Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. *, No. *, *, pp. 1-21
DOI:10.22034/cmde.2025.66899.3165

Enhancing numerical solutions of nonlinear fractional boundary value problems through Genocchi Quasi linearization method

Nesreen Mohamed[1], **Mohamed El-Gamel**[3], **Waleed Adel**[2,3,*]

[1]Department of Basic Science, Faculty of Engineering, Horus University, New Damietta 34517, Egypt.

[2]Nanoelectronics Integrated Systems Center, Nile University, Giza, 12588, Egypt.

[3]Department of Mathematics and Engineering Physics, Faculty of Engineering, Mansoura University, Mansoura 35516, Egypt.

Abstract

This paper presents an enhanced numerical method for solving nonlinear fractional boundary value problems using the Genocchi quasilinearization method (G-QLM). Fractional differential equations (FDEs) are gaining significant attention due to their ability to model various real-world phenomena with memory and hereditary properties in fields such as mechanics, fluid dynamics, chemistry, and astrophysics. However, these equations often lack analytical solutions, necessitating the development of efficient numerical techniques. Traditional methods such as the spectral collocation method, finite difference, and various wavelet approaches have demonstrated limitations in accuracy and computational efficiency when applied to highly nonlinear problems. The proposed approach combines the quasilinearization technique with Genocchi polynomials to transform the nonlinear fractional models into a sequence of linearized subproblems. This iterative process involves solving the Bratu, Troesch, and Lane-Emden equations—three well-known mathematical models with applications in areas such as heat transfer, plasma confinement, and stellar structure. The paper details the formulation of these equations, the application of the Genocchi collocation method, and the generation of approximate solutions through iterative refinement. The accuracy and convergence of the G-QLM method are validated through error analysis in the weighted L^2-norm. Numerical experiments using MATLAB are conducted for various cases of fractional orders β and different initial and boundary conditions. Comparative studies demonstrate that the G-QLM method outperforms existing methods, including Bessel-based techniques, reproducing kernel methods, and Taylor wavelets, in terms of reduced absolute errors and computational efficiency.

Keywords. Nonlinear Fractional Boundary Value Problems; Genocchi Quasilinearization Method (G-QLM); Genocchi Polynomials; Fractional Differential Equations (FDEs); Numerical Methods.

1991 Mathematics Subject Classification. 26A33, 34A08, 65L10, 41A10, 65N35.

1. Introduction

Differential equations play a crucial role in many scientific domains [56]. In mechanics and fluid dynamics, they are employed to predict motion [6], in chemistry to describe chemical reactions, in ecology to explain ecological balance, in finance to model market dynamics, and in epidemiology to study the spread of diseases. However, solving nonlinear differential equations (DEs) often poses significant challenges, requiring advanced numerical methods [13]. Issues such as optimal control problems, stiff and numerically unstable systems, high degrees of nonlinearity, and the "curse of dimensionality" in multi-variable DEs [25] further complicate analytical solutions. Recently, the field of fractional differential equations (FDEs) has seen rapid growth due to its applicability across diverse scientific areas.

Fractional calculus has gained popularity for its utility in various disciplines, including mathematics [22], engineering [47], chemistry [24], physics [15], biology, psychology [43], and economics. Fractional differential equations are particularly effective in capturing memory and hereditary properties in materials and processes, which are often neglected in traditional integer-order models [38]. Since analytical solutions for FDEs are typically infeasible, a variety

Received: 21 April 2025; Accepted: 14 April 2026.
∗ Corresponding author. Email: wauof@nu.edu.eg.

of numerical methods have been developed. These include Cubic B-Splines method [40], Genocchi collocation method [18, 45], the fractional Sumudu homotopy perturbation technique (SHP) [5], finite difference method [59], differential quadrature method (DQM) [46], spectral collocation method [35], multistep Laplace optimized decomposition method [42], extended direct algebraic method (EDAM) [12], collocation discrete least squares meshless method [33], residual power series method (RPSM) [55], and variational iteration method [26].

This paper focuses on solving the general form of a fractional nonlinear boundary value problem:

$$ {}^{LC}D_t^{\beta}\, u(t) = \phi(t, u(t), u'(t)), \quad 1 < \beta \le 2, \quad 0 \le t \le 1, \tag{1.1} $$

where ϕ is a function that includes nonlinear terms and ${}^{LC}D_t^{\beta}$ denotes the Liouville–Caputo fractional derivative. Depending on the application, this equation can take one of the following forms:

- **Bratu equation**: If $\phi(t, u(t), u'(t)) = -\lambda\, e^{\eta\, u(t)}$, then

  $$ {}^{LC}D_t^{\beta}\, u(t) = -\lambda\, e^{\eta\, u(t)}, \tag{1.2} $$

  with one of the initial or boundary conditions:

  $$ u(0) = u_0, \quad u'(0) = u_1, \quad \text{or} \quad u(0) = u_0, \quad u(1) = u_1, \tag{1.3} $$

  where λ is a real constant and $\eta = \pm 1$.
- **Troesch equation**: If $\phi(t, u(t), u'(t)) = \lambda \sinh(\lambda\, u(t))$, then

  $$ {}^{LC}D_t^{\beta}\, u(t) = \lambda \sinh(\lambda\, u(t)), \tag{1.4} $$

  with the boundary conditions:

  $$ u(0) = u_0, \quad u(1) = u_1, \tag{1.5} $$

  where λ is a real number.
- **Lane-Emden equation**: If $\phi(t, u(t), u'(t)) = -\frac{d}{t^{\beta-1}}\, u'(t) - e^{\eta\, u(t)}$, then

  $$ {}^{LC}D_t^{\beta}\, u(t) = -\frac{d}{t^{\beta-1}}\, u'(t) - e^{\eta\, u(t)}, \tag{1.6} $$

  with one of the initial or boundary conditions:

  $$ u(0) = u_0, \quad u'(0) = u_1, \quad \text{or} \quad u'(0) = u_0, \quad u(1) = u_1, \tag{1.7} $$

  where d is a positive number, $\eta = \pm 1$, and u_0, u_1 belong to $\mathbb{R}$.

The Bratu equation's exponential component contributes to its high nonlinearity. This equation has applications in fields like chemistry (e.g., heat reactions), nanotechnology [11], and cosmology [44, 51]. Analytical methods for Bratu problems, such as series summation and perturbation techniques, are used in symmetric geometries like slabs and cylindrical pipes [32]. Numerical methods, including Lucas–Fibonacci series method [20], spectral collocation [8], Legendre spectral method [57], Laplace decomposition [4], fractional residual power series method [36], Chebyshev polynomials [54], and Bessel quasilinearization [30], have also been explored.

Similarly, the Troesch boundary value problem (TBVP) has applications in gas porous electrodes [23] and plasma column confinement [60]. Methods such as cubic trigonometric B-splines [21], Green function-based iterative schemes [1], Taylor wavelets [48], and Bessel quasilinearization [31] are used to solve Troesch's problem, helping engineers optimize parameters for energy confinement and stability.

Finally, the Lane-Emden equation (LEE) arises in models related to stellar structure and celestial mechanics [14]. It describes gaseous spheres in hydrostatic equilibrium. Lane and Emden introduced the classical LEE to model thermal evolution under molecular attraction and surface mass density [52]. Various techniques, including Legendre-QLM method [28], spectral collocation [61], Bessel quasilinearization [30], Laplace-residual methods [53], trigonometric B-spline collocation [3], Haar wavelets [58], Bernoulli wavelets [39], Genocchi wavelet [50] and Mittag-Leffler functions [9], have been applied to solve both classical and fractional cases of LEE.

In this paper, we solve the general form of Eq. (1.1) with three models (1.2), (1.4), and (1.6) by combining the quasilinearization technique with Genocchi polynomials in a method known as the Genocchi quasilinearization method (G-QLM). Using an iteration parameter r, which will be set at most 5, we first apply the quasilinearization technique

to transform it into a sequence of linearized subproblems. In order to solve the resulting sequences of linear equations, we used the Genocchi collocation approach. The contributions and novelty of the study are summarized as follows:

(1) The paper introduces a novel combination of Genocchi polynomials and the quasilinearization technique to improve numerical solutions for nonlinear fractional boundary value problems.
(2) Unlike many existing approaches that focus on a single model, this method is applied to three different types of nonlinear equations—Bratu, Troesch, and Lane-Emden equations—demonstrating its versatility across multiple scientific and engineering problems.
(3) The proposed method achieves a significant reduction in absolute errors compared to traditional methods like the Bessel-based quasilinearization, reproducing kernel, and Taylor wavelet methods, as shown through detailed numerical evaluations.
(4) The paper presents a rigorous error analysis, providing theoretical bounds for the solution error in the weighted L^2-norm. This adds robustness to the validation of the proposed approach.
(5) The G-QLM method delivers accurate results with a smaller number of collocation points and iterations, reducing computational complexity and runtime while maintaining high solution precision.

The organization of this paper is as follows: In section 2, some fundamental facts about fractional calculus and review of Genocchi polynomials is given. Section 3 discusses the quasilinearization method for the general form of the problem with the three models. Section 4 provides the Genocchi collocation method based on the quasilinearization methodology. The error analysis and the residual error of Genocchi quasilinearization method are presented in section 5. Section 6 examines numerical test examples that illustrate the relevant features of the method that is being presented. Finally, a conclusion to the paper is presented in section 7.

2. Fundamentals and definitions

In the following subsection, we will highlight the key theorems and definitions needed for the subsequent sections on fractional calculus and Genocchi polynomials.

2.1. An overview of fractional calculus.

The paper requires an understanding of fractional integrals and derivatives. The Liouville-Caputo derivative and Riemann-Liouville integral will be demonstrated next.

Definition 2.1. Assume that we have the real valued function $f(t)$, $t > 0$. One says that $f(t) \in C_\omega$, $\omega \in \mathbb{R}$ if there exists a real number $\psi > \omega$ as well as a function $\nu(t) \in C([0, \infty))$ in such a way that $f(t) = t^\psi \nu(t)$. If there exists an $m \in \mathbb{N}$ so that $f^{(m)}(t) \in C_\omega$, then we write $f(t) \in C_\omega^m$.

Definition 2.2. Let the function $f(t) \in C_\omega$ for $\omega > -1$ be given. The notion of Riemann-Liouville (RL) integral of order $\beta > 0$ is given by

$$ {}_0\mathcal{I}_t^\beta f(t) = \frac{1}{\Gamma(\beta)} \int_0^t f(z)\,(t-z)^{\beta-1}\,dz. $$

Definition 2.3. Suppose that $f(t) \in C_{-1}^m$ and $\beta \in (m-1, m)$, where $m \in \mathbb{N}$. The Liouville-Caputo (LC) derivative of order β for $f(t)$ is defined as follows:

$$ {}^{LC}D_t^\beta\, f(t) = {}_0\mathcal{I}_t^{m-\beta}\, D^m f(t) = \frac{1}{\Gamma(m-\beta)} \int_0^t \frac{f^{(m)}(z)\,dz}{(t-z)^{1+\beta-m}}, \quad t > 0, $$

where $D^m = \frac{d^m}{dt^m}$.

The operator ${}^{LC}D_t^\beta$ is a linear operator. Additionally, a constant function $f(t) = c$ always has a LC derivative of zero:

$$ {}^{LC}D_t^\beta\, c = 0. \tag{2.1} $$

The LC fractional derivative of $f(t) = t^\mu$ is calculated by the following rule:

$$ {}^{LC}D_t^\beta\, t^\mu = \begin{cases} \dfrac{\Gamma(\mu+1)}{\Gamma(\mu+1-\beta)} t^{\mu-\beta}, & \text{for } \mu \in \mathbb{N}_0 \text{ and } \mu \geq \lceil \beta \rceil \text{ or } \mu \notin \mathbb{N} \text{ and } \mu > \lfloor \beta \rfloor, \\ 0, & \text{for } \mu \in \mathbb{N}_0 \text{ and } \mu < \lceil \beta \rceil. \end{cases} \tag{2.2} $$

In Eq. (2.2), the symbols $\lceil\cdot\rceil$ and $\lfloor\cdot\rfloor$ are utilized to indicate the ceil and floor functions, respectively. Moreover, let us define the symbol $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$.

