Contemporary Mathematics
https://ojs.wiserpub.com/index.php/CM/

Research Article

Spectral Treatment of the Fractional Bratu Equation via Shifted Lucas Polynomials: A Precise Collocation Approach with Error Quantification

M. H. Salama[1], H. A. Zedan[1], W. M. Abd-Elhameed[2], Y. H. Youssri[2,3]*

[1] Department of Mathematics, Faculty of Science, Kafrelsheikh University, Kafrelsheikh, 33516, Egypt
[2] Department of Mathematics, Faculty of Science, Cairo University, Giza, 12613, Egypt
[3] Faculty of Engineering, Egypt University of Informatics, Knowledge City, New Administrative Capital, 19519, Egypt
E-mail: youssri@cu.edu.eg

Received: 5 August 2025; **Revised:** 9 September 2025; **Accepted:** 16 September 2025

Abstract: This paper introduces a novel spectral collocation method based on shifted Lucas polynomials for solving the fractional Bratu differential equation with nonhomogeneous Dirichlet boundary conditions. The method employs a homogenization strategy and an operational matrix formulation in the Caputo sense to transform the problem into a nonlinear algebraic system, which is efficiently solved via the Newton-Raphson method. Detailed error analysis confirms exponential convergence, and extensive numerical experiments demonstrate the method's superior accuracy and efficiency compared to existing approaches, even for strongly nonlinear and fractional-order cases.

***Keywords*:** fractional Bratu equation, shifted Lucas polynomials, spectral collocation method, Caputo derivative, nonhomogeneous boundary conditions, Newton-Raphson method, error analysis, exponential convergence

MSC: 65L10, 65M70, 33C45, 92C10, 26A33

1. Introduction

Numerical solutions of different types of Differential Equations (DEs) are necessary in most cases due to the inability to obtain exact solutions for many DEs that arise in several branches of the applied sciences. Many numerical approaches were followed to obtain approximate solutions for different DEs. For example, the Adomian decomposition method with variational iteration [1], and the wavelets methods [2–6]. Spectral methods are numerical techniques that solve differential equations by approximating the solution with a linear combination of basis functions, typically chosen from orthogonal polynomials or trigonometric functions [7]. These methods are particularly advantageous for problems where the solution is smooth, as they can achieve high accuracy with relatively few degrees of freedom due to their exponential convergence properties [8]. The collocation method is notable for its simplicity and effectiveness among the various spectral methods. In the collocation approach, the differential equation is enforced to vanish at discrete points, called collocation points. This leads to a system of algebraic equations that can be solved for the coefficients of the basis functions [9]. This approach is particularly well-suited for problems exhibiting singularities, sharp gradients, or fractional behavior, as it can effectively capture complex solution structures [10]. In addition, due to its wide applicability for all types of DEs, it was successfully applied to handle several DEs constrained by different underlying conditions, see for example [11–14]. The other two

DOI: https://doi.org/10.37256/cm.6520258150

spectral methods, namely the tau and Galerkin, can also be applied to solve different DEs. In the tau method, we select two different types of basis functions, see [15–18], whereas in the Galerkin methods, the two selected families coincide, see, for example [19, 20].

In recent years, there has been growing interest in applying spectral methods to Fractional Differential Equations (FDEs), which involve derivatives of non-integer order. FDEs are powerful tools for modeling phenomena with memory effects or anomalous diffusion, such as those found in viscoelasticity, biology, and physics [21]. These equations often exhibit behaviors such as singularities or sharp gradients that challenge traditional numerical methods. The collocation spectral method has been adapted to handle FDEs by incorporating fractional derivative operators, typically defined in the Caputo's sense, into the discretization process, allowing for accurate and efficient solutions [10]. Many algorithms were devoted to solving the different FDEs, such as the predictor-corrector method [22], A spatial sixth-order numerical scheme [23], the Laplace Adomian decomposition method [24], a homotopy method [25], a homotopy perturbation method [26], Laplace transform method [27]. The ability of spectral methods to provide a fully diagonal representation of fractional operators enhances their computational efficiency, making them an attractive choice for solving FDEs. The collocation method was successfully applied in [28–30], the tau method in [31–33], and the Galerkin method in [34, 35].

One particular FDE of interest is the fractional Bratu differential equation, a nonlinear equation given by

$$D^{\alpha}u(x)+\lambda e^{u(x)}=0 \quad \text{where} \quad \alpha\in(0,\,2] \quad \text{and} \quad x\in[0,\,1], \tag{1}$$

subject to the nonhomogeneous Dirichlet boundary conditions

$$u(0)=\beta, \qquad u(1)=\gamma. \tag{2}$$

The fractional Bratu equation generalizes the classical Bratu equation, which arises in applications such as thermal runaway in combustion theory, astrophysics (e.g., the Emden-Chandrasekhar equation), and chemical reaction theory [36]. Introducing the fractional derivative allows for modeling non-local effects and memory-dependent processes, making it relevant to a broader range of physical and engineering contexts, including diffusion processes and pattern formation in biological and chemical systems [37]. Solving the fractional Bratu equation poses significant mathematical challenges, including the nonlinearity introduced by the exponential term, the non-local nature of the fractional derivative, and the need to enforce nonhomogeneous boundary conditions accurately.

This paper proposes a novel collocation spectral method based on shifted Lucas polynomials as basis functions to address these challenges. Lucas polynomials are a sequence of polynomials that generalize the Lucas numbers and are closely related to Chebyshev polynomials [38]. The recurrence relation defines them

$$L_0(x)=2, \quad L_1(x)=x, \quad L_n(x)=xL_{n-1}(x)+L_{n-2}(x), \quad n\geq 2.$$

By applying a linear transformation, such as $t=2x-1$, these polynomials are shifted to the interval $[0,\,1]$, yielding shifted Lucas polynomials $L_n^*(x)=L_n(2x-1)$, which are well-suited for problems defined on this domain due to their computational efficiency with smaller coefficients reducing numerical errors compared to Chebyshev and Legendre polynomials, shifted Lucas polynomials offer distinct advantages [39, 40]. Their non-orthogonal structure enables easier generation while maintaining exponential convergence rates [41, 42]. The recurrence relation provides exceptional compatibility with fractional derivative operations in the Caputo sense, making them particularly well-suited for constructing operational matrices that preserve spectral accuracy in FDEs [43]. Previous research has demonstrated the efficacy of orthogonal polynomials, such as Chebyshev and Legendre polynomials, in spectral methods for solving FDEs [44–46]. Moreover, Lucas polynomials have been successfully employed in spectral methods for solving various

differential equations, including fractional initial value problems [47]. However, applying shifted Lucas polynomials to the fractional Bratu equation represents a novel approach that leverages their unique properties to achieve accurate and efficient solutions.

The remainder of this paper is organized as follows: In Section 2, we provide definitions and preliminaries, including the Caputo fractional derivative and the properties of Lucas polynomials. Section 3 presents the theoretical formulation of the numerical scheme, including the boundary homogenization technique to handle nonhomogeneous Dirichlet conditions, and applying the collocation method to construct the nonlinear algebraic system, then solving this system via the Newton-Raphson method. In Section 4, we conduct an error analysis, establishing lemmas and estimating upper bounds for the approximation error to demonstrate the method's convergence properties. Section 5 analyzes the convergence of the Newton-Raphson method that is used to solve the resulting nonlinear system. In Section 6, we present numerical results for various values of the nonlinear coefficient λ, comparing our method with other existing approaches to highlight its accuracy and efficiency. Finally, concluding remarks are reported in Section 7.

2. Essential preliminaries

This section presents some fundamentals of the fractional calculus. In addition, some useful properties of the shifted Lucas polynomials are presented.

2.1 *Caputo fractional derivative*

Definition 1 The Caputo fractional derivative of order $\alpha > 0$ is given in [48] by

$$ {}^{C}D^{\alpha} f(x) = \frac{1}{\Gamma(n-\alpha)} \int_0^x (x-t)^{n-\alpha-1} f^{(n)}(t)\, dt, \quad \alpha > 0, x > 0 $$

where $n-1 \leq \alpha < n$ and $n \in \mathbb{N}$.

The following properties hold:

$$ D^{\alpha} C = 0, \quad (C \text{ is a constant}), $$

$$ D^{\alpha} x^k = \begin{cases} 0, & \text{if } k \in \mathbb{N}_0, \text{ and } k < \lceil \alpha \rceil, \\ \dfrac{k!}{\Gamma(k+1-\alpha)} x^{k-\alpha}, & \text{if } k \in \text{if } k \in \mathbb{N}_0, \text{ and } k \geq \lceil \alpha \rceil, \end{cases} $$

where $\lceil \alpha \rceil$ is the ceiling function, and $\mathbb{N}_0 = \{0,\ 1,\ 2,\ \ldots\}$.

2.2 *Lucas and shifted Lucas polynomials*

Definition 2 The Lucas polynomials $\{L_n(x)\}_{n=0}^{\infty}$ can be generated using the following recurrence relation [49]:

$$\begin{cases} L_0(x) = 2, \\ \\ L_1(x) = x, \\ \\ L_n(x) = xL_{n-1}(x) + L_{n-2}(x), \quad n \geq 2. \end{cases}$$

Definition 3 The shifted Lucas polynomials $\{L_n^*(x)\}_{n=0}^{\infty}$ are defined by the following transformation:

$$L_n^*(x) = L_n(2x-1), \quad x \in [0,\ 1],$$

and meet the following recurrence relation:

$$\begin{cases} L_0^*(x) = 2, \\ \\ L_1^*(x) = 2x - 1, \\ \\ L_n^*(x) = (2x-1)L_{n-1}^*(x) + L_{n-2}^*(x), \quad n \geq 2. \end{cases}$$

Definition 4 The explicit power form of shifted Lucas polynomials is given in [50] by:

$$L_n^*(x) = n \sum_{j=0}^{\lfloor \frac{n}{2} \rfloor} \sum_{k=0}^{n-2j} 2^k (-1)^{n-k} \frac{(n-j-1)!}{j!\ k!(n-2j-k)!} x^k.$$

3. The proposed numerical scheme

It is known that the set of shifted Lucas polynomials $\{L_n^*(x)\}_{n=0}^{\infty}$ can be used as a basis to represent analytic functions on the interval $[0,\ 1]$. Accordingly, we define the following:

Definition 5 Any analytic function $g(x)$ can be represented in the form of an infinite series of shifted Lucas polynomials as:

$$g(x) = \sum_{n=0}^{\infty} d_n L_n^*(x).$$

Definition 6 To find an approximate solution $u_N(x)$, we define the polynomial approximation space S_N and the subspace V_N which satisfies the boundary conditions, as follows:

$$S_N[0,\ 1] = \text{span}\{L_n^*(x) : n = 0,\ 1,\ 2,\ \ldots,\ N\},$$

and

$$V_N = \{u \in S_N : u(0) = \beta, u(1) = \gamma\},$$

such that its approximation $u_N(x)$ may be written in the following form:

$$u_N(x) = \sum_{n=0}^{N} d_n L_n^*(x).$$

3.1 *Boundary condition homogenization*

Lemma 1 For the inhomogeneous Dirichlet conditions $u(0) = \beta$, $u(1) = \gamma$, any approximation in the form: $u_N(x) = \sum_{n=0}^{N} d_n L_n^*(x)$ admits the decomposition:

$$u_N(x) = \underbrace{\sum_{n=0}^{N} d_n L_n^*(x)}_{v_N(x)} + \underbrace{\beta(1-x) + \gamma x}_{w(x)},$$

where $v_N(0) = 0$ and $v_N(1) = 0$ if and only if the coefficients $\{d_n\}$ satisfy:

$$\sum_{n=0}^{N} d_n L_n^*(0) = 0, \quad \sum_{n=0}^{N} d_n L_n^*(1) = 0.$$

Proof. We consider the following transformation:

$$w(x) = \beta(1-x) + \gamma x,$$

which satisfies

$$w(0) = \beta, w(1) = \gamma.$$

Let $v_N(x) = \sum_{n=0}^{N} d_n L_n^*(x)$. The two conditions $v_N(0) = 0$ and $v_N(1) = 0$, lead to

$$\sum_{n=0}^{N} d_n L_n^*(0) = 0, \quad \sum_{n=0}^{N} d_n L_n^*(1) = 0,$$

and, consequently, the decomposition $u_N(x) = v_N(x) + w(x)$ enforces homogeneous boundary conditions on v_N if and only if the coefficient constraints hold. This preserves the spectral structure of v_N while $w(x)$ accounts for the nonhomogeneous Dirichlet conditions. □

3.2 *Collocation system*

Lemma 2 [51] The Caputo fractional derivative of the shifted Lucas polynomials of order α is given by

$$D^{\alpha}L_n^*(x)=\begin{cases} n\sum\limits_{k=\lceil\alpha\rceil}^{n}(-1)^{n-k}\dfrac{(n+k-1)!}{(n-k)!(2k)!}\cdot\dfrac{k!}{\Gamma(k-\alpha+1)}2^{k+1}x^{k-\alpha}, & \ell-1<\alpha\leq\ell,\ \ell=\lceil\alpha\rceil\in\mathbb{N},\ n\geq\alpha,\\ 0, & n<\alpha. \end{cases}$$

Corollary 1 For the homogenized approximate solution

$$u_N(x)=\beta\,(1-x)+\gamma x+\sum_{n=0}^{N}d_nL_n^*(x),$$

the Caputo fractional derivative $D^{\alpha}u_N(x)$ of order $\alpha\in(0,\ 2]$ is given by:

$$D^{\alpha}u_N(x)=\delta_{\lceil\alpha\rceil,\,1}(\gamma-\beta)\frac{x^{1-\alpha}}{\Gamma(2-\alpha)}+\sum_{n=\lceil\alpha\rceil}^{N}d_nn\sum_{k=\lceil\alpha\rceil}^{n}(-1)^{n-k}\frac{(n+k-1)!}{(n-k)!(2k)!}\frac{k!}{\Gamma(k-\alpha+1)}2^{k+1}x^{k-\alpha},$$

where $\delta_{\lceil\alpha\rceil,\,1}$ is the Kroncker-delta function.

Proof. The homogenized solution splits as $u_N(x)=w(x)+v_N(x)$, where $w(x)=\beta(1-x)+\gamma x$, and $v_N(x)=\sum\limits_{n=0}^{N}d_nL_n^*(x)$. Using the linearity of the Caputo derivative:

$$D^{\alpha}u_N(x)=D^{\alpha}w(x)+D^{\alpha}v_N(x),\tag{3}$$

and applying Definition 1 and Lemma 2 to equation (3), yields the following multi-base fractional derivative:

$$D^{\alpha}u_N(x)=\begin{cases}(\gamma-\beta)\dfrac{x^{1-\alpha}}{\Gamma(2-\alpha)}+\sum\limits_{n=1}^{N}d_nn\sum\limits_{k=1}^{n}(-1)^{n-k}\dfrac{(n+k-1)!}{(n-k)!(2k)!}\cdot\dfrac{k!}{\Gamma(k-\alpha+1)}2^{k+1}x^{k-\alpha},\ 0<\alpha\leq1,\\ \sum\limits_{n=2}^{N}d_nn\sum\limits_{k=2}^{n}(-1)^{n-k}\dfrac{(n+k-1)!}{(n-k)!(2k)!}\cdot\dfrac{k!}{\Gamma(k-\alpha+1)}2^{k+1}x^{k-\alpha},\ 1<\alpha\leq2.\end{cases}$$

Yet, we can combine it in a single unified expression as:

$$D^{\alpha}u_N(x)=\delta_{\lceil\alpha\rceil,\,1}(\gamma-\beta)\frac{x^{1-\alpha}}{\Gamma(2-\alpha)}+\sum_{n=\lceil\alpha\rceil}^{N}d_nn\sum_{k=\lceil\alpha\rceil}^{n}(-1)^{n-k}\frac{(n+k-1)!}{(n-k)!(2k)!}\frac{k!}{\Gamma(k-\alpha+1)}2^{k+1}x^{k-\alpha}.$$

□

If we approximate $u(x)$ and $D^{\alpha}u(x)$ as in Corollary 1 and Lemma 1, then the residual $R(x)$ of equation (1)

$$
\begin{aligned}
R(x) =& D^{\alpha} u_N(x) + e^{u_N(x)} \\
=& \delta_{\lceil \alpha \rceil,\, 1}(\gamma - \beta)\frac{x^{1-\alpha}}{\Gamma(2-\alpha)} + \sum_{n=\lceil \alpha \rceil}^{N} d_n n \sum_{k=\lceil \alpha \rceil}^{n} (-1)^{n-k} \frac{(n+k-1)!}{(n-k)!(2k)!} \frac{k!}{\Gamma(k-\alpha+1)} 2^{k+1} x^{k-\alpha} \\
&+ \lambda\, e^{\beta(1-x)+\gamma x + \sum_{n=0}^{N} d_n L_n^*(x)}.
\end{aligned}
$$

The numerical solution is enforced through a dual collocation strategy comprising the boundary conditions:

$$
v_N(0) = \sum_{n=0}^{N} d_n L_n^*(0) = 0, \quad v_N(1) = \sum_{n=0}^{N} d_n L_n^*(1) = 0,
$$

and the interior collocation via selecting $(N-1)$ interior Chebyshev-Gauss-Lobatto nodes:

$$
x_j = \frac{1}{2}\left(1 - \cos\left(\frac{j\pi}{N}\right)\right), \quad j = 1,\, 2,\, \ldots,\, N-1,
$$

with residual enforcement:

$$
R(x_j;\, \mathbf{d}) = 0,
$$

where $\mathbf{d} = [d_0,\, d_1,\, \ldots,\, d_N]^T$ denotes the coefficient vector.

