

APPLICATION OF LDG SCHEME TO SOLVE SEMI-DIFFERENTIAL EQUATIONS

Mohammad Izadi

*Department of Applied Mathematics, Faculty of Mathematics and Computer
Shahid Bahonar University of Kerman, Kerman, Iran
izadi@uk.ac.ir*

Received: 12 June 2019; Accepted: 22 November 2019

Abstract. In the current work, we investigate a technique based on discontinuous Galerkin method for the numerical approximation of semi-differential equations with Caputo's fractional derivative. In this approach, using the natural upwind fluxes enables us to solve the model problem element by element locally in each subintervals and there is no need to solve a full global matrix. Numerical experiments are given to verify the efficiency and accuracy of the proposed method. Numerical solutions are compared with the exact solutions as well as the numerical solutions obtained by other available well-established computational procedures. The results show that the LDG method is more accurate for solving this class of differential equation with relatively low degrees of polynomials and number of elements.

MSC 2010: 26A33, 65M60, 34A08

Keywords: Caputo fractional derivative, local discontinuous Galerkin method, semi-differential equations

1. Introduction

Recent years have seen a growing interest in fractional differential equations due to their numerous applications, specially in the modelling of complex systems in numerous physical phenomena in science and engineering [1, 2]. Indeed, in modelling many real world problems fractional-order models can provide a more realistic insight than integer-order models. This is because of the fact that fractional derivatives and integrals enable the description of the memory properties of various materials and processes.

The goal of this paper is to develop the local discontinuous Galerkin method for the so-called semi-differential equations of order $n \in \mathbb{N}$, see [2–6]. They have the following form

$$[D^{\frac{n}{2}} + c_{n-1}D^{\frac{n-1}{2}} + \dots + c_0D^0]y(t) = f(t), \quad t \in \Omega_T := [0, T], \quad (1a)$$

where c_0, \dots, c_{n-1} are real constants and $f(t) : \Omega_T \rightarrow \mathbb{R}$ is a given function.

The equation is subjected to the following initial conditions

$$y(0) = \omega_0, \quad y'(0) = \omega_1, \quad \dots \quad y^{(m-1)}(0) = \omega_{m-1}, \quad m-1 < \frac{n}{2} \leq m, \quad (1b)$$

where $\omega_0, \omega_1, \dots, \omega_{m-1}$ are given real numbers. Here, D^m is the usual integer differential operator of order m , D^α denotes the standard Caputo fractional derivative operator of order $\alpha \notin \mathbb{N}$ and defined

$$D^\alpha y(t) = J^{m-\alpha} D^m y(t), \quad m-1 < \alpha < m, \quad m \in \mathbb{N}, \quad (2)$$

where J^α is the Riemann-Liouville fractional integral operator of order $\alpha > 0$ and is defined as

$$J^\alpha(t) = \frac{1}{\Gamma(\alpha)} \int_0^t \frac{y(s)}{(t-s)^{1-\alpha}} ds, \quad t > 0,$$

and $\Gamma(\cdot)$ is the well-known Gamma function. In [2, 7], the problem of existence and the uniqueness of the fractional differential equations (1a) subjected to initial conditions is investigated. The study of initial-value problem (1a)-(1b) has now been considered via different models that naturally appear in an increasing number of problems of fluid mechanics. Among others, we emphasize the single degree-of-freedom spring-mass-damper system whose dynamics is described by the following fractional differential equation [5, 8]

$$[mD^2 + cD^{\frac{1}{2}} + kD^0]y(t) = f(t), \quad (3)$$

where m , c , and k represent the mass, damping coefficient, and stiffness, respectively, $f(t)$ is the externally applied force, and $y(t)$ is the displacement. Another example is the Bagely-Torvik equation has the form [9, 10]

$$[mD^2 + 2A\sqrt{\rho\mu}D^{\frac{3}{2}} + kD^0]y(t) = f(t), \quad (4)$$

which describes the motion of an immersed plate bounded in a Newtonian fluid. Here, the function $y(t)$ describes the displacement of the plate of mass m , A denotes the surface area of the plate, and μ and ρ are the viscosity and density of the fluid in which the plate is immersed. Moreover, k is the stiffness of the spring to which the plate is attached and $f(t)$ is a known function representing the external force.

Several numerical and (semi-) analytical techniques are examined for solving the initial-value problem (1a)-(1b). A survey of some of these methods is given in the book [2]. In [6] and [5] a scheme based on the collocation spline procedure is proposed to treat semi-differential equations numerically. In [4], He's variational iteration method (VIM) is applied to solve n th order semi-differential equations. Also homotopy analysis method (HAM) is directly extended to investigate (1a) in [3] and Adomian decomposition method is proposed in [11] for (4). On the other hand, numerous attempts have been made to obtain the numerical solutions of (3) and (4). Among other, we mention the references [8] and [12–17], to name but a few.

It is mentioned in [3, 4] that the main disadvantage of the presented methods in [6] and [5] is the complex and difficult procedure. Since then in order to overcome the demerit, they applied the methods VIM and HAM to solve semi-differential equations of n th order. In this article, a different approach for the numerical solution of semi-differential equations is investigated. We use the local discontinuous Galerkin (LDG) method accurately to discretize (1a). The main idea of the LDG scheme is to rewrite a given fractional-order differential equation as a system of first-order classical ODEs and a fractional integral, then apply the discontinuous Galerkin (DG) method on the system and the fractional integral. The LDG methods for the fractional ODEs including one-term and multi-terms were first discussed in [18]. A key ingredient for the success of LDG schemes is the correct design of interface numerical fluxes.