2.2. **Basic properties of Genocchi polynomials.** The Genocchi polynomials and the Genocchi numbers can be defined by the generating functions [7, 17, 19, 27, 49]

$$\frac{2\tau e^{\tau t}}{e^{\tau}+1} = \sum_{n=0}^{\infty} G_n(t)\frac{\tau^n}{n!}, \quad (|\tau| < \pi), \tag{2.3}$$

$$\frac{2\tau}{e^{\tau}+1} = \sum_{n=0}^{\infty} G_n\frac{\tau^n}{n!}, \quad (|\tau| < \pi). \tag{2.4}$$

Genocchi polynomials $G_n(t)$ of degree n are defined on the interval $[0, 1]$ as:

$$G_n(t) = \sum_{k=0}^{n} \binom{n}{k} G_k t^{n-k}, \tag{2.5}$$

where G_k is the Genocchi numbers which related to Bernoulli numbers as:

$$G_k = 2(1-2^k)B_k, \quad k \geq 1. \tag{2.6}$$

The Genocchi numbers are obtained by evaluating the Genocchi polynomials at $t = 0$ as follows:

$$G_1 = 1, \quad G_2 = -1, \quad G_3 = 0, \quad G_4 = 1, \quad G_5 = 0,$$
$$G_6 = -3, \quad G_7 = 0, \quad G_8 = 17, \quad G_9 = 0, \quad G_{10} = -155.$$

The first few Genocchi polynomials are as follows:

$$G_1(t) = 1, \qquad G_2(t) = 2t - 1,$$
$$G_3(t) = 3t^2 - 3t, \qquad G_4(t) = 4t^3 - 6t^2 + 1,$$
$$G_5(t) = 5t^4 - 10t^3 + 5t, \qquad G_6(t) = 6t^5 - 15t^4 + 15t^2 - 3.$$

Genocchi polynomials have many interesting properties, and some of them are presented in the following:

$$\frac{d}{dt}G_n(t) = n\,G_{n-1}(t), \quad n \geq 1, \tag{2.7}$$

$$G_n(t+1) + G_n(t) = 2n\,t^{n-1}, \quad n \geq 1, \tag{2.8}$$

$$G_n(1) + G_n(0) = 0, \quad n > 1. \tag{2.9}$$

The change of variable t to t^{α} in Eq. (2.5) can be used to define the fractional order Genocchi functions. These polynomials will be represented by the notation $G_{n,\alpha}(t)$. When these polynomials are extended to the interval $[0, 1]$, we get deteriorates

$$G_{n,\alpha}(t) = \sum_{k=0}^{n} \binom{n}{k} G_k t^{n\alpha - k\alpha}, \qquad \alpha > 0. \tag{2.10}$$

3. Quasilinearization technique

In this section, we will explain the core concept and outline the steps of the proposed quasilinearization technique. The original nonlinear model problem (1.1) is intended to be converted into a series of linear equations using the quasilinearization technique. The resulting linearized models will next be subjected to the Genocchi collocation method. In this respect, the fundamental concepts that underlying the quasilinearization method (QLM) solutions of our model problems are briefly explained; for further in-depth discussions and applications, see [2, 29].

Let's examine the nonlinear form (1.1), which represents the standard formulation of our model problems. To solve Eqs. (1.2), (1.4), and (1.6), we must first select an initial approximation $u_0(t)$ for the function $u(t)$. Then, the QLM iteration for Eq. (1.1) is defined as follows:

$$\begin{aligned} {}^{LC}D_t^{\beta} u_{r+1}(t) =& \phi(t, u_r(t), u_r'(t)) + (u_{r+1}(t) - u_r(t))\phi_u(t, u_r(t), u_r'(t)) \\ &+ (u_{r+1}'(t) - u_r'(t))\phi_{u'}(t, u_r(t), u_r'(t)), \quad r = 0, 1, \ldots, 5, \end{aligned} \tag{3.1}$$

with the initial or boundary conditions as mentioned in Eqs. (1.3), (1.5), and (1.7). The functional derivatives of $\phi(t, u(t), u'(t))$ are demonstrated by the functions $\phi_u = \partial\phi/\partial u$ and $\phi_{u'} = \partial\phi/\partial u'$. The following linearized model form is produced after the QLM approach is applied to the nonlinear model problem (1.2) as:

$${}^{LC}D_t^{\beta} u_{r+1}(t) + \lambda\,\eta\, e^{\eta\, u_r(t)} u_r(t)\, u_{r+1}(t) = \lambda\, e^{\eta\, u_r(t)} u_r(t)\,(\eta\, u_r(t) - 1), \tag{3.2}$$

where $1 < \beta \leq 2$ and $\eta = \pm 1$. With one of the initial or boundary conditions

$$u_{r+1}(0) = u_0, \quad u_{r+1}'(0) = u_1, \quad \text{or} \quad u_{r+1}(0) = u_0, \quad u_{r+1}(1) = u_1. \tag{3.3}$$

For the nonlinear model (1.4), we have

$${}^{LC}D_t^{\beta} u_{r+1}(t) - \left[\lambda^2 \cosh(\lambda\, u_r(t))\right] u_{r+1}(t) = -\lambda^2\, u_r(t) \cosh(\lambda\, u_r(t)) + \lambda \sinh(\lambda\, u_r(t)), \tag{3.4}$$

with the boundary conditions

$$u_{r+1}(0) = u_0, \quad u_{r+1}(1) = u_1. \tag{3.5}$$

In the same way, by taking $\eta = \pm 1$ for the model problem (1.6), we get

$${}^{LC}D_t^{\beta} u_{r+1}(t) + \frac{d}{t^{\beta-1}}\, u_r'(t) + \eta\, e^{\eta\, u_r(t)} u_r(t)\, u_{r+1}(t) = e^{\eta\, u_r(t)} u_r(t)\,(\eta\, u_r(t) - 1), \tag{3.6}$$

with one of the initial or boundary conditions

$$u_{r+1}(0) = u_0, \quad u_{r+1}'(0) = u_1, \quad \text{or} \quad u_{r+1}'(0) = u_0, \quad u_{r+1}(1) = u_1. \tag{3.7}$$

Therefore, instead of using the Genocchi collocation method to solve the original Eqs. (1.2), (1.4), and (1.6), we use it to solve the quasilinear model Eqs. (3.2), (3.4), and (3.6) instead. The latter method is known as the G-QLM.

4. Genocchi Quasilinearization Method

In this section, we seek to solve the model problems (1.2), (1.4), and (1.6) approximately so that the desired solutions are represented in terms of the truncated Genocchi series form for the associated approximated quasilinear model problems (3.2), (3.4), and (3.6). Assume for the purposes of this analysis that the model problems in the iteration $r = 0, 1, \ldots, 5$ have an approximated solution $u_N^{(r+1)}(t)$. The next iteration can be approximated using the following form:

$$u_{r+1}(t) \approx u_{N,\alpha}^{(r+1)}(t) = \sum_{n=1}^{N+1} c_n^{(r)} G_{n,\alpha}(t), \quad 0 \leq t \leq 1, \tag{4.1}$$

where the unknown coefficients $c_n^{(r)}$, $n = 1, 2, \ldots, N+1$ have to be sought. By introducing $\mathbf{C}^{(r)} = [c_1^{(r)}\ c_2^{(r)}\ \cdots\ c_{N+1}^{(r)}]^T$, the vector of generalized Genocchi polynomials is given as follows:

$$\mathbf{G}_{\alpha}(t) = \begin{bmatrix} G_{1,\alpha}(t) & G_{2,\alpha}(t) & \ldots & G_{N+1,\alpha}(t) \end{bmatrix}, \qquad \alpha > 0.$$

Next, we can express the approximate solution $u_{N,\alpha}^{(r+1)}(t)$ in Eq. (4.1) in matrix form as follows:

$$u_{N,\alpha}^{(r+1)}(t) = \mathbf{G}_{\alpha}(t)\mathbf{C}^{(r)}. \tag{4.2}$$

Then, we decompose the vector of Genocchi polynomials in the following form:

$$\mathbf{G}_{\alpha}(t) = \mathbf{\Pi}_{\alpha}(t)\mathbf{H}, \tag{4.3}$$

where the vector of standard bases, $\mathbf{\Pi}_\alpha(\mathbf{t})$, is defined by

$$\mathbf{\Pi}_\alpha(t) = \begin{bmatrix} 1 & t^\alpha & t^{2\alpha} & \dots & t^{N\alpha} \end{bmatrix}, \qquad \alpha > 0. \tag{4.4}$$

Additionally, the matrix $\mathbf{H}$ is an upper triangular matrix. Its elements are given as follows:

$$\mathbf{H} = \begin{pmatrix} \binom{1}{1}G_1 & \binom{2}{2}G_2 & \binom{3}{3}G_3 & \dots & \binom{N}{N}G_N & \binom{N+1}{N+1}G_{N+1} \\ 0 & \binom{2}{1}G_1 & \binom{3}{2}G_2 & \dots & \binom{N}{N-1}G_{N-1} & \binom{N+1}{N}G_N \\ 0 & 0 & \binom{3}{1}G_1 & \dots & \binom{N}{N-2}G_{N-2} & \binom{N+1}{N-1}G_{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \binom{N}{1}G_1 & \binom{N+1}{2}G_2 \\ 0 & 0 & 0 & \dots & 0 & \binom{N+1}{1}G_1 \end{pmatrix}_{(N+1)\times(N+1)}. \tag{4.5}$$

Corollary 4.1. *The approximate solution $u_{N,\alpha}^{(r+1)}(t)$ in Eq. (4.1) can be written as follows after combining the two previous relations in Eqs. (4.2) and (4.3)*

$$u_{N,\alpha}^{(r+1)}(t) = \mathbf{\Pi}_\alpha(t)\mathbf{H}\mathbf{C}^{(r)}. \tag{4.6}$$

For determining the fractional derivative of order β of $u_{N,\alpha}^{(r+1)}$, we need to calculate the fractional derivative of the basis function $\mathbf{\Pi}_\alpha(t)$ using Algorithm **??**, which based on the relations in Eqs. (2.1), and (2.2). Thus, we obtain

Algorithm 1 The calculation of the β-derivative of $\mathbf{\Pi}_\alpha(t)$.

procedure $[\mathbf{\Pi}_\alpha^{(\beta)}]$ = compute_DX(N, β, α)
  $\mathbf{\Pi}_\alpha^{(\beta)}[1] := 0;$
for $k := 1, \dots, N$ **do**
if $(k\alpha - \beta < 0)$ **then**
$\mathbf{\Pi}_\alpha^{(\beta)}[k+1] := 0;$
else
if $((k\alpha < \beta)$ && $(k\alpha - \lfloor k\alpha \rfloor == 0))$ **then**
$\mathbf{\Pi}_\alpha^{(\beta)}[k+1] := 0;$
else
$\mathbf{\Pi}_\alpha^{(\beta)}[k+1] := \frac{\Gamma(k\alpha+1)}{\Gamma(k\alpha+1-\beta)} t^{k\alpha-\beta};$
end if
end if
end for
end;

$$^{LC}D_t^\beta u_{N,\alpha}^{(r+1)}(t) = \mathbf{\Pi}_\alpha^{(\beta)}(t)\mathbf{H}\mathbf{C}^{(r)}. \tag{4.7}$$

