Far East Journal of Dynamical Systems

https://pphmjopenaccess.com
https://doi.org/10.17654/0972111826002 **P-ISSN: 0972-1118**
Volume 39, Number 1, June 2026, Pages 23-59 **E-ISSN: 3048-7897**

EXACT SOLUTION OF SOME FRACTIONAL SYSTEMS OF PARTIAL DIFFERENTIAL EQUATIONS VIA THE SBA AND VIM METHODS

Charvelin BILOUMBOU NKOUKA, Rabi Bachire BEKAKO ALI, Joseph BONAZEBI YINDOULA* and Alphonse MASSAMBA

Université Marien NGOUABI
Brazzaville, Congo
e-mail: bonayindoula@yahoo.fr

Abstract

In this paper, the SBA method and VIM are used to obtain an exact solution of fractional systems of partial differential equations. To test the effectiveness of these methods, three numerical examples were solved. The obtained results indicate the identical exact solution for each example test. Thus the accuracy of these methods is in a good agreement with the exact solution. However, a comparison between these methods shows that the SBA method provides more accurate results.

Received: May 20, 2025; Revised: June 9, 2025; Accepted: June 19, 2025
2020 Mathematics Subject Classification: 35R11, 35A20, 35C05, 49M05, 65M70.
Keywords and phrases: Some Blaise Abbo (SBA) method, variational iteration method (VIM), Lagrange multiplier.
*Corresponding author
How to cite this article: Charvelin BILOUMBOU NKOUKA, Rabi Bachir BEKAKO ALI, Joseph BONAZEBI YINDOULA and Alphonse MASSAMBA, Exact solution of some fractional systems of partial differential equations via the SBA and VIM methods, Far East Journal of Dynamical Systems 39(1) (2026), 23-59. https://doi.org/10.17654/0972111826002

Published Online: December 11, 2025

1. Introduction

Water which covers around 70% of the earth's surface, is undoubtedly the most abundant resource on our planet. However, nowadays, with the ever-increasing development of industry, agriculture and other human activities, its diversified pollution is at the root of many health and ecological problems. This pollution stems from industrial waste and discharges, the use of pesticides and fertilizers (through nitrites, chlorides and phosphates) in agriculture, etc. One of the consequences is that, on a global scale, the water quality is deteriorating. Another consequence is that the water in rivers and streams experiences excessive growth of different types of organisms which clogs our waterways, and blocks light from reaching deeper waters. This, in turn, proves very harmful to aquatic organisms as it affects the respiration ability of fish and other invertebrates that reside in water. In addition, pollution in the form of organic material enters waterways in many different forms such as sewages, leaves and grass clippings, or runoff from livestock feedlots and pastures. When natural bacteria and protozoan in the water break down this organic material, they begin to use up the oxygen dissolved (OD) in the water. Many types of fish and others aquatic animal species cannot survive when levels of dissolved oxygen drop below two to five numbers which leads to disruption in the food chain. It is in this context that several mathematical models have been developed by researchers to describe the process of pollution propagation in surface waters [1]. These include the mathematical model of Streeter and Phelps [2], which dates back to 1925 and has been modified in various ways [3], followed by models of contaminant transport [4, 5], bacterial transport [6], biological demand for dissolved oxygen (BOD) versus dissolved oxygen (DO) [7], and models of dissolved organic matter (DOM) transport in water (bacteria, contaminants) [8]. For this purpose, several numerical methods have been used to solve the PDEs and PDE systems associated with the various models, such as the Finite Volume Method (FVM) [9], the Crank-Nicolson numerical scheme method [10], splitting methods [11], the Euler method [12], the Runge-Kutta method [13], etc. In recent decades, in order to gain a better understanding of the problem of water pollution, researchers have increasingly turned their attention to PDEs and systems of fractional-order PDEs modelling the phenomenon [14, 15]. The fractional derivative is well suited to describing the water pollution process, which is strongly conditioned by the geometry of the problem. In this paper, we are interested in the following mathematical model, which describes surface water pollution in terms of BOD and DO concentration:

$$\begin{cases} {}^{c}D_t^{\alpha}u(x, t) + R_1 u(x, t) + N_1(u(x, t), v(x, t)) = f_1(x, t) \\ {}^{c}D_t^{\alpha}v(x, t) + R_2 v(x, t) + N_2(u(x, t), v(x, t)) = f_2(x, t) \\ u(x, 0) = g(x) \\ v(x, 0) = h(x) \end{cases} \text{with } \begin{cases} x \in \mathbb{R} \\ t \geq 0 \\ 0 < \alpha \leq 1, \end{cases} \tag{1}$$

where u and v are, respectively, the BOD and DO concentrations. The aim of this study is to determine the solution of this fractional system of PDEs in order to give real meaning to the model through test problems. We use the Some Blaise-Abbo method (SBA method) and the variational iteration method (VIM) to solve model (1). These techniques enable us to obtain an approximate or exact solution for the considered model. In addition, we compare the results obtained from both techniques, demonstrating their identical nature. The structure of this paper is organized as follows. In Section 2, we revisit fundamental concepts of fractional calculus, outlining their key characteristics and the SBA method and the VIM method. Section 3 discusses the application of the SBA method and the VIM method to three problem-tests. Finally, in Section 4, the concluding remarks of the study are provided.

2. Presentation of Resolution Methods

2.1. Preliminaries

2.1.1. Definition 1 [16]

The *Mittag-Leffler function* is defined by:

Properties

(i) $E_\alpha(t^\alpha)E_\alpha(\nu^\alpha) = E_\alpha((t+\nu)^\alpha), \quad 0 < \alpha \le 1$

(ii) $E_\alpha(t^\alpha)E_\alpha(-\nu^\alpha) = E_\alpha((t-\nu)^\alpha), \quad 0 < \alpha \le 1$

(iii) $\dfrac{d^\alpha E_\alpha(t^\alpha)}{dt^\alpha} = E_\alpha(t^\alpha)$

(iv) $\dfrac{d^\alpha E_\alpha(kt^\alpha)}{dt^\alpha} = kE_\alpha(t^\alpha).$

2.1.2. Definition 2 [16]

Let $f \in C([a, b])$. The operator I_a^α defined on $[a, b]$ by:

$$(I_a^\alpha f)(t) = \frac{1}{\Gamma(\alpha)}\int_a^t (t-\tau)^{\alpha-1} f(\tau)d\tau, \quad \alpha > 0 \tag{2}$$

is called *Riemann-Liouville fractional integral of order* α

Properties

Let α and β be two complex numbers, $f \in C([a, b])$

(i) $I_a^\alpha(I_a^\beta f) = I_a^{\alpha+\beta} f,\ Re(\alpha) > 0,\ Re(\beta) > 0$

(ii) $\dfrac{d}{dt}(I_a^\alpha f)(t) = (I_a^{\alpha+1} f)(t),\ Re(\alpha) > 0$

(iii) $\lim\limits_{a\to 0^+}(I_a^\alpha f)(t) = f(t),\ Re(\alpha) > 0$

(iv) $I_a^\alpha(t^\beta) = \dfrac{\Gamma(\beta+1)}{\Gamma(\alpha+\beta+1)} t^{\alpha+\beta}.$

2.1.3. Definition 3 [16]

The Caputo-type fractional derivative of order $\alpha > 0$ of a function $u \in C_{-1}^m$ $(m = 1, 2, \ldots)$ is given by:

$$\begin{cases} {}^{c}D^{\alpha}u(x, t) = \dfrac{\partial^{\alpha}u(x, s)}{\partial s^{\alpha}} = \dfrac{1}{\Gamma(n-\alpha)} \\ \qquad\qquad \cdot \displaystyle\int_a^t (t-s)^{n-\alpha-1}\dfrac{\partial^{n}u(x, s)}{\partial s^{n}}ds, & \text{if } n-1 < \alpha < n \\ \dfrac{d^n}{dt^n}u(x, t), & \text{if } \alpha = n, \end{cases} \tag{3}$$

where $n = [\alpha] + 1$ is the integer part of the real number.

2.1.4. Definition 4 [16]

The fractional of integration of Caputo derivative for $n - 1 < \alpha < n$, $n \in \mathbb{N}$, $u \in C_{\mu}^{m}$ $(m = 1, 2, ...)$, $\mu \geq -1$, is expressed as

$$I_a^{\alpha}({}^{c}D^{\alpha}u(x, t)) = u(x, t) - \sum_{k=0}^{n-1}\frac{\partial^k u(x, 0)}{\partial t^k}\frac{t^k}{k!}. \tag{4}$$

2.2. SBA method [17-22]

Consider the following functional system:

$$\begin{cases} {}^{c}D_t^{\alpha}u(x, t) + R_1u(x, t) + N_1(u(x, t), v(x, t)) = f_1(x, t) \\ {}^{c}D_t^{\alpha}v(x, t) + R_2v(x, t) + N_2(u(x, t), v(x, t)) = f_2(x, t), \end{cases} \tag{5}$$

where $L_t = {}^{c}D_t^{\alpha}(\cdot)$ is invertible in the Adomian sense, i.e., $L_t^{-1} = I_t^{\alpha}(\cdot)$.