This paper is structured as follows. In Section 2, the main idea of the LDG scheme for the semi-differential equations with initial conditions is presented. In computational Section 3, the accuracy of the proposed algorithm is verified by means of illustrative examples. Comparison of results obtained by using the LDG with the exact solutions and other well-established methods are also reported in this section. Finally, some conclusions are given in Section 4.

2. The basic idea of LDG

Our goal is here to formulate the LDG method for the initial value problems (1a). Assume that $n \geq 4$ for convenience. For this purpose, by the aid of (2) and introducing new variables

$$y_0(t) = y(t), \quad y_1(t) = Dy(t), \quad y_2(t) = D^2y(t), \quad \dots \quad y_m(t) = D^m y(t),$$

we convert (1) into the following first-order system

$$\begin{cases} y_1(t) - Dy_0(t) = 0, \\ y_2(t) - Dy_1(t) = 0, \\ \vdots \\ y_m(t) - Dy_{m-1}(t) = 0, \\ y_m(t) + (c_{n-1}J^{(\frac{1}{2})} + c_{n-2})y_{m-1}(t) + \dots + (c_1D^{\frac{1}{2}} + c_0)y_0(t) = f(t). \end{cases} \quad (5)$$

Note that in the last equation in (5), we have used the fact that n is even, otherwise, this equation needs to be modified slightly. For instance, if $n = 4$ is selected the last equation becomes

$$y_2(t) + (c_3J^{(\frac{1}{2})} + c_2)y_1(t) + (c_1D^{\frac{1}{2}} + c_0)y_0(t) = f(t),$$

while using $n = 5$ we arrive at

$$(J^{(\frac{1}{2})} + c_4)y_2(t) + (c_3J^{(\frac{1}{2})} + c_2)y_1(t) + (c_1D^{\frac{1}{2}} + c_0)y_0(t) = f(t).$$

Applying the initial conditions (1b), the system (5) is subjected to the following conditions

$$y_0(0) - \omega_0 = 0, \quad y_1(0) - \omega_1 = 0, \quad \dots \quad y_{m-1}(0) - \omega_{m-1} = 0. \quad (6)$$

To proceed, assume that M is a positive integer and let Π be a partitioning of Ω_T into M subintervals $\Pi := \{K_j = (t_{j-1}, t_j)\}_{j=1}^M$ with nodes

$$0 =: t_0 < t_1 < \dots < t_{M-1} < t_M := T.$$

Let $\Delta t_j = t_j - t_{j-1}$ for $j = 1, 2, \dots, M$ be the length of each K_j and set $\Delta t := \max\{\Delta t_j\}_{j=1}^M$. Next, we introduce the Sobolev spaces correspond to the partitioning Π as follows

$$\mathcal{W}(\Omega_T, \Pi) = \{v : \Omega_T \rightarrow \mathbb{R} \mid v|_{K_j} \in L_2(K_j), j = 1, 2, \dots, M\},$$

and

$$\mathcal{H}(\Omega_T, \Pi) = \{v : \Omega_T \rightarrow \mathbb{R} \mid v|_{K_j} \in H^1(K_j), j = 1, 2, \dots, M\},$$

where L_2 denotes the usual space of square integrable functions and H^1 is the usual Sobolev space of order one. Now, we assume that the solutions belong to corresponding spaces such that $y_i(t) \in \mathcal{H}(\Omega_T, \Pi)$ for $i = 0, 1, \dots, m-1$ except that $y_m(t) \in \mathcal{W}(\Omega_T, \Pi)$.

We now define $\mathcal{P}_p(K_j)$ as the space of polynomials of degree less than or equal to p on K_j for any given polynomial degree $p \geq 0$ and an element $K_j \in \Pi$. We restrict our approximate solutions to be in a local finite dimensional subspace $\mathcal{V}^{(p)} \subset \mathcal{H}(\Omega_T, \Pi)$. Now, we choose $\mathcal{V}^{(p)}$ to be the space of discontinuous, piecewise polynomial functions defined by

$$\mathcal{V}^{(p)} = \{v : \Omega_T \rightarrow \mathbb{R} : v|_{K_j} \in \mathcal{P}_p(K_j), j = 1, 2, \dots, M\}.$$

Since a function $v \in \mathcal{V}^{(p)}$ is allowed to be discontinuous in t at time level t_j , we let v_j^- and v_j^+ denote the left and right-sided limits of v at t_j

$$v_j^+ = v^+(t_j) = v(t_j^+) := \lim_{s \rightarrow 0^+} v(t_j + s), \quad v_j^- = v^-(t_j) = v(t_j^-) := \lim_{s \rightarrow 0^-} v(t_j + s).$$