The complete system comprises the following $(N+1)$ equations in $\mathbb{R}^{N+1}$:

$$
\mathbf{F}(\mathbf{d}) = \begin{bmatrix} \sum_{n=0}^{N} d_n L_n^*(0) \\ \sum_{n=0}^{N} d_n L_n^*(1) \\ R(x_1;\, \mathbf{d}) \\ \vdots \\ R(x_{N-1};\, \mathbf{d}) \end{bmatrix} = \mathbf{0}. \tag{4}
$$

3.3 *Newton-Raphson system solver*

The Jacobian matrix $\mathbf{J}(\mathbf{d}) \in \mathbb{R}^{(N+1)\times(N+1)}$ is defined by partial derivatives of the residual vector components in (4) as [52]

$$
\mathbf{J}(\mathbf{d}) = \begin{bmatrix} J_{0,\,0} & \cdots & J_{0,\,N} \\ \vdots & \ddots & \vdots \\ J_{N,\,0} & \cdots & J_{N,\,N} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial F_0}{\partial d_0} & \cdots & \dfrac{\partial F_0}{\partial d_N} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial F_N}{\partial d_0} & \cdots & \dfrac{\partial F_N}{\partial d_N} \end{bmatrix}, \tag{5}
$$

where

$$\frac{\partial F_i}{\partial d_m} = \begin{cases} L_m^*(0), & i = 0, \\ L_m^*(1), & i = 1, \\ \begin{cases} \phi_m(x_{i-2}) + \lambda e^{u_N(x_{i-2})} L_m^*(x_{i-2}), & m \geq \lceil \alpha \rceil, \\ \lambda e^{u_N(x_{i-2})} L_m^*(x_{i-2}), & m < \lceil \alpha \rceil, \end{cases} & i \geq 2, \end{cases} \quad \forall m = 0,\ 1,\ \ldots,\ N$$

and

$$\phi_m(x) = m \sum_{k=\lceil \alpha \rceil}^{m} (-1)^{m-k} \frac{(m+k-1)!}{(m-k)!(2k)!} \frac{k!}{\Gamma(k-\alpha+1)} 2^{k+1} x^{k-\alpha}.$$

After setting the Jacobian, we assemble $\mathbf{J}^{(\eta)} = \mathbf{J}(\mathbf{d}^{(\eta)})$ for the iteration counter $\eta = 0,\ 1,\ \ldots$, as follows:

$$\mathbf{J}^{(\eta)} = \begin{bmatrix} J_{0,\,0}^{(\eta)} & \cdots & J_{0,\,N}^{(\eta)} \\ \vdots & \ddots & \vdots \\ J_{N,\,0}^{(\eta)} & \cdots & J_{N,\,N}^{(\eta)} \end{bmatrix}.$$

Then, we solve for the Newton increment $\mathbf{J}^{(\eta)}\Delta\mathbf{d} = -\mathbf{F}^{(\eta)}$ with solution update using $\mathbf{d}^{(\eta+1)} = \mathbf{d}^{(\eta)} + \Delta\mathbf{d}$, where $\mathbf{F}^{(\eta)} = \mathbf{F}(\mathbf{d}^{(\eta)})$ is the iterative residual vector.

Remark 1 Figure 1 presents a comprehensive flowchart illustrating the complete methodology of the proposed numerical scheme. The algorithm begins with the input parameters (α, λ, β, γ, N, tol), proceeds through the construction of the shifted Lucas polynomial basis, homogenization of boundary conditions, setup of the collocation grid and assembly of the nonlinear system $\mathbf{F}(\mathbf{d}) = \mathbf{0}$. The Newton-Raphson iteration loop (shown in the dashed box) continues until the convergence criterion $\|\mathbf{F}^{(\eta)}\| < \text{tol}$ is satisfied, ultimately yielding the approximate solution $u_N(x)$. This systematic approach ensures robust convergence and accurate solutions for the fractional Bratu equation.

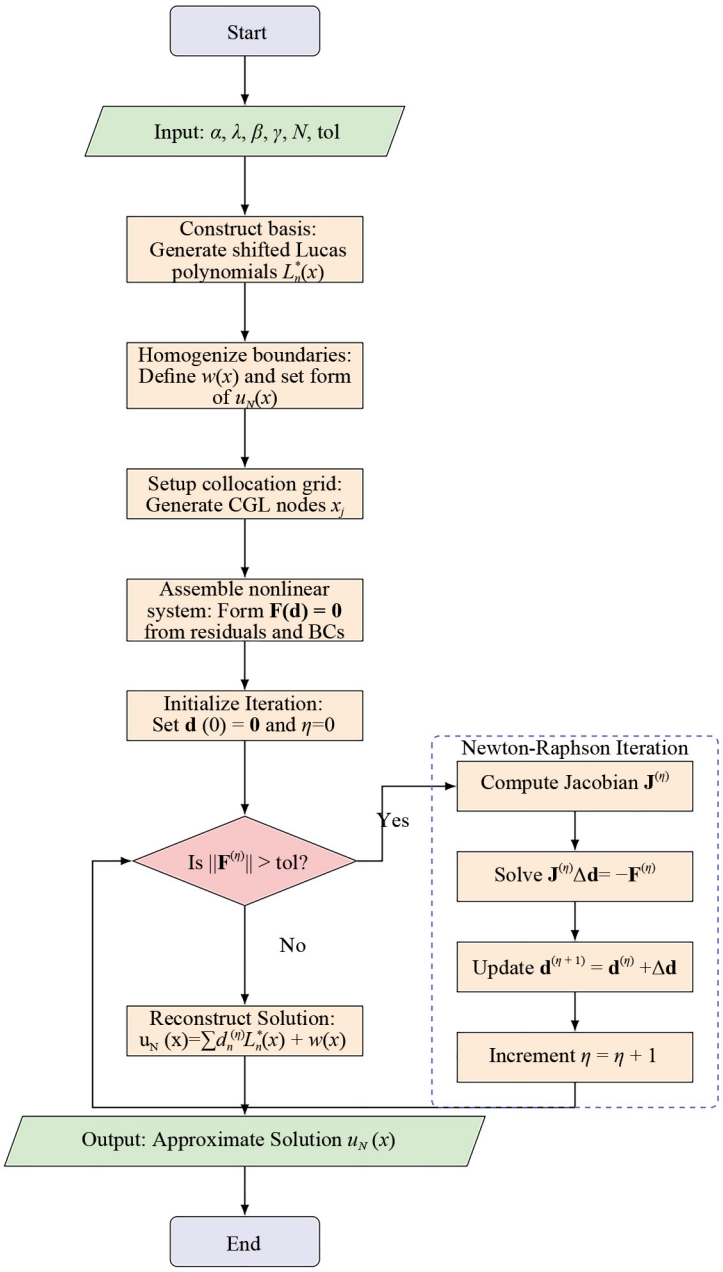


Figure 1. Flowchart for the numerical scheme methodology

4. Error analysis

This section investigates the error analysis of the proposed shifted Lucas expansion. Some lemmas and theorems are stated and proved in this regard.

Lemma 3 For any integer $m \geq 0$, the monomial x^m expands in terms of the shifted Lucas polynomial $L_n^*(x)$ as:

$$x^m = \sum_{i=0}^{m} p_{m,\,i} L_i^*(x),$$

where

$$p_{m,\,i} = \frac{\delta_i}{2^m} \sum_{j=0}^{\left\lfloor \frac{m-i}{2} \right\rfloor} (-1)^r \binom{m}{i+2j} \frac{(i+j+1)_j}{j!},$$

$$\delta_i = \begin{cases} \frac{1}{2},\ i = 0, \\ \\ 1,\ i > 0, \end{cases}$$

and $(z)_j$ represents the Pochhammer notation given by

$$(z)_j = \frac{\Gamma(z+j)}{\Gamma(z)}.$$

Proof. We define $t = 2x - 1$, so $x = \frac{t+1}{2}$ and $t \in [-1,\ 1]$ for $x \in [0,\ 1]$. Then, we get

$$x^m = \left(\frac{t+1}{2}\right)^m = \frac{1}{2^m}(t+1)^m = \frac{1}{2^m} \sum_{k=0}^{m} \binom{m}{k} t^k.$$

If we insert following the inversion formula to t^k [53]:

$$t^k = \sum_{j=0}^{\left\lfloor \frac{k}{2} \right\rfloor} \frac{(-1)^j \delta_{k-2j} (k-j+1)_j}{j!} L_{k-2j}(t),$$

then we get

$$x^m = \frac{1}{2^m} \sum_{k=0}^{m} \binom{m}{k} \sum_{j=0}^{\left\lfloor \frac{k}{2} \right\rfloor} \frac{(-1)^j \delta_{k-2j} (k-j+1)_j}{j!} L_{k-2j}(t),$$

which can be written alternatively as

$$x^m = \frac{1}{2^m} \sum_{i=0}^{m} \sum_{j=0}^{\left\lfloor \frac{m-i}{2} \right\rfloor} \binom{m}{i+2j} \frac{(-1)^j \delta_i (i+j+1)_j}{j!} L_i(t),$$

substituting $L_i(t) = L_i^*(x)$, we get

$$x^m = \sum_{i=0}^{m} \left(\frac{\delta_i}{2^m} \sum_{j=0}^{\lfloor \frac{m-i}{2} \rfloor} (-1)^j \binom{m}{i+2j} \frac{(i+j+1)_j}{j!} \right) L_i^*(x).$$

This confirms the expansion. □

Lemma 4 Let $f(x)$ be an analytic function at $x = 0$, then $f(x)$ admits the expansion:

$$f(x) = \sum_{i=0}^{\infty} \mu_i L_i^*(x),$$

with

$$\mu_i = \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{(-1)^n f^{(i+2n+k)}(0)}{n!k!(i+n)!2^{i+2n+k}},$$

and

$$\delta_i = \begin{cases} \frac{1}{2}, & i = 0, \\ 1, & i \geq 1. \end{cases}$$

Proof. By expanding $f(x)$ via its Taylor series at $x = 0$:

$$f(x) = \sum_{m=0}^{\infty} \frac{f^{(m)}(0)}{m!} x^m.$$

We express x^m in the basis $\{L_i^*(x)\}_{i=0}^{\infty}$ using the inversion formula for shifted Lucas polynomials from Lemma 4.

$$x^m = \sum_{i=0}^{m} p_{m,\,i} L_i^*(x), \quad p_{m,\,i} = \frac{\delta_i}{2^m} \sum_{j=0}^{\lfloor \frac{m-i}{2} \rfloor} (-1)^j \binom{m}{i+2j} \frac{(i+j+1)_j}{j!},$$

Substituting this into the Taylor series gives:

$$f(x) = \sum_{m=0}^{\infty} \frac{f^{(m)}(0)}{m!} \sum_{i=0}^{m} p_{m,\,i} L_i^*(x).$$

Interchanging summations yields:

$$f(x) = \sum_{i=0}^{\infty} \left(\sum_{m=i}^{\infty} \frac{f^{(m)}(0)}{m!} p_{m,\,i} \right) L_i^*(x).$$

The coefficient $\mu_i = \sum_{m=i}^{\infty} \frac{f^{(m)}(0)}{m!} p_{m,\,i}$ expands to:

$$\mu_i = \delta_i \sum_{m=i}^{\infty} \sum_{j=0}^{\lfloor \frac{m-i}{2} \rfloor} \frac{f^{(m)}(0)}{m!} \cdot \frac{(-1)^j}{2^m} \binom{m}{i+2j} \frac{(i+j+1)_j}{j!}.$$

Reindexing with $n = j$ and $k = m - i - 2n$ (so $m = i + 2n + k$, $k \geq 0$):

$$\mu_i = \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{f^{(i+2n+k)}(0)}{(i+2n+k)!} \cdot \frac{(-1)^n}{2^{i+2n+k}} \binom{i+2n+k}{i+2n} \frac{(i+n+1)_n}{n!}.$$

Using the two simple identities:

$$\binom{i+2n+k}{i+2n} = \frac{(i+2n+k)!}{k!(i+2n)!},$$

$$(i+n+1)_n = \frac{(i+2n)!}{(i+n)!},$$

we obtain

$$\mu_i = \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{(-1)^n f^{(i+2n+k)}(0)}{n!k!(i+n)!2^{i+2n+k}}.$$

Thus, $f(x) = \sum_{i=0}^{\infty} \mu_i L_i^*(x)$, completing the proof. □

Lemma 5 Let $f(x)$ be an analytic function at $x = 0$ such that $|f^{(k)}(0)| \leq A^k$ for all $k \geq 0$ and some $A > 0$. Also, Let μ be the coefficients in the expansion of $f(x)$ in shifted Lucas polynomials $L_i^*(x)$:

$$f(x) = \sum_{i=0}^{\infty} \mu_i L_i^*(x), \quad \mu_i = \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{(-1)^n f^{(i+2n+k)}(0)}{n!k!(i+n)!2^{i+2n+k}}.$$

Then,

$$|\mu_i| \leq \frac{(A/2)^i e^{3A/2}}{i!}.$$

Proof. Starting from the expression for $|\mu_i|$:

$$|\mu_i| = \left| \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{(-1)^n f^{(i+2n+k)}(0)}{n!k!(i+n)!2^{i+2n+k}} \right| \leq \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{|f^{(i+2n+k)}(0)|}{n!k!(i+n)!2^{i+2n+k}}.$$

Using the bound $|f^{(i+2n+k)}(0)| \leq A^{i+2n+k}$:

$$|\mu_i| \leq \delta_i \sum_{n=0}^{\infty} \sum_{k=0}^{\infty} \frac{A^{i+2n+k}}{n!k!(i+n)!2^{i+2n+k}} = \delta_i \frac{A^i}{2^i} \sum_{n=0}^{\infty} \frac{(A^2/4)^n}{n!(i+n)!} \sum_{k=0}^{\infty} \frac{(A/2)^k}{k!}.$$

The inner sum over k is the exponential series:

$$\sum_{k=0}^{\infty} \frac{(A/2)^k}{k!} = e^{A/2},$$

and accordingly, we have

$$|\mu_i| \leq \delta_i \frac{A^i}{2^i} e^{A/2} \sum_{n=0}^{\infty} \frac{(A^2/4)^n}{n!(i+n)!}.$$

The series $\sum_{n=0}^{\infty} \frac{(A^2/4)^n}{n!(i+n)!}$ is recognized as $\frac{2^i}{A^i} I_i(A)$, where $I_i(A)$ is the modified Bessel function of the first kind of order i.

