) \\
&\quad + u_1(n_1, n_2 + 1) + u_1(n_1 - 1, n_2 + 1)), \\
(\mathcal{O}_{\leftrightarrow u})_2(n_1, n_2) &= u_2(n_1, n_2), \\
(\mathcal{O}_{\leftrightarrow u})_3(n_1, n_2) &= \frac{1}{2}(u_3(n_1, n_2) + u_3(n_1 - 1, n_2)), \\
(\mathcal{O}_{\leftrightarrow u})_4(n_1, n_2) &= \frac{1}{2}(u_4(n_1, n_2) + u_4(n_1 + 1, n_2)),
\end{aligned}$$

$$\begin{aligned}
(\mathcal{O}_+ u)_1(n_1, n_2) &= \frac{1}{2}(u_1(n_1, n_2) + u_1(n_1, n_2 + 1)), \\
(\mathcal{O}_+ u)_2(n_1, n_2) &= \frac{1}{2}(u_2(n_1, n_2) + u_2(n_1 + 1, n_2)), \\
(\mathcal{O}_+ u)_3(n_1, n_2) &= u_3(n_1, n_2), \\
(\mathcal{O}_+ u)_4(n_1, n_2) &= u_4(n_1, n_2),
\end{aligned}$$

$$\begin{aligned}
(\mathcal{O}_\bullet u)_1(n_1, n_2) &= \frac{1}{2}(u_1(n_1, n_2) + u_1(n_1 - 1, n_2)), \\
(\mathcal{O}_\bullet u)_2(n_1, n_2) &= \frac{1}{2}(u_2(n_1, n_2) + u_2(n_1, n_2 - 1)), \\
(\mathcal{O}_\bullet u)_3(n_1, n_2) &= \frac{1}{4}(u_3(n_1, n_2) + u_3(n_1, n_2 - 1) \\
&\quad + u_3(n_1 - 1, n_2) + u_3(n_1 - 1, n_2 - 1)), \\
(\mathcal{O}_\bullet u)_4(n_1, n_2) &= \frac{1}{4}(u_4(n_1, n_2) + u_4(n_1 + 1, n_2) \\
&\quad + u_4(n_1, n_2 - 1) + u_4(n_1 + 1, n_2 - 1)).
\end{aligned}$$

Theorem 2. Assume that F is convex and that $\lambda \geq 0$. Then the problem

$$\min_x F(x) + \lambda \text{TV}_{dis}(x),$$

is equivalent to the following optimization problem:

$$\begin{aligned}
&\min F(x) + \lambda\{|v_{\uparrow\downarrow}| + |v_{\leftrightarrow}| + |v_\bullet| + |v_+|\}, \\
&\text{subject to} \\
&\mathcal{O}_{\uparrow\downarrow}^*(v_{\uparrow\downarrow})(n_1, n_2) + \mathcal{O}_{\leftrightarrow}^*(v_{\leftrightarrow})(n_1, n_2) + \mathcal{O}_\bullet^*(v_\bullet)(n_1, n_2) \\
&\quad + \mathcal{O}_+^*(v_+)(n_1, n_2) = \mathbb{D}x(n_1, n_2), \quad (n_1, n_2) \in X,
\end{aligned}$$

where

$$\mathcal{O}_{\uparrow\downarrow}^*(v_{\uparrow\downarrow})(n_1, n_2) = \begin{pmatrix} v_{\uparrow\downarrow}^1(n_1, n_2) \\ \frac{1}{4}(v_{\uparrow\downarrow}^2(n_1, n_2) + v_{\uparrow\downarrow}^2(n_1, n_2 + 1) + v_{\uparrow\downarrow}^2(n_1 - 1, n_2) \\ \quad + v_{\uparrow\downarrow}^2(n_1 - 1, n_2 + 1)) \\ \frac{1}{2}(v_{\uparrow\downarrow}^3(n_1, n_2) + v_{\uparrow\downarrow}^3(n_1, n_2 + 1)) \\ \frac{1}{4}(v_{\uparrow\downarrow}^4(n_1, n_2) + v_{\uparrow\downarrow}^4(n_1, n_2 + 1) + v_{\uparrow\downarrow}^4(n_1 - 1, n_2) \\ \quad + v_{\uparrow\downarrow}^4(n_1 - 1, n_2 + 1)) \end{pmatrix},$$

$$\begin{aligned}
\mathcal{O}_{\leftrightarrow}^*(v_{\leftrightarrow})(n_1, n_2) &= \begin{pmatrix} \frac{1}{4}(v_{\leftrightarrow}^1(n_1, n_2) + v_{\leftrightarrow}^1(n_1 + 1, n_2) + v_{\leftrightarrow}^1(n_1, n_2 - 1) \\ + v_{\leftrightarrow}^1(n_1 + 1, n_2 - 1)) \\ v_{\leftrightarrow}^2(n_1, n_2) \\ \frac{1}{2}(v_{\leftrightarrow}^3(n_1, n_2) + v_{\leftrightarrow}^3(n_1 + 1, n_2)) \\ \frac{1}{2}(v_{\leftrightarrow}^4(n_1, n_2) + v_{\leftrightarrow}^4(n_1 - 1, n_2)) \end{pmatrix}, \\
\mathcal{O}_{\bullet}^*(v_{\leftrightarrow})(n_1, n_2) &= \begin{pmatrix} \frac{1}{2}(v_{\bullet}^1(n_1, n_2) + v_{\bullet}^1(n_1 + 1, n_2)) \\ \frac{1}{2}(v_{\bullet}^2(n_1, n_2) + v_{\bullet}^2(n_1, n_2 + 1)) \\ \frac{1}{4}(v_{\bullet}^3(n_1, n_2) + v_{\bullet}^3(n_1, n_2 + 1) + v_{\bullet}^3(n_1 + 1, n_2) \\ + v_{\bullet}^3(n_1 + 1, n_2 + 1)) \\ \frac{1}{4}(v_{\bullet}^4(n_1, n_2) + v_{\bullet}^4(n_1 - 1, n_2) + v_{\bullet}^4(n_1, n_2 + 1) \\ + v_{\bullet}^4(n_1 - 1, n_2 + 1)) \end{pmatrix}, \\
\mathcal{O}_{+}^*(v_{+})(n_1, n_2) &= \begin{pmatrix} \frac{1}{2}(v_{+}^1(n_1, n_2) + v_{+}^1(n_1, n_2 - 1)) \\ \frac{1}{2}(v_{+}^2(n_1, n_2) + v_{+}^2(n_1 - 1, n_2)) \\ v_{+}^3(n_1, n_2) \\ v_{+}^4(n_1, n_2) \end{pmatrix},
\end{aligned}$$

and

$$|v_{\star}| = \sqrt{\sum_{n_1, n_2} |v_{\star}^j|^2}, \star = \uparrow, \leftrightarrow, \bullet, +.$$

Proof. It is a direct conclusion of Fenchel–Rockefeller, primal-dual theorem (see [10], for instance). $\square$

4 Formulation of some image processing problems

4.1 Compressing MRI data

Consider the following problem for reconstructing MR images from under-sampled k -space data:

$$\min_{u \in U} \frac{1}{2\lambda} \|\mathcal{F}_m u - b\|_2^2 + \beta \|\Psi u\|_1 + \text{TV}_{dis}(u), \quad (5)$$

where $u \in U = \mathbb{R}^{n \times n}$ is the MR image, which is defined in spatial domain and $\mathcal{F}_m$ is the undersampled Fourier operator, which converts signals in spatial domain to its corresponding frequency representation and undersample it. In other words, it can be defined as $\mathcal{F}_m u := M \odot \mathcal{F}(u)$, where $\mathcal{F}$ is the two-dimensional fast Fourier transform, $M_{n \times n}$ is the k -space sampling mask with ones at m sampled frequencies and zeros at undersampled locations, and $\odot$ is the component-wise multiplication. We further assume that the sampling rate $\frac{m}{n^2}$ is considerably small (highly undersampled k -space) and that $b \in \mathbb{C}^{n \times n}$ is the partially scanned k -space with only m sampled frequencies and $n^2 - m$ zeros. The so-called zero-filled solution is defined as $u_{zf} := \mathcal{F}^{-1}(b)$. Furthermore, Ψ is a discrete wavelet transform, and the constants $\beta, \lambda \in \mathbb{R}$ are regularization parameters.

Referring to Theorem 2, we can get the following equivalent formulation:

$$\begin{aligned} \min_{u \in U} \quad & \frac{1}{2\lambda} \|\mathcal{F}_m u - b\|_2^2 + \beta \|\Psi u\|_1 + |v_{\downarrow}^*|_1 + |v_{\leftrightarrow}^*|_1 + |v_{\bullet}^*|_1 + |v_+^*|_1, \\ \text{s.t.} \quad & \mathcal{O}_{\downarrow}^*(v_{\downarrow})(n_1, n_2) + \mathcal{O}_{\leftrightarrow}^*(v_{\leftrightarrow})(n_1, n_2) + \mathcal{O}_{\bullet}^*(v_{\bullet})(n_1, n_2) \\ & + \mathcal{O}_+^*(v_+)(n_1, n_2) = \mathbb{D}u(n_1, n_2), \quad (n_1, n_2) \in A. \\ & A = \{(n_1, n_2) : n_1, n_2 = 1, \dots, n\}. \end{aligned} \quad (6)$$

To solve this problem with Algorithm 1, we need to reformulate problem (6) to the form of (3). Now we define the following linear operator K :

$$Kx = \begin{pmatrix} \Psi & 0 & 0 & 0 & 0 \\ \mathcal{F}_m & 0 & 0 & 0 & 0 \\ -\mathbb{D} & \mathcal{O}_{\downarrow}^* & \mathcal{O}_{\leftrightarrow}^* & \mathcal{O}_{\bullet}^* & \mathcal{O}_+^* \end{pmatrix} \begin{pmatrix} u \in \mathbb{R}^{N_1 \times N_2} \\ v_{\downarrow} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_{\leftrightarrow} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_{\bullet} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_+ \in (\mathbb{R}^4)^{N_1 \times N_2} \end{pmatrix} = \begin{pmatrix} \bar{k} \in (\mathbb{R})^{N_1 \times N_2} \\ \bar{h} \in (\mathbb{R})^{N_1 \times N_2} \\ \bar{v} \in (\mathbb{R}^4)^{N_1 \times N_2} \end{pmatrix} = Y.$$

Assume

$$\begin{aligned} F(Y) &= \frac{1}{2\lambda} \|Y_2 - b\|_2^2 + \beta \|Y_1\|_1 + \delta_{\{0\}}(Y_3), \\ G(X) &= |v_{\downarrow}^*|_1 + |v_{\leftrightarrow}^*|_1 + |v_{\bullet}^*|_1 + |v_+^*|_1. \end{aligned}$$

From arguments about calculating adjoint of a given operator, we get

$$F^*(y) = \frac{\lambda}{2} \|y_2\|_2^2 + \langle y_2, b \rangle + \delta_{\|\cdot\|_{\infty} \leq \beta}(y_1) + \langle y_3, \mathbb{D}u \rangle.$$

4.1.1 The y -subproblems

According to Algorithm 1, for $\{Y\} = (\bar{k}, \bar{h}, \bar{v})^\top$, we obtain

Result: u^k

Initialization: Set parameters as $\sigma = \tau = \frac{1}{\sqrt{19}}$ and initial estimates as $u^0 = u_{zf}$, $\tilde{u}^0 = 0$, $v_{\downarrow}^0 = \tilde{v}_{\downarrow}^0 = 0$, $v_{\leftrightarrow}^0 = \tilde{v}_{\leftrightarrow}^0 = 0$, $v_{\bullet}^0 = \tilde{v}_{\bullet}^0 = 0$, $v_{+}^0 = \tilde{v}_{+}^0 = 0$, $\bar{h}^0 = 0$, $\bar{k}^0 = 0$ and $\bar{w}^0 = 0$.

while convergence criterion not met, for $k=0,1,\dots$ **do**

1- $\bar{h}^{k+1} = \text{prox}_{\sigma(\frac{\lambda}{2}\|\cdot\|_2^2 + t_b)}(\bar{h}^k + \sigma(\mathcal{F}_m(\tilde{u}^k)))$, $\text{prox}_{\sigma(\frac{\lambda}{2}\|\cdot\|_2^2 + t_b)}(\alpha) = \frac{\alpha - \sigma b}{1 + \sigma \lambda}$
 2- $\bar{k}^{k+1} = \text{proj}_{\|\cdot\|_{\infty} \leq \beta}(\bar{k}^k + \sigma(\Psi \tilde{u}^k))$, $\left(\text{proj}_{\|\cdot\|_{\infty} \leq \beta}(s)\right)(i, j) =$
 $\min\{\max\{s(i, j), -\beta\}, \beta\}$, $\forall i, j \in \{1, 2, \dots, n\}$,
 3- $\bar{w}^{k+1} = \bar{w}^k + \sigma(\mathcal{O}_{\downarrow}^* \tilde{v}_{\downarrow}^k + \mathcal{O}_{\leftrightarrow}^* \tilde{v}_{\leftrightarrow}^k + \mathcal{O}_{\bullet}^* \tilde{v}_{\bullet}^k + \tilde{v}_{+}^k - \mathbb{D} \tilde{u}^k)$
 4- $u^{k+1} = u^k - \tau(\mathcal{F}_m^* \bar{h}^{k+1} + \Psi^* \bar{k}^{k+1} - \mathbb{D}^* \bar{w}^{k+1})$,
 Use formula $\text{Prox}_{\theta \alpha_0 \|\cdot\|_1}(\alpha)(i, j) = (1 - \frac{\theta \alpha_0}{\max\{\|\alpha(i, j)\|_2, \theta \alpha_0\}}) \alpha(i, j)$ for steps 5-8
 below:
 5- $v_{\downarrow}^{k+1} = \text{Prox}_{\tau \|\cdot\|_1}(v_{\downarrow}^k - \tau(\mathcal{O}_{\downarrow} \bar{w}^{k+1}))$,
 6- $v_{\leftrightarrow}^{k+1} = \text{Prox}_{\tau \|\cdot\|_1}(v_{\leftrightarrow}^k - \tau(\mathcal{O}_{\leftrightarrow} \bar{w}^{k+1}))$,
 7- $v_{\bullet}^{k+1} = \text{Prox}_{\tau \|\cdot\|_1}(v_{\bullet}^k - \tau(\mathcal{O}_{\bullet} \bar{w}^{k+1}))$,
 8- $v_{+}^{k+1} = \text{Prox}_{\tau \|\cdot\|_1}(v_{+}^k - \tau(\mathcal{O}_{+} \bar{w}^{k+1}))$,
 9- $\tilde{u}^{k+1} = 2u^{k+1} - u^k$, $\tilde{v}_{\downarrow}^{k+1} = 2v_{\downarrow}^{k+1} - v_{\downarrow}^k$, $\tilde{v}_{\leftrightarrow}^{k+1} = 2v_{\leftrightarrow}^{k+1} - v_{\leftrightarrow}^k$, $\tilde{v}_{\bullet}^{k+1} =$
 $2v_{\bullet}^{k+1} - v_{\bullet}^k$, $\tilde{v}_{+}^{k+1} = 2v_{+}^{k+1} - v_{+}^k$.

end

Algorithm 2: The algorithm for undersampled MRI reconstruction

$$\begin{aligned} \begin{pmatrix} \bar{k}^{k+1} \\ \bar{h}^{k+1} \\ \bar{w}^{k+1} \end{pmatrix} &= \text{prox}_{\sigma F^*} \left(\begin{pmatrix} \bar{k}^k \\ \bar{h}^k \\ \bar{w}^k \end{pmatrix} + \sigma \begin{pmatrix} \Psi & 0 & 0 & 0 & 0 \\ \mathcal{F}_m & 0 & 0 & 0 & 0 \\ -\mathbb{D} & \mathcal{O}_{\downarrow}^* & \mathcal{O}_{\leftrightarrow}^* & \mathcal{O}_{\bullet}^* & \mathcal{O}_{+}^* \end{pmatrix} \begin{pmatrix} u \in \mathbb{R}^{N_1 \times N_2} \\ v_{\downarrow} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_{\leftrightarrow} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_{\bullet} \in (\mathbb{R}^4)^{N_1 \times N_2} \\ v_{+} \in (\mathbb{R}^4)^{N_1 \times N_2} \end{pmatrix} \right) \\ &= \text{prox}_{\sigma F^*} \begin{pmatrix} \bar{k}^k + \sigma(\Psi(u^k)) \\ \bar{h}^k + \sigma(\mathcal{F}_m \tilde{u}^k) \\ \bar{w}^k + \sigma(\mathcal{O}_{\downarrow}^* v_{\downarrow}^k + \mathcal{O}_{\leftrightarrow}^* v_{\leftrightarrow}^k + \mathcal{O}_{\bullet}^* v_{\bullet}^k + \mathcal{O}_{+}^* v_{+}^k - \mathbb{D} u) \end{pmatrix}. \end{aligned}$$

$$\begin{cases} \bar{h}^{k+1} = \text{prox}_{\sigma(\frac{\lambda}{2}\|\cdot\|_2^2 + t_b)}(\bar{h}^k + \sigma(\mathcal{F}_m(\tilde{u}^k))), \text{prox}_{\sigma(\frac{\lambda}{2}\|\cdot\|_2^2 + t_b)}(\alpha) = \frac{\alpha - \sigma b}{1 + \sigma \lambda}, \\ \bar{k}^{k+1} = \text{proj}_{\|\cdot\|_{\infty} \leq \beta}(\bar{k}^k + \sigma(\Psi \tilde{u}^k)), \left(\text{proj}_{\|\cdot\|_{\infty} \leq \beta}(s)\right)(i, j) \\ = \min\{\max\{s(i, j), -\beta\}, \beta\}, \quad \forall i, j \in \{1, 2, \dots, n\}, \\ \bar{w}^{k+1} = \bar{w}^k + \sigma(\mathcal{O}_{\downarrow}^* v_{\downarrow}^k + \mathcal{O}_{\leftrightarrow}^* v_{\leftrightarrow}^k + \mathcal{O}_{\bullet}^* v_{\bullet}^k + \mathcal{O}_{+}^* v_{+}^k - \mathbb{D} u). \end{cases}$$

4.1.2 The x -subproblems

For $x = (u, v_{\downarrow}, v_{\leftrightarrow}, v_{\bullet}, v_{+})^{\top}$, according to Algorithm 1, we get,

$$\begin{aligned}
\begin{pmatrix} u^{k+1} \\ v_{\downarrow}^{k+1} \\ v_{\leftrightarrow}^{k+1} \\ v_{\bullet}^{k+1} \\ v_{+}^{k+1} \end{pmatrix} &= \text{prox}_{\tau G} \left(\begin{pmatrix} u^k \\ v_{\downarrow}^k \\ v_{\leftrightarrow}^k \\ v_{\bullet}^k \\ v_{+}^k \end{pmatrix} - \tau \begin{pmatrix} \Psi^* \mathcal{F}_m^* - \mathbb{D}^* \\ 0 & 0 & \mathcal{O}_{\downarrow} \\ 0 & 0 & \mathcal{O}_{\leftrightarrow} \\ 0 & 0 & \mathcal{O}_{\bullet} \\ 0 & 0 & \mathcal{O}_{+} \end{pmatrix} \begin{pmatrix} \bar{k}^{k+1} \\ \bar{h}^{k+1} \\ \bar{w}^{k+1} \end{pmatrix} \right) \\
&= \text{prox}_{\tau G} \begin{pmatrix} u^k - \tau(\Psi^* \bar{k}^{k+1} + \mathcal{F}_m^* \bar{h}^{k+1} - \mathbb{D}^* \bar{w}^{k+1}) \\ v_{\downarrow}^k - \tau(\mathcal{O}_{\downarrow} \bar{w}^{k+1}) \\ w_{\leftrightarrow}^k - \tau(\mathcal{O}_{\leftrightarrow} \bar{w}^{k+1}) \\ w_{\bullet}^k - \tau(\mathcal{O}_{\bullet} \bar{w}^{k+1}) \\ w_{+}^k - \tau(\mathcal{O}_{+} \bar{w}^{k+1}) \end{pmatrix},
\end{aligned}$$

where

$$\begin{aligned}
\mathbb{D}^* u(n_1, n_2) &= [u_1(n_1 - 1, n_2) - u_1(n_1, n_2)] + [u_2(n_1 - 1, n_2 - 1) - u_2(n_1, n_2)] \\
&\quad + [u_3(n_1 - 1, n_2 - 1) - u_3(n_1, n_2)] \\
&\quad + [u_4(n_1 + 1, n_2 - 1) - u_4(n_1, n_2)], \quad (n_1, n_2) \in A.
\end{aligned}$$

Based on the above argument, we propose Algorithm 2, to solve undersampled MRI reconstruction problem.

4.2 Image denoising formulation

Let $z \in \mathbb{R}^{N_1 \times N_2}$ be a gray scaled two-dimensional noisy image, and consider the following optimization problem:

$$\min_u \frac{1}{2} \|z - u\|_2^2 + \lambda \text{TV}_{dis}(u).$$

This problem is named denoising problem and can be rewritten with the following problem:

$$\begin{aligned}
\min_{u \in \mathcal{U}} \quad & \frac{1}{2} \|z - u\|_2^2 + |v_{\downarrow}^*|_1 + |v_{\leftrightarrow}^*|_1 + |v_{\bullet}^*|_1 + |v_{+}^*|_1 \\
\text{s.t.} \quad & \mathcal{O}_{\downarrow}^*(v_{\downarrow})(n_1, n_2) + \mathcal{O}_{\leftrightarrow}^*(v_{\leftrightarrow})(n_1, n_2) + \mathcal{O}_{\bullet}^*(v_{\bullet})(n_1, n_2) + \mathcal{O}_{+}^*(v_{+})(n_1, n_2) \\
& = \mathbb{D}u(n_1, n_2), \quad (n_1, n_2) \in X.
\end{aligned}$$

To solve this problem numerically, we need to determine functions F and G as well as the linear operator K in (3). Assume $x = (u, v_{\downarrow}^*, v_{\leftrightarrow}^*, v_{\bullet}^*, v_{+}^*)^\top$. Set

$$K = \begin{pmatrix} I & 0 & 0 & 0 & 0 \\ -\mathbb{D} & \mathcal{O}_{\downarrow}^* & \mathcal{O}_{\leftrightarrow}^* & \mathcal{O}_{\bullet}^* & \mathcal{O}_{+}^* \end{pmatrix}, \quad FY = \frac{1}{2} \|Y_1 - z\|_2^2 + \delta_{\{0\}}(Y_2 - \mathbb{D}z),$$

$$Gx = |v_{\downarrow}^*|_1 + |v_{\leftrightarrow}^*|_1 + |v_{\bullet}^*|_1 + |v_{+}^*|_1.$$

Consequently

$$K^* = \begin{pmatrix} I - \mathbb{D}^* \\ 0 & \mathcal{O}_\downarrow \\ 0 & \mathcal{O}_{\leftrightarrow} \\ 0 & \mathcal{O}_\bullet \\ 0 & \mathcal{O}_+ \end{pmatrix}, \quad F^* y = \|y_1\|_2^2 + \langle z, y_1 \rangle.$$

Analogous to the arguments about Algorithm 2, we can obtain the similar procedure (Algorithm 3), to solve the image denoising problem.

Result: u^k

Initialization: Set parameters as $\sigma = \tau = \frac{1}{\sqrt{19}}$ and initial estimates as $u^0 = z$, $\tilde{u}^0 = 0$, $v_\downarrow^0 = \tilde{v}_\downarrow^0 = 0$, $v_{\leftrightarrow}^0 = \tilde{v}_{\leftrightarrow}^0 = 0$, $v_\bullet^0 = \tilde{v}_\bullet^0 = 0$, $v_+^0 = \tilde{v}_+^0 = 0$, $\bar{h}^0 = 0$ and $\bar{w}^0 = 0$.

while convergence criterion not met, for $k=0, 1, \dots$ **do**

```

1- $\bar{h}^{k+1} = \text{prox}_{\sigma(\frac{1}{2}\|\cdot\|_2^2 + \cdot, z)}(\bar{h}^k + \sigma(\tilde{u}^k)), \text{prox}_{\sigma(\frac{1}{2}\|\cdot\|_2^2 + \cdot, z)}(\alpha) = \frac{\alpha - \sigma z}{1 + \sigma},$ 
2- $\bar{w}^{k+1} = \bar{w}^k + \sigma(\mathcal{O}_\downarrow^* \tilde{v}_\downarrow^k + \mathcal{O}_{\leftrightarrow}^* \tilde{v}_{\leftrightarrow}^k + \mathcal{O}_\bullet^* \tilde{v}_\bullet^k + \mathcal{O}_+^* \tilde{v}_+^k - \mathbb{D} \tilde{u}^k),$ 
3- $u^{k+1} = u^k - \tau(\bar{h}^{k+1} - \mathbb{D}^* \bar{w}^{k+1}),$ 
Use formula  $\text{Prox}_{\theta\alpha_0|\cdot|_1}(\alpha)(i, j) = (1 - \frac{\theta\alpha_0}{\max\{\|\alpha(i, j)\|_2, \theta\alpha_0\}})\alpha(i, j)$  for steps 4-7
below:
4- $v_\downarrow^{k+1} = \text{Prox}_{\tau|\cdot|_1}(v_\downarrow^k - \tau(\mathcal{O}_\downarrow \bar{w}^{k+1})),$ 
5- $v_{\leftrightarrow}^{k+1} = \text{Prox}_{\tau|\cdot|_1}(v_{\leftrightarrow}^k - \tau(\mathcal{O}_{\leftrightarrow} \bar{w}^{k+1})),$ 
6- $v_\bullet^{k+1} = \text{Prox}_{\tau|\cdot|_1}(v_\bullet^k - \tau(\mathcal{O}_\bullet \bar{w}^{k+1})),$ 
7- $v_+^{k+1} = \text{Prox}_{\tau|\cdot|_1}(v_+^k - \tau(\mathcal{O}_+ \bar{w}^{k+1})),$ 
8- $\tilde{u}^{k+1} = 2u^{k+1} - u^k,$ 
9- $\tilde{v}_\downarrow^{k+1} = 2v_\downarrow^{k+1} - v_\downarrow^k,$ 
10- $\tilde{v}_{\leftrightarrow}^{k+1} = 2v_{\leftrightarrow}^{k+1} - v_{\leftrightarrow}^k,$ 
11- $\tilde{v}_\bullet^{k+1} = 2v_\bullet^{k+1} - v_\bullet^k,$ 
12- $\tilde{v}_+^{k+1} = 2v_+^{k+1} - v_+^k.$ 

```

end

Algorithm 3: The algorithm for denoising

5 Numerical results

5.1 Compressed MRI data reconstruction problem

In this section, we run the proposed method on some test images and compare them with other well-known models. We choose a 256×256 in-vivo MR image, which is a T_2 -weighted axial brain scan. The image was chosen based on its variety of displayed feature and the fraction of FOV they occupy (about 46 percent of the FOV). Based on these sizes, we chose to sample 10 percent of the k -space of the images. The random sampling masks were chosen as variable-density patterns, with more samples in the low-frequency area (in a neighborhood of the origin) and fewer ones towards the periphery of the k -space. We used the Sparse MRI MATLAB package [18] to generate the

masks. We compared the performance of our method with the baseline zero-filling solution, TV + wavelet model and the TGV model. The TV + wavelet model solves the unconstrained minimization problem

$$\min_{u \in U} \frac{1}{2} \|\mathcal{F}_m u - b\|_2^2 + \beta \|\Psi u\|_1 + \alpha \|\nabla u\|_{1,2}.$$

This model was first introduced in [19] and then was further investigated in [20, 14, 16]. We followed the experiment setup proposed in these papers in our tests. In particular, we set $\alpha = 0.001$ and $\beta = 0.035$. The TGV method has the form

$$\min_{u \in U} \frac{1}{2\lambda} \|\mathcal{F}_m u - b\|_2^2 + \alpha_1 \|\nabla u - v\|_{1,2} + \alpha_0 \|\mathcal{E}v\|_{1,F}$$

and was proposed in [15]. Again, in order to stay true to the original paper, we chose the parameters as in [15]. In particular, we set $\lambda = 8 \times 10^{-5}$, $\alpha_1 = 1$, and $\alpha_0 = 2$. and the Chambolle–Pock algorithm is applied to them. In the proposed model (5), we also fix $\lambda = 8 \times 10^{-5}$ and $\beta = 0.75$ across the simulations. The 4-tap Daubechies wavelet transform was used the experiment and was implemented through the Sparse MRI package. The algorithm ran for 500 iterations.

We compare the quality of output images of each method based on SNR, SSIM, and HFEN indices. Let u_0 be the ground-truth image. The SNR is defined as

$$\text{SNR}(u) := 20 \log \frac{\|u_0\|_2}{\|u - u_0\|_2}$$

and the high SNR is related to low MSE. The SSIM index is defined as

$$\text{SSIM}(u) := \frac{(2\mu_u \mu_{u_0} + c_1)(2\sigma_{uu_0} + c_2)}{(\mu_u^2 + \mu_{u_0}^2 + c_1)(\sigma_u^2 + \sigma_{u_0}^2 + c_2)},$$

where μ_u is the average of u , σ_u is the standard deviation of u , σ_{uu_0} is the cross-covariance for u and u_0 , and $c_1 = (0.01 \times 255)^2$ and $c_2 = (0.03 \times 255)^2$ are regularizing constants. Another newly-proposed error metric is the HFEN (high frequency error norm), which aims at quantifying the quality of reconstructed edges and fine features [22]. This index is defined as

$$\text{HFEN}(u) := \frac{\|\text{LoG}(u) - \text{LoG}(u_0)\|_2}{\|\text{LoG}(u_0)\|_2},$$

where LoG is a rotationally symmetric Laplacian of Gaussian filter with a kernel size of 15×15 pixels and standard deviation of 1.5 pixels. We note that the perfect reconstruction has $\text{SNR} = \infty$, that $\text{SSIM} = 1$ and that $\text{HFEN} = 0$. We should also remark that none of these metrics perfectly quantify the visual perception of reconstructed images; although rarely the case, it might

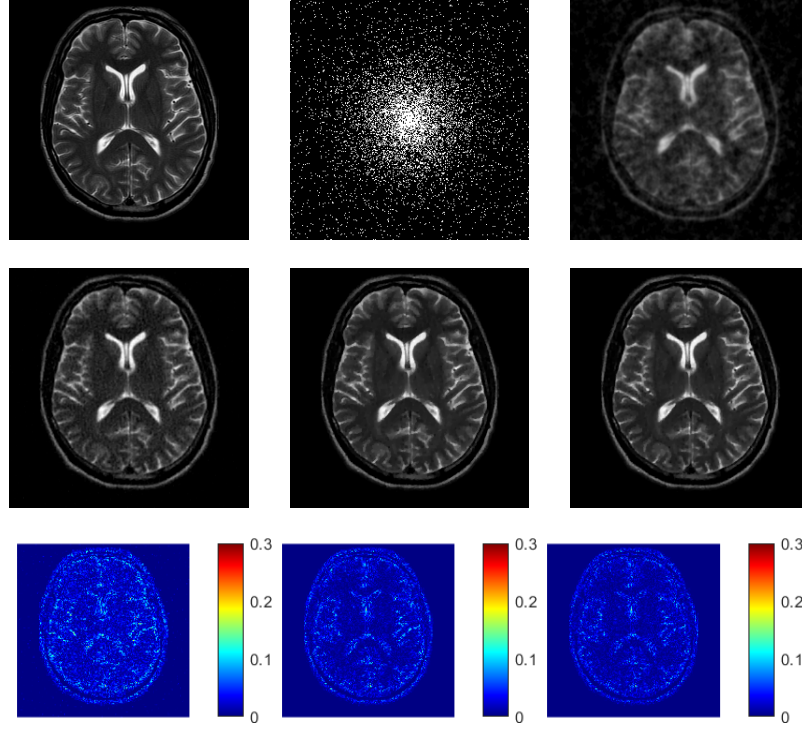


Figure 1: T_2 -weighted axial brain scan results. Top row: ground truth (left), sampling mask at rate %10 (middle), zero-filling solution (right). Middle row: reconstruction by TV+W (left), TGV (middle) and the proposed method (right). Bottom row: error maps for TV+W (left), TGV (middle) and the proposed method (right).

just happen that between two reconstructions, one has a higher value in all of these metrics, yet the other is visually more faithful to the ground-truth.

In Figure 1, the reconstruction results for the T_2 -weighted brain scan are presented. As expected, the zero-filling solution yields the worst reconstruction with too many incoherent artifacts. The TV+W model removes many of these artifacts but still has low structural similarity to the ground truth, which is also easily observed in the reconstructed image. The TGV model significantly improves upon TV + wavelet model as can be perceived visually and quantitatively (Table 1). The proposed framework moderately improves upon the TGV solution as attested by all three error measures and the error maps (the third row in Figure 1), which shows fewer errors in $[0, 0.1]$ interval.

Table 1: Quantitative comparison between various reconstructions for the T_2 -weighted brain scan.

Method	SNR [dB]	SSIM	HFEN
Proposed	20.71	0.9046	0.1013
TGV	20.07	0.9037	0.0995
TV+W	17.58	0.5113	0.2259
ZF	8.83	0.2733	0.7574

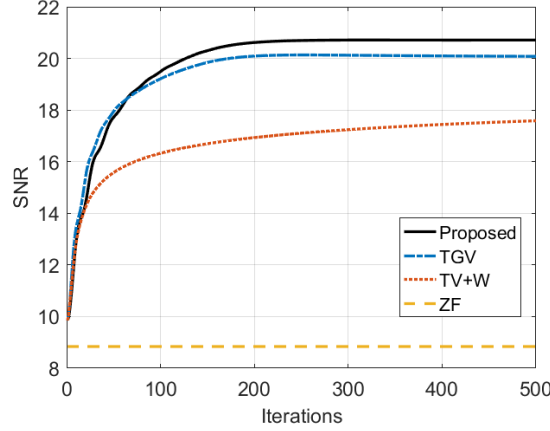


Figure 2: SNR plots for the T_2 -weighted brain scan reconstructions.

5.2 Denoising problem

Consider denoising problem (4.2), assume that x is a clean grayscale image and z is a noisy image that is created by adding Gaussian noise with variance 0.1 to the clean image x . In fact, we create an artificial noisy image by means of adding noise to the original image. Here we solve the denoising problem for the **pirate** test image and compare the results of the proposed model, with Condat [10], TGV [5], and Hosseini's model [13]. Table 2 shows the denoised results. It can be seen that the proposed method has better performance in decreasing noise and preserving the details in comparison with the other methods. On the other hand, Figure 3 shows SSIM and PSNR values with respect to the steps. As a result, it can be seen that the proposed model has better values of SSIM and PSNR, at the solutions that are obtained in the convergence.

Table 2: Results of Pirate denoising problem.

Reference	Noisy	Condat
		
TGV	Hosseini	Proposed
		

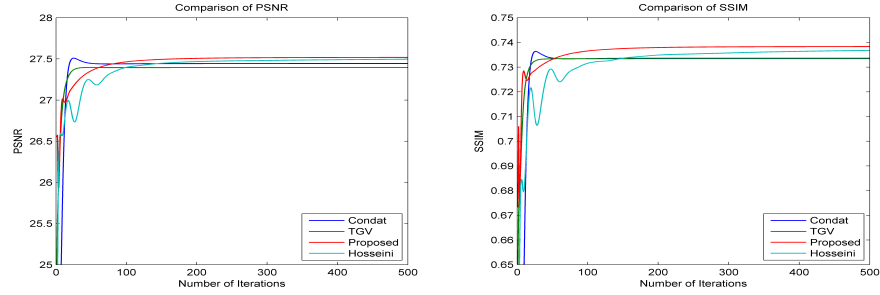


Figure 3: PSNR (left) and SSIM (right) values versus number of iterations for denoising problem.

6 Conclusions

In this paper, a new discretization of total variation functional was proposed. A four directions discrete gradient was used in the structure of the method. Moreover, some linear operators were inserted to the constraints. One of differences of the proposed method with the method in [13], is the new inserted projecting operators to the corners of the pixels. The proposed method

was employed to formulate some image processing problems; reconstructing highly undersampled MR images, and image denoising. The Chambolle–Pock algorithm was applied for the numerical simulations. The wavelet transform was chosen in the formulation of compressed sensing MRI problem and it was compared with zero-filling solution, TGV, and TV+W. In addition, for the image denoising problem, the proposed model was compared with some state of the art methods. In both numerical experiments, the new methods illustrated better quantitative and qualitative results.

References

1. Abergel, R. and Moisan, L. *The Shannon total variation*, J. Math. Imaging Vision 59 (2017), 341–370.
2. Alter, F., Caselles, V. and Chambolle, A. *Evolution of characteristic functions of convex sets in the plane by the minimizing total variation flow*, Interfaces Free Bound. 7 (2005), 29–53 .
3. Beck, A. *First-order methods in optimization*, SIAM, Philadelphia, 2017.
4. Bredies, K. *Recovering piecewise smooth multichannel images by minimization of convex functional with total generalized variation penal*, A. Bruhn et al. eds., Global Optimization Methods, Lecture Notes in Computer Science 8293, (2014), 44–77.
5. Bredies, K., Kunisch, K. and Pock, T. *Total generalized variation*, SIAM J. Imaging Sci. 3 (2010), 492–526.
6. Buades, A., Coll, B. and Morel, J.M. *Image denoising methods. A new nonlocal principle*, SIAM Rev. 52 (2010), 113–147.
7. Chambolle, A., Caselles, V., Novaga, M., Cremers, D. and Pock, T. *An introduction to total variation for image analysis*, Theoretical Foundations and Numerical Methods for Sparse Recovery, De Gruyter, Radon Series Comp. Appl. Math. 9 (2010), 263–340.
8. Chambolle, A., Levine, S.E. and Lucier, B.J. *An upwind finite-difference method for total variation based image smoothing*, SIAM J. Imaging Sci., 4 (2011), 277–299.
9. Chambolle, A. and Pock, T. *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vis. 40 (2010), 120–145.
10. Condat, L. *A primal-dual splitting method for convex optimization involving Lipschitzian, proximable and linear composite terms*, J. Optim. Theory Appl. 158, 460–479 (2013).

11. Ghazel, M., Freeman, G.H. and Vrsnay, E.R. *Fractal image denoising*, IEEE Trans. Image Process. 12 (2003), 1560–1578.
12. Guo, W., Kin, J. and Yin, W. *A new detail-preserving regularization scheme*, SIAM J. Imaging Sci. 7 (2014), 1309–1334.
13. Hosseini, A. *New discretization of total variation functional for image processing tasks*, Signal Process. Image Commun. 78 (2019), 62–76.
14. Huang, J., Zhang, S. and Metaxas, D. *Efficient MR image reconstruction for compressed MR imaging*, Med. Image Anal. 15 (2011), 670–679.
15. Knoll, F., Bredies, K., Pock, T. and Stollberger, R. *Second order total generalized variation (TGV) for MRI*, Magn. Reson. Med. 65 (2011), 480–491.
16. Kou, G., Zhao, Y., Peng, Y. and Shi, Y. *A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data*, IEEE J-STSP. 4 (2010), 288–297.
17. Lore, K.G., Akintayo, A. and Sarkar, S. *LLNet: A deep autoencoder approach to natural low-light image enhancement*, Pattern Recognit, 61 (2015), 650–662.
18. Lustig, M. *Michael Lustig homepage*, <https://people.eecs.berkeley.edu/~mlustig/index.html>, Online, accessed February 2019.
19. Lustig, M., Donoho, D. and Pauly, J. *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magn. Reson. Med. 58 (2007), 1182–1195.
20. Ma, S., Yin, W., Zhang, Y. and Chakraborty, A. *An efficient algorithm for compressed MR imaging using total variation and wavelets*, IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2008), 1–8.
21. Perona, P. and Malik, J. *Scale-space and edge detection using anisotropic diffusion*, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990), 629–639.
22. Ravishankar, S. and Bresler, Y. *MR image reconstruction from highly undersampled k-space data by dictionary learning*, IEEE Trans. Med. Imaging, 30 (2011), 1029–1041.
23. Ravishankar, S. and Bresler, Y. *Efficient blind compressed sensing using sparsifying transforms with convergence guarantees and application to magnetic resonance imaging*, SIAM J. Imaging Sci. 8 (2015), 2519–2557.
24. Rudin, L., Osher, S. and Fatemi, E. *Nonlinear total variation based noise removal algorithms*, Physica D. 60 (1992), 259–268.

25. Wang, N., Tao, D., Gao, X., Li, X. and Li, J. *A comprehensive survey to face hallucination*, Int. J. Comput. Vis. 106 (2014), 9–30.
26. Weickert, J. Anisotropic diffusion in image processing, Teubner, Stuttgart, 1998.



Maximum probability O-D matrix estimation in large-sized networks

M. Abareshi*

Abstract

We propose a maximum probability model to estimate the origin-destination trip matrix in the networks, where the observed traffic counts of links and the target origin-destination trip demands are independent discrete random variables with known probabilities. The problem is formulated by using the least squares approach in which the objective is to maximize the probability that the sum of squared errors between the estimated values and the observed (target) ones does not exceed a pre-specified threshold. An enumeration solution approach is proposed to solve the problem in small-sized networks, while a normal approximation based on the central limit theorem is applied in large-sized networks to transform the problem into a deterministic nonlinear fractional model. Some numerical examples are provided to illustrate the efficiency of the proposed method.

AMS(2010): Primary 90B06, Secondary 90B15, 90C06, 90C32.

Keywords: Transportation; Origin-destination trip matrix; Least squares approach; Probabilistic traffic counts; Fractional programming.

1 Introduction

Origin-destination ($O-D$) trip matrix estimation problem is one of the most important issues in transportation networks, in which the number of trips between each $O-D$ pair is evaluated and inserted in a matrix called $O-D$ trip matrix. The traditional methods used for investigating the $O-D$ trip matrix, including such direct measurement methods as roadside interviews and flagging techniques, are so costly and mistakable. Conducting these methods to update the $O-D$ matrix information is not easily possible due

*Corresponding author

Received 3 April 2019; revised 19 February 2020; accepted 14 July 2020

Maryam Abareshi

Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences,
Hakim Sabzevari University, Sabzevar, Iran. e-mail: abareshi66@gmail.com

to limitations of budgeting and implementation problems. In recent years, using such available information as the observed traffic counts of links or a target $O - D$ matrix, obtained by a sample survey or an outdated model, the problem is formulated as a mathematical model that estimates the $O - D$ matrix in a cheaper and faster way.

Determining the $O - D$ matrix using the traffic counts of links can be interpreted as the inverse of the traffic assignment (TA) problem. In the TA problem, the demands of $O - D$ pairs are given previously and assigned to the paths connecting them so that the total users' cost is minimized. As a result, the flows on paths and the traffic counts of links would be determined. In the $O - D$ matrix estimation problem, the purpose is to estimate the $O - D$ demands by using the observed traffic counts of links. The solution methods for this purpose are divided into two basic classes including (see [5]):

1. The proportional assignment models, in which the congestion effects are not considered and the only reasons to pick and choose a route are the traveler and route characteristics.
2. The equilibrium assignment models, in which the traffic congestion has considerable influences on choosing routes. In this case, the Wardrop's first principle will satisfy that the traffic network is in equilibrium when no traveler can achieve a lower travel cost by switching to another route; see [34].

Considering either the proportional or equilibrium assignment, there are four main mathematical models to estimate the $O - D$ matrix:

1. Entropy maximization model [33, 36, 18], in which by introducing a logarithm-based function, the most unbiased $O - D$ trip matrix consistent with the existing evidence is estimated.
2. Least squares approach [9, 11] with the purpose of minimizing the sum of squared deviations between the estimated (true) variables and the observed (target) ones.
3. Maximum likelihood approach [29], which maximizes the likelihood of observing the target $O - D$ matrix and the observed link traffic counts conditional on the estimated (true) $O - D$ matrix.
4. Bayesian inference approach [22], in which Bayes' theorem is applied to combine information obtained from both the target $O - D$ matrix and the observed traffic counts of links to estimate the true $O - D$ matrix.

Abareshi, Zaferanieh, and Keramati [3] studied a maximum entropy (ME) path flow estimator for disaggregated flows between $O - D$ pairs with a pre-specified level for each disaggregation. Sun et al. [32] applied the ME approach on a subnetwork $O - D$ matrix estimation model by using the total traffic generations (attractions) along with some fixed $O - D$ demands of

the subnetwork as the constraints. Applying a convex combination method, the resulted nonlinear problem was converted to the classical linear transportation problem where a tabular method was implemented to solve it. Xie, Kockelman, and Waller [37] proposed an elastic $O - D$ flow table estimation problem for subnetwork analysis. They proposed a combined maximum entropy-least squares estimator, by which the $O - D$ flows were distributed over the subnetwork in terms of the maximum entropy principle, while demand function parameters were estimated so that the least sum of squared errors was achieved.

Sun et al. [31] introduced two bi-level models to reconstruct $O - D$ demands under congested network using both the observed link and route travel times. Their proposed models aimed to minimize the distances between the observed and estimated traffic information ($O - D$, link and route travel times) in the upper level, and optimize the stochastic user equilibrium (*SUE*) model in the lower level, in which no driver can unilaterally change routes to improve his/her *perceived*, rather than actual, travel times.

In many real applications, the precise values of the observable data as link traffic counts or target $O - D$ trips in a network might be unavailable. For example, the observed number of travelers moving between cities may differ in various situations due to the weather conditions or coming the rush hours; consequently, some measurement errors may occur with known or unknown probability distributions. Therefore, it is desirable to treat the observed information as random or time-dependent variables with certain or uncertain parameters; see [13, 26]. In such cases, the use of statistical modeling to consider explicitly the presence of measurement and sampling errors in the observed information is developed.

Hoang, Vu, and Lo [16] formulated the information-based stochastic user equilibrium dynamic traffic assignment problem in a network, where the real-time information accounted for the (stochastic) uncertainty in demand and network capacity. Jones et al. [17] developed new methods for network assessment and control. Applying partial sensor and survey data while imposing equilibrium conditions during the data collection phase, they considered explicit account of demand variability and uncertainty.

Ma and Qian [20], using day-to-day traffic data collected over many years, estimated the mean and variance/covariance matrix of the $O - D$ demands. They [21] also presented a data-driven framework that estimates the day-to-day dynamic $O - D$ demands using high-granular traffic counts and speed data collected over many years. Pitomberia-Neto, Loureiro, and Carvalho [27] estimated the $O - D$ flows using link traffic volumes over a sequence of days and applied a dynamic hierarchical Bayesian model for estimating the day-to-day $O - D$ demands. Ching, Scholtes, and Zhang [13] assumed that the demands between nodes (zones) over a fixed period of time are formulated as independent random variables with unknown means. They considered both Poisson and normal distributions as the density functions

for the random travel demands and applied some numerical algorithms to estimate the optimal solution.

Shao et al. [28] proposed a bi-level optimization problem using a weighted least squares model in the upper level and a TA model in the lower level to estimate the peak hour $O - D$ traffic demand variations from day-to-day hourly traffic counts throughout the whole year. Abareshi, Zaferanieh, and Safi [4] introduced a Markov chain $O - D$ matrix estimation problem in which the average time between two incoming streams to or outgoing streams from nodes in consecutive time periods was considered as a Markov chain. Besides, a normal distribution with pre-determined parameters in each period was used for traffic counts of links. They proposed a bi-level model where in the upper level, the network flow pattern with the maximum probability in the n th period was to be estimated, while a traffic assignment problem considering the equilibrium conditions was solved in the lower level.

Random variables in location problems have also received lots of attention, recently. Abareshi and Zaferanieh [1] considered prior probabilities to serve the demands of nodes by different facilities and introduced a bi-level model by applying the minimum information (MI) approach to determine the most probable allocation solution in the customer's point of view. Berman and Wang [7] studied locating p medians to serve clients with discrete probabilistic demand weights, where the purpose was maximizing the probability that the total weighted distance does not exceed a given threshold value. To overcome the difficulty of evaluating the objective function, they [7] suggested to use a normal approximation of the problem based on the central limit theorem (CLT) when the size of the network is large enough. Berman and Wang [6] also considered four probabilistic network location problems with independent discrete demand weights and proposed efficient algorithms to solve the problem. Abareshi and Zaferanieh [2] studied the 1-median problem in the case, where the demand weights of nodes and the travel times of links were both discrete random variables.

In this paper, the observed traffic counts of links and the entries of the target $O - D$ matrix are considered to be independent random variables, where unlike [13], the corresponding probabilities are known. In this case, the squared errors between the estimated and the observed values are also independent random variables. Therefore, instead of minimizing the sum of squared errors, it is suggested to maximize the probability that this amount is less than or equal to a pre-threshold; see [6, 7]. Indeed, a predetermined bound is given for the estimated sum of squared errors. The purpose is to estimate the traffic flows on paths in an equilibrium approach to maximize the probability that the sum of squared errors between the estimated and observed (target) values does not exceed the given upper bound.

The problem is first examined for small-sized networks and a mixed-integer quadratically constrained ($MIQC$) model is proposed, which could be solved by using semidefinite programming (SDP) to reformulate the model as a quadratic convex problem; see [15]. We also propose an enumeration

method to solve the problem and compare the results with the ones obtained by *MIQC* methods. Then, the *CLT* is applied to the large-sized networks whereby the *MIQC* model is transformed into a nonlinear fractional programming (*NLFP*) problem. The resulted problem would be solved by such appropriate methods as parametric algorithm or linearization approaches; see [30]. The main contributions of the paper are summarized below:

- A maximum probability approach is proposed to estimate the $O - D$ trip matrix in a probabilistic network in which the observed link traffic counts and the target $O - D$ matrix are discrete random variables with known probabilities
- The problem is formulated as an *MIQC* model being solved via both *MIQC* methods and an enumeration approach in small-sized networks, while a normal approximation based on the *CLT* is applied to large-sized networks to transform the problem into a nonlinear fractional programming model.
- Three different cases are investigated by which the problem in large-sized networks is solved via the parametric algorithm, generating a sequence which superlinearly converges to the solution.
- Providing some numerical examples in both small and large-sized networks, the validity of the model as well as the solution approach is verified.

The rest of the paper is organized as follows: Problem formulation and the proposed *MIQC* model as well as the enumeration approach are given in Section 2. In Section 3, the *CLT* is applied to find the solution in large-sized networks, which results in an *NLFP* problem. In Section 4, the solution approach for the resulted fractional model is stated. In Section 5, some numerical examples are provided to examine the added value of the proposed methods. Summary and conclusions are given in the last section.

2 Problem formulation

Let $G = (N, E)$ be a network with the node set N and the link set E . The observed traffic counts of links as well as the entries of the target $O - D$ matrix are assumed to be independent random variables with known discrete probability distributions. Let $State = \{1, 2, \dots, H\}$ be the set of all realizations of the network. The vectors of the link traffic counts and target $O - D$ matrix in state $h \in State$ are, respectively, denoted by $\bar{V}^h$ and $\bar{X}^h$. Therefore, $\bar{v}_e^h$ and $\bar{x}_{rs}^h$ denote the observed traffic counts of link $e \in E$ and the target number of trips between the $O - D$ pair $(r, s) \in N \times N$ in the state h .

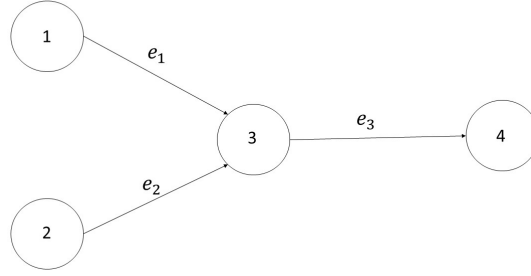


Figure 1: Small network

Note that if there is neither origin nor destination in the network, the traffic counts of links cannot be treated independently. For example, in Figure 1, the flow of link e_3 is equal to the sum of the traffic counts of links e_1 and e_2 , that is, $v_{e_1} + v_{e_2} = v_{e_3}$. Then counting traffic on the link e_3 is redundant, while two counts v_{e_1} and v_{e_2} are independent. Moreover, if node 3 is considered as an origin or a destination, then flows in the network are independent values. Therefore, the expression $v_{e_1} + v_{e_2} = v_{e_3}$ is not satisfied; see [35].

Ching, Scholtes, and Zhang [13] considered the traffic volumes between different zones ($O - D$ pairs) over a fixed period of time as independent random variables with unknown means. In this paper, due to the assumption of existence of origin and destination nodes, both the link traffic counts and the target $O - D$ matrix components are assumed to be independent random variables. A threshold D is given and the purpose is to estimate the $O - D$ matrix that maximizes the probability that the sum of squared errors between the estimated and the observed information is less than or equal to D .