Applying L_t^{-1} to (5), we obtain the Adomian canonical form:

$$\begin{cases} u(x, t) = g(x) + I_t^{\alpha}(f_1) - I_t^{\alpha}(R_1u(x, t)) - I_t^{\alpha}(N_1(u(x, t), v(x, t))) \\ v(x, t) = h(x) + I_t^{\alpha}(f_2) - I_t^{\alpha}(R_2v(x, t)) - I_t^{\alpha}(N_2(u(x, t), v(x, t))). \end{cases} \tag{6}$$

Applying the method of successive approximation to (6), we obtain:

$$\begin{cases} u^k(x, t) = g(x) + I_t^{\alpha}(f_1) - I_t^{\alpha}(R_1 u^k(x, t)) \\ \qquad\qquad - I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v^k(x, t) = h(x) + I_t^{\alpha}(f_2) - I_t^{\alpha}(R_2 v^k(x, t)) \\ \qquad\qquad - I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \end{cases}, \quad \forall k \geq 1. \tag{7}$$

By applying the Adomian algorithm to (7), we obtain

$$\begin{cases} \begin{cases} u_0^k(x, t) = g(x) + I_t^{\alpha}(f_1) - I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ u_{n+1}^k(x, t) = -I_t^{\alpha}(R_1 u_n^k(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^k(x, t) = h(x) + I_t^{\alpha}(f_2) - I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v_{n+1}^k(x, t) = -I_t^{\alpha}(R_2 v_n^k(x, t)), \quad \forall n \geq 0 \end{cases} \end{cases}, \forall k \geq 1. \tag{8}$$

We apply the Picard principle to equation (8). Let u^0 and v^0 be such that $N_1(u^0(x, t), v^0(x, t)) = N_2(u^0(x, t), v^0(x, t)) = 0.$ Thus

For $k = 1,$ we get:

$$\begin{cases} \begin{cases} u_0^1(x, t) = g(x) + I_t^{\alpha}(f_1) \\ u_{n+1}^1(x, t) = -I_t^{\alpha}(R_1 u_n^1(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = h(x) + I_t^{\alpha}(f_2) \\ v_{n+1}^1(x, t) = -I_t^{\alpha}(R_2 v_n^1(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{9}$$

If the series

$$\begin{cases} \displaystyle\sum_{n=0}^{\infty} u_n^1(x, t) \\ \qquad \text{and} \\ \displaystyle\sum_{n=0}^{\infty} v_n^1(x, t) \end{cases} \tag{10}$$

converges, then

$$\begin{cases} u^1(x, t) = \sum_{n=0}^{\infty} u_n^1(x, t) \\ \quad\text{and} \\ u^1(x, t) = \sum_{n=0}^{\infty} v_n^1(x, t). \end{cases} \tag{11}$$

For $k = 2,$ we get

$$\begin{cases} \begin{cases} u_0^2(x, t) = g(x) + I_t^{\alpha}(f_1) - I_t^{\alpha}(N_1(u^1(x, t), v^1(x, t))) \\ u_{n+1}^2(x, t) = -I_t^{\alpha}(R_1 u_n^2(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = h(x) + I_t^{\alpha}(f_2) - I_t^{\alpha}(N_2(u^1(x, t), v^1(x, t))) \\ v_{n+1}^2(x, t) = -I_t^{\alpha}(R_2 v_n^2(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{12}$$

Here, we must verify that $N_1(u^1(x, t), v^1(x, t)) = N_2(u^1(x, t), v^1(x, t)) = 0,$ then

$$\begin{cases} \begin{cases} u_0^2(x, t) = g(x) + I_t^{\alpha}(f_1) \\ u_{n+1}^2(x, t) = -I_t^{\alpha}(R_1 u_n^2(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = h(x) + I_t^{\alpha}(f_2) \\ v_{n+1}^2(x, t) = -I_t^{\alpha}(R_2 v_n^2(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{13}$$

If the series

$$\begin{cases} \sum_{n=0}^{\infty} u_n^2(x, t) \\ \quad\text{and} \\ \sum_{n=0}^{\infty} v_n^2(x, t) \end{cases} \tag{14}$$

converges, then

$$\begin{cases} u^2(x, t) = \sum_{n=0}^{\infty} u_n^2(x, t) \\ \quad \text{and} \\ u^2(x, t) = \sum_{n=0}^{\infty} v_n^2(x, t). \end{cases} \tag{15}$$

We reapply the same procedure for $k \geq 3$ so that the solution of system (5) is

$$\begin{cases} u(x, t) = \lim_{k \to +\infty} u^k(x, t) \\ v(x, t) = \lim_{k \to +\infty} v^k(x, t). \end{cases} \tag{16}$$

2.3. Variational iteration method (VIM) [23-26]

We can construct a functional correction for (1) using the following variational iteration method

$$\begin{cases} u_{n+1} = u_n + \int_0^t \lambda_1 [{}^c D_t^{\alpha} u_n + N_1(\breve{u}_n, \breve{v}_n) + R\breve{u}_n - f_1] ds \\ v_{n+1} = v_n + \int_0^t \lambda_2 [{}^c D_t^{\alpha} v_n + N_1(\breve{u}_n, \breve{v}_n) + R\breve{v}_n - f_2] ds, \end{cases} \tag{17}$$

where λ_1, λ_2 are Lagrange multipliers, the index n represents the nth approximation and $(\widetilde{u}_n, \widetilde{v}_n)$ considered as a restricted variation, i.e., $\delta\widetilde{u}_n = \delta\widetilde{v}_n = 0$.

To solve the system (6) using the VIM method, we must first determine the Lagrange multipliers λ_1, λ_2 which will be identified via integration by parts. The successive approximations $(u_n(x, t), v_n(x, t))$ of the solution (u, v) will be obtained using λ_1, λ_2 and the functions $u_0(x, t), v_0(x, t)$ which must at least satisfy the initial conditions.

Consequently, the exact solution will be the limit

$$\begin{cases} \lim\limits_{n \to +\infty} u_n(x, t) = u(x, t) \\ \lim\limits_{n \to +\infty} v_n(x, t) = v(x, t). \end{cases} \tag{18}$$

3. Examples

3.1. Example 1

Consider the problem with initial values of the following system of Cauchy-type partial differential equations:

$$\begin{cases} {}^c D_t^{\alpha} u(x, t) = 3u(x, t) + 2 - v(x, t) \dfrac{\partial u(x, t)}{\partial x} \\ {}^c D_t^{\alpha} v(x, t) = -3v(x, t) + 2 + u(x, t) \dfrac{\partial v(x, t)}{\partial x} \\ u(x, 0) = e^{2x} \\ v(x, 0) = e^{-2x} \end{cases} \tag{19}$$

with $0 < \alpha \leq 1$ and $\begin{cases} x \in \mathbb{R} \\ t \geq 0. \end{cases}$

3.1.1. Resolution using the SBA method

Consider the state system of (19)

$$\begin{cases} {}^c D_t^{\alpha} u(x, t) = 3u(x, t) + 2 - v(x, t) \dfrac{\partial u(x, t)}{\partial x} \\ {}^c D_t^{\alpha} v(x, t) = -3v(x, t) + 2 + u(x, t) \dfrac{\partial v(x, t)}{\partial x}. \end{cases} \tag{20}$$

Putting $L(\cdot) = {}^c D_t^{\alpha}(\cdot)$ and

$$\begin{cases} N_1(u(x, t), v(x, t)) = 2 - v(x, t) \dfrac{\partial u(x, t)}{\partial x} \\ N_2(u(x, t), v(x, t)) = 2 + u(x, t) \dfrac{\partial v(x, t)}{\partial x}. \end{cases} \tag{21}$$

We have

$$\begin{cases} {}^cD_t^{\alpha}u(x, t) = 3u(x, t) + N_1(u(x, t), v(x, t)) \\ {}^cD_t^{\alpha}v(x, t) = -3v(x, t) + N_2(u(x, t), v(x, t)). \end{cases} \tag{22}$$

Applying $L^{-1}(\cdot) = I_t^{\alpha}(\cdot)$ to (22), we get

$$\begin{cases} u(x, t) = u(x, 0) + I_t^{\alpha}(3u(x, t)) + I_t^{\alpha}(N_1(u(x, t), v(x, t))) \\ v(x, t) = v(x, 0) - I_t^{\alpha}(3v(x, t)) + I_t^{\alpha}(N_2(u(x, t), v(x, t))) \end{cases} \tag{23}$$

either

$$\begin{cases} u(x, t) = e^{2x} + I_t^{\alpha}(3u(x, t)) + I_t^{\alpha}(N_1(u(x, t), v(x, t))) \\ v(x, t) = e^{-2x} - I_t^{\alpha}(3v(x, t)) + I_t^{\alpha}(N_2(u(x, t), v(x, t))) \end{cases} \tag{24}$$

which is the canonical form of Adomian.

Applying the method of successive approximations to (24), we get

$$\begin{cases} u^k(x, t) = e^{2x} + 3I_t^{\alpha}(u^k(x, t)) + I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))), \ k \geq 1 \\ v^k(x, t) = e^{-2x} - 3I_t^{\alpha}(v^k(x, t)) + I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))), \ k \geq 1. \end{cases} \tag{25}$$

Thus, we obtain the following SBA algorithm

$$\begin{cases} \begin{cases} u_0^k(x, t) = e^{2x} + I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ u_{n+1}^k(x, t) = 3I_t^{\alpha}(u_n^k(x, t)), \quad \forall n \geq 0 \end{cases}, \quad \forall k \geq 1 \\ \begin{cases} v_0^k(x, t) = e^{-2x} + I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v_{n+1}^k(x, t) = -3I_t^{\alpha}(v_n^k(x, t)), \quad \forall n \geq 0 \end{cases}, \quad \forall k \geq 1. \end{cases} \tag{26}$$

The solution to each step is given by

$$u^k(x, t) = \sum_{n=0}^{\infty} u_n^k(x, t), \quad k = 1, 2, 3, ... \tag{27}$$

$$v^k(x, t) = \sum_{n=0}^{\infty} v_n^k(x, t), \quad k = 1, 2, 3, ... \tag{28}$$

At step $k = 1,$ the algorithm (26) becomes

$$\begin{cases} \begin{cases} u_0^1(x, t) = e^{2x} + I_t^{\alpha}(N_1(u^0(x, t), v^0(x, t))) \\ u_{n+1}^1(x, t) = 3I_t^{\alpha}(u_n^1(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = e^{-2x} + I_t^{\alpha}(N_2(u^0(x, t), v^0(x, t))) \\ v_{n+1}^1(x, t) = -3I_t^{\alpha}(v_n^1(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{29}$$

From (29), we apply Picard's principle. We look for u^0 and v^0 such that $N_1(u^0(x, t), v^0(x, t)) = N_2(u^0(x, t), v^0(x, t)) = 0.$ Choosing $u^0(x, t) = v^0(x, t) = 0,$ the algorithm (29) becomes

$$\begin{cases} \begin{cases} u_0^1(x, t) = e^{2x} \\ u_{n+1}^1(x, t) = 3I_t^{\alpha}(u_n^1(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = e^{-2x} \\ v_{n+1}^1(x, t) = -3I_t^{\alpha}(v_n^1(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{30}$$

