Research

Author for correspondence:
Igor Podlubny
e-mail: igor.podlubny@tuke.sk

Matrix approach to discrete fractional calculus III: non-equidistant grids, variable step length and distributed orders

Igor Podlubny[1], Tomas Skovranek[1],
Blas M. Vinagre Jara[2], Ivo Petras[1],
Viktor Verbitsky[3] and YangQuan Chen[4]

[1]BERG Faculty, Technical University of Kosice, B. Nemcovej 3, 04200 Kosice, Slovakia
[2]School of Industrial Engineering, University of Extremadura, Campus Universitario, s/n, Badajoz 06071, Spain
[3]Institute of Mathematics, Economics, and Mechanics, Odessa National University, Dvoryanskaya 2, Odessa 65082, Ukraine
[4]Mechatronics, Embedded Systems and Automation Laboratory, School of Engineering, University of California, Merced, 5200 North Lake Road, Merced, CA 95343, USA

In this paper, we further develop Podlubny's matrix approach to discretization of integrals and derivatives of non-integer order. Numerical integration and differentiation on non-equidistant grids is introduced and illustrated by several examples of numerical solution of differential equations with fractional derivatives of constant orders and with distributed-order derivatives. In this paper, for the first time, we present a variable-step-length approach that we call 'the method of large steps', because it is applied in combination with the matrix approach for each 'large step'. This new method is also illustrated by an easy-to-follow example. The presented approach allows fractional-order and distributed-order differentiation and integration of non-uniformly sampled signals, and opens the way to development of variable- and adaptive-step-length techniques for fractional- and distributed-order differential equations.

1. Introduction

The results that we present in this paper were motivated by two important challenges in applied numerical methods of fractional calculus.

First, until recent times, the fractional derivatives were discretized using equidistant nodes. This has roots in the famous Grünwald–Letnikov definition of fractional-order differentiation, which is based on generalization of finite differences defined on an equidistant grid, and which gives the simplest and very efficient approximation for numerical evaluation of fractional derivatives. This Grünwald–Letnikov-based approach to discretization of fractional derivatives had very strong impact on the way of thinking in fractional calculus, that even fractional integrals were routinely discretized on equidistant grids, too. However, it is clear that for fractional integrals it was not a necessity at all. On the other hand, it was unclear what one could do with approximation of fractional integrals on non-equidistant grids, if one wants to have a uniform approach to discretization of both fractional derivatives and fractional integrals.

Second, solution of fractional differential equations in large time intervals is a computational problem due to the fact that the number of points taken into account in computations grows with the growing value of the time variable. This is caused by the non-local character of fractional-order differentiation. So far, the only aid in this respect was the 'short memory principle' [1]. Methods known from classical numerical solutions of integer-order differential equations, especially variable-step-length techniques, were not available for fractional differential equations.

The systematic and continuous development of the 'matrix approach' [2] allowed us to find some answers to both challenges, and in this paper, we present them as two mutually related methods for solving problems of discrete fractional calculus on non-uniformly spaced discretization grids. Moreover, we extend this approach to distributed-order operators and distributed-order differential equations.

We start with demonstrating how the matrix approach can be extended to numerical evaluation of fractional-order integrals and derivatives on non-equidistant grids, and how fractional differential equations with constant-order derivatives can be solved on such grids. This finally unifies the discretization of fractional derivatives and fractional integrals on arbitrary (equidistant and non-equidistant) grids.

After that we make the next step and extend the matrix approach to discretization of distributed-order operators and to numerical solution of distributed-order differential equations.

Then, we move the focus onto using variable step length for solving fractional differential equations. In this paper, we, for the first time, present the method that we call 'the method of large steps'. We provide the general framework and illustrate this method by a numerical example that, for verification purposes, allows easy exact solution as well.

Because each 'large step' consists of a set of 'small steps', it can be done using the matrix approach, and the 'small steps' can be equidistant or non-equidistant. This is illustrated by included little pieces of Matlab code using our publicly available toolbox [3,4].

The methods presented in this paper finally allow fractional-order differentiation and integration of non-uniformly sampled signals, and the development of variable-step-length techniques for solving fractional differential equations (ordinary and partial).

The standard basic notation and basic definitions of fractional derivatives and fractional integrals can be found in Podlubny [1], Samko *et al.* [5] and Kilbas *et al.* [6].

2. Fractional-order integration and differentiation on non-equidistant grids

Although equidistant grids are frequently used in application, in many situations, the use of non-equidistant grids brings notable advantages. For example, many numerical methods for solving differential equations use a variable-time-step technique, so the time step can increase or decrease depending on how rapidly the resulting solution is changing.
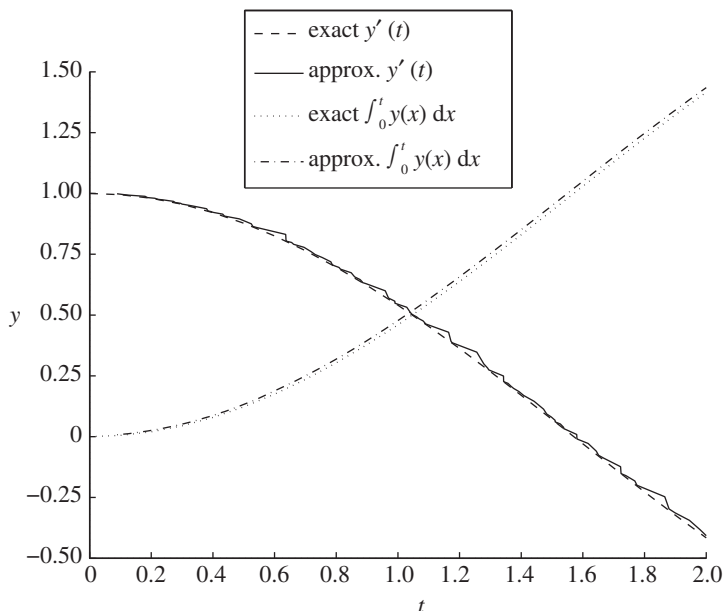
Figure 1. Exact and approximate evaluation of the first-order derivative and the onefold integral of $y = \sin(t)$ on the grid of 200 random non-equidistant nodes.

