Journal of Fractional Calculus and Applications
Vol. 12(3). No.12, pp. 1-13
1st. Inter. E-Conf. in Math. Sciences and Fractional Calculus(ICMSFC Feb 2021).
ISSN: 2090-5858.
http://math-frac.org/Journals/JFCA/

EFFICIENCY IMPROVEMENT OF FRACTIONAL EULER METHOD BY A NEW MEMORY SELECTION METHOD

L. W. SOMATHILAKE

Abstract. Fractional operators are not local operators and hence solutions of time fractional order model at a particular time (say t_1) depends on the past ($t \in [0, t_1)$) values of the solution called memory. As a result of that, numerical integration of time fractional differential equations (FDEs) on large time intervals or finer meshes are time consuming processes. Reducing the computational cost of time integration of time FDEs is a challenge and the aim of this paper is to improve the efficiencies of the fractional Euler method. Two established memory selection methods applying for numerical schemes of FDEs are the short (fixed) memory method (SMM) and the full memory method (FMM). In SMM computational cost is less but computational error is higher due to cut off the tail of the memory at each time step. The computational error of FMM is less but the computational cost is higher. In the proposed method, the number of memory points in the past are chosen such a way that linearly decreasing along the tail of the memory (hereinafter, say *Linearly Decreasing Memory Method* (LDMM) for the proposed method). This paper considers fractional Euler numerical scheme with three memory selection methods FMM, LDMM, and SMM, (denoted by FEM-FM, FEM-LDM, and FEM-SM respectively), and compares these numerical schemes by simulating some FDEs whose analytical solutions are known. It is observed that the experimental order of convergence (EOC) of FEM-FM and FEM-LDM are almost the same. Also, the computational cost of FEM-LDM is less than that of FEM-FM, and FEM-LDM is more accurate than that of FEM-SM. Therefore, the proposed method is more suitable than the short memory method for numerical integration of fractional differential equations. Also, FEM-LDM is more suitable than FEM-FM for long-time integrations and integrations on finer meshes as the computational cost of FEM-LDM is less than that of FEM-FM.

1. Introduction

Fractional calculus and fractional differential equations have been popularizing as its powerful applications in various areas. A large number of mathematical models based on fractional differential equations have been developed in various

2010 *Mathematics Subject Classification.* 26A33, 34D30.
Key words and phrases. Fractional Derivatives, Fractional Eular method, Full Memory method, Short Memory method.
Submitted March. 10, 2021.

areas such as mechanics ([1], [2], [8]), chemistry ([3], engineering ([4], [5], [6], [9]), medicine ([10], [11], [7]), biology [12], physics ([13], [14]) control theory [2], finance [15] etc.

1.1. **Fractional derivatives.** There is no accepted unique definition for fractional derivative. Different definitions for fractional derivatives. Riemann–Liouville definition and Caputo definitions are the popular definitions.

Definition 1.1 (Riemann–Liouville integral)**.** *[2] The Riemann–Liouville fractional integral of order $\alpha \geq 0$ of a function $u(t)$ is defined as*

$$J^{\alpha}u(t) = \begin{cases} \dfrac{1}{\Gamma(\alpha)} \displaystyle\int_0^t (t-s)^{\alpha-1}u(s)ds, & \alpha > 0, \\ u(t), & \alpha = 0. \end{cases}$$

Fractional order integral satisfies following properties

(1) $J^{\alpha}J^{\beta}u(t) = J^{\alpha+\beta}u(t)$,
(2) $J^{\alpha}J^{\beta}u(t) = J^{\beta}J^{\alpha}u(t)$.
(3) $J^{\alpha}t^{\gamma} = \dfrac{\Gamma(\gamma+1)}{\Gamma(\gamma+\alpha+a)}t^{\gamma+\alpha}$.

Definition 1.2 (Riemann–Liouville derivative)**.** *[2] The Riemann– Liouville derivative of order γ, ${}_a^RD_t^{\gamma}u(t)$, is defined as ${}_a^RD_t^{\gamma}u(t) = D^nJ^{(n-\gamma)}u(t)$. Where n is the smallest integer greater than γ. That is*

$${}_a^RD_t^{\gamma}u(t) = \begin{cases} \dfrac{1}{\Gamma(n-\gamma)}\dfrac{d^n}{dt^n}\displaystyle\int_a^t (t-\tau)^{n-\gamma-1}u(\tau)d\tau, & n-1 \leq \gamma < n, \\ \\ \dfrac{d^nu(t)}{dt^n}, \quad \gamma = n \in \mathbb{N}, \end{cases} \tag{1}$$

where, $\Gamma(z)$ $(z \in \mathbb{C})$ denotes the Euler Gamma function defined by $\Gamma(z) = \int_0^{\infty} x^{z-1}e^{-x}dx$, γ is the order of the derivative, a is the initial value of function u.