Based on the identity:

$$I_i(A) = \left(\frac{A}{2}\right)^i \sum_{n=0}^{\infty} \frac{(A^2/4)^n}{n!(i+n)!},$$

we can write

$$|\mu| \leq \delta_i \frac{A^i}{2^i} e^{A/2} \cdot \frac{2^i}{A^i} I_i(A) = \delta_i e^{A/2} I_i(A).$$

We make use of the following bound: $|I_i(A)| \leq \frac{(A/2)^i \cosh A}{i!}$ [53]:

$$|\mu_i| \leq \delta_i e^{A/2} \cdot \frac{(A/2)^i \cosh A}{i!}.$$

In addition, since $\cosh A \leq e^A$ for $A > 0$, thus, we have

$$e^{A/2}\cosh A \le e^{A/2}e^{A} = e^{3A/2},$$

and accordingly, we can write

$$|\mu_i| \le \delta_i \frac{(A/2)^i e^{3A/2}}{i!}.$$

Noting that $\delta_i \le 1$ for all $i \ge 0$, we get

$$|\mu_i| \le \frac{(A/2)^i e^{3A/2}}{i!}.$$

This completes the proof. □

Theorem 1 Let $u(x) = v(x) + w(x)$ be the exact solution and $u_N(x) = v_N(x) + w(x)$ the approximate solution, where $w(x) = \beta(1-x) + \gamma x$, $v(x) = \sum_{n=0}^{\infty} d_n L_n^*(x)$, and $v_N(x) = \sum_{n=0}^{N} d_n L_n^*(x)$, then the truncation error $e_N(x) = u(x) - u_N(x)$ is bounded by:

$$|e_N(x)| \le e^{(\rho+3)A/2} \frac{(\rho A)^{N+1}}{2^N (N+1)!},$$

where $\rho = \dfrac{1+\sqrt{5}}{2}$ is the golden ratio.

Proof. The truncation error is:

$$e_N(x) = u(x) - u_N(x) = [v(x) + w(x)] - [v_N(x) + w(x)] = v(x) - v_N(x) = \sum_{n=N+1}^{\infty} d_n L_n^*(x).$$

Applying the triangle inequality and the bound $|L_n^*(x)| \le 2\rho^n$ for $x \in [0,\ 1]$ from [42]:

$$|e_N(x)| = \left| \sum_{n=N+1}^{\infty} d_n L_n^*(x) \right| \le \sum_{n=N+1}^{\infty} |d_n| \cdot |L_n^*(x)| \le 2 \sum_{n=N+1}^{\infty} \rho^n |d_n|.$$

By Lemma 5, we substitute the coefficient bound $|d_n| \le \dfrac{(A/2)^n e^{3A/2}}{n!}$ to get

$$|e_N(x)| \le 2 \sum_{n=N+1}^{\infty} \rho^n \frac{(A/2)^n e^{3A/2}}{n!} = 2e^{3A/2} \sum_{n=N+1}^{\infty} \frac{(\rho A/2)^n}{n!}.$$

The series $\sum_{n=N+1}^{\infty} \dfrac{(\rho A/2)^n}{n!}$ is the tail of the exponential series $e^{\rho A/2} = \sum_{n=0}^{\infty} \dfrac{(\rho A/2)^n}{n!}$. This tail is bounded by:

$$\sum_{n=N+1}^{\infty} \frac{(\rho A/2)^n}{n!} \leq \frac{(\rho A/2)^{N+1}}{(N+1)!} e^{\rho A/2}.$$

This can be proved as follows: Since,

$$\sum_{n=N+1}^{\infty} \frac{(\rho A/2)^n}{n!} = \frac{(\rho A/2)^{N+1}}{(N+1)!} \left(1 + \frac{\rho A/2}{N+2} + \frac{(\rho A/2)^2}{(N+2)(N+3)} + \cdots\right).$$

We factorize out $\frac{(\rho A/2)^{N+1}}{(N+1)!}$ as:

$$= \frac{(\rho A/2)^{N+1}}{(N+1)!} \left(1 + \frac{\rho A/2}{N+2} + \frac{(\rho A/2)^2}{(N+2)(N+3)} + \cdots\right) \leq \frac{(\rho A/2)^{N+1}}{(N+1)!} \sum_{m=0}^{\infty} \frac{(\rho A/2)^m}{m!} = \frac{(\rho A/2)^{N+1}}{(N+1)!} e^{\rho A/2}.$$

By substituting this into the error bound:

$$|e_N(x)| \leq 2e^{3A/2} \cdot \frac{(\rho A/2)^{N+1}}{(N+1)!} e^{\rho A/2} = 2e^{\frac{A}{2}(\rho+3)} \frac{(\rho A/2)^{N+1}}{(N+1)!},$$

and accordingly, the following estimation can be obtained:

$$|e_N(x)| \leq e^{(\rho+3)A/2} \frac{(\rho A)^{N+1}}{2^N (N+1)!}.$$

The proof is now complete. □

5. Newton-Raphson convergence analysis

The successful implementation of the Newton-Raphson method for solving the nonlinear system in (4) requires establishing rigorous conditions that guarantee convergence. In this section, we provide a comprehensive theoretical framework for the convergence of the iterative scheme presented in Section 3.3. We systematically verify the fundamental requirements through four lemmas; Lemma 6 establishes the smoothness of the residual function $\mathbf{F}$, then proving the existence of a solution with non-singular Jacobian at the fixed point in Lemma 7, followed by Lemma 8 that demonstrates the boundedness of the inverse Jacobian in a neighborhood of the solution, and finally Lemma 9 verifies the Lipschitz continuity of the Jacobian mapping. These results culminate in Theorem 2, which establishes the local quadratic convergence of the Newton-Raphson iteration for the discretized fractional Bratu equation. The theoretical guarantees provided here ensure that for appropriate initial guesses and parameter ranges, the numerical scheme converges rapidly to the exact solution, as demonstrated by the computational results in Section 6.

Lemma 6 (Smoothness of the Residual Function). Let $\mathbf{F} : \mathbb{R}^{N+1} \to \mathbb{R}^{N+1}$ be the residual function defined in equation (4). Then, $\mathbf{F} \in C^2(\Omega)$ for any open convex set $\Omega \subseteq \mathbb{R}^{N+1}$ containing the solution $\mathbf{d}^*$ of $\mathbf{F}(\mathbf{d}) = \mathbf{0}$.

Proof. The function $\mathbf{F}$ is a vector-valued function with components $F_0, F_1, \ldots, F_N$.

The components F_0 and F_1 have the following expressions:

$$F_0(\mathbf{d}) = \sum_{n=0}^{N} d_n \; L_n^*(0), \quad F_1(\mathbf{d}) = \sum_{n=0}^{N} d_n \; L_n^*(1),$$

and they are are linear functions of $\mathbf{d}$, hence infinitely differentiable (C^∞), and in particular C^2.

The components F_i :

For $j = 1, \ldots, N-1$, let $x = x_j$. Then,

$$F_{j+1}(\mathbf{d}) = R(x; \mathbf{d}) = D^\alpha u_N(x) + \lambda e^{u_N(x)}.$$

The function $u_N(x)$ is affine in $\mathbf{d}$, as it is a linear combination of the coefficients d_n plus a constant term. The Caputo derivative D^α is a linear operator, so $D^\alpha u_N(x)$ is also affine in $\mathbf{d}$. The exponential function $e^{u_N(x)}$ is the composition of the exponential (which is C^∞) with an affine function, hence C^∞. Therefore, $R(x; \mathbf{d})$ is C^∞ in $\mathbf{d}$, and in particular C^2.

Since all components of $\mathbf{F}$ are C^2 (in fact, C^∞) on $\mathbb{R}^{N+1}$, it follows that $\mathbf{F} \in C^2(\Omega)$ for any open set $\Omega \subseteq \mathbb{R}^{N+1}$, including any open convex set containing $\mathbf{d}^*$. □

Lemma 7 (Properties at the Exact Solution). Let $\mathbf{F} : \mathbb{R}^{N+1} \to \mathbb{R}^{N+1}$ be the residual function defined by (4). If $\mathbf{d}^* \in \mathbb{R}^{N+1}$ is the exact solution vector corresponding to the solution $u(x)$ of equation (1) subject to (2), then:

i. $\mathbf{F}(\mathbf{d}^*) = \mathbf{0}$;

ii. The Jacobian matrix $\mathbf{J}(\mathbf{d}^*)$ is non-singular.

Proof. First, we prove that $\mathbf{F}(\mathbf{d}^*) = \mathbf{0}$.

The residual function $\mathbf{F}(\mathbf{d})$ is constructed such that its components enforce the boundary conditions: $u_N(0) = \beta$, and $u_N(1) = \gamma$, via the first two rows

$$F_0(\mathbf{d}) = \sum_{n=0}^{N} d_n L_n^*(0), \quad F_1(\mathbf{d}) = \sum_{n=0}^{N} d_n L_n^*(1),$$

and the differential equation: $D^\alpha u_N(x) + \lambda e^{u_N(x)} = 0$ at the interior collocation points x_j ($j = 1, \ldots, N-1$) via the remaining rows

$$F_{j+2}(\mathbf{d}) = R(x_j; \mathbf{d}) = D^\alpha u_N(x_j) + \lambda e^{u_N(x_j)}.$$

Since $\mathbf{d}^*$ corresponds to the exact solution $u(x)$, the approximation $u_N(x) = \beta(1-x) + \gamma x + \sum_{n=0}^{N} d_n^* L_n^*(x)$ satisfies that $u_N(0) = \beta$ and $u_N(1) = \gamma$, so $F_0(\mathbf{d}^*) = 0$ and $F_1(\mathbf{d}^*) = 0$. Also, $D^\alpha u_N(x_j) + \lambda e^{u_N(x_j)} = 0$ at each collocation point x_j, so $F_{j+2}(\mathbf{d}^*) = 0$. Hence, all components of $\mathbf{F}(\mathbf{d}^*)$ are zeros.

Second, we prove that $\mathbf{J}(\mathbf{d}^*)$ is non-singular. Assume the contrary, that is $\mathbf{J}(\mathbf{d}^*)$ is singular. Then there exists a nonzero vector $\mathbf{v} = (v_0, v_1, \ldots, v_N)^\top \in \mathbb{R}^{N+1}$ such that $\mathbf{J}(\mathbf{d}^*)\mathbf{v} = \mathbf{0}$.

We define the function $\psi(x) = \sum_{m=0}^{N} v_m L_m^*(x)$. The condition $\mathbf{J}(\mathbf{d}^*)\mathbf{v} = \mathbf{0}$ implies the boundary conditions, from the first two rows of $\mathbf{J}$,

$$\sum_{m=0}^{N} v_m L_m^*(0) = 0, \quad \sum_{m=0}^{N} v_m L_m^*(1) = 0,$$

so, $\psi(0) = 0$ and $\psi(1) = 0$.

In addition, for each collocation point x_j ($j = 1, \ldots, N-1$), the $(j+2)$-th row of $\mathbf{J}$ gives:

$$\sum_{m=0}^{N} v_m \left[D^{\alpha} L_m^*(x_j) + \lambda e^{u_N(x_j)} L_m^*(x_j) \right] = 0.$$

By linearity of D^{α}, this is equivalent to

$$D^{\alpha} \psi(x_j) + \lambda e^{u_N(x_j)} \psi(x_j) = 0.$$

Thus, $\psi(x)$ is a nonzero polynomial (since $\mathbf{v} \neq \mathbf{0}$) of degree at most N that satisfies

$$\psi(0) = 0, \quad \psi(1) = 0 \quad \text{and} \quad D^{\alpha} \psi(x_j) + \lambda e^{u_N(x_j)} \psi(x_j) = 0 \text{ at } N-1 \text{ distinct points } x_j \in (0,\ 1).$$

However, for the fractional Bratu equation with λ below the critical value, the linearized operator $\mathscr{L}\psi = D^{\alpha}\psi + \lambda e^{u(x)}\psi$ is injective on functions satisfying $\psi(0) = \psi(1) = 0$. Since $u_N(x)$ approximates $u(x)$ and the collocation system is well-posed, the only solution to the discrete system is $\psi(x) \equiv 0$, contradicting $\mathbf{v} \neq \mathbf{0}$. Therefore, $\mathbf{J}(\mathbf{d}^*)$ must be non-singular. □

Lemma 8 (Bound on the Inverse Jacobian). For the residual function $\mathbf{F} : \mathbb{R}^{N+1} \to \mathbb{R}^{N+1}$, there exists a constant $K > 0$ such that for all $\mathbf{d}$ in a neighborhood of $\mathbf{d}^*$, where $\mathbf{F}(\mathbf{d}^*) = \mathbf{0}$, the Jacobian matrix $\mathbf{J}(\mathbf{d})$ is invertible and

$$\|\mathbf{J}^{-1}(\mathbf{d})\|_{\infty} \leq K.$$

Proof. For the Jacobian matrix $\mathbf{J}(\mathbf{d})$ given by (5).

Each entry of $\mathbf{J}(\mathbf{d})$ is a continuous function of $\mathbf{d}$: For $i = 0,\ 1$, the entries $L_m^*(0)$ and $L_m^*(1)$ are constants.

For $i \geq 2$, the entries depend on $u_N(x) = \beta(1-x) + \gamma x + \sum_{n=0}^{N} d_n L_n^*(x)$, which is linear in $\mathbf{d}$, so $u_N(x)$ is continuous in $\mathbf{d}$. The exponential function $e^{u_N(x)}$ is continuous in $u_N(x)$, hence continuous in $\mathbf{d}$. The functions $\phi_m(x)$ and $L_m^*(x)$ are polynomials, hence continuous.

From Lemma 7, $\mathbf{J}(\mathbf{d}^*)$ is non-singular, i.e., $\det(\mathbf{J}(\mathbf{d}^*)) \neq 0$.

The determinant $\det(\mathbf{J}(\mathbf{d}))$ is a polynomial in the entries of $\mathbf{J}(\mathbf{d})$, hence continuous. Since $\det(\mathbf{J}(\mathbf{d}^*)) \neq 0$, by continuity, there exists a neighborhood U of $\mathbf{d}^*$ and a constant $\Gamma > 0$ such that

$$|\det(\mathbf{J}(\mathbf{d}))| > \Gamma \quad \text{for all } \mathbf{d} \in U.$$

Thus, $\mathbf{J}(\mathbf{d})$ is invertible for all $\mathbf{d} \in U$.

We show that on U, the entries of $\mathbf{J}(\mathbf{d})$ are uniformly bounded.

