On Failed Methods of Fractional Differential Equations: the Case of Multi-step Generalized Differential Transform Method

Sima Sarv Ahrabi and Alireza Momenzadeh

Abstract. Numerical treatment of fractional differential equations, in a reliable and accurate way, is very challenging in comparison with classical integer-order differential equations. This difficulty is primarily related to the effect of non-local structure of fractional differentiation operators, to the solution of nonlinear equations involved in implicit methods and so forth. In this paper, a so-called method for fractional differential equations (FDEs) is briefly described: the multi-step generalized differential transform method (MSGDTM). It is shown that the method takes the incorrect approach in dealing with FDEs. The goal is to demonstrate that the MSGDTM is based on failed assumptions and therefore unfit for FDEs. For further verification, an illustrative example is given, in which the MSGDTM is compared with other effective and accurate methods such as fractional linear multi-step methods (FLMMs) and predictor-corrector (PC) method of Adams-Bashforth-Moulton type. The obtained results show that the MSGDTM is unfit to FDEs.

Mathematics Subject Classification (2010). Primary 34A08; Secondary 34K28.

Keywords. Fractional differential equations, numerical solution, differential transform method, Adams-Bashforth-Moulton, fractional linear multi-step methods.

1. Introduction

Fractional calculus has been studied by mathematicians for years and fractional differential equations (FDEs) are used to present the mathematical model of many real-life phenomena in diverse branches of science. Fractional differentiation operators (i.e. derivative operators of any real positive order) have non-local property, in the sense that they depend on the all previous time history and therefore are more appropriate to the systems, which

possess persistent memory characteristics. A detailed introduction to fractional calculus can be found in [5,15,21,24]. Frequent utilization of fractional differential equations can be observed in, for instance, the fluid dynamics, physics, mechanics of solids, civil engineering, control theory and biological systems [3,13,14,16,20,22]. Increasing applications of fractional calculus has been a major cause of the study and the development of numerical methods, which are specifically devised to deal with FDEs. Compared with classical (integer-order) differential equations, the construction of numerical methods of solving FDEs is much more difficult. These difficulties are primarily related to the non-local property of fractional differentiation operators, the low-order accuracy of the majority of the numerical methods, and so forth. Due to the fact that the fractional derivatives are not local in nature, multi-step methods are obvious choice for FDEs. In multi-step methods (in contrast to one-step methods), more previously approximated evaluations are required to compute the solution in each step (for a detailed explanation, see [10]). Fractional linear multi-step methods [8, 10, 17, 18] and product-integration (PI) rules [10, 27] are two of the most effective and reliable classes of numerical methods for fractional-order problems. Other approaches can be also mentioned such as Predictor-Corrector approaches [4, 6, 11], generalized exponential integrators [9], spectral methods [28], methods based on matrix functions [25] and so on (a more detailed literature can be observed in [10]).

The main goal of this article is to sufficiently describe an inadequate approach, which has been proposed to be a method of solving FDEs, and to demonstrate that the basis on which the method has been devised is not appropriate to FDEs. In order to clarify the issue, the MSGDTM will be examined by referring to reliable and effective methods for FDEs. In Section 2 the predictor-corrector method of Adams-Bashforth-Moulton type (PC method of ABM) and the fractional linear multi-step methods (FLMMs) are briefly described. Section 3 is allocated to the MSGDTM. A demonstration of the MSGDTM is given and it is illustrated that the method fails in solving FDEs. An illustrative example is represented in Section 4. For further verification, the results obtained by the MSGDTM will be compared with those of the FLMMs and the PC method of ABM.