Definition 1.3 (Caputo fractional derivative)**.** *[2] The Caputo fractional derivative operator of order $\gamma > 0$, ${}_a^CD_t^{\gamma}u(t)$, is defined as ${}_a^CD_t^{\gamma}u(t) = J^{(n-\gamma)}D^{\gamma}u(t)$. That is*

$${}_a^CD_t^{\gamma}u(t) = \begin{cases} \dfrac{1}{\Gamma(n-\gamma)}\displaystyle\int_a^t (t-\tau)^{n-\gamma-1}\dfrac{d^nu(\tau)}{d\tau^n}d\tau, & n-1 < \gamma < n, \\ \\ \dfrac{d^nu(t)}{dt^n}, \quad \gamma = n \in \mathbb{N}. \end{cases} \tag{2}$$

Fractional derivatives do not have convenient physical meanings as integer order derivatives. However, some geometric and physical interpretations of Riemann-Liouville and Caputo fractional derivatives are reported in [16].

In most of the physical processes, the next state depends not only on the current state but also on its past states starting from the initial time. This behavior is known as the non-local behavior of the physical systems. Fractional differential operators are non-local. That is those operators taken into account the fact that the next state of a system not only depends on its current state but also on its past states starting from the initial time. In ordinary differential operators, it is considered that the next state of a system depends only on states in a neighborhood of the present state. In other words, fractional differential operators are non-local while ordinary differential operators are local. When modelling real-world processes considering this non-local property is very important as this property bring the models closer to reality. Therefore, fractional differential equations models are more realistic than ordinary differential equations models. However, the non-local property of fractional differential operators causes to increase in the complexity and computation cost of numerical integration of FDEs.

In the literature, several analytical techniques to solve fractional differential equations are reported. Some of such analytical techniques are Laplace transform [17, 18, 19, 9], Fourier transform [18, 9], Mellin transform [9, 20] etc. These techniques are limited to solve linear FDEs with constant coefficients only and do not work for non-linear FODEs. Therefore, numerical techniques play an important role in finding approximate solutions of non-linear FODEs.

There are several numerical methods for solving fractional differential equations. Some of those methods are Fractional Euler method [21], Fractional Adams method [21], Block by Block method [22, 23], Predictor-Corrector method [4, 24] etc.

In [25] predictor-corrector method has been implemented for multi-term fractional differential equations. In [26] a detailed error analysis for a fractional Adams method with graded meshes has been done. In [27] an efficient numerical scheme for fractional differential equations considering finite difference formula for Caputo fractional differential operator is proposed. In that method number of memory points in the past are chosen randomly and exponentially decreasing along the tail of the memory.

This paper investigates the efficiencies and computation cost of the fractional Euler method when the memory points of the tail are chosen in such a way that linearly decreasing along the tail of the memory. In numerical simulations of fractional differential equations, initial conditions are required. The physical interpretations of the initial conditions are required to take measurements of the initial states of the physical real-world problems. In the case of using Riemann-Liouville fractional derivatives in numerical simulations, fractional order initial conditions are required. But making measurements of such terms is practically impossible because physical interpretations of fractional derivatives are not well defined. But, in the use of Caputo derivatives integer-order derivatives of initial states are sufficient. Therefore, the Caputo derivative is convenient in numerical simulations of fractional differential equations.

The computational cost of integrating FDEs is very high due to the non-local behaviour of the fractional order derivative operator. Therefore, reducing the computational cost of integration of FDEs is the main challenge. If all the memory points in the past are taken into account (full memory method (FMM)) in the numerical integration of FDEs, then the computational cost is high in integration on large time intervals or finer meshes. Short memory principle is one of the established technique which uses to reduce the computational cost of integrating FDEs. In the short memory method (SMM) only a fixed memory length in the recent past is taken into account in calculations and the rest of the memory points in the past (tail of the memory) are ignored. However, in this method, the computational error becomes higher when the short memory length becomes shorter. This paper aims to reduce the computational cost of the fractional Euler method by choosing a part of the memory points in the past in such a way that linearly decreasing along the tail of the memory. Hereupon we call this method "linearly decreasing memory method (LDMM)". In this paper three numerical schemes based on fractional Euler methods using FMM, SMM and LDMM are introduced. These three numerical schemes are the fractional Euler method with FMM, SMM, and LDMM (denoted by FEM-FM, FEM-SM, FEM-LDM). The author compares the convergence order of FEM-FM and FEM-LDM by calculating the experimental order of convergence (EOC) of these two numerical schemes relevant to three FDEs. The computational costs (CC) of these numerical schemes are compared using numerical solutions of three FODEs.

This paper is organized as follows: The newly proposed memory selection method (linearly decreasing memory method) is introduced section in 2.1. Three explicit numerical schemes constructed based on three memory selection methods are introduced in section 2.2. To compare the proposed numerical schemes three fractional differential equations are introduced in section 3.1. The convergence of the proposed numerical schemes FEM-FM and FEM-SM are compared based on the experimental order of convergence (EOC) in section 3.2. Computational costs of the proposed three numerical schemes are compared in section 3.3. Finally, the conclusion is given in section 4.

2. Methodology

This section explain three numerical schemes for numerical solutions of non-linear FDEs of the form:

$$\begin{aligned} &D^{\gamma} y(t) = f(t, y(t)), \quad t \in (0,T], \quad T > 0, \\ &y^{(i)}(0) = y_0^{(i)}, \qquad\qquad i = 0, 1, 2, ..., n-1, \end{aligned} \tag{3}$$

where $\gamma > 0$, and $n = \lceil \gamma \rceil$ is the smallest integer greater than α.

