AIMS Mathematics, 10(12): 29263–29284.
DOI: 10.3934/math.20251286
Received: 07 October 2025
Revised: 21 November 2025
Accepted: 04 December 2025
Published: 12 December 2025

https://www.aimspress.com/journal/Math

Research article

Bell polynomial-based semi-discretization approach for the extended Fisher-Kolmogorov equations

M. J. Huntul*

Department of Mathematics, College of Science, Jazan University, P.O. Box. 114, Jazan 45142, Saudi Arabia

* **Correspondence:** Email: mhantool@jazanu.edu.sa.

Abstract: This work introduces an integrated numerical framework for obtaining numerical approximations to the extended Fisher-Kolmogorov (EFK) equation model. The methodology entails a two-stage process: initial linearization of the governing equation through a Taylor series approach, followed by the application of a spectral collocation method utilizing Bell polynomials to resolve the resultant linear system. Comprehensive theoretical considerations, including a detailed error estimate in the weighted L^2-norm, are established. The numerical investigation, comprising three illustrative test cases, confirms the scheme's computational efficiency and demonstrates a marked improvement in accuracy when compared to current state-of-the-art results and exact solutions.

Keywords: Bell polynomials; Taylor approach; extended Fisher-Kolmogorov equation; collocation nodes; error analysis
Mathematics Subject Classification: 41A10, 41A58, 65L60, 65M12

1. Introduction

The extended Fisher-Kolmogorov (EFK) equation represents a significant fourth-order generalization of the traditional Fisher-Kolmogorov equation, exhibiting rich nonlinear dynamics and pattern formation phenomena. Early work by [1] established its connection to spatial chaos in physical systems, while [2, 3] systematically analyzed front propagation mechanisms and stability criteria in unstable states. Particularly, [3] provided crucial distinctions between linear and nonlinear marginal stability regimes. The bistable nature of EFK systems was experimentally validated by [4], demonstrating how propagating fronts lead to spontaneous pattern generation. From a rigorous mathematical perspective, [5] developed topological shooting methods to prove existence theorems for kink-type solutions, revealing the equation's complex solution structure. These collective studies underscore the EFK equation's fundamental role in modeling phase transitions, reaction-diffusion

processes, and other phenomena where higher-order spatial derivatives dominate the system dynamics.

We focus on obtaining approximate solutions for a family of partial differential equations (PDEs) that incorporate a nonlinear term denoted by $H(W)$, defined as follows:

$$W_\tau + \lambda\, W_{xxxx} - W_{xx} - H(W) = g(\tau, x), \quad (\tau, x) \in D_\tau \times D_x, \tag{1.1}$$

where $H(W) = W(1 - W^2)$, $D_x := [x_l, x_r]$, and $D_\tau := [0, T]$ with the initial conditions

$$W(\tau = 0, x) = \alpha(x), \quad x \in D_x. \tag{1.2}$$

The boundary conditions are prescribed as follows:

$$\begin{cases} W(\tau, x_l) = \beta_l(\tau), & W(\tau, x_r) = \beta_r(\tau), \\ W_{xx}(\tau, x_l) = 0, & W_{xx}(\tau, x_r) = 0, \end{cases} \qquad \tau \in D_\tau, \tag{1.3}$$

where the functions $\beta_l(\tau)$, $\beta_r(\tau)$, and $\alpha(x)$ are some familiar functions.

The extended Fisher-Kolmogorov equation has motivated the development of diverse numerical approaches. For spline-based methods, [6] developed an orthogonal cubic spline collocation approach that maintains stability while handling the fourth-order derivative terms. Also, [7] employed truncation techniques to establish the existence of periodic solutions, providing theoretical foundations for numerical approximations. Finite difference schemes have been extensively studied, with [8] constructing a second-order accurate difference scheme that balances computational efficiency and accuracy. Previous work on the Fisher-Kolmogorov equation includes the study by Mittal and Dahiya [9], who applied a quintic B-spline based differential quadrature method to a semi-linear class of the problem. Alternative discretization approaches include the boundary integral equation method by [10], which reformulates the problem to reduce dimensionality. Spectral and wavelet methods have shown promise, particularly the Gegenbauer wavelet collocation method by [11] that effectively handles two-dimensional EFK problems. Mesh-free approaches have gained attention, including the interpolating element-free Galerkin method by [12], which was specifically applied to brain tumor dynamics modeling. In addition, [13] introduced a finite pointset method based on mixed formulation that avoids mesh generation entirely. More recently, [14, 15] designed a three-level linearized and compact high-order difference scheme that achieves improved temporal convergence rates. Advanced time integration techniques include the convex splitting backward differentiation formula-2 (BDF2) approach utilizing variable time-steps by [16] that maintains energy stability. For finite element approaches, [17] developed two-grid nonconforming finite element methods (FEMs) that achieve unconditional superconvergence through careful error analysis. The authors in [18] proposed an improved collocation algorithm that demonstrates enhanced accuracy in solving the EFK equation through optimized basis functions and node selection. Recent studies continue to explore this area, including the work of Thottoli et al. [19], who applied a quintic trigonometric B-spline collocation technique to the EFK equation.

The numerical analysis of PDEs has evolved into a rich and diverse landscape, with methodologies tailored to the specific demands of different physical models, dimensionalities, and computational challenges. This is evident in the wide array of techniques developed for complex problems, ranging from phase-field models like the EFK equation to dynamic fracture mechanics. For instance, the EFK

equation has been tackled using finite difference methods [14, 15, 20], finite element schemes [17, 21], and a variety of B-spline-based collocation techniques [18, 19, 22, 23]. More recently, advanced mesh-free methods have gained prominence for their flexibility in handling complex geometries and moving boundaries. This is exemplified by the work of Qu et al. [24], who developed a stable numerical framework for long-time dynamic crack analysis, and Sun et al. [25], who applied a high-order generalized finite difference method (GFDM) with an enhanced Krylov deferred correction technique to analyze bi-material interfacial cracks. Beyond these, the numerical toolkit continues to expand with specialized approaches such as hybrid block methods for biological models like the FitzHugh-Nagumo equation [26]; robust techniques for inverse and free boundary problems [27]; and novel machine learning-based adaptive spectral methods for unbounded domains [28]. This collective progress underscores the dynamic and interdisciplinary nature of contemporary computational mathematics.

The motivation for integrating the Taylor series expansion in time with advanced polynomial approximations in space is rooted in the pursuit of high-order, stable, and efficient numerical solvers for time-dependent PDEs. While traditional low-order methods are often adequate for simple problems, they suffer from significant numerical dissipation and dispersion errors when applied to complex nonlinear phenomena, such as those described by the Korteweg-de Vries, Burgers, or Hunter-Saxton equations [29–31]. The Taylor-Galerkin methodology, pioneered by Donea et al. [32], provides a robust framework to derive temporally accurate schemes by systematically incorporating higher-order time derivatives directly from the PDE itself, effectively reducing the temporal truncation error. When this powerful time-stepping approach is synergistically coupled with a high-order spatial discretization such as a spectral collocation, the Galerkin method utilizing basis functions like B-splines [30], or Bell polynomials, the result is a fully discrete scheme that excels in both temporal and spatial accuracy. This combination, as demonstrated in [33, 34], leverages the strengths of each component: The Taylor method provides a precise time-marching structure, while the spectral polynomial approximation ensures excellent spatial resolution and minimal phase error, leading to superior long-term numerical stability and convergence properties that are essential for capturing the intricate behavior of nonlinear wave dynamics and transport problems.

This study successfully designs, analyzes, and implements a novel computational framework, the Taylor-Bell matrix collocation method, for obtaining accurate numerical solutions to the EFK equation. The proposed hybrid approach effectively combines the strengths of the Taylor series for temporal linearization with the superior approximation capabilities of Bell polynomials for spatial discretization. The primary achievements of the current investigation can be summarized as follows:

- A robust linearization process is established by applying the Taylor series expansion to the high-order, nonlinear EFK equation, systematically reducing it to a more tractable system of linear equations.
- The spectral collocation method, utilizing a basis of Bell polynomials, is demonstrated to be highly effective in solving the derived linear system. The well-established properties of these polynomials, including their smoothness and spectral accuracy, are crucial for this task.
- A rigorous theoretical analysis is provided, confirming the convergence of the method and establishing an error bound in the weighted L^2-norm, thus providing a solid mathematical foundation for the approach.
- The numerical experiments are conducted on three distinct test cases unequivocally demonstrate

the high accuracy and efficacy of the suggested scheme. The results show excellent agreement with existing exact and numerical solutions available in the literature, often achieving superior precision.

The outline of the ongoing research investigation is as follows: Section 2 details the application of the Taylor methodology and the linearization process for the EFK model (1.1). In Section 3, we introduce the Bell polynomials and their essential properties, followed by a discussion on their convergence and error analysis. The complete algorithm of the proposed Taylor-Bell matrix collocation method is elaborated in Section 4. The efficacy of the suggested scheme is then demonstrated through three numerical examples in Section 5. A summary of findings and concluding discussions are provided in Section 6.