For $n = 0,$ we have

$$\begin{cases} u_1^1(x, t) = \dfrac{(3t^{\alpha})^1}{\Gamma(\alpha + 1)} e^{2x} \\ v_1^1(x, t) = \dfrac{(-3t^{\alpha})^1}{\Gamma(\alpha + 1)} e^{-2x}. \end{cases} \tag{31}$$

For $n = 1$

$$\begin{cases} u_2^1(x, t) = \dfrac{(3t^{\alpha})^2}{\Gamma(2\alpha + 1)} e^{2x} \\ v_2^1(x, t) = \dfrac{(-3t^{\alpha})^2}{\Gamma(2\alpha + 1)} e^{-2x}. \end{cases} \tag{32}$$

For $n = 2$

$$\begin{cases} u_3^1(x, t) = \dfrac{(3t^{\alpha})^3}{\Gamma(3\alpha + 1)} e^{2x} \\ v_3^1(x, t) = \dfrac{(-3t^{\alpha})^3}{\Gamma(3\alpha + 1)} e^{-2x}. \end{cases} \tag{33}$$

Recursively, we have

$$\begin{cases} u_0^1(x, t) = e^{2x} \\ u_1^1(x, t) = \frac{(3t^{\alpha})^1}{\Gamma(\alpha+1)} e^{2x} \\ u_2^1(x, t) = \frac{(3t^{\alpha})^2}{\Gamma(2\alpha+1)} e^{2x} \\ u_3^1(x, t) = \frac{(3t^{\alpha})^3}{\Gamma(3\alpha+1)} e^{2x} \\ \quad \vdots \\ u_n^1(x, t) = \frac{(3t^{\alpha})^n}{\Gamma(n\alpha+1)} e^{2x} \end{cases} \quad \text{and} \quad \begin{cases} v_0^1(x, t) = e^{-2x} \\ v_1^1(x, t) = \frac{(-3t^{\alpha})^1}{\Gamma(\alpha+1)} e^{-2x} \\ v_2^1(x, t) = \frac{(-3t^{\alpha})^2}{\Gamma(2\alpha+1)} e^{-2x} \\ v_3^1(x, t) = \frac{(-3t^{\alpha})^3}{\Gamma(3\alpha+1)} e^{-2x} \\ \quad \vdots \\ v_n^1(x, t) = \frac{(-3t^{\alpha})^n}{\Gamma(n\alpha+1)} e^{-2x}. \end{cases} \tag{34}$$

The solution of the problem at step $k = 1$ is

$$\begin{cases} u^1(x, t) = e^{2x}\left(\sum_{n=0}^{\infty} \frac{(3t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^{2x} E_{\alpha}(3t^{\alpha}) \\ v^1(x, t) = e^{-2x}\left(\sum_{n=0}^{\infty} \frac{(-3t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^{-2x} E_{\alpha}(-3t^{\alpha}), \end{cases} \tag{35}$$

where $E_{\alpha}(3t^{\alpha})$ and $E_{\alpha}(-3t^{\alpha})$ is the Mittag-Leffler functions.

At step $k = 2$ the algorithm (26) becomes

$$\begin{cases} \begin{cases} u_0^2(x, t) = e^{2x} + I_t^{\alpha}(N_1(u^1(x, t), v^1(x, t))) \\ u_{n+1}^2(x, t) = 3I_t^{\alpha}(u_n^2(x, t)), \ \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = e^{-2x} + I_t^{\alpha}(N_2(u^1(x, t), v^1(x, t))) \\ v_{n+1}^2(x, t) = -3I_t^{\alpha}(v_n^2(x, t)), \ \forall n \geq 0. \end{cases} \end{cases} \tag{36}$$

Under-mentioned, we calculate

$N_1(u^1(x, t), v^1(x, t))$ and $N_2(u^1(x, t), v^1(x, t))$

$$\begin{cases} N_1(u^1(x, t), v^1(x, t)) = 2 - v^1(x, t)\dfrac{\partial u^1(x, t)}{\partial x} \\ \qquad = 2 - (e^{-2x}E_\alpha(-3t^\alpha))\dfrac{\partial(e^{2x}E_\alpha(3t^\alpha))}{\partial x} \\ \qquad = 2 - 2E_\alpha(-3t^\alpha)E_\alpha(3t^\alpha) \\ \qquad = 2 - 2 \\ \qquad = 0 \end{cases}$$

and

$$\begin{cases} N_2(u^1(x, t), v^1(x, t)) = 2 + u^1(x, t)\dfrac{\partial v^1(x, t)}{\partial x} \\ \qquad = 2 + (e^{2x}E_\alpha(3t^\alpha))\dfrac{\partial(e^{-2x}E_\alpha(-3t^\alpha))}{\partial x} \\ \qquad = 2 - 2E_\alpha(3t^\alpha)E_\alpha(-3t^\alpha) \\ \qquad = 2 - 2 \\ \qquad = 0. \end{cases} \tag{37}$$

The algorithm (36) becomes

$$\begin{cases} \begin{cases} u_0^2(x, t) = e^{2x} \\ u_{n+1}^2(x, t) = 3I_t^\alpha(u_n^2(x, t)), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = e^{-2x} \\ v_{n+1}^2(x, t) = -3I_t^\alpha(v_n^2(x, t)), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{38}$$

The algorithm at step $k = 2$ is the same as the algorithm at step $k = 1$. So, we have

$$\begin{cases} u^2(x, t) = e^{2x}\left(\displaystyle\sum_{n=0}^{\infty}\frac{(3t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^{2x}E_\alpha(3t^\alpha) \\ v^2(x, t) = e^{-2x}\left(\displaystyle\sum_{n=0}^{\infty}\frac{(-3t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^{-2x}E_\alpha(-3t^\alpha). \end{cases} \tag{39}$$

Recursively, we have

$$\begin{cases} u^k(x, t) = e^{2x}\left(\sum_{n=0}^{\infty}\frac{(3t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^{2x}E_{\alpha}(3t^{\alpha}) \\ v^k(x, t) = e^{-2x}\left(\sum_{n=0}^{\infty}\frac{(-3t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^{-2x}E_{\alpha}(-3t^{\alpha}). \end{cases} \tag{40}$$

The solution of the problem is

$$\begin{cases} u(x, t) = \lim\limits_{k\to+\infty} u^k(x, t) = e^{2x}E_{\alpha}(3t^{\alpha}) \\ v(x, t) = \lim\limits_{k\to+\infty} v^k(x, t) = e^{-2x}E_{\alpha}(-3t^{\alpha}), \end{cases} \tag{41}$$

where $E_{\alpha}(\cdot)$ is the Mittag-Leffler function with one parameter α.

The solution of the problem for $\alpha = 1$ is

$$\begin{cases} u(x, t) = e^{2x}E_1(3t) = e^{3t+2x} \\ v(x, t) = e^{-2x}E_1(-3t) = e^{-3t-2x}. \end{cases} \tag{42}$$

3.1.2. Solving with the variational iteration method

The correction functional applied to the system (19) is expressed as follows:

$$\begin{cases} u_{n+1}(x, t) \\ = u_n(x, t) + I_t^{\alpha}\left[\lambda_1(x, t)\left({}^cD_t^{\alpha}u_n(x, t) + \tilde{v}_n(x, t)\frac{\partial\tilde{u}_n(x, t)}{\partial x} - 3\tilde{u}_n(x, t) - 2\right)\right] \\ v_{n+1}(x, t) \\ = v_n(x, t) + I_t^{\alpha}\left[\lambda_2(x, t)\left({}^cD_t^{\alpha}v_n(x, t) - \tilde{u}_n(x, t)\frac{\partial\tilde{v}_n(x, t)}{\partial x} + 3\tilde{v}_n(x, t) - 2\right)\right]. \end{cases} \tag{43}$$

To estimate the values of the Lagrange multipliers, we use the approximation of the following functional correction

$$\begin{cases} u_{n+1}(x,t) \\ = u_n(x,t) + \int_0^t \left[\lambda_1(x,s) \left(\frac{\partial^m u_n(x,s)}{\partial s^m} + \tilde{v}_n(x,s) \frac{\partial \tilde{u}_n(x,s)}{\partial x} - 3\tilde{u}_n(x,s) - 2 \right) \right] ds \\ v_{n+1}(x,t) \\ = v_n(x,t) + \int_0^t \left[\lambda_2(x,s) \left(\frac{\partial^m v(x,s)}{\partial s^m} - \tilde{u}_n(x,s) \frac{\partial \tilde{v}_n(x,s)}{\partial x} + 3\tilde{v}_n(x,s) - 2 \right) \right] ds, \end{cases} \tag{44}$$

where $m - 1 < \alpha \leq m.$

To determine the Lagrange multipliers, we multiply two sides of the equations of the previous system by the corrector such that $\delta\tilde{u}_n = 0$ and $\delta\tilde{v}_n = 0.$ Thus, we have:

$$\begin{cases} \delta u_{n+1}(x,t) \\ = \delta u_n(x,t) + \int_0^t \left[\lambda_1(x,s) \delta \left(\frac{\partial^m u_n(x,s)}{\partial s^m} + \tilde{v}_n(x,s) \frac{\partial \tilde{u}_n(x,s)}{\partial x} - 3\tilde{u}_n(x,s) - 2 \right) \right] ds \\ \delta v_{n+1}(x,t) \\ = \delta v_n(x,t) + \int_0^t \left[\lambda_2(x,s) \delta \left(\frac{\partial^m v(x,s)}{\partial s^m} - \tilde{u}_n(x,s) \frac{\partial \tilde{v}_n(x,s)}{\partial x} + 3\tilde{v}_n(x,s) - 2 \right) \right] ds. \end{cases} \tag{45}$$

Subsequently, we have

$$\begin{cases} \delta u_{n+1}(x,\, t) = \delta u_n(x,\, t) + \int_0^t \lambda_1(x,\, s) \delta \left(\frac{\partial^m u_n(x,\, s)}{\partial s^m} \right) ds \\ \delta v_{n+1}(x,\, t) = \delta v_n(x,\, t) + \int_0^t \lambda_2(x,\, s) \delta \left(\frac{\partial^m v_n(x,\, s)}{\partial s^m} \right) ds. \end{cases} \tag{46}$$

