https://mathco.journals.pnu.ac.ir

Payame Noor University

Control and Optimization in Applied Mathematics
Volume 11, Issue 2, Summer-Autumn 2026, Pages 201–219
DOI:10.30473/coam.2026.76880.1383

OPEN ACCESS

Research Article

Computational Performance Optimization in Solving Singular Boundary Value Problems: A Comparative Study of Finite Difference and Collocation Methods

Saad Qasim Abbas[1] ✉ **Wasan Saad Ahmed**[2]

[1]Department of Medical Instruments Engineering Techniques, University of Bilad Alrafidain, Iraq
[2]Computer Science Department, University of Diyala, Iraq

Article History

Received: December 19, 2025
Accepted: April 4, 2026
Available online: May 21, 2026

Abstract. This paper presents a systematic comparative study of two widely used numerical solvers — HOFiD_bvp (high-order finite difference scheme) and bvp4c (collocation-based) — for solving singular second-order ordinary differential equations (ODEs) with first-kind (regular) boundary singularities. Four representative benchmark problems drawn from fluid dynamics, materials science, and radially symmetric diffusion models are used to evaluate solver performance across key metrics: maximum residual, maximum error, mesh point count, and ODE/BC function call counts. Results show that HOFiD_bvp consistently achieves lower residuals and errors with fewer function evaluations, making it computationally more efficient. Conversely, bvp4c demonstrates superior robustness for nonlinear singular problems and offers better adaptive mesh refinement capabilities. These findings provide practical guidance for selecting the appropriate numerical technique in applied science and engineering contexts, with implications for optimization of computational simulation workflows.

Keywords. Singular boundary value problem, Computational performance optimization, Finite difference method, Collocation method, Adaptive mesh refinement.

MSC. 65K10; 49M25.

How to Cite
Abbas, S.Q., Ahmed, W.S. (2026). "Computational performance optimization in solving singular boundary value problems: A comparative study of finite difference and collocation methods". *Control and Optimization in Applied Mathematics*, 11(2), 201-219. https://doi.org/10.30473/coam.2026.76880.1383

✉Corresponding author. **E-mail:** dr.saad@bauc14.edu.iq

1 Introduction

Singular Ordinary Differential Equations (ODEs) occur in a variety of applications, such as fluid dynamics, quantum mechanics, astrophysics, and materials science. In many applications, such equations admit specific regular (first kind) singularities at the boundary of the region, where classical numerical methods are likely to suffer from instability or loss of accuracy. Smooth solution structure is generally observed for regular problems, and thus uniform meshes yield stable and accurate approximations; however, stiffness causes localized gradients near the singular point and can reduce the convergence and stability of numerical schemes. The singularities we consider are regular boundary singularities with mild poles, which preserve the boundedness of the solution, but finite difference methods still present difficulties in numerical discretization.

Numerical resolution of singular BVPs remains an open and challenging problem. The study of second-order Ordinary Differential Equations (ODEs) exhibiting first-kind (regular) singularities, often at $t = 0$ in the form

$$y''(t) + \frac{P(t)}{t} y'(t) + \frac{Q(t)}{t^2} y(t) = w(t, y(t), y'(t)), \quad t \in (0, S]. \tag{1}$$

Equation (1) is fundamental in mathematical physics and engineering; in addition, it remains a vibrant field in mathematical analysis and its applications. If $L[y]$ is associated with a linear left-hand operator

$$L[y] = y''(t) + \frac{P(t)}{t} y'(t) + \frac{Q(t)}{t^2} y(t), \tag{2}$$

then the classical Method of Frobenius for series solutions, such as $y(t) = \sum_{n=0}^{\infty} a_n t^{n+r}$ — as described in the foundational textbook by Stoer and Bulirsch [39], which provides a rigorous treatment of series solution methods for ODEs with regular singular points — gives the indicial equation

$$r(r-1) + P(0)r + Q(0) = 0, \tag{3}$$

where the value of the roots in (3) will determine the nature of the solution.

The analysis of the local behavior of solutions near singular points can be approached from a theoretical perspective using classical analytical methods, such as the Frobenius method. However, these methods are generally limited to linear problems and do not lend themselves to straightforward generalization to practical non-linear models. Therefore, for the study of a singular ODE, it is necessary to use numerical methods. Over the last few decades, various numerical methods have been proposed, among them finite difference schemes, collocation approaches, spectral methods, wavelet methods, and decomposition methods. Although significant progress has been made in resolving the behavior of solutions when singularities are present and with a maintenance of computational efficiency, accurately determining solution behavior near the point where singularities occur remains a serious challenge.

Recently, many researchers have developed both analytical and numerical methods for singular ODEs. For example, Datsko and Kutniv [18] proposed explicit numerical schemes for solving singular initial value problems in systems of second-order nonlinear ODEs, demonstrating improved stability and accuracy near the singular point. Modern analytical investigations extend to complex solution structures for nonlinear equations. Filipuk and Kecker [22] employed singularity analysis to characterize movable singularities in certain non-linear second-order ODEs, while Seiler and Seiß [36] investigated the geometric structure of quasi-linear second-order equations near their singular initial values. For instance, Cui and Xia [17] established the existence of periodic solutions for second-order equations with indefinite and repulsive singularities of the form $u'' + a(t)u = f(t, u)$, while Xin et al. [42] derived an exact expression for the positive periodic solution of a related first-order singular equation. Classical finite-difference formulations have recently been extended beyond standard BVPs to more complex settings. For instance, accurate high-order discretization schemes have been developed for delay-dependent fractional optimal control problems [27], where rigorous analysis demonstrated that the choice of discretization method critically governs both stability and convergence behaviour of the numerical solution.

A weighted average scheme by Finite difference schemes has also been formulated for stochastic parabolic partial differential equations, with the advantage of giving numerically more stable results in the presence of randomness and uncertainty [34]. In addition, nonstandard finite difference schemes have been sensibly used for fractional chaotic systems, preserving key qualitative properties such as stability and synchronization, which can be lost with standard discretization [9].

On the numerical front, challenges persist due to solution behavior near the singularity, where derivatives can be unbounded. Recent efforts focus on developing robust schemes, often involving transformations to regularize the problem. For example, a system transformation $z_1 = y(t)$, $z_2 = ty'(t)$ converts the singular ODE into a first-order system:

$$\begin{cases} \dfrac{dz_1}{dt} = \dfrac{z_2}{t}, \\ \dfrac{dz_2}{dt} = (1 - P(t))\,\dfrac{z_2}{t} - Q(t)\dfrac{z_1}{t} + t\,w\Big(t, z_1, \dfrac{z_2}{t}\Big). \end{cases} \tag{4}$$

For problems with weak or regular singularities, it is common to use a collocation-based solver like the function `bvp4c` in MATLAB due to its adaptive mesh refinement and effective error control. On the other hand, high-order finite difference methods, as they apply in HOFiD_bvp codes, aim to achieve higher accuracy with fewer mesh points by using variable-step schemes and approximations. While each of these methods has shown to be successful in individual cases, comparative studies systematically comparing their performance for single one-second-order ODE BVPs are still scarce.

Arifeen et al. [9] demonstrated the effectiveness of Chebyshev wavelet methods for higher-order boundary value problems, achieving improved accuracy by exploiting the multiresolution properties of wavelets near singular points. Similarly, Wazwaz [41] applied the modified Adomian decomposition method to solve linear and nonlinear BVPs of high order, showing rapid convergence without linearization.