If $\beta = 1$, we have the first integer order of derivative as

$$\frac{d}{dt}u_{N,\alpha}^{(r+1)}(t) = \mathbf{\Pi}_\alpha^{(1)}(t)\mathbf{H}\mathbf{C}^{(r)}. \tag{4.8}$$

Using collocation points defined as

$$t_k = 0.5\left[1 - \cos\left(\frac{\pi\,(2k+1)}{2\,(N+1)}\right)\right], \qquad k = 0, 1, ..., N, \tag{4.9}$$

in Eqs. (4.6), (4.7), and (4.8), we obtain

$$\mathbf{U}_{r+1} = \mathbf{\Pi}\,\mathbf{H}\mathbf{C}^{(r)} = \begin{pmatrix} u_{N,\alpha}^{(r+1)}(t_0) \\ u_{N,\alpha}^{(r+1)}(t_1) \\ \vdots \\ u_{N,\alpha}^{(r+1)}(t_N) \end{pmatrix}, \quad \mathbf{\Pi} = \begin{pmatrix} \Pi_\alpha(t_0) \\ \Pi_\alpha(t_1) \\ \vdots \\ \Pi_\alpha(t_N) \end{pmatrix}, \tag{4.10}$$

$$\mathbf{U}_{r+1}^{(\beta)} = \mathbf{\Pi}^{(\beta)}\mathbf{H}\mathbf{C}^{(r)} = \begin{pmatrix} D^{(\beta)}u_{N,\alpha}^{(r+1)}(t_0) \\ D^{(\beta)}u_{N,\alpha}^{(r+1)}(t_1) \\ \vdots \\ D^{(\beta)}u_{N,\alpha}^{(r+1)}(t_N) \end{pmatrix}, \quad \mathbf{\Pi}^{(\beta)} = \begin{pmatrix} \Pi_\alpha^{(\beta)}(t_0) \\ \Pi_\alpha^{(\beta)}(t_1) \\ \vdots \\ \Pi_\alpha^{(\beta)}(t_N) \end{pmatrix}, \tag{4.11}$$

$$\dot{\mathbf{U}}_{r+1} = \mathbf{\Pi}^{(1)}\mathbf{H}\mathbf{C}^{(r)} = \begin{pmatrix} D^{(1)}u_{N,\alpha}^{(r+1)}(t_0) \\ D^{(1)}u_{N,\alpha}^{(r+1)}(t_1) \\ \vdots \\ D^{(1)}u_{N,\alpha}^{(r+1)}(t_N) \end{pmatrix}, \quad \mathbf{\Pi}^{(1)} = \begin{pmatrix} \Pi_\alpha^{(1)}(t_0) \\ \Pi_\alpha^{(1)}(t_1) \\ \vdots \\ \Pi_\alpha^{(1)}(t_N) \end{pmatrix}. \tag{4.12}$$

Now, by using the Genocchi collocation technique, we construct an approximate solution in the form (4.1) for each of the quasilinear model problems (3.2), (3.4), and (3.6) in this way.
Next, we will provide detailed explanations on how to address each of the proposed problems separately using the quasilinearization technique.

4.1. **Quasilinear Bratu equation.** Initially, we should focus on the Eq. (3.2). This equation can be collocated at the collocation points defined in Eq. (4.9) to demonstrate the following equation as

$${}^{LC}D_t^{\beta}u_{r+1}(t_k) + \lambda\,\eta\,e^{\eta\,u_r(t_k)}u_{r+1}(t_k) = \lambda\,e^{\eta\,u_r(t_k)}(\eta\,u_r(t_k) - 1), \quad k = 0, 1, \ldots, N, \tag{4.13}$$

where $\eta = \pm 1$. We can write the above equation as compactly as possible by using the matrix notations specified in Eqs. (4.10) and (4.11) as:

$$\mathbf{U}_{r+1}^{(\beta)} + \mathbf{A}_r\mathbf{U}_{r+1} = \mathbf{\Omega}_r, \tag{4.14}$$

where the right-hand side vector $\mathbf{\Omega}_r$ and the coefficient matrix $\mathbf{A}_r$ of size $(N+1)\times(N+1)$ have the following representations as:

$$\mathbf{A}_r = \begin{pmatrix} \lambda\,\eta\,e^{\eta\,u_r(t_0)} & 0 & \cdots & 0 \\ 0 & \lambda\,\eta\,e^{\eta\,u_r(t_1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda\,\eta\,e^{\eta\,u_r(t_N)} \end{pmatrix}, \quad \mathbf{\Omega}_r = \begin{pmatrix} \lambda\,e^{\eta\,u_r(t_0)}(\eta\,u_r(t_0) - 1) \\ \lambda\,e^{\eta\,u_r(t_1)}(\eta\,u_r(t_1) - 1) \\ \vdots \\ \lambda\,e^{\eta\,u_r(t_N)}(\eta\,u_r(t_N) - 1) \end{pmatrix}_{(N+1)\times 1}.$$

This provides us the fundamental matrix equation

$$\mathbf{\Psi}_r\mathbf{C}^{(r)} = \mathbf{\Omega}_r, \quad r = 0, 1, \ldots, 5, \tag{4.15}$$

where

$$\mathbf{\Psi}_r = (\mathbf{\Pi}^{(\beta)} + \mathbf{A}_r)\mathbf{H}. \tag{4.16}$$

We must first define an initial approximation $u_0(t)$ for the solution before we are permitted to begin the calculation. Additionally, the initial or boundary conditions that have been satisfied in Eq. (3.3) must be considered in each iteration. This is achieved by replacing the first two rows of the augmented matrix $[\mathbf{\Psi}_r; \mathbf{\Omega}_r]$ by the row matrices

$[\hat{\mathbf{\Psi}}_0; u_0]$ and $[\hat{\mathbf{\Psi}}_1; u_1]$ which can be formulated as follows:
For the initial conditions:

$$[\hat{\mathbf{\Psi}}_0; u_0] = [\mathbf{\Pi}_\alpha(0)\mathbf{H}\mathbf{C}^{(r)}; u_0], \qquad [\hat{\mathbf{\Psi}}_1; u_1] = [\mathbf{\Pi}_\alpha^{(1)}(0)\mathbf{H}\mathbf{C}^{(r)}; u_1].$$

For the boundary conditions:

$$[\hat{\mathbf{\Psi}}_0; u_0] = [\mathbf{\Pi}_\alpha(0)\mathbf{H}\mathbf{C}^{(r)}; u_0], \qquad [\hat{\mathbf{\Psi}}_1; u_1] = [\mathbf{\Pi}_\alpha(1)\mathbf{H}\mathbf{C}^{(r)}; u_1].$$

The resulting algebraic system of linear equations becomes

$$\hat{\mathbf{\Psi}}_r \mathbf{C}^{(r)} = \hat{\mathbf{\Omega}}_r. \tag{4.17}$$

By solving this system, the unknown Genocchi coefficients in Eq. (4.1) can be accurately determined.

4.2. **Quasilinear Troesch equation.** For the linearized form of Troesch model in Eq. (3.4), we use collocation points given in Eq. (4.9) to obtain

$${}^{LC}D_t^\beta u_{r+1}(t_k) - \left[\lambda^2 \cosh(\lambda\, u_r(t_k))\right] u_{r+1}(t_k) = -\lambda^2 u_r(t_k)\cosh(\lambda\, u_r(t_k)) + \lambda \sinh(\lambda\, u_r(t_k)), \tag{4.18}$$

for $r = 0, 1, \ldots, 5$. We can express the previous equation in a more compact form by using the matrix notations defined in Eqs. (4.10) and (4.11) as follows:

$$\mathbf{U}_{r+1}^{(\beta)} + \mathbf{A}_r \mathbf{U}_{r+1} = \mathbf{\Omega}_r, \tag{4.19}$$

where

$$\mathbf{A}_r = \begin{pmatrix} -\lambda^2\cosh(\lambda u_r(t_0)) & 0 & \cdots & 0 \\ 0 & -\lambda^2\cosh(\lambda u_r(t_1)) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & -\lambda^2\cosh(\lambda u_r(t_N)) \end{pmatrix}_{(N+1)\times(N+1)},$$

$$\mathbf{\Omega}_r = \begin{pmatrix} -\lambda^2 u_r(t_0)\cosh(\lambda u_r(t_0)) + \lambda\sinh(\lambda u_r(t_0)) \\ -\lambda^2 u_r(t_1)\cosh(\lambda u_r(t_1)) + \lambda\sinh(\lambda u_r(t_1)) \\ \vdots \\ -\lambda^2 u_r(t_N)\cosh(\lambda u_r(t_N)) + \lambda\sinh(\lambda u_r(t_N)) \end{pmatrix}_{(N+1)\times 1}.$$

In a concise form, Eq. (4.19) can be expressed as:

$$\mathbf{\Psi}_r \mathbf{C}^{(r)} = \mathbf{\Omega}_r, \quad r = 0, 1, \ldots, 5, \tag{4.20}$$

where

$$\mathbf{\Psi}_r = (\mathbf{\Pi}^{(\beta)} + \mathbf{A}_r)\mathbf{H}. \tag{4.21}$$

Finally, it is necessary to define the initial guess $u_0(t)$ as an approximation of the solution. Additionally, each iteration must incorporate the boundary conditions satisfied in Eq. (3.5). This requires replacing the first two rows of the augmented matrix $[\mathbf{\Psi}_r; \mathbf{\Omega}_r]$ with the row matrices $[\hat{\mathbf{\Psi}}_0; u_0]$ and $[\hat{\mathbf{\Psi}}_1; u_1]$, expressed as

$$[\hat{\mathbf{\Psi}}_0; u_0] = [\mathbf{\Pi}_\alpha(0)\mathbf{H}\mathbf{C}^{(r)}; u_0], \qquad [\hat{\mathbf{\Psi}}_1; u_1] = [\mathbf{\Pi}_\alpha(1)\mathbf{H}\mathbf{C}^{(r)}; u_1].$$

The resulting system of algebraic linear equations is given by

$$\hat{\mathbf{\Psi}}_r \mathbf{C}^{(r)} = \hat{\mathbf{\Omega}}_r. \tag{4.22}$$

Solving this system allows for the determination of the unknown Genocchi coefficients in Eq. (4.1).