2. Temporal advancement strategy

In the current portion, let us begin by applying the Taylor expansion to discretize the given class of PDEs given in (1.1) in the temporal direction. To achieve this, we introduce a uniform partitioning of the interval D_τ into $\mathcal{M}$ subintervals, defined by the nodes

$$\Gamma_\Delta : \{\tau_m = m\Delta\tau, \quad 0 \le m \le \mathcal{M}\},$$

where $\Delta\tau = \tau_{m+1} - \tau_m$ denotes the timestep size, for $m \in \mathbb{M} := \{0, 1, \ldots, \mathcal{M} - 1\}$. We denote by W^m the numerical approximation to the actual true solution $W(\tau, x)$ at the discrete time level τ_m. Specifically, we define

$$W^m \equiv W^m(x) := W(\tau_m, x), \quad x \in D_x.$$

The EFK Eq (1.1), evaluated at the time frame τ_m, takes the form

$$W^m_\tau = W^m_{xx} - \lambda\, W^m_{xxxx} - H(W^m) + g^m(x), \tag{2.1}$$

where $g^m(x) := g(\tau_m, x)$. Utilizing the Taylor expansion form about τ_m, we derive the subsequent relation:

$$W^m_\tau = \frac{1}{\Delta\tau}\left(W^{m+1} - W^m\right) - \frac{1}{2}\,(\Delta\tau\, W^m_{\tau\tau}) + O(\Delta\tau^2). \tag{2.2}$$

Differentiating Eq (2.1) with respect to τ and subsequently evaluating at $\tau = \tau_m$ yields

$$\Delta\tau\, W^m_{\tau\tau} = (W^{m+1}_{xx} - W^m_{xx}) - \lambda(W^{m+1}_{xxxx} - W^m_{xxxx}) - H'(W^m)\left(W^{m+1} - W^m\right), \tag{2.3}$$

in which all terms of the form W^m_τ have been replaced by their first-order difference quotient, namely $\frac{W^{m+1}-W^m}{\Delta\tau}$, on the right-hand side. Substituting (2.3) into (2.2) and matching the left-hand sides with those in (2.1), we simplify and regroup terms to obtain the discretized linearized version of (1.1). The resulting expression is

$$\lambda\,\Delta\tau\, W^{m+1}_{xxxx}(x) - \Delta\tau\, W^{m+1}_{xx}(x) + [2 + \Delta\tau\, H'(W^m)]W^{m+1}(x) = \Xi^m(x), \quad m \in \mathbb{M}, \tag{2.4}$$

where

$$\Xi^m(x) := \Delta\tau\,\left[W^m_{xx}(x) - \lambda\, W^m_{xxxx}(x)\right] + W^m(x)\left[\Delta\tau\, H'(W^m) + 2\right] - 2\Delta\tau\, H(W^m) + \Delta\tau\left[g^m(x) + g^{m+1}(x)\right].$$

As part of the proposed method, we define the following set of coefficients:

$$\Theta_1^m(x) := \lambda\,\Delta\tau, \quad \Theta_2^m(x) := -\Delta\tau, \quad \Theta_3^m(x) := 2 + \Delta\tau\,H'(W^m).$$

In order to compute the approximate solution of (2.4), one first requires the expression $W^0(x)$, which is directly obtained from the initial condition $W^0(x) = W_0(x)$. In addition to this function, the fourth- and second-order derivatives of $W^0(x)$ also appear in the right-hand side function $\Xi^0(x)$. The boundary conditions (1.3) will be accordingly transformed. Thus, at $x = x_l$ and $x = x_r$, we have

$$\begin{cases} W^{m+1}(x_l) = \beta_l^{m+1} := \beta_l(\tau_{m+1}), & W^{m+1}(x_r) = \beta_r^{m+1} := \beta_r(\tau_{m+1}), \\ \frac{d^2}{dx^2}W^{m+1}(x_l) = 0, & \frac{d^2}{dx^2}W^{m+1}(x_r) = 0. \end{cases} \tag{2.5}$$

By setting $\sigma_m(x) := W^m(x)$ and considering the case where $H(W) = W^3 - W$, we need to solve the following set of equations for $m \in \mathbb{M}$:

$$\Theta_1^m(x)\,\sigma_{m+1}^{(4)}(x) + \Theta_2^m(x)\,\sigma_{m+1}''(x) + \Theta_3^m(x)\,\sigma_{m+1}(x) = \Xi^m(x), \quad m \in \mathbb{M}, \tag{2.6}$$

with $\sigma_0(x) = \alpha(x)$ and the subsequent boundary conditions:

$$\begin{cases} \sigma_{m+1}(x_l) = \beta_l^{m+1}, & \sigma_{m+1}(x_r) = \beta_r^{m+1}, \\ \sigma_{m+1}''(x_l) = 0, & \sigma_{m+1}''(x_r) = 0. \end{cases} \tag{2.7}$$

3. Analytical review of Bell polynomials: convergence perspectives

The main purpose of this portion is to review the definition of Bell polynomials (BPs) and discuss their main properties. We then analyze the convergence of BPs expansions.

3.1. Bell polynomials: A brief overview

In combinatorial mathematics, Bell polynomials form an important class of polynomials that were first systematically studied by Eric Temple Bell [35]. These polynomials are closely related to Stirling numbers (of the second kind) and serve as a fundamental tool in various areas of mathematics. For additional applications and detailed discussions, we refer to [36–38]. The sequence $\{\mathbb{B}_e^p(x)\}$ is the associated Sheffer sequence for the function $f(t) = \ln(1 + t)$. Consequently, the exponential generating function of these polynomials is given by

$$\sum_{p=0}^{\infty} \frac{t^p}{p!}\mathbb{B}_e^p(x) = e^{(e^t-1)x}.$$

The second-kind Stirling numbers, denoted by $\mathbb{S}_2(j, p)$, are defined as [39, Chap. 5]:

$$\mathbb{S}_2(j, p) := \frac{1}{j!}\sum_{i=1}^{j}(-1)^{j-i}\binom{j}{i}i^p, \quad 1 \le j \le p,$$

with $\mathbb{S}_2(j, p) = 0$ when $1 \le p < j$, $\mathbb{S}_2(0, 0) = 1$ and $\mathbb{S}_2(j, 0) = 0$ for $j \ge 1$. These numbers count the partitions of a set of size p into j non-empty subsets.

Definition 3.1. *The Bell polynomial BPs of degree p is given by*

$$\mathbb{B}_e^p(x) := \sum_{j=0}^{p} \mathbb{S}_2(j,p)\, x^j, \quad p \in \mathbb{N}, \tag{3.1}$$

with $\mathbb{B}_e^0(x) := 1$.

The first few Bell polynomials BPs are

$$\begin{aligned} &\mathbb{B}_e^1(x) = x, && \mathbb{B}_e^2(x) = x^2 + x, \\ &\mathbb{B}_e^3(x) = x^3 + 3x^2 + x, && \mathbb{B}_e^4(x) = x^4 + 6x^3 + 7x^2 + x, \\ &\mathbb{B}_e^5(x) = x^5 + 10x^4 + 25x^3 + 10x^2 + x, && \mathbb{B}_e^6(x) = x^6 + 15x^5 + 65x^4 + 90x^3 + 31x^2 + x. \end{aligned}$$

Some of key properties of BPs include:

- $\mathbb{B}_e^p(0) = 0$ for all $p \in \mathbb{N}$,
- $\mathbb{B}_e^p(1) = B_p$, where B_p are the Bell numbers ($B_0 = B_1 = 1$, $B_2 = 2$, $B_3 = 5$, $B_4 = 15$, etc.)

3.2. Matrix representation

For $P \in \mathbb{N}$, define the vector of $(P + 1)$ BPs as follows:

$$\boldsymbol{B}_P(x) := \begin{bmatrix} \mathbb{B}_e^0(x) & \mathbb{B}_e^1(x) & \dots & \mathbb{B}_e^P(x) \end{bmatrix}. \tag{3.2}$$

Thus, we have:

Lemma 3.2. *The vector* $\boldsymbol{B}_P(x)$ *can be expressed as*

$$\boldsymbol{B}_P(x) = \boldsymbol{\Phi}_P(x)\, \boldsymbol{N}_P, \tag{3.3}$$

where $\boldsymbol{\Phi}_P(x) = \begin{bmatrix} 1 & x & x^2 & \dots & x^P \end{bmatrix}$ *is the monomial basis vector and* $\boldsymbol{N}_P$ *is the* $(P + 1) \times (P + 1)$ *upper-triangular (nonsingular) matrix*

$$\boldsymbol{N}_P = \begin{bmatrix} 1 & \mathbb{S}_2(1,1) & \mathbb{S}_2(2,2) & \cdots & \mathbb{S}_2(P,P) \\ 0 & 1 & \mathbb{S}_2(2,1) & \cdots & \mathbb{S}_2(P,P-1) \\ 0 & 0 & 1 & \cdots & \mathbb{S}_2(P,P-2) \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}.$$

Proof. The result follows directly from Definition 3.1 by induction on P. □

3.3. Convergent and error bound of BPs

We aim to study the sequence of Bell polynomials BPs on the domain D_x. To analyze their convergence properties rigorously, we first construct an appropriate function space associated with D_x. Let the weight function be defined as $w(x) := \frac{1}{x_r - x_l}$ for simplicity. We then introduce the weighted L^2-space [40]

$$L_w^2(D_x) := \left\{ \sigma \colon D_x \to \mathbb{R} \;\middle|\; \sigma \text{ is measurable and } \|\sigma\|_w < \infty \right\},$$

equipped with the norm

$$\|\sigma\|_w := \left(\int_\Omega |\sigma(x)|^2 w(x)\, dx \right)^{1/2}.$$

This framework will allow us to systematically investigate the convergence behavior of the BPs.

Assume that a function $\sigma(x) \in L^2_w(D_x)$ is given. We can express $\sigma(x)$ as a series expansion in terms of the basis functions $\{\mathbb{B}^p_e(x)\}_{p=0}^{\infty}$ as

$$\sigma(x) = \sum_{p=0}^{\infty} \zeta_p \, \mathbb{B}^p_e(x), \quad x \in D_x. \tag{3.4}$$

Our primary objective is to determine the unknown coefficients ζ_p for $p \geq 0$.

