Proceedings of the Czech–Japanese Seminar in Applied Mathematics 2006
Czech Technical University in Prague, September 14-17, 2006
pp. 9–24

IMPROVED NUMERICAL METHOD FOR MULTIDIRECTIONAL FRACTIONAL ADVECTION-DISPERSION EQUATION IN 1-D AND 2-D WITH GENERAL BOUNDARY CONDITIONS

PETR BEDNAŘÍK[1]

Abstract. In this work we develop a finite difference method for the solution of the 1-D two-sided fractional advection-dispersion equation which enables more general boundary conditions than found in literature. Based on the 1-D method a generalization is developed into 2-D, where we solve the most general equation with a multidirectional fractional differential operator. The method is demostrated on several numerical examples.

Key words. 1-D and 2-D multidirectional fractional advection-dispersion equation, fractional diffusion, fractional partial differential equations, finite difference method, numerical solution, general boundary conditions.

1. Introduction. The concept of fractional derivative, which is almost as old as its integer counterpart has only recently began to find applications in many fields of physics, finance or hydrology ([1], [2], [20], [3], [9]). Partial differential equations with fractional derivatives in time and/or space are used to create models in these fields, where classical derivatives do not suffice or give unsatisfying results. Fractional derivatives in space can be used to model anomalous diffusion, which is diffusion not in accord with the classical model of Brownian motion. When the second derivative in the classical diffusion equation is replaced by a derivative of fractional order α, where $0 < \alpha < 2$, this leads to faster than classical diffusion (also called superdiffusion). If $\alpha > 2$, the result is slower than classical diffusion, also called subdiffusion.

In this paper we are going to deal with a generalization of the classical advection-diffusion (dispersion) equation, where the second derivative will be replaced by a derivative of fractional order. We thus have the following partial differential equation on the domain $x \in \langle L, R\rangle, t \in \langle 0, T\rangle$, with the following initial and boundary conditions. We will call this equation the Fractional Advection-Dispersion Equation (FADE):

$$\begin{aligned} &\frac{\partial c}{\partial t}(x,t) = -v\,(x) * \frac{\partial c}{\partial x}(x,t) + \beta * d\,(x) * D_\alpha^-\,(c)\,(x,t) + (1-\beta) * d\,(x) * D_\alpha^+\,(c)\,(x,t)\,, \\ &c(x,0) = c_0(x), \\ &c(L,t) = d_L, \text{ or } \frac{\partial c}{\partial x}(L,t) = n_L, \quad c(R,t) = d_R, \text{ or } \frac{\partial c}{\partial x}(R,t) = n_R. \end{aligned} \tag{1.1}$$

The paper will be organized in the following way. First we will give a brief introduction into the theory of fractional derivatives. We will then present current numerical methods used to solve the 1-D FADE and we will demonstrate problems, which these methods have. In the following we will try to address these problems by developing an improved method based on the previous. In the last section we will generalize our method into 2-D and give some numerical examples.

[1]Department of Mathematics, Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University, Prague.

2. Fractional derivative. In this section we will give a brief introduction to the theory of fractional derivatives, for a more thorough account see [17], [14], [19] or [18].

2.1. Riemann-Liouville definition. The fractional derivative is defined with use of the Cauchy formula for n-times repeated integration on the domain $\mathbb{R}$.

$$f^{[n]}(x) = \int_{-\infty}^{x} \int_{-\infty}^{\sigma_1} ... \int_{-\infty}^{\sigma_{n-1}} f(\sigma_n)\, d\sigma_n ... d\sigma_2 d\sigma_1 = \frac{1}{(n-1)!} \int_{-\infty}^{x} (x-y)^{n-1} f(y)\, dy.$$

This formula can easily be generalized for fractional orders of integration. We obtain the following definition of the left-sided fractional integration operator of order α.

$$J_{\alpha}^{-}(f)(x) := \frac{1}{\Gamma(\alpha)} \int_{-\infty}^{x} (x-y)^{\alpha-1} f(y)\, dy.$$

The right-sided operator can be defined in a similar way.

$$J_{\alpha}^{+}(f)(x) := \frac{1}{\Gamma(\alpha)} \int_{x}^{\infty} (x-y)^{\alpha-1} f(y)\, dy.$$

To calculate the α-order fractional derivative of the function f, let us denote $n = [\alpha] + 1$. We calculate the $n - \alpha$ order fractional integral and we differentiate this n-times, which is a well-defined operation, because n is integer. This is the so-called Riemann-Liouville operator, which defines the left-sided fractional derivative.

$$D_{\alpha}^{-}(f)(x) = \frac{d^n}{dx^n} \left(J_{n-\alpha}^{-}(f)(x) \right).$$

The right-sided fractional derivative is defined in a similar manner.

$$D_{\alpha}^{+}(f)(x) = \frac{d^n}{dx^n} \left(J_{n-\alpha}^{+}(f)(x) \right).$$

2.2. Grűnwald formula. Another way to arrive at the fractional derivative is by generalizing finite difference schemes.

The first derivative can be approximated by the backward difference on a lattice of box size h.

$$f'(x) \approx \frac{f(x) - f(x-h)}{h}.$$

To obtain an approximation of the second derivative, we perform the backward difference twice.

