Caspian Journal of Mathematical Sciences (CJMS)
University of Mazandaran, Iran
http://cjms.journals.umz.ac.ir
ISSN: 1735-0611

CJMS. **7**(1)(2018), 88-101

Using shifted Legendre scaling functions for solving fractional biochemical reaction problem

Haman Deilami Azodi [1]
[1]Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran

Abstract. In this paper, biochemical reaction problem is given in the form of a system of non–linear differential equations involving Caputo fractional derivative. The aim is to suggest an instrumental scheme to approximate the solution of this problem. To achieve this goal, the fractional derivation terms are expanded as the elements of shifted Legendre scaling functions. Then, applying operational matrix of fractional integration and collocation technique, the main problem is transformed to a set of non–linear algebraic equations. This obtained algebraic system can be solved by available standard iterative procedures. Numerical results of applying the proposed method are investigated in details.

Keywords: Legendre scaling functions, Fractional biochemical reaction problem, Caputo derivative, Collocation method.

2000 Mathematics subject classification: 26A33, 33C45, 65N35

1. Introduction

In order to explain enzyme processes and their basic enzymatic reaction, Michaelis and Menten recommended a clear and useful plan which is represented as [18]

$$E + S \rightleftharpoons I \rightarrow E + P, \tag{1.1}$$

in which E, S, I and P denote enzyme, substrate, enzyme–substrate intermediate complex and product, respectively.

[1]Corresponding author: haman.d.azodi@gmail.com
Received: 4 March 2018
Revised: 8 May 2018
Accepted: 15 May 2018

Considering the law of mass action, which expresses that reaction rates are proportional to the concentrations of the reactants, the time evolution of (1.1) can be identified from the solution of the following system of non–linear ordinary differential equations [19]

$$\begin{cases} S'(t) = -r_1 E(t)S(t) + r_{-1} I(t), \\ E'(t) = -r_1 E(t)S(t) + (r_{-1} + r_2) I(t), \\ I'(t) = r_1 E(t)S(t) - (-r_{-1} + r_2) I(t), \\ P'(t) = r_2 I(t), \end{cases} \tag{1.2}$$

under initial conditions $S(0) = S_0,\ \ E(0) = E_0,\ \ I(0) = I_0,\ \ P(0) = P_0$, where r_1, r_{-1} and r_2 are positive rate constants for each reaction. The system (1.2) can be converted to two equations for S and I and in dimensionless form of concentrations of substrate, u, and intermediate complex between enzyme and substrate, v, it is stated by

$$\begin{cases} u'(t) = -u(t) + (\beta - \alpha)v(t) + u(t)v(t), \\ v'(t) = \frac{1}{\gamma}\left(u(t) - \beta v(t) - u(t)v(t)\right), \end{cases} \tag{1.3}$$

subject to the initial conditions $u(0) = 1,\ \ v(0) = 0$ so that α, β and γ are dimensionless parameters. For knowing more details about converting (1.2) to (1.3), see [19]. In order to solve (1.3), some methods have been used. These methods can be categorized in the following: Multi–stage variational iteration method [6], Multi–stage homotopy perturbation method [7], Legendre wavelets Picard method [8], Picard Pade technique [10], and also Adomian decomposition method [19].

It has been indicated in [6] and [7] that variational iteration and homotopy perturbation methods do not work for solving the classical biochemical reaction problem. Hence, the authors proposed modification of the mentioned methods. However, these schemes are based on the analytical calculations and need to large computational effort.

It is worth mentioning that most of biological systems have memory and after–effects [16, 17]. In the models interpreted by integer–order derivatives, these features are ignored. Due to connection of fractional calculus with the systems involving memory and after–effects, modelling biological phenomena by using fractional differential equations is more appropriate [1–3, 5, 20]. In this way, the following system of fractional differential equations is presented

$$\begin{cases} D_*^{\mu_1} u(t) = -u(t) + (\beta - \alpha)v(t) + u(t)v(t),\ \ 0 < \mu_1 \leq 1, \\ D_*^{\mu_2} v(t) = \frac{1}{\gamma}\left(u(t) - \beta v(t) - u(t)v(t)\right),\ \ 0 < \mu_2 \leq 1, \end{cases} \tag{1.4}$$

with the initial conditions

$$u(0) = 1,\ \ v(0) = 0. \tag{1.5}$$

In this regard, Molliq et al. utilized modified step variational iteration method for solving (1.4) with the conditions (1.5) [13]. Herein, D_* refers to the Caputo fractional derivative. The Caputo derivative of order $0 < \mu < 1$ for an arbitrary function f is given as

$$D_*^{\mu} f(t) = \frac{1}{\Gamma(1-\mu)} \int_0^t (t-s)^{-\mu} f'(s) ds. \tag{1.6}$$

Besides this, fractional integral of Riemann–Liouville type of order $\mu > 0$ for $f : (0, \infty) \to \mathbb{R}$ is defined as follows

$$I^{\mu} f(t) = \frac{1}{\Gamma(\mu)} \int_0^t (t-s)^{\mu-1} f(s) ds. \tag{1.7}$$

It is notable that the identity below holds for Caputo fractional derivative and fractional integral of Riemann–Liouville type for $0 < \mu < 1$,

$$I^{\mu} D_*^{\mu} f(t) = f(t) - f(0). \tag{1.8}$$

Keep in mind that (1.6) and (1.7) constitute the basic and fundamental definitions of fractional calculus [4,15]. It is also clear that (1.4) reduces to (1.3) for $\mu_1 = \mu_2 = 1$.