For practical computations, we consider a finite-dimensional subspace $\mathcal{W}_P \subseteq L^2_w(D_x)$ defined as

$$\mathcal{W}_P := \text{Span}\langle \mathbb{B}^0_e(x), \mathbb{B}^1_e(x), \ldots, \mathbb{B}^P_e(x) \rangle.$$

Clearly, $\mathcal{W}_P$ is a closed, finite-dimensional subspace (of dimension $P+1$) and consequently complete in $L^2_w(D_x)$. This guarantees the existence of a unique best approximation $\sigma_\star(x) \in \mathcal{W}_P$ satisfying:

$$\|\sigma - \sigma_\star\|_w \leq \|\sigma - \phi\|_w, \quad \forall \phi \in \mathcal{W}_P.$$

As previously mentioned, we use only the first $(P+1)$ BPs to approximate $\sigma(x)$. It follows that

$$\sigma(x) \approx \sigma_P(x) := \sum_{p=0}^{P} \zeta_p \, \mathbb{B}^p_e(x), \quad x \in D_x. \tag{3.5}$$

We derive the following compact expression for the approximate solution $\sigma_P(x)$:

$$\sigma_P(x) = \boldsymbol{B}_P(x) \boldsymbol{Z}_P, \tag{3.6}$$

where $\boldsymbol{B}_P(x)$ is provided in (3.2). In addition, the unknowns ζ_p for $p = 0, 1, \ldots, P$ in a vector framework is defined by

$$\boldsymbol{Z}_P := \begin{bmatrix} \zeta_0 & \zeta_1 & \ldots & \zeta_P \end{bmatrix}^\top.$$

The subsequent outcome establishes that as P (the number of basis functions) increases, the approximation error between $\sigma(x)$ and its projection $\sigma_P(x)$ from (3.5) converges to zero. Let $\mathcal{E}_P(x) := \sigma(x) - \sigma_P(x)$ denote the approximation error.

Theorem 3.3. *Let $\sigma_P(x) = \boldsymbol{B}_P(x)\boldsymbol{Z}_P$ be the best approximation of $\sigma(x)$ from the subspace $\mathcal{W}_P$, where $\sigma(x) \in L^2_w(D_x) \cap C^{P+1}(D_x)$. Then, the error $E_P(x)$ converges to zero as $P \to \infty$ with the rate*

$$\|\mathcal{E}_P(x)\|_w \leq \frac{\sigma_\infty}{(P+1)!} L^{P+1}, \tag{3.7}$$

where $\sigma_\infty := \max_{x \in D_x} |\sigma^{(P+1)}|$, $L = x_r - x_l$.

Proof. Consider an arbitrary point $x^* \in D_x = (x_l, x_r)$. By Taylor's theorem, for any $x \in D_x$, there exists $\theta \in (x^*, x)$ such that

$$\sigma(x) = \sum_{\ell=0}^{P} \frac{\sigma^{(\ell)}(x^*)}{\ell!}(x - x^*)^\ell + \frac{\sigma^{(P+1)}(\theta)}{(P+1)!}(x - x^*)^{P+1} = \Sigma_P(x) + \mathcal{R}_P(x).$$

The remainder term $\mathcal{R}_P(x)$ satisfies

$$|\mathcal{R}_P(x)| = |\sigma(x) - \Sigma_P(x)| = \left|\frac{\sigma^{(P+1)}(\theta)}{(P+1)!}(x - x^*)^{P+1}\right| \leq \frac{\sigma_\infty}{(P+1)!} L^{P+1}. \tag{3.8}$$

We then utilize the fact that $\sigma_P(x)$ shows the finest approximation to $\sigma(x)$ out of $\mathcal{W}_P$. Based on the above discussion, one finds that

$$\|\sigma(x) - \sigma_P(x)\|_w \leq \|\sigma(x) - \phi(x)\|_w, \quad \forall \phi \in \mathcal{W}_P. \tag{3.9}$$

The last inequality (3.9) is still true for a specific selection for $\phi(x)$ to be $\sigma_P(x)$. Therefore, we conclude

$$\|\sigma(x) - \sigma_P(x)\|_w^2 \leq \|\sigma(x) - \Sigma_P(x)\|_w^2 = \int_{D_x} |\sigma(x) - \Sigma_P(x)|^2 \; w(x)dx.$$

Now, by virtue of (3.8), we proceed as follows:

$$\|\mathcal{E}_P(x)\|_w^2 \leq \Big[\frac{\sigma_\infty}{(P+1)!} L^{P+1}\Big]^2 \int_{D_x} w(x)dx = \Big[\frac{\sigma_\infty}{(P+1)!} L^{P+1}\Big]^2.$$

The conclusion follows upon taking square roots. □

4. The Taylor-Bell hybrid method

The solution to the EFK Eq (1.1) can be obtained by solving the system of discretized Eq (2.6) subject to the boundary conditions (2.7). Given the approximate solution at time frame $m-1$ (where $m \geq 1$), with the initial condition $\sigma^0(x) = \alpha(x)$ at $\tau = 0$, we express the solution at time level m as a truncated expansion in terms of $(P+1)$ Bell basis functions

$$\sigma^m(x) \approx \Sigma_m^P(x) = \sum_{p=0}^{P} \zeta_p^{(m)} \, \mathbb{B}_e^p(x). \tag{4.1}$$

Our primary objective is to determine the unknown coefficients $\zeta_p^{(m)}$ for $p = 0, 1, \ldots, P$. These coefficients can be represented in vector form as

$$\boldsymbol{Z}_P^{(m)} := \begin{bmatrix} \zeta_0^{(m)} & \zeta_1^{(m)} & \cdots & \zeta_P^{(m)} \end{bmatrix}^\top.$$

Recalling the basis vector $\boldsymbol{B}_P(x)$ defined in (3.2), we can reformulate the approximation (4.1) in the compact form

$$\Sigma_m^P(x) = \boldsymbol{B}_P(x)\boldsymbol{Z}_P^{(m)}. \tag{4.2}$$

By application of Lemma 3.2 (specifically relation (3.3)), we derive the simplified form

$$\Sigma_m^P(x) = \mathbf{\Phi}_P(x)\, \boldsymbol{N}_P\, \boldsymbol{Z}_P^{(m)}. \tag{4.3}$$

Examining Eq (2.6), we observe that approximations of $\frac{d^\nu}{dx^\nu}\Sigma_m^P(x)$ are required for $\nu = 2, 4$. To compute these derivatives, we employ the identity from (4.3), which reduces the problem to calculating derivatives of the basis vector $\mathbf{\Phi}_P(x)$. Straightforward differentiation yields

$$\frac{d}{dx}\mathbf{\Phi}_P(x) = \mathbf{\Phi}_P(x)\, \boldsymbol{H}_P, \quad \boldsymbol{H}_P = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 2 & \dots & 0 \\ \vdots & \vdots & 0 & \vdots & \vdots \\ 0 & 0 & 0 & \ddots & P \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}_{(P+1)\times(P+1)}. \tag{4.4}$$

By successive differentiation, we obtain the higher-order derivatives

$$\frac{d^\nu}{dx^\nu}\mathbf{\Phi}_P(x) = \mathbf{\Phi}_P(x)\, (\boldsymbol{H}_P)^\nu, \quad \nu = 2, 4. \tag{4.5}$$

An application of (4.5) leads to the fundamental relation

$$\frac{d^\nu}{dx^\nu}\Sigma_m^P(x) = \mathbf{\Phi}_P(x)\, (\boldsymbol{H}_P)^\nu\, \boldsymbol{N}_P\, \boldsymbol{Z}_P^{(m)}, \quad \nu = 2, 4. \tag{4.6}$$

To solve for the $P + 1$ coefficients in (4.1), we employ $P + 1$ equally spaced collocation points

$$x_\omega = x_l + \frac{(x_r - x_l)\omega}{P}, \quad \omega = 0, 1, \dots, P. \tag{4.7}$$

Evaluating (2.6) at these points generates the discrete system

$$\Theta_1^{m-1}(x_\omega)\, \frac{d^4}{dx^4}\Sigma_P^m(x_\omega) + \Theta_2^{m-1}(x_\omega)\, \frac{d^2}{dx^2}\Sigma_P^m(x_\omega) + \Theta_3^{m-1}(x_\omega)\, \Sigma_P^m(x_\omega) = \Xi^{m-1}(x_\omega), \tag{4.8}$$

for $\omega = 0, 1, \dots, P$ and $m = 1, 2, \dots, \mathcal{M}$. To facilitate the following analysis, we define the subsequent matrix quantities and vector:

$$\boldsymbol{T}_{m-1}^{\ell} = \begin{bmatrix} \Theta_\ell^{m-1}(x_0) & 0 & \dots & 0 \\ 0 & \Theta_\ell^{m-1}(x_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Theta_\ell^{m-1}(x_P) \end{bmatrix}, \ \ell = 1, 2, 3, \quad \boldsymbol{E}_{m-1} = \begin{bmatrix} \Xi^{m-1}(x_0) \\ \Xi^{m-1}(x_1) \\ \vdots \\ \Xi^{m-1}(x_P) \end{bmatrix}.$$

For clarity in subsequent derivations, we introduce the following notation:

$$\boldsymbol{S}_m := \begin{bmatrix} \Sigma_P^m(x_0) \\ \Sigma_P^m(x_1) \\ \vdots \\ \Sigma_P^m(x_P) \end{bmatrix} \quad \boldsymbol{S}_m^{(2)} := \begin{bmatrix} \frac{d^2}{dx^2}\Sigma_P^m(x_0) \\ \frac{d^2}{dx^2}\Sigma_P^m(x_1) \\ \vdots \\ \frac{d^2}{dx^2}\Sigma_P^m(x_P) \end{bmatrix}, \quad \boldsymbol{S}_m^{(4)} := \begin{bmatrix} \frac{d^4}{dx^4}\Sigma_P^m(x_0) \\ \frac{d^4}{dx^4}\Sigma_P^m(x_1) \\ \vdots \\ \frac{d^4}{dx^4}\Sigma_P^m(x_P) \end{bmatrix}.$$

The matrix formalism allows us to express (4.8) equivalently as

$$\boldsymbol{T}_{m-1}^1\, \boldsymbol{S}_m^{(4)} + \boldsymbol{T}_{m-1}^2\, \boldsymbol{S}_m^{(2)} + \boldsymbol{T}_{m-1}^3\, \boldsymbol{S}_m = \boldsymbol{E}_{m-1}, \qquad m = 1, 2, \dots, \mathcal{M}. \tag{4.9}$$

Our framework relies on the following mathematical foundations:

Lemma 4.1. *By enforcing relations* (4.3) *and* (4.6) *at the collocation nodes, one derives the matrix representations for* $\boldsymbol{S}_m$ *as well as* $\boldsymbol{S}_m^{(\nu)}$ *for* $\nu = 2, 4$ *as*

$$\boldsymbol{S}_m = \boldsymbol{\Pi}_P \boldsymbol{N}_P \boldsymbol{Z}_P^{(m)}, \qquad \boldsymbol{S}_m^{(\nu)} = \boldsymbol{\Pi}_P (\boldsymbol{H}_P)^\nu \boldsymbol{N}_P \boldsymbol{Z}_P^{(m)}. \tag{4.10}$$

