Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 192 (2003) 406–421
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Numerical methods for the solution of partial differential equations of fractional order

V.E. Lynch *, B.A. Carreras, D. del-Castillo-Negrete, K.M. Ferreira-Mejias, H.R. Hicks

Oak Ridge National Laboratory, Computational Sciences and Engineering, P.O. Box 2009, Oak Ridge, TN 37831 8070, USA

Received 31 January 2003; received in revised form 25 June 2003; accepted 10 July 2003

Abstract

Anomalous diffusion is a possible mechanism underlying plasma transport in magnetically confined plasmas. To model this transport mechanism, fractional order space derivative operators can be used. Here, the numerical properties of partial differential equations of fractional order α, $1 \leqslant \alpha \leqslant 2$, are studied. Two numerical schemes, an explicit and a semi-implicit one, are used in solving these equations. Two different discretization methods of the fractional derivative operator have also been used. The accuracy and stability of these methods are investigated for several standard types of problems involving partial differential equations of fractional order.

Keywords: Fractional derivatives; Partial differential equations; Anomalous diffusion; Plasma transport

1. Introduction

Brownian motion of particles results in molecular diffusion, which is characterized by the moments of the distribution of particle positions evolving in time as $\langle x^2(t) - \langle x(t)\rangle^2\rangle = Dt^{2v}$ with $2v = 1$. This process is called normal diffusion. The scaling with time of moments of tracer particle distributions has been studied for a variety of different dynamical conditions. It is not always found that particle diffusion is the dominant transport mechanism. When a scaling with time like the one described above is found but $2v \neq 1$, the process is called anomalous diffusion [1]. Anomalous diffusion may be a consequence of particles being trapped in certain positions along the trajectory, then we have $2v < 1$ and the process is called subdiffusion. If there are jets of particles along the trajectory, $2v > 1$ and the process is superdiffusion. See for example [2]. A continuous random walk with steps obeying a Levy distribution [3] is one example of superdiffusion.

In magnetically confined plasmas, transport is a complex process [4]. The basic mechanisms underlying plasma transport are not yet understood. By looking at several possible turbulence mechanisms in three-

* Corresponding author.
E-mail address: lynchve@ornl.gov (V.E. Lynch).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2003.07.008

dimensional calculations, it has been found that all of them lead to anomalous particle diffusion [5,6]. This result may be consistent with some experimental observations that are difficult to reconcile with a local diffusion process [4,7,8]. Therefore, anomalous diffusion may be present in plasma transport.

A study [5] of tracer particle evolution in three-dimensional pressure-gradient-driven plasma turbulence model has shown the existence of a regime with self-similar tracer particle distribution. In this regime, the moments of the tracer particle displacement scales with an exponent $v > 0.5$ indicating that the tracer particle transport is not diffusive but superdiffusive. Furthermore, the calculated tracer particle distribution has the characteristic properties of a Levy distribution with an algebraic decaying tail. This is further evidence of the non-diffusive character of the process.

The cause of the anomalous diffusion is the existence of long-range correlations in the dynamics and/or the presence of anomalously large particle displacements or trapping. From the latter perspective, the tracer particle transport in a plasma turbulence model can be interpreted as a random walk in which the particles can be trapped in eddies for a certain time and then they jump through successive hyperbolic points of the eddies. The trapping times and the displacements are described by broad probability functions with algebraic tails. In particular, for systems that exhibit anomalous diffusion caused by Levy flights [1], the probability distribution of particle displacements, $p(l)$, is broad in the sense that $\langle l^2 \rangle = \infty$. As it is well known, for these kinds of systems the central limit theorem cannot be applied; and as $N \to \infty$, the probability distribution function of $x = \sum_n^N l_n$, rather than being Gaussian, is a α-stable Levy distribution [9].

In deriving macroscopic equations for plasma transport from plasma turbulence models, Gaussian closures are assumed [4]. They lead to local diffusion operators with diffusion coefficients derived in terms of the plasma turbulence parameters. This closure cannot work in the case of Levy statistics for the particles. The class of operators that should describe the macroscopic transport should have solutions of the Levy type distributions. Fractional derivative operators verify this constraint. Therefore, we expect that the particle distribution function obeys fractional kinetics [5,10]. Under this assumption, a generic macroscopic function such as the particle density, $n(x,t)$, may be the solution of a partial differential equation of fractional order. Here we consider a simple form for such transport equations

$$\frac{\partial n}{\partial t} = F(x,n) + \chi D^{\alpha}[n(x,t) - n(0,t)], \quad t > 0, \;\; x > 0. \tag{1}$$

This equation is derived asymptotically from basic random walk models in [11]. In Eq. (1), D^{α} is a fractional derivative operator, which (following [12]) is defined as

$$D^{\alpha}(f) = \frac{1}{\Gamma(m-\alpha)} \frac{\partial^m}{\partial x^m} \int_0^x \frac{f(y)}{(x-y)^{\alpha-m+1}} \mathrm{d}y, \tag{2}$$

where $\alpha > 0$ and m is an integer such that $m > \alpha > m-1$. The function F is either a given source term or a reaction term expressed as a nonlinear function of n. The transport coefficient χ is taken to be a constant. As is clear from Eq. (1), in this paper we only consider fractional derivatives in space. In this case, there is a simple relation between the v exponent and the index α, $\alpha = 1/v$. Therefore for the problems considered here, the index α should be in the range $1 \leqslant \alpha \leqslant 2$ and $m = 2$.

Although there are several numerical methods developed to solve differential equations of fractional order [12–15], there is little information on the numerical solution of *partial* differential equations of fractional order. Here, we consider two time evolution methods and two different discretizations of the fractional derivative operator. The efficiency and accuracy of these schemes is analyzed.

The rest of this paper is organized as follows. In Section 2, two discretization methods for the fractional derivative operators are introduced. In Section 3 we describe the numerical schemes used in solving Eq. (1). In Section 4 the numerical test and solutions of the transport equilibrium equation are given and two more

problems are examined. In Section 5 a pulse evolution problem is considered and in Section 6 a reaction-fractional-diffusion equation is discussed. Finally, the conclusions of this paper are given in Section 7.