On the other hand, the application of Legendre scaling functions has been investigated for various problems. For instance, these functions have been used successfully for the solution of Emden–Fowler differential equations [12], Riccati differential equation [14], and one–dimensional parabolic inverse problems [22]. Some advantages and profitable features of Legendre scaling functions can be listed as follows: (a) They are suitable for the computer programming. (b) The operational matrix obtained by scaling functions is often sparse. (c) By applying a few number of basis functions, it is possible to obtain a nice approximate solution. (d) The solution is multi–resolution type.

This paper suggests a new computer–oriented method for solving (1.4) under the initial values (1.5). The proposed method is based on shifted Legendre scaling functions, operational matrices and collocation points. Using shifted Legendre scaling functions, one can find the approximate solution of problem on any positive interval. Applying the operational matrix of fractional integration, which will be fabricated employing Block–Pulse functions, and collocation nodes belong to given interval, (1.4) is transformed to a system of non–linear algebraic equations. After solving the achieved algebraic system by every suitable method such as the Newton–Raphson iterative scheme, the solution of (1.4) with the initial conditions (1.5) can be determined. We display all of the calculations of the method in the matrix forms. This representation causes simplicity in the computer programming. The computations of

this paper are performed by MATLAB R2013a software on a 64-bit PC with 2.20 GHz Processor and 8 GB RAM.

The rest of this paper is organized as follows: In Sect. 2, Legendre scaling functions are first constructed on any positive interval. Then, some properties of them are described. Sect. 3 is devoted to the operational matrix of fractional integral of Riemann–Liouville type and method of solution. In Sect. 4, numerical solutions of (1.4) with the initial conditions (1.5) are presented by some graphs. Some concluding remarks are gathered in Sect. 5.

2. Shifted Legendre scaling functions

In what follows, the shifted Legendre scaling functions are formed by using the characteristics of shifted Legendre polynomials and scaling functions.

2.1. **Shifted Legendre polynomials.** The Legendre polynomials, L_i, defined on $[-1,1]$ are determined by the following recurrence relation

$$L_{i+1}(x) = \frac{2i+1}{i+1} x L_i(x) - \frac{i}{i+1} L_{i-1}(x), \;\; i \in \mathbb{N},$$

with $L_0(x) = 1$ and $L_1(x) = x$. The orthogonality of L_i is announced as

$$\int_{-1}^{1} L_i(x) L_j(x) dx = \begin{cases} 0, & i \neq j, \\ \frac{2}{2i+1}, & i = j. \end{cases}$$

Now, let τ be a real positive constant. Changing variables by $x = \frac{2t}{\tau} - 1$ transforms $x \in [-1,1]$ to $t \in [0,\tau]$. So, if $L_{i,\tau}$ allude to shifted Legendre polynomials for a given τ, then one implies that $L_{i,\tau}(t) = L_i(\frac{2t}{\tau} - 1)$. According to this denomination, we get

$$\int_{0}^{\tau} L_{i,\tau}(t) L_{j,\tau}(t) dt = \begin{cases} 0, & i \neq j, \\ \frac{\tau}{2i+1}, & i = j. \end{cases}$$

2.2. **Shifted Legendre scaling functions.** Assume that $M \in \mathbb{N}$. For $i = 0, 1, \ldots, M$, consider the following normalized set of functions

$$\varphi_\tau^i(t) = \begin{cases} \sqrt{\frac{2i+1}{\tau}} L_{i,\tau}(t), & t \in [0,\tau], \\ 0, & otherwise. \end{cases}$$

Notice coefficients $\sqrt{\frac{2i+1}{\tau}}$ have been inserted for normality. The shifted Legendre scaling functions are determined for a fixed $J \in \mathbb{N} \cup \{0\}$ by

$$\varphi_{J,k,\tau}^i(t) = \varphi_\tau^i(2^J t - k\tau), \tag{2.1}$$

where $i = 0, 1, \ldots, M$ and $k = 0, 1, \ldots, 2^J - 1$.