The mathematical formulation of the maximum probability least squares (*MPLS*) problem to estimate the flow pattern in a probabilistic network is proposed as follows:

$$\max_f Z(f) = P_r\left(\sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2 \leq D\right), \quad (1)$$

$$s.t. \quad \sum_{rs} \sum_{k \in K_{rs}} f_{rs}^k \delta_{e,k} = v_e \quad \text{for all } e \in E, \quad (2)$$

$$\sum_{k \in K_{rs}} f_{rs}^k \delta_{k,rs} = x_{rs} \quad \text{for all } (r, s) \in N \times N, \quad (3)$$

$$f_{rs}^k \geq 0 \quad \text{for all } k \in K_{rs}, (r, s), \quad (4)$$

where $\bar{v}_e$ and $\bar{x}_{rs}$ are, respectively, the probabilistic observed traffic count of link e and target demand between $O - D$ pair (r, s) . Also, the decision

variables v_e and x_{rs} represent the corresponding estimated values. The set K_{rs} includes the K -shortest paths connecting $O - D$ pair (r, s) , which is determined by Eppstein's K -shortest path ranking algorithm; see [14]. The variable f_{rs}^k denotes the traffic flow on the path k connecting $O - D$ pair (r, s) . The notation $\delta_{e,k}$ is the arc-path incident indicator, that is, $\delta_{e,k} = 1$ if the link e is a part of path k and zero, otherwise. Similarly, $\delta_{k,rs} = 1$ if the path k is used to connect the $O - D$ pair (r, s) and zero otherwise. Constraint (2) states that the estimated traffic count of link e equals the sum of flows of all paths passing through this link. Besides, by constraint (3), the travel demand between each $O - D$ pair (r, s) equals the sum of flows on all paths connecting them.

One of the fundamental methods to estimate the solution in probabilistic cases, is using the expected value model (E -model) that converts the probabilistic model into a deterministic problem by applying the expected values of the objective function or constraints; see [12]. Indeed in real applications, applying this approach may be useless and result in ineffective solutions; see the following example.

Small Example. Consider the network with two links, given in Figure 2. There are two paths including links e_1 and e_2 connecting $O - D$ pair $(1, 2)$. Two realizations for the network with probabilities 0.7 and 0.3 are specified. The target $O - D$ matrix and the observed link traffic counts in the first and second realizations are assumed to be $\bar{x}_{1,2}^1 = 6.5$, $\bar{v}_{e_1}^1 = 3$, $\bar{v}_{e_2}^1 = 2$ and $\bar{x}_{1,2}^2 = 7.5$, $\bar{v}_{e_1}^2 = 4$, $\bar{v}_{e_2}^2 = 5$, respectively.

To estimate the optimal flow pattern with the least squared errors, either each of the distinct realizations or the expected values of the observed variables, instead of their random amounts, may be considered. The optimal flow pattern applying the least squares approach in the first state is $v_{e_1}^* = 3.5$, $v_{e_2}^* = 2.5$ with the total sum of squared errors 0.75 in the first state and 8.75 in the second one. Also the optimal solution with the least squared errors in the second state is $v_{e_1}^{**} = 3.5$, $v_{e_2}^{**} = 4.5$ with the total sum of squared errors 6.75 in the first state and 0.75 in the second one.

For the first state optimal solution, $v_{e_1}^* = 3.5$, $v_{e_2}^* = 2.5$, the probability of the total sum of the squared errors being less than 1 is 0.7, while this value for the second state optimal solution, $v_{e_1}^{**} = 3.5$, $v_{e_2}^{**} = 4.5$, is 0.3. If the expected values of $\bar{x}_{1,2}$, $\bar{v}_{e_1}$, and $\bar{v}_{e_2}$ are used, then the problem can be written as follows:

$$\min_f Z_{exp}(f) = \sum_e (v_e - \mu_e)^2 + \sum_{rs} (x_{rs} - \mu_{rs})^2, \quad (5)$$

subject to constraints (2), (3), and (4). The notations μ_e and μ_{rs} are used for the expected values of random variables $\bar{v}_e$ and $\bar{x}_{rs}$, respectively. Let the

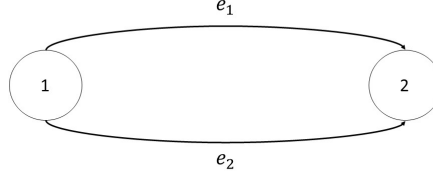


Figure 2: Network with two links

optimal flow vector of problem (5) be denoted by f_{exp}^* . Then, the total sum of squared errors in the first state is 1.47 and in the second one is 4.67. Hence the total sum of squared errors is larger than 1 in both cases.

If the purpose is to estimate a solution with the least sum of squared errors in both states simultaneously, then f_{exp}^* is preferred to the solutions of the first and second states. But, if there is a pre-determined upper bound $D = 1$ for the total sum of squared errors, then the solution $v_{e_1}^* = 3.5$, $v_{e_2}^* = 2.5$ satisfies this condition with probability 0.7 and obviously is preferred to the other ones.

Let $Z_h(f) = \sum_e (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2$, for each state $h \in State$, where v_e and x_{rs} are defined by constraints (2) and (3), respectively. We define the characterization function $Y_h(f)$ as follows:

$$Y_h(f) = \begin{cases} 1 & Z_h(f) \leq D, \\ 0 & otherwise. \end{cases}$$

Therefore problem (1) is rewritten as follows:

$$\max_f Z(f) = \sum_h Y_h(f) P_r^h, \quad (6)$$

where P_r^h is the probability of state h . Let the optimal path flow vector for the following least squares problem, in state $h \in State$, be represented by f_h^* :

$$\min_f Z_h(f) = \sum_e (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2,$$

subject to constraints (2), (3), and (4). The flow vector f_h^* can be estimated by using such solution methods for the least squares $O - D$ matrix estimation problem as Lagrangian dual approach or active-set constraints; see [10, 11, 24].

Definition 1. The set of indices $h \in State$ with $Z_h(f_h^*) \leq D$ is defined as the candidate set and denoted by $C_{set} = \{h \in State; Z_h(f_h^*) \leq D\}$.

If $h \notin C_{set}$, then $Y_h(f_h^*) = 0$, and consequently $Y_h(f) = 0$ for all vectors f . The definition of the characterization function $Y_h(f)$ for each vector f results in the following constraint using the binary variable y_h , where y_h is used instead of $Y_h(f)$ to simplify the model:

$$\sum_{e \in E} (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2 - D \leq M(1 - y_h), \quad (7)$$

where M is chosen sufficiently large. Inequality (7) together with the objective function $\max_f Z(f) = \sum_h y_h P_r^h$ is equivalent to the definition of $Y_h(f)$. Therefore, the following *MIQC* problem is resulted:

$$\begin{aligned} MIQC : \quad & \max_f Z(f) = \sum_h y_h P_r^h, \\ s.t. \quad & \sum_{e \in E} (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2 - D \leq M(1 - y_h), \\ & \sum_{rs} \sum_{k \in K_{rs}} f_{rs}^k \delta_{e,k} = v_e \quad \text{for all } e \in E, \\ & \sum_{k \in K_{rs}} f_{rs}^k \delta_{k,rs} = x_{rs} \quad \text{for all } (r, s) \in N \times N, \\ & f_{rs}^k \geq 0 \quad \text{for all } k, (r, s), \quad y_h \in \{0, 1\} \quad \text{for all } h \in State. \end{aligned} \quad (8)$$

There are some solution approaches to solve problem (8). Galli and Letchford [15] studied the possibility of extending the reformulation approach, using the *SDP*, proposed in [8] for equality-constrained 0 – 1 quadratic programs, to the more general case of mixed-integer quadratically constrained quadratic (*MIQCQ*) problems. The proposed reformulation strengthened the continuous relaxation of the problem, while the optimal solution remained unchanged. Misener and Floudas [23] studied the global optimization of *MIQCQ* problems by using a branch and bound algorithm applying a relaxation approach. They integrated the edge-concave relaxation with the piecewise-linear programming to tightly underestimate the *MIQCQ* problems.

We propose an enumeration solution approach to solve problem (1) and compare the results with the ones obtained by solving *MIQC* problem (8). First, it is required to find the vectors f that minimize the maximum of the functions $Z_h(f)$, where indices h belong to the subsets of C_{set} . Let $S_i \subseteq C_{set}$ be a subset with cardinality i , the optimal solution of the following problem should be determined:

$$\min_f \max_{h \in S_i} Z_h(f) = \sum_{e \in E} (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2, \quad (9)$$

subject to constraints (2), (3), and (4). Problem (9) could be reformulated as a quadratically constrained (QC) problem:

$$\begin{aligned}
 QC : \quad & \min_f \quad \theta, \\
 s.t. \quad & \sum_{rs} \sum_{k \in K_{rs}} f_{rs}^k \delta_{e,k} = v_e \quad \text{for all } e \in E, \\
 & \sum_{k \in K_{rs}} f_{rs}^k \delta_{k,rs} = x_{rs} \quad \text{for all } (r, s) \in N \times N, \\
 & \sum_{e \in E} (v_e - \bar{v}_e^h)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs}^h)^2 \leq \theta \quad \text{for all } h \in S_i, \\
 & f_{rs}^k \geq 0 \quad \text{for all } k, (r, s).
 \end{aligned} \tag{10}$$

Problem (10) would be solved by using such nonlinear programming methods as Lagrangian dual approach or other quadratically constrained programming techniques; see [19, 25]. Let the optimal solution of problem (10) be $f_{S_i}^*$. If $Y_h(f_{S_i}^*) = 1$ for all $h \in S_i$, then $Z(f_{S_i}^*) = \sum_{h \in S_i} P_r^h$. By comparing the values of $Z(f_{S_i}^*)$ for all subsets $S_i \subseteq C_{set}$, the optimal solution of problem (1) is determined. Note that if a solution f^* is obtained, then any subset S_i that the sum of its elements' probabilities is less than $Z(f^*)$ can be ignored.

In Section 5, the solution of problem (1) is obtained for Yang's network by both solving the resulted *MIQC* problem (8) and applying the proposed enumeration method using the *QC* problem (10). In Section 3, a normal approximation is applied to investigate the problem in large-sized networks, which results in an *NLFP* problem equivalent to problem (1).

3 Normal approximation

In this section, the *MPLS* problem in large-sized networks, having an enormous number of links and $O - D$ pairs, is studied. Let the expected mean and variance of the probabilistic observed traffic count $\bar{v}_e$ be, respectively, denoted by μ_e and σ_e^2 , and the expected mean and variance of $\bar{x}_{rs}$ be represented by μ_{rs} and σ_{rs}^2 . Using Lemmas 1 and 2, first the mean and variance of term $\sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2$ are determined. Then applying the *CLT*, problem (1) is reduced to an *NLFP* problem.

Lemma 1. The mean of term $(v_e - \bar{v}_e)^2$ is $\hat{\mu}_e = (v_e - \mu_e)^2 + \sigma_e^2$, and its variance is $\hat{\sigma}_e^2 = 4v_e^2\sigma_e^2 + K_e$, where $K_e = \text{var}(\bar{v}_e^2)$.

Proof. Using the expansion of term $(v_e - \bar{v}_e)^2$ together with equality $\sigma_e^2 = E(\bar{v}_e^2) - \mu_e^2$ yields:

$$E(v_e - \bar{v}_e)^2 = E(v_e^2 - 2v_e\bar{v}_e + \bar{v}_e^2) = v_e^2 - 2v_e\mu_e + \sigma_e^2 + \mu_e^2 = \hat{\mu}_e$$

and

$$\text{var}(v_e - \bar{v}_e)^2 = 0 + 4v_e^2\sigma_e^2 + \text{var}(\bar{v}_e^2).$$

□

Lemma 2. The mean and variance of term $(x_{rs} - \bar{x}_{rs})^2$ are $\hat{\mu}_{rs} = (x_{rs} - \mu_{rs})^2 + \sigma_{rs}^2$ and $\hat{\sigma}_{rs}^2 = 4x_{rs}^2\sigma_{rs}^2 + K_{rs}$, respectively, where $K_{rs} = \text{var}(\bar{x}_{rs}^2)$.

The proof of Lemma 2 is similar to Lemma 1 and has been omitted. Applying the *CLT* and noting the independence of the probabilistic variables $\bar{v}_e$ s and $\bar{x}_{rs}$ s, the term $W = \sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2$ has a normal distribution with the expected mean $\mu_W = \sum_e \hat{\mu}_e + \sum_{rs} \hat{\mu}_{rs}$ and variance $\sigma_W^2 = \sum_e \hat{\sigma}_e^2 + \sum_{rs} \hat{\sigma}_{rs}^2$. Hence,

$$\frac{\sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2 - \mu_W}{\sigma_W} \sim \text{normal}(0, 1).$$

Therefore,

$$\begin{aligned} & Pr\left(\sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2 \leq D\right) \\ &= Pr\left(\frac{\sum_{e \in E} (v_e - \bar{v}_e)^2 + \sum_{rs} (x_{rs} - \bar{x}_{rs})^2 - \mu_W}{\sigma_W} \leq \frac{D - \mu_W}{\sigma_W}\right) \\ &= \phi\left(\frac{D - \mu_W}{\sigma_W}\right), \end{aligned}$$

where ϕ is the cumulative distribution function of the standard normal distribution. Since ϕ is an increasing function, problem (1) would be rewritten as the following *NLFP* problem:

$$\max_f Z_{frac}(f) = \frac{D - \sum_e \hat{\mu}_e - \sum_{rs} \hat{\mu}_{rs}}{\sqrt{\sum_e \hat{\sigma}_e^2 + \sum_{rs} \hat{\sigma}_{rs}^2}}, \quad (11)$$

subject to constraints (2), (3), and (4), where variables $\hat{\mu}_e$, $\hat{\sigma}_e^2$, $\hat{\mu}_{rs}$ and $\hat{\sigma}_{rs}^2$ are the same as given in Lemmas 1 and 2. Problem (11) could be solved by using such nonlinear fractional programming methods as parametric algorithm or linearization approaches; see [30]. In the next section, a parametric method is employed to solve problem (11) in three cases.

4 Solution approach

Consider the following fractional programming problem:

Algorithm 1 Parametric method

-
- 1: Take $\lambda = \lambda_1$ such that $F(\lambda_1) \geq 0$. (There is such a λ_1 , since $F(0) \geq 0$ considering the assumption that $f(x) \geq 0$ for at least one $x \in S$.)
 - 2: Solve problem (13). If $|F(\lambda)| \leq \epsilon$, stop. Otherwise, go to Step (3).
 - 3: Set $\lambda = \frac{f(x^*)}{g(x^*)}$, where x^* is the optimal solution of problem (13) obtained in Step (2). Go to Step (2).
-

$$P : \max_x \{q(x) = \frac{f(x)}{g(x)} | x \in S\}, \quad (12)$$

where f and g are real-valued differentiable functions on $S \subseteq R^n$. The parametric method is used to solve problem (12) in the case that $S \subseteq R^n$ is a compact and nonempty set and functions $f, g : S \rightarrow R$ are continuous with $g(x) > 0$ for all $x \in S$ and $f(x) \geq 0$ for at least one $x \in S$; see [30]. The equivalent parametric problem is introduced as follows:

$$Q(\lambda) : \max_x \{f(x) - \lambda g(x) | x \in S\}, \quad (13)$$

where λ is a real-valued parameter. Let $F(\lambda)$ be the optimal value of the objective function in problem $Q(\lambda)$. The reader is referenced to [30] to see the proofs of the following lemmas and more details.

Lemma 3. Let $\bar{x}$ be the optimal solution to problem (12) and $\bar{\lambda} = \frac{f(\bar{x})}{g(\bar{x})}$. Then

1. $F(\lambda) < 0$ if and only if $\lambda > \bar{\lambda}$,
2. $F(\lambda) = 0$ if and only if $\lambda = \bar{\lambda}$,
3. $F(\lambda) > 0$ if and only if $\lambda < \bar{\lambda}$.

Corollary 1. If $F(\lambda) = 0$, then the optimal solution to problem (13) is also optimal to problem (12).

Lemma 4. The function $F(\lambda)$ is continuous, convex, and strictly decreasing on R .

Therefore, problem (12) is solved by determining the root of $F(\lambda) = 0$. It can be shown that the root is unique and nonnegative; see [30]. The methods used for solving the equation $F(\lambda) = 0$ will generate the solution algorithms for the *NLFP* problem (12). The solution steps are outlined in Algorithm 1. The sequence generated by Algorithm 1 superlinearly converges to the root of $F(\lambda) = 0$; see [30]. Next, we give the solution method for problem (11) by introducing three subproblems under specified conditions.

- 1: **The First case:** $D - \sum_e \hat{\mu}_e - \sum_{rs} \hat{\mu}_{rs} > 0$. In this case, the numerator and denominator of the objective function in problem (11) are both continuous, differentiable, and positive functions. We use the parametric method described in Algorithm 1 to solve problem (11). In Step (2) of Algorithm 1, the following problem should be solved:

$$\max_f D' - \sum_e (v_e - \mu_e)^2 - \sum_{rs} (x_{rs} - \mu_{rs})^2 - \lambda \sqrt{\sum_e 4v_e^2 \sigma_e^2 + \sum_{rs} 4x_{rs}^2 \sigma_{rs}^2} + k'', \quad (14)$$

subject to constraints (2), (3), and (4). In this problem, $D' = D - \sum_e \sigma_e^2 - \sum_{rs} \sigma_{rs}^2$ and $k'' = \sum_{rs} K_{rs} + \sum_e K_e$. Using such nonlinear constrained programming methods as penalty methods or augmented Lagrangian approaches, problem (14) could be solved; see [25]. However, if the first case is infeasible, the second one should be investigated.

- 2: **The Second case:** $D - \sum_e \hat{\mu}_e - \sum_{rs} \hat{\mu}_{rs} = 0$. In this case, the following least squares programming problem is considered:

$$\min_f \sum_e (v_e - \mu_e)^2 + \sum_{rs} (x_{rs} - \mu_{rs})^2, \quad (15)$$

subject to constraints (2), (3), and (4). Problem (15) could be solved by using such methods for least squares $O-D$ matrix estimation problems as Lagrangian dual approach or active-set constraints; see [10, 11, 24]. If the optimal value of problem (15) equals D' , then the obtained solution would be also optimal for problem (11) with the objective value equal to zero, else the third case should be considered.

- 3: **The third case:** $D - \sum_e \hat{\mu}_e - \sum_{rs} \hat{\mu}_{rs} < 0$. This case would be considered if neither the first nor the second case is feasible. Here, the parametric method is used with small modifications, to solve problem (12) in which $f(x)$ is negative while $g(x)$ is positive. Consider the problem:

$$Q'(\lambda) : \max_x \{f(x) + \lambda g(x) | x \in S\}. \quad (16)$$

Let $F'(\lambda)$ be the optimal value of problem (16). Then, Lemma 5, Corollary 2, and Lemma 6 are applied to find the optimal solution. For more detail see [30].

Lemma 5. Let $\bar{x}$ be an optimal solution to problem (12) in the case that the numerator is negative and $\bar{\lambda} = -\frac{f(\bar{x})}{g(\bar{x})}$. Then

- (a) $F'(\lambda) < 0$ if and only if $\lambda < \bar{\lambda}$,
- (b) $F'(\lambda) = 0$ if and only if $\lambda = \bar{\lambda}$,

(c) $F'(\lambda) > 0$ if and only if $\lambda > \bar{\lambda}$.

Proof. We only examine case (a), the other cases are similarly proved. Let $F'(\lambda) < 0$; then $f(x') + \lambda g(x') < 0$ for all $x' \in S$. Since $g(\cdot)$ is a positive function, then $-\frac{f(x')}{g(x')} > \lambda$ for all $x' \in S$. Therefore, $-\frac{f(\bar{x})}{g(\bar{x})} > \lambda$ and consequently $\bar{\lambda} > \lambda$.

Conversely, if $\bar{\lambda} > \lambda$, then $-\frac{f(\bar{x})}{g(\bar{x})} > \lambda$, and consequently $\frac{f(\bar{x})}{g(\bar{x})} < -\lambda$. Since $\bar{x}$ is the optimal solution to problem (12), then $\frac{f(x)}{g(x)} < -\lambda$ for all $x \in S$. Hence, $f(x) + \lambda g(x) < 0$ for all $x \in S$ and consequently $F'(\lambda) < 0$. $\square$

Corollary 2. Let $F'(\lambda) = 0$; then the optimal solution to problem (16) is also optimal to problem (12).

Lemma 6. The function $F'(\lambda)$ is continuous, convex, and strictly increasing on R .

Proof. The continuity and convexity are proved similar to Lemma 4; see [30]. Let $\lambda_1 < \lambda_2$ and x_1 and x_2 be the optimal solutions to problems $Q'(\lambda_1)$ and $Q'(\lambda_2)$, respectively. Then $g(x) > 0$ yields

$$F'(\lambda_1) = f(x_1) + \lambda_1 g(x_1) < f(x_1) + \lambda_2 g(x_1) \leq f(x_2) + \lambda_2 g(x_2) = F'(\lambda_2).$$

$\square$

In this case, Algorithm 1 is modified in Step (1) by choosing λ such that $F'(\lambda) \leq 0$. In Step (2), the problem $Q'(\lambda)$ should be solved and in Step (3) the parameter λ is updated by equality $\lambda = -\frac{f(x^*)}{g(x^*)}$.

Using Algorithm 1 with the mentioned modifications, problem (11) could be solved in the third case, where in the second step of Algorithm 1, the following problem should be solved:

$$\max_f D' - \sum_e (v_e - \mu_e)^2 - \sum_{rs} (x_{rs} - \mu_{rs})^2 + \lambda \sqrt{\sum_e 4v_e^2 \sigma_e^2 + \sum_{rs} 4x_{rs}^2 \sigma_{rs}^2 + k''},$$

subject to constraints (2), (3), and (4), by using nonlinear constrained programming methods such as penalty methods or augmented Lagrangian approaches; see [25].

Efficiency of the method

Applying the proposed methods in Sections 2 and 3 for small and large-sized networks, respectively, the nonlinear problem (1) with probabilistic variables in the objective function would be reduced to deterministic models, problems (10) and (11).

It should be noted that by applying the enumeration method proposed in Section 2, $2^{|C_{set}|}$ nonlinear constrained problems should be solved where C_{set} is the candidate set introduced in Section 2. Therefore, in the worst case that $C_{set} = State$, 2^H NP-hard problems are to be solved, which cause the running time of the solution method to increase dramatically specially for large networks. Whereas, using the *CLT* approach for medium and large-scale networks, it is required to solve just one deterministic fractional model, problem (11), instead of several nonlinear models, while simultaneously the number of constraints is also decreased.

In order to obtain the solution of problem (11), three different cases are considered in a certain order, where in the case of feasibility of each one, the remaining cases are neglected. The involved problems in cases 1 and 3 are readily solved by applying the parametric method, generating a sequence that superlinearly converges to the solution. The least squares problem (15) in the second case could be also solved by active-set constraints or Lagrangian approaches; see [24]. It is worth mentioning that, applying the *CLT*, an approximation of the optimal value of original problem (1) is provided, where the reliability of the solution would be investigated via the numerical examples in the next section.

5 Numerical examples

In this section, some numerical examples are provided to illustrate the efficiency of the proposed approaches discussed in the previous sections.

Example 2. In the first example, a small-sized network is studied, using the method stated in Section 2. Consider Yang's network given in Figure 3. It is assumed, there are five realizations for the observed information in the network with probability vector $Probability = \{0.2, 0.1, 0.2, 0.2, 0.3\}$. The necessary data, including the free flow travel times of links and the observed (target) flows in different states, are summarized in Tables 1 and 2. Ten equilibrium paths are picked up by Eppstein's K -shortest paths ranking algorithm [14], corresponding to four $O - D$ pairs (1, 3), (1, 4), (2, 3), (2, 4) shown in Table 2.

The optimal path flow vectors f^* obtained by solving *MIQC* problem (8) and the proposed enumeration (*ENM*) method using problem (10), are given in Table 3, under column headings *MIQC* and *ENM*, respectively. Since

MATLAB solvers could not provide a feasible solution for *MIQC* problem (8), the corresponding model was implemented in *CPLEX* 12.6, while the *QC* problem (10) was solved by *MATLAB R2014a*. The optimal solutions with their corresponding objective values $Z(f^*)$ as well as the states that $Y_h(f^*) = 1$ for different values of D are inserted in Table 3 (Note that the optimal solution is not unique).

As it is seen in Table 3, both methods *MIQC* and *ENM* reach the same values of the objective function for different amounts of threshold D . In addition, increasing the value of D results in increasing the *MIQC* and *ENM* objective functions. In other words, when the threshold for the sum of squared errors rises, the total probability of states h with $Y_h(f^*) = 1$ grows.

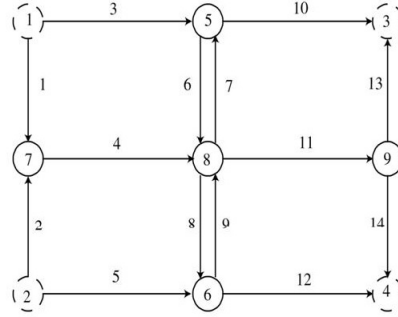


Figure 3: Yang's network

Next, to examine the stability of the resulted solutions in the case of occurring some errors in the observed information, we have made some modifications in the values of the observed link traffic counts and target $O-D$ demands. To this purpose, first, the values of $\bar{v}_e^h$ and $\bar{x}_{rs}^h$ for all links e and $O-D$ pairs (r, s) are multiplied by $\alpha = 1.5, 0.5$ in states $h = 1, 4$. The solutions for two values $D = 50000$ and $D = 95000$ are obtained by the *ENM* algorithm and represented in Table 4. Then to investigate the effect of probabilities, the probability vector is changed to $Probability = [0.05 \ 0.8 \ 0.05 \ 0.05 \ 0.05]$ and the solutions are again obtained for $\alpha = 1.5, 0.5$.

As it is seen in Table 4, multiplying the values of $\bar{v}_e^2$ and $\bar{x}_{rs}^2$ by $\alpha = 1.5$ does not change the solution in both cases $D = 50000$ and 95000 compared with that of Table 3. While, when the values of $\bar{v}_e^4$ and $\bar{x}_{rs}^4$ are multiplied by 1.5, the solution changes for both $D = 50000$ and $D = 95000$. This is because, as it is perceived by Table 3, the value of the function $Z_h(f^*)$ corresponding to the optimal solution f^* was less than $D = 50000$ for state $h = 4$; therefore, any change in the observed information in state $h = 4$ would affect the objective function. In other words, since in case $D = 50000$, the candidate set C_{set} consists of just $h = 4$, multiplying the values $\bar{v}_e^4$ and

Table 1: The information of links in Yang's network

Index of link	From	To	Free flow travel time	The observed traffic counts in states				
				1	2	3	4	5
1	1	5	13.18	130	100	120	150	100
2	1	7	4.29	100	210	200	180	180
3	2	6	11.49	180	80	80	150	130
4	2	7	4.46	130	50	80	130	100
5	5	3	13.24	200	80	120	100	90
6	5	8	3.00	30	90	90	90	100
7	6	4	12.16	200	150	100	110	130
8	6	8	5.00	10	180	180	150	100
9	7	8	11.89	230	60	60	100	120
10	8	5	4.00	30	330	330	130	200
11	8	6	4.00	50	85	120	140	150
12	8	9	13.05	210	150	150	130	130
13	9	3	4.19	115	140	140	100	100
14	9	4	3.11	90	200	200	100	140

$\bar{x}_{rs}^4$ by 1.5 for all links e and $O - D$ pairs (r, s) , cause the candidate set to be empty and consequently, the optimal objective value would be equal to zero. While, the state $h = 2$ did not even belong to the candidate set C_{set} in Table 3. By multiplying its observed values by 1.5, the value of $Z_2(f_2^*)$ is still greater than D which means that $2 \notin C_{set}$ for both $D = 50000$ and 95000 .

However, the case is different for $\alpha = 0.5$. Multiplying $\alpha = 0.5$ in $\bar{v}_e^2$ and $\bar{x}_{rs}^2$ brings the state $h = 2$ to C_{set} for both $D = 50000, 95000$. Although, since the probability of state $h = 2$ is less than the others, if it is necessary, the model might neglect it to find a solution with greater probability. When D increases to 95000 , the model finds a solution with $Z_h(f^*) \leq D$ for $h = 2, 3, 4, 5$.

Next, we have changed the used probability vector to $Probability = [0.05 \ 0.8 \ 0.05 \ 0.05 \ 0.05]$ to see the influence of probabilities on the optimal solution. Even, using the new vector of probabilities, the solutions corresponding to $\alpha = 1.5$ are the same as previous, since the candidate set has not changed. But, for $\alpha = 0.5$, where the candidate set includes the state $h = 2$, the model attempts to find a solution with $Z_2(f^*) \leq D$. Note in the case that $\bar{v}_e^4$ and $\bar{x}_{rs}^4$ are multiplied by 0.5, the candidate set dose not include the second state for $D = 50000$ and 95000 .

Generally, changing the observed information of some states, might or might not result in changing the solution. Indeed, it depends on the state itself, whether it is in the candidate set or not, as well as its observed (target) values and probability.

Example 3. In this example, a large-sized network is investigated by using

Table 2: Target $O - D$ demands with the equilibrium shortest paths in Yang's network

Index	Target demands in states					From	To	Path	Path track
	1	2	3	4	5				
1	150	150	100	200	120	1	3	1	$1 \rightarrow 5 \rightarrow 3$
2	120	200	120	150	130	1	4	2	$1 \rightarrow 5 \rightarrow 8 \rightarrow 6 \rightarrow 4$
								3	$1 \rightarrow 5 \rightarrow 8 \rightarrow 9 \rightarrow 4$
								4	$1 \rightarrow 7 \rightarrow 8 \rightarrow 6 \rightarrow 4$
								5	$1 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 4$
3	820	140	180	90	90	2	3	6	$2 \rightarrow 7 \rightarrow 8 \rightarrow 5 \rightarrow 3$
								7	$2 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 3$
								8	$2 \rightarrow 6 \rightarrow 8 \rightarrow 5 \rightarrow 3$
								9	$2 \rightarrow 6 \rightarrow 8 \rightarrow 9 \rightarrow 3$
4	300	80	80	100	170	2	4	10	$2 \rightarrow 6 \rightarrow 4$

the normal approximation stated in Sections 3 and 4. We have considered *Anahiem* network shown in Figure 4 with 419 nodes and 914 links. Figure 4 and such necessary information as the free flow travel times and capacities of links are available in address [http : //www.bgu.ac.il/ bargera/tntp/](http://www.bgu.ac.il/~bargera/tntp/). Using Eppstein's K -shortest path ranking algorithm [14], the equilibrium cyclic free paths for 80, 90, 100, and 110 selected $O - D$ pairs are determined. We have considered different samples of sizes 200, 300, 400, and 500 for network realizations, taking random values with a discrete uniform distribution in the interval $[0, 8500]$, for the observed link traffic counts and target $O - D$ demands.

The proposed approximation method based on the *CLT* is written in *MATLAB R2014a* and applied to find the optimal solutions for different values of D and different numbers of $O - D$ pairs, applying taken random samples. To investigate the acceptability of the obtained solutions, the estimated optimal solution vector f^* obtained by problem (11) is given to problem (6) and $Z(f^*)$ is calculated. Then the values of $\phi(\frac{D - \mu_W}{\sigma_W})$ for f^* and $Z(f^*)$ are compared. The absolute errors between $Z(f^*)$ and $\phi(\frac{D - \mu_W}{\sigma_W})$ for different sample sizes (SS) and different values of D are shown in Table 5. The small absolute errors indicate the validity of the estimated solutions by the *CLT* approach.

The optimal values of parameter λ in Algorithm 1 obtained in one of the three cases discussed in Section 4, for 80 $O - D$ pairs, are inserted in Table 6. As the table indicates, the optimal value of λ mostly belongs to the interval $[0, 2]$. In other words, the numerator in problem (11) is at most two times of the denominator. However, this may change for different problems.

Table 3: The estimated path flows with their objective functions values for different amounts of D

Paths	Optimal flows					
	$D = 50000$		$D = 65000$		$D = 80000$	
	ENM	$MIQC$	ENM	$MIQC$	ENM	$MIQC$
1	111.81	95.16	67.55	77.74	67.55	79.89
2	60.24	51.94	50.87	49.75	50.87	50.94
3	35.46	35.93	32.25	44.50	32.25	49.44
4	53.88	42.78	61.62	47.1	61.62	47.93
5	20.52	31.51	31.18	42.38	31.18	46.67
6	11.95	26.74	34.43	41.22	34.43	46.23
7	23.59	29.30	17.89	38.60	17.89	44.49
8	46.60	40.54	49.35	45.67	49.35	46.77
9	53.66	46.83	31.80	41.75	31.80	44.38
10	54.67	61.11	81.92	49.42	81.92	85.49
States	{4}	{4}	{4, 5}	{4, 5}	{4, 5}	{4, 5}
$Z(f^*)$	0.2000	0.2000	0.5000	0.5000	0.5000	0.5000
Paths	$D = 95000$		$D = 11000$		$D = 125000$	
	ENM	$MIQC$	ENM	$MIQC$	ENM	$MIQC$
1	28.30	32.37	33.59	42.53	33.59	64.83
2	36.13	36.51	53.37	44.52	53.37	45.89
3	34.41	32.82	29.95	35.03	29.95	42.52
4	66.80	64.44	81.51	77.77	81.51	56.01
5	54.42	54.69	58.09	50.32	58.09	50.38
6	38.55	36.15	10.15	23.73	10.15	41.40
7	11.53	17.73	9.73	16.52	9.73	32.42
8	127.70	125.08	131.02	107.1	131.02	67.56
9	28.26	25.73	18.44	27.88	18.44	41.73
10	11.15	18.61	25.73	38.12	25.73	61.162
States	{3, 4, 5}	{3, 4, 5}	{2, 3, 4, 5}	{2, 3, 4, 5}	{2, 3, 4, 5}	{2, 3, 4, 5}
$Z(f^*)$	0.7000	0.7000	0.8000	0.8000	0.8000	0.8000

Table 4: The changes in the solution under some modifications in the observed information

	$\bar{v}_e^2 = 1.5\bar{v}_e^2, \quad \bar{x}_{rs}^2 = 1.5\bar{x}_{rs}^2$	$\bar{v}_e^4 = 1.5\bar{v}_e^4, \quad \bar{x}_{rs}^4 = 1.5\bar{x}_{rs}^4$
Paths	$D = 50000$	$D = 95000$
1	111.81	28.30
2	60.24	36.14
3	35.46	36.41
4	53.88	66.81
5	20.53	54.42
6	11.95	38.55
7	23.60	11.53
8	46.61	127.71
9	53.67	28.27
10	54.67	11.16
$Z(f^*)$	0.2	0.7
States	{4}	{3, 4, 5}
	$\bar{v}_e^2 = 0.5\bar{v}_e^2, \quad \bar{x}_{rs}^2 = 0.5\bar{x}_{rs}^2$	$\bar{v}_e^4 = 0.5\bar{v}_e^4, \quad \bar{x}_{rs}^4 = 0.5\bar{x}_{rs}^4$
Paths	$D = 50000$	$D = 95000$
1	111.81	27.45
2	60.24	37.62
3	35.46	30.79
4	53.88	62.70
5	20.53	54.72
6	11.95	31.54
7	23.60	15.57
8	46.61	127.64
9	53.67	22.32
10	54.67	11.28
$Z(f^*)$	0.2	0.8
States	{4}	{2, 3, 4, 5}
<i>Probability</i> = [0.05 0.8 0.05 0.05 0.05]		
	$\bar{v}_e^2 = 1.5\bar{v}_e^2, \quad \bar{x}_{rs}^2 = 1.5\bar{x}_{rs}^2$	$\bar{v}_e^4 = 1.5\bar{v}_e^4, \quad \bar{x}_{rs}^4 = 1.5\bar{x}_{rs}^4$
Paths	$D = 50000$	$D = 95000$
1	111.81	28.30
2	60.24	36.14
3	35.46	36.41
4	53.88	66.81
5	20.53	54.42
6	11.95	38.55
7	23.60	11.53
8	46.61	127.71
9	53.67	28.27
10	54.67	11.16
$Z(f^*)$	0.05	0.15
States	{4}	{3, 4, 5}
	$\bar{v}_e^2 = 0.5\bar{v}_e^2, \quad \bar{x}_{rs}^2 = 0.5\bar{x}_{rs}^2$	$\bar{v}_e^4 = 0.5\bar{v}_e^4, \quad \bar{x}_{rs}^4 = 0.5\bar{x}_{rs}^4$
Paths	$D = 50000$	$D = 95000$
1	16.80	27.45
2	26.69	37.62
3	14.98	30.79
4	40.76	62.70
5	29.05	54.72
6	6.41	31.54
7	3.53	15.57
8	64.18	127.64
9	10.55	22.32
10	12.87	11.28
$Z(f^*)$	0.8	0.95
States	{2}	{2, 3, 4, 5}

Table 5: The absolute errors between $Z(f^*)$ and $\phi(\frac{D - \mu_W}{\sigma_W})$

$O - D$	80					90				
$SS \setminus D (10^{10})$	1.7	1.75	1.8	1.85	1.9	1.7	1.75	1.8	1.85	1.9
200	0.08	0.04	0.04	0.04	0.03	0.08	0.07	0.02	0.09	0.05
300	0.05	0.03	0.06	0.07	0.02	0.06	0.07	0.01	0.05	0.06
400	0.08	0.05	0.03	0.07	0.04	0.07	0.07	0.03	0.05	0.04
500	0.08	0.06	0.07	0.05	0.04	0.05	0.10	0.03	0.06	0.05
$O - D$	100					110				
$SS \setminus D (10^{10})$	1.7	1.75	1.8	1.85	1.9	1.7	1.75	1.8	1.85	1.9
200	0.06	0.09	0.05	0.06	0.08	0.03	0.08	0.07	0.06	0.08
300	0.07	0.06	0.00	0.10	0.07	0.04	0.08	0.06	0.04	0.08
400	0.03	0.09	0.03	0.06	0.07	0.03	0.07	0.04	0.03	0.06
500	0.04	0.08	0.03	0.06	0.08	0.03	0.07	0.05	0.01	0.08

The comparison between the average value of $Z(f^*)$ and $\phi(\frac{D - \mu_W}{\sigma_W})$, for different values of D and different numbers of $O - D$ pairs, are given by Figures 5a–5d. As it is seen, the value of $\phi(\frac{D - \mu_W}{\sigma_W})$ is a reasonable approximation for $Z(f^*)$, the objective function of problem (6).

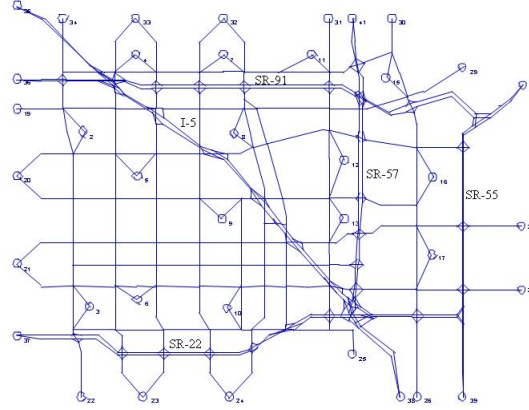


Figure 4: Anaheim network

Example 4. In the last example, in order to examine the efficiency of the *CLT* method in comparison to other existing nonlinear solvers, we consider a 50-node network with 998 links and 56 $O - D$ pairs, where the network topology and necessary data are constructed randomly. Indeed, the existence

Table 6: The optimal values of parameter $\bar{\lambda}$ for 80 $O - D$ pairs

$SS \setminus D$	1.7 (10^{10})	1.75 (10^{10})	1.8 (10^{10})	1.85 (10^{10})	1.9 (10^{10})
200	0.96	0.41	0.32	1.02	1.58
300	1.02	0.39	0.22	0.96	1.55
400	1.00	0.39	0.22	0.88	1.54
500	1.02	0.35	0.28	0.91	1.55

Table 7: The comparison between the *CLT* method and *MIQC* solvers

D	1.9 (10^8)		1.8 (10^8)		1.7 (10^8)	
Solver	RT (sec)	$OBJF$	RT (sec)	$OBJF$	RT (sec)	$OBJF$
<i>CLT</i>	54.13	1.00	46.71	0.97	43.47	0.50
<i>BONMIN</i>	11.75	0.98	—	<i>Infeasible</i>	—	<i>Infeasible</i>
<i>BARON</i>	—	<i>Infeasible</i>	—	<i>Infeasible</i>	—	<i>Infeasible</i>
<i>OQNLP</i>	—	<i>Infeasible</i>	—	<i>Infeasible</i>	—	<i>Infeasible</i>

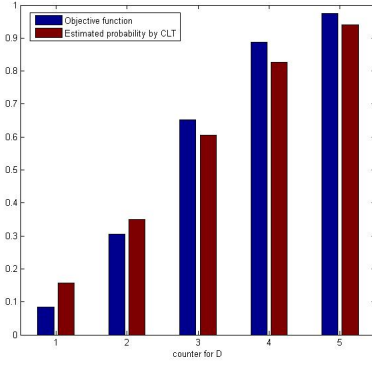
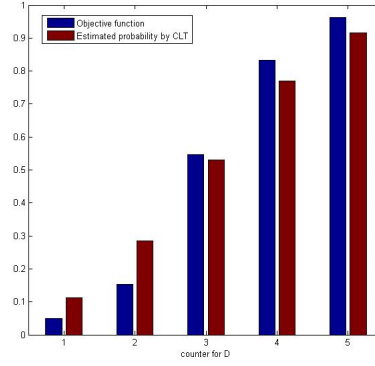
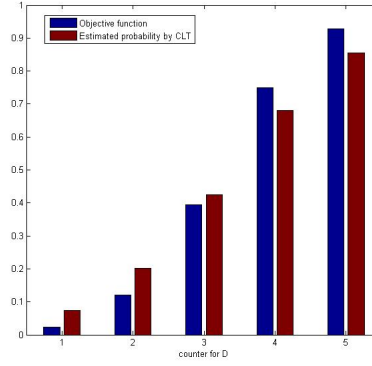
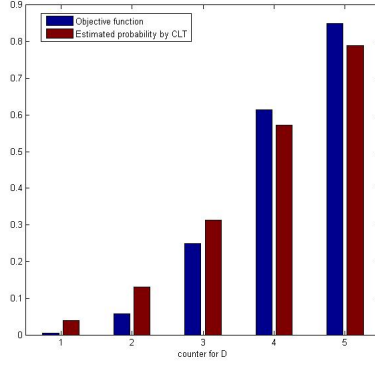
or nonexistence of links between nodes is determined by binary random variables, while the free flow travel times of links are randomly selected from the interval $[10, 30]$. Using Eppstein K -shortest paths ranking algorithm [14], 65 paths are picked up. In addition, 200 realizations are considered for the network where the values of the observed link traffic counts and target $O - D$ demands are taken from the interval $[0, 750]$ by the discrete uniform distribution.

Problem (1) is solved via both applying the proposed normal approximation based on the *CLT* and employing *MIQC* solvers including *BONMIN*, *BARON*, *OQNLP* to problem (8). The corresponding running time (RT) and the objective function value ($OBJF$) for different values of D are inserted in Table 7. The sign — indicates that the solver could not find any feasible solution.

As it is seen by Table 7, in most cases, the *MIQC* solvers did not find any feasible solution, while the *CLT*-based method was always able to find the optimal solution in an acceptable time. We have also examined such other existing solvers as *LINDOGLOBAL*, *COUNNE*, and *DICOPT* on smaller and larger networks, where almost in all cases, the solvers did not provide any solution. Since the results were similar to Table 7, they have not been reported again.

6 Summary and conclusions

The available data in such transportation problems as the $O - D$ matrix estimation may be treated as nondeterministic information due to some external conditions. In this paper, we have considered different states for the

(a) Comparison for 80 $O - D$ pairs(b) Comparison for 90 $O - D$ pairs(c) Comparison for 100 $O - D$ pairs(d) Comparison for 110 $O - D$ pairsFigure 5: Comparison between $\phi(\frac{D - \mu_W}{\sigma_W})$ and $Z(f^*)$

observed link traffic counts and target $O - D$ matrix. The purpose was to estimate the solution that maximizes the probability of the sum of the squared errors being less than or equal to a pre-selected threshold. We investigated the problem in small and large-sized networks. An enumeration-based solution algorithm of the exponential time complexity was presented for small networks, where the validity of the solutions was verified via comparing with *MIQC* solutions. As it was shown through examples, both the *MIQC* solvers and enumeration algorithm provided equal objective function values in the optimality. Due to the large running time of the enumeration method, a normal approximation was proposed for large-sized networks. Using the *CLT*, the probabilistic problem in large-sized networks was transformed into one deterministic nonlinear fractional programming model. To reduce the complexity of the resulted model, three cases were considered to be solved in a

certain order where in the case of feasibility of each one, the next problems would be neglected. A parametric algorithm was implemented to solve the problems in the first and third cases, where the obtained sequence would superlinearly converge to the optimal solution. Providing some numerical examples, the approximated objective function by the *CLT* approach was compared with its true value where considering the small differences between them, the efficiency of the *CLT* approach was verified.

References

1. Abareshi, M. and Zaferanieh, M. *A bi-level capacitated P-median facility location problem with the most likely allocation solution*, Transport. Res. B-Meth. 123 (2019), 1–20.
2. Abareshi, M. and Zaferanieh, M. *The network 1-median with discrete demand weights and travel times*, Iranian journal of numerical analysis and optimization, 9(1) (2019), 69–92.
3. Abareshi, M., Zaferanieh, M. and Keramati, B. *Path flow estimator in an entropy model using a nonlinear L-shaped algorithm*, Netw. Spat. Econ. 17 (2017), 293–315.
4. Abareshi, M., Zaferanieh, M. and Safi M. R. *Origin-destination matrix estimation problem in a Markov chain approach*, Netw. Spat. Econ. 19 (2019), 1069–1096.
5. Abrahamsson, T. *Estimation of origin-destination matrices using traffic counts - A literature survey*, IIASA Interim Report. IIASA, Laxenburg, Austria: IR-98-021 (1998).
6. Berman, O. and Wang, J. *Probabilistic location problems with discrete demand weights*, Networks, 44 (2004), 47–57.
7. Berman, O. and Wang, J. *The network p-median problem with discrete probabilistic demand weights*, Comput. Oper. Res. 37 (2010), 1455–1463.
8. Billionnet, A., Elloumi, S. and Plateau, M. C. *Improving the performance of standard solvers for quadratic 0-1 programs by a tight convex reformulation: the QCR method*, Discrete. Appl. Math. 157 (2009), 1185–1197.
9. Carey, M., Hendrickson, C. and Siddharthan, K. *A method for direct estimation of origin/destination trip matrices*, Transport. Sci. 15 (1981), 32–49.
10. Carey M. and Revelli, R. *Constrained estimation of direct demand functions and trip matrices*, Transport. Sci. 20 (1986), 143–152.

11. Cascetta, E. *Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator*, Transport. Res. B-Meth. 18 (1984), 289–299.
12. Charnes, A. and Cooper, W.W. *Deterministic equivalents for optimizing and satisficing under chance constraints*, Oper. Res. 11 (1963), 18–39.
13. Ching, W., Scholtes, S. and Zhang S. *Numerical algorithms for dynamic traffic demand estimation between zones in a network*, Eng. Optimiz. 36 (2004), 379–400.
14. Eppstein D. *Finding the K-shortest paths*, SIAM J. Comput. 28 (1998), 652–673.
15. Galli, L. and Letchford, A. N. *Reformulating mixed-integer quadratically constrained quadratic programs*, Working Paper. The Department of Management Science, Lancaster University, 2011.
16. Hoang, N.H., Vu, H.L. and Lo, H.K. *An informed user equilibrium dynamic traffic assignment problem in a multiple origin-destination stochastic network*, Transport. Res. B-Meth. 115 (2018), 207–230.
17. Jones, L.K., Gartner, N.H., Shubov, M., Stamatiadis, C. and Einstein, D. *Modeling origin-destination uncertainty using network sensor and survey data and new approaches to robust control*, Transport. Res. C-Emer. 94 (2018), 121–132.
18. Jornsten, K. and Nguyen, S. *On the estimation of a trip matrix from network data*, Technical report. LiTH-MAT-R-79-36, Department of Mathematics, University of Linköping, Sweden, 1979.
19. Lu, C., Fang, S.C., Jin, Q., Wang, Z. and Xing, W. *KKT solution and conic relaxation for solving quadratically constrained quadratic programming problems*, SIAM J. Optim. 21 (2011), 1475–1490.
20. Ma, W. and Qian, Z. *Statistical inference of probabilistic origin-destination demand using day-to-day traffic data*, Transport. Res. C-Emer. 88 (2018), 227–256.
21. Ma, W. and Qian, Z. *Estimating multi-year 24/7 origin-destination demand using high-granular multi-source traffic data*, Transport. Res. C-Emer. 96 (2018), 96–121.
22. Maher, M.J. *Inferences on trip matrices from observations on link volumes: A Bayesian statistical approach*, Transport. Res. B-Meth. 17 (1983), 435–447.
23. Misener, R. and Floudas, C.A. *Global optimization of mixed-integer quadratically-constrained quadratic programs (MIQCQP) through piecewise-linear and edge-concave relaxations*, Math. Program. 136 (2012), 155–182.

24. Nie, Y., Zhang, H.M. and Recker, W.W. *Inferring origin-destination trip matrices with a decoupled GLS path flow estimator*, Transport. Res. B-Meth. 39 (2005), 497–518.
25. Nocedal, J. and Wright, S. *Numerical Optimization*, 2nd ed, Springer-Verlag New York, 2006.
26. Parry, K. and Hazelton, M.L. *Bayesian inference for day-to-day dynamic traffic models*, Transport. Res. B-Meth. 50 (2013), 104–115.
27. Pitombeira-Neto, A.R., Loureiro, C.F.G. and Carvalho, L.E. *A dynamic hierarchical Bayesian model for the estimation of day-to-day origin-destination flows in transportation networks*, Netw. Spat. Econ. 20 (2020), 499–527.
28. Shao, H., Lam, W.H.K., Sumalee, A., Chen, A. and Hazelton, M. L. *Estimation of mean and covariance of peak hour origin-destination demands from day-to-day traffic counts*, Transport. Res. B-Meth. 68 (2014), 52–75.
29. Spiess, H. *A maximum likelihood model for estimating origin-destination matrices*, Transport. Res. B-Meth. 21 (1987), 395–412.
30. Stancu-Minasian, I.M. *Fractional programming, theory, methods and applications*, 1st ed, Springer Netherlands, 1997.
31. Sun, C., Chang, T., Luan, X., Tu, Q. and Tang, W. *Origin-destination demand reconstruction using observed travel time under congested network*, Netw. Spat. Econ. <https://doi.org/10.1007/s11067-020-09496-4> (2020).
32. Sun, C., Chang, Y., Shi, Y., Cheng, L. and Ma, J. *Subnetwork origin-destination matrix estimation under travel demand constraints*, Netw. Spat. Econ. 19 (2019), 1123–1142.
33. Van Zuylen, H. and Willumsen, L.G. *The most likely trip matrix estimated from traffic counts*, Transport. Res. B-Meth. 14 (1980), 281–293.
34. Wardrop, J. *Some theoretical aspects of road traffic research*, Proceedings of the institution of civil engineers, (1952) 325–378.
35. Willumsen, L.G. *Estimation of an $O - D$ matrix from traffic counts- A review*, Working paper, Institute of Transport Studies, University of Leeds, Leeds, UK 1978.
36. Willumsen, L.G. *Simplified transport models based on traffic counts*, Transportation, 10 (1981), 257–278.
37. Xie, C., Kockelman, K.M. and Waller, S.T. *A maximum entropy-least squares estimator for elastic origin-destination trip matrix estimation*, Transport. Res. B-Meth. 45 (2011), 1465–1482.



Numerical methods for solving nonlinear Volterra integro-differential equations based on Hermite–Birkhoff interpolation

S. Fazeli*

Abstract

We introduce a new family of multivalued and multistage methods based on Hermite–Birkhoff interpolation for solving nonlinear Volterra integro-differential equations. The proposed methods that have high order and extensive stability region, use the approximated values of the first derivative of the solution in the m collocation points and the approximated values of the solution as well as its first derivative in the r previous steps. Convergence order of the new methods is determined and their linear stability is analyzed. Efficiency of the methods is shown by some numerical experiments.

AMS(2010): 65R20

Keywords: Volterra integro-differential equations; Multistep collocation methods; Hermite–Birkhoff interpolation; Convergence; Linear stability.

1 Introduction

Consider Volterra integro-differential equations (VIDEs) of the form

$$y'(t) = g(t, y(t)) + \int_0^t K(t, \tau, y(\tau)) d\tau, \quad t \in I := [0, T], \quad y(0) = y_0, \quad (1)$$

where $g \in C(S)$ and the kernel $K \in C(\Omega)$ and satisfies the Lipschitz condition with respect to y , with $S = \{(t, y) : t \in I, y \in \mathbb{R}\}$, $\Omega = \{(t, \tau, y) : 0 \leq \tau \leq t \leq T, y \in \mathbb{R}\}$. It is well known that, under the suitable conditions, (1) possesses a unique solution $y(t) \in C^1[0, T]$ (see [3]).

*Corresponding author

Received 25 February 2020; revised 17 May 2020; accepted 29 July 2020

Somayyeh Fazeli

Marand Technical College, University of Tabriz, Tabriz, Iran. e-mail: fazeli@tabrizu.ac.ir

VIDEs arise in a variety of applications, for example, in viscoelasticity, control theory, epidemiology, and population dynamics [10, 18, 26].