2. PC method of ABM and FLMMs

The PC method of ABM can be considered as a fractional variant of the classical second-order Adams-Bashforth-Moulton method, which has been introduced in [6] and a detailed stability properties of the method has been discussed in [11]. The main emphasis will be placed on the single-term Caputo fractional differential equations for $0 < \alpha \leq 1$, where α is the order of the fractional derivative [5]. Consider the initial value problem

$$\begin{cases} D_{t_0}^{\alpha} y(t) = f(t, y(t)) , \\ y(t_0) = y_0 . \end{cases} \tag{2.1}$$

In order to assure the existence and uniqueness of the solution to Eq. (2.1), it is assumed that $f(t, y)$ is continuous and fulfils a Lipschitz condition with respect to the second variable [5, Theorem 6.5]. Initial value problem (2.1) can be reformulated in terms of the weakly-singular Volterra integral equation (VIE)

$$y(t) = y_0 + \frac{1}{\Gamma(\alpha)} \int_{t_0}^{t} (t-s)^{\alpha-1} f(s, y(s)) \, \mathrm{d}s \,. \tag{2.2}$$

The method presents a numerical approach in solving Eq.(2.2) and is said to be PECE (Predict, Evaluate, Correct and Evaluate) type because an initial approximation y_k^P , the so-called predictor, is first evaluated:

$$y_k^P = y_0 + \frac{1}{\Gamma(\alpha)} \sum_{j=0}^{k-1} b_{j,k} f(t_j, y_j) \,, \tag{2.3}$$

where the nodes $t_k = t_0 + kh$ are used to calculate y_k^P with a constant step-size $h > 0$ for the sake of simplicity. The weights $b_{j,k}$ are computed by

$$b_{j,k} = \frac{h^\alpha}{\alpha} \left((k-j)^\alpha - (k-1-j)^\alpha \right) . \tag{2.4}$$

Then the method gives the corrector formula:

$$y_k = y_0 + \frac{1}{\Gamma(\alpha)} \left(a_{k,k} f\left(t_k, y_k^P\right) + \sum_{j=0}^{k-1} a_{j,k} f(t_j, y_j) \right) , \tag{2.5}$$

where the weights $a_{j,k}$ are given by

$$\begin{cases} a_{0,k} = \dfrac{h^\alpha}{\alpha(\alpha+1)} \left((k-1)^{\alpha+1} - k^\alpha (k-1-\alpha) \right) , \\ a_{j,k} = \dfrac{h^\alpha}{\alpha(\alpha+1)} \left((k+1-j)^{\alpha+1} + (k-1-j)^{\alpha+1} - 2(k-j)^{\alpha+1} \right) , \\ \qquad\qquad\qquad\qquad\qquad\qquad 1 \le j \le k-1, \\ a_{k,k} = \dfrac{h^\alpha}{\alpha(\alpha+1)} \,. \end{cases} \tag{2.6}$$

The basic algorithm, the PC method of ABM, can be completely described by Eqs. (2.3) and (2.5) with the weights $b_{j,k}$ and $a_{j,k}$ defined by Eqs. (2.4) and (2.6).

The FLMMs have been introduced by Lubich in [17]. The main feature of the FLMMs is the generalization of quadrature rules, which are obtained from classical linear multi-step methods (LMMs). They have been specifically developed in order to yield a solid theoretical basis for the numerical treatment of FDEs. The FLMMs are one of the most effective methods of solving FDEs. Detailed explanations of the methods have been given in [8,10,12]. The key element in FLMMs is the approximation of Riemann-Liouville integral (on the interval $[t_0, t]$ and of the order $\alpha > 0$)

$$I_{t_0}^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_{t_0}^{t} (t-\tau)^{\alpha-1} f(\tau) \, \mathrm{d}\tau \,, \tag{2.7}$$

with the help of convolution quadrature. In the sense of Lubich, the generalization about an LMM in order to deal with Eq. (2.7) results in the corresponding FLMM as

$$ {}_hI_{t_0}^{\alpha} f(t_k) = h^{\alpha} \sum_{j=0}^{k} \omega_{k-j} f(t_j) + h^{\alpha} \sum_{j=0}^{v} w_{k,j} f(t_j), \tag{2.8} $$

on uniform nodes $t_k = t_0 + kh$. The convolution and starting quadrature wights ω_k and $w_{k,j}$ are independent of h. Starting quadrature weights $w_{k,j}$ play the major role in dealing successfully with the possible singularity of the integrand function at t_0. The features of the FLMM is specified by convolution quadrature weights ω_k.