4.3. **Quasilinear Lane Emden equation.** In the third model, we use the collocation points given in Eq. (4.9) within its linearized form in Eq. (3.6) to obtain

$$ {}^{LC}D_t^{\beta} u_{r+1}(t_k) + \frac{d}{t_k^{\beta-1}} u'_{r+1}(t_k) + \eta\, e^{\eta\, u_r(t_k)} u_{r+1}(t_k) = e^{\eta\, u_r(t_k)} (\eta\, u_r(t_k) - 1),\ k = 0, 1, ..., N, \tag{4.23} $$

for $r = 0, 1, \ldots, 5$ and $\eta = \pm 1$. Utilizing the relations in Eqs. (4.10), (4.11), and (4.12), the above equation can be rewritten in matrix form as

$$ \mathbf{U}_{r+1}^{(\beta)} + \mathbf{B}_r \dot{\mathbf{U}}_{r+1} + \mathbf{A}_r \mathbf{U}_{r+1} = \mathbf{\Omega}_r, \tag{4.24} $$

where

$$ \mathbf{B}_r = \begin{pmatrix} \frac{d}{t_0^{\beta-1}} & 0 & \cdots & 0 \\ 0 & \frac{d}{t_1^{\beta-1}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{d}{t_N^{\beta-1}} \end{pmatrix}_{(N+1)\times(N+1)}, \quad \mathbf{\Omega}_r = \begin{pmatrix} e^{\eta\, u_r(t_0)} (\eta\, u_r(t_0) - 1) \\ e^{\eta\, u_r(t_1)} (\eta\, u_r(t_1) - 1) \\ \vdots \\ e^{\eta\, u_r(t_N)} (\eta\, u_r(t_N) - 1) \end{pmatrix}_{(N+1)\times 1}, $$

$$ \mathbf{A}_r = \begin{pmatrix} \eta\, e^{\eta\, u_r(t_0)} & 0 & \cdots & 0 \\ 0 & \eta\, e^{\eta\, u_r(t_1)} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \eta\, e^{\eta\, u_r(t_N)} \end{pmatrix}_{(N+1)\times(N+1)}. $$

Thus, the fundamental matrix equation corresponding to Eq. (4.24) is given by

$$ \mathbf{\Psi}_r \mathbf{C}^{(r)} = \mathbf{\Omega}_r, \quad r = 0, 1, \ldots, 5, \tag{4.25} $$

where

$$ \mathbf{\Psi}_r = \left(\mathbf{\Pi}^{(\beta)} + \mathbf{B}_r \mathbf{\Pi}^{(1)} + \mathbf{A}_r \mathbf{\Pi} \right) \mathbf{H}. \tag{4.26} $$

Finally, it is essential to consider the initial or boundary conditions given in Eq. (3.7). Accordingly, we replace the first two rows of the augmented matrix $[\mathbf{\Psi}_r; \mathbf{\Omega}_r]$ with the row matrices $[\hat{\mathbf{\Psi}}_0; u_0]$ and $[\hat{\mathbf{\Psi}}_1; u_1]$ given by:
For the initial conditions:

$$ [\hat{\mathbf{\Psi}}_0; u_0] = [\mathbf{\Pi}_\alpha(0) \mathbf{H} \mathbf{C}^{(r)}; u_0], \qquad [\hat{\mathbf{\Psi}}_1; u_1] = [\mathbf{\Pi}_\alpha^{(1)}(0) \mathbf{H} \mathbf{C}^{(r)}; u_1]. $$

For the boundary conditions:

$$ [\hat{\mathbf{\Psi}}_0; u_0] = [\mathbf{\Pi}_\alpha^{(1)}(0) \mathbf{H} \mathbf{C}^{(r)}; u_0], \qquad [\hat{\mathbf{\Psi}}_1; u_1] = [\mathbf{\Pi}_\alpha(1) \mathbf{H} \mathbf{C}^{(r)}; u_1]. $$

The resulting system of algebraic linear equations takes the form

$$ \hat{\mathbf{\Psi}}_r \mathbf{C}^{(r)} = \hat{\mathbf{\Omega}}_r. \tag{4.27} $$

By solving this system of linear equations, we determine the unknown Genocchi coefficients in Eq. (4.1).
Next, we will conduct a detailed error analysis for the proposed G-QLM technique.

5. Error analysis and residual error

In this section, we present the error analysis and residual error estimation for the G-QLM in solving the general form of the model given in Eq. (1.1). The analysis aims to evaluate the accuracy and efficiency of the proposed approach by examining the residual error behavior and its convergence properties.

5.1. **Error analysis of G-QLM.** Our goal in this subsection is to analysis the suggested G-QLM's error analysis in the weighted L_2-norm which defined as

$$\|u\|_{2,w} = \sqrt{\int_0^1 |u(t)|^2 \, w(t) dt},$$

where $w(t)$ is the weight function and given as $w(t) = t^{\alpha-1}$.

Let $u_{r+1}(t)$ represent the QLM solution of Eq. (3.1), which is obtained after the r-th iteration. Assume that $u(t)$ is the exact solution of model (1.1) with the initial or boundary conditions specified in Eqs. (1.3), (1.5), and (1.7). We know that $u_{N,\alpha}^{(r+1)}(t)$ is the approximate solution to $u_{r+1}(t)$ based on Eq. (4.1). The error bound for the proposed technique can be estimated using the following equation:

$$\left\| u(t) - u_{N,\alpha}^{(r+1)}(t) \right\|_{2,w} \leq \|u(t) - u_{N,\alpha}(t)\|_{2,w} + \left\| u_{N,\alpha}(t) - u_{N,\alpha}^{(r+1)}(t) \right\|_{2,w}. \tag{5.1}$$

Now, as a result, we need to approximate each of these terms, and thus, the following two theorems demonstrate the first term $\|u(t) - u_{N,\alpha}(t)\|_{2,w}$ and the second term $\left\| u_{N,\alpha}(t) - u_{N,\alpha}^{(r+1)}(t) \right\|_{2,w}$.

Theorem 5.1. *Let us assume that* $\mathbb{G}_N^\alpha = Span\{\mathbb{G}_{1,\alpha}(t), \mathbb{G}_{2,\alpha}(t), \ldots, \mathbb{G}_{N,\alpha}(t)\}$ *and* $u(t) \in C^{N+1}[0,1]$. *If* $u_{N,\alpha}(t) = \mathbf{G}_\alpha(t)\mathbf{C}$ *denotes the best approximation to* $u(t)$ *from* $\mathbf{G}_N^\alpha$, *then the error bound is given by*

$$|u(t) - u_{N,\alpha}(t)|_{2,w} \leq \frac{M}{((N+1)\alpha)!\sqrt{(2N+3)\alpha}}, \tag{5.2}$$

where $\mathbf{C} = \begin{bmatrix} c_0 & c_1 & \ldots & c_N \end{bmatrix}^T$, *and* $M = max_{t\in[0,1]}|u^{(N+1)}(t)|$.

Proof. Based on [41], we start the proof. Let $u_{N,\alpha}(t)$ is a Taylor expansion for approximate of $u(t)$ about $t = 0$:

$$u_{N,\alpha}(t) = \sum_{j=0}^{N} \frac{u^{(j\alpha)}(t)}{(j\alpha)!}(t)^{j\alpha},$$

then

$$|u(t) - u_{N,\alpha}(t)| \leq \frac{M}{((N+1)\alpha)!} |t|^{\alpha(N+1)},$$

where M is a maximum value of $|u^{((N+1)\alpha)}|$ on $[0,1]$.
Since $u_{N,\alpha}(t)$ is the best approximation of $u(t)$ and the best approximation is unique, then we can write

$$\begin{aligned}
\|u(t) - u_{N,\alpha}(t)\|_{2,w} \leq & \| \frac{M}{((N+1)\alpha)!} |t|^{\alpha(N+1)} \|_{2,w} \\
= & \sqrt{\int_0^1 \left(\frac{M}{((N+1)\alpha)!} |t|^{\alpha(N+1)} \right)^2 w(t) dt} \\
= & \frac{M}{((N+1)\alpha)!} \sqrt{\int_0^1 t^{2\alpha N + 2\alpha} \, t^{\alpha-1} \, dt} \\
= & \frac{M}{((N+1)\alpha)!} \sqrt{\int_0^1 t^{2\alpha N + 3\alpha - 1} \, dt},
\end{aligned}$$

then, by solving the integral we obtian

$$|u(t) - u_{N,\alpha}(t)|_{2,w} \leq \frac{M}{((N+1)\alpha)!\sqrt{(2N+3)\alpha}}.$$

■

Theorem 5.2. *Let $u(t)$ and $u_{N,\alpha}^{(r+1)}(t)$ be the exact and G-QLM solutions to Eqs. (1.1) and (3.1), respectively, under the hypothesis of Theorem (5.1). Then, the upper bound that follows is appropriate*

$$\|u(t)-u_{N,\alpha}^{(r+1)}(t)\|_{2,w}\leq\frac{M}{(N\alpha+1)!\sqrt{(2N+3)\alpha}}+\frac{Q_N}{\alpha}\|H\|_2\|C-C^{(r)}\|_2. \tag{5.3}$$

Proof. Based on Theorem (5.1), which establishes the first part of Eq. (5.1), we now analyze the second part of Eq. (5.1). By utilizing Eq. (4.6), we derive the following relationship:

$$\|u_{N,\alpha}(t)-u_{N,\alpha}^{(r+1)}(t)\|_{2,w}=\left\|\mathbf{\Pi}_\alpha(t)\mathbf{HC}-\mathbf{\Pi}_\alpha(t)\mathbf{HC}^{(r)}\right\|_{2,w}, \tag{5.4}$$

where $\mathbf{C}=\begin{bmatrix}c_1 & c_2 & \dots & c_{N+1}\end{bmatrix}^T$ and $\mathbf{C}^{(r)}=\begin{bmatrix}c_1^{(r)} & c_2^{(r)} & \dots & c_{N+1}^{(r)}\end{bmatrix}^T$. Applying the norm's properties reveals that

$$\|u_{N,\alpha}(t)-u_{N,\alpha}^{(r+1)}(t)\|_{2,w}\leq\|\mathbf{\Pi}_\alpha(t)\|_{2,w}\|\mathbf{H}\|_2\|\mathbf{C}-\mathbf{C}^{(r)}\|_2. \tag{5.5}$$

For the first term in Eq. (5.5), we obtain

$$\|\mathbf{\Pi}_\alpha(t)\|_{2,w}^2=\int_0^1|\mathbf{\Pi}_\alpha(t)|^2\,w(t)dt=\int_0^1|\mathbf{\Pi}_\alpha(t)|^2\,t^{\alpha-1}dt, \tag{5.6}$$

where the weight function $w(t)=t^{\alpha-1}$. However, we also have

$$|\mathbf{\Pi}_\alpha(t)|^2=\sum_{i=0}^{N}t^{2i\alpha}, \tag{5.7}$$

which gives us:

$$\|\mathbf{\Pi}_\alpha(t)\|_{2,w}^2=\sum_{i=0}^{N}\int_0^1 t^{2i\alpha+\alpha-1}dt=\frac{1}{\alpha}Q_N,\quad Q_N=\sum_{i=0}^{N}\frac{1}{2i+1}. \tag{5.8}$$

Thus, the proof is complete. ■

5.2. **Residual error functions.** In this subsection, we will further validate the results by calculating the residual error for the model (1.1). For $1<\beta\leq 2$, the exact analytical solution for most fractional-order differential equations is generally unknown. Validating the accuracy of the provided collocation technique is essential in this case. Let $\mathcal{R}_{N,\alpha}^{(r+1)}(t)$ represent the residual error function. This function is determined by substituting the truncated Genocchi series solutions from Eq. (4.1) into the Eqs. (1.2), (1.4), and (1.6). As a result, we obtain:

$$\mathcal{R}_{N,\alpha}^{(r+1)}(t)=\begin{cases}{}^{LC}D_t^\beta u_{N,\alpha}^{(r+1)}(t)+\lambda\,e^{\eta\,u_{N,\alpha}^{(r+1)}(t)}\approx 0,\\ {}^{LC}D_t^\beta u_{N,\alpha}^{(r+1)}(t)-\lambda\,\sinh\left(\lambda\,u_{N,\alpha}^{(r+1)}(t)\right)\approx 0,\\ {}^{LC}D_t^\beta u_{N,\alpha}^{(r+1)}(t)+\frac{d}{t^{\beta-1}}\,u_{N,\alpha}'^{(r+1)}(t)+e^{\eta\,u_{N,\alpha}^{(r+1)}(t)}\approx 0,\end{cases}\qquad t\in[0,1], \tag{5.9}$$

where $1<\beta\leq 2$, $\eta=\pm 1$, and $r=0,1,\dots,5$. Indeed, at the collocation points specified in Eq. (4.9), the residual error functions $\mathcal{R}_{N,\alpha}^{(r+1)}(t)$ become zero. As a result, we anticipate that as N increases, the residual error $\mathcal{R}_{N,\alpha}^{(r+1)}(t)$ converges to zero, ensuring that the approximate solution becomes increasingly accurate.

6. Numerical simulations

In this section, we present various examples for Bratu equation (BE), Troesch equation (TE), and Lane Emden equation (LEE) solving by the proposed G-QLM. All calculations were performed through utilizing MATLAB software version R2023a. We begin by the following example.