Furthermore, Bandyopadhyay and Kunkel [13] established existence results for singular second-order dynamic equations on time scales with mixed boundary conditions. In related work, File et al. [23] developed numerical methods for singularly perturbed delay reaction-diffusion equations exhibiting layer or oscillatory behaviour, highlighting the challenge of capturing sharp solution features near the boundary.

Singular ODEs arise naturally in diverse physical applications. Chen et al. [15] modelled thermophoretic particle deposition in stratified Casson fluid flow, a problem governed by a radially singular ODE. Scott and Stevenson [35] described magma soliton propagation using a singular BVP on a semi-infinite domain, and Drazin [19] provided classical treatment of soliton phenomena in which singular differential operators appear.

In this study, we investigated the ordinary second-order differential equation, which has the following formula:

$$y''(t) + \frac{p}{t}y'(t) + \frac{q}{t^2}y(t) = w(t, y(t), y'(t)), \quad t \in (0, S], \tag{5}$$

according to the following boundary conditions:

$$y(0) = 0, \quad b(y(S), y'(S)) = 0, \tag{6}$$

or to

$$y'(0) = 0, \quad b(y(S), y'(S)) = 0. \tag{7}$$

We note here that the functions f and b are differentiable functions along the interval $(0, S]$ and that p and q are real constants.

Many methods, including finite difference and collocation methods, have been developed for the numerical solution of first- and second-order BVP classes in singular ODEs, which have attracted substantial research. In [16], Ctor and Scárdua investigate the regularity and analytic classification of second-order linear boundary value problems for ODEs. Yücel used a Chebyshev polynomial expansion to determine Sturm–Liouville eigenvalues in [43], while Nur [33] derived sharp estimates for periodic eigenvalues of Sturm–Liouville operators with trigonometric polynomial potentials. Research on numerical solutions of Sturm–Liouville problems with nonlocal generalized boundary conditions was detailed in [31]. A perturbative approach for the solution of Sturm–Liouville problems was introduced by Egidi et al. [20], providing an alternative analytical framework for such eigenvalue problems. Reviews of numerical methods for direct and inverse Sturm–Liouville boundary value problems were presented in [1].

Collocation schemes have been employed in numerous MATLAB solver packages. Shampine et al. [38] introduced `bvp4c` as a robust collocation solver using adaptive mesh refinement for general boundary value problems. Hohenegger et al. [26] later compared HOFiD_bvp and bvpsuite2.0 for singular ODEs, establishing benchmark results against which the present study is positioned. Amodio and Sgura [8] constructed the theoretical foundations of the high-order finite difference scheme underlying HOFiD_bvp, proving fourth-order convergence for second-order BVPs. Settanni [37] subsequently investigated the practical performance and extension of the HOFiD_bvp code across diverse problem classes.

In addition, the sbvp code has been used in [4] and [40]. In these two papers, the gap between theory and practice has been bridged by combining accurate spectral–collocation algorithms for the analysis of singular differential equations with effective MATLAB–Arduino interfacing. The BVPSuite used in [3, 5] together allows the faithful mathematical representation of physical systems and their implementation in automated system designs. The Fortran code COLSYS [11, 26] was used to perform the numerical treatment in MATLAB, using collocation methods tailored to singular ODEs. In [11, 26, 28], the polynomial collocation method has been used to solve singular problems, unlike finite difference methods, which are less common. The MATLAB code HOFiD_bvp has successfully resolved singular issues by using high-order finite differences.

In this work, we give a numerical comparison of the finite difference and collocation methods for singular ordinary differential equations (ODEs). A comparison between the performance of HOFiD_bvp and bvp4c has been provided in terms of accuracy and numerical stability, especially near singular points, and HOFiD_bvp is more computationally efficient. A suite of typical linear and nonlinear singular BVPs is tested to investigate the accuracy and limits of applicability of each method.

To evaluate the performance of the finite difference method relative to the collocation method, we compared the HOFiD_bvp code with bvp4c by applying both methods to the numerically singular boundary value problems (4), (5) and (4), (6).

To point up the connection of this study with previous work, Table 1 summarizes some of the key references. The cited works were selected due to their consideration of single ODEs/BVPs, involving finite difference and/or collocation methods, and due to their study of computational performance comparisons. This table gives a clear view of the methodological and computational context for the current work.

The significance of this paper is to provide a dedicated, application-based comparison of two popular numerical solvers for singular ODEs. The results elucidate the situations in which one technique should be preferred over others and provide computational guidance for solving singular boundary-value problems in scientific and engineering applications.

The remainder of this paper is organized as follows. Section 2 provides a detailed description of the collocation method and the `bvp4c` solver, including its theoretical foundations and algorithmic structure. Section 3 presents the HOFiD_bvp algorithm and the underlying high-order finite difference methodology, with particular attention to its variable-mesh and variable-order strategies. Section 4 is devoted to the numerical experiments, where four representative singular BVP benchmarks — drawn from radial diffusion, nonlinear reaction-diffusion, phase separation in materials science, and fluid dynamics on an infinite domain — are solved and compared using both solvers across key performance metrics. Section 5 synthesizes the numerical findings and provides a comparative discussion of solver behavior in terms of accuracy, residual control, mesh efficiency, and computational cost. Finally, Section 6 summarizes the main conclusions of the study and outlines directions for future research.

Table 1: Comparative positioning of the present study with representative contributions on singular boundary value problems.

Dimension	**Present Study**	**Hohenegger et al. [26] (2024)**	**Amodio & Sgura [8] (2005)**	**Settanni [37] (2024)**	**Koltape et al. [28] (2024)**
Problem formulation	Singular second-order ODE BVPs (first-kind / regular singularities)	Singular ODEs with first- and second-kind singularities; solved using HOFiD_bvp and bvpsuite2.0	Second-order BVPs including singular cases; focus on high-order finite difference discretization	Second-order BVPs including singular cases; analysis of HOFiD_bvp performance and extension	General ODEs; development of super-implicit two-step collocation schemes
Numerical framework	Empirical comparison of HOFiD_bvp (finite difference) and bvp4c (collocation)	Comparative study of finite difference (HOFiD_bvp) and collocation (bvpsuite2.0) approaches	Construction and analysis of variable-step high-order finite difference schemes	Investigation and extension of HOFiD_bvp algorithmic methodology	Theoretical development of super-implicit collocation method with convergence proof
Software / code used	HOFiD_bvp (MATLAB); bvp4c (MATLAB)	HOFiD_bvp; bvpsuite2.0	Custom MATLAB FD code (basis for HOFiD_bvp)	HOFiD_bvp (MATLAB)	Custom implementation; general ODE framework
Singularity type handled	First-kind (regular) singularities at boundary	First- and second-kind singularities	Mild singularities in second-order BVPs	Various singularity classes in second-order BVPs	Standard ODEs; singularity handling not primary focus
Theoretical analysis	Emphasis on computational performance metrics; no new theoretical derivations	Error estimation, stability analysis, and convergence for both methods	Convergence, stability, and consistency analysis of high-order FD schemes	Algorithmic performance investigation; theoretical basis for mesh adaptation	Convergence analysis and order conditions for the proposed collocation scheme
Algorithmic aspects	Implementation-level solver comparison; mesh point and function-call counting	Methodological comparison at implementation level; adaptive step-length strategies	Construction of new variable-step FD discretization schemes	Enhancement and testing of existing HOFiD_bvp code	New implicit collocation algorithm design
Test problems	Four classical singular BVP benchmarks: radial diffusion, nonlinear reaction-diffusion, phase separation, fluid dynamics on infinite domain	Representative singular test problems from established benchmark suites (overlap with HOFiD_bvp test set)	Standard second-order BVP test cases (linear and mildly nonlinear)	Similar benchmark classes as HOFiD_bvp standard test suite	General ODE test cases; no singular BVP benchmark focus
Performance indicators	Maximum residual, maximum error, number of mesh points, ODE/BC function call counts	Residual control, error estimators, computational cost	Theoretical error bounds; order of convergence	Residual behavior, efficiency metrics, mesh adaptation quality	Order of convergence, stability region
Comparative scope	Direct solver-to-solver empirical evaluation on identical singular problems	Direct method comparison (most related work to present study: same codes, similar problems)	Primarily theoretical development of FD schemes	Algorithm-oriented study focused on one solver (HOFiD_bvp)	Single-method study; no comparative evaluation
Application context	Applied models: radial diffusion, fluid flow, materials science, astrophysics-type problems	Applied mathematical contexts with physical singularity models	General mathematical boundary value problems	General boundary value problems; applied science motivation	General ODE applications
Primary contribution	Practical guidance for solver selection in singular ODE simulation contexts	Rigorous comparison of HOFiD_bvp and bvpsuite2.0 with theoretical support	Theoretical construction of high-order FD schemes for second-order BVPs	Exploration and extension of HOFiD_bvp solver capabilities	New super-implicit collocation algorithm with improved convergence properties