For $i = 0,\ 1$, we have $|L_m^*(0)| \leq 2\rho^m$ and $|L_m^*(1)| \leq 2\rho^m$. Since $m \leq N$, $\rho^m \leq \rho^N$, so

$$|L_m^*(0)|,\ |L_m^*(1)| \leq 2\rho^N.$$

For $i \geq 2$, we use the bound on d_n from Lemma 5 to get

$$|d_n| \leq 2\rho^n \frac{(A/2)^n e^{3A/2}}{n!},$$

and thus, we can write

$$|u_N(x)| \leq |\beta(1-x)+\gamma x| + \sum_{n=0}^{N} |d_n||L_n^*(x)| \leq \max(|\beta|,\ |\gamma|) + \sum_{n=0}^{N} \left(2\rho^n \frac{(A/2)^n e^{3A/2}}{n!}\right)(2\rho^n).$$

Simplifying,

$$|u_N(x)| \leq \max(|\beta|,\ |\gamma|) + 4e^{3A/2} \sum_{n=0}^{N} \frac{(A\rho^2/2)^n}{n!} \leq \max(|\beta|,\ |\gamma|) + 4e^{3A/2} e^{A\rho^2/2} = \max(|\beta|,\ |\gamma|) + 4e^{A(3+\rho^2)/2}.$$

Let $M_u = \max(|\beta|,\ |\gamma|) + 4e^{A(3+\rho^2)/2}$. Then

$$e^{u_N(x)} \leq e^{M_u}.$$

For $\phi_m(x)$, note that for $x \in [0,\ 1]$, $x^{k-\alpha} \leq 1$ for $k \geq \lceil \alpha \rceil$. Then

$$|\phi_m(x)| \leq m \sum_{k=\lceil \alpha \rceil}^{m} \frac{(m+k-1)!}{(m-k)!(2k)!} \frac{k!}{\Gamma(k-\alpha+1)} 2^{k+1}.$$

This is a finite sum. For each m, let $M_\phi(m)$ be an upper bound, and let $M_\phi = \max\limits_{0 \leq m \leq N} M_\phi(m)$.
Also, $|L_m^*(x)| \leq 2\rho^m \leq 2\rho^N$.
Therefore, for $i \geq 2$, we have

$$\left|\frac{\partial F_i}{\partial d_m}\right| = \left|\lambda e^{u_N(x_{i-2})} L_m^*(x_{i-2})\right| \leq \lambda e^{M_u} \cdot 2\rho^N, \quad m < \lceil \alpha \rceil,$$

$$\left|\frac{\partial F_i}{\partial d_m}\right| \leq |\phi_m(x_{i-2})| + \lambda e^{u_N(x_{i-2})} |L_m^*(x_{i-2})| \leq M_\phi + \lambda e^{M_u} \cdot 2\rho^N,\ m \geq \lceil \alpha \rceil.$$

Thus, there exists a constant $M > 0$ such that

$$|J_{i,\ j}(\mathbf{d})| \leq M \quad \text{for all } i,\ j \text{ and all } \mathbf{d} \in U.$$

For $\mathbf{d} \in U$, the inverse matrix $\mathbf{J}^{-1}(\mathbf{d})$ has entries

$$(\mathbf{J}^{-1}(\mathbf{d}))_{ij} = \frac{C_{ji}(\mathbf{d})}{\det(\mathbf{J}(\mathbf{d}))},$$

where $C_{ji}(\mathbf{d})$ is the $(j,\ i)$-cofactor of $\mathbf{J}(\mathbf{d})$. Each cofactor is a determinant of an $N \times N$ submatrix of $\mathbf{J}(\mathbf{d})$, so by the Leibniz formula

$$|C_{ji}(\mathbf{d})| \leq N! M^N.$$

Since $|\det(\mathbf{J}(\mathbf{d}))| > \Gamma$, we have

$$|(\mathbf{J}^{-1}(\mathbf{d}))_{ij}| \leq \frac{N! M^N}{\Gamma}.$$

The infinity norm of a matrix is the maximum row sum of absolute values

$$\|\mathbf{J}^{-1}(\mathbf{d})\|_\infty = \max_{1 \leq i \leq N+1} \sum_{j=1}^{N+1} |(\mathbf{J}^{-1}(\mathbf{d}))_{ij}| \leq (N+1)\frac{N! M^N}{\Gamma} =: K.$$

Therefore, for all $\mathbf{d} \in U$,

$$\|\mathbf{J}^{-1}(\mathbf{d})\|_\infty \leq K.$$

This completes the proof. □

Lemma 9 (Lipschitz Continuity of the Jacobian). Let $\mathbf{J}(\mathbf{d})$ be the Jacobian matrix of the residual function $\mathbf{F}(\mathbf{d})$ as defined in (5). Then there exists a constant $L > 0$ such that for all $\mathbf{d}_1,\ \mathbf{d}_2$ in the neighborhood of $\mathbf{d}^*$

$$\|\mathbf{J}(\mathbf{d}_1) - \mathbf{J}(\mathbf{d}_2)\|_\infty \leq L\|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

Proof. The Jacobian matrix $\mathbf{J}(\mathbf{d})$ has entries:
For $i = 0,\ 1$:

$$\frac{\partial F_0}{\partial d_m} = L_m^*(0), \quad \frac{\partial F_1}{\partial d_m} = L_m^*(1), \quad m = 0,\ 1,\ \ldots,\ N.$$

These are constants, independent of $\mathbf{d}$.
For $i \geq 2$, let $j = i - 2$, so $x = x_j$. Then

$$\frac{\partial F_i}{\partial d_m} = \begin{cases} \lambda e^{u_N(x)} L_m^*(x), & m < \lceil \alpha \rceil, \\ \\ \phi_m(x) + \lambda e^{u_N(x)} L_m^*(x), & m \geq \lceil \alpha \rceil, \end{cases} \qquad m = 0,\ 1,\ \ldots,\ N,$$

where $\phi_m(x)$ is independent of $\mathbf{d}$.

For $i = 0,\ 1$, the entries are constant, so:

$$\left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| = 0.$$

For $i \geq 2$, the difference is

$$\left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| = \begin{cases} \lambda \left| L_m^*(x) \right| \left| e^{u_N(x;\, \mathbf{d}_1)} - e^{u_N(x;\, \mathbf{d}_2)} \right|, & m < \lceil \alpha \rceil, \\ \\ \lambda \left| L_m^*(x) \right| \left| e^{u_N(x;\, \mathbf{d}_1)} - e^{u_N(x;\, \mathbf{d}_2)} \right|, & m \geq \lceil \alpha \rceil, \end{cases}$$

since $\phi_m(x)$ cancels out. Thus, for all m:

$$\left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| = \lambda |L_m^*(x)| \left| e^{u_N(x;\, \mathbf{d}_1)} - e^{u_N(x;\, \mathbf{d}_2)} \right|.$$

For each row $i \geq 2$, the sum of absolute differences is:

$$\sum_{m=0}^{N} \left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| = \lambda \left| e^{u_N(x;\, \mathbf{d}_1)} - e^{u_N(x;\, \mathbf{d}_2)} \right| \sum_{m=0}^{N} |L_m^*(x)|.$$

Using the bound $|L_m^*(x)| \leq 2\rho^m$, we have

$$\sum_{m=0}^{N} |L_m^*(x)| \leq 2 \sum_{m=0}^{N} \rho^m = 2 \cdot \frac{\rho^{N+1} - 1}{\rho - 1} =: C_L.$$

Thus

$$\sum_{m=0}^{N} \left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| \leq \lambda C_L \left| e^{u_N(x;\, \mathbf{d}_1)} - e^{u_N(x;\, \mathbf{d}_2)} \right|. \tag{6}$$

Since $u_N(x)$ is linear in $\mathbf{d}$, we have

$$|u_N(x;\,\mathbf{d}_1) - u_N(x;\,\mathbf{d}_2)| \le \sum_{n=0}^{N} |L_n^*(x)||d_{1,\,n} - d_{2,\,n}| \le C_L \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

By the mean value theorem and the bound $|u_N(x)| \le M_u$ from Lemma 8

$$\left| e^{u_N(x;\,\mathbf{d}_1)} - e^{u_N(x;\,\mathbf{d}_2)} \right| \le e^{M_u} |u_N(x;\,\mathbf{d}_1) - u_N(x;\,\mathbf{d}_2)| \le e^{M_u} C_L \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

Substituting back into (6), we obtain

$$\sum_{m=0}^{N} \left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| \le \lambda C_L \cdot e^{M_u} C_L \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty = \lambda e^{M_u} C_L^2 \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

For rows $i = 0,\ 1$, the sum is 0.
The infinity norm of a matrix is the maximum row sum of absolute values. Therefore:

$$\|\mathbf{J}(\mathbf{d}_1) - \mathbf{J}(\mathbf{d}_2)\|_\infty = \max_{0 \le i \le N} \sum_{m=0}^{N} \left| \frac{\partial F_i}{\partial d_m}(\mathbf{d}_1) - \frac{\partial F_i}{\partial d_m}(\mathbf{d}_2) \right| \le \lambda e^{M_u} C_L^2 \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

Let $L = \lambda e^{M_u} C_L^2$. Then

$$\|\mathbf{J}(\mathbf{d}_1) - \mathbf{J}(\mathbf{d}_2)\|_\infty \le L \|\mathbf{d}_1 - \mathbf{d}_2\|_\infty.$$

□

Theorem 2 (Local Quadratic Convergence of Newton's Method). Let $\mathbf{F}$ be the residual function and $\mathbf{J}$ be its Jacobian defined in (4) and (5) respectively such that satisfies the conditions in Lemmas 6-9. Then there exists $\zeta > 0$ such that for any initial guess $\mathbf{d}^{(0)}$ with $\|\mathbf{d}^{(0)} - \mathbf{d}^*\|_\infty < \zeta$, the Newton-Raphson iteration

$$\mathbf{d}^{(\eta+1)} = \mathbf{d}^{(\eta)} - \mathbf{J}^{-1}(\mathbf{d}^{(\eta)})\mathbf{F}(\mathbf{d}^{(\eta)}),$$

converges quadratically to $\mathbf{d}^*$. Specifically, there is a constant $C > 0$ such that

$$\|\mathbf{d}^{(\eta+1)} - \mathbf{d}^*\|_\infty \le C \|\mathbf{d}^{(\eta)} - \mathbf{d}^*\|_\infty^2 \quad \text{for all } \eta \ge 0.$$

Proof. Let $\mathbf{e}^{(\eta)} = \mathbf{d}^{(\eta)} - \mathbf{d}^*$. Since $\mathbf{F} \in C^1(\Omega)$ and $\mathbf{F}(\mathbf{d}^*) = \mathbf{0}$, the integral form of the mean value theorem gives

$$\mathbf{F}(\mathbf{d}^{(\eta)}) = \int_0^1 \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\mathbf{e}^{(\eta)} dt.$$

The Newton iteration step is

$$\mathbf{d}^{(\eta+1)} = \mathbf{d}^{(\eta)} - \mathbf{J}^{-1}(\mathbf{d}^{(\eta)})\mathbf{F}(\mathbf{d}^{(\eta)}).$$

Substituting into the error expression,

$$\mathbf{e}^{(\eta+1)} = \mathbf{e}^{(\eta)} - \mathbf{J}^{-1}(\mathbf{d}^{(\eta)})\mathbf{F}(\mathbf{d}^{(\eta)}) = \mathbf{J}^{-1}(\mathbf{d}^{(\eta)}) \left[\mathbf{J}(\mathbf{d}^{(\eta)})\mathbf{e}^{(\eta)} - \mathbf{F}(\mathbf{d}^{(\eta)})\right].$$

Using the expression for $\mathbf{F}(\mathbf{d}^{(\eta)})$,

$$\mathbf{J}(\mathbf{d}^{(\eta)})\mathbf{e}^{(\eta)} - \mathbf{F}(\mathbf{d}^{(\eta)}) = \int_0^1 \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} dt,$$

and thus, we have

$$\mathbf{e}^{(\eta+1)} = \mathbf{J}^{-1}(\mathbf{d}^{(\eta)}) \int_0^1 \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} dt.$$

Taking the infinity norm,

$$\|\mathbf{e}^{(\eta+1)}\|_\infty \leq \|\mathbf{J}^{-1}(\mathbf{d}^{(\eta)})\|_\infty \left\| \int_0^1 \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} dt \right\|_\infty.$$

By the triangle inequality for integrals,

$$\left\| \int_0^1 \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} dt \right\|_\infty \leq \int_0^1 \left\| \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} \right\|_\infty dt.$$

For each $t \in [0,\ 1]$

$$\left\| \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\right] \mathbf{e}^{(\eta)} \right\|_\infty \leq \|\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\|_\infty \|\mathbf{e}^{(\eta)}\|_\infty.$$

By the Lipschitz condition

$$\|\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)})\|_\infty \leq L\|\mathbf{d}^{(\eta)} - (\mathbf{d}^* + t\mathbf{e}^{(\eta)})\|_\infty = L\|(1-t)\mathbf{e}^{(\eta)}\|_\infty = L(1-t)\|\mathbf{e}^{(\eta)}\|_\infty.$$

Therefore,

$$\left\| \left[\mathbf{J}(\mathbf{d}^{(\eta)}) - \mathbf{J}(\mathbf{d}^* + t\mathbf{e}^{(\eta)}) \right] \mathbf{e}^{(\eta)} \right\|_\infty \leq L(1-t)\|\mathbf{e}^{(\eta)}\|_\infty^2.$$

Integrating:

$$\int_0^1 L(1-t)\|\mathbf{e}^{(\eta)}\|_\infty^2 dt = L\|\mathbf{e}^{(\eta)}\|_\infty^2 \int_0^1 (1-t)dt = \frac{L}{2}\|\mathbf{e}^{(\eta)}\|_\infty^2.$$

Also, by the bound on the inverse Jacobian:

$$\|\mathbf{J}^{-1}(\mathbf{d}^{(\eta)})\|_\infty \leq K.$$

Combining:

$$\|\mathbf{e}^{(\eta+1)}\|_\infty \leq K \cdot \frac{L}{2}\|\mathbf{e}^{(\eta)}\|_\infty^2 = \frac{KL}{2}\|\mathbf{e}^{(\eta)}\|_\infty^2.$$

Let $C = \dfrac{KL}{2}$. Then

$$\|\mathbf{e}^{(\eta+1)}\|_\infty \leq C\|\mathbf{e}^{(\eta)}\|_\infty^2.$$

To ensure the iterates remain in Ω, choose $\zeta > 0$ such that the closed ball $B(\mathbf{d}^*,\ \zeta) \subseteq \Omega$ and $C\zeta < 1$.
If $\|\mathbf{e}^{(0)}\|_\infty < \zeta$, then

$$\|\mathbf{e}^{(1)}\|_\infty \leq C\|\mathbf{e}^{(0)}\|_\infty^2 \leq C\zeta^2 < \zeta,$$

so $\mathbf{d}^{(1)} \in B(\mathbf{d}^*,\ \zeta) \subseteq \Omega$.

By induction, if $\|\mathbf{e}^{(\eta)}\|_\infty < \zeta$, then

$$\|\mathbf{e}^{(\eta+1)}\|_\infty \leq C\|\mathbf{e}^{(\eta)}\|_\infty^2 \leq C\zeta^2 < \zeta,$$

so all iterates remain in $B(\mathbf{d}^*,\ \zeta) \subseteq \Omega$, and the inequality holds for all $\eta \geq 0$. This implies quadratic convergence. □

Corollary 2 From the inequality $\|\mathbf{e}^{(\eta+1)}\|_\infty \leq C\|\mathbf{e}^{(\eta)}\|_\infty^2$, since $\|\mathbf{e}^{(0)}\|_\infty < \zeta \leq \dfrac{1}{2C}$, we have $C\|\mathbf{e}^{(0)}\|_\infty < \dfrac{1}{2}$, and thus

$$\|\mathbf{e}^{(\eta)}\|_\infty \leq \frac{1}{C}(C\|\mathbf{e}^{(0)}\|_\infty)^{2^\eta} = C^{2\eta-1}\|\mathbf{e}^{(0)}\|^{2\eta},$$

which tends to zero as $\eta \to \infty$.

6. Numerical examples

This section displays some numerical examples to ensure the applicability and efficiency of our proposed numerical algorithm.

Example 1 Consider the following equation:

$$D^{\alpha}u + e^{u} = 0; \quad 0 \leq x \leq 1,$$

subject to the boundary conditions:

$$u(0) = u(1) = 0,$$

where the exact solution is $u(x) = -2\log\left(\dfrac{\cosh\left(\dfrac{\theta}{2}\left(x-\dfrac{1}{2}\right)\right)}{\cosh\left(\dfrac{\theta}{4}\right)}\right)$ at $\theta = 1.51716$ when $\alpha = 2$.

Table 1 displays the maximum absolute errors (MAEs) resulting from the application of our method in addition to the CPU time needed to implement the algorithm for various N. Table 2 provides a detailed convergence analysis showing the residual norms and convergence orders for the Newton-Raphson iterations with $N = 10$, demonstrating near-quadratic convergence with order approaching 2.00077 by the fourth iteration. In contrast, Table 3 displays a comparison of the exact solution, the approximate solution at $N = 17$, and the Absolute Errors (AEs) of different methods. Figure 2 displays a numerical study of the obtained spectral approximation for the case corresponding to $\lambda = 1$. In Figure 2a, the comparison between numerical solutions for various polynomial degrees N and the exact solution shows excellent agreement across the entire domain $x \in [0,\ 1]$, with all approximate solution curves virtually indistinguishable from the exact solution. The logarithmic plot of absolute point-wise errors in Figure 2b shows that for large values of N, for example $(N = 14,\ 17)$, a high-order precision $(\sim 10^{-16})$. The convergence behavior illustrated in Figure 2c demonstrates exponential convergence of the maximum absolute error as a function of polynomial degree N until machine precision is reached at $N \approx 14$, after which the error plateaus at approximately 10^{-16}. Figure 2d illustrates the non-linear relationship between polynomial degree (N) and CPU time, showing that computation time increases at an accelerating rate. These results confirm the theoretical exponential convergence rate expected for smooth solutions and validate that the method achieves machine precision for polynomial degrees $N \geq 14$.

Figure 3 shows an investigation of the Newton-Raphson convergence for Example 1 with $N = 10$. In Figure 3a, the error decline that isn't monotonic stabilises at about 1.5487×10^{-11}. Figure 3b shows how spatial errors change during the first six iterations. It shows that the patterns are always the same. Figures 3c and d demonstrate that after rapid initial convergence, the error stabilizes in both shape and magnitude, with iterations beyond $\eta \approx 10$ yielding no further accuracy improvement.