2.1. Proposed memory selection method: Linearly decreasing memory method.

In this section, the author introduces a method to improve the efficiency of the fractional-order Eular method by reducing the number of memory points taken into account for the computational process. In this method memory points on uniform meshes are chosen such a way that linearly decreasing along the tail of the memory. Consider the fractional differential equation (3). Now descritise $[0, T]$ into N number of partitions P_1, P_2,...,P_N and each of these partitions descritise into M number of partitions with step size Δt. That is step size of each partition P_i $(i = 1, 2, ..., N)$ is $M\Delta t$ (see Figure 1).

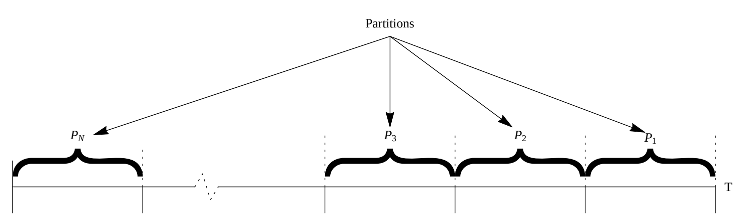


Figure 1. Sketch for the descritisation of the full time interval $[0, T]$ into main partitions.

Now consider the integration of the FDE (3) up to $t = t_m = m\Delta t$. Let $n_m = \lceil \frac{m}{M} \rceil - 1$, and $M_{m,i} = \text{Round}\left(p + (N-i)\left(\dfrac{M-p}{N-1}\right)\right)$, $i = 1,\ 2,\ldots,n_m$. Here $\lceil x \rceil$ and Round(x) denote the smallest integer greater than the real number x and rounding off integer of the real number x respectively. p is the predefined number of memory points on the partition P_N in the case of last integration from $t = 0$ to $t = T$. In the process of integration up to $t = t_m$, choose memory points, $MP(m)$, as follows:

$$MP(m) = \begin{cases} \text{All the memory points;} & \text{from } Mn_m \text{ to } m, \\ M_{m,i}; & \text{on partition } P_i\ (i = 1, 2, \ldots, n_m). \end{cases} \tag{4}$$

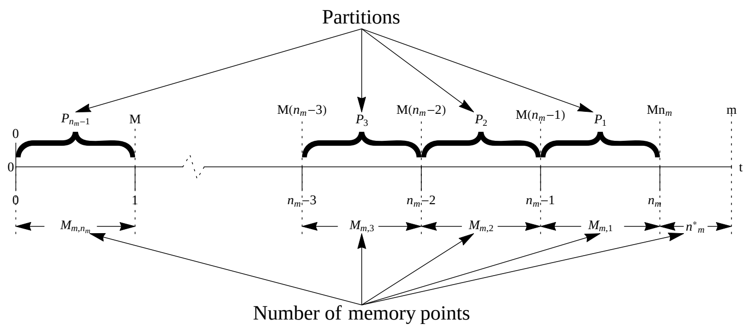


FIGURE 2. A sketch for the selection of memory points for the m^{th} step of integration (integration from 0 to $t = m\Delta t$).

A sketch for the selection of memory points on $[0, t_m]$ is shown in Figures 2.

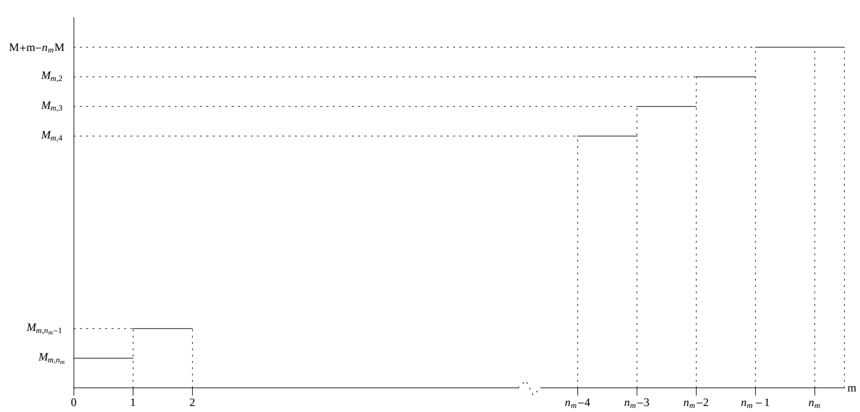


FIGURE 3. Sketch for the number of memory points on the m^{th} step of integration. (integration from 0 to $t = m\Delta t$).

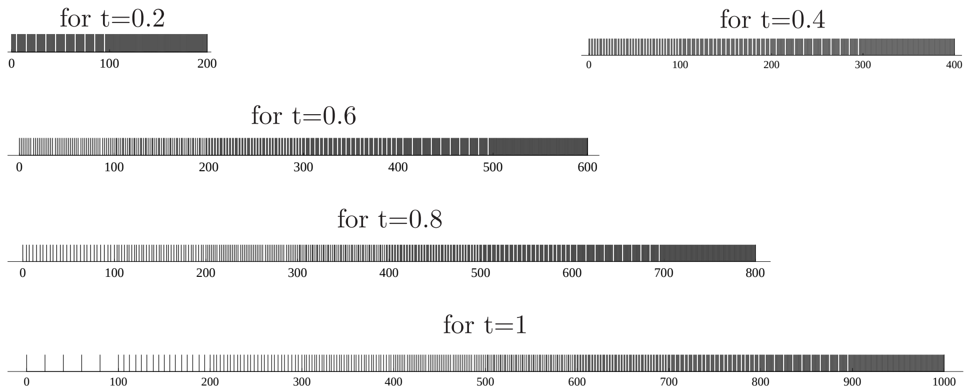


FIGURE 4. Distribution of random memory points over different time intervals for $\Delta t = 0.001$, $T = 1$ and $p_1 = 5$.