On the class of discretization methods for the numerical solution of VIDEs, Linz [19] proposed algorithms for the numerical solution of (1), which consist of linear multistep methods, of the type commonly used for the numerical solution of the initial value problem for an ordinary differential equation

$$y'(t) = f(t, y(t)), \quad y(t_0) = y_0, \quad (2)$$

combined with a class of quadrature formulas. In [27], the construction of the quadrature rules generated by the backward differentiation formulas was discussed in detail and their linear stability properties were analyzed. Also, Brunner [2] introduced Runge–Kutta–Nyström methods, which are based on collocation techniques in certain polynomial spline spaces. One of the most popular numerical methods for solving this kind of equations is the class of collocation methods. In these methods, after discretization of the domain, the approximated solution in every subinterval depends on the fixed number m of the collocation points. These methods are of convergence order m for any choice of collocation parameters; see [5, 3]. Recently, a family of multistep collocation methods for (1) has also been introduced, which considers interpolation conditions in the previous r step points [7, 8]. Indeed, in [12] superimplicit multistep collocation methods (SIMCMs) are used for numerical solving VIDEs. In these methods, approximated values of the solution in the r previous steps and its first derivative in the m collocation points in the current and next subinterval are used, where the r step SIMCMs with m collocation points are of convergence order $2m+r$. Recently, a method based on general linear methods [6] for the numerical solution of (1) has been introduced and studied in [20, 21]. 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 [17] and nonlinear VIEs [13], specially for stiff problems.

The purpose of this paper is to construct higher order methods with extensive region of stability for solving (1). To do this, we introduce a new class of multistep collocation methods, which approximate the solution in each subinterval depending on the values of approximated solution and its first derivative in the fixed number r of previous time steps, and also the values of the first derivative of the approximated solution in the m collocation points. These methods will enable us, in deal with stiff equations, to make a sensible choice for the steplength of the algorithm.

This paper is organized as follows: In Section 2, the construction of multistep Hermite collocation methods (MHCMS) is described, and in Section 3, the convergence orders of these methods are determined. The linear stability is analyzed in Section 4. The paper is closed in Section 5, by showing efficiency of the methods by some numerical examples.

2 Construction of the methods

In this section, we describe the construction of the MHCMs for solving (1). Let $I_h = \{t_n : 0 = t_0 < t_1 < \dots < t_N = T\}$ be a partition of the interval $[0, T]$ with constant stepsize $h := t_{n+1} - t_n$, $n = 0, 1, \dots, N-1$. In these methods, to compute the approximated solution of (1) in the subinterval $[t_n, t_{n+1}]$, we use the approximated values of the solution and approximate values of the first derivative of the solution in the r previous steps and its first derivative in the m collocation points in the subinterval $[t_n, t_{n+1}]$. Denoting the fixed collocation parameters by $0 < c_1 < \dots < c_m \leq 1$ and the collocation points by $t_{n,j} = t_n + c_j h$, $j = 1, \dots, m$, the collocation polynomials restricted to subinterval $[t_n, t_{n+1}]$ are defined by

$$u_n(t_n + sh) = \sum_{k=0}^{r-1} \varphi_k(s) y_{n-k} + h \sum_{j=1}^m \psi_j(s) U_{n,j} + h \sum_{k=0}^{r-1} \chi_k(s) y'_{n-k}, \quad (3)$$

where $s \in [0, 1]$, $n = r-1, \dots, N-1$ and $U_{n,j} = u'_n(t_{n,j})$. By differentiating from (3) with respect to s , an approximation for $y'(t)$ is in the form

$$hu'_n(t_n + sh) = \sum_{k=0}^{r-1} \varphi'_k(s) y_{n-k} + h \sum_{j=1}^m \psi'_j(s) U_{n,j} + h \sum_{k=0}^{r-1} \chi'_k(s) y'_{n-k}. \quad (4)$$

The functions $\varphi_k(s)$, $\psi_j(s)$, and $\chi_k(s)$, $k = 0, 1, \dots, r-1$ and $j = 1, 2, \dots, m$ are polynomials of degree $2r+m-1$. By imposing the interpolation conditions at points $-k$, $k = 0, \dots, r-1$, in the polynomial (3) and (4) and at points c_j , $j = 1, 2, \dots, m$, in the polynomial (4), we obtain

$$\begin{aligned} \varphi_i(-k) &= \delta_{ik}, & \psi_j(-k) &= 0, & \chi_i(-k) &= 0, \\ \varphi'_k(c_j) &= 0, & \psi'_l(c_j) &= \delta_{lj}, & \chi'_i(c_j) &= 0, \\ \varphi'_k(-i) &= 0, & \psi'_j(-i) &= 0, & \chi'_k(-i) &= \delta_{ik}, \end{aligned} \quad (5)$$

where $i, k = 0, 1, \dots, r-1$ and $l, j = 1, 2, \dots, m$. The construction of these polynomials is obtained by Hermite–Birkhoff interpolation [24]. Let us assume the polynomials $\varphi_k(s)$, $\psi_j(s)$ and $\chi_j(s)$ in the form

$$\varphi_k(s) = \sum_{i=0}^{2r+m-1} \Phi_i^{[k]} \frac{s^i}{i!}, \quad \psi_j(s) = \sum_{i=0}^{2r+m-1} \Psi_i^{[j]} \frac{s^i}{i!}, \quad \chi_j(s) = \sum_{i=0}^{2r+m-1} \chi_i^{[j]} \frac{s^i}{i!}. \quad (6)$$

Now by setting $s = c_i$ in (6) and differentiating from these polynomials and setting $s = c_i$, $i = 1, 2, \dots, m$ and $s = -k$, $k = 0, 1, \dots, r-1$, a linear system for coefficient of these polynomials is obtained. The coefficient matrix $A \in \mathbb{R}^{(2r+m) \times (2r+m)}$ is in the form

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ 1 & \frac{(-1)^1}{1!} & \frac{(-1)^2}{2!} & \frac{(-1)^3}{3!} & \dots & \frac{(-1)^{2r+m-1}}{(2r+m-1)!} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \frac{(-r+1)^1}{1!} & \frac{(-r+1)^2}{2!} & \frac{(-r+1)^3}{3!} & \dots & \frac{(-r+1)^{2r+m-1}}{(2r+m-1)!} \\ 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & \frac{-1^1}{1!} & \frac{-1^2}{2!} & \dots & \frac{-1^{2r+m-2}}{(2r+m-2)!} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \frac{(-r+1)^1}{1!} & \frac{(-r+1)^2}{2!} & \dots & \frac{(-r+1)^{2r+m-2}}{(2r+m-2)!} \\ 0 & 1 & \frac{c_1^1}{1!} & \frac{c_1^2}{2!} & \dots & \frac{c_1^{2r+m-2}}{(2r+m-2)!} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \frac{c_m^1}{1!} & \frac{c_m^2}{2!} & \dots & \frac{c_m^{2r+m-2}}{(2r+m-2)!} \end{bmatrix},$$

and the right-hand vectors are defined by $\mathbf{u}_r^{[k]} \in \mathbb{R}^r$, $k = 1, 2, \dots, r$, $\mathbf{v}_m^{[j]} \in \mathbb{R}^m$, $j = 1, 2, \dots, m$ as

$$(\mathbf{u}_r^{[k]})_i = \begin{cases} 0, & i \neq k, \\ 1, & i = k, \end{cases} \quad (\mathbf{v}_m^{[j]})_i = \begin{cases} 0, & i \neq j, \\ 1, & i = j, \end{cases}$$

Their coefficients are obtained by solving the systems

$$\begin{aligned} A\Phi^{[k]} &= [\mathbf{u}_r^{[k+1]}, \mathbf{0}_r, \mathbf{0}_m]^T, & k = 0, 1, \dots, r-1, \\ A\Psi^{[j]} &= [\mathbf{0}_r, \mathbf{0}_r, \mathbf{v}_m^{[j]}]^T, & j = 1, 2, \dots, m, \\ A\chi^{[k]} &= [\mathbf{0}_r, \mathbf{u}_r^{[k+1]}, \mathbf{0}_m]^T, & k = 0, 1, \dots, r-1. \end{aligned} \quad (7)$$

Now, we discuss on the uniqueness of the solution of (7). For this purpose, we briefly review some definitions and known theorems about Hermite–Birkhoff interpolation (see [15])

Let k and n be natural numbers and let

$$E = \|\epsilon_{ij}\|, \quad i = 1, \dots, k, \quad j = 0, 1, \dots, n-1,$$

be a matrix with k rows and n columns having elements

$$\epsilon_{ij} = 0 \text{ or } 1,$$

which are such that

$$\sum_{i,j} \epsilon_{ij} = n.$$

We shall also assume that E has no row entirely composed of zeros. Let also

$$x_1 < x_2 < \cdots < x_k.$$

We shall also assume that E has no row entirely composed of zeros. Let also

$$x_1 < x_2 < \cdots < x_k,$$

be increasing reals. We also need the set of ordered pairs

$$e = \{(i, j) \mid \epsilon_{ij} = 1\}.$$

The reals x_i and the “incidence matrix” E describe the interpolation problem

$$f^{(j)}(x_i) = y_i^{(j)} \quad \text{for } (i, j) \in E. \quad (8)$$

It is appropriate to refer to (8) as a Hermite–Birkhoff interpolation problem, which we shall abbreviate to HB-problem.

Definition 1. We shall say that the HB-problem (8) is poised, provided that if

$$P(x) \in \pi_{n-1}, P^{(j)}(x_i) = 0, \quad \text{for all } (i, j) \in E,$$

then $P(x) \equiv 0$, in the other words, the matrix E is called poised if the associated interpolation problem is uniquely solvable for any set of constants $y_i^{(j)}$, regardless of the choice of the ordered points $x_1, x_2, \dots, x_k$.

Define

$$\tilde{m}_j = \sum_{i=1}^k \epsilon_{i,j}, \quad \tilde{M}_l = \sum_{j=0}^l \tilde{m}_j, \quad j, l = 0, 1, \dots, n-1.$$

Schoenberg [24] showed that a necessary condition for E , to be poised is that

$$\tilde{M}_l \geq l + 1, \quad l = 0, 1, \dots, n-1,$$

in which these inequalities are called the Polya condition.

Definition 2. Let the incidence matrix E have k rows. Let f_i be the column index of the first one that appears in the row i . Then E is called a pyramid matrix if, for each i , $\epsilon_{ij} = 1$ implies $\epsilon_{ij'} = 1$ for $f_i \leq j' \leq j$, and there is some value of $1 \leq i \leq k$ such that $f_1 \geq f_2 \geq \cdots \geq f_i$ and $f_i \leq f_{i+1} \leq \cdots \leq f_k$.

Then Ferguson [15] declared the following theorem for poising the matrix E with respect to the ordering $x_1 < x_2 < \cdots < x_k$.

Theorem 1. If E is a pyramid matrix with k rows, satisfying the Polya conditions, then E is poised with respect to the ordering $x_1 < x_2 < \cdots < x_k$.

Now, we show that our interpolation problems (5) have unique solution. For these problems, the interpolation points can be considered by

$$-r + 1 < -r + 2 < \cdots < -1 < 0 < c_1 < c_2 < \cdots < c_m.$$

Hence the matrix E with $m + r$ rows and $2r + m$ columns can be defined by

$$E = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 1 & 0 & \cdots & 0 \end{bmatrix}.$$

Thus, we have

$$\tilde{m}_0 = r, \quad \tilde{m}_1 = m + r, \quad \tilde{m}_3 = 0, \quad \dots, \tilde{m}_{n-1} = 0,$$

$$\tilde{M}_0 = r, \quad \tilde{M}_1 = 2r + m, \quad \tilde{M}_3 = 2r + m, \quad \dots, \tilde{M}_{n-1} = 2r + m = n.$$

It can be easily seen that the the matrix E satisfies in the polya condition. Also by considering

$$f_1 = 0, \quad f_2 = 0, \quad \dots, f_r = 0, \quad f_{r+1} = 1, \quad \dots, f_{r+m} = 1,$$

one can see that the matrix E is a pyramid matrix. Thus by Theorem 1, we conclude that E is poised with respect to the ordering $-r + 1 < -r + 2 < \cdots < -1 < 0 < c_1 < c_2 < \cdots < c_m$ and the interpolation problems (5) have the unique solution. In the other word, the matrix A is nonsingular.

The exact MHCM is then obtained by imposing the collocation conditions for equation (1), that is, the collocation polynomials (3) exactly satisfy (1) at the collocation points $t_{n,i}$, which leads to the system of m equations in the unknowns $U_{n,i}$ in the form

$$U_{n,i} = F_{n,i} + \Phi_{n,i}, \quad (9)$$

where

$$\begin{aligned} F_{n,i} &= g(t_{n,i}, u_n(t_{n,i})) + h \sum_{\nu=0}^{n-1} \int_0^1 K(t_{n,i}, t_\nu + sh, u_\nu(t_\nu + sh)) ds, \\ \Phi_{n,i} &= h \int_0^{c_i} K(t_{n,i}, t_n + sh, u_n(t_n + sh)) ds, \quad i = 1, \dots, m. \end{aligned} \quad (10)$$

Then $y_{n+1} = u_n(t_{n+1})$ and $hy'_{n+1} = hu'_n(t_{n+1})$ are computed by

$$\begin{cases} y_{n+1} = \sum_{k=0}^{r-1} \varphi_k(1)y_{n-k} + h \sum_{j=1}^m \psi_j(1)U_{n,j} + h \sum_{k=0}^{r-1} \chi_k(1)y'_{n-k}, \\ hy'_{n+1} = \sum_{k=0}^{r-1} \varphi'_k(1)y_{n-k} + h \sum_{j=1}^m \psi'_j(1)U_{n,j} + h \sum_{k=0}^{r-1} \chi'_k(1)y'_{n-k}. \end{cases} \quad (11)$$

Also, the discretized MHCM is obtained by using suitable quadrature formulas for approximating $F_{n,i}$ and $\Phi_{n,i}$. The discretized multistep Hermite collocation polynomials for approximating $y(t_n + sh)$ and $y'(t_n + sh)$ take the forms

$$P_n(t_n + sh) = \sum_{k=0}^{r-1} \varphi_k(s)y_{n-k} + h \sum_{j=1}^m \psi_j(s)Y_{n,j} + h \sum_{k=0}^{r-1} \chi_k(s)y'_{n-k}, \quad (12)$$

$$hP'_n(t_n + sh) = \sum_{k=0}^{r-1} \varphi'_k(s)y_{n-k} + h \sum_{j=1}^m \psi'_j(s)Y_{n,j} + h \sum_{k=0}^{r-1} \chi'_k(s)y'_{n-k}, \quad (13)$$

where $n = r-1, \dots, N-1$ and the unknowns $Y_{n,j} = P'_n(t_{n,j})$ are determined by solving the nonlinear system

$$Y_{n,i} = \bar{F}_{n,i} + \bar{\Phi}_{n,i}. \quad (14)$$

By using quadrature formulas with the weights b_l and nodes ξ_l , $l = 1, \dots, \mu_1$, for integrating on $[0, 1]$, and the weights $w_{i,l}$ and nodes $d_{i,l}$, $l = 1, \dots, \mu_0$, for integrating on $[0, c_i]$, with positive integers μ_0 and μ_1 , one can write

$$\begin{aligned} \bar{F}_{n,i} &= g(t_{n,i}, P_n(t_{n,i})) + h \sum_{\nu=0}^{n-1} \sum_{l=1}^{\mu_1} b_l K(t_{n,i}, t_\nu + \xi_l h, P_\nu(t_\nu + \xi_l h)), \\ \bar{\Phi}_{n,i} &= h \sum_{l=1}^{\mu_0} w_{i,l} K(t_{n,i}, t_n + d_{i,l} h, P_n(t_n + d_{i,l} h)). \end{aligned} \quad (15)$$

Substituting from (12) in (15) yields

$$\begin{aligned} \bar{F}_{n,i} &= g(t_{n,i}, \sum_{k=0}^{r-1} \alpha_{ik} y_{n-k} + h \sum_{j=1}^m \beta_{ij} Y_{n,j} + h \sum_{k=0}^{r-1} \rho_{ik} y'_{n-k}) \\ &\quad + h \sum_{\nu=0}^{r-2} \sum_{l=1}^{\mu_1} b_l K(t_{n,i}, t_\nu + \xi_l h, \bar{y}_\nu(t_\nu + \xi_l h)) \\ &\quad + h \sum_{\nu=r-1}^{n-1} \sum_{l=1}^{\mu_1} b_l K(t_{n,i}, t_\nu + \xi_l h, \sum_{k=0}^{r-1} \delta_{lk} y_{\nu-k} + h \sum_{j=1}^m \eta_{lj} Y_{\nu,j} + h \sum_{k=0}^{r-1} \zeta_{lk} y'_{\nu-k}), \\ \bar{\Phi}_{n,i} &= h \sum_{l=1}^{\mu_0} w_{i,l} K(t_{n,i}, t_n + d_{i,l} h, \sum_{k=0}^{r-1} \bar{\gamma}_{ilk} y_{n-k} + h \sum_{j=1}^m \bar{\beta}_{ilj} Y_{n,j} + h \sum_{k=0}^{r-1} \bar{\rho}_{ilk} y'_{n-k}), \end{aligned}$$

where $\bar{y}_\nu(t_\nu + sh)$, $\nu = 0, 1, \dots, r-2$ are the starting approximated solutions, which are obtained by classical collocation method such that

$$\begin{aligned}\alpha_{ik} &= \varphi_k(c_i), & \beta_{ij} &= \psi_j(c_i), & \rho_{ik} &= \chi_k(c_i), \\ \delta_{lk} &= \varphi_k(\xi_l), & \eta_{lj} &= \psi_j(\xi_l), & \zeta_{lk} &= \chi_k(\xi_l), \\ \bar{\gamma}_{ilk} &= \varphi_k(d_{i,l}), & \bar{\beta}_{ilj} &= \psi_j(d_{i,l}), & \bar{\rho}_{ilk} &= \chi_k(d_{i,l}).\end{aligned}$$

3 Convergence analysis

In this section, we study the zero-stability of the method. When the method is applied on the equation $y' = 0$, the equations in (11) reduce to

$$\begin{aligned}y_{n+1} &= \sum_{k=0}^{r-1} \varphi_k(1)y_{n-k}, \\ hy'_{n+1} &= \sum_{k=0}^{r-1} \varphi'_k(1)y_{n-k}.\end{aligned}$$

Therefore, in analogy to [8, 22], zero-stability is defined as follows.

Definition 3. The MHCMS is said to be zero-stable if and only if all roots of the polynomials

$$p(\lambda) = \lambda^r - \sum_{k=0}^r \varphi_k(1)\lambda^{r-k-1}, \quad q(\lambda) = \lambda^r - \sum_{k=0}^r \varphi'_k(1)\lambda^{r-k-1},$$

have module less than or equal to unity and those of modules unity are simple.

Example 1. In the case, $r = 3$, $m = 2$, and $c = [c_1, 1]$, the method is zero-stable for all $c_1 \in [0.345, 1]$.

In what follows, we determine the convergence order of the exact and discretized MHCMS. We start by deriving local error estimate for the exact MHCMS solution $P(t) \in S_{2r+m-1}^{(1)}(I_h)$ for the linear VIDE

$$y'(t) = a(t)y(t) + g(t) + \int_0^t K(t, \tau)y(\tau)d\tau, \quad t \in [0, T]. \quad (16)$$

Using the Peanos theorem [3, 15] for y and differentiating from it, the representation of the local error is obtained as follows.

Lemma 1. Suppose that $p = 2r + m - 1$ and that the given functions in (16) satisfy $a \in C^p(I)$, $g \in C^p(I)$, and $K \in C^p(D)$ with $D := \{(t, \tau) : 0 \leq \tau \leq t \leq T\}$. Then for any choice of collocation parameters $0 < c_1 < \dots < c_m \leq 1$, the exact solution $y(t)$ satisfies

$$y(t_n + sh) = \sum_{k=0}^{r-1} \varphi_k(s)y(t_{n-k}) + h \sum_{j=1}^m \psi_j(s)y'(t_{n,j}) + h \sum_{k=0}^{r-1} \chi_k(s)y'(t_{n-k}) + h^{2r+m} R_{m,r,n}(s), \quad (17)$$

and

$$hy'(t_n + sh) = \sum_{k=0}^{r-1} \varphi'_k(s)y(t_{n-k}) + h \sum_{j=1}^m \psi'_j(s)y'(t_{n,j}) + h \sum_{k=0}^{r-1} \chi'_k(s)y'(t_{n-k}) + h^{2r+m} R'_{m,r,n}(s), \quad (18)$$

with

$$R_{m,r,n}(s) = \int_{-r+1}^1 K_{m,r}(s, \nu) y^{(2r+m)}(t_n + \nu h) d\nu, \\ K_{m,r}(s, \nu) = \frac{1}{(p-1)!} \left\{ (s - \nu)_+^{p-1} - \sum_{k=0}^{r-1} \varphi_k(s)(-k - \nu)_+^{p-1} \right. \\ \left. - h(p-1) \sum_{j=1}^m \psi_j(s)(-c_j - \nu)_+^{p-2} \right. \\ \left. - h(p-1) \sum_{k=0}^{r-1} \chi_k(s)(-k - \nu)_+^{p-2} \right\},$$

and

$$(x - t)_+ = \begin{cases} (x - t), & x \geq t, \\ 0, & x < t. \end{cases}$$

Theorem 2. Let $e(t) = y(t) - u(t)$ be the error of the exact MHCM and let $p = 2r + m - 1$. Suppose that

- (i) $K \in C^p(D \times \mathbb{R})$, $g \in C^p(I)$,
- (ii) the starting error is $\|e\|_{\infty, [0, t_r]} = O(h^p)$,
- (iii) $\rho(\mathbf{H}) < 1$, where ρ denotes the spectral radius and

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & \tilde{\mathbf{A}} \\ \hat{\mathbf{A}} & \bar{\mathbf{A}} \end{bmatrix} \quad (19)$$

with

$$\mathbf{A} = \left[\begin{array}{c|c} \mathbf{0}_{r-1,1} & \mathbf{I}_{r-1} \\ \hline \varphi_{r-1}(1) & \varphi_{r-2}(1), \dots, \varphi_0(1) \end{array} \right], \quad \tilde{\mathbf{A}} = \left[\begin{array}{c|c} \mathbf{0}_{r-1,1} & \mathbf{I}_{r-1} \\ \hline \chi_{r-1}(1) & \chi_{r-2}(1), \dots, \chi_0(1) \end{array} \right], \\ \hat{\mathbf{A}} = \left[\begin{array}{c|c} \mathbf{0}_{r-1,1} & \mathbf{I}_{r-1} \\ \hline \varphi'_{r-1}(1) & \varphi'_{r-2}(1), \dots, \varphi'_0(1) \end{array} \right], \quad \bar{\mathbf{A}} = \left[\begin{array}{c|c} \mathbf{0}_{r-1,1} & \mathbf{I}_{r-1} \\ \hline \chi'_{r-1}(1) & \chi'_{r-2}(1), \dots, \chi'_0(1) \end{array} \right]. \quad (20)$$

Then

$$\|e\|_\infty = O(h^{2r+m-1}).$$

Proof. We point out to the important parts of the proof for MHCMS in the case of linear VIDE. The proof can be straightforwardly extended to the case of a nonlinear VIDE (1) by using the mean value theorem [3]. Suppose that $Z_{n,j} = y'(t_{n,j})$. By subtracting (3) from (17), representation of the local error may be written as

$$\begin{aligned} e(t_n + sh) &= \sum_{k=0}^{r-1} \varphi_k(s) \mathcal{E}_{n-k} + h \sum_{j=1}^m \psi_j(s) \mathcal{E}'_{n,j} + h \sum_{k=0}^{r-1} \chi_k(s) \mathcal{E}'_{n-k} \\ &\quad + h^{2r+m} R_{m,r,n}(s), \end{aligned} \quad (21)$$

with $\mathcal{E}_{n-k} = y(t_{n-k}) - y_{n-k}$ and $\mathcal{E}'_{n,j} = Z_{n,j} - U_{n,j}$. Differentiating from (21) yields

$$\begin{aligned} h e'(t_n + sh) &= \sum_{k=0}^{r-1} \varphi'_k(s) \mathcal{E}_{n-k} + h \sum_{j=1}^m \psi'_j(s) \mathcal{E}'_{n,j} + h \sum_{k=0}^{r-1} \chi'_k(s) \mathcal{E}'_{n-k} \\ &\quad + h^{2r+m} R'_{m,r,n}(s). \end{aligned} \quad (22)$$

Replacing n by $l-1$ in (21) and (22), and $s = 1$, lead to

$$\mathcal{E}_l^{(1)} = \mathbf{A} \mathcal{E}_{l-1}^{(1)} + h \tilde{\mathbf{A}} \mathcal{E}'_{l-1}^{(1)} + h \tilde{\mathbf{S}} \mathcal{E}'_{l-1}^{(2)} + h^p \tilde{\boldsymbol{\rho}}_{m,r,l-1} \quad (23)$$

and

$$h \mathcal{E}'_l^{(1)} = \hat{\mathbf{A}} \mathcal{E}_{l-1}^{(1)} + h \bar{\mathbf{A}} \mathcal{E}'_{l-1}^{(1)} + h \bar{\mathbf{S}} \mathcal{E}'_{l-1}^{(2)} + h^p \bar{\boldsymbol{\rho}}_{m,r,l-1}, \quad (24)$$

where $\mathbf{A}$, $\hat{\mathbf{A}}$, $\bar{\mathbf{A}}$, and $\tilde{\mathbf{A}}$ are given in (20) and

$$\begin{aligned} \tilde{\mathbf{S}} &= \begin{pmatrix} \mathbf{0}_{r-1,m} \\ \boldsymbol{\psi}(1)^T \end{pmatrix}, \quad \tilde{\boldsymbol{\rho}}_{m,r,j} = \begin{pmatrix} \mathbf{0}_{r-1,1} \\ R_{m,r,j}(1) \end{pmatrix}, \\ \bar{\mathbf{S}} &= \begin{pmatrix} \mathbf{0}_{r-1,m} \\ \boldsymbol{\psi}'(1)^T \end{pmatrix}, \quad \bar{\boldsymbol{\rho}}_{m,r,j} = \begin{pmatrix} \mathbf{0}_{r-1,1} \\ R'_{m,r,j}(1) \end{pmatrix}, \end{aligned}$$

$$\boldsymbol{\varepsilon}_l^{(1)} = [\varepsilon_{l-r+1}, \dots, \varepsilon_l]^T \in \mathbb{R}^r,$$

$$\boldsymbol{\varepsilon}'_l^{(1)} = [\varepsilon'_{l-r+1}, \dots, \varepsilon'_l]^T \in \mathbb{R}^r, \quad \boldsymbol{\varepsilon}'_l^{(2)} = [\varepsilon'_{l,1}, \dots, \varepsilon'_{l,m}]^T \in \mathbb{R}^m,$$

$$\boldsymbol{\psi}(1) = [\psi_1(1), \dots, \psi_m(1)]^T, \quad \boldsymbol{\psi}'(1) = [\psi'_1(1), \dots, \psi'_m(1)]^T.$$

Combining (23) and (24) gives the following matrix equation:

$$\begin{pmatrix} \mathcal{E}_l^{(1)} \\ h\mathcal{E}'_l^{(1)} \end{pmatrix} = \mathbf{H} \begin{pmatrix} \mathcal{E}_{l-1}^{(1)} \\ h\mathcal{E}'_{l-1}^{(1)} \end{pmatrix} + \mathbf{G}\mathcal{E}_{l-1}^{(2)} + \mathbf{Q}_{m,r,l-1}h^p, \quad (25)$$

where $\mathbf{H}$ is given by (19) and

$$\mathbf{G} = \begin{pmatrix} h\tilde{\mathbf{S}} \\ h\bar{\mathbf{S}} \end{pmatrix}, \quad \mathbf{Q}_{m,r,j} = \begin{pmatrix} \tilde{\rho}_{m,r,j} \\ \bar{\rho}_{m,r,j} \end{pmatrix}.$$

The solution of difference equation (25) is (see [16])

$$\mathcal{E}_l^{(1)} = \mathbf{H}^{l-r+1}\mathcal{E}_{r-1}^{(1)} + \sum_{j=r-1}^{l-1} \mathbf{H}^{l-j-1}(\mathbf{G}\mathcal{E}_j^{(2)} + h^p\mathbf{Q}_{m,r,j}). \quad (26)$$

On the other hand, setting $s = c_i$, $i = 1, 2, \dots, m$ in (22), by using (4), leads to

$$e'(t_{n,i}) = \mathcal{E}'_{n,i} + h^{2r+m-1}R'_{m,r,n}(c_i). \quad (27)$$

On the other hand, equation (9) for the linear form of VIDE is equivalent to

$$\begin{aligned} U_{n,i} = & a(t_{n,i})u_n(t_{n,i}) + g(t_{n,i}) + h \sum_{l=0}^{n-1} \int_0^1 K(t_{n,i}, t_l + sh)u_l(t_l + sh)ds \\ & + h \int_0^{c_i} K(t_{n,i}, t_n + sh)u_n(t_n + sh)ds \end{aligned} \quad (28)$$

and (16) at points $t_{n,i}$ can be written as

$$\begin{aligned} y'(t_{n,i}) = & a(t_{n,i})y(t_{n,i}) + g(t_{n,i}) + h \sum_{l=0}^{n-1} \int_0^1 K(t_{n,i}, t_l + sh)y(t_l + sh)ds \\ & + h \int_0^{c_i} K(t_{n,i}, t_n + sh)y(t_n + sh)ds. \end{aligned} \quad (29)$$

Now by subtracting (28) from (29), we get

$$\begin{aligned} e'(t_{n,i}) = & a(t_{n,i})e(t_{n,i}) + h \sum_{l=0}^{n-1} \int_0^1 K(t_{n,i}, t_l + sh)e(t_l + sh)ds \\ & + h \int_0^{c_i} K(t_{n,i}, t_n + sh)e(t_n + sh)ds. \end{aligned} \quad (30)$$

Also, by substituting (30) in the (27), we obtain

$$\mathcal{E}'_{n,i} = a(t_{n,i})e(t_{n,i}) + h \sum_{l=0}^{n-1} \int_0^1 K(t_{n,i}, t_l + sh)e(t_l + sh)ds$$

$$\begin{aligned}
& + h \int_0^{c_i} K(t_{n,i}, t_n + sh) e(t_n + sh) ds \\
& - h^{2r+m-1} R'_{m,r,n}(c_i).
\end{aligned} \tag{31}$$

By the hypothesis on the starting error and by substituting (21) in (31), we have

$$\begin{aligned}
\mathcal{E}'_{n,i} = & a(t_{n,i}) \left(\sum_{k=0}^{r-1} \alpha_{i,k} \mathcal{E}_{n-k} + h \sum_{j=1}^m \beta_{i,j} \mathcal{E}'_{n,j} \right. \\
& \left. + h \sum_{k=0}^{r-1} \rho_{i,k} \mathcal{E}'_{n-k} + h^{2r+m} R_{m,r,n}(c_i) \right) \\
& + h \sum_{l=0}^{n-1} \int_0^1 K(t_{n,i}, t_l + sh) \left(\sum_{k=0}^{r-1} \varphi_k(s) \mathcal{E}_{n-k} \right. \\
& \left. + h \sum_{j=1}^m \psi_j(s) \mathcal{E}'_{n,j} + h \sum_{k=0}^{r-1} \chi_k(s) \mathcal{E}'_{n-k} + h^{2r+m} R_{m,r,n}(s) \right) ds \\
& + h \int_0^{c_i} K(t_{n,i}, t_n + sh) \left(\sum_{k=0}^{r-1} \varphi_k(s) \mathcal{E}_{n-k} \right. \\
& \left. + h \sum_{j=1}^m \psi_j(s) \mathcal{E}'_{n,j} + h \sum_{k=0}^{r-1} \chi_k(s) \mathcal{E}'_{n-k} + h^{2r+m} R_{m,r,n}(s) \right) ds \\
& - h^{2r+m-1} R'_{m,r,n}(c_i).
\end{aligned}$$

By the hypothesis on the starting error, it follows that

$$e(t_l + sh) = h^p q_l(s), \quad l = 0, 1, \dots, r-1, \tag{32}$$

with $\|q_l\|_\infty \leq C_1$ independent of h . Hence, we obtain

$$\begin{aligned}
(\mathbf{I}_m - hC_n^{(n)}) \mathcal{E}'_n^{(2)} = & h^{p+1} \sum_{l=0}^n \bar{R}_n^{(l)} + h \sum_{l=r}^{n-1} C_n^{(l)} \mathcal{E}'_n^{(2)} \\
& + h \sum_{l=r}^n B_n^{(l)} \mathcal{E}'_l^{(1)} + h^2 \sum_{l=r}^n D_n^{(l)} \mathcal{E}'_l^{(1)},
\end{aligned} \tag{33}$$

where $\bar{R}_n^{(l)} \in \mathbb{R}^m$, $C_n^{(l)}$, and $B_n^{(l)}$, $D_n^{(l)} \in \mathbb{R}^{m \times r}$ are defined as

$$(\bar{R}_n^{(l)})_i := \begin{cases} \int_0^1 K(t_{n,i}, t_l + sh) q_l(s) ds, & l = 0, 1, \dots, r-1, \\ \int_0^1 K(t_{n,i}, t_l + sh) R_{m,r,l}(s) ds, & l = r, \dots, n-1, \\ \int_0^1 K(t_{n,i}, t_n + sh) R_{m,r,n}(s) ds + a(t_{n,i}) R_{m,r,n}(c_i), & l = n, \end{cases}$$

$$\begin{aligned}
(B_n^{(l)})_{ik} &:= \begin{cases} \int_0^1 K(t_{n,i}, t_l + sh) \varphi_k(s) ds, & l = r, \dots, n-1, \\ \int_0^{c_i} K(t_{n,i}, t_n + sh) \varphi_k(s) ds + a(t_{n,i}) \varphi_k(c_i), & l = n, \end{cases} \\
(C_n^{(l)})_{ij} &:= \begin{cases} \int_0^1 K(t_{n,i}, t_l + sh) \psi_j(s) ds, & l = r, \dots, n-1, \\ \int_0^{c_i} K(t_{n,i}, t_n + sh) \psi_j(s) ds + a(t_{n,i}) \psi_j(c_i), & l = n, \end{cases} \\
(D_n^{(l)})_{ik} &:= \begin{cases} \int_0^1 K(t_{n,i}, t_l + sh) \chi_k(s) ds, & l = r, \dots, n-1, \\ \int_0^{c_i} K(t_{n,i}, t_n + sh) \chi_k(s) ds + a(t_{n,i}) \chi_k(c_i), & l = n. \end{cases}
\end{aligned}$$

Substituting (26) in (33), a recurrence formula for $\mathcal{E}_n^{(2)}$ is obtained. Then by the same way as described in [9, Theorem 4.2] and by considering the starting errors and the fact that $\rho(H) < 1$ lead to the estimate

$$\|\mathcal{E}_n^{(2)}\| \leq M_2 h^p,$$

and then from (26), a bound for $\|\mathcal{E}_n^{(1)}\|$ is obtained in the form

$$\|\mathcal{E}_n^{(1)}\| \leq M_1 h^p.$$

Using the local error representation and two above inequalities together, completes the proof. $\square$

Theorem 3. Suppose that the hypotheses of Theorem 2 hold. If the quadrature formulas defined in (15) have order $2r + m$ and $2r + m - 1$, respectively, then the uniform order of the discretized MHCMS is equal to $2r + m - 1$.

Now, in the table 1, we give a comparison between the methods MCMs [8], SIMCMs [12], and the new proposed method (MHCMS) in view of computational costs with respect to convergence order.

Table 1: Comparison with multistep and superimplicit multistep collocation methods.

	MCMs	SIMCMs (type 1)	SIMCMs (type 2)	MHCM
collocation points	m	m	m	m
dimensional of system	m	m	$2m$	m
order	$m + r - 1$	$2m + r - 1$	$2m + r - 1$	$2r + m - 1$

4 Linear stability analysis

In this section, we analyze the stability properties of the introduced methods with respect to the basic test equation (see [5, 3, 4, 23])

$$y'(t) = g(t) + \xi y(t) + \eta \int_0^t y(\tau) d\tau, \quad t > 0, \quad y(0) = y_0, \quad (34)$$

where $\xi, \eta \in \mathbb{C}$. The solution of (34) is stable if $Re(\lambda_1) < 0$ and $Re(\lambda_2) < 0$, where $\lambda_{1,2} = (\xi \pm \sqrt{\xi^2 + 4\eta})/2$ (see [1]). We observe that, particularly for real ξ and η , these conditions reduce to $\xi < 0$ and $\eta < 0$. As usual, we look for sufficient conditions for the stability of the numerical solution of (34).

Definition 4. Absolute stability region of the method, $\mathcal{R}$, is the set of all $(z := \xi h, w := \eta h^2) \in \mathbb{C} \times \mathbb{C}$, such that the numerical solution y_n of test equation (34) with a constant stepsize h , tends to zero as $n \rightarrow \infty$. The method is A_0 -stable if $\mathcal{R} \supseteq \mathbb{R}^- \times \mathbb{R}^-$ and is A -stable if it is stable for any value of (z, w) such that $Re(\lambda_1) < 0$ and $Re(\lambda_2) < 0$. An A -stable method is A_0 -stable too.

To state the main results of stability properties of MHCM, let us define

$$\begin{aligned} \alpha_k &= \int_0^1 \varphi_k(s) ds, & \beta_j &= \int_0^1 \psi_j(s) ds, & \gamma_k &= \int_0^1 \chi_k(s) ds, \\ \Omega_{ik} &= \int_0^{c_i} \varphi_k(s) ds, & \Gamma_{ij} &= \int_0^{c_i} \psi_j(s) ds, & \Delta_{ik} &= \int_0^{c_i} \chi_k(s) ds, \\ (\phi(\mathbf{c}))_{ik} &= \varphi_k(c_i), & (\psi(\mathbf{c}))_{ij} &= \psi_j(c_i), & (\chi(\mathbf{c}))_{ik} &= \chi_k(c_i), \end{aligned}$$

and consider the vectors and matrices

$$\begin{aligned} \varphi(1) &= [\varphi_0(1), \dots, \varphi_{r-1}(1)]^T, & \psi(1) &= [\psi_1(1), \dots, \psi_m(1)]^T, \\ \chi(1) &= [\chi_0(1), \dots, \chi_{r-1}(1)]^T, & \chi'(1) &= [\chi'_0(1), \dots, \chi'_{r-1}(1)]^T, \\ \varphi'(1) &= [\varphi'_0(1), \dots, \varphi'_{r-1}(1)]^T, & \psi'(1) &= [\psi'_1(1), \dots, \psi'_m(1)]^T, \\ \mathbf{y}_n^{(r)} &= [y_n, \dots, y_{n-r+1}]^T, & \mathbf{y}'_n^{(r)} &= [y'_n, \dots, y'_{n-r+1}]^T, \\ \mathbf{U}_n &= [U_{n,1}, \dots, U_{n,m}]^T, & \mathbf{u} &= [1, 1, \dots, 1]^T \in \mathbb{R}^m, \end{aligned}$$

$$\mathbf{E}_1 = \begin{bmatrix} \varphi(1)^T \\ \mathbf{I}_{r-1} & \mathbf{0}_{r-1,1} \end{bmatrix}, \quad \mathbf{F}_1 = \begin{bmatrix} \psi(1)^T \\ \mathbf{0}_{r-1,m} \end{bmatrix}, \quad \mathbf{G}_1 = \begin{bmatrix} \chi(1)^T \\ \mathbf{0}_{r-1,r} \end{bmatrix}, \quad (35)$$

$$\mathbf{E}_2 = \begin{bmatrix} \varphi'(1)^T \\ \mathbf{I}_{r-1} & \mathbf{0}_{r-1,1} \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} \psi'(1)^T \\ \mathbf{0}_{r-1,m} \end{bmatrix}, \quad \mathbf{G}_2 = \begin{bmatrix} \chi'(1)^T \\ \mathbf{0}_{r-1,r} \end{bmatrix}. \quad (36)$$

Theorem 4. Applying the exact MHCM on the test equation (34), leads to the following recurrence relation:

$$\begin{bmatrix} \mathbf{y}_n^{(r)} \\ h\mathbf{U}_n \\ \mathbf{y}'_n^{(r)} \end{bmatrix} = R(z, w) \begin{bmatrix} \mathbf{y}_{n-1}^{(r)} \\ h\mathbf{U}_{n-1} \\ \mathbf{y}'_{n-1}^{(r)} \end{bmatrix} + h\overline{\mathbf{G}}_n,$$

where $z := \xi h$, $w = \eta h^2$,

$$R(z, w) = [Q(z, w)]^{-1} M(z, w),$$

and

$$\begin{aligned} \overline{\mathbf{G}}_n &= \begin{bmatrix} \mathbf{0}_{r,m} \\ \mathbf{0}_{r,m} \\ \mathbf{g}_n - \mathbf{g}_{n-1} \end{bmatrix}, \\ Q(z, w) &= \begin{bmatrix} \mathbf{I}_r & \mathbf{0}_{r,m} & \mathbf{0}_{r,r} \\ \mathbf{0}_{r,r} & \mathbf{0}_{r,m} & \mathbf{I}_r \\ -z\boldsymbol{\varphi}(\mathbf{c}) - w\boldsymbol{\Omega} & \mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\boldsymbol{\Gamma} & -z\boldsymbol{\chi}(\mathbf{c}) - w\boldsymbol{\Delta} \end{bmatrix}, \\ M(z, w) &= \begin{bmatrix} \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{G}_1 \\ \mathbf{E}_2 & \mathbf{F}_2 & \mathbf{G}_2 \\ M_1 & M_2 & M_3 \end{bmatrix} \end{aligned}$$

with

$$\begin{aligned} M_1 &= -z\boldsymbol{\varphi}(\mathbf{c}) - w\boldsymbol{\Omega} + w\boldsymbol{\alpha}, \\ M_2 &= \mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\boldsymbol{\Gamma} + w\boldsymbol{\beta}, \\ M_3 &= -z\boldsymbol{\chi}(\mathbf{c}) - w\boldsymbol{\Delta} + w\boldsymbol{\gamma}. \end{aligned}$$

Proof. By the notations of this section, we rewrite the first relation in (11) in the form

$$y_{n+1} = \boldsymbol{\varphi}(1)^T y_n^{(r)} + \boldsymbol{\psi}(1)^T h \mathbf{U}_n + \boldsymbol{\chi}(1)^T h \mathbf{y}'_n^{(r)}, \quad (37)$$

or equivalently

$$\mathbf{y}_n^{(r)} = \mathbf{E}_1 y_{n-1}^{(r)} + \mathbf{F}_1 h \mathbf{U}_{n-1} + \mathbf{G}_1 h \mathbf{y}'_{n-1}^{(r)}. \quad (38)$$

Also, the second relation in (11) can be written in the form

$$h \mathbf{y}'_n^{(r)} = \mathbf{E}_2 y_{n-1}^{(r)} + \mathbf{F}_2 h \mathbf{U}_{n-1} + \mathbf{G}_2 h \mathbf{y}'_{n-1}^{(r)}. \quad (39)$$

Now we apply (10) on the test equation to get

$$U_{n,i} = g(t_{n,i}) + \xi u_n(t_{n,i}) + F_n(t_{n,i}) + \Phi_n(t_{n,i}), \quad (40)$$

where it can be written in the matrix form

$$\mathbf{U}_n = \mathbf{g}_n + \xi \left(\boldsymbol{\phi}(\mathbf{c}) y_n^{(r)} + \boldsymbol{\psi}(\mathbf{c}) h \mathbf{U}_n + \boldsymbol{\chi}(\mathbf{c}) h \mathbf{y}'_n^{(r)} \right) + \mathbf{F}_{n-1} + \boldsymbol{\Phi}_n, \quad (41)$$

with $(\mathbf{g}_n)_i = g(t_{n,i})$ and

$$(\mathbf{F}_{n-1})_i := F_n(t_{n,i}) = \eta h \sum_{l=0}^{n-1} \int_0^1 u_l(t_l + sh)$$

$$= \eta h \sum_{l=0}^{n-1} \int_0^1 \left(\sum_{k=0}^{r-1} \varphi_k(s) y_{l-k} + h \sum_{j=1}^m \psi_j(s) U_{l,j} + h \sum_{k=0}^{r-1} \chi_j(s) y'_{l-k} \right) ds,$$

$$\begin{aligned} (\Phi_n)_i &:= \Phi_n(t_{n,i}) = \eta h \int_0^{c_i} u_n(t_n + sh) ds \\ &= \eta h \int_0^{c_i} \left(\sum_{k=0}^{r-1} \varphi_k(s) y_{n-k} + h \sum_{j=1}^m \psi_j(s) U_{n,j} + h \sum_{j=1}^m \chi_j(s) y'_{n-k} \right) ds. \end{aligned}$$

Thus

$$\begin{aligned} \mathbf{F}_{n-1} &= \eta h \sum_{l=0}^{n-1} \boldsymbol{\alpha} y_l^{(r)} + \beta h \mathbf{U}_l + \gamma h y_l'^{(r)}, \\ \Phi_n &= \eta h \left(\boldsymbol{\Omega} y_n^{(r)} + \boldsymbol{\Gamma} h \mathbf{U}_n + \boldsymbol{\Delta} h y_n'^{(r)} \right). \end{aligned}$$

Therefore, we can write

$$\mathbf{F}_{n-1} - \mathbf{F}_{n-2} = \boldsymbol{\alpha} y_{n-1}^{(r)} + \beta h \mathbf{U}_{n-1} + \gamma h y_{n-1}'^{(r)}. \quad (42)$$

Now for obtaining a recurrence relation, substituting (42) in (41) and multiplying it by h yield

$$\begin{aligned} &(\mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\boldsymbol{\Gamma}) h \mathbf{U}_n + (-z\boldsymbol{\varphi}(\mathbf{c}) - w\boldsymbol{\Omega}) y_n^{(r)} + (-z\boldsymbol{\chi}(\mathbf{c}) - w\boldsymbol{\Delta}) h y_n'^{(r)} \\ &= (\mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\boldsymbol{\Gamma} + w\boldsymbol{\beta}) h \mathbf{U}_{n-1} + (-z\boldsymbol{\varphi}(\mathbf{c}) - w\boldsymbol{\Omega} + w\boldsymbol{\alpha}) y_{n-1}^{(r)} \\ &\quad + (-z\boldsymbol{\chi}(\mathbf{c}) - w\boldsymbol{\Delta} + w\boldsymbol{\gamma}) h y_{n-1}'^{(r)} + h(\mathcal{G}_n - \mathcal{G}_{n-1}). \end{aligned} \quad (43)$$

Hence the relations (38), (39), and (43) complete the proof. $\square$

Let us define

$$\begin{aligned} \tilde{\alpha}_k &= \sum_{l=1}^{\mu_1} b_l \varphi_k(\xi_l), \quad \tilde{\beta}_j = \sum_{l=1}^{\mu_1} b_l \psi_j(\xi_l), \quad \tilde{\gamma}_k = \sum_{l=1}^{\mu_1} b_l \chi_k(\xi_l), \\ \tilde{\Omega}_{ik} &= \sum_{l=1}^{\mu_0} w_{i,l} \varphi_k(d_{i,l}), \quad \tilde{\Gamma}_{ij} = \sum_{l=1}^{\mu_0} w_{i,l} \psi_j(d_{i,l}), \quad \tilde{\Delta}_{ik} = \sum_{l=1}^{\mu_0} w_{i,l} \chi_k(d_{i,l}). \end{aligned}$$

Theorem 5. The discretized MHCM, applied to the test equation (34), leads to the following recurrence relation:

$$\begin{bmatrix} \mathbf{y}_n^{(r)} \\ h\mathbf{Y}_n \\ \mathbf{y}'_n^{(r)} \end{bmatrix} = \tilde{R}(z, w) \begin{bmatrix} \mathbf{y}_{n-1}^{(r)} \\ h\mathbf{Y}_{n-1} \\ \mathbf{y}'_{n-1}^{(r)} \end{bmatrix} + h\tilde{\mathbf{G}}_n,$$

where $z := \xi h$, $w = \eta h^2$,

$$\tilde{R}(z, w) = [\tilde{Q}(z, w)]^{-1} \tilde{M}(z, w),$$

and

$$\tilde{Q}(z, w) = \begin{bmatrix} \mathbf{I}_r & \mathbf{0}_{r,m} & \mathbf{0}_{r,r} \\ \mathbf{0}_{r,r} & \mathbf{0}_{r,m} & \mathbf{I}_r \\ -z\boldsymbol{\varphi}(\mathbf{c}) - w\tilde{\boldsymbol{\Omega}} & \mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\tilde{\boldsymbol{\Gamma}} & -z\boldsymbol{\chi}(\mathbf{c}) - w\tilde{\boldsymbol{\Delta}} \end{bmatrix},$$

$$\tilde{M}(z, w) = \begin{bmatrix} \mathbf{E}_1 & \mathbf{F}_1 & \mathbf{G}_1 \\ \mathbf{E}_2 & \mathbf{F}_2 & \mathbf{G}_2 \\ \tilde{M}_1 & \tilde{M}_2 & \tilde{M}_3 \end{bmatrix}$$

with

$$\begin{aligned} \tilde{M}_1 &= -z\boldsymbol{\varphi}(\mathbf{c}) - w\tilde{\boldsymbol{\Omega}} + w\tilde{\boldsymbol{\alpha}}, \\ \tilde{M}_2 &= \mathbf{I}_m - z\boldsymbol{\psi}(\mathbf{c}) - w\tilde{\boldsymbol{\Gamma}} + w\tilde{\boldsymbol{\beta}}, \\ \tilde{M}_3 &= -z\boldsymbol{\chi}(\mathbf{c}) - w\tilde{\boldsymbol{\Delta}} + w\tilde{\boldsymbol{\gamma}}. \end{aligned}$$

Proof. It is similar to the proof of Theorem 4. $\square$

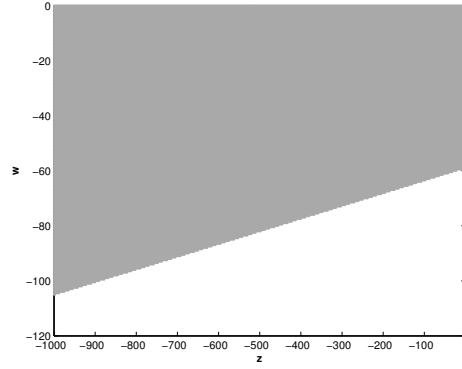
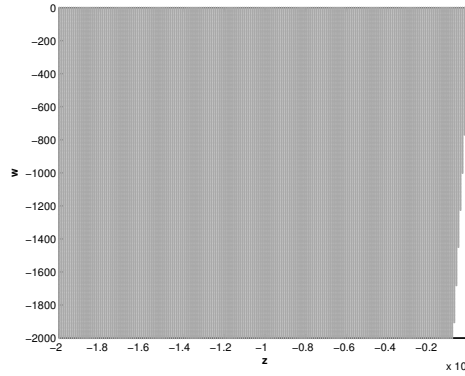
The $R(z, w)$ is called the stability matrix of the method. Now, the method is stable if $\rho(R(z, w)) < 1$ and the stability region of the method is $\mathcal{R} = \{(z, w) \in \mathbb{C} \times \mathbb{C} : \rho(R(z, w)) < 1\}$. Here, the term G_n does not influence stability. The stability function of the method with respect to (34) is defined as

$$p(z, w; \lambda) = \det(\lambda \mathbf{I}_{2r+m} - R(z, w)). \quad (44)$$

To investigate the stability properties of the exact MHCM, it is more convenient to work with the polynomial obtained by multiplying the stability function (44) by its denominator. The resulting polynomial will be denoted by the same symbol $p(z, w; \lambda)$. This polynomial takes the form

$$p(z, w; \lambda) = \sum_{i=0}^{2r+m} p_i(z, w) \lambda^i, \quad (45)$$

where $p_i(z, w)$, $i = 0, 1, \dots, 2r + m$ are polynomials of degree less than or equal to m . Denoting the roots of the polynomial $p(z, w; \lambda)$ by $\lambda_1, \lambda_2, \dots, \lambda_{2r+m}$, the absolute stability region of the method is then defined by

Figure 1: Stability region with $r = 2$, $m = 2$, $c = [\frac{8}{10}, 1]$.Figure 2: Stability region with $r = 3$, $m = 2$, $c = [\frac{19}{20}, 1]$.

$$\mathcal{R} = \{(z, w) \in \mathbb{C}^- \times \mathbb{C}^- : |\lambda_i(z, w)| < 1, \quad i = 1, 2, \dots, 2r + m\}.$$

We did not find A_0 -stable methods within this class, but wide stability regions exist. For example, in the cases $r = 2$, $m = 2$ with collocation parameters $c = [\frac{8}{10}, 1]$, and $r = 3$, $m = 2$ with collocation parameters $c = [\frac{19}{20}, 1]$, the regions of stability are unbounded. These regions are given in Figures 1 and 2, respectively.

5 Numerical results

In this section, to check the numerical performance of the method, we have considered a variety of problems. Here, the starting values $y_1, \dots, y_{r-1}$ are obtained by a one step MHCs of the same order of the present method.