Example 6.1. *[30] In the first example, let us consider the Bratu model problem on* $[0,1]$ *with* $\lambda = -2$ *and* $\eta = 1$ *as*

$$ {}^{LC}D_t^{\beta}\, u(t) - 2e^{u(t)} = 0, \quad 1 < \beta \leq 2,$$

with the initial conditions

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

If $\beta = 2$*, the exact solution is given by* $u(t) = ln(1/cos^2 t)$*.*

We assume that $\beta = 2$ *and* $\alpha = 1$ *and utilize* $N = 10$*. The approximate solution* $u_{10,1}^{(6)}(t)$ *on* $0 \leq t \leq 1$ *with iteration parameter* $r = 5$ *and initial guess* $u_0(t) = t^2$ *utilizing the Genocchi basis functions via the G-QLM approach, is given by*

$$\begin{aligned} u_{10,1}^{(6)}(t) = {} & 0.50335716\, t^{10} - 2.50223272\, t^9 + 5.60021948\, t^8 - 7.20065056\, t^7 + 5.9108073\, t^6 \\ & - 3.12163685\, t^5 + 1.25238525\, t^4 - 0.23937328\, t^3 + 1.03166445\, t^2 \\ & + 2.35098870 \times 10^{-38}\, t - 9.18354962 \times 10^{-41}. \end{aligned}$$

Figure 1 illustrates the exact solution alongside the approximate solution for $\beta = 2, \alpha = 1$*, and* $N = 10$*. Table 1 presents absolute errors using G-QLM for different values of* $N = 20, 25,$ *and* 30 *with* $r = 5$ *at selected points* $t \in [0,1]$*, and CPU times* $4.008, 5.050, 6.350$ *sec, respectively, comparing our approach with the Bessel-QLM method [30]. Tables 2 and 3 provide comparisons for G-QLM method for* $N = 7$ *with CPU time* 2.177 *sec and* $N = 10$ *with CPU time* 2.530 *sec, with other numerical methods, including Bessel-QLM [30], the Legendre spectral method (LSM) [57], the reproducing kernel method (RKM) [10], and the Taylor wavelets method (TWM) [34]. The data in Tables 1 and 3 indicates that our proposed linearized G-QLM method yields smaller errors than Bessel-QLM, TWM, RKM, and LSM. We notice from Table 2 that Bessel-QLM yields slightly more accurate results than G-QLM when* N *is small. On the other hand, in Table 3, G-QLM is more accurate than Bessel-QLM when* N *increases.*

Returning to the fractional-order case, we analyze different values of $\beta = 2, 1.9, ...1.6$*, maintaining* $\alpha = \beta$ *in the calculations for each* β*. The approximate solutions for* $N = 10$ *and* $r = 5$ *are shown in Figure 2. Additionally, Table 4 presents further comparisons of numerical solutions obtained using G-QLM, Bessel-QLM [30], RKM [10], and the fractional differential transform method (FDTM) [16] for various values of* $\alpha = \beta = 1.6, 1.7, 1.8, 1.9$ *evaluated at* $t = 0.1$*. The results demonstrate that increasing* N *leads to improved accuracy.*

TABLE 1. The comparison of absolute errors using G-QLM in Example 6.1 for $\beta = 2$, $\alpha = 1$, $N = 20, 25, 30$, and various $t \in [0,1]$ with Bessel-QLM.

t	*G- QLM* ($\alpha = 1$)			*Bessel- QLM* ($\alpha = 1$) *[30]*		
	$N = 20$	$N = 25$	$N = 30$	$N = 20$	$N = 25$	$N = 30$
0.1	1.2640×10^{-9}	2.3190×10^{-12}	3.7435×10^{-15}	2.5768×10^{-8}	1.1549×10^{-10}	5.0955×10^{-13}
0.2	2.6556×10^{-9}	4.8023×10^{-12}	7.6883×10^{-15}	5.6983×10^{-8}	2.4911×10^{-10}	1.0880×10^{-12}
0.3	4.1029×10^{-9}	7.3867×10^{-12}	1.1810×10^{-14}	8.9392×10^{-8}	3.8795×10^{-10}	1.6892×10^{-12}
0.4	5.6408×10^{-9}	1.0134×10^{-11}	1.6182×10^{-14}	1.2378×10^{-7}	5.3536×10^{-10}	2.3278×10^{-12}
0.5	7.3125×10^{-9}	1.3123×10^{-11}	2.0928×10^{-14}	1.6110×10^{-7}	6.9550×10^{-10}	3.0217×10^{-12}
0.6	9.1761×10^{-9}	1.6455×10^{-11}	2.6257×10^{-14}	2.0265×10^{-7}	8.7388×10^{-10}	3.7949×10^{-12}
0.7	1.1312×10^{-8}	2.0277×10^{-11}	3.2307×10^{-14}	2.5022×10^{-7}	1.0782×10^{-09}	4.6088×10^{-12}
0.8	1.3841×10^{-8}	2.4801×10^{-11}	3.9524×10^{-14}	3.0647×10^{-7}	1.3199×10^{-09}	5.7288×10^{-12}
0.9	1.6949×10^{-8}	3.0365×10^{-11}	4.8406×10^{-14}	3.7556×10^{-7}	1.6170×10^{-09}	7.0170×10^{-12}
1.0	2.0955×10^{-8}	3.7537×10^{-11}	5.9730×10^{-14}	4.6455×10^{-7}	1.9996×10^{-09}	8.6847×10^{-12}

Example 6.2. *[30] In the second example, let us consider the boundary value Bratu equation with* $\lambda = -\pi^2$*,* $\eta = 1$

$$ {}^{LC}D_t^{\beta}\, u(t) - \pi^2 e^{u(t)} = 0, \quad 1 < \beta \leq 2, \quad 0 < t < 1,$$

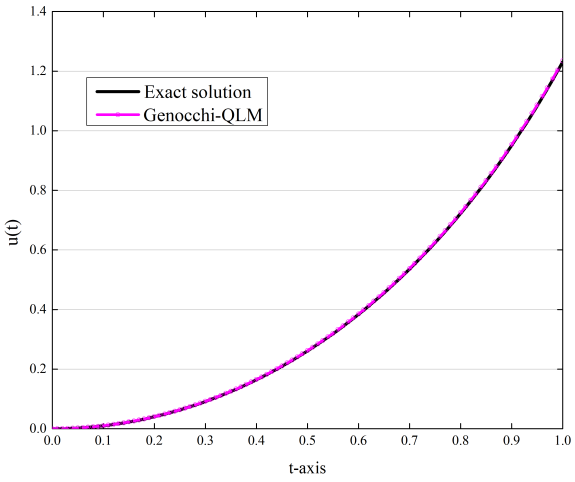

FIGURE 1. The graph of approximated G-QLM and exact solutions for $\beta = 2$, $\alpha = 1$, $N = 10$, and $r = 5$ in Example 6.1.

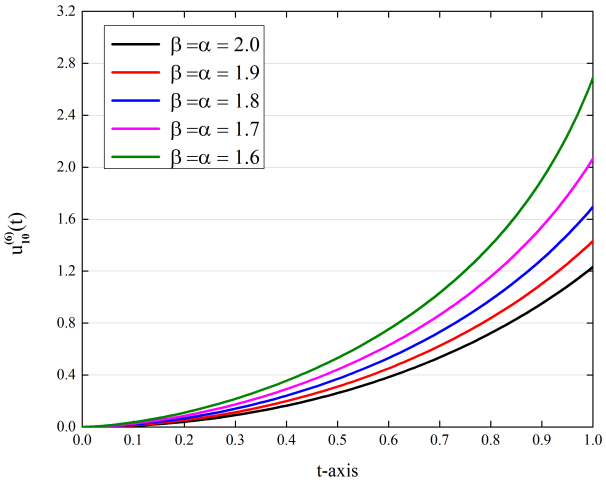

FIGURE 2. The approximated solution with different $\beta = \alpha = 2, 1.9, 1.8, 1.7, 1.6$ for $N = 10$ using G-QLM with $r = 5$ in Example 6.1.

TABLE 2. The comparison of absolute errors using G-QLM in Example 6.1 for $\beta = 2$, $\alpha = 2$, $N = 7, 10$, with Bessel-QLM.

t	*G-QLM* ($\alpha = 2$)		*Bessel-QLM* ($\alpha = 2$) *[30]*	
	$N = 7$	$N = 10$	$N = 7$	$N = 10$
0.1	3.4444×10^{-6}	1.5316×10^{-9}	4.1979×10^{-8}	2.0195×10^{-11}
0.2	1.1879×10^{-5}	4.0414×10^{-9}	1.2167×10^{-7}	4.6897×10^{-11}
0.3	1.8565×10^{-5}	6.3507×10^{-9}	1.8565×10^{-7}	7.4344×10^{-11}
0.4	3.0592×10^{-5}	9.5300×10^{-9}	2.6108×10^{-7}	1.0203×10^{-10}
0.5	4.0337×10^{-5}	1.1713×10^{-8}	3.5454×10^{-7}	1.3688×10^{-10}
0.6	5.2484×10^{-5}	1.4669×10^{-8}	4.1000×10^{-7}	1.5831×10^{-10}
0.7	6.6027×10^{-5}	2.0527×10^{-8}	5.7919×10^{-7}	2.6467×10^{-10}
0.8	7.9968×10^{-5}	2.1239×10^{-8}	6.8266×10^{-7}	1.8342×10^{-10}
0.9	9.8443×10^{-5}	2.9898×10^{-8}	3.0427×10^{-7}	6.5824×10^{-9}
1.0	1.2283×10^{-4}	3.4831×10^{-8}	3.2344×10^{-5}	3.9874×10^{-7}

with the boundary conditions

$$u(0) = 0, \quad u(1) = 0.$$

If $\beta = 2$, the exact solution is given by

$$u(t) = -2 \ln \cosh \left[\frac{(t - 0.5)\xi}{2} \right] + 2 \ln \cosh \frac{\xi}{4},$$

where ξ satisfies $\xi = \sqrt{2\lambda} \cosh \frac{\xi}{4}$. In particular, two exact solutions are given by $u(t) = -\ln\left(1 \pm \cos\left(\frac{\pi}{2}t + \pi\right)\right)$.

TABLE 3. The comparison of absolute errors using G-QLM in Example 6.1 for $\beta = 2$, $\alpha = 2$, $N = 7, 10$, with LSM, TWM, and RKM.

t	*G-QLM* ($\alpha = 2$)		*LSM* ($n = 9$) *[57]*		*TWM [34]*	*RKM [10]*
	$N = 7$	$N = 10$	*Method (a)*	*Method (b)*	$k = 1, M_1 = 7$	$n, N = 10$
0.1	3.4444×10^{-6}	1.5316×10^{-9}	1.4169×10^{-5}	1.7830×10^{-7}	2.69611×10^{-5}	1.6674×10^{-5}
0.2	1.1879×10^{-5}	4.0414×10^{-9}	3.2272×10^{-5}	4.5055×10^{-7}	2.38968×10^{-5}	2.1551×10^{-7}
0.3	1.8565×10^{-5}	6.3507×10^{-9}	5.1244×10^{-5}	7.1998×10^{-7}	1.01329×10^{-5}	1.1310×10^{-6}
0.4	3.0592×10^{-5}	9.5300×10^{-9}	7.1441×10^{-5}	1.0081×10^{-6}	2.12408×10^{-5}	2.1200×10^{-6}
0.5	4.0337×10^{-5}	1.1713×10^{-8}	9.2812×10^{-5}	1.3195×10^{-6}	1.15316×10^{-5}	2.9000×10^{-6}
0.6	5.2484×10^{-5}	1.4669×10^{-8}	1.1720×10^{-4}	1.6653×10^{-6}	1.85136×10^{-5}	4.1000×10^{-6}
0.7	6.6027×10^{-5}	2.0527×10^{-8}	1.4496×10^{-4}	2.0620×10^{-6}	1.15473×10^{-5}	6.5000×10^{-6}
0.8	7.9968×10^{-5}	2.1239×10^{-8}	1.7718×10^{-4}	2.2525×10^{-6}	2.26494×10^{-5}	7.5000×10^{-6}
0.9	9.8443×10^{-5}	2.9898×10^{-8}	2.1791×10^{-4}	3.1212×10^{-6}	1.13933×10^{-5}	3.3500×10^{-5}
1.0	1.2283×10^{-4}	3.4831×10^{-8}	2.6999×10^{-4}	3.6311×10^{-6}	8.55545×10^{-5}	4.3700×10^{-5}

TABLE 4. The comparison of the numerical results using G-QLM for $N = 10$ and various $\beta = \alpha = 1.9, 1.8, 1.7, 1.6$ at $t = 0.1$ with other methods in Example 6.1.

