Iranian Journal of Numerical Analysis and Optimization

Volume 9 , Number 1

Winter 2019

Serial Number: 15

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 9, Number 1, Winter 2019

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

Publisher: Faculty of Mathematical Sciences, Ferdowsi University of Mashhad

Published by: Ferdowsi University of Mashhad Press

Circulation: 50

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:

• Zentralblatt

• ISC

• SID

به اطلاع کلیه محققان، پژوهشگران، اساتید ارجمند، دانشجویان تحصیلات تکمیلی و نویسندگان محترم می رساند که نشریه ایرانی آنالیز عددی و بهینه سازی - IJNAO - طبق مجوز شماره ۳/۱۸/۵۴۸۹۱۳/ مورخه ۱۳۹۲/۱۰/۲۵ مدیر کل محترم سیاست گذاری و برنامه ریزی امور پژوهشی وزارت علوم، تحقیقات و فناوری، علمی – پژوهشی میباشد.
بر اساس نامه شماره ۹۲/۱۵۳۶/پ مورخه ۱۳۹۲/۱۱/۲۱ سرپرست محترم معاونت پژوهشی وفناوری پایگاه استنادی علوم جهان اسلام، نشریه ایرانی آنالیز عددی و بهینه سازی در پایگاه ISC نیز نمایه می شود.

Iranian Journal of Numerical Analysis and Optimization

Volume 9, Number 1, Winter 2019

Ferdowsi University of Mashhad - Iran

Iranian Journal of Numerical Analysis and Optimization

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

---

* Full Professor

** Associate Professor

We would like to acknowledge the help of 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 biannually 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

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 1–16
DOI:10.22067/ijnao.v9i1.53910

Transformation to a fixed domain in LP modelling for a class of optimal shape design problems

H.H. Mehne* and M.H. Farahi

Abstract

A class of optimal shape design problems is studied in this paper. The boundary shape of a domain is determined such that the solution of the underlying partial differential equation matches, as well as possible, a given desired state. In the original optimal shape design problem, the variable domain is parameterized by a class of functions in such a way that the optimal design problem is changed to an optimal control problem on a fixed domain. Then, the resulting distributed control problem is embedded in a measure theoretical form, in fact, an infinite-dimensional linear programming problem. The optimal measure representing the optimal shape is approximated by a solution of a finite-dimensional linear programming problem. The method is evaluated via a numerical example.

Keywords: Approximation; Optimal shape design; Linear programming; Measure theory.

1 Introduction

An optimal shape design (OSD) problem is concerned with the optimization of a performance index depending on the shape of some region. Many-faceted problems naturally arise in engineering applications with the goal of designing a specific structure in an optimal sense. Typical applications are design of a nozzle [12], airfoil boundary [4, 10], and spacecraft shape [6] with respect to specific optimality conditions.

*Corresponding author
Received 16 February 2016; revised 3 July 2016; accepted 11 November 2017
H.H. Mehne
Aerospace Research Institute, Tehran, 14665-834, Iran. e-mail: hmehne@ari.ac.ir

M.H. Farahi
Department of Applied Mathematics, Faculty of Mathematical Sciences, Ferdowsi University of Mashhad, Mashhad, Iran. e-mail: farahi@math.um.ac.ir

There are several methods for a numerical solution of an OSD problem, for example, gradient based methods [2], the Newton method [9,15], sequential quadratic method [19], and fictitious domain method [8]. In the above mentioned methods, computation of solution is a time-consuming task, because they need to solve many boundary value problems. Moreover, these methods are iterative and they need to have an initial guess for solution.

The main focus of this article is to find an appropriate formulation of the optimal shape design problem that is attractive for consistent numerical computation. The advantages of the proposed method lie in the fact that the method is not iterative, that it is self-starting, and that it does not need to solve the corresponding boundary value problem. Moreover, in the modeling of the OSD problem, differentiability of the cost function being a limiting condition in the above mentioned methods, is reduced to a measurability condition. In the methods of [9], the control and state functions are discretized and the linearity of variational forms is an inherent condition, while it is not a restrictive condition in the proposed method, where a general variational form is studied.

In the present work, the variable domain is first parameterized by a class of functions in such a way that the OSD problem changes to an optimal control problem accompanied with partial differential equations on a fixed domain. The converted problem can be solved by extending the measure theoretical method proposed by Rubio [16]. The methods of this type were frequently and successfully applied to different OSD problems. In [3] the problem of finding the upper wall of a nozzle was discussed where the goal is to match the velocity of flow to a constant prescribed value in a given subregion near exit. It also includes details about the measure theoretical approach to solve OSD problems. The same problem for pressure of the flow was given in [12] with more numerical simulations to study the sensitivity analysis and stability of the method. In [4] the problem of finding the optimum shape for a low speed airfoil has been solved by the method. A problem concerning with the control of thermoelastic deformation was also presented in [13], where the extension of the method of [3] has been used to solve it.

The main difference of the present work with the above mentioned researches is that the optimization occurs in a completely unknown two-dimensional area and that there is no restriction to any special form of the problem. Such a generalization of the domain is also presented in [7], which is based on the domain discretization, while in the present paper, the transformation of domain is used.

2 The optimal shape design problem

The set of admissible shapes in the definition of an OSD problem is the set of all possible shapes in which we seek for the best. In our problem definition, let

consider a class of OSD problems where the admissible shapes are described by two-dimensional domains with a free boundary:

$$\tilde{\Omega} = \{\tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T \in \mathbb{R}^2 \ \mid \ \tilde{x}_2 \in I, \ \tilde{x}_1 \in (0, u(\tilde{x}_2))\},$$

where $I = (0, 1)$. The set of admissible shapes is usually parameterized or discretized in order to reduce the search process from the class of admissible shapes to a set of admissible functions. So, we parameterize the above domains by a function $u \in \mathcal{U}_{ad}$, where

$$\mathcal{U}_{ad} = \{u \in C^1(I) \mid u(0) = u(1) = 1, \ u(\tilde{x}_2) \in [\beta_1, \beta_2], \ \frac{du}{d\tilde{x}_2}(\tilde{x}_2) \in [\alpha_1, \alpha_2]\},$$

where $\beta_1 > 0$, β_2, α_1, and α_2 are given.

By measuring the desirability of admissible shapes, let $\tilde{J}$ be a cost function depending on a function $\tilde{y}$ and its gradient as follows:

$$\tilde{J} = \int_{\tilde{\Omega}} \tilde{f}(\tilde{y}, \nabla\tilde{y}) d\tilde{x}, \tag{1}$$

where $\tilde{f}$ is a nonlinear real-valued measurable function and $\tilde{y}$ is defined by the solution of a second order elliptic boundary value problem on $\tilde{\Omega}$. The boundaries of the feasible domains are divided into the moving boundary part

$$\tilde{\Gamma}^M = \{(\tilde{x}_1, \tilde{x}_2)^T \in \mathbb{R}^2 \ \mid \ \tilde{x}_1 = u(\tilde{x}_2) \ \text{ for all } \tilde{x}_2 \in I\},$$

and the fixed boundary part $\tilde{\Gamma}^F$ with $\partial\tilde{\Omega} = \tilde{\Gamma}^M \cup \tilde{\Gamma}^F$ and $\tilde{\Gamma}^M \cap \tilde{\Gamma}^F = \emptyset$.

Now, the OSD problem is defined by

$$\min_{u \in \mathcal{U}_{ad}} \tilde{J}(u),$$

with the equality constraint

$$\int_{\tilde{\Omega}} \tilde{a}(\tilde{y}, \tilde{\eta}) d\tilde{x} = \int_{\partial\tilde{\Omega}} \tilde{l}(\tilde{\eta}) d\tilde{\Gamma} \qquad \text{for all } \tilde{\eta} \in \tilde{V},$$

which is a variational form of the underlying PDE, $\tilde{V} \subset H^1(\tilde{\Omega})$ is the space of corresponding test functions, and $H^1(\tilde{\Omega})$ is a Sobolev space of square integrable functions with square integrable first derivatives on $\tilde{\Omega}$. Functions $\tilde{a}$ and $\tilde{l}$ are assumed to be nonlinear in general. We use the notations of [9] in our article.

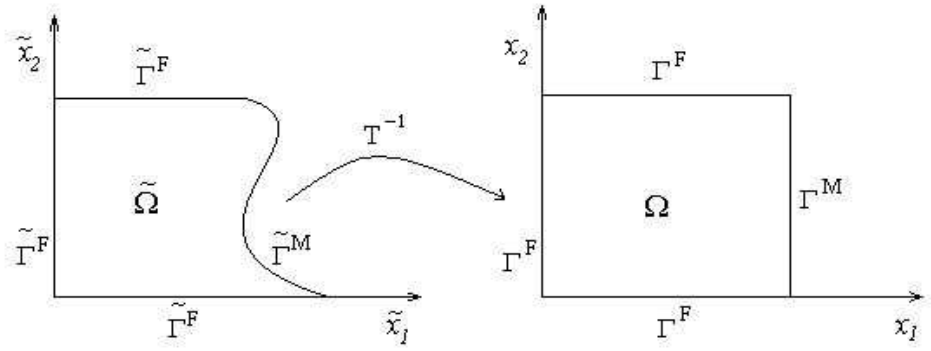

Figure 1: Transformation of the domain.

3 Transformation to a fixed domain

To cope with the variation of the domain, we transform the optimal shape design problem into an optimal control problem on a fixed domain. This routine approach has been used in literature before, for example [9, 13].

As sketched in Figure 1, the transformation defined by

$$\begin{aligned} T^{-1} : \tilde{\Omega} &\to \Omega \\ (\tilde{x}_1, \tilde{x}_2)^t &\mapsto (x_1, x_2)^t = \left(\frac{\tilde{x}_1}{u(\tilde{x}_2)}, \tilde{x}_2 \right)^t \end{aligned} \tag{2}$$

maps the moving domain $\tilde{\Omega}$ to the fixed reference area $\Omega = I \times I$.

So, we consider the problem over a fixed domain

$$\min_{u \in \mathcal{U}_{ad}} J(u, y),$$

with the equality constraint

$$\int_\Omega a(y, \eta, u, \frac{du}{dx_2}) dx = l_\eta \qquad \text{for all } \eta \in V, \tag{3}$$

where l_η is a function depending only on the test function η. Here, a includes all terms depending on y, η, u, and $\frac{du}{dx_2}$.

By this transformation, the unknown geometrical factors are transformed from boundary to equations, that is, the equations must be solved on a fixed known domain while they contain unknown parameters. It will be shown in the next sections that this type of problems may be solved by the described method.

The transformation will also changes the integrand of functional (1) as a function f dependent on y, ∇y, u and $\frac{du}{dx_2}$:

$$J = \int_{\Omega} f(u, y, \nabla y, \frac{du}{dx_2})dx \tag{4}$$

Now, we choose the derivative of $u(x_2)$ as a control variable w in order to obtain the following control system:

$$\frac{du}{dx_2} = w, \qquad \text{a.e.} \quad x_2 \in I, \tag{5}$$

$$u(0) = u(1) = 1. \tag{6}$$

The differential form of this control system should be changed as a variational form consistent to (3). To do this, let define an admissible pair for (5)–(6).

Definition 1. *The pair $p = (w, u)$ is said admissible if the control function $w : I \to [\alpha_1, \alpha_2]$ is measurable and $u \in \mathcal{U}_{ad}$ is the response of the system (5)–(6) to w.*

As unknown factors in (3)-(4) are depending to u, then for a given admissible pair $p = (w, u)$, there exists a solution for (3) with a cost measured by (4). So, the problem will be finding optimal admissible pair in the set of all admissible pairs which is assumed nonempty and is denoted by $\mathcal{P}$. However to find the best admissible pair we have to determine characteristic properties of an admissible pair.

Let $\Upsilon = \Omega \times [\beta_1, \beta_2] \times [\alpha_1, \alpha_2] \times \mathcal{A}$, where $\mathcal{A} \subset \mathbb{R}^3$ is the set that $(y(x_1, x_2), \nabla y(x_1, x_2))$ gets its values in it. Moreover, assume that B is an open ball in $\mathbb{R}^2$ containing $[\beta_1, \beta_2] \times I$, and let $C^1(B)$ be the space of all real-valued continuous differentiable functions with continuous first partial derivatives on B. For $\varphi \in C^1(B)$ and $p = (w, u) \in \mathcal{P}$, define $\varphi^w \in C(\Upsilon)$ as follows:

$$\varphi^w(x_1, x_2, u(x_2), w(x_2), y, \nabla y) = \varphi_1(u(x_2), x_2)w(x_2) + \varphi_2(u(x_2), x_2). \tag{7}$$

In (7), φ_1, φ_2 denote the first partial derivatives of φ with respect to its variables. Admissibility of $p = (w, u)$ implies:

$$\begin{aligned} &\int_0^1 \varphi^w(x_1, x_2, u(x_2), w(x_2), y, \nabla y)dx_2 \\ &\quad = \int_0^1 \{\varphi_1(u(x_2), x_2)w(x_2) + \varphi_2(u(x_2), x_2)\}dx_2 \\ &\quad = \varphi(u(1), 1) - \varphi(u(0), 0) := \Delta_\varphi, \end{aligned}$$

for all $\varphi \in C^1(B)$, where $\varphi(u(1), 1)$ and $\varphi(u(0), 0)$ are known. Integrating the above equation with respect to x_2 on I, the integral form of (5)–(6) can be written as

$$\int_{\Omega} \varphi^w(x_1, x_2, u(x_2), w(x_2), y, \nabla y)dx = \Delta_\varphi \qquad \text{for all } \varphi \in C^1(B). \tag{8}$$

Therefore, the optimal control formulation of the OSD problem may be interpreted as the minimization of (4) over variational constraints (3) and (8).

4 Measure theoretical approach

By Definition 1, each shape corresponds to exactly one admissible pair $p = (w, u) \in \mathcal{P}$. This is an injection between $\mathcal{U}_{ad}$ and $\mathcal{P}$. So, the minimization of (1) over the class of shapes is equivalent to the minimization of (4) over $\mathcal{P}$. Now, we transfer the problem of minimization of (4) over $\mathcal{P}$ into another nonclassical problem. Let us define the following mapping:

$$\Lambda_p : F \to \int_\Omega F(x_1, x_2, u(x_2), w(x_2), y, \nabla y)dx, \qquad F \in C(\Upsilon),$$

where $C(\Upsilon)$ is the space of all continuous real-valued functions defined on Υ. As an injection, the transformation $p \mapsto \Lambda_p$ provides us to describe the set of all admissible pairs $\mathcal{P}$ as a subset of the set of all linear continuous mappings on $C(\Upsilon)$. Moreover, by the Riesz representation theorem (see [18]), corresponding to Λ_p, there is a positive Borel measure μ on Υ such that

$$\Lambda_p(F) = \mu(F), \qquad \text{for all } F \in C(\Upsilon).$$

Now, the problem of minimization of the functional (4) over $\mathcal{P}$ is enlarged to the minimization of

$$\mathcal{I} \ : \ \mu \to \mu(f) \tag{9}$$

over the set of all measures μ satisfying

$$\mu(F_\eta) = l_\eta, \qquad \text{for all } \eta \in V, \tag{10}$$
$$\mu(G_\varphi) = \Delta_\varphi, \qquad \text{for all } \varphi \in C^1(B), \tag{11}$$

where $F_\eta(x_1, x_2, u(x_2), w(x_2), y, \nabla y) = a(y, \eta, u, \frac{du}{dx_2})$ and $G_\varphi = \varphi^w$. In other words, (10) and (11) are a measure theoretical interpretation of (3) and (8) respectively.

We denote the set of all positive Borel measures on Υ satisfying (10)–(11) as Q. We also assume that $\mathcal{M}^+(\Upsilon)$ is the set of all positive Borel measures on Υ. If we consider the space $\mathcal{M}^+(\Upsilon)$ with weak*-topology, it can be seen from [16], that Q is compact. In the sense of this topology, the functional $\mathcal{I} : Q \to \mathbb{R}$ defined by (9) is a linear continuous functional on the compact set Q, thus it attains its minimum on Q (see [16, Theorem III.1]), and then the measure-theoretical problem, which consists of finding the minimum of the functional (9) over a subset of $\mathcal{M}^+(\Upsilon)$, possesses a minimizing solution, say μ^*, in Q.

The minimization problem of (9)–(11) is an infinite-dimensional LP problem, and we are mainly interested in approximating it. It is possible to approximate the solution of the problem (9)–(11) by the solution of a finite-dimensional linear program of sufficiently large dimension.

We first consider the minimization of (9) not only over the set Q but over a subset of it defined by requiring that only a finite number of constraints (10)–(11) are satisfied.

Consider equalities (10)–(11), let the sets of functions $\{\eta_i,\ i = 1, 2, 3, \dots\}$ and $\{\varphi_j,\ j = 1, 2, 3, \dots\}$ have dense linear combinations in V and $C^1(B)$, respectively. Now we can prove the following.

Proposition 1. *Let $Q(M_1, M_2)$ be a subset of $\mathcal{M}^+(\Upsilon)$ consisting of all measures satisfying*

$$\begin{aligned} \mu(F_{\eta_i}) &= l_{\eta_i}, \qquad i = 1, 2, \dots, M_1, \\ \mu(G_{\varphi_j}) &= \Delta_{\varphi_j}, \qquad j = 1, 2, \dots, M_2. \end{aligned}$$

If $\theta(M_1, M_2) = \inf_{Q(M_1,M_2)} \mu(f)$ and $\theta = \inf_Q \mu(f)$, then $\theta(M_1, M_2) \to \theta$ as $M_1, M_2 \to \infty$.

Proof. See the proof of [3, Proposition 2]. □

The first stage of approximation is completed successfully. As the second stage, it is possible to develop a finite-dimensional linear program whose solution can be used to construct the solution of the infinite-dimensional linear program of minimizing (9) subject to (10)–(11). From [16, Theorem A.5], we can characterize a measure, say μ^*, in the set $Q(M_1, M_2)$ at which the function $\mu \to \mu(f)$ attains its minimum.

Proposition 2. *The measure μ^* in the set $Q(M_1, M_2)$ in which the function $\mu \to \mu(f)$ attains its minimum has the form*

$$\mu^* = \sum_{k=1}^{M_1+M_2} \varrho_k^* \delta(z_k^*) \tag{12}$$

with $z_k^* \in \Upsilon$ and $\varrho_k^* \geq 0, k = 1, 2, \dots, M_1 + M_2$.

Here $\delta(z)$ is the atomic measure concentrated at z, characterized by

$$\delta(z)(S) = \begin{cases} 1 & \text{if } z \in S, \\ 0 & \text{if } z \notin S, \end{cases}$$

for all $S \subset \Upsilon$; see [17]. This also implies that $\delta(z)(H) = H(z)$, where $H \in C(\Upsilon)$ and $z \in \Upsilon$. Now, the measure theoretical optimization problem is equivalent to a nonlinear optimization problem, in which unknowns are coefficients ϱ_k^* and supports $z_k^*, \text{for} k = 1, 2, \dots, M_1 + M_2$. It would be convenient if we could minimize the function $\mu \to \mu(f)$ only with respect

to the coefficients ϱ_k^*, for$k = 1, 2, \ldots, M_1 + M_2$, in (12), which would be a linear programming problem. However, we do not know the support of the optimal measure. The answer lies in the approximation of this support by introducing a dense set in Υ.

Proposition 3. *Let E be a countable dense subset of Υ. Given $\epsilon > 0$, a measure $\lambda \in \mathcal{M}^+(\Upsilon)$ can be found such that*

$$\begin{aligned} |(\mu^* - \lambda)(f)| &\leq \epsilon, \\ |(\mu^* - \lambda)(F_{\eta_i})| &\leq \epsilon, \qquad i = 1, 2, \ldots, M_1, \\ |(\mu^* - \lambda)(G_{\varphi_j})| &\leq \epsilon, \qquad j = 1, 2, \ldots, M_2, \end{aligned}$$

and the measure λ has the form

$$\lambda = \sum_{k=1}^{M_1+M_2} \varrho_k^* \delta(z_k),$$

where the coefficients ϱ_k^* are the same as in the optimal measure (12) and $z_k \in E$.

Proof. See the proof of [16, Proposition III.3]. □

Thus, the infinite-dimensional linear programming (9) with restrictions defined by (10)–(11) can be approximated by another LP problem in which z_k, for $k = 1, 2, \ldots, N$, belongs to the known dense subset of Υ. To construct such a subset, we grid entire of Υ into $N \gg M_1 + M_2$ nodes $z_k = (x_1^k, x_2^k, u^k, w^k, y^k, \nabla y^k)$. This gird may be generated by dividing Ω, $[\beta_1, \beta_2]$, $[\alpha_1, \alpha_2]$, and $\mathcal{A}$ separately and rearranging nodes from 1 to N. For example, one could divide $[0, 1]$ for x_1 to n_1 points: $x_{11}, x_{12}, \ldots, x_{1n_1}$ which repeat in arrangement of z_k's.

Now, we have the following LP problem with $\varrho_1, \varrho_2, \ldots, \varrho_N$ as nonnegative decision variables:

$$\text{Minimize} \qquad \sum_{k=1}^{N} \varrho_k f(z_k)$$

subject to

$$\begin{aligned} \sum_{k=1}^{N} \varrho_k F_{\eta_i}(z_k) &= l_{\eta_i} \qquad i = 1, 2, \ldots, M_1, \\ \sum_{k=1}^{N} \varrho_k G_{\varphi_j}(z_k) &= \Delta_{\varphi_j}, \qquad j = 1, 2, \ldots, M_2, \end{aligned}$$

The solution of this LP problem determines nodes that make the optimal measure and finally the optimal shape. As the solution may be trivial (zero), it is necessary to prevent from trivial situation. To do this, we benefit from

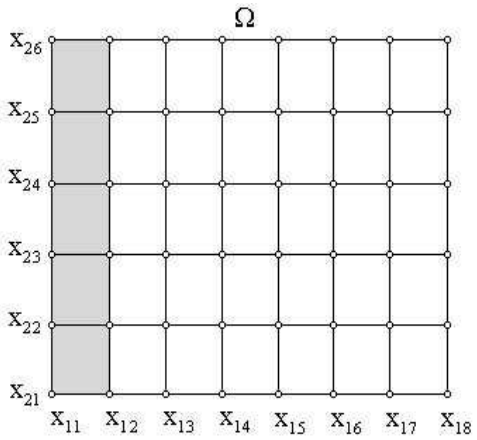

Figure 2: A typical grid on Ω.

theoretical properties of measure. Let consider a simple example in which Ω is covered with 48 nodes as given in Figure 2. If H is the characteristic function on the gray area in the picture, then the measure of H will be the area of band. On the other hand, this measure is equal to $\sum \varrho_k$, where the summation is on the set of all points with x_1^k equaling to x_{11}. So, we have a new constraint preventing the measure to be zero on this area:

$$\sum_{\{k:x_1^k=x_{11}\}} \varrho_k = x_{12} - x_{11}$$

This constraint forces the solution to be nonzero and preserving the properties of measure in the mentioned vertical band. We have to add similar constraints for all vertical and horizontal bands of this picture.

4.1 Extraction of optimal control

After solving the LP problem, we will find the optimal values for decision variables. The form of measure given in (12) is a combination of coefficients ϱ_k and support points z_k. This tel us that, if a coefficient is zero, then the corresponding support does not have a role in the optimal measure. Therefore, nonzero variables will make the optimal measure in a fashion which we discuss here. In fact, we use the effect of this optimal measure to find the approximate solution for the control function. First, define $\gamma_0 = 0$ and

$$\gamma_l = \sum_{k \in A_l} \varrho_k,$$

$$w_l = \frac{1}{|A_l|} \sum_{k \in A_l} w^k,$$

where

$$A_l = \{k : \varrho_k > 0,\ x_1^k = x_{1l}\}, \qquad l = 1, 2, \ldots, n_1.$$

Now the approximation for optimal control function $w(.)$ is a piecewise constant function that equals w_l on each interval $[\gamma_{l-1}, \gamma_l]$. The trajectory is then simply found by solving the differential equation (5) with the initial condition $u(0) = 1$. The resulting solution is a piecewise linear function.

We may also use numerical methods such as interpolation or fitting methods to make a smooth shape. Choosing the convenient method depends on design purposes. A numerical treatment is also given in [12] which may be useful in this connection; see also [1] for numerical smoothing and approximation methods.

5 Numerical example

In this section, a problem with applications in aerodynamic shape optimization is studied to validate the presented method from the computational standpoint.

The velocity $\tilde{v}(\tilde{x})$ at a point $\tilde{x} \in \tilde{\Omega}$ in a nonviscous incompressible potential flow (such as for air or water at moderate speed) may be approximated by

$$\tilde{v}(\tilde{x}) = \nabla \tilde{y}(\tilde{x}), \qquad \tilde{x} \in \tilde{\Omega},$$

where $\tilde{y}$ satisfies the second order partial differential equation on $\tilde{\Omega}$:

$$\Delta \tilde{y} = 0. \tag{13}$$

This type of potential flow occurs mainly in design of low speed airfoils, wings, and blades. The variational form of the above PDE is as follows (see [14]):

$$\int_{\tilde{\Omega}} \nabla \tilde{y} \nabla \tilde{\eta} d\tilde{x} = \int_{\partial \tilde{\Omega}} \tilde{\eta} \nabla \tilde{y}.\vec{n} d\tilde{\Gamma}, \qquad \text{for all } \tilde{\eta} \in \tilde{V}, \tag{14}$$

where $\vec{n}$ is the outward normal vector to $\tilde{\Gamma} = \partial \tilde{\Omega}$.

We also assume the following Dirichlet condition on the fixed boundary part

$$y|_{\tilde{\Gamma}^F} = 0. \tag{15}$$

So, the above variational form is changed to

$$\int_{\tilde{\Omega}} \nabla \tilde{y} \nabla \tilde{\eta} d\tilde{x} = \int_0^1 \tilde{\eta} \tilde{y}_1 d\tilde{x_2}, \qquad \text{for all } \tilde{\eta} \in \tilde{V},$$

where $\tilde{V} = \{\eta \in H^1(\Omega) : \eta|_{\tilde{\Gamma}^F} = 0\}$.

Now, the specific transformation (2) can be used on the variable domain. This leads to the following transformed bilinear form

$$\int_\Omega \left[(\frac{1}{u} + \frac{x_1^2 w^2}{u}) y_1 \eta_1 - x_1 w y_2 \eta_1 - x_1 w y_1 \eta_2 + u y_2 \eta_2 \right] dx$$
$$= \int_0^1 \eta(1, x_2) y_1(1, x_2) \sqrt{1 + w^2} dx_2.$$

The above integral is now defined on the fixed domain Ω, where coefficient functions depend on the parameter function $u \in \mathcal{U}_{ad}$ and its derivative w.

It is easy to check that $\tilde{y} = \sinh(\pi \tilde{x}_1) \sin(\pi \tilde{x}_2)$ is a solution of the boundary value problem (13), (15), or equivalently (14)–(15).

To make a comparison between exact and numerical solution, we follow an inverse scheme to construct an OSD problem with known exact solution. To do this, we choose a cost function as

$$\int_{\tilde{\Gamma}^M} \|\nabla \tilde{y} - V_0\| d\tilde{\Gamma},$$

which leads to a velocity matching problem. In other words, we want to find the moving boundary $\tilde{\Gamma}^M$ in such a way that the velocity of the flow matches, as well as possible, a given velocity V_0. For applications of this type of matching OSD, we may address, for example, the pressure distribution matching in airfoil section design (see [5]).

The above cost function finds the following format after transformation:

$$\int_0^1 \|\nabla y - V_0\| \sqrt{1 + w^2} dx_2$$

or

$$\int_\Omega \|\nabla y - V_0\| \sqrt{1 + w^2} dx. \tag{16}$$

In this example, for test functions of type η, polynomials of x_1 and x_2 with compact support on Ω of the following forms are chosen:

$$x_1^r (1 - x_2)^s, \quad x_2^r (1 - x_1)^s, \qquad r, s = 1, 2, \ldots.$$

For test functions of type φ, we use polynomials of u, functions depending only on x_2, and functions in $C^1(B)$ with compact support on I; see [3, Sec.5] for more details about test functions.

To illustrate the scheme, two specific moving boundaries are implemented:

a: If the moving part of the boundary $\tilde{\Gamma}^M$, is $\tilde{x_1} = 1$, then the the above solution gives a velocity vector as

$$\nabla y = (\pi \cosh(\pi) \sin(\pi \tilde{x}_2), \pi \sinh(\pi) \cos(\pi \tilde{x}_2))$$

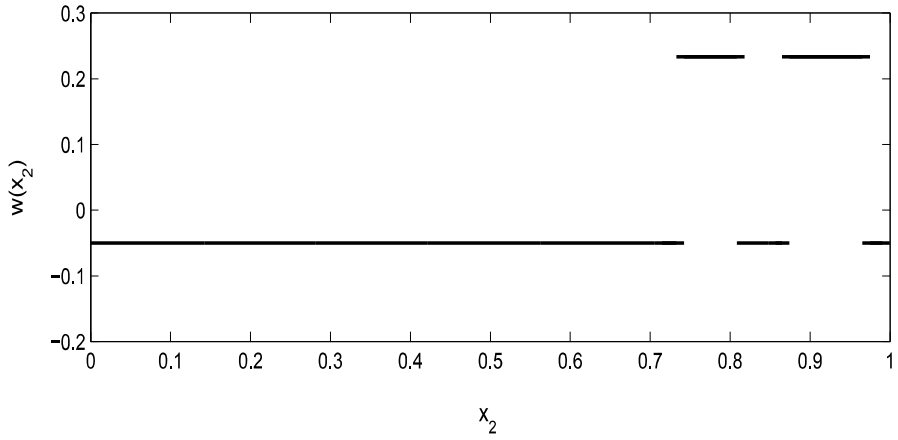

Figure 3: Resulting control function, part a.

on $\tilde{\Gamma}^M$. Now, if we let $V_0 = (\pi\cosh(\pi)\sin(\pi\tilde{x}_2), \pi\sinh(\pi)\cos(\pi\tilde{x}_2))$, then the OSD problem of minimizing (16) subject to (14)–(15) has the unique solution $\tilde{x}_1 = 1$ or equivalently $u = 1$. The solution of the corresponding linear programming problem is used to construct the control function w as shown in Figure 3. The corresponding optimal trajectory, u, is compared with the exact solution in Figure 4. It is clear that the numerical solution approximates the exact solution with a small error.

In this case, we choose $N = 30625$, $\beta_1 = 0.4$, $\beta_2 = 1.3$, $\alpha_1 = -0.8$, $\alpha_2 = 0.9$, $M_1 = 1$, and $M_2 = 12$. The LP problem has been solved by two phase revised simplex method, and the corresponding optimal objective was found as 0.01194.

b: If the moving part of the boundary $\tilde{\Gamma}^M$, is $\tilde{x_1} = 1+\sin(\pi\tilde{x_2})$, then the above solution gives a velocity distribution on $\tilde{\Gamma}^M$ as

$$\nabla y = (\pi\cosh(\pi(1+\sin(\pi\tilde{x_2})))\sin(\pi\tilde{x_2}), \pi\sinh(\pi(1+\sin(\pi\tilde{x_2})))\cos(\pi\tilde{x_2})).$$

Now if we set

$$V_0 = (\pi\cosh(\pi(1+\sin(\pi\tilde{x_2})))\sin(\pi\tilde{x_2}), \pi\sinh(\pi(1+\sin(\pi\tilde{x_2})))\cos(\pi\tilde{x_2})),$$

then the OSD problem of minimizing (16) over (14)–(15) has the unique solution $u = 1+\sin(\pi\tilde{x}_2)$. The solution of the corresponding linear programming problem is used to construct the control function w as shown in Figure 5. To enable a comparison, Figure 6 shows the resulting numerical shape and the exact solution of the problem that indicates the accuracy of results.

In this part of the example, we chose $N = 21875$, $\beta_1 = 0.62$, $\beta_2 = 1$, $\alpha_1 = -1.96$, $\alpha_2 = 1.96$, $M_1 = 1$, and $M_2 = 12$. The optimal objective of the corresponding LP problem was found as 2.1275×10^{-4}. Figure 7 shows the

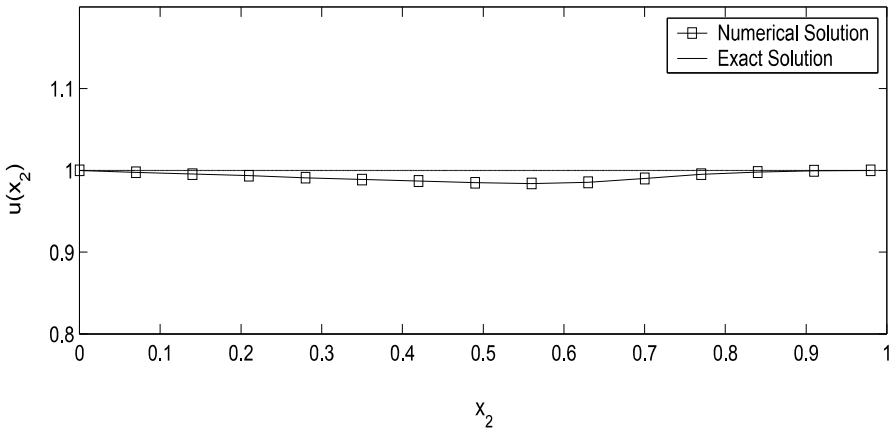

Figure 4: Numerical and exact solution for part a.

decreasing behavior of the objective function and the rate of convergence in the second phase of LP solving.

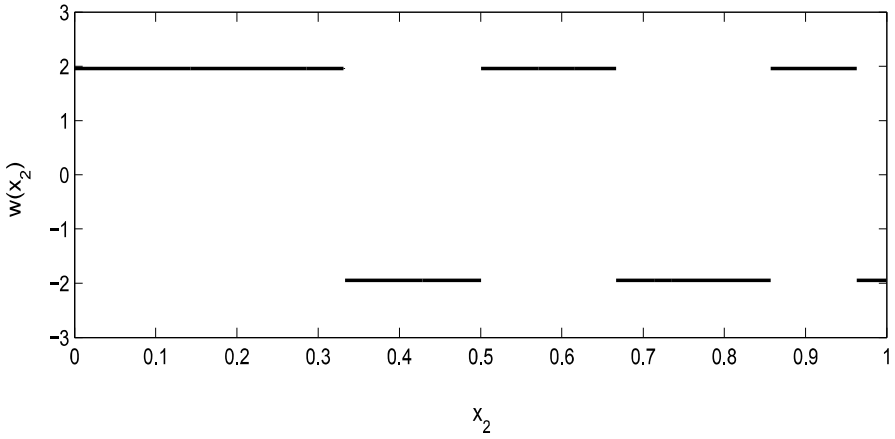

Figure 5: Resulting control function, part b.

6 Conclusions

The method of embedding admissible shapes into a subset of measures is extended to an LP formulation for a class of optimal shape design problems resulting in a numerical solution scheme. Recent works on this subject

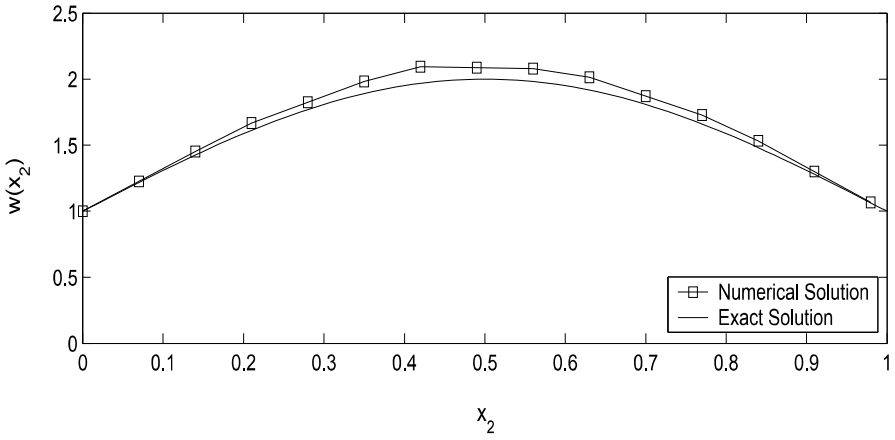

Figure 6: Numerical and exact solution for, part b.

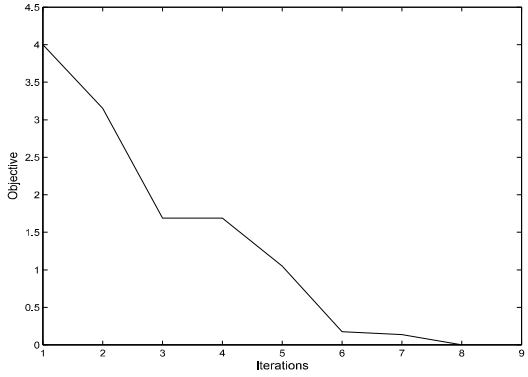

Figure 7: Objective function in iterations of the simplex method.

deal with optimization on a specified known part of the shape, while in the present paper, the optimization takes place on entirely unknown domain. The method is self starting by means that it does not need any initial guess of the solution to be started. It is also independent from type of partial differential equations and has been presented for a relatively general variational form of equations. Numerical examples were used to interpret the implementation of the method and to check its validity.

References

1. Burden R.L. and Faires, J.D. *Numerical Analysis*, (7th Edition), Brooks/Cole (ITP) 2000.

2. De Santis, D., Zahr, M.J. and Farhat, C. *Gradient based aerodynamic shape optimization using the FIVER embedded boundary method*, 54th AIAA Aerospace Sciences Meeting, AIAA SciTech, (2016), 1–11.

3. Farahi, M. H., Borzabadi, A. H., Mehne H. H. and Kamyad, A. V. *Measure theoretical approach for optimal shape design of a nozzle*, J. Appl. Math. Comput. 17 (2005), 315–328.

4. Farahi, M.H., Mehne, H.H. and Borzabadi, A.H. *Wing drag minimization by using measure theory*, Optim. Method. Softw. 21(2) (2006), 169–177.

5. Hacioğlu, A. and Özkol, İ. *Inverse airfoil design by using an accelerated genetic algorithm via distribution strategies*, Inverse. Probl. Sci. En. 13(6) (2005), 563–579.

6. Ghaedamini Harouni, A. and Mehne, H.H. *Multi-Disciplinary Multi-Objective Shape Optimization of Orion Type Re-entry Capsule*, Modares. Mech. Eng. 19(3) (2019), 665–675.

7. Jahromi, A.F. *A shape-measure method for solving free-boundary elliptic systems with boundary control function*, Iran. J. Numer. Anal. Optim. 3(2) (2013), 47–65.

8. Kurahashi, T. *Investigations on boundary temperature control analysis considering a moving body based on the adjoint variable and the fictitious domain finite element methods*, Int. J. Numer. Meth. Eng. 103(8) (2015), 582–597.

9. Laumen, M. *Newton's method for a class of optimal shape design problems*, SIAM J. Optim. 10(2) (2000), 503–533.

10. Li, D. and Hartmann, R. *Adjoint-based airfoil optimization with discretization error control*, Int. J. Numer. Meth. Fl. 77(1) (2015), 1–17.

11. Mehne, H.H. *On solving constrained shape optimization problems for finding the optimum shape of a bar cross-section*, Appl. Numer. Math. 58(8) (2008), 1129–1141

12. Mehne, H.H., Farahi M.H. and Esfahani, J.A. *Slot nozzle design with specified pressure in a given subregion by using embedding method*, Appl. Math. Comput. 168(2) (2005), 1258–1272.

13. Mehne, H.H., Farahi M.H. and Esfahani, J.A. *On an optimal shape design problem to control a thermoelastic deformation under a prescribed thermal treatment*, Appl. Math. Model. 31(4) (2007), 663–675.

14. Mikhailov, V.P. *Partial Differential Equations*, MIR publishers, Moscow 1978.

15. Nguyen, Q. and Yang, C. *Design of a longitudinal cooling fin with minimum volume by a modified Newton–Raphson method*, Appl. Therm. Eng. 98 (2016), 169–178.

16. Rubio, J. E. *Control and Optimization: the linear treatment of non-linear problems*, Manchester University Press, Manchester 1986.

17. Rudin, W. *Real and Complex Analysis* (3rd Edition), McGraw-Hill, New York 1987.

18. Rudin, W. *Functional Analysis* (2ed Edition), McGraw-Hill, New York 1991.

19. Zhang, Z., Liu, J., Tang, Y. and Meng, X. *Optimizing the shape of top piston ring face using inverse method*, Ind. Lubr. Tribol. 68(1) (2016), 9–15.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 17–35
DOI:10.22067/ijnao.v9i1.61735

Approximation of the Huxley equation with nonstandard finite-difference scheme

M. Namjoo* and S. Zibaei

Abstract

In this paper, an explicit exact finite-difference scheme for the Huxley equation is presented based on the nonstandard finite-difference (NSFD) scheme. Afterwards, an NSFD scheme is proposed for the numerical solution of the Huxley equation. The positivity and boundedness of the scheme is discussed. It is shown through analysis that the proposed scheme is consistent, stable, and convergence. The numerical results obtained by the NSFD scheme is compared with the exact solution and some available methods, to verify the accuracy and efficiency of the NSFD scheme.

Keywords: The Huxley equation; Nonstandard finite-difference scheme; Positivity and boundedness; Consistency; Stability; Convergence.

1 Introduction

Nonlinear partial differential equations (PDEs) play an important role in the various fields of physical science and engineering such as plasma physics, fluid mechanics, optimal fibbers, solid state physics, chemical kinetics, and geochemistry. Recently, it also began to become important in various other fields of science, for example, biology and economics [14, 18]. Behaviors of many physical systems encountered in models of reaction mechanisms, convection effects, and diffusion transports give rise to the Burgers–Huxley (BH) equation. Consider the BH equation

*Corresponding author
Received 11 January 2017; revised 25 November 2017; accepted 13 June 2018
M. Namjoo
Department of Mathematics, Vali–e–Asr University of Rafsanjan, Rafsanjan, Iran.
e-mail: namjoo@vru.ac.ir

S. Zibaei
Department of Mathematics, Vali–e–Asr University of Rafsanjan, Rafsanjan, Iran.
e-mail: z.sadegh133@gmail.com

$$u_t + \alpha u u_x - u_{xx} = \beta u(1-u)(u-\gamma), \tag{1}$$

where $\alpha, \beta \geq 0$ and $\gamma \in (0,1)$. When $\beta = 0$, equation (1) reduces to the Burgers equation. If $\alpha = 0$, then Equation (1) recovers the Huxley equation

$$u_t - u_{xx} = \beta u(1-u)(u-\gamma), \tag{2}$$

which describes nerve pulse propagation in nerve fibers and wall motion in liquid crystals. An analytical solution to this equation subject to the following initial and boundary conditions:

$$\begin{aligned} u(x,0) &= \frac{\gamma}{1+e^{-2A_1 x}}, && 0 \leq x \leq 1, \\ u(0,t) &= \frac{\gamma}{1+e^{2A_1 A_2 t}}, && t \geq 0, \\ u(1,t) &= \frac{\gamma}{1+e^{-2A_1(1-A_2 t)}}, && t \geq 0, \end{aligned}$$

derived by Wang in [17], is given by

$$u(x,t) = \frac{\gamma}{2} + \frac{\gamma}{2}\tanh[A_1(x-A_2 t)] = \frac{\gamma}{1+e^{-2A_1(x-A_2 t)}}, \tag{3}$$

where

$$A_1 = \frac{\sqrt{8\beta}}{8}\gamma, \qquad A_2 = \frac{4-2\gamma}{\gamma}A_1.$$

In the past few years, various powerful mathematical methods such as the Adomian decomposition method (ADM) [7, 8, 10], the variational iteration method (VIM) [1, 2], the differential transform method (DTM) [3] and the Haar wavelet method [6] have been used in attempting to solve (2). Among various techniques for solving PDEs, the NSFD schemes have been proved to be one of the most efficient approaches in recent years [13]. These schemes are developed for compensating the weaknesses, such as numerical instabilities that may be caused by standard finite- difference schemes. One of the most important advantages of this scheme is that choosing a complicate denominator function instead of the stepsize h, better results can be obtained. If the stepsize h is chosen small enough, the obtained results do not change significantly but if the stepsize h gets larger, then this advantage comes into focus [13, 19–24].

This paper is organized as follows: In Section 2, we provide a brief overview of the important features of the procedures for constructing NSFD schemes for PDEs. In Section 3, we begin with proposing the exact finite-difference scheme for the Huxley equation. In Section 4, we give an NSFD scheme for numerical solution of the Huxley equation and also positivity and boundedness are studied for the proposed NSFD scheme. Moreover, a brief study of consistency is presented here, along with a Von Neumann analysis

of stability and convergence. Finally, in the last section, numerical results are given to show the efficiency of the proposed NSFD scheme.

2 Nonstandard finite-difference schemes

The initial foundation of NSFD schemes came from the exact finite-difference schemes. NSFD schemes were firstly proposed by Mickens [12,13] for ordinary differential equations (ODEs) and, successively, their use has been investigated in several fields. Regarding the positivity and boundedness of solutions, NSFD schemes have a better performance over the standard finite-difference schemes, due to its flexibility to construct an NSFD scheme that can preserve certain properties and structures, which are obeyed by the original equations. The advantages of NSFD schemes have been shown in many numerical applications. Gonzalez-Parra et al. [5] and Zibaei and Namjoo [20–23] developed NSFD schemes to solve population and biological models.

This class of schemes and their formulations center on two issues. First, how should discrete representations be determined for derivatives, and second, what are the proper forms to be used for nonlinear terms. The forward Euler method is one of the simplest discretization schemes. In this method the derivative term $\frac{dy}{dt}$ is replaced by $\frac{y(t+h)-y(t)}{h}$, where h is the stepsize. However, in the Mickens schemes this term is replaced by $\frac{y(t+h)-y(t)}{\phi(h)}$, where $\phi(h)$ is an increasing continuous function of h, and the function $\phi(h)$ satisfies the following conditions:

$$\phi(h) = h + O(h^2), \qquad 0 < \phi(h) < 1, \qquad h \to 0.$$

Examples of functions $\phi(h)$ that satisfy these conditions are [12, 19–24]:

$$h, \qquad \sin h, \qquad \frac{1 - e^{-\lambda h}}{\lambda}.$$

Note that in taking the $\lim h \to 0$ to obtain the derivative, the use of any of these $\phi(h)$ will lead to the usual result for the first derivative

$$\frac{dy}{dt} = \lim_{h\to 0} \frac{y\big(t + \phi_1(h)\big) - y(t)}{\phi_2(h)} = \lim_{h\to 0} \frac{y(t+h) - y(t)}{h},$$

where $\phi_1(h)$ and $\phi_2(h)$ are continuous functions of the stepsize h. A scheme is called nonstandard if at least one of the following conditions is satisfied:

1. A nonlocal approximation is used.
2. The denominator function for the discrete derivative, in general, expressed in terms of more complicated functions of the stepsize h than those conventionally used.

Nonlinear terms approximated in a nonlocal way, for example, if there are nonlinear terms in the differential equation [12,19–24], these are replaced by

$$y^2(t) \to y_n(t)y_{n+1}(t), \qquad y^2(t) \to y_{n-1}(t)y_n(t).$$

In the two-dimensional case, nonlinear terms such as $y(t)x(t)$ are either replaced by

$$y(t)x(t) \to y_n(t)x_{n+1}(t), \qquad y(t)x(t) \to y_{n+1}(t)x_n(t).$$

One can say that there is no appropriate general method to choose the function $\phi(h)$ or to choose which nonlinear terms are to be replaced, some special techniques may be found in [15,19–24].

3 An explicit exact finite-difference scheme

In this section, we obtain an exact finite-difference scheme for the Huxley equation. If we choose $h = A_2\Delta t$, then we easily obtain $u(x-h,t) = u(x,t+\Delta t)$, and the following equations

$$\begin{aligned}
\frac{1}{u(x+h,t)} &= \frac{1}{\gamma}(1+e^{-2A_1(x+h-A_2t)}),\\
\frac{1}{u(x-h,t)} &= \frac{1}{\gamma}(1+e^{-2A_1(x-h-A_2t)}),\\
\frac{1}{u(x,t-\Delta t)} &= \frac{1}{\gamma}(1+e^{-2A_1(x-A_2(t-\Delta t))}).
\end{aligned} \tag{4}$$

According to (4), we can write

$$\begin{aligned}
\frac{1}{u(x,t)} - \frac{1}{u(x+h,t)} &= (\frac{1}{u(x,t)} - \frac{1}{\gamma})(1-e^{-2A_1h}),\\
\frac{1}{u(x,t)} - \frac{1}{u(x-h,t)} &= (\frac{1}{u(x,t)} - \frac{1}{\gamma})(1-e^{2A_1h}),\\
\frac{1}{u(x,t)} - \frac{1}{u(x,t-\Delta t)} &= (\frac{1}{u(x,t)} - \frac{1}{\gamma})(1-e^{-2A_1A_2\Delta t}).
\end{aligned}$$

If we set the stepsize functions as

$$\psi_1 = \frac{1-e^{-2A_1h}}{2A_1}, \qquad \psi_2 = \frac{e^{2A_1h}-1}{2A_1}, \qquad \phi_1 = \frac{1-e^{-2A_1A_2\Delta t}}{2A_1A_2},$$

we will get the following forward and backward difference quotients with the special stepsize functions

$$\begin{aligned}
\partial_x u = &\quad \frac{u(x+h,t)-u(x,t)}{\psi_1} = 2A_1 u(x+h,t)(1-\frac{u(x,t)}{\gamma}),\\
\bar{\partial}_x u = &\quad \frac{u(x,t)-u(x-h,t)}{\psi_2} = 2A_1 u(x-h,t)(1-\frac{u(x,t)}{\gamma}),\\
\bar{\partial}_t u = &\ \frac{u(x,t)-u(x,t-\triangle t)}{\phi_1} = 2A_1A_2 u(x,t-\Delta t)(\frac{u(x,t)}{\gamma}-1).
\end{aligned} \tag{5}$$

Consequently, it follows that

$$\begin{aligned}
\bar{\partial}_x\partial_x u &= \frac{\bar{\partial}_x u(x+h,t)-\bar{\partial}_x u(x,t)}{\psi_1}\\
&= \frac{2A_1}{\gamma}u(x,t)\frac{u(x-h,t)-u(x+h,t)}{\psi_1} + 2A_1\frac{u(x,t)-u(x-h,t)}{\psi_1}\\
&= -A_2\frac{u(x,t)-u(x-h,t)}{\psi_1} + \frac{2A_1}{\gamma}u(x,t)\frac{u(x-h,t)-u(x+h,t)}{\psi_1}\\
&\quad +(2A_1+A_2)\frac{u(x,t)-u(x-h,t)}{\psi_1}.
\end{aligned} \tag{6}$$

Since $\frac{A_2}{\psi_1}=\frac{1}{\phi_1}$, according to (5), (6), and $u(x-h,t)=u(x,t+\Delta t)$, we can write

$$\begin{aligned}
\bar{\partial}_x\partial_x u &= \frac{u(x,t+\Delta t)-u(x,t)}{\phi_1} + \frac{2A_1}{\gamma}u(x,t)\big(\frac{u(x-h,t)-u(x,t)}{\psi_1}\\
&\quad +\frac{u(x,t)-u(x+h,t)}{\psi_1}\big) + (2A_1+A_2)\frac{u(x,t)-u(x-h,t)}{\psi_1}\\
&= \frac{u(x,t+\Delta t)-u(x,t)}{\phi_1} + \frac{2A_1}{\gamma}u(x,t)\big(-2A_1u(x,t)(1-\frac{u(x-h,t)}{\gamma})\\
&\quad -2A_1u(x+h,t)(1-\frac{u(x,t)}{\gamma})\big) + (2A_1+A_2)2A_1u(x,t)(1-\frac{u(x-h,t)}{\gamma}).
\end{aligned} \tag{7}$$

We notice that

$$2A_1\frac{2A_1}{\gamma}=\frac{\beta\gamma}{2}, \qquad 2A_1(2A_1+A_2)=\beta\gamma,$$

hence according to (7), we have

$$\begin{aligned}
\bar{\partial}_x\partial_x u &= \frac{u(x,t+\Delta t)-u(x,t)}{\phi_1} + \beta(u(x,t))^2\frac{u(x+h,t)+u(x-h,t)}{2}\\
&\quad -\beta\gamma u(x,t)\frac{u(x,t)+u(x+h,t)}{2} + \beta\gamma u(x,t) - \beta u(x,t)u(x-h,t).
\end{aligned}$$

Now, using approximation $u_{xx}\approx\bar{\partial}_x\partial_x u$ and also assuming

$$U_j^n = u(x_j, t_n) = \frac{\gamma}{1 + e^{-2A_1(x_j - A_2 t_n)}},$$

we construct the following explicit exact finite-difference scheme:

$$\begin{aligned}\frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\psi_1 \psi_2} = \frac{U_j^{n+1} - U_j^n}{\phi_1} + \beta (U_j^n)^2 \frac{U_{j+1}^n + U_{j-1}^n}{2} \\ -\beta\gamma U_j^n \frac{U_j^n + U_{j+1}^n}{2} - \beta U_j^n U_{j-1}^n + \beta\gamma U_j^n.\end{aligned} \tag{8}$$

The following theorem is the main result of the previous discussion.