We know that $0 < \alpha \leq 1,$ which implies that $m = 1,$

$$\begin{cases} \delta u_{n+1}(x,\, t) = \delta u_n(x,\, t) + \int_0^t \lambda_1(x,\, s) \delta \left(\frac{\partial u_n(x,\, s)}{\partial s} \right) ds \\ \delta v_{n+1}(x,\, t) = \delta v_n(x,\, t) + \int_0^t \lambda_2(x,\, s) \delta \left(\frac{\partial v(x,\, s)}{\partial s} \right) ds. \end{cases} \tag{47}$$

Integrating the above system by parts with respect to the variable *t*, we have:

$$\begin{cases} \delta u_{n+1}(x, t) = (1 + \lambda_1(x, t))\delta u_n(x, t) + \int_0^t \frac{\partial \lambda_1(x, s)}{\partial s} \delta u_n(x, s) ds \\ \delta v_{n+1}(x, t) = (1 + \lambda_2(x, t))\delta v_n(x, t) + \int_0^t \left(\frac{\partial \lambda_2(x, s)}{\partial s} \right) \delta v(x, s) ds. \end{cases} \tag{48}$$

The existence of extremum conditions for $u_{n+1}(x, t)$ and $v_{n+1}(x, t)$ implies that $\delta u_{n+1}(x, t) = 0$ and $\delta v_{n+1}(x, t) = 0,$ it follows:

$$\begin{cases} (1 + \lambda_1(x, t))\delta u_n(x, t) + \int_0^t \frac{\partial \lambda_1(x, s)}{\partial s} \delta u_n(x, s) ds = 0 \\ (1 + \lambda_2(x, t))\delta v_n(x, t) + \int_0^t \left(\frac{\partial \lambda_2(x, s)}{\partial s} \right) \delta v(x, s) ds = 0. \end{cases} \tag{49}$$

This gives us:

$$\begin{cases} \begin{cases} (1 + \lambda_1(x, t)) = 0 \\ \frac{\partial \lambda_1(x, s)}{\partial s} = 0 \end{cases} \\ \begin{cases} (1 + \lambda_2(x, t)) = 0 \\ \frac{\partial \lambda_2(x, s)}{\partial s} = 0 \end{cases} \end{cases} \Rightarrow \lambda_1(x, t) = \lambda_2(x, t) = -1. \tag{50}$$

Substituting the Lagrange multipliers $\lambda_1(x, t)$ and $\lambda_2(x, t)$ by their respective values in the system (43), we obtain the following system:

$$\begin{cases} u_{n+1}(x, t) \\ = u_n(x, t) - I_t^\alpha \left[\left({}^c D_t^\alpha u_n(x, t) + v_n(x, t) \frac{\partial u_n(x, t)}{\partial x} - 3u_n(x, t) - 2 \right) \right] \\ v_{n+1}(x, t) \\ = v_n(x, t) - I_t^\alpha \left[\left({}^c D_t^\alpha v_n(x, t) - u_n(x, t) \frac{\partial v_n(x, t)}{\partial x} + 3v_n(x, t) - 2 \right) \right]. \end{cases} \tag{51}$$

From (51), we can obtain $u_1; ...; u_n$ and $v_1; ...; v_n$ as follows:

$$\begin{cases} u_0(x, t) = u(x, 0) = e^{2x} \\ u_1(x, t) = e^{2x}\left[1 + \frac{3t^\alpha}{\Gamma(\alpha+1)}\right] \\ u_2(x, t) = e^{2x}\left[1 + \frac{3t^\alpha}{\Gamma(\alpha+1)} + \frac{(3t^\alpha)^2}{\Gamma(2\alpha+1)}\right] + \frac{18\Gamma(2\alpha+1)}{\Gamma^2(2\alpha+1)\Gamma(3\alpha+1)} t^{3\alpha} \\ u_3(x, t) = e^{2x}\left[1 + \frac{3t^\alpha}{\Gamma(\alpha+1)} + \frac{(3t^\alpha)^2}{\Gamma(2\alpha+1)} + \frac{(3t^\alpha)^3}{\Gamma(3\alpha+1)}\right] + \text{noise term} \\ \quad\vdots \\ u_3(x, t) = e^{2x}\left[1 + \frac{3t^\alpha}{\Gamma(\alpha+1)} + \frac{(3t^\alpha)^2}{\Gamma(2\alpha+1)} + \frac{(3t^\alpha)^3}{\Gamma(3\alpha+1)} + \cdots + \frac{(3t^\alpha)^n}{\Gamma(n\alpha+1)}\right] \\ \qquad + \text{noise term} \end{cases} \tag{52}$$

or

$$\begin{cases} v_0(x, t) = v(x, 0) = e^{-2x} \\ v_1(x, t) = e^{-2x}\left[1 - \frac{3t^\alpha}{\Gamma(\alpha+1)}\right] \\ v_2(x, t) = e^{-2x}\left[1 - \frac{3t^\alpha}{\Gamma(\alpha+1)} + \frac{(-3t^\alpha)^2}{\Gamma(2\alpha+1)}\right] + \frac{18\Gamma(2\alpha+1)}{\Gamma^2(\alpha+1)\Gamma(3\alpha+1)} t^{3\alpha} \\ v_3(x, t) = e^{-2x}\left[1 - \frac{3t^\alpha}{\Gamma(\alpha+1)} + \frac{(-3t^\alpha)^2}{\Gamma(2\alpha+1)} + \frac{(-3t^\alpha)^3}{\Gamma(3\alpha+1)}\right] + \text{noise term} \\ \quad\vdots \\ v_n(x, t) = e^{-2x}\left[1 + \frac{(-3t^\alpha)^1}{\Gamma(\alpha+1)} + \frac{(-3t^\alpha)^2}{\Gamma(2\alpha+1)} + \frac{(-3t^\alpha)^3}{\Gamma(3\alpha+1)} + \cdots + \frac{(-3t^\alpha)^n}{\Gamma(n\alpha+1)}\right] \\ \qquad + \text{noise term} \end{cases} \tag{53}$$

This gives us

$$\begin{cases} u_n(x, t) = e^{2x}\left(\displaystyle\sum_{k=0}^{n} \frac{(3t^{\alpha})^k}{\Gamma(k\alpha+1)}\right) + \text{noise term} \\ v_n(x, t) = e^{-2x}\left(\displaystyle\sum_{k=0}^{n} \frac{(-3t^{\alpha})^k}{\Gamma(k\alpha+1)}\right) + \text{noise term.} \end{cases} \tag{54}$$

Thus, the solution of (19) is given as follows:

$$\begin{cases} u(x, t) = \lim\limits_{n\to+\infty} u_n(x, t) = e^{2x}\displaystyle\sum_{k=0}^{+\infty} \frac{(3t^{\alpha})^k}{\Gamma(k\alpha+1)} = e^{2x}E_{\alpha}(3t^{\alpha}) \\ v(x, t) = \lim\limits_{n\to+\infty} v_n(x, t) = e^{-2x}\displaystyle\sum_{k=0}^{+\infty} \frac{(-3t^{\alpha})^k}{\Gamma(k\alpha+1)} = e^{-2x}E_{\alpha}(-3t^{\alpha}). \end{cases} \tag{55}$$

The solution of the example for $\alpha = 1$ is

$$\begin{cases} u(x, t) = e^{2x}E_1(3t) = e^{3t+2x} \\ v(x, t) = e^{-2x}E_1(-3t) = e^{-3t-2x}. \end{cases} \tag{56}$$

3.1.3. Partial conclusion

Both solutions are similar.

3.2. Example 2

Consider the problem with initial values of the following system of Cauchy-type partial differential equations:

$$\begin{cases} {}^c D_t^{\alpha} u(x, t) = \dfrac{\partial^2 u(x, t)}{\partial x^2} + 2u(x, t)\dfrac{\partial u(x, t)}{\partial x} - \dfrac{\partial}{\partial x}(u(x, t)v(x, t)) \\ {}^c D_t^{\alpha} v(x, t) = \dfrac{\partial^2 v(x, t)}{\partial x^2} + 2v(x, t)\dfrac{\partial v(x, t)}{\partial x} - \dfrac{\partial}{\partial x}(u(x, t)v(x, t)) \\ u(x, 0) = \sin 2x \\ v(x, 0) = \sin 2x, \end{cases} \tag{57}$$

where $t \geq 0$, $1 < \alpha \leq 1$, ${}^c D_t^{\alpha}(\cdot)$ is the derivative in Caputo sense, $I_t^{\alpha}(\cdot)$ is the integral in the Riemann-Liouville sense.