2.3. **Function approximation.** The function $f(t)$ defined on $[0,\tau]$ can be approximated using Legendre scaling functions specified in (2.1) for given J and M as

$$f(t) \approx \sum_{k=0}^{2^J-1} \sum_{i=0}^{M} c^i_{J,k} \varphi^i_{J,k,\tau}(t) = \mathbf{C}^T \mathbf{\Phi}(t), \tag{2.2}$$

so that

$$\mathbf{C} = \left[c^0_{J,0}, \ldots, c^M_{J,0}, |c^0_{J,1}, \ldots, c^M_{J,1}|, \ldots, |c^0_{J,2^J-1}, \ldots, c^M_{J,2^J-1}\right]^T,$$

$$\mathbf{\Phi}(t) = \left[\varphi^0_{J,0,\tau}(t), \ldots, \varphi^M_{J,0,\tau}(t), |\ldots, |\varphi^0_{J,2^J-1,\tau}(t), \ldots, \varphi^M_{J,2^J-1,\tau}(t)\right]^T. \tag{2.3}$$

Here, T is the transpose of a vector. In (2.2), $\mathbf{C}$ and $\mathbf{\Phi}(t)$ are column vectors with $n = 2^J(M+1)$ entries. In addition, the orthonormality of shifted Legendre scaling functions results in

$$c^i_{J,k} = \int_0^\tau \varphi^i_{J,k,\tau}(t) f(t) dt.$$

In the following theorem, the convergence of shifted Legendre scaling functions expansion (2.2) is verified.

Theorem 2.1. *Let $\tau > 0$. Assume that the function $f : [0,\tau] \to \mathbb{R}$ is i times continuously differentiable, namely $f \in C^i[0,\tau]$. Then, $\mathbf{C}^T\mathbf{\Phi}(t)$ approximates $f(t)$ with mean error bounded as follows*

$$\|f(t) - \mathbf{C}^T\mathbf{\Phi}(t)\| \leq \frac{\sqrt{\tau}}{i! 2^{iJ}} \sup_{0\leq t\leq \tau} |f^{(i)}(t)|,$$

in which $\|\cdot\|$ denotes to the usual norm in $L^2[0,\tau]$.

Proof. The interval $[0,\tau]$ may be divided into subintervals $\left[\frac{k\tau}{2^J}, \frac{(k+1)\tau}{2^J}\right]$ and if $\mathbf{C}^T\mathbf{\Phi}(t)$ and $\tilde{f}(t)$ be shifted Legendre scaling functions approach and interpolating polynomial of the function $f(t)$, respectively, then one

can deduce

$$\begin{aligned}
\|f(t)-\mathbf{C}^T\mathbf{\Phi}(t)\|^2 &= \int_0^{\tau}\left(f(t)-\mathbf{C}^T\mathbf{\Phi}(t)\right)^2 dt\\
&= \sum_{k=0}^{2^J-1}\int_{\frac{k\tau}{2^J}}^{\frac{(k+1)\tau}{2^J}}\left(f(t)-\mathbf{C}^T\mathbf{\Phi}(t)\right)^2 dt\\
&\le \sum_{k=0}^{2^J-1}\int_{\frac{k\tau}{2^J}}^{\frac{(k+1)\tau}{2^J}}\left(f(t)-\tilde{f}(t)\right)^2 dt\\
&\le \sum_{k=0}^{2^J-1}\int_{\frac{k\tau}{2^J}}^{\frac{(k+1)\tau}{2^J}}\left(\frac{1}{i!2^{iJ}}\sup_{0\le t\le\tau}|f^{(i)}(t)|\right)^2 dt\\
&\le \int_0^{\tau}\left(\frac{1}{i!2^{iJ}}\sup_{0\le t\le\tau}|f^{(i)}(t)|\right)^2 dt\\
&= \tau\left(\frac{1}{i!2^{iJ}}\sup_{0\le t\le\tau}|f^{(i)}(t)|\right)^2.
\end{aligned}$$

Ultimately, taking the square roots completes the proof. □

The error of shifted Legendre scaling functions $\mathbf{C}^T\mathbf{\Phi}(t)$ decays like 2^{-iJ}. As it was mentioned before, the number of basis functions is $n = 2^J(M+1)$. It is expressible that M refers to the degree of shifted scaling functions and J is the level of resolution. Theorem 2.1 exposes that by increasing M or J sufficiently, one can refine accuracy of the approximation.

3. Operational matrix and method of solution

In this section, we first construct operational matrix of fractional integration of the shifted Legendre scaling functions with the aid of Block–Pulse functions. Then, the proposed method is implemented for solving (1.4) under initial conditions (1.5). Hence, it is necessary to review some relevant materials of Block–Pulse functions [9].

3.1. Block–Pulse functions.

The n-set of Block–Pulse Functions on $[0,\tau]$ is defined in the following

$$b_i(t) = \begin{cases} 1, & \frac{(i-1)\tau}{n} \le t < \frac{i\tau}{n},\\ 0, & otherwise, \end{cases}$$

where $i = 1,\dots,n$. They are also disjoint and orthogonal, that is

- $b_i(t)b_j(t) = \begin{cases} 0, & i \ne j,\\ b_i(t), & i = j. \end{cases}$

- $\int_0^\tau b_i(t)b_j(t)dt = \begin{cases} 0, & i \neq j, \\ \frac{\tau}{n}, & i = j. \end{cases}$