Theorem 1. *An explicit exact finite-difference scheme for the Huxley equation (2) is given by (8). The stepsize satisfies $h = A_2 \Delta t$, and the stepsize functions satisfy*

$$\psi_1 = \frac{1 - e^{-2A_1 h}}{2A_1}, \qquad \psi_2 = \frac{e^{2A_1 h} - 1}{2A_1}, \qquad \phi_1 = \frac{1 - e^{-2A_1 A_2 \Delta t}}{2A_1 A_2}.$$

4 Analysis of NSFD scheme

In this section, we present a NSFD scheme for the Huxley equation. In the classical sense, a special difference scheme of the Huxley equation can be written as

$$\frac{U_j^{n+1} - U_j^n}{\Delta t} - \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{h^2} = \beta\Big(- (U_j^n)^3 + (1+\gamma)(U_j^n)^2 - \gamma U_j^n \Big), \tag{9}$$

where h and Δt are space and time stepsizes, respectively; and U_j^n is an approximation to $u(x_j, t_n)$. Similar to the classical difference scheme (9), we obtain a NSFD scheme for the Huxley equation as follows:

$$\frac{U_j^{n+1} - U_j^n}{\Phi} - \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Psi} = \beta\big(- U_j^{n+1}(U_j^n)^2 + (1+\gamma) U_j^{n+1} U_j^n - \gamma U_j^n \big) \tag{10}$$

where the functions Φ and Ψ are given by

$$\Phi = \frac{1 - e^{-2A_1 A_2 \Delta t}}{2A_1 A_2}, \qquad \Psi = \Big(\frac{1 - e^{-2A_1 h}}{2A_1}\Big)^2.$$

We conclude that $\Phi \to \Delta t$ and $\Psi \to h^2$ as Δt and h approach zero.

Comparing (10) with (2), we infer that the nonlinear terms on the right-hand side of (2) are in the forms

$$\big(u(x_j, t_n)\big)^3 \approx (U_j^{n+1})(U_j^n)^2, \qquad \big(u(x_j, t_n)\big)^2 \approx U_j^{n+1} U_j^n.$$

It can be easily noticed that the NSFD scheme (10) is explicit. Setting $R = \frac{\Phi}{\Psi}$, we can rewrite (10) as

$$U_j^{n+1} = \frac{(1-2R)U_j^n + R(U_{j+1}^n + U_{j-1}^n) - \beta\gamma\Phi U_j^n}{1 - \beta(1+\gamma)\Phi U_j^n + \beta\Phi(U_j^n)^2}. \tag{11}$$

We can find positivity and boundedness properties in the presented NSFD scheme. The following theorem shows these properties.

Theorem 2. *For any nonnegative initial data, if* $1 - 2R - \beta\Phi\gamma \geq 0$, *then the numerical solution* (11) *satisfies*

$$0 < U_j^n < \gamma \Longrightarrow 0 < U_j^{n+1} < \gamma$$

for all relevant values of n *and* j.

Proof. Since $1 - 2R - \beta\Phi\gamma \geq 0$ and $U_j^n > 0$, we deduce that

$$\begin{aligned}(1-2R)U_j^n + R(U_{j+1}^n + U_{j-1}^n) - \beta\gamma\Phi U_j^n \\ = (1 - 2R - \beta\gamma\Phi)U_j^n + R(U_{j+1}^n + U_{j-1}^n) > 0.\end{aligned} \tag{12}$$

Moreover, $\beta\gamma\Phi \leq 1 - 2R \leq 1$ and $U_j^n < \gamma < 1$ implies

$$\begin{aligned}1 - \beta(1+\gamma)\Phi U_j^n + \beta\Phi(U_j^n)^2 &\geq \beta\gamma\Phi - \beta(1+\gamma)\Phi\gamma + \beta\Phi(U_j^n)^2 \\ &= \beta\Phi(U_j^n - 1)(U_j^n - \gamma) > 0.\end{aligned} \tag{13}$$

So the inequalities (12) and (13) imply that $U_j^{n+1} > 0$. Consider

$$\begin{aligned}&(1 - 2R - \beta\gamma\Phi)U_j^n + R(U_{j+1}^n + U_{j-1}^n) - \gamma\big(1 - \beta(1+\gamma)\Phi U_j^n + \beta\Phi(U_j^n)^2\big) \\ &\leq (1 - 2R - \beta\gamma\Phi)\gamma + 2R\gamma - \gamma + \beta\gamma(1+\gamma)\Phi U_j^n - \beta\gamma\Phi(U_j^n)^2 \\ &= -\beta\gamma\Phi\big(\gamma - (1+\gamma)U_j^n + (U_j^n)^2\big) = -\beta\gamma\Phi\big((U_j^n - \gamma)(U_j^n - 1)\big) < 0,\end{aligned}$$

and the last inequality shows that $U_j^{n+1} < \gamma$. Hence, the solution of NSFD scheme (11) has the positivity and boundedness properties. □

Now, we establish the properties of consistency and stability of the NSFD scheme (11).

For appropriate R, setting $u_j^n = u(x_j, t_n)$ precisely, we have Taylor's formula for the solution of the Huxley equation, with appropriate $\bar{x}_j \in (x_j, x_{j+1})$ and $\bar{t}_n, \bar{\bar{t}}_n, \bar{\bar{\bar{t}}}_n \in (t_n, t_{n+1})$. For functions defined on the grid, we introduce these difference quotients

$$\partial_t u_j^n = \frac{U_j^{n+1} - U_j^n}{\Phi}, \qquad \partial_x \bar{\partial}_x u_j^n = \frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Psi}.$$

By using the method in [4], the local truncation error τ_j^n is shown as follows:

$$\begin{aligned}
\tau_j^n &= \Big(\partial_t u_j^n - u_t(x_j,t_n)\Big) - \Big(\partial_x\bar{\partial}_x u_j^n - u_{xx}(x_j,t_n)\Big) + \beta\gamma\Big(u_j^n - u(x_j,t_n)\Big)\\
&\quad -\beta(1+\gamma)\Big(u_j^{n+1}u_j^n - u^2(x_j,t_n)\Big) + \beta\Big(u_j^{n+1}(u_j^n)^2 - u^3(x_j,t_n)\Big)\\
&= (\tfrac{\Delta t}{\Phi}-1)u_t(x_j,t_n) + \tfrac{\Delta t^2}{2\Phi}u_{tt}(x_j,t_n) + \tfrac{\Delta t^3}{6\Phi}u_{ttt}(x_j,\bar{t}_n)\\
&\quad -\Big((\tfrac{h^2}{\Psi}-1)u_{xx}(x_j,t_n) + \tfrac{h^4}{12\Psi}u_{xxx}(\bar{x}_j,t_n)\Big)\\
&\quad -\beta(1+\gamma)\Big(\Delta t u(x_j,t_n)u_t(x_j,t_n) + \tfrac{\Delta t^2}{2}u(x_j,t_n)u_{tt}(x_j,t_n)\\
&\qquad\qquad + \tfrac{\Delta t^3}{3}u(x_j,t_n)u_{ttt}(x_j,\bar{\bar{t}}_n)\Big)\\
&\quad +\beta\Big(\Delta t u^2(x_j,t_n)u_t(x_j,t_n) + \tfrac{\Delta t^2}{2}u^2(x_j,t_n)u_{tt}(x_j,t_n)\\
&\qquad + \tfrac{\Delta t^3}{3}u^2(x_j,t_n)u_{ttt}(x_j,\bar{\bar{\bar{t}}}_n)\Big).
\end{aligned}$$

When $\Delta t \to 0$ and $h \to 0$, we have $\Phi \approx \Delta t$ and $\Psi \approx h^2$. Therefore, $\tau_j^n = O(\Delta t) + O(h^2)$ if $\Delta t \to 0$ and $h \to 0$. This shows that the NSFD scheme (11) is consistent with the Huxley equation. In [24], Namjoo et al. considered the approximation of the BH equation using an NSFD scheme, and investigated their numerical results. In fact this scheme is consistent for the BH equation with a consistency of the order of $O(\Delta t) + O(h)$.

Von Neumann introduced a method to prove stability, using Fourier analysis, so that it can give a sufficient condition for the stability of finite-difference schemes.

Theorem 3. *Sufficient conditions for the NSFD scheme* (11) *to be stable, are that* $\gamma \in (0,1)$ *and* $\beta\Psi(1+\gamma) \leqslant 2$.

Proof. Let $U_j^n = e^{\alpha nk}e^{-ij\lambda h}$, for every $n \in \{0,1,2,\ldots,N\}$ and every $j \in \{0,1,2,\ldots,M\}$ where α is a real constant. Substituting in the NSFD scheme (11) and simplifying, we get

$$e^{\alpha k} = \frac{(1-2R) + R(e^{-i\lambda h} + e^{i\lambda h}) - \beta\gamma\Phi}{1-\beta(1+\gamma)\Phi e^{n\alpha k - ij\lambda h} + \beta\Phi e^{2(n\alpha k - ij\lambda h)}},$$

so we deduce that

$$\begin{aligned}
&\beta\gamma\Phi + e^{\alpha k} - \beta(1+\gamma)\Phi e^{(n+1)\alpha k}\Big(\cos(j\lambda h) - i\sin(j\lambda h)\Big)\\
&+\beta\Phi e^{(2n+1)\alpha k}\Big(\cos(2j\lambda h) - i\sin(2j\lambda h)\Big) = 1 - 4R\sin^2(\tfrac{\lambda h}{2}).
\end{aligned} \tag{14}$$

Expanding (14) into separate real and imaginary parts, we obtain the following identities:

$$\beta(1+\gamma)\Phi e^{(n+1)\alpha k}\sin(j\lambda h) - \beta\Phi e^{(2n+1)\alpha k}\sin(2j\lambda h) = 0, \tag{15}$$

$$\begin{aligned}\beta\gamma\Phi + e^{\alpha k} - \beta(1+\gamma)\Phi e^{(n+1)\alpha k}\cos(j\lambda h) + \beta\Phi e^{(2n+1)\alpha k}\cos(2j\lambda h)\\ = 1 - 4R\sin^2(\tfrac{\lambda h}{2}).\end{aligned} \tag{16}$$

Therefore, from (15), we have

$$e^{\alpha k} = (\frac{1+\gamma}{2\cos(j\lambda h)})^{\frac{1}{n}}.$$

Hence, according to the Fourier analysis, the NSFD scheme (11) is stable in the sense of Von Neumann if $|e^{\alpha k}| \leqslant 1$, or equivalently:

$$-1 \leqslant \frac{1+\gamma}{2\cos(j\lambda h)} \leqslant 1. \tag{17}$$

In order to show inequalities (17), we need to consider the following two cases.
Case1. Assume that $\cos(j\lambda h) > 0$, to prove (17), it is sufficient to show that

$$\frac{1+\gamma}{2\cos(j\lambda h)} \leqslant 1. \tag{18}$$

Since $\cos(j\lambda h) > 0$, there exists a number γ^*, such that

$$0 < \gamma^* < \cos(j\lambda h) \leqslant 1. \tag{19}$$

Now pick a number $\gamma \in (0,1)$ such that $\gamma^* = \frac{1+\gamma}{2}$. Substituting for γ^* in (19), we obtain the desired result, (18).
Case 2. Now suppose that $\cos(j\lambda h) < 0$. To prove (17), it is sufficient to show that

$$-1 \leqslant \frac{1+\gamma}{2\cos(j\lambda h)}. \tag{20}$$

Since $\cos(j\lambda h) < 0$, there exists a number γ^{**}, such that

$$-1 \leqslant \cos(j\lambda h) < \gamma^{**} < 0. \tag{21}$$

Choose a number $\gamma \in (0,1)$ such that

$$\gamma^{**} = -\frac{(1+\gamma)}{2}. \tag{22}$$

Substituting (22) into (21), we get the desired result (20).

On the other hand, by using the following inequality

$$\left|1-4R\sin^2(\frac{\lambda h}{2})\right| \leqslant 1+4R,$$

and the fact that $|e^{\alpha k}| \leqslant 1$, from (16), it follows that

$$\left|1-4R\sin^2(\frac{\lambda h}{2})\right| = \Big|\beta\gamma\Phi + e^{\alpha k} - \beta(1+\gamma)\Phi e^{(n+1)\alpha k}\cos(j\lambda h)$$
$$+\beta\Phi e^{(2n+1)\alpha k}\cos(2j\lambda h)\Big| \leqslant \beta\gamma\Phi + 1 + \beta(1+\gamma)\Phi + \beta\Phi = 1 + 2\beta\gamma\Phi + 2\beta\Phi.$$

Now, by imposing the condition

$$1 + 2\beta\gamma\Phi + 2\beta\Phi \leqslant 1 + 4R,$$

we obtain the following inequality

$$\frac{\beta\Phi(1+\gamma)}{2} \leqslant R,$$

where $R = \frac{\Phi}{\Psi}$. Consequently, the sufficient conditions for stability the NSFD scheme (11) are given as follows:

$$\beta\Psi(1+\gamma) \leqslant 2, \qquad \gamma \in (0,1). \tag{23}$$

□

Corollary 1. *The NSFD scheme* (11) *is convergent for* $\gamma \in (0,1)$ *and* $\beta\Psi(1+\gamma) \leqslant 2$ *with the Huxley equation* (2).

Proof. The result follows from Theorem 3 and the Lax theorem. □

5 Numerical results

To verify the effectivity of the NSFD scheme in section 4, we simulate initial-boundary value problem

$$\begin{aligned} u_t - u_{xx} &= \beta u(1-u)(u-\gamma), \qquad 0 \le x \le 1, \qquad t \ge 0, \\ u(x,0) &= \frac{\gamma}{1+e^{-2A_1 x}}, \qquad 0 \le x \le 1, \\ u(0,t) &= \frac{\gamma}{1+e^{2A_1 A_2 t}}, \qquad t \ge 0, \\ u(1,t) &= \frac{\gamma}{1+e^{-2A_1(1-A_2 t)}}, \qquad t \ge 0. \end{aligned} \tag{24}$$

We use the NSFD scheme (11); then we give the initial and boundary conditions

$$\begin{aligned} U_j^0 &= \frac{\gamma}{1+e^{-2A_1 x_j}}, \qquad j = 0, 1, \ldots, M, \\ U_0^n &= \frac{\gamma}{1+e^{2A_1 A_2 t_n}}, \qquad n = 0, 1, \ldots, N, \\ U_M^n &= \frac{\gamma}{1+e^{-2A_1(x_M - A_2 t_n)}}, \qquad n = 0, 1, \ldots, N. \end{aligned} \tag{25}$$

To verify the efficiency and to measure its accuracy and the versatility of the NSFD scheme (11) for the current problem in comparison with the exact solution, absolute error for different values of β and γ is reported which is defined by

$$\big|u(x_j, t_n) - U(x_j, t_n)\big|,$$

in the point (x_j, t_n). Here $U(x_j, t_n)$ as the solution portraying the behaviors of physical system is obtained by the present scheme while $u(x_j, t_n)$ stands for the exact solution. Consider the Huxley equation in the form (24) with the initial and boundary conditions (25), and the exact solution (3). The results are compared with the exact solution. The numerical computations were performed by using uniform grids. The differences between the computed solution and the exact solution for some values of the constants β and γ are shown in Tables 1–4. As various problems of science were modeled by the Huxley equation, hence various values of the parameters have been considered in the following examples. In all numerical examples reported in this section, the Von Neumann stability conditions (23) are fulfilled for the NSFD scheme (11).

Example 1. In Table 1, the absolute errors have been shown for various values of β, x, and t with $\gamma = 0.001$. A comparison between the NSFD scheme [24] and the NSFD scheme (11) is given in Table 1. The numerical results show the high accuracy of the NSFD scheme (11).

Example 2. Table 2 shows the absolute errors for various values of γ, x, and t with $\beta = 10$. The results of the NSFD scheme (11) and the NSFD scheme [24] for different values of the parameters are shown in Table 2. Comparison of

Table 1: The absolute errors for various values of β, x, and t with $\gamma = 0.001$

x	t	NSFD [24] for $\beta = 1$	Presented NSFD for $\beta = 1$	NSFD [24] for $\beta = 10$	Presented NSFD for $\beta = 10$
	0.3	1.3158E–12	7.4809E–18	3.9633E–11	9.3750E–16
0.2	0.5	1.3753E–12	7.5894E–18	4.1390E–11	9.5875E–16
	0.9	1.3846E–12	7.6978E–18	4.1589E–11	9.1983E–16
	0.3	2.0465E–12	1.1384E–17	6.1653E–11	1.4631E–15
0.5	0.5	2.1477E–12	1.2034E–17	6.4648E–11	1.5020E–15
	0.9	2.1635E–12	1.2143E–17	6.4999E–11	1.4430E–15
	0.3	1.3159E–12	7.4809E–18	3.9651E–11	9.4271E–16
0.8	0.5	1.3755E–12	7.5894E–18	4.1408E–11	9.6363E–16
	0.9	1.3847E–12	7.9146E–18	2.3405E–11	9.2525E–16

current and exact results shows the efficiency and accuracy of the NSFD scheme (11).

Table 2: The absolute errors for various values of γ, x, and t with $\beta = 10$

x	t	NSFD [24] for $\gamma = 10^{-2}$	Presented NSFD for $\gamma = 10^{-2}$	NSFD [24] for $\gamma = 10^{-5}$	Presented NSFD for $\gamma = 10^{-5}$
	0.2	2.0599E–08	4.5578E–12	2.0529E–17	6.7762E–21
0.1	0.4	2.2927E–08	4.0248E–12	2.3054E–17	5.9292E–21
	0.6	2.3035E–08	2.8500E–12	2.3408E–17	7.6232E–21
	0.2	5.6063E–08	1.3031E–11	5.5677E–17	1.1011E–20
0.5	0.4	6.3727E–08	1.1893E–11	6.3850E–17	1.5246E–20
	0.6	6.4195E–08	8.6771E–12	6.4996E–17	2.1175E–20
	0.2	2.0724E–08	4.9406E–12	2.0527E–17	1.6940E–21
0.9	0.4	2.3063E–08	4.4080E–12	2.3053E–17	5.0821E–21
	0.6	2.3172E–08	3.2331E–12	2.3407E–17	9.3173E–21

Example 3. In Table 3 the absolute errors have been shown for various values of x, t, and γ with $\beta = 50$. The numerical results obtained by the NSFD scheme (11) show good accuracies and agreements with exact solutions.

Example 4. Table 4 shows the absolute errors for various values of x, t, and γ with $\beta = 100$. The results of the present scheme and the NSFD scheme [24] for the above values of the parameters are shown in Table 4. The numerical results reported in Table 4 indicate the high accuracy of the NSFD scheme (11).

Table 3: The absolute errors for various values of γ, x, and t with $\beta = 50$

x	t	NSFD [24] for $\gamma = 10^{-3}$	Presented NSFD for $\gamma = 10^{-3}$	NSFD [24] for $\gamma = 10^{-5}$	Presented NSFD for $\gamma = 10^{-5}$
	0.1	5.3170E–12	9.2720E–15	1.5906E–16	7.6232E–21
0.1	0.5	2.3504E–10	1.2583E–14	2.3744E–16	1.1858E–20
	1	2.3317E–10	1.2682E–14	2.3895E–16	9.3173E–21
	0.1	5.3230E–12	2.3789E–14	4.0521E–16	1.9481E–20
0.5	0.5	6.5311E–10	3.5184E–14	6.5885E–16	2.7952E–20
	1	6.4861E–10	3.0349E–14	6.6378E–16	2.8799E–20
	0.1	5.3289E–12	9.3692E–15	1.5906E–16	8.4703E–21
0.9	0.5	2.3553E–10	1.2682E–14	2.3745E–16	7.6232E–21
	1	2.3366E–10	1.0925E–14	2.3896E–16	1.1011E–20

Table 4: The absolute errors for various values of γ, x, and t with $\beta = 100$

x	t	NSFD [24] for $\gamma = 10^{-3}$	Presented NSFD for $\gamma = 10^{-3}$	NSFD [24] for $\gamma = 10^{-4}$	Presented NSFD for $\gamma = 10^{-4}$
	0.2	1.2498E–09	1.0843E–13	1.2564E–12	1.1004E–17
0.3	0.5	1.4195E–09	1.1179E–13	1.4482E–12	1.2590E–17
	0.8	1.4027E–09	9.7739E–14	1.4561E–12	1.2529E–17
	0.2	1.4802E–09	1.2868E–13	1.4865E–12	1.3030E–17
0.5	0.5	1.6913E–09	1.3352E–13	1.7241E–12	1.4989E–17
	0.8	1.6718E–09	1.1683E–13	1.7336E–12	1.4928E–17
	0.2	1.2522E–09	1.0890E–13	1.2566E–12	1.1018E–17
0.7	0.5	1.4222E–09	1.1226E–13	1.4488E–12	1.2590E–17
	0.8	1.4053E–09	9.8212E–14	1.4563E–12	1.2549E–17

Comparing the numerical results in Tables 1–4, it follows that the NSFD scheme (11) is more accurate than the NSFD scheme [24]. This follows because the local truncation error the NSFD scheme (11) is $O(\Delta t) + O(h^2)$, while the local truncation error the NSFD scheme [24] is $O(\Delta t) + O(h)$.

Figures 1 and 2 indicate the numerical solutions and the solitary wave solutions. Figures 1(a) and 11(b) compare the numerical results with the exact one for $\beta = 1$ and $\gamma = 0.001$ with stepsize $\Delta t = 0.001$ and $h = 0.1$ for a given fixed value of $x = 0.5$. Figures 2(a) and 2(b) show the error between two solutions of different formats. The two simulations show that the presented NSFD scheme is efficient. From the numerical results of this example, we conclude that the obtained results quite agreed with the exact one.

In Figures 3(a)–(d), we simulate absolute errors between the exact solution and the NSFD scheme for various stepsizes. In fact, we have compared absolute errors when $x = 0.5$, for two values of β and γ. It is noticeable

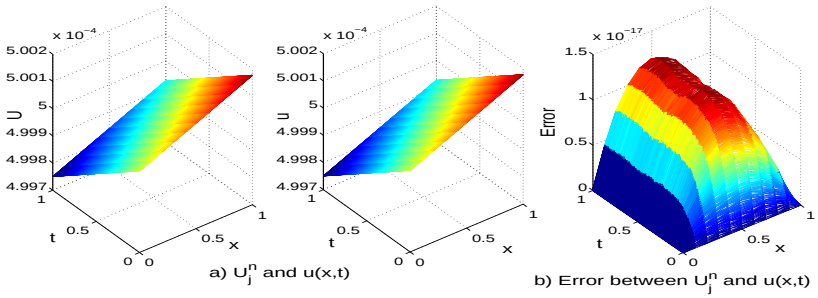

Figure 1: Simulations of the NSFD scheme for the Huxley equation with stepsize $\Delta t = 0.001$, $h = 0.1$.

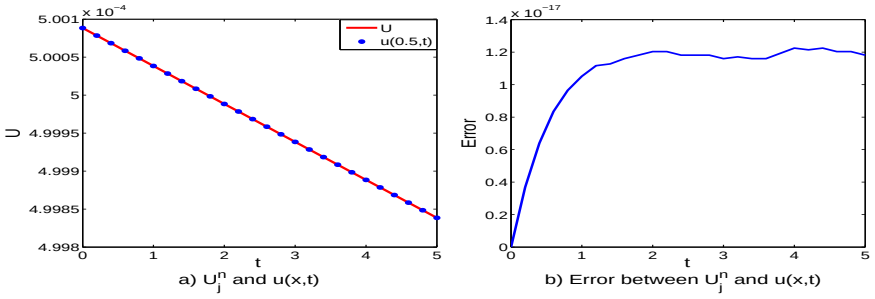

Figure 2: U and $u(x,t)$ at a fixed value $x = 0.5$ for the NSFD scheme.

that increasing stepsizes do not decrease accuracy of the scheme significantly. These comparisons show that the NSFD scheme (11) has good efficiency even for a high space stepsize.

In Table 5, we present the absolute errors of the computed solution for the Huxley equation, by ADM [7,8,10], VIM [1,2], DTM [3], Crank–Nicolson (CN) and NSFD scheme [24], with the NSFD scheme (11). Observing the numerical results, we conclude that the NSFD scheme (11) presents remarkably accurate in comparison with the other methods. In Tables 6 and 7,

Table 5: The absolute errors for $\beta = 1$ and $\gamma = 0.01$

x	t	ADM	VIM	DTM	CN	NSFD [24]	Presented NSFD
	0.05	2.4852E–06	2.4874E–06	2.4875E–06	3.7048E–07	3.5421E–10	2.6543E–14
0.1	0.1	4.9727E–06	4.9749E–06	4.9749E–06	4.0679E–07	5.2191E–10	3.9261E–14
	1 .0	4.9743E–05	4.9749E–05	4.9749E–05	4.4552E–07	7.8134E–10	4.3048E–14
	0.05	2.4763E–06	2.4875E–06	2.4874E–06	5.5746E–08	7.9198E–10	6.3115E–14
0.5	0.1	4.9638E–06	4.9749E–06	4.9749E–06	1.2894E–07	1.3305E–09	1.0499E–13
	1.0	4.9738E–05	4.9749E–05	4.9749E–05	2.4749E–07	2.1718E–09	1.2556E–13
	0.05	2.4673E–06	2.4874E–06	2.4874E–06	2.1847E–09	3.5492E–10	2.9874E–14
0.9	0.1	4.9548E–06	4.9749E–06	4.9749E–06	1.4873E–08	5.2275E–10	4.2973E–14
	1.0	4.9728E–05	4.9788E–05	4.9749E–05	4.9497E–08	7.8230E–10	4.6822E–14

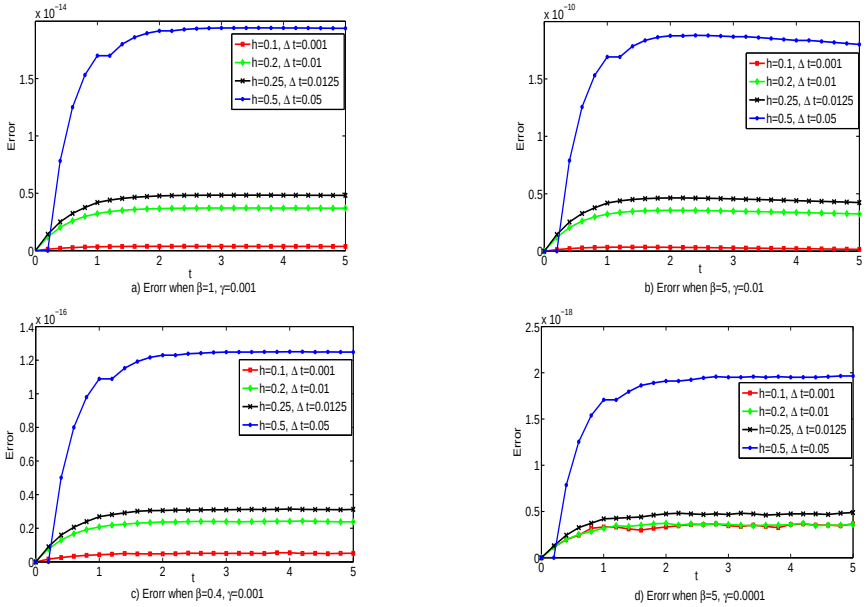

Figure 3: The absolute errors between $u(x_j, t_n)$ and U_j^n for various space and time stepsizes.

the numerical results of the presented NSFD scheme are compared with the NSFD scheme [24] and MCB–DQM, FDS4, ADM, OHAM, GCG, ECG, ELG methods in [16]. These comparisons show that the presented NSFD scheme for various values of x and t with $\beta = 1$ and $\gamma = 0.001$ offers better results than the others.

Table 8 shows comparison of the presented NSFD scheme with explicit exponential finite difference method (EFDM) in [9] for various values of x and t with $\beta = 1$ and $\gamma = 0.001$. According to the results presented in this table, the presented NSFD scheme offers high accuracy for the numerical solutions of the Huxley equation.

In Table 9, the numerical results of the meshless RBFs method in [11] are compared with the presented NSFD scheme for various values of x and t with $\gamma = 0.001$ and $\beta = 1$. The numerical results presented guaranteed the effectiveness of the presented NSFD scheme.

Table 6: Absolute errors for $\beta = 1$ and $\gamma = 0.001$

t	x	MCB-DQM	FDS4	ADM	OHAM	Presented NSFD
	0.1	1.0044E–08	2.4988E–08	1.8747E–07	2.4988E–08	1.8431E–18
0.05	0.5	2.3047E–08	2.4988E–08	1.8749E–07	2.4988E–08	4.3368E–18
	0.9	1.0044E–08	2.4987E–08	1.8751E–07	2.4988E–08	1.8431E–18
	0.1	1.4790E–08	4.9975E–08	3.7493E–07	4.9975E–08	2.9273E–18
0.1	0.5	3.8252E–08	4.9975E–08	3.7498E–07	4.9975E–08	7.2641E–18
	0.9	1.4790E–08	4.9975E–08	3.7502E–07	4.9975E–08	2.9273E–18
	0.1	2.2205E–08	4.9975E–07	3.7500E–06	4.9975E–07	4.2283E–18
1	0.5	6.2169E–08	4.9975E–07	3.7504E–06	4.9975E–07	1.1817E–17
	0.9	2.2205E–08	4.9975E–07	3.7509E–06	4.9975E–07	4.3368E–18

Table 7: Absolute errors for $\beta = 1$ and $\gamma = 0.001$

t	x	GCG	ECG	ELG	NSFD [24]	Presented NSFD
	0.1	1.0698E–08	1.0683E–08	9.2752E–09	3.5194E–13	1.8431E–18
0.05	0.5	9.2595E–09	9.2595E–09	9.2595E–09	7.8629E–13	4.3368E–18
	0.9	7.8921E–09	7.8701E–09	9.2845E–09	3.5201E–13	1.8431E–18
	0.1	2.3188E–08	2.3188E–08	2.3173E–08	5.2107E–13	2.9273E–18
0.1	0.5	2.1749E–08	2.1748E–08	2.1749E–08	1.3292E–12	7.2641E–18
	0.9	2.0382E–08	2.0381E–08	2.0360E–08	5.2115E–13	2.9273E–18
	0.1	2.4872E–07	2.4870E–07	2.4729E–07	1.3846E–12	4.2283E–18
1	0.5	2.4728E–07	2.4728E–07	2.4728E–07	2.1636E–12	1.1817E–17
	0.9	2.4591E–07	2.4585E–07	2.4530E–07	7.7895E–13	4.3368E–18

6 Conclusions

In this paper, we present an exact finite-difference scheme for the Huxley equation based on the method in Mickens papers. Moreover, we present an NSFD scheme for the Huxley equation. We also provided analysis of positivity, boundedness, consistency, stability, and convergence of the NSFD scheme. The numerical results obtained by the scheme, compared to ADM, VIM, DTM, CN, and NSFD [24] with the other methods, which reveal that the NSFD scheme is significantly more effective and accurate than the other methods in the literature.

Table 8: The absolute errors for value of γ, x, and t with $\gamma = 0.001$ and $\beta = 1$

x	t	EFDM	Presented NSFD
	0.05	1.030307E–08	1.84314E–18
0.1	0.1	1.506294E–08	2.92734E–18
	1	2.248771E–08	4.22838E–18
	0.05	2.313697E–08	4.33680E–18
0.5	0.1	3.843952E–08	7.26415E–18
	1	6.246539E–08	1.18178E–17
	0.05	1.030307E–08	1.84314E–18
0.9	0.1	1.506294E–08	2.92734E–18
	1	2.248771E–08	4.33680E–18

Table 9: The absolute errors for value of γ, x, and t with $\gamma = 0.001$ and $\beta = 1$

x	t	meshless RBFs method (MQ)	Presented NSFD
	0.1	0.0E–09	1.9E–18
0.05	0.5	1.0E–09	4.4E–18
	0.9	1.0E–09	1.8E–18
	0.1	1.0E–09	3.1E–18
0.1	0.5	0.0E–09	7.1E–18
	0.9	0.0E–09	2.8E–18
	0.1	1.0E–09	4.4E–18
1	0.5	0.0E–09	1.2E–17
	0.9	1.0E–09	4.7E–18

References

1. Batiha, B., Noorani, M.S.M. and Hashim, I. *Application of variational iteration method to the generalized Burgers–Huxley equation*, Chaos Solitons Fractals, 36(3) (2008), 660–663.

2. Batiha, B., Noorani, M.S.M. and Hashim, I. *Numerical simulation of the generalized Huxley equation by He's variational iteration method,* Appl. Math. Comput. 186(2) (2007), 1322–1325.

3. Biazar, J. and Mohammadi, F. *Application of differential transform method to the generalized Burgers–Huxley equation,* Appl. Appl. Math. 5(10) (2010), 1726–1740.

4. Erdogan, U. and Ozis, T. *A smart nonstandard finite difference scheme for second order nonlinear boundary value problems*, J. Comput. Phys. 230(17) (2011), 6464–6474.

5. Gonzalez–Parra, G., Arenas, A.J. and Chen–Charpentier, B.M. *Combination of nonstandard schemes and Richardsons extrapolation to improve the numerical solution of population models*, Math. Comput. Modelling, 52(7-8) (2010), 1030–1036.

6. Hariharan, G. and Kannan, K. *Haar wavelet method for solving FitzHugh–Nagumo equation*, World Acad Sci, Eng Technol, 67(43) (2010), 560–574.

7. Hashim, I., Noorani, M.S.M. and Batiha, B. *A note on the Adomian decomposition method for the generalized Huxley equation*, Appl. Math. Comput. 181(2) (2006), 1439–1445.

8. Hashim, I., Noorani, M.S.M. and Said Al–Hadidi, M.R. *Solving the generalized Burgers–Huxley equation using the Adomian decomposition method*, Math. Comput. Modelling, 43(11-12) (2006),1404–1411.

9. Inan, B. *Finite difference methods for the generalized Huxley and Burgers-Huxley equations*, Kuwait J. Sci. 44(3) (2017), 20–27.

10. Ismail, H.N.A., Raslan, K. and Abd Rabboh, A.A. *Adomian decomposition method for Burgers– Huxley and Burgers–Fisher equations*, Appl. Math. Comput. 159(1) (2004), 291–301.

11. Khattak, A.J. *A computational meshless method for the generalized Burgers–Huxley equation*, Applied Mathematical Modelling 33 (2009), 3718–3729.

12. Mickens, R.E. *Calculation of denominator functions for nonstandard finite difference schemes for differential equations satisfying a positivity condition*, Numer. Methods Partial Differential Equations, vol 23(3) (2007), 672–691.

13. Mickens, R.E. *Advances in the Applications of Nonstandard Finite Difference Schemes*, Wiley–Interscience, Singapore, 2005.

14. Murray, J.D. *Mathematical Biology I, II*, Third edition, Springer, 2003.

15. Namjoo, M. and Zibaei, S. *Numerical solutions of FitzHugh–Nagumo equation by exact finite-difference and NSFD schemes*, Comp. Appl. Math., (2016) 1–17.

16. Singh, B.K., Arora, G and Singh, M.K. *A numerical scheme for the generalized Burgers–Huxley equation*, Journal of the Egyptian Mathematical Society (2016), 1–9.

17. Wang, X.Y., Zhu, Z.S. and Lu, Y.K. *Solitary wave solutions of the generalised Burgers–Huxley equation*, J. Phys. A: Math. Gen., 23 (1990), 271–274.

18. Wazwaz, A.M. *Partial Differential Equations and Solitary Waves Theory–Higher Education Press*, Springer, New York, 2009.

19. Zeinadini, M. and Namjoo, M. *A Numerical Method for Discrete Fractional–Order Chemostat Model Derived from Nonstandard Numerical Scheme*, Bull. Iranian Math. Soc., (2016), in press.

20. Zibaei, S. and Namjoo, M. *A Nonstandard Finite Difference Scheme for Solving Fractional–Order Model of HIV–1 Infection of CD* 4^+ *T–cells*, Iran. J. Math. Chem., vol 6(2) (2015), 145–160.

21. Zibaei, S. and Namjoo, M. *A Nonstandard Finite Difference Scheme for Solving Three–Species Food Chain with Fractional–Order Lotka–Volterra Model*, Iran. J. Numer. Anal. Optim., vol 6(1) (2016), 53–78.

22. Zibaei, S. and Namjoo, M. *A NSFD scheme for Lotka–Volterra food web model*, Iran. J. Sci. Technol. Trans. A Sci., vol 38(4) (2014), 399–414.

23. Zibaei, S. and Namjoo, M. *Solving fractional–order competitive Lotka–Volterra model by NSFD schemes*, TWMS J. App. Eng. Math., vol 6(2) (2016), 264–277.

24. Zibaei, S., Zeinadini, M. and Namjoo, M. *Numerical solutions of Burgers–Huxley equation by exact finite difference and NSFD schemes*, J. Difference Equ. Appl., (2016) 1–16.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 37–68
DOI:10.22067/ijnao.v9i1.66730

An operational approach for solving fractional pantograph differential equation

H. Ebrahimi* and K. Sadri

Abstract

The aim of the current paper is to construct the shifted fractional-order Jacobi functions (SFJFs) based on the Jacobi polynomials to numerically solve the fractional-order pantograph differential equations. To achieve this purpose, first the operational matrices of integration, product, and pantograph, related to the fractional-order basis, are derived (operational matrix of integration is derived in Riemann–Liouville fractional sense). Then, these matrices are utilized to reduce the main problem to a set of algebraic equations. Finally, the reliability and efficiency of the proposed scheme are demonstrated by some numerical experiments. Also, some theorems are presented on existence of solution of the problem under study and convergence of our method.

Keywords: Fractional pantograph differential equation; Fractional-order Jacobi functions; Operational matrices; Caputo derivative; Riemann–Liouville integral.

1 Introduction

In recent decades, fractional calculus and fractional differential equations have attracted the interest of many mathematicians and researchers. Fractional differential equations appear in various fields of sciences and engineering such as visco-elastic materials [1], propagation of spherical flames [16], electromagnetism [7], fluid mechanics [13], and continuum and statistical

*Corresponding author
Received 10 August 2017; revised 27 June 2018; accepted 29 July 2018
H. Ebrahimi
Department of Mathematics, Rasht branch, Islamic Azad University, Rasht, Iran, P. O. Box 41335-3516, P. C. 4147654919. e-mail: h.ebrahimi@iaurasht.ac.ir
K. Sadri
Young Researchers and Elite Club, Rasht Branch, Islamic Azad University, Rasht, Iran. e-mail: kh.sadri@uma.ac.ir

mechanics [15]. The fractional nature of this class of equations makes their solving very difficult. For this reason, many researchers and authors have tried to generalize the existing methods to simply apply and numerically or analytically solve the fractional-order functional equations. These methods include Spectral method [5, 6], Tau method [2], Muntz–Legendre Tau Method [17], Collocation method [27], and many others. In this regard, valuable works have been presented, for example ,Dehghan and et al. applied the Jacobi polynomials to solve fractional variational problems [4]. In [8], the authors utilized new fractional-order Legendre functions to solve weakly singular Volterra integral equations. Esmail and et al. in [9] presented the collocation Muntz method for the solution of fractional differential equations. Kazem generalized the Jacobi integral operational matrix to solve a class of nonlinear fractional differential equations [11]. The authors in [12] utilized the Tau method together with the d-dimensional orthogonal functions to solve partial differential equations.

In this paper, we consider fractional–order pantograph differential equations as follows:

$$\begin{aligned} &D^{\gamma}u(t) = au(t) + \sum_{i=1}^{l} b_i D^{\gamma_i} u(q_i t) + f(t), \qquad m-1 < \gamma \leqslant m, \;\; t \in [0,1], \\ &u^{(j)}(0) = d_j, \qquad j = 0, 1, \ldots, m-1, \end{aligned} \tag{1}$$

where $a, b_i \in \mathbb{R}$, $0 \leqslant \gamma_i < \gamma \leqslant m$, $i = 1, 2, \ldots, l$, $0 < q_i < 1$, D is the fractional derivative in the Caputo sense, and $u(t)$ is an unknown function.

The pantograph equation is one of the kinds of delay differential equations and has many applications in electro-dynamic and biology; see [14]. Several authors have solved the pantograph differential equations of integer-order such as, Jacobi operational method [3], Chebyshev polynomials [21], Bernoullie polynomials [23], variational iteration method [26], and so on. But there exist few methods applied to numerical solution of pantograph differential equations of fractional-order. For example, we can mention the Hermite wavelet method [20], spectral-collocation method [24], and Legendre multiwavelet collocation method [25]. This motivates us to present a low cost algorithm for solving this kind of fractional-order equations. To this end, we define the shifted fractional-order Jacobi functions based on shifted Jacobi polynomial on $[0, 1]$. Then, the operational matrices of fractional integration, product, and pantograph are constructed. The resultant matrices are utilized to approximate the various terms in equation (1). Finally, the main problem is converted to an algebraic equation, which is collocated at the roots of $P_{N+1}^{(\alpha,\beta)}(t)$. Consequently, an algebraic system is achieved, which solving it leads to determine the unknown coefficients vector and thereupon an approximate solution is obtained. It must be mentioned that the resultant nonlinear algebraic system (corresponding to a nonlinear equation) will be solved by the well-known Newton iteration method.

The current paper is organized as follows: Section 1 contains the introduction; in Section 2, some basic definitions of fractional calculus are introduced; the study of existence and uniqueness of solution of (1) is presented in Section 3; the shifted fractional-order Jacobi functions (SFJFs) and their properties are given in Section 4; in Section 5, the Jacobi operational of fractional integration, product, and pantograph are derived; the proposed method is described to solve the fractional-order pantograph differential equations in Section 6; in Section 7, the convergence of proposed approach is studied and an error bound is obtained in the weighted-Jacobi Sobolev space; In section 8, our numerical results are reported. For this purpose, several examples are presented; finally, a conclusion is presented in Section 9.

2 Basic definitions of fractional calculus

Some basic definitions and properties of fractional calculus theory are presented in this section, which are used in this paper.

Definition 1. *The Riemann–Liouville fractional integral operator of order* $\gamma \geqslant 0$ *is defined as follows:*

$$I^{\gamma}u(t) = \frac{1}{\Gamma(\gamma)}\int_0^t (t-s)^{\gamma-1}u(s)ds, \qquad \gamma > 0, \;\; t > 0,$$
$$I^{0}u(t) = u(t).$$

Definition 2. *The Caputo fractional derivative operator of order* $\gamma \geqslant 0$ *is given by*

$$D^{\gamma}u(t) = I^{m-\gamma}D^{m}u(t) = \frac{1}{\Gamma(m-\gamma)}\int_0^t (t-s)^{m-\gamma-1}\frac{d^m}{ds^m}u(s)ds, \;\; t > 0,$$

where $m = \lceil \gamma \rceil$.

For the Riemann–Liouville fractional integral and Caputo fractional derivative operators, we have

1. $I^{\gamma_1}I^{\gamma_2}u(t) = I^{\gamma_1+\gamma_2}u(t)$,
2. $I^{\gamma}(\lambda_1 u_1(t) + \lambda_2 u_2(t)) = \lambda_1 I^{\gamma}u_1(t) + \lambda_2 I^{\gamma}u_2(t)$,
3. $I^{\gamma}x^{\nu} = \dfrac{\Gamma(\nu+1)}{\Gamma(\gamma+\nu+1)}x^{\gamma+\nu}, \quad \nu > -1$,
4. $D^{\gamma}I^{\gamma}u(t) = u(t)$,
5. $I^{\gamma}D^{\gamma}u(t) = u(t) - \sum\limits_{i=0}^{m-1}\dfrac{u^{(i)}(0^+)}{i!}x^i$,

6. $D^{\gamma}x^{\nu} = \begin{cases} 0, & \gamma > \nu, \\ \frac{\Gamma(\nu+1)}{\Gamma(\nu-\gamma+1)}x^{\nu-\gamma}, & otherwise, \end{cases}$

7. $D^{\gamma}\lambda = 0,$

where λ_1, λ_2, and λ are constant.

3 Existence and uniqueness of solutions of fractional-order pantograph differential equations

In this section, the existence of solutions of (1) is established by using the fixed point theorem.

Let Y be a Banach space and let $C(J, Y)$ be the Banach space of continuous $y(t)$ with $y(t) \in Y$ and $t \in J = [0, 1]$ and $\|y\| = \max_{t\in J} |y(t)|$. Moreover, $B_r(y, Y)$ represents the closed ball with center at y and radius r in Y. It is seen that (1) is equivalent to the following integral equation:

$$\begin{aligned} u(t) =& g(t) + \frac{a}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} u(s)ds \\ &+ \sum_{i=1}^{l} \frac{b_i}{\Gamma(\gamma-\gamma_i)} \int_0^t (t-s)^{\gamma-\gamma_i-1} u(q_i s)ds \\ &+ \frac{1}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} f(s)ds, \end{aligned}$$

where

$$g(t) = \sum_{k=0}^{m-1} \frac{d_k\ t^{k+\gamma}}{\Gamma(k+\gamma+1)} - \sum_{i=1}^{n} \frac{b_i}{\Gamma(\gamma-\gamma_i)} \sum_{k=0}^{m_i-1} q_i^k\ d_k\ t^{k+\gamma-\gamma_i}, \quad m_i - 1 < \gamma_i \leqslant m_i, \quad i = 1, 2, \ldots, l.$$

Theorem 1. *If* $(|a|/\Gamma(\gamma+1) + l\ b_0/Z) < 1/2$, *where* $b_0 = \max_{1\leqslant i\leqslant l}\{b_i\}$ *and* $Z = \min_{1\leqslant i\leqslant n}\{\Gamma(\gamma-\gamma_i+1)\}$, *then, the fractional-order pantograph equation* (1) *has a unique solution.*

Proof. Let $W = C(J, Y)$. A mapping $\Psi u(t) : W \to W$ is defined by

$$\begin{aligned} \Psi u(t) =& g(t) + \frac{a}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} u(s)ds \\ &+ \sum_{i=1}^{l} \frac{b_i}{\Gamma(\gamma-\gamma_i)} \int_0^t (t-s)^{\gamma-\gamma_i-1} u(q_i s)ds \\ &+ \frac{1}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} f(s)ds. \end{aligned}$$

It should be shown that Ψ has a fixed point and that this fixed point is a solution of (1). Set $r \geqslant 2\ (G + F/\Gamma(\gamma+1))$, where $G = \max_{t\in J} \|g(t)\|$ and $F = \max_{t\in J} \|f(t)\|$. Then, it can be shown that $\Psi B_r \subset B_r$, where $B_r = \{y \in W | \parallel y \parallel \leqslant r\}$. So, one has

$$\begin{aligned}
\parallel \Psi u(t) \parallel \leqslant & \parallel g(t) \parallel + \frac{|a|}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} \parallel u(s) \parallel ds \\
& + \sum_{i=1}^{l} \frac{|b_i|}{\Gamma(\gamma-\gamma_i)} \int_0^t (t-s)^{\gamma-\gamma_i-1} \parallel u(q_i s) \parallel ds \\
& + \frac{1}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} \parallel f(s) \parallel ds \\
\leqslant & G + r\frac{|a|}{\gamma\Gamma(\gamma)} + r\sum_{i=1}^{l} \frac{|b_i|}{(\gamma-\gamma_i)\Gamma(\gamma-\gamma_i)} + \frac{F}{\gamma\Gamma(\gamma)} \\
\leqslant & G + r\left(\frac{|a|}{\Gamma(\gamma+1)} + \frac{l\ b_0}{Z}\right) + \frac{F}{\Gamma(\gamma+1)} \\
\leqslant & r.
\end{aligned}$$

Thus, Ψ maps B_r into itself. Now, for $u_1(t), u_2(t) \in W$, one has,

$$\begin{aligned}
\parallel \Psi u_1(t) - \Psi u_2(t) \parallel \leqslant & \frac{|a|}{\Gamma(\gamma)} \int_0^t (t-s)^{\gamma-1} \parallel u_1(s) - u_2(s) \parallel ds \\
& + \sum_{i=1}^{l} \frac{|b_i|}{\Gamma(\gamma-\gamma_i)} \int_0^t (t-s)^{\gamma-\gamma_i-1} \parallel u_1(q_i s) - u_2(q_i s) \parallel ds \\
\leqslant & \frac{|a|}{\Gamma(\gamma+1)} \parallel u_1(s) - u_2(s) \parallel + \frac{l\ b_0}{Z} \parallel u_1(s) - u_2(s) \parallel \\
= & \left(\frac{|a|}{\Gamma(\gamma+1)} + \frac{l\ b_0}{Z}\right) \parallel u_1(s) - u_2(s) \parallel .
\end{aligned}$$

As $|a|/\Gamma(\gamma+1) + l\ b_0/Z < 1/2$, the mapping Ψ is a contraction and therefore there exists a unique fixed point $u(t) \in B_r$ such that $\Psi u(t) = u(t)$. □

4 Fractional-order Jacobi functions

In this section, first we recall the definitions and some useful properties of the orthogonal Jacobi polynomials in the interval $[0, 1]$. Then, we define the fractional-order Jacobi functions to obtain the operational matrices of fractional integration, product, and pantograph.