On the element K_j we use the quantities $\mathcal{Y}_i(t) \in \mathcal{V}^{(p)}$ for $i = 0, 1, \dots, m$ represent the computed DG approximations to the exact solutions $y_i(t)$ respectively of the system (5). To obtain the weak DG formulation for (5), we multiply the first m equations by test functions $v_i \in \mathcal{V}^{(p)}$, $i = 0, 1, \dots, m-1$ follow by integrating over K_j and then

integrating by parts to obtain that

$$\int_{K_j} \mathcal{Y}_{i+1}(t) v_i(t) dt + \int_{K_j} \mathcal{Y}_i(t) \frac{dv_i(t)}{dt} dt - \mathcal{Y}_i(t_j^-) v_i(t_j^-) + \mathcal{Y}_i(t_{j-1}^+) v_i(t_{j-1}^+) = 0, \quad (7)$$

for $i = 0, 1, \dots, m-1$. We then multiply the last equation in (5) by a test function $v_m \in \mathcal{V}^{(p)}$ and integrate over K_j . To solve the equations forward in time, we utilize the upwind flux $\mathcal{Y}_i(t_{j-1}^-)$ instead of $\mathcal{Y}_i(t_{j-1}^+)$ for $i = 0, 1, \dots, m$ in (7). Putting all together, the discrete formulation consists of determining $\mathcal{Y}_i(t) \in \mathcal{V}^{(p)}$ such that for all $v_i(t) \in \mathcal{V}^{(p)}$ for $i = 0, 1, \dots, m$ becomes

$$\begin{aligned} \int_{K_j} \mathcal{Y}_1(t) v_0(t) dt + \int_{K_j} \mathcal{Y}_0(t) \frac{dv_0(t)}{dt} dt - \mathcal{Y}_0(t_j^-) v_0(t_j^-) + \mathcal{Y}_0(t_{j-1}^-) v_0(t_{j-1}^+) &= 0, \\ \int_{K_j} \mathcal{Y}_2(t) v_1(t) dt + \int_{K_j} \mathcal{Y}_1(t) \frac{dv_1(t)}{dt} dt - \mathcal{Y}_1(t_j^-) v_1(t_j^-) + \mathcal{Y}_1(t_{j-1}^-) v_1(t_{j-1}^+) &= 0, \\ &\vdots \\ \int_{K_j} \mathcal{Y}_m(t) v_{m-1}(t) dt + \int_{K_j} \mathcal{Y}_{m-1}(t) \frac{dv_{m-1}(t)}{dt} dt - \mathcal{Y}_{m-1}(t_j^-) v_{m-1}(t_j^-) \\ &+ \mathcal{Y}_{m-1}(t_{j-1}^-) v_{m-1}(t_{j-1}^+) = 0, \end{aligned}$$

and

$$\begin{aligned} \int_{K_j} \mathcal{Y}_m(t) v_m(t) dt + c_{n-1} \int_{K_j} (J^{\frac{1}{2}} \mathcal{Y}_{m-1}(t)) v_m(t) dt + c_n \int_{K_j} \mathcal{Y}_{m-1}(t) v_m(t) dt \\ + \dots + c_1 \int_{K_j} (D^{\frac{1}{2}} \mathcal{Y}_0(t)) v_m(t) dt + c_0 \int_{K_j} \mathcal{Y}_0(t) v_m(t) dt = \int_{K_j} f(t) v_2(t) dt, \end{aligned}$$

for all $j = 1, 2, \dots, M$. These equations are also accompanied by the following initial conditions

$$\mathcal{Y}_0(t_0^-) - \omega_0 = 0, \quad \mathcal{Y}_1(t_0^-) - \omega_1 = 0, \quad \dots \quad \mathcal{Y}_{m-1}(t_0^-) - \omega_{m-1} = 0.$$

Although, on the initial step $K_1 = (t_0, t_1)$ we will use the above given initial conditions, however, on the K_j for $j \geq 2$ the values of $\mathcal{Y}_i(t_{j-1}^-), i = 0, 1, \dots, m-1$ will be extrapolated from the previous element K_{j-1} . We also emphasize that using the upwind fluxes as natural choices enable us to solve the equation interval by interval on each subinterval K_j for $j = 1, 2, \dots, M$. This implies that we just invert a local low-order $(p+1) \times (p+1)$ matrix rather than a global full matrix.

Due to the fact that the functions in $\mathcal{V}^{(p)}$ are allowed to have discontinuities across element interfaces, one can exploit various local basis for finite element approximation, see [19] for practical implementation. Below, to implement the above LDG scheme, in the numerical experiments we use the Legendre polynomials as a basis for the space $\mathcal{V}^{(p)}$. Utilizing this special orthogonal basis functions, our discrete LDG scheme will be transformed to a linear system of algebraic equations.

3. Applications

In this section, we present some results of computations using the proposed LDG schemes described in the preceding sections to test their accuracy and efficiency when applied to the semi-differential equations. To measure the accuracy of the numerical algorithms, we compute the difference between the analytic and numerical solutions. We also compare our results with existing computational procedures available in the standard literature.

Example 1 As the first example, we use $n = 4$ in (1a) with $c_3 = c_2 = 0$, $c_1 = 1/2$, and $c_0 = 1$ to get

$$\left[D^2 + \frac{1}{2}D^{\frac{1}{2}} + D^0\right]y(t) = 3 + t^2 + \frac{t^{\frac{3}{2}}}{\Gamma(\frac{5}{2})} + \frac{t^{-\frac{1}{2}}}{2\Gamma(\frac{1}{2})}, \quad 0 < t \leq 1. \quad (8)$$

The initial conditions are given as $y(0) = 1$ and $y'(0) = 0$. It can be easily verified that $y(t) = 1 + t^2$ is the exact solution of (8). This equation is known as Bagley-Torvik equation of fractional order “1/2” taken from [13]. \square

The numerical solution for this example while using $p = 2$ and $M = 1$ are depicted in Figure 1, left plot. The corresponding exact solution is plotted by solid line while the numerical solution is visualized by (coloured) curve. Note that the time step is equal to $\Delta t = 1$. On the same figure but right plot, the corresponding absolute error is shown. It can be seen from Figure 1 that the numerical solution obtained by the LDG method is in excellent agreement with the exact solution and hardly to distinguishable from each other. To see the behaviour of numerical solution more closely, we magnify these solutions at time $t = 0.630$ in Figure 1.