For the sake of convenience, it is mentioned that the MATLAB code fde12.m has been devised to solve systems in which all equations have the same order. The code fde12.m is mainly devised on the basis of PC method of ABM. The MATLAB code flmm2.m, which implements three different FLMMs (i.e. the generalizations of the trapezoidal rule, the Newton-Gregory formula and the backward differentiation formula) has been introduced in [12] (rigorous information on several more MATLAB routines than those mentioned above can be observed in [10]).

3. Multi-step generalized differential transform method

The differential transform method (DTM) deals with the approximated solutions to integer-order differential equations and is based on polynomial approximations (a thorough literature on the DTM is available in [2]). The authors of [1] extended the DTM in order to solve non-integer differential equations, called as the fractional differential transform method (FDTM). Then, a rather different formulation of generalizing the DTM was introduced in [23], named as generalized differential transform method (GDTM). The GDTM provides the expansion

$$ y(t) = \sum_{k=0}^{\infty} Y_k (t - t_0)^{k\alpha}, \tag{3.1} $$

as the solution to Eq. (2.1), where the coefficients Y_k are evaluated by the recurrence equation

$$ Y_{k+1} = \frac{\Gamma(\alpha k + 1)}{\Gamma(\alpha(k+1) + 1)} F(k, Y_k) . \tag{3.2} $$

The first coefficient Y_0 is assessed to be equal to the initial condition, i.e. $Y_0 = y(t_0)$. The term $F(k, Y_k)$, mentioned as the differential transform of $f(t, y(t))$, is determined by using the methods provided in [1, 7, 23].

It must be mentioned that, under general circumstances, these methods do not provide an accurate solution to FDEs. As demonstrated by Lubich [19] (see also [5]), the solution to VIE (2.2) is expanded in mixed (integer

and fractional) powers, i.e. $\sum_{i,j\in\mathbb{N}} Y_{i,j}(t-t_0)^{i+j\alpha}$ and therefore shows a non-smooth behaviour at t_0. In the presence of non-smoothness at $t = t_0$, the solution cannot be properly approximated by methods based on polynomial approximations.

Moreover, these methods are originally based on Taylor expansion which allows to obtain the results only in a small neighborhood of t_0. In order to deal with this restriction, the authors of [7] applied the GDTM to FDEs by using step-by-step procedures, and mentioned it as multi-step generalized differential transform method (MSGDTM). It is simply formed on the idea of dividing the time interval $[t_0, T]$ into n sub-intervals $[t_j, t_{j+1}]$ with a constant step-size $h = t_{j+1} - t_j$ $(j = 0, 1, \ldots, n-1)$:

$$y(t) = \begin{cases} y_1(t) = y_0 + \sum\limits_{k\in\mathbb{N}} Y_k^{(1)} (t-t_0)^{k\alpha}, & t \in [t_0, t_1], \\ y_2(t) = y_1(t_1) + \sum\limits_{k\in\mathbb{N}} Y_k^{(2)} (t-t_1)^{k\alpha}, & t \in [t_1, t_2], \\ . \\ . \\ y_n(t) = y_{n-1}(t_{n-1}) + \sum\limits_{k\in\mathbb{N}} Y_k^{(n)} (t-t_{n-1})^{k\alpha}, & t \in [t_{n-1}, T], \end{cases} \tag{3.3}$$

where the coefficients $Y_k^{(i)}$ $(i = 1, 2, \ldots, n)$ are computed by Eq. (3.2).