4.1 Shifted Jacobi polynomials

The shifted Jacobi polynomials are defined on the interval $[0,1]$, with weighted function $w^{(\alpha,\beta)}(x) = (1-x)^{\alpha}\ x^{\beta}$, and can be determined with the following recurrence formula [22]:

$$\begin{aligned} P_{i+1}^{(\alpha,\beta)}(x) =& A(\alpha,\beta,i)\ P_i^{(\alpha,\beta)}(x) + (2x-1)\ B(\alpha,\beta,i)\ P_i^{(\alpha,\beta)}(x) \\ & - E(\alpha,\beta,i)\ P_{i-1}^{(\alpha,\beta)}(x), \qquad i = 1,2,\ldots, \end{aligned} \tag{2}$$

where

$$\begin{aligned} A(\alpha,\beta,i) &= \frac{(2i+\alpha+\beta+1)(\alpha^2-\beta^2)}{2(i+1)(i+\alpha+\beta+1)(2i+\alpha+\beta)}, \\ B(\alpha,\beta,i) &= \frac{(2i+\alpha+\beta+2)(2i+\alpha+\beta+1)}{2(i+1)(i+\alpha+\beta+1)}, \\ E(\alpha,\beta,i) &= \frac{(i+\alpha)(i+\beta)(2i+\alpha+\beta+2)}{(i+1)(i+\alpha+\beta+1)(2i+\alpha+\beta)}, \end{aligned}$$

and

$$P_0^{(\alpha,\beta)}(x) = 1, \qquad P_1^{(\alpha,\beta)}(x) = \frac{\alpha+\beta+2}{2}(2x-1) + \frac{\alpha-\beta}{2}.$$

The orthogonality relation of Jacobi polynomials is defined as follows:

$$\int_0^1 {P_j}^{(\alpha,\beta)}(x)\ {P_k}^{(\alpha,\beta)}(x)\ w^{(\alpha,\beta)}(x)\ dx = h_k\ \delta_{jk},$$

where

$$h_k = \frac{\Gamma(k+\alpha+1)\Gamma(k+\beta+1)}{(2k+\alpha+\beta+1)k!\ \Gamma(k+\alpha+\beta+1)}.$$

The analytic form of Jacobi polynomials is given by

$$P_i^{(\alpha,\beta)}(x) = \sum_{k=0}^{i} \frac{(-1)^{i-k}\Gamma(i+\beta+1)\Gamma(i+k+\alpha+\beta+1)x^k}{\Gamma(k+\beta+1)\Gamma(i+\alpha+\beta+1)\ (i-k)!\ k!}, \qquad i = 0,1,\ldots\ .$$

A continuous function $u(t)$, in the interval $[0,1]$, can be expressed in terms of shifted Jacobi polynomials as follows:

$$u(x) = \sum_{j=0}^{\infty} C_j\ P_j^{(\alpha,\beta)}(x), \tag{3}$$

where the coefficients C_j are given by

$$C_j = \frac{1}{h_j}\int_0^1 u(t)\ P_j^{(\alpha,\beta)}(x)\ w^{(\alpha,\beta)}(x)\ dx, \qquad j = 0,1,\ldots\ .$$

In practice, only the first $(N+1)$-terms of shifted Jacobi polynomials are considered. Therefore, one has

$$u_N(x) = \sum_{j=0}^{N} C_j \; P_j^{(\alpha,\beta)}(x) = \Phi^T(x) \; C = C^T \; \Phi(x), \tag{4}$$

where the vectors C and $\Phi(t)$ are given by

$$C = [C_0, C_1, \ldots, C_N]^T, \qquad \Phi(x) = [P_0^{(\alpha,\beta)}(x), P_1^{(\alpha,\beta)}(x), \ldots, P_N^{(\alpha,\beta)}(x)]^T. \tag{5}$$

Also, the following auxiliary relation is obtained from (2).

$$\begin{aligned} 2x \; P_i^{(\alpha,\beta)}(x) =& \frac{1}{B(\alpha,\beta,i)} \; P_{i+1}^{(\alpha,\beta)}(x) - \frac{A(\alpha,\beta,i)}{B(\alpha,\beta,i)} \; P_i^{(\alpha,\beta)}(x) \\ &+ \frac{E(\alpha,\beta,i)}{B(\alpha,\beta,i)} \; P_{i-1}^{(\alpha,\beta)}(x), \qquad i = 1, 2, \ldots, N. \end{aligned} \tag{6}$$

Two other properties of the shifted Jacobi polynomials are defined below:

$$\begin{aligned} & P_i^{(\alpha,\beta)}(0) = (-1)^i \binom{i+\alpha}{i}, \\ & \frac{d^i P_n^{(\alpha,\beta)}(x)}{dx^i} = \frac{\Gamma(n+i+\alpha+\beta+1)}{\Gamma(n+\alpha+\beta+1)} P_{n-i}^{(\alpha+i,\beta+i)}(x). \end{aligned} \tag{7}$$

4.2 Shifted fractional-order Jacobi functions

We define the shifted fractional-order Jacobi functions (SFJFs) by transformation x to t^σ $(\sigma > 0)$ based on Jacobi polynomials (2) in the interval $[0,1]$. For convenience, the fractional-order Jacobi functions, $P_i^{(\alpha,\beta)}(t^\sigma)$, $i = 0, 1, 2, \ldots$, are denoted by $P_i^{(\alpha,\beta,\sigma)}(t)$. It is clear, the classic Jacobi polynomials are resulted for $\sigma = 1$. The weight function is $w^{(\alpha,\beta,\sigma)}(t) = \sigma t^{\sigma(\beta+1)-1}(1-t^\sigma)^\alpha$. Moreover, the relations (2)–(7) are converted to:

$$\begin{aligned} P_{i+1}^{(\alpha,\beta,\sigma)}(t) =& A(\alpha,\beta,i) \; P_i^{(\alpha,\beta,\sigma)}(t) + (2t^\sigma - 1) \; B(\alpha,\beta,i) \; P_i^{(\alpha,\beta,\sigma)}(t) \\ & - E(\alpha,\beta,i) \; P_{i-1}^{(\alpha,\beta,\sigma)}(t), \qquad i = 1, 2, \ldots, \end{aligned} \tag{8}$$

$$P_0^{(\alpha,\beta,\sigma)}(t) = 1, \qquad P_1^{(\alpha,\beta,\sigma)}(t) = \frac{\alpha+\beta+2}{2}(2t^\sigma - 1) + \frac{\alpha-\beta}{2},$$

where coefficients $A(\alpha,\beta,i)$, $B(\alpha,\beta,i)$, and $E(\alpha,\beta,i)$ are same in (2).

$$\int_0^1 P_j^{(\alpha,\beta,\sigma)}(t) \; P_k^{(\alpha,\beta,\sigma)}(t) \; w^{(\alpha,\beta,\sigma)}(t) \; dt = h_k \; \delta_{jk},$$

$$P_i^{(\alpha,\beta,\sigma)}(t) = \sum_{k=0}^{i} \frac{(-1)^{i-k}\Gamma(i+\beta+1)\Gamma(i+k+\alpha+\beta+1)t^{\sigma k}}{\Gamma(k+\beta+1)\Gamma(i+\alpha+\beta+1)\ (i-k)!\ k!}, \tag{9}$$

$$u(t) = \sum_{j=0}^{\infty} C_j\ P_j^{(\alpha,\beta,\sigma)}(t),$$

where

$$C_j = \frac{1}{h_j}\int_0^1 u(t)\ P_j^{(\alpha,\beta,\sigma)}(t)\ w^{(\alpha,\beta,\sigma)}(t)\ dt, \qquad j = 0, 1, \ldots,$$

$$u_N(t) = \sum_{j=0}^{N} C_j\ P_j^{(\alpha,\beta,\sigma)}(t) = \Phi^{(\sigma)T}(t)\ C = C^T\ \Phi^{(\sigma)}(t),$$

where the vectors C and $\Phi^{(\sigma)}(t)$ are given by

$$C = [C_0, C_1, \ldots, C_N]^T, \qquad \Phi^{(\sigma)}(t) = [P_0^{(\alpha,\beta,\sigma)}(t), P_1^{(\alpha,\beta,\sigma)}(t), \ldots, P_N^{(\alpha,\beta,\sigma)}(t)]^T. \tag{10}$$

$$\begin{aligned} 2t^{\sigma}\ P_i^{(\alpha,\beta,\sigma)}(t) =& \frac{1}{B(\alpha,\beta,i)}\ P_{i+1}^{(\alpha,\beta,\sigma)}(t) - \frac{A(\alpha,\beta,i)}{B(\alpha,\beta,i)}\ P_i^{(\alpha,\beta,\sigma)}(t) \\ &+ \frac{E(\alpha,\beta,i)}{B(\alpha,\beta,i)}\ P_{i-1}^{(\alpha,\beta,\sigma)}(t), \qquad i = 1, 2, \ldots, N, \end{aligned} \tag{11}$$

$$\frac{dP_n^{(\alpha,\beta,\sigma)}(t)}{dt} = \sigma\Gamma(n+\alpha+\beta+1)t^{\sigma-1}P_{n-1}^{(\alpha+1,\beta+1,\sigma)}(t).$$

5 Fractional-order Jacobi operational matrices of integration, product, and pantograph

The objective of this section is to derive the operational matrices of fractional integration, product, and pantograph related to the SFJFs. The fractional-order operational matrices of product and pantograph are utilized for approximating the product of the vectors $\Phi^{(\sigma)}(t)$ and $\Phi^{(\sigma)T}(t)$ and the terms, which include the delay in equations under study, respectively. First, some properties of the shifted fractional-order Jacobi functions are presented as follows.

Lemma 1 (see [19]). *The ith shifted fractional-order Jacobi function, $P_i^{(\alpha,\beta,\sigma)}(t)$, $t \in [0,1]$, can be obtained in the form*

$$P_n^{(\alpha,\beta,\sigma)}(t) = \sum_{k=0}^{n} \gamma_k^{(n)}\ t^{\sigma k},$$

where coefficients $\gamma_k^{(i)}$ *are given as*

$$\gamma_k^{(n)} = (-1)^{n-k} \binom{n+k+\alpha+\beta}{k} \binom{n+\alpha}{n-k}.$$

Lemma 2. *If* $n \in \mathbb{N}$ *and* $p \geqslant n$*, then we have*

$$\int_0^1 t^p \; P_n^{(\alpha,\beta,\sigma)}(t) \; w^{(\alpha,\beta,\sigma)}(t) \; dt$$
$$= \sum_{k=0}^{n} \frac{(-1)^{n-k} \; \Gamma(n+\beta+1) \; \Gamma(n+k+\alpha+\beta+1) \; \Gamma(k+\beta+p/\sigma+1) \; \Gamma(\alpha+1)}{\Gamma(k+\beta+1) \; \Gamma(n+\alpha+\beta+1) \; \Gamma(k+\alpha+\beta+p/\sigma+2) \; (n-k)! \; k!}.$$

Proof. The lemma can be easily proved by multiplying relation (9) in t^p and then integrating of the resulting equation. □

Now, the Jacobi operational matrix of fractional integration is generally derived.

Theorem 2. *Suppose that* $\Phi^{(\sigma)}(t)$ *is the shifted fractional-order Jacobi vector in equation* (10)*. Then, we have*

$$I^{(\nu)}\Phi^{(\sigma)}(t) \simeq \mathbf{P}^{(\nu)} \; \Phi^{(\sigma)}(t),$$

where $I^{(\nu)}$ *is the Riemann–Liouville fractional integral operator of order* ν *and* $\mathbf{P}^{(\nu)}$ *is the* $(N+1)\times(N+1)$ *operational matrix of fractional integration which defined by*

$$\mathbf{P}^{(\nu)} = \begin{bmatrix} \pi(0,0) & \pi(0,1) & \dots & \pi(0,N) \\ \pi(1,0) & \pi(1,1) & \dots & \pi(1,N) \\ \vdots & \vdots & \ddots & \vdots \\ \pi(N,0) & \pi(N,1) & \dots & \pi(N,N) \end{bmatrix},$$

where

$$\pi(i,j) = \sum_{k=0}^{i} \omega_{ijk}, \qquad i = 0,1,\dots,N, \quad j = 1,\dots,N, \tag{12}$$

and ω_{ijk} *are given by*

$$\omega_{ijk} = \frac{(-1)^{i-l} \; \Gamma(j+\beta+1) \; \Gamma(i+\beta+1) \; \Gamma(i+l+\alpha+\beta+1) \; \Gamma(\alpha+1)\Gamma(\sigma l+1)}{h_j \; \Gamma(j+\alpha+\beta+1) \; \Gamma(l+\beta+1) \; \Gamma(i+\alpha+\beta+1) \; l! \; (i-l)!\Gamma(\sigma l+\nu+1)}$$
$$\times \sum_{l=0}^{j} \frac{(-1)^{j-k} \; \Gamma(j+k+\alpha+\beta+1) \; \Gamma(k+l+\beta+\nu/\sigma+2)}{\Gamma(k+\beta+1) \; \Gamma(k+l+\alpha+\beta+\nu/\sigma+2) \; k!(j-k)!}, \; i,j = 0,1,\dots,N.$$

Proof. By using the analytical form (9) and the properties of operator I^{ν}, we have

$$I^{\nu} P_i^{(\alpha,\beta)}(t) \; dt = \sum_{l=0}^{i} \frac{(-1)^{i-l} \; \Gamma(i+\beta+1) \; \Gamma(i+l+\alpha+\beta+1) \; \Gamma(\sigma l+1) \; t^{\sigma l+\nu}}{\Gamma(l+\beta+1) \; \Gamma(i+\alpha+\beta+1) \; \Gamma(\sigma l+\nu+1) \; l! \; (i-l)!}. \tag{13}$$

Now, $t^{\sigma l+\nu}$ can be approximated in terms of the SFJFs as follows:

$$t^{\sigma l+\nu} \simeq \sum_{j=0}^{N} a_{l,j}\ P_j^{(\alpha,\beta,\sigma)}(t),$$

where

$$a_{l,j} = \frac{1}{h_j}\ \int_0^1 t^{\sigma l+\nu}\ P_j^{(\alpha,\beta,\sigma)}(t)\ w^{(\alpha,\beta,\sigma)}(t)\ dt.$$

According to Lemma 2, one has

$$\begin{aligned}&\int_0^1 t^{\sigma l+\nu}\ P_j^{(\alpha,\beta,\sigma)}(t)\ w^{(\alpha,\beta,\sigma)}(t)\ dt\\ &\quad= \sum_{k=0}^{j} \frac{(-1)^{j-k}\ \Gamma(\alpha+1)\ \Gamma(j+\beta+1)\ \Gamma(k+l+\beta+\nu/\sigma+1)\ \Gamma(j+k+\alpha+\beta+1)}{\Gamma(k+\beta+1)\ \Gamma(j+\alpha+\beta+1)\ \Gamma(k+l+\alpha+\beta+\nu/\sigma+2)\ k!\ (j-k)!}.\end{aligned}$$

Therefore, equation (13) is written as follows:

$$\begin{aligned}&I^{\nu} P_i^{(\alpha,\beta,\sigma)}(t)\ dt\\ &\quad\simeq \sum_{j=0}^{N} \Bigg\{ \sum_{k=0}^{i} \frac{(-1)^{i-l}\ \Gamma(j+\beta+1)\ \Gamma(i+\beta+1)\ \Gamma(i+l+\alpha+\beta+1)\ \Gamma(\alpha+1)\ \Gamma(\sigma l+1)}{h_j\ \Gamma(j+\alpha+\beta+1)\ \Gamma(l+\beta+1)\ \Gamma(i+\alpha+\beta+1)\ \Gamma\sigma l+\nu+1)\ l!\ (i-l)!}\\ &\qquad\qquad \times \sum_{k=0}^{j} \frac{(-1)^{j-k}\ \Gamma(j+k+\alpha+\beta+1)\ \Gamma(l+k+\beta+\nu/\sigma+1)}{\Gamma(k+\beta+1)\ \Gamma(l+k+\alpha+\beta+\nu/\sigma+2)\ (j-k)!\ (k)!} \Bigg\}\ P_j^{(\alpha,\beta,\sigma)}(t)\\ &\quad= \sum_{j=0}^{N} \pi(i,j)\ P_j^{(\alpha,\beta,\sigma)}(t),\end{aligned} \tag{14}$$

where $\pi(i,j)$ are given by equation (12). Accordingly, equation (14) can be rewritten as the following vector form:

$$I^{\nu} P_i^{(\alpha,\beta,\sigma)}(t) \simeq [\pi(i,0), \pi(i,1), \ldots, \pi(i,N)]\ \Phi^{(\sigma)}(t), \quad i = 0, 1, \ldots, N.$$

This leads to the desired result. □

The following lemmas are useful to obtain the fractional operational matrix of product.

Lemma 3 (see [19]). *If $P_j^{(\alpha,\beta,\sigma)}(t)$ and $P_k^{(\alpha,\beta,\sigma)}(t)$ are respectively jth and kth SFJFs, then the product of $P_j^{(\alpha,\beta,\sigma)}(t)$ and $P_k^{(\alpha,\beta,\sigma)}(t)$ can be written as*

$$Q_{j+k}^{(\alpha,\beta,\sigma)}(t) = \sum_{r=0}^{j+k} \lambda_r^{(j,k)}\ t^{\sigma r},$$

where coefficients $\lambda_r^{(j,k)}$ are determined as follows:

If $j \geq k$ **:**

$r = 0, 1, \ldots, j + k,$
$if \;\; r > j, \;\; then$
$\lambda_r^{(j,k)} = \sum_{l=r-j}^{k} \gamma_{r-l}^{(j)} \; \gamma_l^{(k)},$
$else$
$r_1 = min\{r, k\},$
$\lambda_r^{(j,k)} = \sum_{l=0}^{r_1} \gamma_{r-l}^{(j)} \; \gamma_l^{(k)},$
$end.$

If $j < k$ **:**

$r = 0, 1, \ldots, j + k,$
$if \;\; r \leq j, \;\; then$
$r_1 = min\{r, j\},$
$\lambda_r^{(j,k)} = \sum_{l=0}^{r_1} \gamma_{r-l}^{(j)} \; \gamma_l^{(k)},$
$else$
$r_2 = min\{r, k\},$
$\lambda_r^{(j,k)} = \sum_{l=r-j}^{r_2} \gamma_{r-l}^{(j)} \; \gamma_l^{(k)},$
$end.$

The quantities $\gamma_l^{(k)}$ *and* $\gamma_{r-l}^{(j)}$ *are introduced, respectively, for* $P_k^{(\alpha,\beta,\sigma)}(t)$ *and* $P_j^{(\alpha,\beta,\sigma)}(t)$ *based on the Lemma 1.*

Lemma 4. *If* $P_i^{(\alpha,\beta,\sigma)}(t)$, $P_j^{(\alpha,\beta,\sigma)}(t)$, *and* $P_k^{(\alpha,\beta,\sigma)}(t)$ *are respectively i-, j-, and kth SFJFs, then,*

$$q_{ijk} = \int_0^1 P_i^{(\alpha,\beta,\sigma)}(t) \; P_j^{(\alpha,\beta,\sigma)}(t) \; P_k^{(\alpha,\beta,\sigma)}(t) \; w^{(\alpha,\beta,\sigma)}(t) \; dt$$
$$= \sum_{n=0}^{j+k} \sum_{l=0}^{i} \frac{(-1)^{i-l} \lambda_n^{(j,k)} \; \Gamma(i+\beta+1) \; \Gamma(i+l+\alpha+\beta+1) \; \Gamma(n+l+\beta+1) \; \Gamma(\alpha+1)}{\Gamma(l+\beta+1) \; \Gamma(i+\alpha+\beta+1) \; \Gamma(n+l+\alpha+\beta+2) \; (i-l)! \; l!},$$

where $\lambda_l^{(j,k)}$ *are computed by Lemma 3.*

Now, a general formula is presented for finding the $(N+1) \times (N+1)$ operational matrix of fractional product $\tilde{U}$ whenever,

$$\Phi^{(\sigma)}(t) \; \Phi^{(\sigma)T}(t) \; U \simeq \tilde{U} \; \Phi^{(\sigma)}(t), \tag{15}$$

and U is a $(N+1)$ given vector.

Theorem 3 (see [3])**.** *The entries of the matrix* $\tilde{U}$ *in* (15) *are computed as follows:*

$$\tilde{U}_{jk} = \frac{1}{h_k} \sum_{i=0}^{N} U_i \; q_{ijk}, \qquad j, k = 0, 1, \ldots, N,$$

where q_{ijk} are introduced by Lemma 4 and U_i are the components of the vector U in (15).

Corollary 1. *If $u(t) \simeq U^T\ \Phi^{(\sigma)}(t) = \Phi^{(\sigma)T}(t)\ U$ and $\Phi^{(\sigma)}(t)\ \Phi^{(\sigma)T}(t)\ U \simeq \tilde{U}\ \Phi^{(\sigma)}(t)$, where U is an $(N+1)$ vector and $\tilde{U}$ is the $(N+1)\times(N+1)$ operational matrix of fractional product, corresponding to the vector U, it is easily shown that,*

$$u^2(t) \simeq U^T\ \tilde{U}\ \Phi^{(\sigma)}(t).$$

To approximate the terms including the delay, the fractional-order Jacobi operational matrix of pantograph is derived. For this end, we apply the auxiliary (11).
Replacing t with qt in definition (10), leads to

$$\Phi^{(\sigma)}(qt) = [P_0^{(\alpha,\beta,\sigma)}(qt), P_1^{(\alpha,\beta,\sigma)}(qt), \ldots, P_N^{(\alpha,\beta,\sigma)}(qt)]^T.$$

The following theorem expresses the relation between vectors $\Phi^{(\sigma)}(qt)$ and $\Phi^{(\sigma)}(t)$.

Theorem 4. *The relation between the delay vector $\Phi^{(\sigma)}(qt)$, $0<q<1$, and vector $\Phi^{(\sigma)}(t)$ can be expressed as follows:*

$$\Phi^{(\sigma)}(tq) = \mathbf{L}_{(\mathbf{q},\sigma)}\ \Phi^{(\sigma)}(t),$$

where $\mathbf{L}_{(\mathbf{q},\sigma)}$ is an $(N+1)\times(N+1)$ lower triangular matrix called fractional operational matrix of pantograph.

Proof. Replacing t with qt in (8) and rearranging leads to

$$\begin{aligned} P_{i+1}^{(\alpha,\beta,\sigma)}(qt) =& A(\alpha,\beta,i)\ P_i^{(\alpha,\beta,\sigma)}(qt) + (2q^\sigma t^\sigma - 1)\ B(\alpha,\beta,i)\ P_i^{(\alpha,\beta,\sigma)}(qt) \\ & - E(\alpha,\beta,i)\ P_{i-1}^{(\alpha,\beta,\sigma)}(qt), \qquad i = 1,2,\ldots\ . \end{aligned} \tag{16}$$

The ith delay SFJF , $P_i^{(\alpha,\beta,\sigma)}(qt)$, can be expressed in terms of the SFJFs themselves as follows:

$$P_i^{(\alpha,\beta,\sigma)}(qt) = \sum_{k=0}^{i} l_{i,k}\ P_k^{(\alpha,\beta,\sigma)}(qt), \qquad i = 0,1,\ldots\ . \tag{17}$$

From the first few polynomials in (17), it easily obtains

$$l_{0,0} = 1, \quad l_{0,i} = 0,\ \ i = 1,2,\ldots,N, \quad l_{1,0} = (\beta+1)(q^\sigma - 1), \quad l_{1,1} = q^\sigma. \tag{18}$$

It is clearly found that,

$$l_{i,i} = q^{i\sigma}, \qquad i = 0,1,\ldots,N. \tag{19}$$

Using auxiliary (11) and substituting (17) into equation (16), we have

$$\begin{aligned}
&\sum_{k=0}^{i+1} l_{i+1,k}\ P_k^{(\alpha,\beta,\sigma)}(t)\\
&= \sum_{k=0}^{i} l_{i,k}\ \Big\{\Big(A(\alpha,\beta,i) + q^{\sigma}\ B(\alpha,\beta,i)(1-\frac{A(\alpha,\beta,k)}{B(\alpha,\beta,k)})\ P_{k+1}^{(\alpha,\beta,\sigma)}(t) - B(\alpha,\beta,i)\Big)\ P_k^{(\alpha,\beta,\sigma)}(t)\\
&\qquad + q^{\sigma} B(\alpha,\beta,i)\ \Big(\frac{1}{B(\alpha,\beta,k)} + \frac{E(\alpha,\beta,k)}{B(\alpha,\beta,k)}\ P_{k-1}^{(\alpha,\beta,\sigma)}(t)\Big)\Big\}\\
&- E(\alpha,\beta,i)\ \sum_{k=0}^{i-1} l_{i-1,k}\ P_k^{(\alpha,\beta,\sigma)}(t).
\end{aligned} \tag{20}$$

Equating the coefficients of $P_k^{(\alpha,\beta,\sigma)}(t)$ on both sides of equation (20), for $k = 0, 1, \ldots, i$, leads to

$$\begin{aligned}
l_{i+1,0} =& \Big\{ A(\alpha,\beta,i)\\
&+ q^{\sigma}\ B(\alpha,\beta,i)\Big(1 - \frac{\alpha-\beta}{\alpha+\beta+2} + \frac{2(\alpha+1)(\beta+1)}{(\alpha+\beta+2)(\alpha+\beta+3)}\Big)\\
&- B(\alpha,\beta,i)\Big\}\ l_{i,0} - E(\alpha,\beta,i)\ l_{i-1,0}, \quad i = 1,2,\ldots,N-1,
\end{aligned}$$

$$\begin{aligned}
l_{i+1,k} =& \Big\{ A(\alpha,\beta,i) + q^{\sigma}\ B(\alpha,\beta,i)(1 - \frac{A(\alpha,\beta,i)}{B(\alpha,\beta,i)} - B(\alpha+\beta+i))\Big\}\ l_{i,k}\\
&+ \Big\{ \frac{1}{B(\alpha,\beta,k-1)}\ l_{i,k-1} + \frac{E(\alpha,\beta,k+1)}{B(\alpha,\beta,k+1)}\ l_{i,k+1}\Big\}\ q^{\sigma} B(\alpha,\beta,i)\\
&- C(\alpha,\beta,i)\ l_{i-1,k}, \qquad k = 1,2,\ldots,i-1,
\end{aligned}$$

$$\begin{aligned}
l_{i+1,i} =& \Big(A(\alpha,\beta,i) + q^{\sigma} B(\alpha,\beta,i)(1 - \frac{A(\alpha,\beta,i)}{B(\alpha,\beta,i)}) - B(\alpha,\beta,i)\Big) q^{i\sigma}\\
&+ q^{\sigma} \frac{B(\alpha,\beta,i)}{B(\alpha,\beta,i-1)}\ l_{i,i-1}, \qquad i = 1,2,\ldots,N-1.
\end{aligned}$$

The starting values are given by equations, (18)–(19). Finally, for the proportional delay q, we have

$$\begin{aligned}
\Phi^{(\sigma)}(qt) &= [P_0^{(\alpha,\beta,\sigma)}(qt), P_1^{(\alpha,\beta,\sigma)}(tq), \ldots, P_N^{(\alpha,\beta,\sigma)}(tq)]^T\\
&= \begin{bmatrix}
1 & 0 & 0 & \ldots & 0 & 0\\
l_{1,0} & q^{\sigma} & 0 & \ldots & 0 & 0\\
l_{2,0} & l_{2,1} & q^{2\sigma} & \ldots & 0 & 0\\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\
l_{N-1,0} & l_{N-1,1} & l_{N-1,2} & \ldots & q^{(N-1)\sigma} & 0\\
l_{N,0} & l_{N,1} & l_{N,2} & \ldots & l_{N,N-1} & q^{N\sigma}
\end{bmatrix}
\begin{bmatrix}
P_0^{(\alpha,\beta,\sigma)}(t)\\
P_1^{(\alpha,\beta,\sigma)}(t)\\
P_2^{(\alpha,\beta,\sigma)}(t)\\
\vdots\\
P_{N-1}^{(\alpha,\beta,\sigma)}(t)\\
P_N^{(\alpha,\beta,\sigma)}(t)
\end{bmatrix}
\end{aligned}$$

$$= \mathbf{L}_{(\mathbf{q},\sigma)}\ \Phi^{(\sigma)}(x).$$

□

Corollary 2. *The operational matrices of fractional integration and product are converted to the following forms for the delay q.*

$$I^{\nu}\Phi^{(\sigma)}(qt)\ \simeq \mathbf{L}_{(\mathbf{q},\sigma)}\ \mathbf{P}^{(\nu)}\ \Phi^{(\sigma)}(t),$$
$$\Phi^{(\sigma)}(qt)\ \Phi^{(\sigma)T}(qt)\ C \simeq \mathbf{L}_{(\mathbf{q},\sigma)}\ \tilde{F}\ \Phi^{(\sigma)}(t), \qquad F = \mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}\ C,$$

where the matrices $\mathbf{P}^{(\nu)}$ and $\mathbf{L}_{(\mathbf{q},\sigma)}$ were defined by Theorems 2 and 4 and $\tilde{F}$ is the fractional operational matrix of product corresponding to the vector $F = \mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}\ C$.

6 Description of method

Consider the problem given in (1) again. We expand $D^{\gamma}u(t)$ by means of the SFJFs as follows:

$$D^{\gamma}u(t) \simeq \Phi^{(\sigma)T}(t)C. \tag{21}$$

According the properties of Caputo fractional derivative, we have

$$\begin{aligned} u(t) &\simeq I^{\gamma}\Phi^{(\sigma)T}(t)C + \sum_{k=0}^{m-1}\frac{d_k}{k!}t^k \\ &\simeq \Phi^{(\sigma)T}(t)\mathbf{P}^{(\gamma)\mathbf{T}}C + \Phi^{(\sigma)T}(t)F, \qquad m-1 < \gamma \leqslant m, \end{aligned} \tag{22}$$

where F is a known vector and $\sum_{k=0}^{m-1}\frac{d_k}{k!}t^k$ is approximated as $\Phi^{(\sigma)T}F$. To approximate the $D^{\gamma_i}u(t)$, $0 \leqslant \gamma_i < \gamma \leqslant m$, $i = 1, 2, \ldots, l$, the following procedure can be pursued.

$$D^{\gamma}u(t) = D^{\gamma-\gamma_i}D^{\gamma_i}u(t) \simeq \Phi^{(\sigma)T}(t)C. \tag{23}$$

Using the Caputo derivative, we get

$$\begin{aligned} D^{\gamma_i}u(t) &\simeq I^{\gamma-\gamma_i}\Phi^{(\sigma)T}(t)C + \sum_{k=\lceil\gamma_i\rceil}^{m-1}\frac{d_k}{\Gamma(k+1-\gamma_i)}t^{k-\gamma_i} \\ &\simeq \Phi^{(\sigma)T}(t)\mathbf{P}^{(\gamma-\gamma_i)\mathbf{T}}C + \Phi^{(\sigma)T}F_i, \qquad 0 \leqslant \gamma_i < \gamma \leqslant m, \quad i = 1, 2, \ldots, l. \end{aligned} \tag{24}$$

Now, by applying equations (21)–(24) to the terms containing the delay, we get

$$
\begin{aligned}
&u(qt) \simeq \Phi^{(\sigma)T}(t)\mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}\mathbf{P}^{(\gamma)\mathbf{T}}C + \Phi^{(\sigma)T}(t)\mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}F,\\
&D^{\gamma_i}u(qt) \simeq \Phi^{(\sigma)T}(t)\mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}\mathbf{P}^{(\gamma-\gamma_\mathbf{i})\mathbf{T}}C + \Phi^{(\sigma)T}(t)\mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}F_i, \qquad i = 1,2,\ldots,l,\\
&u^2(t) \simeq U^T\tilde{U}\Phi^{(\sigma)T}(t), \qquad U = \mathbf{P}^{(\gamma)\mathbf{T}}C + F, \quad D^{\gamma}u(qt) \simeq \Phi^{(\sigma)T}(t)\mathbf{L}^{\mathbf{T}}_{(\mathbf{q},\sigma)}C,
\end{aligned}
\tag{25}
$$

where $\tilde{U}$ is the fractional operational matrix of product corresponding to the vector U. By substituting the expressions (21)–(25) into (1), the following algebraic equations is resulted.

$$
\begin{aligned}
\Phi^{(\sigma)T}(t)C - a\Phi^{(\sigma)T}(t)(\mathbf{P}^{(\gamma)\mathbf{T}}C + F) - \sum_{i=1}^{n} b_i\Phi^{(\sigma)T}(t)(\mathbf{L}^{\mathbf{T}}_{(\mathbf{q_i},\sigma)}\mathbf{P}^{(\gamma-\gamma_\mathbf{i})\mathbf{T}}C\\
+ \mathbf{L}^{\mathbf{T}}_{(\mathbf{q_i},\sigma)}F_i) - f(t) \approx 0.
\end{aligned}
$$

In this way, the main equation is converted into an algebraic equation. The resultant algebraic equation is collocated at $(N+1)$ roots of the $(N+1)$th shifted Jacobi polynomials on the interval $(0,1)$. By solving the system of generated algebraic equations, the unknown vector C can be determined. Thus, an approximate solution can be acquired for various values of α and β parameters using equation (22).

Corollary 3. *The nonlinear terms in the problems under study are approximated as a matrix form. For this reason, it is necessary that the initial conditions of the problems to be approximated in terms of fractional-order Jacobi basis.*

7 Convergence analysis and error estimation

In this section, the upper bounds are presented for the errors of the residual function, proposed algorithm, and operational matrix of fractional integration, $\mathbf{P}^{(\nu)}$. It is seen that the upper bounds tend to zero when the number of the SFJFs increases. First the following definitions and theorems are stated for $\sigma = 1$.

The set of all algebraic polynomials of degree at most N is denoted by $\mathbb{P}_N$. The orthogonal projection $\mathcal{P}_{N,\alpha,\beta}L^2_{w^{(\alpha,\beta)}}(J) \to \mathbb{P}_N$, $J = [0,1]$, is considered for $v(t) \in L^2_{w^{(\alpha,\beta)}}(J)$ and defined by

$$
(\mathcal{P}_{N,\alpha,\beta}v - v,\ \phi) = 0, \qquad \text{for all } \phi \in \mathbb{P}_N.
$$

The Jacobi-weighted space is introduced as

$$
\mathcal{J}^r_{w^{(\alpha,\beta)}}(J) = \{v|\ v \textit{ is measurable and } \|v\|_{r,w^{(\alpha,\beta)}} < \infty\}, \qquad r \in \mathbb{N},
$$

equipped with the following norm and seminorm,

$$\|v\|_{r,w^{(\alpha,\beta)}} = \left(\sum_{k=0}^{r} \| \partial_x^k v \|^2_{w^{(\alpha+k,\beta+k)}} \right)^{\frac{1}{2}}, \qquad |v|_{r,w^{(\alpha,\beta)}} = \| \partial_x^r v \|_{w^{(\alpha+r,\beta+r)}} .$$

Theorem 5 (see [10]). *For any $v \in \mathcal{J}^r_{w^{(\alpha,\beta)}}(J)$, $r \in \mathbb{N}$, and $0 \leqslant \mu \leqslant r$, we have*

$$\|v - \mathcal{P}_{N,\alpha,\beta} v\|_{\mu,w^{(\alpha,\beta)}} \leqslant c(N(N+\alpha+\beta))^{\frac{\mu-r}{2}} |v|_{r,w^{(\alpha,\beta)}},$$

where c is a positive constant independent of N, α, and β.

Lemma 5. *If $n \in \mathbb{N}$, $v \in \mathcal{J}^r_{w^{(\alpha,\beta)}}(J)$, and $D^n v(t)$ is the derivative of $v(t)$ of order n, then*

$$\|D^n v - D^n \mathcal{P}_{N,\alpha,\beta} v\|_{\mu,w^{(\alpha,\beta)}} \leqslant c^* \left((N+n)(N+n+\alpha+\beta) \right)^{\frac{\mu-r}{2}} |v^{(n)}|_{r,w^{(\alpha,\beta)}}.$$

Proof. By derivation of the shifted Jacobi polynomials and applying Theorem 5, the desired result is acquired. □

Corollary 4. *For $v \in \mathcal{J}^r_{w^{(\alpha,\beta)}}(J)$ and $0 < q < 1$, we have*

$$\|v(qt) - \mathcal{P}_{N,\alpha,\beta} v(qt)\|_{\mu,w^{(\alpha,\beta)}} \leqslant c(qN(N+\alpha+\beta))^{\frac{\mu-r}{2}} |v(qt)|_{r,w^{(\alpha,\beta)}}. \quad (26)$$

Theorem 6. *If $\gamma > 0$, $\gamma \in \mathbb{R}$, and $D^\gamma v(t)$ denotes the Caputo fractional derivative of $v(t) \in \mathcal{J}^r_{w^{(\alpha,\beta)}}(J)$ of order γ, then*

$$\|D^\gamma v - D^\gamma \mathcal{P}_{N,\alpha,\beta} v\|_{\mu,w^{(\alpha,\beta)}} \leqslant \frac{c'}{\Gamma(m-\gamma+1)} \left((N+m)(N+m+\alpha+\beta) \right)^{\frac{\mu-r}{2}} \times |v^{(m)}|_{r,w^{(\alpha,\beta)}}, \quad m = \lceil \gamma \rceil.$$

Proof. Using properties of the Riemann–Liouville fractional integral operator leads to

$$\begin{aligned} D^\gamma v(t) = I^{m-\gamma} D^m v(t) &= \frac{1}{\Gamma(m-\gamma)} \int_0^t (t-s)^{m-\gamma-1} D^m v(s) ds \\ &= \frac{1}{\Gamma(m-\gamma)} t^{m-\gamma-1} * D^m v(t), \end{aligned}$$

where the star symbol denotes the convolution of two functions $t^{m-\gamma-1}$ and $D^m v(t)$. So, by means of Lemma 5, we have

$$\begin{aligned} \|D^\gamma v - D^\gamma \mathcal{P}_{N,\alpha,\beta} v\|_{\mu,w^{(\alpha,\beta)}} &= \|I^{m-\gamma}(D^m v - D^m \mathcal{P}_{N,\alpha,\beta} v)\|_{\mu,w^{(\alpha,\beta)}} \\ &= \left\| \frac{t^{m-\gamma-1}}{\Gamma(m-\gamma)} * (D^m v - D^m \mathcal{P}_{N,\alpha,\beta} v) \right\|_{\mu,w^{(\alpha,\beta)}} \\ &\leqslant \frac{1}{(m-\gamma)\Gamma(m-\gamma)} \|D^m v - D^m \mathcal{P}_{N,\alpha,\beta} v)\|_{\mu,w^{(\alpha,\beta)}} \end{aligned}$$

$$\leqslant \frac{c'}{\Gamma(m-\gamma+1)}\Big((N+m)(N+m+\alpha+\beta)\Big)^{\frac{\mu-r}{2}}|v^{(m)}|_{r,w^{(\alpha,\beta)}}.$$

□

Now, we consider equation (1). If $u_N(t)$ is the approximate solution to $u(t)$, then the residual function is as follows:

$$H_N(t) = D^{\gamma}u_N(t) - au_N(t) - \sum_{i=1}^{l} b_i\ D^{\gamma_i}u_N(q_it) - f(t). \tag{27}$$

The next lemma estimates a bound for the residual function (27).

Lemma 6. *Consider the residual function given in (27). A bound for it can be estimated as follows:*

$$\begin{aligned}\|H_N(t)\|_{\mu,w^{(\alpha,\beta)}} \leqslant U^*\Bigg(&\frac{c'}{\Gamma(m-\gamma+1)}\Big((N+m)(N+m+\alpha+\beta)\Big)^{\frac{\mu-r}{2}}\\ &+\sum_{i=1}^{l}\frac{|b_i|\ c_i'}{\Gamma(m_i-\gamma_i+1)}\Big(q_i(N+m_i)(N+m_i+\alpha+\beta)\Big)^{\frac{\mu-r}{2}}\\ &+|a|c(N(N+\alpha+\beta))^{\frac{\mu-r}{2}}\Bigg),\end{aligned}$$

where $U^* = \max_{t\in J}|u^{(k)}(t)|_{r,w^{(\alpha,\beta)}}, k = 0,1,\ldots,m,\ m_i - 1 < \gamma_i \leqslant m_i,$ *and* $m-1<\gamma\leqslant m.$

Proof. By using Theorems 5 and 6 and (26), the above inequality can be resulted. □

The following theorem presents an error bound for the proposed method.

Theorem 7. *If* $u_N(t)$ *is the Jacobi approximate solution to* $u(t)$ *and* $e_N(t) = u(t) - u_N(t)$ *is the error function, then the error bound can be estimated as follows:*

$$\begin{aligned}\|e_N(t)\|_{\mu,w^{(\alpha,\beta)}} \leqslant U^*\Bigg(&\frac{(1/|a|+1)c'}{\Gamma(m-\gamma+1)}\Big((N+m)(N+m+\alpha+\beta)\Big)^{\frac{\mu-r}{2}}\\ &+\sum_{i=1}^{l}\frac{(1/|a|+1)|b_i|\ c_i'}{\Gamma(m_i-\gamma_i+1)}\Big(q_i(N+m_i)(N+m_i+\alpha+\beta)\Big)^{\frac{\mu-r}{2}}\\ &+c(N(N+\alpha+\beta))^{\frac{\mu-r}{2}}\Bigg).\end{aligned}$$

Proof. By subtracting equation (26) from equation (1), we get

$$D^{\gamma}e_N(t) = ae_N(t) + \sum_{i=1}^{l} b_i \; D^{\gamma_i}e_N(q_i t) - H_N(t).$$

Now, applying Theorems 5 and 6, equation (26), and Lemma 6 leads to the desired result. □

Now, we consider the operational matrix of fractional integration $\mathbf{P}^{(\nu)}$ in Theorem 2. We define the error vector E_ν as follows:

$$E_\nu = I^\nu \Phi^{(\sigma)}(t) - \mathbf{P}^{(\nu)}\Phi^{(\sigma)}(t) = [E_{0,\nu}, E_{1,\nu}, \ldots, E_{N,\nu}]^T,$$

where

$$E_{k,\nu} = I^\nu P_k^{(\alpha,\beta,\sigma)}(t) - \sum_{j=0}^{N} \mathbf{P}^{(\nu)}{}_{kj} P_j^{(\alpha,\beta,\sigma)}(t), \qquad k = 0, 1, \ldots, N.$$

The following theorem presents an upper bound of Riemann–Liouville fractional integral operator.

Theorem 8. *If $E_{k,\nu} = I^\nu P_k^{(\alpha,\beta,\sigma)}(t) - \sum_{j=0}^{N} \mathbf{P}^{(\nu)}{}_{kj} P_j^{(\alpha,\beta,\sigma)}(t)$, then an error bound of Riemann–Liouville fractional integral operator of order ν can be expressed by*

$$\begin{aligned}\|E_{k,\nu}\|^2_{\mu,w^{(\alpha,\beta)}} \leqslant & c^2(N(N+\alpha+\beta))^{\mu-r} \\ & \times \sum_{j=0}^{k}\sum_{i=0}^{k}\Bigg\{\frac{\gamma_j^{(k)}\gamma_i^{(k)}\Gamma(\sigma j+1)\Gamma(\sigma i+1)\Gamma(\sigma j+\nu+2)\Gamma(\sigma i+\nu+2)}{\Gamma(\sigma j+\nu+1)\Gamma(\sigma i+\nu+1)\Gamma(\sigma j+\nu-r+2)\Gamma(\sigma i+\nu-r+2)} \\ & \times B(i+j+r+\beta+2(\nu-r)/\sigma+1, \alpha+r+1)\Bigg\},\end{aligned}$$

where $\gamma_j^{(k)}$ were introduced by Lemma 1 and $B(u,v)$ is the well-known Beta function.

Proof. By using Lemma 1 and the property of Riemann–Liouville fractional integral, we have

$$I^\nu P_k^{(\alpha,\beta,\sigma)}(t) = \sum_{j=0}^{k} \gamma_j^{(k)} \frac{\Gamma(\sigma j+1)}{\Gamma(\sigma j+\nu+1)} t^{\sigma j+\nu}.$$

By setting $Z = I^\nu P_k^{(\alpha,\beta,\sigma)}(t)$, we get

$$\begin{aligned}|Z|^2_{\mu,w^{(\alpha,\beta)}} &= \|\partial_t^r Z\|^2_{\mu,w^{(\alpha+r,\beta+r)}} \\ &= \left\| \frac{\Gamma(\sigma j+1)\Gamma(\sigma j+\nu+2)}{\Gamma(\sigma j+\nu+1)\Gamma(\sigma j+\nu-r+2)} t^{\sigma j+\nu-r} \right\|^2_{\mu,w^{(\alpha+r,\beta+r)}} \\ &= \sum_{j=0}^{k}\sum_{i=0}^{k}\Bigg\{\frac{\sigma\gamma_j^{(k)}\gamma_i^{(k)}\Gamma(\sigma j+1)\Gamma(\sigma i+1)\Gamma(\sigma j+\nu+2)\Gamma(\sigma i+\nu+2)}{\Gamma(\sigma j+\nu+1)\Gamma(\sigma i+\nu+1)\Gamma(\sigma j+\nu-r+2)\Gamma(\sigma i+\nu-r+2)}\end{aligned}$$

$$\times \int_0^1 t^{\sigma(i+j+r+\beta+1)+2(\nu-r)-1}(1-t^\sigma)^{\alpha+r}dt\Big\}$$
$$= \sum_{j=0}^{k}\sum_{i=0}^{k}\Big\{\frac{\gamma_j^{(k)}\gamma_i^{(k)}\Gamma(\sigma j+1)\Gamma(\sigma i+1)\Gamma(\sigma j+\nu+2)\Gamma(\sigma i+\nu+2)}{\Gamma(\sigma j+\nu+1)\Gamma(\sigma i+\nu+1)\Gamma(\sigma j+\nu-r+2)\Gamma(\sigma i+\nu-r+2)}$$
$$\times \int_0^1 x^{i+j+r+\beta+2(\nu-r)/\sigma}(1-x)^{\alpha+r}dx\Big\}.$$

Therefore, the theorem is proved. □

8 Numerical examples

In this section, the method presented in this paper is applied to solve some examples taken from [18] to demonstrate the efficiency and reliability of the proposed algorithm. As a result, a comparison is presented.

Example 1. First, consider the following fractional pantograph differential equation from [10],

$$D^\gamma u = \frac{3}{4}u(t) + u(\frac{t}{2}) - t^2 + 2, \qquad 1 < \gamma \leqslant 2, \tag{28}$$

with conditions,

$$u(0) = 0, \qquad u'(0) = 0.$$

The exact solution is $u(t) = t^2$ in the case $\gamma = 2$. We first approximate the unknown function and its derivative of fractional order γ as follows:

$$u^\gamma(t) \simeq \Phi^{(\sigma)T}(t)\, C,\ \ u(t) \simeq \Phi^{(\sigma)T}(t)\, \mathbf{P}^{(\gamma)T}\, C,\ \ u(\frac{t}{2}) \simeq \Phi^{(\sigma)T}(t)\, \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\, \mathbf{P}^{(\gamma)\mathbf{T}}\, C,$$

Substituting the approximations into (28) leads to the following algebraic equation:

$$\Phi^{(\sigma)T}(t)\, C - \frac{3}{4}\Phi^{(\sigma)T}(t)\, \mathbf{P}^{(\gamma)T}\, C - \Phi^{(\sigma)T}(t)\, \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\, \mathbf{P}^{(\gamma)\mathbf{T}}\, C + t^2 - 2 \approx 0. \tag{29}$$

By choosing $N = 4$, equation (29) is collocated at roots of $P_5^{(\alpha,\beta)}(t)$. Then, by solving the resultant algebraic system, the unknown vector C can be determined for some values of α and β parameters. Table 1 displays the absolute errors at some arbitrary points for $\alpha = \beta = 0$, $\gamma = 2$, $\sigma = 0.5, 1, 1.5, 2$, and $N = 4$. Table 1 shows that the results of our algorithm are more accurate than those which reported by [18]. In Figure 1, the approximate solutions are compared to the exact solution for $\alpha = \beta = 0$, $N = 4$, $\gamma = 1.3, 1.5, 1.7, 1.9, 2$, and $\sigma = 1$. From these results, it is seen that the approximate solutions converge to the exact solution when γ approaches 2.

Table 1: Errors for $\alpha = \beta = 0$, $N = 4$, and various values of σ for Example 1

t_i	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$	$\sigma = 0.5$ in [18]	$\sigma = 1$ in [18]	$\sigma = 1.5$ in [18]
0.0	5.3967×10^{-2}	1.0616×10^{-5}	1.5319×10^{-3}	6.4935×10^{-5}	5.02×10^{-2}	1.64×10^{-1}	1.33×10^{-1}
0.2	9.5185×10^{-4}	5.6130×10^{-5}	4.2342×10^{-4}	1.1404×10^{-4}	$--$	$--$	$--$
0.4	1.8965×10^{-3}	9.4037×10^{-4}	2.0913×10^{-3}	9.7976×10^{-4}	3.08×10^{-2}	6.51×10^{-2}	1.17×10^{-2}
0.6	4.4029×10^{-3}	4.7621×10^{-3}	3.4555×10^{-3}	4.8110×10^{-3}	$--$	$--$	$--$
0.8	1.4375×10^{-2}	1.5160×10^{-2}	1.5934×10^{-2}	1.5206×10^{-2}	1.59×10^{-1}	2.60×10^{-2}	3.78×10^{-1}
1.0	3.8299×10^{-2}	3.7313×10^{-2}	3.1948×10^{-2}	1.5206×10^{-2}	$--$	$--$	$--$

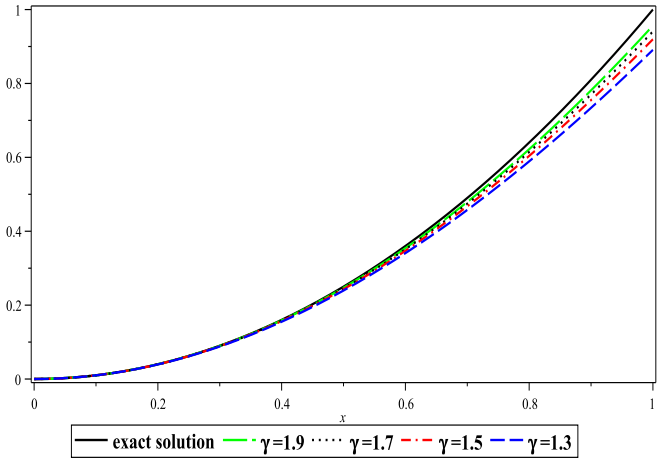

Figure 1: Plots of $u(t)$ for $N = 4$, $\sigma = 1$, $\gamma = 1.3, 1.5, 1.7, 1.9$ and exact solution for Example 1.