3.2.1. Resolution using the SBA method

Consider the state system of (57)

$$\begin{cases} {}^{c}D_t^{\alpha}u(x,t) = \dfrac{\partial^2 u(x,t)}{\partial x^2} + N_1(u(x,t), v(x,t)) \\ {}^{c}D_t^{\alpha}v(x,t) = \dfrac{\partial^2 v(x,t)}{\partial x^2} + N_2(u(x,t), v(x,t)), \end{cases} \tag{58}$$

where

$$\begin{cases} N_1(u(x,t), v(x,t)) = 2u(x,t)\dfrac{\partial u(x,t)}{\partial x} - \dfrac{\partial}{\partial x}(u(x,t)v(x,t)) \\ N_2(u(x,t), v(x,t)) = 2v(x,t)\dfrac{\partial v(x,t)}{\partial x} - \dfrac{\partial}{\partial x}(u(x,t)v(x,t)). \end{cases} \tag{59}$$

Applying the Riemann-Liouville integral $I_t^{\alpha}(\cdot)$ to (58), we have

$$\begin{cases} I_t^{\alpha}({}^{c}D_t^{\alpha}u(x,t)) = I_t^{\alpha}\left(\dfrac{\partial^2 u(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u(x,t), v(x,t))) \\ I_t^{\alpha}({}^{c}D_t^{\alpha}v(x,t)) = I_t^{\alpha}\left(\dfrac{\partial^2 v(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u(x,t), v(x,t))). \end{cases} \tag{60}$$

This gives the following canonical form

$$\begin{cases} u(x,t) = u(x,0) + I_t^{\alpha}\left(\dfrac{\partial^2 u(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u(x,t), v(x,t))) \\ v(x,t) = v(x,0) + I_t^{\alpha}\left(\dfrac{\partial^2 v(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u(x,t), v(x,t))) \end{cases} \tag{61}$$

either

$$\begin{cases} u(x,t) = \sin 2x + I_t^{\alpha}\left(\dfrac{\partial^2 u(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u(x,t), v(x,t))) \\ v(x,t) = \sin 2x + I_t^{\alpha}\left(\dfrac{\partial^2 v(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u(x,t), v(x,t))). \end{cases} \tag{62}$$

Applying the method of successive approximations, we obtain

$$\begin{cases} u^k(x,t) = \sin 2x + I_t^{\alpha}\left(\dfrac{\partial^2 u^k(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u^{k-1}(x,t), v^{k-1}(x,t))) \\ v^k(x,t) = \sin 2x + I_t^{\alpha}\left(\dfrac{\partial^2 v^k(x,t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u^{k-1}(x,t), v^{k-1}(x,t))) \end{cases}, \forall k \geq 1. \tag{63}$$

The solution at each stage is

$$u^k(x, t) = \sum_{n=0}^{\infty} u_n^k(x, t), \; k = 1, 2, 3, \ldots \tag{64}$$

and

$$v^k(x, t) = \sum_{n=0}^{\infty} v_n^k(x, t), \; k = 1, 2, 3, \ldots. \tag{65}$$

Substituting (64) and (65) into (63), we obtain the following SBA algorithm

$$\begin{cases} \begin{cases} u_0^k(x, t) = \sin(2x) + I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ u_{n+1}^k(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^k(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases}, \; k \geq 1 \\ \begin{cases} v_0^k(x, t) = \sin(2x) + I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v_{n+1}^k(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^k(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases}, \; k \geq 1 \end{cases} \tag{66}$$

At step $k = 1$, we have

$$\begin{cases} \begin{cases} u_0^1(x, t) = \sin(2x) + I_t^{\alpha}(N_1(u^0(x, t), v^0(x, t))) \\ u_{n+1}^1(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = \sin(2x) + I_t^{\alpha}(N_2(u^0(x, t), v^0(x, t))) \\ v_{n+1}^1(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{67}$$

From (67), we apply Picard's principle, we look for u^0 and v^0 such that

$$N_1(u^0(x, t), v^0(x, t)) = N_2(u^0(x, t), v^0(x, t)) = 0.$$

Choosing $u^0(x, t) = v^0(x, t) = 0,$ the algorithm (67) becomes

$$\begin{cases} \begin{cases} u_0^1(x, t) = \sin(2x) \\ u_{n+1}^1(x, t) = I_t^\alpha\left(\dfrac{\partial^2 u_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = \sin(2x) \\ v_{n+1}^1(x, t) = I_t^\alpha\left(\dfrac{\partial^2 v_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{68}$$

This gives us

$$\begin{cases} u_0^1(x, t) = \sin 2x \\ u_1^1(x, t) = \dfrac{(-4t^\alpha)^1}{\Gamma(\alpha+1)}\sin 2x \\ u_2^1(x, t) = \dfrac{(-4t^\alpha)^2}{\Gamma(2\alpha+1)}\sin 2x \\ \quad\vdots \\ u_n^1(x, t) = \dfrac{(-4t^\alpha)^n}{\Gamma(n\alpha+1)}\sin 2x \end{cases} \text{ and } \begin{cases} v_0^1(x, t) = \sin 2x \\ v_1^1(x, t) = \dfrac{(-4t^\alpha)^1}{\Gamma(\alpha+1)}\sin 2x \\ v_2^1(x, t) = \dfrac{(-4t^\alpha)^2}{\Gamma(2\alpha+1)}\sin 2x \\ \quad\vdots \\ v_n^1(x, t) = \dfrac{(-4t^\alpha)^n}{\Gamma(n\alpha+1)}\sin 2x. \end{cases} \tag{69}$$

The solution of the problem at step $k = 1$ is

$$\begin{cases} u^1(x, t) = \sin 2x\left(\displaystyle\sum_{n=0}^{\infty}\frac{(-4t^\alpha)^n}{\Gamma(n\alpha+1)}\right) = \sin 2x E_\alpha(-4t^\alpha) \\ v^1(x, t) = \sin 2x\left(\displaystyle\sum_{n=0}^{\infty}\frac{(-4t^\alpha)^n}{\Gamma(n\alpha+1)}\right) = \sin 2x E_\alpha(-4t^\alpha), \end{cases} \tag{70}$$

where $E_\alpha(-4t^\alpha)$ is the Mittag-Leffler function.

At step $k = 2$, we have

$$\begin{cases} \begin{cases} u_0^2(x, t) = \sin(2x) + I_t^\alpha(N_1(u^1(x, t), v^1(x, t))) \\ u_{n+1}^2(x, t) = I_t^\alpha\left(\dfrac{\partial^2 u_n^2(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = \sin(2x) + I_t^\alpha(N_2(u^1(x, t), v^1(x, t))) \\ v_{n+1}^2(x, t) = I_t^\alpha\left(\dfrac{\partial^2 v_n^2(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{71}$$

Thus, we calculate $N_1(u^1(x, t), v^1(x, t))$ and $N_2(u^1(x, t), v^1(x, t))$

$$\begin{cases} N_1(u^1(x, t), v^1(x, t)) \\ = 2u^1(x, t)\dfrac{\partial u^1(x, t)}{\partial x} - \dfrac{\partial}{\partial x}(u^1(x, t)v^1(x, t)) \\ = 2(\sin 2xE_\alpha(-4t^\alpha))\dfrac{\partial(\sin 2xE_\alpha(-4t^\alpha))}{\partial x} \\ \quad - \dfrac{\partial}{\partial x}(\sin 2xE_\alpha(-4t^\alpha)\sin 2xE_\alpha(-4t^\alpha)) \\ = (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) - \dfrac{\partial}{\partial x}(\sin^2 2xE_\alpha^2(-4t^\alpha)) \\ = (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) - (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) \\ = 0 \end{cases} \tag{72}$$

$$\begin{cases} N_2(u^1(x, t), v^1(x, t)) \\ = 2v^1(x, t)\dfrac{\partial v^1(x, t)}{\partial x} - \dfrac{\partial}{\partial x}(u^1(x, t)v^1(x, t)) \\ = 2(\sin 2xE_\alpha(-4t^\alpha))\dfrac{\partial(\sin 2xE_\alpha(-4t^\alpha))}{\partial x} \\ \quad - \dfrac{\partial}{\partial x}(\sin 2xE_\alpha(-4t^\alpha)\sin 2xE_\alpha(-4t^\alpha)) \\ = (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) - \dfrac{\partial}{\partial x}(\sin^2 2xE_\alpha^2(-4t^\alpha)) \\ = (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) - (4\sin 2x\cos 2x)E_\alpha^2(-4t^\alpha) \\ = 0. \end{cases} \tag{73}$$

The algorithm (71) becomes

$$\begin{cases} \begin{cases} u_0^2(x,\ t) = \sin(2x) \\ u_{n+1}^2(x,\ t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^2(x,\ t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x,\ t) = \sin(2x) \\ v_{n+1}^2(x,\ t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^2(x,\ t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{74}$$

The algorithm at step $k = 2$ is the same as the algorithm at step $k = 1$. So, we have

$$\begin{cases} u^2(x,\ t) = \sin 2x\left(\sum_{n=0}^{\infty} \dfrac{(-4t^{\alpha})^n}{\Gamma(n\alpha + 1)}\right) = \sin 2x E_{\alpha}(-4t^{\alpha}) \\ v^2(x,\ t) = \sin 2x\left(\sum_{n=0}^{\infty} \dfrac{(-4t^{\alpha})^n}{\Gamma(n\alpha + 1)}\right) = \sin 2x E_{\alpha}(-4t^{\alpha}). \end{cases} \tag{75}$$

Recursively, we have

$$\begin{cases} u^k(x,\ t) = \sin 2x\left(\sum_{n=0}^{\infty} \dfrac{(-4t^{\alpha})^n}{\Gamma(n\alpha + 1)}\right) = \sin 2x E_{\alpha}(-4t^{\alpha}) \\ v^k(x,\ t) = \sin 2x\left(\sum_{n=0}^{\infty} \dfrac{(-4t^{\alpha})^n}{\Gamma(n\alpha + 1)}\right) = \sin 2x E_{\alpha}(-4t^{\alpha}). \end{cases} \tag{76}$$

The solution of the problem is

$$\begin{cases} u(x,\ t) = \lim_{k \to +\infty} u^k(x,\ t) = \sin 2x E_{\alpha}(-4t^{\alpha}) \\ v(x,\ t) = \lim_{k \to +\infty} v^k(x,\ t) = \sin 2x E_{\alpha}(-4t^{\alpha}), \end{cases} \tag{77}$$

The solution of the problem for $\alpha = 1$ is

$$\begin{cases} u(x,\ t) = \sin 2x E_1(-4t) = e^{-4t} \sin 2x \\ v(x,\ t) = \sin 2x E_1(-4t) = e^{-4t} \sin 2x. \end{cases} \tag{78}$$

3.2.2. Solving with the variational iteration method

The correction functionals for (57) is written as follows:

$$\begin{cases} u_{n+1}(x,t) \\ = u_n(x,t) + I_t^{\alpha}\left[\lambda_1(x,t)\left({}^cD_t^{\alpha}u_n(x,t) - \frac{\partial^2\tilde{u}_n(x,t)}{\partial x^2} - 2\tilde{u}_n(x,t)\frac{\partial\tilde{u}_n(x,t)}{\partial x} + \frac{\partial}{\partial x}(\tilde{u}_n(x,t)\tilde{v}_n(x,t))\right)\right] \\ v_{n+1}(x,t) \\ = v_n(x,t) + I_t^{\alpha}\left[\lambda_2(x,t)\left({}^cD_t^{\alpha}v_n(x,t) - \frac{\partial^2\tilde{v}_n(x,t)}{\partial x^2} - 2\tilde{v}_n(x,t)\frac{\partial\tilde{v}_n(x,t)}{\partial x} + \frac{\partial}{\partial x}(\tilde{u}_n(x,t)\tilde{v}_n(x,t))\right)\right]. \end{cases} \tag{79}$$