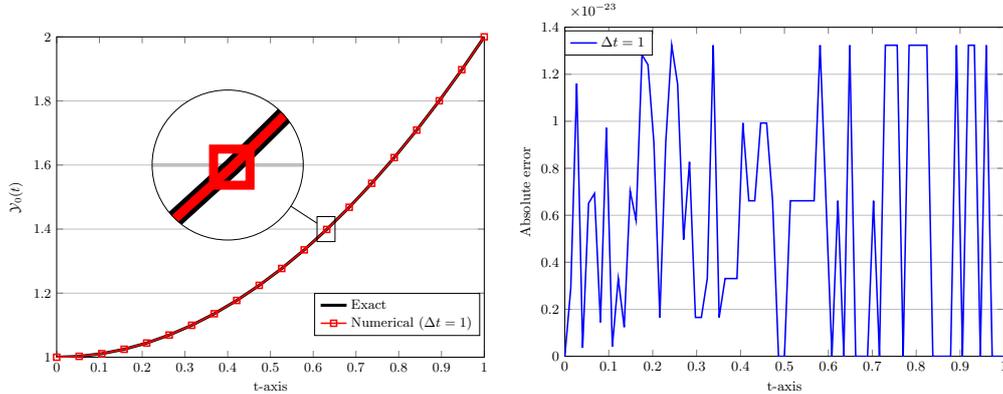


Fig. 1. Comparing the approximated LDG and exact solutions (left) and the corresponding absolute error (right) for $M = 1$, $p = 2$

Indeed, the approximated analytical solution $\mathcal{Y}_0(t)$ obtained by the LDG in Figure 1 for $p = 1$ and $M = 1$ on the interval $(0, 1]$ takes the form $\mathcal{Y}_0(t) = 1.0 + t^2$,

which is the exact solution. Meanwhile, the corresponding approximations for the first and the second derivatives of the solution are also obtainable. In this case, we find that

$$\begin{aligned}\mathcal{Y}_1(t) &= -5.3291 \times 10^{-15} t^2 + 2.000000000000001 t - 8.8818 \times 10^{-16}, \\ \mathcal{Y}_2(t) &= -6.5281 \times 10^{-14} t^2 + 6.8390 \times 10^{-14} t + 1.9999999999999999.\end{aligned}$$

Clearly, for this example we have $y'(t) = 2t$ and $y''(t) = 2$.

To show further the benefit of the proposed method and to validate our numerical solutions, we make a comparison in Table 1 between the LDG for different $M = 1, 2, 3$ and the reproducing kernel algorithm (RKA) [13].

Table 1. Comparison of absolute errors for $\mathcal{Y}_0(t)$ in Example (1) for $M = 1, 2, 3$, $p = 2$

t	$M = 1$			$M = 2$		$M = 3$		RKA
	Exact	Numerical	Error	Numerical	Error	Numerical	Error	Error
0.0	1.00	1.00	0	1.00	0	1.00	0	0
0.1	1.01	1.01	8.27×10^{-27}	1.01	8.27×10^{-27}	1.01	8.27×10^{-27}	1.93×10^{-12}
0.2	1.04	1.04	3.31×10^{-26}	1.04	3.31×10^{-26}	1.04	3.31×10^{-26}	3.16×10^{-11}
0.3	1.09	1.09	1.32×10^{-25}	1.09	1.32×10^{-25}	1.09	1.32×10^{-25}	3.68×10^{-10}
0.4	1.16	1.16	1.32×10^{-25}	1.16	1.32×10^{-25}	1.16	6.75×10^{-18}	3.66×10^{-09}
0.5	1.25	1.25	0	1.25	0	1.25	1.58×10^{-17}	3.30×10^{-09}
0.6	1.36	1.36	5.29×10^{-25}	1.36	5.58×10^{-18}	1.36	1.54×10^{-17}	2.75×10^{-09}
0.7	1.49	1.49	1.85×10^{-24}	1.49	7.70×10^{-17}	1.49	1.85×10^{-16}	2.10×10^{-10}
0.8	1.64	1.64	5.29×10^{-25}	1.64	1.90×10^{-16}	1.64	2.35×10^{-16}	1.40×10^{-11}
0.9	1.81	1.81	2.12×10^{-24}	1.81	3.45×10^{-16}	1.81	2.47×10^{-16}	7.00×10^{-12}
1.0	2.00	2.00	0	2.00	5.41×10^{-16}	2.00	2.20×10^{-16}	0

Example 2 As the second example, we consider the following equation [3, 4, 6]

$$[D^2 - 2D + D^{\frac{1}{2}} + D^0]y(t) = t^3 + \frac{16}{5\Gamma(\frac{1}{2})}t^{\frac{5}{2}} - 6t^2 + 6t. \quad (9)$$