Example 2. Consider the following linear fractional-order pantograph differential equation of [18],

$$D^{\gamma}u(t) = -u(t) + 0.1u(\frac{4}{5}t) + 0.5D^{\gamma}u(\frac{4}{5}t) + (0.32t - 0.5)\exp(-0.8t) + \exp(-t),$$

where $0 < \gamma \leqslant 1$. With condition $u(0) = 0$ and the exact solution is $u(t) = t\exp(-t)$. By using the fundamental matrices obtained in Section 5, the following algebraic equation is resulted:

$$\begin{aligned}\Phi^{(\sigma)T}(t)\ C + \Phi^{(\sigma)T}(t)\ \mathbf{P}^{(\gamma)T}\ C - 0.1\Phi^{(\sigma)T}(t)\ \mathbf{L}_{(\frac{4}{5},\sigma)}{}^{T}\ \mathbf{P}^{(\gamma)\mathbf{T}}\ C \\ - 0.5\Phi^{(\sigma)T}(t)\ \mathbf{L}_{(\frac{4}{5},\sigma)}{}^{T}\ C - (0.32t - 0.5)\exp(-0.8t) - \exp(-t) \approx 0.\end{aligned} \tag{30}$$

By choosing $N = 10$, equation (30) is collocated at roots of $P_{11}^{(\alpha,\beta)}(x)$. Then, by solving the resultant algebraic system, the unknown vector C can be determined for some values of α and β parameters. Table 2 displays the maximum absolute errors for various values of α and β parameters, $N = 10$, and $\gamma = \sigma = 1$. The data of Table 2 shows that the approximate and exact solutions are in a good agreement. In Table 3, the values of absolute error are compared to the values reported in [18] at some arbitrary points for $N = 10$, $\alpha = \beta = 1/3$, and $\gamma = \sigma = 1$. It is seen that the errors of our method are lesser than the errors of methods applied in [18] on the interval $[0, 1]$. Also, the absolute errors are listed in Table 4 for $N = 10$, $\alpha = \beta = 1/3$, $\gamma = \sigma$ and various values of γ in some arbitrary points within interval $[0, 1]$. A graphical comparison between the exact and approximate solutions is depicted in part (a) of Figure 2 for values of $\alpha = \beta = 1/3$, $\gamma = \sigma = 1$, and $N = 10$. From part (b), it can be shown that the numerical results are well consistent with the analytical solution. In addition, the plots of $u(t)$ are shown in part (b) of Figure 2 for $\sigma = 1$ and various values of γ. It is seen that the approximate solutions converge to the exact solution as γ approaches 1.

Table 2: Maximum absolute errors for various values of α and β, $\gamma = \sigma = 1$, and $N = 10$ for Example 2

(α, β)	$Error_{Abs}$	(α, β)	$Error_{Abs}$
$(0, 0)$	2.2652×10^{-13}	$(\frac{1}{2}, \frac{1}{2})$	4.5600×10^{-13}
$(1, 1)$	7.6421×10^{-13}	$(-\frac{1}{3}, -\frac{1}{3})$	1.1765×10^{-13}
$(-\frac{2}{3}, -\frac{2}{3})$	8.6704×10^{-14}	$(\frac{2}{3}, \frac{2}{3})$	5.5014×10^{-13}
$(\frac{3}{2}, \frac{3}{2})$	1.1469×10^{-12}	$(2, 2)$	1.5979×10^{-12}

Table 3: Comparison of absolute errors with other methods for Example 2

t_i	present method	Bernoulli method in [18]	Runge–Kutta method in [18]	Variational iteration method in [18]	θ-method in [18]
0.1	6.4471×10^{-14}	1.98×10^{-8}	8.68×10^{-4}	1.30×10^{-3}	4.65×10^{-3}
0.3	4.4701×10^{-14}	7.78×10^{-9}	1.90×10^{-3}	2.63×10^{-3}	2.57×10^{-2}
0.5	4.7819×10^{-16}	6.34×10^{-5}	2.28×10^{-3}	2.83×10^{-3}	4.43×10^{-2}
0.7	4.5170×10^{-14}	4.36×10^{-5}	2.28×10^{-3}	2.39×10^{-3}	5.37×10^{-2}
0.9	6.4785×10^{-14}	2.80×10^{-5}	2.03×10^{-3}	1.64×10^{-3}	5.35×10^{-2}

Example 3. In current example, we consider the following fractional pantograph differential equation of [18] :

$$D^{\gamma}u(t) = \frac{3}{4}u(t) + u(\frac{t}{2}) + D^{\gamma_1}u(\frac{t}{2}) + \frac{1}{2}D^{\gamma}u(\frac{t}{2}) - t^2 - t + 1, \qquad (31)$$

Table 4: Absolute errors for various values of $\gamma = \sigma$, $\alpha = \beta = \frac{1}{3}$, and $N = 10$ for Example 2

t_i	$\gamma = 0.6$	$\gamma = 0.7$	$\gamma = 0.8$	$\gamma = 0.9$
0.2	2.5022×10^{-5}	1.1429×10^{-4}	2.6347×10^{-4}	3.3616×10^{-4}
0.4	1.0801×10^{-5}	5.3195×10^{-5}	1.3757×10^{-4}	1.9924×10^{-4}
0.6	6.3865×10^{-6}	3.1052×10^{-5}	7.9303×10^{-5}	1.1741×10^{-4}
0.8	4.2784×10^{-6}	1.8443×10^{-5}	4.4219×10^{-5}	6.5747×10^{-5}
1.0	5.0278×10^{-6}	2.0332×10^{-5}	4.6987×10^{-5}	6.8103×10^{-5}

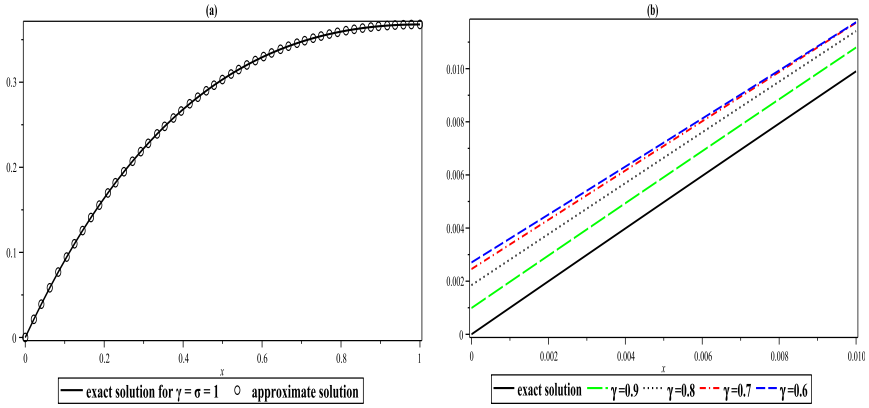

Figure 2: Plots of (a) exact and approximate solutions, (b) $u(t)$ for $\alpha = \beta = \frac{1}{3}$, $\sigma = 1$, $\gamma = 0.6, 0.7, 0.8, 0.9$, and $N = 10$ for Example 2.

where $0 < \gamma_1 < \gamma \leqslant 2$, with conditions $u(0) = u'(0) = 0$. The exact solution is $u(t) = t^2$ for case $\gamma = 2$ and $\gamma_1 = 1$. Substituting the approximations introduced in the previous sections into (31), leads to the following algebraic equation:

$$\Phi^{(\sigma)T}(t)\ C - \frac{3}{4}\Phi^{(\sigma)T}(t)\ \mathbf{P}^{(\gamma)T}\ C - \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ \mathbf{P}^{(\gamma)\mathbf{T}}\ C$$
$$- \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ \mathbf{P}^{(\gamma-\gamma_1)\mathbf{T}}\ C - \frac{1}{2}\Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ C + t^2 + t - 1 \approx 0. \tag{32}$$

By choosing $N = 10$, equation (32) is collocated at roots of $P_{11}^{(\alpha,\beta)}(x)$. Then, by solving the resultant algebraic system, the unknown vector C can be determined for some values of α and β parameters. Table 5 displays the maximum absolute errors for various values of α and β parameters, $\gamma = 2$, $\gamma_1 = 1$, $\sigma = 1$, and $N = 10$. The approximate solutions are in good agreement with exact solution. Equation (31) was solved in [18] by Bernoulli collocation method. In Table 6, the absolute errors are compared to the values reported

by [18] at some arbitrary points for $N = 10$, $\alpha = \beta = 2$, $\gamma = 2$, $\gamma_1 = 1$, $\sigma = 1$. Table 7 shows the maximum absolute errors for $\alpha = \beta = 0$, $\gamma = 1.9, 2$, $\gamma_1 = 0.9, 1$, and $\sigma = 0.5, 1, 1.5, 2$. For the case $\gamma = 1.9$, $\gamma_1 = 0.9$, and the case $\gamma = 2$, $\gamma_1 = 1$, the best results obtain for $\sigma = 0.5$ and $\sigma = 1$, respectively. The errors given in [18] are greater than the errors of our method at the selected points on the interval $[0, 0.5]$. A graphical comparison between the exact and approximate solutions for values of $\alpha = \beta = 0$, $\gamma_1 = \sigma = 1$, $\gamma = 1.2, 1.4, 1.6, 1.8$, and $N = 10$ are shown in Figure 3.

Table 5: Maximum absolute errors for various values of α and β, $\gamma = 2$, $\gamma_1 = \sigma = 1$, and $N = 10$ for Example 3

(α, β)	$Error_{Abs}$	(α, β)	$Error_{Abs}$
$(0, 0)$	7.2345×10^{-22}	$(\frac{1}{2}, \frac{1}{2})$	1.9150×10^{-19}
$(1, 1)$	1.8407×10^{-20}	$(-\frac{4}{5}, -\frac{4}{5})$	8.7002×10^{-22}
$(-\frac{1}{3}, -\frac{1}{3})$	2.2935×10^{-20}	$(2, 2)$	1.2297×10^{-20}

Table 6: Comparison of absolute errors with other methods for Example 3

t_i	present method	Bernoulli method in [18]	Runge–Kutta method in [18]	Variational iteration method in [18]	θ-method in [18]
0.1	0.0000	3.47×10^{-16}	1.00×10^{-3}	1.67×10^{-4}	6.10×10^{-3}
0.2	1.0000×10^{-21}	5.55×10^{-16}	2.02×10^{-3}	7.15×10^{-4}	2.58×10^{-2}
0.3	2.0000×10^{-21}	3.05×10^{-16}	3.07×10^{-3}	1.73×10^{-3}	6.47×10^{-2}
0.4	0.0000	1.14×10^{-15}	4.17×10^{-3}	3.30×10^{-3}	1.37×10^{-2}
0.5	1.0000×10^{-20}	1.39×10^{-13}	5.34×10^{-3}	5.50×10^{-3}	2.81×10^{-2}

Table 7: Comparison of maximum absolute errors for different values of γ, γ_1, and σ for Example 3

	$\gamma = 1.9$	$\gamma_1 = 0.9$	
$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$
2.2058×10^{-3}	1.0822×10^{-2}	1.0081×10^{-2}	1.5358×10^{-1}
	$\gamma = 2$	$\gamma_1 = 1$	
$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$
3.4229×10^{-18}	7.2345×10^{-22}	6.8343×10^{-3}	1.1564×10^{-1}

Example 4. Consider the following pantograph fractional differential equation [18] :

$$D^{\gamma}u(t) = -\frac{5}{6}u(t) + 4u(\frac{t}{2}) + 9u(\frac{t}{3}) + t^2 - 1, \qquad 0 < \gamma \leqslant 1.$$

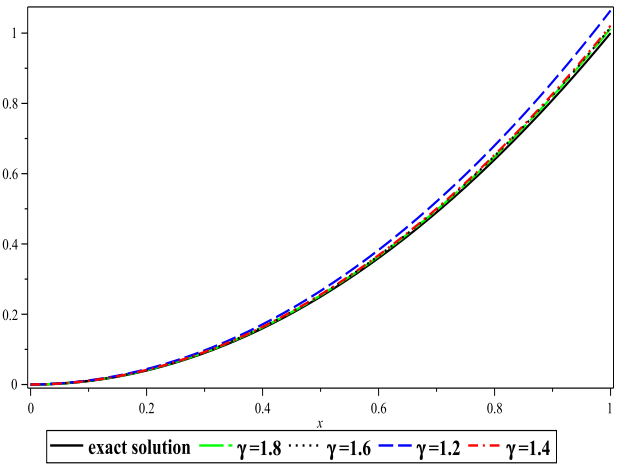

Figure 3: Comparison of exact and approximate solutions for $\gamma_1 = \sigma = 1$, $\gamma = 1.2, 1.4, 1.6, 1.8$, and $N = 10$ for Example 3.

The initial condition is $u(0) = 1$. The exact solution is $u(t) = 1 + \frac{67}{6}x + \frac{1675}{72}x^2 + \frac{12157}{1296}x^3$ in the case $\gamma = 1$. By applying the fundamental matrices presented in the previous sections, we get the following algebraic equation:

$$\begin{aligned}\Phi^{(\sigma)T}(t)\ C + \frac{5}{6}\{\Phi^{(\sigma)T}(t)\ \mathbf{P}^{(\gamma)T}\ C + \Phi^{(\sigma)T}(t)\ F\} - 4\{\Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ \mathbf{P}^{(\gamma)\mathbf{T}}\ C \\ + \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ F\} - 9\{\Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{3},\sigma)}\ \mathbf{P}^{(\gamma)\mathbf{T}}\ C \\ + \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{3},\sigma)}\ F\} - t^2 + 1 \approx 0.\end{aligned} \tag{33}$$

where the matrices $\mathbf{L}_{(\frac{1}{2},\sigma)}$ and $\mathbf{L}_{(\frac{1}{2},\sigma)}$ are pantograph operational matrices corresponding to the delays $q = \frac{1}{2}, \frac{1}{3}$ and $u(0)$ is approximated as $\Phi^{(\sigma)T}(t)\ F$. By choosing $N = 10$, equation (33) is collocated at roots of $P_{11}^{(\alpha,\beta)}(x)$. By solving the resultant algebraic systems, the unknown vector C can be determined for some values of α and β parameters. In Table 8, the absolute errors of the approximate solutions are listed at some arbitrary points for $N = 10$, $\alpha = \beta = \gamma = 1$, and various values of σ. As it is seen, the best results are related to the values $\sigma = 0.5, 1$. A graphical comparison between the exact and approximate solutions is found in part (a) of Figure 4 for $\alpha = \beta = \gamma = \sigma = 1$. The plots of $u(t)$ have been compared in part (b) for $\gamma = \sigma = 0.75, 0.85, 0.95$. Also, the error functions of these solutions are depicted in part (c). It is seen that the approximate solutions converge to the exact solution as γ and σ approach 1.

Table 8: Absolute errors for various values of σ, $\alpha = \beta = \gamma = 1$, and $N = 10$ for Example 4

t_i	$\sigma = 0.5$	$\sigma = 0.75$	$\sigma = 1$	$\sigma = 1.5$
0.2	1.5100×10^{-15}	1.1748×10^{-3}	2.0000×10^{-19}	4.5227×10^{-1}
0.4	3.5176×10^{-15}	2.3520×10^{-3}	9.0000×10^{-19}	1.0366×10^{-0}
0.6	6.5190×10^{-15}	4.5162×10^{-3}	0.0000	1.9337×10^{-0}
0.8	1.0666×10^{-14}	7.3187×10^{-3}	5.0000×10^{-18}	3.1543×10^{-0}
1.0	1.6149×10^{-14}	9.6215×10^{-3}	3.0000×10^{-18}	4.7377×10^{-0}

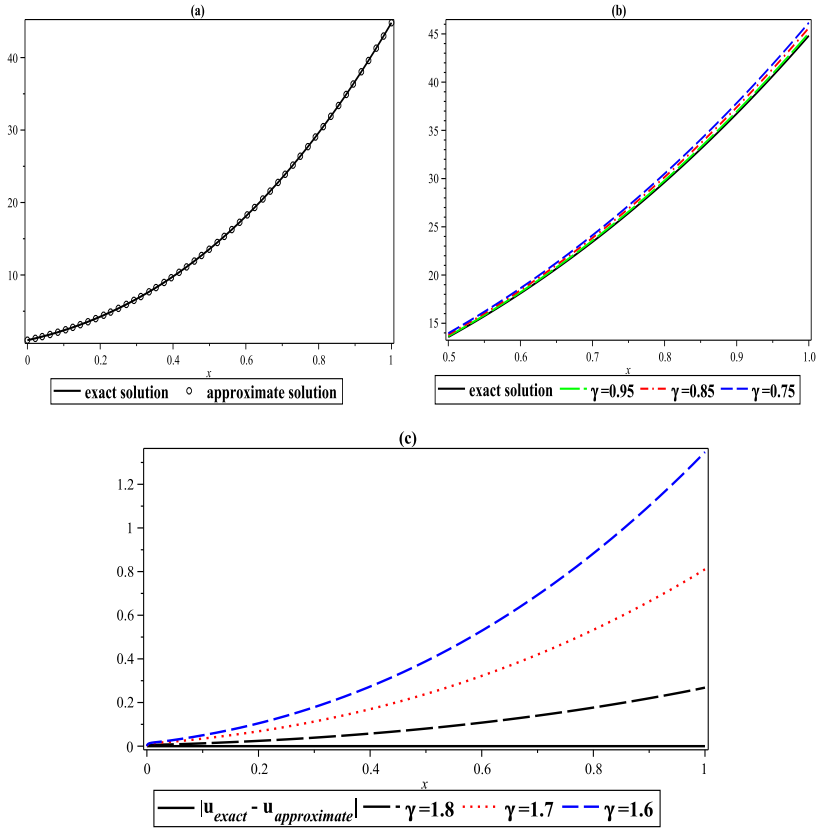

Figure 4: (a) Exact and approximate solutions for $\alpha = \beta = \gamma = \sigma = 1$, and $N = 10$, (b) comparison of u(t) for $\gamma = \sigma = 0.75, 0.85, 0.95, 1$, ($c$) maximum absolute errors for $\gamma = \sigma = 0.75, 0.85, 0.95$ for Example 4.

Table 9: Absolute errors for various values of σ, and $N = 10$ for Example 5

t_i	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$
0.2	2.2499×10^{-18}	0.0000	5.0015×10^{-5}	2.4119×10^{-2}
0.4	1.3489×10^{-17}	1.0000×10^{-21}	6.8587×10^{-5}	7.0770×10^{-2}
0.6	3.9130×10^{-17}	0.0000	5.8770×10^{-6}	1.6689×10^{-1}
0.8	8.4580×10^{-17}	4.0000×10^{-20}	2.1854×10^{-4}	3.1842×10^{-1}
1.0	1.5572×10^{-16}	1.0000×10^{-20}	2.3271×10^{-4}	5.5809×10^{-1}

Example 5. In this example, we consider the following fractional pantograph differential equation of [18] :

$$D^{\gamma}u(t) = u(t) + D^{\gamma_1}u(\frac{t}{2}) + D^{\gamma_2}u(\frac{t}{3}) + \frac{1}{2}D^{\gamma}u(\frac{t}{4}) - t^4 - \frac{1}{2}t^3 - \frac{4}{3}t^2 + 21t,$$

where $0 < \gamma_1 < \gamma_2 \leqslant \gamma \leqslant 3$. The initial conditions are $u(0) = u'(0) = u''(0) = 0$ and the exact solution is $u(t) = t^4$ in case $\gamma_1 = 1$, $\gamma_2 = 2$, and $\gamma = 3$. By using the approximations introduced in the previous sections, we get the following algebraic equation:

$$\begin{aligned} &\Phi^{(\sigma)T}(t)\ C - \Phi^{(\sigma)T}(t)\ \mathbf{P}^{(\gamma)T}\ C - \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\ \mathbf{P}^{(\gamma-\gamma_1)\mathbf{T}}\ C \\ &\quad - \Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{3},\sigma)}\ \mathbf{P}^{(\gamma-\gamma_2)\mathbf{T}}\ C - \frac{1}{2}\Phi^{(\sigma)T}(t)\ \mathbf{L}^{\mathbf{T}}_{(\frac{1}{4},\sigma)}\ C \\ &\quad - t^4 + \frac{1}{2}t^3 + \frac{4}{3}t^2 - 21t \approx 0. \end{aligned} \tag{34}$$

By choosing $N = 10$, equation (34) is collocated at roots of $P_{11}^{(\alpha,\beta)}(x)$. Then, by solving the resultant algebraic system, the unknown vector C is determined for some values of α and β parameters. Table 9 displays the absolute errors for $\alpha = \beta = 0$, $\gamma = 3$, $\gamma_2 = 2$, $\gamma_1 = 1$, and various values of σ. In Table 10, the values of the approximate solutions are shown at some arbitrary points for $N = 10$, $\alpha = \beta = 0$, $\gamma = 3$, $\gamma_2 = 2$, $\gamma_1 = 1$, and $\sigma = 1$. A graphical comparison between the exact and approximate solutions is found in part (a) of Figure 5 for $\alpha = \beta = 0$, $\sigma = \gamma_1 = 1$, $\gamma_2 = 2$, and $\gamma = 3$. The plots of $u(t)$ have been compared in part (b) for $\gamma = 2.2, 2.4, 2.6, 2.8$, $\sigma = \gamma_1 = 1$, and $\gamma_2 = 2$. It is seen that the approximate solutions converge to the exact solution as γ approaches 3.

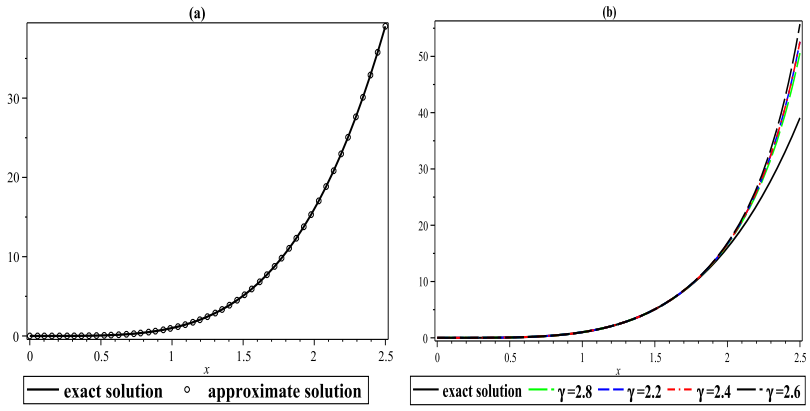

Figure 5: Comparison of (a) exact and approximate solutions, (b) $u(t)$ for $\alpha = \beta = 0$, and $N = 10$, $\gamma_2 = 2$, $\gamma_1 = 1$, and $\sigma = 1$, and $\gamma = 2.2, 2.4, 2.6, 2.8$ for Example 5.

Table 10: Comparison of absolute errors with other methods for Example 5

t_i	present method	Bernoulli method in [18]	Runge–Kutta method in [18]	Variational iteration method in [18]
0.2	0.0000	5.31×10^{-13}	4.43×10^{-4}	2.98×10^{-10}
0.4	1.0000×10^{-21}	1.24×10^{-12}	3.85×10^{-3}	1.01×10^{-8}
0.6	0.0000	2.89×10^{-10}	1.39×10^{-2}	8.24×10^{-8}
0.8	4.0000×10^{-20}	6.49×10^{-10}	3.53×10^{-2}	3.76×10^{-7}
1.0	1.0000×10^{-20}	1.23×10^{-9}	7.34×10^{-2}	1.26×10^{-6}

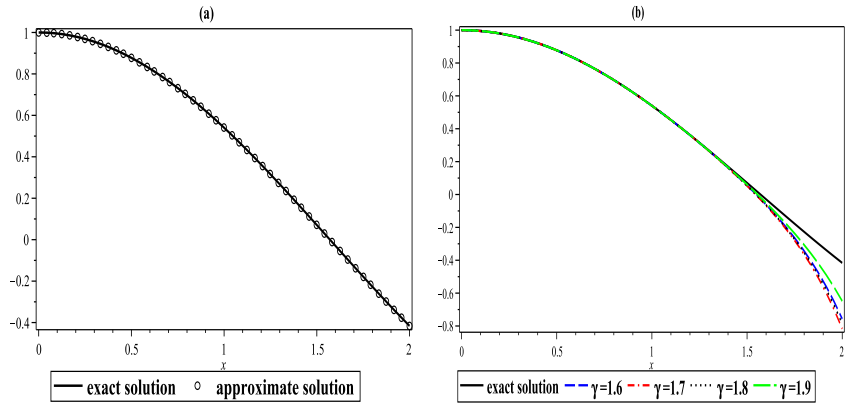

Figure 6: Comparison of (a) exact and approximate solutions, (b) $u(t)$ for $\alpha = \beta = -\frac{1}{4}$, and $N = 7$, $\sigma = \frac{1}{2}$, and $\gamma = 1.6, 1.7, 1.8, 1.9$ for Example 6.

Example 6. Consider the following nonlinear fractional pantograph differential equation of [18] :

$$D^{\gamma}u(t) = 1 - 2u^2(\frac{t}{2}), \qquad 1 < \gamma \leqslant 2.$$

The initial conditions are $u(0) = 1$, $u'(0) = 0$ and the exact solution is $u(t) = \cos(t)$ in case $\gamma = 2$. By using the approximations introduced in the previous sections, we get the following algebraic equation:

$$\Phi^{(\sigma)T}(t)\, C + 2U^T\, \tilde{U}\, \Phi^{(\sigma)}(t) - 1 \approx 0, \qquad U = \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\, \mathbf{P}^{(\gamma-)\mathbf{T}}\, C + \mathbf{L}^{\mathbf{T}}_{(\frac{1}{2},\sigma)}\, F, \quad (35)$$

where $u(0) = 1$ is approximated by $\Phi^{(\sigma)T}(t)\, F$ and $\tilde{U}$ is the operational matrix of product corresponding to the vector U. By choosing $N = 7$, equation (35) is collocated at roots of $P_8^{(\alpha,\beta)}(x)$. Then, by solving the resultant algebraic system by means of Newton's iteration method, the unknown vector C is determined for some values of α and β parameters. Table 11 displays the absolute errors for $\alpha = \beta = 3/2$, $\gamma = 2$, and various values of σ. In Table 12, the values of the approximate solutions are shown at some arbitrary points for $N = 7$, $\alpha = \beta = -1/4$, $\sigma = 1/2$, and various values of γ. A graphical comparison between the exact and approximate solutions is found in part (a) of Figure 6 for $\alpha = \beta = -1/4$, $\sigma = 1$, and $\gamma = 2$. The plots of $u(t)$ have been compared in part (b) for $\gamma = 1.6, 1.7, 1.8, 1.9$, and $\sigma = 1/2$. It is seen that the approximate solutions converge to the exact solution as γ approaches 2.

Table 11: Absolute errors for various values of σ, and $N = 7$ for Example 6

t_i	$\sigma = 0.5$	$\sigma = 1$	$\sigma = 1.5$	$\sigma = 2$
0.0	2.4053×10^{-7}	6.0568×10^{-9}	5.6194×10^{-5}	1.1748×10^{-2}
0.2	4.0376×10^{-8}	7.7581×10^{-11}	2.3175×10^{-3}	1.0518×10^{-2}
0.4	4.1630×10^{-8}	4.1069×10^{-10}	1.7587×10^{-3}	7.6767×10^{-3}
0.6	8.1380×10^{-8}	5.3919×10^{-10}	9.8358×10^{-4}	3.3268×10^{-3}
0.8	2.3864×10^{-7}	2.3463×10^{-10}	3.8368×10^{-4}	2.4282×10^{-3}
1.0	5.8922×10^{-6}	5.4779×10^{-9}	4.2678×10^{-4}	9.5443×10^{-3}

Table 12: Absolute errors for various values of γ, and $N = 7$ for Example 6

t_i	$\gamma = 1.6$	$\gamma = 1.7$	$\gamma = 1.8$	$\gamma = 1.9$
0.0	1.9622×10^{-7}	1.1006×10^{-7}	2.1920×10^{-7}	5.1996×10^{-8}
0.2	1.1452×10^{-5}	2.9183×10^{-5}	4.5715×10^{-5}	2.5122×10^{-5}
0.4	1.4694×10^{-5}	4.2206×10^{-5}	7.3289×10^{-5}	7.9829×10^{-5}
0.6	1.4420×10^{-5}	4.6407×10^{-5}	8.8547×10^{-5}	1.0437×10^{-5}
0.8	1.0683×10^{-5}	4.2124×10^{-5}	9.0686×10^{-5}	1.1652×10^{-4}
1.0	4.2390×10^{-6}	3.0183×10^{-5}	7.9795×10^{-5}	11529×10^{-4}

9 Conclusion

In this paper, the Jacobi operational method was used to numerically solve the linear and nonlinear fractional pantograph differential equations. This approach was based on the shifted fractional-order Jacobi functions on the interval $[0, 1]$, which converted a given linear or nonlinear fractional pantograph differential equation into a set of linear or nonlinear algebraic equations. Another advantage of the proposed method was that the coefficients of the Jacobi series solution were easily found by using the presented algorithms. These algorithms decrease the computational costs compared to other methods such as the Bernoulli collocation method as it applies the lesser numbers of operational matrices. By using the introduced algorithms, the operational matrices of fractional integration, product, and pantograph were simply constructed. The illustrative examples confirmed the validity and simplicity of the proposed approach. It was found that the absolute error of this method is lesser than the errors of other common methods and the errors of our proposed method were almost constant on the studied interval. This demonstrates the stability of the applied Jacobi collocation method. Illustrative examples with satisfactory results were used to demonstrate the application of this method. Suggested approximations make this method very interesting and effective to the good agreement between approximate and exact solutions. Almost the results obtained by the technique developed

in this paper were more accurate than the results obtained by the Bernoulli collocation, Runge–Kutta, variational iteration, and θ- methods.

Acknowledgements

The authors are grateful to the referees for many useful comments and suggestions which have improved the presentation of this paper.

References

1. Bagley, R.L. and Torvik, P.J. *Fractional calculus in the transient analysis of viscoelastically damped structures*, J. AIAA., 23 (1985), 918–925.

2. Bhrawy, A.H. and Zaky, M.A. *Shifted fractional–order Jacobi orthogonal functions: Application to a system of fractional differential equations*, Appl. Math. Model. 40 (2016), 832–845.

3. Borhanifar, A. and Sadri, K. *A new operational approach for numerical solution of generalized functional integro-differential equations*, J. Comput. Appl. Math. 279 (2015), 80–96.

4. Dehghan, M., Hamedi, E. A. and Khosravian-Arab, H. *A numerical scheme for the solution of a class of fractional variational and optimal control problems using the modified Jacobi polynomials*, J.Vib. .Control, 22 (6) (2014), 1547–1559.

5. Doha, E.H., Bhrawy, A.H., and Ezz-Eldien, S.S. *A new Jacobi operational matrix: An application for solving fractional differential equations*, Appl. Math. Model. 36 (10) (2012), 4931–4943.

6. Duan, B., Zheng, Z. and Cao, W. *Spectral approximation methods and error estimates for Caputo fractional derivative with applications to initial-value problems*, J. Comput. Phys. 319 (2016), 108–128.

7. Engheta, N. *On fractional calculus and fractional multipoles in electromagnetism*, IEEE Trans. Antennas and Propagation 44 (1996), 554–566.

8. Eshaghi , J., Adibi, H., and Kazem, K. *Solution of nonlinear weakly singular Volterra integral equations using the fractional-order Legendre functions and pseudospectral method*, Math. Methods Appl. Sci. 39 (12) (2015), 3411–3425.

9. Esmaeil, S., Shamsi, M. and Luchko, Y. *Numerical solution of fractional differential equations with a collocation method based on Muntz polynomials*, Comp. Math. Appl. 62 (3) (2011), 918–929.

10. Guo, B.Y. and Wang, L.L. *Jacobi approximations in non-uniformly Jacobi-weighted Sobolev spaces*, J. Approx. Theory,. 128 (2004), 1–41.

11. Kazem, S. *An integral operational matrix based on Jacobi polynomials for solving fractional-order differential equations*, Appl. Math. Model. . 37 (3) (2013), 1126–1136.

12. Kazem, S., Shaban, M. and Amani Rad, J. *Solution of the coupled Burgers equation based on operational matrices of d-dimensional orthogonal functions*, Zeitschrift fr Naturforschung A, 67 (2012), 267–274.

13. Kulish, V.V., Lage, J.L. *Application of fractional calculus to fluid mechanics*, J. Fluids. Eng., 124(3) (2002), 803–806.

14. Magin, R.L. *Fractional calculus models of complex dynamics in biological tissues*, Comput. Math. Appl. 59 (2010), 1586–1593.

15. Mainardi, F. *Fractional calculus: some basic problems in continuum and statistical mechanics*, Fracta. Fracti. Calcu. Continu. Mecha., 378 (1997), 291–348.

16. Meral, F.C., Royston, T.J. and Magin, R. *Fractional calculus in viscoelasticity: an experimental study*, Commun. Nonlinear Sci. Numer. Simul.15 (2010), 939–945.

17. Mokhtary, P., Ghoreishi, F. and Srivastava, H.M. *The Muntz–Legendre Tau Method for Fractional Differential Equations*, Appl. Math. Model. 40 (2016), 671–684.

18. Rahimkhani, P., Ordokhani, Y. and Babolian, E. *Numerical solution of fractional pantograph differential equations by using generalized fractional-order Bernoulli wavelet*, J. Comput. Appl. Math.309 (2017), 493–510.

19. Sadri, K., Amini, A. and Cheng, C. *Low cost numerical solution for three-dimensional linear and nonlinear integral equations via three-dimensional Jacobi polynomials*, J. Comput. Appl. Math. 319 (2017), 493–513 .

20. Saeed, U. and Rehman, M. *Hermite wavelet method for fractional delay differential equations*, J. Difference Equ., doi: 10.1155/2014/359093, (2014), 8 pp.

21. Sedaghat, S., Ordokhani, Y., and Dehghan, M. *Numerical solution of the delay differential equations of pantograph type via Chebyshev polynomials*, Commun. Nonlinear Sci. Numer. Simul. 17 (2012), 4815–4830.

22. Szego, G. *Orthogonal polynomials*, American Mathematical Society. Providence, Rhodes Island, 1939.

23. Tohidi, E., Bhrawy, A.H. and Erfani, K.A. *Collocation method based on Bernoulli operational matrix for numerical solution of generalized pantograph equation*, Appl. Math. Model. 37 (2012), 4283–4294.

24. Yang, Y. and Huang, Y. *Spectral-collocation methods for fractional pantograph delay integro-differential equations*, Adv. Math. Phys., doi: 10.1155/2013/821327, (2013) 14 pp.

25. Yousefi, S.A. and Lotfi, A. *Legendre multiwavelet collocation method for solving the linear fractional time delay systems*, Cent. Eur. J. Phys. 11(10) (2013), 1463–1469.

26. Yu, Z.H. *Variational iteration method for solving the multi–pantograph delay equation*, Phys. Lett. A., 372 (2008) , 6475–6479.

27. Yuzbasi, S. *Numerical solutions of fractional Riccati type differential equations by means of the Bernstein polynomials*, Appl. Math. Comput. 219 (2013), 6328–6343.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 69–92
DOI:10.22067/ijnao.v9i1.68851

The network 1-median problem with discrete demand weights and traveling times

M. Abareshi and M. Zaferanieh*

Abstract

In this paper, the 1-median location problem on an undirected network with discrete random demand weights and traveling times is investigated. The objective function is to maximize the probability that the expected sum of weighted distances from the existing nodes to the selected median does not exceed a prespecified given threshold. An analytical algorithm is proposed to get the optimal solution in small-sized networks. Then, by using the central limit theorem, the problem is studied in large-sized networks and reduced to a nonlinear problem. The numerical examples are given to illustrate the efficiency of the proposed methods.

Keywords: Facility location; 1-median problem; probabilistic weights; probabilistic traveling times.

1 Introduction

In the classical deterministic p-median problem, the purpose is to locate p new facilities on the links or nodes of a given network G, so that the total weighted distances from all nodes that are considered as demand points to the nearest facility, is minimized. The p-median problem on general networks is NP-hard; see [8]. In a tree network, Kariv and Hakimi [12] proved that the problem can be solved in $O(p^2n^2)$ time. Tamir [18] improved the

*Corresponding author
Received 21 November 2017; revised 25 July 2018; accepted 29 August 2018
M. Abareshi
Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Hakim Sabzevari University, Sabzevar, Iran. e-mail: abareshi66@gmail.com

M. Zaferanieh
Department of Applied Mathematics, Faculty of Mathematics and Computer Sciences, Hakim Sabzevari University, Sabzevar, Iran. e-mail: m.zaferanieh@hsu.ac.ir

time complexity on tree networks to $O(pn^2)$. Hakimi [10] demonstrated that the classical 1-median problem has nodal optimal solution, and Goldman [9] presented a linear time algorithm to find the optimal solution. For the case $p = 2$, an $O(n \log n)$ time algorithm was provided by Gavish and Sridhar in [7].

Determining precise demand weight of each node may be impossible. For example, the number of patients arising from different regions going into a hospital cannot be predicted precisely. Therefore, it is desirable to treat demand weights as random variables. Frank considered a probabilistic case, in which the demand weights are random variables with arbitrary known or estimated probability distributions, [5]. He discussed the median and center problems on a network with independent random demand weights and extended the classical 1-median problem to the maximum probability absolute median ($MPAM$) problem. The $MPAM$ problem seeks to find a new facility maximizing the probability that the total weighted distances from all nodes to the new facility is less than or equal to a given threshold. He also defined a maximum probability center to maximize the probability that the maximum weighted distance is less than a prespecified threshold value. Later, the results for the maximum probability median were extended by Frank to the case where demand weights are correlated normal random variables, [6].

Berman and Wang [2] extended the problem to some single facility location problems as maximum probability 1-median, 1-anti-median, 1-center, and 1-anti-center problems with independent discrete random weights and presented efficient algorithms to solve these problems. In [3], They also studied the p-facility location problem with maximum probability and formulated it as a linear integer programming problem, then determined the solutions in large-scaled networks by using the normal approximation.

Puga and Tancrez [16] studied a location-inventory problem for the design of large supply chain networks with uncertain demand and proposed a heuristic algorithm to solve the problem. Bieniek [4] considered the single source capacitated facility location problem with independent and identically distributed random demands. The unified of a priori solution for the locations of facilities and for the allocation of customers to the operating facilities was found.

The distances between nodes can also be considered as nondeterministic parameters. The uncertainty about the lengths of the links of a network often occurs when the lengths are measured in units of traveling times instead of geographical distances. Mirchandani and Odoni studied the p-median problem when traveling times on the links are discrete random variables, [13]. The p-median problem with continuous random lengths of links, was studied by Handler and Mirchandani, [11]. They formulated the problem for locating p medians and provided an algorithm for the 1-median problem. The uncertainty in link traveling times can also be considered in some other location problems. Berman et al [1] studied the maximum covering location problem

on a network with traveling time uncertainty represented by different traveling time scenarios. The proposed models were applied to the analysis of the location of fire stations in Toronto, Canada.

In this paper, a new case is proposed in which the weights of nodes and the traveling times on links are both discrete independent random variables. The problem is to find the location of the new facility in such a way the expected sum of weighted distances is less than a prespecified value with maximum probability. The problem is studied in both small and large-sized networks. In Section 2, the problem formulation is stated and an expected 1-median with the maximum probability is defined. The solution approach for small-sized networks is stated in Section 3, and a precise algorithm to get the optimal solution is presented; then a numerical example is provided. In Section 4, by using the normal approximation, the solution is determined in large-sized networks and a numerical example is presented. Summary and conclusions are stated in the last section.

2 Problem formulation

Let $G = (N, E)$ be a network, where N is the set of nodes and E is the set of links with $|N| = n$ and $|E| = m$. The weight of each node $v_h \in N, \quad h = 1, \ldots, n$ is denoted by w_h. Let the link connecting two nodes v_i and v_j and its length be denoted by e_{ij} and l_{ij}, respectively. In a deterministic network, the shortest distance between two points $y, x \in G$ is denoted by $d(y, x)$. Let x be a point on the link e_{ij} at distance x from node v_i; then the shortest distance between x and an arbitrary point y is given as follows:

$$d(y, x) = \min\{d(y, v_i) + x, d(y, v_j) + l_{ij} - x\}. \tag{1}$$

Hence the shortest distance from point y to $x \in e_{ij}$ is covered via node v_i or node v_j, whichever that is shorter. In a stochastic network, each link e_{ij} has a deterministic geographical length l_{ij}, that is a real positive fixed value. Moreover the traveling time along the link e_{ij} is a random variable dependent on the traveling speed. In this case, the distances are computed in terms of traveling times. It is clear that the traveling speed along each link may be changed under such various conditions as traffic congestion.

To introduce the traveling time of a link, we use the inverse of traveling speed. Let the traveling time and inverse of traveling speed along the link e_{ij} be denoted by T_{ij} and SP_{ij}, respectively. If SP_{ij} is a discrete random variable, T_{ij} is also a discrete random variable equal to $SP_{ij}l_{ij}$. In this paper, the random values for the inverse of traveling speed of the link e_{ij} are considered to be independent from the other links and chosen from the set $\{sp_{ij}^1, sp_{ij}^2, \ldots, sp_{ij}^{U_{ij}}\}$, where U_{ij} is the number of all states of traveling

speed in the link e_{ij}. The relevant probability to sp_{ij}^u on the link e_{ij} is P_{ij}^u for $u = 1, \ldots, U_{ij}$.

Because of random values for traveling times, the network G gets into different state-based networks G_r. Let $G_{state} = \{G_r,\ r = 1, \ldots, R\}$, where $R = \prod_{e_{ij} \in E} U_{ij}$. We define

$$WD_r(x) = \sum_{h=1}^{n} w_h d_r(v_h, x), \quad r = 1, \ldots, R \tag{2}$$

as the total weighted traveling time from all nodes to the point x in the rth state $G_r \in G_{state}$, in which $d_r(v_h, x)$ is the distance between v_h and $x \in G_r$, that is determined by equation (1) noting that the traveling time of each link $e_{ij} \in E$ is computed by $sp_{ij}^r l_{ij}$. Then the expected 1-median problem is defined as below.

Definition 1. *The expected* 1*-median problem in network* G *with probabilistic distances seeks to find a point* x *which minimizes*

$$E(WD(x)) = \sum_{r=1}^{R} P_r WD_r(x), \tag{3}$$

where $WD(x)$ *is the random variable denotes the weighted traveling time from all nodes to the point* x *having the values* $WD_r(x),\ r = 1, \ldots, R$ *computed by expression* (2)*, and* P_r *is the probability of the* r*th state in set* G_{state}*. The function* $E(WD(x))$ *is known as the expected 1-median function.*

In continue, an independent discrete random weight W_h is allocated for each node $v_h \in N$. The discrete values of W_h are denoted by $w_h[k],\ k = 1, \ldots, K_h$, where K_h is the number of different situations for the weight of node $v_h \in N$ and it is assumed that $w_h[k] \leq w_h[k+1]$ for $k = 1, \ldots, K_h - 1$. The probability corresponding to $w_h[k]$ is denoted by $P'_h[k],\ k = 1, \ldots, K_h$. Let T be a preselected admissible threshold. The proposed objective function is maximizing the probability that the expected 1-median function (3) does not exceed the given threshold T. Here T is an upper level, which is given by the decision maker.

Definition 2. *Let the weights of nodes and the lengths of links be independent discrete random variables with known probabilities. A point* $x \in G$ *is an expected* 1*-median with the maximum probability if it maximizes the following probabilistic objective function:*

$$P'(\sum_{r=1}^{R} P_r WD_r(x) \leq T) \tag{4}$$

and consequently

$$P'(\sum_{r=1}^{R} P_r \sum_{h=1}^{n} W_h d_r(v_h, x) \leq T),$$

where $P'(.)$ *is the probability corresponding to the random weight vector of nodes.*

The expected 1-median with maximum probability is NP-hard on general networks; see [2].

2.1 Calculating a lower bound

In this section, we provide a lower bound approximation for the objective function (4) at each point x of G. The lower bound can be calculated as an expected function, which would be readily calculated in comparison with the probabilistic objective function (4); see Lemma 1.

Lemma 1. *If* $\mu_h = \sum_{k=1}^{K_h} w_h[k]P'_h[k]$, $h = 1, \ldots, n$ *is the expected value of* W_h, *then the probability* $P'(\sum_{r=1}^{R} P_r W D_r(x) < T)$ *is greater than or equal to* $1 - \sum_{r=1}^{R} P_r \sum_{h=1}^{n} \frac{\mu_h}{T} d_r(v_h, x)$.

Proof. Based on Markov's inequality, if X is a nonnegative random variable and $a \geq 0$, then $P(X \geq a) \leq \dfrac{E(X)}{a}$ or in other words $P(X < a) \geq 1 - \dfrac{E(X)}{a}$, where $E(X)$ is the expected value of X. Therefore, Since $W_1, \ldots, W_n$ are independent random variables, the following inequalities hold:

$$\begin{aligned} P'(\sum_{r=1}^{R} P_r \sum_{h=1}^{n} W_h d_r(v_h, x) < T) &\geq 1 - E(\sum_{r=1}^{R} P_r \sum_{h=1}^{n} \frac{W_h}{T} d_r(v_h, x)) \\ &= 1 - \sum_{r=1}^{R} P_r \sum_{h=1}^{n} \frac{E(W_h)}{T} d_r(v_h, x) \\ &= 1 - \sum_{r=1}^{R} P_r \sum_{h=1}^{n} \frac{\mu_h}{T} d_r(v_h, x), \end{aligned}$$

which completes the proof. □

Lemma 1 determines a lower bound approximation for the function (4), when a given point is available as the solution for the expected 1-median with maximum probability problem. This helps to correct the available value of T and estimates a more reasonable amount to the preselected threshold. To determine the solution of the expected 1-median with maximum probability, we define the *antipode points* which are originally introduced by Frank, [6]. Using the antipode points, we bring forward the *state-based primary regions* in links, which are useful to estimate the distances; see [2].

Definition 3. *For each node $v_h \in N$ and each state $r, r = 1, \ldots, R$, the point $z^{ij}_{r,h}$ on the link e_{ij} is called an antipode associated with the node v_h in state r, if the following equation holds:*

$$z^{ij}_{r,h} sp^r_{ij} + d_r(v_h, v_i) = (l_{ij} - z^{ij}_{r,h}) sp^r_{ij} + d_r(v_h, v_j).$$

An antipode $z^{ij}_{r,h}$ on the link e_{ij} provides two regions $[v_i, z^{ij}_{r,h}]$ and $[z^{ij}_{r,h}, v_j]$, where the distances between the points on the link e_{ij} in each of these regions and the node v_h in state r can be distinctly computed. Considering all antipode points and gathering them up together yield some basic intervals on e_{ij}, which we call them the state-based primary regions; see Definition 4.

Definition 4. *The set of all points between two consecutive antipodes $z^{ij}_{r',l}$ and $z^{ij}_{r'',k}$ in the link e_{ij} associated with the node $v_l \in N$ in the state r' and node $v_k \in N$ in the state r'', $(r', r'' \in \{1, \ldots, R\})$ is called a state-based primary region $[z^{ij}_{r',l}, z^{ij}_{r'',k}]$.*

Recall that the state-based primary regions are special with a remarkable property that the distances from any node v_h in each state r to all points $x \in [z^{ij}_{r',l}, z^{ij}_{r'',k}]$ are computed either by $x sp^r_{ij} + d_r(v_h, v_i)$ or $(l_{ij} - x) sp^r_{ij} + d_r(v_h, v_j)$ and it remains unchanged throughout this region. By using this property, the 1-median objective function in the state r, that is, $WD_r(x) = \sum_{h=1}^{n} w_h d_r(v_h, x)$, in each state-based primary region can be computed.

Next, a reformulation of the expected 1-median problem with maximum probability is given. Since the weights of nodes are discrete random variables, there are different situations in network. Let the set of all situations for the vector of weights be denoted by $W_{state} = \{W^s | W^s = (w^s_1, \ldots, w^s_n), s = 1, \ldots, S\}$, where $S = \prod_{h=1}^{n} K_h$ and W^s denotes the vector of weights in situation s. Therefore, the weight of node v_h, $h = 1, \ldots, n$, in situation s, $s = 1, \ldots, S$ and its probability are denoted by w^s_h and $P^{'s}_h$, respectively. For all $W^s \in W_{state}$ and $x \in G$, the following characterization function is provided to calculate the objective function (4):

$$Y_{W^s}(x) = \begin{cases} 1, & \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w^s_h d_r(v_h, x) \leq T, \\ 0 & otherwise. \end{cases} \tag{5}$$

By applying this characterization function, the objective function (4) would be rewritten as follows:

$$\max_{x \in G} f(x) = \sum_{s=1}^{S} Y_{W^s}(x) P'_{W^s}, \tag{6}$$

where $P'_{W^s} = \prod_{v_h \in N} P^{'s}_h$; see [2].

3 The solution approach

In this section, two lemmas are provided to determine the optimal solution of the expected 1-median problem with maximum probability when the threshold T is selected from some regular intervals. Furthermore, when T is selected out of these regular intervals, an algorithm is represented to find the optimal solution; see [2]. The 1-anti-median and 1-median problems are used in Lemmas 2 and 3 to indicate some regular intervals for threshold T, where all points in the network G are optimal with probabilistic objective function equal to 1 and zero, respectively. In 1-anti-median problem, the purpose is to find the location of a facility in the network, where the total weighted distances from all nodes to the facility is maximized. The point x is 1-anti-median, if it maximizes $\sum_{h=1}^{n} w_h d(v_h, x)$, where the weights and distances are fixed values. For more details and the solution approaches, see [14].

Let x_r be the optimal solution to the following 1-deterministic-anti-median problem:

$$\max_{x \in G_r} \sum_{h=1}^{n} w_h[K_h] d_r(v_h, x),$$

for $r = 1, \ldots, R$, where $w_h[K_h]$ is the largest probabilistic weight corresponding to node v_h. Also let x'_r be the optimal solution to the following 1-deterministic-median problem:

$$\min_{x \in G_r} \sum_{h=1}^{n} w_h[1] d_r(v_h, x),$$

for $r = 1, \ldots, R$, where $w_h[1]$ is the smallest probabilistic weight corresponding to node v_h. Lemma 2 provides an upper level value to the threshold T, while Lemma 3 provides a lower level value to it. If T is selected greater than the estimated upper level or smaller than the estimated lower level, then the maximum probability expected 1-median solution would be readily determined.

