

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/320859603>

# Benchmark problems for Caputo fractional-order ordinary differential equations

Article in *Fractional Calculus and Applied Analysis* · October 2017

DOI: 10.1515/fca-2017-0068

---

CITATIONS

7

READS

235

2 authors, including:



Dingyu Xue

Northeastern University (Shenyang, China)

168 PUBLICATIONS 5,120 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



fractional order modelling [View project](#)



Deep Learning for Medical Images [View project](#)

## SHORT PAPER

### BENCHMARK PROBLEMS FOR CAPUTO FRACTIONAL-ORDER ORDINARY DIFFERENTIAL EQUATIONS

Dingyü Xue <sup>1</sup>, Lu Bai <sup>2</sup>

*This paper is dedicated to Professor Virginia Kiryakova  
for her 65th birthday, and thanks for her 20 years'  
persistent efforts to establish FCAA as a great journal*

#### Abstract

There are many numerical algorithms for solving the fractional-order ordinary differential equations (FODEs). They are usually very different in nature, and it is difficult to compare their performances. To solve this problem, a set of five benchmark problems of different categories of FODEs with known analytical solution are designed and proposed, they can be used as benchmark problems for testing the numerical algorithms. A Simulink block diagram scheme is used for solving these benchmark problems, with computing errors and the running times reported.

*MSC 2010:* Primary 34A08; Secondary 65Y20, 26A33, 46N40, 37M05

*Key Words and Phrases:* benchmark problem; fractional calculus, Caputo differential equation; numerical algorithm, simulation

#### 1. Introduction

In recent years, fractional-order ordinary differential equations (FODEs) are widely used in the accurate descriptions of complicated dynamic systems in diverse fields of science and engineering. A prominent example is the Bagley–Torvik equation [18] - a linear multi-term FODE, and many fractional-order control systems are described by this kind of the equations.

Since very few FODEs can be solved analytically, numerical algorithms are needed.

Many classical and recently published numerical algorithms for solving FODEs are available, such as the fractional-order linear multi-step algorithm, and it is very efficient to the FODEs with zero initial conditions [12–14]; the closed-form numerical solution is presented for linear FODEs with zero initial conditions in [23]. However, the higher precision solutions cannot be obtained by these algorithms, when there exist nonzero initial conditions in the FODEs. Although the FODEs with nonzero initial conditions can be solved by the matrix algorithm [17], the unified approach for solving nonlinear FODEs is not presented. The nonlinear FODEs can be solved by the predictor–corrector-like algorithms [4–8, 10, 11], however, they are not quite efficient. A valuable nonlinear FODE is presented in [6], the corrected version can be used as a benchmark problem for testing algorithms. For fractional-order state space equations, a sequential updating discrete algorithm is used in [16]. A numerical algorithm for distributed order differential equations is proposed in [9]. Some new algorithms are also presented recently, such as the algorithms in [2, 3, 21]. Some of the existing algorithms claimed to be of high accuracy, but the orders are restricted to the range  $(0, 1)$ , while others are restricted to handling only zero initial condition problems. Besides, there is no efficient algorithms available for solving implicit FODEs.

It is also noted that in most of the original papers of the existing algorithms, each algorithm comes with its own dedicated numerical examples, therefore it might not be fair to cross comparing the performances of them. Benchmark problems of different categories of FODEs are badly needed to assess the accuracy and efficiency of the algorithms. A set of five benchmark problems with analytical solutions are designed and proposed in [1]. In this paper, they are further enhanced and deliberately made more challenging.

To the authors' knowledge, there is not a single existing numerical algorithm capable of solving all the five benchmark problems. An attempt is made in [20] by introducing unified block diagram-based solution schemes aiming at handling Caputo differential equations of any complexity. Some high precision algorithms for many kinds of FODEs are also proposed in [20], with their MATLAB implementations provided in the FOTF Toolbox [19]. In this paper, the difficulties, accuracy and time elapse analysis for all the benchmark problems are provided using the block diagram-based schemes and other methods, and it is believed that the results given here are the most accurate ones so far obtainable by any existing approaches, under double-precision data type, and the accuracy are usually several orders of magnitude higher.