Up to now, the fractional derivatives were discretized using equidistant nodes. This was, of course, due to the famous Grünwald–Letnikov definition of fractional-order differentiation, which is based on generalization of finite differences defined on an equidistant grid.

The matrix approach to discretization of integrals and derivatives of arbitrary real order, developed by Podlubny [2,7], allows us to generalize discretization of fractional-order integrals and derivatives to non-equidistant grids.

The idea is to create first a discretization matrix $\boldsymbol{I}^{\alpha}$ for integration of order α. After the matrix $\boldsymbol{I}^{\alpha}$ for discrete fractional integration on a non-equidistant grid is obtained, we can easily derive the matrix $\boldsymbol{D}^{\alpha}$ for discretization of fractional-order derivatives by matrix inversion:

$$\boldsymbol{D}^{\alpha} = (\boldsymbol{I}^{\alpha})^{-1}.$$

In the simplest case, the function under differentiation can be approximated by a piecewise constant function, and for the non-equidistant discretization nodes t_k $(k=1,\ldots,N)$, the coefficients of the lower triangular matrix $\boldsymbol{I}^{\alpha}$ can be evaluated as

$$I_{k,j} = \frac{(t_k - t_{j-1})^{\alpha} - (t_k - t_j)^{\alpha}}{\Gamma(\alpha+1)}, \tag{2.1}$$

with

$$j = 1,\ldots,k \quad \text{and} \quad k = 1,\ldots,N. \tag{2.2}$$

Other formulae for numerical integration will give other expressions for the coefficients $I_{k,j}$; however, as we demonstrate below, even this simple approximation works well.

In the case of non-equidistant nodes, however, the matrices $\boldsymbol{I}^{\alpha}$ and $\boldsymbol{D}^{\alpha}$ are not strip matrices, although for one-sided fractional integrals and derivatives they are still triangular.

In the examples given below, we use non-equidistant nodes generated with random steps. We generate N random points between 0 and 1, sort them in ascending order, and then scale to the considered interval of length L. After that, we replace the first and the last randomly generated nodes with the exact left and right bounds of the considered interval.
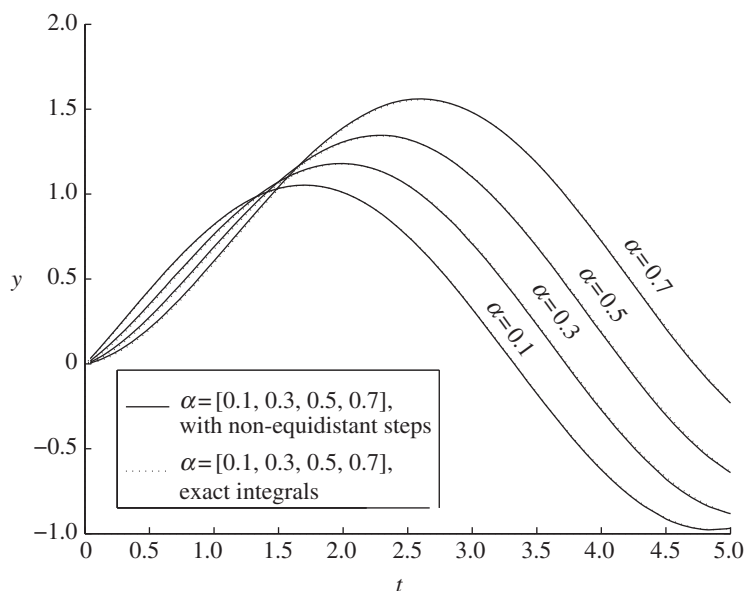
Figure 2. Exact and approximate evaluation of αth-order integrals of $y = \sin(t)$ on the grid of 200 random non-equidistant nodes for $\alpha = 0.1, 0.3, 0.5$ and 0.7.
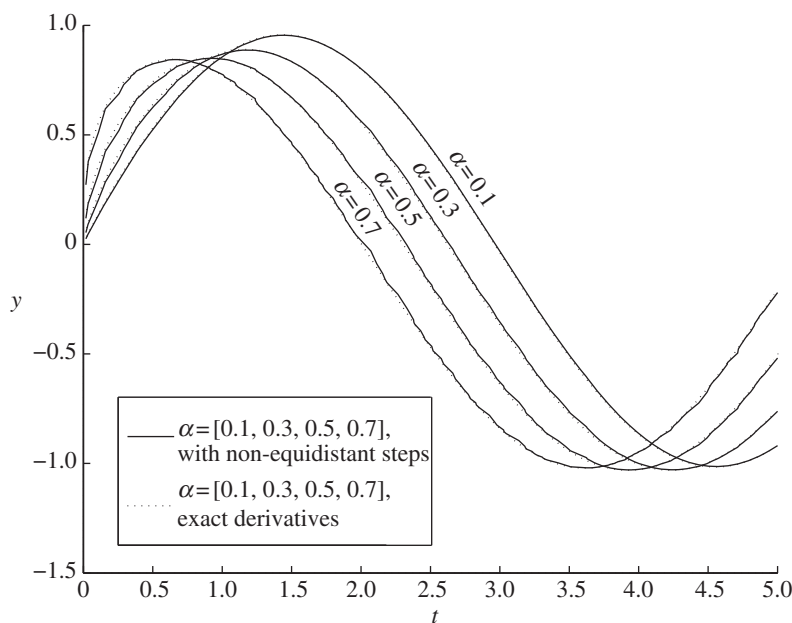

Figure 3. Exact and approximate evaluation of αth-order derivatives of $y = \sin(t)$ on the grid of 200 random non-equidistant nodes for $\alpha = 0.1, 0.3, 0.5$ and 0.7.