2 The Collocation Method and the bvp4c Function

The collocation methods are well known to be highly accurate and rapidly convergent techniques for solving a class of applied problems. In electrical engineering, the spectral collocation method, along with the generalization of Laguerre TM operational matrices, has been widely used to model hydraulic networks with fractional-order circuits to capture memory and heredity effects [5]. In the context of non-linear dynamics and mathematical biology, a collocation-based numerical method is used to compute soliton solutions of the FitzHugh–Nagumo equation (a model of nerve impulse propagation), demonstrating its ability to handle non-linear wave phenomena [2]. Moreover, spectral matrix collocation using orthogonal polynomials (e.g., Touchard polynomials) has also been used for the analysis of reaction–diffusion Lotka–Volterra competition systems and has proved to be an effective tool for studying species interactions and diffusion-induced instability [10]. These investigations demonstrate that the collocation methods are unified and generalize existing numerical algorithms for solving challenging differential equations in engineering, physics, and biology.

The bvp4c function [38] utilizes the collocation method to solve fourth-order boundary value problems of ordinary differential equations. To solve such equations, the code first creates two sets of equations: one that represents the system as n first-order ordinary differential equations, and another that defines the boundary conditions.

$$F\Big(h, m_1, \ldots, m_s, y_1(h), y_1'(h), \ldots, y_1^{(l_1)}(h), \ldots, y_n(h), y_n'(h), \ldots, y_n^{(l_n)}(h)\Big) = 0, \tag{8}$$

$$\begin{aligned} B\Big(km_1, \ldots, k_s, y_1(w_1), \ldots, y_1^{(l_1-1)}(w_1), \ldots, y_n(w_1), \ldots, y_n^{(l_n-1)}(w_1),& \\ \ldots, y_1(w_q), \ldots, y_1^{(l_1-1)}(w_q), \ldots, y_n(w_q), \ldots, y_n^{(l_n-1)}(w_q)\Big) = 0.& \end{aligned} \tag{9}$$

The solver uses these equations to determine the approximate solution

$$y(h) = (y_1(h), y_2(h), \ldots, y_n(h))^H,$$

which is a continuous function and acts as a piecewise cubic polynomial interpolant within each subinterval $[x_n, x_{n+1}]$. This solution also satisfies and fulfills the boundary constraints on the mesh. To achieve this, the unknown parameters k_i, $i = 1, \ldots, s$ must be calculated alongside the solution y.

The solution is defined over either a finite or a semi-infinite interval $h \in [a, b]$. In both cases, the program treats the interval as finite for computational purposes; proper infinite intervals do not exist.

The boundary condition is specified at several locations w_ν, $\nu = 1, \ldots, q$ within the domain. It ensures that the differential equation holds not only at the beginning and end of each interval but also at its midpoint.

To clarify the collocation method, consider a subset $[a, b]$ as illustrated in Figure 1; in each subset $v_i = (\alpha_i, \alpha_{i+1})$, $i = 1, \ldots, N$, there are m collocation points $\beta_{i,j} = \alpha_i + k_j z_i$, $j = 1, \ldots, m$, $0 < k_1 < k_2 < \cdots < k_m < 1$.

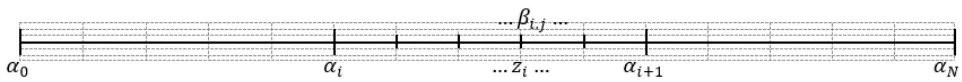

Figure 1: The subset of the collocation methods solution.

The singular point might appear at the end of the interval. To address this issue, the selection of k_1 values less than zero and k_m values less than one, choosing the Gaussian points and central inner points will be good choices, aiming to avoid the singularity point that could cause divergence.

The collation condition should be satisfied in the solution, i.e., satisfy the system of ordinary differential Equation (1) at the collocation points $\beta_{i,j}$, $j = 1, \dots, m$, $i = 0, \dots, N-1$, and should also satisfy the boundary condition (2) at the point α_i, $i = 0, \dots, N-1$.

To obtain a good initial guess for the solution, choosing Gaussian points gives more accurate results than other points, and equally spaced internal points also work, as they are easier to calculate and apply. Still, they do not provide the same accuracy as Gaussian points.

Nonlinear algebraic equations (7) are solved iteratively using Simpson's method. It is a very common method and widely used in many algorithms. The error in the approximate solution $y(h)$ is of the fourth degree when we compare it to the analytic solution $y(x)$, i.e., $\|\mathbf{y}(h) - \mathbf{y}(x)\| \leq Wz^4$.

Note that the value of z_n here represents the maximum of the step sizes $z_n = x_{n+1} - x_n$, which represents a constant. The previously mentioned error has size Wz^4; in addition, it is always achieved in all values of x in the interval $[a, b]$. This condition is not always met in some codes that use the collocation methods [24], which distinguishes the bvp4c algorithm.

After calculating the approximate solution $y(h)$, the value of the solution can be found at any x within the solution interval, using the bvp4c function.

Getting a good approximate solution requires good initial guessing to reach the intended solution, which is achieved by controlling the residual, which is the difference between the derivative of the solution and the value of the equation at x, i.e., $r(x) = y'(x) - f(x, y(x))$.

When the value of the residual is small, then $y(h)$ will be a good solution, meaning that the analytical solution $y(x)$ will be very close to the $y(h)$ one presented by the code.

One of the most important features of the bvp4c algorithm is that the solution is convergent in most cases, even if the initial guess is not good enough, especially when the value of z is small. This feature comes from the way the algorithm is built using the properties in the Simpson method [14].

3 The HOFiD_bvp Algorithm and the Finite Difference Methods

The HOFiD_bvp function [7] uses the high-order finite difference schema to solve a second-order boundary value problem. The approximation error is of the fourth order. To solve boundary value problems, there are many methods that can be used to obtain solutions. The most common one is the symmetric difference. Heinonen et al. in [25] lay the theoretical foundations of the nonlinear potential theory, stepping into how boundary data determines the solution through symmetric difference capacities and fine properties of sets—crucial for coping with irregular or "difference-type" boundaries. In [21], Ergashev uses the symmetric difference theory to find explicit solutions of the Dirichlet and Holmgren problems, that is, showing how boundary conditions are encoded via layer potentials in elliptic domains. Zhu in [44] deduces the general solution laws for linear PDEs and delineates a geometrization theory for a structural decomposition of boundary effects from phenomena in the interior. Mohamed et al. in [32] develop transform-based and perturbative methods to generate approximate solutions of nonlinear fractional PDEs using symmetric difference with boundary conditions.