2. Discretization of the fractional derivatives

In solving Eq. (1), the space variable x is defined as a unit length interval, $x \in [0, 1]$. In this interval, we set up a grid of N points and define $\Delta = 1/N$. The first step towards a solution is to discretize the fractional derivative. Here, we follow the method given in [12]. First, it is useful to write explicitly the singular terms in the integral Equation (2) and write the fractional derivative in the following way [12]:

$$D^{\alpha}(f) = \frac{f(0)}{\Gamma(1-\alpha)}x^{-\alpha} + \frac{f'(0)}{\Gamma(2-\alpha)}x^{1-\alpha} + \frac{1}{\Gamma(2-\alpha)}\int_0^x \frac{f''(y)\mathrm{d}y}{(x-y)^{\alpha-1}}. \tag{3}$$

For our problem, we keep explicit the two first terms because they are directly determined by boundary conditions. The second step in the discretization is to write the integral in Eq. (3) as a sum of integrals. That is,

$$\int_0^x \frac{f''(y)\mathrm{d}y}{(x-y)^{\alpha-1}} = \sum_{j=0}^{i-1}\int_{x_j}^{x_{j+1}} \frac{f''(x-y)}{y^{\alpha-1}}\mathrm{d}y. \tag{4}$$

In [12], each integral in the sum is discretized the following way:

$$\int_{x_j}^{x_{j+1}} \frac{f''(x-y)}{y^{\alpha-1}}\mathrm{d}y \approx \frac{f(x-x_{j+1}) + f(x-x_{j-1}) - 2f(x-x_j)}{\Delta^2}\int_{x_j}^{x_{j+1}} \frac{\mathrm{d}y}{y^{\alpha-1}}. \tag{5}$$

As it is clear from Eq. (5), the second derivative is evaluated at position x_j. There are obvious alternative ways of evaluating this derivative as we discuss later in this paper. In [12], this discretization method is called the L2 method. We keep the same notation here. Just from integrating over y in Eq. (5) and substituting back in Eq. (3), we obtain

$$D^{\alpha}(f)_i = \frac{f(0)}{\Gamma(1-\alpha)}x_i^{-\alpha} + \frac{f'(0)}{\Gamma(2-\alpha)}x_i^{1-\alpha} + \frac{1}{\Gamma(3-\alpha)\Delta^{\alpha}}\sum_{j=0}^{i-1}\left[(j+1)^{2-\alpha} - j^{2-\alpha}\right](f_{i-j+1} + f_{i-j-1} - 2f_{i-j}). \tag{6}$$

Here, the subindex i refers to the space position x_i where the derivative is taken and $f_k \equiv f(k\Delta)$. We can now reorganize the sum and write the fractional derivative in the form

$$D^{\alpha}(f)_i = \frac{f(0)}{\Gamma(1-\alpha)}x_i^{-\alpha} + \frac{f'(0)}{\Gamma(2-\alpha)}x_i^{1-\alpha} + \sum_{j=-1}^{i} W_j(\alpha)f_{i-j}. \tag{7}$$

In Eq. (7), the weights $W_j(\alpha)$ are

$$\begin{cases} W_j = \frac{(j+2)^{2-\alpha} - 3(j+1)^{2-\alpha} + 3j^{2-\alpha} - (j-1)^{2-\alpha}}{\Delta^{\alpha}\Gamma(3-\alpha)}, \quad 1 \leqslant j \leqslant i-2, \\ W_{-1} = 1/(\Delta^{\alpha}\Gamma(3-\alpha)), \\ W_0 = (2^{2-\alpha} - 3)/(\Delta^{\alpha}\Gamma(3-\alpha)), \\ W_{i-1}(\alpha) = \frac{-2i^{2-\alpha} + 3(i-1)^{2-\alpha} - (i-2)^{2-\alpha}}{\Delta^{\alpha}\Gamma(3-\alpha)}, \\ W_i(\alpha) = \frac{i^{2-\alpha} - (i-1)^{2-\alpha}}{\Delta^{\alpha}\Gamma(3-\alpha)}. \end{cases} \tag{8}$$

In the limit $\alpha = 2$, the D^α operator becomes a second derivative corresponding to normal diffusion. The only weights that are non-zero are

$$W_{-1} = \frac{1}{\Delta^2}, \quad W_0 = -\frac{2}{\Delta^2}, \quad W_1 = \frac{1}{\Delta^2}. \tag{9}$$

They correspond to the Δ^2-accurate discretization of a second order derivative. There is a qualitative difference between the $\alpha = 2$ case and the non-integer α's. In the later case all grid points with $j < i$ are coupled through the sum in Eq. (7), although the strength of the coupling decreases with increasing distance between x_i and x_j. This decrease of the weights is illustrated in Fig. 1, where we have plotted the weights for a case with $\alpha = 3/2$.

In the limit $\alpha = 1$, the numerical representation of the first order derivative using the L2 scheme is a backward two-point derivative. This implies that the derivative is accurate to order Δ for $\alpha = 1$. Therefore, the accuracy of this discretization method depends on the value of α. Since this discretization might not be accurate enough for α near 1, a variant of the method of discretization developed by [12] was developed. To do so, let us go back to Eq. (5) and evaluate the integral in a more symmetric form. We use a four-point discretization of the second derivative in the integrand and we obtain

$$\int_{x_j}^{x_{j+1}} \frac{f''(x-y)}{y^{\alpha-1}}\,\mathrm{d}y \approx \frac{f(x-x_{j+2}) - f(x-x_{j+1}) + f(x-x_{j-1}) - f(x-x_j)}{2\Delta^2} \int_{x_j}^{x_{j+1}} \frac{\mathrm{d}y}{y^{\alpha-1}}. \tag{10}$$

Then doing the same manipulations as before, we reach an expression analogous to Eq. (7)

$$D^\alpha(f)_i = \frac{f(0)}{\Gamma(1-\alpha)} x_i^{-\alpha} + \frac{f'(0)}{\Gamma(2-\alpha)} x_i^{1-\alpha} + \sum_{j=-1}^{i+1} \hat{W}_j(\alpha) f_{i-j}, \tag{11}$$

where the new weights are

$$\begin{cases} \hat{W}_{-1}(\alpha) = \frac{1}{2\Delta^\alpha \Gamma(3-\alpha)}, \\ \hat{W}_0(\alpha) = \frac{2^{2-\alpha}-2}{2\Delta^\alpha \Gamma(3-\alpha)}, \\ \hat{W}_1(\alpha) = \frac{3^{2-\alpha}-2^{3-\alpha}}{2\Delta^\alpha \Gamma(3-\alpha)}, \\ \hat{W}_j(\alpha) = \frac{(j+2)^{2-\alpha}-2(j+1)^{2-\alpha}+2(j-1)^{2-\alpha}-(j-2)^{2-\alpha}}{2\Delta^\alpha \Gamma(3-\alpha)}, \quad 1 < j \leqslant i-2, \\ \hat{W}_{i-1}(\alpha) = \frac{-i^{2-\alpha}-(i-3)^{2-\alpha}+2(i-2)^{2-\alpha}}{2\Delta^\alpha \Gamma(3-\alpha)}, \\ \hat{W}_i(\alpha) = \frac{-i^{2-\alpha}+2(i-1)^{2-\alpha}-(i-2)^{2-\alpha}}{2\Delta^\alpha \Gamma(3-\alpha)}, \\ \hat{W}_{i+1}(\alpha) = \frac{i^{2-\alpha}-(i-1)^{2-\alpha}}{2\Delta^\alpha \Gamma(3-\alpha)}. \end{cases} \tag{12}$$