$$f''(x) \approx \frac{f(x) - 2f(x-h) + f(x-2h)}{h^2}.$$

After applying the backward difference n-times we obtain the following approximation of the n-th derivative.

$$f^{(n)}(x) \approx \frac{1}{h^n} \sum_{k=0}^{n} (-1)^k \binom{n}{k} f(x-kh).$$

The previous relation can be generalized to non-integer orders by use of the gamma function as a generalization of the factorial. The limit $h \to 0$ gives us another definition of the left-sided fractional derivate. This is called the Grűnwald formula.

$$D_{\alpha}^{-}(f)(x) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{\infty} \frac{\Gamma(k-\alpha)}{\Gamma(-\alpha)\Gamma(k+1)} f(x-kh).$$

Similarly we obtain a right-sided derivative.

$$D_{\alpha}^{+}(f)(x) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{k=0}^{\infty} \frac{\Gamma(k-\alpha)}{\Gamma(-\alpha)\Gamma(k+1)} f(x+kh).$$

It is known that under fairly mild assumptions, the Riemann-Liouville and Grűnwald definitions are equivalent (see [19]).

3. Current methods for numerical solution of 1-D FADE. In recent years several authors have developed numerical methods of the solution of 1-D FADE. Most use a FDM approximation based on or similar to the Grűnwald formula ([15], [21], [5], [16], [24], [25], [4]). Roop et at. also developed a FEM method ([7], [8]). Meerschaert et al. developed a numerical method of second-order precision ([22]).

In this section we will briefly present the numerical method for the solution of equation (1.1), which was developed by Meerschaert and Tadjeran([10],[11]).

Consider a lattice in space with nodes $x_0 = L, x_1, x_2, ..., x_n = R$, where $\Delta x = \frac{R-L}{n}$ and $x_i = i\Delta x$ and in time $t^{(0)} = 0, t^{(1)}, ..., t^{(m)} = T$, where $\Delta t = T/m$ and $t^{(k)} = k\Delta t$.

3.1. Numerical approximation of fractional derivative. To approximate the fractional differentiation operator in FADE we will use the Grűnwald formula, let us denote.

$$g_k = \frac{\Gamma(k-\alpha)}{\Gamma(-\alpha)\Gamma(k+1).}$$

The Grűnwald formulas thus take the form

$$D_{\alpha}^{-}(f)(x_i) = \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{\infty} g_k f(x_{i-k}),$$

$$D_{\alpha}^{+}(f)(x_i) = \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{\infty} g_k f(x_{i+k}).$$

Because we are working on a finite domain $\langle L, R \rangle$, we will approximate the fractional derivatives in equation (1.1) in the following manner.

$$D_{\alpha}^{-}(f)(x_i) \approx \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{i} g_k f(x_{i-k}),$$

$$D_{\alpha}^{+}(f)(x_i) \approx \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{n-i} g_k f(x_{i+k}).$$

Meerschaert and Tadjeran proved ([10],[11]) that to get a unconditionally stable scheme it is better to use the a modified Grűnwald formula which is shifted by 1.

The approximation we are going to use is the following.

$$D_{\alpha}^{-}(f)(x_i) \approx \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{i+1} g_k f(x_{i+1-k}), \tag{3.1}$$
$$D_{\alpha}^{+}(f)(x_i) \approx \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{n-i+1} g_k f(x_{i-1+k}).$$

3.2. Numerical scheme for FADE. The time derivative is discretized by means of a backward difference.

$$\frac{\partial c}{\partial t}(x_i, t^{(n+1)}) \approx \frac{c_i^{(n+1)} - c_i^{(n)}}{\Delta t}.$$

An explicit scheme using the previous approximations of the derivatives would be computationally more feasible, but unfortunately such a scheme is not stable. Therefore we will use the following implicit scheme, which is stable (see [10],[11]).

$$\frac{c_i^{(n_t+1)} - c_i^{(n_t)}}{\Delta t} = -v_i \frac{c_{i+1}^{(n_t+1)} - c_i^{(n_t+1)}}{\Delta x} + \beta d_i \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{i+1} g_k c_{i+1-k}^{(n_t+1)}$$
$$+ (1-\beta) d_i \frac{1}{\Delta x^{\alpha}} \sum_{k=0}^{n-i+1} g_k c_{i-1+k}^{(n_t+1)}, \quad n_t = 0, \ldots, m-1, \quad i = 1, \ldots, n-1. \tag{3.2}$$

Let us denote

$$E_i = v_i \Delta t / \Delta x, \quad B_i = d_i \Delta t / \Delta x^{\alpha}.$$

We then get the following system of linear equations.

$$-(\beta g_0 + (1-\beta) g_2) B_i c_{i+1}^{(n_t+1)} + (1 + E_i - g_1 B_i) c_i^{(n_t+1)}$$
$$- (E_i + \beta g_2 B_i + (1-\beta) g_0 B_i) c_{i-1}^{(n_t+1)} - \beta B_i \sum_{k=3}^{i+1} g_k c_{i-k+1}^{(n_t+1)} - (1-\beta) B_i \sum_{k=3}^{n-i+1} g_k c_{i-1+k}^{(n_t+1)}$$
$$= c_i^{(n)} + \Delta t f_i^{(n_t+1)}, i = 1, \cdots, n-1$$

We also have equations that come from the boundary conditions (depending on whether we have Dirichlet or Neumann conditions).

$$c_0 = d_L \text{ or } \frac{c_1 - c_0}{\Delta x} = n_L, \quad c_n = d_R \text{ or } \frac{c_n - c_{n-1}}{\Delta x} = n_R.$$

To move forward one time-step we have to solve a system of linear equations, where the values of c in the previous time are on the right-hand side. If the coefficients v_i and d_i are constant in time then the matrix of the linear system will remain the same. We thus calculate its inverse in the beginning and then only change right-hand sides, which is an advantage in computations.

