

https://dspace.org/

Adaptive Multidomain Numerical Solution for Singularly Perturbed Fractional Differential Equation Chebyshev Pseudospectral Method

Y Tijani; Olumuyiwa Otegbeye; S Oloniiju

https://hdl.handle.net/10539/46477
WOS

Downloaded from a DSpace repository. https://dspace.org/

Journal of Nonlinear Science (2025) 35:100
https://doi.org/10.1007/s00332-025-10199-8

Journal of Nonlinear Science

Adaptive Multidomain Numerical Solution for Singularly Perturbed Fractional Differential Equation: Chebyshev Pseudospectral Method

Yusuf Olatunji Tijani[1] · Olumuyiwa Otegbeye[2] · Shina Daniel Oloniiju[1]

Received: 24 October 2024 / Accepted: 28 July 2025
© The Author(s) 2025

Abstract

Singularly perturbed differential equations pose significant challenges for many numerical and semi-analytical solution methods in classical calculus. The complexity of these equations increases when dealing with fractional differential equations, which exhibit unique properties, including the nonlocal nature of the arbitrary non-integer order differential operators. This study presents an adaptive multidomain Chebyshev pseudospectral method to effectively approximate the solutions of singularly perturbed fractional differential equations. In each subdomain, the fractional differential operator in the Caputo sense accounts for both local derivative effects and memory by dividing the discrete fractional operator into two distinct components, one representing local behavior and the other capturing memory effects. The adaptivity of the method is controlled by ensuring that the maximum residual error in each domain remains below a predetermined tolerance value. If the maximum error exceeds the set tolerance, the size of the subinterval is reduced by a specified ratio, and the solution is recomputed within that interval. To assess the performance of the adaptive method, a comparative study with the multidomain pseudospectral method with uniform step length is used. The accuracy of the adaptive multidomain Chebyshev pseudospectral method is validated through a series of numerical tests on various fractional differential equations, thus demonstrating the effectiveness of the approach. The adaptive multidomain pseudospectral method offers a robust solution for handling the challenges posed by

Communicated by Anthony Bloch.

✉ Yusuf Olatunji Tijani
olodyusuf@gmail.com

Olumuyiwa Otegbeye
olumuyiwa.otegbeye@wits.ac.za

Shina Daniel Oloniiju
s.oloniiju@ru.ac.za

[1] Department of Mathematics, Rhodes University, Makhanda, PO Box 94, Grahamstown 6140, South Africa

[2] School of Computer Science and Applied Mathematics, University of the Witwatersrand, Private Bag 3, Braamfontein, Johannesburg 2050, South Africa

Published online: 14 August 2025

singularly perturbed differential equations common in fields such as wave propagation, astrophysics, and quantum mechanics.

Keywords Adaptive · Singularly perturbed · Multidomain · Fractional differential equation · Spectral method · Errors

Mathematics Subject Classification 65L11 · 34A08 · 65L05 · 65L50

1 Introduction

Fractional differentiation and integration, which originated around the same time as their classical counterparts, form the foundation for fractional differential equations (Podlubny 1999). In recent years, the understanding and practical application of fractional differential equations has gained significant recognition across various disciplines, including elasticity theory, transport theory, experimental fluid dynamics, and mathematical biology, among others (Ninghu 2020; Diethelm and Freed 1999; Rossikhin and Shitikova 1997). The calculus of arbitrary order has expanded the range of mathematical tools available for modeling real-world phenomena. The operators used in fractional differential equations, like the Riemann–Liouville and Caputo derivatives, are nonlocal, meaning that the derivative at a point depends on all values to the left or right rather than only those in a local neighborhood of the point. This nonlocal property of fractional operators characterizes fractional differential equations as global in nature (Owolabi and Atangana 2019). A significant advantage of fractional differential equations is their ability to model problems that involve nonlocality and memory effects, aspects that classical differential equations cannot address (Podlubny 1999; Huseynov et al. 2021). It can be argued that most classical differential equations do not readily lend themselves to closed-form solutions (Evans 1998). This may be due to nonlinearity, higher orders, or non-homogeneity of the differential equations. Fractional differential equations are typically more intricate and challenging to solve than classical differential equations (Garrappa 2018). As a result, numerical approximations have emerged as efficient techniques for solving differential equations and addressing uncertainties in differential models. The nonlinearity in the equation itself or in its initial or boundary conditions fundamentally complicates a differential equation (Logan 1994). An example of this complexity is the singularly perturbed differential equation (Kadalbajoo and Patidar 2002).

Singularly perturbed problems are a key aspect of initial-boundary value problems and have significant applications in geomechanics, enzyme models, and mathematical biology problems (Kadalbajoo and Patidar 2002). These equations arise when a small parameter multiplies the highest order of the differential equation (Kadalbajoo and Patidar 2002; Endrie and Duressa 2024). Consider the equation

$$-\epsilon \frac{\mathrm{d}^2 u(t)}{\mathrm{d}t^2} + u(t) = 1, \quad t \in (0,1), \quad u(0) = u(1) = 0. \tag{1}$$

From Eq. (1), it can be seen that if the parameter ϵ is set to zero, it yields $u(t) = 1$, which is inconsistent with the boundary conditions. Consequently, the reduced problem has no solution in $\mathcal{C}^2([0, 1])$, the space of twice continuously differentiable functions. This indicates that the solution to Eq. (1) is not well posed when ϵ becomes very small. This implies that the solution to the problem may exhibit large derivatives near the boundary, and the presence of boundary layers typically makes numerical approximations of singularly perturbed problems challenging (Kadalbajoo and Patidar 2002). It also renders certain classical numerical techniques unstable, resulting in inaccurate numerical results. Finite difference-based methods, such as Crank–Nicholson, may produce reasonable approximations when very fine meshes are used for the grid discretization, but this can lead to a computationally demanding scheme (Sharma et al. 2013). Therefore, developing an efficient, fast, and accurate numerical method for approximating solutions to such differential equations is essential. In this study, we will consider singularly perturbed fractional initial-value differential equations of the form:

$$ {}_C D^{\theta}_{0,t} u(t) = g[t, u(t), u(t)^2, u'(t), u''(t), \epsilon]. \tag{2}$$

Here, ${}_C D^{\theta}_{0,t}$ is the Caputo fractional differential operator of order θ, $g : \mathbb{R} \to \mathbb{R}^n$.

Following the study of Shen et al. (2011), spectral methods belong to a class of methods known as the method of weighted residuals, which use infinitely differentiable functions to approximate the solutions of differential equations. These methods use a truncated series of orthogonal functions, such as Chebyshev, Legendre, Jacobi, and Hermite polynomials, to approximate the solutions of differential equations (Canuto et al. 2006). Spectral methods are considered global methods as they discretize differential equations globally to achieve high-order approximations of their solutions (Gottlieb and Gottlieb 2009). Spectral methods can frequently achieve higher accuracy, surpassing the accuracy typically obtained through other traditional numerical methods like the finite difference or finite element methods (Trefethen 2000; Zayernouri and Em Karniadakis 2014). Sweilam and Khader (2010) applied the Legendre pseudospectral method to discretize the fractional advection-dispersion equation involving the Caputo fractional differential operator. Oloniiju et al. (2019) used the Chebyshev pseudospectral method to tackle a continuum mechanics problem involving fractional constitutive equations for a fluid dynamics problem. Other applications of the spectral-based methods to classical and fractional differential equations can be found in the following studies (Motsa 2012; Otegbeye 2018; Oloniiju et al. 2021; Doha et al. 2011; Li et al. 2012; Ezz-Eldien and Doha 2023). Ezz-Eldien (2018) applied the spectral tau method, using Jacobi polynomials as basis functions, to solve a system of multi-pantograph equations. The performance of the method was validated through numerical simulations. The spectral Galerkin (SG) technique using the shifted Jacobi polynomials was employed to approximate the solution of a delay time-fractional partial differential equation by Alsuyuti et al. (2023). The effectiveness of the proposed method was assessed by comparing the numerical solutions from five examples with known analytic solutions and results from other techniques in the literature. Spectral-based methods typically use the extrema of orthogonal polynomials as collocation points, which are densely packed near the boundaries. This is particularly

advantageous for problems like Rayleigh–Bénard convection or channel flow, where a higher density of collocation points is needed to accurately resolve viscous sublayers (Nnakenyi et al. 2015). However, for problems where rigidity is concentrated in the interior points, the direct application of a single-domain pseudospectral method may lead to some numerical trade-offs. Despite advances in spectral methods, their direct application to certain differential equations, such as fractional singularly perturbed equations within a single domain, can lead to significant accuracy issues (Motsa et al. 2013). This is due to the nonlinearity introduced by the singular perturbation parameter and the nonlocal nature of these fractional differential equations. To minimize the loss of accuracy, domain decomposition or multidomain techniques can be applied (Samuel and Motsa 2019).