Here, two matrices $\boldsymbol{N}_P, \boldsymbol{H}_P$ *are defined in* (3.3) *and* (4.4)*, respectively. Additionally, we define the collocation matrix*

$$\boldsymbol{\Pi}_P = [\boldsymbol{\Phi}_P(x_0) \quad \boldsymbol{\Phi}_P(x_1) \quad \dots \quad \boldsymbol{\Phi}_P(x_P)]^\top,$$

where the basis vector $\boldsymbol{\Phi}_P$ *is defined in Lemma 3.2.*

The combination of (4.10) with (4.9) produces the governing matrix equation

$$\left\{\boldsymbol{T}_{m-1}^1 \boldsymbol{\Pi}_P (\boldsymbol{H}_P)^4 + \boldsymbol{T}_{m-1}^2 \boldsymbol{\Pi}_P (\boldsymbol{H}_P)^2 + \boldsymbol{T}_{m-1}^3 \boldsymbol{\Pi}_P\right\} \boldsymbol{N}_P \boldsymbol{Z}_P^{(m)} = \boldsymbol{E}_{m-1}, \quad m = 1, 2, \dots, \mathcal{M}.$$

Rewriting the previous equations in simplified form, we obtain

$$\boldsymbol{A}_m \boldsymbol{Z}_P^{(m)} = \boldsymbol{E}_{m-1}, \quad \text{or} \quad [\boldsymbol{A}_m; \boldsymbol{E}_{m-1}], \quad m = 1, 2, \dots, \mathcal{M}, \tag{4.11}$$

where

$$\boldsymbol{A}_m := \left\{\boldsymbol{T}_{m-1}^1 \boldsymbol{\Pi}_P (\boldsymbol{H}_P)^4 + \boldsymbol{T}_{m-1}^2 \boldsymbol{\Pi}_P (\boldsymbol{H}_P)^2 + \boldsymbol{T}_{m-1}^3 \boldsymbol{\Pi}_P\right\} \boldsymbol{N}_P.$$

To complete the formulation, we must enforce the boundary conditions (2.7) within the matrix system (4.11). This is achieved by

(1) Evaluating (4.3) and (4.6) for $\nu = 2$ at the left boundary point $x \to x_l^-$ and
(2) Evaluating (4.3) and (4.6) for $\nu = 2$ at the right boundary point $x \to x_r^+$.

In both cases, the limiting process yields

$$\begin{cases} \boldsymbol{A}_m^{l,0} \boldsymbol{Z}_P^{(m)} = \beta_l^{m+1}, & \boldsymbol{A}_m^{l,0} := \boldsymbol{\Phi}_P(x_l) \boldsymbol{N}_P, \\ \boldsymbol{A}_m^{l,0} \boldsymbol{Z}_P^{(m)} = \beta_r^{m+1}, & \boldsymbol{A}_m^{r,0} := \boldsymbol{\Phi}_P(x_r) \boldsymbol{N}_P, \end{cases} \qquad \begin{cases} \boldsymbol{A}_m^{l,2} \boldsymbol{Z}_P^{(m)} = 0, & \boldsymbol{A}_m^{l,2} := \boldsymbol{\Phi}_P(x_l) (\boldsymbol{H}_P)^2 \boldsymbol{N}_P, \\ \boldsymbol{A}_m^{r,2} \boldsymbol{Z}_P^{(m)} = 0, & \boldsymbol{A}_m^{r,2} := \boldsymbol{\Phi}_P(x_r) (\boldsymbol{H}_P)^2 \boldsymbol{N}_P. \end{cases}$$

To enforce boundary conditions, we modify the fundamental matrix equation $[\boldsymbol{A}_m; \boldsymbol{E}_{m-1}]$ by replacing four rows with

$$[\boldsymbol{A}_m^{l,0}, \beta_l^{m+1}], [\boldsymbol{A}_m^{l,2}, 0], \quad \text{(left boundary)} \quad \text{and} \quad [\boldsymbol{A}_m^{r,0}, \beta_r^{m+1}], [\boldsymbol{A}_m^{r,2}, 0], \quad \text{(right boundary)}.$$

This yields the modified linear system

$$[\widehat{\boldsymbol{A}}_m; \widehat{\boldsymbol{E}}_{m-1}], \quad m = 1, 2, \dots, \mathcal{M}, \tag{4.12}$$

where $\widehat{\boldsymbol{A}}_m$ is the updated system matrix and $\widehat{\boldsymbol{E}}_{m-1}$ contains the adjusted right-hand side terms. Solving (4.12) provides the Bell coefficients $\zeta_p^{(m)}$ $(p = 0, 1, \dots, P)$ at each time level τ_m for $m = 1, 2, \dots, \mathcal{M}$. At each time level τ_m, the linear system in Eq (4.12) is solved using MATLAB's direct solver via the command `linsolve(`$\widehat{\boldsymbol{A}}_m, \widehat{\boldsymbol{E}}_{m-1}$`)`, ensuring efficient and stable computation of the Bell coefficients $\zeta_p^{(m)}$.

Algorithm 4.1 encapsulates the full sequence of operations in the Taylor-Bell method (TBeM) through a structured pseudocode formulation.

Algorithm 4.1 Stepwise algorithmic implementation of the Taylor-Bell collocation approach.

1: **procedure** TBeM($\mathcal{M}, P, \lambda, g(\tau, x), \alpha(x), \beta_l(\tau), \beta_r(\tau), x_l, x_r, T$)
2: $\Delta\tau := T/\mathcal{M}$; $H(W) = W - W^3$; $H'(W) = 1 - 3W^2$;
3: $\boldsymbol{\Phi}_P(x) := \begin{bmatrix} 1 & x & x^2 & \dots & x^P \end{bmatrix}$;
4: Calculate two matrices $\boldsymbol{N}_P$ and $\boldsymbol{H}_P$ as in (3.3) and (4.4), respectively;
5: $W^0(x) = \alpha(x)$;
{Loop over timesteps}
6: **for** $m = 0, 1, \dots, \mathcal{M} - 1$ **do**
7: $\tau_m := m\,\Delta\tau$; $\tau_{m+1} := \tau_m + \Delta\tau$;
8: $g^{m+1}(x) := g(\tau_{m+1}, x)$; $g^m(x) := g(\tau_m, x)$;
9: $\Theta_1^m(x) := \Delta\tau\lambda$; $\Theta_2^m(x) := -\Delta\tau$; $\Theta_3^m(x) := 2 + \Delta\tau\, H'(W^m(x))$;
10: $\Xi^m(x) := \Delta\tau\,\left[W^m_{xx}(x) - \lambda\, W^m_{xxxx}(x)\right] + W^m(x)\,\Theta_3^m(x) - 2\Delta\tau\, H(W^m) + \Delta\tau\left[g^m(x) + g^{m+1}(x)\right]$;
{Utilize the collocation points (4.7) for ($\ell = 1, 2, 3$)}
11: $\boldsymbol{\Pi}_P := \begin{pmatrix} \boldsymbol{\Phi}_P(x_0) \\ \boldsymbol{\Phi}_P(x_1) \\ \vdots \\ \boldsymbol{\Phi}_P(x_P) \end{pmatrix}$; $T_m^\ell = \begin{bmatrix} \Theta_\ell^m(x_0) & 0 & \dots & 0 \\ 0 & \Theta_\ell^m(x_1) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \Theta_\ell^m(x_P) \end{bmatrix}$; $\boldsymbol{E}_m = \begin{bmatrix} \Xi^m(x_0) \\ \Xi^m(x_1) \\ \vdots \\ \Xi^m(x_P) \end{bmatrix}$;
{Form the algebraic system}
12: LHS_Sys$\equiv \boldsymbol{A}_m := \left\{\boldsymbol{T}_m^1\,\boldsymbol{\Pi}_P\,(\boldsymbol{H}_P)^4 + \boldsymbol{T}_m^2\,\boldsymbol{\Pi}_P\,(\boldsymbol{H}_P)^2 + \boldsymbol{T}_m^3\,\boldsymbol{\Pi}_P\right\}\boldsymbol{N}_P$; RHS_Sys:$=\boldsymbol{E}_m$;
{Enter the B.Cs.}
13: LHS_Sys[1:2,:]:$=[\boldsymbol{\Phi}_P(x_l)\,\boldsymbol{N}_P, \boldsymbol{\Phi}_P(x_r)\,\boldsymbol{N}_P]$; RHS_Sys[1:2]:$=[\beta_l^m, \beta_r^m]^\top$;
14: LHS_Sys[3:4,:]:$=[\boldsymbol{\Phi}_P(x_l)\;(\boldsymbol{H}_P)^2\boldsymbol{N}_P, \boldsymbol{\Phi}_P(x_r)\;(\boldsymbol{H}_P)^2\boldsymbol{N}_P]$; RHS_Sys[3:4]:$=[0, 0]^\top$;
15: $\boldsymbol{Z}_P^{(m)}$:=**LinSolve** $(\widehat{\boldsymbol{A}}_m, \widehat{\boldsymbol{E}}_m)$;
16: Calculate the approximation $\Sigma_m^P(x) = \boldsymbol{\Phi}_P(x)\,\boldsymbol{N}_P\,\boldsymbol{Z}_P^{(m)}$ at level τ_m;
17: Update the initial condition $W^0(x)$ in line 5 by $\Sigma_m^P(x)$;
18: **end for**
19: **end;**

5. Numerical verification and discussion

This portion presents experimental simulations to validate the Taylor-Bell matrix collocation method for the EFK Eq (1.1). We examine several test cases with different parameter configurations to demonstrate the algorithm's accuracy and robustness. The results are quantified through error analysis and visualized graphically to illustrate the method's performance. The computational implementation was executed in MATLAB® R2023a using standard double-precision arithmetic. All simulations were performed on a desktop workstation equipped with an Intel® Core™i7 processor and 16GB RAM. Complete implementation details and datasets are available in the supplementary materials.