In practice, we need quadrature rules to obtain the numerical solutions. For this propose, we have to apply the rules that preserve order of the main

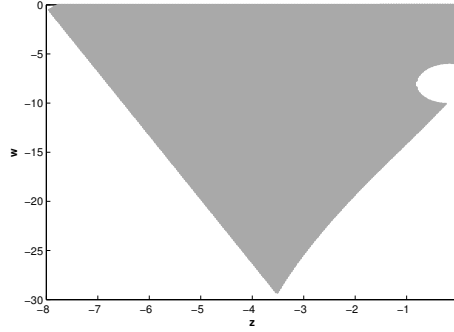


Figure 3: Stability region with $r = 3$, $m = 2$, $c = [\frac{2}{3}, 1]$.

method. A suitable choice is the 3-times Romberg quadrature formula with the Simpson rule [25]. In this rule that is of order 8, the nodes and weights for integration in the interval $[0, h]$ are

$$\zeta_i = \frac{ih}{8}, \quad i = 0, 1, \dots, 8$$

and

$$b = \frac{1}{5760} [217 \ 1024 \ 352 \ 1024 \ 436 \ 1024 \ 352 \ 1024 \ 217].$$

In what follows, we describe details of the implemented methods:

- **Method 1:** MHCM of convergence order 5 with $r = 2$, $m = 2$, and collocation parameters $c_1 = \frac{2}{3}$ and $c_2 = 1$.
- **Method 2:** MHCM of convergence order 7 with $r = 3$, $m = 2$, and collocation parameters $c_1 = \frac{2}{3}$ and $c_2 = 1$, with bounded stability region. (see Figure 3)
- **Method 3:** MHCM of convergence order 7 with $r = 3$, $m = 2$, and collocation parameters $c_1 = \frac{19}{20}$, and $c_2 = 1$, with unbounded stability region.
- **Method 4:** Multistep collocation method [8] of convergence order 4 with $r = 3$, $m = 2$, and collocation parameters $c_1 = \frac{7}{10}$ and $c_2 = 1$.

Computational experiments are done by applying the Methods 1–4 on the following problems:

I. The linear VIDE

$$y'(t) = 1 + 2t - y(t) + \int_0^t \tau(1 + 2\tau)e^{\tau(t-\tau)} y(\tau) d\tau, \quad t \in [0, 2],$$

Table 2: The results for problem I.

	k	4	5	6	7	8	9
Method 1	cd	2.74	4.20	5.69	7.18	8.68	10.19
	$p(h)$		4.86	4.93	4.97	4.98	4.99
Method 3	cd	4.02	6.06	8.14	10.23	12.33	14.44
	$p(h)$		6.77	6.90	6.96	6.98	6.99
Method 4	cd	2.94	4.07	5.22	6.41	7.60	8.81
	$p(h)$		3.71	4.84	3.92	3.96	3.98

$$y(0)=1,$$

with the exact solution $y(t) = e^{t^2}$.

II. The nonlinear VIDE

$$y'(t) = -t - \frac{1}{(1+t)^2} + \frac{1}{y(t)} \ln\left(\frac{2+2t}{2+t}\right) + \int_0^t \frac{d\tau}{1+(1+t)y(\tau)}, \quad t \in [0, 4],$$

$$y(0)=1,$$

with the exact solution $y(t) = \frac{1}{1+t}$.

III. The stiff nonlinear VIDE

$$y'(t) = -\frac{\lambda}{2}y(t) + 1 + \frac{\lambda}{2}t + \frac{\lambda}{2}te^{-t^2} - \lambda \int_0^t \tau e^{-y^2(\tau)} d\tau, \quad t \in [0, 4],$$

$$y(0)=0,$$

with the exact solution $y(t) = t$.

We have implemented the methods with a fixed stepsize $h = \frac{T}{2^k}$, with several integer values of k . In the following tables, the maximal end point error is written as 10^{-cd} , where cd is the number of correct significant digits. Also, a numerical estimation of the order of convergence of the methods is computed by the formula $p(h) = \log_2(\frac{e(2h)}{e(h)})$, where $e(h)$ is the maximal absolute end point error.

The results in Tables 2 and 3 confirm the proved convergence order. In Table 4, we show the effect of linear stability properties of the methods in solving stiff problems. For Method 2, the interval of absolute stability on real line is bounded. When $z = \lambda h$ or $w = \lambda h^2$ lies out of this interval, increasing of absolute error is evidently seen while for Method 3, with unbounded stability region, better results are obtained.

Table 3: The results for problem **II**.

	k	5	6	7	8	9
Method 1	cd	6.45	7.85	9.31	10.79	12.28
	$p(h)$		4.67	4.84	4.92	4.96
Method 3	cd	7.47	9.33	11.31	13.36	15.43
	$p(h)$		6.17	6.57	6.78	6.89
Method 4	cd	4.61	5.67	6.77	7.92	9.06
	$p(h)$		3.47	3.65	3.82	3.91

Table 4: Comparison of the methods (absolute errors of the methods for problem **III** with $\lambda = 500$).

t	Method 2		Method 3	
	$k = 7$	$k = 8$	$k = 7$	$k = 8$
0.25	1.44E-20	1.28E-22	3.25E-20	1.38E-22
0.5	5.69E-19	8.35E-22	2.14E-19	8.58E-22
0.75	2.20E-17	1.89E-21	4.79E-19	1.90E-21
1	6.21E-16	2.93E-21	7.34E-19	2.90E-21
1.5	4.86E-13	7.34E-21	1.82E-18	7.34E-21
2	4.24E-10	1.71E-20	4.28E-18	1.72E-20
3	3.47E-04	3.09E-20	7.72E-18	3.09E-20
3.5	3.13E-01	3.57E-20	9.03E-18	3.61E-20
4	2.73E+02	4.13E-20	1.03E-17	4.13E-20

6 Conclusion

The introduced methods for VIDEs, based on Hermite–Birkhoff interpolation, use of the approximated values of the solution in the m collocation points and the approximated values of the solution as well as its first derivative in the r previous steps. Using of this technique caused to get methods of higher orders and also with extensive stability regions. As we showed in Table 4, the extensive stability region of the method let us to make a sensible choice for the steplength of the algorithm for solving stiff equations.

7 Acknowledgments

This paper is published as part of a research project supported by the University of Tabriz Research Affairs Office.

References

1. Baker, C.T.H., Makroglou, A. and Short, E. *Regions of stability in the numerical treatment of Volterra integro-differential equations*, SIAM J. Numer. Anal. 16 (1979), 890–910.
2. Brunner, H. *Implicit Runge-Kutta-Nystrom methods for general second-order Volterra integro-differential equations*, Comput. Math. Appl. 14 (1987), 549–559.
3. Brunner, H. *Collocation methods for Volterra integral and related functional equations*, Cambridge University Press, 2004.
4. Brunner, H. and Lambert, J.D. *Stability of numerical methods for Volterra integro-differential equations*, Computing (Arch. Elektron. Rechnen.) 12 (1974), 75–89.
5. Brunner, H. and Van der Houwen, P.J. *The Numerical solution of Volterra equations*, CWI monographs, vol. 3, North Holland, Amsterdam, 1986.
6. Butcher, J.C. *Numerical methods for ordinary differential equations*, Third edition. With a foreword by J. M. Sanz-Serna. John Wiley & Sons, Ltd., Chichester, 2016.
7. Cardone, A., Conte, D. and Paternoster, B. *A family of multistep collocation methods for Volterra integro-differential equations*, AIP Conference Proceedings, 1168 (2009), 358–361.
8. Cardone, A. and Conte, D. *Multistep collocation methods for Volterra integro-differential equations*, Appl. Math. Comput. 221 (2013), 770–785.
9. Conte, D. and Paternoster, B. *Multistep collocation methods for Volterra Integral equations*, Appl. Numer. Math. 59 (2009), 1721–1736.
10. Cushing, J.M. *Integrodifferential equations and delay models in population dynamics*. Lecture Notes in Biomathematics, vol. 20. Berlin-Heidelberg-New York: Springer 1977.
11. Dahlquist, G. *A special stability problem for linear multistep methods*, Nordisk Tidskr. Informationsbehandling (BIT) 3 (1963), 27–43.
12. Fazeli, S. and Hojjati, G. *Numerical solution of Volterra integro-differential equations by superimplicit multistep collocation methods*, Numer. Algor. 68 (2015), 741–768.
13. Fazeli, S., Hojjati, G. and Shahmorad, S. *Multistep Hermite collocation methods for solving Volterra integral equations*, Numer. Algor. 60 (2012), 27–50.

14. Ferguson D. *The question of uniqueness for G.D. Birkhoff interpolation problems*, J. Approx. Theory, 2 (1969), 1–28.
15. Ferguson, D. *Some interpolation theorems for polynomials*, J. Approx. Theory, 9 (1973), 327–348.
16. Grimaldi R.P. *Discrete and combinatorial mathematics: An Applied Introduction*, Reading, Mass: Addison-Wesley, 1994.
17. Hairer, E. and Wanner, G. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems*, Springer-Verlag, Berlin, 1996.
18. Hale, J.K. *Functional differential equations*, Applied Mathematical Sciences, Vol. 3. Springer-Verlag New York, New York-Heidelberg, 1971.
19. Linz, P. *Linear multistep methods for Volterra integro-differential equations*, J. Ass. Comput. Mach. 16 (1969), 295–301.
20. Mahdi, H., Abdi, A. and Hojjati, G. *Efficient general linear methods for a class of Volterra integro-differential equations*, Appl. Numer. Math. 127 (2018), 95–109.
21. Mahdi, H. Hojjati, G. and Abdi, A. *Explicit general linear methods with a large stability region for Volterra integro-differential equations*, Math. Model. Anal. 24 (2019), 478–493.
22. Matthys, J. *A-stable linear multistep methods for Volterra integro-differential equations*, Numer. Math. 27 (1976), 85–94.
23. Miller, R.K. *Almost-periodic behavior of solutions of nonlinear Volterra system*, Quart. Appl. Math. 28 (1971), 553–570.
24. Schoenberg, I.J. *On Hermite–Birkhoff interpolation*, J. Math. Anal. Appl. 16 (1966), 538–543.
25. Stoer, J. and Bulirsch, R. *Introduction to numerical analysis*, Springer, 2002.
26. Volterra, V. *Theory of functional and of integral and integro-differential equations*, Moscow, Nauka, 1982.
27. Wolkenfelt, P.H.M. *The construction of reducible quadrature rules for Volterra integral and integro-differential equations*, IMA J. Numer. Anal. 2 (1982), 131–152.



The most probable allocation solution for the p -median problem[†]

M. Zaferanieh*

Abstract

The most important purpose in location problems is usually to locate some facilities and allocate the demands of nodes so that the total transportation cost of the network is minimized. However, in real networks, there are some other influencing factors, aside from the transportation costs, for determining the allocation mode. In this paper, a minimum information approach is applied to the capacitated p -median problem to estimate the most likely allocation solution based on some prior probabilities. Indeed, the most probable solution is achieved through minimizing a log-based objective function, while the total transportation cost should be less than or equal to a predetermined budget. The problem is solved by using a decomposition method combined with the Karush–Kuhn–Tucker optimality conditions, and some numerical examples are provided to verify the added value of the proposed model and solution approach

AMS(2010): Primary 90BXX; Secondary 90B80, 90B18.

Keywords: Network; Location problems; Capacitated p -median; Benders decomposition; Minimum information.

1 Introduction

The p -median problem is one of the known subjects in the field of locating facilities with a minisum objective function, where its purpose is to locate

*Corresponding author

Received 1 January 2020; revised 2 March 2020; accepted 11 August 2020

Mehdi Zaferanieh

Department of Applied Mathematics, Faculty of Mathematics and computer sciences, Hakim Sabzevari University, Sabzevar, Iran. e-mail: m.zaferanieh@hsu.ac.ir

[†] This article was suggested by the Scientific Committee of the Third Iranian Seminar on Control and Optimization for publication in IJNAO, which was accepted after independent review.

p facilities and allocate the demands of nodes so that the total demand-weighted distance is minimized. ReVelle and Swain [26] introduced the first integer linear programming formulation for the p -median, which has played an important role in solving the problem by applying such exact methods as the branch and bound [24] and Lagrangian dual relaxation [5]. Hakimi [18] proved that the p -median problem on the general networks is NP -hard; however, it has at least one nodal solution. Kariv and Hakimi [20] suggested an $O(p^2n^2)$ time algorithm to solve the p -median problem on the tree networks. Then, Tamir [30] improved the time complexity on the tree networks to $O(pn^2)$. For a complete bibliography on the p -median problem; see [25].

In the classical p -median model, which is sometimes referred to as the uncapacitated p -median ($UCpM$) problem, there is no constraint on the capacities of facilities; see [17]. In the case of limited capacities, the problem is usually called the capacitated p -median (CpM) problem. Both the $UCpM$ and CpM problems are NP -hard, although considering capacity limitations makes the latter problem more difficult; see [14]. A column generation approach, along with a Lagrangian/surrogate relaxation technique has been applied by Lorena and Senne [21] for solving the CpM problem. Ceselli and Righini [8] presented a branch and price algorithm that utilizes a column generation, heuristics, and branch-and-bound to determine the optimal solutions. Such Meta-heuristic methods as a Hybrid scatter search and path relinking algorithm [10], variable neighborhood search [12], neural network [27], and genetic algorithm [23] have been recently applied to solve the CpM problem.

When there is a lack of information in the network, the maximum entropy (ME) principle or its extended concept, the minimum information theory, is used to consider the solution with the most unbiased probability distribution; see [22]. In the ME models, there is no assumption about the missing information, and all possible states consistent with the system conditions are considered equally; then the most likely one is selected; see [9, 19]. The application of ME to the location problems was employed by Teye, Bell, and Bliemer [32, 33] to solve a multiuser inter-modal terminal (IMT) location problem wherein there is no priority to use an IMT . Based on the ME principle, the most unbiased probability distribution was achieved through considering all possible states of IMT usage and selecting the most likely one consistent with the problem constraints. Teye, Bell, and Bliemer [31] also proposed an ME approach for locating competitive multiuser freight facilities in general, and inland multiuser IMT s, in particular, when the multiple users have the choice whether to use a facility or not.

The ME approach to the origin-destination ($O - D$) matrix estimation problems was studied by Van Zuylen and Willumsen [34] to determine the most probable $O - D$ demands consistent with the available traffic counts. Abareshi, Zaferanieh, and Keramati [2] introduced an ME path flow estimator for disaggregated flows between $O - D$ pairs with a prespecified level for each disaggregation.

When, in the network, there is some prior information, but insufficient, it is reasonable to choose a solution that adds the least extra information to the available knowledge, which results in the most possible unbiased probability distribution. The minimum information (MI) approach provides an extended measure of the likelihood of a certain macro-state on the existence of some appropriate micro-state space; see [28]. Abareshi and Zaferanieh [1] proposed a new *bi*-level p -median problem in which the total cost of locating facilities and serving demands was minimized through the upper level while the MI approach was applied in the lower level to determine the most unbiased allocation solution based on some prior partial information. The prior information was given in terms of the probabilities of serving the demands of client nodes by different facilities, which might be determined by considering some attributes such as distance, locality, and geographical features, taken into account by multiple attributes decision making procedures; see [1, 29].

In this paper, considering some prior probabilities for serving the demands of nodes, we attempt to locate p capacitated facilities and determine the most probable allocation solution, while the total transportation cost should be less than or equal to a predetermined budget. The model is formulated as a mixed-integer nonlinear programming ($MINLP$) problem, which is known to be NP -hard. Applying such alternative methods as the Lagrangian dual approach or Benders decomposition algorithm to separate the model into some easier subproblems, the solution could be iteratively obtained.

The generalized Benders decomposition (GBD) algorithm has been suggested by Geoffrion [16] as a solution procedure for certain nonlinear programming (NLP) and $MINLP$ problems. The decomposition procedure was first developed by Benders [6] for the solution of mixed-variable programming problems. However, some restrictions regarding the convexity and other properties of the involved functions were considered. Floudas, Aggarwal, and Ciric [13] studied this technique and revealed its potential for applying in a chemical process design. They also proposed a computational implementation to reach the global optimal solution for nonconvex NLP and $MINLP$ problems. This claim was sustained by solving different examples, though it has not been proven mathematically.

Bagajewicz and Manousiouthakis [3] investigated the importance of two properties defined by Geoffrion [16], called *property (P)* and *L-dual-adequacy*, for the proper application of the GBD algorithm. They solved several examples satisfying or violating these properties and analyzed the obtained solutions. Through examples and mathematical analysis, they showed that dual gaps may prevent this procedure from converging to the global solution, and therefore only bounds on the global optimum might be obtained. Here, we apply the GBD algorithm studied in [3] to solve the proposed $MINLP$ model of the p -median problem in the MI approach.

The remainder of this paper is organized as follows: In Section 2, the model of the CpM problem in the MI approach is introduced. Using the scheme proposed by Bagajewicz and Manousiouthakis [3], the GBD algo-

Table 1: The notation used in the model

Notation	Definition
N	$\{v_1, \dots, v_n\}$, the set of nodes
E	The set of links
I	The index set of client nodes
J	The index set of candidate points for establishing facilities
y_j	The binary variable representing whether node $v_j, j \in J$ is used to locate a facility or not
c_j	The capacity of the facility at node $v_j, j \in J$
w_i	The total demand of the client node $v_i, i \in I$
p_{ij}	The probability of serving the demand of node $v_i, i \in I$ by the facility at node $v_j, j \in J$
x_{ij}	The decision variable representing the amount of demand of node $v_i, i \in I$ provided by the facility at node $v_j, j \in J$
d_{ij}	Per unit cost to serve the demand of node $v_i, i \in I$ by the facility at node $v_j, j \in J$
B	The total available budget

rithm and the Karush–Kuhn–Tucker (*KKT*) optimality conditions are applied in Sections 3 and 4 whereby the problem is reduced to easier subproblems. The feasibility and optimality constraints are added iteratively and the upper and lower bounds are updated. The convergence of the *GBD* algorithm on the proposed problem is also established in Section 4. To investigate the added value of the proposed model, some numerical examples are provided in Section 5, while the efficiency of the implemented algorithm is compared with some existing *MINLP* solvers by applying the performance profile test proposed by Dolan and More [11]. The summary and conclusions are given in the last section.

2 CpM problem with the most likely allocation solution

In this section, the *CpM* model applying the *MI* approach, to determine the most probable allocation solution, is proposed. Consider a network $G = (N, E)$, where the other frequently used notations are listed in Table 1.

To apply the concept of *MI* to the *CpM* problem, consider several demand nodes $v_i, i \in I$ with w_i clients that should be served by the facilities at candidate nodes $v_j, j \in J$. Also, let x_{ij} be the amount of clients of the demand node v_i that are served by the facility at node v_j , where its corresponding probability is given by p_{ij} . Then, using the multinomial distribution, the probability of observing an allocation solution with the components x_{ij} is given as follows (see [1]):

$$E(x) = \prod_i \left(\frac{w_i!}{\prod_j x_{ij}!} \prod_j p_{ij}^{x_{ij}} \right).$$

Following the scheme presented by Abareshi and Zaferanieh [1], by taking the logarithm, using Stirling's approximation, removing constant values, and reducing the problem to the minimum case, the model of the minimum infor-

mation capacitated p -median ($MICpM$) problem is introduced as follows:

$$MICpM : \min_{x,y} Z_0(x,y) = \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij}), \quad (1)$$

$$\sum_{j \in J} y_j = p, \quad (2)$$

$$\sum_{j \in J} x_{ij} = w_i \quad \text{for all } i \in I, \quad (3)$$

$$\sum_{i \in I} x_{ij} \leq y_j c_j \quad \text{for all } j \in J, \quad (4)$$

$$\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B, \quad (5)$$

$$y_j \in \{0, 1\}, \quad x_{ij} \geq 0 \quad \text{for all } i \in I, j \in J. \quad (6)$$

Constraint (2) assures that the number of established facilities is equal to p and constraints (3) practically guarantee that the demands of all nodes v_i are supplied. Constraints (4) make all open facilities serve the demands less than or equal to their capacities. In addition, constraint (5) states that the total spent cost does not exceed the available budget B . Due to the existing of nonlinear terms and mixed-integer variables, the $MICpM$ problem (1)–(6) is NP -hard; see [14]. In the following, the GBD algorithm is introduced whereby the problem is reduced to smaller subproblems to which the constraints are added iteratively to update the upper and lower bounds.

3 Generalized Benders decomposition algorithm

Here, the GBD algorithm studied by Bagajewicz and Manousiouthakis [3], for general $MINLP$ is stated, while its application to the $MICpM$ problem (1)–(6) is examined through the next section. Consider the following problem

$$\begin{aligned} & \min_{x,y} F(x,y), \\ & s.t. \quad G(x,y) \leq 0, \\ & x \in X, \quad y \in Y. \end{aligned} \quad (7)$$

The projection of the above problem on Y was proposed by Geoffrion [16] as

$$\begin{aligned} & \min_y v(y), \\ & s.t. \quad v(y) = \min_{x \in X} \{F(x,y) : s.t. G(x,y) \leq 0\}, \end{aligned} \quad (8)$$

$$y \in Y \cap V,$$

where $V = \{y : G(x, y) \leq 0 \text{ for some } x \in X\}$. Introducing the following two problems, Geoffrion [16] proposed a decomposition of (8) when X is a convex set, and the functions $F(x, y)$ and $G(x, y)$ are convex with respect to the variable x :

[1.] Primal problem:

$$\begin{aligned} & \min_x F(x, \bar{y}), \\ & s.t. \quad G(x, \bar{y}) \leq 0, \\ & \quad x \in X, \end{aligned} \tag{9}$$

where $\bar{y}$ is an arbitrary but fixed point in Y .

[2.] Master problem:

$$\begin{aligned} & \min_{y \in Y} \{ \max_{u \geq 0} \{ \min_{x \in X} F(x, y) + u^t G(x, y) \} \}, \\ & s.t. \quad \min_{x \in X} \{ \lambda^t G(x, y) \} \leq 0 \quad \text{for all } \lambda \in \Lambda, \end{aligned} \tag{10}$$

where $\Lambda = \{ \lambda \in \mathbb{R}^m : \lambda \geq 0, \sum_{i=1}^m \lambda_i = 1 \}$ (m is the size of vector G).

Bagajewicz and Manousiouthakis [3] pointed out that the master problem is equivalent to the projection (8), in the case, where X is a convex set and $F(x, y)$ and $G(x, y)$ are convex functions with respect to the variable x . In addition, they showed that the master problem is also equivalent to the following problem:

$$\min_{y, y_0} y_0, \tag{11}$$

$$s.t. \quad L^*(y, u) = \min_{x \in X} \{ F(x, y) + u^t G(x, y) \} \leq y_0 \quad \text{for all } u \geq 0, \tag{12}$$

$$L_*(y, \lambda) = \min_{x \in X} \{ \lambda^t G(x, y) \} \leq 0 \quad \text{for all } \lambda \in \Lambda. \tag{13}$$

Geoffrion [16] suggested to solve a relaxed version of problem (11)–(13) in which all but a few constraints are ignored. Indeed, in each iteration, the constraints corresponding to a subset of vectors u and λ , instead of all of them, are considered. Due to ignoring some of constraints, the obtained problem is called the relaxed master problem; see [3]. Since constraints are continuously added, the optimal values of this problem form a monotone nondecreasing sequence as the lower bounds to problem (7). Next, the steps of the *GBD* algorithm in the general form are represented.

3.1 General form of the *GBD* algorithm

Step 1. Let a point $\bar{y} \in Y \cap V$ be available. Solve the primal problem (9) and obtain the optimal solution x^* and the optimal multiplier vector u^* . Put the counters $K^f = 1$, $K^i = 0$. Set $UB = F(x^*, \bar{y})$. Select a tolerance $\epsilon \geq 0$ and set $u^{(1)} = u^*$. Finally, determine the function $L^*(y, u^{(1)})$.

Step 2. Solve globally the current relaxed master problem:

$$\begin{aligned} & \min_{y, y_0} y_0, \\ & s.t. \quad L^*(y, u^{(k_1)}) \leq y_0, \quad k_1 = 1, \dots, K^f, \quad (\text{optimality constraint}) \\ & \quad \quad L_*(y, \lambda^{(k_2)}) \leq 0, \quad k_2 = 1, \dots, K^i, \quad (\text{feasibility constraint}). \end{aligned}$$

Let $(\hat{y}, \hat{y}_0)$ be the globally optimal solution. Set $LB = \hat{y}_0$. If $UB \leq LB + \epsilon$, then terminate.

Step 3. Solve globally the primal problem (9) using $\bar{y} = \hat{y}$.

Step 3.a. Feasible primal problem: Solve the primal problem (9) by using $\bar{y}$. If $v(\bar{y}) \leq LB + \epsilon$, then terminate. Otherwise, determine the optimal multiplier vector u^* , set $K^f = K^f + 1$ and $u^{(K^f)} = u^*$. If $v(\bar{y}) < UB$, then set $UB = v(\bar{y})$. Finally, determine the function $L^*(y, u^{(K^f)})$ and return to Step 2.

Step 3.b. Infeasible primal problem: Determine a set of values of $\lambda^* \in \Lambda$ satisfying $\min_{x \in X} \{(\lambda^*)^t G(x, \bar{y})\} > 0$. Set $K^i = K^i + 1$ and $\lambda^{(K^i)} = \lambda^*$. Determine the function $L_*(y, \lambda^{(K^i)})$. Return to Step 2.

Remark 1. To determine the value of λ^* in Step 3.b, the following problem should be solved (see [3]):

$$\begin{aligned} & \min_{x, \alpha} \alpha, \\ & s.t. \quad G(x, \bar{y}) - \alpha \underline{1} \leq 0, \\ & \quad \quad x \in X, \quad \alpha \in \mathbb{R}, \end{aligned}$$

where $\underline{1} = (1, 1, \dots, 1)^t$.

Remark 2. If the functions $F(x, y)$ and $G(x, y)$ are separable with respect to x and y and convex with respect to x , the global optimal solutions for x within constraints (12) and (13) are obtained independently of y ; see [3]. In this case, functions $L^*(y, u^{(k^f)})$ and $L_*(y, \lambda^{(k^i)})$ are defined in explicit forms.

Geoffrion stated some conditions ([16, Theorem 2.4] for the finite discrete set Y and [16, Theorem 2.5] for the infinite cardinality set Y) for problem (7),

where the lowest solution of the primal and the global solution of the master would approach one another, and consequently, the global optimum of the overall problem would be provided within a prespecified tolerance. However, when the convexity in x does not hold, dual gaps may exist. Nevertheless, the global solution to the master problem will still provide a valid lower bound to problem (7). Next, an application of the *GBD* algorithm on problem (1)–(6) is investigated.

4 Implementing the *GBD* algorithm on the *MICpM* problem

To apply the proposed *GBD* algorithm, we reformulate problem (1)–(6) as follows:

$$\begin{aligned} \min_{x,y} Z_0(x,y) &= \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij}), \\ \sum_{i \in I} x_{ij} &\leq y_j c_j \quad \text{for all } j \in J, \\ x \in X &= \{x_{ij} \geq 0 : \sum_{j \in J} x_{ij} = w_i \text{ for all } i \in I, \sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B\}, \\ y \in Y &= \{y : y_j \text{ is binary and } \sum_{j \in J} y_j = p\}. \end{aligned} \tag{14}$$

Note that the set X as well as functions $Z_0(x,y)$ and $G_j(x,y) = \sum_{i \in I} x_{ij} - y_j c_j$, for all $j \in J$, are convex with respect to x . Following the steps of Algorithm 3.1, the implementation of the *GBD* algorithm to solve the *MICpM* problem is summarized as below.

4.1 *GBD* algorithm on the *MICpM* problem

Step 1 Select a tolerance $\epsilon \geq 0$ and a feasible vector $\bar{y} \in Y$ for which there exists a feasible solution $x \in X$ satisfying in (14).

Step 1.a. Solve the corresponding primal problem (15)–(16)

$$\begin{aligned} \min_{x \in X} Z_0(x, \bar{y}) &= \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij}), \\ \sum_{i \in I} x_{ij} &\leq \bar{y}_j c_j \quad \text{for all } j \in J, \end{aligned} \tag{15}$$

$$\sum_{i \in I} x_{ij} \leq \bar{y}_j c_j \quad \text{for all } j \in J, \tag{16}$$

and obtain the optimal solution x^* and the optimal multipliers u_j^* , $j \in J$ corresponding to constraints (16). Set $K^f = 1, K^i = 0$, $UB = Z_0(x^*, \bar{y})$ and vector $u^{(1)} = u^*$.

Step 1.b. Determine the optimal solution $\hat{x}$ corresponding to the following subproblem and set $x^{(K^f)} = \hat{x}$:

$$\min_{x \in X} \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij}) + \sum_{j \in J} u_j^{(K^f)} \sum_{i \in I} x_{ij}. \quad (17)$$

Step 2 Solve globally the current relaxed master problem:

$$\min_{y_0, y \in Y} y_0, \quad (18)$$

$$\begin{aligned} s.t. \quad L^*(y, u^{(k_1)}) &= \sum_{j \in J} \sum_{i \in I} x_{ij}^{(k_1)} (\ln x_{ij}^{(k_1)} - 1 - \ln p_{ij}) \\ &\quad + \sum_{j \in J} u_j^{(k_1)} \left(\sum_{i \in I} x_{ij}^{(k_1)} - c_j y_j \right) \leq y_0, \quad k_1 = 1, \dots, K^f, \\ L_*(y, \lambda^{(k_2)}) &= \sum_{j \in J} \lambda_{j \in J}^{(k_2)} \left(\sum_{i \in I} x_{ij}'^{(k_2)} - c_j y_j \right) \leq 0, \quad k_2 = 1, \dots, K^i, \lambda \in \Lambda. \end{aligned} \quad (19)$$

Note that in each iteration, the variables $u^{(k_1)}, x^{(k_1)}$, and $x'^{(k_2)}$ are specified fixed values that have been obtained during the previous steps of the algorithm; therefore, problem (18) is a linear mixed-binary model, which could be globally solved by such existing methods as branch and bound. Let $(\hat{y}, \hat{y}_0)$ be the global optimal solution. Set $LB = \hat{y}_0$. If $UB \leq LB + \epsilon$, then terminate.

Step 3. Solve globally the following problem to examine the feasibility of the primal problem (15)–(16) corresponding to $\bar{y} = \hat{y}$.

$$\begin{aligned} \min_{x, \alpha} \quad & \alpha, \\ \sum_i x_{ij} - c_j \bar{y}_j - \alpha & \leq 0 \quad \text{for all } j \in J, \\ x \in X, \alpha & \in \mathbb{R}. \end{aligned} \quad (20)$$

Step 3.a. If $\alpha \leq 0$, then the primal problem (15)–(16) is feasible. Solve the primal problem for the given vector $\bar{y}$ to find the optimal solution x^* . If $Z_0(x^*, \bar{y}) \leq LB + \epsilon$, then terminate. Otherwise, determine the optimal multiplier vector u^* corresponding to constraints (16), and set $K^f = K^f + 1$ and $u^{(K^f)} = u^*$. If $Z_0(x^*, \bar{y}) < UB$, then put $UB = Z_0(x^*, \bar{y})$. Finally, return to Step 1.b.

Step 3.b. If $\alpha > 0$, then the primal problem (15)–(16) is infeasible. Determine the optimal multipliers λ_j^* corresponding to constraints (20).

Set $K^i = K^i + 1$ and $\lambda^{(K^i)} = \lambda^*$. To find the optimal solution $\tilde{x}$, solve the following model:

$$\min_{x \in X} \sum_{j \in J} \lambda_j^{(K^i)} \sum_{i \in I} x_{ij}.$$

Put $x'^{(K^i)} = \tilde{x}$ and go to Step 2.

The purpose of the algorithm is to minimize the gap between the upper and lower bounds. Next, the convergence conditions stated by Geoffrion [16], are verified for problem (1)–(6).

Theorem 1. The GBD Algorithm 4.1 for the *MICpM* problem (1)–(6) terminates in a finite number of steps, for any given $\epsilon \geq 0$.

Proof. Since $Y = \{y : y_j \text{ is binary and } \sum_{j \in J} y_j = p\}$ is a finite discrete set, the convergence of Algorithm 4.1 should be established via [16, Theorem 2.4]. Indeed, the following conditions must hold:

- i. X is a nonempty convex set and functions $G_j(x, y) = \sum_{i \in I} x_{ij} - y_j c_j$ are convex on X for each fixed $y \in Y$. Further, the set $Z_y = \{z = (z_1, \dots, z_{|J|}) : G_j(x, y) \leq z_j \text{ for some } x \in X\}$ is closed for each fixed $y \in Y$.
- ii. Functions $Z_0(x, y)$ and $G_j(x, y)$ are convex on X for each fixed $y \in Y$. In addition, for each fixed $\bar{y} \in Y \cap V$, either the optimal value $Z_0(x, \bar{y})$ is infinite, or the primal problem (15)–(16) possesses an optimal multiplier vector.

Part i. Obviously, the set $X = \{x_{ij} \geq 0 : \sum_{j \in J} x_{ij} = w_i, \forall i \in I, \sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B\}$ is nonempty ($x_{ii} = w_i$, for all $i \in I$ and $x_{ij} = 0$ for $i \neq j$ is a feasible solution), closed and bounded. Noting the continuity of functions $G_j(x, y)$ on X , the closeness of Z_y is implied; see [16].

Part ii. The convexity of functions $Z_0(x, y) = \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij})$ and $G_j(x, y) = \sum_{i \in I} x_{ij} - y_j c_j$ with respect to x for each fixed y is straightforward by the definition. Following the saddle point optimality conditions stated in [4], the existence of optimal multiplier vector is inferred; see [15]. $\square$

The overall scheme of the algorithm is represented in Figure 1.

4.2 Solving the primal problem

The primal problem (15)–(16) is a convex nonlinear problem, which can be solved by applying the *KKT* optimality conditions. Let $\bar{y}$ be given. Consider the Lagrangian dual vectors u_j, γ_i , and η corresponding to constraints

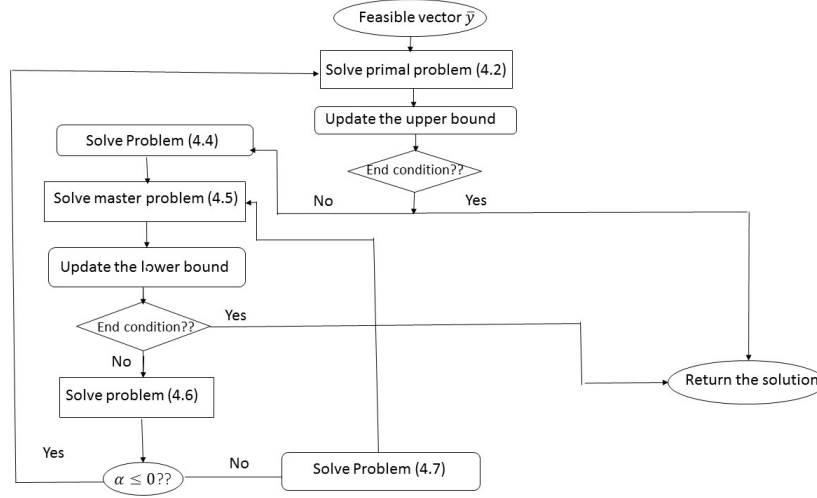


Figure 1: The overall framework of the proposed algorithm

(22), (23), and (24). Then Lemma 1 expresses the relationship between the primal and dual variables:

$$\min_x Z_0(x, \bar{y}) = \sum_{j \in J} \sum_{i \in I} x_{ij} (\ln x_{ij} - 1 - \ln p_{ij}), \quad (21)$$

$$\sum_{i \in I} x_{ij} \leq c_j \bar{y}_j \quad \text{for all } j \in J, \quad (22)$$

$$\sum_{j \in J} x_{ij} = w_i \quad \text{for all } i \in I, \quad (23)$$

$$\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B, \quad (24)$$

$$x_{ij} \geq 0 \quad \text{for all } i \in I, j \in J. \quad (25)$$

Lemma 1. The optimal solution of problem (21)–(25) is obtained by $x_{ij} = p_{ij} e^{-(u_j + \gamma_i + \eta d_{ij})}$, for all $j \in J, i \in I$, where u_j, γ_i , and η are the optimal multiplier vectors corresponding to constraints (22), (23), and (24), respectively.

Proof. The *KKT* optimality conditions of model (21)–(25) with respect to x obtain in the following expressions:

$$\ln x_{ij} - \ln p_{ij} + u_j + \gamma_i + \eta d_{ij} \geq 0, \quad (26)$$

$$x_{ij} (\ln x_{ij} - \ln p_{ij} + u_j + \gamma_i + \eta d_{ij}) = 0. \quad (27)$$

The above conditions state that one of the following cases holds in the optimality:

$$\begin{aligned} x_{ij} = 0 \text{ and } \ln x_{ij} - \ln p_{ij} + u_j + \gamma_i + \eta d_{ij} &\geq 0, \\ \rightarrow x_{ij} &\geq p_{ij} e^{-(u_j + \gamma_i + \eta d_{ij})} \geq 0, \end{aligned}$$

or

$$\begin{aligned} x_{ij} \geq 0 \text{ and } \ln x_{ij} - \ln p_{ij} + u_j + \gamma_i + \eta d_{ij} &= 0, \\ \rightarrow x_{ij} &= p_{ij} e^{-(u_j + \gamma_i + \eta d_{ij})} \geq 0. \end{aligned}$$

The former case holds if $p_{ij} = 0$; therefore, the optimal solution would be written as $x_{ij} = p_{ij} e^{-(u_j + \gamma_i + \eta d_{ij})} \geq 0$ in both cases. $\square$

Note that to obtain the optimal solution of problem (21)–(25), the relation obtained for x_{ij} must be considered together with the other optimality conditions of the problem, resulting in the following system to be solved:

$$x_{ij} = p_{ij} e^{-(u_j + \gamma_i + \eta d_{ij})} \quad \text{for all } i \in I, \quad j \in J, \quad (28)$$

$$\sum_{i \in I} x_{ij} \leq c_j \bar{y}_j \quad \text{for all } j \in J, \quad (29)$$

$$u_j \left(\sum_{i \in I} x_{ij} - c_j \bar{y}_j \right) = 0 \quad \text{for all } j \in J, \quad (30)$$

$$\sum_{j \in J} x_{ij} = w_i \quad \text{for all } i \in I, \quad (31)$$

$$\eta \left(\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} - B \right) = 0, \quad (32)$$

$$\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B, \quad (33)$$

$$\eta, u \geq 0. \quad (34)$$

To transform the nonlinear terms in the system of equations (28)–(34) into linear ones, a linearization method is applied. Setting $v_j = e^{-u_j}$, $s_i = e^{-\gamma_i}$, $k_{ij} = e^{-\eta d_{ij}}$ and noting that $u, \eta \geq 0$, the system of equations (28)–(34) is rewritten as follows:

$$\begin{aligned} x_{ij} &= p_{ij} v_j s_i k_{ij} \quad \text{for all } i \in I, \quad j \in J, \\ \sum_{i \in I} x_{ij} &\leq c_j \bar{y}_j \quad \text{for all } j \in J, \\ (v_j - 1) \left(\sum_{i \in I} x_{ij} - c_j \bar{y}_j \right) &= 0 \quad \text{for all } j \in J, \\ \sum_{j \in J} x_{ij} &= w_i \quad \text{for all } i \in I, \end{aligned} \quad (35)$$

$$(k_{ij} - 1)(B - \sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij}) = 0 \quad \text{for all } i \in I, j \in J, \quad (36)$$

$$\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B, \quad (37)$$

$$\begin{aligned} v_j \leq 1, \quad k_{ij} \leq 1 \quad & \text{for all } i \in I, j \in J, \\ v_j, s_i, k_{ij} > 0 \quad & \text{for all } i \in I, j \in J. \end{aligned} \quad (38)$$

Since $u_j = 0$ results in $v_j = 1$, for all $j \in J$, and visa versa. Therefore, constraints (30) and (35) are equivalent. The same statement holds for constraints (32) and (36). Using the approach introduced in [7], the product of variables $f_{ij} = h_{ij}k_{ij} = v_j s_i k_{ij}$ can be linearized by adding some constraints; see [1]. The final linear mixed-integer system that should be solved is equivalent to the following:

$$x_{ij} = p_{ij} f_{ij} \quad \text{for all } i \in I, j \in J, \quad (39)$$

$$\sum_{i \in I} x_{ij} \leq c_j \bar{y}_j \quad \text{for all } j \in J,$$

$$\sum_{j \in J} x_{ij} = w_i \quad \text{for all } i \in I,$$

$$s_i - h_{ij} \leq M r_j, \quad c_j \bar{y}_j - \sum_i x_{ij} \leq M(1 - r_j) \quad \text{for all } j \in J,$$

$$h_{ij} - f_{ij} \leq M q, \quad B - \sum_i \sum_j d_{ij} x_{ij} \leq M(1 - q) \quad \text{for all } i \in I, j \in J,$$

$$\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} \leq B, \quad (40)$$

$$f_{ij} \leq h_{ij}, \quad h_{ij} \leq s_i \quad \text{for all } i \in I, j \in J, \quad (41)$$

$$s_i, h_{ij}, f_{ij} > 0, \quad r_j, q \in \{0, 1\} \quad \text{for all } i \in I, j \in J,$$

in which M is a sufficiently large number. Problem (17) in Step 1.b can be similarly solved while the other problems in Steps 2 and 3 are linear or mixed-binary linear that would be solved by existing methods.

5 Numerical examples

In this section, the proposed *GBD* algorithm is implemented in *GAMS 23.7* environment to determine the most probable allocation solution on some small and medium-sized networks. During the steps of the algorithm, the obtained mixed-binary linear problem (18) is solved by such mixed-integer programming solvers as *CPLEX*.

Although by Theorem 1, the algorithm would converge after a finite number of iterations, implementing numerical examples revealed that the optimal solution would be obtained much sooner than the convergence. Indeed, after a number of iterations, the upper bound values might marginally change or even remain unchanged, which results in an optimal or near-optimal solution. Therefore, we have considered some other stopping criteria, by which the algorithm continues until no progress is achieved in the upper or lower bounds during a predetermined number of successive iterations, or the total number of iterations reaches its maximum value. The used stopping criteria are summarized below:

- i. To reach a solution with $UB \leq LB + \epsilon$, where ϵ is a predetermined tolerance.
- ii. Proceeding the algorithm with no improvement in the values of UB and LB for k successive iterations.
- iii. To reach the maximum number of iterations Max_{iter} .

Example 1. Consider the small network shown in Figure 2 wherein $I = J = N$ and the costs of links have been given next to them. The value of d_{ij} for each pair (i, j) is calculated by the shortest distance between i and j . The capacities and demands of nodes, (c, w) , are inserted in Table 2. To estimate the probabilities, the approximate distances with 5 additional criteria indicated by random integers, representing such geographical and local features as proximity to transportation services, security, and surrounding facilities, are considered. The compared weights of different nodes j for a certain node i with respect to the first attribute, that is, the approximate distance, are determined as follows:

$$\begin{cases} \frac{0.45}{n_i^1} & \text{if } j \in S_i^1 = \{j : d_{ij} \leq 25\}, \\ \frac{0.25}{n_i^2} & \text{if } j \in S_i^2 = \{j : 25 < d_{ij} \leq 35\}, \\ \frac{0.15}{n_i^3} & \text{if } j \in S_i^3 = \{j : 35 < d_{ij} \leq 45\}, \\ \frac{0.10}{n_i^4} & \text{if } j \in S_i^4 = \{j : 45 < d_{ij} \leq 55\}, \\ \frac{0.05}{n_i^5} & \text{if } j \in S_i^5 = \{j : 55 < d_{ij}\}, \end{cases}$$

where $n_i^q = |S_i^q|$, $q = 1, \dots, 5$. In addition, the compared weights of nodes with respect to the other 5 attributes $r = 2, \dots, 6$ are given in Table 2. We use the multiple attributes decision making procedures to calculate the probabilities p_{ij} for different weight vectors $Weight$, which defines the relative importance of 6 criteria; see [1, 29].

To see the effect of the available budget as well as emphasizing on different attributes, the solution of the $MICpM$ problem (1)–(6) for $p = 3, 2$ is estimated for $B = 15000$ and $B = 12000$, by applying weight vectors $Weight_1$ and $Weight_2$; see Table 3. The solutions are estimated via both the GBD algorithm with $Max_{iter} = 15$, $\epsilon = 100$, and $k = 5$ as well as the

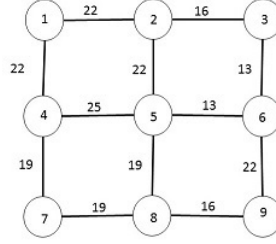


Figure 2: The grid network with 9 nodes

Table 2: The capacities and demands of nodes along with their weights with respect to criteria $r = 2, \dots, 6$

Nodes	1	2	3	4	5	6	7	8	9
c	300	200	350	320	250	250	350	100	200
w	50	60	80	40	90	100	80	60	50
Criteria	1	2	3	4	5	6	7	8	9
2	0.0541	0.0270	0.1351	0.1622	0.0811	0.1081	0.0541	0.1351	0.2432
3	0.0244	0.0488	0.1220	0.0976	0.1707	0.1951	0.2195	0.0732	0.0488
4	0.0455	0.1136	0.1591	0.0227	0.0682	0.1591	0.2045	0.1818	0.0455
5	0.2162	0.0270	0.0811	0.1351	0.1081	0.0541	0.2432	0.0270	0.1081
6	0.2286	0.0571	0.1429	0.0286	0.1143	0.0857	0.2571	0.0571	0.0286

“*Dicopt*” solver of *GAMS* 23.7, where the allocation solutions along with their objective function values (*OBJF*) are inserted in Table 3.

As it is seen, the optimal values of the objective function provided by the *GBD* algorithm are better than or at least equal to those provided by “*Dicopt*” solver. Note that, the optimal solution might change by changing the *GAMS* solver. However, examining different cases showed that the solutions obtained by “*Dicopt*” are usually better than or at least near to the other solvers of *GAMS* and hence have been reported in Table 3.

Referring to the estimated solutions, it can be realized if the capacities are large enough, the shares of the selected facilities in the *MICpM* allocation solution are proportional to their probabilities. For example, the probabilities $p(1, j)$, $j = 1, \dots, 9$ for *Weight*₁ are estimated as

$$p(1, :) = [0.1617, 0.1072, 0.1090, 0.1423, 0.0983, 0.0895, 0.1471, 0.0642, 0.0808].$$

Among the selected medians 3, 6, 7, using the *GBD* algorithm, the median 7 has the most probability and consequently the most share to provide the demand of node 1. Indeed, the higher is the probability p_{ij} , the more would be the share of the median v_j to provide the demand of node v_i . The same statement holds for medians 1, 3, 7 found by the *GAMS* solver. However, decreasing the budget B to 12000 results in a different allocation solution to assign as much demand as possible to the closer facilities; compare the column of $i = 6$ in cases $B = 12000$ and $B = 15000$ for the weight vector *Weight*₁ in the *GBD* solution.

For the weight vector *Weight*₂, wherein the criterion $r = 6$ is ranked as the most important one, the locations of 2 medians are to be selected. As it is seen, both the *GBD* and “*Dicopt*” solver determine the nodes 3, 7 for

Table 3: The allocation solution for the *MICpM* problem

Facility \ Node	1	2	3	4	5	6	7	8	9	<i>OBJF</i>	<i>Time</i>
$B = 15000$	$p = 3, Weight_1 = [0.3462 \ 0.1923 \ 0.1154 \ 0.0385 \ 0.1154 \ 0.1923]$										
	The <i>GBD</i> algorithm										
3	16.739	27.739	36.479	8.315	26.951	37.234	12.206	13.266	13.765	2533.355	4
6	11.952	17.815	29.058	9.391	34.628	40.470	13.927	15.904	18.429		
7	21.308	14.446	14.463	22.294	28.421	22.296	53.867	30.831	17.806		
	GAMS solver " <i>Dicopt</i> "										
1	31.095	24.002	15.583	15.006	21.182	16.798	12.873	7.646	5.906	2583.869	8
3	8.491	27.644	54.607	5.016	36.048	60.544	6.251	12.448	19.226		
7	10.414	8.354	9.810	19.978	32.770	22.659	60.876	39.906	24.869		
$B = 12000$	$p = 3, Weight_1 = [0.3462 \ 0.1923 \ 0.1154 \ 0.0385 \ 0.1154 \ 0.1923]$										
	The <i>GBD</i> algorithm										
3	18.245	35.331	46.724	5.194	25.673	37.088	4.160	9.138	12.350	2598.169	4
6	9.937	17.310	28.391	7.690	43.242	52.847	6.222	14.362	21.675		
7	21.818	7.359	4.885	27.117	21.086	10.065	69.617	36.499	15.975		
	GAMS solver " <i>Dicopt</i> "										
1	39.397	34.999	18.405	14.815	14.836	9.744	8.310	4.658	2.983	2655.921	10
6	2.925	17.626	53.375	4.664	50.319	77.113	4.155	13.557	26.266		
7	7.678	7.375	8.219	20.521	24.845	13.144	67.535	41.785	20.751		
$B = 15000$	$p = 2, Weight_2 = [0.0714, \ 0.0714, \ 0.0714, \ 0.0714, \ 0.0714, \ 0.6429]$										
	The <i>GBD</i> algorithm										
3	20.035	41.620	63.996	7.688	41.899	65.449	6.953	13.305	19.109	2654.990	4
7	29.965	18.380	16.004	32.312	48.101	34.551	73.047	46.695	30.891		
	GAMS solver " <i>Dicopt</i> "										
1	49.154	32.910	5.434	38.559	27.072	6.799	75.123	21.472	3.478	2867.986	5
3	0.846	27.090	74.566	1.441	62.928	93.201	4.877	38.528	46.522		
$B = 12000$	$p = 2, Weight_2 = [0.0714, \ 0.0714, \ 0.0714, \ 0.0714, \ 0.0714, \ 0.6429]$										
	The <i>GBD</i> algorithm										
3	21.181	54.045	77.430	3.191	50.387	86.269	0.998	6.688	19.109	2778.247	4
7	28.819	5.955	2.570	36.809	39.613	13.731	79.002	53.312	30.891		
	GAMS solver " <i>Dicopt</i> "										
3	21.181	54.045	77.430	3.191	50.387	86.269	0.998	6.688	19.109	2778.247	4
7	28.819	5.955	2.570	36.809	39.613	13.731	79.002	53.312	30.891		

$B = 12000$ as the optimal locations, while for $B = 15000$, the *GBD* provides a better solution. Referring to Table 2, it is realized that the model finds the optimal location of facilities in nodes having more weights with respect to the attribute $r = 6$, while simultaneously the total transportation cost is limited to the given budget. In the case of enough capacities, the shares of the selected facilities to provide the demands of nodes are again based on their corresponding probabilities. In addition, reducing the budget changes the allocation solution so that the closer is the median, the larger will be the assigned share.

To analyze the convergence of the *GBD* algorithm, the variations of the upper and lower bounds through successive iterations for weight vector $Weight_1$ with $B = 15000$, $p = 3$ and $Weight_2$ with $B = 15000$, $p = 2$, are illustrated in Figures 3 and 4, respectively. Here, no limit has been considered on iterations, that is, the only stopping criterion is satisfying the case i and the other two ones are neglected. In Figure 3, the algorithm has stopped after 76 iterations, where the lower bound becomes greater than the upper bound. However, the upper bound values remain almost unchanged after 20 iterations. Indeed, the optimal value of the objective function (1) has been obtained in the 20th iteration. For the weight vector $Weight_2$ with $B = 15000$, $p = 2$ in Figure 4, the algorithm converges in the 7th iteration while the optimal value of the objective function (1) has been achieved in the first iteration.

Example 2. In the second example, the efficiency of the proposed algorithm on larger problems is investigated. First we consider a 100-node network, where the network topology, the lengths of links, and the demands and capacities of nodes are taken from the URL address <http://people.brunel.ac.uk/mastjjb/jeb/orlib/pmedinfo.html>. To calculate the probabilities p_{ij} , ap-

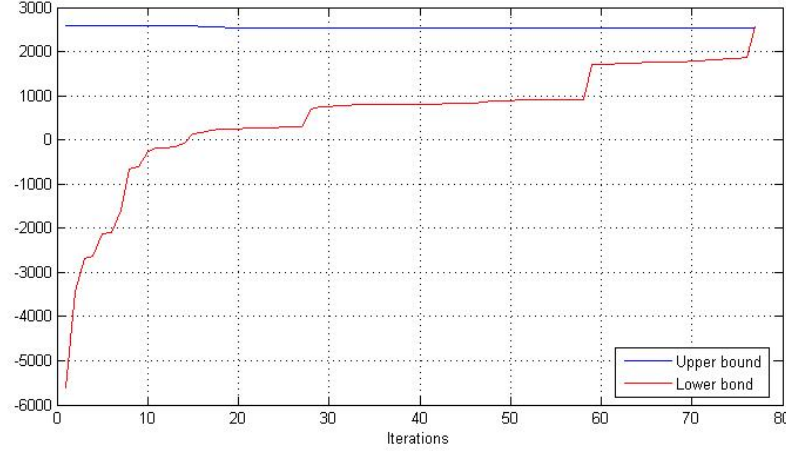


Figure 3: The variations of the upper and lower bounds for $Weight_1$, $B = 15000$, $p = 3$

Table 4: Comparison between GBD and $GAMS$ solvers

Solver	$B = 800000$, $p = 10$		$B = 900000$, $p = 10$		$B = 500000$, $p = 20$	
	Objective function	Running time	Objective function	Running time	Objective function	Running time
Knitro	—	—	—	—	—	—
Baron	—	—	—	—	—	—
Bonmin	40639.753	2077	41851.221	1004	40396.948	1006
Dicopt	41263.215	90	40557.247	16	40340.818	32
Sbb	—	—	40247.635	1005	—	—
Lindo global	—	—	—	—	—	—
Oqnlp	—	—	—	—	—	—
GBD	41929.25	26	41474.553	29	39942.716	26

proximate distances plus 5 additional criteria, indicated by random integers assigned to network nodes, are considered. The problem is solved for $p = 10$ and $B = 800000$, 900000 and $p = 20$, $B = 500000$, by applying different $MINLP$ solvers of $GAMS$ as well as the GBD algorithm. The stopping criteria for the GBD algorithm are the same as those used in Example ?? 1. The optimal value of the objective function and the elapsed running time in each case have been reported in Table 4. The sign (—) indicates that the solver has not been able to find any solution for the problem. As it is seen, the GBD algorithm generally provides an adequate solution in reasonable running time in comparison with the used solvers.