Lemma 2. *If* $A = arg \max_{r=1,\ldots,R} \sum_{h=1}^{n} w_h[K_h] d_r(v_h, x_r)$ *while* x_A *is the corresponding* 1*-deterministic-anti-median solution to the following problem:*

$$\max_{x \in G_A} \sum_{h=1}^{n} w_h[K_h] d_A(v_h, x)$$

and T *is given in such a way to satisfy the constraint*

$$\sum_{h=1}^{n} w_h[K_h] d_A(v_h, x_A) \leq T,$$

then each point $x \in G$ *is the expected* 1*-median solution with the maximum probability equal to* 1*.*

Proof. The weights of the nodes $v_h \in N$ are assumed to be arranged in ascending order; therefore, the following inequalities hold:

$$w_h[1] < w_h[2] < \cdots < w_h[K_h].$$

Considering the above inequalities together with the definition of x_A yields the following inequalities for an arbitrary x:

$$\begin{aligned}
&\sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[k] d_r(v_h, x) \\
&\quad \le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[K_h] d_r(v_h, x) \le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[K_h] d_r(v_h, x_r) \\
&\quad \le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[K_h] d_A(v_h, x_A) = \sum_{h=1}^{n} w_h[K_h] d_A(v_h, x_A) \le T.
\end{aligned}$$

Hence $Y_{W^s}(x) = 1$ for all $W^s \in W_{state}$ and $x \in G$; therefore, the objective function value at every point $x \in G$ is equal to 1, that is, all $x \in G$ are optimal. □

Lemma 3. *Let* $B = arg\min_{r=1,\ldots,R} \sum_{h=1}^{n} w_h[1] d_r(v_h, x'_r)$, *where* x'_B *is the* 1*-deterministic-median solution to the following problem:*

$$\min_{x \in G_B} \sum_{h=1}^{n} w_h[1] d_B(v_h, x).$$

If $T < \sum_{h=1}^{n} w_h[1] d_B(v_h, x'_B)$, *then all points* $x \in G$ *are optimal to the expected* 1*-median problem with maximum probability equal to zero.*

Proof. Considering the definitions of x'_B, the following inequalities hold:

$$\begin{aligned}
T &< \sum_{h=1}^{n} w_h[1] d_B(v_h, x'_B) = \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[1] d_B(v_h, x'_B) \\
&\le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[1] d_r(v_h, x'_r) \le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[1] d_r(v_h, x) \\
&\le \sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h[k] d_r(v_h, x).
\end{aligned}$$

The foregoing inequalities yield $Y_{W^s}(x) = 0$ for all $x \in G$ and $W^s \in W_{state}$; therefore, all points $x \in G$ are optimal with probability equal to zero. □

Lemma 4 provides a sufficient condition to ignore some links, which along them, the characterization function (5) is equal to zero.

Lemma 4. *Let v_i and v_j be two adjacent nodes, where $Y_{W^s}(v_i) = Y_{W^s}(v_j) = 0$, for some $W^s \in W_{state}$; then $Y_{W^s}(x)$ is also equal to zero for all $x \in e_{ij}$.*

Proof. By the assumption, $Y_{W^s}(v_i) = Y_{W^s}(v_j) = 0$; therefore, the following inequalities hold:

$$\sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h^s d_r(v_h, v_i) > T,$$

$$\sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h^s d_r(v_h, v_j) > T.$$

Furthermore, each function $d_r(v_h, x)$ is concave in each link e_{ij}; see Figure 1. Therefore, the function:

$$\sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h^s d_r(v_h, x)$$

is concave on the link e_{ij}; see [14]. This yields the inequality

$$\sum_{r=1}^{R} P_r \sum_{h=1}^{n} w_h^s d_r(v_h, x) > T$$

for all $x \in e_{ij}$ and consequently $Y_{W^s}(x) = 0$, which completes the proof. □

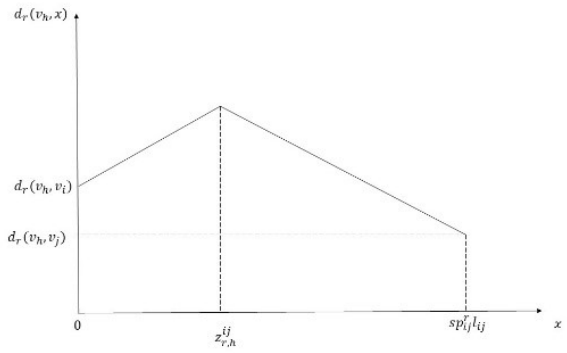

Figure 1: The figure of function $d_r(v_h, x)$ on the link e_{ij}.

Let the threshold T be selected between the upper and lower bound values provided by Lemmas 2 and 3,

$$\sum_{h=1}^{n} w_h[1] d_B(v_h, x'_B) \leq T < \sum_{h=1}^{n} w_h[K_h] d_A(v_h, x_A).$$

In this case, the solution of the expected 1-median problem with maximum probability would be determined by defining some new points x in network links, where the characterization function $Y_{W^s}(x)$ for the weight vectors $W^s \in W_{state}$ would change. First some special points in stated-based primary regions, called jump-points, are specified as introduced in [2]. Then it is shown the objective function (6) would be changed only at the jump-points. The shortest path from the points in a state-based primary region on the link e_{ij} to each node v_h would be obtained through the node v_i or v_j without switching to the other node. Hence, the value of objective function (6) can be readily computed in all jump-points in each state-based primary region. Therefore, the optimal solution in each primary region would be determined. By comparing all solutions in the state-based primary regions together, the local optimal solution in each link would be obtained.

Let $[z^{ij}_{r',l}, z^{ij}_{r'',k}]$ be a state-based primary region associated with the nodes l and k in the link e_{ij} in states r' and r'', respectively. Then the set of nodes N and the set of states corresponding to the traveling times of links, G_{state}, can be partitioned into two sets $L^-_{r'}$ and $K^+_{r''}$ as follows:

$$L^-_{r'} = \{(h,r) | z^{ij}_{r,h} \le z^{ij}_{r',l}\}, \qquad K^+_{r''} = \{(h,r) | z^{ij}_{r,h} \ge z^{ij}_{r'',k}\}.$$

Indeed, the set $L^-_{r'}$ includes the pair indices (h,r), where h corresponds to the node $v_h \in N$ and r corresponds to $G_r \in G_{state}$, so that the position of antipode $z^{ij}_{r,h}$ in the link e_{ij} is before the position of the antipode $z^{ij}_{r',l}$ associated with the node v_l in the state r'. The set $K^+_{r''}$ is defined similarly. Therefore, for $(h,r) \in L^-_{r'}$, the distance between any point $x \in [z^{ij}_{r',l}, z^{ij}_{r'',k}]$ and the node v_h in the state r is determined through the node v_j, that is, $d_r(v_h, x) = sp^r_{ij}(l_{ij} - x) + d_r(v_h, v_j)$, where sp^r_{ij} is the inverse of the traveling speed on the link e_{ij} in the state r. The corresponding value for $d_r(v_h, x)$, where $(h,r) \in K^+_{r''}$, would be determined through node v_i in a similar manner.

As an example, consider the small network shown in Figure 2 with three state-based networks G_1, G_2, and G_3 with $sp^r_{12} = 0.5$, $sp^r_{14} = 0.3$, $sp^r_{23} = 0.3$, $sp^r_{24} = 0.5$ for $r = 1,2,3$ and $sp^1_{13} = sp^1_{34} = 0.2$, $sp^2_{13} = sp^2_{34} = 0.3$ and $sp^3_{13} = sp^3_{34} = 0.4$. Consider the link e_{12}, then the set of antipodes associated with the node 3 is $\{z^{12}_{1,3}, z^{12}_{2,3}, z^{12}_{3,3}\} = \{3.7, 2.7, 2.7\}$ and with the node 4 is $\{z^{12}_{1,4}, z^{12}_{2,4}, z^{12}_{3,4}\} = \{3.4, 4.1, 4.8\}$; see Figure 2. For the primary region $[z^{12}_{2,3}, z^{12}_{1,4}] = [2.7, 3.4]$, we have $L^-_2 = \{(2,1),(2,2),(2,3),(3,2),(3,3)\}$ and $K^+_1 = \{(1,1),(1,2),(1,3),(3,1)\ (4,1),(4,2),(4,3)\}$. Note that for all states r, the antipodes corresponding to the nodes 1 and 2, are the points $l_{12} = 6$ and 0, respectively. Now consider one of the weight vectors $W^s \in W_{state}$ with components $w^s_h, h = 1, \ldots, n$. So for any point $x \in [z^{ij}_{r',l}, z^{ij}_{r'',k}]$, the following relationships hold [2]:

$$\sum_r P_r \sum_h w^s_h d_r(v_h, x) \le T$$

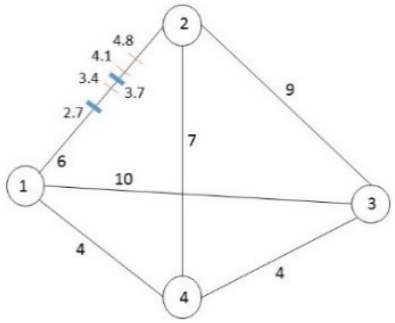

Figure 2: The state-based primary regions in the link e_{12} in a small-sized network.

$$\Rightarrow \sum_{(r,h)\in L^-_{r'}} P_r w^s_h d_r(v_h, x) + \sum_{(r,h)\in K^+_{r''}} P_r w^s_h d_r(v_h, x) \le T$$

or

$$\begin{aligned}
&\sum_{(r,h)\in L^-_{r'}} P_r w^s_h (d_r(v_h, v_j) + sp^r_{ij}(l_{ij} - x)) + \sum_{(r,h)\in K^+_{r''}} P_r w^s_h (d_r(v_h, v_i) + sp^r_{ij} x) \le T \\
&\quad \Rightarrow \sum_{(r,h)\in L^-_{r'}} P_r w^s_h (d_r(v_h, v_j) + sp^r_{ij} l_{ij}) + \sum_{(r,h)\in K^+_{r''}} P_r w^s_h d_r(v_h, v_i) \\
&\quad + \Big(\sum_{(h,r)\in K^+_{r''}} P_r w^s_h sp^r_{ij} - \sum_{(h,r)\in L^-_{r'}} P_r w^s_h sp^r_{ij} \Big) x \le T.
\end{aligned} \tag{7}$$

Definition 5. *If inequality (7) is binding at a point x^{W^s}, then x^{W^s} is called a jump-point with respect to weight vector $W^s \in W_{state}$.*

If we set

$$\sum_{(r,h)\in L^-_{r'}} P_r w^s_h (d_r(v_h, v_j) + sp^r_{ij} l_{ij}) = A^{W^s}_{r',L}, \qquad \sum_{(r,h)\in K^+_{r''}} P_r w^s_h d_r(v_h, v_i) = B^{W^s}_{r'',K}$$

and

$$\sum_{(h,r)\in K^+_{r''}} P_r w^s_h sp^r_{ij} - \sum_{(h,r)\in L^-_{r'}} P_r w^s_h sp^r_{ij} = C^{W^s}_{(r',L),(r'',K)},$$

then

$$x^{W^s} = \frac{T - A^{W^s}_{r',L} - B^{W^s}_{r'',K}}{C^{W^s}_{(r',L),(r'',K)}}.$$

Let J be the set of all jump-points in state-based primary region $[z^{ij}_{r',l}, z^{ij}_{r'',k}]$ unified with two boundary points $\{z^{ij}_{r',l}, z^{ij}_{r'',k}\}$.

Lemma 5. *The expected 1-median objective function would change only in the points belonging to the set J.*

Proof. The objective function value would change in some point x, if the value of $Y_{W^s}(x)$ is changes from 0 to 1 or vice versa for some vectors W^s. This guarantees that these points belong to the set J. □

If a local optimal solution in a state-based primary region in some links is given, Lemma 6 provides some optimal intervals in this region. The coefficient of variable x in inequality (7) is a criterion, which determines the optimal intervals; for more details see [2].

Lemma 6. *Let $x^{W^s} \in [z^{ij}_{r',l}, z^{ij}_{r'',k}]$ be the local optimal jump-point while $y_1^{W^1}$ and $y_2^{W^2}$ are the left and right consecutive jump-points to x^{W^s}, respectively. Then the optimal interval in $[z^{ij}_{r',l}, z^{ij}_{r'',k}]$ is determined as follows:*

1. *If $C^{W^s}_{(r',L),(r'',K)} \geq 0$ and $C^{W^1}_{(r',L),(r'',K)} \leq 0$, then the interval $[y_1^{W^1}, x^{W^s}]$ is optimal.*
2. *If $C^{W^s}_{(r',L),(r'',K)} \leq 0$ and $C^{W^2}_{(r',L),(r'',K)} \geq 0$, then the interval $[x^{W^s}, y_2^{W^2}]$ is optimal.*
3. *If $C^{W^s}_{(r',L),(r'',K)} \geq 0$ and x^{W^s} is the first jump-point after $z^{ij}_{r',L}$, then the interval $(z^{ij}_{r',L}, x^{W^s}]$ is optimal, where the starting point of the interval can be distinctly verified.*
4. *If $C^{W^s}_{(r',L),(r'',K)} \leq 0$ and x^{W^s} is the last jump-point before $z^{ij}_{r'',K}$, then the interval $[x^{W^s}, z^{ij}_{r'',K})$ is optimal, where the end point of the interval can be distinctly verified.*

Proof. The coefficient of variable x in inequality (7) is $C^{W^s}_{(r',L),(r'',K)}$. If $C^{W^s}_{(r',L),(r'',K)} \geq 0$ and $C^{W^1}_{(r',L),(r'',K)} \leq 0$, then $Y_{W^s}(x) = 1$ for all $x \leq x^{W^s}$ and $Y_{W^1}(x) = 1$ for all $x \geq y_1^{W^1}$. For other vectors W', which their respective jump-points x lie outside the interval $[y_1^{W^1}, x^{W^s}]$, the values of $Y_{W'}(x)$ remain unchanged in this interval. Note that if $C^{W^1}_{(r',L),(r'',K)} > 0$ and $C^{W^s}_{(r',L),(r'',K)} > 0$, then $Y_{W^1}(x^{W^s}) = 0$. Therefore, $f(x^{W^s}) < f(y_1^{W^1})$, which contradicts the optimality of point x^{W^s}. The other cases can be proved similarly. Note that the rest of states for $C^{W^s}_{(r',L),(r'',K)}$, $C^{W^1}_{(r',L),(r'',K)}$, and $C^{W^2}_{(r',L),(r'',K)}$ are impossible, since they contradict the optimality of point x^{W^s}. □

3.1 The Algorithm: finding a global optimal solution

In this section, Algorithm 1 is represented to describe that how a local optimal solution can be obtained on an arbitrary link and consequently a global optimal solution is determined. Recall that if two arbitrary adjacent nodes, say v_i and v_j, have the same zero objective function values, then all points on the link e_{ij} have also objective values equal to zero. Therefore, these edges should be ignored in the proposed algorithm. In other side, as described in Lemmas 2 and 3, the solution can be determined directly for spacial values of T. The idea of Algorithm 1 is to specify the state-based primary regions on all links and obtain the local optimal solution in all regions. Hence, G_{state} the set of all state-based situations for the network G, , and the set of all weight situations, W_{state}, are determined. Then the primary regions are located by computing the antipodes while the set of jump-points are determined in each primary region, as well. The corresponding value of the objective function (6) is estimated for all jump-points x by evaluating $Y_{W^s}(x)$ for vectors $W^s \in W_{state}$. By comparing these solutions, the optimal solution in each primary region and consequently on each link would be determined. Finally, the global optimal solution in the network would be determined by comparing the local optimal solutions on all links, while the optimal regions are fully specified by Lemma 6.

In the represented algorithm, the maximum number of state-based primary regions in each link is $(n-2)R+1$ and the maximum number of jump-points in each link is $S((n-2)R+1)$, where S and R are the number of states for the weights of nodes and the traveling times of edges, respectively. To specify the solution, the values of $Y_{W^s},\ s=1,\ldots,S$ for each jump-point should be estimated. Therefore, if any link is not ignored by Lemma 4, in the worst case, the time of algorithm in network would be $mS^2((n-2)R+1)$. This time would be so long, particularly for large-sized networks. In section 5, we propose a method to solve the problem in large-sized networks without using the Algorithm 1.

3.2 Numerical example

In the following, a numerical example is provided. The optimal solutions are estimated for different values of threshold T by performing the steps of Algorithm 1 in MATLAB R2014a, and the sensitivity analysis for different values of weights is proposed. Using Lemmas 2 and 3, some regular intervals for threshold T are introduced, where the optimal solutions along them would be readily obtained.

Example 1. In this example, we study an squared network given in Figure 3 with 25 nodes and 40 links. The lengths of the links have been written next

to the links while the weights of the nodes with their relevant probabilities are shown in Table 1. The traveling speeds in different states for each link e_{ij} with their relevant probabilities are given in Table 2.

Algorithm 1 Determining the expected 1-median with maximum probability in the link e_{ij}

1: Determine the solution of the deterministic problems $\min_{r=1,\ldots,R} \min_{x' \in G_r} \sum_{h=1}^{n} w_h[1] d_r(v_h, x')$ and $\max_{r=1,\ldots,R} \max_{x \in G_r} \sum_{h=1}^{n} w_h[K_h] d_r(v_h, x)$, set their optimal values as L and U, respectively. If $T < L$ or $T \geq U$, then the optimal solutions are determined by Lemmas 3 and 2, respectively. Otherwise go to step (2).
2: Consider the set of all state-based situations for the network G as $G_{state} = \{G_r,\ r = 1, \ldots, R\}$ in which $R = \prod_{e_{ij} \in E} U_{ij}$ with relevant probabilities $P_r = \prod_{e_{ij} \in E} P_{ij}^r$, where P_{ij}^r is the probability that the link e_{ij} has the traveling time $d_r(v_i, v_j) = sp_{ij}^r l_{ij}$.
3: Arrange the demand weight situations in the set $W_{state} = \{W^s |\ W^s = (w_1^s, \ldots, w_n^s), s = 1, \ldots, S\}$ with relevant probabilities $P'_{W^s} = \prod_{v_h \in N} P_h^{'s}$ for $s = 1, \ldots, S$.
4: Determine and sort the set of $Z^{ij} = \{z_{r,h}^{ij} | h = 1, \ldots, n,\ h \neq i, j,\quad r = 1, \ldots, R\} \cup \{0, l_{ij}\}$ on the link $e_{ij} \in E$ in ascending order. Rename the members of Z^{ij} as $z_k, k = 1, \ldots, |Z^{ij}|$.
5: Set $W'_{state} = W_{state} \backslash \{W^s | Y_{W^s}(v_i) = Y_{W^s}(v_j) = 0\}$ and $k = 1$.
6: Determine the state-based primary regions $[z_k, z_{k+1}]$, where $z_k, z_{k+1} \in Z^{ij}$, and the jump-point set $J_k = \{x_d^k | x_d^k = x^{W^s} \in [z_k, z_{k+1}]$ *for some vector* $W^s \in W'_{state},\ d = 1, \ldots, D\}$, where $D = |J_k|$. Set $q = 1$, $d = 1$ and $f(x_d^k) = 0$.
7: If $Y_{W^q}(x_d^k) = 1$, where W^q is the qth member of W'_{state}, set $f(x_d^k) = f(x_d^k) + P'_{W^q}$, else go to step (8).
8: Set $q = q + 1$. If $q \leq |W'_{state}|$, then go to step (7), else set $d = d + 1$ and go to step (9).
9: If $d \leq D$, set $f(x_d^k) = 0$, $q = 1$, and go to step (7), else go to step (10).
10: Set $x_{opt}^k = \arg\max_d f(x_d^k)$.
11: Set $k = k + 1$, if $k \leq |Z^{ij}| - 1$, go to step (6), else go to step (12).
12: Set $x_{e_{ij}} = argmax_{k=1,\ldots,|Z^{ij}|} f(x_{opt}^k)$.

The upper and lower bound values provided by Lemmas 2 and 3 are 5716.435 and 310.2833, respectively. Therefore, if $T \geq 5716.435$ or $T < 310.2833$, then all points of network G are optimal with probabilities equal to one and zero, respectively. In addition, using the proposed algorithm, it can be numerically verified that, for values of $T \leq 1050$ and $T \geq 1900$, the optimal objective function value is equal to zero and one, respectively.

To verify Algorithm 1 in details, we consider the threshold $T = 1100$. By applying Lemma 4, only the links $e_{7,12}, e_{8,13}, e_{11,12}, e_{12,13}, e_{12,17}, e_{13,14}$ and

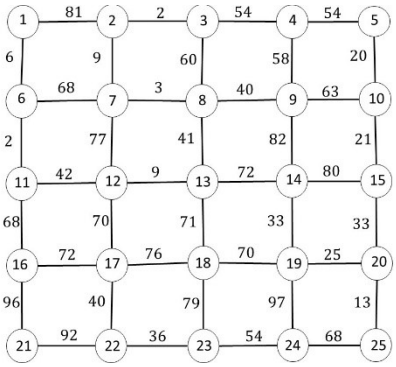

Figure 3: A squared network with 25 nodes.

Table 1: The weights of nodes in different states

nodes	1	2	3	4	5	6	7	8	9	10	11	12	13
state 1	75	35	33	59	47	57	56	72	23	86	78	62	53
state 2	69	68	18	88	74	54	28	5	90	90	40	50	96
state 3	44	36	4	67	9	50	57	25	72	41	32	55	4
state 4	47	94	57	28	80	43	82	27	7	38	50	78	39
state 5	75	28	33	76	43	33	53	72	60	44	41	15	68
nodes	14	15	16	17	18	19	20	21	22	23	24	25	Probability
state 1	82	18	51	75	48	27	72	8	32	46	74	4	0.2
state 2	77	18	10	54	6	9	25	88	51	1	66	73	0.25
state 3	57	6	19	16	69	23	41	27	56	51	63	62	0.15
state 4	101	32	38	97	107	46	90	68	7	28	24	87	0.1
state 5	47	17	36	9	57	29	58	44	48	66	72	48	0.3

Table 2: The different traveling speeds

State	1	2	3	4	5	6
link traveling speed	$100+i+j$	$70+i+j$	$50+i+j$	$40+i+j$	$80+i+j$	$90+i+j$
probability	0.15	0.2	0.1	0.25	0.23	0.07

$e_{13,18}$ have to be searched for state $s = 3$ and the other states $s = 1, 2, 4, 5$ as well as the other links would be ignored. We follow the steps of algorithm for links $e_{12,13}$ and $e_{13,14}$. The antipodes of the link $e_{12,13}$ are sorted ascending according to their distance from vertex 12, as follows:

$$Z^{12,13} = \{0,\ 5.5061,\ 5.6330,\ 5.8108,\ 5.8753,\ 5.9289,\ 5.9742,\ 9\}.$$

The jump-points obtained by equality (7) are all outside the primary regions and the only jump-points of the link $e_{12,13}$ would be its antipodes. Computing the values of $\sum_{r=1}^{6} P_r \sum_{h=1}^{25} w_h^3 d_r(v_h, z_k)$ for $z_k \in Z^{12,13},\ k = 1, \ldots, 8$ results in $Y_{W^3}(z_k) = 1$. So all points in the link $e_{12,13}$ are optimal with the objective function value equals to 0.15. The other evaluated link is $e_{13,14}$, where its antipode set is

$$\begin{aligned} Z^{13,14} = \{&0,\ 1.5085,\ 1.6971,\ 1.9164,\ 2.1749,\ 2.8602,\ 3.3285,\ 3.8673,\ 5.3977,\\ &7.4310,\ 8.1409, 8.7211,\ 9.2043,\ 33.5780,\ 34.0291,\ 34.6129,\\ &34.8126,\ 34.9741,\ 35.1074, 44.9147,\ 45.7261,\ 45.7886,\ 46.9351,\\ &46.9441,\ 47.4125,\ 47.8139,\ 48.0984,\ 48.1616,\ 48.2600,\ 48.5373,\\ &48.6747,\ 48.9968,\ 49.1737,\ 49.2538,\ 49.4117,\\ &49.6128,\ 49.7849,\ 69.6898,\ 71.4483,\ 72\} \end{aligned}$$

Again, the jump-points obtained by equality (7) are outside the primary regions and the set of jump-points includes just the antipodes. For antipodes

$$\{0,\ 1.5085,\ 1.6971,\ 1.9164,\ 2.1749,\ 2.8602,\ 3.3285,\ 3.8673,\ 5.3977\} \subset Z^{13,14}$$

the value of Y_{W^3} is equal to 1. Therefore, the optimal interval obtained by Lemma 6 is [0 , 5.3977] with optimal probability 0.15. Using Algorithm 1, the solutions for different values of T with their objective (Obj) function values are determined and inserted in Table 3. The obtained optimal intervals (Int) in corresponding links with running times (t) are also reported. Moreover, the variations of the objective function (4) with respect to different values of T are released in Figure 4.

Next, we develop a sensitivity analysis to the solutions by certain perturbations in the weights of nodes and traveling speeds of links. The weights of all nodes are assumed to be equal to 75 in all 5 states with relevant probabilities $\{0.55,\ 0.1,\ 0.1,\ 0.1,\ 0.15\}$ while the lengths of all links are considered equal to 50. In addition, 6 states are considered for links' traveling speeds that are assumed to be the same for all links equal to $\{100,\ 70,\ 50,\ 40,\ 80,\ 90\}$ in states $r = 1, \ldots, 6$ with relevant probabilities given in Table 2. The weight of node 1 is multiplied by 100 in the first state, that is, $w_1^1 = 7500$. This alteration results in the solution of maximum probability expected 1-median problem for $T = 3700$ being attained in node 13 with probability 0.45. Moreover, $Y_{W^s}(x^*) = 1$ for the states $s = 2, 3, 4, 5$ and $Y_{W^1}(x^*) = 0$, where x^*

denotes the optimal solution, also the interval $[0,\ 6.8090]$ in the link $e_{13,18}$ is optimal.

By increasing the value of T from $T = 3700$ to $T = 6000$, the optimal value of the objective function remains unchanged while the optimal solutions would be spread in the network. For $T = 6000$, all nodes except 1, 5, 21 and 25 lie in the optimal solution set. If $T = 6100$, only the node 1 is optimal with objective function value 1. In this case, the value of T is larger than the expected sum of weighted distances from all nodes to node 1 in all states $s = 1, 2, 3, 4, 5$. Notice that if the median is in node 1, the expected sum of weighted distances from all nodes to node 1 is the same in all states $s = 1, \ldots, 5$.

Next, we multiply the weight of node 25 in state 5 by 100 while the other information remain unchanged. For $T = 3700, \ldots, 6000$, the optimal value of the objective function is 0.3. The solution for $T = 3700$ is node 13, by increasing the value of T to $T = 6000$ the value of optimal probability does not change but the number of optimal points is increased. For $T = 6000$ all nodes expect 1, 5, 21 and 25 lie in the optimal points' set. For $T = 6100$ the optimal probability is changed to 0.85 whereas the optimal solution set is reduced to node 1.

If the probability of the first and fifth states are changed together, that is, $P'_{W^1} = 0.15$ and $P'_{W^5} = 0.55$, then the solution would be in the last node with optimal probability 0.85. Also, if the first and the fifth states of nodes' weights have just the same probability equal to 0.35, then for $T = 6100$, both nodes 1 and 25 would be optimal with optimal probability 0.65.

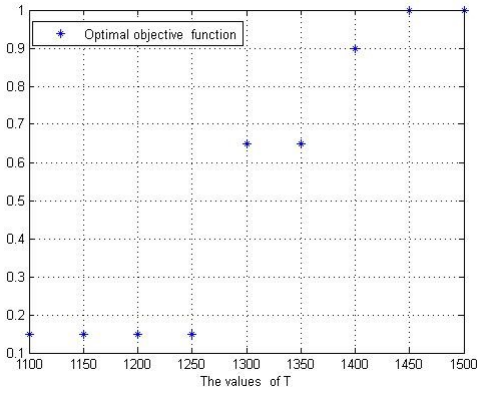

Figure 4: Variations of objective function (4) for different values of T

Table 3: The solutions for different values of T

$T = 1100$	$link$	$e_{12,13}$	$e_{13,14}$							$t = 0.51$
$Obj = 0.15$	Int	$[0\ ,\ 9]$	$[0\ ,\ 5.3977]$							
$T = 1150$	$link$	$e_{8,13}$	$e_{11,12}$	$e_{12,13}$	$e_{12,17}$	$e_{13,14}$				$t = 0.49$
$Obj = 0.15$	Int	$[28.8133\ ,\ 41]$	$[34.9193\ ,\ 42]$	$[0\ ,\ 9]$	$[0\ ,\ 11.0125]$	$[0\ ,\ 9.2043]$				
$T = 1200$	$link$	$e_{8,13}$	$e_{11,12}$	$e_{12,13}$	$e_{12,17}$	$e_{13,14}$	$e_{13,18}$			$t = 0.66$
$Obj = 0.15$	Int	$[8.5634\ ,\ 41]$	$[32.5787\ ,\ 42]$	$[0\ ,\ 9]$	$[0\ ,\ 13.3575]$	$[0\ ,\ 9.2043]$	$[0\ ,\ 71]$			
$T = 1250$	$link$	$e_{2,7}$	$e_{6,7}$	$e_{7,8}$	$e_{7,12}$	$e_{8,9}$	$e_{8,13}$	$e_{11,12}$	$e_{12,13}$	$t = 1.51$
$Obj = 0.15$	Int	$[6.8588\ ,\ 9]$	$[65.7371\ ,\ 68]$	$[0\ ,\ 3]$	$[64.0216\ 77]$	$[0\ ,\ 4.1260]$	$[0\ ,\ 41]$	$[0\ ,\ 42]$	$[0\ ,\ 9]$	
$T = 1250$	$link$	$e_{12,17}$	$e_{13,14}$	$e_{13,18}$	$e_{14,19}$	$e_{17,18}$	$e_{18,19}$	$e_{19,20}$	$e_{20,25}$	$t = 1.51$
$Obj = 0.15$	Int	$[0\ ,\ 13.3575]$	$[0\ ,\ 9.2043]$	$[0\ ,\ 30.9657]$	$[0\ ,\ 33]$	$[71.4691\ ,\ 76]$	$[0\ ,\ 11.1206]$	$[0\ ,\ 3.3058]$	0	
			$[71.4483\ ,\ 72]$				$[66,8237\ ,\ 70]$			
$T = 1300$	$link$	$e_{12,13}$	$e_{13,14}$							$t = 2.29$
$Obj = 0.65$	Int	$[7.0710\ ,\ 9]$	$[0\ ,\ 2.1749]$							
$T = 1350$	$link$	$e_{7,12}$	$e_{11,12}$	$e_{12,13}$	$e_{13,14}$					$t = 3.25$
$Obj = 0.65$	Int	$[74.6709\ ,\ 77]$	$[38.7520\ ,\ 42]$	$[0\ ,\ 9]$	$[0\ ,\ 7.4310]$					
$T = 1400$	$link$	$e_{13,14}$								$t = 3.66$
$Obj = 0.9$	Int	0								
$T = 1450$	$link$	$e_{12,13}$	$e_{13,14}$							$t = 4.58$
$Obj = 1$	Int	$[5.5061\ ,\ 9]$	$[0\ ,\ 2.8602]$							
$T = 1500$	$link$	$e_{12,13}$	$e_{13,14}$							$t = 5.66$
$Obj = 1$	Int	$[0\ ,\ 9]$	$[0\ ,\ 9.2043]$							

4 Central limit theorem

In this section, the expected 1-median problem with maximum probability in the large-sized networks is considered. Considering all state-based primary regions, determining the jump-points, and finding the value of objective function in all of them make a large number of computations to be done that practically is useless. Therefore, we state an approximation method, which provides a near optimal solution to the expected 1-median with maximum probability, the approach proposed in [3].

Let the mean and variance of W_h, which is the independent discrete random weight of node $v_h \in N$, be denoted by μ_h and σ_h^2, respectively. Using Lemma 7, the mean and variance of term $\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} W_h d_r(v_h, x)$ are determined. Then by applying the central limit theorem, the problem is reduced to a nonlinear fractional programming problem.

Lemma 7. *Based on the central limit theorem, the term* $\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} W_h d_r(v_h, x)$ *in large-sized networks has the normal distribution with the expected mean value*

$$\mu_{EWD} = \sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} \mu_h d_r(v_h, x)$$

and variance $\sigma^2_{EWD} = \sum_{r=1,\ldots,R} P_r^2 \sum_{h=1,\ldots,n} \sigma_h^2 d_r^2(v_h, x)$.

Therefore, $\dfrac{\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} W_h d_r(v_h, x) - \mu_{EWD}}{\sigma_{EWD}} \sim N(0,1)$, which yields the following:

$$\begin{aligned} &P'\Big(\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} W_h d_r(v_h, x) \le T\Big) \\ &= P'\Big(\frac{\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,N} W_h d_r(v_h, x) - \mu_{EWD}}{\sigma_{EWD}} \le \frac{T - \mu_{EWD}}{\sigma_{EWD}}\Big) \\ &= \phi\Big(\frac{T - \mu_{EWD}}{\sigma_{EWD}}\Big), \end{aligned}$$

where ϕ is the cumulative distribution function of the standard normal distribution. By considering the ascending property of ϕ, the main problem can be transformed into

$$\max_x \frac{T - \sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} \mu_h d_r(v_h, x)}{\sqrt{\sum_{r=1,\ldots,R} P_r^2 \sum_{h=1,\ldots,n} \sigma_h^2 d_r^2(v_h, x)}}. \tag{8}$$

The above nonlinear fractional programming problem can be solved by using the existed approaches such as parametric or direct methods; see [17]. But here we use a method that considers the problem in three cases. Let x^*

be the optimal solution to the problem (8); then three cases can be occurred associated with $T - \mu_{EWD}(x^*)$ that are discussed in details below; see [3].

1. **First case**: $T - \mu_{EWD}(x^*) > 0$. Setting $T - \mu_{EWD}(x) = \frac{1}{\gamma}$, the problem is transformed into the following problem:

$$\min_{x} \sum_{r=1,\ldots,R} P_r^2 \sum_{h=1,\ldots,n} \sigma_h^2 \gamma^2 d_r^2(v_h, x) \tag{9}$$

$$s.t. \quad T\gamma - \sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} \mu_h \gamma d_r(v_h, x) = 1.$$

$$\gamma \geq 0$$

   To solve the above problem, the optimal point x in each state-based primary region should be determined. Let $[z^{ij}_{r',l}, z^{ij}_{r'',k}]$ be a state-based primary region associated with the nodes l and k in states r' and r'' in the link e_{ij}, respectively. Then for all $x \in [z^{ij}_{r',l}, z^{ij}_{r'',k}]$ and an arbitrary state r and node v_h, either $d_r(v_h, x) = d_r(v_h, v_i) + sp^r_{ij}x$ or $d_r(v_h, x) = d_r(v_h, v_j) + sp^r_{ij}(l_{ij} - x)$. Substituting $d_r(v_h, x)$ in problem (9) and setting $\gamma x = x^{'}$, the problem can be rewritten as a second order problem with one linear constraint in each primary region, which can be solved by using such nonlinear convex programming algorithms as Lagrangian dual method or linearization approaches; see [15]. Comparing the objective function values of the optimal solutions in all intervals yields the global optimal solution in each link and consequently in the network.

2. **Second case**: $T - \mu_{EWD}(x^*) = 0$. The second case is considered when the first one is infeasible. In this case, it is sufficient to solve the following problem:

$$\min_{x} \sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} \mu_h d_r(v_h, x). \tag{10}$$

   Using the concavity of the objective function as mentioned in the proof of Lemma 4, this problem has at least a nodal optimal solution. So it is sufficient to search among the nodes of the network. If the optimal value of the objective function (10) equals to T, then the obtained solution is optimal to the fractional programming problem (8) with the optimal value equal to zero. Otherwise, the third case should be considered.

3. **Third case**: $T - \mu_{EWD}(x^*) < 0$. If the first and second cases are infeasible, this case is considered similar to the first one, but the constraint is changed as follows:

$$\sum_{r=1,\ldots,R} P_r \sum_{h=1,\ldots,n} \mu_h \gamma d_r(v_h, x) - T\gamma = 1,$$

and the objective function (9) should be maximized. The local optimal solutions of the resulted problem can also be obtained in each primary region by using such nonlinear programming algorithms as the Lagrangian dual method or linearization approaches; see [15].

Example 2. To illustrate the efficiency of the proposed method based on central limit theorem, a test problem from the address ($http://people.brunel.ac.uk/mastjjb/jeb/info.html$) with 100 nodes and 396 links is considered. Four scenarios with probabilities $\{0.1, 0.25, 0.45, 0.2\}$ for the network links' traveling speeds have been considered. For each link e_{ij} the traveling speed in state $r = 1, 2, 3, 4$ is assumed as $sp_{ij}^r = (r + i + j)^{0.25}$.

We have considered 1000 different situations for vector of weights with discrete uniform distribution and each of components for random weights of nodes are taken from the interval $[0, 850]$. Using the calculated mean and variance for $W_h, h = 1, \ldots, 100$, the proposed method based on central limit theorem is applied for different values of threshold $T = 10^4(1300 + 50i); i = 1, \ldots, 10$. In each case, the resulted nonlinear problems are solved by using the interior-point solver in Matlab R2014a, and the best optimal solutions are determined. Then, the optimality of the solutions is verified by comparing the values of estimated objective function $\phi(\frac{T - \mu_{EWD}(x^*)}{\sigma_{EWD}(x^*)})$ by the central limit theorem with the values obtained from the objective function (6) for the obtained solutions x^*; see Figure 5.

In addition, the variations of the estimated relative error $\frac{|\phi(\frac{T - \mu_{EWD}(x^*)}{\sigma_{EWD}(x^*)}) - f(x^*)|}{f(x^*)}$, where $f(x^*)$ is computed by (6) for the optimal solution x^* and T is selected from the set $\{10^4(1300 + 50i); i = 1, \ldots, 10\}$ is depicted in Figure 6. As it is seen, increasing the value of T results in increasing the objective function optimal value and therefore decreasing the relative error.

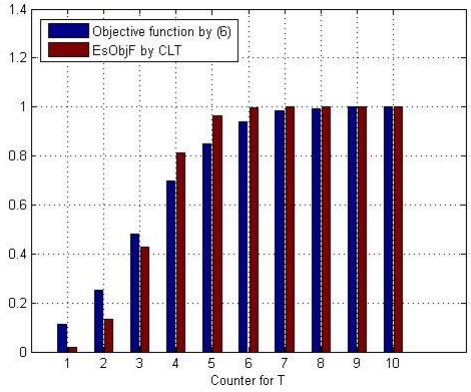

Figure 5: Comparison between $\phi(\frac{T-\mu_{EWD}(x^*)}{\sigma_{EWD}(x^*)})$ and objective function (6).

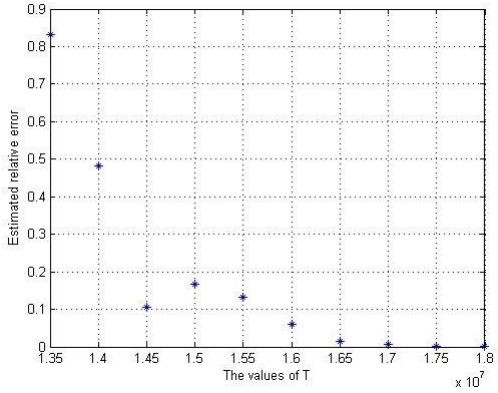

Figure 6: Variations of relative error for different values of T.

5 Summary and conclusions

The median problem is one of the most common problems in location theory specially with nondeterministic parameters. In this paper, the 1-median problem on a network with independent discrete demand weights and traveling times is investigated. The objective function is devoted to maximizing the probability that the expected 1-median function does not exceed a given threshold. First, a precise algorithm to obtain the optimal solution in small-sized networks is presented. Next, by using the central limit theorem, an approach to find the optimal solution in large-sized networks is proposed,

where the original problem is reduced to some linear and nonlinear problems with a linear constraint. The numerical examples are given to illustrate the efficiency of the proposed methods.

References

1. Berman, O., Hajizadeh, I. and Krass, D. *The maximum covering problem with travel time uncertainty*, IIE Transactions, 45 (2013), 81–96.

2. Berman, O. and Wang, J. *Probabilistic location problems with discrete demand weights*, Networks, 44 (2004), 47–57.

3. Berman, O. and Wang, J. *The network p-median problem with discrete probabilistic demand weights*, Computers and Operations Research, 37 (2010), 1455–1463.

4. Bieniek, M. *A note on the facility location problem with stochastic demands*, Omega, 55 (2015), 53–60.

5. Frank, H. *Optimum locations on a graph with probabilistic demands*, Operations Research, 14 (1966), 409–421.

6. Frank, H. *Optimum locations on graphs with correlated normal demands*, Operations Research, 15 (1967), 552–557.

7. Gavish, B. and Sridhar, S. *Computing the 2-median on tree networks in* $O(n \log n)$ *time*, Networks, 26 (1995), 305–317.

8. Gray, M.R. and Johnson D.S. *Computers and intractability: A guide to the theory of Np-completeness* W. H. Freeman. New York, 1979.

9. Goldman, A.J. *Optimal center location in simple networks*, Transportation Science, 5 (1971), 212–221.

10. Hakimi, S.L. *Optimal locations of switching centers and the absolute centers and medians of a graph*, Operations Research, 12 (1964), 450–459.

11. Handler, G.Y. and Mirchandani, P.B. *Location on networks: theory and algorithm*, The MIT Press, Cambridge, 1979.

12. Kariv, O. and Hakimi, S.L. *An algorithmic approach to network location problems. Part II: p-medians*, SIAM Journal on Applied Mathematics, 37 (1979), 539–560.

13. Mirchandani, P.B. and Odoni, A.R. *Location of medians on stochastic networks*, Transportation Science, 13 (1979), 85–97.

14. Minieka, E. *Anticenters and Antimedians of a Network*, Networks, 13 (1983), 359–365.

15. Nocedal, J. and Wright, S.J. *Numerical Optimization*, Springer-Verlag, New York, 2006.

16. Puga, M.S and Tancrez, J.S. *A heuristic algorithm for solving large location-inventory problems with demand uncertainty*, European Journal of Operational Research, 259 (2017), 413–423.

17. Stancu-Minasian, I.M. *Fractional Programming: Theory, Methods and Applications*, Springer, Netherlands, 1997.

18. Tamir, A. *An $O(pn^2)$ algorithm for the p-median and related problems on tree graphs*, Operations Research Letters, 19 (1996), 59–64.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 93–104
DOI:10.22067/ijnao.v9i1.65852

On the finding 2-(k,l)-core of a tree with arbitrary real weight

S.M. Ashkezari and J. Fathali*

Abstract

Let $T = (V, E)$ be a tree with $\mid V \mid = n$. A 2-(k, l)-core of T is two subtrees with at most k leaves and with a diameter of at most l, which the sum of the distances from all vertices to these subtrees is minimized. In this paper, we first investigate the problem of finding 2-(k, l)-core on an unweighted tree and show that there exists a solution that none of (k, l)-cores is a vertex. Also in the case that the sum of the weights of vertices is negative, we show that one of (k, l)-cores is a single vertex. Then an algorithm for finding the 2-(k, l)-core of a tree with the pos/neg weight is presented.

Keywords: Core; Facility location; Median subtree; Semi-obnoxious.

1 Introduction

Location theory is one of the important fields in operations research. In the classical location theory, we want to find the optimal location of a set of single points, as the facilities, on a network. However, in many real applications, the facility to be located is too large to be modeled as a point, so the extensive facilities, which have the form of a path or tree, are considered in many researches.

A tree T is given. A core of a tree is a path of T, so that the sum of the weighted distances from all vertices to this path is minimized. Morgan and Slater [5] and Becker [1] presented linear time algorithms for finding a core

*Corresponding author
Received 6 July 2017; revised 25 July 2018; accepted 1 September 2018
S.M. Ashkezari
Faculty of Mathematical Sciences, Shahrood University of Technology, University Blvd., Shahrood, Iran. email: samane.motevalli@gmail.com

J. Fathali
Faculty of Mathematical Sciences, Shahrood University of Technology, University Blvd., Shahrood, Iran. email: fathali@shahroodut.ac.ir

of a tree, where the weights of all vertices are nonnegative. Zaferanieh and Fathali [16] considered the problem of finding the core of a general network and presented an ant colony and simulated annealing algorithms for solving this problem. Rahbari et al. [10] presented a hybrid genetic and an ant colony algorithm for the path center problem on general networks. Motevalli and Fathali [6] presented a linear time algorithm for finding the core of a tree with interval weights.

A 2-core of a tree is a set of two paths minimizing the sum of the distances of all vertices of the tree from these two paths. Becker and perl [3] considered both cases of disjoint paths and intersecting paths for a tree. Wang [13] presented a linear time algorithm for disjoint paths case.

Some authors deal with the case that there is a constraint on the length of path, so that the length can be at most l (see, e.g., [2,4,8]). Peng et al. [9] considered the problem of finding a subtree containing exactly k leaves, called k-tree core, such that it minimizes the sum of the distances of all vertices to this tree. A linear time algorithm is given for this problem by Shioura and Uno [11]. Wang [12] and Wang et al. [14] considered the parallel algorithms for finding the k-tree core of a tree.

An special case of k-tree core is (k, l)-core, which the diameter of k-tree is at most l. Becker et al. [2] presented an efficient algorithm for finding a (k, l)-core of a tree with time complexity of $O(n^2 logn)$. Their algorithm is started by the tree T and constructs new rooted trees where the maximum length of a path is at most l. Then, for each new tree, apply a greedy-type procedure to find a subtree containing the root with at most k leaves and which minimizes the sum of the distances. The problem of finding the (k, l)-core of a tree also has been considered in [15].

Recently the semi-obnoxious location problems have found an increasing interest. Zaferanieh and Fathali [17] considered the problem of finding a core of a tree with pos/neg weights. The problem of finding a (k, l)-core of a tree with pos/neg weights is considered by Motevalli et al. [7]. They proved that, when the sum of the weights of vertices is negative, the (k, l)-core must be a single vertex. They also presented an algorithm with time complexity of $O(n^2 logn)$ for finding the (k, l)-core of a tree with the pos/neg weight, which is in fact a modification of the one proposed by Becker et al. [2]. Recently Zhou et al. [19] presented polynomial time algorithms for finding 2-core of a tree with pos/neg weights.

In this paper, we consider the problem of finding 2-(k, l)-core of T, where the weights of vertices in T can be any positive or negative numbers. A 2-(k, l)-core of tree is two disjoint subtrees T_1 and T_2 of T that each of them has at most k leaves and diameter of at most l, so that the sum of the distances from all vertices to these subtrees is minimized. Note that if these two subtrees are intersecting, then the problem converts to a (k, l)-core problem that has been solved in [7].

In what follows, we define the problem in Section 2. Then we consider the unweighted tree in Section 3 and show that there exists a solution that

any of 2-(k,l)-core trees is not a vertex. In Section 3, also we show that when the sum of the weights of vertices is negative, one of 2-(k,l)-cores is a single vertex. At last, we propose an algorithm for finding the 2-(k,l)-core of a tree with the pos/neg weight in Section 4.

2 Problem Formulation

Let $T = (V, E)$ be a tree with $|V| = n$. Let $w(v_i)$, for simplicity w_i, be the weight of vertex $v_i \in V$ and let $a(i,j)$ be the length of edge (i,j). Also let $d(v_i, v_j)$ be the length of path from v_i to v_j; then the length of shortest path between subtree T_1 and vertex v is given by

$$d(v, T_1) = min_{u \in T_1}\ d(u, v).$$

For every path P, we show the length of P by $L(P)$, and for every subtree T' of T, we show the weight of tree T' by $w(T')$, that is, $w(T') = \sum_{v_i \in T'} w_i$. The diameter $d_{T'}$ of any tree T' is the maximum distance between two vertices of T'. Any path whose length equals $d_{T'}$, is called a diameter path of T'.

A (k,l)-core of T is a subtree $T_1 = (V_1, E_1)$ of T with at most k leaves and with a diameter of at most l, so that the sum of weighted distances from all vertices to T_1 is minimized, that is,

$$minF(T_1) = \sum_{v_i \in V \setminus V_1} w_i d(v_i, T_1).$$

Also a 2-(k,l)-core of T is a set of two subtrees $T_1 = (V_1, E_1)$ and $T_2 = (V_2, E_2)$ with at most k leaves and with a diameter of at most l, so that the following function is minimized:

$$F(T_1, T_2) = \sum_{v_i \in V \setminus (V_1 \cup V_2)} w_i d(v_i, T_1, T_2),$$

where $d(v_i, T_1, T_2)$ is the minimum distances from the vertex v_i to T_1 and T_2, that is,

$$d(v_i, T_1, T_2) = min\{d(v_i, T_1), d(v_i, T_2)\}.$$

The applications of 2-(k,l)-core is the same as (k,l)-core. It can be applied to design of communication networks such as railroad lines, high ways, pipelines, and transit routes. For more details on semi-obnoxious (k,l)-core and its application to real world, we refer the reader to [7] and [17].

3 The properties for special cases

In this section, we investigate some properties of 2-(k,l)-core in special cases. The basic idea of solving 2-(k,l)-core is the edge deletion method, which is tried to find an optimal solution by removing any edges and find a (k,l)-core in each obtained subtree. In the following theorems, we use this idea to show the properties of problem.

First, we consider the unweighted tree where all edges and vertices have the same positive weight. The following theorem shows that none of two subtrees of 2-(k,l)-core on an unweighted tree with $d_T > 2$ is a single vertex.

Theorem 1. *Let $T = (V, E)$ be an unweighted tree. Then, for the case $l > 0$ and $d_T > l+1$, there is a set of two subtrees T_1' and T_2', which are 2-(k,l)-core of T, that non of them is a single vertex.*

Proof. First, we show that a leaf cannot be one of 2-(k,l)-core of T. Let q be an edge between an inner vertex and a leaf v_i of T. By removing q, two subtrees T_1 and $T_2 = v_i$ are obtained. Let T_1' and $T_2' = v_i$ be the (k,l)-core of T_1 and T_2, respectively, and let $v_j \in T_1$ be the adjacent vertex to v_i. If $v_j \notin T_1'$, then by adding v_j to T_2', the objective function will not be increased. In the case when $v_j \in T_1'$, let $v_r \in T_1'$ be a non leaf adjacent vertex to v_j and let $T_1'(v_j)$ be a subtree of T_1' containing v_j obtained by deleting the edge (v_j, v_r). Then by adding all vertices in $T_1'(v_j)$ to T_2' and deleting them from T_1', the objective function does not increase. Note that since $d_T > l+1$, then there exists a non leaf vertex $v_r \in T_1'$ adjacent to v_j.

In the case that one of (k,l)-cores is an inner vertex, obviously, we can reduce the objective function by extending the core toward a leaf. □

Note that Theorem 1 is not true for the case $d_T \leq 2$. A counter example, for this case is the star graph, which is a tree that have just one vertex with degree more than one.