(a) Example 1: evaluation of integer-order integrals and derivatives

Let us consider the function $y = \sin(t)$. In figure 1, its exact first-order derivative (dashed line) and its exact (dotted line) onefold integral (dash-dotted line) are depicted.
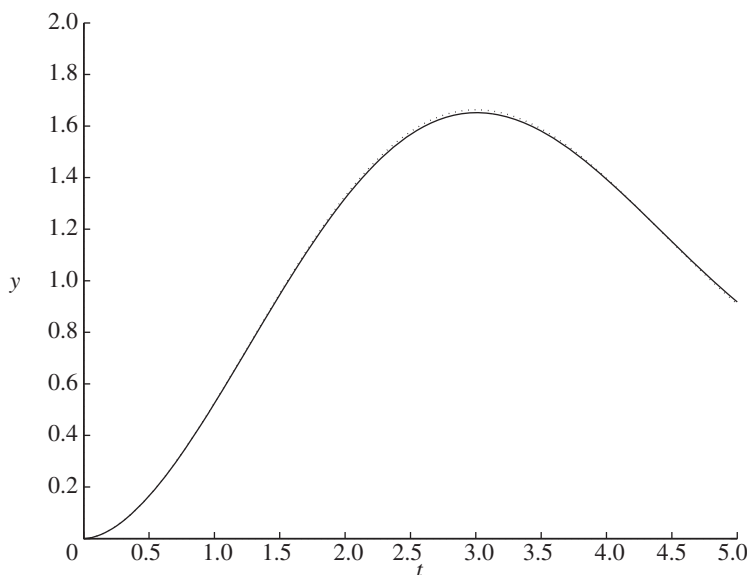
Figure 4. Analytical and numerical solution of problem (3.1). Dotted line denotes analytical solution. Solid line denotes numerical solution (non-equidistant steps).

The results of numerical differentiation (solid line) and numerical integration (dash-dotted line) of the same function using the matrix approach on a non-equidistant grid of 200 randomly generated nodes in the interval $[0, 2]$ are also shown in figure 1.

(b) Example 2: evaluation of fractional-order integrals and derivatives

The proposed approach is suitable also for evaluating fractional-order integrals and derivatives. In figures 2 and 3, fractional-order integrals and derivatives of orders $\alpha = 0.1$, $\alpha = 0.3$, $\alpha = 0.5$ and $\alpha = 0.7$ of function $y = \sin(t)$ are plotted. Each derivative obtained using non-equidistant steps is compared with the solution obtained using the 'matrix approach' with equidistant steps. The match shows good agreement of the results.

3. Solution of fractional differential equations on non-equidistant grids

Using discrete analogues of fractional-order derivatives on a non-equidistant grid, we can easily and conveniently perform discretization and numerical solution of fractional differential equations on such grids. We illustrate the developed approach on two classical benchmark problems: the two-term fractional relaxation equation and the Bagley–Torvik equation.

(a) Example 3: fractional relaxation equation

In the first work on the matrix approach to discrete fractional calculus [2], the following sample two-term fractional differential equation in terms of the Caputo derivatives [1] under zero initial conditions was considered:

$$\left. \begin{array}{l} y^{(\alpha)}(t) + y(t) = 1 \\ y(0) = 0, \quad y'(0) = 0. \end{array} \right\} \tag{3.1}$$

and

The exact analytical solution of this problem can be expressed using the Mittag–Leffler function:

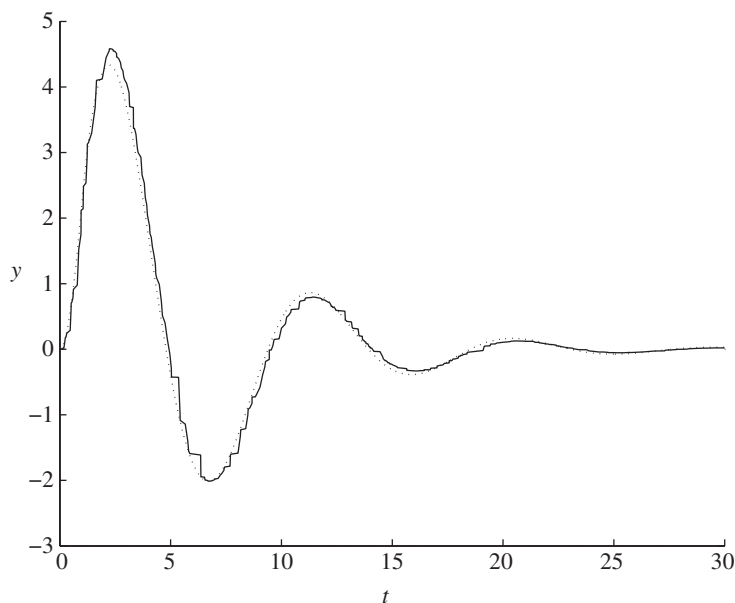$$y(t) = t^\alpha E_{\alpha,\alpha+1}(-t^\alpha). \tag{3.2}$$