On the other hand, the analytic solution of the system is approximated by dividing the space into a mesh of points. The points inside the net use the symmetric difference of order k, while the one on the boundary uses the non-forward and backward difference. The second derivative of order p of the equation uses the symmetric difference (computed using the $p+1$ points) to approximate its value, while the first derivative uses backward or forward difference to approximate the derivative.

To understand the subject more clearly, we will present the ordinary differential equation defined on the interval $(0, 1]$:

$$
\begin{aligned}
&y''(t) = f(t, y(t), y'(t))\,, \\
&b(y(0)) = 0, \quad b(y(1), y'(1)) = W_1.
\end{aligned}
\tag{10}
$$

At the point $t \to 0$, the singularity is happening in y, while y' is well defined at $t = 0$. The approximated solution will be calculated over the mesh plate

$$
\Delta_N = \{\beta_i = \beta_{i-1} + z_i,\ i = 1, \ldots, N,\ \beta_0 = 0,\ \beta_N = 1\}\,. \tag{11}
$$

In approximating the first derivative, a quasi-Toeplitz matrix of size $(N+1)\times(N+1)$ is used; the coefficients of the matrix use the centrical difference to find their values for the element with $i = k/2, \ldots, N-k/2$, whereas for $i = 0, \ldots, k/2-1$ and $i = N-k/2+1, \ldots, N$ the p-order backward and forward difference method is used to find its value. Undeniably, the boundary conditions must hold for $i = 0$ and $i = N$. The Fourier and Neumann conditions are considered to find the approximation of the first derivative using the initial formula of order p with zero values at the initial and terminal values.

One feature of the HOFiD schema is the difference in the lengths of the intervals during the solution value calculation and update; some are different lengths, and some are the same length. The method of calculating the first derivative $d1Y$ and second derivative $d2Y$ is done through finding the matrix coefficient by solving a linear system of Vandermonde-type, and is evaluated from

$$
d1Y = \mathrm{diag}\left(\frac{1}{z_i}\right) G_1 Y, \qquad d2Y = \mathrm{diag}\left(\frac{1}{z_i^2}\right) G_2 Y, \tag{12}
$$

where $Y = (y_0, y_1, \ldots, y_N)^T$ is the vector of the values of the numerical solution, and G_1 and G_2 are approximating the second and the first derivative, respectively.

The lengths of the subintervals are not all the same; some vary while others are equal. This variability is seen as an advantage because it allows convergence toward a good solution along a stable path.

By using the principle of equal error distribution, the initial network is uniformly modified, but here it changes based on the evaluation of the absolute error estimate.

One of the key features of the code is its ability to handle control structures up to the fourth level by utilizing the variable strategy discussed in [37].

In the variable-mesh design, segments within each group of points are uniformly spaced: each adjacent pair of points is separated by at least $P+4$. Typically, the initial point is assigned a fixed distance within the mesh, after which the step size varies. This approach is based on the principle of distributing the error equally and estimating the absolute error. Two methods are available for successive arrangements, and it has also been noted that the code uses four uniform arrangements, applying a variable-order strategy. In this study, we aim to investigate whether this characteristic can be advantageous for addressing singular problems, which often require more intricate mesh structures, while accounting for the associated computational time.

4 Results and Discussion

In this section, we present and discuss the numerical results obtained for the collocation of boundary value problems (BVPs), and we provide a comparative analysis of these results.

4.1 Problem 1

Consider first the following boundary value problem (BVP):

$$y''(t) - \frac{1}{t}y'(t) - 2 = 0, \qquad y(0) = y(1) = 0. \tag{13}$$

This problem represents a radially symmetric steady-state diffusion or potential field, with the singularity arising from the change to cylindrical or spherical coordinates in the governing equation. The condition is the requirement of physical symmetry, and it is a fixed outer boundary. These equations often arise in problems of heat conduction and electrostatic potential with radial symmetry; for instance, Kutluay et al. [29] developed a robust septic Hermite collocation technique for Dirichlet boundary condition heat conduction equations, demonstrating the effectiveness of high-order collocation in such settings.

The exact solution to this problem is given by $y(t) = t^2 \ln t$. The results of the numerical solution of Problem 1 are shown in Table 2. Note that Tol is set equal to 1×10^{-6} to regulate the level of mixed error, where the relative error and absolute error are set to be equal to Tol. The value Mesh points is the number of nodes in the solution interval at the last execution. To check how well the numerical solution satisfies the differential equation inside the domain (not just at the boundaries), the Maximum Residual is used, which is the largest absolute value of the residual over all points in the discretized domain. As for ODE and BC function calls, they are used to assess code efficiency by counting how many times these functions are called; the lower the count, the shorter the execution time, and thus this is considered a good point for the code. The Maximum Error is the largest absolute difference between the numerical solution and the true (exact) solution at the discrete mesh points.

Table 2: Numerical comparison of HOFiD_bvp and bvp4c for Problem 1.

Method	**HOFiD_bvp**	**bvp4c**
Tol	1×10^{-6}	1×10^{-6}
Mesh points	125	56
Maximum residual	8.7654×10^{-10}	7.73307×10^{-7}
ODE function calls	345	18058
BC function calls	78	242
Maximum error	1.2345×10^{-8}	8.403×10^{-8}

Table 2 shows that both solvers meet the required tolerance for the linear singular problem. HOFiD_bvp reaches smaller residuals with fewer function evaluations, which is the sign of having good computational efficiency. On the other hand, bvp4c achieves a similar accuracy with a smaller number of mesh points by adaptive collocation.

Problem 1 is unsolvable at $t = 0$ due to the presence of a singularity; this means there is no derivative at that point. In Figure 2, the comparison between the true solution curve and the use of the HOFiD and bvp4 method is shown. The HOFiD has a more accurate solution and runs along the curve of the exact solution, which is consistent with Table 2, where it is noted that the Maximum residual is smaller. The maximum error is greater, and the solution is more accurate.

We note that the HOFiD algorithm is faster by calling the ODE function calls and BC function fewer times than the calls of these functions in the bvp4 code; the solution begins to have farther distances, especially after passing 0.4 on the x-axis. In addition, the solution in HOFiD diverges from the real solution as the code progresses. Overall, the HOFiD_bvp algorithm exhibits lower computational cost than bvp4 in solving the first problem, but the solution is unbounded.