Note that one problem in this new scheme is that it requires knowing f_{-1}. However, this is not a problem for the problems considered here, because we always use zero derivative boundary conditions at the origin. Therefore, $f_{-1} = f_1$. We call this discretization method L2C.

It is interesting to compare the two discretization methods in the limit when α is an integer. For $\alpha = 1$, the L2 method gives

$$\left.\frac{\partial f}{\partial x}\right|_i = \frac{f_{i+1} - f_i}{\Delta}. \tag{13}$$

This is a forward derivative, with a numerical error of the order Δ. The L2C method gives

$$\left.\frac{\mathrm{d}f}{\mathrm{d}x}\right|_i = \frac{f_{i+1} - f_{i-1}}{2\Delta}. \tag{14}$$

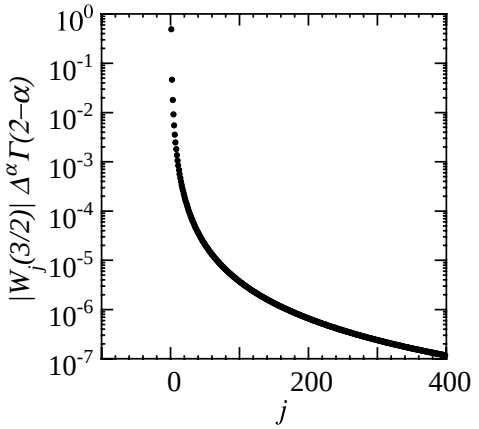

Fig. 1. L2 weights for $\alpha = 3/2$ and $n = 400$.

That is the centered derivative with a discretization error of order of Δ^2. On the other hand, in the limit $\alpha = 2$, the L2 method gives

$$\left.\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}\right|_i = \frac{f_{i+1} + f_{i-1} - 2f_i}{\Delta^2}. \tag{15}$$

This is a centered derivative with an error of the order of Δ^2. The L2C method gives

$$\left.\frac{\mathrm{d}^2 f}{\mathrm{d}x^2}\right|_i = \frac{f_{i+1} + f_{i-2} - f_i - f_{i-1}}{2\Delta^2} \tag{16}$$

with an accuracy of the order Δ. Therefore, the L2 method is more accurate than the L2C method for $\alpha = 2$, but the reverse happens for $\alpha = 1$. We can expect these methods to be useful in different ranges of α.

3. Numerical schemes

The range of independent variables is restricted to $0 \leqslant x \leqslant 1$, $t \geqslant 0$. The initial condition, $n(x, 0)$, is given in functional form with each type example in the paper. In solving Eq. (1), we take as boundary conditions $n_x(0, t) = 0$, $n(1, t) = 0$. The first boundary conditions removes one of the singular terms in the fractional derivative, Eq. (3), as we discuss below. These boundary condition are used in all the examples in the paper, except the second boundary condition is different for solutions of the reaction-fractional-diffusion equation propagating from right to left. For these calculations since the initial condition is $n(1, 0) = 1$, we use $n_x(1, t) = 0$ as the edge boundary condition. In Eq. (1), we consider the fractional derivative of the function $f(x, t) = n(x, t) - n(0, t)$. This removes the other singular term in the fractional derivative.

We have considered two numerical schemes for time advancing Eq. (1). One is a simple explicit scheme and the second one is semi-implicit. The explicit scheme can be written in the following way:

$$n_i^{t+\Delta t} = n_i^t + \Delta t \left[\chi \sum_{j=-1}^{j=i} W_j (n_{i-j}^t - n_0^t) + F(x_i, n_i^t) \right]. \tag{17}$$

Table 1

	0	1	2	3	4
0	$1-\lambda W_0-\lambda\kappa W_1$	$-\lambda W_{-1}+2^{-\alpha}\lambda\kappa W_1$	0	0	0
1	$-\lambda W_1$	$1-\lambda W_0$	$-\lambda W_{-1}$	0	0
2	0	$-\lambda W_1$	$1-\lambda W_0$	$-\lambda W_{-1}$	0
3	0	0	$-\lambda W_1$	$1-\lambda W_0$	$-\lambda W_{-1}$
4	0	0	0	$-\lambda W_1$	$1-\lambda W_0$

Here, the weights W_j are either the ones of the L2 method or the L2C method. The value of n at $i=0$ is advanced in time using the boundary condition of zero first derivative at the origin. Doing an expansion of the solution near the origin, we have

$$n(\delta)=n(0)+n'(0)\delta+n_\alpha\delta^\alpha. \tag{18}$$

Applying this expansion to the two first points in the grid and setting the first derivative to zero, we have

$$n_0^{t+\Delta t}=\frac{n_1^{t+\Delta t}-2^{-\alpha}n_2^{t+\Delta t}}{1-2^{-\alpha}}. \tag{19}$$

The weights are stored at the beginning of each calculation so they will not have to be calculated each time a fractional derivative is needed. Initially $j^{2-\alpha}$ is stored in an array and each weight is calculated using this array. This reduced the time taken for the calculations by a factor of 19 for the case of $N=2000$.

To be able to have a more efficient numerical scheme, we have also developed a semi-implicit scheme, by extracting the elements in the expansion in Eq. (17) that form a tridiagonal matrix and using them in the time advanced part of the equation. We can write the semi-implicit scheme in the following way:

$$\begin{aligned} &n_i^{t+\Delta t}-\Delta t\chi\left[W_1 n_{i-1}^{t+\Delta t}+W_0 n_i^{t+\Delta t}+W_{-1}n_{i+1}^{t+\Delta t}\right]\\ &\quad=n_i^t+\Delta t\left[\chi\sum_{j=2}^{i}W_j(n_{i-j}^t-n_0^t)-\chi(W_{-1}+W_0+W_1)n_0^t+F(x_i,n_i^t)\right]. \end{aligned} \tag{20}$$

Using Eq. (19), the term $n_0^{t+\Delta t}$ in the left-hand side can be calculated in terms of the values of the density in the first two grid points. In this way, we have a tridiagonal matrix on the left-hand side. Table 1 shows the upper left hand corner of this matrix.