Paying attention to the orthogonality of Block–Pulse functions, the function $f(t) \in L^2[0,\tau]$ can be written as

$$f(t) \approx \sum_{i=1}^{n} f_i b_i(t) = \mathbf{F}^T \mathbf{B}_n(t),$$

where

$$\mathbf{F} = [f_1, f_2, \ldots, f_n]^T, \ \ \mathbf{B}_n(t) = [b_1(t), b_2(t), \ldots, b_n(t)]^T,$$
$$f_i = \frac{n}{\tau} \int_0^\tau f(t) b_i(t) dt.$$

Moreover, if $f(t) = \mathbf{F}^T \mathbf{B}_n(t)$ and $g(t) = \mathbf{G}^T \mathbf{B}_n(t)$, then we can write $f(t)g(t) = \left(\mathbf{F}^T \otimes \mathbf{G}^T\right) \mathbf{B}_n(t)$. Here, the notation $\otimes$ denotes to the tensor product of two vectors $\mathbf{F} = [f_1, \ldots, f_n]^T$ and $\mathbf{G} = [g_1, \ldots, g_n]^T$ which is defined in the form of $\mathbf{F}^T \otimes \mathbf{G}^T = [f_1 g_1, \ldots, f_n g_n]^T$.

3.1.1. *Operational matrix of fractional integration.* In order to obtain the operational matrix of fractional integration for shifted Legendre scaling functions, we first expand the shifted Legendre scaling functions vector, $\mathbf{\Phi}(t)$, into an n–set of Block–Pulse functions, $\mathbf{B}_n(t)$. As it was noted before $n = 2^J(M+1)$. Then, operational matrix of fractional integration of $\mathbf{B}_n(t)$ is used to find operational matrix of fractional integration of $\mathbf{\Phi}(t)$. To the best of our knowledge, construction of the operational matrix of fractional integration using this process is easy to perform and has low computational effort. From [11], the fractional integration of order μ of $\mathbf{B}_n(t)$ is assigned as

$$I^\mu \mathbf{B}_n(t) \approx \mathbf{F}^\mu_{n\times n} \mathbf{B}_n(t), \tag{3.1}$$

where

$$\mathbf{F}^\mu_{n\times n} = \left(\frac{\tau}{n}\right)^\mu \frac{1}{\Gamma(\mu+2)} \begin{bmatrix} 1 & \xi_1 & \xi_2 & \ldots & \xi_{n-1} \\ 0 & 1 & \xi_1 & \ldots & \xi_{n-2} \\ 0 & 0 & 1 & \ldots & \xi_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 1 \end{bmatrix},$$

in which $\xi_l = (l+1)^{\mu+1} - 2l^{\mu+1} + (l-1)^{\mu+1}, \ \ l = 1, \ldots, n-1$. Now, let $\mathbf{P}^\mu_{n\times n}$ be the operational matrix of shifted Legendre scaling functions, that is

$$I^\mu \mathbf{\Phi}(t) \approx \mathbf{P}^\mu_{n\times n} \mathbf{\Phi}(t). \tag{3.2}$$

By choosing the collocation nodes $t_i = \frac{(2i-1)\tau}{2n},\ i = 1, \ldots, n$, we define

$$\mathbf{\Psi}_{n\times n} = [\mathbf{\Phi}(t_1), \mathbf{\Phi}(t_2), \ldots, \mathbf{\Phi}(t_n)]\,. \tag{3.3}$$

By this definition, the relation between $\mathbf{B}_n(t)$ and $\mathbf{\Phi}(t)$ for every $t = t_i$ becomes

$$\mathbf{\Phi}(t) = \mathbf{\Psi}_{n\times n}\mathbf{B}_n(t). \tag{3.4}$$

From (3.1) and (3.4), we have

$$I^{\mu}\mathbf{\Phi}(t) = \mathbf{\Psi}_{n\times n}I^{\mu}\mathbf{B}_n(t) \approx \mathbf{\Psi}_{n\times n}\mathbf{F}^{\mu}_{n\times n}\mathbf{B}_n(t) = \mathbf{\Psi}_{n\times n}\mathbf{F}^{\mu}_{n\times n}\mathbf{\Psi}^{-1}_{n\times n}\mathbf{\Phi}(t). \tag{3.5}$$

Note that $\mathbf{\Psi}^{-1}_{n\times n}$ is the inverse of $\mathbf{\Psi}_{n\times n}$. Eventually, comparison (3.2) with (3.5) entails

$$\mathbf{P}^{\mu}_{n\times n} \approx \mathbf{\Psi}_{n\times n}\mathbf{F}^{\mu}_{n\times n}\mathbf{\Psi}^{-1}_{n\times n}. \tag{3.6}$$

3.2. **Method of solution.** Expanding fractional derivative terms of (1.4) as the elements of $\mathbf{\Phi}(t)$, one can write

$$\begin{cases} D^{\mu_1}_{*}u(t) = \mathbf{C}^T\mathbf{\Phi}(t), \\ D^{\mu_2}_{*}v(t) = \mathbf{D}^T\mathbf{\Phi}(t), \end{cases} \tag{3.7}$$

in which $\mathbf{C}$ and $\mathbf{D}$ are the unknown column vectors. As we know, each of them has n entries. Integrating of fractional order from (3.7) and imposing initial conditions (1.5), enables one to get

$$\begin{cases} u(t) = \mathbf{C}^T I^{\mu_1}\mathbf{\Phi}(t) + 1 = \mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Phi}(t) + 1, \\ v(t) = \mathbf{D}^T I^{\mu_2}\mathbf{\Phi}(t) = \mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Phi}(t). \end{cases} \tag{3.8}$$