This equation is subjected with the initial conditions $y(0) = 0$ and $y'(0) = 0$ and the computational domain is again the interval $[0, 1]$. It is not a difficult task to show that $y(t) = t^3$ is the exact solution of (9). \square

First, we use the step-size $\Delta t = 1$ and the degree of polynomial is $p = 3$. In Table 2, we present the exact results correspond to $y(t)$ and its numerical approximation $\mathcal{Y}_0(t)$ for various points $t = 0, 0.1, \dots, 0.9, 1$ obtained by the LDG scheme while using different values of $M = 1, 2$. To measure the high accuracy of the method, the absolute errors as well as the relative errors are calculated in this table. Clearly, using only $M = 1$ and $M = 2$ intervals, the desired accuracy is achieved and one observes an excellent agreement between the exact and numerical solutions.

Table 2. Comparison of numerical solutions, absolute, and relative errors for $\mathcal{U}_0(t)$ in Example (2) for $M = 1, 2$ and $p = 3$

t	Numerical		Absolute errors		Relative errors		Exact
	$M = 1$	$M = 2$	$M = 1$	$M = 2$	$M = 1$	$M = 2$	
0.0	0.000	0.000	0	0	—	—	0.000
0.1	0.001	0.001	0	0	0	0	0.001
0.2	0.008	0.008	0	0	0	0	0.008
0.3	0.027	0.027	0	0	0	0	0.027
0.4	0.064	0.064	0	0	0	0	0.064
0.5	0.125	0.125	0	0	0	0	0.125
0.6	0.216	0.216	0	1.03×10^{-15}	0	4.79×10^{-15}	0.216
0.7	0.343	0.343	0	8.70×10^{-16}	0	2.54×10^{-15}	0.343
0.8	0.512	0.512	0	2.17×10^{-16}	0	4.23×10^{-16}	0.512
0.9	0.729	0.729	0	5.29×10^{-16}	0	7.25×10^{-16}	0.729
1.0	1.000	1.000	0	9.67×10^{-16}	0	9.67×10^{-16}	1.000

To further justify the results shown in Table 2, we compute the approximate solutions of the problem in each subinterval. In the LDG scheme we are able to find not only $\mathcal{U}_0(t)$, but also its first and second derivatives $\mathcal{U}_1(t)$ and $\mathcal{U}_2(t)$ during computations. Using $M = 1$ and $p = 3$, we find that

$$\mathcal{U}_0(t) = t^3,$$

$$\mathcal{U}_1(t) = 9.3814 \times 10^{-14} t^3 + 2.99999999999986 t^2 + 5.6288 \times 10^{-14} t - 6.9222 \times 10^{-14},$$

$$\mathcal{U}_2(t) = 1.3300 \times 10^{-12} t^3 - 2.0114 \times 10^{-12} t^2 + 6.000000000000081 t - 6.9222 \times 10^{-14}.$$

Similarly, if one utilizes $M = 2$ with the same p as above, we arrive at for $t \in K_0$

$$\mathcal{U}_0(t) = t^3,$$

$$\mathcal{U}_1(t) = -9.9920 \times 10^{-15} t^3 + 3.00000000000001 t^2 - 1.4988 \times 10^{-15} t + 6.2450 \times 10^{-17},$$

$$\mathcal{U}_2(t) = 1.4655 \times 10^{-12} t^3 - 1.2836 \times 10^{-12} t^2 + 6.000000000000031 t - 1.6848 \times 10^{-14},$$

and for $t \in K_1$

$$\mathcal{U}_0(t) = 1.000000000000007 t^3 - 1.6393 \times 10^{-13} t^2 + 1.2713 \times 10^{-13} t - 3.0574 \times 10^{-14},$$

$$\mathcal{U}_1(t) = -6.9602 \times 10^{-13} t^3 + 3.000000000000184 t^2 - 1.5956 \times 10^{-12} t + 4.4780 \times 10^{-13},$$

$$\mathcal{U}_2(t) = -5.1549 \times 10^{-11} t^3 + 1.1942 \times 10^{-10} t^2 + 5.99999999990979 t + 2.2154 \times 10^{-11}.$$

Obviously, all approximated analytical solutions using step $\Delta t = 0.2$ are perfectly aligned with the corresponding exact solutions up to machine epsilon.

Next, we verify the high accuracy of the proposed LDG method compared to existing numerical methods in standard literature. For this purpose, we exploit the analogous results of [4] for the underlying He's variational iteration method (HVIM) and the homotopy perturbation method (HPM) as well as homotopy analysis method (HAM) from [3]. In this comparison, we present the absolute errors $|y(t) - \mathcal{Y}_0(t)|$ at different $t \in [0, 1]$ as shown in Table 3. In our LDG method we take $p = 3$. A similar comparison for $|y'(t) - \mathcal{Y}_1(t)|$ and $|y''(t) - \mathcal{Y}_2(t)|$ is made in Table 4.