## 2. Benchmark Problems

A set of five benchmark problems for the assessment of FODE numerical algorithms are designed with known analytical solutions. Some of the benchmarks problems are inspired by the existing examples in literature, and deliberately made more challenging.

**2.1. Benchmark Problem 1.** A simple one-term FODE is given by

$${}^C_0\mathcal{D}_t^{0.7}y(t) = \begin{cases} \frac{1}{\Gamma(1.3)}t^{0.3}, & 0 \leq t \leq 1, \\ \frac{1}{\Gamma(1.3)}t^{0.3} - \frac{2}{\Gamma(2.3)}(t-1)^{1.3}, & t > 1. \end{cases} \quad (2.1)$$

The initial condition is  $y(0) = 0$ , and the solution interval is  $t \in (0, 2)$ . The analytical solution of equation (2.1) is a piecewise function

$$y(t) = \begin{cases} t, & 0 \leq t \leq 1, \\ t - (t-1)^2, & t > 1. \end{cases}$$

**2.2. Benchmark Problem 2.** A linear multi-term FODE is given by

$$y'''(t) + {}^C_0\mathcal{D}_t^{2.5}y(t) + y''(t) + 4y'(t) + {}^C_0\mathcal{D}_t^{0.5}y(t) + 4y(t) = 6 \cos t, \quad (2.2)$$

with the nonzero initial conditions  $y(0) = 1$ ,  $y'(0) = 1$ ,  $y''(0) = -1$ , and the analytical solution is  $y(t) = \sqrt{2} \sin(t + \pi/4)$ , with  $t \in (0, 5000)$ .

**2.3. Benchmark Problem 3.** A nonlinear explicit FODE is given by

$${}^C_0\mathcal{D}_t^{\sqrt{2}}y(t) = -t^{1.5-\sqrt{2}} \frac{E_{1,3-\sqrt{2}}(-t)}{E_{1,1.5}(-t)} e^t y(t) {}^C_0\mathcal{D}_t^{0.5}y(t) + e^{-2t} - [y'(t)]^2, \quad (2.3)$$

where  $E_{1,1.5}(\cdot)$  is a two-parameters' Mittag-Leffler function. The initial conditions are  $y(0) = 1$ ,  $y'(0) = -1$ , and the analytical solution is  $y(t) = e^{-t}$ ,  $t \in (0, 2)$ . Equation (2.3) is inspired by [6], where the one-parameter Mittag-Leffler functions are corrected to two-parameter ones, and the highest order is deliberately changed to an irrational number  $\sqrt{2}$ , such that the FODE is no longer a commensurate-order one.

**2.4. Benchmark Problem 4.** An implicit nonlinear FODE is given by

$$\begin{aligned} {}^C_0\mathcal{D}_t^{0.2}y(t) {}^C_0\mathcal{D}_t^{1.8}y(t) + {}^C_0\mathcal{D}_t^{0.3}y(t) {}^C_0\mathcal{D}_t^{1.7}y(t) \\ = -\frac{t}{8} \left[ E_{1,1.8}(-t/2) E_{1,1.2}(-t/2) - E_{1,1.7}(-t/2) E_{1,1.3}(-t/2) \right], \end{aligned} \quad (2.4)$$

with the initial conditions  $y(0) = 1$ ,  $y'(0) = -1/2$ , and the analytical solution is  $y(t) = e^{-t/2}$ , and  $t \in (0, 10)$ .

**2.5. Benchmark Problem 5.** A fractional-order state space model is given by

$$\begin{cases} {}_0^C\mathcal{D}_t^{0.5}x(t) = \frac{1}{\sqrt{\pi}} \left( \sqrt[6]{(y(t) - 0.5)(z(t) - 0.3)} + \sqrt{t} \right) \\ {}_0^C\mathcal{D}_t^{0.2}y(t) = \Gamma(2.2)(x(t) - 1) \\ {}_0^C\mathcal{D}_t^{0.6}z(t) = \frac{\Gamma(2.8)}{\Gamma(2.2)}(y(t) - 0.5). \end{cases} \quad (2.5)$$

The initial conditions are  $x(0) = 1$ ,  $y(0) = 0.5$ ,  $z(0) = 0.3$ , and the analytical solutions are  $x(t) = t + 1$ ,  $y(t) = t^{1.2} + 0.5$ ,  $z(t) = t^{1.8} + 0.3$ , and  $t \in (0, 5)$ .