In Table 1, $\lambda\equiv\Delta t\chi$ and $\kappa\equiv(1-2^{-\alpha})^{-1}$. The weights for the semi-implicit scheme are the same as for the explicit scheme, namely, either (8) or (12).

The tridiagonal matrix is inverted by using the Thomas Algorithm of LU decomposition for tridiagonal systems [16].

Variants of these two schemes that are Δt^2 accurate have also been implemented. They consist of a two-step process. In the first step, the density is advanced to time $t+\Delta t/2$. This density is then used to calculate the right-hand side for the full time step advance.

4. Numerical tests using an analytic model for the equilibrium transport equation

To test the numerical schemes, it is important to use simple analytical models. For that purpose, we first consider the case of evolution of a density profile to an equilibrium solution. The equilibrium density profile is the solution of the equation,

$$S(x)+\chi D^\alpha(n-n_0)=0. \tag{21}$$

Here, $S(x)$ is a given source function and χ is a constant transport coefficient that we take to be 1. In order to have a simple solution of the equilibrium equation, we use a simple form for the source function, $S(x) = 1 - x^{2-\alpha}$, In this case, Eq. (21) can be analytically solved for any value of $\alpha > 1$ and the solution for the density function is

$$n(x) = \frac{1 - x^{\alpha}}{\Gamma(1+\alpha)} - \frac{\Gamma(3-\alpha)}{2}(1 - x^2). \tag{22}$$

If now we go back to Eq. (1) with $F = S(x)$, for a given initial condition, $n(x,0)$, not too far from the equilibrium density profile in Eq. (22), the density evolves towards this equilibrium solution. We calculate $n(x,0) = -D^{-\alpha}(S(x)/\chi) + a + bx$, choosing a and b so that the initial condition satisfies the boundary conditions. We have studied this evolution for different values of α and determined the asymptotic solution for large t. Having an analytical solution has also allowed us to determine the numerical error for the different numerical discretizations and time evolution schemes. We calculate the error as

$$\delta n = \sqrt{\frac{\sum_{i=0}^{N}[n_n(x_i) - n(x_i)]^2}{\sum_{i=0}^{N} n(x_i)^2}}, \tag{23}$$

where n_n is the numerical solution in the large t regime when we find that the calculation has converged to a steady state and $n(x)$ is given by Eq. (22).

The calculated error for a case with $\alpha = 3/2$ and $N = 400$ is 0.00148 for both the explicit and semi-implicit methods for small Δt. The calculations were stopped when the solutions reached steady state for all values of α. There is a stability limit, $\Delta t < \Delta^{\alpha}/\chi$, on the time step that can be used with the explicit scheme. For this case, the maximum step is $\Delta t = 5 \times 10^{-5}$ which is below the stability limit of 1.25×10^{-4}. The semi-implicit method allows us to use larger time steps without a large increase in the errors. Time steps of the order of $\Delta t = 1$ can be used without loss of accuracy.

The number of operations in the right-hand side of Eqs. (17) and (20) for both numerical schemes is very large because all grid points are coupled. Therefore, the relative increase of overhead operations in the semi-implicit method is relatively small. The time taken by the code for the same step size of 10^{-5} going to $t = 10$ increases from 41.05 min for the explicit to 42.21 min for the semi-implicit. Of course, increasing the time step, the time taken for the calculation is reduced by a factor inversely proportional to the step size. Therefore, the semi-implicit scheme leads to considerably faster results.

We also tested the accuracy of the L2 and L2C discretization methods. We use the semi-implicit method for a grid of $N = 400$ points and a time step $\Delta t = 2 \times 10^{-6}$ to ensure accuracy for all values of α. In Fig. 2, we have plotted the error as a function of α for a scan varying α between 1 and 2. The results are consistent with the discussion at the end of Section 2. For $\alpha <\approx 1.2$, the L2C scheme has a smaller error than the L2. For $\alpha >\approx 1.6$, the L2 scheme has a smaller error than the L2C. In the intermediate region, both methods lead to similar results.

The L2 and L2C methods have similar results with all grid sizes tested. Table 2 shows the results of changing the grid by factors of two from $N = 100$ to $N = 3200$ for three values of α with $\Delta t = 2\ 10^{-6}$. For $\alpha = 1.1$, the L2C scheme has a smaller error than the L2 for all grid sizes. For $\alpha = 1.9$, the L2 scheme has a smaller error than the L2C for all grid sizes where the time step is below the stability limit. Errors from the two largest grids were omitted from the table for $\alpha = 1.9$ because the time step used is above the stability limit for those grid sizes. For $\alpha = 1.5$, the L2 scheme has a slightly smaller error than the L2C for grids below $N = 800$, but the largest grids have a smaller error with the L2C method.

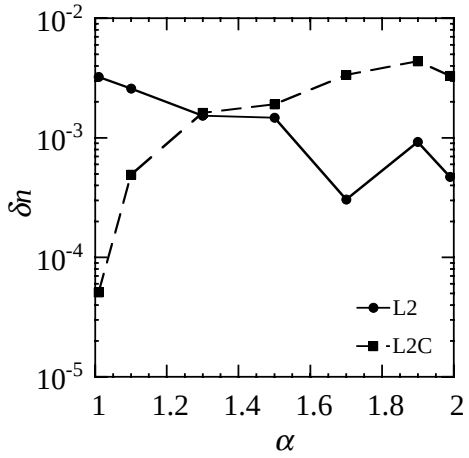

Fig. 2. Error as calculated by Eq. (23) for L2 and L2C methods for $N = 400$ and $\Delta t = 2 \times 10^{-6}$ as a function of α.