3.3. Problems of method.

3.3.1. Problem 1: Boundary conditions. The method described in the previous section works well for problems, where we expect, that the function c outside of the boundaries (i.e. the intervals $(-\infty, L) \cup (R, \infty)$) is equal to 0. In other words, the support of the function c is all inside the interval (L, R). This is only the case when there are Dirichlet boundary conditions on both sides with $d_L = d_R = 0$. When we have Neumann conditions on the boundaries and the values of c near the boundaries are close to 0 then this is a good approximation. But for all other cases (i.e. non-zero Dirichlet boundaries or Neumann boundaries, where c is non-negligibly higher than 0) the method leads to fundamentally wrong results. This is illustrated in Figure 3.1, which shows the evolution of two different numerical solutions to equation 1.1, when using the above described method of Meerschaert and Tadjeran. The blue line is the solution with a delta function initial condition and zero Dirichlet boundary conditions. The green line corresponds to the first problem shifted up by 0.5 (i.e. $c(x,0) = 0.5 + \delta(x), d_L = d_R = 0.5$). It would be expected, that the two solutions will behave in an identical manner only shifted by 0.5 for all times t. But we can see that in the second case its behavior is different at the boundaries (a very fast decrease of c) and soon this artifact changes the look of the whole function.

The problem is caused by the incorrect interpretation of boundary conditions. For simplicity, let us consider in the following only the left fractional derivative (i.e. $\beta = 1$), all of the also following applies for the right fractional derivative in a similar manner. When evaluating the left fractional derivative at a point near the left boundary only the information in the few points between the point itself and the boundary is used. Whereas when the left fractional derivative is evaluated at a point further away from the left boundary, information about the function in more points is used. The more points we use for the evaluation of the Grűnwald formula (i.e. the less members in the Grűnwald sum we cut off) the better approximation of the real fractional derivative we get. But the number of points used depends on the distance from the boundary, so for the points close to the left boundary, the left fractional derivative is approximated very badly.

3.3.2. Proposed solution to problem 1. When numerically solving partial differential equations (i.e. non-fractional) the number of points which are used to approximate the derivatives in the equation is always constant in space. Also, the number of points on the edge, which have to be prescribed by a boundary condition is equal to the distance of the furthest point in the derivative approximation scheme. For example when solving the ordinary diffusion equation by use of the scheme $f_i'' \approx \frac{f_{i-1} - 2f_i + f_{i+1}}{\Delta x^2}$, only one point on the each boundary has to be prescribed, f_0 and f_n. But if we used a five-point symmetric scheme for the approximation of the second derivative, 2 points on each boundary would have to be prescribed, f_{-1}, f_0 and f_n, f_{n+1}.

But when solving the FADE by use of the method described above (scheme (3.2)) we use only one point on each boundary although the number of points in the approximation of the derivative ranges from 1 to n. But scheme (3.2) can also be interpreted in a different way. We could use an n-point approximation of the left fractional derivative in each point,

$$D_\alpha^-(f)(x_i) = \frac{1}{\Delta x^\alpha} \sum_{k=0}^{n} g_k f(x_{i-k}), i = 1, ..., n-1,$$