### 3. Solutions and Comments

In this section, our results to all the benchmark problems are provided, and the reusable solvers and Simulink models are provided with the new versions of FOTF Toolbox. Detailed Simulink model structures and related reusable MATLAB codes are reported in a companion PDF file in the toolbox. In this paper, only major results are provided. All the elapse time are measured on the same laptop - Lenovo Yoga 5 Pro, equipped with Intel i7-7500U CPU at 2.90GHz, 16GB RAM and 1TB SSD.

(1) Solutions to benchmark problem 1. The difficulties in the equation is that the right hand side function in the equation is a piecewise one. The design of this benchmark problem is motivated by Dr Yiheng Wei of University of Science and Technology of China.

The Simulink model for the differential equation is created and stored in file bp1model.slx, downloadable with FOTF Toolbox. By choosing the expected frequency range as  $(10^{-5}, 10^5)$  in the Oustaloup's filter [15], and an order of  $N = 25$ , the maximum error obtained is  $6.0761 \times 10^{-6}$ , and the solutions can be obtained with 10 seconds, with 78468 points computed. If stiff solver ode15s in Simulink is adopted, the time elapse is only 0.38 seconds, with only 1467 points computed, and the maximum error is virtually the same.

(2) Solutions to benchmark problem 2. The time duration is deliberately set to  $t \in (0, 5000)$ . It is obvious that none of the existing fixed-step algorithms can be used, since if the step-size is selected relatively small, the total computational load may be too heavy even for modern computers.

Simulink model bp2model.slx can be established, and with the parameters  $(10^{-5}, 10^5)$ ,  $N = 25$ , maximum error as low as  $1.2945 \times 10^{-10}$  can be achieved, with time elapse of 12.63 seconds, and a total of 617581 points computed. If Runge–Kutta–Fehlberg algorithm is used, a smaller time interval  $t \in (0, 100)$  can be tried, and 7231268 points may be computed, with around 151 seconds. The maximum error is virtually the same. It is strongly suggested that the stiff equation solvers in Simulink be adopted for efficient solutions.

(3) Solutions to benchmark problem 3. Many existing algorithms are established based on the assumptions that the original equations can be expressed by commensurate-order systems, while in this particular example, since an irrational order is involved, the equation is an incommensurate-order one. Many of the existing algorithms cannot be used at all in solving incommensurate-order FODEs.

With Simulink modeling technique, the model bp3model.slx can be established. When the parameters of the Oustaloup filters are set to  $(10^{-5}, 10^6)$ ,  $N = 25$ , the maximum error is  $2.0917 \times 10^{-6}$ , and the time needed is only 0.57 seconds.

If the  $o(h^p)$  algorithm in [20] is used, with step-size  $h = 0.0001$  and  $p = 2$ , the maximum error is as small as  $3.5314 \times 10^{-8}$ , about 60 times more accurate than the Simulink result, however, the time required is 318 seconds, about 550 times more time needed. Large step-sizes are also allowed in the  $o(h^p)$  algorithm. For instance, if  $h = 0.01$ ,  $p = 4$ , the maximum error is  $4.2143 \times 10^{-8}$ , with 83.2 seconds.

If the algorithm in [16] is used, for the interval  $t \in (0, 1)$ , with step-size of  $h = 0.00001$ , the maximum error is  $2.1064 \times 10^{-5}$ , and the time elapse is 156 seconds, and it begins to diverge when  $t$  is larger. It should be noted that the time consumption in this algorithm is highly nonlinear with respect to  $h$ . It is not suitable to further reduce  $h$  to achieve more accurate results.

(4) Solutions to benchmark problem 4. Since the original FODE is an implicit one, and the initial value of  ${}_0^C\mathcal{D}_t^{0.2}y(t)$  at  $t = 0$  is likely to be zero, the original FODE cannot be converted into an explicit one, by merely dividing both sides by  ${}_0^C\mathcal{D}_t^{0.2}y(t)$ . To the authors' knowledge, there is no other efficient algorithms capable of tackling implicit FODEs, the Simulink scheme may be the only choice. The Simulink model bp4model.slx can be used in describing the implicit Caputo equation, and with  $(10^{-5}, 10^5)$ ,  $N = 25$ , the maximum error is  $3.6868 \times 10^{-5}$ , and the time needed is 289.47 seconds.