In (3.8), we have used the property (1.8). Collocating (3.8) at the collocation points t_i, $(i = 1, \ldots, n)$, and using (3.4), it is obvious that

$$\begin{cases} u(t) = \mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) + [1, \ldots, 1]_{1\times n}\mathbf{B}_n(t), \\ v(t) = \mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t). \end{cases} \tag{3.9}$$

Substituting (3.7) and (3.9) into (1.4) and considering (3.4), one points out

$$\begin{cases} \mathbf{C}^T\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) = -\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) - [1, \ldots, 1]_{1\times n}\mathbf{B}_n(t) \\ \qquad +(\beta-\alpha)\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) \\ \qquad +\left\{\left(\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Psi}_{n\times n} + [1, \ldots, 1]_{1\times n}\right)\otimes\left(\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Psi}_{n\times n}\right)\right\}\mathbf{B}_n(t), \\ \mathbf{D}^T\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) = \frac{1}{\gamma}\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) + \frac{1}{\gamma}[1, \ldots, 1]_{1\times n}\mathbf{B}_n(t) \\ \qquad -\frac{\beta}{\gamma}\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Psi}_{n\times n}\mathbf{B}_n(t) \\ \qquad -\frac{1}{\gamma}\left\{\left(\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\mathbf{\Psi}_{n\times n} + [1, \ldots, 1]_{1\times n}\right)\otimes\left(\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\mathbf{\Psi}_{n\times n}\right)\right\}\mathbf{B}_n(t). \end{cases}$$

Consequently,

$$\begin{cases} \mathbf{C}^T\boldsymbol{\Psi}_{n\times n} = -\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\boldsymbol{\Psi}_{n\times n} - [1,\ldots,1]_{1\times n} + (\beta-\alpha)\mathbf{C}^T\mathbf{P}^{\mu_2}_{n\times n}\boldsymbol{\Psi}_{n\times n} \\ \qquad\qquad + \left(\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\boldsymbol{\Psi}_{n\times n} + [1,\ldots,1]_{1\times n}\right)\otimes\left(\mathbf{C}^T\mathbf{P}^{\mu_2}_{n\times n}\boldsymbol{\Psi}_{n\times n}\right), \\ \mathbf{D}^T\boldsymbol{\Psi}_{n\times n} = \frac{1}{\gamma}\left(\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\boldsymbol{\Psi}_{n\times n} + [1,\ldots,1]_{1\times n}\right) - \frac{\beta}{\gamma}\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\boldsymbol{\Psi}_{n\times n} \\ \qquad\qquad -\frac{1}{\gamma}\left(\mathbf{C}^T\mathbf{P}^{\mu_1}_{n\times n}\boldsymbol{\Psi}_{n\times n} + [1,\ldots,1]_{1\times n}\right)\otimes\left(\mathbf{D}^T\mathbf{P}^{\mu_2}_{n\times n}\boldsymbol{\Psi}_{n\times n}\right). \end{cases} \tag{3.10}$$

Clearly, (3.10) is a non–linear algebraic system including $2n$ equations and $2n$ unknowns. This system can be solved by the Newton–Raphson method or *fsolve* function of MATLAB software. At the end, solution of problem can be uncovered by (3.8) for every $t \in [0,\tau]$.

The algorithm of the proposed method is summarized as follows:

Algorithm.
Input: $M \in \mathbb{N}$, $J \in \mathbb{N}\cup\{0\}$, $\tau \in \mathbb{R}^+$; $0 < \mu_1, \mu_2 \leq 1$, and the parameters α, β, γ.
Step 1: Define $\varphi^i_{J,k,\tau}(t)$ by (2.1).
Step 2: Construct the vector $\boldsymbol{\Phi}(t)$ from (2.3).
Step 3: Construct the matrix $\boldsymbol{\Psi}_{n\times n}$ according to (3.3).
Step 4: Compute the fractional operational matrices, $\mathbf{P}^{\mu_1}_{n\times n}$ and $\mathbf{P}^{\mu_2}_{n\times n}$, by (3.6).
Step 5: Define unknown vectors $\mathbf{C} = [c_1,\ldots,c_n]^T$ and $\mathbf{D} = [d_1,\ldots,d_n]^T$.
Step 6: Constitute the system of algebraic equations from (3.10).
Step 7: Solve system of Step 6 using Newton–Raphson scheme or *fsolve* function.
Step 8: Substitute the obtained $\mathbf{C}$ and $\mathbf{D}$ into (3.8).
Output: The approximate solutions of $u(t)$ and $v(t)$ on $[0,\tau]$.