The multidomain technique for initial-value problems involves dividing the time interval into multiple subintervals and solving the differential equation on each subinterval sequentially. The process begins with the initial condition, which serves as the starting point or left boundary for the first subinterval. The solution obtained at the last discrete point in the first subinterval is then used as the initial condition for the next subinterval. This procedure continues through all subintervals until the final domain is reached (Oloniiju et al. 2024; Xu and Hesthaven 2014). This approach offers several advantages, such as improving the condition of the algebraic system and allowing larger step sizes compared to single-domain methods. It is particularly effective for solving problems that involve interface tracking and immersed boundary methods (Motsa 2012), and it significantly enhances the accuracy of approximations for stiff and chaotic systems. The multidomain technique is also efficient for problems with long-term dynamics (Oloniiju et al. 2024). However, its direct application to singularly perturbed differential equations may give rise to some significant numerical challenges. To overcome these challenges, an adaptive multidomain approach, which dynamically adjusts each subdomain's length, may improve efficiency and accuracy.

An adaptive numerical technique dynamically adjusts the subdomains based on the complexity of the differential equation or the error estimation of the approximate solution (Peter and Martin 2012). These numerical techniques aim to improve efficiency, accuracy, and computational time by adapting their step sizes, mesh resolutions, or other parameters during the computation (Qureshi et al. 2023). An example of an adaptive numerical technique is adaptive quadrature for numerical integration (Gander and Gautschi 2000). Instead of using a fixed number of equally spaced intervals, adaptive quadrature methods automatically refine the intervals in regions where the integrand changes rapidly and coarsen them in smoother areas. This approach enables more accurate results with fewer function evaluations.

Another example is adaptive time-stepping ordinary differential equation solvers, like the Runge–Kutta–Fehlberg (RKF) methods, which dynamically adjust their time steps to efficiently handle stiff or non-stiff differential models. This allows the technique to focus computational resources where they are most needed. However, the effectiveness of these methods deteriorates when applied to fractional differential equations. This is because the RKF method is local, while fractional differential equations are inherently nonlocal. Global methods, such as the Chebyshev pseudospectral method, are better suited for this purpose (Oloniiju et al. 2024). Adaptive techniques

have become particularly important for handling nonlinear models, chaotic systems, stiff equations, and singular perturbation problems (Hans-Gorg et al. 1994).

A thorough literature review reveals that no existing studies have explored the performance of an adaptive multidomain Chebyshev pseudospectral method for singularly perturbed fractional differential equations. To demonstrate the efficiency of the adaptive multidomain technique, we will use established numerical metrics, such as absolute error, residual error, and computational time, to compare the accuracy and efficiency of this proposed technique with a multidomain pseudospectral method with equal subintervals and other numerical methods.

The rest of this manuscript is structured as follows: Section 2 provides the preliminaries and definitions required to develop the adaptive multidomain scheme in the context of the calculus of arbitrary orders. Section 3 details the development of the adaptive scheme using the Chebyshev–Gauss–Lobatto quadrature and discusses the development of a nonoverlapping multidomain pseudospectral scheme for differential equations of arbitrary non-integer orders. This section also briefly addresses the error estimates associated with using Chebyshev–Gauss–Lobatto points. Section 4 presents the numerical performance of the method through five distinct numerical examples, and finally, Section 5 concludes the study.

2 Preliminaries and Definitions

This section outlines vital preliminary concepts that will be referenced later, establishing the foundation for the terminology used throughout the article.

Definition 1 The gamma function, which extends the factorial function, is defined as Podlubny (1999)

$$\Gamma(n) = (n-1)! = \int_0^\infty t^{n-1} e^{-t} \mathrm{d}t, \tag{3}$$

where Γ is the Euler gamma function.

Definition 2 Suppose $t > 0$ and $g \in \mathcal{C}\big([0,t]\big)$ is continuously defined on a closed interval $[0,t]$. From Cauchy multiple integrals (Sikora 2023)

$$\begin{aligned}&\int_0^t \left(\int_0^{\sigma_n} \left(\cdots \left(\int_0^{\sigma_2} g(\sigma_1) \mathrm{d}\sigma_1 \right) \cdots \right) \mathrm{d}\sigma_{n-1} \right) \mathrm{d}\sigma_n \\ &= \frac{1}{(n-1)!} \int_0^t (t-\sigma)^{n-1} g(\sigma) \mathrm{d}\sigma, \quad n \in \mathbb{N}. \end{aligned} \tag{4}$$

Definition 3 (Let $\theta \in \mathbb{R}_+$, $t \in \mathbb{R}_+$, and $g \in \mathcal{C}\big([0,t]\big)$, then the Riemann–Liouville fractional integral of order, θ is Podlubny (1999))

$$I^\theta g(t) = \frac{1}{\Gamma(\theta)} \int_0^t (t-\sigma)^{\theta-1} g(\sigma) \mathrm{d}\sigma \tag{5}$$

Definition 4 Let $g(t) \in \mathcal{C}^n([0,t])$, the space of n-times continuously differentiable functions, then the left-sided arbitrary non-integer θ-order differential operator in the Caputo sense is defined as (Oloniiju et al. 2024; Caputo 1967)

$$ {}_C D^{\theta}_{0,t}\, g(t) = \begin{cases} \dfrac{1}{\Gamma(n-\theta)} \displaystyle\int_0^t \frac{g^n(\sigma)}{(t-\sigma)^{\theta+1-n}} \mathrm{d}\sigma, & \text{for} \quad n-1 < \theta < n, \\ \dfrac{\mathrm{d}^n g(t)}{\mathrm{d}t^n}, & \text{for} \quad \theta = n, \quad n \in \mathbb{Z}_+. \end{cases} \tag{6} $$

In Caputo's definition of the fractional differential operator, the integer order derivative of the function is computed first, followed by applying the Riemann–Liouville integral of arbitrary order. For $\theta \in (n-1, n)$, the Caputo fractional derivative ${}_C D^{\theta}_{0,t} g(t)$ exists if $g \in \mathcal{C}^n\big([0,t]\big)$. As the Caputo fractional derivative is expressed in integral form, it operates as a nonlocal operator with memory properties, where the current state depends on past states.

Remark 1 The Caputo derivative defined in Eq. (6) satisfies several fundamental properties, including linearity, consistency with the classical result for the derivative of a constant, and adherence to both the generalized Leibniz rule and the index law for fractional orders (Podlubny 1999).

Definition 5 The fractional derivative of t^{μ} in the Caputo sense is defined as (Li et al. 2012)

$$ {}_C D^{\theta}_{0,t} t^{\mu} = \begin{cases} 0, & \mu \in \mathbb{N}_0,\ \mu < \lceil \theta \rceil \\ \dfrac{\Gamma(\mu+1)}{\Gamma(\mu+1-\theta)} t^{\mu-\theta}, & \mu \in \mathbb{N}_0,\ \mu \geq \lceil \theta \rceil, \end{cases} \tag{7} $$

where $\lceil \theta \rceil$ is the ceiling function of θ and $\lceil \theta \rceil \geq \theta$.

3 Numerical Discretization via Pseudospectral Approximation

In this section, we derive the multidomain schemes used in this study. The procedure for approximating the solution $u(t)$ of a differential equation of arbitrary real order is first demonstrated, followed by the approximation of the arbitrary real-order derivatives of the function and the subsequent implementation of the multidomain schemes.