2.2. **Numerical Schemes.** This section explains fractional Euler method with full memory, short memory and linearly decreasing memory methods.

2.2.1. *Fractional Eular method with full memory(FEM-FM).* Fractional Eular method [21] with full memory for fractional differential equations of the form (3) can be written as follows:

$$y_{m+1} = \sum_{i=0}^{n-1} \frac{t_{m+1}^i y_0^{(i)}}{i!} + \frac{(\Delta t)^\gamma}{\Gamma(\gamma+1)} \sum_{i=0}^{m} b_{i,m+1} f(t_i, y_i), \text{ where } b_{i,m+1} = (m-i+1)^\gamma - (m-i)^\gamma.$$

2.2.2. *Fractional Eular method with short memory(FEM-SM).* In short memory principle only a fixed memory length in the recent history is taken into account for the computations. Suppose that L_s is the considered short (fixed) memory length and $n_s = \lceil \frac{L_s}{n} \rceil$. Then the fractional Euler method with short memory takes the form:

$$y_{m+1} = \begin{cases} \sum_{i=0}^{n-1} \frac{t_{m+1}^i y_0^{(i)}}{i!} + \frac{(\Delta t)^\gamma}{\Gamma(\gamma+1)} \sum_{i=0}^{m} b_{i,m+1} f(t_i, y_i), & \text{if } m \le n_s \\ \sum_{i=0}^{n-1} \frac{t_{m+1}^i y_0^{(i)}}{i!} + \frac{(\Delta t)^\gamma}{\Gamma(\gamma+1)} \sum_{i=m-n_s}^{m} b_{i,m+1} f(t_i, y_i), & \text{if } m > n_s. \end{cases} \tag{5}$$

2.2.3. *Fractional Eular method with linearly decreasing memory(FEM-LDM).* Fractional Euler method with linearly decreasing memory (FEM-LDM) can be written in the form:

$$y_{m+1} = \sum_{i=0}^{n-1} \frac{t_{m+1}^i y_0^{(i)}}{i!} + \frac{(\Delta t)^\gamma}{\Gamma(\gamma+1)} \left(\sum_{i=n_m^*}^{m} b_{i,m+1} f(t_i, y_i) + \sum_{i=1}^{n_m} \frac{M}{M_{m,i}} \sum_{j=1}^{M_{m,i}} b_{l_{m,i,j},m+1} f(t_{l_{m,i,j}}, y_{l_{m,i,j}}) \right). \tag{6}$$

where, $b_{i,m+1} = (m-i+1)^\gamma - (m-i)^\gamma$, $l_{m,i,j} = M(n_m - i) + \text{round}\left(\frac{M}{M_{m,i}} j\right)$, $b_{i,m+1} = (m-i+1)^\gamma - (m-i)^\gamma$, and $n_m^* = M n_m$.

3. RESULTS AND DISCUSSION

3.1. **Numerical Examples.** Computational efficiencies and accuracies of the above three numerical schemes are compared with the exact solutions of the following fractional differential equations.

$$D^{\gamma}y(t)=\frac{\Gamma(9)}{\Gamma(9-\gamma)}t^{8-\gamma}+3\frac{\Gamma(5+\gamma/2)}{\Gamma(5-\gamma/2)}t^{4-\gamma/2},\quad 0<\gamma<1,\ y(0)=0. \tag{7}$$

$$D^{0.5}y(t)=2y(t)+t^{2},\quad y(0)=0. \tag{8}$$

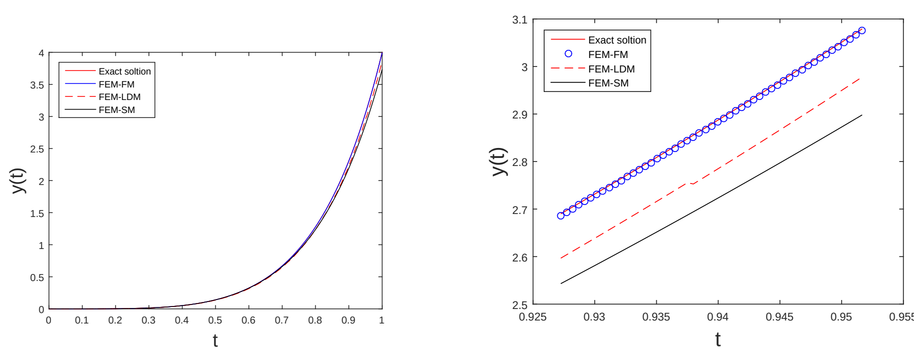
$$\begin{aligned} D^{4\gamma}y(t)+D^{3\gamma}y(t)+y(t)= & \frac{\Gamma(5/2+\gamma)}{\Gamma(5/2-3\gamma)}t^{3/2-3\gamma}+\frac{\Gamma(5/2+\gamma)}{\Gamma(5/2-2\gamma)}t^{3/2-2\gamma} \\ & +t^{(3/2+\gamma)},\quad y^{(n)}=0,\ \text{for } n=1,2,...\lceil 4\gamma\rceil. \end{aligned} \tag{9}$$