To quantitatively assess the numerical scheme's accuracy, we evaluate the absolute value of error at time instant τ_m. Define the error values

$$e_m(x) = |W(\tau_m, x) - \Sigma_P^m(x)|, \quad x \in D_x, \quad 1 \le m \le \mathcal{M}. \tag{5.1}$$

At the final time $\tau = T$, we also compute the infinity error norm as follows:

$$e_\infty \equiv e_{P,\mathcal{M}}^\infty := \max_{x \in D_x} e_{\mathcal{M}}(x). \tag{5.2}$$

The temporal convergence rate is obtained by computing

$$\text{rate}_\tau := \log_2 \left(\frac{e_{P,\mathcal{M}}^\infty}{e_{P,2\mathcal{M}}^\infty} \right). \tag{5.3}$$

The numerical convergence rate with respect to variable x is computed using

$$\text{rate}_x := \log_2 \left(\frac{e_{P,\mathcal{M}}^\infty}{e_{2P,\mathcal{M}}^\infty} \right). \tag{5.4}$$

Example 5.1. *(inhomogeneous model) Let us consider the EFK Eq* (1.1) *with the subsequent initial condition [14, 18]*

$$\alpha(x) = \sin(2\pi x), \quad x \in D_x = (0, 1).$$

The boundary conditions are prescribed as

$$W(\tau, x + 1) = W(\tau, x), \quad x \in D_x, \quad \tau \in D_\tau.$$

One can verify that the actual solution of this model equation is $W(\tau, x) = \sin(2\pi x)\, e^{-\tau}$. *The source function is* $g(\tau, x) = \left[4\pi^2 - 2 + 16\lambda\pi^4 + \sin^2(2\pi x)e^{-2\tau}\right] \sin(2\pi x)\, e^{-\tau}$. *It should be emphasized that the above-mentioned periodic boundary condition is prescribed to be consistent with the periodic nature of the spatial part of the given exact solution,* $\sin(2\pi x)$*, which has a period of* 1.

We consider the initial computational parameters $P = 10$, $\Delta\tau = 0.1$, $\lambda = 0.01$, and final time $T = 1$. The approximate solution obtained via the TBeM at the first time level ($\tau = \Delta\tau$) is

$$\begin{aligned}\Sigma_1^{10}(x) = {} & -4.6737 \times 10^{-6}\, x^{10} - 32.22715193\, x^9 + 145.0222377\, x^8 - 221.3687421\, x^7 \\ & + 98.02009001\, x^6 + 33.06741181\, x^5 + 10.66651552\, x^4 - 38.89198477\, x^3 + 5.711628383\, x.\end{aligned}$$

The numerical approximation obtained at time level $\tau = T$ is as follows:

$$\begin{aligned}\Sigma_{10}^{10}(x) = {} & 4.6737 \times 10^{-6}\, x^{10} - 13.76418467\, x^9 + 61.9385025\, x^8 - 95.58426526\, x^7 + 45.4989839\, x^6 \\ & + 10.12712348\, x^5 + 5.457599306\, x^4 - 16.00023301\, x^3 + 2.32644576\, x.\end{aligned}$$

Figure 1 displays the complete approximate solutions obtained using the specified parameters over the space-time domain $[0, 1] \times [0, 1]$. The left plot illustrates the approximate solution surface while the right panel shows corresponding absolute error distributions at discrete time levels $\tau = \tau_m$ for $m = 1, 2, \ldots, 10$.

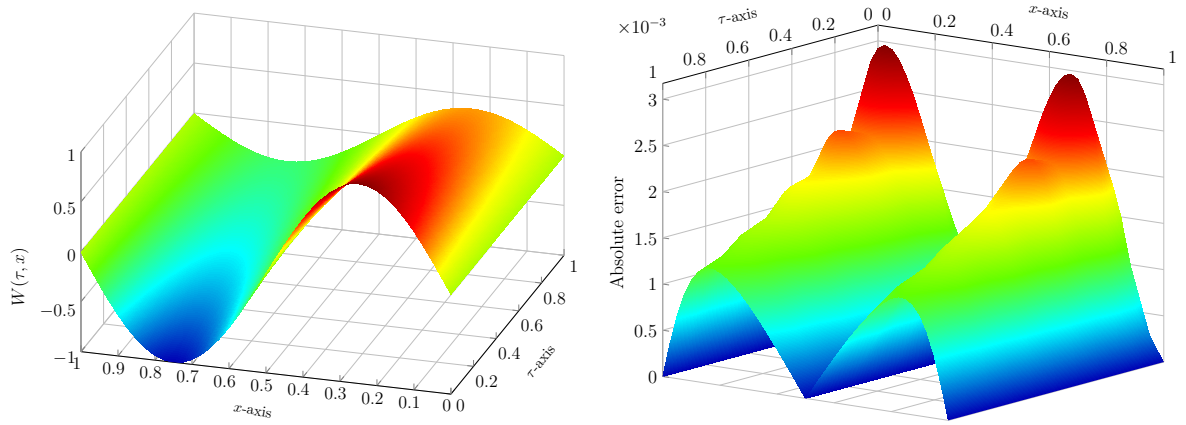

Figure 1. Numerical results for Example 5.1 using the TBeM ($P = 10$, $\mu = 0.01$, $\Delta\tau = 0.1$): (left) approximate solution surface $W(\tau, x)$ and (right) corresponding absolute error distribution, computed over the space-time domain $[0, 1]^2$.

Comparison of numerical (TBeM) and analytical solutions for Example 5.1 ($P = 10$, $\lambda = 0.01$, $\Delta\tau = 0.1$), demonstrating spatial-temporal accuracy at $\tau = T = 1$, is further visually plotted in Figure 2.

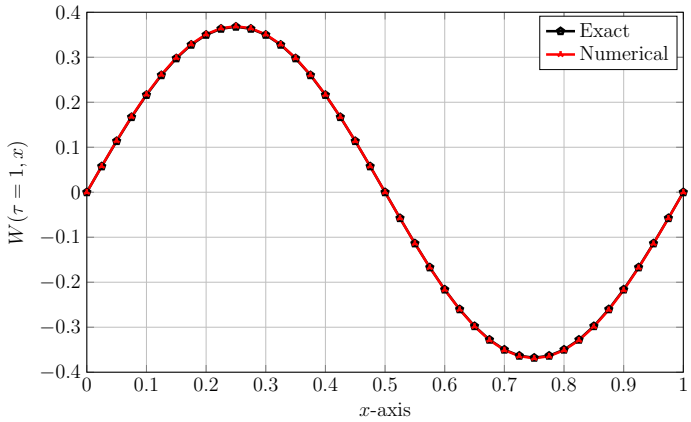

Figure 2. Approximate solution computed by the TBeM at final time $T = 1$ for Example 5.1 ($P = 10$, $\lambda = 0.01$, $\Delta\tau = 0.1$).

Table 1 compares the maximum error norms achieved by our method with those of the three-level linearized finite difference method (TLFDM) studied in [14], demonstrating second-order temporal convergence at $T = 1$. The results numerically confirm that the proposed TBeM not only attains second-order accuracy in time but also achieves higher precision with lower computational cost compared to the established TLFDM. Additionally, the table includes the elapsed CPU time (in seconds) for the TBeM, and the outcomes are compared with those of TLFDM. The CPU time for the TBeM is significantly lower than that of TLFDM at comparable refinement levels, demonstrating superior computational efficiency. For instance, at the finest timestep ($\Delta\tau = 2^{-4}$), the TBeM requires only 7.6 seconds compared to TLFDM's 18.7 seconds, while achieving comparable accuracy.

Table 1. Error norms (e_∞) along with corresponding convergence rates (w.r.t. τ) and CPU time required for the TBeM with $P = 17$ and $\lambda = 0.01$ computed at $T = 1$ for varying timesteps ($\Delta\tau$) in Example 5.1.

	TBeM ($P = 17$)				TLFDM [14] ($h = \frac{1}{500}$)		
$\Delta\tau$	e_∞	rate$_\tau$	CPU [s]	k	$e_\infty(h,k)$	q = rate2	CPU [s]
2^{-1}	4.8286×10^{-3}	–	0.9508	$\frac{1}{20}$	3.4825×10^{-4}	–	2.0156
2^{-2}	1.0504×10^{-3}	2.2007	1.8993	$\frac{1}{40}$	7.1211×10^{-5}	2.2899	4.3594
2^{-3}	2.3412×10^{-4}	2.1656	3.8038	$\frac{1}{80}$	1.7651×10^{-5}	2.0123	8.4531
2^{-4}	4.0866×10^{-6}	5.8402	7.6049	$\frac{1}{160}$	4.3935×10^{-6}	2.0062	18.6719

Similarly, in Table 2, we investigate the numerical rate of convergence (rate$_x$) with regard to space variable x. For these results, we use the same parameters as in Table 1 except that we fix $\Delta\tau = 0.0125$ and vary $P = 2^j$ for $j = 2, 3, 4, 5$. We compare our results with those reported by two previously published approaches: the TLFDM [14] (using the same parameters) and the quintic trigonometric B-spline method (QTBSM) [19] with $\lambda = 1$.

Table 2. Error norms (e_∞) along with corresponding convergence rates (w.r.t. x) for the TBeM with $\Delta\tau = 0.0125$ and $\lambda = 0.01$ computed at $T = 1$ for varying basis number (P) in Example 5.1.