Figure 4 further illustrates the Newton-Raphson convergence dynamics for Example 1 at $N = 10$, providing a complementary visualization of the update norm convergence in panel (a) and the estimated convergence order evolution in panel (b). The convergence order rapidly approaches the theoretical quadratic value of 2, validating the convergence analysis presented in Theorem 2.

Figure 5 shows the numerical solution and error analysis for Example 1, where $\lambda = 1$ and $N = 17$. Figure 5a illustrates that the approximate and exact solutions are quite close to each other for all values of x in the range $[0,\ 1]$. Figure 5b shows a pointwise absolute error that is limited by machine accuracy (around 4.5×10^{-16}). It exhibits high-frequency oscillations, a common feature of spectral approaches, due to the spread of collocation points.

Table 1. Maximum absolute error and CPU time of Example 1 for different values of N

N	5	8	11	14	17
Maximum AE	9.40398×10^{-6}	1.59176×10^{-9}	1.33164×10^{-11}	8.46545×10^{-16}	2.498×10^{-16}
CPU time (s)	0.113121	0.401707	1.49257	3.26759	6.4302

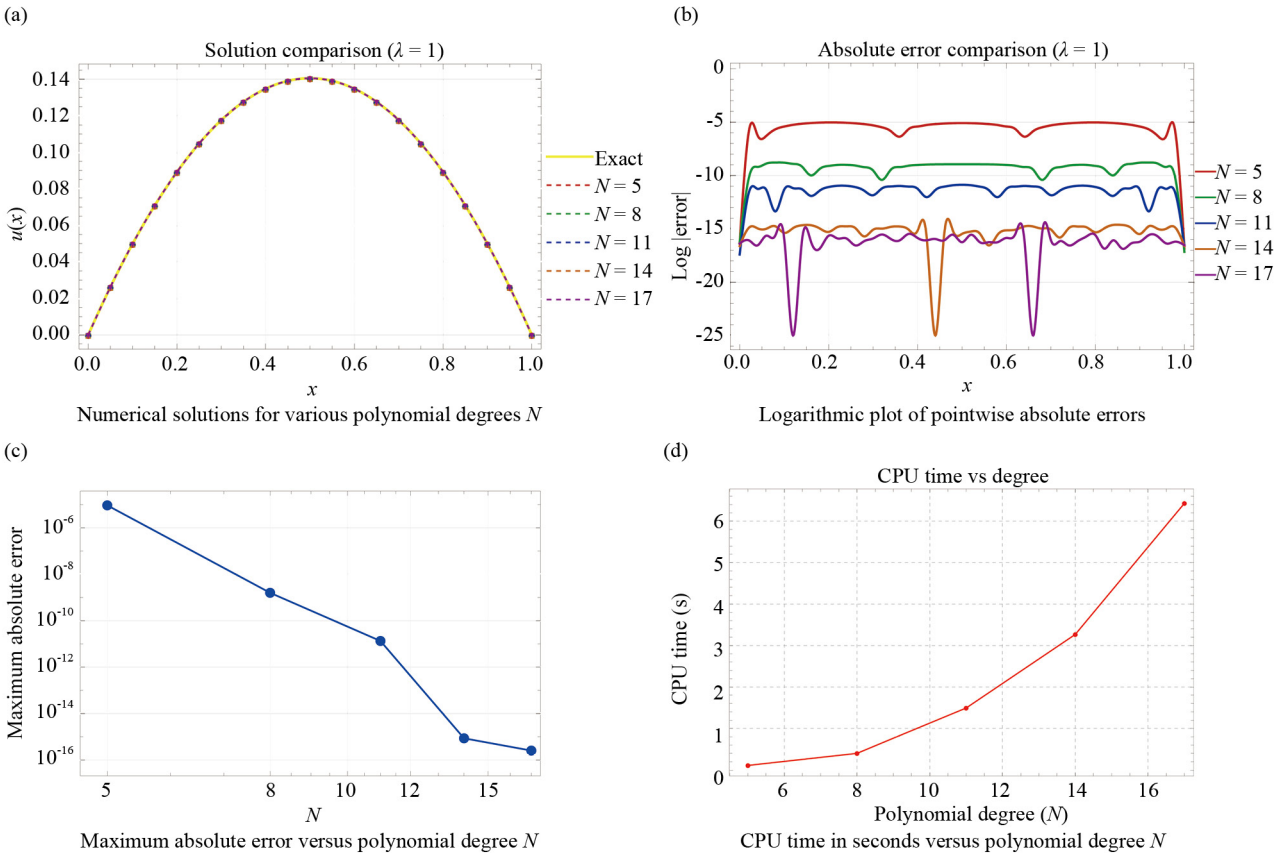
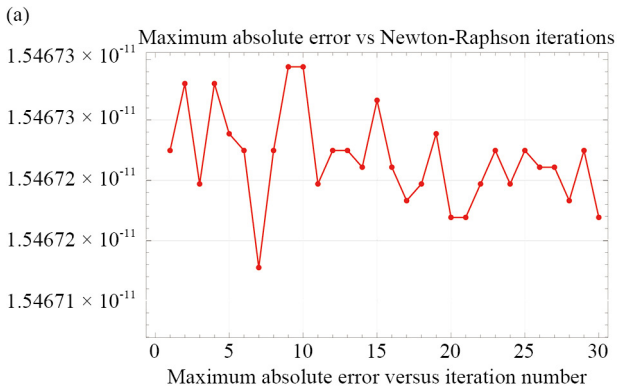


Figure 2. Spectral approximation results for Example 1



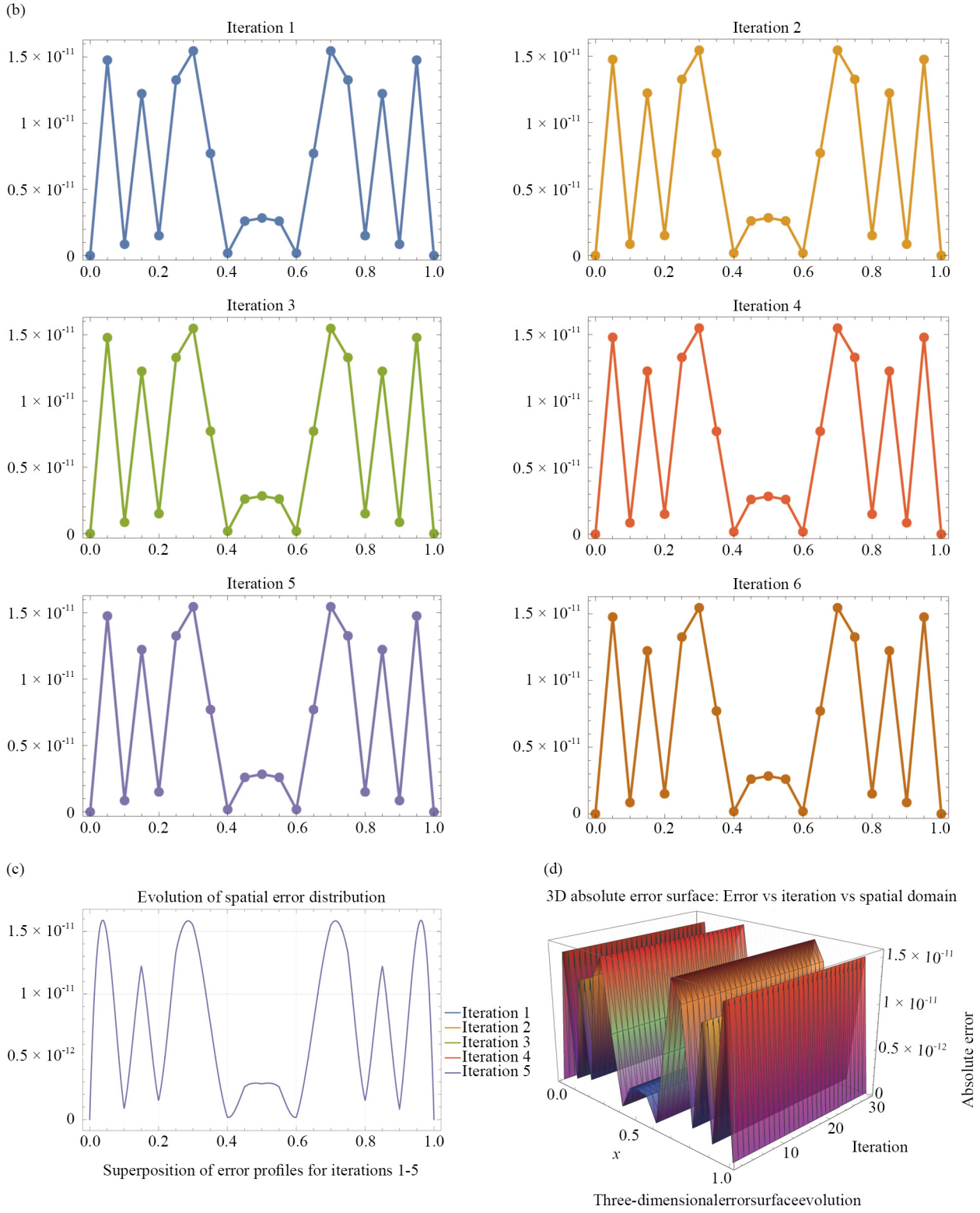


Figure 3. Newton-Raphson convergence analysis for Example 1 at $N = 10$

Table 2. Example 1 convergence orders and residuals' norms with the corresponding iterations for $N = 10$

Iteration	Residual norm $\|\|\mathbf{F}^{(\eta)}\|\|$	Convergence order
1	1	-
2	1.67134×10^{-2}	-
3	9.63107×10^{-7}	1.88078
4	3.17418×10^{-15}	2.00077

Table 3. Example 1 comparison of exact solution, approximate solution at $N = 17$, and absolute errors of various methods

x	Exact Sol.	Approx. Sol.	Presented method	L-RKM [54]	ANN-SOS [55]	RCNN [56]	Green's Func. Tech. [57]	Adomian Decomp. [58]
0.1	0.0498468	0.0498468	1.38778×10^{-16}	5.53×10^{-17}	1.32×10^{-6}	2.5726×10^{-7}	1.8×10^{-12}	2.887×10^{-9}
0.2	0.0891899	0.0891899	2.22045×10^{-16}	1.09×10^{-16}	6.33×10^{-7}	5.8002×10^{-7}	4.1×10^{-12}	5.745×10^{-9}
0.3	0.117609	0.117609	1.38778×10^{-16}	1.63×10^{-16}	3.29×10^{-6}	7.5634×10^{-7}	6.4×10^{-12}	8.539×10^{-9}
0.4	0.13479	0.13479	1.38778×10^{-16}	2.15×10^{-16}	2.81×10^{-6}	7.3394×10^{-7}	8.2×10^{-12}	1.123×10^{-8}
0.5	0.140539	0.140539	5.55112×10^{-17}	2.64×10^{-16}	4.75×10^{-7}	7.0454×10^{-7}	9.1×10^{-12}	1.380×10^{-8}
0.6	0.13479	0.13479	1.38778×10^{-16}	3.11×10^{-16}	1.48×10^{-6}	7.7639×10^{-7}	8.8×10^{-12}	1.622×10^{-8}
0.7	0.117609	0.117609	8.32667×10^{-17}	3.54×10^{-16}	2.31×10^{-6}	8.0204×10^{-7}	7.4×10^{-12}	1.843×10^{-8}
0.8	0.0891899	0.0891899	2.498×10^{-16}	3.92×10^{-16}	2.26×10^{-6}	5.8306×10^{-7}	5.2×10^{-12}	2.024×10^{-8}
0.9	0.0498468	0.0498468	6.245×10^{-17}	4.27×10^{-16}	8.62×10^{-7}	2.2869×10^{-7}	2.5×10^{-12}	1.961×10^{-8}

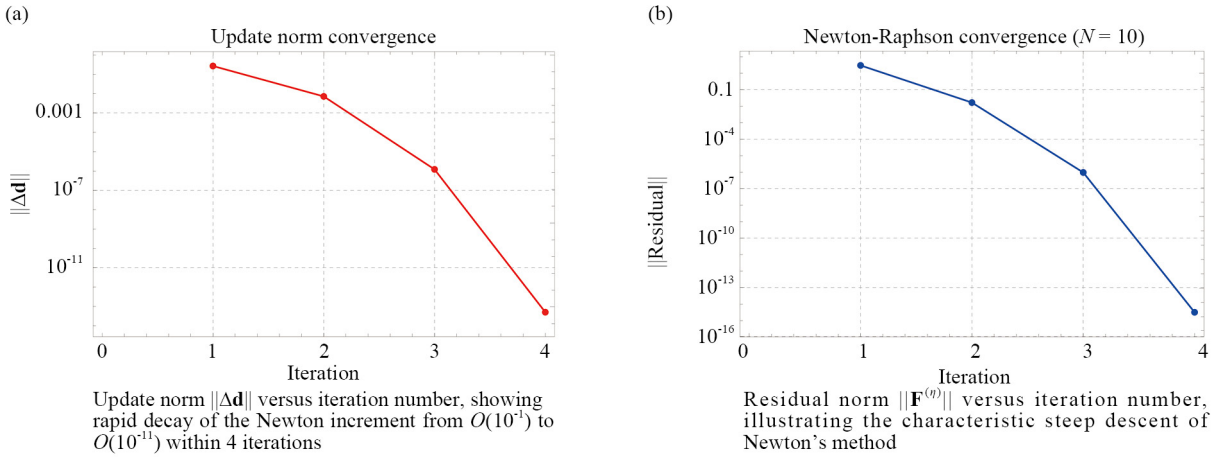


Figure 4. Newton-Raphson iterations convergence analysis of Example 1 for $N = 10$

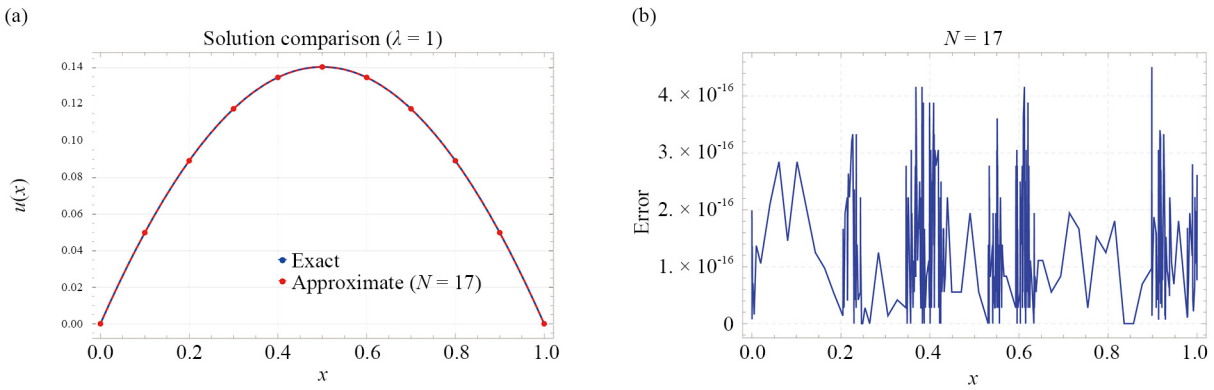


Figure 5. Numerical solution and error analysis for Example 1 with $\lambda = 1$ and $N = 17$

Example 2 Consider the following equation:

$$D^{\alpha}u + 3.51e^{u} = 0; \quad 0 \leq x \leq 1,$$

subject to the boundary conditions:

$$u(0) = u(1) = 0, \tag{7}$$

where the exact solution is $u(x) = -2\log\left(\dfrac{\cosh\left(\dfrac{\theta}{2}\left(x - \dfrac{1}{2}\right)\right)}{\cosh\left(\dfrac{\theta}{4}\right)}\right)$ at $\theta = 4.66781$ when $\alpha = 2$.

Table 4 presents the comparison of numerical solutions and absolute errors for Example 2 across different methods.

In Table 5, all maximum errors happened at position $x = 1/2$. The error values show stable convergence. Maximum errors varied between 2.22853×10^{-10} and 2.26195×10^{-10}, while average errors ranged from 1.27211×10^{-10} to 1.2922×10^{-10}. In addition, from Figure 6, The method showed convergence with errors around 10^{-10}.