Figure 5. Solution of the Bagley–Torvik equation for $A = 1$, $B = 1$, $C = 1$. Dotted line denotes equidistant steps; solid line denotes non-equidistant steps.
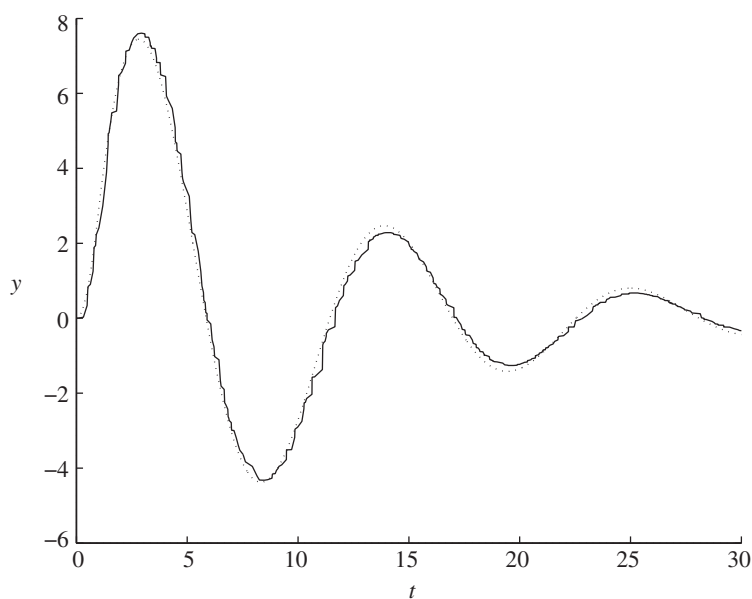

Figure 6. Solution of the Bagley–Torvik equation for $A = 1$, $B = 0.5$, $C = 0.5$. Dotted line denotes equidistant steps; solid line denotes non-equidistant steps.

In figure 4, the comparison of the exact analytical solution (dotted line) and numerical solution obtained with the help of developed approach (solid line) using non-equidistant nodes (with $N = 500$) for the case of $\alpha = 1.8$ is shown.

(b) Example 4: fractional oscillator equation

As mentioned earlier, the proposed approach allows easy solution of ordinary differential equations with derivatives of arbitrary real order (integer and non-integer). Let us consider the

following classical initial value problem for the Bagley–Torvik equation (also known as damped fractional oscillator equations):

$$Ay''(t) + By^{(3/2)}(t) + Cy(t) = F(t),$$

and
$$F(t) = \begin{cases} 8 & (0 \le t \le 1), \\ 0 & (t > 1), \end{cases} \qquad y(0) = y'(0) = 0.$$

(3.3)

The solutions of the Bagley–Torvik equation for $A = 1$, $B = 1$, $C = 1$ in the time interval $[0; 30]$, obtained using two approaches, are shown in figure 5, and the solutions of the same problem for $A = 1$, $B = 0.5$, $C = 0.5$ are depicted in figure 6. In both cases, the number of discretizaton steps is 400. The dotted lines represent the numerical solutions obtained using equidistant steps (with $h = 0.075$), and the solid lines represent the numerical solutions with the same number of randomly generated non-equidistant steps.

4. Solution of distributed-order differential equations on non-equidistant grids

The presented extension of the matrix approach to discretization of non-integer order integrals and derivatives and to numerical solution of equations with such operators on non-equidistant grids can be used for solving distributed-order differential equations, too. In this paper, we use the following definition of distributed-order differential/integral operators [8]:

$$_0D_t^{w(\alpha)}f(t) = \int_{\gamma_1}^{\gamma_2} w(\alpha) \, _0D_t^\alpha f(t) \, d\alpha,$$

(4.1)

where $w(\alpha)$ denotes the weight function of distribution of order $\alpha \in [\gamma_1, \gamma_2]$. The weight function $w(\alpha)$ is normalized, so

$$\int_{\gamma_1}^{\gamma_2} w(\alpha) \, d\alpha = 1.$$

(4.2)

The idea of distributed-order differential equations was first introduced most probably by Caputo [9,10].

As Jiao et al. [8] showed recently, distributed-order derivatives can be discretized as

$$_0D_t^{w(\alpha)}f(t) \approx \mathbf{B}^{w(\alpha)}f_n,$$

(4.3)

and

$$\mathbf{B}^{w(\alpha)} = \sum_{k=1}^p \mathbf{B}^{\alpha_k} w(\alpha_k) \, \Delta\alpha_k,$$

(4.4)

where matrices $\mathbf{B}^{\alpha_k}$ are discrete analogues of fractional derivatives of orders α_k on a given grid—in our case, on a non-equidistant grid. The matrices $\mathbf{B}^{\alpha_k}$ for discrete differentiation of order α_k on a non-equidistant grid are obtained as described earlier, and then the matrix $\mathbf{B}^{w(\alpha)}$ for discrete distributed-order differentiation on the same non-equidistant grid is computed using relationship (4.4).

(a) Example 5: distributed-order relaxation equation

Let us consider the following initial value problem for the distributed-order relaxation equation:

$$_0D_t^{w(\alpha)}x(t) + bx(t) = f(t)$$

(4.5)

and

$$x(0) = 1,$$

(4.6)