Next, we use the method proposed by Dolan and More [11] to analyze the performance profile of the GBD algorithm as well as three solvers *Bonmin*, *Dicopt*, and *Sbb*. To this purpose, we have considered 30 problems $p \in P$ taken from the URL address <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/pmedinfo.html> on networks with 100, 250, 300 and 400 nodes, and solved them by the three mentioned solvers $s = 1, 2, 3$ and the GBD algorithm, as well. Let

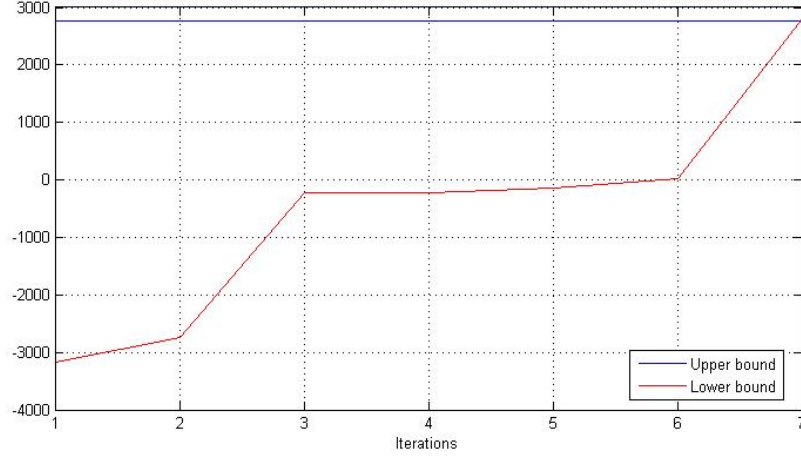


Figure 4: The variations of the upper and lower bounds for $Weight_2$, $B = 15000$, $p = 2$

$t_{s,p}$ = the running time required to solve problem p by solver s .

Referring to [11], the performance on problem $p \in P$ by solver s is compared with the best performance realized among all the used solvers on this problem. Therefore, we use the performance ratio

$$r_{s,p} = \frac{t_{s,p}}{\min\{t_{s,p} : s = 1, 2, 3\}}.$$

A parameter $r_M \geq r_{s,p}$ for all p, s is chosen, where $r_{s,p} = r_M$ if and only if the solver s does not solve problem p . Then, the probability for the solver s that the performance ratio $r_{s,p}$ is less than or equal to a factor $\tau \in \mathbb{R}$, is demonstrated by

$$\rho_s(\tau) = \frac{1}{|P|} \text{size}\{p, r_{s,p} \leq \tau\}.$$

The function $\rho_s(\cdot)$ is the (cumulative) distribution function for the performance ratio and is termed as the performance profile; see [11]. Figure 5 illustrates the comparison between the performance profiles of the three solvers and the *GBD* algorithm for $\tau = 1, \dots, 200$. As it is realized, the *GBD* algorithm has much better performance in comparison with the applied *GAMS* solvers. In addition, even for those problems $p \in P$, where the solvers could not provide any solution, the *GBD* algorithm reached a reasonable upper bound in less than 200 seconds.

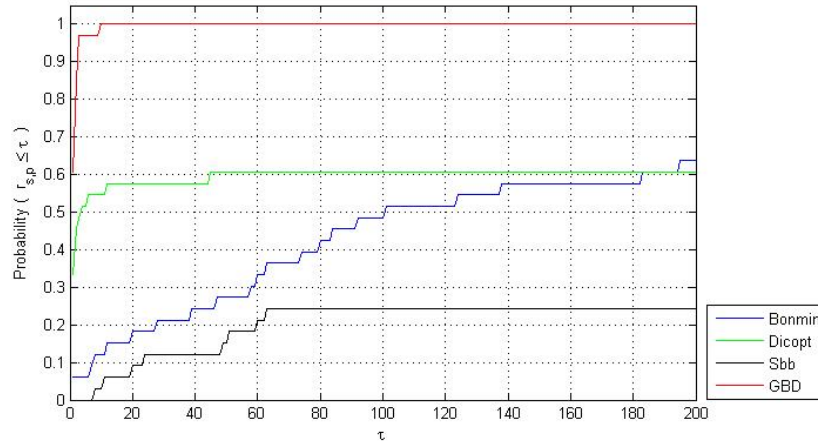


Figure 5: Comparison between the performance profiles of the *GAMS* solvers and the *GBD* algorithm

6 Summary and conclusions

In this paper, we proposed a log-based model using the minimum information approach, for the capacitated p -median problem to estimate the most likely allocation solution, while the total transportation cost should be less than or equal to a predetermined budget. Using the generalized benders decomposition method, the proposed problem was reduced to easier subproblems to which some constraints were added iteratively, and the upper and lower bounds have been updated. The convergence of the algorithm was established analytically, while its efficiency in comparison to *MINLP* solvers was investigated practically by applying the performance profile test on some numerical examples.

Acknowledgements

Author is grateful to anonymous referees and editor for their constructive comments.

References

1. Abareshi, M. and Zaferanieh, M. *A bi-level capacitated p -median facility location problem with the most likely allocation solution*, *Transport. Res. B-Meth.* 123 (2019) 1–20.
2. Abareshi, M., Zaferanieh, M. and Keramati, B. *Path flow estimator in an entropy model using a nonlinear L -shaped algorithm*, *Netw. Spat. Econ.* 17(1) (2017) 293–315.
3. Bagajewicz, M.J. and Manousiouthakis, V. *On the generalized Benders decomposition*, *Comput. Chem. Eng.* 15(10) (1991) 691–700.
4. Bazaraa, M.S., Sherali H.D. and Shetty C.M. *Nonlinear programming: Theory and algorithms*, John Wiley and Sons; 2013 .
5. Beltran, C., Tadonki, C. and Vial, J.P. *Solving the p -median problem with a semi-Lagrangian relaxation*, *Comput. Optim. Appl.* 35(2) (2006) 239–260.
6. Benders, J.F. *Partitioning procedures for solving mixed-variables programming problems*, *Comput. Manag. Sci.* 2(1) (2005) 3–19.
7. Bisschop, J. *AIMMS optimization modeling*, Lulu. com. 2006.
8. Ceselli, A. and Righini, G. *A branch-and-price algorithm for the capacitated p -median problem*, *Networks.* 45(3) (2005) 125–142.
9. Cover, T.M. *Elements of information theory*, John Wiley & Sons, 1999.
10. Diaz, J.A. and Fernandez, E. *Hybrid scatter search and path relinking for the capacitated p -median problem*, *Eur. J. Oper. Res.* 169(2) (2006) 570–785.
11. Dolan, E.D. and More, J.J. *Benchmarking optimization software with performance profiles*, *Math. Program.* 91(2) (2002) 201–213.
12. Fleszar, K. and Hindi, K.S. *An effective VNS for the capacitated p -median problem*, *Eur. J. Oper. Res.* 191(3) (2008) 612–622.
13. Floudas, C.A., Aggarwal, A. and Ciric, A.R. *Global optimum search for nonconvex NLP and MINLP problems*, *Comput. Chem. Eng.* 13(10) (1989) 11171–1132.
14. Garey, M.R. and Johnson, D.S., *Computers and intractability: A guide to the theory of NP-completeness*, First Edition, W. H. Freeman, 1979.
15. Geoffrion, A.M. *Duality in nonlinear programming: A simplified applications-oriented development*, *SIAM Rev.* 13(1) (1971) 1–37.

16. Geoffrion, A.M. *Generalized benders decomposition*, J. Optim. Theory App. 10(4) (1972) 237–260.
17. Goldengorin, B., Krushinsky, D. and Pardalos, P.M. *Cell formation in industrial engineering*, New York, NY: Springer; 2013.
18. Hakimi, S.L. *Optimum distribution of switching centers in a communication network and some related graph theoretic problems*, Oper. Res. 13(3) (1965) 462–475.
19. Jaynes, E.T. *Information theory and statistical mechanics*, Phys. Rev. 106(4) (1957) 620.
20. Kariv, O. and Hakimi, S.L. *An algorithmic approach to network location problems II: the p -medians*, SIAM J. Appl. Math. 37 (1979) 539–560.
21. Lorena, L.A. and Senne, E.L. *A column generation approach to capacitated p -median problems*, Comput. Oper. Res. 31(6) (2004) 863–876.
22. McFadden, D.L. *Conditional logit analysis of qualitative choice behaviour*, Zarembka, P. (ed.): Frontiers in Econometrics, Wiley, New York, 1973, 105–142.
23. Maniezzo, V., Mingozzi, A. and Baldacci, R. *A bionomic approach to the capacitated p -median problem*, J. Heuristics. 4(3) (1998) 263–280.
24. Neebe, A.W. *Branch and Bound Algorithm for the p -Median Transportation Problem*, J. Oper. Res. Soc. 29(10) (1978) 989–995.
25. Reese, J. *Solution methods for the p -median problem: An annotated bibliography*, Networks. 48(3) (2006) 125–142.
26. ReVelle, C.S. and Swain, R.W. *Central facilities location*, Geogr. Anal. 2(1) (1970) 30–42.
27. Shamsipoor, H., Sandidzadeh, M.A. and Yaghini, M. *Solving capacitated p -median problem by a new structure of neural network*, Int. J. Ind. Eng-Theory. 19(8) (2012) 305–319.
28. Snickars, F. and Weibull, J.W. *A minimum information principle: Theory and practice*, Reg. Sci. Urban Econ. 7(1-2) (1977) 137–168.
29. Sule, D.R. *Manufacturing facilities: Location, planning, and design*, CRC press; 2008.
30. Tamir, A. *An $O(pn^2)$ algorithm for the p -median and related problems on tree graphs*, Oper. Res. Lett. 19(2) (1996) 59–64.
31. Teye, C., Bell, M.G. and Bliemer, M.C. *Locating urban and regional container terminals in a competitive environment: An entropy maximising approach*, Transp. Res. Proc. 23 (2017) 208–227.

32. Teye, C., Bell, M.G. and Bliemer, M.C. *Entropy maximising facility location model for port city intermodal terminals*, Transp. Res. E Logist. Transp. Rev. 100 (2017) 1–6.
33. Teye, C., Bell, M.G. and Bliemer, M.C. *Urban intermodal terminals: The entropy maximising facility location problem*, Transport. Res. B-Meth. 100 (2017) 64–81.
34. Van Zuylen, H.J. and Willumsen, L.G. *The most likely trip matrix estimated from traffic counts*, Transport. Res. B-Meth. 14(3) (1980) 281–293.



Numerical solution of the Allen–Cahn equation by using ”shifted” surface spline radial basis functions

C.G. Keshavarzi and F. Ghoreishi*

Abstract

We consider a fully-discrete approximation of the Allen–Cahn equation, such that the forward Euler/Crank–Nicolson scheme (in time) combined with the RBF collocation method based on “shifted” surface spline (in space). Numerical solvability and stability of the method, by using second order finite difference matrices are discussed. We show that, in the proposed scheme, the nonlinear term can be treated explicitly and the resultant numerical scheme is linear and easy to implement. Numerical results that show the efficiency and reliability of the proposed method are presented, and two types of collocation nodes for solving this equation are compared.

AMS(2010): Primary 65N35; Secondary 65L07.

Keywords: Allen–Cahn equation; RBF collocation method; Shifted surface spline; Stability; Solvability.

1 Introduction

The radial basis function method is one of the most useful tools for interpolating multidimensional scattered data. The method has remarkable properties, to handle arbitrarily scattered data, to adjust to several space dimensions easily, and to provide powerful convergence properties, which have made it popular in sciences and mathematics; see [16, 26, 6, 28]. Some of the more

*Corresponding author

Received 6 April 2020; revised 16 May 2020; accepted 20 August 2020

Changiz Goli Keshavarzi

Department of Mathematics, K. N. Toosi University of Technology, Tehran, Iran. e-mail: ckeshavarzi@mail.kntu.ac.ir.

Farideh Ghoreishi

Department of Mathematics, K. N. Toosi University of Technology, Tehran, Iran. e-mail: ghoreishif@kntu.ac.ir

recent applications of radial basis functions include measurements of potential or temperature on the earth's surface [3], mathematical finance [11], data mining [19], neural network [18], tomographic reconstruction, medical imaging [4], and the numerical solution of partial differential equations (PDEs).

The Allen–Cahn equation was initially introduced by Allen and Cahn [1] in 1979 for antiphase domain coarsening in a binary alloy. This equation, as the simplest model of phase-field, describes the evolution of diffuse phase boundary, concentrated in a small region. It is known that the Allen–Cahn equation is well-posed and has a unique solution that satisfies the maximum principle; see [12]. The Allen–Cahn equation has been widely used in many complicated moving interface problems in fluid dynamics and material science applications, such as crystal growth [7] and tumor growth [27] on flat surfaces.

During the past two decades, a great deal of mathematical efforts have been devoted to the study of numerical solution for the Allen–Cahn equation [20, 31, 12], and finite difference [13], finite element [9], Fourier spectral with periodic boundary condition [15], and RBF methods [17] are some of the recently proposed methods.

In this paper, a fully-discrete approximation of the Allen–Cahn equation is considered, such that a first order (Euler/Crank–Nicolson) finite difference scheme (in time) combined with the RBF collocation method (in space). Here, the main goal is how to construct an efficient numerical scheme based on the RBF collocation method that preserves the maximum principle and energy dissipation law in the Allen–Cahn equation. We will use a special kind of RBFs as the shifted surface splines [8, 29, 28, 30]. These functions include all of the multiquadric, thin plate, and power RBFs, which their properties quite well understood, theoretically and practically [3, 8, 26]. For example, some features of multiquadric, thin plate, and power RBFs are as follows:

- Multiquadrics are strictly positive definite, but surface splines and powers are conditionally positive definite.
- Powers and surface splines give an algebraic rate of convergence, but multiquadrics can achieve spectral rate of convergence.
- Multiquadrics interpolation is always solvable, provided that the points are disjoint, but for surface splines and powers, for solvability, we need to add some appropriate number of functions (polynomials). It is worth noting the shifted surface spline $(-1)^{m-s/2+1}(x^2 + \delta^2)^{m-s/2} \log(x^2 + \delta^2)^{1/2}$, where s is even and $\delta \neq 0$, is strictly conditionally positive definite (see [3, p. 104]). Hence, for the well-posedness of the interpolation problem, we need to add some appropriate polynomial to this function.
- Powers and surface splines are piecewise smooth, but multiquadrics are infinitely smooth.
- Franke's research [10] showed that multiquadrics, in terms of simplicity of implementation, stability, and accuracy, have much better performance among 29 available interpolation methods used for interpolating two-dimensional analytic functions.

In the proposed scheme, the nonlinear term can be treated explicitly such that, the resultant numerical scheme will be linear and easy to implement. The numerical solvability and stability of the method by using the RBF-FD procedure based on the classical finite difference method are discussed. The RBF-FD method uses the simplicity of FDM formulation and the meshfree property of RBFs, which was proposed in [22, 21]. Also, Tolstykh, Lipavskii, and Shirobokov [24] and Wright and Fornberg [25] introduced this method independently for the numerical approximation of the ODEs and PDEs. For flat RBF, Wright and Fornberg [25] showed that the RBF-FD scheme is the same as the conventional finite difference scheme. The RBF-FD method is used with other names such as local radial basis function method [23] and local radial basis functions based on the differential quadrature method. By using some numerical experiments, we show that the proposed method with equally spaced collocation nodes may give better results in comparison with scattered nodes.

The paper is organized as follows. In Section 2, the RBF interpolation is introduced in brief. Section 3 reviews the RBF collocation methods. In Section 4, numerical results that show the efficiency of the proposed method are given. The last section ends with concluding remarks.

2 The RBF interpolation method

Before we pose the RBF interpolation, we need to recall the following definitions.

Definition 1. A continuous and even function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is said to be strictly conditionally positive definite of order k (SCPD(k)) if, for all $N \in \mathbb{N}$, all sets $\chi = \{X_0, \dots, X_N\} \subset \mathbb{R}^d$, and all $\lambda \in \mathbb{R}^N$ satisfying

$$\sum_{i=0}^N \lambda_i q(X_i) = 0,$$

for all real-valued polynomials q of degree less than k , the quadratic form

$$\lambda^t A \lambda = \sum_{i=0}^N \sum_{j=0}^N \lambda_i \lambda_j f(\|X_i - X_j\|),$$

is positive, unless λ is zero.

Let $\Pi_k(\mathbb{R}^d)$ be the space of d -variate polynomials of degree at most k ; then we have the following definition.

Definition 2. A set of distinct points $\{X_0, \dots, X_N\}$ in $\mathbb{R}^d$ is said to be

$\Pi_k(\mathbb{R}^d)$ -unisolvent if the only element of $\Pi_k(\mathbb{R}^d)$ that vanishes at each x_i is the zero polynomial.

The definition of multivariate RBF interpolation can be stated as follows:

Definition 3. Given a set $\chi = \{X_i\}_{i=0}^N$ of distinct centers in $\mathbb{R}^d$ and a target function $v : \mathbb{R}^d \rightarrow \mathbb{R}$, the RBF interpolant is given by

$$Sv(X) = \sum_{j=0}^N \lambda_j \phi(\|X - X_j\|) + \sum_{k=0}^M \gamma_k p_k(X), \quad M \leq N, \quad (1)$$

where $\phi(\|X - X_j\|)$ is a strictly conditionally positive definite RBF of order m , centered on the points x_j , and $\{p_k(X)\}_{k=0}^M$ forms a basis for the polynomial space $\Pi_{m-1}(\mathbb{R}^d)$, where $M = \binom{d+m-1}{m-1}$. The expansion coefficient λ_j , γ_k are determined by the interpolation conditions,

$$Sv(X_j) = v(X_j), \quad 0 \leq j \leq N,$$

and the following additional conditions are considered to handle the additional degree of freedom

$$\sum_{j=0}^N \lambda_j p_k(X_j) = 0, \quad 0 \leq k \leq M.$$

These conditions lead to a system of equations that can be written down in the matrix form as

$$\begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\gamma} \end{pmatrix} = \begin{pmatrix} V \\ 0 \end{pmatrix}, \quad (2)$$

where

$$\Phi_{ij} = \phi(\|X_i - X_j\|), \quad P_{ij} = p_j(X_i), \quad \boldsymbol{\lambda} = (\lambda_0, \dots, \lambda_N)^t, \\ \boldsymbol{\gamma} = (\gamma_0, \dots, \gamma_M)^t, \quad V = (v_0, \dots, v_N)^t.$$

As it is shown in [26] that system (2) is uniquely solvable when ϕ is strictly conditionally positive definite of order m and χ is $\Pi_{m-1}(\mathbb{R}^d)$ -unisolvent set of centers; so the coefficient matrix in (2) is invertible.

In this paper, we use “shifted” surface splines as radial basis functions ϕ , which include the multiquadric and thin plate spline and are defined as follows:

$$\phi_\delta(x) := \begin{cases} (-1)^{\lceil m-s/2 \rceil} (x^2 + \delta^2)^{m-s/2}, & s \text{ is odd,} \\ (-1)^{m-s/2+1} (x^2 + \delta^2)^{m-s/2} \log(x^2 + \delta^2)^{1/2}, & s \text{ is even,} \end{cases}$$

where $s, m \in N$, $m > s/2$ and $\lceil s \rceil$ indicates the smallest integer greater than s . For odd s , ϕ_δ is called the multiquadric RBF, and for $\delta = 0$, the function ϕ_δ is called the surface spline.

In what follows, we consider the number h as the "density" of $\chi = \{X_0, \dots, X_N\}$ in Ω , as follows:

$$h = h(\Omega; \chi) := \sup_{X \in \Omega} \min_{X_k \in \chi} |X - X_k|.$$

3 The RBF collocation method, stability and numerical solvability

In this section, we use the forward Euler/Crank–Nicolson scheme for the time and classical RBF method for spatial discretization in numerical approximation of the Allen–Cahn equation. Let us define $t_k = k * \tau, k = 0, 1, \dots, \mathcal{M}$, where $\tau = \frac{T}{\mathcal{M}}$ and $\mathcal{M}$ is the number of the time steps. we introduce notations $v_j(X) = v(X, t_j)$ and $\partial_t v_j = \frac{v_{j+1} - v_j}{\tau}$. Also we define $\hat{v}_j = \frac{v_{j+1} + v_j}{2}$.

Consider the following time-dependent Allen–Cahn equation:

$$\begin{cases} \frac{\partial v(X, t)}{\partial t} - \Delta v(X, t) + \frac{1}{\eta^2} f(v(X, t)) = 0, & X \in \Omega \subset \mathbb{R}^d, t \in [0, T] \\ v(X, t) = g(X), & X \in \partial\Omega, \end{cases} \quad (3)$$

which is supplemented with the following initial condition:

$$v(X, 0) = v_0(X) \text{ in } \Omega,$$

where Ω is a bounded convex domain with Lipschitz boundary $\partial\Omega$, $T > 0$ is a positive constant, and $f(v) = v^3 - v$ is the reaction term with $F(v) = \frac{1}{4}(v^2 - 1)^2$, a given energy potential that drives the solution into the two pure states $v = \pm 1$. The nonlinearity $f(v)$ is monotone outside the interval $[-1, 1]$, so solutions of the Allen–Cahn equation satisfy a maximum principle; see [12, 2]. Also, $g(X)$ and $X \in \partial\Omega$ are known constants. The parameter η , which usually stands for the inter-facial width, is a small constant compared to the characteristic length of the laboratory scale. The function $v(X, t)$ is a distribution function of the concentration for one of the two metallic components of the alloy.

In the RBF collocation method, the solution $v(X, t)$ is approximated by a linear combination of RBFs as

$$v_N(X, t) = \sum_{j=0}^N \lambda_j(t) \phi(\|X - X_j\|) + \sum_{k=0}^M \gamma_k(t) p_k(X), \quad M \leq N, \quad (4)$$

where ϕ and $p_k(X)$ are defined in (1). The collocation points $\{X_i\}_{i=0}^N$ has been selected in such a way that $\{X_0, \dots, X_{N_I}\} \subset \Omega$ and $\{X_{N_I+1}, \dots, X_N\} \subset \partial\Omega$. In RBF collocation method, the requirement is that (3) is satisfied exactly by (4) at the following relations.

$$\begin{aligned} \frac{\partial v_N(X_k, t)}{\partial t} - \Delta v_N(X_k, t) + \frac{1}{\eta^2} f(v_N(X_k, t)) &= 0, \quad k = 0, 1, \dots, N_I, \\ v_N(X_k, t) &= g(X_k), \quad k = N_I + 1, \dots, N, \end{aligned}$$

and the initial conditions are

$$v_N(X_k, 0) = v_0, \quad k = 0, 1, \dots, N.$$

According to the forward Euler/Crank–Nicolson scheme in time, we discretize the Allen–Cahn equation as follows:

$$\begin{aligned} \frac{v_N(X_k, t_{j+1}) - v_N(X_k, t_j)}{\tau} - \frac{\Delta v_N(X_k, t_{j+1}) + \Delta v_N(X_k, t_j)}{2} \\ + \frac{1}{\eta^2} f(v_N(X_k, t_j)) &= 0, \quad k = 0, 1, \dots, N, \end{aligned}$$

where τ is a mesh size in time and $X_k = (x_i, y_l)$, $i, l = 0, \dots, n$, with $N = (n+1)^2 - 1$. Equivalently,

$$\begin{aligned} (v_N)_k^{j+1} - \frac{\tau}{2}(\Delta v_N)_k^{j+1} &= (v_N)_k^j + \frac{\tau}{2}(\Delta v_N)_k^j \\ &\quad - \frac{\tau}{\eta^2} f((v_N)_k^j) = 0, \quad k = 0, 1, \dots, N \end{aligned} \quad (5)$$

where

$$(v_N)_k^j = v_N(X_k, t_j), \quad (\Delta v_N)_k^j = \Delta v_N(X_k, t_j) = \left((v_N)_{xx} + (v_N)_{yy} \right)_k^j.$$

In addition, we impose the following constraints,

$$\sum_{j=0}^N \lambda_j(t) p_i(X_j) = 0, \quad 0 \leq i \leq M. \quad (6)$$

From (5) and (6), we have a system with $N + M + 2$ unknowns $\{\lambda_k\}_{k=0}^N$ and $\{\gamma_k\}_{k=0}^M$ and $N + M + 2$ equations. But, before imposing boundary conditions, we will simplify systems (5) and (6), by using finite difference matrices.

Now, If we set $\Phi_{ik} = \phi(\|X_i - X_k\|)$ and $\Delta \Phi_{ik} = \Delta \phi(\|X_i - X_k\|)$, then we can rewrite collocation equation (5) and (6) in the matrix notation as

$$\begin{aligned} \left[\begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix} - \frac{\tau}{2} \begin{pmatrix} \Delta \Phi & \Delta P \\ P^t & 0 \end{pmatrix} \right] \Lambda^{j+1} &= \frac{\tau}{\eta^2} \begin{pmatrix} -f((v_N)_k^j) \\ 0 \end{pmatrix} \\ &\quad + \begin{pmatrix} \frac{\tau}{2} \Delta (v_N)_k^j + (v_N)_k^j \\ 0 \end{pmatrix}, \end{aligned} \quad (7)$$

where $\Lambda^{j+1} = \begin{pmatrix} \lambda^{j+1} \\ \gamma^{j+1} \end{pmatrix}$. Now, by using (2), we may write (7) as

$$\left[\begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix} - \frac{\tau}{2} \begin{pmatrix} \Delta \Phi & \Delta P \\ P^t & 0 \end{pmatrix} \right] A^{-1} \begin{pmatrix} V^{j+1} \\ 0 \end{pmatrix} = -\frac{\tau}{\eta^2} \begin{pmatrix} f((v_N)_k^j) \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\tau}{2} \Delta (v_N)_k^j + (v_N)_k^j \\ 0 \end{pmatrix}, \quad (8)$$

where $V^l = [(v_N)_0^l, (v_N)_1^l, \dots, (v_N)_N^l]^t, l = 0, \dots, \mathcal{M}$ and

$$A = \begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix},$$

which is a dense linear system and when solved, we obtain an RBF based on the numerical approximation for (3). However, for a stable computation, we have to consider, stability issues for time-dependent problems and adapt the proposed method for nonlinear equations.

To obtain stability criterion for system (8), we will use a recently proposed local method that causes more flexibility for handling nonlinearities. In this method, essentially a classical finite difference (FD) method or generalization of them to scattered node layouts are considered, which is known as RBF-FD methods. This kind of methods have been investigated independently by Shu et al. [22, 21], Tolstykh, Lipavskii, and Shirobokov [24], Cecil, Qian, and Osher [5], Wright and Fornberg [25]. Here, we only consider its most general form, in which, the FD method consists of approximating second order derivative of base functions at a given point, based on a linear combination of the value of radial basis functions at some surrounding nodes. In other words, we will use a classical 2-dimensional case as follows:

$$\frac{d^2}{dx^2} \phi_s(\|X\|) |_{x=x_j} \simeq \sum_{i=0}^k w_{j,i}^{(2)} \phi_s(\|X_i\|), \quad 0 \leq j \leq n, \quad 1 \leq k \leq n,$$

where $w_{i,j}^{(2)}$ is called the FD weights and $\phi_s(\|X\|) = \phi(\|X - X_s\|)$. This formula can be combined to create FD formulas for partial derivatives in two and higher dimensions. In classical central finite difference approximation of order two, FD weights can be considered as:

$$\begin{aligned} \frac{d^2}{dx^2} \phi_s(\|X\|) |_{x=x_j} &\simeq \sum_{i=0}^k w_{j,i}^{(2)} \phi(\|X_i\|) \\ &= \frac{\phi(\|X_{j+1}\|) - 2\phi(\|X_j\|) + \phi(\|X_{j-1}\|)}{h^2}, \end{aligned}$$

where h is the spatial distance of nodes. In two-dimensional case, we have

$$\begin{aligned} \frac{d^2}{dx^2} \phi_s(\|X\|) |_{x=x_j} + \frac{d^2}{dy^2} \phi_s(\|X\|) |_{y=y_i} &\simeq \frac{\phi_s(\|X_{j+1,i}\|) - 2\phi_s(\|X_{j,i}\|) + \phi_s(\|X_{j-1,i}\|)}{h^2} \\ &\quad + \frac{\phi_s(\|X_{j,i-1}\|) - 2\phi_s(\|X_{j,i}\|) + \phi_s(\|X_{j,i+1}\|)}{h^2}. \end{aligned}$$

In the matrix notation, we may write $\Delta\Phi = \frac{1}{h^2}F^{(2)}\Phi$, where the finite difference matrix $F^{(2)}$ is as follows:

$$F^{(2)} = \begin{pmatrix} -4 & 1 & 0 & \dots & 0 & 1 & 0 & 0 & \dots \\ 1 & 0 & 0 & & 1 & -4 & 1 & 0 & \\ 0 & 0 & 0 & \dots & 0 & 1 & 0 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots \\ 1 & -4 & 1 & & 0 & 0 & 1 & 0 & \\ 0 & 1 & 0 & \dots & 0 & 1 & -4 & 1 & \dots \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \vdots & \ddots & \end{pmatrix}.$$

From $\Delta\Phi = \frac{1}{h^2}F^{(2)}\Phi$ and $\Delta P = \frac{1}{h^2}F^{(2)}P$, collocation equation (8) can be written as

$$\left[\begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix} - \frac{\tau}{2h^2} \begin{pmatrix} F^{(2)} & 0 \\ 0 & I \end{pmatrix} \cdot \begin{pmatrix} \Phi & P \\ P^t & 0 \end{pmatrix} \right] A^{-1} \begin{pmatrix} V^{j+1} \\ 0 \end{pmatrix} = R, \quad (9)$$

where R is the right-hand side of (8). For example, for $n = 2$, ($N = (n+1)^2$), we have

$$F^{(2)} = \begin{pmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{pmatrix}.$$

Now, (9) can be written equivalently as

$$\left[I_{N+1} - \frac{\tau}{2h^2} F^{(2)} \right] V^{j+1} = \left[-\frac{\tau}{\eta^2} f((v_N)_k^j) + \frac{\tau}{2} \Delta(v_N)_k^j + (v_N)_k^j \right]_{k=0}^N. \quad (10)$$

It is well known that the eigenvalues of $F_{xx}^{(2)} \left(F_{yy}^{(2)} \right)$ are $\lambda_i = -4 \sin^2(\frac{i\pi}{2(N+1)})$, or $\rho(F^{(2)}) \leq 8$, that is, $|1 - \frac{4\tau}{h^2}| \leq 1$. Hence for a stable computation, we need $\frac{2\tau}{h^2} \leq 1$.

Note that the eigenvalues of $F^{(2)}$ after imposing boundary conditions are less than or equal to zero. Therefore, $1 - \frac{\tau\lambda_i}{2h^2} \neq 0$ for all $0 \leq i \leq N$. Thus, clearly system (10) is numerically solvable.

According to considered order of mesh points $\{x_i, y_j\}$,

$$\{\{x_0, y_0\}, \{x_0, y_1\}, \dots, \{x_0, y_n\}, \{x_1, y_0\}, \{x_1, y_1\}, \dots, \{x_1, y_n\}, \dots, \{x_n, y_n\}\}$$

and given boundary conditions on x_0, x_n, y_0, y_n , system (8) after imposing boundary conditions is completed. This is a system with $N + 1$ unknowns and equations. Therefore, after imposing given boundary conditions, (8) can be solved directly, by using stability condition $\frac{2\tau}{h^2} \leq 1$.

4 Numerical illustrations

In this section, we present four numerical examples to illustrate the accuracy and efficiency of the proposed scheme. According to our observations, when equally spaced nodes are used as a set of collocation points, we obtain better results in comparison with scattered mesh points. In all of test problems, we have used multiquadratic radial basis functions with $\tau \leq \eta^2$ and $\tau \leq h^2/2$. In Example 1, the standard surface spline is considered too. For comparison, at time $t = T$, the infinity norm of the error

$$\|v - Sv\|_\infty = \max_{1 \leq i \leq N} |v(X_i, T) - Sv(X_i, T)|,$$

and the root mean square (RMS) norm of the error

$$\|v - Sv\|_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N (v(X_i, T) - Sv(X_i, T))^2},$$

are used, where N is the number of collocation points.

Example 1. Consider two-dimensional Allen–Cahn equation with the following initial and boundary conditions:

$$\begin{cases} \frac{\partial v}{\partial t} - \Delta v + \frac{1}{\eta^2}(v^3 - v) = f, & (x, y) \in (-1, 1)^2, t > 0, \\ v(\pm 1, y, t) = 0, v(x, \pm 1, t) = 0, & t > 0, \\ v(x, y, 0) = 0, & (x, y) \in [-1, 1]^2, \end{cases}$$

where $\eta = 100$, the exact solution is $v(x, y, t) = \tanh((x^2 - 1)(y^2 - 1)t)$ and the function f is computed accordingly.

In this example, we apply the proposed method by using two types of nodes. As shown in Table 1, the obtained errors for equally spaced nodes are slightly better than scattered nodes. We plot the absolute error of this example in Figure 1 and the contour plot of the error in Figure 2. The comparison of errors of Example 1 by using different time steps for $t = 1000\tau$ at $N = 100$ is presented in Table 2.

Table 1: The comparison of errors of Example 1, by considering “shifted” surface spline, at $t = 1000\tau$ by using scattered and equally spaced nodes where time step is $\tau = 10^{-4}$.

N	Equally Spaced		Scattered	
	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
49	$4.82043E - 6$	$1.41187E - 5$	$4.26723E - 6$	$1.30836E - 5$
64	$4.07495E - 6$	$1.33603E - 5$	$3.35978E - 6$	$1.34205E - 5$
81	$1.31830E - 6$	$5.20152E - 6$	$7.72547E - 7$	$3.28669E - 6$
100	$1.12669E - 6$	$5.04211E - 6$	$7.65370E - 7$	$2.91935E - 6$
121	$7.02509E - 7$	$3.32090E - 6$	$7.56430E - 7$	$1.94812E - 6$
144	$6.73415E - 7$	$3.52046E - 6$	$6.22881E - 7$	$2.15101E - 6$

Table 2: The comparison of errors of Example 1, by using various time steps at $N = 100$ and $t = 1000\tau$.

Time Step ($\tau = 10^{-4}$)	Equally Spaced		Scattered	
	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
τ	$1.12669E - 6$	$5.04211E - 6$	$7.65370E - 7$	$2.91935E - 6$
$1/2 \times \tau$	$2.78156E - 7$	$1.17651E - 6$	$1.33866E - 7$	$6.79208E - 7$
$1/4 \times \tau$	$6.91386E - 8$	$2.85469E - 7$	$3.36881E - 8$	$1.65689E - 7$
$1/8 \times \tau$	$1.72370E - 8$	$7.04178E - 8$	$8.64618E - 9$	$4.10731E - 8$
$1/16 \times \tau$	$4.30344E - 9$	$1.74946E - 8$	$2.19114E - 9$	$1.02357E - 8$

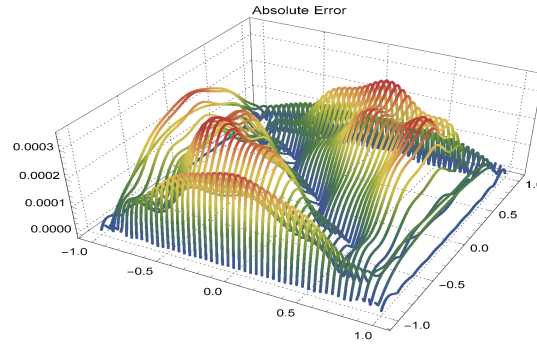


Figure 1: The plot of the absolute error in Example 1, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

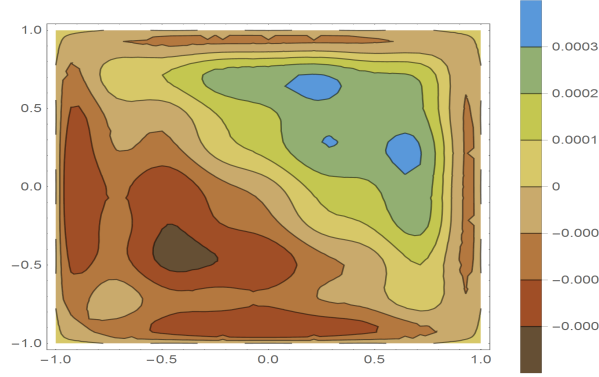


Figure 2: The contour plot of the error in Example 1, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

In what follows, we solve Example 1, by considering the standard surface spline $\phi(r) = r^2 \log r$. Comparing the results obtained in Tables 1–4, it can be seen that the “shifted” surface spline provides better results than the standard surface spline. The absolute error and contour plot of the error, for this case, is plotted in Figures 1 and 4, respectively.

Table 3: The comparison of errors of Example 1, by considering standard surface spline, at $t = 1000\tau$ by using scattered and equally spaced nodes where time step is $\tau = 10^{-4}$.

Equally Spaced			Scattered	
N	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
49	$6.51165E - 6$	$1.84565E - 5$	$5.57870E - 6$	$1.78168E - 5$
64	$5.59294E - 6$	$1.74854E - 5$	$4.96732E - 6$	$1.79523E - 5$
81	$1.84662E - 6$	$6.64194E - 6$	$1.13404E - 6$	$4.34977E - 6$
100	$1.58833E - 6$	$6.32184E - 6$	$8.10437E - 7$	$3.54860E - 6$
121	$7.04900E - 7$	$3.59848E - 6$	$7.72250E - 7$	$1.96949E - 6$
144	$6.74292E - 7$	$3.73803E - 6$	$6.20674E - 7$	$2.14926E - 6$

Table 4: The comparison of errors of Example 1, by considering standard surface spline, by using various time steps at $N = 100$ and $t = 1000\tau$.

Equally Spaced			Scattered	
Time Step ($\tau = 10^{-4}$)	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
τ	$1.58833E - 6$	$6.32184E - 6$	$8.10437E - 7$	$3.54860E - 6$
$1/2 \times \tau$	$3.91406E - 7$	$1.50928E - 6$	$1.85967E - 7$	$8.74747E - 7$
$1/4 \times \tau$	$9.71936E - 8$	$3.69865E - 7$	$5.01409E - 8$	$2.20779E - 7$
$1/8 \times \tau$	$2.42193E - 8$	$9.16348E - 8$	$1.30555E - 8$	$5.57276E - 7$
$1/16 \times \tau$	$6.04515E - 9$	$2.28113E - 8$	$3.33334E - 9$	$1.40169E - 8$

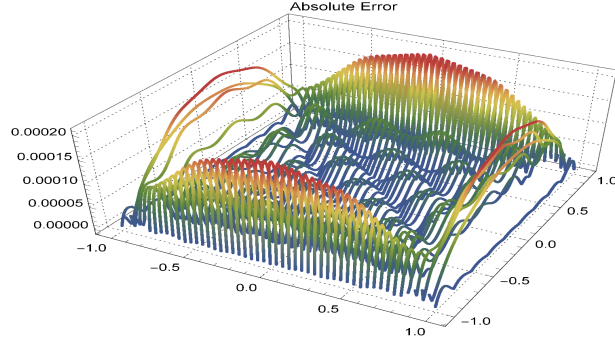


Figure 3: The plot of the absolute error in Example 1, by considering the standard surface spline, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

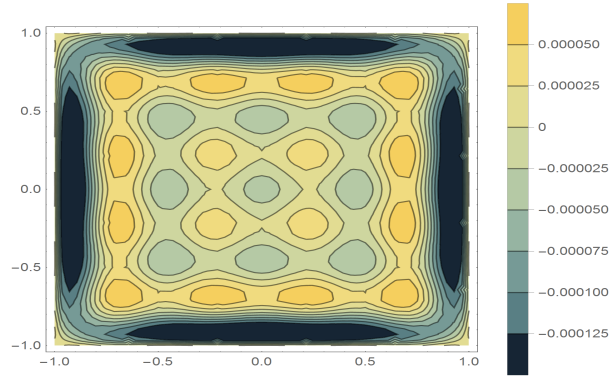


Figure 4: The contour plot of the error in Example 1, by considering the standard surface spline at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

Example 2. Consider the Allen–Cahn equation with the following initial and boundary conditions

$$\begin{cases} \frac{\partial v}{\partial t} - \Delta v + \frac{1}{\eta^2}(v^3 - v) = f, & (x, y) \in (-1, 1)^2, t > 0, \\ v(\pm 1, y, t) = e^{-\pi} \sin((y^2 - 1)t), & v(x, \pm 1, t) = 0, t > 0, \\ v(x, y, 0) = 0, & (x, y) \in [-1, 1]^2, \end{cases}$$

where $\eta = 100$, the exact solution is $v(x, y, t) = e^{-\pi x^2} \sin((y^2 - 1)t)$ and the right-hand side function f can be computed accordingly.

The comparison of errors for two types of nodes is presented in Table 5, and the plot of the absolute error is given in Figure 5. We also plot the

contour of the error in Figure 6. The obtained errors for this example, by using different time steps, are also presented in Table 6.

Table 5: The comparison of errors of Example 2 for $t = 1000\tau$ by using scattered and equally spaced nodes where time step is $\tau = 10^{-4}$.

Equally Spaced			Scattered	
N	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
49	$1.07720E - 5$	$3.00790E - 5$	$1.31170E - 5$	$3.15498E - 5$
64	$8.74748E - 6$	$2.65163E - 5$	$1.14132E - 5$	$3.47482E - 5$
81	$1.67367E - 6$	$4.61698E - 6$	$1.55723E - 6$	$4.72437E - 6$
100	$1.33128E - 6$	$4.66584E - 6$	$1.43747E - 6$	$5.21502E - 6$
121	$6.75815E - 7$	$2.56911E - 6$	$7.37549E - 7$	$1.53253E - 6$
144	$6.28635E - 7$	$2.73066E - 6$	$5.62016E - 7$	$1.69418E - 6$

Table 6: The comparison of errors of Example 2 by using various time steps at $N = 100$ and $t = 1000\tau$.

Equally Spaced			Scattered	
Time Step ($\tau = 10^{-4}$)	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
τ	$1.33128E - 6$	$4.66584E - 6$	$1.43747E - 6$	$5.21502E - 6$
$1/2 \times \tau$	$3.29225E - 7$	$1.12282E - 6$	$3.27643E - 7$	$1.21081E - 6$
$1/4 \times \tau$	$8.18399E - 8$	$2.75750E - 7$	$7.82939E - 8$	$2.92148E - 7$
$1/8 \times \tau$	$2.04010E - 8$	$6.83517E - 8$	$1.91410E - 8$	$7.17766E - 8$
$1/16 \times \tau$	$5.09283E - 9$	$1.70169E - 8$	$4.73236E - 9$	$1.77901E - 8$

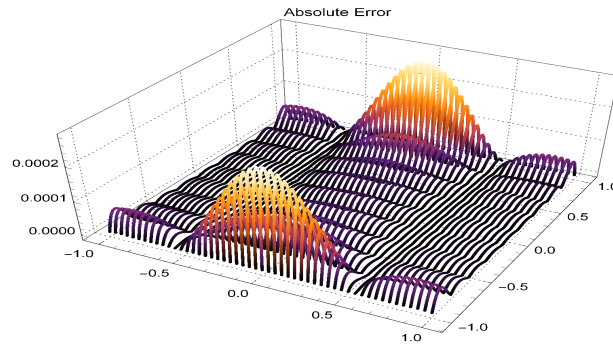


Figure 5: The plot of the error in Example 2, at $t = 1000\tau$ and $N = 100$ for equally spaced nodes.

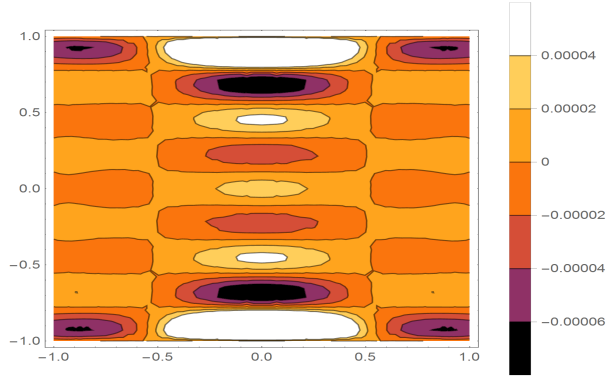


Figure 6: The contour plot of the error in Example 2, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

Example 3. For this example, consider the following equation:

$$\begin{cases} \frac{\partial v}{\partial t} - \Delta v + \frac{1}{\eta^2}(v^3 - v) = f, & (x, y) \in (-1, 1)^2, t > 0, \\ v(\pm 1, y, t) = 0, v(x, \pm 1, t) = 0, & t > 0, \\ v(x, y, 0) = \sin(\pi y) \cos(\pi x/2), & (x, y) \in [-1, 1]^2, \end{cases} \quad (11)$$

where $\eta = 100$ and the exact solution is $v(x, y, t) = \sin(\pi y) \cos(\pi x/2)(1 + t)$ and the function f can be computed accordingly.

The comparison of errors for the equally spaced and the scattered nodes for this example is shown in Table 7. As we expected, better results are obtained for equally spaced nodes. The plot of absolute error, in the case of equally spaced nodes, is presented in Figure 7, and the contour plot of the error is given in Figure 8. In Table 8, the comparison of errors for two types of nodes, by using different time steps is shown.

Table 7: The comparison of errors of Example 3, at $t = 1000\tau$ by using scattered and equally spaced nodes where time step is $\tau = 10^{-4}$.

N	Equally Spaced		Scattered	
	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
49	$3.07732E - 3$	$6.27540E - 3$	$5.39392E - 3$	$7.59093E - 3$
64	$5.49423E - 4$	$1.12731E - 3$	$1.08340E - 3$	$1.38534E - 3$
81	$3.37897E - 4$	$8.97249E - 4$	$6.02756E - 4$	$1.14500E - 3$
100	$5.51337E - 5$	$1.40970E - 4$	$1.58731E - 4$	$2.24005E - 4$
121	$3.34659E - 5$	$1.06018E - 4$	$4.93030E - 5$	$1.29525E - 4$
144	$5.23636E - 6$	$1.69188E - 5$	$1.28415E - 5$	$2.64670E - 5$

Table 8: The comparison of errors of Example 3, by using various time steps at $N = 100$ and $t = 1000\tau$.

Time Step ($\tau = 10^{-4}$)	Equally Spaced		Scattered	
	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
τ	$5.51337E-5$	$1.40970E-4$	$1.58731E-4$	$2.24005E-4$
$1/2 \times \tau$	$2.72150E-5$	$6.72841E-5$	$6.89102E-5$	$1.04333E-4$
$1/4 \times \tau$	$1.34159E-6$	$3.27630E-5$	$3.20549E-5$	$5.04380E-5$
$1/8 \times \tau$	$6.65022E-6$	$1.61537E-5$	$1.54562E-5$	$2.48065E-5$
$1/16 \times \tau$	$3.30963E-6$	$8.01907E-6$	$7.58896E-6$	$1.23023E-5$

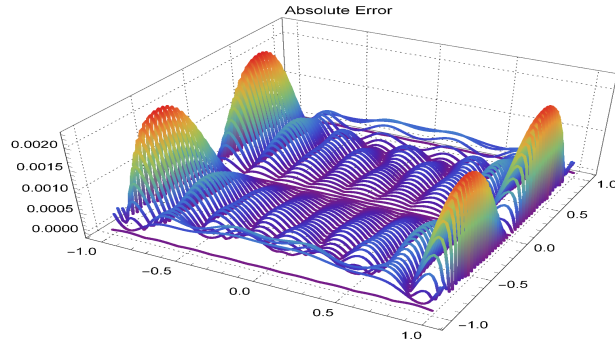


Figure 7: The plot of the error in Example 3, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

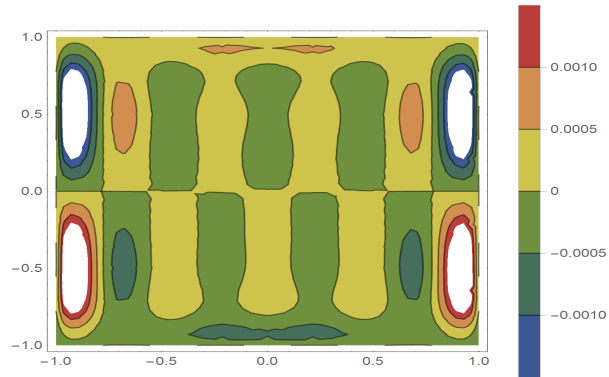


Figure 8: The contour plot of the error in Example 3, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

Example 4. For the final example, we solve the following equation in a polygonal domain.

$$\begin{cases} \frac{\partial v}{\partial t} - \Delta v + \frac{1}{\eta^2}(v^3 - v) = f, & (x, y) \in (-1, 1)^2, t > 0, \\ v(\pm 1, y, t) = 0, v(x, \pm 1, t) = 0, & t > 0, \\ v(x, y, 0) = (x^2 - 1)(y^2 - 1), & (x, y) \in [-1, 1]^2, \end{cases} \quad (12)$$

where $\eta = 100$, the exact solution is $v(x, y, t) = (x^2 - 1)(y^2 - 1)e^{-\pi^2 t}$ and the function f is computed accordingly.

In this example, for showing the flexibility of our method, we consider a polygonal domain with the following vertices that are arranged in clockwise order:

$$\{(-1, 1), (-1, 0), (-0.286, 0), (0.5, 1), (1, 1), (1, -1)\}.$$

Figure 9 shows the shape of the domain and equally spaced points. As before, the comparison of errors for the equally spaced and the scattered nodes is presented in Table 9. we also plot the absolute error in Figure 10 and the contour plot of the error in Figure 11.

Table 9: The comparison of errors of Example 4, by using various time steps at $N = 165$ and $t = 1000\tau$.

Time Step ($\tau = 10^{-4}$)	Equally Spaced		Scattered	
	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$	$\ u - u_N\ _\infty$	$\ u - u_N\ _2$
τ	$3.20101E-4$	$1.48746E-3$	$3.69452E-4$	$1.00703E-3$
$1/2 \times \tau$	$1.03216E-4$	$4.58850E-4$	$1.49190E-4$	$3.25760E-4$
$1/4 \times \tau$	$3.15830E-5$	$1.28779E-4$	$6.47536E-5$	$1.11007E-4$
$1/8 \times \tau$	$9.89283E-6$	$3.56560E-5$	$3.01340E-5$	$4.73288E-5$
$1/16 \times \tau$	$3.35178E-6$	$1.11966E-5$	$1.45773E-5$	$2.35022E-5$

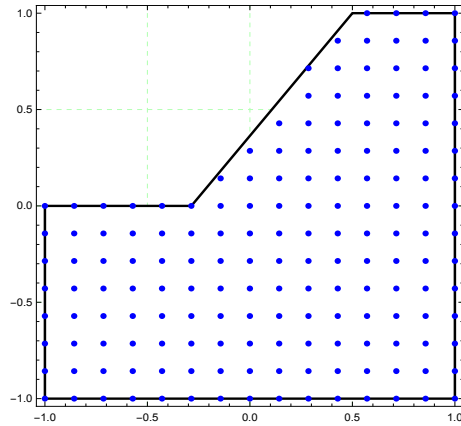


Figure 9: Polygonal domain including 165 equally spaced points for Example 4.

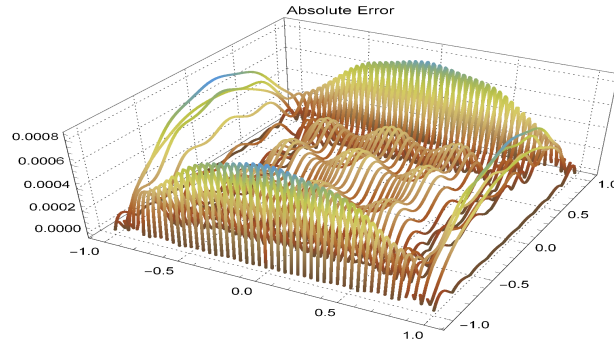


Figure 10: The plot of the error in Example 4, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

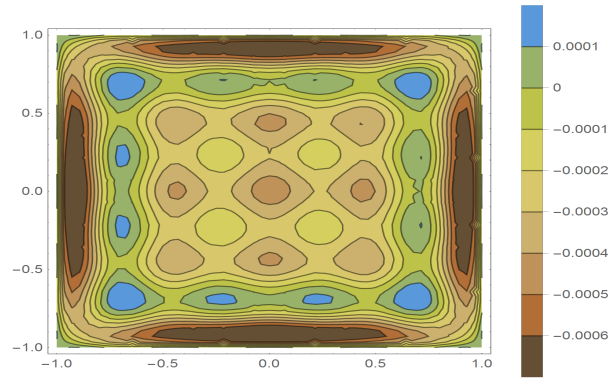


Figure 11: The contour plot of the error in Example 4, at $t = 1000\tau$ and $N = 100$ with equally spaced nodes.

5 Conclusion

In this paper, a fully-discrete approximation of the Allen–Cahn equation was considered. Based on the forward Euler/Crank–Nicolson scheme (in time) and the RBF collocation method (in space), we proposed a numerical scheme, in which the nonlinear term could be treated explicitly and the resultant numerical scheme was linear and easy to implement. Numerical solvability and stability of the method, by using second order finite difference matrices were discussed. Four numerical test problems were considered, which showed the efficiency and reliability of the proposed method. The obtained numerical

results showed that the equally spaced nodes are better in comparison with scattered nodes.