β	*G-QLM* ($\alpha = \beta$)	*Bessel-QLM* ($\alpha = \beta$) *[30]*	*RKM* ($N = 10$) *[10]*	*FDTM* ($N = 5$) *[16]*
1.9	1.3814×10^{-2}	1.3814×10^{-2}	1.3082×10^{-2}	1.3814×10^{-2}
1.8	1.8983×10^{-2}	1.8983×10^{-2}	1.7221×10^{-2}	1.8983×10^{-2}
1.7	2.6003×10^{-2}	2.5993×10^{-2}	2.2504×10^{-2}	2.5993×10^{-2}
1.6	3.5918×10^{-2}	3.5473×10^{-2}	2.9221×10^{-2}	3.5472×10^{-2}

We first consider the case where $\beta = 2$ and $\alpha = 1$. Using the G-QLM approach with initial approximate $u_0(t) = t^2 - t$ and $r = 5$, the following approximate solution is obtained for $N = 8$ over the interval $t \in [0, 1]$:

$$u_{8,1}^{(6)}(t) = 1.412556535\,t^8 - 6.01217037 71\,t^7 + 11.615174302\,t^6 - 13.067682472\,t^5 + 10.261254812\,t^4 - 6.218550918\,t^3 + 5.175424735\,t^2 - 3.166006618\,t.$$

Table 5 presents a comparison of the approximate solutions and absolute errors obtained using the G-QLM for the Bratu equation with $\beta = 2, \alpha = 1, N = 30$, $r = 5$, and CPU time 6.262 sec. The results are compared against those obtained using the Bessel-QLM method [30]. Furthermore, for different values of $\beta = 1.5, 1.7, 1.9$, with $N = 15, \alpha = 1$, $\lambda = -\pi^2$, and CPU times 3.204, 3.188, 3.191 sec, we use the G-QLM method to compute approximate solutions. Table 6 provides a comparison of these solutions with those derived from the Bessel-QLM method.

Example 6.3. *[37] In this example, consider the general form of Troesch boundary value problem*

$$^{LC}D_t^{\beta} u(t) = \lambda \sinh(\lambda u(t)), \quad 0 < t < 1, \quad 1 < \beta \leq 2,$$

subject to the boundary conditions

$$u(0) = 0, \quad u(1) = 1.$$

By setting $N = 15$, $\alpha = 1$, $r = 5$ and taking the initial approximate $u_0(t) = t$, we compare the absolute residual errors for different values of $\lambda = 0.5$, 1, 1.5 at fractional derivative orders $\beta = 1.5$, 1.9. These comparisons are detailed in Tables 7, 8, and 9, as follows:

- *$\lambda = 0.5$ and $\beta = 1.5$, 1.9:*
  Table 7 presents the absolute residual errors obtained using the G-QLM method with CPU times 3.401 and 3.388 sec compared to the results from the Picard method [37].

TABLE 5. Comparison of approximate solutions and absolute errors for $N = 30$, $\beta = 2$, $\alpha = 1$ using G-QLM and Bessel-QLM in Example 6.2.

t	*G-QLM*		*Bessel-QLM [30]*	
	$u^{(6)}_{30,1}(t)$	$\mathcal{E}^{(6)}_{30,1}(t)$	$u^{(6)}_{30,1}(t)$	$\mathcal{E}^{(6)}_{30,1}(t)$
0.1	*-0.269276469559262*	7.7716×10^{-16}	*-0.26927646955801*	7.6043×10^{-13}
0.2	*-0.462340122126475*	6.1062×10^{-16}	*-0.462340122126269*	2.0527×10^{-13}
0.3	*-0.592783600716709*	4.4409×10^{-16}	*-0.592783607107044*	3.3595×10^{-13}
0.4	*-0.668371029081565*	3.3307×10^{-16}	*-0.668371020982459*	8.9516×10^{-13}
0.5	*-0.693147180559946*	3.3307×10^{-16}	*-0.693147180561445*	1.4998×10^{-12}
0.6	*-0.668371029081564*	1.1102×10^{-16}	*-0.668371020982473*	2.1792×10^{-12}
0.7	*-0.592783600716708*	1.1102×10^{-16}	*-0.592783607019678*	2.9697×10^{-12}
0.8	*-0.462340122126475*	5.5511×10^{-17}	*-0.462340122130420*	3.9452×10^{-12}
0.9	*-0.269276469559262*	$0.0000 \times 10^{+00}$	*-0.269276469564491*	5.2293×10^{-12}

TABLE 6. Numerical solutions for $\beta = 1.9, 1.7, 1.5$, $N = 15$, $\lambda = -\pi^2$, and $\alpha = 1$ using G-QLM and Bessel-QLM in Example 6.2.

	G-QLM			*Bessel-QLM [30]*		
t	$\beta = 1.5$	$\beta = 1.7$	$\beta = 1.9$	$\beta = 1.5$	$\beta = 1.7$	$\beta = 1.9$
0.1	*-0.293611*	*-0.294465*	*-0.279943*	*-0.294195*	*-0.294574*	*-0.279940*
0.2	*-0.449542*	*-0.475222*	*-0.471698*	*-0.450191*	*-0.475380*	*-0.471703*
0.3	*-0.545317*	*-0.589202*	*-0.598030*	*-0.545938*	*-0.589367*	*-0.598037*
0.4	*-0.598920*	*-0.652218*	*-0.669682*	*-0.599428*	*-0.652366*	*-0.669690*
0.5	*-0.616292*	*-0.670707*	*-0.691838*	*-0.616646*	*-0.670828*	*-0.691846*
0.6	*-0.597832*	*-0.646155*	*-0.666091*	*-0.597974*	*-0.646235*	*-0.666096*
0.7	*-0.539747*	*-0.576277*	*-0.591039*	*-0.539612*	*-0.576307*	*-0.591401*
0.8	*-0.433549*	*-0.454803*	*-0.462078*	*-0.433043*	*-0.454679*	*-0.462076*
0.9	*-0.263517*	*-0.269881*	*-0.270319*	*-0.262495*	*-0.269766*	*-0.270311*

- $\lambda = 1$ *and* $\beta = 1.5, 1.9$*:*
  Table 8 provides a comparison of the absolute residual errors between the G-QLM method with CPU times 3.400 *and* 3.419 *sec and the Picard method [37].*
- $\lambda = 1.5$ *and* $\beta = 1.5$*:*
  Table 9 presents the approximate solutions and the absolute residual errors obtained using the G-QLM method with CPU times 3.460 *and* 3.484 *sec and compares them with those obtained from the Picard method [37].*

These comparisons highlight the accuracy and efficiency of the G-QLM method in solving fractional-order of Troesch differential equations, demonstrating its superiority in reducing absolute residual errors compared to the Picard method.

Example 6.4. *[53] Consider the following initial value Lane–Emden equation with* $d = 2$

$$ {}^{LC}D^{\beta}_{t}u(t) + \frac{2}{t^{\beta-1}}u'(t) + e^{\eta\, u(t)} = 0, \quad 1 < \beta \leq 2, \quad 0 < t < 1, $$

with the initial conditions

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

After applying the G-QLM method to solve the Lane-Emden equation with $u_0(t) = t^2$, $\alpha = 1$, *and* $d = 2$, *we compare the approximate solutions and absolute residual errors at* $N = 7$, $\beta = 2$, $\eta = 1$, *and* $r = 5$ *with the results*

TABLE 7. Comparison of absolute residual errors using G-QLM for $N = 15$, $\alpha = 1$, $\lambda = 0.5$, $\beta = 1.5, 1.9$, and Picard method in Example 6.3.

t	*G-QLM* $\mathcal{E}_{15,1}^{(6)}(t)$		*Picard Method* $\mathcal{E}(t)$*[37]*	
	$\beta = 1.5$	$\beta = 1.9$	$\beta = 1.5$	$\beta = 1.9$
0.1	2.5181×10^{-6}	4.8066×10^{-7}	3.6×10^{-4}	1.2×10^{-6}
0.2	4.3977×10^{-7}	6.5740×10^{-8}	6.8×10^{-4}	2.3×10^{-6}
0.3	2.3859×10^{-7}	3.0767×10^{-8}	9.5×10^{-4}	3.1×10^{-6}
0.4	1.2214×10^{-7}	1.4156×10^{-8}	1.1×10^{-3}	3.5×10^{-6}
0.5	7.0497×10^{-8}	7.5142×10^{-9}	1.2×10^{-3}	3.6×10^{-6}
0.6	4.4714×10^{-8}	4.4485×10^{-9}	1.2×10^{-3}	3.3×10^{-6}
0.7	2.9221×10^{-8}	2.7417×10^{-9}	1.1×10^{-3}	2.6×10^{-6}
0.8	1.4139×10^{-8}	1.2608×10^{-9}	6.4×10^{-4}	1.7×10^{-6}
0.9	1.0550×10^{-8}	8.9925×10^{-10}	4.3×10^{-4}	6.3×10^{-7}

TABLE 8. Comparison of absolute residual errors using G-QLM for $N = 15$, $\alpha = 1$, $\lambda = 1$, $\beta = 1.5, 1.9$, and Picard method in Example 6.3.

t	*G-QLM* $\mathcal{E}_{15,1}^{(6)}(t)$		*Picard Method* $\mathcal{E}(t)$*[37]*	
	$\beta = 1.5$	$\beta = 1.9$	$\beta = 1.5$	$\beta = 1.9$
0.1	7.9756×10^{-6}	1.6688×10^{-6}	8.6×10^{-4}	3.6×10^{-4}
0.2	1.3940×10^{-6}	2.2837×10^{-7}	1.4×10^{-3}	6.8×10^{-4}
0.3	7.5649×10^{-7}	1.0692×10^{-7}	1.8×10^{-3}	9.5×10^{-4}
0.4	3.8724×10^{-7}	4.9203×10^{-8}	1.9×10^{-3}	1.1×10^{-3}
0.5	2.2343×10^{-7}	2.6120×10^{-8}	1.7×10^{-3}	1.2×10^{-3}
0.6	1.4164×10^{-7}	1.5463×10^{-8}	1.3×10^{-3}	1.2×10^{-3}
0.7	9.2484×10^{-8}	9.5290×10^{-9}	6.8×10^{-4}	1.1×10^{-3}
0.8	4.4698×10^{-8}	4.3806×10^{-9}	3.0×10^{-4}	8.1×10^{-4}
0.9	3.3296×10^{-8}	3.1230×10^{-9}	3.8×10^{-4}	4.3×10^{-4}

TABLE 9. Comparison of approximate and absolute residual errors using G-QLM for $N = 15$, $\alpha = 1$, $\lambda = 1.5$, $\beta = 1.5$ with Picard method in Example 6.3.

t	*G-QLM*		*Picard Method[37]*	
	$u_{15,1}^{(6)}(t)$	$\mathcal{E}_{15,1}^{(6)}(t)$	$u(t)$	$\mathcal{E}(t)$
0.1	0.04995387987972	1.1044×10^{-5}	*0.04750460845553*	2.7×10^{-2}
0.2	0.10388246906098	1.7747×10^{-6}	*0.09970714745690*	4.8×10^{-2}
0.3	0.16391237534080	8.6350×10^{-7}	*0.15888723821570*	6.1×10^{-2}
0.4	0.23208497878280	3.8077×10^{-7}	*0.22705646204360*	6.7×10^{-2}
0.5	0.31068788583293	1.7652×10^{-7}	*0.30635574105404*	6.4×10^{-2}
0.6	0.40249457065939	7.7705×10^{-8}	*0.39932950411375*	5.5×10^{-2}
0.7	0.51108043107923	2.2064×10^{-8}	*0.50926124239340*	4.0×10^{-2}
0.8	0.64134418853010	7.7803×10^{-9}	*0.64070236071652*	1.9×10^{-2}
0.9	0.80047988823889	2.4959×10^{-8}	*0.80044390768920*	7.4×10^{-3}

obtained from the Laplace-residual power series (LRPS) method [53]. These comparisons are presented in Table 10. Additionally, Figure 3 illustrates the behavior of the approximate solutions for different values of β, specifically $\beta = 1.5, 1.6, 1.7, 1.8, 1.9,$ and 2. The figure is divided into two sections:

- *The left side of the figure displays the approximate solutions for $\eta = +1$.*
- *The right side of the figure shows the approximate solutions for $\eta = -1$.*

This comparative analysis further validates the efficiency and accuracy of the G-QLM method in solving the Lane-Emden equation across various fractional orders.