The MSGDTM takes the incorrect approach in numerical treatment of FDEs; in fact, the MSGDTM implements a step-by-step procedure which is not adequate to discretize nonlocal operators such as fractional derivatives (as it is normal for integer-order differential equations). For initial value problem

$$\begin{cases} \dot{y}(t) = f(t, y(t)), \\ y(t_0) = y_0, \end{cases} \tag{3.4}$$

it is possible to compute the solution at any point t_{k+1} as

$$y_{k+1} = y_k + \int_{t_k}^{t_{k+1}} f(s, y(s))\, \mathrm{d}s. \tag{3.5}$$

Differently, the solution to Eq. (2.2) at any point t_{k+1} cannot be evaluated as the solution at the previous point t_k plus the increment term related to the interval $[t_k, t_{k+1}]$ (usually done in integer-order differential equations). This is due to the presence of a real power in the kernel.

4. Illustrative example

The statement about the MSGDTM is further verified by making a comparison with other effective and accurate methods such as the FLLMs and the PC method of ABM. Thus, the MATLAB codes flmm2.m (by choosing fractional backward differentiation formula) and fde12.m are employed to achieve the

goal. Consider the fractional Riccati differential equation (see [23]):

$$\begin{cases} D_0^{\alpha} y(t) = 2y - y^2 + 1, & t > 0, \quad 0 < \alpha \leq 1, \\ y(0) = 0. \end{cases} \tag{4.1}$$

The goal is to follow the MSGDTM for the interval $I = [0, 0.4]$ by dividing it into two sub-intervals $I_1 = [0, 0.2]$ and $I_2 = [0.2, 0.4]$. The differential transform of Eq. (4.1) is

$$\frac{\Gamma(\alpha(k+1)+1)}{\Gamma(\alpha k+1)} Y_{k+1} = 2Y_k - \sum_{k_1=0}^{k} (Y_{k_1} Y_{k-k_1}) + \delta_k, \tag{4.2}$$

where $Y(0) = y(0)$ and δ_k is computed as (see, for instance [7,23])

$$\delta_k = \begin{cases} 1, & \text{if } k = 0, \\ 0, & \text{otherwise} \end{cases}$$

By using Eqs. (3.1), (3.3) and (4.2), the results (up to $O\left(t^{3.5}\right)$) are obtained as:

$$\begin{aligned} y(t) &= 1.10\, t^{0.7} + 1.61\, t^{1.4} + 1.14\, t^{2.1} \\ &\quad -0.60\, t^{2.8} - 2.54\, t^{3.5}, \qquad 0 \leq t \leq 0.2, \end{aligned}$$

and

$$\begin{aligned} y(t) &= 0.55 + 1.98\,(t-0.2)^{0.7} + 1.30\,(t-0.2)^{1.4} - 1.54\,(t-0.2)^{2.1} \\ &\quad -3.07\,(t-0.2)^{2.8} + 0.66\,(t-0.2)^{3.5}, \qquad 0.2 \leq t \leq 0.4. \end{aligned}$$

where $\alpha = 0.7$. Figure 1 illustrates the obtained results. The problem is solved

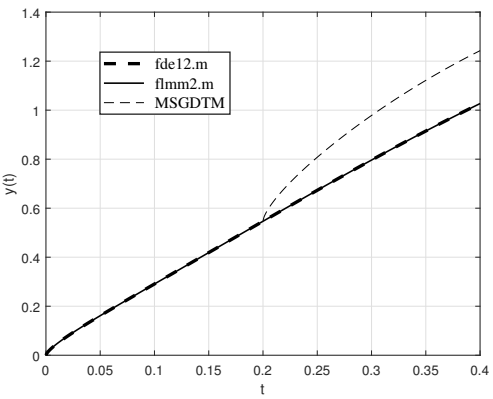

FIGURE 1. The result obtained by the MSGDTM compared with the solutions to Eq. (4.1) obtained by the FLMMs and the PC method of ABM, for $\alpha = 0.7$.

one more time, for $t \in [0, 3]$ with a constant step-size $h = 0.01$ and the results are given in Fig. 2. The MSGDTM fails obviously to give the solution of Eq.