The exact solutions of the FDEs (7), (8) and (9) are $y(t)=\left(t^{8}+3t^{4+\gamma/2}\right)$, $y(t)=E_{0.5,1}(2x^{0.5})+\Gamma(3)x^{2.5}E_{0.5,3.5}$ and $y(t)=t^{(3/2-\gamma)}$ respectively. FDE (9) is a multi order fractional differential equation. It can be transformed into following system of single order fractional differential equations.

$$\begin{aligned} &D^{\gamma}y_1(t)=y_2(t),\\ &D^{\gamma}y_2(t)=y_3(t),\\ &D^{\gamma}y_3(t)=y_4(t),\\ &D^{\gamma}y_4(t)=-y_4(t)-y_1(t)+\frac{\Gamma(5/2+\gamma)}{\Gamma(5/2-3\gamma)}t^{3/2-3\gamma}+\frac{\Gamma(5/2+\gamma)}{\Gamma(5/2-2\gamma)}t^{3/2-2\gamma}+t^{(3/2+\gamma)}, \end{aligned} \tag{10}$$

whee $y_1(t)=y(t)$. The initial conditions are $y_1(0)=y(0)=0$, $y_2(0)=0$, $y_3(0)=0$, $y_4(0)=0$.

The numerical solutions obtained by FEM-FM, FEM-LDM, FEM-SM, and the exact solutions of FDE (7) are shown in Figures 5. According to the Figure 5(b), the solution obtained by FEM-LDM ($p_1=2$, $M=32$) is closer to the exact solution than numerical solution obtained by FEM-SM when percentage memory length (PML) is 40.



(a) Solutions for $0\le t\le 1$ (b) Solutions for $0.925\le t\le 0.955$

FIGURE 5. (a), (b): Numerical solutions and exact solution of FDE (7) . (In these simulations $p_1=2$, $M=32$ in LDM, PML=40 in SMM and $\Delta t=1/2^{11}$).

3.2. **Estimated Order of Convergence.** The estimated order of convergence (EOC) is measured by

EOC $= \log_2\left(\frac{E_{\max}(\Delta t)}{E_{\max}(\Delta t/2)}\right)$, where $E_{\max}(\Delta t) = \max_{1\le i\le N} |u(t_i) - U^i_{\Delta t})|$, where $N = T/\Delta t$. Here, $U^i_{\Delta t}$ is the numerical approximation for $u(t)$ at $t = t_i (\equiv i\Delta t)$. EOCs of FEM-FM and FEM-LDM correspond to FDE (7) ($\gamma = 0.9$), (8) ($\gamma = 0.5$) and (9) ($\gamma = 0.9$) are shown in Tables 1, 2 and 3 respectively.

Step Size	$E_{\max}(\Delta t)$ of FDE			EOC of FDE	
$(\Delta t = 1/2^p)$	EEM-FM	FEM-LDM	FEM-SM	EEM-FM	FEM-LDM
$1/2^{10}$	0.0104	0.2669	1.7869	-	-
$1/2^{11}$	0.0052	0.1340	1.7897	1.0000	0.9946
$1/2^{12}$	0.0026	0.0673	1.7911	1.0001	0.9933
$1/2^{13}$	0.0013	0.0337	1.7918	1.0000	0.9995
$1/2^{14}$	6.5×10^{-4}	0.0168	1.7922	1.0000	0.9997

Table 1. $E_{\max}$ and EOC of the schemes FEM-FM, FEM-LDM ($M = 2^{p-6}$, $p_1 = M/2^4$) and FEM-SM ($L = 0.4T$) of FDE (7).

Step Size	$E_{\max}(\Delta t)$ of FDE			EOC of FDE	
$(\Delta t = 1/2^p)$	EEM-FM	FEM-LDM	FEM-SM	EEM-FM	FEM-LDM
$1/2^{10}$	2.0393	38.0522	111.5062	-	-
$1/2^{11}$	1.0173	21.0099	111.5217	1.0033	0.8569
$1/2^{12}$	0.5070	11.08675	78.7738	1.0048	0.9222
$1/2^{13}$	0.2527	5.6898	78.7609	1.0046	0.9624
$1/2^{14}$	0.1260	2.8822	78.7547	1.0039	0.9812

Table 2. $E_{\max}$ and EOC of the schemes FEM-FM, FEM-LDM ($M = 2^{p-6}$, $p_1 = M/2^4$) and FEM-SM ($L = 0.4T$) of FDE (8).

Step Size	$E_{\max}(\Delta t)$ of FODE			EOC of FDE	
$(\Delta t = 1/2^p)$	EEM-FM	FEM-LDM	FEM-SM	EEM-FM	FEM-LDM
$1/2^{10}$	0.0027	0.2102	0.2764	-	-
$1/2^{11}$	0.0013	0.1091	0.0714	1.00085	0.9459
$1/2^{12}$	6.6220×10^{-4}	0.0557	0.0711	1.0061	0.9709
$1/2^{13}$	3.3011×10^{-4}	0.0281	0.0711	1.0043	0.9866
$1/2^{14}$	1.6471×10^{-4}	0.0141	0.0710	1.0031	0.9934

Table 3. $E_{\max}$ and EOC of the schemes FEM-FM, FEM-LDM ($M = 2^{p-6}$, $p_1 = M/2^4$) and FEM-SM ($L = 0.4T$) of FDE (9).