TABLE 10. Comparison of approximate and absolute residual errors using G-QLM for $N = 7$, $\alpha = 1$, $\eta = 1$, $\beta = 2$ with LRPS method in Example 6.4.

t	*G-QLM*		*LRPS [53]*	
	$u_{7,1}^{(6)}(t)$	$\mathcal{E}_{7,1}^{(6)}(t)$	$u_7(t)$	$\mathcal{E}_7(t)$
0.2	$-6.65340793 \times 10^{-3}$	5.3357×10^{-08}	$-6.65346031 \times 10^{-3}$	1.4774×10^{-8}
0.4	$-2.64555302 \times 10^{-2}$	1.1958×10^{-09}	$-2.64541446 \times 10^{-2}$	9.3774×10^{-7}
0.6	$-5.89441309 \times 10^{-2}$	3.5902×10^{-10}	$-5.89292571 \times 10^{-2}$	1.0535×10^{-5}
0.8	$-1.03386109 \times 10^{-1}$	4.7795×10^{-11}	$-1.03305346 \times 10^{-1}$	5.8046×10^{-5}
1.0	$-1.58827732 \times 10^{-1}$	4.5437×10^{-09}	$-1.58531746 \times 10^{-1}$	2.1591×10^{-4}

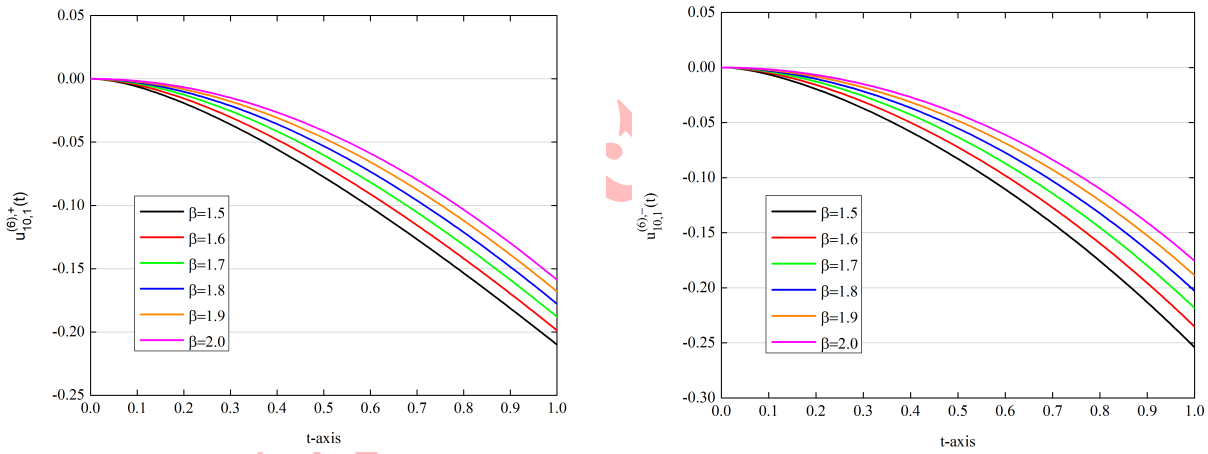

FIGURE 3. Approximated solutions with $\eta = +1$ (left) and $\eta = -1$ (right) in G-QLM for $N = 10$, $\alpha = 1$, $r = 5$, and various $\beta = 1.9, 1.8, ..., 1.5$ in Example 6.4.

Example 6.5. *[30, 62] Consider the boundary value Lane–Emden problem with $d = 1$ and $\eta = 1$*

$$ {}^{LC}D_t^{\beta} u(t) + \frac{1}{t^{\beta-1}} u'(t) + e^{u(t)} = 0, \quad 1 < \beta \leq 2, \quad 0 < t < 1, $$

with the boundary conditions

$$ u'(0) = 0, \quad u(1) = 0. $$

The exact solution for $\beta = 2$ is given by

$$ u(t) = 2 \ln \frac{C+1}{Ct^2+1}, $$

where $C = 3 - 2\sqrt{2}$.

In Table 11, we compare the absolute errors obtained using the G-QLM method with those from the Bessel-QLM method [30] and the Laguerre wavelets operational matrix method (LWOMM) [62]. This comparison is conducted for initial guess $u_0(t) = 0$, $\beta = 2, \alpha = 1, r = 5$, *and two different values of* N, *specifically* $N = 10$ *with CPU time* $3.449\,sec$ *and* $N = 15$ *with CPU time* $4.968\,sec$. *Furthermore, Figure 4 presents the graph of approximate solutions for different fractional-order values,* $\beta = 1.5, 1.6, 1.7, 1.8, 1.9, 2$, *with* $\alpha = 1$ *and* $N = 10$. *Additionally, it includes the exact solution for* $\beta = 2$. *The results demonstrate that our numerical solutions provide higher accuracy while requiring less computational effort, highlighting the efficiency of the G-QLM method.*

TABLE 11. Comparison of absoulte error of G-QLM with $N = 10, 15, \beta = 2, \alpha = 1, r = 5$, Bessel-QLM, and LWOMM methods for Example 6.5.

t	*G-QLM*		*Bessel-QLM [30]*		*LWOMM [62]*
	$\mathcal{E}^{(6)}_{10,1}(t)$	$\mathcal{E}^{(6)}_{15,1}(t)$	$\mathcal{E}^{(6)}_{10,1}(t)$	$\mathcal{E}^{(6)}_{15,1}(t)$	$k = 3, M_1 = 7$
0.1	6.5541×10^{-09}	1.3661×10^{-13}	1.1135×10^{-08}	2.3444×10^{-12}	1.12567×10^{-11}
0.2	4.7394×10^{-09}	1.0031×10^{-13}	8.2300×10^{-09}	1.7229×10^{-12}	1.06202×10^{-11}
0.3	3.7054×10^{-09}	7.7993×10^{-14}	6.3672×10^{-09}	1.3380×10^{-12}	9.71828×10^{-12}
0.4	2.8922×10^{-09}	6.0896×10^{-14}	4.9738×10^{-09}	1.0495×10^{-12}	8.57608×10^{-12}
0.5	2.2413×10^{-09}	4.7268×10^{-14}	3.8355×10^{-09}	7.28159×10^{-13}	7.28159×10^{-12}
0.6	1.6857×10^{-09}	3.5388×10^{-14}	2.8611×10^{-09}	6.1151×10^{-13}	5.78329×10^{-12}
0.7	1.1903×10^{-09}	2.5063×10^{-14}	2.0053×10^{-09}	4.3376×10^{-13}	4.31280×10^{-12}
0.8	7.5265×10^{-10}	1.5807×10^{-14}	1.2419×10^{-09}	2.7506×10^{-13}	2.82023×10^{-12}
0.9	3.5707×10^{-10}	7.5009×10^{-15}	5.5412×10^{-10}	1.3218×10^{-13}	1.33990×10^{-12}

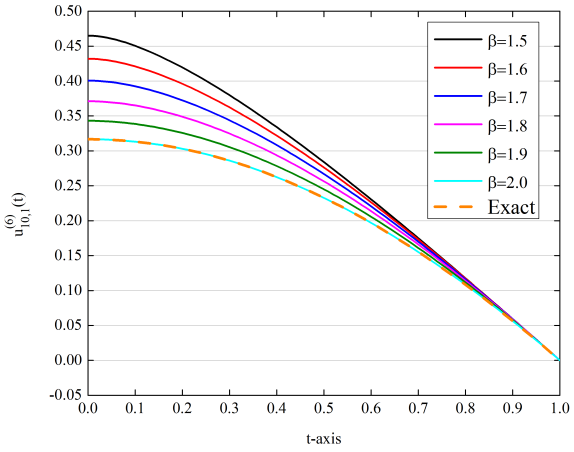

FIGURE 4. Numerical approximations in G-QLM for various $\beta = 2, 1.9, ..., 1.5, \alpha = 1, N = 10$ and the exact solution for Example 6.5.

7. CONCLUSION

In this paper, we introduced the Genocchi quasilinearization method (G-QLM) as an effective numerical chnique for solving nonlinear fractional boundary value problems. The method integrates the quasilinearization approach

with Genocchi polynomials to handle three well-known models: the Bratu, Troesch, and Lane-Emden equations. These models are widely used to describe complex phenomena in engineering, physics, and applied mathematics. By iteratively transforming nonlinear equations into linear subproblems, the G-QLM method demonstrated superior accuracy and efficiency compared to existing methods. We conducted a detailed error analysis and demonstrated the convergence of the method under various conditions, including different fractional orders and boundary constraints. Numerical experiments verified the reliability of the approach, with results showing significant reductions in absolute errors when compared to methods such as the Bessel quasilinearization, Taylor wavelets, and reproducing kernel methods. Additionally, the method proved to be computationally efficient, making it suitable for solving a broad range of nonlinear fractional models.

The G-QLM method provides a robust framework for solving fractional boundary value problems and holds great potential for applications requiring accurate numerical solutions to complex differential equations. Future work may explore further enhancements, including adaptive mesh refinement and hybrid polynomial techniques, to improve the method's performance for higher-dimensional and more complex systems.

Competing interests declaration: All the authors declare that there is no competing interests, or other interests that might be perceived to influence the results and/or discussion reported in this paper.

Data availability: Since no datasets were created or examined for this research, data sharing is not applicable.

Funding: Not applicable.

Acknowledgment: The authors would like to convey their thanks to the Editor and Reviewers for the helpful comments and suggestions which improved the work.