Table 2

N	α	δn(L2)	δn(L2C)
100	1.1	0.0099700	0.0021538
200	1.1	0.0050403	0.0010675
400	1.1	0.0025800	0.00049200
800	1.1	0.0013526	0.00019806
1600	1.1	0.00073861	6.3133e – 05
3200	1.1	0.00043167	6.1263e – 05
100	1.5	0.0044301	0.0082408
200	1.5	0.0025417	0.0040501
400	1.5	0.0014769	0.0019081
800	1.5	0.00089989	0.00082656
1600	1.5	0.00059500	0.00028872
3200	1.5	0.00043642	7.0800e – 05
100	1.9	0.0059026	0.016662
200	1.9	0.0024905	0.0086793
400	1.9	0.00092500	0.0042800
800	1.9	0.00023665	0.0012937

The rate of convergence of the solution depends on both Δ and Δt. Tables 3 and 4 show the error as a function of time for calculations varying Δt three orders of magnitude for $\alpha = 1.5$ and $N = 400$. For this grid size, the error of the converged solution is less for the L2 method shown in Table 3 than for the L2C shown in Table 4, but both methods converge at approximately the same time.

5. Time evolution of a pulse

Let us consider another problem, the time evolution of a pulse in a system governed by Eq. (1). For this problem, the basic equation to be solved is a linear equation without source term

$$\frac{\partial n}{\partial t} = \chi D^{\alpha}[n(x,t) - n(0,t)] \tag{24}$$

Table 3

t	$\delta n(\Delta t = 2 \times 10^{-6}$, L2)	$\delta n(\Delta t = 2 \times 10^{-5}$, L2)	$\delta n(\Delta t = 2 \times 10^{-4}$, L2)	$\delta n(\Delta t = 2 \times 10^{-3}$, L2)
1.	0.00073504	0.00071151	0.00049513	0.0016257
2.	0.0013838	0.0013773	0.0013013	0.00034482
3.	0.0014655	0.0014643	0.0014474	0.00083649
4.	0.0014755	0.0014753	0.0014719	0.0011701
5.	0.0014767	0.0014767	0.0014761	0.0013315
6.	0.0014769	0.0014769	0.0014768	0.0014082
7.	0.0014769	0.0014769	0.0014769	0.0014445
8.	0.0014769	0.0014769	0.0014769	0.0014616
9.	0.0014769	0.0014769	0.0014769	0.0014697
10.	0.0014769	0.0014769	0.0014769	0.0014735

Table 4

t	$\delta n(\Delta t = 2 \times 10^{-6}$, L2C)	$\delta n(\Delta t = 2 \times 10^{-5}$, L2C)	$\delta n(\Delta t = 2 \times 10^{-4}$, L2C)	$\delta n(\Delta t = 2 \times 10^{-3}$, L2C)
1.	0.0022849	0.0023289	0.0027439	0.0041704
2.	0.0019542	0.0019655	0.0021345	0.0036145
3.	0.0019138	0.0019160	0.0019695	0.0031835
4.	0.0019088	0.0019092	0.0019248	0.0028592
5.	0.0019082	0.0019083	0.0019127	0.0026171
6.	0.0019081	0.0019082	0.0019094	0.0024366
7.	0.0019081	0.0019081	0.0019085	0.0023021
8.	0.0019081	0.0019081	0.0019082	0.0022019
9.	0.0019081	0.0019081	0.0019082	0.0021272
10.	0.0019081	0.0019081	0.0019081	0.0020715

with the initial condition that is localized in x space. Depending on the value of α, the evolution of the pulse is a combination of a decay and advection of this pulse. In the case $\alpha = 1$, the problem is pure advection and Eq. (24) is just

$$\frac{\partial n}{\partial t} = \chi \frac{\partial n}{\partial x}. \tag{25}$$

The general solution of this equation is $n(x,t) = n_0(x + \chi t)$. That is, the initial pulse, whatever shape it has, moves inward at a constant velocity χ without changing its shape. In the numerical calculations, we use a Gaussian pulse as an initial condition centered at $x_0 = 0.5$ with a given width W, $n(x,0) = \exp[-(x - x_0)^2/(2W)]$. Since the general solution of Eq. (25) is just a shifted Gaussian, the value at the peak of the pulse should be constant in time. In the calculations presented here, we have used an initial width $W = 0.1$ and a transport coefficient $\chi = 0.05$. In Fig. 3, we have plotted, as a function of time, the maximum value of n for three different numbers of spatial grid points and the two numerical schemes.

For both numerical schemes, as we increase the number of grid points, the time step must decrease in order to verify $\Delta t \leqslant \Delta/\chi$. Using the L2 method, there is a serious accuracy problem, the maximum value of the pulse decreases as it moves out. The decay rate, caused by numerical diffusion, is significant and depends linearly with Δ (the system is only Δ accurate). To reduce the error to an acceptable value (a few percents during the motion out of the box), it is necessary to use 10^4 points. This makes the calculations impossibly slow. However, the L2C method, Δ^2 accurate, gives very accurate results for all the grid sizes considered. In Fig. 3, the effect of the change in grid is hardly visible because all curves are close each other. Although with the L2 method there is strong distortion of the pulse, the propagation velocity of the pulse is equally accurate using either discretization method.

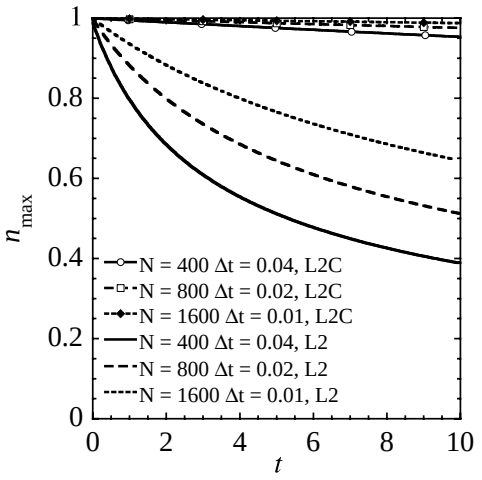

Fig. 3. Maximum value of $n(x,t)$ for three different numbers of grid points for the L2 and L2C methods.

For $1 < \alpha \leqslant 2$, Eq. (24) can be solved [17] in an infinite domain and the solution for an initial condition of the form $n(x,0) = \delta(x)$ is

$$n(x,t) = \frac{1}{(\chi t)^{1/\alpha}} \Phi\left[\frac{x}{(\chi t)^{1/\alpha}}\right] \tag{26}$$

with

$$\Phi(\zeta) = \frac{1}{\pi}\int_0^\infty \cos\left[\sin\left(\frac{\alpha\pi}{2}\right)k^\alpha + k\zeta\right] \exp\left[\cos\left(\frac{\alpha\pi}{2}\right)k^\alpha\right] \mathrm{d}k. \tag{27}$$

For a well-localized initial condition and for times short enough that the effects of the boundaries are not affecting the solution, we can expect that Eq. (26) gives a good representation of the numerical result. The self-similar property reflected in Eq. (26) can then be used as a measure of the accuracy of the solution. A way of visualizing the accuracy of the solution is plotting $(\chi t)^{1/\alpha} n(x,t)$ versus $x/(\chi t)^{1/\alpha}$ at different times within the time interval of self-similar evolution; we should obtain a single curve.