Since this equation is an implicit one, algebraic loops in Simulink model are unavoidable, which means that within each computation step, an algebraic equation is solved, which makes the execution of the model very time consuming.

(5) Solutions to benchmark problem 5. Again, this benchmark problem is less challenging, compared with others. If the parameters of the Oustaloup filters are selected as  $(10^{-5}, 10^5)$ ,  $N = 25$ , the maximum error is  $3.5656 \times 10^{-4}$ , within 3.83 seconds.

It should be noted that since signal  $z(t)$  increases with  $t$  rapidly, absolute maximum error on  $z(t)$  is not a good measure to describe the accuracy of the numerical algorithms. Relative errors may be more meaningful.

For this particular example, if the algorithm in [16] is used, with fixed step-size of  $h = 0.00001$ , the maximum error is reduced to  $1.1071 \times 10^{-5}$ , and the codes runs about two hours. The result is more accurate than the Simulink results, however is is too time demanding. If time interval is changed to  $t \in (0, 10)$ , such a small step-size definitely cannot be used, while in Simulink model, the time consumption is almost linear with respect to terminate time.

#### 4. Concluding Remarks

A set of five benchmark problems, covering wide categories of FODE types, is designed and proposed. They can be used in the assessment of numerical algorithms for fractional-order ordinary Caputo equations. It can be seen that all of them can be solved with the unified high efficiency Simulink based solvers.

The Simulink models involved are usually stiff, therefore, it is suggested that stiff equation solvers such as ode15s and ode113 in Simulink can be adopted for fast and efficient solutions.

The parameters  $(\omega_b, \omega_h)$  and  $N$  of Oustaloup filters or their modified versions [22] are important in approximating fractional-order derivative or integral activities. For all the solutions, they are selected the same. One may fine-tune them accordingly if necessary. For instance, if the error is large when  $t$  is large,  $\omega_b$  can be reduced.

All the FODEs studied in the paper come with analytical solutions, however, in real applications, it is not the case, since most of the FODEs have no analytical solutions, and numerical solutions are the only way to study them. If the analytical solutions are not known, of course, the validation process in this paper cannot be used. Alternatively, two different set of Oustaloup filter parameters, or Simulink solvers or other control parameters can be used. If they yield virtually the same results, the solutions can be regarded as validated, otherwise, use better parameters, for instance, wider frequency band, or higher order in Oustaloup filters and try again, until satisfactory results are obtained.

Since Simulink with various of variable-step algorithms are involved in the solutions, it is not possible to mathematically prove the precisions or error bands of the algorithms.

#### Acknowledgement

The research work in the paper is supported financially by the National Natural Science Foundation of China under Grants 61174145 and 61673094. The kind suggestions and recommendations of Professor Igor Podlubny are also acknowledged.