Table 6 quantifies the convergence behavior with residual norms and convergence orders for successive Newton-Raphson iterations, showing the transition from linear (order ≈ 0.78735) to near-quadratic convergence (order ≈ 1.98378) within eight iterations. Figure 7 presents a comprehensive error analysis of the Newton-Raphson method for Example 2 with $N = 20$, using an extremely low tolerance of 10^{-20} to demonstrate the method's stability and robustness even under extended iteration counts.

Table 4. Example 2 comparison of numerical solutions and absolute errors for various methods

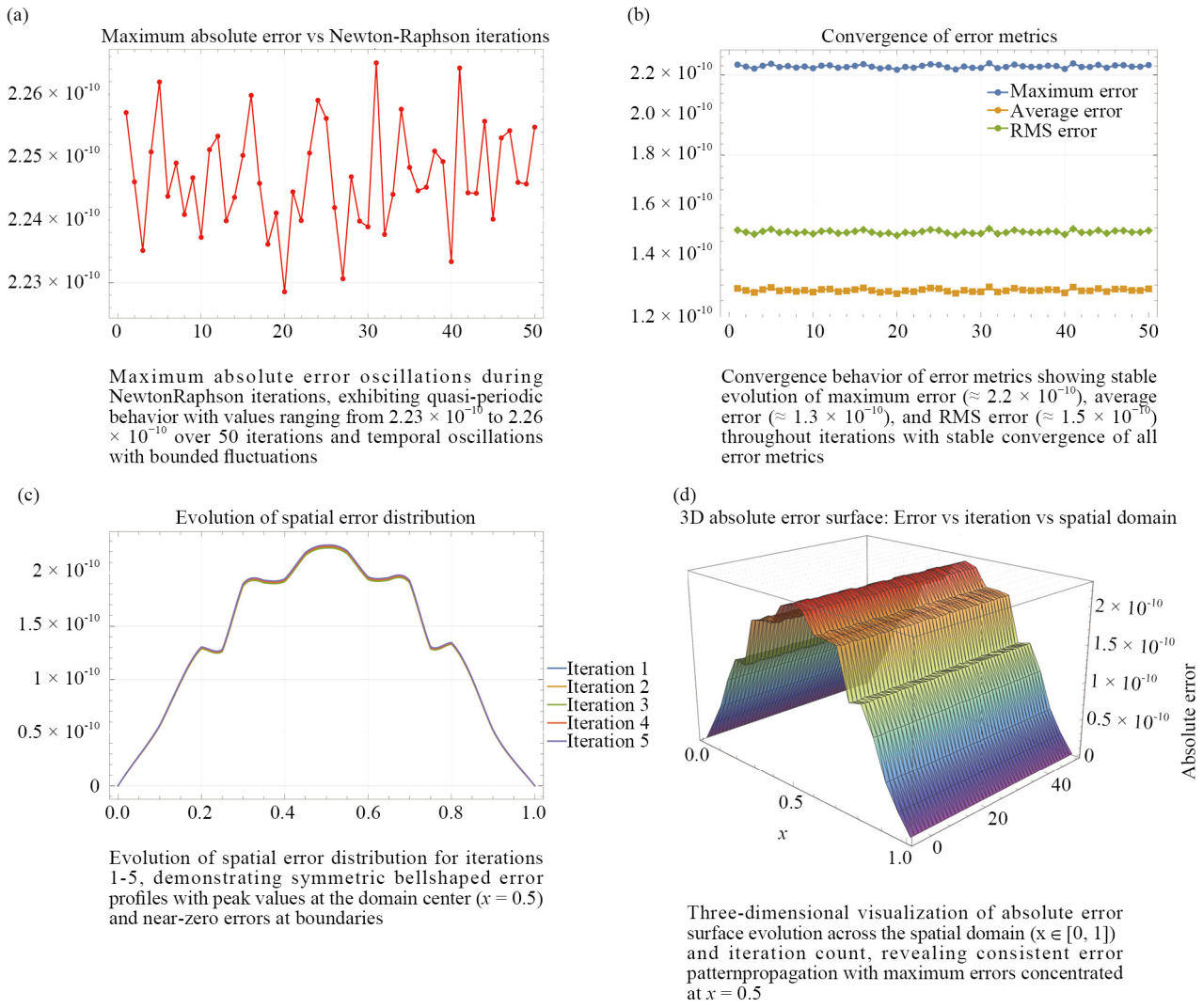
x	Exact solution	Approx. solution	Presented method ($N = 20$)	Genocchi polynomials [59]	GFIM [57]	Taylor wavelets [60] ($k = 1$, $M = 11$)	RCNN [56] ($\lambda = 3.51$)	ANN-SOS [55] ($k = 1$, $M = 11$)
0.1	0.364336	0.364336	5.67795×10^{-11}	3.318×10^{-5}	2.0×10^{-5}	2.49×10^{-7}	1.5627×10^{-5}	1.79×10^{-4}
0.2	0.67787	0.67787	1.30128×10^{-10}	2.551×10^{-5}	4.0×10^{-5}	2.61×10^{-7}	1.2411×10^{-5}	3.18×10^{-4}
0.3	0.922214	0.922214	1.89501×10^{-10}	2.005×10^{-4}	5.6×10^{-5}	2.68×10^{-7}	4.9981×10^{-5}	4.38×10^{-4}
0.4	1.07863	1.07863	1.94062×10^{-10}	1.677×10^{-4}	6.8×10^{-5}	1.25×10^{-6}	3.3924×10^{-5}	5.22×10^{-4}
0.5	1.13262	1.13262	2.25535×10^{-10}	1.568×10^{-4}	7.2×10^{-5}	1.31×10^{-6}	7.8505×10^{-6}	5.46×10^{-4}
0.6	1.07863	1.07863	1.9412×10^{-10}	1.677×10^{-4}	6.3×10^{-5}	1.25×10^{-6}	2.7385×10^{-5}	5.16×10^{-4}
0.7	0.922214	0.922214	1.85813×10^{-10}	2.005×10^{-4}	5.6×10^{-5}	2.68×10^{-7}	5.0464×10^{-5}	4.35×10^{-4}
0.8	0.67787	0.67787	1.44922×10^{-10}	2.551×10^{-4}	4.0×10^{-5}	2.61×10^{-7}	2.6432×10^{-5}	3.11×10^{-4}
0.9	0.364336	0.364336	4.92833×10^{-11}	3.318×10^{-4}	2.0×10^{-5}	2.49×10^{-7}	1.6634×10^{-6}	1.68×10^{-4}

Table 5. Example 2 shows the convergence of the iterative Newton-Raphson method across 50 iterations for $N = 20$. The maximum absolute error and average error at each 5-iteration step are in the table

Iteration	Maximum error	Average error
1	2.25709×10^{-10}	1.28918×10^{-10}
5	2.26195×10^{-10}	1.2922×10^{-10}
10	2.2372×10^{-10}	1.27729×10^{-10}
15	2.25027×10^{-10}	1.28536×10^{-10}
20	2.22853×10^{-10}	1.27211×10^{-10}
25	2.25616×10^{-10}	1.2887×10^{-10}
30	2.23889×10^{-10}	1.2783×10^{-10}
35	2.24835×10^{-10}	1.28411×10^{-10}
40	2.23334×10^{-10}	1.27502×10^{-10}
45	2.24011×10^{-10}	1.27904×10^{-10}
50	2.25476×10^{-10}	1.28788×10^{-10}

Table 6. Example 2 convergence orders and residuals' norms against the corresponding iterations

Iteration	Residual norm $\|\|\mathbf{F}^{(\eta)}\|\|$	Convergence order
1	0.152997	-
2	2.3806	-
3	0.55018	0.78735
4	0.125359	1.0097
5	2.411×10^{-2}	1.11459
6	2.4414×10^{-3}	1.38913
7	3.97089×10^{-5}	1.79854
8	1.123059×10^{-8}	1.98378



Maximum absolute error oscillations during NewtonRaphson iterations, exhibiting quasi-periodic behavior with values ranging from 2.23×10^{-10} to 2.26×10^{-10} over 50 iterations and temporal oscillations with bounded fluctuations

Convergence behavior of error metrics showing stable evolution of maximum error ($\approx 2.2 \times 10^{-10}$), average error ($\approx 1.3 \times 10^{-10}$), and RMS error ($\approx 1.5 \times 10^{-10}$) throughout iterations with stable convergence of all error metrics

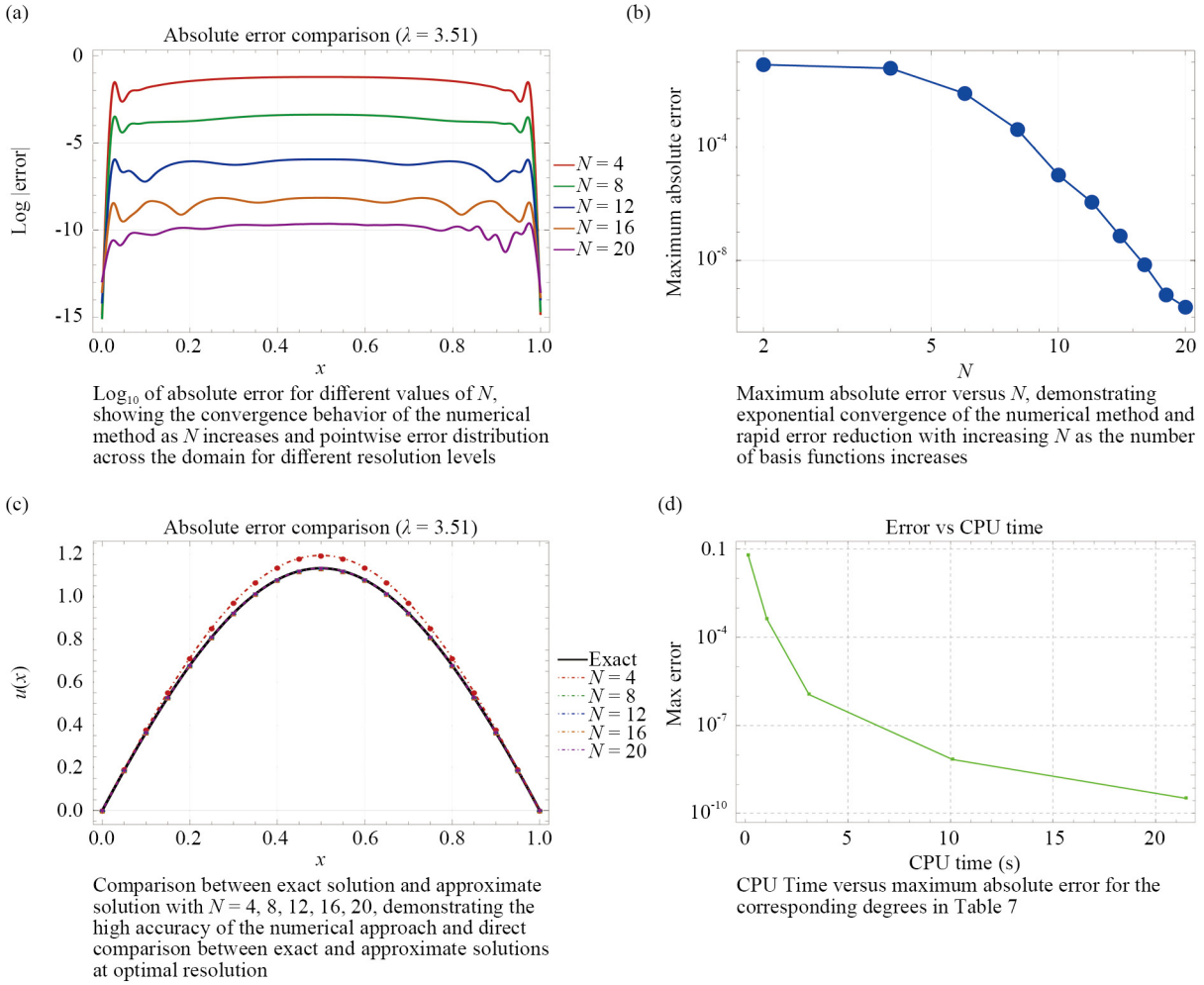
Evolution of spatial error distribution for iterations 1-5, demonstrating symmetric bellshaped error profiles with peak values at the domain center ($x = 0.5$) and near-zero errors at boundaries

Three-dimensional visualization of absolute error surface evolution across the spatial domain ($x \in [0, 1]$) and iteration count, revealing consistent error patternpropagation with maximum errors concentrated at $x = 0.5$

Figure 6. An error analysis of the Newton-Raphson iterative method in Example 2 was performed, using a tolerance of 10^{-20} for $N = 20$

Table 7. Example 2 maximum absolute error and CPU time for different values of N when $\lambda = 3.51$

N	4	8	12	16	20
Max. A. E.	5.99351×10^{-2}	4.13964×10^{-4}	1.13476×10^{-6}	7.04313×10^{-9}	2.25535×10^{-10}
CPU time (s)	0.140218	1.04936	3.10342	10.1049	21.4841



Log_{10} of absolute error for different values of N, showing the convergence behavior of the numerical method as N increases and pointwise error distribution across the domain for different resolution levels

Maximum absolute error versus N, demonstrating exponential convergence of the numerical method and rapid error reduction with increasing N as the number of basis functions increases

Comparison between exact solution and approximate solution with $N = 4, 8, 12, 16, 20$, demonstrating the high accuracy of the numerical approach and direct comparison between exact and approximate solutions at optimal resolution

CPU Time versus maximum absolute error for the corresponding degrees in Table 7

Figure 7. An error analysis of the Newton-Raphson iterative method in Example 2 was performed, using a tolerance of 10^{-20} for $N = 20$

Table 7 summarizes the computational performance for Example 2, documenting maximum absolute errors and CPU times for polynomial degrees $N \in \{4,\ 8,\ 12,\ 16,\ 20\}$. The results demonstrate exponential error decay from $\mathcal{O}(10^{-2})$ to $\mathcal{O}(10^{-10})$ as N increases, while CPU time exhibits polynomial growth, consistent with the complexity of solving the Newton-Raphson systems.

Figure 6 illustrates the algorithm's behavior over 50 iterations using an extremely low tolerance (10^{-20}). The purpose is to demonstrate the robustness and stability of the Newton-Raphson method; it is observed that the solution does not diverge (no maximum error inflation), even with a large number of iterations, confirming the method's reliability [61].

In contrast, Figure 8 focuses on the analysis of the rate of convergence. The order of convergence was calculated using only the first 8 iterations with a tolerance of 10^{-10}. It was observed that the residual value reaches a state of stagnation after the eighth iteration. Beyond this point, the convergence becomes non-monotonic and oscillates within a very narrow range. This is an expected phenomenon when the machine precision limits of floating-point arithmetic are reached. This effect is amplified by the large polynomial degree ($N = 20$), as the resulting large-scale system is more susceptible to the accumulation of floating-point errors, which hinders the accurate computation of the residual at this level of precision [62]. Furthermore, this sensitivity is a direct consequence of the physics of the Bratu problem itself. The problem possesses a critical parameter, λ_c, beyond which no real solution exists. As the chosen parameter $\lambda = 3.51$ approaches this critical

value, the Jacobian matrix becomes severely ill-conditioned, magnifying the effects of floating-point errors and directly causing the observed slowdown and oscillation in convergence.

However, it is crucial to note that by the eighth iteration, the calculated order of convergence closely approaches 2, the theoretical value of the ideal quadratic convergence. This confirms the high efficiency and effectiveness of the applied method. It is anticipated that maintaining the quadratic convergence rate to lower error levels would require the use of arbitrary-precision arithmetic [63].

Example 3 Consider the following equation:

$$D^{\alpha}u + e^{u} = 0; \quad 0 \leq x \leq 1,$$

subject to the boundary conditions:

$$u(0) = u(1) = 0, \tag{8}$$

where the exact solution is $u(x) = -2\log\left(\dfrac{\cosh\left(\dfrac{\theta}{2}\left(x-\dfrac{1}{2}\right)\right)}{\cosh\left(\dfrac{\theta}{4}\right)}\right)$ at $\theta = 1.51716$ when $\alpha = 1.5$.