	TBeM ($\Delta\tau = 0.0125$)			TLFDM [14] ($k = 10^{-5}$)			QTBSM [19] ($k = 10^{-2}, \lambda = 1$)	
P	e_∞	rate$_x$	h	$e_\infty(h,k)$	p = rate2	M	L_∞	R_c
2^2	3.6789×10^{-1}	–	$\frac{1}{20}$	1.4252×10^{-4}	–	20	4.414×10^{-2}	–
2^3	1.3498×10^{-2}	4.7684	$\frac{1}{40}$	8.9808×10^{-6}	3.9880	25	2.312×10^{-2}	2.90
2^4	5.4700×10^{-5}	7.9470	$\frac{1}{80}$	5.5832×10^{-7}	4.0076	45	6.551×10^{-3}	2.68
2^5	1.1191×10^{-7}	8.9331	$\frac{1}{160}$	3.0827×10^{-8}	4.0178	–	–	–

The TBeM demonstrates a significantly higher order of spatial convergence compared to both TLFDM and QTBSM, achieving rates of up to 8.93 as P increases, while TLFDM plateaus near fourth order and QTBSM remains below third order. This superior convergence highlights the TBeM's exceptional efficiency in refining spatial accuracy with minimal computational resources.

Example 5.2. *(homogeneous model) We define the second test problem with the initial condition [15, 18]*

$$\alpha(x) = 10(1-x)^4 x^4, \quad x \in D_x = [0,1],$$

and with zero boundary conditions

$$W(\tau, x_l) = W(\tau, x_r) = 0, \quad W_{xx}(\tau, x_l) = W_{xx}(\tau, x_r) = 0.$$

To the best of our knowledge, no exact solution for this model has been reported in the literature.

We executed the TBeM algorithm with the parameters $\lambda = 0.001$, $P = 10$, and a timestep of $\Delta\tau = 0.0125$. The resulting approximate solution, computed at the first timestep $\tau = \Delta\tau$ for the spatial domain $0 \le x \le 1$, is

$$\Sigma_1^{10}(x) = -9.648924143\, x^{10} + 48.2446287\, x^9 - 99.84515935\, x^8 + 109.9128359\, x^7 - 69.36008863\, x^6$$

$$+ 26.0128205\,x^5 - 6.244025368\,x^4 + 0.8757124707\,x^3 + 0.05219990834\,x.$$

Applying the same TBeM configuration at time $\tau = 0.2$ yields the following result:

$$\begin{aligned}\Sigma_{16}^{10}(x) = &-0.0001142214197\,x^{10} + 0.000571105921\,x^9 - 0.0001803357687\,x^8 - 0.002705288169\,x^7\\ &- 0.00005018014546\,x^6 + 0.01201768819\,x^5 - 0.00000306518523\,x^4 - 0.02432172826\,x^3\\ &+ 1.74406035 \times 10^{-105}\,x^2 + 0.01478602484\,x.\end{aligned}$$

To evaluate the accuracy of the TBeM in lieu of an exact solution, we compute the residual error function (REF). According to (2.6), for $x \in D_x$ at $\tau = \tau_{m+1}$, this function is given by

$$\Re_{m+1,P}(x) =: \Theta_1^m(x)\,\sigma_{m+1}^{(4)}(x) + \Theta_2^m(x)\,\sigma_{m+1}''(x) + \Theta_3^m(x)\,\sigma_{m+1}(x) - \Xi^m(x), \quad m \in \mathbb{M}. \tag{5.5}$$

The numerical results for Example 5.2 are presented in Figure 3. The left panel shows the approximate solution while the right panel displays the absolute values of the corresponding REFs for all timesteps $m = 0, 1, \ldots, M-1$, where the total number of steps is $M = 80$.

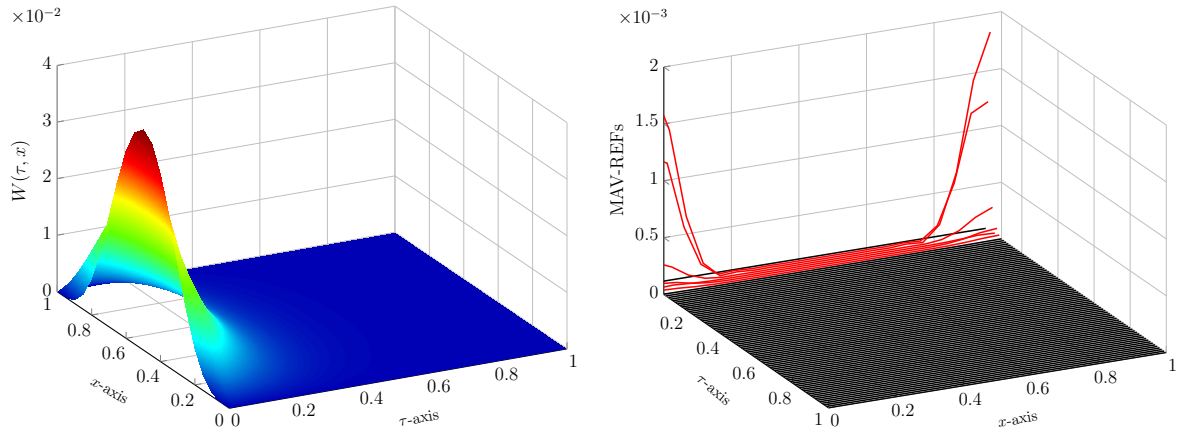

Figure 3. Numerical results for Example 5.2 using the TBeM ($P = 10$, $\lambda = 0.001$, $\Delta\tau = 0.0125$): (left) approximate solution surface $W(\tau, x)$ and (right) corresponding residual error (RE) distribution, computed over the space-time domain $[0, 1]^2$.

Note that the parameters for the aforementioned plots remain unchanged, with a final time of $T = 1$. We emphasize that the REFs at the initial time levels $\tau = \tau_1, \tau_2, \ldots, \tau_6$ are plotted in red to highlight their initially high magnitude. This phenomenon can be attributed to the startup behavior of our semi-discrete method, where the spatial discretization using Bell polynomials is coupled with a Taylor series-based time integration. The first few timesteps involve the highest-order temporal derivatives approximated by the Taylor series, which are sensitive to the initial spatial discretization error and the solution's rapid initial evolution. As the computation progresses, the scheme stabilizes, and the maximum absolute value of the REFs (MAV-REFs) becomes very small. For instance, at $\tau = \tau_7$, the MAV-REF is 3.8121×10^{-6}, demonstrating the method's rapid convergence after the initial transient phase.

The following analysis involves the computation of the REF's supremum norm. Evaluated at the final timestep τ_M, the norm is given by

$$\Re^\infty \equiv \Re_{P,\mathcal{M}}^\infty := \max_{x \in D_x} |\Re_{\mathcal{M},P}(x)|. \tag{5.6}$$

The convergence rates of the proposed method can be computed in an analogous manner to Eqs (5.3) and (5.4).

The following results, compiled in Table 3, present the RE norms and the associated convergence rates at the final time $T = 0.2$. These were obtained through two separate numerical experiments: first, by fixing the polynomial degree $P = 12$ and varying the timestep $\Delta\tau = 2^{-j}$ for $2 \leq j \leq 5$; and second, by fixing $\Delta\tau = 0.0125$ and varying the approximation parameter $P = 4, 6, 8, 10$.

Table 3. Error norms ($\mathfrak{R}^\infty$) along with corresponding convergence rates (w.r.t. τ/x) for the TBeM with $\Delta\tau = 0.0125/P = 12$ and $\lambda = 0.001$ computed at $T = 0.2$ in Example 5.1.

TBeM						CFDM [15] ($\tau = 10^{-5}$)		
$\Delta\tau$	$\mathfrak{R}^\infty$ ($P = 12$)	rate_τ	P	$\mathfrak{R}^\infty$ ($\Delta\tau = 0.0125$)	rate_x	h	$\|e^n\|_{h,\infty}$	order 1
2^{-2}	3.4947×10^{-1}	–	4	7.9324×10^{-5}	–	$\frac{1}{20}$	8.3284×10^{-7}	–
2^{-3}	9.8852×10^{-2}	1.8218	6	3.5617×10^{-6}	4.4771	$\frac{1}{40}$	5.1579×10^{-8}	4.0132
2^{-4}	3.5071×10^{-3}	1.4950	8	1.2095×10^{-7}	4.8801	$\frac{1}{80}$	3.3453×10^{-9}	3.9466
2^{-5}	9.3328×10^{-3}	1.9099	10	4.0374×10^{-9}	4.9049	$\frac{1}{160}$	1.9204×10^{-10}	4.1227

To validate the performance of the TBeM, we compare its results against those from an established fourth-order compact finite difference method (CFDM) [15]. The CFDM results were generated using a fixed, highly refined timestep of $\tau = 10^{-5}$ and various spatial stepsizes h, with its corresponding order of convergence included for reference. An analysis of the tabulated data demonstrates that the TBeM achieves high-order accuracy while consuming significantly fewer computational resources compared to the CFDM.

Example 5.3. *(homogeneous model on non-unit interval) Let us pay attention to the subsequent initial conditions [6, 9, 18]*

$$\alpha(x) = \pm\, 10^{-3} e^{-x^2}, \quad x \in D_x = [-4, 4].$$

The positive and negative signs in the initial condition correspond to the following sets of boundary conditions, respectively:
Case (i) $W(\tau, x_l) = W(\tau, x_r) = +1$,
Case (ii) $W(\tau, x_l) = W(\tau, x_r) = -1$,
together with

$$W_{xx}(\tau, x_l) = W_{xx}(\tau, x_r) = 0.$$

As with the second test problem, no exact solution is available for these two models either.

The numerical approximations for Cases (i) and (ii), computed with $P = 10$, $\Delta\tau = 0.0125$, and $\lambda = 0.0001$, are presented below for the initial simulation time $\tau = \Delta\tau$:

$$\begin{aligned} {}^{(i)}\Sigma_1^{10}(x) = {} & -0.000003503782877\, x^{10} + 6.3509 \times 10^{-19}\, x^9 + 0.000116500311\, x^8 - 8.0394 \times 10^{-18}\, x^7 \\ & - 0.0008355986019\, x^6 - 3.6527 \times 10^{-16}\, x^5 + 0.001897456247\, x^4 + 3.3873 \times 10^{-15}\, x^3 \\ & - 0.001568681732\, x^2 - 2.4961592 \times 10^{-15}\, x + 0.0009802319541, \end{aligned}$$

and

$${}^{(ii)}\Sigma_1^{10}(x) = 0.000003503782877\, x^{10} - 6.3509 \times 10^{-19}\, x^9 - 0.000116500311\, x^8 + 8.0394 \times 10^{-18}\, x^7$$

$$+ 0.0008355986019\, x^6 + 3.6527 \times 10^{-16}\, x^5 - 0.001897456247\, x^4 - 3.3873 \times 10^{-15}\, x^3$$
$$+ 0.001568681732\, x^2 + 2.4961592 \times 10^{-15}\, x - 0.0009802319541.$$

It can be seen that the coefficients of both approximations are the same but have an opposite sign up to machine epsilon. The evolution of the solution at successive time intervals is presented in Figure 4. Consistent with the findings in [6, 9, 19], the solution for a small initial condition decays over time, asymptotically approaching the steady-state value of ±1 corresponding to Cases (i) and (ii).