To estimate the values of the Lagrange multipliers, we use the approximation of the following correction functional

$$\begin{cases} u_{n+1}(x,t) \\ = u_n(x,t) + \int_0^t\left[\lambda_1(x,s)\left(\frac{\partial^m u_n(x,s)}{\partial s^m} - \frac{\partial^2\tilde{u}_n(x,s)}{\partial x^2} - 2\tilde{u}_n(x,s)\frac{\partial\tilde{u}_n(x,s)}{\partial x} + \frac{\partial}{\partial x}(\tilde{u}_n(x,s)\tilde{v}_n(x,s))\right)\right]ds \\ v_{n+1}(x,t) \\ = v_n(x,t) + \int_0^t\left[\lambda_2(x,s)\left(\frac{\partial^m v_n(x,s)}{\partial s^m} - \frac{\partial^2\tilde{v}_n(x,s)}{\partial x^2} - 2\tilde{v}_n(x,s)\frac{\partial\tilde{v}_n(x,s)}{\partial x} + \frac{\partial}{\partial x}(\tilde{u}_n(x,s)\tilde{v}_n(x,s))\right)\right]ds, \end{cases} \tag{80}$$

where $m - 1 < \alpha \le m$.

The stationary conditions are given by

$$\begin{cases} \begin{cases} (1 + \lambda_1(x, t)) = 0 \\ \frac{\partial\lambda_1(x, s)}{\partial s} = 0 \end{cases} \\ \begin{cases} (1 + \lambda_2(x, t)) = 0 \\ \frac{\partial\lambda_2(x, s)}{\partial s} = 0. \end{cases} \end{cases} \tag{81}$$

So that

$$\lambda_1(x, t) = \lambda_2(x, t) = -1. \tag{82}$$

Substituting these values of the Lagrange multipliers into the functionals

(79) gives the iteration formulas

$$\begin{cases} u_{n+1}(x,t) \\ = u_n(x,t) - I_t^{\alpha}\left[\left({}^{c}D_t^{\alpha} u_n(x,t) - \dfrac{\partial^2 u_n(x,t)}{\partial x^2} - 2u_n(x,t)\dfrac{\partial u_n(x,t)}{\partial x} + \dfrac{\partial}{\partial x}(u_n(x,t)v_n(x,t))\right)\right] \\ v_{n+1}(x,t) \\ = v_n(x,t) - I_t^{\alpha}\left[\left({}^{c}D_t^{\alpha} v_n(x,t) - \dfrac{\partial^2 v_n(x,t)}{\partial x^2} - 2v_n(x,t)\dfrac{\partial v_n(x,t)}{\partial x} + \dfrac{\partial}{\partial x}(u_n(x,t)v_n(x,t))\right)\right]. \end{cases} \tag{83}$$

The zeroth approximations $u_0(x, t) = \sin 2x$ and $v_0(x, t) = \sin 2x$ are selected using the given initial conditions. Therefore, we obtain the following successive approximations

$$\begin{cases} u_0(x,t) = v_0(x,t) = \sin(2x) \\ u_1(x,t) = v_1(x,t) = \left[1 - \dfrac{4t^{\alpha}}{\Gamma(\alpha+1)}\right]\sin(2x) \\ u_2(x,t) = v_2(x,t) = \left[1 - \dfrac{4t^{\alpha}}{\Gamma(\alpha+1)} + \dfrac{16t^{2\alpha}}{\Gamma(2\alpha+1)}\right]\sin(2x) \\ u_3(x,t) = v_3(x,t) = \left[1 - \dfrac{4t^{\alpha}}{\Gamma(\alpha+1)} + \dfrac{16t^{2\alpha}}{\Gamma(2\alpha+1)} - \dfrac{64t^{3\alpha}}{\Gamma(3\alpha+1)}\right]\sin(2x) \\ \qquad\vdots \\ u_n(x,t) = v_n(x,t) = \left[1 + \dfrac{(-4t^{\alpha})^1}{\Gamma(\alpha+1)} + \dfrac{(-4t^{\alpha})^2}{\Gamma(2\alpha+1)} + \dfrac{(-4t^{\alpha})^3}{\Gamma(3\alpha+1)} + \cdots + \dfrac{(-4t^{\alpha})^n}{\Gamma(n\alpha+1)}\right] \\ \qquad \cdot \sin(2x). \end{cases} \tag{84}$$

This gives

$$\begin{cases} u_n(x,t) = \sin(2x)\left(\displaystyle\sum_{k=0}^{n} \dfrac{(-4t^{\alpha})^k}{\Gamma(k\alpha+1)}\right) \\ v_n(x,t) = \sin(2x)\left(\displaystyle\sum_{k=0}^{n} \dfrac{(-4t^{\alpha})^k}{\Gamma(k\alpha+1)}\right). \end{cases} \tag{85}$$

The solution is formulated as follows:

$$\begin{cases} u(x, t) = \lim\limits_{n \to +\infty} u_n(x, t) = \left(\sum\limits_{k=0}^{+\infty} \dfrac{(-4t^{\alpha})^k}{\Gamma(k\alpha + 1)} \right) \sin(2x) = E_{\alpha}(-4t^{\alpha}) \sin(2x) \\ v(x, t) = \lim\limits_{n \to +\infty} v_n(x, t) = \left(\sum\limits_{k=0}^{+\infty} \dfrac{(-4t^{\alpha})^k}{\Gamma(k\alpha + 1)} \right) \sin(2x) = E_{\alpha}(-4t^{\alpha}) \sin(2x), \end{cases} \tag{86}$$

where $E_{\alpha}(-4t^{\alpha})$ is the Mittag-Leffler function with one parameter α.

The solution of the problem for $\alpha = 1$ is

$$\begin{cases} u(x, t) = \sin 2x E_1(-4t) = e^{-4t} \sin 2x \\ v(x, t) = \sin 2x E_1(-4t) = e^{-4t} \sin 2x. \end{cases} \tag{87}$$

3.2.3. Partial conclusion

Both solutions are similar.

3.3. Example 3

Consider the problem with initial values of the following system of Cauchy-type partial differential equations:

$$\begin{cases} {}^c D_t^{\alpha} u(x, t) = \dfrac{\partial^2 u(x, t)}{\partial x^2} + \dfrac{\partial u(x, t)}{\partial x} \dfrac{\partial v(x, t)}{\partial x} - u(x, t) v(x, t) \\ {}^c D_t^{\alpha} v(x, t) = \dfrac{\partial^2 v(x, t)}{\partial x^2} + \left(\dfrac{\partial v(x, t)}{\partial x} \right)^2 - u^2(x, t) \\ u(x, 0) = e^x \\ v(x, 0) = e^x, \end{cases} \tag{88}$$

where

$$\begin{cases} 0 < \alpha \le 1 \\ x \in \mathbb{R} \\ t \ge 0 \end{cases}$$

and $t \ge 0$ and $x \in \mathbb{R}$.

3.3.1. Solution using the SBA method

Consider the state system of (88)

$$\begin{cases} {}^{c}D_t^{\alpha}u(x, t) = \dfrac{\partial^2 u(x, t)}{\partial x^2} + N_1(u(x, t), v(x, t)) \\ {}^{c}D_t^{\alpha}v(x, t) = \dfrac{\partial^2 v(x, t)}{\partial x^2} + N_2(u(x, t), v(x, t)), \end{cases} \tag{89}$$

where

$$\begin{cases} N_1(u(x, t), v(x, t)) = \dfrac{\partial u(x, t)}{\partial x}\dfrac{\partial v(x, t)}{\partial x} - u(x, t)v(x, t) \\ N_2(u(x, t), v(x, t)) = \left(\dfrac{\partial v(x, t)}{\partial x}\right)^2 - u^2(x, t). \end{cases} \tag{90}$$

Applying the Riemann-Liouville integral $I_t^{\alpha}(\cdot)$ member by member to (89), we obtain the following canonical form

$$\begin{cases} u(x, t) = u(x, 0) + I_t^{\alpha}\left(\dfrac{\partial^2 u(x, t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u(x, t), v(x, t))) \\ v(x, t) = v(x, 0) + I_t^{\alpha}\left(\dfrac{\partial^2 v(x, t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u(x, t), v(x, t))) \end{cases} \tag{91}$$

$\Leftrightarrow$

$$\begin{cases} u(x, t) = e^x + I_t^{\alpha}\left(\dfrac{\partial^2 u(x, t)}{\partial x^2}\right) + I_t^{\alpha}(N_1(u(x, t), v(x, t))) \\ v(x, t) = e^x + I_t^{\alpha}\left(\dfrac{\partial^2 v(x, t)}{\partial x^2}\right) + I_t^{\alpha}(N_2(u(x, t), v(x, t))). \end{cases} \tag{92}$$

Applying the successive approximation method to (92), we obtain

$$\begin{cases} u^k(x, t) = e^x + I_t^{\alpha}\left(\dfrac{\partial^2 u^k(x, t)}{\partial x^2}\right) \\ \qquad\qquad + I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v^k(x, t) = e^x + I_t^{\alpha}\left(\dfrac{\partial^2 v^k(x, t)}{\partial x^2}\right) \\ \qquad\qquad + I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \end{cases}, \quad \forall k \geq 1. \tag{93}$$

The solution to each step is:

$$\begin{cases} u^k(x, t) = \sum_{n=0}^{\infty} u_n^k(x, t) \\ v^k(x, t) = \sum_{n=0}^{\infty} v_n^k(x, t) \end{cases}, \quad k = 1, 2, 3, \dots. \tag{94}$$

Substituting (94) into (93) gives the following SBA algorithm:

$$\begin{cases} \begin{cases} u_0^k(x, t) = e^x + I_t^{\alpha}(N_1(u^{k-1}(x, t), v^{k-1}(x, t))) \\ u_{n+1}^k(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^k(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^k(x, t) = e^x + I_t^{\alpha}(N_2(u^{k-1}(x, t), v^{k-1}(x, t))) \\ v_{n+1}^k(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^k(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \end{cases}, \quad \forall k \geq 1. \tag{95}$$