where the distribution of the orders α is given by the function $w(\alpha) = 6\alpha(1 - \alpha)$ $(0 \le \alpha \le 1)$. To be able to use the matrix approach, we need a zero initial condition. Introducing an auxiliary
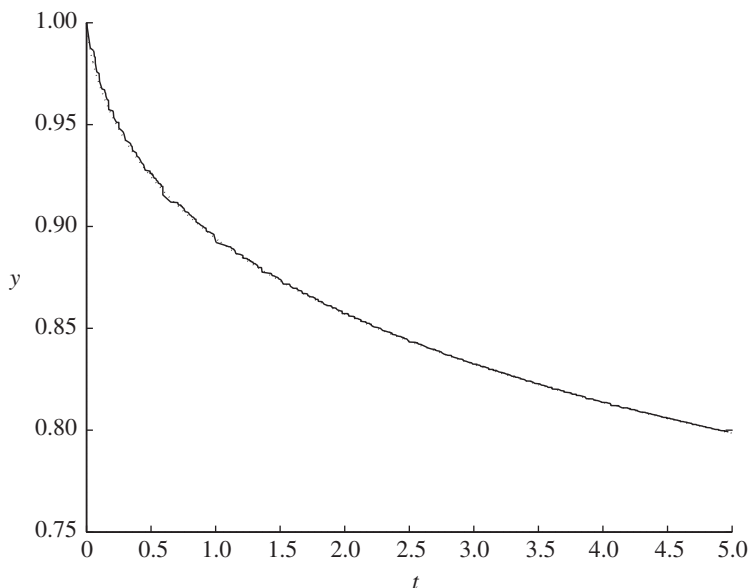
Figure 7. Solution of the distributed-order relaxation equation on equidistant (dotted line) and on non-equidistant grids (solid line).

function $u(t)$,

$$x(t) = u(t) + 1,$$

gives the following initial value problem for the new unknown $u(t)$:

$$_0D_t^{w(\alpha)}u(t) + bu(t) = f(t) - b \tag{4.7}$$

and

$$u(t) = 0. \tag{4.8}$$

The discretization of equation (4.7) gives, as usual in the matrix approach, the following system of algebraic equations in matrix form:

$$(\boldsymbol{B}^{w(\alpha)} + b\boldsymbol{E}_n)\boldsymbol{U}_n = \boldsymbol{F}_n, \tag{4.9}$$

where $\boldsymbol{U}_n$ is the vector of the values of $u(t)$ at the discretization nodes, and $\boldsymbol{F}_n$ is the vector of the values of the right-hand side, $f(t) - b$, at the same nodes; $\boldsymbol{E}_n$ is the identity matrix.

The results of computations for the case of 500 randomly generated nodes in the interval $[0, 5]$ for $b = 0.1$ are shown in figure 7 by the solid line. They are in agreement with the numerical solution obtained for the same number of uniformly spaced nodes (dotted line in figure 7).

(b) Example 6: distributed-order oscillator

Let us consider an initial value problem for the Bagley–Torvik equation, where the damping term is expressed in terms of distributed-order derivatives:

$$ay''(t) + b_0D_t^{w(\alpha)}y(t) + cy(t) = f(t), \quad f(t) = \begin{cases} 8 & (0 \le t \le 1), \\ 0 & (t > 1), \end{cases} \tag{4.10}$$

and

$$y(0) = y'(0) = 0. \tag{4.11}$$

Similar to example 5, we just replace continuous operators with their corresponding matrix-form discrete analogues on the considered non-equidistant grids, and the known and unknown
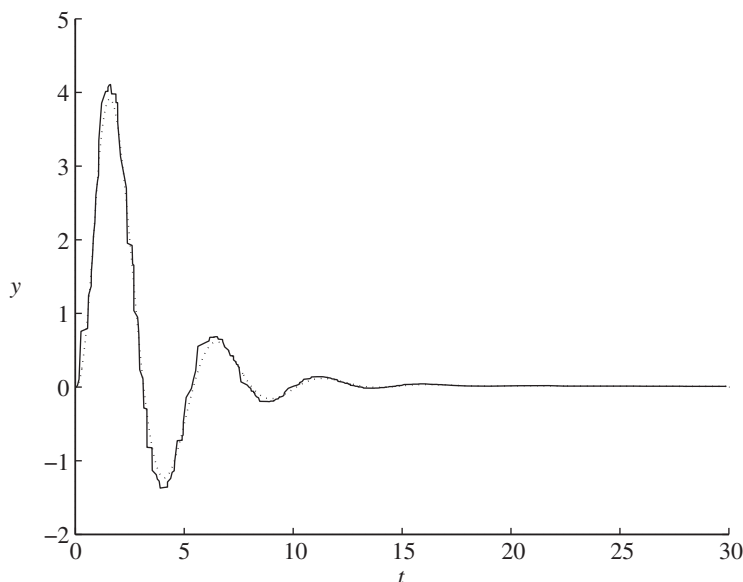
Figure 8. Solution of the distributed-order Bagley–Torvik equation for $A = 1$, $B = 1$, $C = 1$, $\varphi(\alpha) = 6\alpha(1 - \alpha)$, $\alpha \in [0, 1]$, on equidistant (dotted line) and on non-equidistant grids (solid line).

function by the vectors of their values in the discretization nodes. This gives the following algebraic system in matrix form:

$$(a\,\boldsymbol{B}_n^2 + b\,\boldsymbol{B}^{w(\alpha)} + c)\boldsymbol{Y}_n = \boldsymbol{F}_n. \tag{4.12}$$

The results of computations for the case of 400 randomly generated nodes in the interval $[0, 30]$ for $A = 1$, $B = 1$, $C = 1$, $\varphi(\alpha) = 6\alpha(1 - \alpha)$, $\alpha \in [0, 1]$, are shown in figure 8 by the solid line. They are in obvious agreement with the numerical solution obtained for the same number of uniformly spaced nodes (dotted line in figure 8).

The Matlab implementation of the matrix approach to discretization of distributed-order operators and numerical solution of distributed-order differential equations can be found in the update to the available toolbox [11].

5. Method of 'large steps'

We will illustrate the idea of the proposed 'method of large steps' on an easy-to-follow and sample problem, which allows exact analytical solution. In this section, we prefer using small snippets of the Matlab code in order to illustrate the simplicity of the procedure, and the idea of how 'large steps' is performed.