3.1 Approximation of the Function $u(t)$

Given the function $u(t)$, which belongs to the space $\mathcal{C}^n([0, \mathbf{T}])$, it can be approximated using the shifted Chebyshev polynomials of the first kind. The following recursive expression defines the Chebyshev polynomials:

$$ \Phi_{\mathbf{T},0}(t) = 1, $$

$$\begin{aligned}
\Phi_{\mathbf{T},1}(t) &= \frac{2t}{\mathbf{T}} - 1, \\
\Phi_{\mathbf{T},2}(t) &= \frac{8t^2}{\mathbf{T}^2} - \frac{8t}{\mathbf{T}} + 1, \\
\Phi_{\mathbf{T},m+1}(t) &= 2\Phi_{\mathbf{T},1}(t) \cdot \Phi_{\mathbf{T},m}(t) - \Phi_{\mathbf{T},m-1}(t), \quad m \geq 3.
\end{aligned} \tag{8}$$

The series expression for the shifted Chebyshev polynomials takes the form:

$$\Phi_{\mathbf{T},m}(t) = m \sum_{j=0}^{m} \frac{(-1)^{m-j}(m+j-1)!2^{2j}}{(m-j)!(2j)!\mathbf{T}^j} t^j, \quad m > 0. \tag{9}$$

Hence, using the Chebyshev polynomial, the function $u(t)$ can be approximated in terms of a truncated series as

$$u(t) = \bar{\mathbf{U}}(t) \approxeq \sum_{m=0}^{M} \hat{\Omega}_m \Phi_{\mathbf{T},m}(t). \tag{10}$$

Here, $\hat{\Omega}_m$ are coefficients to be determined and satisfy the orthogonality condition, defined by the following inner product

$$\hat{\Omega}_m = \langle \bar{\mathbf{U}}(t), \Phi_{\mathbf{T},m}(t) \rangle = \frac{1}{k_m} \int_0^{\mathbf{T}} \frac{\bar{\mathbf{U}}(t)\Phi_{\mathbf{T},m}(t)}{\sqrt{\mathbf{T}t - t^2}} \mathrm{d}t = k_m^{-1} \sum_{j=0}^{\infty} \bar{\mathbf{U}}(t_j)\Phi_{\mathbf{T},m}(t_j)\lambda_j, \tag{11}$$

where

$$k_m = \begin{cases} \pi & m = 0 \\ \dfrac{\pi}{2} & \text{otherwise,} \end{cases} \quad \text{and} \quad \lambda_j = \begin{cases} \dfrac{\pi}{2M} & j = 0, M \\ \dfrac{\pi}{M} & \text{otherwise.} \end{cases} \tag{12}$$

Suppose $u(t)$ is a square-integrable and analytic function; therefore, $u(t)$ is approximated using a linear combination of $M+1$ shifted Chebyshev polynomials at $M+1$ shifted Chebyshev–Gauss–Lobatto nodes, t_r, as

$$u(t_r) \approxeq \sum_{j=0}^{M} \bar{\mathbf{U}}(t_j)\lambda_j \sum_{m=0}^{M} \frac{1}{k_m} \Phi_{\mathbf{T},m}(t_j)\Phi_{\mathbf{T},m}(t_r) = \sum_{j=0}^{M} \bar{\mathbf{U}}(t_j) F_j(t_r) = \mathbf{F}_{\mathbf{T}}^{\mathsf{T}} \bar{\mathbf{U}}, \tag{13}$$

where $\mathbf{F_T} = \mathbf{K}\Phi_{\mathbf{T}}^{\top}\boldsymbol{\lambda}\Phi_{\mathbf{T}}$, and

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_0 & & & \\ & \lambda_1 & & \\ & & \ddots & \\ & & & \lambda_M \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} \frac{1}{k_0} & & & \\ & \frac{1}{k_1} & & \\ & & \ddots & \\ & & & \frac{1}{k_M} \end{bmatrix},$$
$$\Phi_{\mathbf{T}} = \begin{bmatrix} \Phi_{\mathbf{T},0}(t_0) & \Phi_{\mathbf{T},0}(t_1) & \dots & \Phi_{\mathbf{T},0}(t_M) \\ \Phi_{\mathbf{T},1}(t_0) & \Phi_{\mathbf{T},1}(t_1) & \dots & \Phi_{\mathbf{T},1}(t_M) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{\mathbf{T},M}(t_0) & \Phi_{\mathbf{T},M}(t_1) & \cdots & \Phi_{\mathbf{T},M}(t_M) \end{bmatrix}. \tag{14}$$

The next step is to find the Caputo fractional derivative of Eq. (13). Assume that $u(t) \in \mathcal{C}^q([0, \mathbf{T}])$, where $M + \lceil\theta\rceil + 1 \leq q$, denoting the space of continuously differentiable functions for $t > 0$ and a fractional order $\theta > 0$, so that

$${}_C D^{\theta}_{0,t} u(t) \approx \sum_{j=0}^{M} \bar{\mathbf{U}}(t_j)\lambda_j \sum_{m=0}^{M} \frac{1}{k_m} \Phi_{\mathbf{T},m}(t_j) {}_C D^{\theta}_{0,t} \Phi_{\mathbf{T},m}(t). \tag{15}$$

The Caputo derivative of the space of shifted first kind Chebyshev polynomials, ${}_C D^{\theta}_{0,t} \Phi_{\mathbf{T},m}(t)$, can be derived using the series expansion of the polynomials (Oloniiju et al. 2024)

$$\begin{aligned} {}_C D^{\theta}_{0,t} \Phi_{\mathbf{T},m}(t) &= m \sum_{j=0}^{m} \frac{(-1)^{m-j}(m+j-1)!2^{2j}}{(m-j)!(2j)!\mathbf{T}^j} {}_C D^{\theta}_{0,t} t^j \\ &= m \sum_{j=\lceil\theta\rceil}^{m} \frac{(-1)^{m-j}(m+j-1)!2^{2j}}{(m-j)!(2j)!\mathbf{T}^j} \frac{\Gamma(j+1)}{\Gamma(j-\theta+1)} t^{j-\theta}. \end{aligned} \tag{16}$$

The expression, $t^{j-\theta}$, is now written as a linear combination of the shifted Chebyshev polynomials of the first kind at the Chebyshev–Gauss–Lobatto points. Therefore, Eq. (16) can be written as:

$${}_C D^{\theta}_{0,t} \Phi_{\mathbf{T},m}(t_r) \approx \mathrm{D}^{(\theta)} \Phi_{\mathbf{T}}, \tag{17}$$

where $\mathrm{D}^{(\theta)} = \mathrm{Z}\,\boldsymbol{\eta}\,\Psi\,\mathrm{Z}^{\mathsf{T}}\,\mathbf{K}$. The entries of the matrices Z, $\boldsymbol{\eta}$, and Ψ are defined by the following:

$$\left.\begin{aligned} Z_{m,j} &= \begin{cases} 1 & m = 0,\ j = 0, \\ \dfrac{m(-1)^{m-j}(m+j-1)!2^{2m}}{(m-j)!(2j)!\mathbf{T}^{j}} & m = 1(1)M,\ \ j = 0(1)m, \\ 0 & \text{otherwise}, \end{cases} \\ \eta_{j,j} &= \begin{cases} 0 & j < \lceil\theta\rceil, \\ \dfrac{\Gamma(j+1)}{\Gamma(j-\theta+1)} & j = \lceil\theta\rceil(1)M, \end{cases} \\ \Psi_{v,j} &= \begin{cases} 0 & j < \lceil\theta\rceil, \\ \dfrac{\sqrt{\pi}h^{j-\theta+v}\Gamma(j-\theta+v+\frac{1}{2})}{\Gamma(j-\theta+v+1)} & j = \lceil\theta\rceil(1)M,\ v = 0(1)s,\ \ s = 0(1)M. \end{cases} \end{aligned}\right\} \tag{18}$$

Thus, ${}_C D^{\theta}_{0,t} u(t_r)$ is given as:

$${}_C D^{\theta}_{0,t} u(t_r) \approx ({}_{0,\mathbf{T}}\mathbf{D}^{\theta})^{\mathsf{T}}\,\bar{\mathbf{U}}, \tag{19}$$

where ${}_{0,\mathbf{T}}\mathbf{D}^{\theta} = \boldsymbol{\lambda}\,\Phi^{\mathsf{T}}_{\mathbf{T}}\,\mathbf{K}\,\mathrm{D}^{(\theta)}\,\Phi_{\mathbf{T}}$. The matrix $({}_{0,\mathbf{T}}\mathbf{D}^{\theta})^{\mathsf{T}}$ is the fractional differentiation matrix which evaluates the fractional derivative of a function at the shifted Chebyshev–Gauss–Lobatto nodes in the domain $[0, \mathbf{T}]$.