Acknowledgements

The authors would like to thank the anonymous referees for their valuable comments and suggestions, that greatly improved the final version of this article.

References

1. Allen, S.M. and Cahn, J.W. *A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening*, Acta Metall. 27 (1979) 1085–1095.
2. Bartels, S. *Numerical methods for nonlinear partial differential equations*, Springer Series in Computational Mathematics, vol. 47, Springer, 2015.
3. Buhmann, M.D. *Radial basis functions: Theory and implementations*, Cambridge University Press, Cambridge, 2004.
4. Carr, J.C., Fright, W.R. and Beatson, R.K. *Surface interpolation with radial basis functions for medical imaging*, IEEE Trans. Med. Imaging 16 (1997) 96–107.
5. Cecil, T., Qian, J. and Osher, S. *Numerical methods for high dimensional Hamilton-Jacobi equations using radial basis functions*, J. Comput. Phys. 196 (2004) 327–347.
6. Cheng, A.H.D., Golberg, M.A., Kansa, E.j. and Zangeni, G. *Exponential convergence and h-c multiquadric collocation method for partial differential equations*, Numer. Meth. Part. Differ. Equ. 19 (2003) 571–594.
7. Cheng, M. and Warren, J.A. *An efficient algorithm for solving the phase field crystal model*, J. Comput. Phys. 227 (2008) 6241–6248.
8. Fasshauer, G.E. *Meshfree approximation methods with MATLAB*, World Scientific, Hackensack, 2007.
9. Feng, X., Li, Y. and Zhang, Y. *Finite element methods for the stochastic Allen-Cahn equation with gradient-type multiplicative noise*, SIAM J. Numer. Anal. 55(1) (2017) 194–216.
10. Franke, R. *Scattered data interpolation: tests of some methods*, Math. Comput. 38 (1982) 181–200.

11. Hon, Y.C. and Mao, X.Z. *A radial basis function method for solving options pricing model*, Financial Eng. 8 (1) (1999) 31–50.
12. Kessler, D., Nochetto, R. and Schmidt, A. *A posteriori error control for the Allen–Cahn problem: circumventing Gronwall’s inequality*, M2AN Math. Model. Numer. Anal. 38 (2004) 129–142.
13. Kim, J., Jeong, D., Yang, S. and Choi, Y. *A finite difference method for a conservative Allen–Cahn equation on non-flat surfaces*, J. Comput. Phys. 334 (2017) 170–181.
14. Kufner, A. and Persson, L.E. *Weighted inequalities of Hardy type*, World Scientific Publishing Co., Inc., River Edge, NJ, 2003.
15. Lee, H. and Lee, J. *A semi-analytical Fourier spectral method for the Allen–Cahn equation*, Comput. Math. Appl. 68 (3) (2014) 174–184.
16. Madych, W.R. and Nelson, S.A. *Multivariable interpolation and conditionally positive definite functions II*, Math. Comput. 54 (1990) 211–230.
17. Mohammadi, V., Mirzaei, D. and Dehghan, M. *Numerical simulation and error estimation of the time-dependent Allen–Cahn equation on surfaces with radial basis functions*, J. Sci. Comput. 79 (2019) 493–516.
18. Park, J. and Sandberg, I.W. *Universal approximations using radial-basis-function networks*, Neu. Comput. 3 (1991) 246–257.
19. Pham, H. *Handbook of engineering statistics*, Springer, 2006.
20. Shen, J., Tang, T. and Wang, L.L. *Spectral methods: Algorithms, analysis and applications*, Springer-Verlag, Berlin, 2011.
21. Shu, C. and Ding, H. *Numerical comparison of least square based finite difference (LSFD) and local multiquadric-differential quadrature (LMQDQ) methods*, Comput. Math. Appl. (submitted).
22. Shu, C., and Ding, H. and Yeo, K.S. *Local radial basis function-based differential quadrature method and its application to solve two-dimensional incompressible N Navier-Stokes equations*, Comput. Methods Appl. Mech. Eng. 192 (2003) 941–954.
23. Sarra, S.A. *A local radial basis function method for advection-diffusion-reaction equations on complexly shaped domains*, Appl. Math. Comput. 218, 19 (2012) 9853–9865.
24. Tolstykh, A.I., Lipavskii, M.V. and Shirobokov, D.A. *High-accuracy discretization methods for solid mechanics*, Arch. Mech. 55 (2003) 531–553.
25. Wright, G.B. and Fornberg, B. *Scattered node compact finite difference-type formulas generated from radial basis functions*, J. Comput. Phys. 212 (2006) 99–123.

26. Wendland, H. *Scattered data approximations*, Cambridge Monographs on Applied Computational Mathematics, Cambridge University Press, 2005.
27. Wu, X., Zwieten, G.J. and Zee, K.G. *Stabilized second-order convex splitting schemes for Cahn-Hilliard models with application to diffuse-interface tumor-growth models*, Int. J. Numer. Methods Biomed. Eng. 30 (2014) 180–203.
28. Yoon, J. *Spectral approximation order of radial basis function interpolation on the Sobolev space*, SIAM J. Math. Anal. 33 (2001) 946–958.
29. Yoon, J. *L_p -error estimates for “shifted” surface spline interpolation on Sobolev Space*, Math. Comput. 72 (243) (2003) 1349–1367.
30. Yoon, J. *Improved accuracy of L_p -approximation to derivatives by radial basis function interpolation*, Appl. Math. Comput. 161 (2005) 109–119.
31. Zhang, J. and Du, Q. *Numerical studies of discrete approximations to the Allen-Cahn equation in the sharp interface limit*, SIAM J. Sci. Comput. 31 (4) (2009) 3042–3063.



On the numerical solution of optimal control problems via Bell polynomials basis

M. R. Dadashi*, A. R. Haghighi, F. Soltanian and A. Yari

Abstract

We present a new numerical approach to solve the optimal control problems (OCPs) with a quadratic performance index. Our method is based on the Bell polynomials basis. The properties of Bell polynomials are explained. We also introduce the operational matrix of derivative for Bell polynomials. The chief feature of this matrix is reducing the OCPs to an optimization problem. Finally, we discuss the convergence of the new technique and present some illustrative examples to show the effectiveness and applicability of the proposed scheme. Comparison of the proposed method with other previous methods shows that this method is accurate.

AMS(2010): Primary 49J15; Secondary 65M70.

Keywords: Optimal control problems; Bell polynomial; Best approximation; Operational matrix of derivative.

*Corresponding author

Received 14 May 2020 ; revised 10 August 2020 ; accepted 31 August 2020

Mohammad Reza Dadashi

Department of Mathematics, Payame Noor University, Tehran, Iran.

e-mail: dadashi_m.reza@yahoo.com

Ahmad Reza Haghighi

Department of Mathematics, Technical and Vocational University, Tehran, Iran.

e-mail: ah.haghighi@gmail.com

Fahimeh Soltanian

Department of Mathematics, Payame Noor University, Tehran, Iran.

e-mail: f.soltanian@pnu.ac.ir

Ayatollah Yari

Department of Mathematics, Payame Noor University, Tehran, Iran.

e-mail: a.yari@yahoo.com

1 Introduction

Researchers believe that the analytical solution of optimal control problems (OCPs) is usually complicated, or, most of the time, impossible [14]. Hence we use numerical methods to solve them. The first attempt to solve OCPs was applying numerical methods to related Bellman and Pontryagin in 1950; see [16]. The main problem in numerical methods is the accuracy of computations and the amount of manipulations.

In recent years, some studies have been done in this regard, which we will refer only [1, 7, 8, 9, 11, 18, 20, 21, 25]. The investigation of OCPs and solving them by using the Bezier polynomials were proposed by Yari, Mirnia, and Lakestani [25]. To be more specific, they chose Bezier polynomials as basic functions and considered the state vector $x(t)$ and control vector $u(t)$ as Bezier polynomials with unknown coefficients. Grigoryev, Mustafina, and Larin [9] proposed the numerical solution of OCPs, using a successive approximations method. Ramezani [18] presented the numerical solution of OCPs, using the second kind Chebyshev wavelet. Rose [20] described numerical methods to resolve control issues. Kafash and his colleagues [11] proposed a numerical approach for solving OCPs, using the Boubaker polynomials expansion scheme.

In the present study, we offer a numerical solution for the linear constrained quadratic OCPs via Bell polynomials basis. The Bell polynomials were studied extensively by Bell [4] in 1934. These polynomials naturally occur from differentiating a composite function several times. Bell polynomials have many applications in number theory and classical analysis, and there is a vast literature about their applications. They are frequently applied in combinatorial analysis and statistics. Also, these polynomials have been used in many other contexts such as the Blissard problem; see [15]. This method relies on approximating the state and control variables with Bell polynomials. It includes reducing the OCPs to an optimization one by first expanding the state rate $\dot{x}(t)$ and the control $u(t)$ as a Bell polynomial with unknown coefficients. The operational matrix of differentiation D_ϕ is given to approximate the differential part of the problem and the performance index.

This paper is arranged as follows. In Section 2, we present some important preliminaries needed for our subsequent development and approximation of a function by using the Bell polynomials basis. Section 3 is devoted to the numerical method of solving the OCPs. In Section 4, we discuss the proposed method for solving the OCPs. In Section 5, we present the convergence of the proposed method. In Section 6, we consider the applicability as well as the accuracy of the proposed technique with several examples. Finally, the main results are summarized in Section 7.

2 Preliminaries

In this section, we present some preliminaries needed in some other parts of the paper.

2.1 Bell polynomials and their properties

The incomplete exponential Bell polynomials are given as

$$B_{m,n}(t_1, \dots, t_{m-n+1}) = \sum_{\pi(m,n)} \frac{m!}{d_1! \dots d_{m-n+1}!} \left(\frac{t_1}{1!}\right)^{d_1} \dots \left(\frac{t_{m-n+1}}{(m-n+1)!}\right)^{d_{m-n+1}},$$

where $m, n \in \mathbb{Z}^+$, $n \leq m$, and the summation takes place over all integers $d_1, d_2, \dots, d_{m-n+1} \geq 0$, such that

$$\begin{aligned} d_1 + d_2 + \dots + d_{m-n+1} &= n, \\ d_1 + 2d_2 + \dots + (m-n+1)d_{m-n+1} &= m. \end{aligned}$$

The sum

$$B_m(t_1, t_2, \dots, t_m) = \sum_{n=1}^m B_{m,n}(t_1, t_2, \dots, t_{m-n+1}),$$

is called the m th complete exponential Bell polynomial [4].

Suppose that $t_1 = t_2 = \dots = t_m = t$, and let $B_m(t, t, \dots, t) = B_m(t)$, then

$$B_m(t) = \sum_{n=0}^m \{m, n\} t^n, \quad (1)$$

is called the complete Bell polynomial (Touchard polynomial), where

$$\{m, n\} = \frac{1}{n!} \sum_{j=0}^n (-1)^j \binom{n}{j} (n-j)^m,$$

are called Sterling numbers of the second kind and some properties of them are as follows (see [4, 23]):

- (1) $\{m, 0\} = \delta_{m,0}$,
- (2) $\{m, 1\} = 1$,
- (3) $\{m, 2\} = 2^{m-1} - 1$,
- (4) $\{m, n\} = n\{m-1, n\} + \{m-1, n-1\}$,
- (5) $\{m, m\} = 1$,

$$(6) \{m, n\} = 0, n > m.$$

Some sterling numbers of the second kind are given in Table 1:

Table 1: Sterling numbers of the second kind

m	$\{m, 0\}$	$\{m, 1\}$	$\{m, 2\}$	$\{m, 3\}$	$\{m, 4\}$	$\{m, 5\}$	$\{m, 6\}$	$\{m, 7\}$	$\{m, 8\}$	$\{m, 9\}$
0	1									
1	0	1								
2	0	1	1							
3	0	1	3	1						
4	0	1	7	6	1					
5	0	1	15	25	10	1				
6	0	1	31	90	65	15	1			
7	0	1	63	301	350	140	21	1		
8	0	1	127	966	1701	1050	266	28	1	
9	0	1	255	3025	7770	6951	2646	462	36	1

The first six Bell polynomials are as

$$\begin{aligned}
B_0(t) &= 1, \\
B_1(t) &= t, \\
B_2(t) &= t^2 + t, \\
B_3(t) &= t^3 + 3t^2 + t, \\
B_4(t) &= t^4 + 6t^3 + 7t^2 + t, \\
B_5(t) &= t^5 + 10t^4 + 25t^3 + 15t^2 + t, \\
B_6(t) &= t^6 + 15t^5 + 65t^4 + 90t^3 + 31t^2 + t.
\end{aligned}$$

Some properties of Bell polynomials are as follows (see [4, 5, 6]):

- (1) $B_m(1) = B_m$ (Bell number),
- (2) $B_m(-1) = \tilde{B}_m$ (complement of Bell number),
- (3) for all $m, n \in \mathbb{Z}^+$ and $n \leq m$,

1.

$$B_m(t) = e^{-t} \sum_{n=0}^{\infty} \frac{n^m}{n!} t^n,$$

2.

$$\left(t \frac{d}{dt}\right)^j e^{at} = \sum_{i=0}^j \{j, i\} t^i \left(\frac{d}{dt}\right)^j e^{at} = B_j(at) e^{at},$$

3.

$$B_{m+1}(t) = t(B_m(t) + B'_m(t)), \quad (2)$$

4.

$$t \frac{d}{dt} B_m(t) e^t = B_{m+1}(t) e^t,$$

5.

$$t \frac{d}{dt} B_m(t) e^t = t(B_m(t) + B'_m(t)) e^t,$$

6.

$$B_{m+1}(t) = t \sum_{n=0}^m \binom{m}{n} B_n(t), \quad (3)$$

7.

$$B_{m+1}(p+q) = \sum_{n=0}^m \binom{m}{n} B_n(p) B_{m-n}(q).$$

Theorem 1. [semi-orthogonality of $B_m(t)$] For all $m, n \in N$, we have

$$\int_0^\infty B_m(-t) B_n(-t) \frac{e^{-2t}}{t} dt = (-1)^{m-1} \frac{2^{m+n} - 1}{m+n} \mathcal{B}_{m+n}, \quad (4)$$

where $\mathcal{B}_{m+n}$ are the Bernoulli numbers. Note that the right-hand side of (4) is zero, when $m+n$ is odd. Also all Bernoulli numbers $\mathcal{B}_k$ with indices odd and larger than one, are zero.

Proof. For the proof, see [5]. □

We can express the standard basis $\{1, t, t^2, \dots, t^m\}$ in terms of the Bell polynomials as follows:

$$\begin{aligned} 1 &= B_0(t), \\ t &= B_0(t) + B_1(t), \\ t^2 &= B_0(t) - B_1(t) + B_2(t), \\ t^3 &= B_0(t) + 2B_1(t) - 3B_2(t) + B_3(t), \\ t^4 &= B_0(t) - 6B_1(t) + 11B_2(t) - 6B_3(t) + B_4(t). \end{aligned}$$

We expand t^m in terms of Bell polynomials as follows:

$$t^m = \sum_{n=0}^m (-1)^{m-n} [m, n] B_n(t), \quad (5)$$

where $[m, n]$ are called Sterling numbers of the first kind and given as follows (see [22]):

$$[m, n] = \sum_{0 \leq i_1 < \dots < i_{m-n} < m} i_1 i_2 \dots i_{m-n} = \sum_{i_1=0}^{m-1} \sum_{i_2=0}^{i_1-1} \dots \sum_{i_{m-n}=0}^{i_{m-n-1}-1} i_1 i_2 \dots i_{m-n}.$$

We can write (5) as in the following way:

$$t^j = d_j B_j(t), \quad j = 0, 1, 2, \dots, m-1,$$

where d_j , $j = 0, 1, 2, \dots, m-1$, are constant vectors of order $1 \times (m+1)$ and are given as

$$d_j = \sum_{i=0}^j (-1)^{j-i} [j, i].$$

Some properties of Sterling numbers of the first kind are as follows (see [4, 23]):

- (1) $[m, 0] = \delta_{m,0}$,
- (2) $[m, 1] = (n-1)!$,
- (3) $[m, n] = (m-1)[m-1, n] + [m-1, n-1]$,
- (4) $[m, m] = 1$,
- (5) $[m, n] = 0, n > m$.

Some sterling numbers of the first kind are given in Table 2:

Table 2: Sterling numbers of the first kind

m	$[m, 1]$	$[m, 1]$	$[m, 2]$	$[m, 3]$	$[m, 4]$	$[m, 5]$	$[m, 6]$	$[m, 7]$	$[m, 8]$	$[m, 9]$
0	1									
1	0	1								
2	0	1	1							
3	0	2	3	1						
4	0	6	11	6	1					
5	0	24	50	35	10	1				
6	0	120	274	225	85	15	1			
7	0	720	1764	1624	735	175	21	1		
8	0	5040	13068	13132	6769	1960	322	28	1	
9	0	40320	109584	118124	67284	22249	4536	546	36	1

2.2 The operational matrix of the Bell polynomials

In this section, we define the operational matrix of derivative for the Bell polynomials. Suppose that $\Phi_m(t_0)$ on $[\alpha, \beta]$ is defined as

$$\Phi_m(t_0) = [B_0(t_0), B_1(t_0), \dots, B_m(t_0)]^T. \quad (6)$$

Then we have

$$\Phi'_m(t_0) = D_\phi \Phi_m(t_0), \quad (7)$$

where D_ϕ is the $(m+1) \times (m+1)$ operational matrix of derivative for the Bell polynomials and it is given as follows:

$$D_\phi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 \\ 1 & 0 & 0 & 0 & 0 & \dots & \dots & 0 \\ 1 & 2 & 0 & 0 & 0 & \dots & \dots & 0 \\ 1 & 3 & 3 & 0 & 0 & \dots & \dots & 0 \\ 1 & 4 & 6 & 4 & 0 & \dots & \dots & 0 \\ 1 & 5 & 10 & 10 & 5 & \dots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ \binom{m}{0} & \binom{m}{1} & \binom{m}{2} & \binom{m}{3} & \binom{m}{4} & \dots & \binom{m}{m-1} & 0 \end{bmatrix}.$$

Theorem 2. For all $i, j \in Z^+$ and $i \leq j$, we have

$$B'_j(t_0) = \sum_{i=0}^{j-1} \binom{j}{i} B_i(t_0).$$

Proof. Using (2) and (3), we have

$$t_0(B_j(t_0) + B'_j(t_0)) = t_0 \sum_{i=0}^j \binom{j}{i} B_i(t_0);$$

then

$$B_j(t_0) + B'_j(t_0) = \sum_{i=0}^j \binom{j}{i} B_i(t_0).$$

Therefore

$$B'_j(t_0) = \sum_{i=0}^j \binom{j}{i} B_i(t_0) - B_j(t_0) = \sum_{i=0}^{j-1} \binom{j}{i} B_i(t_0) + \binom{j}{j} B_j(t_0) - B_j(t_0),$$

so that

$$B'_j(t_0) = \sum_{i=0}^{j-1} \binom{j}{i} B_i(t_0).$$

□

2.3 Function approximation

Suppose that $H = L^2([\alpha, \beta])$ and that $Y = \text{span}\{B_0(t), B_1(t), \dots, B_m(t)\}$. Since Y is a finite-dimensional and closed subspace, so Y is a complete subspace of H ; see [12]. Thus for an arbitrary element $h(t)$ in H , there exists a unique best approximation out of Y such as $h_0(t)$, such that

$$\text{for all } g(t) \in Y, \quad \|h(t) - h_0(t)\| \leq \|h(t) - g(t)\|.$$

Since $h_0(t) \in Y$, there exist the unique coefficients $c_0, \dots, c_m$, such that

$$h(t) \simeq h_0(t) = \sum_{j=0}^m c_j B_j(t) = C^T \Phi_m(t), \quad (8)$$

where C and $\Phi_m(t)$ are given by

$$\Phi_m(t_0) = [B_0(t_0), B_1(t_0), \dots, B_m(t_0)]^T, \quad C = [c_0, c_1, \dots, c_m]^T,$$

and T denotes the transposition. For computing the coefficients c_j , we let

$$k_i = \langle h(t), B_i(t) \rangle = \int_{\alpha}^{\beta} h(t) B_i(t) dt, \quad i = 0, 1, 2, \dots, m.$$

Hence, using (8), we have

$$\begin{aligned} k_i &= \int_{\alpha}^{\beta} h(t) B_i(t) dt = \int_{\alpha}^{\beta} \sum_{j=0}^m c_j B_j(t) B_i(t) dt = \sum_{j=0}^m c_j \int_{\alpha}^{\beta} B_j(t) B_i(t) dt \\ &= \sum_{j=0}^m c_j \langle B_j(t), B_i(t) \rangle dt = \sum_{j=0}^m c_j r_{ji} dt, \quad i = 0, 1, 2, \dots, m, \end{aligned} \quad (9)$$

where

$$r_{ji} = \langle B_j(t), B_i(t) \rangle = \int_{\alpha}^{\beta} B_j(t) B_i(t) dt,$$

and $\langle \cdot, \cdot \rangle$ denotes the inner product. Let

$$R = [r_{ji}]_{(m+1) \times (m+1)}, \quad K = [k_1, k_2, \dots, k_m]^T.$$

The matrix R is called the dual matrix of Φ_m , which can be obtained as follows:

$$R_{i+1, j+1} = \langle B_i, B_j \rangle = \int_{\alpha}^{\beta} B_i(t) B_j(t) dt = \int_{\alpha}^{\beta} \sum_{p=0}^i \{i, p\} t^p \sum_{q=0}^j \{j, q\} t^q dt$$

$$= \sum_{p=0}^i \sum_{q=0}^j \{i, p\} \{j, q\} \frac{\beta^{p+q+1} - \alpha^{p+q+1}}{p+q+1}.$$

From (9), we have

$$K^T = C^T R \Rightarrow C^T = K^T R^{-1}.$$

Therefore, any function $h(t) \in L^2([\alpha, \beta])$ can be expanded in terms the Bell polynomials as

$$h(t) \simeq C^T \Phi_m(t), \quad (10)$$

where

$$C^T = \langle h(t), \Phi_m(t) \rangle R^{-1}.$$

Also, the Bell polynomials form a complete basis over the interval $[\alpha, \beta]$. Because using (1), we write

$$\Phi_m(t) = SX(t), \quad (11)$$

where

$$\Phi_m(t) = [B_0(t), B_1(t), \dots, B_m(t)]^T, \quad X(t) = [1, t, t^2, \dots, t^m]^T,$$

and S is a lower triangular matrix with nonzero diagonal elements. Since from [15] the matrix S is nonsingular, hence S^{-1} exists. Let

$$\sum_{j=0}^m c_j B_j(t) = C^T \Phi_m(t) = 0, \quad (12)$$

where $C = [c_0, c_0, \dots, c_m]^T$. using (11) and (12), we have

$$C^T \Phi_m(t) = C^T SX(t) = 0. \quad (13)$$

According to the fundamental theorem of algebra, $X(t) = \{1, t, t^2, \dots, t^m\}$ forms a basis; see [12]. Hence they are a linearly independent set. Therefore in (13), we have

$$C^T S = 0 \rightarrow C^T = [c_0, c_0, \dots, c_m] = 0;$$

hence $\Phi_m(t)$ is a linearly independent set. Therefore using (10), the Bell polynomials form a complete basis in $[\alpha, \beta]$.

3 Problem statement

In this section, we discuss the numerical solution of the OCPs. Consider the nonlinear system [17]

$$\dot{x}(t) = \mathcal{A}(t)x(t) + \mathcal{B}(t)u(t), \quad (14)$$

$$x(\alpha) = x^0, \quad x(\beta) = x^1, \quad t \in [\alpha, \beta], \quad (15)$$

where $[\mathcal{A}(t)]_{m \times m}$ and $[\mathcal{B}(t)]_{n \times n}$ are matrix functions and $[x(t)]_{m \times 1}$ and $[u(t)]_{n \times 1}$ are state and control vectors, respectively. The purpose is finding the optimal control $u(t)$ and the corresponding state trajectory $x(t)$, which satisfy (14) and (15) while maximize (or minimize) the quadratic performance index

$$\mathcal{Z} = \frac{1}{2}x^T(\beta)\mathcal{G}(t)x(\beta) + \frac{1}{2} \int_{\alpha}^{\beta} (x^T(t)\mathcal{Q}(t)x(t) + u^T(t)\mathcal{R}(t)u(t))dt, \quad (16)$$

where $[\mathcal{G}(t)]_{m \times m}$ and $[\mathcal{Q}(t)]_{m \times m}$ are symmetric positive semi-definite matrices and $[\mathcal{R}(t)]_{n \times n}$ is a symmetric positive definite matrix.

4 The proposed method

In this section, we discuss a new approach to solve the OCPs by using Bell polynomials. Let

$$x_i(t_0) \simeq \Phi_m^T(t_0)X^i, \quad i = 1, \dots, m, \quad (17)$$

$$u_j(t_0) \simeq \Phi_m^T(t_0)U^j, \quad j = 1, \dots, n, \quad (18)$$

where $\Phi_m(t_0)$ is defined in (6) and $[X^i]_{(m+1) \times 1}$ and $[U^j]_{(m+1) \times 1}$ are state and control coefficient vectors, respectively. Then by using (7), we get

$$\dot{x}_i(t_0) \simeq [D_{\phi}\Phi_m(t_0)]^T X^i,$$

$$\dot{u}_j(t_0) \simeq [D_{\phi}\Phi_m(t_0)]^T U^j.$$

Using (17) and (18), we have

$$x(t_0) \simeq [\Phi_m^T(t_0)X]^T = \left[\sum_{i=0}^m B_i(t_0)X_i^1, \dots, \sum_{i=0}^m B_i(t_0)X_i^m \right]^T, \quad (19)$$

$$u(t_0) \simeq [\Phi_m^T(t_0)U]^T = \left[\sum_{j=0}^m B_j(t_0)U_j^1, \dots, \sum_{j=0}^m B_j(t_0)U_j^n \right]^T, \quad (20)$$

where $X = (X_i^r)_{(m+1) \times m}$ and $U = (U_j^s)_{(m+1) \times n}$ are state and control coefficient matrices, respectively. The boundary conditions in (15) can be rewritten as

$$\begin{aligned} x(\alpha) &= x^0 = d^0 \otimes E \Phi_m(t_0), \\ x(\beta) &= x^1 = d^1 \otimes E \Phi_m(t_0), \end{aligned} \quad (21)$$

where $[d^0]_{m \times 1}$, $[d^1]_{m \times 1}$ and $E = [1, \dots, 1]_{1 \times (m+1)}$ are constant vectors, and the symbol $\otimes$ denotes the Kronecker product [13]. If $x(\alpha)$ or $x(\beta)$ is unknown in (15), then we put

$$\begin{aligned} x(\alpha) &\simeq [\Phi_m^T(\alpha)X]^T = [\sum_{i=0}^m B_i(\alpha)X_i^1, \dots, \sum_{i=0}^m B_i(\alpha)X_i^m]^T, \\ x(\beta) &\simeq [\Phi_m^T(\beta)X]^T = [\sum_{j=0}^m B_j(\beta)X_j^1, \dots, \sum_{j=0}^m B_j(\beta)X_j^n]^T. \end{aligned} \quad (22)$$

4.1 Performance index approximation

By substituting (19), (20), and (21) in (16), we obtain

$$\begin{aligned} \max(\min) \mathcal{Z} &= \frac{1}{2} x^{\beta T} \mathcal{G}(\beta) x^\beta + \frac{1}{2} X^T \left[\int_\alpha^\beta \Phi_m^T(t) \mathcal{Q}(t) \Phi_m(t) dt \right] X \\ &\quad + \frac{1}{2} U^T \left[\int_\alpha^\beta \Phi_m^T(t) \mathcal{R}(t) \Phi_m(t) dt \right] U. \end{aligned} \quad (23)$$

Let

$$P_x^* = \int_\alpha^\beta \Phi_m^T(t) \mathcal{Q}(t) \Phi_m(t) dt \quad \text{and} \quad P_u^* = \int_a^\beta \Phi_m^T(t) \mathcal{R}(t) \Phi_m(t) dt. \quad (24)$$

By replacing (22) and (24) in (23), we obtain

$$\mathcal{Z}[X, U] = \frac{1}{2} X^T (P^* + P_x^*) X + \frac{1}{2} U^T P_u^* U, \quad (25)$$

where

$$P^* = \Phi_m^T(\beta) \mathcal{G}(\beta) \Phi_m(\beta).$$

The boundary conditions in (15) can be expressed as

$$q_k^0 = x_k(a) - x_k^0, \quad k = 1, \dots, m, \quad (26)$$

$$q_k^1 = x_k(b) - x_k^1, \quad k = 1, \dots, m. \quad (27)$$

We now find the extremum of (25) subject to (26) and (27) by using the Lagrange multiplier technique. Let

$$\mathcal{Z}[X, U, \lambda^0, \lambda^1] = \mathcal{Z}[X, U] + \lambda^0 \mathcal{Q}^0 + \lambda^1 \mathcal{Q}^1, \quad (28)$$

where $\mathcal{Q}^0 = [q_k^0]_{m \times 1}$ and $\mathcal{Q}^1 = [q_k^1]_{m \times 1}$ are constant vectors. The necessary condition for the extremum of (28), is

$$\nabla \mathcal{Z}[X, U, \lambda^0, \lambda^1] = 0.$$

5 Convergence analysis

In this section, we present the convergence of the proposed method.

Theorem 3. Suppose that the function $h : [\alpha, \beta] \rightarrow R$ is $m + 1$ times continuously differentiable (i.e., $f \in C^{m+1}[\alpha, \beta]$) and that $S_m = \text{span}\{\Phi_m(t)\}$. If $C^T B$ is the best approximation of h out of S_m , then

$$\|h - C^T B\|_{L^2[\alpha, \beta]} \leq \frac{\hat{U}}{(m+1)!} \sqrt{\frac{\beta^{2m+3} - \alpha^{2m+3}}{2m+3}},$$

where $\hat{U} = \max |h^{(m+1)}(t)|$, $t \in [\alpha, \beta]$.

Proof. We know that, the set $\{1, t, t^2, \dots, t^m\}$ is a basis for polynomials space of degree m . Considering

$$y_1(t) = h(\alpha) + th'(\alpha) + \dots + \frac{t^m}{m!} h^{(m)}(\alpha),$$

and applying Taylor expansion, we have

$$|h(t) - y_1(t)| = |h^{(m+1)}(\xi_t) \frac{t^{(m+1)}}{(m+1)!}|, \quad \xi_t \in (\alpha, \beta). \quad (29)$$

Since $C^T B$ is the best approximation of h out of S_m and $y_1(t) \in S_m$, using (29), we obtain

$$\begin{aligned} \|h - C^T B\|_{L^2[\alpha, \beta]}^2 &\leq \|h - y_1\|_{L^2[\alpha, \beta]}^2 = \int_{\alpha}^{\beta} |h(t) - y_1(t)|^2 dx \\ &= \int_{\alpha}^{\beta} |h^{(m+1)}(\xi_t)|^2 \left(\frac{t^{(m+1)}}{(m+1)!}\right)^2 dt \\ &\leq \left(\frac{\hat{U}}{(m+1)!}\right)^2 \int_{\alpha}^{\beta} t^{2m+2} dt = \left(\frac{\hat{U}}{(m+1)!}\right)^2 \frac{\beta^{2m+3} - \alpha^{2m+3}}{2m+3}, \end{aligned}$$

so

$$\|h - C^T B\|_{L^2[\alpha, \beta]} \leq \frac{\hat{U}}{(m+1)!} \sqrt{\frac{\beta^{2m+3} - \alpha^{2m+3}}{2m+3}}.$$

Also, we can write

$$\frac{\hat{U}}{(m+1)!} \sqrt{\frac{\beta^{2m+3} - \alpha^{2m+3}}{2m+3}} = \frac{\hat{U}}{(m+1)!} \sqrt{\frac{\beta^{2m+3}(1 - (\frac{\alpha}{\beta})^{2m+3})}{2m+3}}. \quad (30)$$

Therefore, using (30), we have

$$\|h - C^T B\|_{L^2[\alpha, \beta]} \leq \frac{\hat{U}}{(m+1)!} \sqrt{\frac{\beta^{2m+3}(1 - (\frac{\alpha}{\beta})^{2m+3})}{2m+3}} \approx \frac{\hat{U} \beta^m}{(m+1)!} \sqrt{\frac{b^3}{2m+3}},$$

which shows that the error vanishes as $m \rightarrow \infty$. $\square$

6 Illustrative examples

To illustrate the proposed technique, we solve some numerical examples and make a comparison with some of the results in the literature in this section. We implemented our method with MATLAB.

Example 1. This example has been taken from [25]. Consider

$$\min \mathcal{Z} = \int_0^1 (3x^2(t) + u^2(t))dt, \quad (31)$$

subject to

$$\dot{x}(t) = x(t) + u(t), \quad x(0) = 1.$$

We solve the problem (31), for $m = 3$. Let

$$x_{app}(t) = X\Phi_3(t), \quad u_{app}(t) = U\Phi_3(t),$$

we obtain

$$X = [1, -\frac{905}{236}, \frac{285}{118}, -\frac{35}{118}], \quad U = [-\frac{641}{236}, \frac{1835}{236}, -\frac{195}{59}, \frac{35}{118}].$$

Then, the approximate solutions are as follows:

$$\begin{aligned} x_{app}(t) &= B_0 - \frac{905}{236}B_1 + \frac{285}{118}B_2 - \frac{35}{118}B_3 \\ &= -\frac{35}{118}t^3 + \frac{90}{59}t^2 - \frac{405}{236}t + 1, \end{aligned} \quad (32)$$

$$\begin{aligned}
u_{app}(t) &= -\frac{641}{236}B_0 + \frac{1835}{236}B_1 - \frac{195}{59}B_2 + \frac{35}{118}B_3 \\
&= \frac{35}{118}t^3 - \frac{285}{118}t^2 + \frac{1125}{236}t - \frac{641}{236}.
\end{aligned} \tag{33}$$

The analytical solutions are as follows:

$$x_{exa}(t) = \frac{3e^{-4}}{3e^{-4} + 1}e^{2t} + \frac{1}{3e^{-4} + 1}e^{-2t}, \tag{34}$$

$$u_{exa}(t) = \frac{3e^{-4}}{3e^{-4} + 1}e^{2t} - \frac{3}{3e^{-4} + 1}e^{-2t}, \tag{35}$$

and the optimal value of objective function is $\mathcal{Z}_{exa} = 2.791659975310063$.

In Table 3, we compare the absolute errors of objective function $\mathcal{Z}$ of the proposed method with methods of [25, 24] for different values of m . Obviously, the estimated results of our proposed scheme are coincided with the results in [25, 24]. Figures 1 and 2 show, respectively, plots of errors for state and control functions for Example 1, by using the presented method.

Table 3: Comparing absolute errors of $\mathcal{Z}$ of the proposed method with the methods of [24, 25] for different values of m for Example 1.

m	proposed method	model of [25]	model of [24]
5	1.9512×10^{-15}	1.9512×10^{-15}	1.9512×10^{-15}
6	1.3590×10^{-10}	1.3590×10^{-10}	1.3590×10^{-10}
7	3.7582×10^{-12}	3.7582×10^{-12}	3.7582×10^{-12}
8	1.9512×10^{-15}	2.0000×10^{-15}	2.0228×10^{-15}

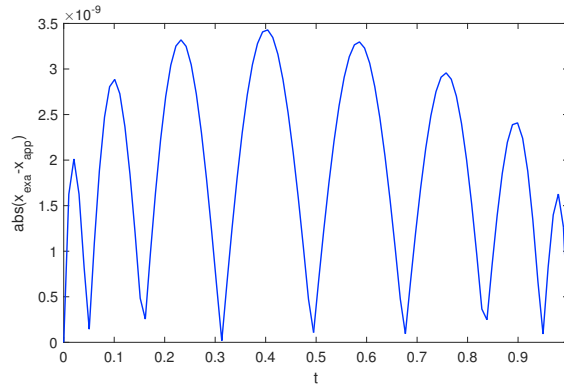


Figure 1: Absolute error for the state function $x(t)$ for Example 1, $m = 8$.

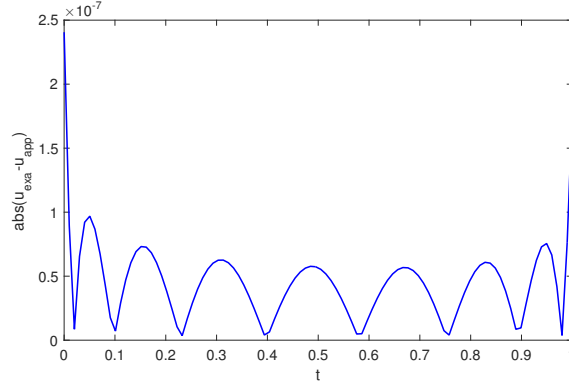


Figure 2: Absolute error for the control function $u(t)$ for Example 1, $m = 8$.

Example 2. This example has been adapted from [19]. Consider

$$\min \mathcal{Z} = \int_0^1 (x_2(t) + u^2(t))dt, \quad (36)$$

subject to

$$\dot{x}_1(t) = x_2(t), \quad x_1(0) = 0, \quad x_1(1) = 1, \quad \dot{x}_2(t) = u(t), \quad x_2(0) = 0.$$

We solve the problem (36), for $m = 3$. Let

$$x_{app}^1(t) = \Phi_3^T(t)X_1, \quad x_{app}^2(t) = \Phi_3^T(t)X_2, \quad u_{app}(t) = \Phi_3^T(t)U.$$

we obtain

$$X_1 = [0, -\frac{5}{2}, 3, -\frac{1}{2}], \quad X_2 = [0, \frac{9}{2}, -\frac{3}{2}, 0], \quad U = [3, -3, 0, 0].$$

Then, the approximate solutions are as follows:

$$\begin{aligned} x_{app}^1(t) &= -\frac{5}{2}B_1 + 3B_2 - \frac{1}{2}B_3 = \frac{3}{2}t^2 - \frac{1}{2}t^3 \\ x_{app}^2(t) &= \frac{9}{2}B_1 - \frac{3}{2}B_2 = 3t - \frac{3}{2}t^2, \\ u_{app}(t) &= 3B_0 - 3B_1 = 3 - 3t. \end{aligned}$$

The analytical solutions are as follows:

$$x_{exa}^1(t) = \frac{3}{2}t^2 - \frac{1}{2}t^3, \quad x_{exa}^2(t) = 3t - \frac{3}{2}t^2, \quad u_{exa}(t) = 3 - 3t,$$

and the optimal value of objective function is $\mathcal{Z}_{exa} = 4$.

Example 3. In this example, extracted from [10], the vibration of a spring-mass-damper system subjected to an external force is considered. In particu-

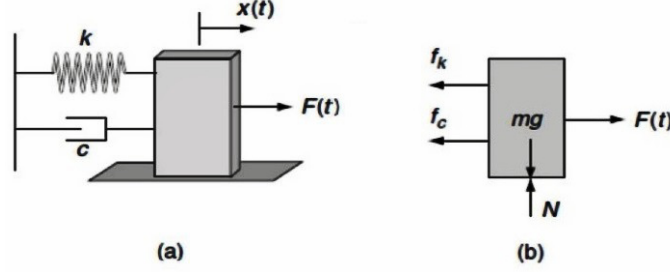


Figure 3: (a) Schematic of the forced-mass-damper system assuming no friction the surface and (b) free body diagram of the system of part (a) for Example 2.

lar, the response to harmonic excitations, impulses, and step forcing functions is examined. In many environments, rotation machinery, motors, and so on cause periodic motions of structures to induce vibrations into other mechanical devices and structures nearby. On summing the forces, the equation for the forced vibration of the system in Figure 3 becomes. It is common to approximate the driving forces, $F(t)$,

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = F(t),$$

where m, c , and k are fixed numbers and $F(t)$ represents the control force derived from the action of one actuator force represented by $F(t) = bu(t)$, where b is a fixed number. The linear regulator problem specific application in vibration suppression, the performance index for this problem is

$$\mathcal{Z} = \frac{1}{2} \int_{t_0}^{t_f} (x^2(t) + au^2(t))dt, \quad (37)$$

where t_0 and t_f are initial and final times, respectively. We introduce the usual state variable notation

$$x_1(t) = x(t), \quad x_2(t) = \dot{x}(t).$$

Then

$$\dot{x}_1(t) = x_2(t), \quad \dot{x}_2(t) = \ddot{x}(t) = -\frac{k}{m}x_1 - \frac{c}{m}x_2 + \frac{b}{m}.$$

The boundary conditions for this example are considered as

$$x(0) = \dot{x}(0) = 1, \quad x(1) = \dot{x}(1) = 0 \quad \text{or} \quad x_1(0) = x_2(0) = 1, \quad x_1(1) = x_2(1) = 0.$$

We solve the problem (37), using Bell polynomials for $m = 3$, $t_0 = 0$, $t_f = 1$, $c = 2$, and $a = b = m = k = 1$. Let

$$x_{app}^1(t) = \Phi_3^T(t)X_1, \quad x_{app}^2(t) = \Phi_3^T(t)X_2, \quad u_{app}(t) = \Phi_3^T(t)U.$$

We obtain

$$X_1 = [1, 12, -14, 3], \quad X_2 = [1, -19, 9, 0], \quad U = [-7, -8, 4, 3].$$

Then, the approximate solutions are as follows:

$$\begin{aligned} x_{app}^1(t) &= B_0 + 12B_1 - 14B_2 + 3B_3 = 3t^3 - 5t^2 + t + 1, \\ x_{app}^2(t) &= B_0 - 19B_1 + 9B_2 = 9t^2 - 10t + 1, \\ u_{app}(t) &= -7B_0 - 8B_1 + 4B_2 + 3B_3 = 3t^3 + 13t^2 - t - 7. \end{aligned}$$

The analytical solutions are as follows:

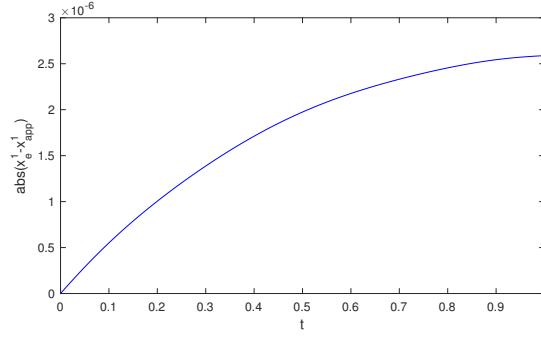
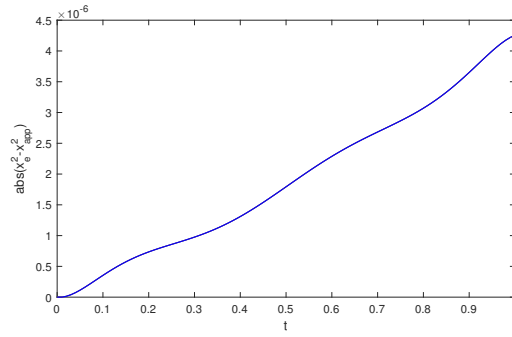
$$\begin{aligned} x_{exa}^1(t) &= \left[\frac{1393}{95} e^{-\frac{167}{152}t} + \frac{679}{219} e^{\frac{167}{152}t} \right] \sin\left(\frac{67}{167}t\right) \\ &\quad + \left[\frac{1574}{423} e^{-\frac{167}{152}t} - \frac{1151}{423} e^{\frac{167}{152}t} \right] \cos\left(\frac{67}{167}t\right), \\ x_{exa}^2(t) &= \left[\frac{2838}{613} e^{-\frac{167}{152}t} - \frac{12872}{723} e^{\frac{167}{152}t} \right] \sin\left(\frac{67}{167}t\right) \\ &\quad + \left[\frac{1021}{395} e^{-\frac{167}{152}t} - \frac{1151}{423} e^{\frac{626}{423}t} \right] \cos\left(\frac{67}{167}t\right), \\ u_{exa}(t) &= \left[-\frac{2931}{1145} e^{-\frac{167}{152}t} + \frac{15449}{851} e^{\frac{167}{152}t} \right] \sin\left(\frac{67}{167}t\right) \\ &\quad + \left[\frac{2591}{1263} e^{-\frac{167}{152}t} + \frac{2326}{421} e^{\frac{167}{152}t} \right] \cos\left(\frac{67}{167}t\right), \end{aligned}$$

and the optimal value of objective function is $\mathcal{Z}_{exa} = 13.00484054100315$.

Table 4 shows the errors of x_1, x_2, u and approximate values of objective function $\mathcal{Z}$, and Figures 4, 5, and 6 show the absolute errors for states of x_1, x_2 and control of u functions for Example 3, by using the presented method.

Table 4: Approximations of $\mathcal{Z}$ and errors of x_1, x_2, u , for Example 3.

m	$\mathcal{Z}_{app}$	errors of x_1	errors of x_2	errors of u
4	13.01222484276730	1.9070×10^{-3}	1.2649×10^{-2}	1.2145×10^{-1}
5	13.01900000000000	7.6740×10^{-6}	7.2401×10^{-5}	9.4197×10^{-4}
6	13.00484749962340	2.9364×10^{-6}	2.6746×10^{-5}	4.1151×10^{-4}
7	13.00484741498875	1.9166×10^{-6}	2.2206×10^{-6}	8.6153×10^{-6}

Figure 4: Absolute errors of x_1 for Example 3, $m = 7$.Figure 5: Absolute errors of x_2 for Example 3, $m = 7$.

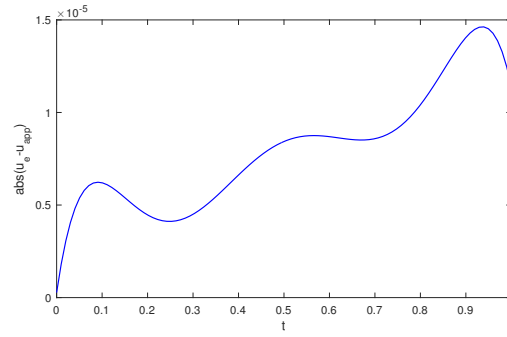
Example 4. This example has been taken from [2]. Consider

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

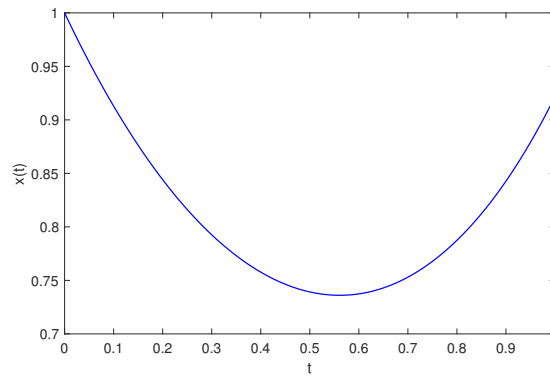
subject to

$$\dot{x}(t) = tx(t) + u(t), \quad x(0) = 1.$$

In Table 5, we compare the results of the optimal values of objective function $\mathcal{Z}$ of the proposed method with methods of [2, 3] for $m = 8$. Obviously, the estimated results for the minimum values of $\mathcal{Z}$ of the proposed method are more accurate than the results in models of [2, 3]. Figures 7 and 8 illustrate the behavior of state variable $x(t)$ and control variable $u(t)$ respectively, by using the presented method for $m = 8$.

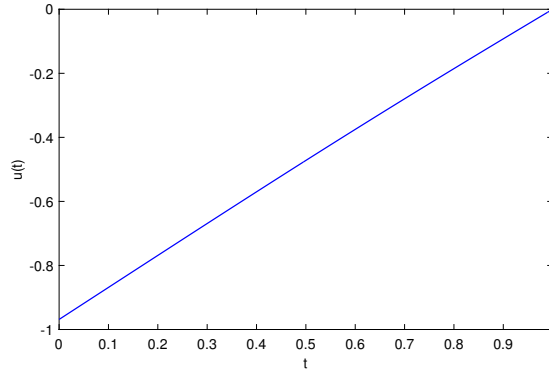
Figure 6: Absolute errors of u for Example 3, $m = 7$.Table 5: Comparing approximate values of $\mathcal{Z}$ of the proposed method with the methods of [2, 3] for Example 4, $m = 8$.

m	proposed method	model of [2]	model of [3]
8	0.48426769622877	0.484268	0.48427

Figure 7: Behavior of $x(t)$ for Example 4, $m = 8$.

Example 5. This example has been adapted from [26]. Consider

$$\min \mathcal{Z} = \frac{1}{2} \int_0^1 (x_1^2(t) + x_2^2(t) + u^2(t)) dt,$$

Figure 8: Behavior of $u(t)$ for Example 4, $m = 8$.

subject to

$$\begin{aligned}\dot{x}_1(t) &= -x_1(t) + x_2(t) + u(t), \quad x(0) = 1, \\ \dot{x}_2(t) &= -2x_2(t), \\ x_1(0) &= x_2(0) = 1.\end{aligned}$$

The analytical solutions of this problem are as follows:

$$\begin{aligned}x_1(t) &= \frac{-3}{2}e^{-2t} + 2.48165e^{-\sqrt{2}t} + 0.018352e^{\sqrt{2}t}, \\ x_2(t) &= e^{-2t}, \\ u(t) &= \frac{1}{2}e^{-2t} - 1.027922e^{-\sqrt{2}t} + 0.044305e^{\sqrt{2}t},\end{aligned}$$

and $\mathcal{Z} = 0.43198832823734928111$. In Tables 6, 7, and 8, we compare the results of the absolute errors of $x_1(t)$, $x_2(t)$, and $u(t)$ of the proposed method with the method of [26] for different points of time and values of m , respectively. Obviously, the estimated results of the proposed method are more accurate than the results in [26]. Figures 9, 10, and 11 illustrate the behavior of state variables $x_1(t)$, $x_2(t)$ and control variable $u(t)$ by using the presented method for $m = 8$, respectively.

Table 6: Comparing absolute errors of $x_1(t)$ of the proposed method with the method of [26] for Example 5, different values of m .

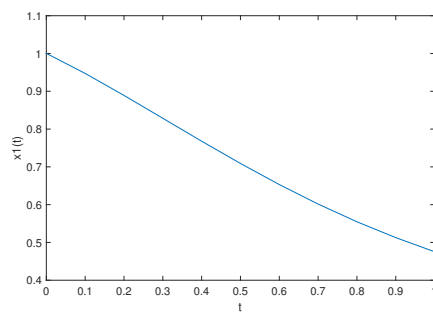
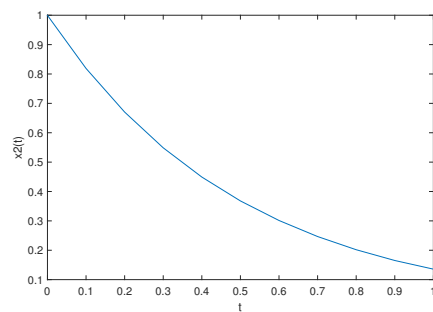
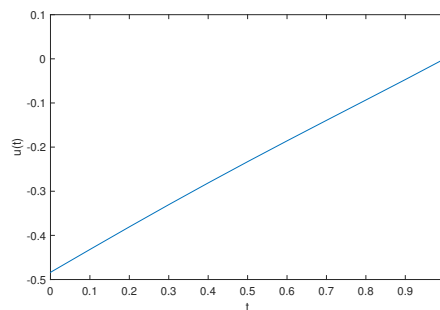
t	proposed method		model of [26]	
	$m = 4$	$m = 6$	$m = 4$	$m = 6$
0.1	1.5215×10^{-4}	2.4791×10^{-6}	6.2840×10^{-5}	6.9470×10^{-6}
0.2	2.9911×10^{-5}	4.1683×10^{-7}	3.8617×10^{-4}	5.6627×10^{-6}
0.3	1.2214×10^{-4}	1.5184×10^{-6}	4.7516×10^{-4}	1.4208×10^{-6}
0.4	1.0726×10^{-4}	2.1591×10^{-6}	2.5623×10^{-4}	5.4834×10^{-7}
0.5	2.2672×10^{-5}	8.9684×10^{-7}	1.1356×10^{-4}	3.5375×10^{-6}
0.6	1.2331×10^{-4}	8.4596×10^{-8}	4.0262×10^{-4}	5.8476×10^{-6}
0.7	8.1944×10^{-5}	9.0533×10^{-7}	4.2904×10^{-4}	3.9087×10^{-6}
0.8	8.3818×10^{-5}	1.8245×10^{-6}	1.5915×10^{-4}	1.1543×10^{-7}
0.9	1.4779×10^{-4}	2.4645×10^{-7}	2.1417×10^{-4}	1.8431×10^{-6}

Table 7: Comparing absolute errors of $x_2(t)$ of the proposed method with the models of [26] for Example 5, different values of m .

t	proposed method		model of [26]	
	$m = 4$	$m = 6$	$m = 4$	$m = 6$
0.1	1.6805×10^{-4}	6.0534×10^{-7}	5.9651×10^{-5}	1.8022×10^{-6}
0.2	4.1122×10^{-5}	9.0967×10^{-7}	4.0570×10^{-4}	1.2818×10^{-6}
0.3	1.2262×10^{-4}	1.4305×10^{-7}	5.1060×10^{-4}	1.7540×10^{-6}
0.4	1.1136×10^{-4}	8.0849×10^{-7}	2.8780×10^{-4}	2.1099×10^{-6}
0.5	2.4713×10^{-5}	1.0834×10^{-7}	1.0392×10^{-4}	6.6340×10^{-7}
0.6	1.3364×10^{-4}	8.2323×10^{-7}	4.2076×10^{-4}	2.8671×10^{-6}
0.7	9.2876×10^{-5}	6.4668×10^{-8}	6.6499×10^{-4}	1.5552×10^{-6}
0.8	8.3272×10^{-5}	8.9367×10^{-7}	1.9246×10^{-4}	1.5283×10^{-6}
0.9	1.5574×10^{-4}	7.4550×10^{-7}	1.9951×10^{-4}	1.2855×10^{-6}

Table 8: Comparing absolute errors of $u(t)$ of the proposed method with the models of [26] for Example 5, different values of m .

t	proposed method		model of [26]	
	$m = 4$	$m = 6$	$m = 4$	$m = 6$
0.1	2.9430×10^{-5}	1.1055×10^{-6}	1.0466×10^{-5}	2.5978×10^{-6}
0.2	9.1300×10^{-7}	1.2850×10^{-6}	8.0579×10^{-5}	2.1213×10^{-6}
0.3	3.4541×10^{-5}	5.5935×10^{-7}	8.9374×10^{-5}	8.6406×10^{-7}
0.4	2.7359×10^{-5}	9.289×10^{-10}	2.5833×10^{-5}	5.0892×10^{-7}
0.5	4.9785×10^{-6}	1.4324×10^{-8}	6.7342×10^{-5}	1.3463×10^{-6}
0.6	2.7466×10^{-5}	5.9741×10^{-8}	1.3411×10^{-5}	1.9603×10^{-6}
0.7	1.5686×10^{-5}	7.1574×10^{-7}	1.3507×10^{-4}	1.3192×10^{-6}
0.8	2.2792×10^{-5}	1.3485×10^{-6}	6.8786×10^{-5}	1.1611×10^{-6}
0.9	3.4034×10^{-5}	1.1203×10^{-6}	1.3754×10^{-5}	1.0032×10^{-7}

Figure 9: Behavior of $x_1(t)$ for Example 5, $m = 8$.Figure 10: Behavior of $x_2(t)$ for Example 5, $m = 8$.Figure 11: Behavior of $u(t)$ for Example 5, $m = 8$.