(a) Sample problem in the interval (0, 2)

It is worth remembering that we use the Caputo derivatives [1]. Let us consider the following sample problem for the large-steps method:

$$ {}_0^C D_t^{1/2} y(t) + y(t) = \frac{t^{0.5}}{\Gamma(1.5)} + t \quad (t > 0) \tag{5.1}$$

and

$$y(0) = 0. \tag{5.2}$$

One can easily verify that the exact solution of this problem is $y(t) = t$.
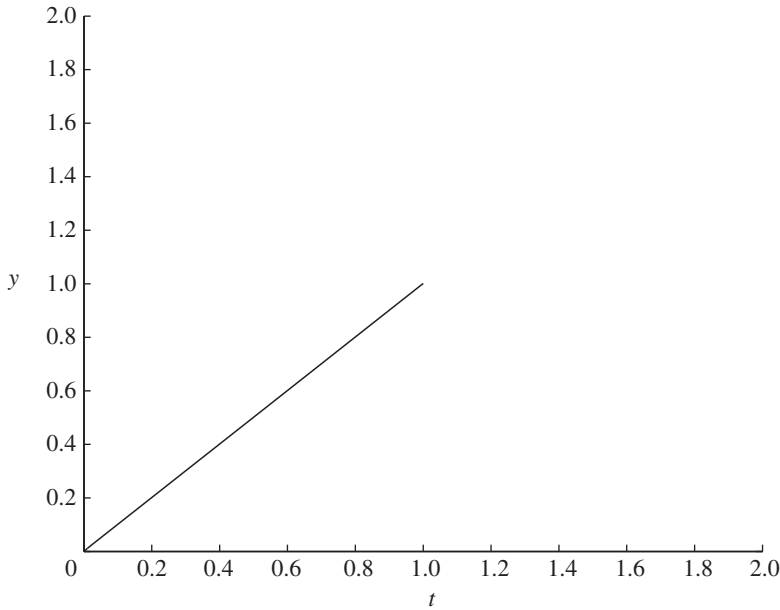
Figure 9. Solution of (5.1) and (5.2) after the first 'large step.'

(b) First 'large step': interval (0, 1)

We can solve the problem (5.1)–(5.2) numerically in the interval (0,1) using the recently developed matrix approach [2,7] and the corresponding MATLAB toolbox [2]:

```
clear all
h = 0.01;
t = 0:h:1;
N = 1/h + 1;
M = zeros(N,N);
M = ban(0.5, N, h) + eye(N,N);
F = (t.^(0.5)/gamma(1.5) + t)';
M = eliminator(N,[1])*M*eliminator(N,[1])';
F = eliminator(N,[1])*F;
Y = M\F;
Y0 = [0; Y];
plot (t,Y0,'b')
set(gca, 'xlim', [0 2], 'ylim', [0 2] )
grid on, hold on
```

So, we have solved (see figure 9) the previous problem in (0,1) and we know $y(t)$ for t in (0,1). How can we continue from the point at which we have arrived?

(c) Second 'large step': interval (1,2)

Taking into account that for $t > 1$ (we recall that we use the Caputo derivatives here)

$$
{}^C_0 D^{1/2}_t y(t) = {}^C_1 D^{1/2}_t y(t) + \frac{1}{\Gamma(0.5)} \int_0^1 \frac{y'(\tau)\,\mathrm{d}\tau}{(t-\tau)^{1/2}} \quad (t > 1) \tag{5.3}
$$

and that we already have $y(t) = t$ in the interval $(0,1)$, the problem (5.1)–(5.2) can be written as

$$_1^C D_t^{1/2} y(t) + y(t) = \frac{t^{0.5}}{\Gamma(1.5)} + t - \frac{1}{\Gamma(0.5)} \int_0^1 \frac{d\tau}{(t-\tau)^{1/2}} \quad (t > 1). \tag{5.4}$$

The integral in the last term can be easily evaluated as

$$\int_0^1 \frac{d\tau}{(t-\tau)^{1/2}} = 2t^{0.5} - 2(t-1)^{0.5} \quad (t > 1). \tag{5.5}$$

Now, we are ready to make the second 'large step,' i.e. solution in the interval $(1,2)$. In the interval $(1,2)$ (second step), we have to solve the following problem:

$$_1^C D_t^\alpha y(t) + y(t) = \frac{t^{0.5}}{\Gamma(1.5)} + t - \frac{2t^{0.5}}{\Gamma(0.5)} + \frac{2(t-1)^{0.5}}{\Gamma(0.5)} \quad (t > 1) \tag{5.6}$$

and

$$y(1) = 1. \tag{5.7}$$

To solve this problem using the matrix approach [2,7], we need to obtain zero initial conditions. For this, we substitute

$$y(t) = u(t) + 1, \tag{5.8}$$

and for the auxiliary function $u(t)$ we have the desired initial value problem with zero initial condition:

$$_1^C D_t^\alpha u(t) + u(t) = \frac{t^{0.5}}{\Gamma(1.5)} + t - \frac{2t^{0.5}}{\Gamma(0.5)} + \frac{2(t-1)^{0.5}}{\Gamma(0.5)} - 1 \quad (t > 1) \tag{5.9}$$

and

$$u(1) = 0. \tag{5.10}$$

Now we solve the problem for $u(t)$ using the same matrix approach toolbox, and plot the solution:

```
clear all
h = 0.01;
t = 1:h:2;
N = 1/h + 1;
M = zeros(N,N);
M = ban(0.5, N, h) + eye(N,N);
F = (t.^(0.5)/gamma(1.5) + t - 2*t.^(0.5)/gamma(0.5) ...
        + 2*(t-1).^(0.5)/gamma(0.5) - 1)';
M = eliminator(N,[1])*M*eliminator(N,[1])';
F = eliminator(N,[1])*F;
U = M \F;
U0 = [0; U];
Y0 = U0 + 1;
plot(t, Y0, 'g')
```