### References

- [1] L. Bai, D.Y. Xue, The benchmark problems for the assessment of numerical algorithms on fractional-order differential equations. In: *Proc. of the 29th Chinese Control and Decision Conference, CCDC 2017* (2017), 1004–1009.
- [2] J.X. Cao, C.P. Li, Y.Q. Chen, High-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (II). *Fract. Calc. Appl. Anal.* **18**, No 3 (2015), 735–761; DOI: 10.1515/fca-2015-0045; <https://www.degruyter.com/view/j/fca.2015.18.issue-3/issue-files/fca.2015.18.issue-3.xml>.
- [3] J.X. Cao, C.P. Li, Y.Q. Chen, High-order approximation to Caputo derivatives and Caputo-type advection-diffusion equations (III). *J. Comput. Appl. Math.* **18**, No 3 (2015), 159–175.
- [4] K. Diethelm, Efficient solution of multi-term fractional differential equations using P(EC)<sup>m</sup>E methods. *Computing* **71**, No 4 (2003), 305–319.
- [5] K. Diethelm, An investigation of some nonclassical methods for the numerical approximation of Caputo-type fractional derivatives. *Numer. Algor.* **47**, No 3 (2008), 361–390.
- [6] K. Diethelm, *The Analysis of Fractional Differential Equations: An Application-oriented Exposition Using Differential Operators of Caputo Type*. Springer, Berlin (2010).
- [7] K. Diethelm, An efficient parallel algorithm for the numerical solution of fractional differential equations. *Fract. Calc. Appl. Anal.* **14**, No 3 (2011), 475–490; DOI: 10.2478/s13540-011-0029-1; <https://www.degruyter.com/view/j/fca.2011.14.issue-3/issue-files/fca.2011.14.issue-3.xml>.
- [8] K. Diethelm, N.J. Ford, Multi-order fractional differential equations and their numerical solution. *Appl. Math. Comput.* **154**, No 3 (2004), 621–640.
- [9] K. Diethelm, N.J. Ford, Numerical analysis for distributed-order differential equations. *J. Comput. Appl. Math.* **225**, No 1 (2009), 96–104.
- [10] K. Diethelm, N.J. Ford, A.D. Freed, A predictor-corrector approach for the numerical solution of fractional differential equations. *Nonlinear Dyn.* **29**, No 1-4 (2002), 3–22.
- [11] K. Diethelm, N.J. Ford, A.D. Freed, Detailed error analysis for a fractional Adams method. *Numer. Algor.* **36**, No 1 (2004), 31–52.
- [12] C. Lubich, On the stability of linear multistep methods for Volterra convolution equations. *IMA J. Numer. Anal.* **3**, No 4 (1983), 439–465.

- [13] C. Lubich, Discretized fractional calculus. *SIAM J. Math. Anal.* **17**, No 3 (1986), 704–719.
- [14] C. Lubich, Fractional linear multistep methods for Abel–Volterra integral equations of the first kind. *IMA J. Numer. Anal.* **7**, No 1 (1987), 97–106.
- [15] A. Oustaloup, F. Levron, F. Nanot, B. Mathieu, Frequency band complex non integer differentiator: characterization and synthesis. *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.* **47**, No 1 (2000), 25–40.
- [16] I. Petráš, *Fractional-order Nonlinear Systems - Modeling, Simulation and Analysis*. Higher Education Press, Beijing (2011).
- [17] I. Podlubny, Matrix approach to discrete fractional calculus. *Fract. Calc. Appl. Anal.* **3**, No 4 (2000), 359–386.
- [18] P.J. Torvik, R.L. Bagley, On the appearance of the fractional derivative in the behavior of real materials. *J. Appl. Mech.* **51** (1984), 294–298.
- [19] D.Y. Xue, *FOTF Toolbox*, downloadable from  
<http://cn.mathworks.com/matlabcentral/fileexchange/60874-fotf-toolbox>.
- [20] D.Y. Xue, *Fractional-order Control Systems – Fundamentals and Numerical Implementations*. De Gruyter, Berlin (2017).
- [21] D.Y. Xue, L. Bai, Numerical algorithms for Caputo fractional-order differential equations. *Int. J. Control* **90**, No 6 (2017), 1201–1211.
- [22] D.Y. Xue, C. N. Zhao, Y. Q. Chen, A modified approximation method of fractional order system. In: *IEEE Internat. Conf. Mechatronics and Automation* (2006), 1043–1048.
- [23] C.N. Zhao, D.Y. Xue, Closed-form solutions to fractional-order linear differential equations, *Front. Electr. Electr. Eng. in China* **3**, No 2 (2008), 214–217.

<sup>1</sup> School of Information Science and Engineering  
Northeastern University, Shenyang 110006, CHINA

e-mail: [xuedingyu@mail.neu.edu.cn](mailto:xuedingyu@mail.neu.edu.cn) Received: July 8, 2017

<sup>2</sup> School of Information Science and Engineering  
Northeastern University, Shenyang 110006, CHINA  
and: School of Information Engineering, Shenyang University  
Shenyang 110044, CHINA  
e-mail: [bailu820601@163.com](mailto:bailu820601@163.com)

---

Please cite to this paper as published in:

*Fract. Calc. Appl. Anal.*, Vol. **20**, No 5 (2017), pp. 1305–1312,  
DOI: [10.1515/fca-2017-0068](https://doi.org/10.1515/fca-2017-0068)