Table 3. Comparison of absolute errors for $\mathcal{Y}_0(t)$ in Example (2) for $M = 1, 2, 3$, and $p = 3$

t	LDG			HVIM	HAM	HPM
	$M = 1$	$M = 2$	$M = 3$			
0.1	0	0	0	$0.254373331 \times 10^{-4}$	3.5997×10^{-13}	2.5435×10^{-16}
0.2	0	0	0	$0.413093948 \times 10^{-3}$	1.5021×10^{-12}	4.2573×10^{-13}
0.3	0	0	0	$0.211853620 \times 10^{-2}$	9.6606×10^{-12}	3.1145×10^{-11}
0.4	0	0	1.16×10^{-15}	$0.677080535 \times 10^{-2}$	2.9515×10^{-11}	6.3416×10^{-10}
0.5	0	0	1.37×10^{-16}	$0.166874021 \times 10^{-1}$	4.1907×10^{-11}	6.4131×10^{-09}
0.6	0	1.03×10^{-15}	3.00×10^{-16}	$0.348739983 \times 10^{-1}$	3.3163×10^{-10}	4.1671×10^{-08}
0.7	0	8.70×10^{-16}	6.25×10^{-15}	$0.650089365 \times 10^{-1}$	5.0078×10^{-10}	1.9955×10^{-07}
0.8	0	2.17×10^{-16}	1.23×10^{-16}	$0.111412924 \times 10^{+0}$	2.0880×10^{-09}	7.6424×10^{-07}
0.9	0	5.29×10^{-16}	3.96×10^{-15}	$0.179004637 \times 10^{+0}$	8.0036×10^{-09}	2.4674×10^{-06}
1.0	0	9.67×10^{-16}	9.79×10^{-16}	$0.273243226 \times 10^{+0}$	4.4094×10^{-09}	6.9604×10^{-06}

Table 4. Comparison of absolute errors for $\mathcal{Y}_1(t)$ and $\mathcal{Y}_2(t)$ in Example (2) for various $M = 1, 2, 3$, and $p = 3$

t	$ \mathcal{Y}_1(t) - y'(t) $			$ \mathcal{Y}_2(t) - y''(t) $		
	$M = 1$	$M = 2$	$M = 3$	$M = 1$	$M = 2$	$M = 3$
0.0	4.6907×10^{-15}	6.2450×10^{-17}	2.4546×10^{-16}	7.3996×10^{-14}	1.6182×10^{-14}	1.3769×10^{-14}
0.1	3.7525×10^{-16}	2.2482×10^{-17}	1.0800×10^{-16}	7.2381×10^{-15}	2.7778×10^{-15}	5.8054×10^{-15}
0.2	1.6886×10^{-15}	1.7486×10^{-17}	6.8730×10^{-17}	2.5251×10^{-14}	5.7498×10^{-15}	2.8441×10^{-15}
0.3	2.0639×10^{-15}	1.7486×10^{-17}	1.9637×10^{-17}	3.1934×10^{-14}	9.5479×10^{-16}	3.4318×10^{-16}
0.4	1.3133×10^{-15}	2.2482×10^{-17}	6.5481×10^{-15}	2.1274×10^{-14}	3.3873×10^{-15}	5.2749×10^{-13}
0.5	0	6.2450×10^{-17}	1.6369×10^{-14}	1.7347×10^{-15}	9.4369×10^{-16}	5.9212×10^{-14}
0.6	1.3134×10^{-15}	2.4326×10^{-15}	1.1795×10^{-14}	1.8221×10^{-14}	1.1743×10^{-13}	5.4146×10^{-13}
0.7	2.0639×10^{-15}	6.3453×10^{-15}	8.5580×10^{-14}	3.0130×10^{-14}	1.6083×10^{-13}	5.0655×10^{-13}
0.8	1.6886×10^{-15}	7.5595×10^{-15}	1.1680×10^{-13}	2.5528×10^{-14}	1.9120×10^{-14}	1.8974×10^{-12}
0.9	3.7526×10^{-16}	5.3862×10^{-15}	2.1526×10^{-14}	4.0462×10^{-15}	1.1312×10^{-13}	2.8887×10^{-12}
1.0	4.6907×10^{-15}	4.0014×10^{-15}	7.7506×10^{-15}	6.7057×10^{-14}	1.8812×10^{-13}	6.6132×10^{-12}

It can be observed from Tables 3 and 4 that using a small number of subintervals and a low degree of polynomial is sufficient to get the desired exact solutions not only

for $y(t)$ itself, but also for $y'(t)$ and $y''(t)$. From Table 3, we can see that the absolute errors given in this paper have higher accuracy than the other results obtained by the well-established methods, i.e., HVIM, HAM, and HPM. This shows the thoroughness of the proposed method.

Example 3 As the last example arises in application, we consider the Bagley-Torvik equation of order “3/2” of the form [3, 4, 6]

$$[D^2 + D^{\frac{3}{2}} + D^0]y(t) = 2 + 4\sqrt{\frac{t}{\pi}} + t^2,$$

on the computational domain $[0, 1]$ with zero initial conditions $y(0) = 0, y'(0) = 0$. The exact solution of this fractional initial-value problem takes the form $y(t) = t^2$. \square

Figure 2 visualizes the numerical solutions as well as the corresponding absolute errors for the approximated solutions $\mathcal{Y}_0(t), \mathcal{Y}_1(t)$, and $\mathcal{Y}_2(t)$ in Example (3). In these plots we take $\Delta t = 1$ and the degree of polynomial is $p = 2$.