We see (figure 10) that finally we obtain the solution of the original problem in the interval $(0,2)$ using two 'large steps': the first step was numerical solution in $(0,1)$, and the second step was numerical solution in $(1,2)$. In the right-hand side of the equation for the interval $(1,2)$ two additional terms appeared as the result of considering fractional differentiation with a different lower terminal, namely $_1^C D_t^{1/2} y(t)$.
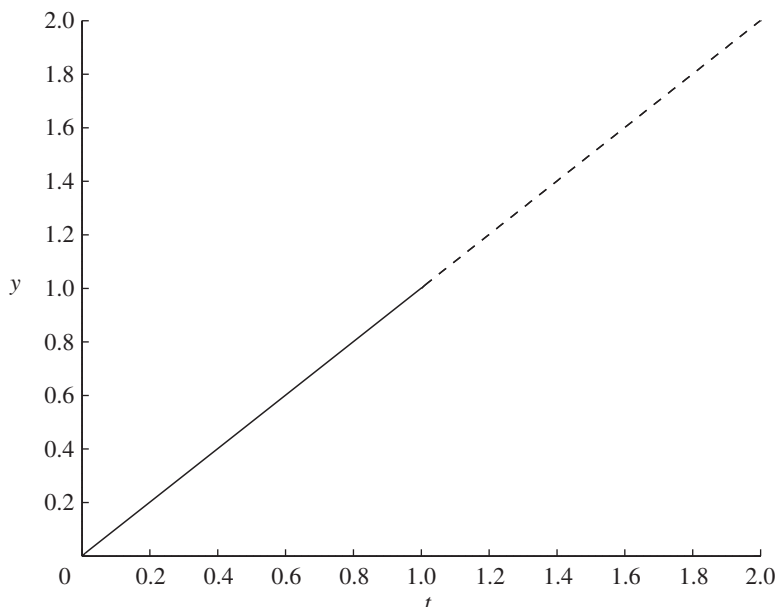
Figure 10. Solution of (5.6) and (5.7) after the second 'large step'.

6. Method of 'large steps': general scheme

In general, if we have considered the problem ($0 < \alpha < 1$)

$$\substack{C\\0}D_t^\alpha y(t) = f(y(t), t) \quad (t > 0) \tag{6.1}$$

and

$$y(0) = 0, \tag{6.2}$$

and obtained its solution in the interval $(0, a)$ (and the final value y_a at $t = a$), then we can use this for transforming the problem to

$$\substack{C\\a}D_t^\alpha y(t) = f(y(t), t) - {}_0P_a^\alpha y(t) \quad (t > a) \tag{6.3}$$

and

$$y(a) = y_a, \tag{6.4}$$

where

$${}_0P_a^\alpha y(t) = \frac{1}{\Gamma(1-\alpha)} \int_0^a (t-\tau)^{\alpha-1} y'(\tau)\,d\tau \quad (t > a) \tag{6.5}$$

is the contribution of the 'past' of the process $y(t)$ in the interval $[0, a]$ to the differential equations describing its current state in the interval $[a, b]$.

It is useful to note here that ${}_0P_a^\alpha y(t)$ can be evaluated as a fractional derivative of the function $y^*(t) = (1 - H(t-a))y(t)$, where $H(t)$ is the Heaviside unit step function:

$${}_0P_a^\alpha y(t) = \substack{C\\0}D_t^\alpha((1 - H(t-a))y(t)). \tag{6.6}$$

Introducing an auxiliary function $y(t) = u(t) + y_a$, we arrive at the problem with zero initial condition for the function $u(t)$, which can be solved numerically:

$$\substack{C\\a}D_t^\alpha u(t) = f(u(t) + y_a, t) - {}_0P_a^\alpha y(t) - y_a \quad (t > a) \tag{6.7}$$

and

$$u(a) = 0. \tag{6.8}$$

This process of making 'large steps' can be continued as long as necessary.

7. Linear fractional differential equations

Let us consider a linear fractional differential equation with constant coefficients in the interval $(0, b)$,

$$\sum_{k=1}^{m} p_k \, {}^{C}_{0}D^{\alpha_k}_t y(t) + p_0 y(t) = f(t) \quad (0 < t < b). \tag{7.1}$$

If we assume that $a_k < n$ $(k = 1, \ldots, m)$ and $n - 1 < \max_k a_k < n$, then we have to add n initial conditions, for example,

$$y^{(k)}(0) = 0, \quad k = 0, \ldots, n - 1. \tag{7.2}$$

The equation for the second 'large step' in the interval (a, b) will be

$$\sum_{k=1}^{m} p_k \, {}^{C}_{a}D^{\alpha_k}_t y(t) + p_0 y(t) = f(t) - \sum_{k=1}^{m} p_k \, {}_{0}P^{\alpha_k}_a y(t) \quad (a < t < b), \tag{7.3}$$

and the initial conditions for the second 'large step' will have the values of the final values $y^{(k)}_a$ of the solution in the first interval:

$$y^{(k)}(a) = y^{(k)}_a, \quad k = 0, \ldots, n - 1. \tag{7.4}$$

The initial conditions should be, as usual, transformed to zero initial conditions. For this, we have to introduce the auxiliary function $u(t)$:

$$y(t) = u(t) + \sum_{k=0}^{n-1} y^{(k)}_a \frac{t^k}{k!}. \tag{7.5}$$

This process of making 'large steps' can be continued as long as necessary.

8. Method of 'large steps' and the problem of initialization of fractional derivatives

Lorenzo & Hartley [12,13] raised the question about initialization of fractional derivatives. Their motivation was to use or recover the information about the process $y(t)$ in the interval $(0, a)$, if we consider fractional derivatives of $y(t)$ in (a, b).