---

4. Numerical results

In the following, the competence of present method and the treatment of the fractional model are discussed. We offer the outcomes of applying the recommended algorithm through several figures.

Consider that $\mu_1 = \mu_2 = \mu$, $\tau = 1$, $\alpha = 0.375$, $\beta = 1$ and $\gamma = 0.1$. To test the accuracy and efficiency of the present method, we compute the differences of our solution (for $J = 2$) in the case of $\mu = 1$ from the fourth order Runge–Kutta method (with step–size=0.001). These values are as follows

$$\begin{cases} err(u) = |u(t) - u_{rk}(t)|, \\ err(v) = |v(t) - v_{rk}(t)|, \end{cases}$$

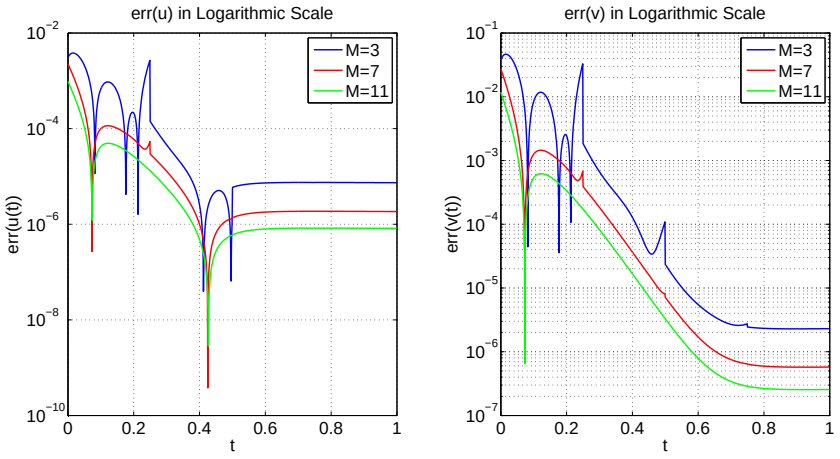

FIGURE 1. The graphs of $err(u)$ and $err(v)$ for $J = 2$

where $(u(t), v(t))$ and $(u_{rk}(t), v_{rk}(t))$ are the present solution and RK4 solution, respectively. Figure 1 demonstrates the graphs of $err(u)$ and $err(v)$ for $J = 2$ with different M in logarithmic scale. As we expect, by augmentation of M sufficiently, $err(u)$ and $err(v)$ decrease. This means that the solutions of present method for $\mu = 1$ and RK4 method are closer to each other when M increases.

Figure 2 illustrates the behaviour of solution for aforesaid α, β, γ and $J = 2$, $M = 7$ (or $n = 32$) with various amounts of μ. We have chosen $\mu = 0.65, 0.75, 0.85, 0.95$ and 1.00. It is obvious that when μ tends to 1, then the solution of fractional biochemical reaction model $(0 < \mu < 1)$ closes to the solution of classical model $(\mu = 1)$. Furthermore, the present approach for $\mu = 1.00$ is well–adapted to the RK4 solution.

Now, the solution of (1.4) is evaluated for each dimensionless reaction parameter (namely α, β, γ) we want to vary it. Indeed, we are interested to assess the effects of α, β, γ on the concentration of $u(t)$ and $v(t)$ for the problem (1.4).

Let $\mu = 0.9$ is given and $n = 32$. Figure 3 shows the behaviour of $u(t)$ and $v(t)$ for $\beta = 1$, $\gamma = 0.1$ and variant α. Here, $\alpha = 0.2, 0.4, 0.6, 0.8, 1.0$. It is visible that when the amount of α increases, the concentration of $u(t)$ and $v(t)$ decreases.

Figure 4 portrays the treatment of $u(t)$ and $v(t)$ for $\alpha = 0.375$, $\gamma = 0.1$ and different quantities of β; $\beta = 0.2, 0.4, 0.6, 0.8, 1$. It is seen that if the value of β increases, then the concentration of $u(t)$ increases and the concentration of $v(t)$ decreases.

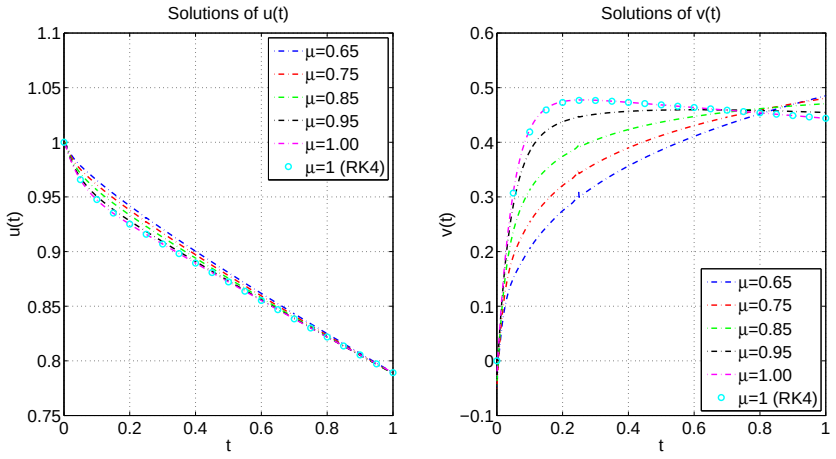

FIGURE 2. Solutions of $u(t)$ and $v(t)$ for $\alpha = 0.375$, $\beta = 1$, $\gamma = 0.1$ and $n = 32$