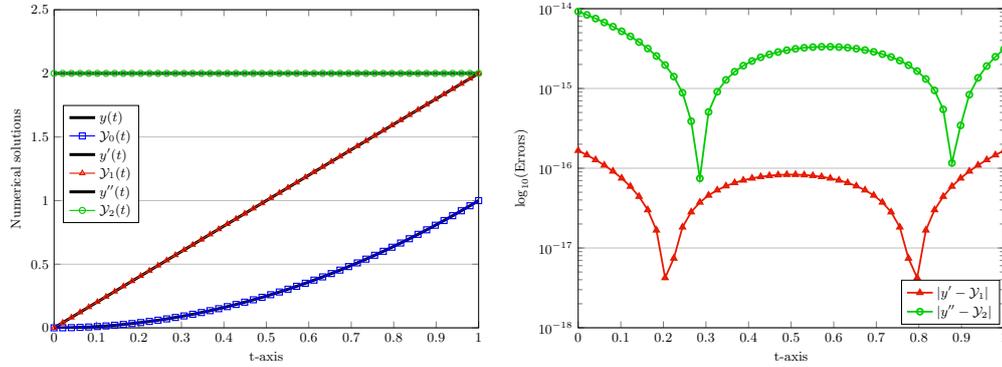


Fig. 2. Comparison of numerical and exact solutions for $\mathcal{Y}_0(t), \mathcal{Y}_1(t), \mathcal{Y}_2(t)$ (left) and the corresponding absolute errors (right) using $M = 2$ and $p = 3$

Notice in Figure 2, right plot that the plot of $|y(t) - \mathcal{Y}_0(t)|$ is omitted due to the fact that difference between the approximated and exact solutions are really zero and it is not possible to depict in log-scale. Indeed, for $M = 1$ and $p = 2$ we obtain

$$\begin{aligned} \mathcal{Y}_0(t) &= t^2, \quad \mathcal{Y}_1(t) = 9.9920 \times 10^{-16} t^2 + 2.0t + 1.6653 \times 10^{-16}, \\ \mathcal{Y}_2(t) &= 3.6970 \times 10^{-14} t^2 - 4.3077 \times 10^{-14} t + 2.000000000000001, \end{aligned}$$

which further justify the validation of the plotted curves in Figure 2.

In the next simulation, we examine the behaviour of absolute errors when using more than one subintervals. Figure 3 shows the absolute errors for two different values of $M = 1$ and $M = 2$. In both cases we utilize $p = 2$. Note that in the first intervals I_0 the approximated and exact solutions for both M are exactly the same and therefore the absolute errors in these intervals are completely zero. This is the reason why we could not show the errors in log-scale format. Finally, we make a comparison

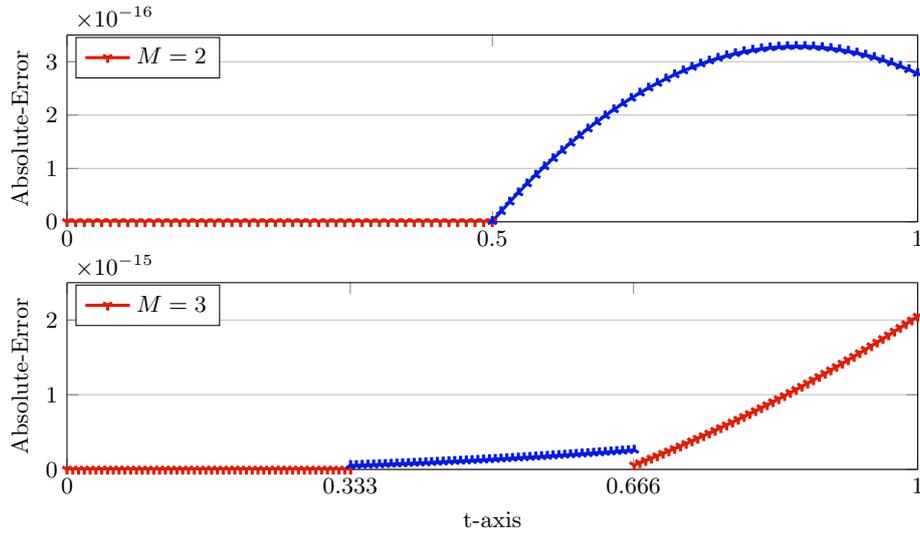


Fig. 3. Comparison of absolute errors in LDG for Example (3) with $M = 2, 3$, and $p = 2$

between our results and some of the previously published works available in literature for this example. Analogue comparison as for Example 2 is done here with the schemes HVIM, HAM, HPM. The results for various M and different $t \in [0, 1]$ are reported in Table 5. However, we use $p = 2$ here.

Table 5. Comparison of absolute errors for $\mathcal{B}_6(t)$ in Example (3) for $M = 1, 2, 3$, and $p = 2$