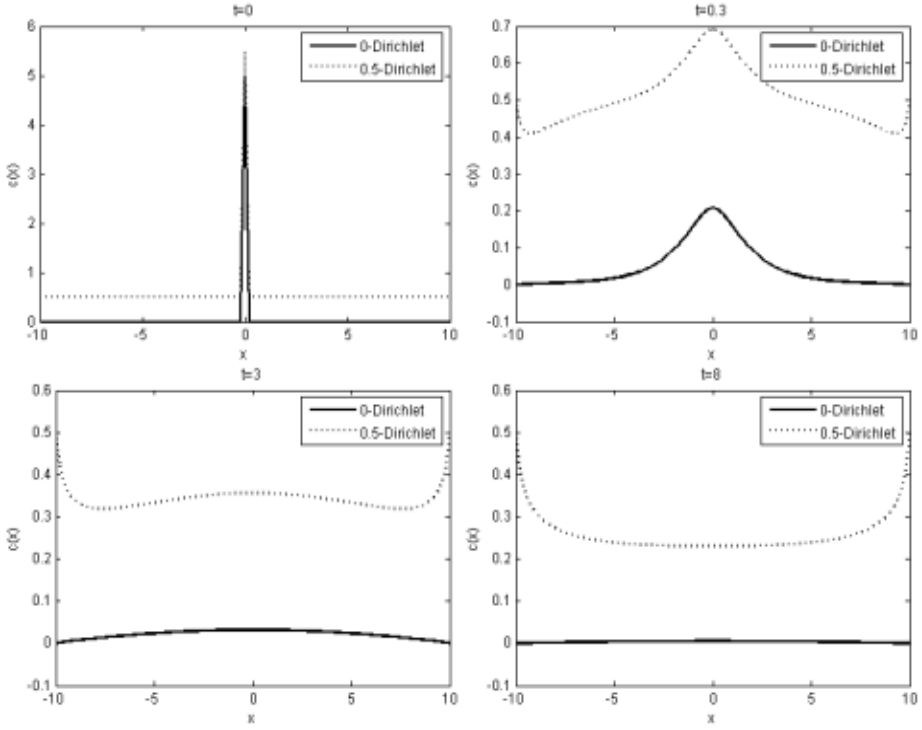

FIG. 3.1. *Problem with non-zero Dirichlet boundary conditions. Artifacts are created at the boundaries.*

and prescribe a boundary condition in not just one point on the boundary but in n points $f_0, f_{-1}, f_{-2}, ..., f_{-(n-1)}$. When we put

$$c_{-1}, c_{-2}, ..., c_{-(n-1)} = 0, c_0 = c_L,$$

we get a scheme, which is exactly the same as scheme (3.2). Therefore scheme (3.2) can be interpreted as an n-point scheme with zero boundary conditions everywhere but c_0. This explains the artifacts which arise near the boundary in Figure 3.1. When $c_0 = 0.5$ to solution is forced to have this value on the boundary, but because $c_{-1}, ..., c_{-(n-1)} = 0$ there is a very steep downward gradient near the boundary, which causes the sharp decrease which eventually deforms the whole solution. Our answer to this problem is to prescribe all the boundary conditions in a consistent manner

$$c_{-1}, ..., c_{-(n-1)} = c_0 = c_L.$$

All of the previous applies similarly for the right-sided fractional derivative and the right boundary conditions

$$c_{n+1}, ..., c_{2n-1} = c_n = c_R.$$

Our proposed method will we covered in more detail in section 4.

3.3.3. Problem 2: Derivative of constant function. When the first problem is removed as indicated above (more in section 4) another problem demonstrates itself. The problem can be best shown on a simple FADE with Dirichlet boundary

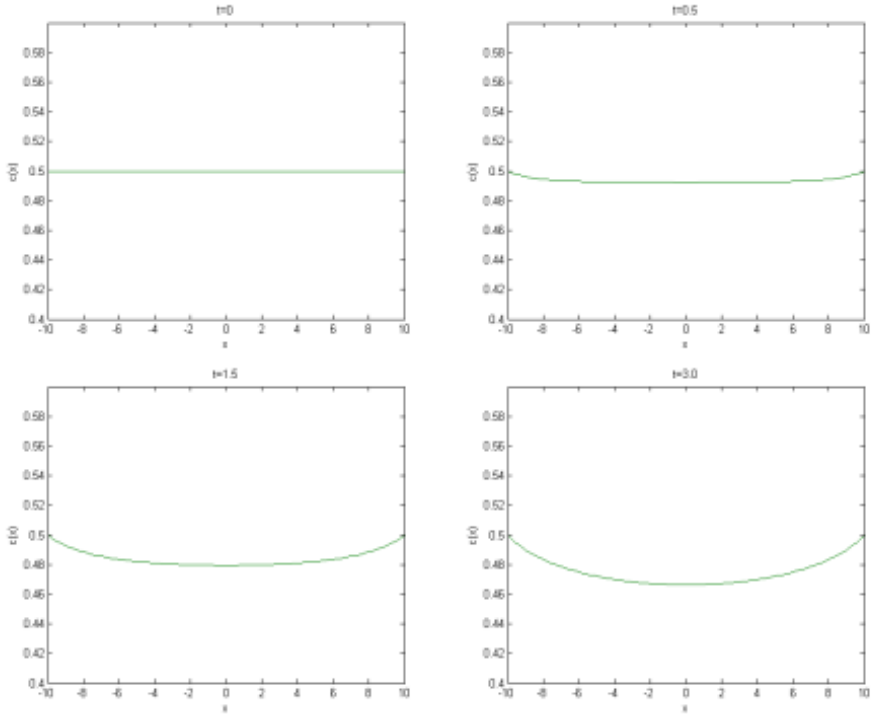

FIG. 3.2. *Illustration of problem 2. Derivative of constant function is not zero.*

conditions on both edges and a constant initial condition. The derivative (even of fractional order) of a constant is always 0 so the solution of the FADE in this case will not change and will remain constantly identical to the initial condition. But the numerical results are different as is shown on Figure 3.2. The reason behind this again lies in way the fractional derivative is approximated. In classical partial differential equations when approximating the derivative by means of a finite difference scheme, the coefficients always add up to zero. This causes the derivative of a constant function to be zero. But in our case, when we approximate the fractional derivative by means of the Grűnwald formula, we always have to cut some members of the sum off because there is a infinite number of them. And only the whole infinite sum of coefficients adds up to zero. When we cut the sum at some stage this causes the sum of what's left to be less than one. Therefore the fractional derivative of a constant calculated using this approximation is slightly negative, which causes the global decrease of the whole solution in time as shown on Figure 3.2.