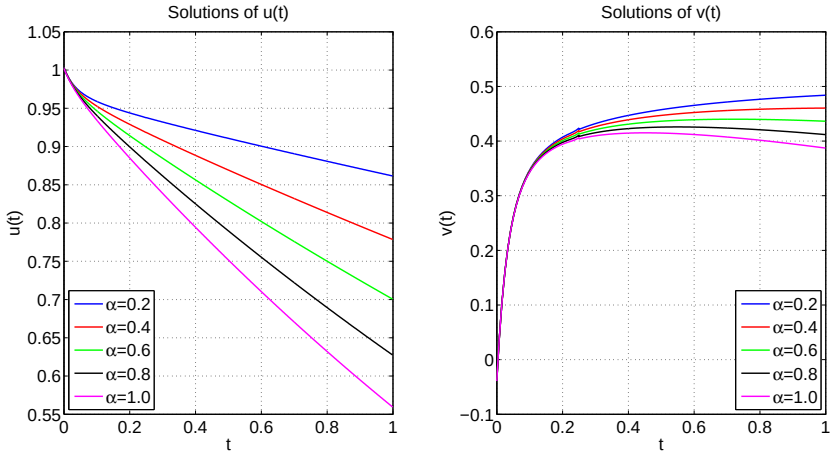

FIGURE 3. Solutions of $u(t)$ and $v(t)$ for $\beta = 1$, $\gamma = 0.1$, $\mu = 0.9$ and $n = 32$

Figure 5 exhibits behaviour of $u(t)$ and $v(t)$ for $\alpha = 0.375$, $\beta = 0.1$ and diverse values of γ; $\gamma = 0.2, 0.4, 0.6, 0.8, 1$. One can observe that when the value of γ increases, the concentration of $u(t)$ and $v(t)$ decreases.

Briefly, from Figures 3, 4, and 5, we conclude the following points

(i) For fixed β and γ, by increasing α, the concentration of $u(t)$ and $v(t)$ decreases.

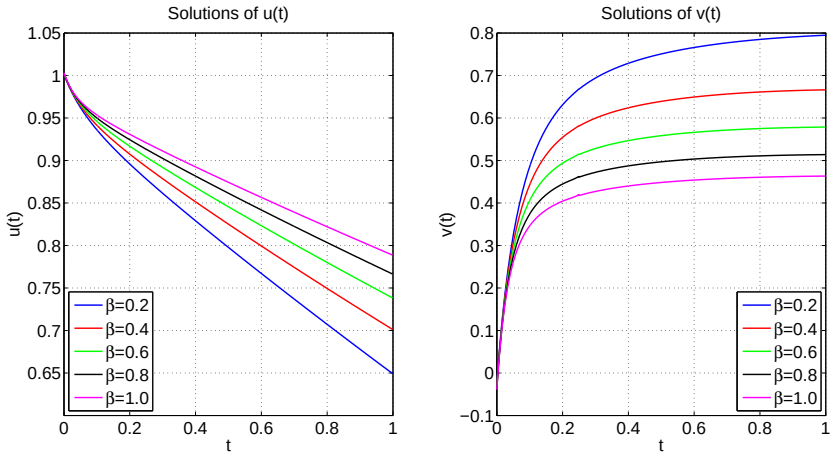

FIGURE 4. Solutions of $u(t)$ and $v(t)$ for $\alpha = 0.375$, $\gamma = 0.1$, $\mu = 0.9$ and $n = 32$

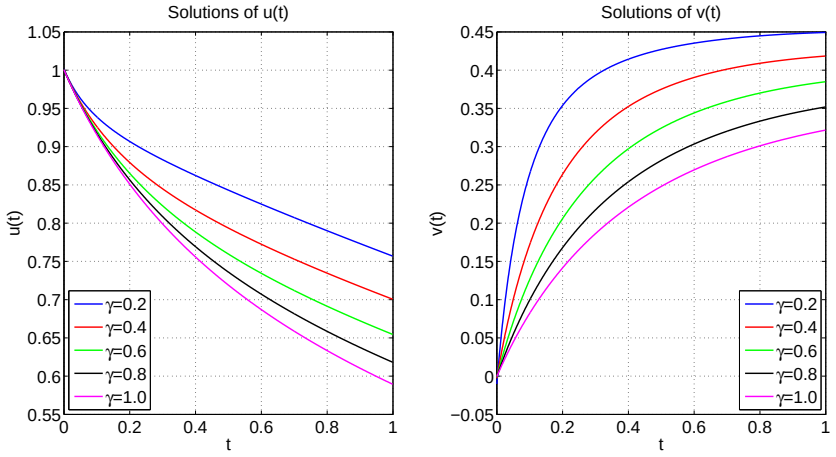

FIGURE 5. Solutions of $u(t)$ and $v(t)$ for $\alpha = 0.375$, $\beta = 1$, $\mu = 0.9$ and $n = 32$

(ii) For fixed α and γ, by increasing β, the concentration of $u(t)$ increases and the concentration of $v(t)$ decreases.