7 Conclusion

In the presented study, we have provided a computational approach to solve linear constrained quadratic OCPs by employing the Bell polynomials. Several test problems have been studied to illustrate the applicability and efficiency of the proposed method. Results have revealed that the new technique can solve the OCPs effectively. Comparison of the proposed method with other previous methods showed that this method is accurate.

References

1. Aguilar, C. and Krener, A. *Numerical solutions to the Bellman equation of optimal control*, J. Optim. Theory Appl. 160 (2014) 527–552.
2. Ahmed, H.F. and Melad, M.B. *A new approach for solving fractional optimal control problems using shifted ultraspherical polynomials*, Prog. Fract. Differ. Appl. 4(3) (2018) 179–195.
3. Akbarian, T. and Keyanpour, M. *A new approach to the numerical solution of FOCPs*, Applications and Applied Mathematics, 8 (2) (2013) 523–534.
4. Bell, E.T. *Exponential polynomials*. Ann. Math. (2) 35 (1934), no. 2, 258–277.
5. Boyadzhiev, K.N. *Exponential polynomials, Stirling numbers and evaluation of some gamma integrals*, Abstr. Appl. Anal. 2009, Art. ID 168672, 18 pp.
6. Feng, Q. and Guo B.N. *Relations, among Bell polynomials, central factorial numbers, and central Bell polynomials*, Mathematical Sciences and Applications, 7 (2) (2019) 191–194.
7. Frego, M. *Numerical methods for optimal control problems with application to autonomous vehicles*, Ph.D. Thesis, University of Trento, 2014.
8. Ghomanjani, F. and Farahi, M.H. *Optimal control of switched systems based on Bezier control points*, Int. J. Intell. Syst. Appl. 7 (2012) 16–22.
9. Grigoryev, I., Mustafina, S. and Larin, O. *Numerical solution of optimal control problems by the method of successive approximations*, Int. J. Pure Appl. Math. 112(3) (2017) 599–604.
10. Inman, D.J. *Vibration with control*, John Wiley Sons, Ltd. 2006.

11. Kafash, B., Delavarkhalafi, A., Karbassi, M. and Boubaker, K. *A numerical approach for solving optimal control problems using the Boubaker polynomials expansion scheme*, J. Interpolat. Approx. Sci. Comput. 2014, Art. ID 00033, 18 pp.
12. Kreyszig, E. *Introductory functional analysis with applications*, John Wiley & Sons, New York-London-Sydney, 1978.
13. Lancaster, P. *Theory of Matrices*, New York, Academic Press, 1969.
14. Lewis, F.L., Vrabie, D.L. and Syrmos, V.L. *Optimal control*, Third edition. John Wiley & Sons, Inc., Hoboken, NJ, 2012.
15. Mirzaee, F. *Numerical solution of nonlinear Fredholm-Volterra integral equations via Bell polynomials*, Comput. Methods Differ. Equ. 5(2) (2017) 88–102.
16. Oruh, I.B. and Agwu, U.E. *Application of Pontryagin's maximum principles and Runge-Kutta methods in optimal control problems*, IOSR Journal of mathematics, 11(5) (2015) 43–63.
17. Pesch, H.J. *A practical guide to the solution of real life optimal control problems Parametric optimization*, Control Cybernet. 23 (1994) 7–60.,
18. Ramazani, M. *Numerical solution of optimal control problems by using a new second kind Chebyshev wavelet*, Comput. Methods Differ. Equ. 4 (2016), 4(2) (2016) 162–169.
19. Rogalsky, T. *Bezier parameterization for optimal control by differential evolution*, Proceedings of the 14th annual conference companion on Genetic and evolutionary computation, (2012) 523–530.
20. Rose, G.R. *Numerical methods for solving optimal control problems*, Master's Thesis, University of Tennessee, 2015.
21. Sharif, H.R., Vali, M.A., Samava M. and Gharavisi, A.A. *A new algorithm for optimal control of time-delay systems*, Appl. Math. Sci. (Ruse) 5(12) (2011) 595–606.
22. Stanley, R.P. *Enumerative combinatorics*, Cambridge University Press, 2011.
23. Wakhare, T. *Refinements of the Bell Stirling numbers*, Trans. Comb. 7(4) (2018) 25–42.
24. Yari, A.A. and Mirnia, M. *Solving optimal control problems by using hermite polynomials*, Comput. Methods Differ. Equ. 8(2) (2020) 314–329.
25. Yari, A.A., Mirnia M. and Lakestani, M. *Investigation of optimal control problems and solving them by using Bezier polynomials*, Appl. Comput. Math., 16(2) (2017) 133–147.

26. Yousefi, S.A., Lotfi, A. and Dehghan, M. *The use of a Legendre multi-wavelet collocation method for solving the fractional optimal control problems*, J. Vib. Control 17(13) (2011) 2059–2065.



An approximate method based on Bernstein polynomials for solving fractional PDEs with proportional delays

A. Ketabdari, M.H. Farahi* and S. Effati

Abstract

We apply a new method to solve fractional partial differential equations (FPDEs) with proportional delays. The method is based on expanding the unknown solution of FPDEs with proportional delays by the basis of Bernstein polynomials with unknown control points and uses operational matrices with the least-squares method to convert the FPDEs with proportional delays to an algebraic system in terms of Bernstein coefficients (control points) approximating the solution of FPDEs. We use the Caputo derivatives of degree $0 < \alpha \leq 1$ as the fractional derivatives in our work. The main advantage of using this technique is that the method can easily be employed to a variety of FPDEs with or without proportional delays, and also the method offers a very simple and flexible framework for direct approximating of the solution of FPDEs with proportional delays. The convergence analysis of the present method is discussed. We show the effectiveness and superiority of the method by comparing the results obtained by our method with the results of some available methods in two numerical examples.

AMS(2010): 26A33; 34A25; 35R11.

Keywords: Fractional partial differential equation; Bernstein polynomial; Operational matrix; Caputo derivative.

*Corresponding author

Received 5 April 2020; revised 22 June 2020; accepted 4 September 2020

Ali Ketabdari

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: aliketabdari@mail.um.ac.ir

Mohammad Hadi Farahi

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: Farahi@math.um.ac.ir

&

The Center of Excellence on Modelling and Control Systems (CEMCS), Mashhad, Iran

Sohrab Effati

Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: s-effati@um.ac.ir

1 Introduction

Fractional partial differential equations (FPDEs) plays an important role in recent years. Many significant phenomena in mechanics, biology, networks, signal processing, reaction process, and other fields in sciences and engineering are modeled by FPDEs. Although FPDEs are exploited to describe these phenomena carefully, but it is very challenging and irritating to obtain the exact solutions or at last approximate solutions of FPDEs. In recent years, many numerical methods are used to solve FPDEs. For sample, Chebyshev collocation method [1, 6, 19], Legendre collocation method [29], Jacobi collocation method [9, 28], variational iteration method (VIM) [13, 26], homotopy perturbation method (HPM) [15, 23], Laplace transform method [4], Adomian decomposition method [25], and finite difference method [30].

FPDE with proportional delays is a particular type of FPDEs. This case of FPDEs models various phenomena such as shallow-water waves or nonlinear waves, which come from quantum field theory, medicine, biology, and climate models. For more details and various applications, we refer the interested reader to [20, 27]. Recently, some scholars have attempted to solve these types of equations. Sakar, Uludag, and Erdogan [20] applied the homotopy perturbation method to obtain an approximate solution of FPDEs with proportional delays. Singh and Kumar [22] used VIM for solving FPDEs with proportional delays. Shah et al. [21] suggested the natural transform decomposition method for constructing approximate solutions of FPDEs with proportional delays.

Bezier functions are appropriate tools for obtaining the solution to these cases of problems. Easy definition, well performance, and interpolation of end-points are some virtues of these tools. Many scholars have applied control points of the Bernstein–Bezier form in their papers. Zheng, Sederberg, and Johnson [31] introduced the Bernstein–Bezier form for solving ordinary differential equations. Chakrabarti and Martha [3] used Bezier curves for solving the Fredholm integral equations of the second kind. Ghomanjani and Khorram [5] used this approach for solving the quadratic Riccati differential equation. Karimi, Bahadorimehr, and Mansoorzadeh [8] proposed the use of this method to solve PDEs. Also, some researchers applied the Bezier control points method to obtain an approximate solution of fractional problems; one can see [2, 7, 10, 11, 14, 17, 18] and references therein. We suggest a technique similar to the one used in [10], for solving two-dimensional fractional optimal control problems.

In this paper, we consider the following FPDE with proportional delays

$${}_0D_t^\alpha u(x, t) = h(x, t, u(a_0x, b_0t), \frac{\partial u(a_1x, b_1t)}{\partial x}, \dots, \frac{\partial^{n_1} u(a_{n_1}x, b_{n_1}t)}{\partial x^{n_1}}), \quad (1)$$

with initial conditions

$$u^{(k)}(x, 0) = s^{(k)}(x), \quad k = 0, 1, 2, \dots, q-1, \quad (2)$$

where $x, t \in [0, 1]$, $a_i, b_i \in [0, 1]$ for $i \in \{0, 1, 2, \dots, n_1\}$, also $s^{(k)}(x)$, $k = 0, 1, 2, \dots, q-1$, where $q \in \mathbb{N}$, are smooth functions and h is the partial differential operator. Also the fractional derivative is defined in Caputo sense. The real numbers a_i 's and b_i 's are given constant proportional delays. The necessary conditions for the existence of the solution of (1)–(2) and regularity properties of solutions can be found in [20, 21, 22].

Bezier control points with least-squares method are employed for solving FPDEs with proportional delays (1)–(2). The novelty of our strategy is in producing operational matrices—based on Bernstein polynomials— together with the least-squares method converting FPDEs with proportional delays to a simple optimization problem.

This article is formed as follows: Section 2 introduces some properties of fractional derivatives and Bernstein polynomials. In Section 3, we define a class of FPDEs with proportional delays and employ operational matrices based on Bernstein polynomials to convert these equations to a mathematical programming problem in terms of some unknowns control points. In the fourth section, the convergence of the proposed method is studied. The applicability and efficiency of the proposed method are shown through two numerical examples, and comparison with some other methods is investigated in Section 5. Finally, the conclusion is presented in Section 6.

2 Preliminaries

This section contains some basic definitions and properties of fractional calculus, Bernstein polynomials, and function approximation.

2.1 Some preliminaries in fractional derivatives

Definition 1. Let $\mu \in \mathbb{R}$ and let $n \in \mathbb{N}$. A real function $f(x)$, $x > 0$ belongs to the space C_μ if there exist a real number $r > \mu$ and $g(x) \in C[0, \infty]$ such that $f(x) = x^r g(x)$. Moreover, $f \in C_\mu^n$ if and only if $f^{(n)} \in C_\mu$.

Definition 2. The Riemann–Liouville fractional integral of order $\alpha > 0$ is defined as follows:

$${}_0I_x^\alpha(f(x)) = \frac{1}{\Gamma(\alpha)} \int_0^x (x - \tau)^{\alpha-1} f(\tau) d\tau, \quad x > 0,$$

where $f(x) \in C_\mu$ and $\Gamma(\alpha) = \int_0^\infty \tau^{(\alpha-1)} e^{-\tau} d\tau$ is Euler's gamma function.

Definition 3. The Caputo fractional derivative of order $\alpha > 0$ of the given function $f(x) \in C_\mu$ is defined as

$$\begin{aligned} {}_0D_x^\alpha f(x) &= {}_0I_x^{n-\alpha} D^n f(x) \\ &= \frac{1}{\Gamma(n-\alpha)} \int_0^x (x-\tau)^{n-\alpha-1} f^{(n)}(\tau) d\tau, \quad n-1 < \alpha < n, \end{aligned}$$

where n is a nonnegative integer number; in fact $n = [\alpha] + 1$ while $[\alpha]$ indicates the integer part of $\alpha > 0$. For $\alpha \in \mathbb{N}$, the Caputo derivative operator is the usual derivative operator of integer order.

We list some properties of fractional integral and fractional derivative for $f \in C_\mu$, $\mu > 1$, and $\alpha, \beta > 0$, as follows:

$$\begin{aligned} {}_0I_x^0(f(x)) &= f(x), \\ {}_0I_x^\alpha(x^k) &= \frac{\Gamma(k+1)}{\Gamma(k+1+\alpha)} x^{k+\alpha}, \quad k \in \mathbb{N} \cup 0, x > 0, \\ {}_0D_x^\alpha {}_0I_x^\alpha(f(x)) &= f(x), \\ {}_0D_x^\alpha(x^k) &= \frac{\Gamma(k+1)}{\Gamma(k+1-\alpha)} x^{k-\alpha}, \quad k \in \mathbb{N} \cup 0, x > 0, k > [\alpha], \\ {}_0D_x^\alpha(C) &= 0, \quad C \in \mathbb{R}, \\ {}_0D_x^\alpha\left(\sum_{i=0}^n c_i f_i(x)\right) &= \sum_{i=0}^n c_i {}_0D_x^\alpha f_i(x), \end{aligned}$$

where $\{c_i\}_{i=0}^n$ are constants.

2.2 Some preliminaries in Bernstein polynomials

Definition 4. The Bernstein polynomials of degree n over the interval $[0, 1]$ are defined as follows:

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i},$$

for $i = 0, 1, 2, \dots, n$, where $\binom{n}{i} = \frac{n!}{i!(n-i)!}$.

If one uses the binomial expansion for $(1-t)^{n-i}$, then

$$B_i^n(t) = \sum_{k=0}^{n-i} \binom{n-i}{k} \binom{n}{i} (-1)^k t^{i+k}. \quad (3)$$

Definition 5. We define a Bernstein vector $\phi_n(t)$ as

$$\phi_n(t) = [B_0^n(t) \ B_1^n(t) \ B_2^n(t) \ \dots \ B_n^n(t)]^T;$$

the Bezier polynomial of degree n over the interval $[0, 1]$ is defined as follows:

$$P_n(t) = C\phi_n(t), \quad (4)$$

where

$$C = [c_0 \ c_1 \ c_2 \ \dots \ c_n], \quad (5)$$

is the vector of constant coefficients that we recall its entry as the control points.

Thus by (4),

$$P_n(t) = \sum_{i=0}^n c_i B_i^n(t).$$

Lemma 1. By using (3), we define

$$\phi_n(t) = \psi_n T_n(t),$$

where $T_n(t) = [1 \ t \ t^2 \ \dots \ t^n]^T$ and $\psi_n = \Psi^{n+1}$ is an $(n+1) \times (n+1)$ upper triangular matrix that can be expressed by

$$\Psi_{i+1,j+1}^{n+1} = \begin{cases} \frac{(-1)^{j-i} n!}{(n-j)!(j-i)!i!}, & i \leq j, \\ 0, & i > j, \end{cases} \quad (6)$$

where $i, j = 0, 1, \dots, n$.

Proof. See [2]. □

Theorem 1. Suppose that $f(x) \in C^m[0, 1]$; then for each $k \in \mathbb{N}$, $k \leq m$, and $k \leq n$, there exists an $(n+1) \times (n+1)$ matrix D such that

$$f^{(k)}(x) \simeq P_n^{(k)}(x) = \mathbf{C} D^k \phi_n(x),$$

where $\mathbf{C}$ is shown in (5) and $D = \psi_n \Lambda \mathcal{V}$, with

$$\Lambda_{i+1,j+1} = \begin{cases} i, & i = j + 1, \\ 0 & \text{otherwise,} \end{cases}$$

for $i = 0, \dots, n$ and $j = 0, \dots, n-1$, and $\mathcal{V}$ can be expressed by

$$\mathcal{V}_k = \psi_{n,k}^{-1}, \quad k = 1, 2, \dots, n,$$

where $\psi_{n,k}^{-1}$ is the k th row of ψ_n^{-1} .

Proof. For details, see [8]. $\square$

2.3 Function approximation

Definition 6. For the product of two Bernstein polynomials of degree $n, m \in \mathbb{N} \cup \{0\}$, we define

$${}_i^n B_j^m(x, t) = B_i^n(x) B_j^m(t),$$

where $t, x \in [0, 1]$.

Lemma 2. The set

$$Y = \text{Span} \{ {}_0^n B_0^m(x, t), {}_0^n B_1^m(x, t), \dots, {}_0^n B_m^m(x, t), {}_1^n B_0^m(x, t), \dots, {}_n^n B_m^m(x, t) \}$$

is a complete basis for the Hilbert space $L^2[\bar{\Omega}]$, where $\bar{\Omega} = [0, 1] \times [0, 1]$.

Proof. Since Y is a finite subset of $L^2[\bar{\Omega}]$, it is complete. For more details, see [12]. $\square$

Lemma 3. If

$$Y = \text{Span} \{ {}_0^n B_0^m(x, t), {}_0^n B_1^m(x, t), \dots, {}_0^n B_m^m(x, t), {}_1^n B_0^m(x, t), \dots, {}_n^n B_m^m(x, t) \}$$

and $u(x, t) \in L^2[\bar{\Omega}]$, then $\phi_n^T(x) \mathbf{U} \phi_m(t)$ is the best approximation of $u(x, t)$ out of Y , that is,

$$\begin{aligned} u(x, t) &\simeq u_{nm}(x, t) = \sum_{i=0}^n \sum_{j=0}^m u_{i,j} {}_i^n B_j^m(x, t) = \sum_{i=0}^n \sum_{j=0}^m u_{i,j} B_i^n(x) B_j^m(t) \\ &= \phi_n^T(x) \mathbf{U} \phi_m(t), \end{aligned}$$

where $\mathbf{U}$ is the $(n+1) \times (m+1)$ matrix such that

$$\mathbf{U} = Q_n^{-1} \langle \phi_n(x), \langle u(x, t), \phi_m(t) \rangle \rangle Q_m^{-1}$$

and $Q_n = \psi_n G_n \psi_n^T$,

$$G_n = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n+1} & \frac{1}{n+2} & \frac{1}{n+3} & \cdots & \frac{1}{2n+1} \end{bmatrix},$$

where ψ_n is defined in (6). Also the inner product $\langle \cdot, \cdot \rangle$ on the space $C[0, 1]$ is defined by

$$\langle f, g \rangle = \int_0^1 f(x)g(x)dx, \quad f(x), g(x) \in C[0, 1].$$

Proof. See [12]. $\square$

To compute Caputo fractional derivatives for an arbitrary function $f(x)$, we define an operational fractional derivative matrix of order $\alpha > 0$ by using Bernstein polynomials.

Theorem 2. Let $\phi_n(x)$ be a vector of Bernstein polynomials. Then for each $\alpha > 0$, we have

$${}_0D_x^\alpha \phi_n(x) = D^\alpha \phi_n(x),$$

while $D^\alpha = \psi_n K_n^\alpha H^\alpha$, where K_n^α is an $(n+1) \times (n+1)$ matrix, defined as follows:

$$K_n^\alpha(i+1, j+1) = \begin{cases} \frac{\Gamma(i+1)}{\Gamma(i-\alpha+1)} (B(p_i, q_i) - B(\frac{a}{x}; p_i, q_i)), & i \in \mathbb{N} \cup 0, i \geq [\alpha] \text{ or} \\ & i \notin \mathbb{N} \cup 0, i > [\alpha], i = j, \\ 0, & \text{otherwise,} \end{cases}$$

where the floor function $[\alpha]$ and the ceiling function $\lceil \alpha \rceil$ indicate the largest integer less than or equal to α and the smallest integer greater than or equal to α , respectively. Also, $B(p, q)$ and $B(x; p, q)$ are, respectively, the beta function and incomplete beta function, for $i, j = 0, 1, \dots, n$, $p_i = i - [\alpha]$, and $q_i = [\alpha] - \alpha + 1$. Also H^α is the following $(n+1) \times (n+1)$ matrix:

$$H^\alpha = [H^\alpha(1), H^\alpha(2), \dots, H^\alpha(n+1)],$$

where $H^\alpha(i+1) = \langle x^{i+1-\alpha}, \phi_n(x) \rangle A_n^{-1}$, $i = 0, 1, \dots, n$, and $A_n = \langle \phi_n(x), \phi_n(x) \rangle$. The matrix D^α is called the operational derivative matrix for the fractional Caputo derivative of order α .

Proof. see [10] $\square$

3 Approximate solution of FPDEs with proportional delays

In this section, we consider the FPDE (1)–(2). One may assume that the existence conditions for problem (1)–(2) are satisfied. We develop using Bernstein polynomials for solving FPDE (1)–(2). To achieve this aim, by using the least-squares method, we convert problem (1)–(2) to a mathematical programming problem. For this purpose, define the residual function $R(x, t)$ as

follows:

$$R(x, t) = {}_0D_t^\alpha u(x, t) - h(x, t, u(a_0x, b_0t), \frac{\partial u(a_1x, b_1t)}{\partial x}, \dots, \frac{\partial^{n_1} u(a_{n_1}x, b_{n_1}t)}{\partial x^{n_1}}). \quad (7)$$

It is clear that when we get the exact solution, then the residual function $R(x, t)$ is zero for $(x, t) \in (0, 1) \times (0, 1)$. To get the approximate solution of problem (1)–(2), we can follow Step 1 up to Step 6.

Step 1: Choose natural numbers n and m and approximate the solution of $u(x, t)$ by Bernstein polynomials.

$$u(x, t) \simeq u_{nm}(x, t) = \omega(x, t)\phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t) + \xi(x), \quad (8)$$

where $\omega(x, t)$ and $\xi(x)$ are given functions such that $u(x, t)$ satisfies the initial conditions (2). If one chooses $\omega(x, t) = t$ and $\xi(x) = s(x)$, then we have

$$u_{nm}(x, t) = t\phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t) + \xi(x).$$

But, by Lemma 3.5 in [2], the function $t\phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t)$ can be rewritten as follows:

$$\begin{aligned} t\phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t) &= \phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t)t = \phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t)(\mathbf{C}\phi_m(t))^\top \\ &= \phi_n^\top(x)\bar{\mathbf{U}}\phi_m(t)\phi_m^\top(t)\mathbf{C}^\top \simeq \phi_n^\top(x)\bar{\mathbf{U}}\mathbf{C}\phi_m(t) \\ &= \phi_n^\top(x)\mathbf{U}\phi_m(t). \end{aligned}$$

So,

$$u(x, t) \simeq u_{nm}(x, t) = \phi_n^\top(x)\mathbf{U}\phi_m(t) + \xi(x). \quad (9)$$

Step 2: According to the previous section, one can easily find that

$$\begin{aligned} {}_0D_t^\alpha u(x, t) &\simeq \phi_n^\top(x)\mathbf{U}D^\alpha\phi_m(t), \\ u(a_0x, b_0t) &\simeq \phi_n^\top(a_0x)\mathbf{U}\phi_m(b_0t) + \xi(a_0x), \\ \frac{\partial u(a_1x, b_1t)}{\partial x} &\simeq \phi_n^\top(a_1x)D^\top\mathbf{U}\phi_m(b_1t) + \frac{d\xi(a_1x)}{dx}, \\ &\vdots \\ \frac{\partial^{n_1} u(a_{n_1}x, b_{n_1}t)}{\partial x^{n_1}} &\simeq \phi_n^\top(a_{n_1}x)D^{n_1\top}\mathbf{U}\phi_m(b_{n_1}t) + \frac{d^{n_1}\xi(a_{n_1}x)}{dx^{n_1}}, \end{aligned}$$

and

$$u^{(k)}(x, 0) \simeq \phi_n^\top(x)D^{k\top}\mathbf{U}\phi_m(0) + \frac{d^k\xi(x)}{dx^k} = s^{(k)}(x), \quad k = 0, 1, \dots, q-1.$$

Step 3: We express the residual function (7) by the Bernstein polynomials as follows:

$$R(x, t) \simeq \phi_n^T(x) \mathbf{R} \phi_m(t) = \sum_{i=0}^n \sum_{j=0}^m r_{i,j} B_i^n(x) B_j^m(t),$$

where the control points $r_{i,j}$ are functions of coefficients $u_{i,j}$, $i = 0, 1, 2, \dots, n$, $j = 0, 1, 2, \dots, m$. In other word,

$$r_{i,j} = r_{i,j}(u_{0,0}, u_{0,1}, u_{0,2}, \dots, u_{0,m}, u_{1,0}, \dots, u_{n,m}).$$

Step 4: Create a cost function $F = \sum_{i=0}^n \sum_{j=0}^m r_{i,j}^2$. It is obvious that the square function F is a function of $u_{0,0}, u_{0,1}, u_{0,2}, \dots, u_{0,m}, u_{1,0}, \dots, u_{n,m}$.

Step 5: Construct a mathematical programming problem as follows:

$$\text{minimize } F = \sum_{i=0}^n \sum_{j=0}^m r_{i,j}^2(u_{0,0}, u_{0,1}, \dots, u_{0,m}, u_{1,0}, \dots, u_{n,m}), \quad (10)$$

s.t.

$$\begin{aligned} & \sum_{i=0}^n \sum_{j=0}^m \left(\sum_{l=0}^n d_{i,l}^k u_{l,j} \right) B_i^n(x) B_j^m(0) + \xi^{(k)}(x) = s^{(k)}(x), \\ & k = 0, 1, 2, \dots, q-1, \\ & 0 \leq x \leq 1, \\ & 0 \leq t \leq 1. \end{aligned} \quad (11)$$

Step 6: Optimization problem (10)–(11) can be solved by some useful computational softwares such as MATLAB. By solving this optimization problem, one can find the values of control points $u_{i,j}$ $i = 0, 1, 2, \dots, n$, $j = 0, 1, 2, \dots, m$. Then the approximate solution of $u(x, t)$ could be found by using these control points in (8). We need to mention that we have solved optimization problem (10)–(11) by using discrete points $x_p \in (0, 1)$ and $t_h \in (0, 1)$.

4 Convergence analysis

In this section, we show the convergence of the approximate solution that has been found by Bernstein polynomials for the FPDE (1)–(2) with proportional delays. In this way, without loss of generality, we suppose that $n_1 = 1$, that $q = 1$, and that $0 < \alpha \leq 1$ for the FPDE (1)–(2). Consider the following

problem:

$$\begin{aligned} L \left[x, t, u(a_0x, b_0t), \frac{\partial u(a_1x, b_1t)}{\partial x}, {}_0D_t^\alpha u(x, t) \right] \\ = {}_0D_t^\alpha u(x, t) - h(x, t, u(a_0x, b_0t), \frac{\partial u(a_1x, b_1t)}{\partial x}), \end{aligned} \quad (12)$$

with initial conditions

$$u(x, 0) = s(x), \quad (13)$$

where $(x, t) \in \bar{\Omega} = [0, 1] \times [0, 1]$, the constant proportional delays a_0, a_1, b_0 , and b_1 are in the interval $[0, 1]$, also $s(x)$ is a given function and L is the partial differential operator.

Consider $(C(\bar{\Omega}), \|\cdot\|)$ as the normed space of all differentiable continuous functions, where $\|\cdot\|$ is defined as follows:

$$\|f\| = \|f\|_\infty + \left\| \frac{\partial f}{\partial t} \right\|_\infty + \left\| \frac{\partial f}{\partial x} \right\|_\infty, \quad f \in C(\bar{\Omega}).$$

Lemma 4. If $\bar{\Omega} = [0, 1] \times [0, 1]$, then the defined function $L : C(\bar{\Omega}) \rightarrow \mathbb{R}$ in (12) is uniformly continuous on $(C(\bar{\Omega}), \|\cdot\|)$.

Proof. See [16]. □

Now, we prove the convergence of the approximate solution for the FPDE (1)–(2) when the degree of the Bernstein polynomials increase. For this intention, we remark the following lemma, that is an obvious result of the Stone–Weierstrass theorem.

Lemma 5. If $\bar{u}(x, t)$ is a continuous real-valued function defined on $\bar{\Omega}$ and $\varepsilon > 0$, then there exists a sequence of Bernstein polynomials $\{u_{nm}(x, t)\}_{n,m \in \mathbb{N}}$ and $M \in \mathbb{N}$ such that

$$\|\bar{u}(x, t) - u_{nm}(x, t)\| < \varepsilon, \quad (x, t) \in \bar{\Omega}, \quad n, m \geq M.$$

Proof. See [24]. □

Theorem 3. Consider the functional L on $C(\bar{\Omega})$. If η_{nm} is the minimum of the functional L on $C(\bar{\Omega}) \cap Y$, then

$$\eta_{nm} \rightarrow 0$$

where L and Y are defined in (12) and Lemma 2, respectively.

Proof. Let $u^*(x, t) \in C(\bar{\Omega})$ and $\varepsilon > 0$ be such that $L[u^*(x, t)] < \varepsilon$. By the minimum properties, it is possible to find such a function $u^*(x, t)$. Considering Lemma 4, if $u(x, t) \in C(\bar{\Omega})$ and $\|u - u^*\| < \delta$, then

$$|L[u] - L[u^*]| < \varepsilon.$$

We use Lemma 5, for suitable values of n and m , there exist $u_{nm}(x, t) \in C(\bar{\Omega}) \cap Y$ such that $\|u_{nm} - u^*\| < \delta$. If we show $L[u_{nm}]$ by η_{nm} , then we have

$$\eta_{nm} = |L[u_{nm}] - L[u^*] + L[u^*]| \leq |L[u_{nm}] - L[u^*]| + |L[u^*]| \leq 2\varepsilon.$$

Because ε is arbitrary, this completes the proof. Note that the minimum value of the functional L in (12) on $C(\bar{\Omega})$ is zero. $\square$

5 Numerical examples

In this section, we give two numerical examples and apply the method presented in Section 3 for solving them. These test problems are solved by using powerful *MATLAB 2018b* software on an *Intel Core i7-7500U*. These test problems demonstrate the validity and efficiency of this technique.

Example 1. Consider the following fractional partial generalized Burgers equation with proportional delay (see [20]):

$${}_0D_t^\alpha u(x, t) = \frac{\partial^2 u(x, \frac{t}{2})}{\partial x^2} u(x, \frac{t}{2}) - u(x, t),$$

where $x, t \in [0, 1]$, $0 < \alpha < 1$, and the initial condition is

$$u(x, 0) = x^2.$$

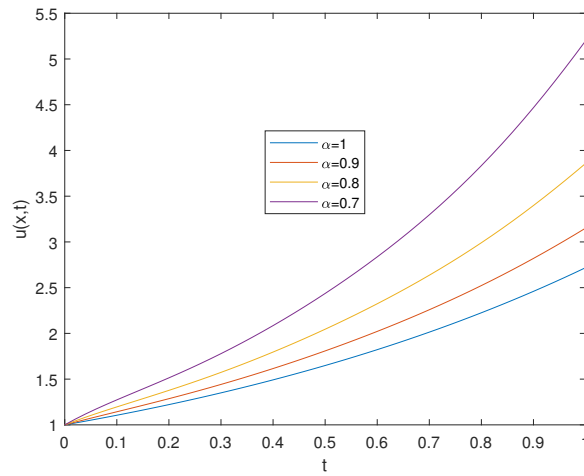
The exact solution of this example is $u(x, t) = x^2 e^t$ when $\alpha = 1$. If one assume that $\bar{u}(x, t)$ is the approximate solution, then the error between exact and approximate solution can be shown by

$$|u(x, t) - \bar{u}(x, t)|, \quad (x, t) \in (0, 1) \times (0, 1).$$

The comparison of the absolute errors between exact and approximate solution by employing HPM [20], AVIM [22], and the proposed method for $n = 2$ and $m = 5$ is shown in Table 1. Figure 1 shows the graph of the approximate solution of $u(x, t)$ for different values of α when $x = 1$.

Table 1: The absolute value of error for $\alpha = 1$ in Example 1.

x	t	presented method	HPM[20]	AVIM[22]
0.25	0.25	$8.7365E-08$	$5.30E-07$	$2.1974E-08$
0.25	0.5	$5.6597E-08$	$1.7735E-05$	$1.4596E-06$
0.25	0.75	$3.6812E-08$	$1.4087E-04$	$1.7274E-05$
0.25	1	$1.7890E-07$	$6.2178E-04$	$1.0095E-04$
0.5	0.25	$1.0598E-07$	$2.1230E-06$	$8.7896E-08$
0.5	0.5	$1.4489E-07$	$7.0943E-05$	$5.8385E-06$
0.5	0.75	$6.2958E-07$	$5.6348E-04$	$6.9096E-05$
0.5	1	$1.3441E-06$	$2.4871E-03$	$4.0379E-04$
0.75	0.25	$2.4535E-07$	$4.7760E-06$	$1.9777E-07$
0.75	0.5	$9.9618E-07$	$1.5962E-04$	$1.3137E-05$
0.75	0.75	$2.2682E-06$	$1.2678E-03$	$1.5547E-04$
0.75	1	$4.1328E-06$	$5.5960E-03$	$9.0853E-04$

Figure 1: Approximated graphs of $u(x, t)$ in $\alpha = 0.7, 0.8, 0.9, 1$ and $x = 1$ for Example 1.

Example 2. In this example, the following fractional generalized Burgers equation with proportional delay is considered (see [20]):

$${}_0D_t^\alpha u(x, t) = \frac{\partial^2 u(x, t)}{\partial x^2} + \frac{\partial u(x, \frac{t}{2})}{\partial x} u(\frac{x}{2}, \frac{t}{2}) + \frac{1}{2} u(x, t),$$

such that

$$u(x, 0) = x,$$

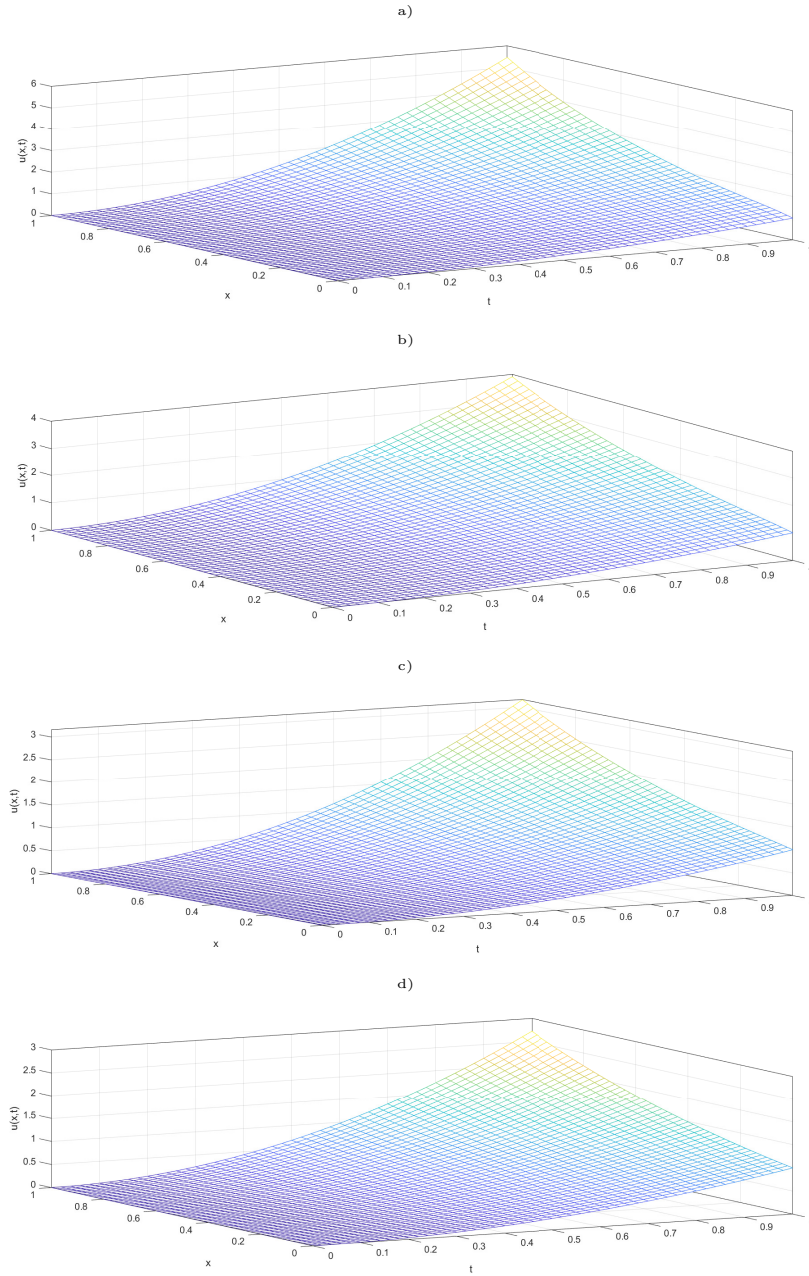


Figure 2: Approximated graphs of $u(x, t)$ in Example 1 for a) $\alpha = 0.7$; b) $\alpha = 0.8$; c) $\alpha = 0.9$; d) $\alpha = 1$;

where $x, t \in [0, 1]$ and $0 < \alpha < 1$.

The exact solution of this example is $u(x, t) = xe^t$ when $\alpha = 1$. The comparison of the absolute errors between exact and approximate solution by employing NTDM [21], HPM [20], AVIM [22], and the proposed method for $n = 2$ and $m = 6$ is shown in Table 2. Figure 3 shows the graph of the approximate solution of $u(x, t)$ for different values of α when $x = 1$.

Table 2: The absolute value of error for $\alpha = 1$ in Example 2.

x	t	presented method	NTDM [21]	HPM [20]	AVIM[22]
0.25	0.25	$1.4976E-06$	$2.1224E-06$	$2.1230E-06$	$8.7896E-08$
0.25	0.5	$7.1515E-06$	$7.0943E-05$	$7.0943E-05$	$5.8386E-06$
0.25	0.75	$1.7332E-05$	$5.6348E-04$	$5.6348E-04$	$6.9096E-05$
0.25	1	$3.4298E-05$	$2.4871E-03$	$2.4871E-03$	$4.0379E-04$
0.5	0.25	$1.0884E-06$	$4.2550E-06$	$4.2450E-06$	$1.7579E-07$
0.5	0.5	$6.5797E-06$	$1.4189E-04$	$1.4189E-04$	$1.1677E-05$
0.5	0.75	$1.6607E-05$	$1.1270E-03$	$1.1270E-03$	$1.3819E-04$
0.5	1	$3.3493E-05$	$4.9742E-03$	$4.9743E-03$	$8.0758E-04$
0.75	0.25	$1.6021E-06$	$6.3675E-06$	$6.3670E-06$	$2.6369E-07$
0.75	0.5	$7.5613E-06$	$2.1283E-04$	$2.1283E-04$	$1.7516E-05$
0.75	0.75	$1.8304E-05$	$1.6904E-03$	$1.6905E-03$	$2.0729E-04$
0.75	1	$3.6085E-05$	$7.4614E-03$	$7.4614E-03$	$1.2114E-03$

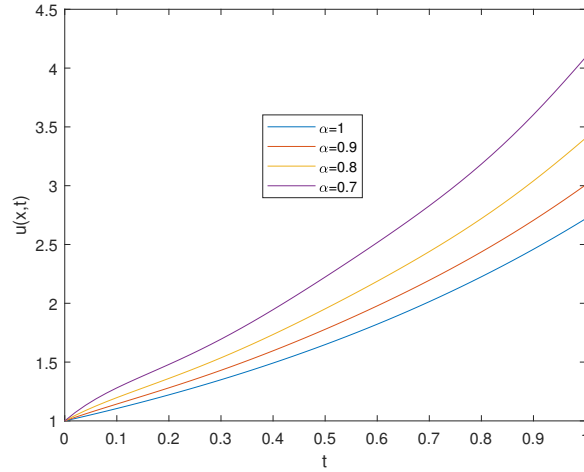


Figure 3: Approximated graphs of $u(x, t)$ in $\alpha = 0.7, 0.8, 0.9, 1$ and $x = 1$ for Example 2.

6 Conclusion

The aim of this paper was to solve fractional partial differential equations with multiple proportional delays by using Bernstein polynomials. To reach this goal, we generated some operational matrices based on Bernstein polynomials for fractional-order derivative in the Caputo sense. Operational matrices, together with the least-squares method, were used in an algorithm for solving these kinds of problems. The steps of the algorithm were easy to implement. The convergence of the method was investigated. Two numerical examples were given to illustrate that this proposed method is accurate and precise.

Acknowledgments

We thank the anonymous reviewer for helpful comments, which led to definite improvements in the manuscript.

References

1. Aghdam, Y.E., Mesgrani, H., Javidi, M. and Nikan, O. *A computational approach for the space-time fractional advection-diffusion equation arising in contaminant transport through porous media*, Eng. Comput. (2020), 1–13.
2. Alipour, M., Rostamy, D. and Baleanu, D. *Solving multi-dimensional fractional optimal control problems with inequality constraint by Bernstein polynomials operational matrices*, J. Vib. Control, 19 (2012), 2523–2540.
3. Chakrabarti, A. and Martha, S.C. *Approximate solutions of Fredholm integral equations of the second kind*, Appl. Math. Comput. 211 (2009), 459–466.
4. Elagan, S.K., Sayed, M. and Higazy, M. *An analytical study on fractional partial differential equations by Laplace transform operator method*, Int. J. Appl. Eng. Res. 13(1) (2018), 545–549.
5. Ghomanjani, F. and Khorram, E. *Approximate solution for quadratic Riccati differential equation*, J. Taibah Univ. Sci. 11(2) (2017), 246–250.
6. Hassani, H., Tenreiro Machado, J.A., Avazzadeh, Z. and Naraghirad, E. *Generalized shifted Chebyshev polynomials: solving a general class of nonlinear variable order fractional PDE*, Commun. nonlinear Sci. 85 (2020), 1–16.

7. Javadi, S., Jani, M. and Babolian, E. *A numerical scheme for space-time fractional advection-dispersion equation*, Int. J. Nonlinear Anal. Appl. 7(2) (2016), 331–343.
8. Karimi, K. Bahadorimehr, A. and Mansoorzadeh, S. *Numerical solution of nonlocal parabolic partial differential equation via Bernstein polynomial method*, J. Math. 48 (2016), 47–53.
9. Kazem, S. *An integral operational matrix based on Jacobi polynomials for solving fractional-order differential equations*, Appl. Math. Model. 37(3) (2013), 1126–1136.
10. Ketabdari, A., Farahi, M.H. and Effati, S. *A novel approximate method for solving 2D fractional optimal control problems using generalized fractional order of Bernstein functions*, Manuscript in press.
11. Khan, H., Alipour, M., Jafari, H. and Khan, R.A. *Approximate analytical solution of a coupled system of fractional partial differential equations by Bernstein polynomials*, Int. J. Appl. Comput. Math. 2 (2016), 85–96.
12. Kreyszig, E. *Introduction to functional analysis with applications*, New York: Wiley, 1978.
13. Li, M., Li, G., Li, Z. and Jia, X. *Determination of Time-Dependent Coefficients in Time-Fractional Diffusion Equations by Variational Iteration Method*, J. Math. Res. 12(1) (2020), 1–74.
14. Li, W., Bai, L., Chen, Y., Dos Santos, S. and Li, B. *Solution of linear fractional partial differential equations based on the operator matrix of fractional Bernstein polynomials and error correction*, Int. J. Innov. Comput. I. 14 (2018), 211–226.
15. Momani, S. and Odibat, Z. *Homotopy perturbation method for nonlinear partial differential equations of fractional order*, Phys. Lett. A, 365 (2007), 345–350.
16. Nemati, A. and Yousefi, S.A. *A numerical scheme for solving two-dimensional fractional optimal control problems by the Ritz method combined with fractional operational matrix*, IMA J. Math. Control Inform. 34(4) (2017), 1079–1097.
17. Rostamy, D., Karimi, K. and Mohamadi, E. *Solving fractional partial differential equations by an efficient new basis*, Int. J. Appl. Math. Comput. 5 (2013), 6–21.
18. Safaie, E. and Farahi, M.H. *An approximation method for numerical solution of multi-dimensional feedback delay fractional optimal control problems by Bernstein polynomials*, Iran J. Numerical Analysis and Optimization, 4(1) (2014), 77–94.

19. Safdari, H., Mesgarani, H., Javidi, M. and Esmaealzade Aghdam, Y. *Convergence analysis of the space fractional-order diffusion equation based on the compact finite difference scheme*, J. Comput. Appl. Math. 39(2) (2020), 1–15.
20. Sakar, M.G., Uludag, F. and Erdogan, F. *Numerical solution of time-fractional nonlinear PDEs with proportional delays by homotopy perturbation method*, Appl. Math. Model. 40 (2016), 6639–6649.
21. Shah, R., Khan, H., Kumam, P., Arif, M. and Baleanu, D. *Natural transform decomposition method for solving fractional-order partial differential equations with proportional delay*, Mathematics, 7 (2019), 1–14.
22. Singh, B.K. and Kumar, P. *Fractional variational iteration method for solving fractional partial differential equations with proportional delay*, Int. J. Differ. Equ. 4 (2017), 1–11.
23. Sripacharasakullert, P., Sawangtong, W. and Sawangtong, P. *An approximate analytical solution of the fractional multi-dimensional Burgers equation by the homotopy perturbation method*, Adv. Differ. Equ. 252 (2019), 1–12.
24. Stone, M. *The generalized weierstrass approximation theorem*, Math. Mag. 21 (1948), 237–254.
25. Thabet, H. and Kendre, S. *New modification of Adomian decomposition method for solving a system of nonlinear fractional partial differential equations*, Int. J. Adv. Appl. Math. Mech. 6(3) (2019), 1–13.
26. Turut, V. and Güzel, N. *On solving partial differential equations of fractional order by using the variational iteration method and multivariate pad approximations*, Eur. j. pure appl. math. 6(2) (2013), 147–171.
27. Wu, J. *Theory and applications of partial functional differential equations*, New York: Springer, 1996.
28. Wu, Q. and Zeng, X. *Jacobi collocation methods for solving generalized space-fractional Burgers equations*, Commun. Appl. Math. Comput. 2 (2020), 305–318.
29. Yaslan, H.. *Legendre collocation method for the nonlinear space-time fractional partial differential equations*, Iran J. Sci. Technol. Trans. Sci. 44 (2020), 239–249.
30. Zhang, Y. *A finite difference method for fractional partial differential equation*, Appl. Math. Comput. 215 (2009), 524–529.
31. Zheng, J., Sederberg, T.W. and Johnson, R.W. *Least squares methods for solving differential equations using Bezier control points*, Appl. Numer. Math. 48 (2004), 237–252.



Nonlinear vibrations of functionally graded cylindrical shell by using numerical analysis in the wave propagation method

M.R. Assari*, A. Kavooosi Nejad and Sh. Amirshirzad

Abstract

Functionally graded materials (FGMs) are materials that show different properties in different areas due to the gradual change of chemical composition, distribution, and orientation, or the size of the reinforcing phase in one or more dimensions. In this paper, the free vibrations of a thin cylindrical shell made of FGM is investigated. In order to investigate this problem, the first-order shear theory is used, by using relations related to the propagation of waves and fluid-structure interaction. Also, due to the rotational inertia of first-order shear deformation and the fluid velocity potential, dynamic equation of functionally graded cylinder shell, containing current is obtained. Convergence of the solutions obtained from this method in different modes of boundary conditions as well as different geometric characteristics for the submerged cylinder and results of other studies and articles is showed. Also the effects of different parameters on the FGM cylindrical shell frequencies for the classical boundary conditions (compositions of simple, clamped, and free boundary conditions) are investigated against the ratio of length to the radius and the ratio of thickness to radius for different values of exponential power (exponential order) of FGM material. The results show that if the more density of the fluid in which the cylinder is submerged is lower, then the frequency values will be higher. Also, by examining the different fluid velocities, it can be seen that the effect of thickness change so that increasing thickness causes the increase of effect of speed on the natural frequency reduction, especially in higher modes.

*Corresponding author

Received 27 June 2019; revised 12 December 2019; accepted 16 April 2020

Mohammad Reza Assari

Department of Mechanical Engineering , Jundi-Shapur University of Technology, Dezful, Iran e-mail: mr_assari@yahoo.com

Ali Kavooosi Nejad

Department of Mechanical Engineering , Jundi-Shapur University of Technology, Dezful, Iran e-mail: Alibox2006@yahoo.com

Shahryar Amirshirzad

Department of Mechanical Engineering , Jundi-Shapur University of Technology, Dezful, Iran e-mail: shahryaramirshirzad@gmail.com

AMS(2010): 76L05.

Keywords: Functionally graded materials; Natural frequencies of cylindrical shell; First-order shear deformation theory; Fluid-structure interaction; Nonlinear vibrations; Propagation method.

1 Introduction

In recent years, with the development of high-powered engines for aerospace industries, turbines, reactors, and other machines, there is a need for materials with higher thermal resistance and more resistant dynamically. Functionally graded materials (FGMs) are in fact composite materials in which the composition of materials or microscopic structure is changed in such a way that the mechanical and thermal properties of the structure made of these materials are ideal for applications that include high thermal gradients from thermal structures in advanced aircraft and aerospace engines and computer circuit boards. According to the many variables that affect the design of FGMs, the full exploitation of the potential of FGMs requires the development of appropriate modeling methods for their response to the combination of mechanical and thermal loads. Emergence and entry of FGMs, by defining the various modes of the existing compounds of human-made, the possibility of the deliberate construction is provided according to these materials. This possibility was proposed as a concept by Bohr and Dowz in 1972, and the exploitation of these materials in individual efforts in the 1950s, 60s, and 70s, and the early 1980s was by many researchers and prominently in the United States. Then Japanese require these materials to grow and progress in space research to focus on the spaceship project that the result was that many of the various hard and precise requirements were made for increasing the temperature of the components of hybrid and gradually microstructural structures; see [25, 19, 4, 11].

Robinson and Palmer [23] carried out a modal analysis of a rectangular sheet suspended on fluid. They obtained the response for a time-harmonic point load, but their analysis can be cited for a limited number of initial frequencies. Kwak [18] examined the free vibration of a rectangular sheet suspended on an infinite fluid. The basis of this analysis is based on the Rayleigh–Ritz method and the Green’s function. Haddara and Cao [10] obtained an approximate expansion for the modal analysis of a rectangular sheet of a curved side, horizontally immersed in the fluid, empirically, and analytically. Zhou and Cheung [36] studied the vibrations of a rectangular sheet that is in contact with the fluid in one side, using the Rayleigh–Ritz method. The fluid is in contact with the sheet inside a semi-infinite rigid reservoir. Liang et al. [20] proposed a simple method for obtaining natural frequencies and the shape of the immersed sheet of a clamped side based on

empirical formulas. Yadykin et al. [32] analyzed an immersed rectangular sheet in fluid with ratio of different aspects. The sheets analyzed in their work are with a clamped edge and other edges free. Jeong et al. [15] presented an analytical method for estimating the frequencies of two specific sheets with finite fluid. They used polynomial functions for analysis and satisfied the boundary conditions of the fluid with finite Fourier extension, by expanding the fluid velocity potential. Tarjoman and Carlo [6] studied the sheets of a curved side immersed in fluid and studied the effect of the length-to-width ratio. Zhou and Liu [37] presented a three dimensional method method for dynamically analyzing a flexible rectangular reservoir filled with fluid, using a combination of Rayleigh–Ritz methods and Galerkin’s method. FGMs are heterogeneous microstructure materials whose mechanical properties change smoothly and continuously from one surface to another.