Table 8 shows the rapid convergence of the approximate solution as the order of approximation increases from $N = 3$ to $N = 12$, with the solution values stabilizing to at least 4-5 significant digits and exhibiting the expected symmetry about $x = 0.5$ (maximum value ≈ 0.1405). The boundary conditions are satisfied with increasing accuracy, particularly at $x = 1$, where the error reduces from $\mathcal{O}(10^{-2})$ at $N = 3$ to $\mathcal{O}(10^{-6})$ at $N = 12$. Table 9 validates the method's superior performance by comparing residual errors with the Green's Function-Picard's Fixed Point (GFPF) iteration method; the presented spectral approach achieves residual errors on the order of 10^{-13}, approximately six to seven orders of magnitude smaller than the GFPF method's errors (10^{-6} to 10^{-5}), with the maximum residual occurring near at $x = 0.9$. This significant improvement in accuracy with only $N = 12$ collocation points underscores the efficiency of spectral methods for fractional differential equations with smooth solutions.

Table 8. Example 3 approximate solutions at various N values

x	$N = 3$	$N = 6$	$N = 9$	$N = 12$
0.0	$5.551115123 \times 10^{-17}$	0	0	0
0.1	0.05165009259	0.04983292324	0.04984681147	0.04984678027
0.2	0.09274950319	0.08916273141	0.08918997457	0.08918991339
0.3	0.1228412806	0.1175692253	0.1176091552	0.1176090656
0.4	0.1414684735	0.1347388244	0.1347903349	0.1347902167
0.5	0.1481741308	0.1404777514	0.1405393433	0.1405391728
0.6	0.1425013013	0.1347200106	0.1347906108	0.1347902067
0.7	0.1239930336	0.1175301600	0.1176105810	0.1176090091
0.8	0.09219237662	0.08910087676	0.08919613192	0.08918958854
0.9	0.04664237909	0.04974531592	0.04986961651	0.04984514551
1.0	−0.01311391021	−0.0001157366766	0.00007373727535	$-6.997328434 \times 10^{-6}$

Table 9. Example 3 residual errors comparison between our method at $N = 12$ and the Green's Function-Picard's fixed point iteration method in [64] after 5 iterations

x	Presented method	GFPF [64]
0.1	1.87758×10^{-13}	3.69×10^{-6}
0.2	2.57517×10^{-13}	5.65×10^{-6}
0.3	4.51808×10^{-13}	5.45×10^{-6}
0.4	4.30561×10^{-13}	3.60×10^{-6}
0.5	4.04231×10^{-17}	1.08×10^{-6}
0.6	1.26368×10^{-12}	1.18×10^{-6}
0.7	4.34115×10^{-12}	2.57×10^{-6}
0.8	1.10266×10^{-11}	3.01×10^{-6}
0.9	9.99175×10^{-11}	1.12×10^{-5}

Table 10 documents the computational efficiency for Example 3, showing CPU times for $N = 3, 6, 9$, and 12. The polynomial growth in computational cost is expected given the increasing size of the nonlinear algebraic system.

Table 10. Example 3 CPU time for different values of N

N	3	6	9	12
CPU Time (s)	0.1229621	0.6697874	2.4432161	8.1343948

Figure 9 shows spatial error distributions for polynomial degrees $N \in \{3,\ 6,\ 9,\ 12\}$, illustrating the transition from low- to high-order approximations. Errors decrease significantly with increasing N: from monotonic growth up to 0.15 at $N = 3$ (Figure 9a), to localized exponential growth at $N = 6$ (Figure 9b), a sharp spike near $x \approx 0.95$ at $N = 9$ (Figure 9c), and a smooth, similarly-shaped profile at $N = 12$ (Figure 9d). This progression reveals boundary-layer behavior and non-monotonic convergence, with error magnitudes spanning seven orders of magnitude, demonstrating the dramatic impact of polynomial degree on approximation quality.

Figure 10 demonstrates spectral convergence for polynomial degrees $N \in \{3,\ 6,\ 9,\ 12\}$. Figure 10a shows close agreement between approximate and exact solutions, with only slight deviation at $x = 1$ for $N = 3$. All approximations capture the symmetric bell-shaped profile peaking near $x = 0.5$. Figure 10b displays exponential decay in maximum absolute error, from $\mathcal{O}(10^{-1})$ at $N = 3$ to $\mathcal{O}(10^{-8})$ at $N = 12$, confirming spectral accuracy and rapid convergence for smooth problems.

Figure 11 complements the spectral convergence analysis by illustrating the relationship between computational performance metrics and accuracy. Panel (a) demonstrates the correlation between maximum residual errors and CPU time, showing that achieving machine precision requires approximately 8 seconds for $N = 12$. Panel (b) reveals the polynomial scaling of CPU time with respect to the polynomial degree N, confirming the expected computational complexity of the method.

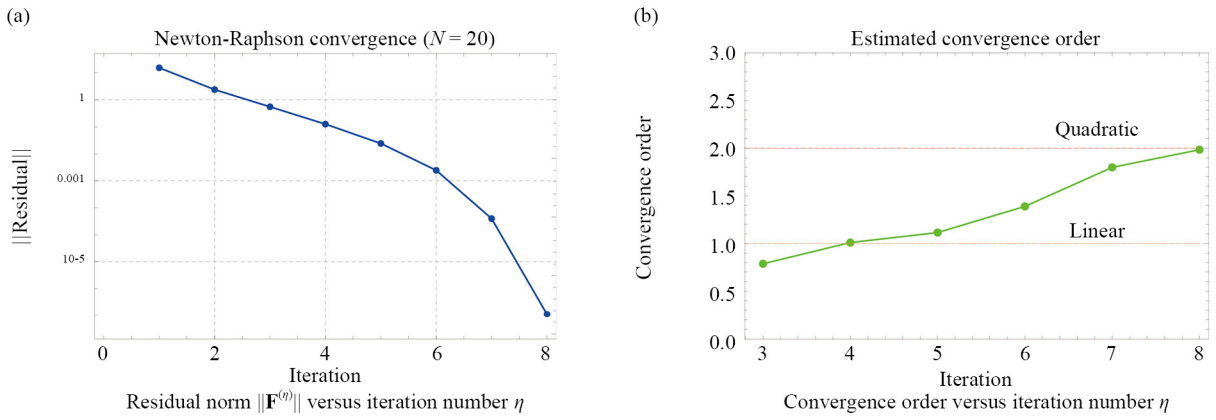


Figure 8. Convergence analysis of the Newton-Raphson iterative method in Example 2

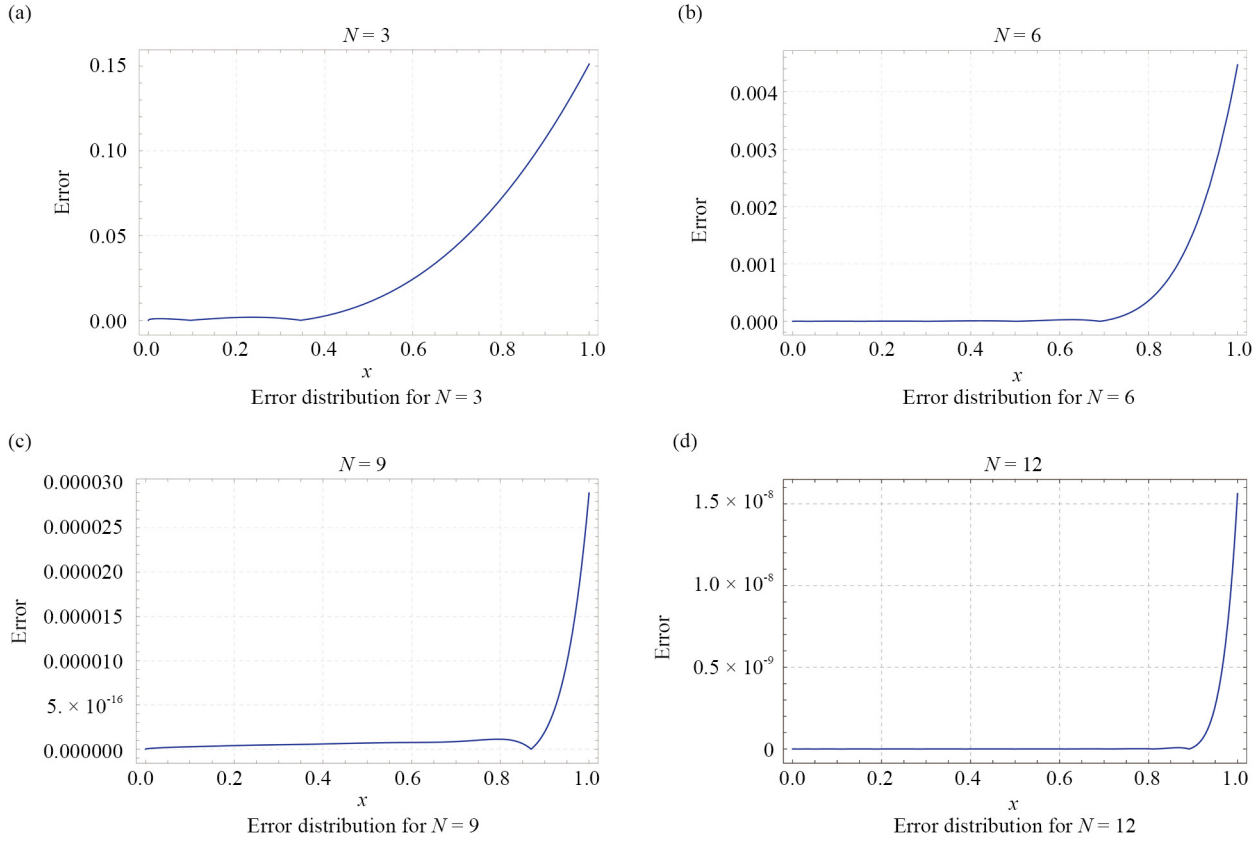


Figure 9. Example 3 Spatial error distributions for polynomial approximations of degrees $N \in \{3, 6, 9, 12\}$

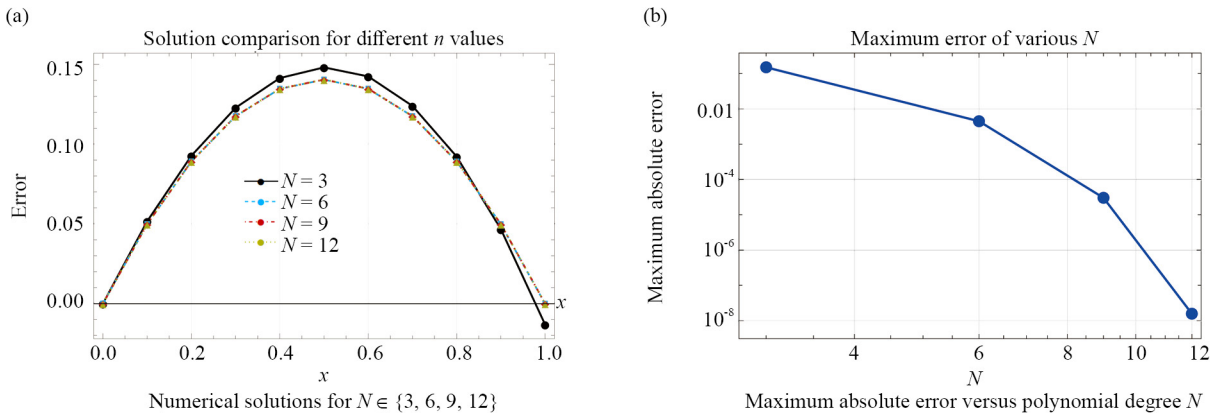


Figure 10. Example 3 spectral approximation performance for varying polynomial degrees

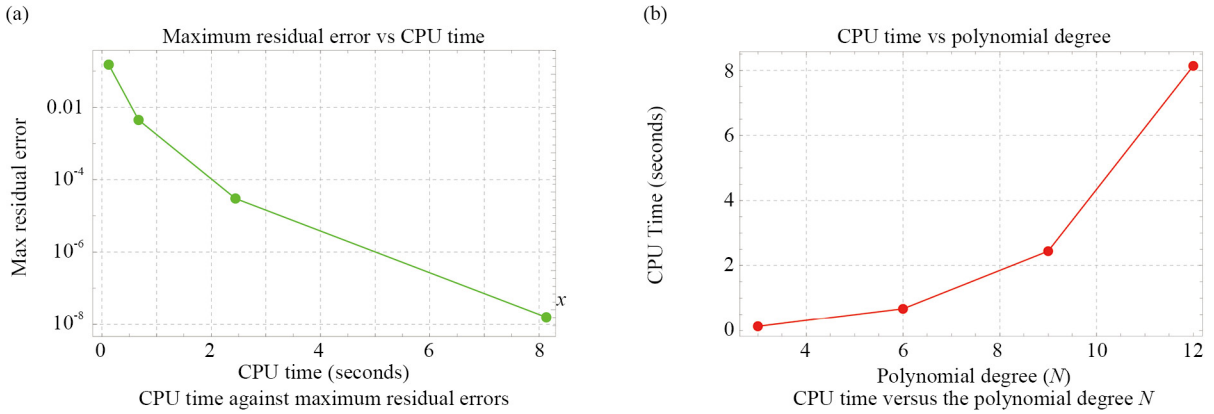


Figure 11. Example 3 computational performance for various polynomial degrees and corresponding maximum residual errors

Having presented the numerical experiments for various fractional orders and nonlinearity parameters, we now spot some of the computational aspects of our implementation. All numerical experiments were performed using Wolfram Mathematica 12.1 on a PC equipped with an Intel Core i3-4150 processor (3.50 GHz frequency) and 16.0 GB DDR3 RAM, running Windows 10 Home. CPU times were measured using the built-in timing functions, averaged over multiple runs to minimize system variability. The Newton-Raphson convergence order p was computed using the formula

$$p \approx \frac{\log(\|\mathbf{F}(\mathbf{d}^{(\eta)})\|/\|\mathbf{F}(\mathbf{d}^{(\eta-1)})\|)}{\log(\|\mathbf{F}(\mathbf{d}^{(\eta-1)})\|/\|\mathbf{F}(\mathbf{d}^{(\eta-2)})\|)},$$

where $\|F(\mathbf{d}^{(\eta)})\|$ represents the Euclidean norm of the residual vector at iteration η. The reported CPU times exhibit polynomial growth with respect to the degree N, consistent with the complexity of solving the $(N+1)\times(N+1)$ Newton systems. As demonstrated in the convergence plots, the method typically achieves linear convergence during initial iterations when distant from the solution, transitioning to quadratic convergence ($p \approx 2$) within the basin of attraction.

7. Concluding remarks

This study presented a collocation approach to treat the one-dimensional fractional Bratu differential equation governed by nonhomogeneous Dirichlet boundary conditions, using the shifted Lucas polynomials as basis functions. The method quickly converges and accurately deals with the fractional derivatives and nonlinear terms. The application of the algorithm led to reducing the equation with its underlying conditions into a nonlinear algebraic system efficiently handled by the Newton-Raphson method. A thorough error analysis for the proposed shifted Lucas expansion was presented. We presented some numerical tests and comparisons to ensure the algorithm's applicability and high convergence. The shifted Lucas polynomials may be employed to solve some other FDEs using the various versions of spectral methods. In addition, we aim to introduce some generalizations of the shifted Lucas polynomials in the near future and employ them in other applications.

Availability of data and material

The authors did not use any scientific data during this research.

Acknowledgments

The authors would like to express their sincere gratitude to the anonymous referees for their useful remarks, which helped us to enhance the paper in its present form.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Al Baghdadi SK, Ahammad NA. A comparative study of Adomian decomposition method with variational iteration method for solving linear and nonlinear differential equations. *Journal of Applied Mathematics and Physics*. 2024; 12(8): 2789-2819. Available from: https://doi.org/10.4236/jamp.2024.128166.

[2] Lakshmi R, Gireesha BJ, Venkatesh P, Gowtham KJ. Hermite wavelet approach to analyze the entropy generation of MHD Williamson hybrid nanofluid flow through an inclined channel with particle shape effects. *International Journal of Applied and Computational Mathematics*. 2025; 11(2): 40. Available from: https://doi.org/10.1007/s40819-025-01853-6.

[3] Gireesha BJ, Gowtham KJ. Efficient hypergeometric wavelet approach for solving Lane-Emden equations. *Journal of Computational Science*. 2024; 82: 102392. Available from: https://doi.org/10.1016/j.jocs.2024.102392.