3.2 Approximation in Equal Nonoverlapping Partitioned Domains

Consider a function $u(t)$ defined on the interval $t \in [0, \mathbf{T}]$, which is subdivided into p nonoverlapping subintervals. The partitioned points are given by $0 = t_0 < t_1 < \cdots < t_p = \mathbf{T}$, where the length of each subinterval is denoted as $\delta_k = t_k - t_{k-1}$ for $k = 1, 2, \ldots, p$. Assume that $t \in [t_k, t_{k+1}]$ and $\theta \in (\lceil\theta\rceil - 1, \lceil\theta\rceil]$, then, the fractional derivative ${}_C D^{\theta}_{0,t}$ can then be expressed as:

$${}_C D^{\theta}_{0,t} u(t) = \frac{1}{\Gamma(\lceil\theta\rceil - \theta)} \int_0^t \frac{u^{\lceil\theta\rceil}(\sigma)}{(t-\sigma)^{\theta+1-\lceil\theta\rceil}}\,\mathrm{d}\sigma. \tag{20}$$

This integral representation in Eq. (20) can be further decomposed into a sum of terms corresponding to the 'memory' and 'local' fractional derivatives in each subinterval as

$${}_C D^{\theta}_{0,t} u(t) = \sum_{f=1}^{k} \left[{}_C D^{\theta}_{t_{f-1},t_f} u(t) \right] + {}_C D^{\theta}_{t_k,t} u(t), \tag{21}$$

where ${}_C D^{\theta}_{t_{f-1},t_f} u(t)$ represents the contribution of the fractional derivative from the previous f-th subinterval and ${}_C D^{\theta}_{t_k,t} u(t)$ is the local fractional derivative in the current subinterval $[t_k, t_{k+1}]$. Eq. (21) is rewritten as

$$
\begin{aligned}
{}_C D^{\theta}_{0,t} u(t) = & \frac{1}{\Gamma(\lceil\theta\rceil - \theta)} \\
& \left(\sum_{f=1}^{k} \left[\int_{t_{f-1}}^{t_f} \frac{u^{\lceil\theta\rceil}(\sigma)}{(t-\sigma)^{\theta+1-\lceil\theta\rceil}} \, \mathrm{d}\sigma \right] + \int_{t_k}^{t} \frac{u^{\lceil\theta\rceil}(\sigma)}{(t-\sigma)^{\theta+1-\lceil\theta\rceil}} \, \mathrm{d}\sigma \right).
\end{aligned} \tag{22}
$$

The first term in Eq. (22) represents the 'memory' term of the fractional derivative, incorporating contributions from all previous subintervals. In contrast, the second term is the local fractional derivative within the current subinterval, $[t_k, t_{k+1}]$. These terms are then approximated using the pseudospectral method at $M+1$ Chebyshev–Gauss–Lobatto (CGL) points t_r^k in the subinterval $[t_k, t_{k+1}]$. The local fractional derivative in the subinterval $[t_k, t_{k+1}]$ is approximated as:

$$
{}_C D^{\theta}_{t_k,t} u(t_r^k) \approx ({}_{t_k,t_k+\delta_k}\mathbf{D}^{\theta})^{\mathsf{T}} \bar{\mathbf{U}}_k, \quad t \in [t_k, t_{k+1}], \tag{23}
$$

where $\bar{\mathbf{U}}_k$ is $u(t)$ evaluated at the CGL points within the k-th subinterval. The memory term, ${}_C D^{\theta}_{t_{f-1},t_f} u(t)$, is also approximated as

$$
\begin{aligned}
& {}_C D^{\theta}_{t_{f-1},t_f} u(t) \approx \sum_{j=0}^{M} \bar{\mathbf{U}}(t_j^f) \lambda_j \sum_{m=0}^{M} \frac{1}{k_m} \Phi_{\delta_k,m}(t_j^f) \; {}_C D^{\theta}_{t_{f-1},t_f} \Phi_{\delta_k,m}(t), \\
& t_j^f \in [t_{f-1}, t_f].
\end{aligned} \tag{24}
$$

In Eq. (24), the Chebyshev polynomials, $\Phi_{\delta_k,m}(t)$, are evaluated at the $M+1$ shifted CGL points, $t_r^k \in [t_k, t_{k+1}]$. Therefore, the definite integrals, ${}_C D^{\theta}_{t_{f-1},t_f} \Phi_{\delta_k,m}(t_r^k)$, defined as

$$
Q_{m,r} = \begin{cases} 0 & j < \lceil\theta\rceil \\ \dfrac{m!}{\Gamma(\lceil\theta\rceil - \theta)(m - \lceil\theta\rceil)!} \displaystyle\int_{t_{f-1}}^{t_f} \frac{(\sigma - t_{f-1})^{m-\lceil\theta\rceil}}{(t_r^k - \sigma)^{\theta+1-\lceil\theta\rceil}} \mathrm{d}\sigma, & t_r^k \in [t_k, t_{k+1}], \\ j = \lceil\theta\rceil, \ldots, M, \end{cases} \tag{25}
$$

can be evaluated using any quadrature formula. Hence, the approximation of the memory term evaluated at the collocation points, $t_r^f \in [t_{f-1}, t_f]$, becomes

$$
\sum_{f=1}^{k} {}_C D^{\theta}_{t_{f-1},t_f} u(t_r^f) \approx \sum_{f=1}^{k} ({}_{t_{f-1},t_f}\mathbf{D}^{\theta})^{\mathsf{T}} \bar{\mathbf{U}}_f, \tag{26}
$$

where ${}_{t_{f-1},t_f}\mathbf{D}^{\theta} = \boldsymbol{\lambda} \Phi_{\delta_k}^{\mathsf{T}} \mathbf{K} \mathbf{Z} \mathbf{Q}$ and the entries of the matrix, $\mathbf{Q}$, are defined by Eq. (25).

3.3 Adaptive Nonoverlapping Partitioned Domain for Initial-Value Problem

Consider the function, $u(t)$, defined on the interval $t \in [0, \mathbf{T}]$. In contrast to the multidomain method with equal subinterval lengths described in Sect. 3.2, where the interval $[0, \mathbf{T}]$ is divided into p nonoverlapping subintervals, the adaptive approach dynamically adjusts both the length of each subinterval and the number of subintervals p based on a predefined tolerance and how well the numerical scheme approximates the solution of the differential equation. The subdomains and partition points, $0 = t_0, t_1, \ldots, t_p = \mathbf{T}$, are not predetermined but are generated iteratively as the computation progresses. Beginning with $t_0 = 0$, the length of each subinterval $\delta_k = t_k - t_{k-1}$ is adaptively determined according to the following procedure:

1. Initializing the computation: The computation starts at the initial point $t_0 = 0$, with an initial step length δ_1 defined. The solution is computed in the first subinterval $[t_0, t_1]$. A tolerance value, denoted as tol, is set to check and control the accuracy of the solution within each subinterval.
2. Estimating the residual error: The norm of the residual error in each subinterval $[t_{k-1}, t_k]$ is estimated and checked against the preset tolerance using the following formula:

$$\text{Residual error} = \left\| {}_C D^{\theta}_{t_{k-1},t} u(t) - g[t, u(t), u(t)^2, u'(t), u''(t), \epsilon] \right\|_{\infty} < tol. \tag{27}$$

3. Adjusting the step length:
   - If the norm of the residual error is less than or equal to the preset tolerance, the approximate solution in the current subinterval $[t_{k-1}, t_k]$ is accepted, and the procedure doubles the latest step length and moves forward to approximate the solution in the next subinterval, $[t_k, t_{k+1}]$.
   - If the norm of the residual error exceeds the set tolerance, the step length, δ_k, is reduced using the expression

$$\delta_k^{\text{new}} = 0.9 \times \delta_k^{\text{old}} \times \left(\frac{\text{tol}}{\text{Residual error}} \right)^{\frac{1}{M}}, \tag{28}$$

   and the solution is recomputed in the adjusted subinterval.
4. Extending the adaptive domain: The procedure adaptively partitions the interval $[0, \mathbf{T}]$ by sequentially creating new subintervals $[t_k, t_{k+1}]$ based on the adaptive criterion. The number of subintervals, p, varies as needed, and the process continues until the endpoint $t_p = \mathbf{T}$ of the domain is reached. To ensure that the partitioning does not exceed the domain of the differential equation, we set the condition that if $t_k + \delta_k > \mathbf{T}$, then, $\delta_k = \mathbf{T} - t_k$.