According to the Tables 1, 2 and 3 EOC of FEM-FN and FEM-LDM correspond to the FODEs (7), (8) and (9) are approximately equal.

3.3. **Computational cost.** In this section, we compare the computational costs (CPU time) of the three numerical schemes FEM-FM, FEM-LDM, and FEM-SM. Algorithms for the numerical schemes FEM-FM, FEM-SM, and FEM-LDM were developed and solved using Matlab on a 2.3GHz, Intel Core i5 laptop computer that had 8GB of ram and Microsoft Windows 10.

Now define two terms computational cost reduction percentage (CCRP) between two numerical schemes FEM-FM and FEM-LDM as follows:

$$\text{CCRP} = \frac{\text{CC of FEM-FM} - \text{CC of FEM-LDM}}{\text{CC of FEM-FM}} \times 100.$$

Table 5 shows the computational cost (CC), $E_{\max}$, CCRP of FEM-FM and FEM-LDM ($M = 2^{p-6}$, $p_1 = M/2^4$) when integrate FODE (7) up to different time levels. The computational cost of FEM-LDM has been reduced by approximately 20% when compared with FEM-FM. Therefore, the FEM-LDM is computationally more efficient than FEM-FM in the integration of FDEs on finer meshes or on larger time ranges.

Tables 4, 5 and 6 shows the computational time of the above numerical schemes correspond to the FDEs (7), (8) and (9).

Step Size	Computational time (in seconds)			CCRP	
($\Delta t = 1/2^p$)	FEM-FM	FEM-LDM	FEM-SM	FEM-LDM	FEM-SM
$1/2^{10}$	28.5057	22.1814	12.2490	22.1860	57.0295
$1/2^{11}$	107.8149	81.8871	46.9016	24.0481	56.4980
$1/2^{12}$	438.0992	371.5130	186.5630	15.1989	57.4153
$1/2^{13}$	1744.8	1469.1	1096.3	15.8005	37.1665
$1/2^{14}$	6927.9	5001.3	3026.8	27.8100	56.3105

TABLE 4. CC and CCRP of FEM-FM, FEM-LDM and FEM-SM ($L = 0.4T$) correspond to FODE (7) for $\gamma = 0.9$.

Step Size	Computational time (in seconds)			CCRP	
($\Delta t = 1/2^p$)	FEM-FM	FEM-LDM	FEM-SM	FEM-LDM	FEM-SM
$1/2^{10}$	1.98	1.3	1.27	34.25	35.99
$1/2^{11}$	6.54	4.66	4.56	28.22	30.24
$1/2^{12}$	24.71	17.39	16.86	29.65	31.76
$1/2^{13}$	101.08	65.5	64.18	35.20	36.59
$1/2^{14}$	449.54	274.19	279.46	37.62	36.42

TABLE 5. CC and CCRP of FEM-FM, FEM-LDM and FEM-SM ($L = 0.4T$) correspond to FDE (8) for $\gamma = 9$.

Step Size ($\Delta t = 1/2^p$)	Computational time (in seconds)			CCRP	
	FEM-FM	FEM-LDM	FEM-SM	FEM-LDM	FEM-SM
$1/2^{10}$	12.45	8.20	6.72	34.13	46.06
$1/2^{11}$	41.57	33.85	27.53	18.57	33.77
$1/2^{12}$	139.56	84.39	112.73	39.53	19.22
$1/2^{13}$	524.75	372.98	461.79	28.92	11.88
$1/2^{14}$	2617	2246	1897.68	14.19	27.50

Table 6. CC and CCRP of FEM-FM, FEM-LDM ($p_1 = M/10^4$) and FEM-SM ($L = 0.7T$) correspond to FDE (9) for $\gamma = 0.9$.

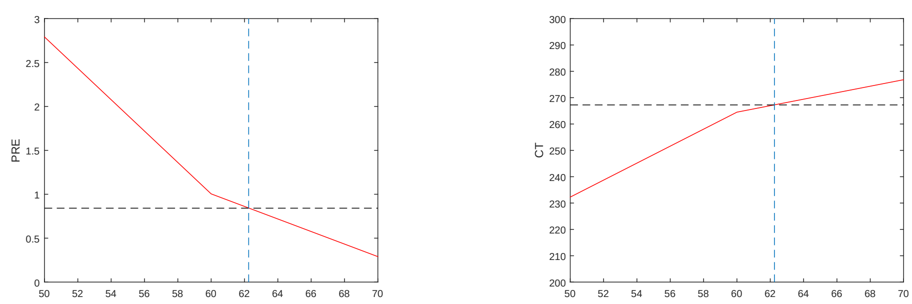
Computational error of FEM-SM is higher for FDE (9) for smaller L. Therefore, larger value for L ($L = 0.7T$) is chosen for the simulations.