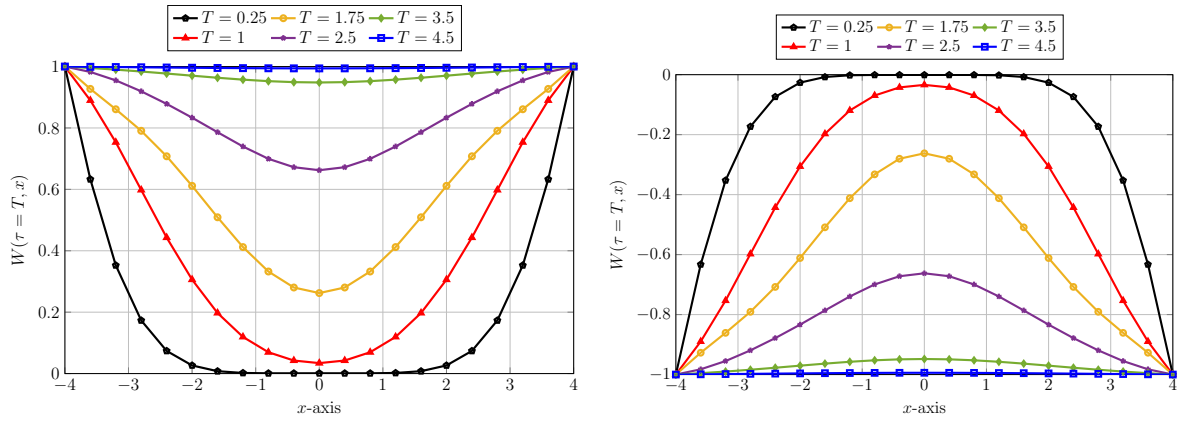

Figure 4. Numerical results for Example 5.3 using the TBeM ($P = 10$, $\lambda = 0.0001$, $\Delta\tau = 0.0125$): Approximate solution surface $W(T, x)$ in Case (i) (left) and Case (ii) (right) computed at different final time T.

The three-dimensional representation of the numerical solutions in Cases (i)-(ii), computed with $\lambda = 0.0001$, $\Delta\tau = 0.05$, and $P = 10$ for $\tau \in [0, 5]$, is shown in Figure 5. The results demonstrate a decay process converging to a stable state near ±1, which is consistent with the known qualitative behavior of the EFK equation.

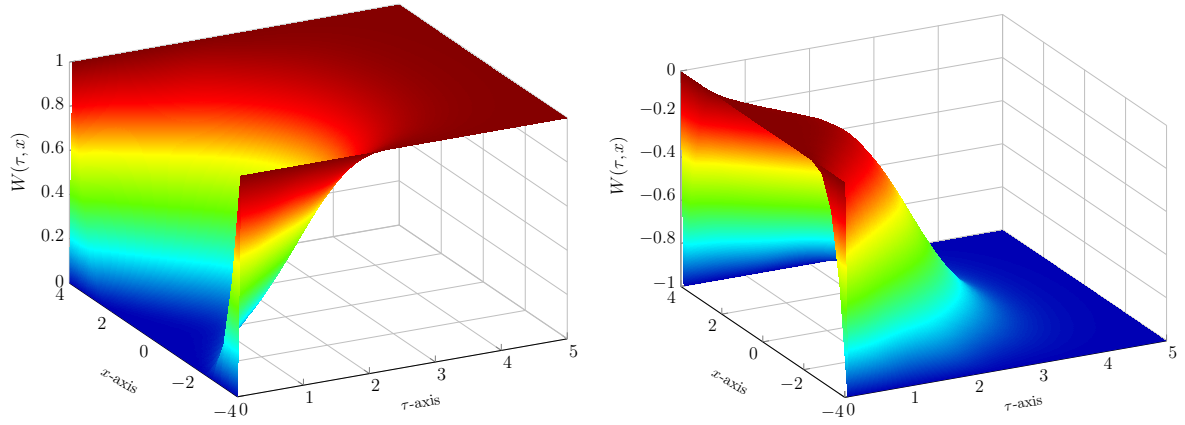

Figure 5. Numerical results for Example 5.3 using the TBeM ($P = 10$, $\lambda = 0.0001$, $\Delta\tau = 0.05$): 3D visualization of $W(\tau, x)$ in Case (i) (left) and Case (ii) (right) computed on the whole domain $(\tau, x) \times [0, 5] \times [-4, 4]$.

In the final analysis, we compute the L^2 error norm of the REF at the final timestep $\tau = T$ using the definition

$$\Re_2 \equiv \Re^2_{P,\mathcal{M}} := \frac{1}{\sqrt{P+1}} \left(\int_{D_x} |\Re_{\mathcal{M},P}(x)|^2 dx \right)^{\frac{1}{2}}, \tag{5.7}$$

where the REF $\Re_{\mathcal{M},P}(x)$ is given in (5.5). The resulting values of $\Re_2$ at $T = 1$, computed with a fixed timestep $\Delta\tau = 0.003125$ and a parameter $\lambda = 0.0001$, are presented in Table 4 for both Cases (i) and (ii). These results were generated using approximation orders $P = 4, 8, 16$. We note that the results for both cases are identical, which is attributable to the symmetry of the model problems described in Example 5.3. For comparison, Table 4 also includes results from the established trigonometric B-spline method (TBSM) method [19], computed with a timestep of $\Delta\tau = 10^{-4}$ and problem sizes $M = 20, 40, 80$. The data clearly demonstrate that the proposed TBeM achieves superior accuracy with lower computational complexity than the TBSM approach.

For the sake of completeness, we evaluate the condition number of the coefficient matrix $\widehat{\boldsymbol{A}}_m$ appearing in (4.12) at the last time frame $\tau_{\mathcal{M}}$. Formally, it is given by

$$\kappa(\widehat{\boldsymbol{A}}_{\mathcal{M}}) = \|\widehat{\boldsymbol{A}}_{\mathcal{M}}\|_\infty \, \|\widehat{\boldsymbol{A}}^{-1}_{\mathcal{M}}\|_\infty.$$

In practice, this quantity is computed using the MATLAB command cond(·). The result for different values of P are reported in the sixth column of Table 4. Note that the results for Cases (i) and (ii) coincide up to machine epsilon, owing to the symmetry induced by initial conditions. It becomes apparent that enlarging the matrix dimension, or equivalently increasing P, leads to a coefficient matrix that is ill-conditioned. Nevertheless, this does not pose a significant difficulty and imposes no serious limitation on the efficiency of our algorithm. On the contrary, a larger choice of P enhances the accuracy of the results.

Table 4. Error norms ($\Re_2$) along with corresponding convergence rates (w.r.t. x) and condition numbers for the TBeM with $\Delta\tau = 0.003125$ and $\lambda = 0.0001$ computed at $T = 1$ in Example 5.3.

	TBeM ($\Delta\tau = 0.003125$)					TBSM [19] ($\Delta\tau = 10^{-4}$)		
	Case (i)		Case (ii)		Cases (i)-(ii)	Case (i)		
P	$\Re_2$	rate$_x$	$\Re_2$	rate$_x$	$\kappa(\widehat{\boldsymbol{A}}_{\mathcal{M}})$	M	L_2	R_c
4	3.1264×10^{-3}	–	3.1264×10^{-3}	–	$7.7907 \times 10^{+2}$	20	2.909×10^{-2}	–
8	7.4254×10^{-4}	2.0740	7.4254×10^{-4}	2.0740	$4.1811 \times 10^{+7}$	40	9.250×10^{-3}	1.65
16	1.8159×10^{-4}	2.0318	1.8159×10^{-4}	2.0318	$1.4457 \times 10^{+24}$	80	2.587×10^{-3}	1.84

6. Conclusions

This paper has presented a robust and efficient numerical technique for solving the EFK equation. The suggested approach synergistically combines the Taylor series for temporal linearization with a spectral matrix collocation methodology based upon Bell polynomials for spatial discretization. The theoretical convergence analysis, established in the weighted L^2-norm, is validated through extensive

numerical simulations. The results from three test cases confirm that the TBeM achieves high accuracy and outperforms existing solutions for the EFK equation. The framework developed herein is not only effective for the current problem but also holds significant potential for application to other high-order nonlinear partial differential equations.

Despite its strong performance, the proposed TBeM has certain limitations that should be acknowledged. The method's reliance on Taylor series for time integration necessitates sufficient temporal smoothness in the solution, making it less suitable for problems with nonsmooth initial conditions or solutions that develop discontinuities. Furthermore, the current formulation is implemented on rectangular space-time domains, and its extension to complex geometries remains an open challenge. The global nature of the Bell polynomial basis can also lead to dense matrix systems, which may pose computational challenges for problems requiring extremely high spatial resolution or extension to higher spatial dimensions. Future work will focus on addressing these limitations by extending this methodology to more complex domains, developing hybrid approaches for nonsmooth problems, and implementing adaptive strategies to enhance computational efficiency. The application of this framework to other classes of high-order nonlinear PDEs also presents a promising research direction.

Use of Generative-AI tools declaration

The author declares that he has not used Artificial Intelligence (AI) tools in the creation of this article.

Conflict of interest

The author declares no conflicts of interest.

Acknowledgments

The author would like to thank the anonymous referee and editor very much for their valuable comments and suggestions, which greatly help us improve the presentation of this article.