3.4 Error Estimates

This subsection presents the general error estimates of the approximation in terms of the Chebyshev polynomials. Assume that $\Phi_{\mathbf{T},m}u$ is the projection of $u(t)$ onto the space of $M+1$ Chebyshev polynomials. The error estimate for $u - \Phi_{\mathbf{T},m}u$ in the Sobolev space equipped with its usual norm for $u(t) \in \mathcal{H}^n(0,\mathbf{T})$ and $n \in [1,\infty)$ is defined as

$$\|u - \Phi_{\mathbf{T},m}u\|_{\ell^2(0,\mathbf{T})} \leq CM^{-n}|u|_{\mathcal{H}^{n,M}(0,\mathbf{T})}. \tag{29}$$

We have the following error estimate for interpolation on the Gauss–Lobatto nodes

$$\|u - \Phi_{\mathbf{T},m}u\|_{\ell^2(0,\mathbf{T})} \leq CM^{1-n}|u|_{\mathcal{H}^{n,M}(0,\mathbf{T})}, \tag{30}$$

where C is independent of M and n. The proof of the above estimates is provided in Canuto et al. (2006).

4 Numerical Experimentation and Results

This section presents a detailed analysis of the performance of the proposed adaptive multidomain method. Various numerical examples are used to illustrate the accuracy, efficiency, and stability of the technique when applied to singularly perturbed fractional differential equations. The results are compared to the nonadaptive multidomain scheme using key metrics such as the absolute error (the difference between the exact and approximate solutions), the norm of the residual error, and the computational time to demonstrate the effectiveness of the adaptive technique.

Example 4.1 Consider the singularly perturbed linear fractional differential equation

$${}_0^C D_t^\theta u(t) + \epsilon u''(t) + u'(t) = \frac{2}{\Gamma(3-\theta)}t^{2-\theta} + 2\epsilon + 2t, \quad \text{for} \quad 0 < \theta < 1, \tag{31}$$

subject to the initial conditions $u(0) = 0,$ and $u'(0) = 0$. The closed-form solution of Eq. (31) is $u(t) = t^2$. The right-hand side belongs to the space of at least second-order continuously differentiable functions, i.e., $\mathcal{C}^2([0,T])$.

Table 1 compares the maximum absolute errors, residual errors, and computational times between the adaptive multidomain pseudospectral method and the multidomain pseudospectral method with uniform step sizes (i.e., $p = 50$, 200 and 300) (Oloniiju et al. 2024). The results indicate that the adaptive Chebyshev pseudospectral method yields lower absolute and residual errors than the multidomain technique with uniform step size. This improvement demonstrates the effectiveness of the adaptive approach in achieving higher accuracy. For this comparative analysis, the tolerance of the residual error for the adaptive technique was set at 10^{-3}. The table also suggests that the absolute errors of the equally spaced multidomain method decrease as the number of subdomains increases. However, this improvement comes at the cost of a higher

Table 1 The ℓ_∞-norms of the absolute and residual errors and CPU time of Example 4.1 with $M = 10$

Metrics	$\epsilon = 0.01,\ \theta = 0.5,\ \mathbf{T} = 5.0,\ \xi = 5.0$			
	Adaptive scheme ($p = 1$)	$p = 50$	$p = 200$	$p = 300$
Absolute error	9.3478×10^{-4}	2.2464×10^{-2}	2.1780×10^{-2}	1.9003×10^{-2}
Residual error	1.3855×10^{-13}	5.6626×10^{-11}	2.4131×10^{-10}	7.3949×10^{-10}
CPU time	0.012s	7.818s	119.515s	269.316s

Table 2 The ℓ_∞-norms of the absolute and residual errors of Example 4.1 for different values of the perturbation parameter, ϵ with $\theta = 0.5$, $M = 10$, and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \xi = 5.0,\ \mathbf{T} = 5.0,\ p = 1$			
	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$	$\epsilon = 0.0001$
Absolute error	1.4324×10^{-3}	9.3478×10^{-4}	8.9589×10^{-4}	8.9209×10^{-4}
Residual error	3.5171×10^{-13}	1.3855×10^{-13}	6.7501×10^{-14}	5.0626×10^{-14}

Table 3 The maximum absolute and residual errors of Example 4.1 for different initial step size, ξ with $\epsilon = 0.01$, $M = 10$, $\theta = 0.5$, and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \epsilon = 0.01,\ \mathbf{T} = 5.0$			
	$\xi = 1.0$	$\xi = 2.0$	$\xi = 3.0$	$\xi = 4.0$
Absolute error	8.9307×10^{-3}	8.3432×10^{-3}	8.1970×10^{-3}	5.5004×10^{-3}
Residual error	2.2515×10^{-13}	1.4521×10^{-13}	3.1796×10^{-13}	1.0897×10^{-12}
Number of subdomains (p)	3	2	2	2

computational time. Tables 2 and 3 present the maximum errors obtained by varying the singularly perturbed parameter ϵ and the step size of the first domain, ξ. The adaptive method uses only a single domain for the results presented in Table 2.

The results in Tables 2 and 3 , respectively, clearly show that the absolute errors improve as $\epsilon \to 0$ and the initial step size increases, i.e., $\xi \to T$. The residual error data in Table 2 highlight that the method proposed approximates the solution of Eq. (31) more accurately as ϵ approaches zero. Table 3 shows that the adaptive method uses three subdomains when the initial step size equals one. However, when the initial step sizes equal 2, 3, and 4, the method only requires two subdomains. This demonstrates how the adaptive approach adjusts the number of subdomains based on the complexity of the solution in each interval.

Example 4.2 Consider the nonlinear singularly perturbed fractional differential equation

$$\begin{aligned} {}_0^C D_t^\theta u(t) + \epsilon u''(t) + u'(t) + u^2(t) &= \frac{6}{\Gamma(4-\theta)} t^{3-\theta} \\ - \frac{2}{\Gamma(3-\theta)} t^{2-\theta} &+ \epsilon(6t-2) + 3t^2 - 2t + (t^3 - t^2)^2. \end{aligned} \tag{32}$$

Table 4 The absolute and residual error norms and CPU time of Example 4.2 with $M = 10$

Metrics	$\epsilon = 0.01,\ \theta = 0.5,\ \mathbf{T} = 5.0,\ \xi = 5.0$			
	Adaptive domain ($p = 1$)	$p = 50$	$p = 200$	$p = 300$
Absolute error	8.5023×10^{-4}	1.7049×10^{-2}	2.0310×10^{-3}	1.7557×10^{-3}
Residual error	6.1299×10^{-10}	1.2015×10^{-8}	1.2119×10^{-9}	1.1895×10^{-9}
CPU time	0.035s	7.863s	229.459s	536.195s

Table 5 The norms of the absolute and residual errors of Example 4.2 using different values of the perturbation parameter, ϵ, with $\theta = 0.5$, $M = 10$ and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \xi = 5.0,\ \mathbf{T} = 5.0,\ p = 1$			
	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$	$\epsilon = 0.0001$
Absolute error	1.1346×10^{-3}	8.5023×10^{-4}	8.2330×10^{-4}	8.2065×10^{-4}
Residual error	6.2937×10^{-10}	6.1299×10^{-10}	6.0936×10^{-10}	6.0254×10^{-10}

Table 6 The maximum absolute and residual errors of Example 4.2 for different initial domain length, ξ, with $\epsilon = 0.01$, $M = 10$, and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \epsilon = 0.01,\ \mathbf{T} = 5.0$			
	$\xi = 1.0$	$\xi = 2.0$	$\xi = 3.0$	$\xi = 4.0$
Absolute error	7.7770×10^{-4}	5.0100×10^{-4}	4.4339×10^{-4}	6.1360×10^{-4}
Residual error	1.9707×10^{-8}	1.2778×10^{-8}	1.9955×10^{-8}	1.8087×10^{-8}
Number of subdomains (p)	3	2	2	2

Here, $\theta \in (0, 1]$ with the starting conditions $u(0) = 0,$ and $u'(0) = 0$. The analytical solution to Eq. (32) subject to its initial conditions is given as $u(t) = t^2(t - 1)$. The right-hand side belongs to the space of second-order continuously differentiable functions.