The common type is the combination of ceramic and metal. These materials are obtained by mixing ceramic and metal powder. The advantage of using these materials is that they are able to withstand extreme temperatures and temperature differences, and they are extremely corrosion-resistant and have high resistance to fracture. Nowadays, these materials are used for structures that are resistant to high temperatures. This type of materials is used because of its specific features in thermal shields of rockets, chemical tanks, and abrasive environments. Considering the importance of FGMs in the industries, many researchers investigate the dynamic behavior of this type of material. Hosseini-Hashemi et al. [13] investigated and analyzed the free vibration of a rectangular sheet of FGM with a relatively thick based on the first-order shear deformation theory. In their study, for six modes of combining different boundary conditions of a sheet with two simple parallel edges and other edges, a combination of simple, free, and clamped boundary conditions, the natural frequency of the sheet is obtained by using a precise solution. One of the benefits of that research is the high accuracy of the proposed method. Talha and Sinagh [31] studied the vibration and static analysis of FGM sheets using the third-order shear deformation theory, but with the difference that in this theory, little corrections in the transverse displacement of these plates were done by the limited element method. Zhao et al. [35] provided an analysis for the free vibration of the plates of functionally material. They also used the first-order shear deformation theory to calculate shear strain and inertial rotation. The special equations were converted to energy functions using the Ritz method and then solved. Hosseini-Hashemi et al. [12] presented a closed-loop solution based on the free vibration analysis of thick sheets of FGMs based on the third-order shear deformation theory. The boundary conditions used are two simple parallel supports. Khorshidi [16] investigated the effect of the hydrostatic pressure of vibration of a rectangular coupler sheet with a fluid. Ho and Zheng studied the two-branch phenomenon in symmetric nonlinear vibrations of functionally graded circular sheet, considering the effects of temperature and harmonic transverse force. Khorshidi and Farhadi [17] studied the free vibration of rectangular

composite sheet in contact with limited fluid. Abolghasemi et al. [1] analyzed the buckling of rectangular sheets under uneven plate load based on the first-order shear deformation theory and used the Galerkin method for solving equations of stability for a sheet with simple boundary conditions. Ghasemi et al. [8] studied buckling behavior of conical reinforced composite shells under axial load and calculated them by using the Ritz method. Ebrahimi et al. [5] examined the vibrations of FGM beams in thermal loading. Guo and Zhang [9] studied on static bending, elastic buckling, and free and forced vibrations, and composite-reinforced fiber structures have been investigated. More recently, Song, Kitipornchai, and Yang [30] examined polymer nanoparticle sheets on the free and compulsory vibrations that operate in a categorized manner, in which GPLs are nonuniform and dispersed in matrices. Mirzaei and Kiani [21] examined an iso-geometric formulation for thermal analysis of shaking GRC layers with different compositions in different boundary conditions. They used the assumptions of Shen and his colleagues [26, 29, 27, 28, 33]. Benchouaf and Boutyour [2] studied on the coupled nonlinear static, and dynamic problems were transformed into a sequence of linear ones solved by a finite-element method. Benferhat et al. [3] studied on the effect of foundation stiffness parameters presented for thick to thin plates and for various values of the gradient index, aspect, and the side-to-thickness ratio.

In this paper, using the first-order shear deformation theory, the vibration of sheets made of FGMs in contact with fluid is studied. In order to examine the displacements and the vibrational behavior of the sheet, the boundary conditions are considered differently. The results are obtained by using the Comsol software for FSI, and in numerical results, different geometric conditions, applying fluid to the sheet at various heights, applying different velocities, and obtaining the natural frequencies with checking the different boundary conditions on the natural frequency of the sheet in contact with the fluid with the parametric sweep tool, the effect of volumetric power factor parameters has been investigated and analyzed for several important modes, including the volume ratio and fluid height. To verify the accuracy of the results, numerical results are compared with the results available in references.

2 Statement of the problem

The first step in this research is the modeling of a cylindrical shell consisting of FGM. For this modeling, in the first part, the properties of the materials studied should be considered based on the theories related to FGMs. For this purpose, we consider a cylinder from FGMs with uniform thickness with length L and thickness H and width R that this sheet will consist of ceramic and metal. In all equations, there is the assumption of linear behavior of

materials, displacements, and strains, as well as the elastic properties of materials are in line with the thickness of the variable plate and according to the volume ratio rule. To solve it, using a finite element method, a cylindrical shell model was first drawn in the Comsol software, which is shown in Figure 1 of the cylindrical shell model.

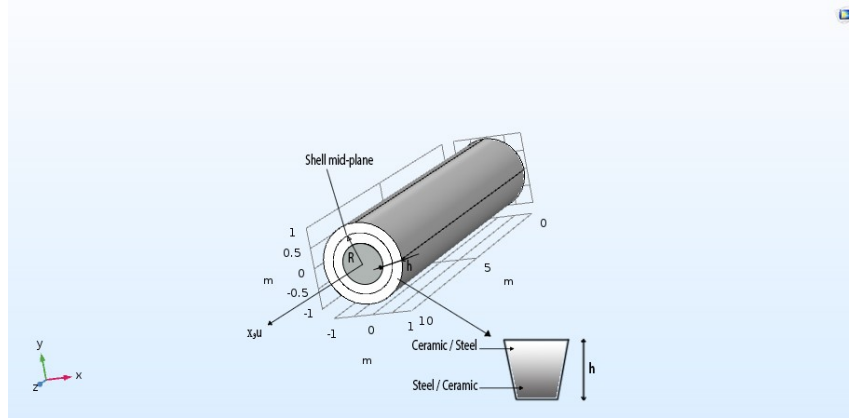


Figure 1: Functionally graded and German cylindrical geometry from cylindrical thickness

The next step in this research is the modeling of the effects of the fluid-structure interaction, which in this paper is modeled by using the coupling motion equations of fluid motion of this interaction. Nonslip and incompressible fluid and the incompressible fluid flow are isotropic and nonrotary. Limited software can be a solution to the condition of observing determinants in the analysis. Thin cylindrical shell is uniformly considered. Equations govern the dynamical behavior of cylinder from the FGM in the presence of fluid.

3 Formulation

The resultant material properties D of an FGM are functions of material properties and volume fractions of the constituent materials and can be modeled as follows (see [14]):

$$D = \sum_{k=1}^l D_k V_{fk},$$

where D_k and V_{fk} , respectively, represent the material properties and volume fraction of the constituent material k . The sum of volume fractions of all the constituent materials equals unity, that is,

$$\sum_{k=1}^l V_{fk} = 1.$$

For a cylindrical shell with uniform thickness h and with the middle surface taken as reference surface, the volume fraction is expressed as follows (see [14]):

$$V_f = \left(\frac{z}{h} + 0.5 \right)^g,$$

where g is the power law index and is taken to be $0 \leq g \leq \infty$. Since the paper study two materials for the construction of a cylindrical shell, three main mechanical quantities, Module Yang (E), Poisson ratio (ν), and density (ρ), can be extracted as follows (see [14]):

$$\begin{cases} E_o = (E_2 - E_1) \left(\frac{z}{h} + 0.5 \right)^g + E_1, \\ \nu_o = (\nu_2 - \nu_1) \left(\frac{z}{h} + 0.5 \right)^g + \nu_1, \\ \rho_o = (\rho_2 - \rho_1) \left(\frac{z}{h} + 0.5 \right)^g + \rho_1. \end{cases}$$

In this case, we know that for $z = -h/2$, the Young's modulus and the Poisson coefficient are, respectively, $\nu = \nu_1$ and $E = E_1$, and also the density is equal to $\rho = \rho_1$. When $z = h/2$, Yang modulus and Poisson coefficients are, respectively, $\nu = \nu_2$, and $E = E_2$ and also the density is equal to $\rho = \rho_2$, which represents the fact that, at the inner surface of the cylinder, the property of the constituent is corresponding with the first material, and at the outer surface of the cylinder, the mechanical properties are the properties of second material.

The thin sheet problem is a three-dimensional problem which becomes a two-dimensional problem, by considering the plate stress condition. For example, the stress and strain of components in the Z direction are negligible. This theory is used to calculate the effect of shear forces on thick shell frequencies. Using Love's thin shell theory, the equation of motion of this cylindrical shell is shown as follows (see [24]):

$$\begin{cases} \frac{\partial N_x}{\partial x} + \frac{1}{R} \frac{\partial N_{x\theta}}{\partial \theta} = \rho T \frac{\partial^2 u_x}{\partial t^2}, \\ \frac{\partial N_{x\theta}}{\partial x} + \frac{1}{R} \frac{\partial N_\theta}{\partial \theta} + \frac{1}{R} \frac{\partial M_{x\theta}}{\partial x} + \frac{1}{R^2} \frac{\partial M_\theta}{\partial \theta} = \rho T \frac{\partial^2 v}{\partial t^2}, \\ \frac{\partial^2 M_x}{\partial x^2} + \frac{2}{R} \frac{\partial^2 M_{x\theta}}{\partial x \partial \theta} + \frac{1}{R^2} \frac{\partial^2 M_\theta}{\partial \theta^2} - \frac{N_\theta}{R} = \rho T \frac{\partial^2 w}{\partial t^2}, \end{cases}$$

where N_x , N_θ , and $N_{x\theta}$ are the force components in the three directions and M_x , M_θ , and $M_{x\theta}$ are the components of the moment in the three main directions. Using the Love's theory for shells, the equations for strains and curves of the reference page will be as follows (see [14]):

$$\begin{aligned}
(N_x, N_\theta, N_{x\theta}) &= \int_{-h/2}^{h/2} (\sigma_x, \sigma_\theta, \sigma_{x\theta}) dz, \\
(M_x, M_\theta, M_{x\theta}) &= \int_{-h/2}^{h/2} (\sigma_x, \sigma_\theta, \sigma_{x\theta}) z dz, \\
\rho_o &= \int_{-h/2}^{h/2} \rho' dz.
\end{aligned}$$

There, the Q_{ij} 's are reduced stiffnesses for a thin cylindrical isotropic shell defined as follows (see [14]):

$$Q_{11} = \frac{E}{1 - \nu^2}, \quad Q_{22} = \frac{E}{1 - \nu^2}, \quad (1)$$

$$Q_{12} = \frac{\nu E}{1 - \nu^2}, \quad (2)$$

$$Q_{11} = \frac{E}{2(1 - \nu)}.$$

Here E and ν , respectively, represent the Young's coefficient and Poisson's coefficient. By placing the above equations in the matrix of force and momentum, we obtain

$$\begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{12} & L_{22} & L_{23} \\ L_{31} & L_{31} & L_{31} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \quad (3)$$

In the above equations, $L_{ij}(i, j = 1, 2, 3)$ represents the differential operator relative to x and θ . Since the velocity field for a fluid outside the shell only contains the effect of pressure and no flow outside the tube is observed, using the cylindrical coordinate system (x, φ, r) , the acoustic pressure equation satisfies the equation below [24]:

$$\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial P_a}{\partial r} \right) + \frac{1}{r^2} \frac{\partial^2 P_a}{\partial \varphi^2} + \frac{\partial^2 P_a}{\partial x^2} = \frac{1}{c^2} \frac{\partial^2 P_a}{\partial t^2}. \quad (4)$$

In the upper equation, P_a stands for the acoustic pressure and c indicates the speed of sound inside the water. The coordinates x and φ in the above equations are in accordance with the cylinder coordinates.

In this method, the modal displacement in shell equations can be written by using the wave propagations method and the longitudinal wave number (k_s), as well as by using the parameter of the wave number in the direction of the environment (n) as follows (see [24]):

$$\begin{cases} u(x, \theta, t) = A \cos(n\theta) \exp(i\omega t - ik_s x) , \\ v(x, \theta, t) = B \sin(n\theta) \exp(i\omega t - ik_s x) , \\ w(x, \theta, t) = C \cos(n\theta) \exp(i\omega t - ik_s x) . \end{cases}$$

The coefficients A , B , and C denote the wave amplitudes in the x , θ , and z directions, respectively. In these equations, n is the number of circumferential waves, and k_s is the axial wave number. It should be noted that by changing the boundary conditions, the number of longitudinal waves varies. Also, ω in the above equations shows the natural frequency for the cylindrical shell, and to analyze the waveforms generated, the peripheral wave number is used.

It is assumed that the fluid and shell are in contact at any moment, and the fluid has a flow with velocity U , which will also show the effects of the emission of waves in both fluids.

The associated form of the acoustic pressure field in the contained fluid, which satisfies the acoustic wave (2), can be expressed in the cylindrical coordinate system, associated with an axial wave number k_s , radial wave number k_r , and circumferential wave number n , and is given as follows (see [24]):

$$P = P_a \cos(n\theta) J_n(k_r r) \exp(i\omega t - ik_s x) ,$$

where J_n is the Bessel function of first kind with order n . For the fluid outside of the surface and the cylindrical shell, we have (see [24])

$$-\left\{ \frac{1}{i\omega\rho_f} \right\} \left(\frac{\partial P_a}{\partial r} \right) \Big|_{r=R} = \left(\frac{\partial w}{\partial t} \right) \Big|_{r=R} . \quad (5)$$

Given equation (4) for acoustic pressure, P_a will be obtained by the following equation (see [24]):

$$P_a = \left[\rho_f \omega^2 / k_r H_n'^{(2)}(Rk_r) \right] , \quad (6)$$

where ρ_f is the density of the contained fluid and the prime on the H_n denotes differentiation with respect to the argument Rk_r .

Equation (6) represents the amplitude of the acoustic pressure on the shell surface from the fluid outside of the shell and, in other words, determines the effect of the shell's submergence in the fluid.

Using equation (3) and considering the effects of external fluid, (3) could be rewritten as the equation of vibration of the shell as follows (see [22]):

$$\begin{bmatrix} L_{11} & L_{12} & L_{13} \\ L_{12} & L_{22} & L_{23} \\ L_{31} & L_{31} & L_{31} + FL \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} .$$

The FL in the equation above shows the effects of the acoustic pressure due to the submergence of the shell in the fluid. To calculate FL , we have

$$FL_o = -\frac{\rho_f \omega^2}{K_r} \frac{H_n^{(2)}(RK_r)}{H_n'^{(2)}(RK_r)}. \quad (7)$$

In other words, if FL is equal to zero, the equations become vibration equations of a free shell, and in general, the special values of equation (7) represent the natural frequencies of the system.

For the out-tube fluid, by applying the wave propagation method, the equation of wave propagation in the fluid in the tube by using the wave propagation equation for the distribution of pressure is obtained as follows:

$$FL_i = \frac{\rho_f}{K_r} \frac{J_n(RK_r)}{J_n'(RK_r)} (\omega^2 - U^2 k_s^2). \quad (8)$$

Using (5), (6) can be written as

$$\begin{aligned} FL &= FL_o + FL_i \\ &= \frac{\rho_f}{k_r} \left(-\frac{H_n^{(2)}(RK_r)}{H_n'^{(2)}(RK_r)} \omega^2 + \frac{J_n(RK_r)}{J_n'(RK_r)} \omega^2 - U^2 k_s^2 \right). \end{aligned}$$

By (1), we have

$$\begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{12} & T_{22} & T_{23} \\ T_{31} & T_{31} & T_{33} + FL \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} o \\ o \\ o \end{bmatrix},$$

where T_{ij} ($i, j = 1, 2, 3$) are coefficients of the stiffness matrix and the 3×3 matrix on the right hand side of (6) represents the mass matrix depending on shell parameters and the type of boundary conditions specified at the ends of a cylindrical shell and are given as (see [24])

$$\begin{aligned} T_{11} &= \left[(K_s^2) A_{11} + \frac{n^2}{R^2} A_{66} \right] - \rho_o h \omega^2, \\ T_{12} &= \left[-n k_s \left(\frac{A_{11} + A_{66}}{R} + \frac{B_{11} + 2B_{66}}{R^2} \right) \right], \\ T_{13} &= \left[i k_s \left(\frac{A_{12}}{R} + B_{11} k_s^2 + n^2 \frac{B_{12} + 2B_{66}}{R^2} \right) \right], \\ T_{22} &= \left[n^2 \left(\frac{A_{22}}{R^2} + \frac{2B_{22}}{R^3} + \frac{C_{22}}{R^4} \right) + k_s^2 \left(A_{66} + \frac{3B_{66}}{R} + \frac{4C_{66}}{R^2} \right) \right] - \rho_o h \omega^2, \\ T_{23} &= \left[-i n \left(\frac{A_{22}}{R^2} + \frac{B_{22}}{R^3} + n^2 \left(\frac{B_{22}}{R^3} + \frac{C_{22}}{R^4} \right) + k_s^2 \left(\frac{B_{12} + 2B_{66}}{R^2} + \frac{C_{12} + 4C_{66}}{R^2} \right) \right) \right], \\ T_{33} &= \left[-\left(\frac{A_{22}}{R^2} + \frac{2B_{12}}{R} k_s^2 + 2n^2 \frac{B_{22}}{R^3} + C_{11} k_s^4 + 2n^2 k_s^2 \left(\frac{C_{12} + 2C_{22}}{R^2} \right) + n^4 \frac{C_{22}}{R^4} \right) \right] - \rho_o h \omega^2. \end{aligned}$$

By solving the special value problem for equation (8), the natural frequency of vibration of a thin cylindrical shell can be obtained, by changing n and s , the various vibrational modes of the shell can be obtained.

4 Result and discussion

In this paper, by using two aluminum and ceramic materials (Al₂O₃), the mechanical properties of each one are given in Table 1, it is assumed that aluminum will be as an outer coating and ceramic as an inner coating of the sheet that is involved with the fluid.

Table 1: Specification of FGM

Type of material	Density	Elasticity module	Poisson coefficient
Al	2702	70	0.3
Al ₂ O ₃	3800	393	0.3

To solve it, using finite element solving method, a cylindrical shell model was first drawn in the software, and then, the problem of structure and fluid interaction was used to consider the effects of the fluid outside and inside the shell. Since in solving a finite element, due to some factors, the response depends on the size of the mesh, it must be ensured that the mesh size is sufficiently small and the dimensions of the mesh have no effect on the solutions. For this purpose, using different mesh sizes, the element solving has been limited. In Figure 2, to show the effect of mesh size on the results of the finite element method for the first frequency, are presented.

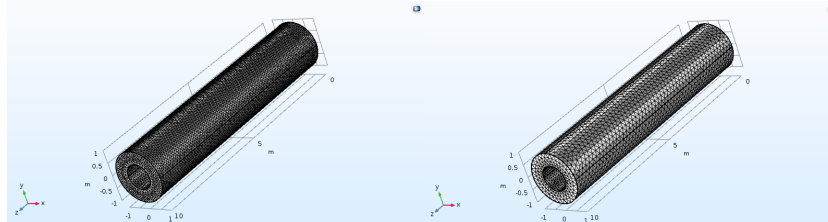


Figure 2: Mesh of cylindrical shell

Because in solving a finite element, due to some factors, the response depends on the size of the mesh, it must be ensured that there is no independence of the mesh network and the dimensions of the mesh do not affect the frequencies. For this purpose, using different sizes of mesh several times, the finite element solving has been performed and the results are presented in Figure 3 for the first frequency.

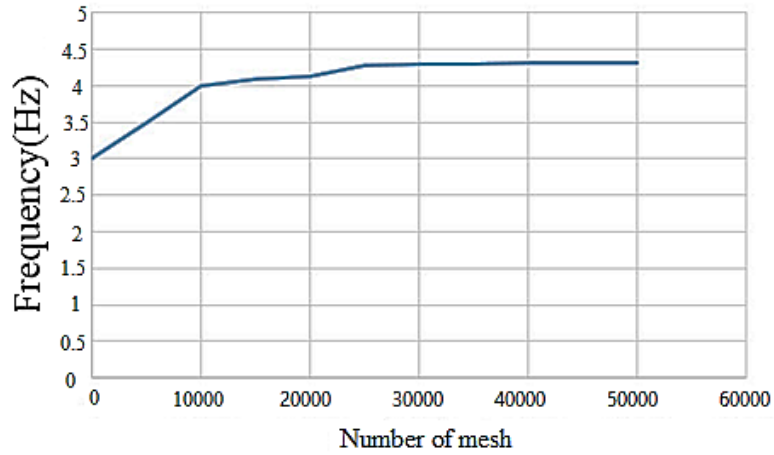


Figure 3: Mesh of cylindrical shell

According to Table 2, the optimal value for solving a finite element in a way that does not depend on the number of meshes, the point of mesh with average element quality is 0.997.

Table 2: Mesh Quality

Mesh shape	Mesh vertices	Number of element	Average element quality
a	70372	387950	0.6643
b	558217	3221150	0.997

After finding the optimal mesh size that has no effect on the solution of the problem, the results for the natural frequency for the form of various modes are presented in Table 3. In Figures 4 and 5, the shape of the vibrational modes associated with the first to third vibrational frequencies in contact with air and in contact with the fluid with fully bounded boundary conditions of the FGM (AL/A12 O3) with $\alpha = 1$ is shown.

Table 3: The results of the analysis for various figures

Mode shapes (m, n)	Natural frequency (Hz)
(1,2)	4.53
(1,3)	8.35
(2,3)	11.31
(2,2)	11.53
(1,3)	15.18
(3,3)	20.39
(2,4)	12.51
(3,4)	22.98

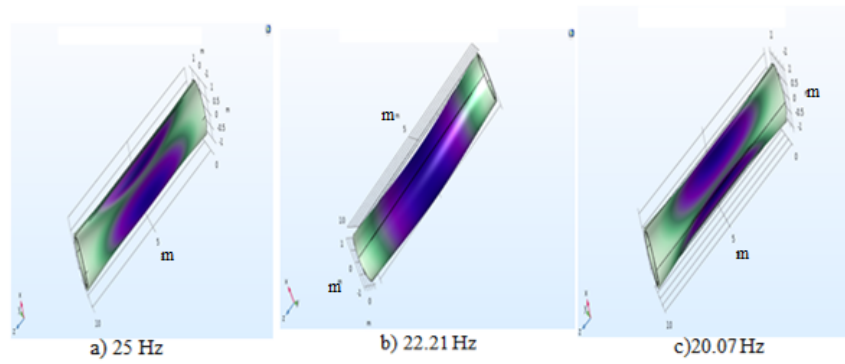


Figure 4: Vibrating base frequency variations of FGM in contact with air by clamped boundary conditions

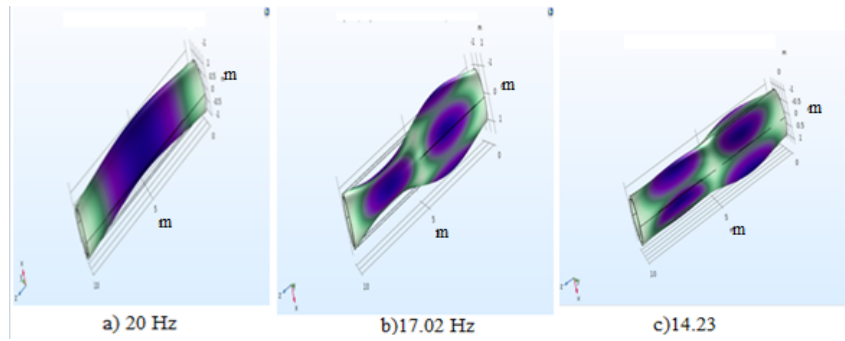


Figure 5: Vibrating base frequency variations of FGM in contact with fluid by clamped boundary conditions

In order to obtain the results of the analysis, with the problem solving, we consider $R = 1m$, $L/R = 15$, and $h/R = 0.005$ and a speed of 1 m/s for the fluid in equation (8). The results is presented in Table 4.

Table 4: Natural frequency in solving software and analytical solution

Mode number	Mode shapes (m, n)	Natural frequency in analytical solution	Natural frequency in solving software
1	(1,2)	5.001	4.92
2	(1,3)	8.71	8.97
3	(2,3)	10.13	11.26
4	(2,2)	10.89	11.42
5	(3,3)	13.85	15.29
6	(1,4)	18.98	20.48
7	(2,4)	20.08	20.57
8	(3,4)	23.01	23.79

As shown in Table 4, in low-frequencies, the results of the analysis of finite element and the analytical results extracted from the above-mentioned method with high accuracy are consistent. In order to better understand the effect of fluid, which is associated with nonrotating flow at a speed of 1 m/s , the comparison between the results of this paper and the results of [7, 14] is presented in Table 5.

Table 5: Comparison of the results of the present study with other similar studies performed in clamped boundary conditions and geometric conditions $L/R = 15$, $h/R = 0.005$, $R = 1m$

Mode number	Mode shapes (m,n)	Natural frequency for fluid contact (submerged cylinder)[14]	Natural frequency for fluid flow inside the tube [7]	Natural frequency for fluid flow with fluid stationary (in this study)
1	(1,2)	5.21	4.93	4.97
2	(1,3)	9.98	8.94	9.31
3	(2,3)	11.36	10.64	10.2
4	(2,2)	11.6	11.48	11.11
5	(3,3)	14.98	14.66	13.64
6	(1,4)	19.01	18.26	17.1
7	(2,4)	19.47	18.73	17.45
8	(3,4)	21.1	19.96	19.1

Given the fact that the fluid velocity can affect the natural vibrational frequencies of the sheet, we consider four different speeds for the fluid. The results for the form of various modes for the geometric conditions $L/R = 15$, $h/R = 0.005$, and $R = 1m$ are shown in table 6.

Table 6: Comparison of natural frequencies for different speeds

Mode number	Mode shapes (m,n)	Natural frequency without speed	Natural frequency for speed 2m/s	Natural frequency for speed 4m/s	Natural frequency for speed 6m/s
1	(1,2)	5.1	5.02	4.58	4.5
2	(1,3)	8.69	9.23	8.61	8.37
3	(2,3)	9.82	9.87	9.77	9.44
4	(2,2)	10.49	10.75	10.23	9.89

In order to better understand the effect of fluid velocity on the natural frequency of the sheet, according to the results of the above table, for different velocities of the fluid, the clamped boundary conditions and the geometric conditions $L/R = 20$, $h/R = 0.002$, and $R = 1m$ for the shape of the various modes in Figure 6 are plotted.

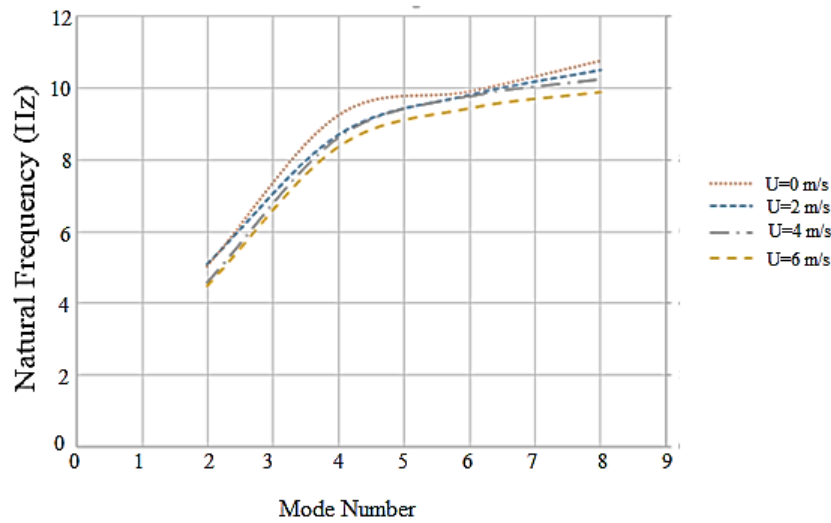


Figure 6: Comparison of the effect of velocity of fluid inside the cylindrical shell on the natural frequency of vibration of shell

As shown in Figure 6, the increase in velocity reduces the natural frequency of the first modes of the shell. It is necessary to note that the effect of the velocity of the fluid on reducing the natural frequencies of the first modes at low velocities is more than high speeds. On the other hand, according to Figure 6, reducing natural frequency in different modes has been done with the change in fluid velocity by approximately one ratio, which can generally be said that increasing the velocity of the fluid in the cylinder reduces the natural frequencies. In other words, it can be expected that by increasing fluid velocity due to increasing fluid dynamic pressure, the con-

nection between the fluid and the structure becomes stronger, which reduces the natural frequencies. This means that the mechanism of the effect of fluid velocity is similar to the mechanism of the effect of fluid and structure continuity, and, likewise [34]. This shows that, considering the effect of the fluid and structure continuity, the natural frequencies of each vibrational mode reduces. Different boundary conditions for free vibration of shell make the numbers of different longitudinal wave. In other words, due to the fact that free vibration of shell must satisfy the boundary conditions (geometric conditions such as displacement, slope, etc.), it is necessary to consider the longitudinal wave number that according to the longitudinal wave number (m), the geometry conditions of the beginning and end of the shell are also realized. According to this issue, it is expected that changing the boundary conditions affects the free vibration of shell.

As shown in Table 7, a change in the boundary conditions makes vibration of sheet undergoing fundamental changes; for this purpose, in the analytical model provided with $U = 2m/s$, the results are as follows:

Table 7: Comparison of the effect of different boundary conditions on natural sheet frequencies in conditions $U = 2m/s$, $L/R = 15$, $h/R = 0.005$, $R = 1m$

Mode number	Mode shapes (m, n)	natural frequencies (HZ)		
		C-C	C-SS	SS-SS
1	(1,2)	5.02	4.66	3.71
2	(1,3)	9.23	7.93	7.18
3	(2,3)	9.89	8.52	6.79
4	(2,2)	10.75	8.95	7.88

As shown in Table 7, different boundary conditions affect the natural frequency of the free vibration of submerged sheet containing fluid flow. According to the results of Table 7, (Clamped-Clamped) C-C mode, which means the clamped boundary conditions for a thin cylindrical shell, the free vibration frequencies have the highest value. In other words, it can be claimed that, by limiting the structure, the first natural frequency will be higher, and consequently, other modes will also have higher frequencies. It is also observed that in (Clamped-Simply Supported) C-SS mode, which means the beam of one side is clamped and one side is simple supported, the natural frequency for each mode is greater than the corresponding mode for the (Simply Supported-Simply Supported) SS-SS mode (simple supported boundary conditions). It should be noted that the SS-SS mode has the least number of constraints among the three modes. It is expected that the flowing fluid shell in this type of boundary condition will require less energy to vibrate, which means that the natural frequencies of free vibrations of the sheet in this case have the least amount than the other boundary conditions. In order to

understand the effect of different boundary conditions on the free vibration frequency, a comparison has been made between different states in Figure 7.

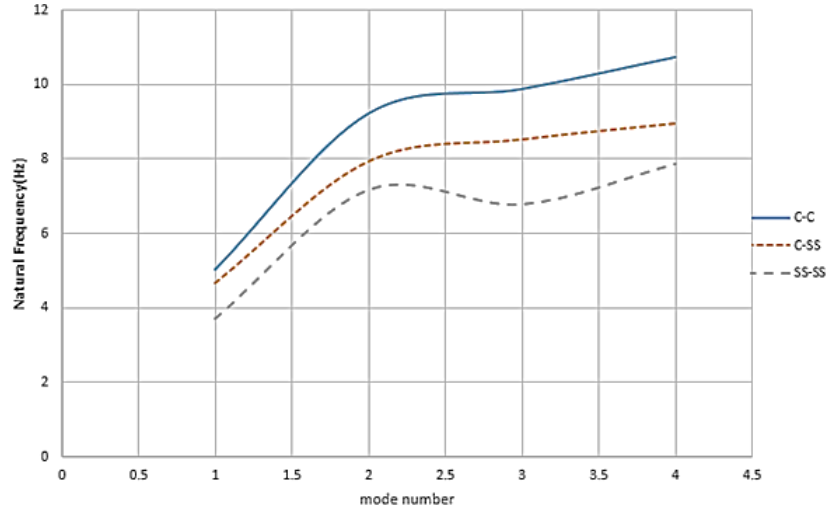


Figure 7: Different boundary conditions $U = 2m/s$, $L/R = 15$, $h/R = 0.005$, $R = 1m$

According to Figure 7, it is concluded that the effect of the SS-SS boundary conditions on the reduction of the natural frequency of sheet vibrations is higher in higher modes. Since the reduction of the natural frequencies of the first vibrational modes has also been observed in other studies due to the change of the boundary conditions of the system from a completely constrain to free state (Iqbal 2009) and (Shah 2011) [14, 24]. According to the results of Figure 7, it can be claimed that in the shell of fluid carrier and submerged in water, the boundary conditions with the number of degrees of constrained release increase the natural frequencies. The main difference observed in this study is that in higher modes, the effect of reducing constraints in boundary conditions on reducing the natural frequency of free shell vibrations is greater than the first and second modes.

Given that the extracted values in the previous sections relate to particular geometric conditions for the shell, it is necessary to examine the effect of geometric parameters on this problem. In the previous sections, parameters $L/R = 15$, $h/R = 0.005$, $R = 1m$ were used. In this section, by changing each of the parameters, the effect of that parameter on the vibrational behavior of the shell is studied. one of the most important geometric parameters is the thickness of the shell, and since in this paper, by assuming the plate stress, the free vibration problem is solved with the wave propagation method, considering the important point that the FGM is used in the structure of this shell, and changes in material from the outer surface to the inner surface are using a shell-thickness function, it can be expected that changing the h/R

ratio affects the natural frequencies of the free vibration of the cylindrical shell.

Table 8: Comparison of the effect of h/R under different velocity conditions of the fluid on the natural frequencies of the structure under conditions of $L/R = 15$ and $R = 1m$

Mode number	Mode shapes (m,n)	$h/R = 0.005$		$h/R = 0.015$		$h/R = 0.03$	
		$U=0m/s$	$U=6m/s$	$U=0m/s$	$U=6m/s$	$U=0m/s$	$U=6m/s$
1	(1,2)	5.1	4.5	9.78	8.23	25.02	19.88
2	(1,3)	8.69	8.37	24.58	23.11	38.98	34.58
3	(2,3)	9.8	9.44	25.01	24.13	40.5	38.8
4	(2,2)	10.49	9.89	35.32	33.58	57.69	47.57

According to Table 8, as expected, the effect of changing the h/R parameter on the natural frequencies of the system is very high, so that it can be said that the fluid velocity change has less effect on the natural frequencies of the cylindrical shell made of FGMs. To understand how affecting the h/R ratio and examining its simultaneous effect with the fluid velocity, Figure 8 is plotted.

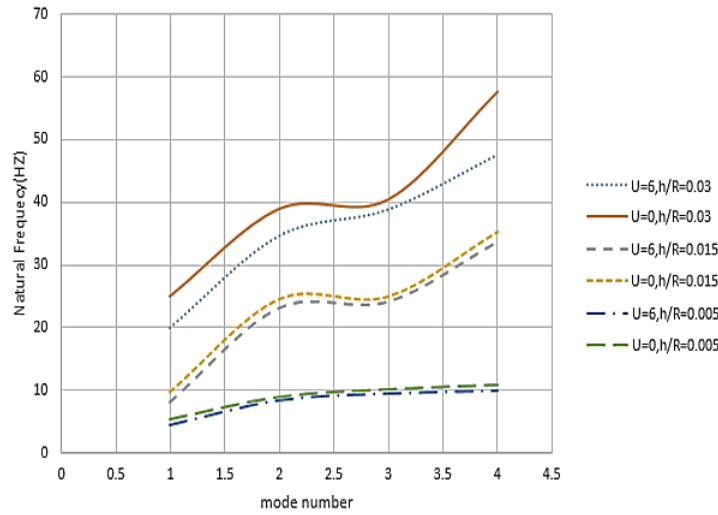


Figure 8: Comparison of the natural frequency of the first modes of sheet under different boundary conditions

Figure 8 shows that by decreasing the h/R ratio, the frequencies also decrease in such a way that at low ratios h/R , by increasing number of modes and a small amount is added to frequency, but in larger values of h/R , by increasing the number of mode and increasing the amount, frequency will

be more. In other words, increasing the h/R ratio will increase the distance between natural frequencies (sequential modes).

In addition, with the increase of h/R ratio, which means increasing the thickness of shell, the natural frequencies of the shell vibration increase, which can be said here, although this problem has been solved for a submerged FGM containing fluid flow, but, as nonemerged isotropic cylinder, the natural frequencies increase by increasing the shell thickness. On the other hand, it can be found from Figure 8 that the increase in the velocity of fluid flow inside the cylindrical shell always has a nearly identical effect on vibrational behavior and by increasing fluid flow due to increasing interaction of the structure and fluid, natural frequencies decrease a little. In addition, the natural frequency variations of the cylinder in contact with the fluid and in contact with air have been discussed in terms of the volumetric power factor.

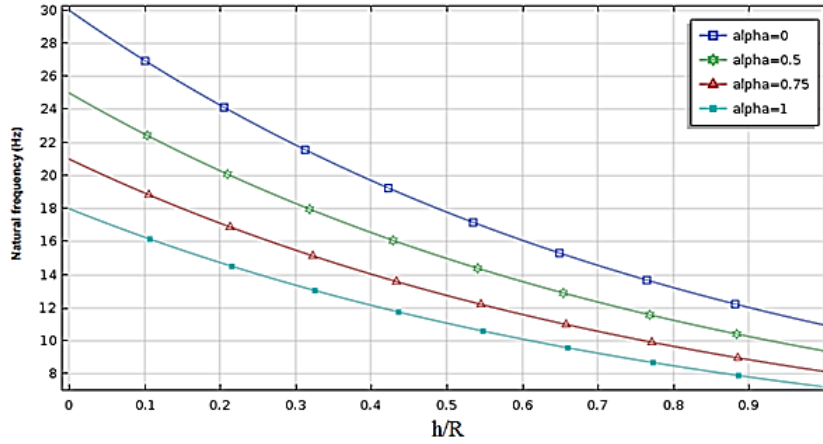


Figure 9: Vibrating base frequency variations of FGM in contact with air in accordance with h/R ratio and four different volumetric ratios with clamped boundary conditions

In Figure 9, the graph of the variations of the vibrating base frequency of sheet from FGM in contact with air with a thickness ratio of $L/R = 15$ and clamped boundary conditions for the power factor of different volume proportions is shown. According to the results presented in the figure above, the more the amount of thickness in the radius increases, the vibration frequency of sheet will be low, and it is also observed that the more the power factor of the volume ratio increases, the frequency of the system also increases. By increasing the volume ratio, the shell rigidity is increased and the ceramic percentage in the sheet increases. The reason is due to the direct relationship between the frequency and the volume ratio in the above figure.

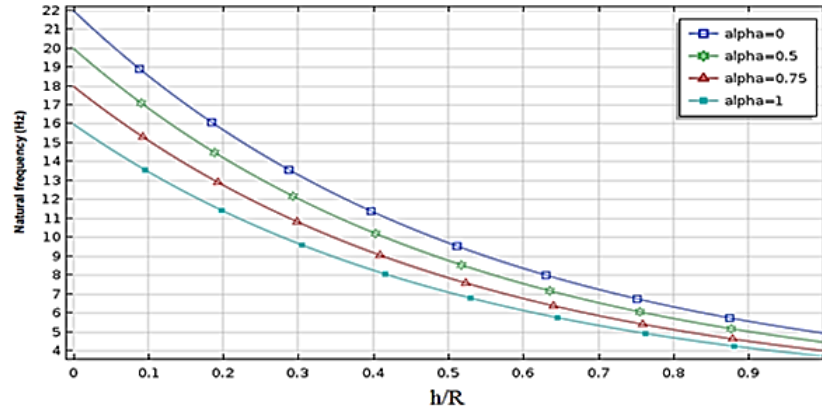


Figure 10: Vibrating base frequency variations of FGM in contact with fluid in accordance with h/R ratio and four different volumetric ratios with clamped boundary conditions

Figure 10 shows the graph of the cylinder base frequency variation from the FGM in contact with the fluid for the power factor of the various volumetric ratios with the clamped boundary conditions. A cylinder with a ratio $L/R = 15$ is in contact with a fluid with a density of $\rho_f = 997 \text{ kg/m}^3$. First, by comparing the values of the shell frequency in contact with air and fluid, under the same conditions, the frequency in the shell contact with the air is more than the shell's contact state with turbulent fluid. Figure 10 shows that the more the power factor of the volume ratio increases, the system frequency also increases.

5 Conclusions

This paper investigated the free vibration of the cylindrical shell of FGM in contact with turbulent fluid and used first-order shear theory to analyze it. Solving equations related to wave propagation was used to solve the problem of determining free frequencies in a cylindrical shell, which has a structure-fluid interaction. In the fluid inside the shell, because the fluid has a velocity along the longitudinal axis, the potential function was written based on the pressure and velocity of fluid flow. In the case of fluid outside the shell, since shell was considered submerged and fluid constant, solving equations was made by using the hydrostatic pressure. To validate, the analytical results were compared with the results of the analysis by using the Comsol software and the results of other studies. In the boundary conditions, we used three movable simple, non-movable simple, and completely clamped

modes, which were discussed more because of the importance of a completely clamped boundary condition.

Using simulation data provided, the effects of different parameters such as the fluid inside and outside the shell, volume fraction exponent, geometrical parameters, and boundary conditions on the natural frequencies were discussed in detail. The natural frequencies corresponding to the fundamental axial mode number and the smaller values of circumferential wave numbers were strongly affected by the fluid.

By examining the results, a summary of the general conclusion is as follows:

- 1) Increasing the thickness causes the increase of effect of speed on the reduction of natural frequencies, especially in higher modes.
- 2) By examining the motion of the fluid at different velocities, it is possible to determine the effect of the thickness change, so that the increase in thickness will increase the effect of speed on the reduction of natural frequencies, especially in higher modes.
- 3) By increasing the volumetric power factor due to the increase in rigidity of the shell, also the natural frequency increases.
- 4) The effect of boundary conditions on the free natural frequency was such that, as the constraints increased, the frequency of vibrations also increased, so that the natural frequency in the clamped mode is more than the other two.
- 5) The natural frequency values of the cylindrical shell in contact with the air are more than contact with the turbulent fluid.

6 Nomenclature

c	speed of sound(m/s)	v	modal displacement Direction θ
H	Thickness(m)	D	Material properties
E	young modulus(GPa)	V	Volume fraction
k	wave number		Greek letters
M	Moment resultant(N.m)	ρ'	The mass density of the shell material
P	pressure(pa)	ρ	Fluid density(kg/m^3)
U	Velocity fluid (m/s)	α	Poisson ratio
N	Force(N)	v	volume ratio
Q	Reduced Stiffness(N/m)	ω	natural frequency(Hz)
			Subscripts

L	differential operator	i, j	components
A	wave amplitude in the x direction	t	time
B	wave amplitude in the θ direction	f	Contained fluid
C	wave amplitude in the z direction	a	acoustic
r	radius	z	Direction z
n	The number of circumferential waves		
w	modal displacement Direction z	x	Direction x
u	modal displacement Direction x	$x\theta$	Direction $x\theta$
J	Bessel function of first kind	θ	Direction θ
T	Coefficients of the stiffness	o	mass
FL	Effect of acoustic pressure	g	Power law
L	Differential operator	s	axial
ω	Natural frequency of cylindrical shell		

References

1. Abolghasemi, S., Eipakchi, H.R. and Shariati, M. *Analytical solution for buckling of rectangular plates subjected to non-uniform in-plane loading based on first-order shear deformation theory*, Modares Mechanical Engineering, 14 (2015), 37–46.
2. Benchouaf, L. and Boutyour, E.I. *Nonlinear vibrations of buckled plates by an asymptotic numerical method*, Comptes Rendus Mécanique, 344 (2016), 151–166.
3. Benferhat, R., Hassaine Daouadji, T. and Said Mansoura, M. *Free vibration analysis of FG plates resting on an elastic foundation and based on the neutral surface concept using higher-order shear deformation theory*, Comptes Rendus Mécanique, 344 (2016), 631–641.
4. Cherradi, D., Delfosse, B. and Kawasaki, A. *La Revue de Metallurgie*, CIT, 34 (1996), 185–196.
5. Ebrahimi, F., Ghasemi, F. and Salari, E. *Investigating thermal effects on vibration behavior of temperature-dependent compositionally graded Euler beams with porosities*, Meccanica, 51 (2016), 223–249.
6. Ergin, A. and Ural B. *Linear vibration analysis of cantilever plates partially Submerged in fluid*, J. Fluid Struct. 17 (2003), 927–939.

7. Farahani, H., Azarafza, R. and Barati, F. *Mechanical buckling of a functionally graded cylindrical shell with axial and circumferential stiffeners using the third-order shear deformation theory*, Comptes Rendus Mécanique, 342 (2014), 501–512.
8. Ghasemi, M.A., Yazdani, M. and Soltan Abadi, E. *Buckling behavior investigation of grid stiffened composite conical shells under axial loading*, Modares Mechanical Engineering, 14 (2015), 170–176.
9. Guo, X.Y. and Zhang, W. *Nonlinear vibrations of a reinforced composite plate with carbon nanotubes*, Compos. Struct. 135 (2016), 96–108.
10. Haddara, M.R. and Cao, S. *A study of the dynamic response of submerged Rectangular flat plates*, Mar. Struct. 9 (1996), 913–933.
11. Cahn, R.W., Haasen, P. and Kramer, E.J. *Materials science and technology: A Comprehensive catchment*, Materials Science and Technology, VCH, 1996.
12. Hosseini-Hashemi, Sh., Fadaee M. and Atashipour S.R. *Study on the free vibration of thick functionally graded rectangular plates according to a new exact closed-form procedure*, Compos. Struct. 93 (2011), 722–735.
13. Hosseini-Hashemi, Sh., Rokni-Damavandi-Taher, H., Akhavan, H. and Omid, M. *Free vibration of functionally graded rectangular plates use first-order shear deformation plate theory*, Appl. Math. Model. 34 (2010), 1276–1291.
14. Igbal, Z., Naeem, M.N. and Sultana, N. *Vibration characteristics of FGM circular cylindrical shells using wave propagation approach*, Acta Mech. (2009), 237–248.
15. Jeong, K.H., Yoo, G.H. and Lee, S.C. *Hydroelastic vibration of two identical rectangular plates*, J. Sound Vib. 272 (2003), 539–555.
16. Khorshidi, K. *Effect of Hydrostatic pressure on vibrating rectangular plates coupled with fluid*, Sci. IRAN. 17 (2010), 415–429.
17. Khorshidi, K. and Farhadi, S. *Free vibration analysis of laminated composite rectangular plate in contact with bounded fluid*, Compos. Struct. 104 (2013) 176–186.
18. Kwak, M.K. *Hydroelastic vibration of rectangular plates*, J. Appl. Mech. Mar. 63 (1996), 110–115.
19. Leissa, A.W. *Vibration of shells*, NASA SP-288, Reprinted by Acoustical Society of America, America institute of Physics, (1993).
20. Liang, C.C., Liao, C.C., Tai, Y.S. and Lai, W.H. *The free vibration analysis of Submerged cantilever plates*, Ocean Eng. 28 (2001), 1225–1245.

21. Mirzaei, M. and Kiani, Y. *Isogeometric thermal buckling analysis of temperature dependent FG graphene reinforced laminated plates using NURBS formulation*, Compos. Struct. 180 (2017), 606–616.
22. Morse M. and Ingard, K.U. *Theoretical Acoustics*, Am. J. Phys. 38 (1970), 666–667.
23. Robinson, N.J. and Palmer, S.C. *A modal analysis of a rectangular plate floating On an incompressible liquid*, J. Sound Vib. 142 (1990), 435–460.
24. Shah, G., Mahmood, T., Naeem, M.N. and Arshad, S.H. *Vibration characteristics of fluid-filled cylindrical shells based on elastic foundations*, Acta Mech. 216 (2011), 17–28.
25. Shen, H.S. and Chen, T.Y. *Buckling and postbuckling behavior of cylindrical shells under combined external pressure and axial compression*, Thin-Walled Struct. 12 (1991), 321–334.
26. Shen, H.S., Xiang, Y. and Lin, F. *Nonlinear bending of functionally graded graphene-reinforced composite laminated plates resting on elastic foundations in thermal environments*, Compos. Struct. 170 (2017), 80–90.
27. Shen, H.S., Xiang, Y. and Lin, F. *Nonlinear vibration of functionally graded graphene-reinforced composite laminated plates in thermal environments*, Comput. Methods Appl. Mech. Eneg. 319 (2017), 175–193.
28. Shen, H.S., Xiang, Y. and Lin, F. *Thermal buckling and postbuckling of functionally graded graphene-reinforced composite laminated plates resting on elastic foundations*, Thin-Walled Struct. 118 (2017), 229–237.
29. Shen, H.S., Xiang, Y., Lin, F. and D. Hui, *Buckling and postbuckling of functionally graded graphene-reinforced composite laminated plates in thermal environments*, Composites, 119 (2017), 67–78.
30. Song, M., Kitipornchai, S. and Yang, J. *Free and forced vibrations of functionally graded polymer composite plates reinforced with graphene nanoplatelets*, Compos. Struct. 159 (2017), 579–588.
31. Talha, M. and Singh, B.N. *Static response and free vibration analysis of FGM Plates using higher order shear deformation theory*, Appl. Math. Model. 34 (2009), 3991–4011.
32. Yadykin, Y., Tenetov, V. and Levin, D. *The added mass of flexible plate oscillating in fluid*, J. Fluids Struct. 17 (2003), 115–123.
33. Yu, Y., Shen, H.S., Wang, H. and Hui, D. *Postbuckling of sandwich plates with grapheme reinforced composite face sheets in thermal environments*, Compos. B. Eng. 135 (2018), 72–83.

34. Zhang, X.M., Liu, G.R. and Lam K.Y. , *Vibration analysis of thin cylindrical shells using wave propagation approach*, J. Sound Vib. 239 (2001), 397–403.
35. Zhao, X., Lee, Y.Y. and Liew, K.M. *Free vibration analysis of functionally graded plates using the element-free kp-Ritz method*, J. Sound Vib. 319 (2009), 918–939.
36. Zhou, D. and Cheung, Y.K. *Vibration of vertical rectangular plate in contact with water on one side*, Earthq. Eng. Struct. Dyn. 29 (2000), 693–710.
37. Zhou, D. and Liu, W. *Hydroelastic vibrations of flexible rectangular tanks partially filled with liquid*, Int. J. Numer. Meth. Eng. 71 (2007), 149–174.

Aims and scope

Iranian Journal of Numerical Analysis and Optimization (IJNAO) is published twice a year by the Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad. Papers dealing with different aspects of numerical analysis and optimization, theories and their applications in engineering and industry are considered for publication.

Journal Policy

After receiving an article, the editorial committee will assign referees. Refereeing process can be followed via the web site of the Journal.

The manuscripts are accepted for review with the understanding that the work has not been published and also it is not under consideration for publication by any other journal. All submissions should be accompanied by a written declaration signed by the author(s) that the paper has not been published before and has not been submitted for consideration elsewhere.

Instruction for Authors

The Journal publishes all papers in the fields of numerical analysis and optimization. Articles must be written in English.

All submitted papers will be refereed and the authors may be asked to revise their manuscripts according to the referee's reports. The Editorial Board of the Journal keeps the right to accept or reject the papers for publication.

The papers with more than one authors, should determine the corresponding author. The e-mail address of the corresponding author must appear at the end of the manuscript or as a footnote of the first page.

It is strongly recommended to set up the manuscript by Latex or Tex, using the template provided in the web site of the Journal. Manuscripts should be typed double-spaced with wide margins to provide enough room for editorial remarks.

References should be arranged in alphabetical order by the surname of the first author as examples below:

- [1] Stoer, J. and Bulirsch, R. *Introduction to Numerical Analysis*, Springer-verlag, New York, 2002.
- [2] Brunner, H. *A survey of recent advances in the numerical treatment of Volterra integral and integro-differential equations*, J. Comput. Appl. Math. 8 (1982), 213-229.

Submission of Manuscripts

Authors may submit their manuscripts by the online submission via the website of the Journal at:

<https://ijnao.um.ac.ir/about/submissions>

Copyright Agreement

Upon the acceptance of an article by the Journal, the corresponding author will be asked to sign a "Copyright Transfer Agreement" (see the website) and send it to the Journal address. This will permit the publisher to publish and distribute the work.

The most probable allocation solution for the p-median problem	155
M. Zaferanieh	
Numerical solution of the Allen-Cahn equation by using "shifted" surface spline radial basis functions.	177
C.G Keshavarzi and F. Ghoreishi	
On the numerical solution of optimal control problems via Bell polynomials basis	197
M.R. Dadashi, A.R. Haghighi, F. Soltanian and A. Yari	
An approximate method based on Bernstein polynomials for solving fractional PDEs with proportional delays.	223
A. Ketabdari, M.H. Farahi and S. Effati	
Nonlinear vibrations of functionally graded cylindrical shell by using numerical analysis in the wave propagation method.	241
M.R. Assari, A. Kavousi Nejad and Sh. Amirshirzad	

2D-fractional Muntz-Legendre polynomials for solving the fractional partial differential equations	1
E. Hengamian Asl, J. Saberi-Nadjafi and M. Gachpazan	
Solving biobjective network flow problem associated with minimum cost-time loading.	33
O. Baghani and S. Ghafoori	
A noniterative domain decomposition method for the forward-backward heat equation	49
S. Banei and K. Shanazari	
Comparison of homotopy perturbation transform method and fractional Adams-Bashforth method for the Caputo-Prabhakar nonlinear fractional differential equations	63
M.H. Derakhshan and A. Aminataei	
A four directions variational method for solving image processing problems ..	87
A.R. Hosseini and E.E. Esfahani	
Maximum probability O-D matrix estimation in large-sized networks	105
M. Abareshi	
Numerical methods for solving nonlinear Volterra integro-differential equations based on Hermite-Birkhoff interpolation.	137
S. Fazeli	
The rest of contents is on back of the page	

web site: <http://ijnao.um.ac.ir>

Email: ijnao@um.ac.ir

ISSN-Print: [2423-6977](#)

ISSN-Online: [2423-6969](#)