Now consider the case $w(T) < 0$. In this case, according to Theorems 2 and 3, the solution of (k,l)-core is a single vertex that may be a leaf. The proof of these theorems can be found in [7].

Theorem 2. *Let T be a tree with $w(T) < 0$. Then the (k,l)-core is the 1-median of T and vice versa.*

For any edge $(u, v) \in E$, suppose that T_{uv} is the subtree of T obtained by deleting edge (u, v), in which $v \in T_{uv}$.

Theorem 3. *Let T be a tree with $w(T) < 0$. Then one of the following two cases holds:*

1. *The (k,l)-core is a leaf.*
2. *The (k,l)-core is an inner vertex u, so that $w(u) \geq 0$ and $w(T_{uv}) \leq 0$ for each vertex v adjacent to u.*

Corollary 1. *Let $T = (V, E)$ be a tree with $w(T) < 0$. There is a solution for a $2-(k,l)$-core problem that one of (k,l)-cores is a single vertex.*

Proof. Let T_1' and T_2' be the $2-(k,l)$-core of T. Also let T_1 and T_2 be two subtrees of T containing all vertices which assigned to T_1' and T_2', respectively. So T_1' and T_2' is (k,l)-core of T_1 and T_2, respectively. Since $w(T) < 0$, then $w(T_1) < 0$ or $w(T_2) < 0$. Therefore by Theorem 2, T_1' or T_2' is a vertex. □

Theorem 4. *Let $T = (V, E)$ be a tree with $w(T) \geq 0$, let T_1 be a subtree of T with $d_{T_1} < l$, and let the number of leaves of T_1 be less than k. If $w(T \setminus T_1) > 0$, then T_1 is not a (k,l)-core of T.*

Proof. Let $T_1^c = T \setminus T_1$. Since $w(T_1^c) > 0$, there exists a component T_2 of T, which is obtained by removing T_1, so that $w(T_2) > 0$. Let $u \in T_2$ be the adjacent vertex to T_1 and $T_3 = T_1 \cup \{u\}$. Then

$$\begin{aligned} F(T_1) &= \sum_{v_i \in T_1^c \setminus T_2} w_i d(v_i, T_1) + \sum_{v_i \in T_2} w_i d(v_i, T_1) \\ &= \sum_{v_i \in T_1^c \setminus T_2} w_i d(v_i, T_1) + \sum_{v_i \in T_2} w_i (d(v_i, u) + d(u, T_1)) \\ &= F(T_3) + d(u, T_1) w(T_2) > F(T_3). \end{aligned}$$

Therefore T_1 is not the (k,l)-core of T. □

Zhou et al. [19] proved that the midpoint of two disjoint facility paths is a vertex of the tree. This property holds for 2-(k,l)-core case and can be written as the following theorem.

Theorem 5. *If two disjoint subtrees T_1 and T_2 are a 2-(k,l)-core of T, then the midpoint of the path between T_1 and T_2 is a vertex of the tree.*

4 Algorithm

As mentioned before to find a 2-(k,l)-core of a tree, we delete an edge of T and divide T to two subtrees T_1 and T_2. Then find the (k,l)-core of each of T_1 and T_2. This procedure is repeated for each edge of T and the best pair of (k,l)-cores is found. To find any (k,l)-core of T_1 and T_2, we can use the algorithm of Motevalli et al. [7]. The complexity of their algorithm is $O(n^2 logn)$, therefor the complexity of finding 2-(k,l)-core is $O(n^3 logn)$.

However for the case $w(T) < 0$, according to Lemma 3, the (k,l)-core is a vertex, which is either a leaf or an inner vertex u, so that $w(u) \geq 0$ and $w(T_{uv}) \leq 0$ for each vertex v adjacent to u. Also in the case $w(T) > 0$, we can use Theorem 4.

Therefore we can present the following algorithm.

Algorithm
Input: a tree T with the pos/neg weight
Output: a 2-(k,l)-core S^* of T and its objective function d^*
begin
$d^* := +\infty$
for each subtree T_1 and T_2 obtained from T by removing each edge $e \in E$ **do**
SUBTREE(T_i) for $i = 1, 2$
end

Where procedure SUBTREE(T) find (k,l)-core of tree T as follows:

Proedure $SUBTREE(T')$
Input: a subtree $T' = (V', E')$ of T with $\mid V' \mid = n'$ and the best current objective function d^*
Output: if the best subtree in T' has an objective function less than the previous value of d^*, the best subtree S^* in T' , having at most k leaves and with a diameter of at most l and its objective function as the new value of d^*
begin
if T' consists of one vertex **then**
$S' = T'$ and let $d(S')$ be its objective function
else
find a central vertex v of T'
if $w(T') < 0$, v is an inner vertex, $w(v) \geq 0$ and for each vertex u adjacent to v, $w(T_{vu}) \leq 0$
then v is the (k,l)-core, let $S' = v$ and set $d(S')$ as its objective function
else let $S' :=$ **BEST-TREE**(T', v)
if $d(S') < d^*$ and $W(T \setminus S') < 0$
$d^* := d(S')$
$S^* := S'$
for each subtree T^i obtained from T' by removing v **do**
SUBTREE(T^i)
end

In this procedure, a central vertex v of the tree T is the centroid of the corresponding unweighted tree of T, that is, a vertex minimizes the maximum number of vertices of the subtrees obtained by removing it. The procedure BEST-TREE(T', v), which is presented by Beker et al. [2], finds the best subtree S' containing v in T'^v having at most k leaves and a diameter of at most l.
Procedure $BEST - TREE(T', v)$
Input: a subtree $T' = (V', E')$ of T with n' vertices rooted at the central vertex v

Output: the best subtree S' containing v in T'^v having at most k leaves and a diameter of at most l and its objective function $d(S')$

begin
find the paths starting from v with the length at most l
for each path P starting from v with the length at most l
do
prune the tree T'^v and let $\hat{T'^v}$ be the new tree (see Prune)
find the distance savings of the paths from v to each vertex $u \neq v$ in $\hat{T'^v}$
if $deg(v) \geq 2$ **then**
find the path P'
find $P'' = P \cup P'$
else
$P'' = P$
find the set LS with all paths having the length no more than l and
containing P'' that their objective function are better than P''
if $\hat{T'^v}$ has more than k leaves, **then**
in LS select the $k-2$ paths to be added to P''
let S' be the best subtree containing v and
let $d(S')$ be its objective function
else
select all the paths in LS
let $S' = \hat{T'^v}$ and let $d(S')$ be its objective function
end for
end

The following notations are used in procedure BEST-TREE.

Let P_{uv} be the path between vertices u and v in the tree T and let T^u be the tree T rooted at u. We also denote by T^u_v the rooted subtree of T^u containing v and all the descendants of v. For each vertex u, let $f(u)$ be its father. The distance saving $sav(v, P_{vu})$ obtained by adding P_{vu} to the root v in T^u_v, is given by

$$sav(v, P_{vu}) = sav(v, P_{vf(u)}) + a(f(u), u)sum_c(u)$$

where if $v = f(u)$, then $sav(v, P_{vf(u)}) = sav(v, v) = 0$ and $sum_c(u)$ is calculated by

$$sum_c(v) = \begin{cases} w(v) & \text{if } v \text{ is a leaf of} T_c, \\ w(v) + \sum_{\text{u is a child of } v} sum_c(u) & otherwise, \end{cases} \quad (1)$$

where T_c is the selected subtree of T as the (k,l)-core in the current iteration. Let the path $P = P_{vu}$ be given and let B be the set of children of v. We denote by $T_{\bar{b}}^v$, $\bar{b} \in B$, the subtree in which P lies. In the algorithm, the tree T^v is pruned as follows:

1. Prune the paths that belong to subtrees T_b^v, with $b \in B \setminus \{\bar{b}\}$, in order to obtain paths with the length at most $min\{l - L(P), L(P)\}$.
2. For the paths that lie in the same subtree as $P = P_{vu}$ (i.e., in $T_{\bar{b}}^v$), prune those paths P_{xw}, with $x \in P$, $x \neq v$ and $w \in T_{\bar{b}}^v \setminus P$, such that $L(P_{xw}) > min\{L(P) - L(P_{vx}), l - L(P_{xu})\}$.

The pruned tree is called $\hat{T^v}$, and the weight of tree $\hat{T^v}$ is calculated by the following function:

$$w'(u) = \begin{cases} w(u) & \text{for each vertex } u \text{ of } \hat{T^v} \text{that is not a leaf,} \\ sum_v(u)a(u, f(u)) & \text{for each vertex } u \text{ that is a leaf of } \hat{T^v}. \end{cases} \quad (2)$$

In the rest of this section, we use the algorithm for a different case of the total weight of a tree.

Example 1. Consider the tree shown in Figure 1. The length of each edge is written on it. Also the weights of vertices are shown in Table 1. The total weight of tree is $w(T) = 3 > 0$. We want to find the 2-(3,3)-core on this tree. Since $w(T) > 0$, according to the algorithm, we should delete an edge of T and obtain two subtrees T_1 and T_2. Let us delete $v_9 - v_{10}$ and consider $T_1 = v_{10}$ and $T_2 = T \setminus T_1$. Now we should find (3,3)-core of each subtree. Because T_1 has only one vertex (v_{10}), so its (3,3)-core is v_{10}. To find the (3,3)-core of T_2, we should find a central vertex of T_2. We start with v_5 as a central vertex. Then we should find all of the paths starting from v_5 with the length at most 3. This paths are shown in Figure 2.

Table 1: The weights of vertices of tree in Figure 1 for Example 1

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}	w_{11}	w_{12}
3	-2	1	-1	1	2	-3	2	2	-1	-2	1

Now for each pat, prune the tree T and continue steps of the algorithm. For example, consider the path number 8 in Figure 2, the new tree after prune is presented in Figure 3.

The distance savings of $p : v_5\_v_4$ and $p : v_5\_v_6$ are $sav(v_5, p_{v_5v_4}) = -2$ and $sav(v_5, p_{v_5v_6}) = +2$. Since $deg(v_5) = 4$, we consider $p' = v_5 - v_6$, so $p'' = p \cup p'$ that is shown in Figure 3. Also since $\hat{T'^v}$ has 2 leaves, so we select all the paths in LS; then $S' = \hat{T'^v}$ and $d(S') = 0$. After performing the above steps for each path, we consider subtrees obtained by removing v_5 one by one and

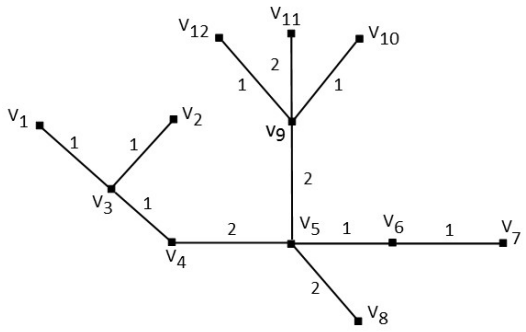

Figure 1: A tree with 12 vertices.

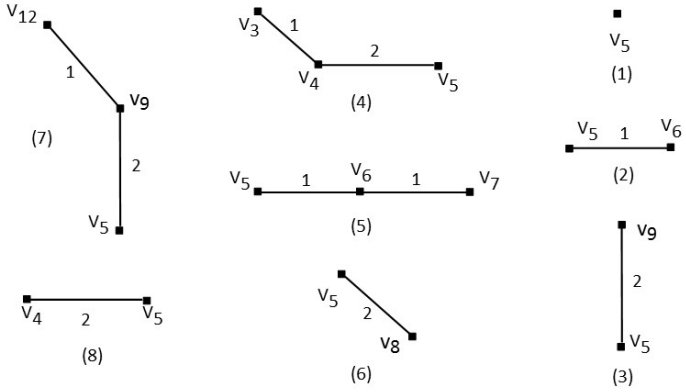

Figure 2: The paths starting from v_5 with the length at most 3.

find a central vertex for each subtree and continue above steps for each subtree. After that we delete other edges of tree such as $v_9 - v_{10}$ one by one and run all of the above steps after removing each edge. Finally 2-(3,3)-core of T is $T_1' = v_1$ and $T_2' = v_8$ that is obtained by removing of edge $v_1 - v_3$, and its objective function is equal to $F(T_1', T_2') = F(v_1) + F(v_8) = 0 + (-19) = -19$.

Example 2. Consider the tree depicted in Figure 4. The edge lengths are written on the edges. The weights of vertices are given in Table 2. The total weight of tree is $w(T) = -1 < 0$. We want to find the 2-$(3,4) - core$ on this tree. Since $w(T) < 0$, according to Theorem 2, the (k,l)-core is a vertex of T. The (k,l)-core of T is v_3, that is 1-median of T, too. The best solution is obtained by removing edge $v_6 - v_8$.Therefore 2-(k,l)-core is $T_1' = v_3$ and $T_2' = v_8$,

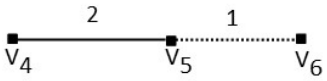

Figure 3: The path number 8 after prune.

and the objective function is $F(T_1', T_2') = F(T_1') + F'(T_2) = -25 + 0 = -25$.

Table 2: The weights of vertices of tree in Figure 1 for Example 1

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8
-1	3	2	-2	1	-3	-2	1

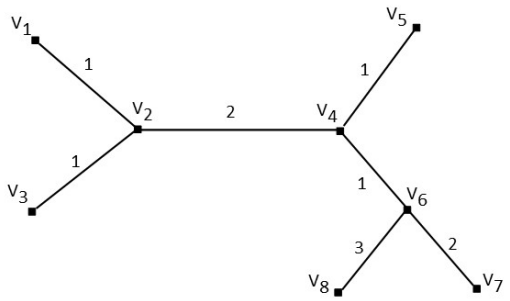

Figure 4: A tree with 8 vertices.

5 Summary and conclusion

In this paper, we considered the 2-(k, l)-core problem on a tree with positive and negative weights. We showed that, in the case when the sum of weights of tree is negative, the solution of 2-(k, l)-core is also 1-median. Some properties also stated for the case that the tree has the positive weight. Then a polynomial algorithm was presented to find the solution of this problem.

For a future research, it would be interesting to develop the algorithm on other special graphs such as extended stars, cactus graphs, interval graphs, and block graphs.

Acknowledgements

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

References

1. Becker R.I. *Inductive algorithms on finite trees,* Quaest Math., 13, (1990), 165–181.

2. Becker R.I., Lari I., Storchi G. and Scozzari A. *Efficient algorithms for finding the (k, l)-core of tree networks,* Networks, 40, (2002), 208–215.

3. Becker R.I. and Perl Y. *Finding the two-core of a tree,* Discrete Applied Mathematics, 11, (1985), 103–113.

4. Minieka E. and Patel N.H. *On finding the core of a tree with a specified length,* J. Alg., 4, (1983), 345–352.

5. Morgan C.A. and Slater P.J. *A linear algorithm for a core of a tree,* Journal of Algorithms, 1, (1980), 247–258.

6. Motevalli S. and Fathali J. *A linear algorithm for finding core of weighted interval trees*, Journal of Operational Research and Its Applications, 13, (2016), 101–111.

7. Motevalli S., Fathali J. and Zaferanieh M. *An efficient algorithm for finding the semi-obnoxious (k,l)-core of a tree,* Journal of Mathematical Modeling, 3, (2015), 129–144.

8. Peng S. and Lo W. *Efficient algorithms for finding a core of a tree with a specified length,* Journal of Algorithms 20, (1996), 445–458.

9. Peng S., Stephens A.B. and Yesha Y. *Algorithms for a core and a k-tree core of a tree*, J. Alg., 15, (1993), 143–159.

10. Rahbari M., Fthali J. and Mortazavi R. *A hybrid algorithm for the path center problem,* Global Analysis and Discrete Mathematics, 1, (2016), 83–92.

11. Shioura A. and Uno T. *A linear time algorithm for finding a k-tree core,* J. Alg., 23, (1997), 281–290.

12. Wang B.F. *Finding a k-tree core and a k-tree center of a tree network in parallel,* IEEE Transactions on Parallel and Distributed Systems, 9, (1998), 186–191.

13. Wang B.F. *Finding a 2-core of a tree in linear time,* SIAM Journal on Discrete Mathematics, 15, (2002), 193–210.

14. Wang Y., Wang D.Q., Liu W. and Tian B.Y. *Efficient parallel algorithms for constructing a k-tree center and a k-tree core of a tree network,* Lecture Notes in Computer Science 3827, Springer-Verlag Berlin Heidelberg, (2005), 553–562.

15. Wang Y. and Wang Y. *Efficient algorithms for constructing a (k,l)-center and a (k,l)-core in a tree network,* Fourth International Conference on Innovative Computing, Information and Control, (ICICIC), 2009.

16. Zaferanieh M. and Fathali J. *Ant colony and simulated annealing algorithms for finding the core of a graph,* World Applied Science Journal, 7, (2009), 1335–1341.

17. Zaferanieh M. and Fathali J. *Finding a core of a tree with pos/neg weight,* Math. Meth. Oper. Res., 76, (2012), 147–160.

18. Zelinka B. *Medians and peripherians of trees,* Archvum Mathematicum, 4, (1968), 87–95.

19. Zhou J., Kang L. and Shan E. *Two paths location of a tree with positive or negative weights,* Theoretical Computer Science, 607, (2015), 296–305.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 105–126
DOI:10.22067/ijnao.v9i1.69454

New S-ROCK methods for stochastic differential equations with commutative noise

A. Haghighi*

Abstract

The class of strong stochastic Runge–Kutta (SRK) methods for stochastic differential equations with a commutative noise proposed by Rößler (2010) is considered. Motivated by Komori and Burrage (2013), we design a class of explicit stochastic orthogonal Runge–Kutta Chebyshev ($SROCKC2$) methods of strong order one for the approximation of the solution of Itô SDEs with an m-dimensional commutative noise.The mean-square and asymptotic stability analysis of the newly proposed methods applied to a scalar linear test equation with a multiplicative noise is presented. Finally, some numerical experiments for stochastic models arising in applications are given that confirm the theoretical discussion.

Keywords: Stochastic differential equations; Runge–Kutta methods; Stochastic mean square stability, Stiff equations; Commutative noise.

1 Introduction

Stochastic differential equations ($SDEs$) arise in a variety of applied mathematics and physics areas; see [4,22]. Such equations are the results of adding random fluctuations in the parameters of a system of ordinary differential equations ($ODEs$). In the literature, there are plenty of work devoted to the study of stochastic differential calculus; see [10,12]. However, in many cases, the analytical solutions of such SDEs are not known; this makes numerical methods as very important tools for solving SDEs, which have been considered by many researchers. For solving different forms of SDEs, many efficient

*Corresponding author
Received 15 December 2017; revised 29 July 2018; accepted 30 September 2018
A. Haghighi
Department of Mathematics, Faculty of Science, Razi University, Kermanshah, Iran.
e-mail: a.haghighi@razi.ac.ir

numerical methods have been constructed, for example, [6, 12, 16, 23]. Especially, SRK methods are an important derivative free class of numerical methods, which are proposed for strong approximations of SDEs; see [5, 15, 17]. In 2010, Rößler [19, 20] introduced a new class of SRK methods in which the number of stages does not depend on the dimension, m, of the Wiener process. On the other hand, the stability of numerical methods for SDEs is an essential factor in order to avoid possible explosion. Generally, the implicit methods have a wider range of acceptable step sizes compared to explicit ones, which makes them suitable for the solution of stiff systems; see [7, 8, 12, 24]. However, the implementation of implicit methods demands to solve a nonlinear system of equations per step, which may increase the computational cost significantly in the case of high dimensional SDEs. Therefore, explicit methods with extended stability regions could greatly help to solve stiff SDEs. In this regard, Abdulle et al. proposed two classes of explicit stochastic orthogonal Runge–Kutta Chebyshev ($SROCK$) methods of strong order 1/2 for SDEs in Stratonovich [1] and Itô [2] sense. Recently, Komori et al. in [14], constructed explicit strong SRK methods ($SROCKD1$ and $SROCKD2$) of order one for SDEs in Itô and Stratonovich sense. These methods are designed for approximation of SDEs with noncommutative noises. As we know, double stochastic integrals must be used in methods of strong order one for the approximation of the solution of a SDE with a noncommutative noise; see [12, 13]. Such an expensive computational cost can be reduced when the SDE has a single Wiener process or has a commutative noise. It should be noted, such SDEs can be used for modeling a variety of nonlinear physical systems; see [12]. In this article, the SRK methods of [20] for the strong approximation of SDEs with a commutative noise is briefly reviewed. Then, motivated by [14], a new class of SROCK methods of strong order one for the approximation of the solution of SDEs with an m-dimensional commutative noise is proposed. The mean-square stability (MS-stability) functions for the new proposed methods, applied to a scalar linear test equations with a multiplicative noise, are determined, and in some special cases, the regions of MS-stability and T-stability are compared between our proposed methods and the scalar test equation.

Consider the autonomous SDE of the Itô type in the form of

$$dX(t) = a(X(t))dt + b(X(t))dW_t, \quad X(t_0) = X_0, \quad t \in [t_0, T]. \tag{1}$$

Here, $X(t) \in \mathbb{R}^d$, $a : [t_0, T] \times \mathbb{R}^d \to \mathbb{R}^d$, is a drift vector, $b = (b_1, b_2, \ldots, b_m) : [t_0, T] \times \mathbb{R}^d \to \mathbb{R}^{d\times m}$ is a diffusion matrix, and $W(t)$ is an m-dimensional standard Wiener process defined on probability space $(\Omega, \mathcal{F}, P)$ with the filtration $\{\mathcal{F}_t\}_{t\in[t_0,T]}$, which satisfies the usual conditions. It is assumed, X_0 is $\mathcal{F}_{t_0}$ -measurable and independent of the Wiener process with $E|X_0|^2 < \infty$, where $|.|$ denotes the Euclidean norm. Suppose that the conditions of the existence and uniqueness theorem [12] are fulfilled for SDE (1). For simplicity in this paper, the equidistant discretization $p_h = \{t_0, t_1, \ldots, t_N = T\}$ of the

time interval $[t_0, T]$ with $t_n = t_0 + nh$, where $h = \frac{T-t_0}{N}$ for $n = 0, 1, \ldots, N$, has been considered. Also, for an equidistant grid point $t_n \in p_h$, the notation y_n denotes a discrete approximation to the solution $X(t_n)$ of (1).

In the following, the SDE (1) is considered with a commutative noise, in the other words, the diffusion matrix b satisfies the commutativity condition

$$\sum_{i=1}^{d} b^{i,r} \frac{\partial b^{k,s}}{\partial x^i}(t,x) = \sum_{i=1}^{d} b^{i,s} \frac{\partial b^{k,r}}{\partial x^i}(t,x), \tag{2}$$

for all $r, s = 1, \ldots, m$, $k = 1, \ldots, d$, and $(t, x) \in [t_0, T] \times \mathbb{R}^d$.

Definition 1 (see [12, 20]). *A discrete time approximation y^h is said to be convergent with the strong order α to the solution X of SDE (1) at time t_n, if there exist constants $C > 0$ and $\delta_0 > 0$ such that, for each $h \in (0, \delta_0)$,*

$$\left(E|X(t_n) - y_n^h|^2\right)^{\frac{1}{2}} \leq Ch^{\alpha}. \tag{3}$$

The constants C and δ_0 are independent of h.

The outline of the paper is as follows: In Section 2, we will introduce second order Chebyshev methods proposed by Abdulle and Medovikov in [3] for ODEs. In Section 3, we introduce the class of SRK methods proposed by Rößler in [20] for the strong approximation of SDEs with a commutative noise. Then, we introduce a new class of methods of strong order one for the approximation of the solution of Itô SDEs with an m-dimensional commutative noise. In Section 4, a brief overview of the stochastic stability concepts of SDEs is presented. We also analyze the MS-stability and T-stability properties of the newly proposed method on a linear scalar SDE. In Section 5, some numerical results for stochastic models arising in applications are provided to confirm theoretical discussions.

2 Chebyshev methods of order two for ODEs

Consider the autonomous d-dimensional ODE system $X'(t) = a(X(t))$ with the initial condition $X(t_0) = y_0$. The s-stage explicit deterministic Runge–Kutta (RK) method for solving this system corresponding to step-size h is as follows:

$$y_{n+1} = y_n + h \sum_{i=1}^{s} \alpha_i a(H_i^{(0)}), \tag{4}$$

where

$$H_i^{(0)} = y_n + h \sum_{j=1}^{i-1} a_{ij} a(H_j^{(0)}), \quad 1 \leq i \leq s.$$

Applying (4) to the scalar test equation yields

$$X'(t) = \lambda X(t), \quad t > 0, \quad y(0) = y_0, \tag{5}$$

where $\Re(\lambda) < 0$ and $y_0 \neq 0$, we have $y_{n+1} = R_s(\lambda h) y_n$. Here, $R_s(z)$ is called a stability function. Consequently, the stability region of (4) can be expressed as

$$R_S = \{z \in \mathbb{C} : |R_s(z)| \leq 1\}. \tag{6}$$

Riha [18] showed that the polynomial

$$R_s(z) = 1 + z + \frac{z^2}{2} + \sum_{j=3}^{s} c_{sj} z^j, \quad c_{sj} \in \mathbb{R},$$

exists such that $|R_s(z)| \leq 1$ for $z \in [-l_s, 0]$ with l_s as large as possible. It is desirable in practice to replace $|R_s(z)| \leq 1$ by $|R_s(z)| \leq \eta \leq 1$ (damping). Abdulle et al. in [3] constructed approximations to these optimal stability polynomials. In this approach, at first for a given η (damping factor) and s (stage value), the values a_s and l_s and also the polynomial $\bar{w}(x)$ of the degree two with complex zeros $\alpha_s \pm \beta_s$ which satisfies $\bar{w}(a_s) = 1$, are calculated. Then, the orthogonal polynomials $\{\bar{Q}_j(x)\}_{j=0}^{s-2}$ with respect to the weight function $\frac{\bar{w}(x)}{\sqrt{1-x^2}}$ on $[-1, 1]$ are constructed in which $\bar{Q}_j(a_s) = 1$ for $j = 0, 1, \ldots, s-2$. Finally, the approximation of optimal stability polynomials is constructed as

$$R_s(x) := w(x) Q_{s-2}(x), \tag{7}$$

where $w(x) := \bar{w}(a_s + x/d_s)$ with $d_s = \frac{l_s}{1+a_s} > 0$ and $Q_j(x) := \bar{Q}_j(a_s + x/d_s)$ for $j = 0, 1, \ldots, s-2$. For more details, you can refer to [3]. The polynomials $Q_0(x), Q_1(x), \ldots, Q_{s-2}(x)$ satisfy in the below three-term recurrence relations as form

$$\begin{aligned} &Q_0(z) = 1, \quad Q_1(z) = 1 + \mu_1 z, \\ &Q_j(z) = (\mu_j z + \kappa_j + 1) Q_{j-1}(z) - \kappa_j Q_{j-2}(z), \end{aligned} \tag{8}$$

for $j = 1, \ldots, s-2$. The values of μ_j and κ_j can be determined by inserting the different suitable values $r_1, r_2 \in \mathbb{R} \setminus \{0\}$ in (8) and solving the nonsingular linear system

$$(\mu_j r_i + \kappa_j + 1) Q_{j-1}(r_i) - \kappa_j Q_{j-2}(r_i) = Q_j(r_i), \quad i = 1, 2.$$

Abdulle et al. in [3], corresponding to stability polynomial (7), constructed the second order explicit RK method $\{y_k\}_{k \geq 0}$ for the approximation of the solution of the deterministic system $X'(t) = a(X(t))$. The mentioned method is implemented in a code called ROCK2 as below:

$$H_1^{(0)} = y_n,$$

$$
\begin{aligned}
H_2^{(0)} &= y_n + h\mu_1 a(H_1^{(0)}),\\
H_{j+1}^{(0)} &= h\mu_j a(H_j^{(0)}) + (\kappa_j + 1)H_j^{(0)} - \kappa_j H_{j-1}^{(0)}, \qquad 2 \le j \le s-2,\\
H_s^{(0)} &= H_{s-1}^{(0)} + h\theta_s a(H_{s-1}^{(0)}),\\
y_{n+1} &= H_s^{(0)} + h\theta_s a(H_s^{(0)}) - h\theta_s(1 - \frac{\tau_s}{\theta_s^2})(a(H_s^{(0)}) - a(H_{s-1}^{(0)})),
\end{aligned} \tag{9}
$$

where the parameters θ_s and τ_s satisfy in the relation $w(x) = 1 + 2\theta_s x + \tau_s x^2$.

3 SRK methods for SDEs with a commutative noise

In this section, we will review the SRK method introduced by Rößler in [20] for the strong approximation of Itô SDEs with m-dimensional commutative noise. In this class, the SRK method is given by $y_0 = X_0$ and

$$
\begin{aligned}
y_{n+1} = y_n &+ \sum_{i=1}^{s} \alpha_i a(t_n + c_i^{(0)} h_n, H_i^{(0)}) h_n\\
&+ \sum_{i=1}^{s}\sum_{k=1}^{m} (\beta_i^{(1)} I_{(k),n} + \beta_i^{(2)} \sqrt{h_n}) b^k(t_n + c_i^{(1)} h_n, H_i^{(k)}),
\end{aligned} \tag{10}
$$

for $n = 0, 1, 2, \ldots$, with stage-values

$$
\begin{aligned}
H_i^{(0)} &= y_n + \sum_{j=1}^{s} A_{ij}^{(0)} a(t_n + c_j^{(0)} h_n, H_j^{(0)}) h_n + \sum_{j=1}^{s}\sum_{l=1}^{m} B_{ij}^{(0)} b^l(t_n + c_j^{(1)} h, H_j^{(l)}) I_{(l),n},\\
H_i^{(k)} &= y_n + \sum_{j=1}^{s} A_{ij}^{(1)} a(t_n + c_j^{(0)} h_n, H_j^{(0)}) h_n - \sum_{j=1}^{s} B_{ij}^{(1)} b^k(t_n + c_j^{(1)} h_n, H_j^{(k)}) \frac{\sqrt{h_n}}{2}\\
&\quad + \sum_{j=1}^{s}\sum_{l=1}^{m} B_{ij}^{(1)} b^l(t_n + c_j^{(1)} h_n, H_j^{(l)}) \frac{I_{(k),n} I_{(l),n}}{2\sqrt{h_n}},
\end{aligned}
$$

for $i = 1, \ldots, s$ and $k = 1, \ldots, m$. The random variables of the method are defined by the stochastic Itô integrals

$$
I_{(k),n} = W^k(t_n + h) - W^k(t_n) = \int_{t_n}^{t_n+h} dW_s^k.
$$

Since the random variables of the method (10) are only increments of the Wiener process and the simulation of the multiple stochastic integrals is not required any more, the computational cost will be reduced significantly.

Table 1: Butcher tableau for the SRK methods $SRIC1$ (left) and $SRIC2$ (right)

0						
0	0			0		
0	0	0		0	0	
0						
0	0			1		
0	0	0		-1		
	1	0	0	1	0	0
				0	$\frac{1}{2}$	$-\frac{1}{2}$

0						
0	1			0		
0	0	0		0	0	
0						
1	1			1		
1	1	0		-1		
	$\frac{1}{2}$	$\frac{1}{2}$	0	1	0	0
				0	$\frac{1}{2}$	$-\frac{1}{2}$

In the following, let $\mathcal{C}^{p,q}([t_0,T]\times\mathbb{R}^d,\mathbb{R}^d)$ denote the space of all mappings from $[t_0,T]\times\mathbb{R}^d$ to $\mathbb{R}^d$, which are p and q times continuously differentiable with respect to the time and the state variables, respectively.

Remark 1. *Frequently, the coefficients of the SRK method (10) are shown by the following Butcher tableau :*

$c^{(0)}$	$A^{(0)}$	$B^{(0)}$
$c^{(1)}$	$A^{(1)}$	$B^{(1)}$
	α^T	${\beta^{(1)}}^T$
		${\beta^{(2)}}^T$

Theorem 1 (Order conditions)**.** *Let $a, b^j \in \mathcal{C}^{1,2}([t_0,T]\times\mathbb{R}^d,\mathbb{R})$ for $j = 1,\dots,m$. Then, the SRK method (10) obtains the strong order* 0.5 *for approximation of the solution of (1), if its coefficients satisfy in the equations*

$$1.\quad \alpha^T e = 1, \qquad 2.\quad {\beta^{(1)}}^T e = 1, \qquad 3.\quad {\beta^{(2)}}^T e = 0.$$

In addition, suppose $a, b^j \in \mathcal{C}^{1,3}([t_0,T]\times\mathbb{R}^d,\mathbb{R})$ for $j = 1,\dots,m$. Then, the SRK method (10) obtains the strong order 1.0*, if the equations*

$$4.\quad {\beta^{(2)}}^T A^{(1)} e = 0, \qquad 5.\quad {\beta^{(2)}}^T (B^{(1)} B^{(1)} e) = 0,$$
$$6.\quad {\beta^{(1)}}^T B^{(1)} e = 0 \qquad 7.\quad {\beta^{(2)}}^T (B^{(1)} e)^2 = 0, \qquad 8.\quad {\beta^{(2)}}^T B^{(1)} e = 1,$$

are fulfilled with $c^{(i)} = A^{(i)} e$ for $i = 1, 2$.

Proof. see [20]. □

To date, suggestions have been presented for the coefficients of SRK method (10). As an example, Rößler in [20] purposed three stages of SRK methods $SRIC1$ and $SRIC2$ with orders $(p_D, p_S) = (1.0, 1.0)$ and $(p_D, p_S) = (2.0, 1.0)$, respectively, in which p_D is the order of the method related to ODEs, that is, $b \equiv 0$ and p_S is the order of the method related to SDEs; see Table 1.

Remark 2. *In the rest of the paper, we set* $B^{(0)} = 0$ *and*

$$B^{(1)} = \begin{pmatrix} 0 \cdots & 0 & 0\,0 \\ \vdots \cdots & \vdots & \vdots\,\vdots \\ 0 \cdots & 0 & 0\,0 \\ 0 \cdots & 1 & 0\,0 \\ 0 \cdots & -1 & 0\,0 \end{pmatrix}_{s\times s}, \beta^{(1)} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}_{s\times 1}, \beta^{(2)} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix}_{s\times 1}, \quad (11)$$

therefore, the equations 2. and 3. as well as the equations 5.–8. in Theorem 1 are satisfied.

3.1 A new class of Chebyshev methods for SDEs

In this section, motivated by Komori et al. [14], we introduce a new class of s-stage explicit stochastic orthogonal Runge–Kutta Chebyshev method for Itô SDEs with a commutative noise. We denote this method by $SROCKC2$, which has the orders $(p_D, p_S) = (2.0, 1.0)$.

For the strong approximation of the solution of SDE (1) with a commutative noise, the s-stage ($s \geq 2$) SROCKC2 method is given by $y_0 = X_0$ and

$$y_{n+1} = H^{(0)}_{s+1} + \sum_{i=1}^{s}\sum_{k=1}^{m}(\beta^{(1)}_i I_{(k),n} + \beta^{(2)}_i \sqrt{h_n}) b^k(t_n + c^{(1)}_i h_n, H^{(k)}_i), \quad (12)$$

with stage values

$$\begin{aligned}
H^{(0)}_1 &= y_n, \qquad H^{(0)}_2 = y_n + h\mu_1 a(H^{(0)}_1), \\
H^{(0)}_{j+1} &= h\mu_j a(H^{(0)}_j) + (\kappa_j + 1)H^{(0)}_j - \kappa_j H^{(0)}_{j-1}, \qquad 2 \leq j \leq s-2, \\
H^{(0)}_s &= H^{(0)}_{s-1} + h\theta_s a(H^{(0)}_{s-1}), \\
H^{(0)}_{s+1} &= H^{(0)}_s + h\theta_s a(H^{(0)}_s) - h\theta_s(1 - \frac{\tau_s}{\theta_s^2})(a(H^{(0)}_s) - a(H^{(0)}_{s-1})), \\
H^{(k)}_i &= Y_n + \sum_{j=1}^{s} A^{(1)}_{ij} a(H^{(0)}_j) h_n - \sum_{j=1}^{s} B^{(1)}_{ij} b^k(H^{(k)}_j) \frac{\sqrt{h_n}}{2} \\
&\quad + \sum_{j=1}^{s}\sum_{l=1}^{m} B^{(1)}_{ij} b^l(H^{(l)}_j) \frac{I_{(k),n} I_{(l),n}}{2\sqrt{h_n}},
\end{aligned}$$

where μ_j and κ_j ($j = 1, \ldots, s-2$) as well as θ_s and τ_s are free parameters in $\mathbb{R}$, which are in accordance with the second method ROCK2 in (9); see [3]

for more details. In addition, $B^{(0)}$, $B^{(1)}$, $\beta^{(1)}$ and $\beta^{(2)}$ would be defined as previously mentioned in relation (11).

Remark 3. *We note that in the deterministic case, $b \equiv 0$, the method (12) coincides with the second order ROCK2 method, introduced in (9). Therefore, assuming $y_n + \sum_{i=1}^{s} \alpha_i a(H_i^{(0)}) = H_{s+1}^{(0)}$ from second order properties of the ROCK2 method, we have $\sum_{i=1}^{s} \alpha_i = 1$, and the equation 1. in Theorem 1 is satisfied.*

4 Stability properties

In this section, we will first investigate the MS-stability of our proposed methods and compare them with the MS-stability regions of the methods SRIC1 and SRIC2 in [20]. Then, the T-stability properties of the proposed schemes on a scalar linear test equation with a multiplicative noise have been analyzed. In this regard, we first review the MS-stability analysis of the methods SRIC1 and SRIC2 on the linear test problem with the multiplicative noise

$$dX(t) = \lambda X(t)dt + \mu X(t)dW_t, \quad \lambda, \mu \in C, \tag{13}$$

with the deterministic initial condition $X(t_0) = y_0 \in \mathbb{R}\backslash\{0\}$. For $\lambda, \mu \in \mathbb{C}$, the solution of (13), $X(t) = y_0 exp\{(\lambda - \frac{1}{2}\mu^2)(t - t_0) + \mu(W_t - W_{t_0})\}$, is stochastically asymptotically stable in the large if and only if

$$\lim_{t\to\infty} |X(t)| = 0 \Leftrightarrow \Re(\lambda - \tfrac{1}{2}\mu^2) < 0. \tag{14}$$

Also, the equilibrium position of SDE (13) is asymptotically MS-stable if and only if

$$\lim_{t\to\infty} E(|X(t)|^2) = 0 \Leftrightarrow 2\Re(\lambda) + |\mu|^2 < 0, \tag{15}$$

for $\lambda, \mu \in \mathbb{C}$; see [12]. It worth mentioning, MS-stability always results in asymptotically stability in the large due to inequality $\Re(2\lambda - \mu^2) \leq 2\Re(\lambda) + |\mu|^2$.

Here, we are going to find conditions in which the SRIC1 and SRIC2 methods, on the scalar test equation (13), have numerically stable solutions. For every $\lambda, \mu \in \mathbb{C}$ a numerical method applied on the test equation (13) is numerically asymptotically stable and MS-stable, respectively, if and only if $\lim_{n\to\infty} |y_n| = 0$ (with probability 1) and $\lim_{n\to\infty} E(|y_n|^2) = 0$. To determine the domain of asymptotic stability of a numerical method, we recall the following lemma from [9].

Lemma 1. *For a given sequence of real-valued, nonnegative, independent, and identically distributed random variables $\{R_n(h, \lambda, \mu)y_n\}_{n=0}^{\infty}$. Consider the sequence of random variable $\{|y_n|\}_{n=0}^{\infty}$ defined by $y_{n+1} = R_n(h, \lambda, \mu)y_n$ and $|y_0| \neq 0$ (w.p.1). If the random variables $\log(|R_n(h, \lambda, \mu)|)$ are square*

integrable, then

$$\lim_{n\to\infty} |y_n| = 0 \Leftrightarrow E\Big(\log(|R_n(h,\lambda,\mu)|\Big) < 0. \tag{16}$$

So the domain of asymptotic stability of a numerical method can be expressed as

$$R_{AS} = \{\lambda, \mu \in \mathbb{C} : E\Big(\log(|R_n(h,\lambda,\mu)|\Big) < 0\}.$$

Set $a_1 := 1 + x - \frac{y^2}{2}$ and $a_2 := 1 + x + \frac{1}{2}x^2 - \frac{y^2}{2}$, where $x = h\lambda$ and $y = \mu\sqrt{h}$. Applying SRIC1 and SRIC2 methods on the scalar test equation (13), respectively, we have

$$y_{n+1} = \Big(a_j + \frac{I_{(1),n}}{\sqrt{h}}y + \frac{I^2_{(1),n}}{2h}y^2\Big)y_n, \quad j = 1, 2. \tag{17}$$

Here, we find the forms

$$y_{n+1} = R^j_n(x, y)y_n, \quad j = 1, 2,$$

for (17), and the MS-stability functions of the methods SRIC1 and SRIC2 will be given, respectively, by

$$E(|y_{n+1}|^2) = \hat{R}^j_n(x, y)E(|y_n|^2), \quad j = 1, 2.$$

Obviously, for $\lambda, \mu \in \mathbb{C}$ and $h > 0$, the methods SRIC1 and SRIC2 are MS-stable, if $\hat{R}^j_n(x, y) < 1$ for $j = 1, 2$. So, the domain of MS-stability of SRIC1 and SRIC2 methods can be expressed, respectively, as

$$R^j_{MS} = \{\lambda, \mu \in \mathbb{C} : \hat{R}^j_n(x, y) < 1\}, \quad j = 1, 2.$$

Because of better visualization of the regions R^1_{MS} and R^2_{MS} in the $x - y$ plane, the restriction $\lambda, \mu \in \mathbb{R}$ is considered. So, one can express the MS-stability function of the SRIC1 and SRIC2 methods, respectively, as

$$\hat{R}^j(x, y) = a_j^2 + y^2 + \frac{3}{4}y^4 + y^2 a_j, \quad j = 1, 2.$$

It should be noted that all the following figures, demonstrating regions of MS-stability for the numerical methods under consideration, are plotted, using the Mathematica software. Figure 1 gives the MS-stability regions of the methods SRIC1 and SRIC2. The light gray area shows the MS-stability region of the test equation (13) and the dark gray area shows the MS-stability regions of the methods. Comparison of the illustrated regions in Figure 1 shows that the SRIC1 and SRIC2 methods have small regions of MS-stability.

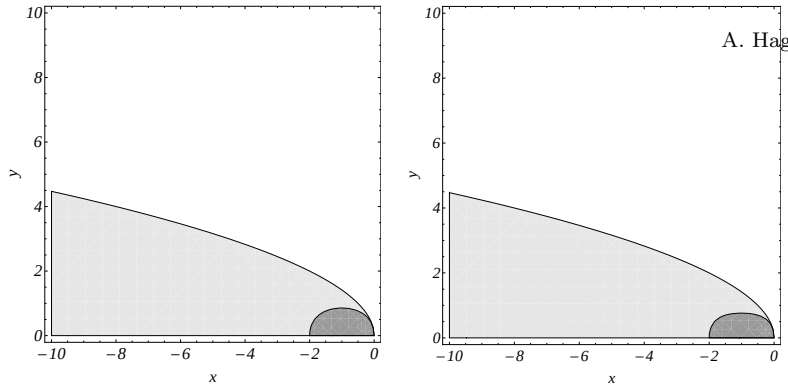

Figure 1: MS-stability region for the test equation (13) (light grey surface) and SRK methods (dark grey surface) (left) SRIC1, (right) SRIC2.

4.1 MS-stability analysis of SROCKC2

Consider the method (12), for simplification similar to [14], we set all components of $A^{(1)}$ as zero except

$$A^{(1)}_{s-2,j} = A^{(1)}_{s-1,j} = A^{(1)}_{s,j} = A^{(0)}_{s-1,j}, \quad 1 \le j \le s-2,$$

where $A^{(0)}$ is an $s \times s$ strictly lower triangular matrix which could be obtained from the following equation:

$$H^{(0)}_i = y_n + \sum_{j=1}^{i-1} A^{(0)}_{i,j} a(H^{(0)}_{j-1})$$

for $i = 1, \ldots, s$. Clearly, from the definition of $A^{(1)}$, the equation 4. in Theorem 1 is satisfied.

Now, suppose that $R_s(x) = w(x)Q_{s-2}(x)$ is the stability polynomial of the s-stage deterministic method ROCKD2 (9) with parameter η; see [3]. When the method (12) is applied to the test equation (13), we find the forms $y_{n+1} := R_s(h, \lambda, \mu, \Delta W_n)y_n$, where

$$R_s(h, \lambda, \mu, \Delta W_n) = \Big(w(\lambda h) + \Delta W_n \mu + \frac{1}{2}[\Delta W_n]^2\mu^2 - \frac{1}{2}h\mu^2\Big)Q_{s-2}(\lambda h).$$

Applying the expectation operator and according to the properties of standard normal distribution in the terms of x and y, we have

$$E\Big(|R_s(x, y)|^2\Big) = \Big(w(x)^2 + y^2 + \frac{y^4}{2}\Big)[Q_{s-2}(x)]^2.$$

Therefore, the mean square stability regions of the methods SROCKC2 is

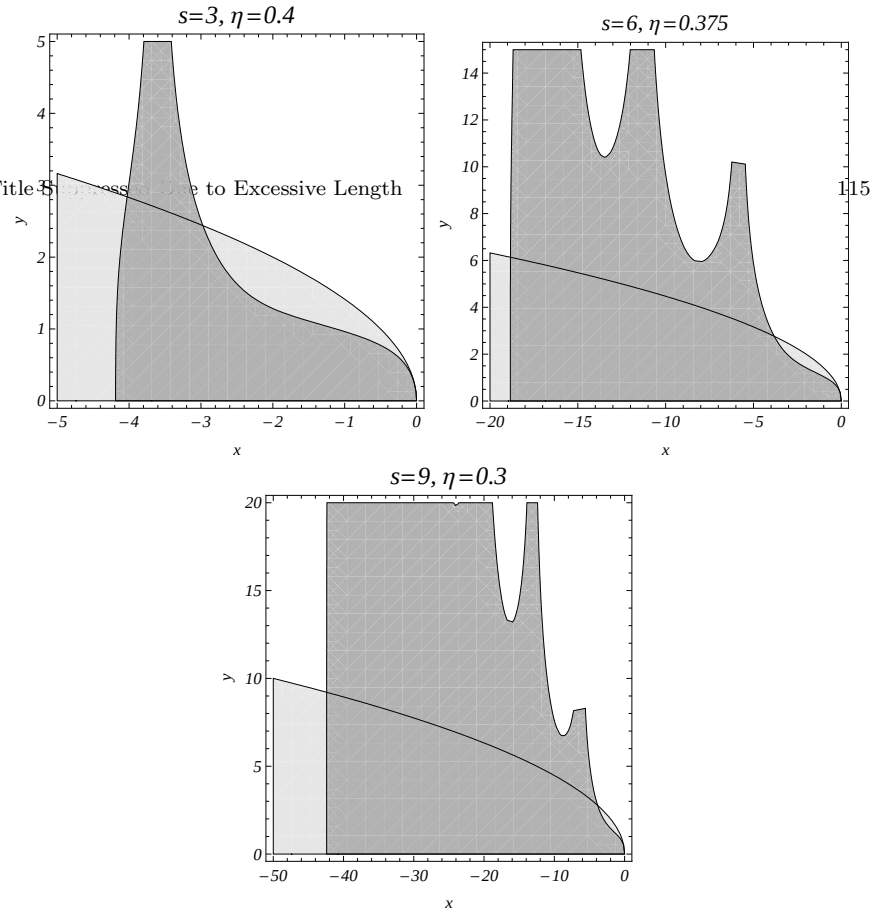

Figure 2: Profile of the MS-stability regions of the SROCKC2 schemes for some special values of η and s.

$$R^{MS} = \{(x, y) \in \mathbb{R}^2 : E\Big[|R_s(x, y)|^2\Big] < 1\}. \tag{18}$$

For more clear results concerning the MS-stability regions of SROCKC2 schemes, in the following, we confine our investigation to some optimal values of η and s, which have been suggested by [14]. The regions of MS-stability for SROCKC2 schemes with different values of η and s (dark gray area) are illustrated in Figure 2. We can see in Figure 2 that the MS-stability region of methods SROCKC2 is larger than the methods SRIC1 and SRIC2. In what follows, SROCKC2 scheme with three stages ($\eta = 0.4$), six stages ($\eta = 0.375$), and nine stages ($\eta = 0.3$) are denoted by $SROCKC2 - 3$, $SROCKC2 - 6$, and $SROCKC2 - 9$, respectively.

4.2 T-stability analysis of SROCKC2

To measure the asymptotic stability of a numerical method with respect to the driving process, Saito and Mitsui established the definition of T-stability in [21].

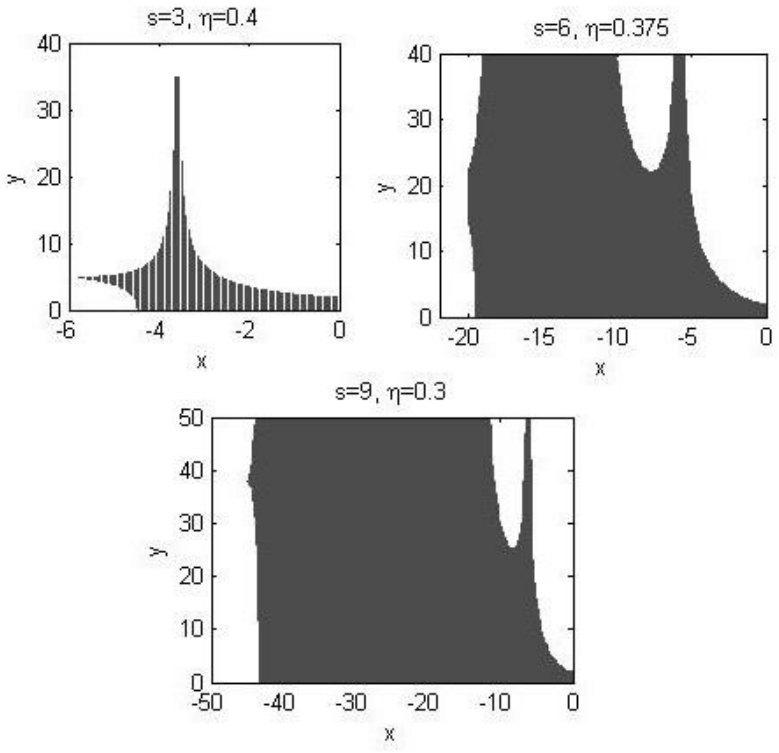

Figure 3: Profile of the T-stability regions of the SROCKC2 schemes for some optimal values of η and s.