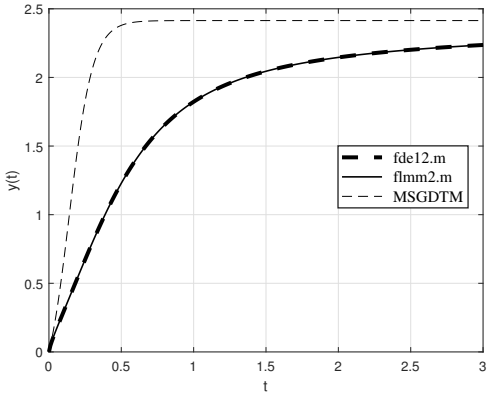

FIGURE 2. The result obtained by the MSGDTM compared with the solution to Eq. (4.1) obtained by the FLMMs and the PC method of ABM, for $\alpha = 0.7$.

(4.1); in fact, in terms of the accuracy there is no comparison. As stated, the MSGDTM approaches the problem by dividing the solution at any time t_{n+1}, into the sum of the solution of previous time t_n and the increment related to the interval $[t_n, t_{n+1}]$.

This is a basic fact that, for $\alpha = 1$, the MSGDTM returns to its classical origin (i.e. the DTM for integer-order differential equations). In this case, solutions will be in agreement with those obtained by other classical methods such as Runge-Kutta approaches for integer-order differential equations. This fact cannot be referred to as a proof of the effectiveness of the MSGDTM for FDEs.

5. Conclusion

The main reasons, due to which the MSGDTM fails to solve FDEs, have been discussed. As proved by Lubich [19], the solution of the VIE (2.2) presents an expansion in mixed (i.e. integer and fractional) powers and shows a non-smooth behavior at $t = t_0$. This is a problem for methods based on polynomial approximations, since methods of this type fail to give accurate results in the presence of some non-smoothness. Furthermore, the use of step-by-step procedures is not adequate to discretize nonlocal operators such as fractional derivatives. Thus, the MSGDTM and other similar methods are not suitable for FDEs.

Acknowledgment

The authors appreciate the reviewer's invaluable advice on improving the paper. The authors also are very grateful to Prof. Roberto Garrappa for the

development of several robust MATLAB routines for numerical treatment of fractional differential equations.

They would like to thank Prof. Paola Loreti and Dr. Mirko D'Ovidio for some helpful suggestions.

Compliance with Ethical Standards

Conflict of Interest: the authors declare that they have no conflict of interest such as: employment, consultancies, stock ownership, paid expert testimony, patent applications/registrations, and grants or other funding.

Ethical approval: this article does not contain any studies with human participants or animals performed by any of the authors, and any work involving chemicals, procedures or equipment that have any unusual hazards inherent in their use.

References

[1] Arikoglu, A., Ozkol, I.: Solution of fractional differential equations by using differential transform method. Chaos Solitons Fractals **34**, 1473–1481 (2007)

[2] Bervillier, C.: Status of the differential transformation method. Appl. Math. Comput. **218** (20), 10158–10170 (2012)

[3] Carpinteri, A., Mainardi, F.: Fractals and fractional calculus in continuum mechanics. Springer, Vienna, Austria (2014)

[4] Diethelm, K., Ford, N.J., Freed, A.D.: A predictor-corrector approach for the numerical solution of fractional differential equations. Nonlinear Dyn. **29**, 3–22 (2002)

[5] Diethelm, K.: The analysis of fractional differential equations. Springer, Berlin, Germany (2010)

[6] Diethelm, K., Freed, A.D.: The FracPECE subroutine for the numerical solution of differential equations of fractional order. Forschung und wissenschaftliches Rechnen **1999**, 57–71 (1998)

[7] Ertürk, V.S., Odibat, Z.M., Momani, S.: An approximate solution of a fractional order differential equation model of human T-cell lymphotropic virus I (HTLV-I) infection of CD4+ T-cells. Comput. Math. Appl. **62** (3), 996–1002 (2011)

[8] Galeone, L., Garrappa, R.: On multistep methods for differential equations of fractional order. Mediterr. J. Math. **3**, 565–580 (2006)