3.3.4. Proposed solution to problem 2. The problem can be solved by increasing the coefficient at the furthest point by the difference needed to get a zero sum of the coefficients. This is equivalent to extending the function to all of $\mathbb{R}$ and postulating that $\forall i \in \mathbb{N}_0, x_{-n} = x_0$. That is, constantly extending the function from the finite interval onto $\mathbb{R}$. Naturally, this extension is not always ideal (it is so for Dirichlet boundary conditions), but it is certainly better than extending the function with zeros.

4. Proposed numerical method for solution of 1-D FADE. In this section we will present our numerical method for the solution of the 1-D FADE, which is based on the method described in the previous chapter but contains improvements, which deal with the problems described above.

Consider a lattice in space with nodes $x_0 = L, x_1, x_2, ..., x_n = R$, where $\Delta x = \frac{R-L}{n}$ and $x_i = L + i\Delta x$ and in time $t^{(0)} = 0, t^{(1)}, ..., t^{(m)} = T$, where $\Delta t = T/m$ and $t^{(k)} = k\Delta t$.

Also let us consider a function f defined on $\langle L, R\rangle$. Then we can define an extension of the function f onto the whole real axis in the following manner

$$\begin{aligned}\tilde{f}(x) &= f(x), x \in \langle L, R\rangle,\\ \tilde{f}(x) &= f(L), x \in \langle -\infty, L\rangle,\\ \tilde{f}(x) &= f(R), x \in \langle R, \infty\rangle.\end{aligned}$$

The function $\tilde{f}$ is simply the extension of f, under the assumption that outside the interval $\langle L, R\rangle$ the function is constant and it's value is equal to the value at the boundary.

4.1. Approximation of fractional derivative. In our method we will again start from the Grűnwald formula. This time we will not cut off the members of the Grűnwald sum which lie out of the interval $\langle L, R\rangle$, but we will use the complete Grűnwald sum of the extended function $\tilde{f}$. Thus our approximation of the left fractional derivative will have the following form.

$$D_\alpha^-(f)(x_i) = \frac{1}{\Delta x^\alpha}\sum_{k=0}^{\infty} g_k \tilde{f}(x_{i-k}).$$

This can be rewritten by use of the definition of $\tilde{f}$.

$$\begin{aligned}D_\alpha^-(f)(x_i) &= \frac{1}{\Delta x^\alpha}\left(\sum_{k=0}^{i} g_k f(x_{i-k}) + \sum_{k=i+1}^{\infty} g_k f(x_0)\right)\\ &= \frac{1}{\Delta x^\alpha}\left(\sum_{k=0}^{i-1} g_k f(x_{i-k}) + g_i' f(x_0)\right),\end{aligned}$$

where $g_i' = g_i + \sum_{k=i+1}^{\infty} g_k = -\sum_{k=0}^{i-1} g_k$. The previous equality is implied by the fact, that $\sum_{k=0}^{\infty} g_k = 0$. Therefore

$$\begin{aligned}D_\alpha^-(f)(x_i) &= \frac{1}{\Delta x^\alpha}\left(\sum_{k=0}^{i-1} g_k f(x_{i-k}) - \left(\sum_{k=0}^{i-1} g_k\right) f(x_0)\right)\\ &= \frac{1}{\Delta x^\alpha}\sum_{k=0}^{i-1} g_k\left(f(x_{i-k}) - f(x_0)\right).\end{aligned}$$

Similarly for the right-sided derivative

$$D_\alpha^+(f)(x_i) = \frac{1}{\Delta x^\alpha}\sum_{k=0}^{n-i-1} g_k\left(f(x_{i+k}) - f(x_n).\right)$$

To improve the stability of the scheme we are going to apply the same shift by one as in the method described in section 3.1. We thus obtain the following approximation of the fractional derivatives.

$$D_\alpha^-(f)(x_i) \approx \frac{1}{\Delta x^\alpha}\sum_{k=0}^{i} g_k \left(f(x_{i+1-k}) - f(x_0)\right), \tag{4.1}$$

$$D_\alpha^+(f)(x_i) \approx \frac{1}{\Delta x^\alpha}\sum_{k=0}^{n-i} g_k \left(f(x_{i-1+k}) - f(x_n)\right).$$

4.2. Numerical scheme. The time derivative is discretized by means of a backward difference.

$$\frac{\partial c}{\partial t}(x_i, t^{(n+1)}) \approx \frac{c_i^{(n+1)} - c_i^{(n)}}{\Delta t}.$$

For reasons of stability we propose the following implicit scheme for the solution of equation (1.1) by use of the approximations (4.1).

$$\frac{c_i^{(n_t+1)} - c_i^{(n_t)}}{\Delta t} = -v_i \frac{c_{i+1}^{(n_t+1)} - c_i^{(n_t+1)}}{\Delta x} + \beta d_i \frac{1}{\Delta x^\alpha}\sum_{k=0}^{i} g_k \left(c_{i+1-k}^{(n_t+1)} - c_0^{(n_t+1)}\right), \tag{4.2}$$

$$+ (1-\beta)\, d_i \frac{1}{\Delta x^\alpha}\sum_{k=0}^{n-i} g_k \left(c_{i-1+k}^{(n_t+1)} - c_n^{(n_t+1)}\right), \tag{4.3}$$

$$n_t = 0, ..., m-1,\ i = 1, n-1.$$

Let us denote

$$E_i = v_i \Delta t/\Delta x,\ B_i = d_i \Delta t/\Delta x^\alpha.$$

We then get the following system of linear equations.