(iii) For fixed α and β, by increasing γ, the concentration of $u(t)$ and $v(t)$ decreases.

5. Conclusion

The fractional biochemical reaction model produces freedom in the growing or decaying of the substrate and enzyme–substrate intermediate complex. Throughout this paper, a contributory method was derived to solve numerically fractional biochemical reaction problem. As it was shown in the present research, the proposed method can solve both the classical and fractional model, effectively.

References

[1] E. Ahmed, and H. A. El–Saka, *On fractional order models for Hepatitis C*, Nonlinear Biomed. Phys., 4(1) (2010), 1–4.

[2] A. M. M. Arafa, S. Z. Rida, and M. Khalil, *A fractional–order model of HIV infection with drug therapy effect*, J. Egypt. Math. Soc., 22(3) (2014), 538–543.

[3] R. S. Damor, S. Kumar, and A. K. Shukla, *Numerical solution of fractional bioheat equation with constant and sinusoidal heat flux condition on skin tissue*, Amer. Math. Soc., 1(2) (2013), 20–24.

[4] K. Diethelm, *The Analysis of Fractional Differential Equations*, Springer-Verlag, Berlin, 2010.

[5] Y. Ding, and H. Ye, *A fractional–order differential equation model of HIV infection of $CD4^+$ T–cells*, Math. Comput. Model., 50(3) (2009), 386–392.

[6] S. M. Goh, M. S. M. Noorani, and I. Hashim, *Introducing variational iteration method to a biochemical reaction model*, Nonlinear Anal. Real World Appl., 11(4) (2010), 2264–2272.

[7] I. Hashim, M. S .H. Chowdhury, and S. Mawa, *On multistage homotopy–perturbation method applied to nonlinear biochemical reaction model*, Chaos Solitons Fract., 36(4) (2008), 823–827.

[8] M. A. Iqbal, and S. T. Mohyud–Din, *A study of nonlinear biochemical reaction model*, Int. J. Biomath., 9(5) (2016), Article ID 1650071, 9 pages.

[9] Z. Jiang, and W. Schaufelberger, *Block Pulse functions and their applications in control systems*, Springer, Berlin, 1992.

[10] M. M. Khader, *On the numerical solutions to nonlinear biochemical reaction model using Picard–Pade technique*, World J. Model. Simul., 9(1) (2013), 38–46.

[11] A. Kilicman, and Z. A. A. Al–Zhour, *Kronecker operational matrices for fractional calculus and some applications*, Appl. Math. Comput., 187(1) (2017), 250–265.

[12] M. Lakestani, and B. Nemati Saray, *Numerical solution of singular IVPs of Emden–Fowler type using Legendre scaling functions*, Int. J. Nonlinear Sci., 13(2) (2012), 211–219.

[13] R. Y. Molliq, M. S. M. Noorani, R. R. Ahmad, and A. K. Alomari, *Modified step variational iteration method for solving fractional biochemical reaction model*, Int. J. Differ. Equ., 2011, Article ID 514384, 12 pages.

[14] R. Najafi Baghchehjoughi, B. Nemati Saray, and M. Lakestani, *Numerical solution of Riccati differential equation using Legendre scaling functions*, J. Curr. Res. Sci., 2(1) (2014), 149–153.

[15] I. Podlubny, *Fractional Differential Equations*, Academic Press, New York, 1999.

[16] F.A. Rihan, *Numerical modelling of fractional–order biological systems*, Abstr. Appl. Anal., 2013, Article ID 816803, 11 pages.

[17] A. Rocco, and B. J. West, *Fractional calculus and the evolution of fractal phenomena*, Physica A Stat. Mech. Appl., 265(3) (1999), 535–546.

[18] S. Schnell, and C. Mendoza, *Closed form solution for time-dependent enzyme kinetics*, J. Theor. Biol., 187(2) (1997), 207–212.

[19] A. K. Sen, *An application of the Adomian decomposition method to the transient behavior of a model biochemical reaction*, J. Math. Anal. Appl., 131(1) (1988), 232–245.

[20] J. Singh, P. K. Gupta, and K. N. Rai, *Solution of fractional bioheat equations by finite difference method and HPM*, Math. Comput. Mode., 54(9) (2011), 2316–2325.

[21] A. Yildirim, A. Gokdogan, and M. Merdan, *Numerical approximations to solution of biochemical reaction model*, Int. J. Chem. React. Eng. 9(1) (2011), DOI: https://doi.org/10.2202/1542-6580.2606.

[22] S. A. Yousefi, and M. Dehghan, *Legendre multiscaling functions for solving the one–dimensional parabolic inverse problem*, Numer. Methods Partial Differ. Equ., 25(6) (2009), 1502–1510.