[4] Gowtham KJ, Gireesha BJ. Associated Laguerre wavelets: Efficient method to solve linear and nonlinear singular initial and boundary value problems. *International Journal of Applied and Computational Mathematics*. 2025; 11(1): 16. Available from: https://doi.org/10.1007/s40819-024-01827-0.

[5] Gowtham KJ, Gireesha BJ. An application of novel vertex covering wavelet (of path graph) for solving singular boundary value problems. *International Journal of Dynamics and Control*. 2025; 13(6): 220. Available from: https://doi.org/10.1007/s40435-025-01732-4.

[6] Manvitha NV, Gireesha BJ, Gowtham KJ. Taylor wavelet solution procedure for the analysis of triangular porous fin made of functionally graded materials in the presence of hybrid nanofluid. *Acta Mechanica*. 2025; 236: 3055-3080. Available from: https://doi.org/10.1007/s00707-025-04312-x.

[7] Canuto C, Hussaini MY, Quarteroni A, Zang TA. *Spectral Methods: Fundamentals in Single Domains*. Heidelberg: Springer; 2006. Available from: https://doi.org/10.1007/978-3-540-30726-6.

[8] Boyd JP. *Chebyshev and Fourier Spectral Methods*. Heidelberg: Springer; 1989.
[9] Shen J, Tang T, Wang LL. *Spectral Methods: Algorithms, Analysis and Applications*. Heidelberg: Springer; 2011. Available from: https://doi.org/10.1007/978-3-540-71041-7.
[10] Li X, Xu C. A space-time spectral method for the time fractional diffusion equation. *SIAM Journal on Numerical Analysis*. 2010; 47(3): 2108-2131. Available from: https://doi.org/10.1137/080718942.
[11] Abd-Elhameed WM, Youssri YH, Amin AK, Atta AG. Eighth-kind Chebyshev polynomials collocation algorithm for the nonlinear time-fractional generalized Kawahara equation. *Fractal and Fractional*. 2023; 7(9): 652. Available from: https://doi.org/10.3390/fractalfract7090652.
[12] Manohara G, Kumbinarasaiah S. Numerical solution of some stiff systems arising in chemistry via Taylor wavelet collocation method. *Journal of Mathematical Chemistry*. 2024; 62(1): 24-61. Available from: https://doi.org/10.1007/s10910-023-01508-1.
[13] Youssri YH, Abd-Elhameed WM, Abdelhakem M. A robust spectral treatment of a class of initial value problems using modified Chebyshev polynomials. *Mathematical Methods in the Applied Sciences*. 2021; 44(11): 9224-9236. Available from: https://doi.org/10.1002/mma.7347.
[14] Yassin NM, Atta AG, Aly EH. Numerical solutions for nonlinear ordinary and fractional Newell-Whitehead-Segel equation using shifted Schröder polynomials. *Boundary Value Problems*. 2025; 2025(1): 57. Available from: https://doi.org/10.1186/s13661-025-02041-7.
[15] Abd-Elhameed WM, Youssri YH, Atta AG. Adopted spectral tau approach for the time-fractional diffusion equation via seventh-kind Chebyshev polynomials. *Boundary Value Problems*. 2024; 2024(1): 102. Available from: https://doi.org/10.1186/s13661-024-01907-6.
[16] Sadri K, Amilo D, Hosseini K, Hinçal E, Seadawy AR. A tau-Gegenbauer spectral approach for systems of fractional integro-differential equations with the error analysis. *AIMS Mathematics*. 2024; 9(2): 3850-3880. Available from: https://doi.org/10.3934/math.2024190.
[17] Abd-Elhameed WM, Machado JAT, Youssri YH. Hypergeometric fractional derivatives formula of shifted Chebyshev polynomials: Tau algorithm for a type of fractional delay differential equations. *International Journal of Nonlinear Sciences and Numerical Simulation*. 2022; 23(7-8): 1253-1268. Available from: https://doi.org/10.1515/ijnsns-2020-0124.
[18] Abd-Elhameed WM, Abu Sunayh AF, Alharbi MH, Atta AG. Spectral tau technique via Lucas polynomials for the time-fractional diffusion equation. *AIMS Mathematics*. 2024; 9: 34567-34587. Available from: https://doi.org/10.3934/math.20241646.
[19] Youssri YH, Hafez RM, Atta AG. An innovative pseudo-spectral Galerkin algorithm for the time-fractional Tricomitype equation. *Physica Scripta*. 2024; 99(10): 105238. Available from: https://doi.org/10.1088/1402-4896/ad74ad.
[20] Abd-Elhameed WM, Alsuyuti MM. New spectral algorithm for fractional delay pantograph equation using certain orthogonal generalized Chebyshev polynomials. *Communications in Nonlinear Science and Numerical Simulation*. 2025; 141: 108479. Available from: https://doi.org/10.1016/j.cnsns.2024.108479.
[21] Podlubny I. *Fractional Differential Equations*. USA: Academic Press; 1999.
[22] Sivalingam SM, Kumar P, Trinh H, Govindaraj V. A novel L1-predictor-corrector method for the numerical solution of the generalized-Caputo type fractional differential equations. *Mathematics and Computers in Simulation*. 2024; 220: 462-480. Available from: https://doi.org/10.1016/j.matcom.2024.01.017.
[23] Zhang X, Feng Y, Luo Z, Liu J. A spatial sixth-order numerical scheme for solving fractional partial differential equation. *Applied Mathematics Letters*. 2025; 159: 109265. Available from: https://doi.org/10.1016/j.aml.2024.109265.
[24] Zhang Y, Wang L. Application of Laplace Adomian decomposition method for fractional Fokker-Planck equation and time fractional coupled Boussinesq-Burger equations. *Engineering Computations*. 2024; 41(4): 793-818. Available from: https://doi.org/10.1108/EC-06-2023-0275.
[25] Kumar CD, Prakasha D, Veeresha P, Kapoor M. A homotopy-based computational scheme for two-dimensional fractional cable equation. *Modern Physics Letters B*. 2024; 38(32): 2450292. Available from: https://doi.org/10.1142/S0217984924502920.
[26] Farhood AK, Mohammed OH. Homotopy perturbation method for solving time-fractional nonlinear variable-order delay partial differential equations. *Partial Differential Equations in Applied Mathematics*. 2023; 7: 100513. Available from: https://doi.org/10.1016/j.padiff.2023.100513.

[27] Baihi A, Kajouni A, Hilal K, Lmou H. Laplace transform method for a coupled system of (p, q)-Caputo fractional differential equations. *Journal of Applied Mathematics and Computing*. 2025; 71(1): 511-530. Available from: https://doi.org/10.1007/s12190-024-02254-6.

[28] Abd-Elhameed WM, Youssri YH, Doha EH. A novel operational matrix method based on shifted Legendre polynomials for solving second-order boundary value problems involving singular, singularly perturbed and Bratu'stype equations. *Mathematical Sciences*. 2015; 9(2): 93-102. Available from: https://doi.org/10.1007/s40096-015-0155-8.

[29] Izadi M, Kamandar M. Fractional-order Liouville-Caputo electrical circuit models: QLM-Dickson matrix collocation algorithms. *Computers and Electrical Engineering*. 2025; 122: 109981. Available from: https://doi.org/10.1016/j.compeleceng.2024.109981.

[30] Huang R, Weng Z, Yuan J. Collocation-based numerical simulation of multi-dimensional nonlinear time-fractional Schrödinger equations. *Computers & Mathematics with Applications*. 2025; 183: 214-233. Available from: https://doi.org/10.1016/j.camwa.2025.02.002.

[31] Marasi HR, Derakhshan MH, Ghuraibawi AA, Kumar P. A novel method based on fractional order Gegenbauer wavelet operational matrix for the solutions of the multi-term time-fractional telegraph equation of distributed order. *Mathematics and Computers in Simulation*. 2024; 217: 405-424. Available from: https://doi.org/10.1016/j.matcom.2023.11.004.

[32] Youssri YH, Abd-Elhameed WM, Ahmed HM. New fractional derivative expression of the shifted third-kind Chebyshev polynomials: Application to a type of nonlinear fractional pantograph differential equations. *Journal of Function Spaces*. 2022; 2022(1): 3966135. Available from: https://doi.org/10.1155/2022/3966135.

[33] Abdelhakem M, Fawzy M, El-Kady M, Moussa H. Legendre polynomials' second derivative tau method for solving Lane-Emden and Ricatti equations. *Applied Mathematics and Information Sciences*. 2023; 17(3): 437-445. Available from: http://dx.doi.org/10.18576/amis/170305.

[34] Hafez RM, Zaky MA. High-order continuous Galerkin methods for multi-dimensional advection-reaction-diffusion problems. *Engineering with Computers*. 2020; 36(4): 1813-1829. Available from: https://doi.org/10.1007/s00366-019-00797-y.

[35] Hafez RM, Zaky MA, Abdelkawy MA. Jacobi spectral Galerkin method for distributed-order fractional Rayleigh-Stokes problem for a generalized second grade fluid. *Frontiers in Physics*. 2020; 7: 240. Available from: https://doi.org/10.3389/fphy.2019.00240.

[36] Jacobsen J, Schmitt K. The Liouville-Bratu-Gelfand problem for radial operators. *Journal of Differential Equations*. 2002; 184(1): 283-298. Available from: https://doi.org/10.1006/jdeq.2001.4151.

[37] Khuri SA, Louhichi I, Sayfy A. Fractional Bratu's problem: A novel approach. *International Journal of Applied and Computational Mathematics*. 2023; 9(3): 107. Available from: https://doi.org/10.1007/s40819-023-01509-3.

[38] Koshy T. *Fibonacci and Lucas Numbers with Applications*. USA: John Wiley and Sons; 2001.

[39] Karami S, Fakharzadeh Jahromi A, Heydari M. A computational method based on the generalized Lucas polynomials for fractional optimal control problems. *Advances in Continuous and Discrete Models*. 2022; 2022(1): 64. Available from: https://doi.org/10.1186/s13662-022-03737-1.

[40] Gumgum S, Savasaneril NB, Kurkcu O, Sezer M. Lucas polynomial solution for neutral differential equations with proportional delays. *TWMS Journal of Applied and Engineering Mathematics*. 2020; 10(1): 259-269.

[41] Mokhtar MM, Mohamed AS. Lucas polynomials semi-analytic solution for fractional multi-term initial value problems. *Advances in Difference Equations*. 2019; 2019(1): 471. Available from: https://doi.org/10.1186/s13662-019-2402-z.

[42] Salama M, Zedan H, Abd-Elhameed W, Youssri YH. Galerkin method with modified shifted Lucas polynomials for solving the 2D Poisson equation. *Journal of Computational Applied Mechanics*. 2025; 56(4): 737-776. Available from: https://doi.org/10.22059/jcamech.2025.398629.1550.

[43] Saadatmandi A, Dehghan M. A new operational matrix for solving fractional-order differential equations. *Computers & Mathematics with Applications*. 2010; 59(3): 1326-1336. Available from: https://doi.org/10.1016/j.camwa.2009.07.006.

[44] Bhrawy AH, Zaky MA. A review of operational matrices and spectral techniques for fractional calculus. *Nonlinear Dynamics*. 2015; 81(3): 1023-1052. Available from: https://doi.org/10.1007/s11071-015-2087-0.

[45] Atta AG, Abd-Elhameed WM, Moatimid GM, Youssri YH. Advanced shifted sixth-kind Chebyshev tau approach for solving linear one-dimensional hyperbolic telegraph type problem. *Mathematical Sciences*. 2023; 17(4): 415-429. Available from: https://doi.org/10.1007/s40096-022-00460-6.

[46] Ahmed HM. Enhanced shifted Jacobi operational matrices of integrals: Spectral algorithm for solving some types of ordinary and fractional differential equations. *Boundary Value Problems*. 2024; 2024(1): 75. Available from: https://doi.org/10.1186/s13661-024-01880-0.

[47] Mokhtar MM, Mohamed AS. Lucas polynomials semi-analytic solution for fractional multi-term initial value problems. *Advances in Continuous and Discrete Models*. 2019; 2019(1): 471. Available from: https://doi.org/10.1186/s13662-019-2402-z.

[48] Podlubny I. *Fractional Differential Equations: An Introduction to Fractional Derivatives, Fractional Differential Equations, to Methods of Their Solution and Some of Their Applications*. Amsterdam, Netherlands: Elsevier; 1998.

[49] Abd-Elhameed WM, Youssri YH. Generalized Lucas polynomial sequence approach for fractional differential equations. *Nonlinear Dynamics*. 2017; 89(2): 1341-1355. Available from: https://doi.org/10.1007/s11071-017-3519-9.

[50] Sahlan MN, Afshari H. Lucas polynomials based spectral methods for solving the fractional order electrohydrodynamics flow model. *Communications in Nonlinear Science and Numerical Simulation*. 2022; 107: 106108. Available from: https://doi.org/10.1016/j.cnsns.2021.106108.

[51] Koundal R, Kumar R, Kumar R, Srivastava K, Baleanu D. A novel collocated-shifted Lucas polynomial approach for fractional integro-differential equations. *International Journal of Applied and Computational Mathematics*. 2021; 7(4): 167. Available from: https://doi.org/10.1007/s40819-021-01108-0.

[52] Burden RL, Faires JD. *Numerical Analysis*. 9th ed. Boston: Cengage Learning; 2010.

[53] Abd-Elhameed WM, Youssri YH. Spectral solutions for fractional differential equations via a novel Lucas operational matrix of fractional derivatives. *Romanian Journal of Physics*. 2016; 61(5-6): 795-813.

[54] Sakar MG, Saldır O, Akgül A. Numerical solution of fractional Bratu type equations with Legendre reproducing kernel method. *International Journal of Applied and Computational Mathematics*. 2018; 4(5): 126. Available from: https://doi.org/10.1007/s40819-018-0562-2.

[55] Ahmad A, Sulaiman M, Aljohani AJ, Alhindi A, Alrabaiah H. Design of an efficient algorithm for solution of Bratu differential equations. *Ain Shams Engineering Journal*. 2021; 12(2): 2211-2225. Available from: https://doi.org/10.1016/j.asej.2020.11.007.

[56] He J, Cao C. A new neural network method for solving Bratu type equations with rational polynomials. *International Journal of Machine Learning and Cybernetics*. 2025; 16(2): 1355-1369. Available from: https://doi.org/10.1007/s13042-024-02340-y.

[57] Tomar S, Dhama S, Ramos H, Singh M. An efficient technique based on Green's function for solving two-point boundary value problems and its convergence analysis. *Mathematics and Computers in Simulation*. 2023; 210: 408-423. Available from: https://doi.org/10.1016/j.matcom.2023.03.015.

[58] Umesh U. Numerical simulation of Bratu's problem using a new form of the Adomian decomposition technique. *International Journal of Numerical Methods for Heat and Fluid Flow*. 2023; 33(6): 2295-2307. Available from: https://doi.org/10.1108/HFF-11-2022-0656.

[59] Swaminathan G, Hariharan G, Selvaganesan V, Bharatwaja S. A new spectral collocation method for solving Bratutype equations using Genocchi polynomials. *Journal of Mathematical Chemistry*. 2021; 59: 1837-1850. Available from: https://doi.org/10.1007/s10910-021-01264-0.

[60] Keshavarz E, Ordokhani Y, Razzaghi M. The Taylor wavelets method for solving the initial and boundary value problems of Bratu-type equations. *Applied Numerical Mathematics*. 2018; 128: 205-216. Available from: https://doi.org/10.1016/j.apnum.2018.02.001.

[61] Quarteroni A, Sacco R, Saleri F. *Numerical Mathematics*. Vol. 37. Heidelberg: Springer; 2006. Available from: https://doi.org/10.1007/b98885.

[62] Higham NJ. *Accuracy and Stability of Numerical Algorithms*. 2nd ed. USA: Society for Industrial and Applied Mathematics; 2002.

[63] Press WH. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. UK: Cambridge University Press; 2007.

[64] Khuri S, Louhichi I, Sayfy A. Fractional Bratu's problem: A novel approach. *International Journal of Applied and Computational Mathematics*. 2023; 9(5): 107. Available from: https://doi.org/10.1007/s40819-023-01509-3.