References

1. P. Coullet, C. Elphick, D. Repaux, Nature of spatial chaos, *Phys. Rev. Lett.*, **58** (1987), 431–434. http://doi.org/10.1103/PhysRevLett.58.431
2. W. Van Saarloos, Dynamical velocity selection: Marginal stability, *Phys. Rev. Lett.*, **58** (1987), 2571–2574. http://doi.org/10.1103/PhysRevLett.58.2571
3. W. Van Saarloos, Front propagation into unstable states. II. Linear versus nonlinear marginal stability and rate of convergence, *Phys. Rev. A*, **39** (1989), 6367–6390. http://doi.org/10.1103/PhysRevA.39.6367
4. G. T. Dee, W. van Saarloos, Bistable systems with propagating fronts leading to pattern formation, *Phys. Rev. Lett.*, **60** (1988), 2641–2644. http://doi.org/10.1103/PhysRevLett.60.2641

5. L. A. Peletier, W. C. Troy, A topological shooting method and the existence of kinks of the extended Fisher-Kolmogorov equation, *Topol. Method. Nonlin.*, **6** (1995), 331–355.
6. P. Danumjaya, A. K. Pani, Orthogonal cubic spline collocation method for the extended Fisher-Kolmogorov equation, *J. Comput. Appl. Math.*, **174** (2005), 101–117. http://doi.org/10.1016/j.cam.2004.04.002
7. P. C. Carriao, L. F. O. Faria, O. H. Miyagaki, Periodic solutions for extended Fisher-Kolmogorov and Swift-Hohenberg equations by truncature techniques, *Nonlinear Anal. Theor.*, **67** (2007), 3076–3083. http://doi.org/10.1016/j.na.2006.09.061
8. T. Kadri, K. Omrani, A second-order accurate difference scheme for an extended Fisher-Kolmogorov equation, *Comput. Math. Appl.*, **61** (2011), 451–459. http://doi.org/10.1016/j.camwa.2010.11.022
9. R. C. Mittal, S. Dahiya, A study of quintic B-spline based differential quadrature method for a class of semi-linear Fisher-Kolmogorov equations, *Alex. Eng. J.*, **55** (2016), 2893–2899. https://doi.org/10.1016/j.aej.2016.06.019
10. M. Ilati, M. Dehghan, Direct local boundary integral equation method for numerical solution of extended Fisher-Kolmogorov equation, *Eng. Comput.*, **34** (2018), 203–213. http://doi.org/10.1007/s00366-017-0530-1
11. I. Celik, Gegenbauer wavelet collocation method for the extended Fisher-Kolmogorov equation in two dimensions, *Math. Method. Appl. Sci.*, **43** (2020), 5615–5628. http://doi.org/10.1002/mma.6300
12. M. Ilati, Analysis and application of the interpolating element-free Galerkin method for extended Fisher-Kolmogorov equation which arises in brain tumor dynamics modeling, *Numer. Algor.*, **85** (2020), 485–502. http://doi.org/10.1007/s11075-019-00823-6
13. L. J. Doss, N. Kousalya, A finite pointset method for extended Fisher-Kolmogorov equation based on mixed formulation, *Int. J. Comp. Meth.*, **18** (2021), 2050019. http://doi.org/10.1142/S021987622050019X
14. K. Ismail, N. Atouani, K. Omrani, A three-level linearized high-order accuracy difference scheme for the extended Fisher-Kolmogorov equation, *Eng. Comput.*, **38** (2022), 1215–1225. http://doi.org/10.1007/s00366-020-01269-4
15. S. Li, D. Xu, J. Zhang, C. Sun, A new three-level fourth-order compact finite difference scheme for the extended Fisher-Kolmogorov equation, *Appl. Numer. Math.*, **178** (2022), 41–51. http://doi.org/10.1016/j.apnum.2022.03.010
16. Q. Sun, B. Ji, L. Zhang, A convex splitting BDF2 method with variable time-steps for the extended Fisher-Kolmogorov equation, *Comput. Math. Appl.*, **114** (2022), 73–82. http://doi.org/10.1016/j.camwa.2022.03.017
17. L. Pei, C. Zhang, D. Shi, Unconditional superconvergence analysis of two-grid nonconforming FEMs for the fourth order nonlinear extended Fisher-Kolmogorov equation, *Appl. Math. Comput.*, **471** (2024), 128602. http://doi.org/10.1016/j.amc.2024.128602

18. Shallu, V. K. Kukreja, Numerical solution and analysis of extended Fisher-Kolmogorov equation using an improved collocation algorithm, *Int. J. Comput. Math.*, **102** (2025), 415–434. http://doi.org/10.1080/00207160.2024.2415701

19. S. R. Thottoli, M. Tamsir, M. Z. Meetei, A. H. Msmali, Numerical investigation of nonlinear extended Fisher-Kolmogorov equation via quintic trigonometric B-spline collocation technique, *AIMS Math.*, **9** (2024), 17339–17358. http://doi.org/10.3934/math.2024843

20. M. J. Huntul, M. Tamsir, A. A. H. Ahmadini, S. R. Thottoli, A novel collocation technique for parabolic partial differential equations, *Ain Shams Eng. J.*, **13** (2022), 101497. http://doi.org/10.1016/j.asej.2021.05.011

21. H. Han, C. Zhang, One-parameter Galerkin finite element methods for neutral reaction-diffusion equations with piecewise continuous arguments, *J. Sci. Comput.*, **90** (2022), 91. http://doi.org/10.1007/s10915-022-01769-z

22. M. Shafiq, M. Abbas, K. M. Abualnaja, M. J. Huntul, A. Majeed, T. Nazir, An efficient technique based on cubic B-spline functions for solving time-fractional advection diffusion equation involving Atangana–Baleanu derivative, *Eng. Comput.*, **38** (2022), 901–917. http://doi.org/10.1007/s00366-021-01490-9

23. M. Tamsir, M. J. Huntul, N. Dhiman, S. Singh, Redefined quintic B-spline collocation technique for nonlinear higher order PDEs, *Comput. Appl. Math.*, **41** (2022), 413. http://doi.org/10.1007/s40314-022-02127-3

24. W. Qu, Y. Gu, C. M. Fan, A stable numerical framework for long-time dynamic crack analysis, *Int. J. Solids Struct.*, **293** (2024), 112768. http://doi.org/10.1016/j.ijsolstr.2024.112768

25. W. Sun, W. Qu, Y. Gu, S. Zhao, Dynamic analysis of bi-material interfacial cracks by the high-order GFDM with an enhanced Krylov deferred correction technique, *Int. J. Solids Struct.*, **318** (2025), 113451. http://doi.org/10.1016/j.ijsolstr.2025.113451

26. M. A. Rufai, A. A. Kosti, Z. A. Anastassi, B. Carpentieri, A new two-step hybrid block method for the FitzHugh-Nagumo model equation, *Mathematics*, **12** (2024), 51. http://doi.org/10.3390/math12010051

27. M. J. Huntul, D. Lesnic, Time-dependent reaction coefficient identification problems with a free boundary, *Int. J. Comput. Method. Eng. Sci. Mech.*, **20** (2019), 99–114. http://doi.org/10.1080/15502287.2019.1568619

28. M. Xia, X. Li, Q. Shen, T. Chou, Learning unbounded-domain spatiotemporal differential equations using adaptive spectral methods, *J. Appl. Math. Comput.*, **70** (2024), 4395–4421. http://doi.org/10.1007/s12190-024-02131-2

29. B. V. Rathish Kumar, M. Mehra, Time-accurate solutions of Korteweg-de Vries equation using wavelet Galerkin method, *Appl. Math. Comput.*, **162** (2005), 447–460. http://doi.org/10.1016/j.amc.2003.12.10

30. I. Dağ, A. Canıvar, A. Şahin, Taylor-Galerkin and Taylor-collocation methods for the numerical solutions of Burgers' equation using B-splines, *Commun. Nonlinear Sci.*, **16** (2011), 2696–2708. http://doi.org/10.1016/j.cnsns.2010.10.009

31. M. Izadi, Numerical approximation of Hunter-Saxton equation by an efficient accurate approach on long time domains, *U.P.B. Sci. Bull. Ser. A*, **83** (2021), 291–300.

32. J. Donea, H. Laval, S. Guiliani, L. Quartapelle, Taylor-Galerkin method for time-dependent transport problems, In: *Transactions of the 7th International Conference on Structural Mechanics in Reactor Technology*, **B** (1983), 77–82.

33. M. Torabi, M. M. Hosseini, A new efficient method for the numerical solution of linear time-dependent partial differential equations, *Axioms*, **7** (2018), 70. http://doi.org/10.3390/axioms7040070

34. M. Izadi, P. Roul, Spectral semi-discretization algorithm for a class of nonlinear parabolic PDEs with applications, *Appl. Math. Comput.*, **429** (2022), 127226. http://doi.org/10.1016/j.amc.2022.127226

35. E. T. Bell, Exponential polynomials, *Ann. Math.*, **35** (1934), 258–277. http://doi.org/10.2307/1968431

36. M. Mihoubi, M. S. Maamra, Touchard polynomials, partial Bell polynomials and polynomials of binomial type, *J. Integer Seq.*, **14** (2011), 3.

37. S. Yüzbaşı, Fractional Bell collocation method for solving linear fractional integro-differential equations, *Math. Sci.*, **18** (2024), 29–40. http://doi.org/10.1007/s40096-022-00482-0

38. P. Yadav, S. Jahan, K. S. Nisar, A new generalized Bell wavelet and its applications for solving linear and nonlinear integral equations, *Comput. Appl. Math.*, **44** (2025), 40. http://doi.org/10.1007/s40314-024-02999-7

39. L. Comtet, *Advanced combinatorics: The art of finite and infinite expansions*, Dordrecht: Springer, 1974. https://doi.org/10.1007/978-94-010-2196-8

40. K. Parand, H. Yousefi, M. Fotouhifar, M. Delkhosh, M. Hosseinzadeh, Shifted Boubaker Lagrangian approach for solving biological systems, *Int. J. Biomath.*, **11** (2018), 1850039. http://doi.org/10.1142/S1793524518500390

AIMS Press