At step $k = 1,$ we have

$$\begin{cases} \begin{cases} u_0^1(x, t) = e^x + I_t^{\alpha}(N_1(u^0(x, t), v^0(x, t))) \\ u_{n+1}^1(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^1(x, t) = e^x + I_t^{\alpha}(N_2(u^0(x, t), v^0(x, t))) \\ v_{n+1}^1(x, t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^1(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{96}$$

From (96), we apply Picard's principle. We look for u^0 and v^0 such that $N_1(u^0(x, t), v^0(x, t)) = N_2(u^0(x, t), v^0(x, t)) = 0.$ Choosing $u^0(x, t) = v^0(x, t) = 0,$ the algorithm (96) becomes

$$\begin{cases}\begin{cases}u_0^1(x,t) = e^x \\ u_{n+1}^1(x,t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^1(x,t)}{\partial x^2}\right), \quad \forall n \geq 0\end{cases} \\ \begin{cases}v_0^1(x,t) = e^x \\ v_{n+1}^1(x,t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^1(x,t)}{\partial x^2}\right), \quad \forall n \geq 0.\end{cases}\end{cases} \tag{97}$$

This gives us

$$\begin{cases}\begin{cases}u_0^1(x,t) = e^x \\ u_n^1(x,t) = \dfrac{(t^{\alpha})^n}{\Gamma(n\alpha+1)} e^x, \quad \forall n \geq 1\end{cases} \\ \begin{cases}v_0^1(x,t) = e^x \\ v_n^1(x,t) = \dfrac{(t^{\alpha})^n}{\Gamma(n\alpha+1)} e^x, \quad \forall n \geq 1.\end{cases}\end{cases} \tag{98}$$

The solution to the problem at step $k = 1$ is

$$\begin{cases}u^1(x,t) = e^x\left(\displaystyle\sum_{n=0}^{\infty}\frac{(t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^x E_{\alpha}(t^{\alpha}) \\ v^1(x,t) = e^x\left(\displaystyle\sum_{n=0}^{\infty}\frac{(t^{\alpha})^n}{\Gamma(n\alpha+1)}\right) = e^x E_{\alpha}(t^{\alpha}).\end{cases} \tag{99}$$

At step $k = 2$, we have

$$\begin{cases}\begin{cases}u_0^2(x,t) = e^x + I_t^{\alpha}(N_1(u^1(x,t), v^1(x,t))) \\ u_{n+1}^2(x,t) = I_t^{\alpha}\left(\dfrac{\partial^2 u_n^2(x,t)}{\partial x^2}\right), \quad \forall n \geq 0\end{cases} \\ \begin{cases}v_0^2(x,t) = e^x + I_t^{\alpha}(N_2(u^1(x,t), v^1(x,t))) \\ v_{n+1}^2(x,t) = I_t^{\alpha}\left(\dfrac{\partial^2 v_n^2(x,t)}{\partial x^2}\right), \quad \forall n \geq 0.\end{cases}\end{cases} \tag{100}$$

Below we calculate $N_1(u^1(x, t), v^1(x, t))$ and $N_2(u^1(x, t), v^1(x, t))$

$$\begin{cases} \begin{cases} N_1(u^1(x, t), v^1(x, t)) = \dfrac{\partial(e^x E_\alpha(t^\alpha))}{\partial x}\dfrac{\partial(e^x E_\alpha(t^\alpha))}{\partial x} \\ \qquad\qquad\qquad - (e^x E_\alpha(t^\alpha))(e^x E_\alpha(t^\alpha)) \\ \qquad = e^{2x}E_\alpha^2(t^\alpha) - e^{2x}E_\alpha^2(t^\alpha) = 0 \end{cases} \\ \begin{cases} N_2(u^1(x, t), v^1(x, t)) = \left(\dfrac{\partial(e^x E_\alpha(t^\alpha))}{\partial x}\right)^2 - (e^x E_\alpha(t^\alpha))^2 \\ \qquad = e^{2x}E_\alpha^2(t^\alpha) - e^{2x}E_\alpha^2(t^\alpha) \\ \qquad = 0. \end{cases} \end{cases} \tag{101}$$

So at step $k = 2$, we have

$$\begin{cases} \begin{cases} u_0^2(x, t) = e^x \\ u_{n+1}^2(x, t) = I_t^\alpha\left(\dfrac{\partial^2 u_n^2(x, t)}{\partial x^2}\right), \quad \forall n \geq 0 \end{cases} \\ \begin{cases} v_0^2(x, t) = e^x \\ v_{n+1}^2(x, t) = I_t^\alpha\left(\dfrac{\partial^2 v_n^2(x, t)}{\partial x^2}\right), \quad \forall n \geq 0. \end{cases} \end{cases} \tag{102}$$

We obtain the same algorithm as in step $k = 1$, which gives

$$\begin{cases} \begin{cases} u_0^2(x, t) = e^x \\ u_n^2(x, t) = \dfrac{(t^\alpha)^n}{\Gamma(n\alpha + 1)} e^x, \quad \forall n \geq 1 \end{cases} \\ \begin{cases} v_0^2(x, t) = e^x \\ v_n^2(x, t) = \dfrac{(t^\alpha)^n}{\Gamma(n\alpha + 1)} e^x, \quad \forall n \geq 1. \end{cases} \end{cases} \tag{103}$$

The solution to the problem at step $k = 2$ is

$$\begin{cases} u^2(x, t) = e^x\left(\sum_{n=0}^{\infty} \frac{(t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^x E_\alpha(t^\alpha) \\ v^2(x, t) = e^x\left(\sum_{n=0}^{\infty} \frac{(t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^x E_\alpha(t^\alpha). \end{cases} \tag{104}$$

Then, for all $k = 3$ by recurrence we can obtain for each step the solution

$$\begin{cases} u^k(x, t) = e^x\left(\sum_{n=0}^{\infty} \frac{(t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^x E_\alpha(t^\alpha) \\ v^k(x, t) = e^x\left(\sum_{n=0}^{\infty} \frac{(t^\alpha)^n}{\Gamma(n\alpha + 1)}\right) = e^x E_\alpha(t^\alpha). \end{cases} \tag{105}$$

The solution of the problem is

$$\begin{cases} u(x, t) = \lim_{k \to +\infty} u^k(x, t) = e^x E_\alpha(t^\alpha) \\ v(x, t) = \lim_{k \to +\infty} v^k(x, t) = e^x E_\alpha(t^\alpha), \end{cases} \tag{106}$$

where $E_\alpha(t^\alpha)$ is the Mittag-Leffler function.

The solution of the problem for $\alpha = 1$ is

$$\begin{cases} u(x, t) = e^x E_1(t) = e^{x+t} \\ v(x, t) = e^x E_1(t) = e^{x+t}. \end{cases} \tag{107}$$

3.3.2. Solving with the variational iteration method

According to the theory of the variational iteration method for fractional-order partial differential equations, the correction functional associated with the system (88) is given by the following system:

$$\begin{cases} u_{n+1}(x, t) \\ = u_n(x, t) + I_t^{\alpha}\left[\lambda_1(x, t)\left({}^cD_t^{\alpha}u_n(x, t) - \dfrac{\partial^2\tilde{u}_n(x, t)}{\partial x^2} - \dfrac{\partial\tilde{u}_n(x, t)}{\partial x}\dfrac{\partial\tilde{v}_n(x, t)}{\partial x} + \tilde{u}_n(x, t)\tilde{v}_n(x, t)\right)\right] \\ v_{n+1}(x, t) \\ = v_n(x, t) + I_t^{\alpha}\left[\lambda_2(x, t)\left({}^cD_t^{\alpha}v_n(x, t) - \dfrac{\partial^2\tilde{v}_n(x, t)}{\partial x^2} - \left(\dfrac{\partial\tilde{v}_n(x, t)}{\partial x}\right)^2 + (\tilde{u}_n(x, t))^2\right)\right]. \end{cases} \tag{108}$$

To estimate the values of the Lagrange multipliers, we use the approximation of the following correction functional

$$\begin{cases} u_{n+1}(x, t) \\ = u_n(x, t) + \displaystyle\int_0^t\left[\lambda_1(x, s)\left({}^cD_t^{\alpha}u_n(x, s) - \dfrac{\partial^2\tilde{u}_n(x, s)}{\partial x^2} - \dfrac{\partial\tilde{u}_n(x, s)}{\partial x}\dfrac{\partial\tilde{v}_n(x, s)}{\partial x} + \tilde{u}_n(x, s)\tilde{v}_n(x, s)\right)\right]ds \\ v_{n+1}(x, t) \\ = v_n(x, t) + \displaystyle\int_0^t\left[\lambda_2(x, s)\left({}^cD_t^{\alpha}v_n(x, s) - \dfrac{\partial^2\tilde{v}_n(x, s)}{\partial x^2} - \left(\dfrac{\partial\tilde{v}_n(x, s)}{\partial x}\right)^2 + (\tilde{u}_n(x, s))^2\right)\right]ds, \end{cases} \tag{109}$$

where $m - 1 < \alpha \leq m$.