The residual and absolute error norms presented in Table 4 mirror the trends observed for Example 4.1. The residual errors are nearly zero, indicating that the numerical scheme closely approximates the solution of the differential equation, albeit with varying accuracy. As shown in Table 4, the adaptive method surpasses the multidomain scheme with uniform step size in terms of error reduction and computational efficiency. The tolerance set for the adaptive scheme for this example is 10^{-3}.

A similar trend to that observed in Table 2 is evident in Table 5, where the proposed method effectively solves the arbitrary real-order differential equation as the perturbation parameter approaches zero. However, unlike the consistent reduction in absolute error with increase in domain length seen in Table 3, Table 6 reveals a distinct pattern. Specifically, the absolute errors decrease for $\xi = 1.0$ to $\xi = 3.0$, but subsequently increase as ξ progresses from 3.0 to 5.0. When the initial domain length is 1.0 (i.e., $\xi = 1.0$), the adaptive method uses three subdomains ($p = 3$). By contrast, for $\xi = 2.0$, $\xi = 3.0$, and $\xi = 4.0$, only two subdomains ($p = 2$) are required.

Table 7 The maximum absolute and residual errors and CPU time for Example 4.3 with $M = 10$

Metrics	$\epsilon = 0.01,\ \theta = 0.5,\ \mathbf{T} = 5.0,\ \xi = 5.0$			
	Adaptive domain ($p = 1$)	$p = 50$	$p = 200$	$p = 300$
Absolute error	3.4525×10^{-5}	4.7580×10^{1}	1.2889×10^{3}	1.1311×10^{2}
Residual error	2.4364×10^{-11}	1.9016×10^{2}	2.2390×10^{-3}	1.6084×10^{-8}
CPU time	0.014s	8.405s	124.019s	281.111s

Table 8 The maximum absolute and residual errors of Example 4.3 for different values of the perturbation parameter, ϵ, with $\theta = 0.5$, $M = 10$ and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \xi = 5.0,\ \mathbf{T} = 5.0,\ p = 1$			
	$\epsilon = 0.1$	$\epsilon = 0.01$	$\epsilon = 0.001$	$\epsilon = 0.0001$
Absolute error	3.4525×10^{-4}	3.4525×10^{-5}	3.4525×10^{-6}	3.4525×10^{-7}
Residual error	2.4364×10^{-9}	2.4364×10^{-11}	2.4364×10^{-13}	2.4364×10^{-15}

Example 4.3 Given the nonlinear singularly perturbed fractional differential equation

$$\epsilon\, {}_0^C D_t^{\theta} u(t) + u^2(t) = \frac{\epsilon^2}{\Gamma(2-\theta)} t^{1-\theta} + \epsilon^2 (1+t)^2, \quad \text{for} \quad 0 < \theta \leq 1, \tag{33}$$

where $u(0) = \epsilon$. The exact solution to Eq. (33) is given as $u(t) = \epsilon(1 + t)$.

A detailed analysis of Table 7 underscores the importance of using the adaptive multidomain technique. Notably, only the solution from the adaptive multidomain Chebyshev pseudospectral method agrees with the closed-form solution with an accuracy of 10^{-5}. In contrast, the multidomain pseudospectral method with uniform step size is unable to accurately resolve the singularly perturbed fractional differential equation in Example 4.3, resulting in an absolute error of 1.1311×10^2 with 300 subdomains. Additionally, Table 8 confirms that the performance of the adaptive method improves as the value of the perturbation parameter approaches zero. The tolerance set for the results in Table 9 is 10^{-1}; the method fails to converge for smaller tolerances. Furthermore, for this particular example, using a multidomain approach does not yield an accurate solution to the differential equation, as evidenced by the results in Tables 7 and 9 . These tables illustrate that while effective in other cases, the multidomain technique struggles to provide sufficient accuracy for this problem. The inaccuracies, even when using multiple subdomains, indicate that the structure of this differential equation requires a numerical scheme in a single domain (Table 8), as a multidomain technique, in this instance, does not improve the solution quality.

Example 4.4 Consider the singularly perturbed fractional differential equation, which is an interior layer problem (Ramos 2005)

$$\epsilon\, {}_0^C D_t^{\theta} u(t) + u(t) = g(t) + \epsilon g'(t), \quad \text{for} \quad 0 < t, \theta \leq 1, \tag{34}$$

Table 9 The norms of the absolute and residual errors of Example 4.3 for different starting domain length, ξ, with $\epsilon = 0.01$ and $\mathbf{T} = 5$

Metrics	$\theta = 0.5,\ \epsilon = 0.01,\ \mathbf{T} = 5.0$			
	$\xi = 1.0$	$\xi = 2.0$	$\xi = 3.0$	$\xi = 4.0$
Absolute error	3.9980×10^{-1}	1.7007×10^{-1}	1.6734×10^{-1}	7.0310×10^{-1}
Residual error	6.6713×10^{-2}	5.3586×10^{-2}	3.2314×10^{-2}	6.9536×10^{-1}
Number of subdomains (p)	3	2	2	2

where $g(t) = 10 - (10 + t)e^{-t}$ and $u(0) = 10$. Typically, this type of differential equation is characterized by solutions with steep gradients or layers that become increasingly thin as the perturbation parameter decreases. When the perturbation parameter ϵ is small, the solution $u(t)$ exhibits a boundary layer near $t = 0$ or another critical point, where the solution changes rapidly. Outside of this boundary layer, the solution changes more gradually. The challenge in numerical methods for this kind of problem is accurately capturing the dynamics in the boundary layer while maintaining the overall accuracy of the solution. The analytical solution of Eq. (34), for the case when $\theta = 1.0$, is given as $u(t) = 10 - (10 + t)e^{-\frac{t}{\epsilon}} + 10e^{-\frac{t}{\epsilon}}$.

Figure 1a illustrates the dynamics of the solution of Eq. (34) as $\epsilon \to 0$, highlighting the presence of a steep gradient $t = 0$. As ϵ decreases beyond 10^{-3}, the solutions of the equation begin to superimpose, and no significant differences are observed. Figure 1b depicts the norms of the absolute and residual errors in each subdomain for different values of the perturbation parameter. The number of subdomains used remains consistent when the value of the perturbation parameter is changed, while the initial step length remains constant. For instance, when $\xi = 0.01$, seven subdomains were used for the different values of ϵ. According to Table 10, for larger ϵ, the choice of initial domain length ξ has less impact on the accuracy of the approximate solution. In contrast, smaller ϵ generally benefits from a smaller initial domain length ξ to achieve lower errors. The table also shows that the accuracy of the approximate solution deteriorates as $\epsilon \to 0$. Table 11 presents the maximum residual error for two different non-integer orders of the differential equation, noting that the residual errors were computed as the exact solutions for these cases are unknown.

Table 11 shows that for both values of θ, decreasing ξ generally leads to more significant residual errors, while smaller values of ϵ result in more minor residual errors. This suggests that the method exhibits greater accuracy for smaller perturbation parameters. As $\theta \to 1$, the residual errors decrease, which indicates that solving the fractional differential equation is more challenging than the classical case. Additionally, it was observed that for a given initial domain length, the number of subdomains used by the adaptive method remains constant, regardless of changes in the perturbation parameter for Example 4.4. The adaptivity criterion was only met once across all the subdomains used in this example.