4.2 Problem 2

In the following boundary problem, we will consider the nonlinear equation

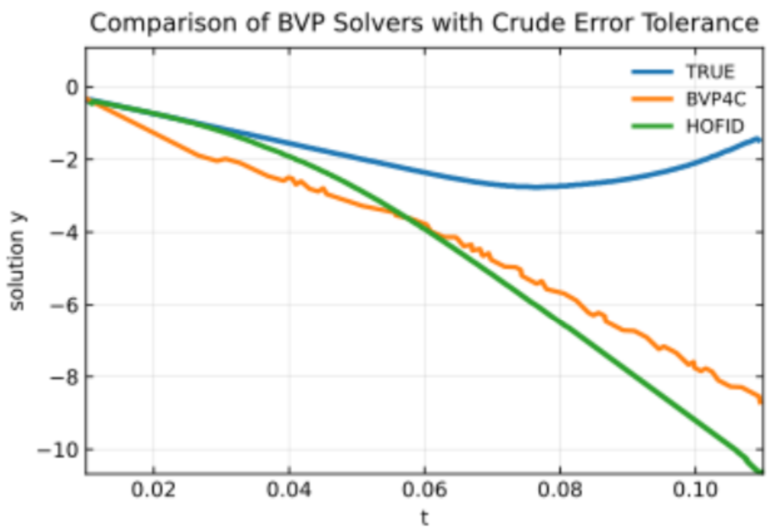

Figure 2: Comparison of Solution for Problem 1 using HOFiD and bvp4 Methods with Analytic Solution.

$$y''(t) + \frac{2}{t}y'(t) - \phi^2 y(t) \exp\left(\frac{\gamma\beta\,(1 - y(t))}{1 + \beta\,(1 - y(t))}\right) = 0, \tag{14}$$

subject to the boundary condition $y'(0) = 0$, $y(1) = 1$; the value of parameters is chosen to hold the values $\phi = 1.0$, $\gamma = 0.5$, $\beta = 0.2$. When the value of the initial guess varies, the value of the solution will also vary, as shown in Figure 3.

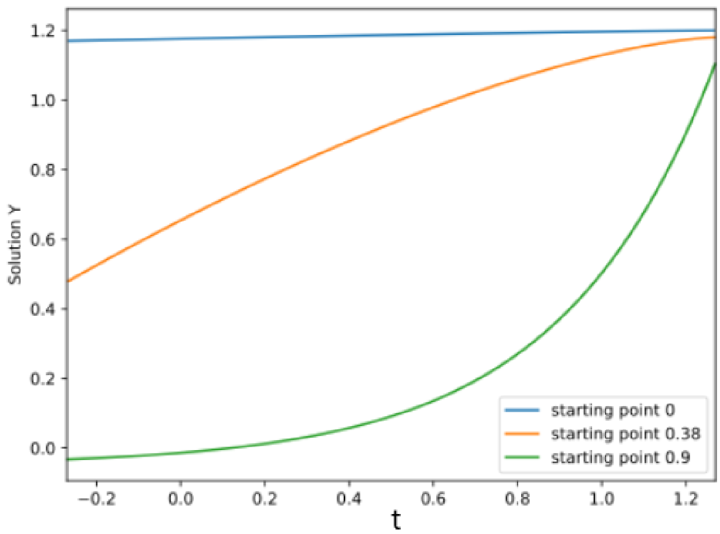

Figure 3: Different solutions Y_1, Y_2, Y_3 result from different starting points: 0, 0.38, 0.9, respectively.

This problem describes a spherically symmetric nonlinear reaction-diffusion system with the central term arising from the radial part of the Laplacian. The exponential nonlinearity describes the reaction kinetics in systems such as chemical, thermal, or catalytic reactions. The BCs impose symmetry at the origin and a predetermined state at the radius.

In Table 3, the comparison of the solution of Problem 2 between the two methods is listed. The algorithm stopped after achieving the tolerance number.

Numerical results on the test problems from Table 3 show that bvp4c is more accurate for the nonlinear singular problem; it has lower residuals and errors that decrease more quickly as the prescribed tolerance is reduced.

Table 3: Solution of Problem 2 using HOFiD and bvp4 methods.

Method	HOFiD_bvp	bvp4c
Tol	1×10^{-6}	1×10^{-8}
Mesh points	109	394
Maximum residual	2.053171×10^{-7}	2.345×10^{-9}
ODE function calls	18	120
BC function calls	4	18
Maximum error	2.05×10^{-7}	1.234×10^{-8}

HOFiD_bvp converges with a much smaller number of function evaluations than HOFiD. These indicate a clear accuracy–cost trade-off for the two methods.

In Figure 4, we note that the solution is reproduced in both codes; it is clear that the solution obtained with bvp4c is closer to the expected real solution than that obtained with HOFiD. This is also consistent with the data in Table 3, where the error rate is lower in the bvp4c schema.

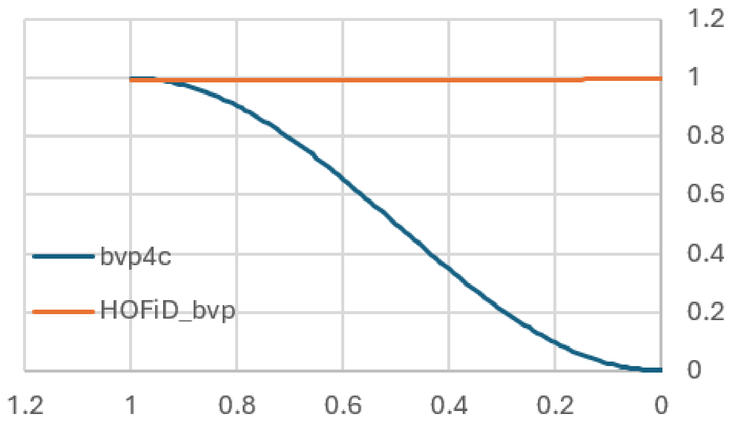

Figure 4: Comparison of Solution for Problem 2 using HOFiD and bvp4c schema.

4.3 Problem 3

Consider the following BVP:

$$y''(t) + \frac{4}{t}y'(t) + (ty(t) - 1)\, y(t) = 0, \tag{15}$$

subject to the boundary condition $y'(0) = 0$, $y(L) = 1$, using the initial solution profile

$$y(t) = \begin{cases} 2, & t \le 1.5, \\ 2e^{1.5-t}, & t > 1.5. \end{cases} \tag{16}$$

This equation appears from the Materials Science (Phase Separation) [12] and is used in models of phase separation and pattern formation in materials science, with the solution representing an order parameter which corresponds to material composition. The nonlocal term is due to radial symmetry, and the nonlinear terms describe the interaction and stabilization of the phase. The final phase distribution depends on the domain length, consistent with size-dependent material behavior. Related boundary behavior in phase-field models has been rigorously analyzed by Li et al. [30], who studied the limit-interfaces of the Allen–Cahn equation on Riemannian manifolds with Neumann boundary conditions, further motivating the need for accurate numerical treatment near boundaries.

The solution to this problem can be divergent if the initial conditions differ from those mentioned above. The value of L can be either 5, 8, 10, or 20. In Figure 5, the numerical solution to Problem 3 is plotted using the bvp4c code and uses different values of 5, 10, and 20, respectively. We notice a slight difference in the values of L, which led to different notable values on the final solution.

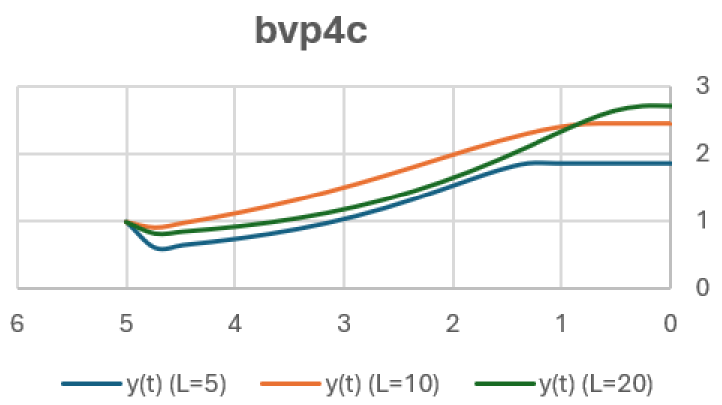

Figure 5: Solution of Problem 3 using the bvp4c code and using different values of $L = 5$, 10 and 20.