References

[1] J. Ahmad, M. Arshad, and Z. Ma, *Numerical solutions of Troesch's problem based on a faster iterative scheme with an application*, AIMS Math., *9* (2024), 9164-9183.
[2] S. Ahmed, S. Jahan, and K. S. Nisar, *Hybrid Fibonacci wavelet method to solve fractional-order logistic growth model*, Math. Methods Appl. Sci., *46* (2023), 16218-16231.
[3] M. P. Alam, T. Begum, and A. Khan, *A high-order numerical algorithm for solving Lane–Emden equations with various types of boundary conditions*, Comput. Appl. Math., *40* (2021).
[4] R. Alchikh and S. A. Khuri, *On the solutions of the fractional Bratu's problem*, J. Interdiscip. Math., *23* (2020), 1093-1107.
[5] L. K. Alzaki and H. K. Jassim, *Time fractional differential equations with an approximate solution*, J. Niger. Soc. Phys. Sci., *818* (2022).
[6] J. D. Anderson and J. Wendt, *Computational Fluid Dynamics*, *206* (1995).
[7] S. Araci, *Novel identities for q-Genocchi numbers and polynomials*, J. Funct. Spaces Appl., *2012* (2012), 1-13.
[8] A. Atta, J. Soliman, E. Elsaeed, M. Elsaeed, and Y. Youssri, *Spectral collocation algorithm for the fractional Bratu equation via hexic shifted Chebyshev polynomials*, Comput. Methods Differ. Equ., *2024* (2024), 1-15.
[9] R. O. Awonusika and O. A. Mogbojuri, *Approximate analytical solution of fractional Lane-Emden equation by Mittag-Leffler function method*, J. Niger. Soc. Phys. Sci., *4* (2022), 265-280.
[10] E. Babolian, S. Javadi, and E. Moradi, *RKM for solving Bratu-type differential equations of fractional order*, Math. Methods Appl. Sci., *39* (2015), 1548-1557.
[11] M. Ben-Romdhane, H. Temimi, and M. Baccouch, *An iterative finite difference method for approximating the two-branched solution of Bratu's problem*, Appl. Numer. Math., *139* (2019), 62-76.
[12] M. Bilal, J. Iqbal, K. Shah, B. Abdalla, T. Abdeljawad, and I. Ullah, *Analytical solutions of the space–time fractional Kundu–Eckhaus equation by using modified extended direct algebraic method*, Partial Differ. Equ. Appl. Math., *11* (2024), 100832.
[13] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*, 2016.
[14] S. Chandrasekhar, *An Introduction to The Study of Stellar Structure*, *2* (1957).

[15] A. Chowdhury and C. Christov, *Memory effects for the heat conductivity of random suspensions of spheres*, Proc. R. Soc. A Math. Phys. Eng. Sci., *466* (2010), 3253-3273.
[16] D. D. Demir, A. Zeybek, A. Atangana, H. Bulut, Z. Hammouch, H. Baskonus, T. Mekkaoui, and F. B. Muhammad Belgacem, *The numerical solution of fractional Bratu-type differential equations*, ITM Web Conf., *13* (2017), 01008.
[17] A. Ebrahimzadeh and E. Hashemizadeh, *Optimal control of non-linear volterra integral equations with weakly singular Kernels based on Genocchi polynomials and collocation method*, J. Nonlinear Math. Phys., *30* (2023), 1758-1773.
[18] M. El-Gamel, N. Mohamed, and W. Adel, *Genocchi collocation method for accurate solution of nonlinear fractional differential equations with error analysis*, Math. Model. Numer. Simul. Appl., *3* (2023), 351-375.
[19] M. El-Gamel, N. Mohamed, and W. Adel, *Numerical study of a nonlinear high order boundary value problems using Genocchi collocation technique*, Appl. Comput. Math., *8* (2022), 143.
[20] M. A. El-Hady, M. El-Gamel, M. Izadi, and A. El-Shenawy, *Numerical computation of nonlinear boundary value problems in heat transfer and chemical engineering: a robust Lucas–Fibonacci series approach*, J. Taibah Univ. Sci., *19* (2025).
[21] A. El-Shenawy, M. El-Gamel, and D. Reda, *Troesch's problem: A numerical study with cubic trigonometric B-spline method*, Partial Differ. Equ. Appl. Math., *10* (2024), 100694.
[22] R. Garrappa, E. Kaslik, and M. Popolizio, *Evaluation of fractional integrals and derivatives of elementary functions: overview and tutorial*, Math., *7* (2019), 407.
[23] D. Gidaspow and B. S. Baker, *A model for discharge of storage batteries*, J. Electrochem. Soc., *120* (1973), 1005.
[24] W. G. Glöckle and T. F. Nonnenmacher, *A fractional calculus approach to self-similar protein dynamics*, Biophys. J., *68* (1995), 46-53.
[25] E. Hairer, G. Wanner, and O. Solving, *II: Stiff and Differential-Algebraic Problems*, 1991.
[26] J. H. He, *Approximate analytical solution for seepage flow with fractional derivatives in porous media*, Comput. Methods Appl. Mech. Eng., *167* (1998), 57-68.
[27] M. Izadi, K. J. Ansari, and H. M. Srivastava, *A highly accurate and efficient Genocchi-based spectral technique applied to singular fractional order boundary value problems*, Math. Methods Appl. Sci., *48* (2024), 905-925.
[28] M. Izadi and D. Baleanu, *An effective QLM-based Legendre matrix algorithm to solve the coupled system of fractional-order Lane-Emden equations*, Appl. Numer. Math., *201* (2024), 608-627.
[29] M. Izadi and H. M. Srivastava, *Applications of modified Bessel polynomials to solve a nonlinear chaotic fractional-order system in the financial market: Domain-splitting collocation techniques*, Comput., *11* (2023), 130.
[30] M. Izadi and H. M. Srivastava, *Generalized Bessel Quasilinearization technique applied to Bratu and Lane–Emden type equations of arbitrary order*, Fractals Fract., *5* (2021), 179.
[31] M. Izadi, Ş. Yüzbaşi, and S. Noeiaghdam, *Approximating solutions of non-Linear Troesch's problem via an efficient Quasi-Linearization Bessel approach*, Math., *9* (2021), 1841.
[32] J. Jacobsen and K. Schmitt, *The Liouville–Bratu–Gelfand problem for radial operators*, J. Differ. Equ., *184* (2002), 283-298.
[33] H. Jafari, B. F. Malidareh, and V. R. Hosseini, *Collocation discrete least squares meshless method for solving nonlinear multi-term time fractional differential equations*, Eng. Anal. Bound. Elem., *158* (2024), 107-120.
[34] E. Keshavarz, Y. Ordokhani, and M. Razzaghi, *The Taylor wavelets method for solving the initial and boundary value problems of Bratu-type equations*, Appl. Numer. Math., *128* (2018), 205-216.
[35] M. Khader, K. M. Saad, Z. Hammouch, and D. Baleanu, *A spectral collocation method for solving fractional KdV and KdV-Burgers equations with non-singular kernel derivatives*, Appl. Numer. Math., *161* (2021), 137-146.
[36] A. Khalouta and A. Kadem, *Solution of the fractional Bratu-type equation via fractional residual power series method*, Tatra Mt. Math. Publ., *76* (2020), 127-142.
[37] S. Khuri, I. Louhichi, and A. Sayfy, *An approach for the approximate solution of the fractional Troesch's problem*, Int. Conf. Fract. Differ. Appl. (ICFDA), *2023* (2023), 1-4.
[38] A. A. Kilbas, H. M. Srivastava, and J. J. Trujillo, *Theory and Applications of Fractional Differential Equations*, North-Holland Math. Stud., *204* (2006).

[39] S. Kumbinarasaiah, G. Manohara, and G. Hariharan, *Bernoulli wavelets functional matrix technique for a system of nonlinear singular Lane Emden equations*, Math. Comput. Simul., *204* (2023), 133-165.
[40] M. Lakestani and R. Tuntaş, *Efficient solution for multi-delay fractional optimal control problems via cubic B-splines*, Optim. Control Appl. Methods, 2025.
[41] M. Lotfi, *Numerical solution of different population balance models using operational method based on Genocchi polynomials*, Comput. Methods Differ. Equ., *13* (2024), 659–675.
[42] B. Maayah, A. Moussaoui, S. Bushnaq, and O. Abu Arqub, *The multistep Laplace optimized decomposition method for solving fractional-order coronavirus disease model (COVID-19) via the Caputo fractional approach*, Demonstr. Math., *55* (2022), 963-977.
[43] F. Mainardi and R. Gorenflo, *Time-fractional derivatives in relaxation processes: a tutorial survey*, arXiv preprint arXiv, 2008.
[44] Z. Masood, K. Majeed, R. Samar, and M. A. Z. Raja, *Design of Mexican Hat Wavelet neural networks for solving Bratu type nonlinear systems*, Neurocomput., *221* (2017), 1-14.
[45] M. A. Moghaddam, Y. E. Tabriz, and M. Lakestani, *Solving fractional optimal control problems using Genocchi polynomials*, Comput. Methods Differ. Equ., *9* (2021), 79-93.
[46] S. A. Mohamed, *A fractional differential quadrature method for fractional differential equations and fractional eigenvalue problems*, Math. Methods Appl. Sci., *47* (2020), 10995-11018.
[47] M. D. Ortigueira, *Fractional Calculus for Scientists and Engineers*, *84* (2011).
[48] G. ÖZALTUN and S. GÜMGÜM, *Numerical solutions of Troesch and Duffing equations by Taylor wavelets*, Hacet. J. Math. Stat., *52* (2023), 292-302.
[49] H. Ozden, Y. Simsek, and H. Srivastava, *A unified presentation of the generating functions of the generalized Bernoulli, Euler and Genocchi polynomials*, Comput. Math. Appl., *60* (2010), 2779-2787.
[50] P. Rahimkhani, Y. Ordokhani, and P. M. Lima, *Numerical solution of stochastic fractional integro-differential/Itô–Volterra integral equations via fractional Genocchi wavelets*, Comput. Methods Differ. Equ., 2025.
[51] M. A. Z. Raja, R. Samar, E. S. Alaidarous, and E. Shivanian, *Bio-inspired computing platform for reliable solution of Bratu-type equations arising in the modeling of electrically conducting solids*, Appl. Math. Model., *40* (2016), 5964-5977.
[52] O. W. Richardson, *The Emission of Electricity from Hot Bodies*, 1921.
[53] R. Saadeh, A. Burqan, and A. El-Ajou, *Reliable solutions to fractional Lane-Emden equations via Laplace transform and residual error function*, Alex. Eng. J., *61* (2022), 10551-10562.
[54] S. M. Sayed, A. S. Mohamed, E. M. Abo-Eldahab, and Y. H. Youssri, *A compact combination of second-kind Chebyshev polynomials for Robin boundary value problems and Bratu-type equations*, J. Umm Al-Qura Univ. Appl. Sci., *2024* (2024), 1-18.
[55] M. Shqair, A. El-Ajou, and M. Nairat, *Analytical solution for multi energy groups of neutron diffusion equations by a residual power series method*, Math., *7* (2019), 633.
[56] G. Simmons, *Differential Equations with Applications and Historical Notes*, 1991.
[57] H. Singh, F. Akhavan Ghassabzadeh, E. Tohidi, and C. Cattani, *Legendre spectral method for the fractional Bratu problem*, Math. Methods Appl. Sci., *43* (2020), 5941-5952.
[58] Swati, M. Singh, and K. Singh, *An efficient technique based on higher order Haar wavelet method for Lane–Emden equations*, Math. Comput. Simul., *206* (2023), 21-39.
[59] A. M. Vargas, *Finite difference method for solving fractional differential equations at irregular meshes*, Math. Comput. Simul., *193* (2022), 204-216.
[60] E. S. Weibel, *On the Confinement of a Plasma by Magnetostatic Fields*, Phys. Fluids., *2* (1959), 52-56.
[61] Y. H. Youssri and A. G. Atta, *Spectral collocation approach via normalized shifted Jacobi polynomials for the nonlinear Lane-Emden equation with fractal-fractional derivative*, Fractal Fract., *7* (2023), 133.
[62] F. Zhou, X. Xu, *Numerical solutions for the linear and nonlinear singular boundary value problems using Laguerre wavelets*, Adv. Differ. Equ., *2016* (2016), 1687-1847.