We have carried the calculation of the evolution of a pulse starting from a Gaussian function centered at $x = 0.5$ and with a width $W = 0.03$. We have used the semi-implicit numerical scheme with the L2 discretization. The self-similarity of the solution is shown in Fig. 4 for $\alpha = 2$ and for $\alpha = 1.999$. In this figure, we have plotted a sequence of solutions at different times. We can see that all curves collapse on a single one showing that the solutions verify the self-similarity property.

Let us now consider a case with α away from the integer values, for instance $\alpha = 1.5$. Using $\chi = 1$, $W = 0.002$, and $N = 1000$, we compare in Fig. 5 the numerical solution to the analytical one for different times during the evolution. We can see that the agreement for the bulk of the solution is very good. In this figure it is difficult to see the agreement on the tails. To have a more quantitative comparison and in order to test the two discretization methods, we will calculate the error, as given by Eq. (23), for several times during the calculation. The error has been calculated over the range $0.3 \leqslant x \leqslant 0.7$ to avoid possible distortion of the estimate by the effects of the boundary. This calculated error for the L2 and L2C methods is plotted in Fig. 6 versus α for the maximum time, $t = 100W^\alpha$. Again we reach the same conclusion as in the previous section. For $\alpha < 3/2$, the L2C method is more accurate than the L2 and the reverse for $\alpha > 3/2$.

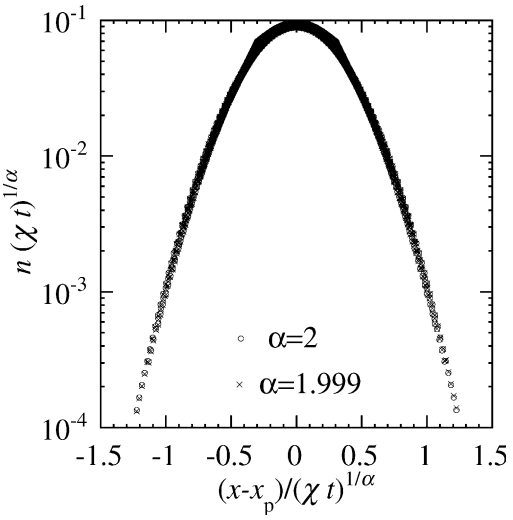

Fig. 4. Self-similarity of n profiles near $\alpha = 2$.

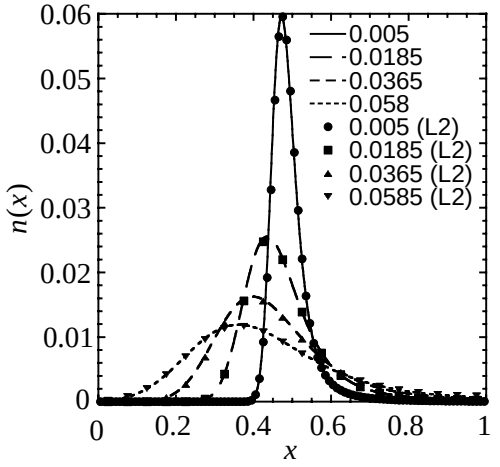

Fig. 5. Comparison of n from Eq. (26) with the L2 method at different times for $\alpha = 1.5$, $N = 1000$ and $\chi = 0.0005$.

By choosing the more accurate method in each range of α, we can obtain accurate results for the pulse decay problem.

6. Reaction-fractional-diffusion equations

In the previous sections, we have been concerned with the accuracy of the solutions from integration of the linear partial differential equation of fractional order. Here we will consider an example of a nonlinear partial differential equation of fractional order. We have investigated the combined effects of time step and space grid resolution for the case of the reaction-fractional-diffusion equations. That is,

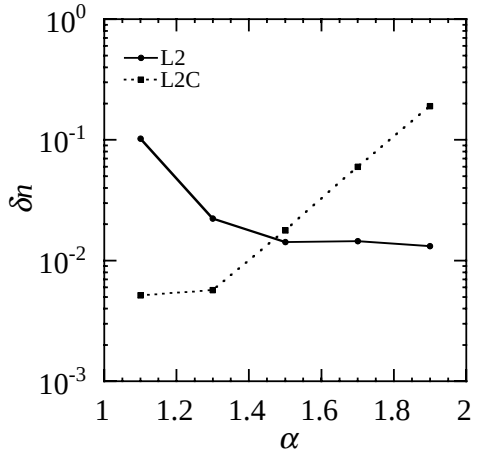

Fig. 6. Error from Eq. (26) and the numerical methods for different α with $N = 1000$ and $\chi = 0.0005$.

$$\frac{\partial n}{\partial t} = \gamma n(1-n) + \chi D^{\alpha}[n(x,t) - n(0,t)]. \tag{28}$$

For $\alpha = 2$, this equation is the Fisher–Kolmogorov equation [18]. The generalization of Fisher–Kolmogorov equation to fractional spatial derivatives has been studied in [17], here we concentrate on the accuracy of its numerical integration.

The Fisher–Kolmogorov equation has front propagating solutions. These solutions are self-similar solutions of the form $n\ (x + ct)$, where c is the velocity of the front. We can plot the solution at different times and shift the x-scale by $x \to x + ct$, where c is the velocity of the front. The solutions will appear to be a single curve.

The generalized Fisher–Kolmogorov equation also has propagation solutions. However, their properties depend on the direction of propagation. This is not surprising because the fractional operator is intrinsically asymmetric in x-space. Solutions propagating from left to right have a complicated behavior and strict self-similarity is not maintained. The properties of these front solutions are discussed in [17]. However, solutions propagating from right to left have sharp fronts with exponentially decaying tails and they propagate in a self-similar manner. Therefore, we consider the latter type of fronts for our numerical studies.