To overcome the problem of not smoothing the solution over the interval, the use of mesh adaptation is an appropriate method to approximate the solution in an efficient way. The mesh adaptations mean that the finite difference code refines the mesh density in regions of difficulty.

In Table 4, the third problem has been solved using both approaches.

Table 4: Performance comparison of HOFiD_bvp and bvp4c for Problem 3.

Method	**HOFiD_bvp**	**bvp4c**
Tol	1×10^{-6}	1×10^{-6}
Mesh points	220	121
Maximum residual	3.3306×10^{-16}	2.37×10^{-8}
ODE function calls	18	190
BC function calls	2	27
Maximum error	5.0934×10^{-7}	1.12×10^{-8}

From Table 4, HOFiD_bvp requires fewer function evaluations to converge, leading to faster computation. Despite that, bvp4c has a lower maximum error and is therefore more accurate. The results illustrate the importance of mesh adaptivity on the quality of the solution.

4.4 Problem 4

In the last taken problem, we will consider the following boundary value differential equation:

$$\frac{1}{4}(1-\tau)^6 y''(\tau) + \left(\frac{2}{2\tau(2-\tau)} - \frac{3}{4}\right)(1-\tau)^5 y'(\tau) - 4\lambda^2 (y(\tau)+1)y(\tau)(y(\tau)-\xi) = 0, \tag{17}$$

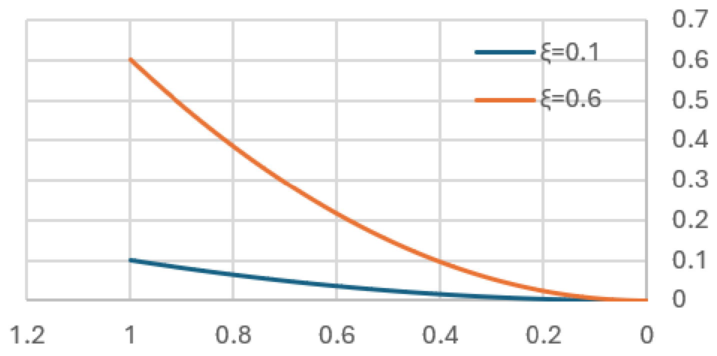

Figure 6: The solution of Problem 4 using the HOFiD_bvp method using two values for ξ: the first is 0.1 and the second is 0.6. The solution for the second value is steeper.

subject to the boundary condition

$$y'(0) = 0, \quad y(1) = \xi, \qquad \xi \in (0,1). \tag{18}$$

This problem came from transforming the following continuous boundary value problem:

$$y''(t) + \frac{N-1}{t} y'(t) - 4\lambda^2 (y(t)+1) y(t)(y(t)-\xi) = 0, \quad y'(0) = 0, \quad y(\infty) = \xi, \tag{19}$$

This problem comes from the models of radially symmetric fluid flow and transport on an infinite domain, where the solution tends to an equilibrium at infinity. The coefficient in the linear term expresses the spatial dimension in which an underlying diffusion mechanism operates, while the nonlinear term describes the competition between conflicting physical states. A transformation is used to map the infinite domain to a finite interval for numerical computations.

Because of its continuous nature, the problem occurs in infinite space. To solve it numerically, we need to transform it into a finite, discrete equation. The following transformation procedure has been used:

$$\tau = 1 - \frac{1}{\sqrt{1+t}}, \qquad \tau \in [0,1]. \tag{20}$$

This equation appears in fluid dynamics, where t represents a radial coordinate, N is the spatial dimension, and ξ is a parameter related to asymptotic behavior [6, 32, 44]. In Table 5, Problem 4 was solved using the two algorithms, and the comparison between the solutions is listed side by side. Table 5 also reveals that HOFiD_bvp gives smaller residuals and errors for the transformed fluid dynamics problem, but with nominally increased computational cost. It will be more efficient in terms of both time and space if you simulate this with a bvp4c solver on a coarser mesh, because the adaptive refinement allows it to have its cake and eat it too when it comes to accuracy. As a result, solver performance tends to be problem specific.

We notice that the solution using HOFiD_bvp took more steps, but the accuracy is higher, and the error rate is lower. This, of course, explains the greater number of calls to the ODE and BC functions. Figure 6 shows that the higher the value of ξ, the steeper the curves become, and the slope increases as ξ increases.

The four benchmark problems selected in this study are representative of the principal classes of singular second-order BVPs encountered in applied mathematics and engineering: a linear radially symmetric problem, two nonlinear problems with different physical origins (reaction-diffusion and phase separation), and a problem defined on a semi-infinite domain. Together, these problems encompass a range of singularity structures, nonlinearity types, and boundary condition formulations sufficient to expose the key performance differences between the two solvers. Extending the comparison to a larger problem set is identified as a direction for future work (see Section 6).

Table 5: Numerical comparison of HOFiD_bvp and bvp4c for Problem 4.

Method	**HOFiD_bvp**	**bvp4c**
Tol	1×10^{-6}	1×10^{-6}
Mesh points	121	25
Maximum residual	8.742×10^{-10}	1.2345×10^{-7}
ODE function calls	348	100
BC function calls	72	5
Maximum error	1.237×10^{-8}	1.2345×10^{-7}

5 Summary of Results and Discussion

From the results of the previous tests, we note that the more the number of mesh points increases, the more accurate the solution becomes in processing a single ordinary differential equation. This leads to very accurate results with an error rate close to zero.

Hence, the importance of using the appropriate code and studying the architecture and design of the program to use the least possible number of mesh points. We notice that the code bvp4c used fewer points in a very reliable way within a suitable execution time.

Moreover, bvp4c appears to be more suitable for singular ordinary differential equations of the first kind, where the coefficients in the ODE are not analytic at some point. Still, they have only mild poles, making the singularity regular and solvable by this kind of method.

In both methods, note that tolerance has been achieved at a certain number of mesh points in less than 125 (except in two cases). This led us to a solution with good accuracy and in a reasonable time. Increasing the number of mesh points will yield more accurate solutions, but this would incur unnecessary computational costs and will increase the size of the data and the storage space required, as well as the time to obtain a solution. This exaggeration is sometimes unnecessary, and it is sufficient to reach reasonable solutions.

Determining the sufficient number of points is necessary when solving the BVP, which can be done through controlling and determining the tolerance number. Reducing the number of mesh points leads to a weak solution, and increasing them leads to unnecessary costs, so estimating the sufficient number is crucial.

As a comparison between HOFiD_bvp and bvp4c, it is obvious that there are differences in accuracy, computational efficiency, and the mesh points. In all computed cases, the HOFiD_bvp obtained consistently lower maximal residuals and errors for a prescribed tolerance, implying better numerical performance in terms of certain accuracy and robustness. In contrast, the bvp4c achieved reasonable error control, especially in combination with smaller tolerances (e.g., 1×10^{-8}), although at the expense of a vast number of mesh points.

One of the most important features of the bvp4c algorithm that makes it reliable is that the solution converges in most cases, even if the initial guess is not good enough, especially when the value of z is small. This feature comes from the way the algorithm is built using the properties in the Simpson method.

From a computational efficiency perspective, HOFiD_bvp was notably more efficient, performing far fewer ODE and boundary condition function evaluations while maintaining equivalent high accuracy. In contrast, bvp4c, with its adaptive mesh refinement, required a significantly higher number of function evaluations, as reflected in the considerably larger number.