It is worth noting that in the second 'large step' in the considered sample problem we used, in fact, the proper initialization of the fractional derivative in the interval $(1, 2)$ based on the known behaviour of $y(t)$ in $(0, 1)$.

In other words, proper initialization in the interval (a, b) can be done only when we know all values of $y(t)$ in the preceding interval $(0, a)$.

9. Concluding remarks

The presented extension of the matrix approach to discretization of non-integer order derivatives and integrals of constant and distributed order allows numerical solution of differential equations with such derivatives on non-equidistant grids.

The matrix approach proves to be a very easy, algorithmic, modular and convenient method that unifies numerical solution of many types of problems in various settings. In the examples in this paper we, for simplicity, used only ordinary differential equations with generalized derivatives; partial differential equations on non-equidistant grids can be solved using the technique published earlier [7].

Of course, the proposed method can be used for equations containing any mixture of left-sided, right-sided and two-sided derivatives of integer orders, constant non-integer orders, and distributed orders, on equidistant (uniform) or non-equidistant (non-uniform) grids. In all cases, those derivatives are simply replaced with their discrete analogues in the form of easily computable matrices.

The methods presented in this paper finally allow fractional-order differentiation and integration, and also distributed-order differentiation and integration, of non-uniformly sampled signals.

The proposed method of 'large steps' allows one to avoid in many situations the limitations of the 'short memory principle' and to obtain numerical solutions in large intervals with higher accuracy. At the same time, the method of 'large steps' finally allows the development of variable- and adaptive-step-length techniques for solving differential equations of non-integer order (ordinary and partial).

We would like to mention that other kinds of efforts towards using non-equidistant grids for numerical solution of fractional differential equations can be found in recent studies [14,15], and that some existing methods, such as the collocation method [16,17] or explicit numerical methods [18], can be reconsidered in terms of non-equidistant grids as well.

References

1. Podlubny I. 1999 *Fractional differential equations*. San Diego, CA: Academic Press.
2. Podlubny I. 2000 Matrix approach to discrete fractional calculus. *Fract. Calc. Appl. Anal.* **3**, 359–386. (doi:10.1016/j.jcp.2009.01.014)
3. Podlubny I, Skovranek T, Vinagre Jara BM. 2008 Matrix approach to discretization of ODEs and PDEs of arbitrary real order (a MATLAB toolbox). See http://www.mathworks.com/matlabcentral/fileexchange/22071.
4. Podlubny I, Skovranek T, Vinagre Jara BM. 2009 Matrix approach to discretization of ordinary and partial differential equations of arbitrary real order: the Matlab toolbox. In *Proc. ASME 2009 IDETC/CIE 2009, San Diego, CA, 30 August–2 September, 2009.* Article no. DETC2009-86944.
5. Samko S, Kilbas A, Marichev O. 1993 *Fractional integrals and derivatives: theory and applications*. Yverdon, Switzerland: Gordon and Breach Science Publishers.
6. Kilbas AA, Srivastava HM, Trujillo JJ. 2006 *Theory and applications of fractional differential equations*. North-Holland Mathematics Studies, vol. 204. Amsterdam, The Netherlands: Elsevier.
7. Podlubny I, Chechkin A, Skovranek T, Chen YQ, Vinagre Jara BM. 2009 Matrix approach to discrete fractional calculus. II. Partial fractional differential equations. *J. Comput. Phys.* **228**, 3137–3153. (doi:10.1016/j.jcp.2009.01.014)
8. Jiao Z, Chen YQ, Podlubny I. 2012 *Distributed-order dynamic systems*. Berlin, Germany: Springer.
9. Caputo M. 1969 *Elasticità e dissipazione*. Bologna, Italy: Zanichelli.
10. Caputo M. 1995 Mean fractional-order-derivatives differential equations and filters. *Annali dell'Universita di Ferrara* **41**, 73–84. (doi:10.1007/BF02826009)
11. Podlubny I. 2012 *Matrix approach to distributed-order ODEs and PDEs*. See http://www.mathworks.com/matlabcentral/fileexchange/.
12. Lorenzo CF, Hartley T. 2000 Initialized fractional calculus. Technical Report NASA/TP-2000-209943. Glenn Research Center.
13. Lorenzo CF, Hartley T. 2007 Initialization, conceptualization, and application in the generalized (fractional) calculus. *Crit. Rev. Biomed. Eng.* **35**, 447–553. (doi:10.1615/CritRevBiomedEng.v35.i6.10)
14. Ford N, Simpson A. 2001 The numerical solution of fractional differential equations: speed versus accuracy. *Numer. Algorithms* **26**, 333–346. (doi:10.1023/A:1016601312158)
15. Sprouse BP, MacDonald ChL, Silva GA. 2010 Computational efficiency of fractional diffusion using adaptive time step memory. In *Proc. 4th IFAC Workshop Fractional Differentiation and*

its Applications, Badajoz, Spain, 18–20 October 2010 (eds I Podlubny, B Vinagre, YQ Fen, V Feliu, I Tejado). Article no. FDA10-051. Kosice, Slovakía: Technical University of Kosice; and Badajoz, Spain: University of Extremadura.

16. Blank L. 1996 Numerical treatment of differential equations of fractional order. Numerical Analysis Report no. 287. Manchester Center for Computational Mathematics.

17. Dubois F, Mengué S. 2003 Mixed collocation for fractional differential equations. *Numer. Algorithms* **34**, 303–311. (doi:10.1023/B:NUMA.0000005367.21295.05)

18. Quintana Murillo J, Yuste SB. 2011 An explicit difference method for solving fractional diffusion and diffusion–wave equations in the Caputo form. *J. Comput. Nonlinear Dyn.* **6**, 021014. (doi:10.1115/1.4002687)