$$-(\beta g_0 + (1-\beta)g_2) B_i c_{i+1}^{(n_t+1)} + (1 + E_i - g_1 B_i)\, c_i^{(n_t+1)}$$

$$-\left(E_i + \beta g_2 B_i + (1-\beta) g_0 B_i\right) c_{i-1}^{(n_t+1)} - \beta B_i \sum_{k=3}^{i} g_k c_{i-k+1}^{(n_t+1)} - (1-\beta)\, B_i \sum_{k=3}^{n-i} g_k c_{i-1+k}^{(n_t+1)}$$

$$+\beta B_i \sum_{k=0}^{i} g_k c_0^{(n_t+1)} + (1-\beta)\, B_i \sum_{k=0}^{n-i} g_k c_n^{(n_t+1)} = c_i^{(n_t)} + \Delta t f_i^{(n_t+1)},\ i = 1, \ldots, n-1.$$

We also have equations that come from the boundary conditions (depending on whether we have Dirichlet or Neumann conditions).

$$c_0 = d_L \text{ or } \frac{c_1 - c_0}{\Delta x} = n_L,\ c_n = d_R \text{ or } \frac{c_n - c_{n-1}}{\Delta x} = n_R.$$

The system has similar properties as the system in the method of Meerschaert and Tadjeran. To move forward one time-step we have to solve a system of linear equations, where the values of c in the previous time are on the right-hand side. If the coefficients v_i and d_i are constant in time then the matrix of the linear system will remain the same. We thus calculate its inverse only once at the beginning and then only change right-hand sides, which is an advantage for computations.

5. Two-dimensional FADE. The numerical solution of the 2-D FADE has also been considered by Meerschaert and Tadjeran ([12], [23]), but only in the simplest form of diffussion along the principal axes x and y (i.e. 4 directions in our notation). Roop et al. developed a multidimensional FEM method ([6]). To our knowledge, these are the only published results on the multidimensional FADE. In the following, we will develop a numerical method for the solution of the 2-D FADE in it's general multi-directional form.

5.1. Generalization of 1-D FADE. The generalization of the 1-D FADE to 2-D is not as straightforward as in the case of the ordinary ADE. This is caused by the fact that any ordinary diffusion in 2-D can by expressed as a combination of diffusion along the axes x and y (or more generally a rotated coordinate system, this would mean the diffusion coefficient d would be a 2by2 tensor). Thus the ordinary 2-D ADE has the following form

$$\frac{\partial c}{\partial t}(x,y,t) = -\mathbf{v}\cdot\nabla c\left(x,y,t\right) + d * \triangle c\left(x,y,t\right), \tag{5.1}$$

where $\mathbf{v} = \left(v_x\left(x,y\right), v_y\left(x,y\right)\right)$, $\nabla c = \left(\frac{\partial c}{\partial x}, \frac{\partial c}{\partial y}\right)$ and $\triangle c = \frac{\partial^2 c}{\partial x^2} + \frac{\partial^2 c}{\partial y^2}$.

To obtain the 2-D FADE we have to define a fractional analogy of the Laplace operator $\triangle$. Because of the non-local character of the fractional derivatives it is not sufficient to use only the derivatives in the direction of the axes x and y. A fully general 2-D FADE contains fractional derivatives in all directions. The definition of the 2-D fractional derivative operator is the following (see [13]).

$$\nabla_M^\alpha f\left(\mathbf{x},t\right) = \frac{1}{\Gamma\left(-\alpha\right)} \int_{\|\theta\|=1} \int_0^\infty r^{-\alpha-1} f\left(\mathbf{x} - r\theta, t\right) dr \cdot m\left(\theta\right) d\theta. \tag{5.2}$$

Here $m\left(\theta\right)$ defines a probability measure on the unit circle, this measure represents the weights of all the different directional derivatives, which are given by the unit vector θ.

Now we will formulate the 2-D FADE on a rectangular domain $x \in \Gamma = \langle L, R\rangle \times \langle D, U\rangle$ and $t \in \langle 0, T\rangle$. Let $\partial\Gamma = \partial\Gamma_d \cup \partial\Gamma_n$.

$$\begin{aligned}
\frac{\partial c}{\partial t}(x,y,t) &= -\mathbf{v}\cdot\nabla c\left(x,y,t\right) + d * \nabla_M^\alpha f\left(\mathbf{x},t\right), \\
c(x,y,0) &= c_0(x,y), \\
c(x,y,t) &= d_{\partial\Gamma_1}\left(x,y\right) \text{ for } (x,y) \in \partial\Gamma_1, \\
\frac{\partial c}{\partial x}(x,y,t) &= n_{\partial\Gamma_2}\left(x,y\right) \text{ for } (x,y) \in \partial\Gamma_2.
\end{aligned} \tag{5.3}$$