3.4. **Comparison of computational cost between FEM-SM and FEM-LDM.**

Now compare the computational cost of the numerical schemes FEM-SM and FEM-LDM when computational errors of the two numerical schemes are same. To do this comparision define percentage relative error (PRE) of a numerical scheme as follows.

$$\text{PRE}= \frac{\text{E}_{\max}(\Delta t) \times 100}{\max_{t\in[0,T]} (\text{Exact solution})}.$$

FDEs (7), (8) and (9) were solved using the numerical scheme FEM-LDM for the case $M = 2^7$, $p_1 = 2^3$, $\Delta t = 1/2^{13}$. For these solutions PRE correspond to the FDEs (7), (8),and (9) are 0.84, 4.97 and 2.84 respectively. Figures 6(a), 7(a) and 8(a) indicate the corresponding PML values of FEM-SM such that maintain above PRE values respectively. Figures 6(b), 7(b) and 8(b) indicate the corresponding computational time relevant to the respective PML values of FEM-SM. These information are tabulated in Table 7.



(a) (b)

Figure 6. Comparision of computational cost between FEM-LDM and FEM-SM correspond to FDE (7). PRE of LDM= 0.84, PML of SM=62.25, CC of FEM-SM=267.25, CC of FEM-LDM=230.31

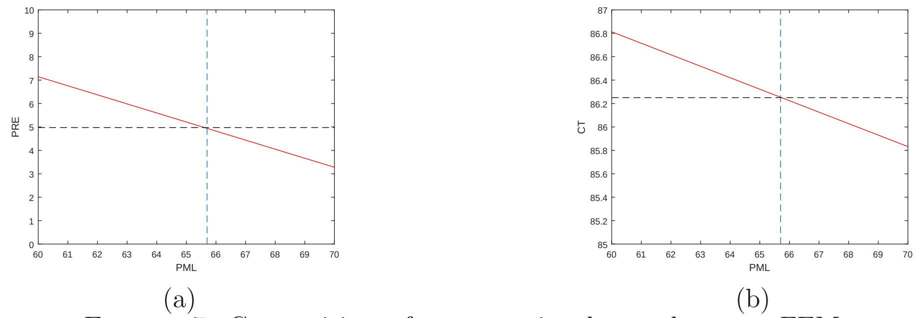


Figure 7. Comparision of computational cost between FEM-LDM and FEM-SM correspond to FDE (8). PRE of LDM= 4.97, PML of SM=65.24, CC of FEM-SM=86.25, CC of FEM-LDM=61.88

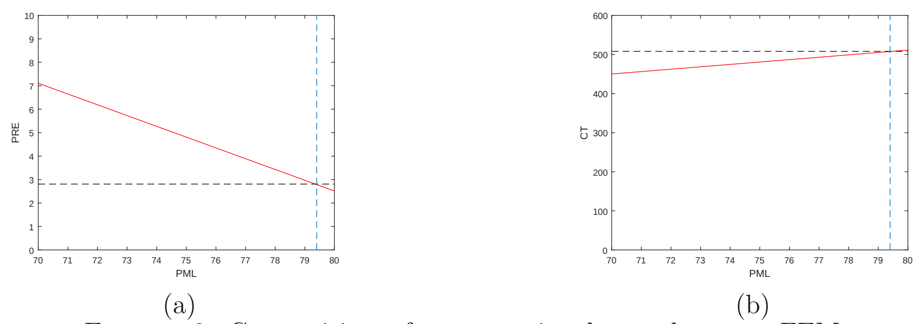


Figure 8. Comparision of computational cost between FEM-LDM and FEM-SM correspond to FDE (9). PRE of LDM= 2.84, PML of SM=79.4, CC of FEM-SM=508.00, CC of FEM-LDM=378.89

FODE	PRE of FEM-LDM	PML of FEM-SM	Computational cost of FEM-SM	Computational cost of LDM
(7)	0.84	62.25	267.25	230.31
(8)	4.97	65.6	86.25	61.88
(9)	2.84	79.4	508.00	378.89

Table 7. Comparison of computational cost of FEM-LDM and FEM-SM for $\Delta t = 1/2^{13}$. $M = 2^7$, $p_1 = 2^3$ for FEM-LDM

According to the information in table 7, one can conclude that the computational cost of FEM-SM is higher than that of FEM-SM when the computational error is maintained to a fixed value in both numerical schemes.

4. Conclusions

The accuracy and computational cost of FEM-LDM depend on the values M and p_1 of the numerical scheme. The computational cost of FEM-FM is higher than that of FEM-LDM in solving FDEs. The computational cost of FEM-LDM is less than that of FEM-SM when the computational error of both methods are equal. Also, FEM-SM is not good for numerical integration of FDEs as it's uncontrollable error. the computational cost of FEM-FM is high for the numerical integration of FODs on finer grids or on large time intervals. Therefore, the newly proposed numerical scheme FEM-LDM is better for numerical integration of FDEs as its high accuracy when compared with FEM-SM and low computational cost when compare with FEM-FM.