t	LDG			HVIM	HAM	HPM
	$M = 1$	$M = 2$	$M = 3$			
0.1	0	0	0	0.5487432×10^{-4}	2.3265×10^{-13}	4.0215×10^{-11}
0.2	0	0	0	$0.63125562 \times 10^{-3}$	1.4385×10^{-11}	5.2739×10^{-09}
0.3	0	0	0	$0.266557142 \times 10^{-2}$	6.1890×10^{-11}	9.2959×10^{-08}
0.4	0	0	8.07×10^{-17}	$0.74801219 \times 10^{-2}$	2.2736×10^{-11}	7.2230×10^{-07}
0.5	0	0	1.36×10^{-16}	$0.167959208 \times 10^{-1}$	1.3680×10^{-10}	3.5881×10^{-06}
0.6	0	1.58×10^{-16}	2.05×10^{-16}	$0.327730722 \times 10^{-1}$	3.5678×10^{-11}	1.3445×10^{-05}
0.7	0	2.65×10^{-16}	2.09×10^{-16}	$0.580653520 \times 10^{-1}$	2.6188×10^{-10}	4.1497×10^{-05}
0.8	0	3.20×10^{-16}	7.30×10^{-16}	$0.958850823 \times 10^{-1}$	4.3416×10^{-10}	1.1118×10^{-04}
0.9	0	3.25×10^{-16}	1.34×10^{-15}	$0.150076845 \times 10^{+0}$	1.0816×10^{-10}	2.6745×10^{-04}
1.0	0	2.78×10^{-16}	2.05×10^{-15}	$0.225199500 \times 10^{+0}$	7.9195×10^{-10}	5.9110×10^{-04}

4. Conclusions

In this paper, the LDG method as an efficient and accurate approximation technique has been adopted for solving semi-differential equations. Using the natural upwind fluxes and a special basis functions the problem reduced to a local low-order system of algebraic equation in each subinterval. Three illustrative examples are provided to demonstrate the validity and applicability of the method presented. Comparison with well-established computational procedures show that the LDG method produces an accurate approximation for the solution as well as its derivatives.

References

- [1] Kilbas, A., Srivastava, H., & Trujillo, J. (2006). *Theory and Applications of Fractional Differential Equations*. Amsterdam: Elsevier.
- [2] Podlubny, I. (1999). *Fractional Differential Equations*. New York: Academic Press.
- [3] Fadravi, H.H., Saberi Nik, H., & Buzhabadi, R. (2011). Homotopy analysis method based on optimal value of the convergence control parameter for solving semi-differential equations. *Journal of Mathematical Extension*, 5, 105-112.
- [4] Ghorbani, A., & Alavi, A. (2008). Application of He's variational iteration method to solve semidifferential equations of n th order. *Mathematical Problems in Engineering*, Vol. 2008, Article ID 627983, 9 pages, DOI:10.1155/2008/627983.
- [5] Hamarshdeh, M.H., & Rawashdeh, E.A. (2010). A numerical method for solution of semidifferential equations. *Matematički Vesnik*, 62(2), 117-126.
- [6] Rawashdeh, E.A. (2006). Numerical solution of semi-differential equations by collocation method. *Applied Mathematics and Computation*, 174(2), 869-876.
- [7] Diethelm, K., & Ford, N.J. (2002). Analysis of fractional differential equation. *Journal of Mathematical Analysis and Applications*, 265, 229-248.
- [8] Yuan, L., & Agrawal, O.P. (2002). A numerical scheme for dynamic systems containing fractional derivatives. *Journal of Vibration and Acoustics*, 124, 321-324.
- [9] Bagley, R.L., & Torvik, P.J. (1984). On the appearance of the fractional derivative in the behaviour of real materials. *Journal of Applied Mechanics*, 51(2), 294-298.
- [10] Bagley, R.L., & Torvik, P.J. (1983). Fractional calculus-a different approach to the analysis of viscoelastically damped structures. *The American Institute of Aeronautics and Astronautics*, 21(5), 741-748.
- [11] Saha Ray, S., & Bera, R.K. (2005). Analytical solution of the Bagley Torvik equation by Adomian decomposition method. *Applied Mathematics and Computation*, 168, 398-410.
- [12] Azizi, M-R., & Khani, A. (2017). Sinc operational matrix method for solving the Bagley-Torvik equation. *Computational Methods for Differential Equations*, 5(1), 56-66.
- [13] Arqub, O.A., & Maayah, B. (2018). Solutions of Bagley-Torvik and Painlevé equations of fractional order using iterative reproducing kernel algorithm with error estimates. *Neural Computing & Applications*, 29(5), 1465-1479.
- [14] Cenesiz, Y., Keskin, Y., & Kurnaz, A. (2010). The solution of the Bagley-Torvik equation with the generalized Taylor collocation method. *Journal of Franklin Institute*, 347(2), 452-466.
- [15] Diethelm, K., & Ford, N.J. (2002). Numerical solution of the Bagley-Torvik equation. *BIT Numerical Mathematics*, 42, 490-507.
- [16] Rahimkhani, P., & Ordokhani, Y. (2018). Application of Müntz-Legendre polynomials for solving the Bagley-Torvik equation in a large interval. *SeMA Journal*, 75(3), 517-533.

-
- [17] Srivastava, H.M., Shah, F.A., & Abass, R. (2019). An application of the Gegenbauer wavelet method for the numerical solution of the fractional Bagley-Torvik equation. *Russian Journal of Mathematical Physics*, 26(1), 77-93.
- [18] Deng, W., & Hesthaven, S. (2015). Local discontinuous Galerkin method for fractional ordinary differential equations. *BIT Numerical Mathematics*, 55, 967-985.
- [19] Negar, M.R., Izadi, M., & Saeedi, H. (2019). Numerical solution of fractional ordinary differential equations by local discontinuous Galerkin method. *Wavelet and Linear Algebra*, 5(3 (Special issue)), 1-25, DOI:10.22072/wala.2018.82269.1161.