5.2. Numerical approximation of 2-D fractional derivative operator. Consider a lattice on Γ defined by the discretization of the axes x and y in the following manner:

$$x_0 = L, x_1, x_2, ..., x_{n_x} = R, \text{ where } \Delta x = \frac{R-L}{n_x} \text{ and } x_i = L + i\Delta x,$$

and

$$y_0 = D, y_1, y_2, ..., y_{n_y} = U, \text{ where } \Delta y = \frac{U-D}{n_y} \text{ and } y_j = D + j\Delta y,$$

and in time

$$t^{(0)} = 0, t^{(1)}, ..., t^{(m)} = T, \text{ where } \Delta t = T/m \text{ and } t^{(k)} = k\Delta t.$$

For the approximation of the operator (5.2) we will again use the modified Grűnwald formula (4.1). For each direction $\theta = (\theta_1, \theta_2)$ there exists a unique angle $\varphi \in \langle 0, 2\pi)$ such that $\theta = (\cos\varphi, \sin\varphi)$. First we will discretize the unit circle (i.e. the possible directions) by diving it into n_φ parts, which is equivalent to diving the interval $\langle 0, 2\pi)$ into n_φ parts:

$$\varphi_0 = 0, \varphi_1 = \frac{2\pi}{n_\varphi}, \varphi_2 = 2\frac{2\pi}{n_\varphi}, ..., \varphi_{n_\varphi - 1} = 2\pi - \frac{2\pi}{n_\varphi}.$$

This measure $m(\theta)$ defines a measure (it's discretized version) m_d on $\varphi_0, ..., \varphi_{n_\varphi - 1}$ such that $\sum_{i=0}^{n_\varphi - 1} m_d(\varphi_i) = 1$.

For simplicity, let us presume that $\Delta x = \Delta y$. We can approximate the 2-D fractional differential operator by using the modified and improved Grűnwald formula (4.1) in each direction φ_k. We obtain

$$\nabla_M^\alpha f(x_i, y_j) \approx \sum_{s=0}^{n_\varphi - 1} m_d(\varphi_s) \frac{1}{\Delta x^\alpha} \sum_{k=0}^{r_s(i,j)} g_k \left(f\left(\mathbf{x}_{k-1}^s(i,j)\right) - f\left(\mathbf{b}_s(i,j)\right)\right). \quad (5.4)$$

Here $\mathbf{x}_{k-1}^s(i,j)$ is the closest point in the grid to the point $\tilde{\mathbf{x}}_{k-1}^s(i,j)$, which is the point with distance $(k-1) * \Delta x$ from the point (x_i, y_j) in the direction φ_s (see Fig. 5.1) .

$$\begin{aligned}\tilde{\mathbf{x}}_{k-1}^s(i,j) &= (x_i + (k-1)\Delta x \cos\varphi_s, y_j + (k-1)\Delta x \sin\varphi_s) = \\ &= \Delta x\,(i + (k-1)\cos\varphi_s, j + (k-1)\sin\varphi_s).\end{aligned}$$

Therefore

$$\begin{aligned}\mathbf{x}_{k-1}^s(i,j) &= \Delta x\,(round\,(i + (k-1)\cos\varphi_s), round\,(j + (k-1)\sin\varphi_s)) = \\ &= \left(x_{round(i+(k-1)\cos\varphi_s)}, y_{round(j+(k-1)\sin\varphi_s)}\right).\end{aligned}$$

The point $\mathbf{b}_s(i,j)$ is the point $\mathbf{x}_{k-1}^s(i,j)$, which lies on the boundary of the domain, $r_s(i,j) = k$ (i.e. this is the distance from the point (x_i, y_j) to the boundary). See Figure 5.1.

5.3. Numerical scheme. The time derivative is discretized by means of a backward difference.

$$\frac{\partial c}{\partial t}(x_i, y_j, t^{(n+1)}) \approx \frac{c_{i,j}^{(n+1)} - c_{i,j}^{(n)}}{\Delta t}.$$

For reasons of stability we propose the following implicit scheme for the solution of equation (5.3) by use of the approximations (5.4).

$$\begin{aligned}&\frac{c_{i,j}^{(n+1)} - c_{i,j}^{(n)}}{\Delta t} = -v_x(i,j)\frac{c_{i+1,j}^{(n+1)} - c_{i,j}^{(n+1)}}{\Delta x} - v_y(i,j)\frac{c_{i,j+1}^{(n+1)} - c_{i,j}^{(n+1)}}{\Delta y} \\ &+ \sum_{s=0}^{n_\varphi - 1} m(\theta_s)\frac{1}{\Delta x^\alpha}\sum_{k=0}^{r_s(i,j)} g_k\left(c\left(\mathbf{x}_{k-1}^s(i,j), t^{(n+1)}\right) - c\left(\mathbf{b}_s(i,j), t^{(n+1)}\right)\right), \\ &n = 0, ..., m-1, \quad i = 1, n_x - 1, \quad j = 1, ..., n_y - 1.\end{aligned}$$