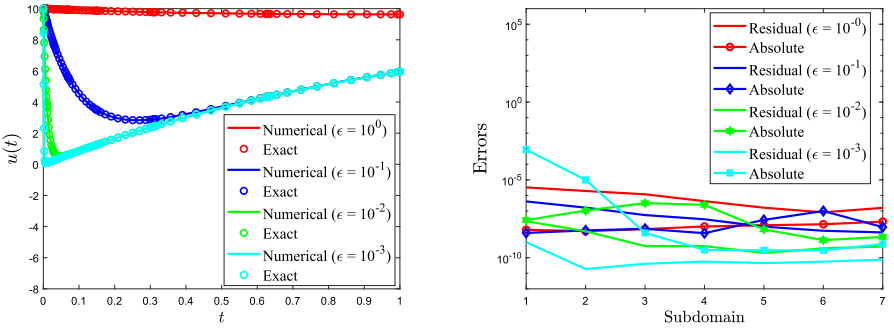


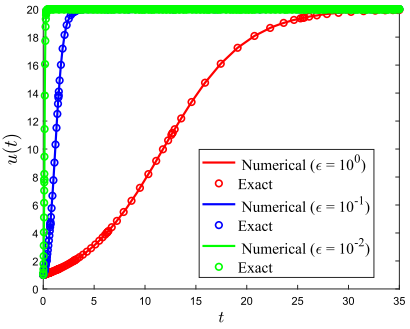
Fig. 1 **a** The exact and numerical solutions of Eq. (34), **b** the norms of the residual and absolute errors for different values of ϵ with $\theta = 1.0,\ \xi = 10^{-2},\ \mathbf{T} = 1.0,\ M = 15$

Table 10 The maximum absolute errors for Example 4.4 using different values of the initial domain length ξ and $M = 15,\ \theta = 1.0,\ \text{tol} = 10^{-3}$

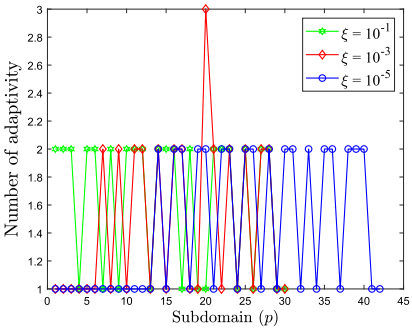
ϵ	$M = 15,\ \theta = 1.0,\ \mathbf{T} = 1.0$				
	$\xi = 10^{-1}$	$\xi = 10^{-2}$	$\xi = 10^{-3}$	$\xi = 10^{-4}$	$\xi = 10^{-5}$
10^{0}	1.698×10^{-8}	2.081×10^{-8}	2.664×10^{-8}	1.880×10^{-8}	–
10^{-1}	3.086×10^{-7}	9.961×10^{-8}	1.036×10^{-7}	1.147×10^{-7}	8.458×10^{-8}
10^{-2}	8.944×10^{-4}	3.225×10^{-7}	1.437×10^{-7}	1.389×10^{-7}	1.207×10^{-7}
10^{-3}	5.990×10^{1}	8.746×10^{-4}	3.058×10^{-7}	1.423×10^{-7}	1.503×10^{-7}
Number of subdomains (p)	4	7	10	14	17

Table 11 The maximum residual errors for Example 4.4 using different values of the initial domain length ξ, $M = 15$ and $\text{tol} = 10^{-3}$

ϵ	$M = 15,\ \mathbf{T} = 1.0$			
	$\xi = 10^{-1}$	$\xi = 10^{-2}$	$\xi = 10^{-3}$	$\xi = 10^{-4}$
$\theta = 0.90$				
510^{-1}	6.575×10^{-8}	8.360×10^{-7}	4.025×10^{-6}	2.574×10^{-5}
10^{-2}	2.424×10^{-9}	9.449×10^{-8}	1.434×10^{-5}	–
10^{-3}	2.072×10^{-10}	1.296×10^{-8}	2.539×10^{-5}	–
$\theta = 0.95$				
10^{0}	3.680×10^{-7}	2.609×10^{-6}	2.480×10^{-5}	1.569×10^{-4}
10^{-1}	2.811×10^{-8}	3.695×10^{-7}	3.541×10^{-6}	2.065×10^{-5}
10^{-2}	1.780×10^{-9}	2.342×10^{-8}	5.668×10^{-7}	3.567×10^{-5}
10^{-3}	7.016×10^{-11}	4.308×10^{-9}	1.880×10^{-6}	1.246×10^{-6}
Number of subdomains (p)	4	7	10	14



(a) $\epsilon = 10^{0},\ \ 10^{-1},\ \ 10^{-2}$



(b) $\epsilon = 10^{-3}$

Fig. 2 **a** The plot of the exact and approximate solutions with $\xi = 10^{-1}$, $\mathbf{T} = 35.0$, and **b** the number of adaptivity in each subdomain with $\epsilon = 10^{-3}$, $\mathbf{T} = 1.0$ for $\theta = 1.0$, $M = 12$

Example 4.5 Next, we evaluate the performance of the adaptive method for the nonlinear singularly perturbed fractional Ricatti equation, given as (Ramos 2005)

$$\epsilon\, {}_0^C D_t^{\theta} u(t) + \frac{1}{80} u(t)\big(u(t) - 20\big) = 0, \quad \text{for} \quad 0 < \theta \leq 1, \quad t \in \mathbb{R}^+, \tag{35}$$

with $u(0) = 1$. The closed-form solution when $\theta = 1$ is $u(t) = \dfrac{20}{1 + 19e^{-\frac{t}{4\epsilon}}}$.

Figure 2 demonstrates the accuracy of the adaptive method in capturing the dynamics of Example 4.5, where the solution plateaus at $u(t) = 20$. As the singularly perturbed parameter decreases, the solution reveals an interior layer near the left boundary. In this example, the number of subdomains the adaptive method uses fluctuates with changes in ϵ. Table 12 presents the maximum absolute errors obtained by the adaptive multidomain method when approximating Equation (35) with $\theta = 1.0$. Notably, the error worsens as the perturbation parameter approaches zero. However, reducing the initial step size ξ improves the error, indicating increased accuracy in the approximate solution. In a multidomain method with a uniform step size of $\xi = 10^{-4}$, the domain $[0, 1]$ would be partitioned into $100,000$ subdomains. In contrast, the adaptive technique resolves the problem using only 42 subdomains, demonstrating its computational efficiency.

5 Conclusion

This study examined the performance of an adaptive multidomain pseudospectral method for solving singularly perturbed fractional differential equations. These equations involve derivatives of arbitrary non-integer order with singular perturbations, where small parameters influence the behavior of the solution, often leading to solutions with steep gradients. The adaptive method is constructed by projecting the solution of the differential equation onto the space of Chebyshev polynomials and using

Table 12 The maximum absolute errors for Example 4.5 using different values of the initial domain length ξ and $M = 12,\ \theta = 1.0,\ \text{tol} = 10^{-3}$

ϵ	$M = 12,\ \theta = 1.0,\ \mathbf{T} = 1.0$				
	$\xi = 10^{-1}$	$\xi = 10^{-2}$	$\xi = 10^{-3}$	$\xi = 10^{-4}$	$\xi = 10^{-5}$
10^{0}	1.98×10^{-11}, $p = 4$	2.19×10^{-11}, $p = 7$	3.37×10^{-11}, $p = 10$	6.53×10^{-11}, $p = 14$	5.37×10^{-11}, $p = 17$
10^{-1}	1.73×10^{-9}, $p = 4$	1.84×10^{-9}, $p = 7$	1.05×10^{-9}, $p = 10$	1.15×10^{-9}, $p = 14$	2.64×10^{-9}, $p = 17$
10^{-2}	1.67×10^{-4}, $p = 4$	1.21×10^{-5}, $p = 7$	1.03×10^{-6}, $p = 10$	9.49×10^{-6}, $p = 14$	5.53×10^{-6}, $p = 17$
10^{-3}	2.79×10^{-1}, $p = 30$	1.85×10^{-4}, $p = 31$	1.27×10^{-5}, $p = 30$	9.56×10^{-7}, $p = 36$	1.13×10^{-7}, $p = 42$

the Gauss–Lobatto points to collocate within each domain. The method's accuracy is assessed through various linear and nonlinear differential equations. The findings of this study reveal the following key points:

- The adaptive multidomain scheme demonstrates superior accuracy and computational efficiency performance compared to multidomain with fixed step sizes.
- In the case of differential equations with interior layers, the adaptive multidomain scheme performed reasonably well, effectively capturing the solution's complex behavior while maintaining accuracy across the layers.
- As the singularly perturbed parameter approaches zero, the adaptive scheme successfully handles the presence of interior layers, showcasing its ability to adjust to regions of rapid variation in the solution, ensuring accurate approximation in these regions.
- Although the adaptive method generally enhances accuracy and improves computational efficiency, the relationship between the initial domain length and the accuracy of the solution remains uncertain, as some problems considered in this study required a large initial step length and others required a smaller step length to achieve the desired accuracy. Further investigation is needed to understand this balance fully.
- For the problems considered in this study, using a small tolerance value often results in residual errors that are approximately as small as those obtained with a more significant tolerance.