Definition 2. *Assume that the test equation* (13) *is stochastically asymptotically stable in the large. The numerical scheme equipped with a specified driving process said to be T-stable if* $\lim_{n\to\infty}|y_n| = 0$ *holds for the driving process.*

It can be deduced that the criterion of T-stability depends both on the scheme and the driving process. Accordingly, similar to the approach, which is used by Lemma 1, the criterion of T-stability of a numerical method with respect to normal random variables can express as

$$\log T(h,\lambda,\mu) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty}\Big(\log(|R_n(h,\lambda,\mu)|e^{-\frac{1}{2}z^2}dz\Big) < 0. \tag{19}$$

It seems it is not so simple to find the closed form of (19). Thus, we use the MATLAB software to approximate the integral in (19). The integral interval $(-\infty,+\infty)$ is approximated by $[-50,50]$ because of the magnitude of the integrand in (19) becomes sufficiently small when $|z| > 50$. In Figure 3, the regions of T-stability for SROCKC2 schemes with different values of

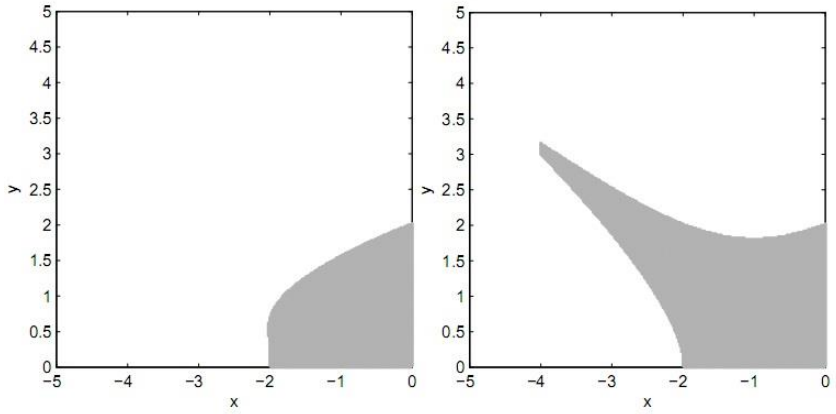

Figure 4: Profile of the T-stability regions of the methods SRIC1 (left) and SRIC2 (right).

η and s (dark gray area) have been illustrated. Similarly, the regions of T-stability for the methods SRIC1, SRIC2, Milstein and Platen (light gray area) are depicted in Figures 4 and 5. Comparison of the illustrated regions confirms that the proposed methods have reasonably larger regions of both MS-stability and T-stability.

5 Numerical experiments

In this section, we apply some optimal cases of SROCKC2 schemes (12) on some examples of Itô stochastic differential equations that arising in applications. These examples illustrate the efficiency of the new proposed methods. Denote $y_N^{(i)}$ and $X^{(i)}(t_N)$ as the numerical solutions and the exact solution at step point t_N in ith simulation, respectively. The root mean square error (RMSE) is used by simulating 1000 independent trajectories for a given step-size h as bellow:

$$RMSE := \left(\frac{1}{1000} \sum_{i=1}^{1000} \left| X^{(i)}(t_N) - y_N^{(i)} \right|^2 \right)^{\frac{1}{2}}.$$

We also investigate computational costs for a given h to measure accuracy of proposed methods. In simulation results, the number of evaluations of the drift function a, of the diffusion function b, and the number of random numbers to be generated in every step is taken as measure of computational costs (S_a). In the following, we will compare the methods SROCKC2-3, SROCKC2-6, and SROCKC2-9 with some popular methods, including Euler-

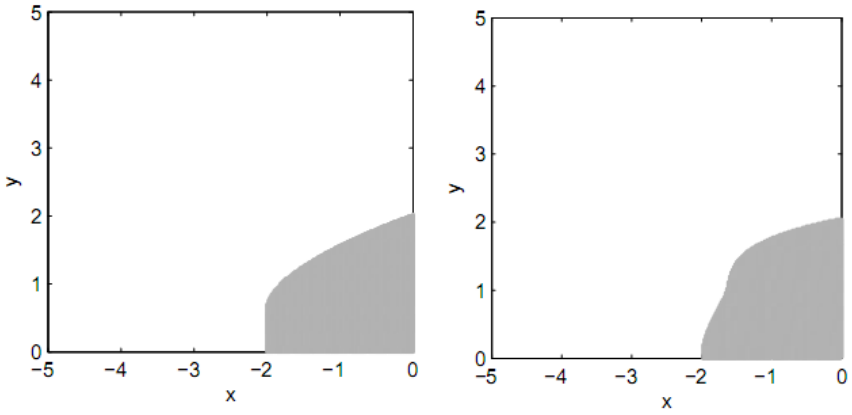

Figure 5: Profile of the T-stability regions of the methods Milstein (left) and Platen (right).

Table 2: A least squares fit for the parameters C and α, for problem 1 with $\lambda = 1, \mu = 1$ $X_{t_0} = 1$

	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC1
α	1.0514	1.0597	1.0615	1.0207	1.0109	1.0109
$\log(C)$	1.8067	1.7033	1.6803	2.0750	2.2317	2.2317

Maruyama [12], Milstein [16], SRIC1 and SRIC2 [20], and also the method proposed by Platen [12].

Problem 1: The first problem is the scalar test equation (13) that is considered on $I = [0, 1]$ with the initial condition $X_{t_0} = 1$. For this setting, the simulation results are presented in Figure 6. In this figure, RSME versus step size of the methods are illustrated at $T = 1$. It is clearly seen in Figure 6 that the slopes of curves appear to match well and (3) for $\alpha = 1$ is valid. Further, with assumption $RSME \approx Ch^{\alpha}$ for some constants C and α, we can write

$$\log(RSME) \approx \log(C) + \alpha \log(h). \tag{20}$$

In Table 2, a least squares fit for the parameters C and α are calculated, respectively. These results confirm that the proposed methods have a convergence with order 1 in mean square sense. Also, in Table 3, the RMSE for time steps $h = 2^{-9}, \ldots, 2^{-5}$ are computed.

As a further aspect of providing comparisons of the computational efficiency of the proposed methods, ratio CPU times of the methods SROCKC2, Milstein, Platen, and SRIC2 to the running time of the method SRIC1 are calculated in Table 4. It can be deduced from Table 4 that the CPU time of the

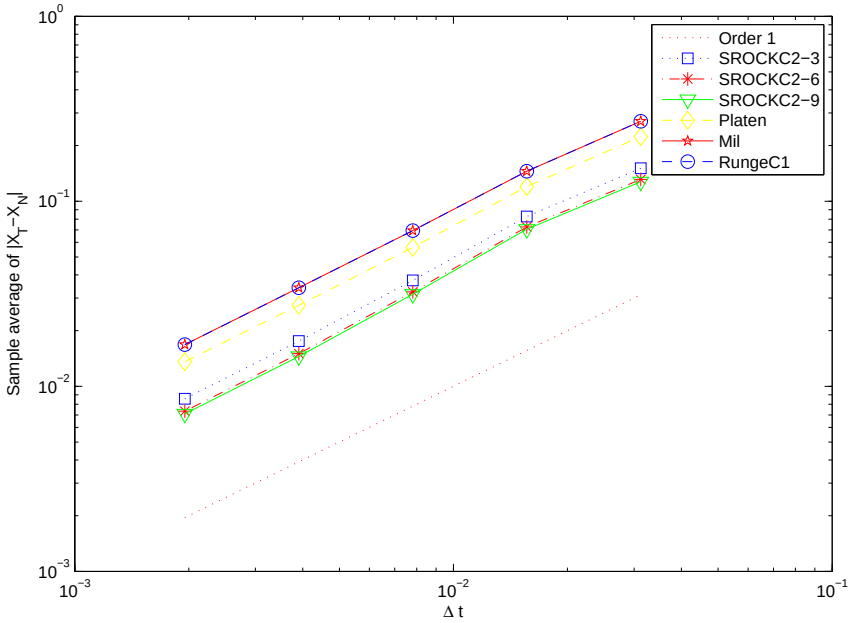

Figure 6: RMSE versus step size for test problem 1.

Table 3: Means of absolute errors, RMSE, for problem 1

h	SROCKC2-3	SROCKC2-3	SROCKC2-3	Platen	Milstein	SRIC2
2^{-5}	0.150	0.131	0.127	0.223	0.271	0.271
2^{-6}	0.082	0.072	0.070	0.121	0.145	0.145
2^{-7}	0.037	0.032	0.031	0.056	0.069	0.068
2^{-8}	0.017	0.015	0.014	0.027	0.035	0.034
2^{-9}	0.008	0.007	0.007	0.013	0.017	0.016

method SROCKC2-3 and SRIC2 are approximately similar to the method SRIC1.

In following, because we do not know exact solutions to problems, reference solutions have been simulated with the Milstein scheme [16] based on the step size $h = 2^{-13}$.

Problem 2: The second problem is a chemical reaction model [12]. The mathematical model is

$$\begin{aligned} dx_1(t) &= \Big(c_1x_1(t) - c_2x_1(t)(x_1(t)-1) + 2c_3x_2(t))\Big)dt \\ &\quad + x_1(t)\Big(\alpha_1 dW_t^1 + \alpha_2 dW_t^2\Big), \\ dx_2(t) &= \Big(\frac{c_2}{2}x_1(t)(x_1(t)-1) - c_3x_2(t) - c_4x_2(t))\Big)dt \\ &\quad + x_2(t)\Big(\beta_1 dW_t^1 + \beta_2 dW_t^2\Big). \end{aligned}$$

The system parameters take the values $c_1 = c_2 = 200$, $c_3 = 100$, and $c_4 = 100$ while the stochastic coefficients are $\alpha_1 = \alpha_2 = 5$ and $\beta_1 = \beta_2 = 0.5$.

Table 4: Ratio CPU times of the proposed methods to the running time of the method SRIC1, for problem 1 with $\lambda = 1, \mu = 1$ $X_{t_0} = 1$ at $T = 1$

h	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC2
2^0	0.842	1.736	2.157	0.115	0.105	0.473
2^{-1}	1.346	1.846	3.891	0.269	0.262	1.000
2^{-2}	0.957	2.015	3.723	0.414	0.106	1.063
2^{-3}	0.970	2.000	3.303	0.284	0.176	0.961
2^{-4}	0.984	2.035	3.533	0.329	0.203	1.011
2^{-5}	1.032	2.060	3.486	0.319	0.151	1.001
2^{-6}	0.995	1.949	3.212	0.386	0.239	1.022

Table 5: A least squares fit for the parameters C and α, for problem 2

	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC1
α	0.7247	0.7237	0.6993	1.5669	1.4207	0.8303
$\log(C)$	8.5043	8.3181	7.9808	19.5330	17.8396	10.1329

Table 6: Means of absolute errors, RMSE, for problem 2

h	SROCKC2-3	SROCKC2-3	SROCKC2-3	SRIC2	Milstein	Euler	Platen
0.01×2^{-9}	unst.	1.17	1.21	unst.	unst.	unst.	unst.
0.01×2^{-10}	0.86	0.91	0.92	unst.	unst.	unst.	unst.
0.01×2^{-11}	0.71	0.60	0.58	0.98	1.72	unst.	1.58
0.01×2^{-12}	0.41	0.34	0.33	0.55	0.51	0.51	0.41
0.01×2^{-13}	0.26	0.22	0.22	0.31	0.24	0.24	0.18

Table 7: Ratio CPU times of the proposed methods to the running time of the method SRIC1, for problem 2

h	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC2
0.01×2^{-6}	1.066	1.333	1.600	0.733	0.667	0.933
0.01×2^{-7}	0.935	1.225	1.581	0.709	0.645	0.967
0.01×2^{-8}	1.034	1.310	1.672	0.706	0.689	1.034
0.01×2^{-9}	1.008	1.288	1.661	0.711	0.678	1.016
0.01×2^{-10}	0.991	1.287	1.665	0.707	0.671	1.042
0.01×2^{-11}	1.006	1.319	1.696	0.724	0.698	1.002
0.01×2^{-12}	1.004	1.304	1.685	0.719	0.691	1.000
0.01×2^{-13}	1.007	1.314	1.684	0.730	0.699	1.009

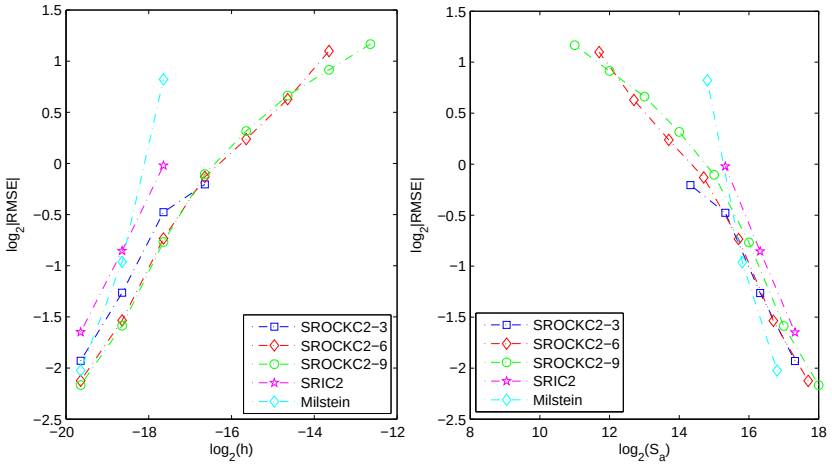

Figure 7: RMSE versus step size (left) and RMSE versus computational cost (right) with double logarithmic (ld) measures for test problem 2.

Table 8: A least squares fit for the parameters C and α, for problem 3

	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC1
α	0.5454	0.5715	0.5244	1.4623	1.3736	1.4243
$\log(C)$	0.6772	0.8068	0.5372	6.7938	6.2749	6.5723

The initial conditions are $x_1(0) = 1000$ and $x_2(0) = 100$, and the integration is performed on the interval $[0, 0.01]$. The results are indicated in Figure 7 and Tables 5–7. Table 5 shows that although all mentioned methods have same slope values, α, but the values of C, for the methods SROCKC2-3, SROCKC2-6, and SROCKC2-9 are much less that other methods. Also, from Table 6, it can be deduced that the methods SROCKC2-6 and SROCKC2-9 are MS-stable for step sizes $h \leq 7.81 \times 10^{-5}$ and $h \leq 1.56 \times 10^{-4}$. Clearly, the simulation results suggest proposed methods as the most efficient methods with respect to the root mean-square errors and computational costs. Also, the evolution of the $x_1(t)$ and $x_2(t)$ is given in Figure 8.

Problem 3: The third problem is Marine bacteriophage infection model, which is a stiff SDEs; see [11]. The mathematical model is

$$\begin{aligned} ds(t) &= \Big(as(t)(1-(i(t)+s(t))) - s(t)p(t)\Big)dt + \sigma_1\Big(s(t)-s^*\Big)dW_t^1, \\ di(t) &= \Big(s(t)p(t) - \ell i(t)\Big)dt + \sigma_2\Big(i(t)-i^*\Big)dW_t^2, \\ dp(t) &= \Big(-s(t)p(t) - mp(t) + b\ell i(t)\Big)dt - \sigma_3\Big(p(t)-p^*\Big)dW_t^3, \end{aligned} \tag{21}$$

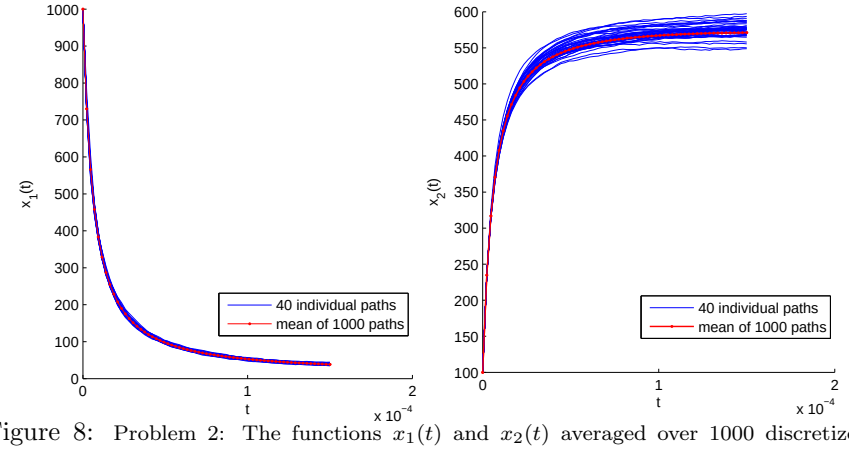

Figure 8: Problem 2: The functions $x_1(t)$ and $x_2(t)$ averaged over 1000 discretized Brownian paths and along 40 individual paths with the SROCKC2-6 method with $h = 0.01 \times 2^{-8}$.

Table 9: Means of absolute errors, RMSE, for problem 3

h	SROCKC2-3	SROCKC2-3	SROCKC2-3	SRIC1	Milstein	Euler	Platen
5×2^{-5}	unst.	0.353	0.351	unst.	unst.	unst.	unst.
5×2^{-6}	0.354	0.350	0.344	unst.	unst.	unst.	unst.
5×2^{-7}	0.312	0.311	0.309	unst.	unst.	unst.	unst.
5×2^{-8}	0.238	0.237	0.235	unst.	unst.	unst.	unst.
5×2^{-9}	0.162	0.161	0.159	unst.	unst.	unst.	unst.
5×2^{-10}	0.108	0.107	0.105	0.365	0.355	0.362	0.372
5×2^{-11}	0.074	0.072	0.073	0.136	0.137	0.133	0.135
5×2^{-12}	0.055	0.054	0.055	0.039	0.039	0.038	0.039

with the initial condition $(s_0, i_0, p_0) = (0.3, 0.2, 5)$. Here s represents the susceptible bacteria, i is the infected bacteria, and p is the phage (viruses). The drift coefficients have the values $a = 8.65$, $\ell = 24.628$, $m = 14.925$, and $b = 60$ while the noise coefficients have the values $\sigma_1 = \sigma_2 = \sigma_3 = 0.4$ and

$$s^* = \frac{m}{b-1}, i^* = \frac{as^*(1-s*)}{\ell + as^*}, p^* = \frac{a\ell(1-s^*)}{\ell + as^*}.$$

In Figure 9, we report the errors versus step size of the methods and errors versus computational effort with double logarithmic (ld) measures on the interval $[0, 5]$. Also, Figure 9 illustrates the reduction in computational effort of methods SROCKC2-3, SROCKC2-6, and SROCKC2-9 in the same level of accuracy. Clearly, the simulation results in Tables 8–10 suggest the proposed methods as the most efficient methods with respect to root mean-square errors and computational costs. The evolution of the three interacting species

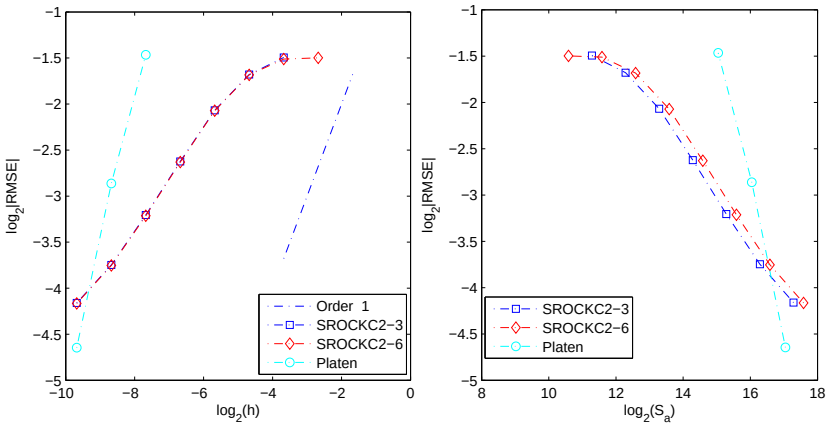

Figure 9: RMSE versus step size (left) and RMSEs versus computational cost (right) with double logarithmic (ld) measures for test problem 3.

Table 10: Ratio CPU times of the proposed methods to the running time of the method SRIC1, for problem 3

h	SROCKC2-3	SROCKC2-6	SROCKC2-9	Platen	Milstein	SRIC2
5×2^{-5}	1.200	1.600	3.000	0.400	0.401	1.200
5×2^{-6}	0.909	1.545	2.454	0.363	0.181	1.000
5×2^{-7}	1.001	1.476	2.428	0.381	0.285	1.095
5×2^{-8}	1.024	1.561	2.243	0.390	0.268	1.048
5×2^{-9}	0.964	1.511	2.178	0.392	0.272	1.023
5×2^{-10}	1.189	1.561	2.298	0.445	0.274	1.158
5×2^{-11}	0.972	1.413	2.011	0.356	0.263	1.087
5×2^{-12}	0.952	1.384	2.158	0.371	0.267	1.069

is given in Figure 10.

6 Conclusions

In this paper, the class of strong stochastic Runge–Kutta methods for stochastic differential equations with commutative noise due to Rößler (2010) is considered. For stiff SDEs, a family of explicit stochastic orthogonal Runge–Kutta Chebyshev methods of strong order one for the approximation of the solution of Itô SDEs with m-dimensional commutative noise are designed. The mean-square and asymptotic stability of newly proposed methods applied to a scalar linear test equation with multiplicative noise have been analyzed. Some numerical results for stochastic models arising in applications are provided to confirm theoretical discussions. In future works, we will con-

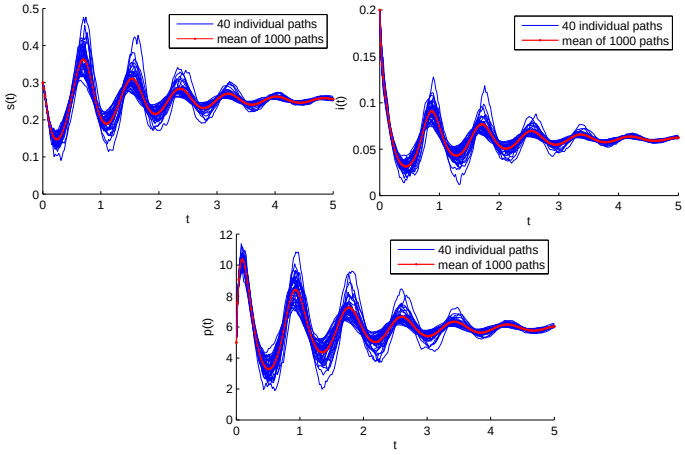

Figure 10: Problem 3: The functions $s(t)$, $i(t)$, and $p(t)$ averaged over 1000 discretized Brownian paths and along 40 individual paths with the SROCKC2-6 method with $h = 5 \times 2^{-8}$.

sider constructing methods with higher strong global convergence orders and better stability properties.

Acknowledgements

Author is grateful to there anonymous referees and editor for their constructive comments.

References

1. Abdulle, A. and Cirilli, S. *S-ROCK: Chebyshev methods for stiff stochastic differential equations*, SIAM J. Sci. Comput. 30(2) (2008), 997–1014.

2. Abdulle, A. and Li, T. *S-ROCK methods for stiff Itô SDEs*, Commun. Math. Sci. 6(4) (2008), 845–868.

3. Abdulle, A. and Medovikov, A. *Second order Chebyshev methods based on orthogonal polynomials*, Numer. Math. 90 (2001), 1–18.

4. Burrage, K., Burrage, P. and Tian, T. *Numerical methods for strong solutions of stochastic differential equations: an overview*, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 460(2041) (2004), 373–402.

5. Burrage, K. and Burrage, P.M. *General order conditions for stochastic Runge–Kutta methods for both commuting and non-commuting stochastic ordinary differential equation systems*, Appl. Numer. Math. 28 (1998), 161–177.

6. Falsone, G. *Stochastic differential calculus for Gaussian and non-Gaussian noises: A critical review*, Commum. Nonlinear. Sci. 56 (2018), 198–216.

7. Haghighi, A. and Hosseini, S.M. *A class of split-step balanced methods for stiff stochastic differential equations*, Numer. Algorithms. 61 (2012), 141–162.

8. Haghighi, A., Hosseini, S.M. and Rößler, A. *Diagonally drift-implicit Runge–Kutta methods of strong order one for stiff stochastic differential systems*, J. Comput. Appl. Math. 293 (2016), 82–93.

9. Higham, D. *Mean-square and asymptotic stability of the stochastic theta method*, SIAM J. Numer. Anal. 38(3) (2000), 753–769.

10. Holden, H., Øksendal, B., Ubøe, J., and Zhang, T. *Stochastic partial differential equations*, In Stochastic Partial Differential Equations, pp. 141–191. Birkhäuser Boston, 1996.

11. Ilie, S., Jackson, K.R. and Enright, W.H. *Adaptive time-stepping for the strong numerical solution of stochastic differential equations*, Numer Algor. 68 (2015), 791–812.

12. Kloeden, P. and Platen, E. *Numerical solution of stochastic differential equations*, Vol. 23, Springer-Verlag, Berlin, 1999.

13. Komori, Y. and Burrage, K. *A bound on the maximum strong order of stochastic Runge–Kutta methods for stochastic ordinary differential equations*, BIT 37(4) (1997), 771–780.

14. Komori, Y. and Burrage, K. *Strong first order S-ROCK methods for stochastic differential equations*, J. Comput. Appl. Math. 242 (2013), 261–274.

15. Komori, Y., Mitsui, T. and Sugiura, H. *Rooted tree analysis of the order conditions of row-type scheme for stochastic differential equations*, BIT 37(1) (1997), 43–66.

16. Milstein, G.N. and Tretyakov, M.V. *Stochastic numerics for mathematical physics*, Scientific Computing, Springer-Verlag, Berlin and New York, 2004.

17. Namjoo, M. *Strong approximation for Itô stochastic differential equations*, Iranian Journal of Numerical Analysis and Optimization 5(1) (2015), 1–12.

18. Riha, W. *Optimal stability polynomials*, Computing 9 (1972), 37–43.

19. Rößler, A. *Second order Runge–Kutta methods for Itô stochastic differential equations*, SIAM J. Numer. Anal. 47(3) (2009), 1713–1738.

20. Rößler, A. *Runge–Kutta methods for the strong approximation of solutions of stochastic differential equations*, SIAM J. Numer. Anal. 48(3) (2010), 922–952.

21. Saito, Y. and Mitsui, T. *T-stability of numerical scheme for stochastic differential equations*, World Sci. Ser. Appl. Anal. 2 (1993), 333–343.

22. Sobczyk, K. *Stochastic differential equations: with applications to physics and engineering*, Vol. 40. Springer Science and Business Media, 2013.

23. Soheili, A.R. and Soleymani, F. *Iterative methods for nonlinear systems associated with finite difference approach in stochastic differential equations*, Numer Algor 71(1) (2016), 89–102.

24. Soleymani, F. and Soheili, A.R. *A revisit of stochastic theta method with some improvements*, Filomat. 31(3) (2017), 585–596.

Iranian Journal of Numerical Analysis and Optimization
Vol. 9, No. 1, (2019), pp 127–150
DOI:10.22067/ijnao.v9i1.63182

Regularization technique and numerical analysis of the mixed system of first and second-kind Volterra–Fredholm integral equations

S. Pishbin* and J. Shokri

Abstract

It is important to note that mixed systems of first and second-kind Volterra–Fredholm integral equations are ill-posed problems, so that solving discretized system of such problems has a lot of difficulties. We will apply the regularization method to convert this mixed system (ill-posed problem) to system of the second kind Volterra–Fredholm integral equations (well-posed problem). A numerical method based on Chebyshev wavelets is suggested for solving the obtained well-posed problem, and convergence analysis of the method is discussed. For showing efficiency of the method, some test problems, for which the exact solution is known, are considered.

Keywords: Mixed systems of first and second-kind Volterra–Fredholm integral equations; Regularization method; Chebyshev wavelets; Convergence analysis.

1 Introduction

The Volterra–Fredholm integral equations [4, 5, 7, 13] arise from parabolic boundary value problems, from the mathematical modeling of the spatiotemporal development of an epidemic, mathematical population dynamics, and from various physical and biological models.

*Corresponding author
Received 8 March 2017; revised 29 August 2018; accepted 15 October 2018
S. Pishbin
Department of Mathematics, Faculty of Sciences, Urmia University, Urmia, P. O. Box 165, Iran. e-mail: s.pishbin@urmia.ac.ir

J. Shokri
Department of Mathematics, Faculty of Sciences, Urmia University, Urmia, P. O. Box 165, Iran. j.shokri@urmia.ac.ir

In this paper, we consider mixed system of Volterra–Fredholm integral equations (VFIEs) consisting of first and second-kind VFIEs as follows:

$$F(t,x) = AU(t,x) + \int_0^t \int_\Omega \mathbf{K}(t,\eta,x,s)U(\eta,s)dsd\eta, \quad t \in I = [0,T], \quad x \in \Omega, \tag{1}$$

where $F(t,x) = [f(t,x), g(t,x)]^T$, $U(t,x) = [u_1(t,x), u_2(t,x)]^T$ and

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad \mathbf{K}(t,\eta,x,s) = \begin{bmatrix} k_{11}(t,\eta,x,s) & k_{12}(t,\eta,x,s) \\ k_{21}(t,\eta,x,s) & k_{22}(t,\eta,x,s) \end{bmatrix}.$$

Here, Ω denotes a (closed) bounded region in $\mathbb{R}^d$ $(d = 1, 2, 3)$ with the (piecewise) smooth boundary $\partial\Omega$.

The reformulation of the initial-boundary-value problem for the linear heat equation in a two-dimensional spatial domain Ω with the boundary $\partial\Omega$ by single-layer techniques leads to a mixed system of Volterra–Fredholm integral equations. Mixed systems of VFIEs are considered as the ill-posed problems. However, we will first apply the method of regularization that received a considerable amount of interest, especially in solving first kind integral equations. The method transforms mixed system to the system of second kind integral equations. The method of regularization was established independently by Phillips [12] and Tikhonov [14]. The method of regularization consists of replacing ill-posed problem by well-posed problem. Some numerical methods have been proposed for solving Volterra–Fredholm integral equations of the second kind; see, for example, [2, 6, 8, 10, 11, 15, 16]. In this paper, the wavelet collocation method is developed for a mixed system of first and second kind Volterra–Fredholm integral equations.

The paper is organized as follows. In Section 2, we consider some applications of Volterra–Fredholm integral equation and in Section 3, we introduce regularization technique to transform mixed system (1) into the system of second kind integral equations. In Section 4, a numerical method based on Chebyshev wavelets is applied for solving the obtained well-posed problem. The convergence analysis is given in Section 5 and the numerical experiments are carried out in Section 6, which will be used to verify the theoretical results.

2 Some applications

We can consider Volterra–Fredholm integral equation of the first kind as a spacial case of system (1)(let $u_1 = 0, k_{12} = 0$). The reformulation of the initial-boundary-value problem for the linear heat equation in a two-dimensional spatial domain Ω with boundary $\partial\Omega$ by singlelayer techniques leads to a Volterra–Fredholm integral equation of the first kind in the follow-

ing form [3]:

$$g_\Gamma(t,\theta) = \int_0^t \int_0^1 \mathbf{K}(t-s, X(\theta) - X(s))U(s,\phi)d\phi ds,$$

where $X(\theta)$ is a smooth 1-periodic parametric representation of the boundary curve $\Gamma = \partial\Omega$, and g_Γ represents the function describing the given boundary condition on $I \times \partial\Omega$.

For the another example, you can consider the following integral equation [1]

$$f(x,t) = (\Lambda(t) - f_*(x)) = \int_0^t \int_\Omega F(t,\tau)k(x,y)\Phi(y,t)dyd\tau,$$
$$(x = \bar{x}(x_1,x_2,x_3), y = \bar{y}(y_1,y_2,y_3), (x,y) \in \Omega, t \in [0,T],)$$

under the condition

$$\int_\Omega \Phi(x,t)dx = P(t),$$

can be investigated from the contact problem of a rigid surface (G,ν) having an elastic material occupying the domain Ω where $f_*(x)$ describing the surface of stamp. This stamp impressed into an elastic layer surface (plane) by a variable known force with respect to time $P(t)$ whose eccentricity of application $e(t), (t \in [0,T])$ that case rigid displacement $\Lambda(t)$. Here G is the displacement magnitude and ν is Poisson's coefficient.

3 Regularization technique

The linear operator defined by second equation in system (1) has not generally a continuous inverse, so that it is difficult to obtain a precise numerical solution by classical discretization methods. Thus regularization techniques could be used instead to transform integral equations such as second equation in system (1) into second-kind integral equations. More precisely, we consider the following integral equations:

$$\begin{cases} f(t,x) = u_1(t,x) + \int_0^t \int_\Omega k_{11}(t,\eta,x,s)u_1(\eta,s)dsd\eta \\ \qquad\qquad + \int_0^t \int_\Omega k_{12}(t,\eta,x,s)u_{2,\varepsilon}(\eta,s)dsd\eta, \\ g(t,x) = \varepsilon u_{2,\varepsilon}(t,x) + \int_0^t \int_\Omega k_{21}(t,\eta,x,s)u_1(\eta,s)dsd\eta \\ \qquad\qquad + \int_0^t \int_\Omega k_{22}(t,\eta,x,s)u_{2,\varepsilon}(\eta,s)dsd\eta, \end{cases} \tag{2}$$

where ε is a fixed positive number.

Under a hypothesis that will be clarified later, it can be proved that the solution $u_{2,\varepsilon}(x,t)$ of equation (2) converges (when $\varepsilon \to 0$) to the solution $u_2(x,t)$ of equation (1). Now, we consider the matrix form of equation (2) as the following system of integral equations:

$$F(t,x) = EU_\varepsilon(t,x) + \int_0^t \int_\Omega \mathbf{K}(t,\eta,x,s)U_\varepsilon(\eta,s)dsd\eta, \tag{3}$$

where $E = \begin{bmatrix} 1 & 0 \\ 0 & \varepsilon \end{bmatrix}$ and $U_\varepsilon(x,t) = [u_1(x,t), u_{2,\varepsilon}(x,t)]^T$.

We need the following definition and lemma to regularization technique.

Definition 1. *A self-adjoint operator $\kappa : H \to H$, where H is a real Hilbert space, is called coercive if there exists a constant $c > 0$ such that*

$$\langle \kappa x, x\rangle \geq c\|x\|^2, \quad \textit{for all } x \in H.$$

Lemma 1. *Suppose that the integral operator of system (3) is continuous and coercive in a Hilbert space, where F, U, U_ε are defined. Then*

- *$\|U_\varepsilon\|$ is bounded independently of ε;*
- *$\|U_\varepsilon - U\|$ tends to 0 when $\varepsilon \to 0$.*

Proof. From (3) we conclude the following:

$$\|E\|\|U_\varepsilon\| = \|F - \int_0^t \int_\Omega \mathbf{K}U_\varepsilon dsd\eta\| \geqslant -\|F\| + \|\int_0^t \int_\Omega \mathbf{K}U_\varepsilon dsd\eta\|, \tag{4}$$

The coercivity property of the integral operator implies

$$\|\int_0^t \int_\Omega \mathbf{K}U_\varepsilon dsd\eta\| \geqslant \alpha\|U_\varepsilon\|, \tag{5}$$

where α is the coercivity constant.

From (4) and (5) we deduce

$$\|E\|\|U_\varepsilon\| \geqslant \alpha\|U_\varepsilon\| - \|F\|; \tag{6}$$

therefore it is obtained:

$$(\alpha - \|E\|)\|U_\varepsilon\| \leqslant \|F\|,$$

which proves the first part of the lemma.

Now, for proving the second result, by using (1) and (3), we have

$$E(U_\varepsilon - U) = -\int_0^t \int_\Omega \mathbf{K}(t,\eta,x,s)[U_\varepsilon(\eta,s) - U(\eta,s)]dsd\eta + \tilde{U}_1 - EU, \tag{7}$$

where $\tilde{U}_1(x,t) = [u_1(x,t), 0]^T$. By rearranging (7), we can write

$$\int_0^t \int_\Omega \mathbf{K}(t,\eta,x,s)[U_\varepsilon(\eta,s) - U(\eta,s)]dsd\eta = -E(U_\varepsilon - U) - \varepsilon\tilde{U}_2, \quad (8)$$

where, by using vectors operations $EU - \tilde{U}_1 = \varepsilon\tilde{U}_2$ is obtained, which $\tilde{U}_2 = [0, u_2]^T$. Taking the norm from the both side of (8) and using the coercivity property imply

$$\alpha\|U_\varepsilon - U\| \leqslant \|E(U_\varepsilon - U) - \varepsilon\tilde{U}_2\| \leqslant \|E\|\|U_\varepsilon - U\| + \varepsilon\|\tilde{U}_2\|. \quad (9)$$

From $\|\tilde{U}_2\| = \|u_2\|$, we deduce that

$$(\alpha - \|E\|)\|U_\varepsilon - U\| \leqslant \varepsilon\|u_2\|, \quad (10)$$

and now from $\|E\| = 1 + \varepsilon \to 1$ when $\varepsilon \to 0$ and $\alpha \gg 1$ then it is concluded that $\|U_\varepsilon - U\| \to 0$ when $\varepsilon \to 0$. This completes the proof. □

The conditions of existence and uniqueness of solutions related to the VFIEs (3) can be investigated by considering the theorem about existence and uniqueness of solution of the second-kind Volterra–Fredholm integral equation in [3].

Theorem 1. *Assume that*
1. $F \in C(I \times \Omega)$
2. $\mathbf{K} \in C(D \times \Omega^2)$ *where* $D = \{(t,\eta), 0 \leq \eta \leq t \leq T\}$.
Then the mixed system (3) possesses a unique solution $U_\varepsilon \in C(I \times \Omega)$.

4 Numerical treatment

4.1 Chebyshev wavelets

Dilations and translations of the "Mother function," or "analyzing wavelet" $\Phi(x)$ define an orthogonal basis, our wavelet basis:

$$\Phi_{(s,l)}(t) = 2^{\frac{-s}{2}}\Phi(2^{-s}t - l),$$

the variables s and l are integers that scale and dilate the mother function Φ to generate wavelets, such as a Daubechies wavelet family. The scale index s indicates wavelet's width, and the location index l gives its position. Notice that the mother functions are rescaled, or "dilated" by powers of two, and translated by integers. What makes wavelet bases especially interesting is the self-similarity caused by the scales and dilations. Once we know about the mother functions, we know everything about the basis. Chebyshev wavelets

$\phi_{(n,m)} = \phi(k,n,m,t)$ have four arguments, $n = 1,2,\ldots,2^{k-1}$, where k can assume any positive integer and m is the degree of Chebyshev polynomials of the first kind. They are defined on the interval $[0,1)$ as

$$\phi_{(n,m)}(t) = \begin{cases} 2^{\frac{k}{2}}\hat{T}_m(2^k t - 2n + 1), & \frac{n-1}{2^{k-1}} \le t < \frac{n}{2^{k-1}}, \\ 0, & otherwise, \end{cases} \tag{11}$$

where

$$\hat{T}_m(t) = \begin{cases} \frac{1}{\sqrt{\pi}}, & m = 0, \\ \sqrt{\frac{2}{\pi}}T_m(t), & m > 0, \end{cases}$$

and $m = 0,1,\ldots,M-1$, $n = 1,2,\ldots,2^{k-1}$. Also $T_m(t)$ are Chebyshev polynomials of degree m which are orthogonal with respect to the function $w(t) = \frac{1}{\sqrt{1-t^2}}$ on the interval $[-1,1]$. We consider the two-dimensional Chebyshev wavelet $\phi_{(n_1,m_1,n_2,m_2)}(x,y)$ as follows:

$$\begin{cases} 2^{\frac{k_1+k_2}{2}}\hat{T}_{m_1}(2^{k_1}x - 2n_1 + 1)\hat{T}_{m_2}(2^{k_2}y - 2n_2 + 1), & \frac{n_1-1}{2^{k_1-1}} \le x < \frac{n_1}{2^{k_1-1}}, \\ & \frac{n_2-1}{2^{k_2-1}} \le y < \frac{n_2}{2^{k_2-1}}, \\ 0, & otherwise, \end{cases} \tag{12}$$

where $m_1 = 0,1,\ldots,M_1-1$, $m_2 = 0,1,\ldots,M_2-1$, $n_1 = 1,\ldots,2^{k_1-1}$, and $n_2 = 1,\ldots,2^{k_2-1}$.

A function $u(x,y) \in L^2([0,1)\times[0,1))$ may be expanded as

$$u(x,y) = \sum_{n_1=1}^{\infty}\sum_{m_1=0}^{\infty}\sum_{n_2=1}^{\infty}\sum_{m_2=0}^{\infty} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1,n_2,m_2)}(x,y). \tag{13}$$

If the infinite series in equation (13) is truncated, then

$$u(x,y) \approx \sum_{n_1=1}^{2^{k_1-1}}\sum_{m_1=0}^{M_1-1}\sum_{n_2=1}^{2^{k_2-1}}\sum_{m_2=0}^{M_2-1} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1,n_2,m_2)}(x,y) = \Phi(x)^T U \Phi(y). \tag{14}$$

where $\Phi(x)$ and $\Phi(y)$ are $(2^{k_1-1})M_1 \times 1$ and $(2^{k_2-1})M_2 \times 1$ matrices, respectively, such that

$$\Phi(x) = [\phi_{(1,0)}(x),\ldots,\phi_{(1,M_1-1)}(x),\ldots,\phi_{(2^{k_1-1},0)}(x),\ldots,\phi_{(2^{k_1-1},M_1-1)}(x)]^T,$$

$$\Phi(y) = [\phi_{(1,0)}(y),\ldots,\phi_{(1,M_2-1)}(y),\ldots,\phi_{(2^{k_2-1},0)}(y),\ldots,\phi_{(2^{k_2-1},M_2-1)}(y)]^T.$$

Also U is a $(2^{k_1-1})M_1 \times (2^{k_2-1})M_2$ matrix whose elements can be calculated as

$$u_{n_1m_1n_2m_2} = \int_0^1\int_0^1 \phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)u(x,y)w_{n1}(x)w_{n2}(y)dydx,$$

where $n_1 = 1, \ldots, 2^{k_1-1}$, $m_1 = 0, 1, \ldots, M_1 - 1$, $n_2 = 1, \ldots, 2^{k_2-1}$, $m_2 = 0, 1, \ldots, M_2 - 1$ and

$$w_n(t) = \begin{cases} w(2^k t - 2n + 1), & \frac{n-1}{2^{k-1}} \leq t < \frac{n}{2^{k-1}}, \\ 0, & otherwise \end{cases}$$

Now, consider system (2) with $I = \Omega = [0, 1]$ and approximate the solution of this system using (13) as

$$\hat{u}_1(t, x) = \Phi(t)^T U_1 \Phi(x). \tag{15}$$

$$\hat{u}_{2,\varepsilon}(t, x) = \Phi(t)^T U_{2,\varepsilon} \Phi(x). \tag{16}$$

Also, we approximate the kernels in system (2) respect to two variables η and s as follows:

$$k_{ij}(t, \eta, x, s) \approx \hat{k}_{ij}(t, \eta, x, s) = \Phi(\eta)^T K_{ij} \Phi(s) \quad (i, j = 1, 2). \tag{17}$$

Inserting (15), (16), (17), and the following collocation points

$$\begin{aligned} t_i &= \frac{i}{M_1 2^{k_1-1} + 1}, \quad i = 1, 2, \ldots, M_1 2^{k_1-1}, \\ x_j &= \frac{j}{M_1 2^{k_2-1} + 1}, \quad j = 1, 2, \ldots, M_2 2^{k_2-1}, \end{aligned} \tag{18}$$

into system (2), we have the following linear system of algebraic equations for the unknown coefficients U_1 and $U_{2,\varepsilon}$:

$$\begin{cases} f(t_i, x_j) = \Phi(t_i)^T U_1 \Phi(x_j) + \int_0^{t_i} \int_0^1 \hat{k}_{11}(t_i, \eta, x_j, s) \Phi(\eta)^T U_1 \Phi(s) ds d\eta \\ \qquad + \int_0^{t_i} \int_0^1 \hat{k}_{12}(t_i, \eta, x_j, s) \Phi(\eta)^T U_{2,\varepsilon} \Phi(s) ds d\eta, \\ g(t_i, x_j) = \varepsilon \Phi(t_i)^T U_{2,\varepsilon} \Phi(x_j) + \int_0^{t_i} \int_0^1 \hat{k}_{21}(t_i, \eta, x_j, s) \Phi(\eta)^T U_1 \Phi(s) ds d\eta \\ \qquad + \int_0^{t_i} \int_0^1 \hat{k}_{22}(t_i, \eta, x_j, s) \Phi(\eta)^T U_{2,\varepsilon} \Phi(s) ds d\eta, \\ \qquad i = 1, 2, \ldots, M_1 2^{k_1-1}, \quad j = 1, 2, \ldots, M_2 2^{k_2-1}. \end{cases} \tag{19}$$

4.2 Normalization

In the language of optimization theory, for the linear bounded operator $K : X \to Y$ and $Kx = y$, determine x_ε that minimizes the Tikhonov functional

$$J_\varepsilon(x) = \|Kx - y\|^2 + \varepsilon\|x\|^2, \quad \text{for all } x \in X.$$

We can consider the following theorem from [9].

Theorem 2. *Let $K : X \to Y$ be a linear bounded operator between Hilbert spaces and let $\epsilon > 0$. Then the Thikhonov functional J_ε has a unique minimum $x_\varepsilon \in X$. This minimum x_ε is the unique solution of the normal equation*

$$\varepsilon x_\varepsilon + K^*Kx_\varepsilon = K^*y.$$

Here, $K^* : Y \to X$ denotes the adjoint of K.

By using Theorem 2, system (2) with $I = \Omega = [0, 1]$ can be written in the normal form as follows:

$$\begin{cases} f(p,q) = u_1(p,q) + \displaystyle\int_0^p \int_0^1 k_{11}(p,\eta,q,s)u_1(\eta,s)dsd\eta \\ \qquad\qquad + \displaystyle\int_0^p \int_0^1 k_{12}(p,\eta,q,s)u_{2,\varepsilon}(\eta,s)dsd\eta, \\ \displaystyle\int_p^1 \int_0^1 k_{22}(p,t,q,x)g(t,x)dxdt \\ \qquad = \varepsilon u_{2,\varepsilon}(p,q) \\ \qquad\quad + \displaystyle\int_p^1 \int_0^1 \int_0^t \int_0^1 k_{22}(p,t,q,x)k_{21}(t,\eta,x,s)u_1(\eta,s)dsd\eta dxdt \\ \qquad\quad + \displaystyle\int_p^1 \int_0^1 \int_0^t \int_0^1 k_{22}(p,t,q,x)k_{22}(t,\eta,x,s)u_{2,\varepsilon}(\eta,s)dsd\eta dxdt, \end{cases} \tag{20}$$

Now, we can consider the numerical method based on Chebyshev wavelets from the previous subsection for the approximate solution of system (20).

5 Convergence analysis

In this section, we investigate the convergence analysis of the proposed Chebyshev wavelet collocation method, using polynomial approximation theory.

Lemma 2. *Assume that $u(x, y) \in L^2([0, 1) \times [0, 1))$ can be expanded in the form of series (13) and that $\hat{u}(t, x)$ is the approximation of $u(x, y)$ which is defined by (14). Then $\hat{u}(t, x)$ converges to $u(t, x)$.*

Proof. We recall the series $u(t, x)$ from (13) and the truncated series $\hat{u}(t, x)$ from (14), respectively, as follows:

$$u(x,y) = \sum_{n_1=1}^{\infty} \sum_{m_1=0}^{\infty} \sum_{n_2=1}^{\infty} \sum_{m_2=0}^{\infty} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1,n_2,m_2)}(x,y),$$

and

$$\hat{u}(t,x) = \sum_{n_1=1}^{2^{k_1-1}} \sum_{m_1=0}^{M_1-1} \sum_{n_2=1}^{2^{k_2-1}} \sum_{m_2=0}^{M_2-1} u_{n_1 m_1 n_2 m_2} \phi_{(n_1,m_1,n_2,m_2)}(t,x),$$

where $m_1 = 0, 1, \ldots, M_1 - 1,\ m_2 = 0, 1, \ldots, M_2 - 1,\ n_1 = 1, \ldots, 2^{k_1-1}$, and $n_2 = 1, \ldots, 2^{k_2-1}$. The functions $\phi_{(n_1,m_1,n_2,m_2)}(t,x)$ are the Chebyshev wavelet. Let $L^2([0,1) \times [0,1))$ be the Hilbert space and let

$$\phi_{(n_1,m_1,n_2,m_2)}(t,x) = 2^{\frac{-(n_1+n_2)}{2}} \phi(2^{\frac{-n_1}{2}} t - m_1)\phi(2^{\frac{-n_2}{2}} x - m_2),$$

where $\phi_{(n_1,m_1,n_2,m_2)}(t,x)$ form a basis of $L^2([0,1) \times [0,1))$ as in (12). From (13), we consider

$$u(t,x) = \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} u_{1m_1 1m_2} \phi_{(1,m_1,1,m_2)}(t,x) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} c_{ij} \phi_{(i,j)}(t,x),$$

where $c_{ij} =< u(t,x), \phi_{(i,j)}(t,x) >$ for $k_1 = 1, k_2 = 1$ and $< .,. >$ represents an inner product. Let us denote $\phi_{(i,j)}(t,x) = \phi(t,x)$ and $\alpha_{ij} =< u(t,x), \phi(t,x) >$.

Define the sequence of partial sums $S_{n,m}$, $n > m$ of $\{\alpha_{ij}\phi(t_i, x_j)\}$. Let S_{n_1,m_1} and S_{n_2,m_2} be arbitrary partial sums with $n_1 > n_2$. We show that $S_{n,m}$ is a Cauchy sequence in a Hilbert space.

Let $S_{n,m} = \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} \alpha_{ij}\phi(t_i, x_j)$. From

$$< u(t,x), \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} \alpha_{ij}\phi(t_i, x_j) >= \sum_{i=0}^{n_1} \sum_{j=0}^{n_2} |\alpha_{ij}|^2,$$

we can show that $\|S_{n_1,n_2} - S_{m_1,m_2}\|^2 = \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} |\alpha_{ij}|^2$ for $n_1 > m_1,\ n_2 > m_2$.