If self-similar propagating solutions exist, we can transform Eq. (28) into a differential equation of fractional order by changing variables, $y = x + ct$, and we have

$$c\frac{\mathrm{d}n}{\mathrm{d}y} = \gamma n(1-n) + \chi D^{\alpha}[n(y) - n(0)]. \tag{29}$$

For fronts decaying exponentially, we can apply the leading edge approximation [18] and solve Eq. (29). This solution allows us to determine the minimum velocity of the front solution [18]. This minimal velocity is,

$$c_{\min} = \frac{\alpha\gamma}{\alpha - 1}\left[(\alpha - 1)\frac{\chi}{\gamma}\right]^{1/\alpha}. \tag{30}$$

For $\alpha = 2$, Eq. (30) recovers the result for the Fisher–Kolmogorov equation, $c_{\min} = 2\sqrt{\gamma\chi}$.

As in the case of the Fisher–Kolmogorov equation, and for initial conditions with a width of the front smaller than the width for the minimal propagation velocity solution, $W_{\min} = [(\alpha - 1)\chi/\gamma]^{1/\alpha}$, the right to left

propagating fronts do so at a velocity very close to the minimal velocity. We will use the determination of the velocity of the front as a way of calculating the accuracy of the discretization methods and of the semi-implicit numerical scheme. We measure this velocity c by a linear fit to the position of the $n = 0.5$ as a function of time.

In the numerical calculations, we use as initial condition

$$n(x,0) = \frac{1}{2}\left\{1 + \tanh\left(\frac{x - 0.9}{2W}\right)\right\}. \tag{31}$$

To have a sharp enough front during the evolution, we have used a small value for the transport coefficient, $\chi = 5 \times 10^{-4}$, and $\gamma = 1/3$. This gives a velocity $c_{\min} = 0.0258$. After a short transient, the shape of the front is established and it propagates at a constant velocity.

For $\alpha = 1.9$ and using the L2 discretization method, we have calculated the velocity of the front, moving from right to left, for a variety of values of the time step and grid sizes. The results are shown in Fig. 7. At first look, the results are surprising. For low spatial resolution, Δ large, we get values of the velocity, which are fairly independent of the grid and close to 0.020. As we increase the spatial resolution with constant time step, we reach a point at which the value of the velocity begins to diverge. The reason is that the time step becomes large for the spatial resolution given the velocity of propagation of the information. The important parameter in these numerical studies is $\eta \equiv \Delta^{\alpha}/\chi\Delta t$. This parameter must be greater than one to get effective transmission of information. We can see that when we plot all the results in Fig. 7 as a function of only one parameter, η, they collapse onto a single curve as shown in Fig. 8.

The value of c as a function of the numerical parameter η is well described by a function of the form

$$c = \frac{\hat{c}}{1 + \lambda/\eta}. \tag{32}$$

For the particular case shown in Fig. 8, $\lambda = 0.0284$ and $\hat{c} = 0.020$. $\hat{c}$ is the numerically converged value of the velocity in the limit of infinite space-time resolution. This result indicates that the calculation of the front velocity is accurate and well converged within an error of order ε when the time step verifies the condition $\Delta t < (\varepsilon\Delta^{\alpha})/(\lambda\chi)$.

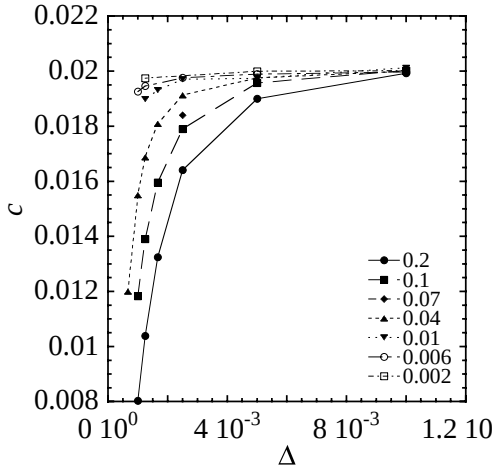

Fig. 7. Velocity of front as a function of Δ for different Δt.

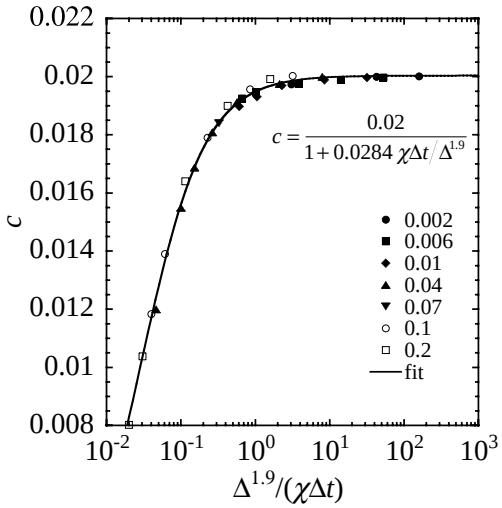

Fig. 8. Velocity of front as a function of the parameter in Eq. (32) for the L2 method.

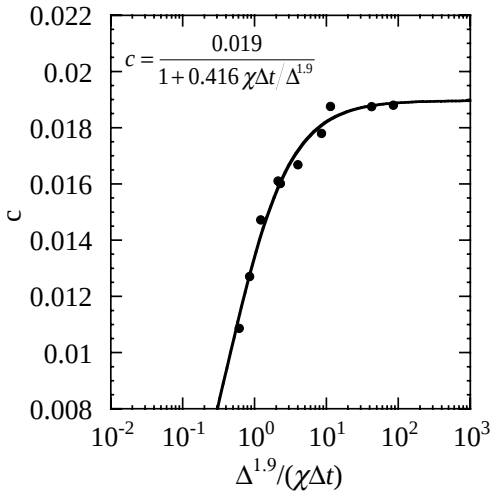

Fig. 9. Velocity of front as a function of the parameter in Eq. (32) for the L2C method.

We can now repeat the same calculation for the L2C method. We have used different values for the time step and grid size covering a similar range of values as before. There is also a unique curve summarizing all the values of the calculated velocity that can be fitted by Eq. (32). The results and the fit are shown in Fig. 9. In this case, $\lambda = 0.416$ and $\hat{c} = 0.019$.

The converged value of the velocity for the L2C method is about a 5% lower than the converged velocity with the L2 method. However, the most dramatic result is that the restriction on time step is a factor of 20 higher for the L2C method than for the L2, at least for this value of α. Therefore, also in the case of reaction-fractional-diffusion equations, at $\alpha = 1.9$ the L2 method is more efficient than the L2C method, because, to achieve the same accuracy, we need a time step ten times lower for the L2C than for the L2.