The stationary conditions are given by

$$\begin{cases} \begin{cases} (1 + \lambda_1(x, t)) = 0 \\ \dfrac{\partial\lambda_1(x, s)}{\partial s} = 0 \end{cases} \\ \begin{cases} (1 + \lambda_2(x, t)) = 0 \\ \dfrac{\partial\lambda_2(x, s)}{\partial s} = 0. \end{cases} \end{cases} \tag{110}$$

So that

$$\lambda_1(x, t) = \lambda_2(x, t) = -1. \tag{111}$$

Substituting these values of the Lagrange multipliers into (108) gives the iteration formulas

$$\begin{cases} u_{n+1}(x, t) \\ = u_n(x, t) - I_t^{\alpha}\left({}^c D_t^{\alpha} u_n(x, t) - \dfrac{\partial^2 u_n(x, t)}{\partial x^2} - \dfrac{\partial u_n(x, t)}{\partial x}\dfrac{\partial v_n(x, t)}{\partial x} + u_n(x, t) v_n(x, t)\right) \\ v_{n+1}(x, t) \\ = v_n(x, t) - I_t^{\alpha}\left({}^c D_t^{\alpha} v_n(x, t) - \dfrac{\partial^2 v_n(x, t)}{\partial x^2} - \left(\dfrac{\partial v_n(x, t)}{\partial x}\right)^2 + (u_n(x, t))^2\right). \end{cases} \tag{112}$$

The zeroth approximations $u_0(x, t) = e^x$ and $v_0(x, t) = e^x$ are selected by using the given initial conditions. Therefore, we obtain the following successive approximations

$$\begin{cases} u_0(x, t) = v_0(x, t) = e^x \\ u_1(x, t) = v_1(x, t) = \left[1 + \dfrac{t^{\alpha}}{\Gamma(\alpha+1)}\right] e^x \\ u_2(x, t) = v_2(x, t) = \left[1 + \dfrac{t^{\alpha}}{\Gamma(\alpha+1)} + \dfrac{t^{2\alpha}}{\Gamma(2\alpha+1)}\right] e^x \\ u_3(x, t) = v_3(x, t) = \left[1 + \dfrac{t^{\alpha}}{\Gamma(\alpha+1)} + \dfrac{t^{2\alpha}}{\Gamma(2\alpha+1)} + \dfrac{t^{3\alpha}}{\Gamma(3\alpha+1)}\right] e^x \\ \qquad \vdots \\ u_n(x, t) = v_n(x, t) \\ \qquad = \left[1 + \dfrac{(t^{\alpha})^1}{\Gamma(\alpha+1)} + \dfrac{(t^{\alpha})^2}{\Gamma(2\alpha+1)} + \dfrac{(t^{\alpha})^3}{\Gamma(3\alpha+1)} + \cdots + \dfrac{(t^{\alpha})^n}{\Gamma(n\alpha+1)}\right] e^x. \end{cases} \tag{113}$$

This gives us

$$\begin{cases} u_n(x, t) = e^x \left(\displaystyle\sum_{k=0}^{n} \frac{(t^{\alpha})^k}{\Gamma(k\alpha+1)} \right) \\ v_n(x, t) = e^x \left(\displaystyle\sum_{k=0}^{n} \frac{(t^{\alpha})^k}{\Gamma(k\alpha+1)} \right). \end{cases} \tag{114}$$

The solution is given in the form below:

$$\begin{cases} u(x, t) = \lim_{n\to+\infty} u_n(x, t) = \left(\sum_{k=0}^{+\infty} \frac{(t^\alpha)^k}{\Gamma(k\alpha+1)}\right)e^x = e^x E_\alpha(t^\alpha) \\ v(x, t) = \lim_{n\to+\infty} v_n(x, t) = \left(\sum_{k=0}^{+\infty} \frac{(t^\alpha)^k}{\Gamma(k\alpha+1)}\right)e^x = e^x E_\alpha(t^\alpha). \end{cases} \tag{115}$$

The solution of the problem for $\alpha = 1$ is

$$\begin{cases} u(x, t) = e^x E_1(t) = e^{x+t} \\ v(x, t) = e^x E_1(t) = e^{x+t}. \end{cases} \tag{116}$$

3.3.3. Partial conclusion

Both methods: SBA and VIM give the same solution.

4. Conclusion

The main aim of this article is to carry out a comparative study between the Somé Blaise Abbo (SBA) method and the variational iteration method (VIM).

In this study, the SBA method provides the components of the exact solution when these components follow the summation given in (10) and (14). However, the variational iteration method (VIM) gives several successive approximations using the iteration of the correction functional. In addition, the SBA method, which is based on the construction of a solution at each step in the form of a series, approximates the exact solution of the given problem. The terms of this series are determined using an iterative scheme known as the SBA algorithm. In contrast, the variational iteration method (VIM) requires the evaluation of the Lagrange multiplier λ. In the examples presented in this article, a closed-form solution is obtained using each of the methods. In sum, these three examples, illustrate well the power and efficiency of both methods, for both methods are so powerful and efficient that they yield more accurate approximations and closed-form solutions if they exist. We can see from these three examples that the two methods give us similar results.

Acknowledgement

We thank the anonymous referees for their comments and feedback on earlier version of this document.

References

[1] Aleksandra Ziemiska-Stolarska and Jerzy Skrzypski, Review of mathematical models of water quality, Ecol. Chem. Eng. S. 19(2) (2012), 197-211.

[2] H. W. Streeter and E. B. Phelps, A study of the pollution and natural purification of the Ohio River, Public health office of United States, Public Health Bulletin, Vol. 146, Washington, DC: US Government Press, 1925.

[3] J. R. James, D. K. Frederik and J. H. Taylor, Use of expert-systems programming techniques for the design of lead-lag compensators, IEE Proceedings D (Control Theory and Applications) 134(3) (1987), 137-144. https://doi-org/10.49/ip-d-1987-0023.

[4] Ebeezer Bongah, Paul Agbekpornu and Canan Unlud, Mathematical modeling of transmission of water pollution, Journal of Prime Research in Mathematics 17(2) (2021), 20-38.

[5] F. B. Agusto and O. M. Bamigbola, Numerical treatment of the mathematical models for water pollution, Journal of Mathematics and Statistics 3(4) (2007), 172-180.

[6] Jonathan King, Reza Ahmadian and Roger A. Falconer, Hydro-epidemiological modelling of bacterial transport and decay in nearshore coastal waters, Water Research 196 (2021), 117049. https://doi.org/10.1016/j.watres.2021.117049.

[7] H. Sing, D. K. Mishra, R. N. Das and S. Kumar, A brief review-mathematical modeling on water pollution and its effects on aquatic species, Resembling Bagnati River, International Journal of Applied Environmental Sciences 17(1) (2022), 67-72. https://doi.org/10.37622/IJAES/17.1.2022.67-72.

[8] Chelsea J. Weiskerger and Mantha S. Phanikumar, Numerical modeling of microbial fate and transport in natural waters: Review and Implications for normal and extreme Storn Events, Water 12 (2020), 1876-1898. DOI: 10.3390/w12071876.

[9] N. Pochai, S. Tangmanee, L. J. Crane and J. J. H. Miller, A Mathematical Model of Water Pollution Control using the Finite Element Method, In PAMM: Proceeding in Applied Mathematics and Mechanics, Berlin: Wiley-VCH Verlag 6(1) (2006), 755-756.

[10] A. Ebrahinzadeh, E. Hashemizadeh and R. Mirabbasi, A numerical solution of the mathematical models for water pollution by shifted Jacobi polynomials, TWMS J. App. and Eng. Math. 15(2) (2025), 298-308.

[11] Tsegaye Simon and Purnachandra Rao Koya, Modeling and numerical simulation of river pollution using diffusion-reaction equation, American Journal of Applied Mathematics 3(6) (2015), 335-340. Doi: 10:11648/j.ajan:20150306:24.

[12] Putsadee Poruphol, Porpattama Hammachukiattikul, Rajarathinan Vadivel Salaudeen Ab-dulwaheed Adebago and Saratha Sathasivan, Compartmental equations hybrid model for modelling water pollution transmission, Journal of Advanced Research in Fluid Mechanics and Thermal Sciences 115(1) (2024), 30-50.

[13] Steven Kent, Doddi Yadianto, Gao Cheng, Finna Fitriana and Wang Qian, Water quality modelling with industrial and domestic point source pollution: a study case of Cikakembang River, Majalaya District, Journal of the Civil Engineering Forum 10(2) (2024), 151-162. Doi: 10.22146/joof:11807.

[14] Y. N. Anjam, M. ur Yavuz, Mu Rahman and A. Batod, Analysis of a fractional pollution model in a system of three interconnecting lakes, AIMS Biophysics 10(2) (2023), 220-240. DOI: 10.3934/bio-phy.2023014.

[15] Soner Aydinlink, A novel approach for fractional model of water pollution management, Journal of Industrial and Management Optimization 20(12) (2024), 3617-3627.

[16] Keit B. Oldhan and Jerome Spanier, The Fractional Calculus, Theory and Applications of Differentiation and Integration Tonarbitrary Order, Dover Publications, 2006.

[17] K. Abbaoui and Y. Cherruault, Convergence of Adomian method applied to differential equations, Math. Comput. Modeling 28(5) (1994), 103-109.

[18] K. Abbaoui, Les fondements de la méthode décompositionnelle d'Adomian et application à la résolution de problèmes issus de la biologie et de la medicine, Thèse de doctorat de l, Université Paris VI, 1995.

[19] K. Abbaoui and Y. Cherruault, The decomposition method applied to non-linear equation, Math. Comput. Modeling 20(9) (1994), 60-73.

[20] Blaise SOME, SBA method for solving mathematical models in the environment, European University Editions, July 2018.

[21] Joseph Bonazebi Yindoula, Gabriel Bissanga, Pare Youssouf, Francis Bassono and Blaise SOME, Application of the Adomian Decomposition Method (ADM) and the SOME BLAISE ABBO (SBA) method to solving the diffusion-reaction equations, Advances in Theoretical and Applied Mathematics 9(2) (2014), 97-104.

[22] Stevy Mikamona Mayembo, Joseph Bonazebi-Yindoula, Youssouf Pare and Gabriel Bissanga, Application of the SBA method to solve the nonlinear biological population models, European Journal of Pure and Applied Mathematics 12(3) (2019), 1096-1105.

[23] J. H. He, Variational iteration method a kind of nonlinear analytical technique: some examples, Int. J. Nonlinear Mech. 34 (1999), 699-708.

[24] A. M. Wazwaz, A comparison between the variational iteration method and Adomian decomposition method, J. Comput. Appl. Math. 207(1) (2007), 129-136.

[25] A. M. Wazwaz, The variational iteration method for solving linear and nonlinear systems of PDEs, Comput. Math. Appl. 54 (2007), 895-902.

[26] A. M. Wazwaz, The variational iteration method: A reliable analytic tool for solving linear and nonlinear wave equations, Comput. Math. Appl. 54 (2007), 926-932.