Taken together, these results indicate that HOFiD_bvp is more appropriate for scenarios with a priority on computational efficiency and stability, whereas bvp4c may be more optimal for scenarios that require adaptive mesh refinement and tolerance-based error control.

6 Conclusion

This study presented a detailed numerical comparison between HOFiD_bvp, a high-order finite difference solver, and bvp4c, a collocation-based solver implemented in MATLAB, for the solution of singular second-order boundary value problems. Four representative test problems with first-kind (regular) singularities, drawn from radially symmetric diffusion, nonlinear reaction-diffusion, phase separation in materials science, and fluid dynamics on infinite domains, were used as benchmarks. The numerical experiments demonstrated that HOFiD_bvp consistently achieves smaller maximum residuals and lower maximum errors with fewer ODE and boundary condition function evaluations, indicating superior computational efficiency for this class of problems. In contrast, bvp4c proved more robust for nonlinear singular problems and provides reliable tolerance-based error control through adaptive mesh refinement, at the cost of a higher number of mesh points and function evaluations. These results confirm that solver selection should be problem-specific: HOFiD_bvp is preferable when computational efficiency and accuracy are the primary concerns, while bvp4c is advantageous when robustness, adaptive refinement, and ease of use are prioritized — particularly for nonlinear problems or when a reliable initial guess is unavailable.
Research Limitations: The present study is subject to several limitations that should be acknowledged. First, the comparison is restricted to second-order singular ODEs with first-kind (regular) singularities; problems with irregular or higher-order singularities were not investigated. Second, only four benchmark problems were tested, and extending this study to a larger and more diverse problem set would yield more generalizable conclusions. Third, the study did not investigate the effect of problem stiffness on solver performance. Finally, parallel or GPU-accelerated implementations of these algorithms were not considered, which may alter the computational cost comparisons.
Future Work: Several directions for future investigation are suggested. Future studies should extend the comparison to systems of singular ODEs, higher-order BVPs, and fractional-order differential equations. The integration of machine learning-based mesh adaptation strategies with these solvers represents a promising research avenue. Additionally, incorporating uncertainty quantification and stochastic perturbations in the problem formulation would extend the applicability of the findings. A rigorous convergence and stability analysis under varying singularity strengths is also warranted. Finally, the development of hybrid solvers that combine the efficiency of HOFiD_bvp with the robustness of bvp4c represents an important optimization target for future computational work.

Declarations

Author Contributions

All authors contributed equally to the design of the study, data analysis, and writing of the manuscript, and share equal responsibility for the content of the paper.

Funding

The authors received no specific funding for this research. The authors would like to thank all colleagues and reviewers whose constructive comments significantly improved the quality of this manuscript.

Conflict of Interest

The authors confirm that no conflict of interest has been declared.

Artificial Intelligence Statement

Artificial intelligence (AI) tools, including large language models, were used solely for language editing and improving readability. AI tools were not used for generating ideas, performing analyses, interpreting results, or writing the scientific content. All scientific conclusions and intellectual contributions were made exclusively by the authors.

Publisher's Note

References

[1] Abeldinov, Y. (2025). "Numerical methods for solving direct and inverse Sturm–Liouville problems." *Proceedings of Young Scientists and Specialists of the Samara University*. https://vmuis.ru/smus/article/view/27439.

[2] Adams, M., Muir, P. (2024). "Differential equation software for the computation of error-controlled continuous approximate solutions." *Numerical Algorithms*, 96(3), 1021–1044. https://doi.org/10.1007/S11075-024-01784-1.

[3] Ahmed, H.M., Abd-Elhameed, W.M. (2024). "Spectral solutions of specific singular differential equations using a unified spectral Galerkin–collocation algorithm." *Journal of Nonlinear Mathematical Physics*, 31(1), 1–22. https://doi.org/10.1007/S44198-024-00194-0.

[4] Al Arfaj, K., Levesley, J. (2023). "Lagrange radial basis function collocation method for boundary value problems in 1D." *AIMS Mathematics*, 8(11), 27542–27572. https://doi.org/10.3934/MATH.20231409.

[5] Al-Doori, V.S., Abbas, S.Q., Kulikov, O., Ismail, M.N. (2024). "Home automation system with a GUI that is powered by Arduino and MATLAB." *Conference of Open Innovation Association, FRUCT*. https://www.scopus.com/pages/publications/85193393535.

[6] Alshanti, W.G., Batiha, I.M., Hammad, M.A., Khalil, R. (2023). "A novel analytical approach for solving partial differential equations via a tensor product theory of Banach spaces." *Partial Differential Equations in Applied Mathematics*, 8, 100531. https://doi.org/10.1016/j.padiff.2023.100531.

[7] Amodio, P., Settanni, G. (2009). "A deferred correction approach to the solution of singularly perturbed BVPs by high order upwind methods: Implementation details". *AIP Conference Proceedings*, 1168, 711–714. https://doi.org/10.1063/1.3241565

[8] Amodio, P., Sgura, I. (2005). "High-order finite difference schemes for the solution of second-order BVPs." *Journal of Computational and Applied Mathematics*, 176(1), 59–76. https://doi.org/10.1016/j.cam.2004.07.008.

[9] Arifeen, S.U., Haq, S., Ghafoor, A., Ullah, A., Kumam, P., Chaipanya, P. (2021). "Numerical solutions of higher order boundary value problems via wavelet approach." *Advances in Difference Equations*, 2021(1), 1–15. https://doi.org/10.1186/S13662-021-03495-6.

[10] Ascher, U., Christiansen, J., Russell, R.D. (1981). "Algorithm 569: COLSYS: Collocation software for boundary-value ODEs [D2]." *ACM Transactions on Mathematical Software*, 7(2), 223–229. https://doi.org/10.1145/355945.355951.

[11] Auzinger, W., Fallahpour, M., Koch, O., Weinmüller, E. (2024). "Implementation of a path-following strategy with an automatic step-length control: New MATLAB package bvpsuite2.0." TU Wien Repository. https://repositum.tuwien.at/handle/20.500.12708/30869.

[12] Balluffi, R.W., Allen, S.M., Carter, W.C. (2005). "Diffusion as a series of discrete jumps." *Kinetics of Materials*, 154–159.

[13] Bandyopadhyay, S., Kunkel, C.J. (2025). "Existence result for singular second order dynamic equations with mixed boundary conditions." https://doi.org/10.48550/arXiv.2506.16505

[14] MATLAB Central. (n.d.). "Tutorial on solving BVPs with BVP4C." https://www.mathworks.com/matlabcentral/fileexchange/3819-tutorial-on-solving-bvps-with-bvp4c.

[15] Chen, S.B., Shahmir, N., Ramzan, M., Sun, Y.L., Aly, A.A., Malik, M.Y. (2021). "Thermophoretic particle deposition in the flow of dual stratified Casson fluid with magnetic dipole and generalized Fourier's and Fick's laws." *Case Studies in Thermal Engineering*, 26, 101186. https://doi.org/10.1016/J.CSITE.2021.101186.

[16] Ctor León, V., Scárdua, B. (2024). "Regularity, synthesis, rigidity and analytic classification for linear ordinary differential equations of second order." *Journal of Applied Mathematics*, 2(4), 1698. https://doi.org/10.59400/JAM.V2I4.1698.

[17] Cui, X., Xia, Y. (2022). "Second-order differential equation with indefinite and repulsive singularities." *Bulletin of the Malaysian Mathematical Sciences Society*, 46(3). https://doi.org/10.1007/s40840-023-01497-z.