We can write

$$\begin{aligned} &\| \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} \alpha_{ij}\phi(t_i, x_j)\|^2 \\ &\quad =< \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} \alpha_{ij}\phi(t_i, x_j), \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} \alpha_{ij}\phi(t_i, x_j) > \\ &\quad = \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} |\alpha_{ij}|^2, \end{aligned}$$

for $n_1 > m_1, n_2 > m_2$. Then it is concluded that

$$\| \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} \alpha_{ij}\phi(t_i,x_j)\|^2 = \sum_{i=m_1+1}^{n_1} \sum_{j=m_2+1}^{n_2} |\alpha_{ij}|^2,$$

for $n_1 > m_1, n_2 > m_2$.
Now from Bessel's inequality, we deduce that $\sum_{i=m_1+1}^{n_1}\sum_{j=m_2+1}^{n_2}|\alpha_{ij}|^2$ is convergent and therefore $\|\sum_{i=m_1+1}^{n_1}\sum_{j=m_2+1}^{n_2}\alpha_{ij}\phi(t_i,x_j)\|^2 \to 0$,
that is, $\|\sum_{i=m_1+1}^{n_1}\sum_{j=m_2+1}^{n_2}\alpha_{ij}\phi(t_i,x_j)\| \to 0$ and $\{S_{n,m}\}$ is a Cauchy sequence thus it converges to a real number like 's'.
In the continuance, we prove that $u(t,x) = s$.

$$\begin{aligned} < s-u(t,x),\phi(t_i,x_i) > &=< s,\phi(t_i,x_i) > - < u(t,x),\phi(t_i,x_i) > \\ &=< \lim_{n\to\infty} S_{n,m},\phi(t_i,x_i) > -\alpha_{ij} \\ &= \lim_{n\to\infty} < S_{n,m},\phi(t_i,x_i) > -\alpha_{ij} = \alpha_{ij}-\alpha_{ij} = 0; \end{aligned}$$

then it is concluded that $u(t,x) = s$ or $\sum_{i=0}^{n_1}\sum_{j=0}^{n_2}\alpha_{ij}\phi(t_i,x_j)$ converges to $u(t,x)$, and by induction it is evident that this result is established for any integer numbers of k_1 and k_2, so for $k_1 \to \infty$ and $k_2 \to \infty$. In the other hand, we can consider $\hat{u}(t,x)$ as $S_{n,m}$, which means $\hat{u}(t,x)$ converges to $u(t,x)$ and this completes the proof. □

Theorem 3. *Consider a function $u(x,y) \in L^2([0,1]\times[0,1])$, with bounded forth partial derivations, $|\frac{\partial^4 u(x,y)}{\partial^2 x \partial^2 y}| < B$. Then the wavelet coefficient, $u_{n_1m_1n_2m_2}$ in (14) , decay as follows:*

$$|u_{n_1m_1n_2m_2}| \le \frac{\pi B}{2^4 (n_1n_2)^{5/2}(m_1^2-1)(m_2^2-1)}. \tag{21}$$

Proof. From (14) it follows that

$$\begin{aligned} u_{n_1m_1n_2m_2} &= \int_0^1\int_0^1 u(x,y)\phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)w_{n_1}(x)w_{n_2}(y)dydx \\ &= \int_0^1 \phi_{(n_1,m_1)}(x)w_{n_1}(x)\Big(\int_0^1 u(x,y)\phi_{(n_2,m_2)}(y)w_{n_2}(y)dy\Big)dx. \\ &= \int_0^1 \phi_{(n_1,m_1)}(x)w_{n_1}(x)\Big(\int_{(n_2-1)/2^{k_2-1}}^{n_2/2^{k_2-1}} 2^{k_2/2}u(x,y) \\ &\qquad \hat{T}_{m_2}(2^{k_2}y-2n_2+1) \\ &\qquad w_{n_2}(2^{k_2}y-2n_2+1)\,dy\Big)dx. \end{aligned} \tag{22}$$

If $m_1, m_2 > 1$, by substituting $2^{k_2}y - 2n_2 + 1 = \cos\theta$ in (22), it yields

$$
\begin{aligned}
u_{n_1m_1n_2m_2} &= \frac{1}{2^{k_2/2}}\int_0^1 \phi_{(n_1,m_1)}(x)w_{n_1}(x)\Big[\int_0^\pi u(x,\frac{\cos\theta_2 n_2-1}{2^{k_2}})\sqrt{\frac{2}{\pi}}\cos m_2\theta\,d\theta\Big]\\
&= \frac{\sqrt{2}}{2^{k_2/2}\sqrt{\pi}}\int_0^1 \phi_{(n_1,m_1)}(x)w_{n_1}(x)\Big[u(x,\frac{\cos\theta+2n_2-1}{2^{k_2}})(\frac{\sin m_2\theta}{m_2})\Big|_0^\pi\\
&\quad+\frac{\sqrt{2}}{2^{3k_2/2}m_2\sqrt{\pi}}\int_0^\pi \frac{\partial u(x,(\cos\theta_2 n_2-1)/2^{k_2})}{\partial y}\sin m_2\theta\sin\theta\,d\theta\Big]\,dx\\
&= \frac{1}{2^{3k_2/2}m_2\sqrt{2\pi}}\int_0^1 \phi_{(n_1,m_1)}(x)w_{k_1}(x)\Big[\int_0^\pi \frac{\partial u(x,(\cos\theta_2 n_2-1)/2^{k_2})}{\partial y}\\
&\qquad\qquad\Big(\frac{\sin(m_2-1)\theta}{m_2-1}-\frac{\sin(m_2+1)\theta}{m_2+1}\Big)\Big|_0^\pi\\
&\quad+\frac{1}{2^{5k_2/2}m_2\sqrt{2\pi}}\int_0^\pi \frac{\partial^2 u(x,(\cos\theta+2n_2-1)/2^{k_2})}{\partial^2 y}h_{m_2}(\theta)\,d\theta\Big]\,dx,
\end{aligned}
\tag{23}
$$

where

$$
h_m(\theta)=\sin\theta\Big(\frac{\sin(m_2-1)\theta}{m_2-1}-\frac{\sin(m_2+1)\theta}{m_2+1}\Big).
$$

Then, we obtain

$$
\begin{aligned}
u_{n_1m_1n_2m_2} = \frac{1}{2^{5k_2/2}m_2\sqrt{2\pi}}\int_0^\pi \Big[\int_0^1 \frac{\partial^2 u(x,(\cos\theta+2n_2-1)/2^{k_2})}{\partial^2 y}\\
\phi_{(n_1,m_1)}(x)w_{k_1}(x)\,dx\Big]h_{m_2}(\theta)\,d\theta.
\end{aligned}
\tag{24}
$$

Now, similar to the discussion in (23), by substituting $2^{k_1}y-2n_1+1=\cos\alpha$ in (24), it yields

$$
\begin{aligned}
&u_{n_1m_1n_2m_2}\\
&\quad=\frac{1}{2^{5(k_1+k_2)/2}m_1m_2(2\pi)}\\
&\qquad\times\int_0^\pi\int_0^\pi \frac{\partial^4 u((\cos\alpha+2n_1-1)/2^{k_1},(\cos\theta+2n_2-1)/2^{k_2})}{\partial^2 x\partial^2 y}h_{m_1}(\alpha)h_{m_2}(\theta)\,d\alpha\,d\theta.
\end{aligned}
$$

Thus we have

$$
\begin{aligned}
&|u_{n_1m_1n_2m_2}|\\
&\quad=\frac{1}{2^{5(k_1+k_2)/2}m_1m_2(2\pi)}\\
&\qquad\Big|\int_0^\pi\int_0^\pi \frac{\partial^4 u((\cos\alpha+2n_1-1)/2^{k_1},(\cos\theta+2n_2-1)/2^{k_2})}{\partial^2 x\partial^2 y}h_{m_1}(\alpha)h_{m_2}(\theta)\,d\alpha\,d\theta\Big|\\
&\quad\le\frac{B}{2^{5(k_1+k_2)/2}m_1m_2(2\pi)}\int_0^\pi |h_{m_1}(\alpha)|\,d\alpha\int_0^\pi |h_{m_2}(\theta)|\,d\theta.
\end{aligned}
\tag{25}
$$

However

$$
\begin{aligned}
\int_0^\pi |h_{m_2}(\theta)|\,d\theta &= \int_0^\pi \Big|\sin\theta\Big(\frac{\sin(m_2-1)\theta}{m_2-1}-\frac{\sin(m_2+1)\theta}{m_2+1}\Big)\Big|d\theta\\
&\le \int_0^\pi \Big|\frac{\sin\theta\sin(m_2-1)\theta}{m_2-1}\Big|+\Big|\frac{\sin\theta\sin(m_2+1)\theta}{m_2+1}\Big|\\
&\le \frac{2m_2\pi}{(m_2^2-1)},
\end{aligned}
\tag{26}
$$

and similarly, it is obtained

$$\int_0^{\pi} |h_{m_1}(\alpha)|\, d\alpha \le \frac{2m_1\pi}{(m_1^2-1)}. \tag{27}$$

Since $n_1 \le 2^{k_1-1}$ and $n_2 \le 2^{k_2-1}$, by substituting (26) and (27) in (25), the desired result is obtained as follows:

$$|u_{n_1m_1n_2m_2}| \le \frac{2\pi B}{2^{5(k_1+k_2)/2}(m_1^2-1)(m_2^2-1)}.$$

□

Theorem 4. *Let $u(x,y) \in L^2([0,1]\times[0,1])$, with bounded forth partial derivations, $|\frac{\partial^4 u(x,y)}{\partial^2 x \partial^2 y}| < B$; then the error bound would be obtained as follows:*

$$\sigma_{k_1,M_1,k_2,M_2} = O\Big(2^{-\frac{5}{2}(k_1+k_2)}\Big), \tag{28}$$

where

$$\begin{aligned}
&\sigma_{k_1,M_1,k_2,M_2}\\
&= \Big(\int_0^1\int_0^1 \Big[u(x,y) - \sum_{n_1=1}^{2^{k_1-1}}\sum_{m_1=0}^{M_1-1}\sum_{n_2=1}^{2^{k_2-1}}\sum_{M_2=0}^{M_2-1} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)\Big]^2\\
&\qquad w_{n_1}(x)w_{n_2}(y)\,dx\,dy\Big)^{\frac{1}{2}}.
\end{aligned}$$

Proof. From the error statement, we have

$$\begin{aligned}
&\sigma^2_{k_1,M_1,k_2,M_2}\\
&= \int_0^1\int_0^1 \Big[u(x,y) - \sum_{n_1=1}^{2^{k_1-1}}\sum_{m_1=0}^{M_1-1}\sum_{n_2=1}^{2^{k_2-1}}\sum_{M_2=0}^{M_2-1} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)\Big]^2\\
&\qquad\qquad w_{n_1}(x)w_{n_2}(y)\,dx\,dy\\
&= \int_0^1\int_0^1 \Big[\sum_{n_1=1}^{\infty}\sum_{m_1=0}^{\infty}\sum_{n_2=1}^{\infty}\sum_{M_2=0}^{\infty} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)\\
&\quad - \sum_{n_1=1}^{2^{k_1-1}}\sum_{m_1=0}^{M_1-1}\sum_{n_2=1}^{2^{k_2-1}}\sum_{m_2=0}^{M_2-1} u_{n_1m_1n_2m_2}\phi_{(n_1,m_1)}(x)\phi_{(n_2,m_2)}(y)\Big]^2 w_{n_1}(x)w_{n_2}(y)\,dx\,dy\\
&= \int_0^1\int_0^1 \sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty} u^2_{n_1m_1n_2m_2}(\phi_{(n_1,m_1)}(x))^2(\phi_{(n_2,m_2)}(y))^2\\
&\qquad\qquad w_{n_1}(x)w_{n_2}(y)\,dx\,dy\\
&= \sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty} u^2_{n_1m_1n_2m_2}\int_0^1\int_0^1 (\phi_{(n_1,m_1)}(x))^2(\phi_{(n_2,m_2)}(y))^2
\end{aligned}$$

$$w_{n_1}(x)w_{n_2}(y)\,dx\,dy$$

$$= \sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty} u^2_{n_1m_1n_2m_2}\int_{\frac{n_1-1}{2^{k_1-1}}}^{\frac{n_1}{2^{k_1-1}}}\frac{\left[2^{\frac{k_1}{2}}\hat{T}_{m_1}(2^{k_1}x-2n_1+1)\right]^2}{\sqrt{1-(2^{k_1}x-2n_1+1)^2}}\,dx$$

$$\times\int_{\frac{n_2-1}{2^{k_2-1}}}^{\frac{n_2}{2^{k_2-1}}}\frac{\left[2^{\frac{k_2}{2}}\hat{T}_{m_2}(2^{k_2}y-2n_2+1)\right]^2}{\sqrt{1-(2^{k_2}x-2n_2+1)^2}}\,dy$$

$$= \sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty} u^2_{n_1m_1n_2m_2}2^{k_1+k_2}\int_{\frac{n_1-1}{2^{k_1-1}}}^{\frac{n_1}{2^{k_1-1}}}\frac{\left[\hat{T}_{m_1}(2^{k_1}x-2n_1+1)\right]^2}{\sqrt{1-(2^{k_1}x-2n_1+1)^2}}\,dx$$

$$\times\int_{\frac{n_2-1}{2^{k_2-1}}}^{\frac{n_2}{2^{k_2-1}}}\frac{\left[\hat{T}_{m_2}(2^{k_2}y-2n_2+1)\right]^2}{\sqrt{1-(2^{k_2}x-2n_2+1)^2}}\,dy.$$

Now, let $2^{k_1}y-2n_1+1=t_1$ and $2^{k_2}y-2n_2+1=t_2$, then it is obtained

$$sigma^2_{k_1,M_1,k_2,M_2} = \sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty} u^2_{n_1m_1n_2m_2}\int_{-1}^{1}\frac{\hat{T}_{m_1}(t_1)}{\sqrt{1-t_1^2}}\,dt_1$$

$$\int_{-1}^{1}\frac{\hat{T}_{m_2}(t_2)}{\sqrt{1-t_2^2}}\,dt_2. \tag{29}$$

For $m_1\geq 1$, $m_2\geq 1$, we have

$$\int_{-1}^{1}\frac{\hat{T}_{m_1}(t_1)}{\sqrt{1-t_1^2}}\,dt_1=\frac{\pi}{2},\qquad \int_{-1}^{1}\frac{\hat{T}_{m_2}(t_2)}{\sqrt{1-t_2^2}}\,dt_2=\frac{\pi}{2};$$

then (29) simplifies as follows:

$$\sigma^2_{k_1,M_1,k_2,M_2}=\frac{\pi^2}{4}\sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty}u^2_{n_1m_1n_2m_2}. \tag{30}$$

Therefore from (21) and (30), we can conclude the desired result as follows:

$$\sigma^2_{k_1,M_1,k_2,M_2}\leq\frac{\pi^4B}{2^{10}}\sum_{n_1=2^{k_1}}^{\infty}\sum_{m_1=M_1}^{\infty}\sum_{n_2=2^{k_2}}^{\infty}\sum_{m_2=M_2}^{\infty}\frac{1}{(n_1n_2)^5(m_1^2-1)^2(m_2^2-1)^2}. \tag{31}$$

Also

$$\sum_{n_1=2^{k_1}}^{\infty}\frac{1}{(n_1)^5}=\frac{1}{(2^{k_1})^5}+\frac{1}{(2^{k_1}+1)^5}\cdots=\frac{1}{(2^{k_1})^5}\sum_{n=0}^{\infty}\frac{1}{(1+\frac{n}{2^{k_1}})^5}. \tag{32}$$

From (31) and (32), we conclude that

$$\sigma_{k_1,M_1,k_2,M_2} = O\Big(2^{-\frac{5}{2}(k_1+k_2)}\Big).$$

□

Theorem 5. *Let us consider* $M = \max \|\mathbf{K}(t,\eta,x,s)\|$, *for* $\eta, t \in I = [0,1], s, x \in \Omega = [0,1]$. *Assume that* $U_\varepsilon(t,x)$ *is the exact solution of system (3) and that* $\hat{U}_\varepsilon(t,x)$ *denotes the Chebyshev wavelet approximation for the exact solution* U *which is given by (15) and (16) . Then* $\hat{U}_\varepsilon(t,x)$ *converges to* $U_\varepsilon(t,x)$ *and the following result can be obtained*

$$\|U_\varepsilon - \hat{U}_\varepsilon\| = O\Big(2^{-\frac{5}{2}(k_1+k_2)}\Big).$$

Proof. According to the proposed method in previous section, we consider (15), (16), and (17) and insert $\hat{U}_\varepsilon(t,x)$ and $\hat{\mathbf{K}}(t,\eta,x,s)$ as approximations of the exact solution U and kernel $\mathbf{K}$ into system (3)

$$F(t,x) = EU_\varepsilon(t,x) + \int_0^t \int_0^1 \hat{\mathbf{K}}(t,\eta,x,s)U_\varepsilon(\eta,s)dsd\eta. \tag{33}$$

Subtracting (3) from (33) and some manipulations, we get

$$\begin{aligned}\|E\|\|U_\varepsilon - \hat{U}_\varepsilon\| \leqslant \| \int_0^t \int_0^1 (\mathbf{K}(t,\eta,x,s) - \hat{\mathbf{K}}(t,\eta,x,s))U_\varepsilon(\eta,s)dsd\eta\| \\ + \| \int_0^t \int_0^1 \hat{\mathbf{K}}(t,\eta,x,s)(U_\varepsilon(\eta,s) - \hat{U}_\varepsilon(\eta,s))dsd\eta\|.\end{aligned}$$

From $\|E\| \geq 1$ it is obtained

$$\|U_\varepsilon - \hat{U}_\varepsilon\| \leqslant \|\mathbf{K} - \hat{\mathbf{K}}\|\|U_\varepsilon\| + M\|U_\varepsilon - \hat{U}_\varepsilon\|. \tag{34}$$

Relation (34) together with Lemma 2, shows the convergence of the exact solution to the approximate solution. Considering Theorem 4, we have

$$\|\mathbf{K} - \hat{\mathbf{K}}\| = O\Big(2^{-\frac{5}{2}(k_1+k_2)}\Big). \tag{35}$$

From (34) and (35), we conclude that

$$\|U_\varepsilon - \hat{U}_\varepsilon\| = O\Big(2^{-\frac{5}{2}(k_1+k_2)}\Big).$$

□

6 Numerical examples

To demonstrate the efficiency and the practicability of the proposed method, we consider the following two examples. All results are computed by using a program written in the Mathematica®.

Example 1. Consider the following TIAEs:

$$\int_0^t \int_0^1 100t(t+x)u(\eta,s)dsd\eta = \frac{25}{3}t^2(3+4t)(t+x), \qquad t \in [0,1], \tag{36}$$

where the exact solution is $u(t,x) = t + x$.

Considering regularization techniques and (3) for (36), we have

$$\varepsilon u_\varepsilon(t,x) + \int_0^t \int_0^1 100t(t+x)u(\eta,s)dsd\eta = \frac{25}{3}t^2(3+4t)(t+x). \tag{37}$$

We assume that $\hat{u}_\varepsilon(t,x)$ is the approximation of the exact solution $u_\varepsilon(t,x)$ which is defined by (15). For analyzing the behavior of the error representations, we consider absolute error as

$$Error = |u(t_i,x_j) - \hat{u}_\varepsilon(t_i,x_j)|.$$

The Chebyshev wavelet method described in Section 4 has been implemented for problem (37) with $M_1 = M_2 = 3, k_1 = k_2 = 2$ and the error for different values of ε has been reported in Table 1, which confirms the theoretical results of Lemma 1. Figure 1 presents the plots of exact and approximate solution with $\varepsilon = 0.000001$, which are found to be in good agreement.

From subsection 4.2, we consider the normal form of equation (36) as follows:

$$\begin{aligned} \varepsilon u_\varepsilon(p,q) + \int_p^1 \int_0^1 \int_0^t \int_0^1 10000pt(p+q)(t+x)u_\varepsilon(\eta,s)dsd\eta dxdt \\ = \int_p^1 \int_0^1 \frac{2500}{3}p(p+q)t^2(3+4t)(t+x)dxdt, \qquad p \in [0,1], \end{aligned} \tag{38}$$

and then solve this equation by the Chebyshev wavelet method with $M_1 = M_2 = 3$ and $k_1 = k_2 = 2$. We report the error for different values of ε in Table 2. From numerical results in Tables 1 and 2, we observe that the results in Table 2 are more accurate than the reported results in Table 1.

Example 2. Consider the following:

$$Au(t,x) + \int_0^t \int_0^1 K(t,\eta,x,s)u(\eta,s)dsd\eta = f(t,x), \qquad t \in [0,1], \tag{39}$$

Table 1: Absolute errors of u_ε for different values of ε in Example 1

(t_i, x_j)	Error $\varepsilon = 0.01$	Error $\varepsilon = 0.001$	Error $\varepsilon = 0.000001$
$(0.1, 0.1)$	3.62×10^{-3}	3.64×10^{-4}	3.65×10^{-7}
$(0.2, 0.2)$	1.62×10^{-4}	1.45×10^{-5}	1.40×10^{-8}
$(0.3, 0.3)$	1.37×10^{-3}	1.39×10^{-4}	1.39×10^{-7}
$(0.4, 0.4)$	1.32×10^{-3}	1.35×10^{-4}	1.37×10^{-7}
$(0.5, 0.5)$	9.09×10^{-3}	9.29×10^{-4}	9.32×10^{-7}
$(0.6, 0.6)$	1.71×10^{-3}	1.74×10^{-4}	1.72×10^{-7}
$(0.7, 0.7)$	1.69×10^{-3}	1.73×10^{-4}	1.70×10^{-7}
$(0.8, 0.8)$	4.19×10^{-5}	9.27×10^{-7}	1.87×10^{-9}
$(0.9, 0.9)$	7.42×10^{-3}	8.20×10^{-4}	8.20×10^{-7}

Table 2: Absolute errors of u_ε for different values of ε for normal equation in Example 1

(t_i, x_j)	Error $\varepsilon = 0.01$	Error $\varepsilon = 0.001$	Error $\varepsilon = 0.000001$
$(0.1, 0.1)$	6.18×10^{-6}	6.11×10^{-6}	1.95×10^{-7}
$(0.2, 0.2)$	6.84×10^{-6}	6.26×10^{-6}	5.22×10^{-8}
$(0.3, 0.3)$	3.79×10^{-6}	3.83×10^{-7}	8.93×10^{-8}
$(0.4, 0.4)$	2.43×10^{-5}	2.43×10^{-6}	5.70×10^{-8}
$(0.5, 0.5)$	2.06×10^{-5}	2.06×10^{-6}	6.66×10^{-7}
$(0.6, 0.6)$	2.14×10^{-5}	2.14×10^{-6}	2.42×10^{-7}
$(0.7, 0.7)$	7.78×10^{-6}	7.78×10^{-7}	1.67×10^{-7}
$(0.8, 0.8)$	2.25×10^{-5}	2.25×10^{-6}	2.41×10^{-8}
$(0.9, 0.9)$	7.18×10^{-5}	7.18×10^{-6}	3.13×10^{-7}

where

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \qquad K(t,\eta,x,s) = \begin{bmatrix} \eta^2(s^2+t^2+1) & (s+t) \\ \eta s^2+\eta+\eta t^2 & 200s+200t \end{bmatrix},$$

$$u(t,x) = \big(u_1(t,x), u_2(t,x)\big)^T, \qquad f(t,x) = \big(f_1(t,x), f_2(t,x)\big)^T,$$

and f_1 and f_2 are such that the exact solution is

$$u_1(t,x) = t^2 + x^2 + 1, \qquad u_2(t,x) = t + x.$$

We apply the regularization method to convert the mixed system (39) to the system of the second kind integral equations. Then, the resulting second kind integral equation will be solved by the proposed numerical scheme in section 4 with $M_1 = M_2 = 3$ and $k_1 = k_2 = 2$. Let $(\hat{u}_1, \hat{u}_{2,\varepsilon})$ be the approximation of the exact solution $(u_1, u_{2,\varepsilon})$ which are defined by (15) and (16). Numerical errors with several values of ε are displayed in Tables 3 and 4.

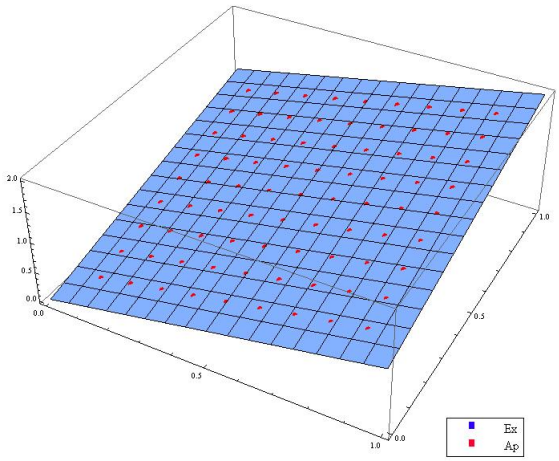

Figure 1: The plots of exact solution u and approximate solution of u with $\varepsilon = 0.000001$ in Example 1.

Similar to Example 2, we transform system (39) into the normal form of system (20) and then solve the normal system by Chebyshev wavelet method with $M_1 = M_2 = 3$ and $k_1 = k_2 = 2$. The error for different values of ε is

Table 3: Absolute errors of u_1 for different values of ε in Example 2

	Error	Error	Error
(x_i, y_j)	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.000001$
$(0.1, 0.1)$	1.00×10^{-5}	1.00×10^{-6}	1.00×10^{-9}
$(0.2, 0.2)$	2.00×10^{-5}	2.00×10^{-6}	2.00×10^{-9}
$(0.3, 0.3)$	3.03×10^{-5}	3.03×10^{-6}	3.03×10^{-9}
$(0.4, 0.4)$	4.11×10^{-5}	4.11×10^{-6}	4.11×10^{-9}
$(0.5, 0.5)$	5.44×10^{-5}	5.44×10^{-6}	5.44×10^{-9}
$(0.6, 0.6)$	6.58×10^{-5}	6.58×10^{-6}	6.58×10^{-9}
$(0.7, 0.7)$	8.25×10^{-5}	8.25×10^{-6}	8.25×10^{-9}
$(0.8, 0.8)$	1.05×10^{-4}	1.05×10^{-5}	1.05×10^{-8}
$(0.9, 0.9)$	1.34×10^{-4}	1.34×10^{-5}	1.34×10^{-8}

Table 4: Absolute errors of $u_{2,\varepsilon}$ for different values of ε in Example 2

	Error	Error	Error
(x_i, y_j)	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.000001$
$(0.1, 0.1)$	1.07×10^{-3}	1.83×10^{-4}	8.51×10^{-6}
$(0.2, 0.2)$	1.58×10^{-4}	2.84×10^{-5}	1.86×10^{-5}
$(0.3, 0.3)$	2.39×10^{-4}	2.36×10^{-5}	1.78×10^{-5}
$(0.4, 0.4)$	4.63×10^{-4}	2.48×10^{-5}	2.37×10^{-6}
$(0.5, 0.5)$	3.84×10^{-3}	8.20×10^{-4}	4.82×10^{-5}
$(0.6, 0.6)$	1.66×10^{-3}	1.12×10^{-4}	1.06×10^{-4}
$(0.7, 0.7)$	1.22×10^{-3}	1.80×10^{-4}	1.77×10^{-4}
$(0.8, 0.8)$	2.85×10^{-3}	2.81×10^{-4}	1.80×10^{-4}
$(0.9, 0.9)$	6.69×10^{-3}	3.88×10^{-4}	3.56×10^{-4}

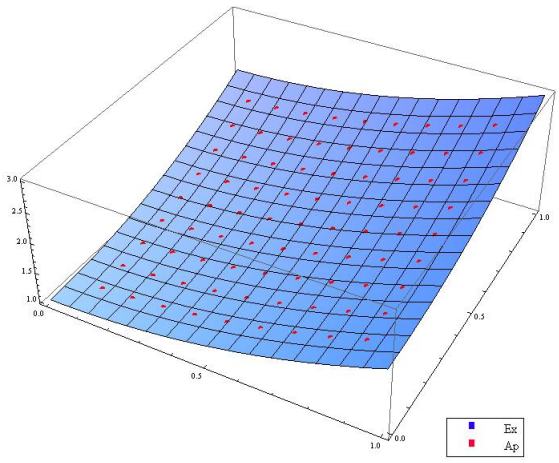

Figure 2: The plots of exact solution u_1 and approximate solution of u_1 with $\varepsilon = 0.000001$ in Example 2.

Table 5: Absolute errors of u_1 for different values of ε for normal equation in Example 2

(x_i, y_j)	Error $\varepsilon = 0.01$	Error $\varepsilon = 0.001$	Error $\varepsilon = 0.000001$
$(0.1, 0.1)$	8.54×10^{-8}	8.54×10^{-9}	8.54×10^{-12}
$(0.2, 0.2)$	2.56×10^{-8}	2.56×10^{-9}	2.56×10^{-12}
$(0.3, 0.3)$	4.19×10^{-8}	4.19×10^{-9}	4.19×10^{-12}
$(0.4, 0.4)$	1.10×10^{-7}	1.10×10^{-8}	1.10×10^{-11}
$(0.5, 0.5)$	5.46×10^{-7}	5.46×10^{-8}	5.46×10^{-11}
$(0.6, 0.6)$	1.73×10^{-6}	1.73×10^{-7}	1.73×10^{-10}
$(0.7, 0.7)$	2.54×10^{-6}	2.54×10^{-7}	2.54×10^{-10}
$(0.8, 0.8)$	2.86×10^{-6}	2.86×10^{-7}	2.86×10^{-10}
$(0.9, 0.9)$	2.58×10^{-6}	2.58×10^{-7}	2.58×10^{-10}

reported in Tables 5 and 6. Comparing displayed errors in the Tables 3, 4

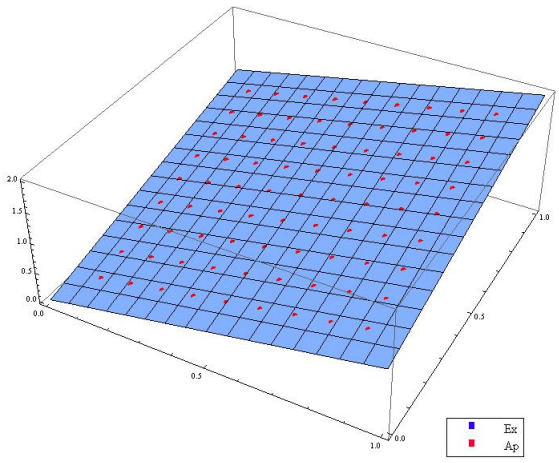

Figure 3: The plots of exact solution u_2 and approximate solution of u_2 with $\varepsilon = 0.000001$ in Example 2.

and 5, 6, we observe that the results in Tables 5 and 6 are more accurate than the reported results in Tables 3 and 4. This is predicted by the theory, in particular by Theorem 2. Figures 2 and 3 present the plots of exact and approximate solution with $\varepsilon = 0.000001$.

6.1 Haar wavelet method

In literature, several wavelets with different properties have been derived and depending upon the applications, different wavelet families are used. The Cheybyshev wavelets method is found to be simple, efficient, accurate, and computationally attractive for solving linear and non-linear problems. The properties of Chebyshev wavelets make the wavelet coefficient matrices sparse which eventually leads to the sparsity of the coefficients matrix of the obtained system. The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines. For comparison, we consider the Haar wavelet method to solve Example 1. The Haar wavelet family is

Table 6: Absolute errors of $u_{2,\varepsilon}$ for different values of ε for normal equation in Example 2

(x_i, y_j)	Error $\varepsilon = 0.01$	Error $\varepsilon = 0.001$	Error $\varepsilon = 0.000001$
$(0.1, 0.1)$	1.25×10^{-7}	1.00×10^{-8}	2.19×10^{-7}
$(0.2, 0.2)$	9.92×10^{-7}	9.83×10^{-8}	8.59×10^{-8}
$(0.3, 0.3)$	9.58×10^{-8}	1.05×10^{-8}	9.59×10^{-8}
$(0.4, 0.4)$	3.64×10^{-6}	3.64×10^{-7}	5.67×10^{-9}
$(0.5, 0.5)$	5.75×10^{-6}	5.67×10^{-7}	2.19×10^{-7}
$(0.6, 0.6)$	4.75×10^{-6}	4.73×10^{-7}	4.37×10^{-7}
$(0.7, 0.7)$	1.18×10^{-6}	1.17×10^{-7}	1.52×10^{-7}
$(0.8, 0.8)$	5.24×10^{-6}	5.24×10^{-7}	7.34×10^{-8}
$(0.9, 0.9)$	1.48×10^{-5}	1.48×10^{-6}	1.39×10^{-7}

$$h_i(t) = \begin{cases} 1, & t \in [\tau_1, \tau_2], \\ -1, & t \in [\tau_2, \tau_3], \\ 0, & elsewhere \end{cases}$$

where

$$\tau_1 = \frac{k}{m}, \quad \tau_2 = \frac{k + \frac{1}{2}}{m}, \quad \tau_3 = \frac{k+1}{m}.$$

The integer $m = 2^j, j = 0, 1, \ldots, p$ and $k = 0, 1, \ldots, m-1$ are the level of the wavelet and translation parameter, respectively. The index i is calculated from the formula $i = m + k + 1$ and the maximal value is $i = 2M (M = 2^p)$. The index $i = 1$ corresponds to the scaling function of the Haar wavelet $h_1(t) = 1$. We can consider an approximate solution of equation (37) as

$$\hat{u}_\epsilon(t, x) = \sum_{i=1}^{2M_1} \sum_{j=1}^{2M_2} a_{ij} h_i(t) h_j(x),$$

where the coefficients a_{ij} can be obtained by inserting approximate solution $\hat{u}_\epsilon(t, x)$ and the following collocation points in equation (37)

$$t_l = \frac{l - (\frac{1}{2})}{2M_1}, l = 1, 2, \ldots, 2M_1, \quad x_r = \frac{r - (\frac{1}{2})}{2M_2}, r = 1, 2, \ldots, 2M_2.$$

By assuming $M_1 = M_2 = 4$, the absolute error for different values of ε have been reported in Table 7. From Tables 1 and 7, we observe that the results obtained by the Chebyshev wavelet method are more accurate than the Haar wavelet method in this case.

Table 7: Absolute errors of u_ε for different values of ε by Haar wavelet method in Example 1

	Error	Error	Error
(t_i, x_j)	$\varepsilon = 0.01$	$\varepsilon = 0.001$	$\varepsilon = 0.000001$
$(0.1, 0.1)$	8.99×10^{-2}	8.07×10^{-2}	7.99×10^{-2}
$(0.2, 0.2)$	8.83×10^{-2}	2.11×10^{-2}	1.41×10^{-2}
$(0.3, 0.3)$	4.04×10^{-2}	1.72×10^{-2}	1.29×10^{-2}
$(0.4, 0.4)$	9.33×10^{-2}	7.99×10^{-2}	7.82×10^{-2}
$(0.5, 0.5)$	4.99×10^{-1}	5.00×10^{-1}	4.99×10^{-1}
$(0.6, 0.6)$	9.58×10^{-2}	8.25×10^{-2}	8.08×10^{-2}
$(0.7, 0.7)$	6.79×10^{-2}	1.98×10^{-2}	1.15×10^{-2}
$(0.8, 0.8)$	5.09×10^{-2}	1.78×10^{-2}	1.77×10^{-2}
$(0.9, 0.9)$	9.26×10^{-2}	8.01×10^{-2}	7.84×10^{-2}

7 Conclusion and future work

This work has been concerned with the regularization method to convert the mixed systems of the first and second-kind Volterra–Fredholm integral equations to the system of the second-kind Volterra–Fredholm integral equations. We presented the numerical method based on Chebyshev wavelets for solving the obtained second-kind problem. Convergence of the method was proved. These results were confirmed by some numerical examples.

In the present work, we considered the mixed system (1) in special case with $A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$. The mixed system (1) in general form with $A(t)$ will be investigated as our future work.

Acknowledgements

The authors would like to express their gratitude for the referees of the paper for their valuable suggestions that improved the final form of the paper.

References

1. M.A. Abdou, F.A. Salama, *Volterra–Fredholm integral equation of the first kind and spectral relationships*, Appl. Math. Comput. 153 (2004), 141–153.

2. J. Ahmadi Shali, P. Darania, A.A. Jodayree, *Collocation method for nonlinear Volterra–Fredholm integral equations*, Open J. Appl. Sci. 2 (2012),115–121.

3. H. Brunner, *Collocation methods for Volterra integral and related functional equations*, Cambridge Monographs on Applied and Computational Mathematics, vol. 15, Cambridge University Press, Cambridge, 2004.

4. O. Diekman, T*hresholds and traveling waves for the geographical spread of infection*, J. Math. Biol. 6 (1978), 109–130.

5. M. Hamina, J. Saranen , *On the spline collocation method for the single-layer heat operator equation*, Math. Comp. 62 (1994), 41–64.

6. L. Hocia, *On Approximate Solution for Integral Equation of the Mixed Type*, ZAMM J. Appl. Math. Mech. 76 (1996), 415–416.

7. Y, Iso, K. Onishi, *On the stability of the boundary element collocation method applied to the linear heat equation*, J. Comput. Appl. Math. 38 (1991), 201–209

8. J.P. Kauthen, *Continuous time collocation for Volterra–Fredholm integral equations*, Numer. Math. 56 (1989), 409–424.

9. A. Kirsch, *An introduction to the mathematical theory of inverse problem*, Second edition. Applied Mathematical Sciences, 120. Springer, New York, 2011.

10. K. Maleknejad, H. Almasieh, M. Roodaki, *Triangular functions (TF) method for the solution of nonlinear Volterra–Fredholm integral equations*, Commun. Nonlinear Sci. Numer. Simul. 15 (2010), 3293–3298

11. K. Maleknejad, M. Hadizadeh, *A New computational method for Volterra–Fredholm integral equations*, Comput. Math. Appl., 37 (1999), 1–8.

12. D.L. Phillips, A *technique for the numerical solution of certain integral equations of the first kind*, J. Assoc. Comput. Mach. 9 (1962), 84–97.

13. H. R. Thieme, *A model for the spatio spread of an epidemic*, J. Math. Biol. 4, (1977), 337–351.

14. A. N. Tikhonov, *On the solution of incorrectly posed problem and the method of regularization*, Soviet Math. 4 (1963), 1035–1038.

15. S. Yousefi, M. Razzaghi, *Legendre Wavelets method for the nonlinear Volterra–Fredholm integral equations*, Math. Comput. Simul. 70 (2005), 1–8.

16. A.M. Wazwaz, *A reliable treatment for mixed Volterra–Fredholm integral equations*, Appl. Math. Comput. 127 (2002), 405–414.

of Translation Persian

Abstracts

انتقال به دامنه ثابت در مدلسازی برنامه ریزی خطی برای رده ای از مسایل طراحی شکل بهینه

سیّدحامد هاشمی مهنه[1] *و محمّد هادی فراهی*[2]

[1] تهران، پژوهشگاه هوافضا
[2] دانشگاه فردوسی مشهد، دانشکده علوم ریاضی

دریافت مقاله ۲۷ بهمن ۱۳۹۴، دریافت مقاله اصلاح شده ۱۳ تیر ۱۳۹۵، پذیرش مقاله ۲۰ آبان ۱۳۹۶

چکیده : رده ای از مسائل طراحی شکل بهینه در این مقاله مطالعه می شوند. شکل دامنه به گونه ای تعیین می شود که پاسخ معادلات با مشتقات جزئی در ناحیه حاصل تا حد ممکن به یک حالت مطلوب نزدیک شود. بدین منظور، دامنه متغیر در مساله اولیه توسط دسته ای از توابع پارامتری می شود تا مساله طراحی شکل بهینه به یک مساله کنترل بهینه روی یک دامنه ثابت تبدیل شود. سپس این مساله به صورت یک مساله در فضای اندازه بیان می شود که خود یک مساله برنامه ریزی خطی با بعد نامتناهی است. در نهایت، اندازه بهینه که مبین شکل بهینه است، به کمک پاسخ یک مساله برنامه ریزی خطی با بعد متناهی تقریب زده می شود. روش پیشنهادی بر چند مثال عددی پیاده سازی و ارزیابی شده است.

کلمات کلیدی : تقریب؛ طراحی شکل بهینه؛ برنامهریزی خطی؛ نظریه اندازه.

تقریب معادله هاکسلی با طرح تفاضل متناهی غیراستاندارد

مهران نامجو و صادق زیبایی

دانشگاه ولی عصر رفسنجان، دانشکده علوم ریاضی

دریافت مقاله ۲۲ دی ۱۳۹۵، دریافت مقاله اصلاح شده ۴ آذر ۱۳۹۶، پذیرش مقاله ۲۳ خرداد ۱۳۹۷

چکیده : در این مقاله یک طرح تفاضل متناهی دقیق از نوع صریح برای معادله هاکسلی براساس طرح تفاضل متناهی غیراستاندارد ارائه شده است. در ادامه یک طرح تفاضل متناهی غیراستاندارد برای جواب عددی معادله هاکسلی پیشنهاد و مثبت بودن، کرانداری، سازگاری، پایداری و همگرایی طرح مورد بررسی قرار گرفته شده است. به منظورنشان دادن دقت و کارایی طرح، نتایج عددی آن با جواب دقیق و برخی روش‌های موجود مقایسه شده است.

کلمات کلیدی : معادله هاکسلی؛ طرح تفاضل متناهی غیراستاندارد؛ کرانداری و مثبت بودن؛ سازگاری، پایداری و همگرایی.

روش جستجوی همسایگی متغیر برای مسئله مینیمم رنگ آمیزی مجموع روی گراف‌های ساده

حمیده ابراهیمی[1] *و خدیجه صدری*[2]

[1] دانشگاه آزاد اسلامی، واحد رشت، گروه ریاضی
[2] دانشگاه آزاد اسلامی، واحد رشت، باشگاه نخبگان و محققان جوان

دریافت مقاله ۱۹ مرداد ۱۳۹۶، دریافت مقاله اصلاح شده ۶ تیر ۱۳۹۷، پذیرش مقاله ۷ مرداد ۱۳۹۷

چکیده : هدف مقاله فعلی، ساختن توابع ژاکوبی کسری انتقال یافته (SFJFs) بر اساس چندجملهای های ژاکوبی برای حل عددی معادلات دیفرانسیل پانتوگراف مرتبه کسری است. برای دستیابی به این هدف، ابتدا ماتریس های عملیاتی انتگرال، حاصل ضرب و پانتوگراف، مربوط به پایه مرتبه کسری، به دست می آیند (ماتریس عملیاتی انتگرال بر حسب تعریف انتگرال کسری ریمان–لیوویل به دست می آید). سپس، از این ماتریس ها برای تبدیل مسأله اصلی به مجموعه ای از معادلات جبری استفاده می شود. سرانجام، اعتبار و کارایی روش پیشنهادی به وسیله مثال های عددی نشان داده می شود. هم چنین، برخی قضایا در مورد وجود جواب مسأله تحت بررسی و همگرایی روش ارائه می شوند.

کلمات کلیدی : معادله دیفرانسیل پانتوگراف کسری؛ توابع ژاکوبی مرتبه کسری؛ مشتق کاپوتو؛ انتگرال ریمان–لیوویل.

بررسی مساله ۱-میانه بر روی شبکه ای با طول یال و زمان حرکت گسسته

مریم ابارشی و مهدی زعفرانیه

دانشگاه حکیم سبزواری، دانشکده ریاضی و علوم کامپیوتر، گروه ریاضی کاربردی

دریافت مقاله ۳۰ آبان ۱۳۹۶، دریافت مقاله اصلاح شده ۳ مرداد ۱۳۹۷، پذیرش مقاله ۷ شهریور ۱۳۹۷

چکیده : در این مقاله، مساله ۱-میانه بر روی شبکه های درختی بدون جهت با طول یال و زمان حرکت تصادفی گسسته بررسی می شود. تابع هدف مساله به دست آوردن بیشینه احتمالی است که مجموع وزنی مورد انتظار فاصله رئوس شبکه تا مکان بهترین سرویس دهنده از یک مقدار کران بالای از پیش تعیین شده کمتر باشد. یک الگوریتم تحلیلی برای حل مساله در شبکه های کوچک و متوسط طراحی شده است. اما برای حل مساله در شبکه های بزرگ قضیه حد مرکزی به کار گرفته شده است که در نتیجه آن مساله اصلی به یک مساله غیرخطی تبدیل می شود. مثالهای ارائه شده در بخش پایانی نشان دهنده تضمین دقت و کارایی روشهای پیشنهادی هستند.

کلمات کلیدی : مساله مکانیابی؛ مساله ۱-میانه؛ وزنهای احتمالی؛ زمان های سفر احتمالی.

مساله پیدا کردن $(k,l)-2$–هسته روی درختهای با وزن حقیقی

سمانه متولی اشکذری و جعفر فتحعلی

دانشگاه صنعتی شاهرود، دانشکده علوم ریاضی

دریافت مقاله ۱۵ تیر ۱۳۹۶، دریافت مقاله اصلاح شده ۳ مرداد ۱۳۹۷، پذیرش مقاله ۱۰ شهریور ۱۳۹۷

چکیده :
فرض کنید $T=(V,E)$ درختی شامل n راس باشد. یک $(k,l)-2$–هسته از درخت T شامل دو زیر درخت مجزا از T می باشد که هر یک از آنها حداکثر دارای T برگ و حداکثر قطر l هستند، همچنین مجموع فاصله تمام رئوس درخت تا این دو زیردرخت کمینه است. در این مقاله در ابتدا به بررسی مساله پیدا کردن $(k,l)-2$–هسته روی درختی که در آن وزن تمام رئوس یکسان است می پردازیم و نشان می دهیم جوابی برای این مساله وجود دارد که هیچ یک از هسته ها یک راس تنها نیست. سپس الگوریتمی برای پیدا کردن $(k,l)-2$–هسته روی درختهایی که وزن رئوس آنها می تواند مثبت یا منفی باشد، ارائه می دهیم.

کلمات کلیدی : هسته درخت؛ مکانیابی نیمه-ناخوشایند؛ زیردرخت میانه.

روشهای S-ROCK جدید برای حل معادلات دیفرانسیل تصادفی با نویز جابجایی

امیر حقیقی

دانشگاه رازی، دانشکده علوم، گروه ریاضی

دریافت مقاله ۲۴ آذر ۱۳۹۶، دریافت مقاله اصلاح شده ۷ مرداد ۱۳۹۷، پذیرش مقاله ۸ مهر ۱۳۹۷

چکیده : در این مقاله، ابتدا کلاس روشهای رونگ-کوتای تصادفی بیان شده توسط روسلر (۲۰۱۰) برای حل معادلات دیفرانسیل تصادفی با نویز جابجایی در نظر گرفته میشود. سپس، با استفاده از ایده بیان شده توسط بُرج و کُموری (۲۰۱۳)، یک کلاس از روشهای رونگ-کوتای تصادفی متعامد صریح (SROCKC۲) از مرتبهی قوی یک برای تقریب جواب معادلات دیفرانسیل تصادفی ایتو با نویز جابجایی طراحی میگردد. در ادامه، پایداری میانگین مربعی و مجانبی روشهای پیشنهادی روی یک معادلهی اسکالر خطی آزمون با نویز ضربی مورد تحلیل قرار میگیرد. در انتها، با ارائه نتایج عددی حاصل از شبیهسازی برخی مدلهای تصادفی کاربردی مطالب بیان شده مورد تایید قرار خواهد گرفت.

کلمات کلیدی : معادلات دیفرانسیل تصادفی؛ روشهای رونگ-کوتا؛ پایداری میانگین مربعی؛ معادلات سخت؛ نویز جابجایی.

منظم سازی و تحلیل عددی دستگاه مرکب معادلات انتگرالی ولترا-فردهلم نوع اول و دوم

سعید پیش بین و جواد شکری

دانشگاه ارومیه، دانشکده علوم، گروه ریاضی

دریافت مقاله ۱۸ اسفند ۱۳۹۵، دریافت مقاله اصلاح شده ۷ مهر ۱۳۹۷، پذیرش مقاله ۲۳ مهر ۱۳۹۷

چکیده : توجه به این نکته ضرروی است که دستگاه معادلات انتگرالی ولترا-فردهلم نوع اول و دوم جز معادلات بد وضع می باشند بنابراین حل دستگاه‌های گسسسته شده مربوط به این معادلات دارای مشکلات فراوانی است. ما در این مقاله ابتدا یک روش منظم سازی را در نظر گرفته و مسئله نوع اول بد وضع را به یک مسئله نوع دوم خوش وضع تبدیل می کنیم. در ادامه روش عددی براساس موجک چبیشف را برای حل دستگاه خوش وضع بکار برده و همگرایی روش مربوطه را تحلیل می‌کنیم. در پایان چند مثال عددی با جوابهای معلوم را برای نشان دادن کارایی روش عددی پیشنهاد شده در نظر می‌گیریم.

کلمات کلیدی : دستگاه مرکب معادلات انتگرالی ولترا-فردهلم نوع اول و دوم؛ روش منظم سازی؛ موجک چبیشف؛ آنالیز همگرایی.

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 either of the following ways:

a) Online submission (pdf or dvi files) via the website of the Journal at:

http://ijnao.um.ac.ir

b) Via journal's email mjms@um.ac.ir

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 web site) and send it to the Journal address. This will permit the publisher to publish and distribute the work.

Reviewers Acknowledgment

Perr reviewers are key to advancing scholarship and contributing to the quality of scholarly journals. We would like to sincerely thank the following reviewers who have taken part in the peer-review process for Iranian Journal of Numerical analysis and Optimization during the last year.

Abbasbandy, Saeid
Abbaszadeh, Mostafa
Abdi, Ali
Akhoundi Roshanavand, Naser
Afshari-Rad, Mariya
Alizadeh, Behrooz
Aminikhah, Hossein
Ansari, Mohammad Reza
Babaie-Kafaki, Saman
Babolian, Esmaiel
Baymani, Mojtaba
Dolati, A.
Effati, Sohrab
Emrouznejad, Ali
Erfanian, Majid
Eslami, Esfandiar
Falsone, Giovanni
Farahi, Mohammad Hadi
Farhadinia, Bahram
Fathali, Jafar
Gao, Wenwu
Garshasbi, Morteza
Ghanbari, Reza
Ghorbani, Asghar
Giyas Sakar, Mehmet
Hadizadeh Yazdi, Mahmoud
Hajarian, Masoud
Hashemi Mehne, Hamed
Hojjati, Gholamreza
Hou, Tianliang
Ibrahimov, Vagif
Janfada, Mohammad
Javidi, Mohammad
Kazem, Saeed
Khatibzadeh, Hadi
Khojasteh Salkuyeh, Davood
Li, Xiaolin
Marzban, Hamid Reza
Mehdizadeh Khalsaraei, Mohammad
Moeen Moghaddas, Furugh
Mokhtari, Reza
Molabahrami, Ahmad
Molavi Arabshahi, Mahboubeh
Monabbati, Ehsan
Mostafaee, Amin
Namjoo, Mehran
Nazemi, Alireza
Noori Skandari, Mohammad Hadi
Ordokhani, Yadollah
Pakdaman, Morteza
Panjeh Ali Beik, Fatemeh
Pishbin, Saeed
Pourgholi, Reza
Rafiei, Amin
Rashidinia, Jalil
Reutskiy, Sergiy
Saeedpoor, Mahdi
Safi, Mohmmadreza
Salahi, Maziar
Sharifi, Soroosh
Shivanian, Elyas
Shokri, Ali
Soleimani, Majid
Tohidi, Emran
Xia Li, Cui
Zakeri, Ali
Zhang, Shuhua

Iranian Journal of Numerical Analysis and Optimization

Former : MRJMS

CONTENTS Vol. 9, No.1, pp 1-150, 2019

web site : http://ijnao.um.ac.ir

Email : ijnao@um.ac.ir

ISSN-Print: : 2423-6977

ISSN-Online: : 2423-6969