References

[1] R. C. Koeller, Polynomial operators, Stieltjes convolution, and fractional calculus in hereditary mechanics, Acta Mechanica, Vol. 58, No, 3-4,251-264, 1986, Springer
[2] I. Podlubny, Fractional Differential Equations, an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications, San Diego : Academic Press, Mathematics in science and engineering, 1999
[3] K. B. Oldham, Fractional differential equations in electrochemistry, Advances in Engineering Software, Vol. 41, No. 1, 9-12, 2010, Elsevier, doi:10.1016/j.advengsoft.2008.12.012
[4] K. Diethelm and A. D. Freed, On the solution of nonlinear fractional-order differential equations used in the modeling of viscoplasticity, Scientific Computing in Chemical Engineering II, 217-224, 1999, Springer
[5] J. Sabatier, O. P. Agrawal and J. A. Tenreiro Machado, Advances in fractional calculus, Vol. 4, No. 9, 2007, Springer
[6] H. Schiessel, R. Metzler, A. Blumen and T. F. Nonnenmacher, Generalized viscoelastic models: their fractional equations with solutions, Journal of physics A: Mathematical and General, Vol. 28, 6567-6584, 1995, IOP Publishing
[7] E. Ahmed, A. Hashish and F. A. Rihan, On fractional order cancer model, Journal of Fractional Calculus and Applied Analysis, Vol. 3, No. 2, 1-6,2012
[8] P. J. Torvik and R. L. Bagley, On the appearance of the fractional derivative in the behavior of real materials, Journal of Applied Mechanics, Vol. 51, No. 2, 294-298, 1984, American Society of Mechanical Engineers
[9] A. A. Kilbas, H. M. Srivastava, and J. J. Trujillo, Theory and applications of fractional differential equations Vol. 204 No. 2006, Elsevier Science Limited
[10] N. Hosein , Fractional Dynamics of Cancer Cells and the Future of Research in Biomedicine, Cancer Research Journal, Vol. 6, No. 1, p. 16, 2018, Science Publishing Group
[11] A. Y. Tuğba and A. Sadia and B. Dumitru, Optimal chemotherapy and immunotherapy schedules for a cancer-obesity model with Caputo time fractional derivative, Mathematical Methods in the Applied Sciences, Vol. 41, No. 18, 9390-940, 2018, Wiley Online Library
[12] F. A. Rihan, Numerical modeling of fractional-order biological systems, Abstract and Applied Analysis, Vol. 2013, 2013, Hindawi

[13] E. Barkai, R. Metzler and J. Klafter, From continuous time random walks to the fractional Fokker-Planck equation, Physical Review E, Vol. 61, No. 1, 132, 2000, APS

[14] G. M. Zaslavsky, Chaos, fractional kinetics, and anomalous transport, Physics reports, Vol. 371, No. 6, 461-580, 2002, Elsevier

[15] E. Scalas, R. Gorenflo and F. Mainardi, Fractional calculus and continuous-time finance, Physica A: Statistical Mechanics and its Applications, Vol. 284, No. 1-4, 376-384, 2000, Elsevier

[16] I. Podlubny, Geometric and physical interpretation of fractional integration and fractional differentiation, arXiv preprint math/0110241, 2001

[17] I. Podlubny, The Laplace transform method for linear differential equations of the fractional order, arXiv preprint funct-an/9710005, 1997

[18] I. Podlubny, Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications, 1998, Elsevier

[19] S. Kazem, Exact solution of some linear fractional differential equations by Laplace transform, International Journal of Nonlinear Science, Vol. 16, No. 1, 3-11, 2013

[20] S. Butera and M. Di. Paola, Fractional differential equations solved by using Mellin transform, Communications in Nonlinear Science and Numerical Simulation, Vol. 19, No. 7, 2220-2227, 2014, Elsevier

[21] C. Li and F. Zeng, The finite difference methods for fractional ordinary differential equations, Numerical Functional Analysis and Optimization, Vol. 34, No. 2, 149-179, 2013, Taylor & Francis

[22] P. Kumar and O. P. Agrawal, An approximate method for numerical solution of fractional differential equations, Signal Processing, Vol. 86, No. 10, 2602-2610, 2006, Elsevier

[23] J. Huang, Y. Tang and L. Vázquez, Convergence analysis of a block-by-block method for fractional differential equations, Numerical Mathematics: Theory, Methods and Applications, Vol. 5, No. 2, 229-241, 2012, Cambridge University Press

[24] K. Diethelm and A. D. Freed, The FracPECE subroutine for the numerical solution of differential equations of fractional order, Forschung und wissenschaftliches Rechnen, Vol. 1999, 57-71, 1998, Gesellschaft für Wisseschaftliche Datenverarbeitung Gottingen, Germany

[25] K. Diethelm, Efficient solution of multi-term fractional differential equations using P (EC) m E methods, Computing, Vol. 71, No. 4, 305-319, 2003, Springer

[26] Y. Liu, J. Roberts and Y. Yan, Detailed error analysis for a fractional Adams method with graded meshes, Numerical Algorithms, Vol. 78, No. 4, 1195-1216, 2018, Springer

[27] L. W. Somathilake, An efficient numerical method for fractional ordinary differential equations based on exponentially decreasing random memory on uniform meshes, Journal of the National Science Foundation of Sri Lanka, Vol. 48, No. 2, 163-174,2020, National Science Foundation of Sri Lanka

L. W. Somathilake

Department of Mathematics, Faculty of Science, University of Ruhuna, Matara, Sri Lanka

E-mail address: lwsoma@gmail.com, sthilake@maths.ruh.ac.lk