[9] Garrappa, R., Popolizio, M.: Generalized exponential time differencing methods for fractional order problems. Comput. Math. Appl. **62**, 876–890 (2011)

[10] Garrappa, R.: Numerical solution of fractional differential equations: a survey and a software tutorial. Mathematics **6**(2), 16 (2018)

[11] Garrappa, R.: On linear stability of predictor-corrector algorithms for fractional differential equations. Int. J. Comput. Math. **87**, 2281–2290 (2010)

[12] Garrappa, R.: Trapezoidal methods for fractional differential equations: theoretical and computational aspects. Math. Comput. Simulat. **110**, 96–112 (2015)

[13] Kamocki, R.: Pontryagin maximum principle for fractional ordinary optimal control problems. Math. Methods. Appl. Sci. **37**(11), 1668–1686 (2014)

[14] Keil, F., Mackens, W., Voß, H., Werther, J.: Scientific Computing in Chemical Engineering II: Computational Fluid Dynamics, Reaction Engineering, and Molecular Properties. Springer, Berlin, Germany (2012)

[15] Kilbas, A.A., Srivastava, H.M., Trujillo, J.J.: Theory and Applications of Fractional Differential Equations. In: North-Holland Mathematics Studies, Vol. 204. Elsevier Science Inc., Amsterdam, Netherlands (2006)

[16] Lopes, A.M., Machado, J.A.T., Pinto, C.M.A., Galhano, A.M.S.F.: Fractional dynamics and MDS visualization of earthquake phenomena. Comput. Math. Appl. **66**(5), 647–658 (2013)

[17] Lubich, C.: Discretized fractional calculus. SIAM J. Math. Anal. **17**(3), 704–719 (1986)

[18] Lubich, C.: Fractional linear multistep methods for Abel-Volterra integral equations of the second kind. Math. Comput. **45**(172), 463–469 (1985)

[19] Lubich, C.: Runge-Kutta theory for Volterra and Abel integral equations of the second kind. Math. Comput. **41**(163), 87–102 (1983)

[20] Magin, R.L.: Fractional calculus models of complex dynamics in biological tissues. Comput. Math. Appl. **59**(5), 1586–1593 (2010)

[21] Miller, K.S., Ross, B.: An introduction to the fractional calculus and fractional differential equations. Wiley-Interscience, New York, NY, USA (1993)

[22] Näsholm, S.P., Holm, S.: On a fractional Zener elastic wave equation. Fract. Calc. Appl. Anal. **16**, 26–50 (2013)

[23] Odibat, Z., Momani, S., Erturk, V.S.: Generalized differential transform method: application to differential equations of fractional order. Appl. Math. Comput. **197** (5), 467–477 (2008)

[24] Podlubny, I.: Fractional differential equations: an introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications. Academic Press Inc. San Diego, CA, USA (1999)

[25] Popolizio, M.: Numerical solution of multiterm fractional differential equations using the matrix Mittag–Leffler functions. Mathematics **6**, 7 (2018)

[26] Rossikhin, Y.A., Shitikova, M. V.: Application of fractional calculus for dynamic problems of solid mechanics: novel trends and recent results. Appl. Mech. Rev. **63**(1), 010801 (2010)

[27] Young, A.: Approximate product-integration. Proc. R. Soc. Lond. A **224**(1159), 552–561 (1954)

[28] Zayernouri, M., Karniadakis, G.E.: Fractional spectral collocation method. SIAM J. Sci. Comput. **36**, A40–A62 (2014)

Sima Sarv Ahrabi
Dipartimento di Scienze di Base e Applicate per l'Ingegneria
Sapienza Università di Roma
Via Antonio Scarpa, 16
00161 Rome
Italy
e-mail: sima.sarvahrabi@sbai.uniroma1.it

Alireza Momenzadeh
Dipartimento di Ingegneria dell'Informazione, Elettronica e Telecomunicazioni
Sapienza Università di Roma
Via Eudossiana, 18
00184 Roma
Italy
e-mail: momenzadeh@diet.uniroma1.it