The adaptive multidomain scheme is generally effective for solving singularly perturbed fractional differential equations. In the future, we aim to broaden the applicability and improve the performance of the proposed method through two key objectives:

- Refinement of the adaptive strategy: We will focus on improving the core algorithm by developing more sophisticated error estimators and mesh-refinement criteria. The goal is to achieve higher levels of accuracy while maintaining or even reducing computational cost.
- Extension to new problem classes: We plan to adapt the technique to address a wider class of challenging fractional differential equations, notably singularly

perturbed fractional boundary value problems and fractional partial differential equations.

Successfully adapting the method to boundary value problems will require a fundamental shift from the way the continuity condition is currently enforced across the subdomains. While initial-value problems ensure continuity sequentially from one subdomain to the next, boundary value problems involve handling the subdomains globally. This will involve formulating and solving the problem globally to ensure that the boundary conditions at both ends of the domain are satisfied in a single, unified computational step.

Author Contributions YOT was involved in conceptualization, initial draft and writing, methodology, software. OO helped in conceptualization, methodology, review and editing, supervision. SDO contributed to conceptualization, review and editing, supervision, methodology, software.

Funding Open access funding provided by Rhodes University. This work is based on the research supported in part by the National Research Foundation of South Africa (Ref Number TTK2204163593). Yusuf Olatunji Tijani acknowledges the support of Rhodes University.

Data Availability No datasets were generated or analyzed during the current study.

Code Availability The code used to generate the results of this study can be made available upon request.

Declarations

Conflict of interest The authors declare that there are no conflict of interest. The findings and conclusions expressed in this article are solely those of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Alsuyuti, M.M., Doha, E.H., Bayoumi, B.I., Ezz-Eldien, S.S.: Robust spectral treatment for time-fractional delay partial differential equations. Comput. Appl. Math. **42**, 63–73 (2023)

Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A.: Spectral Methods: Fundamentals in Single Domains. Springer, Berlin (2006)

Caputo, M.: Linear models of dissipation whose q is almost frequency independent-II. Geophys. J. R. Astr. Soc **13**, 529–539 (1967)

Diethelm, K., Freed, A.D.: On the solution of nonlinear fractional-order differential equations used in the modeling of viscoplasticity. In: Scientific Computing in Chemical Engineering II, pp. 217–224 (1999)

Doha, E.H., Bhrawy, A.H., Ezz-Eldien, S.S.: Efficient Chebyshev spectral methods for solving multi-term fractional orders differential equations. Appl. Math. Model. **35**(12), 5662–5672 (2011)

Endrie, N.A., Duressa, G.F.: Numerical method for second order singularly perturbed delay differential equations with fractional order in time via fitted computational method. Partial Differ. Equ. Appl. Math. **10**, 100717 (2024)

Evans, L.C.: Partial Differential Equations. American Mathematical Society, Providence (1998)

Ezz-Eldien, S.S.: On solving systems of multi-pantograph equations via spectral tau method. Appl. Math. Comput. **321**, 63–73 (2018)
Ezz-Eldien, S.S., Doha, E.H.: Fast and precise sepctral method for solving pantograph type Volterra integro-differential equations. Numer. Alogorithms **81**, 57–77 (2023)
Gander, W., Gautschi, W.: Adaptive quadrature revisited. BIT Numer. Math. **40**, 84–101 (2000)
Garrappa, R.: Numerical solution of fractional differential equations: a survey and a software tutorial. Mathematics **6**(2), 16 (2018)
Gottlieb, S., Gottlieb, D.: Spectral methods. Comput. Methods Appl. Mech. Eng. **4**(9), 7504 (2009)
Hans-Gorg, R., Martin, S., Lutz, T.: Robust numerical methods for singularly perturbed differential equations: convection-diffusion-reaction and flow problems. Springer Ser. Comput. Math. **257**, 1–604 (1994)
Huseynov, I.T., Ahmadova, A., Fernandez, A., Mahmudov, N.I.: Explicit analytical solutions of incommensurate fractional differential equation systems. Appl. Math. Comput. **390**, 125590 (2021)
Kadalbajoo, M.K., Patidar, K.C.: A survey of numerical techniques for solving singularly perturbed ordinary differential equations. Appl. Math. Comput. **130**, 457–510 (2002)
Li, C., Zeng, F., Liu, F.: Spectral approximation to the fractional integral and derivative. Fract. Calc. Appl. Anal. **15**(3), 383–406 (2012)
Logan, J.D.: An Introduction to Nonlinear Partial Differential Equations. Wiley, New York (1994)
Motsa, S.: A new piecewise-quasilinearization method for solving chaotic systems of initial value problems. Cent. Eur. J. Phys. **10**(4), 936–946 (2012)
Motsa, S.S., Dlamini, P., Khumalo, M.: A new multistage spectral relaxation method for solving chaotic initial value systems. Nonlinear Dyn. **72**, 265–283 (2013)
Ninghu, S.: Fractional Calculus for Hydrology, Soil Science and Geomechanics. Taylor & Francis, New York (2020)
Nnakenyi, C.A., Weideman, J.A.C., Hale, N.: Spectral methods for fractional differential equations. AIMS Master Thesis (2015)
Oloniiju, S.D., Goqo, S.P., Sibanda, P.: A Chebyshev spectral method for heat and mass transfer in MHD nanofluid flow with space fractional constitutive model. Front. Heat and Mass Transf. **13**, 3–14 (2019)
Oloniiju, S.D., Goqo, S.P., Sibanda, P.: A pseudo-spectral method for time distributed order two-sided space fractional differential equations. Taiwan. J. Math. **25**(5), 959–979 (2021)
Oloniiju, S.D., Mukwevho, N., Tijani, Y.O., Otegbeye, O.: Chebyshev pseudospectral method for fractional differential equations in non-overlapping partitioned domains. Appl. Math. **4**(3), 950–974 (2024)
Otegbeye, O.: On paired decoupled quasi-linearization methods for solving nonlinear systems of differential equations that model boundary layer fluid flow problems. Ph.D. thesis (2018)
Owolabi, K., Atangana, A.: Numerical Methods for Fractional Differentiation. Springer, Berlin (2019)
Peter, D., Martin, W.: Adaptive Numerical Solution of PDEs. De Gruyter, Berlin (2012)
Podlubny, I.: Fractional Differential Equations. B Academic Press, San Diego. (1999)
Qureshi, S., Akanbi, M.A., Shaikh, A.A., Wusu, A.A., Ogunlaran, O.M., Mahmoud, W., Osman, M.S.: A new adaptive nonlinear numerical method for singular and stiff differential problems. Alex. Eng. J. **74**, 585–597 (2023)
Ramos, J.I.: Linearization techniques for singularly-perturbed initial-value problems of ordinary differential equations. Appl. Math. Comput. **163**, 1143–1163 (2005)
Rossikhin, Y.A., Shitikova, M.V.: Applications of fractional calculus to dynamic problems of linear and nonlinear hereditary mechanics of solids. Appl. Mech. Rev. **50**(1), 15–67 (1997)
Samuel, F.M., Motsa, S.S.: Solving hyperbolic partial differential equations using a highly accurate multidomain bivariate spectral collocation method. Wave Motion **88**, 57–72 (2019)
Sharma, K.K., Rai, R., Patidar, K.C.: A review on singularly perturbed differential equation with turning points and interior layers. Appl. Math. Comput. **219**(3), 10575–10609 (2013)
Shen, J., Tang, T., Wang, L.-L.: Spectral Methods: Algorithms, Analysis and Applications, vol. 41. Springer, Berlin (2011)
Sikora, B.: Remarks on the caputo fractional derivative. Matematyka Informatyka Na Uczelniach Technicznych (MINUT) **5**, 76–84 (2023)
Sweilam, N.H., Khader, M.M.: Efficient chebyshev spectral methods for solving multi-term fractional orders differential equations. ANZIAM J. **51**(4), 464–475 (2010)
Trefethen, L.N.: Spectral Methods in MATLAB. Society for Industrial and Applied Mathematics, Providence (2000)

Xu, Q., Hesthaven, J.S.: Stable multi-domain spectral penalty methods for fractional partial differential equations. J. Comput. Phys. **257**, 241–258 (2014)
Zayernouri, M., Em Karniadakis, G.: Fractional spectral collocation method. SIAM J. Sci. Comput. **36**(1), A40–A62 (2014)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.