[18] Datsko, B.Y., Kutniv, M.V. (2025). "Explicit numerical methods for solving singular initial value problems for systems of second-order nonlinear ODEs." *Numerical Algorithms*, 98(2), 929–942. https://doi.org/10.1007/s11075-024-01820-0.

[19] Drazin, P.G. (1983). Solitons. *Cambridge University Press.* https://doi.org/10.1017/CBO9780511662843.

[20] Egidi, N., Giacomini, J., Maponi, P. (2023). "A perturbative approach for the solution of Sturm–Liouville problems." *Applied and Computational Mathematics*, 12(3), 46–54. https://doi.org/10.11648/J.ACM.20231203.11.

[21] Ergashev, T.G. (2020). "Solving the Dirichlet and Holmgren problems for a three-dimensional elliptic equation by the potential method." *arXiv: Analysis of PDEs*. https://api.semanticscholar.org/CorpusID:213005636.

[22] Filipuk, G., Kecker, T. (2022). "On singularities of certain non-linear second-order ordinary differential equations." *Results in Mathematics*, 77(1), 41. https://doi.org/10.1007/s00025-021-01577-1.

[23] File, G., Gadisa, G., Aga, T., Reddy, Y.N. (2017). "Numerical solution of singularly perturbed delay reaction-diffusion equations with layer or oscillatory behaviour." *American Journal of Numerical Analysis*, 5(1), 1–10. https://doi.org/10.12691/ajna-5-1-1.

[24] Hamidizadeh, K., Manaviyat, R., Mirvakili, S., Davvaz, B. (2024). "Application of soft sets to graph coloring." *Computational and Applied Mathematics*, 43(4). https://doi.org/10.1007/S40314-024-02738-Y.

[25] Heinonen, J., Kilpeläinen, T., Martio, O. (2018). *Nonlinear Potential Theory of Degenerate Elliptic Equations*. Dover Publications, Mineola, New York.

[26] Hohenegger, M., Settanni, G., Weinmüller, E.B., Wolde, M. (2024). "Numerical treatment of singular ODEs using finite difference and collocation methods." *Applied Numerical Mathematics*, 205, 184–194. https://doi.org/10.1016/j.apnum.2024.07.002.

[27] Jajarmi, A., Baleanu, D. (2018). "Suboptimal control of fractional-order dynamic systems with delay argument." *Journal of Vibration and Control*, 24(12), 2430–2446. https://doi.org/10.1177/1077546316687936.

[28] Koltape, L.T., Hojjati, G., Fazeli, S., Abdi, A. (2024). "Super implicit two-step collocation methods for ordinary differential equations." *Computational and Applied Mathematics*, 43(6), 1–22. https://doi.org/10.1007/s40314-024-02848-7.

[29] Kutluay, S., Murat, Y., Karakas, A.S. (2023). "A robust septic Hermite collocation technique for Dirichlet boundary condition heat conduction equation." *arXiv:2312.05042*. https://arxiv.org/pdf/2312.05042.

[30] Li, M.M., Parise, D., Sarnataro, L. (2024). "Boundary behavior of limit-interfaces for the Allen–Cahn equation on Riemannian manifolds with Neumann boundary condition." *Archive for Rational Mechanics and Analysis*, 248(6), 1–43. https://doi.org/10.1007/S00205-024-02070-Z.

[31] Liu, C.S., Chang, C.W., Kuo, C.L. (2024). "Numerical analysis for Sturm–Liouville problems with nonlocal generalized boundary conditions." *Mathematics*, 12(8), 1265. https://doi.org/10.3390/MATH12081265.

[32] Mohamed, Z., Yousif, M., Hamza, A.E. (2022). "Solving nonlinear fractional partial differential equations using the Elzaki transform method and the homotopy perturbation method." *Abstract and Applied Analysis*, 2022, 4743234. https://doi.org/10.1155/2022/4743234.

[33] Nur, C. (2021). "On the estimates of periodic eigenvalues of Sturm–Liouville operators with trigonometric polynomial potentials." *Mathematical Notes*, 109(5–6), 794–807. https://doi.org/10.1134/S0001434621050114.

[34] Omar, S.S., Ahmed, W.S., Ismail, M.N., Sieliukov, O. (2024). "In-depth examination of a fingerprint recognition system using the Gabor filter." *Conference of Open Innovation Association, FRUCT*, 532–543. https://doi.org/10.23919/fruct61870.2024.10516364.

[35] Scott, D.R., Stevenson, D.J. (1984). "Magma solitons." *Geophysical Research Letters*, 11(11), 1161–1164. https://doi.org/10.1029/GL011I011P01161.

[36] Seiler, W.M., Seiß, M. (2024). "Singular initial value problems for some quasi-linear second-order ODEs." *Journal of Dynamics and Differential Equations*, 1–23. https://doi.org/10.1007/s10884-024-10396-1.

[37] Settanni, G. (2024). "Potentiality of the HOFiD_bvp code in solving different kind of second order boundary value problems." *Applied Numerical Mathematics*, 200, 379–388. https://doi.org/10.1016/j.apnum.2023.08.008.

[38] Shampine, L.F., Gladwell, I., Thompson, S. (2003). *Solving ODEs with MATLAB*. Cambridge University Press.

[39] Stoer, J., Bulirsch, R. (1980). *Introduction to Numerical Analysis* (Chapter 7: Ordinary Differential Equations, pp. 404–535). Springer. https://doi.org/10.1007/978-1-4757-5592-3_7.

[40] Trogdon, T. (2024). "The ultraspherical rectangular collocation method and its convergence." https://doi.org/10.48550/arXiv.2401.03608

[41] Wazwaz, A.M. (2000). "The modified Adomian decomposition method for solving linear and nonlinear boundary value problems of tenth-order and twelfth-order." *International Journal of Nonlinear Sciences and Numerical Simulation*, 1(1), 17–24. https://doi.org/10.1515/IJNSNS.2000.1.1.17.

[42] Xin, Y., Cui, X., Liu, J. (2020). "An exact expression of positive periodic solution for a first-order singular equation." *Advances in Difference Equations*, 2020(1), 1–9. https://doi.org/10.1186/s13662-020-02986-2.

[43] Yücel, U. (2015). "Numerical approximations of Sturm–Liouville eigenvalues using Chebyshev polynomial expansions method." *Cogent Mathematics*, 2(1), 1045223. https://doi.org/10.1080/23311835.2015.1045223.

[44] Zhu, H.L. (2022). "General solutions' laws of linear partial differential equations I." *Partial Differential Equations in Applied Mathematics*, 6, 100418. https://doi.org/10.1016/j.padiff.2022.100418.

Authors Bio-sketches

Saad Qasim Abbas is a specialist in applied mathematics with over two decades of university-level teaching experience spanning institutions in Iraq and the United States. His academic work bridges the disciplines of engineering and mathematics, with particular emphasis on engineering mathematics, numerical analysis, and the advancement of educational methodologies. Throughout his career, he has made notable contributions to scholarly research and curriculum development, reflecting a sustained commitment to both applied research and the evolution of mathematical pedagogy at the university level. Corresponding author: Email: dr.saad@bauc14.edu.iq

Wasan Saad Ahmed is a researcher affiliated with the University of Diyala, Iraq, where she holds a verified academic position within the College of Sciences. Her research interests span machine learning, artificial intelligence, and data mining. She has contributed to collaborative scholarly work in applied mathematics and computational science, including studies on the dynamical behavior of complex mathematical systems such as the logistic map and Mandelbrot set in the complex plane. Her work reflects a commitment to advancing interdisciplinary research at the intersection of mathematics and intelligent computing.