Let us now consider the case of front propagation near $\alpha = 1$. For $\alpha = 1.1$, $\chi = 5 \times 10^{-3}$, and $\gamma = 1/3$, let us consider the time evolution of the front. Two calculations for the L2 and L2C method are performed

with the semi-implicit scheme for the same parameters with a time step $\Delta t = 0.001$ and $N = 400$. For the L2 method, the front propagates in a self-similar manner with a velocity of 0.0104. The L2C method gives a velocity of 0.117. Using Eq. (30), the velocity from the analytical model for $\alpha = 1.1$ is 0.009933. The evolution of the front is somewhat faster for the L2 method, while the L2C gives a more accurate answer. In any case, the difference in velocity between the L2 and L2C methods is small.

The self-similarity of the front is maintained very accurately in both calculations. In Figs. 10 and 11, we have plotted the self-similarity of the L2 and L2C methods respectively. After the fast initial readjustment at $t = 1$, it is impossible to distinguish between the curves at different times. The self-similarity is maintained for both methods until the front reaches the origin.

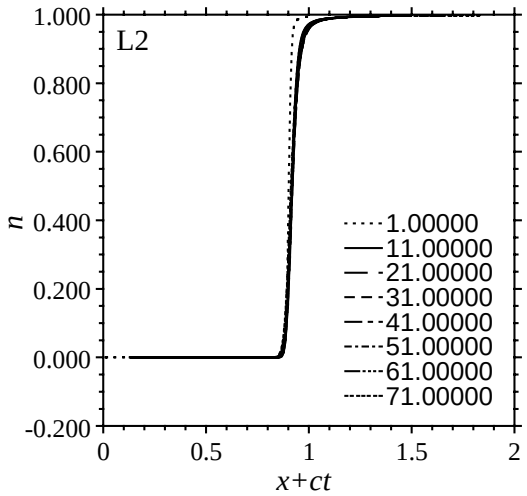

Fig. 10. Self-similarity of front at different times for $\chi = 5 \times 10^{-3}$, and $\gamma = 1/3$ for the L2 scheme.

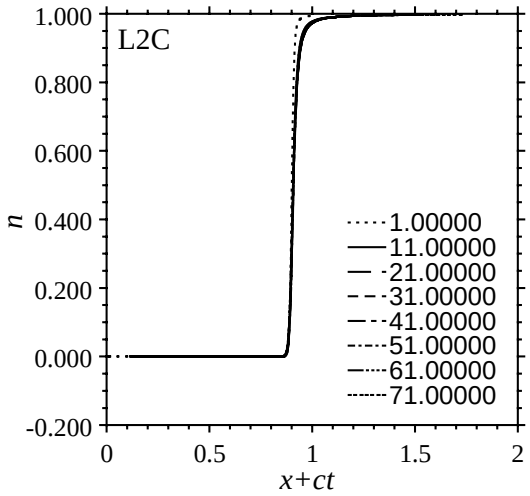

Fig. 11. Self-similarity of front at different times for $\chi = 5 \times 10^{-3}$, and $\gamma = 1/3$ for the L2C scheme.

7. Conclusions

We have considered a simple transport problem with anomalous diffusion modeled by a partial differential equation of fractional order α and studied two different numerical schemes for solving the equation and two different discretization methods for the fractional operators. We have applied these methods to three different types of problem. We found that efficient and accurate calculations are possible by choosing the proper numerical approach.

The semi-implicit method is more effective than the explicit method of numerically advancing the equation. With the former, larger time steps are always possible than the latter. However, constraints on the time step must be imposed to maintain the accuracy of the semi-implicit method.

Two discretization methods of the fractional derivative operators have been considered. Over this range of problems, the L2 method is the most accurate for $\alpha > 1.5$ and the L2C method is the most accurate for $\alpha < 1.5$. Around $\alpha = 1.5$ both methods have similar accuracy.

Therefore, using the semi-implicit method and switching discretization methods around $\alpha = 1.5$, we obtain optimal results for stability, efficiency, and accuracy of the numerical scheme.

Acknowledgements

This research is sponsored by Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U.S. Department of Energy under contract number DE-AC05-00OR22725. K.M.F. was supported by the Energy Research Undergraduate Laboratory Fellowship program of the U.S. Department of Energy.

References

[1] M.F. Shlesinger, G.M. Zaslavsky, J. Klafter, Nature 363 (1993) 31.
[2] D. del-Castillo-Negrete, Phys. Fluids 10 (1998) 576.
[3] E.W. Montroll, M.F. Shlesinger, in: J. Lebowitz, E. Montroll (Eds.), Studies in Statistical Mechanics, North-Holland, Amsterdam, 1984, p. 1.
[4] B.A. Carreras, IEEE Trans. Plasma Sci. 25 (1997) 1281.
[5] B.A. Carreras, V.E. Lynch, G.M. Zaslavsky, Phys. Plasmas 8 (2001) 5096.
[6] B.A. Carreras, V.E. Lynch, L. Garcia, M. Edelman, G.M. Zaslavsky, Chaos 14 (2003), to be published.
[7] N.J.L. Cardozo, Plasma Phys. Contr. Fusion 37 (1995) 799.
[8] K. Gentle, R.V. Bravenec, G. Cima, H. Gasquet, G.A. Hallock, P.E. Phillips, D.W. Ross, W.L. Rowan, A.J. Wootton, T.P. Crowley, J. Heard, A. Ourona, P.M. Schoch, C. Watts, Phys. Plasmas 2 (1995) 2292.
[9] G. Samorodnitsky, M.S. Taqqu, Stable Non-Gaussian Random Processes, Chapman & Hall, New York, 1994.
[10] G.M. Zaslavsky, Chaos 4 (1994) 25.
[11] R. Metzler, J. Klafter, Phys. Rep. 339 (2000) 1–77.
[12] K.B. Oldham, J. Spanier, The Fractional Calculus, Academic Press, New York, 1974.
[13] Luise Blank, Nonlinear World 4 (1997) 473–490.
[14] Alberto Carpinteri, Francesco Mainardi, Fractals and Fractional Calculus in Continuum Mechanics, Springer-Verlag, New York, 1997.
[15] Kai Diethelm, Elec. Trans. Numer. Anal. 5 (1997) 1.
[16] C. Pozrikidis, Numerical Computation in Science and Engineering, Oxford University Press, Oxford, 1998.
[17] D. del-Castillo-Negrete, B.A. Carreras, V.E. Lynch, Phys. Rev. Lett. 91 (2003) 18302.
[18] J.D. Murray, Mathematical Biology, Springer-Verlag, New York, 1989.