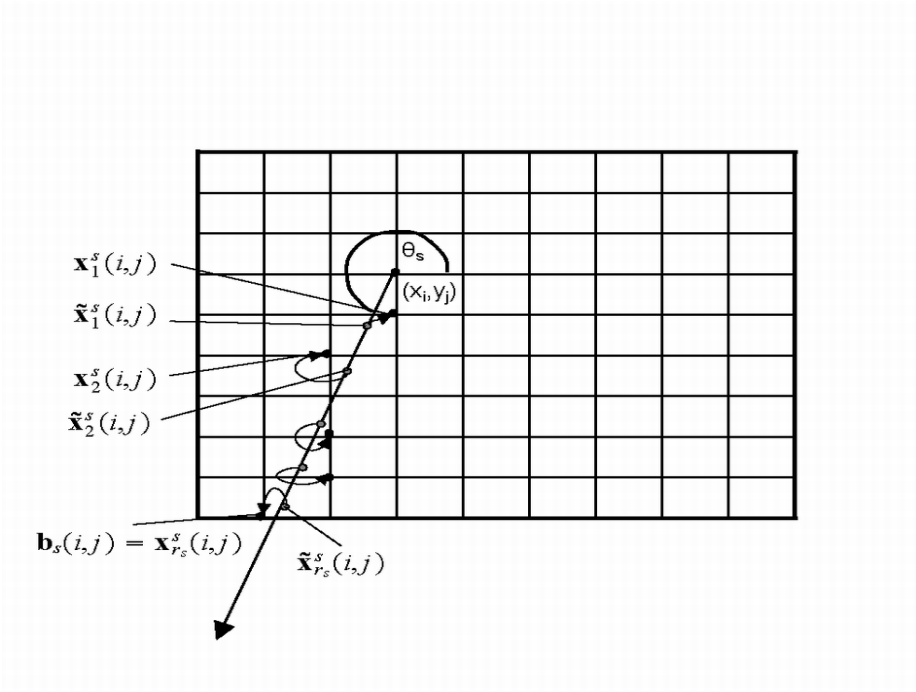

FIG. 5.1. *Illustration of our approximation of the fractional differential operator in 2-D*

This scheme again leads to a system of linear equations, which must be solved in each time step. The problem with this system is, that it's matrix is dense, because the value the fractional derivative of c in each point depends on almost all the points in the grid (depending on the number of directions n_φ). Thus there are few zeros in the matrix. This is a big problem because even at a relatively small number of grid points the matrix is very large and it's inversion is very slow or impossible. The examples we are going to show in the following part of the paper are thus solutions of the 2-D FADE only on grid with a small number of points (cca $(50-100)^2$).

5.4. Numerical examples of our method : example 1 - delta function initial condition. The first example is going to be the simplest case possible, the 2-D FADE with a delta function initial condition. We are going to solve (5.3) with $\alpha = 1.5$ on $\Gamma = \langle 0, 50\rangle \times \langle 0, 50\rangle$ with a grid of $n_x = 50$ points on the x axis and $n_y = 50$ points on the y axis, thus $\Delta x = \Delta y = 1$. We use a timestep of $\Delta t = 1$. The diffusion coefficient $d = 1$, there is no advection, i.e. $v_x = v_y = 0$. The boundary conditions used are Neumann boundary conditions on the whole boundary, i.e. $\frac{\partial c}{\partial x}(x, y, t) = 0$ on $\partial\Gamma$. In the first case (left part of figures) we will use only 4 directions (i.e. $n_\varphi = 4$), in the second case (right part of figures), we will use 17 directions. We will consider in each case a uniform measure $m_d(\varphi_i) = \frac{1}{n_\varphi}$. The initial condition in this case is the discrete approximation of a delta function, i.e. $c\left(x_{n_x/2}, y_{n_y/2}\right) = \frac{1}{\Delta x \Delta y}$ and $c = 0$ everywhere else. The results are shown on Figures 5.2, 5.3, 5.4. The bright points correspond to higher values of c. It is visible that in the case of the 4 principal directions the diffusion is asymmetrical and much quicker along the axes. In the case of 17 directions the result is very close to a symmetrical shape.

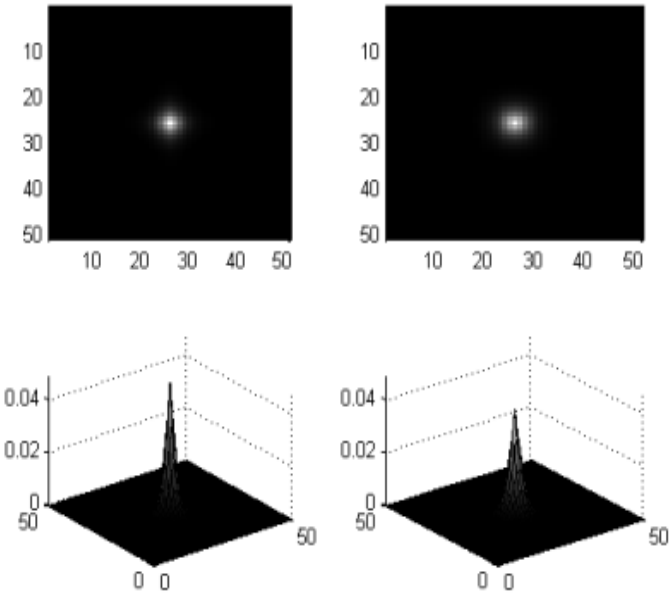

FIG. 5.2. *Example 1 - t=5s, left 4 directions, right 17 directions*

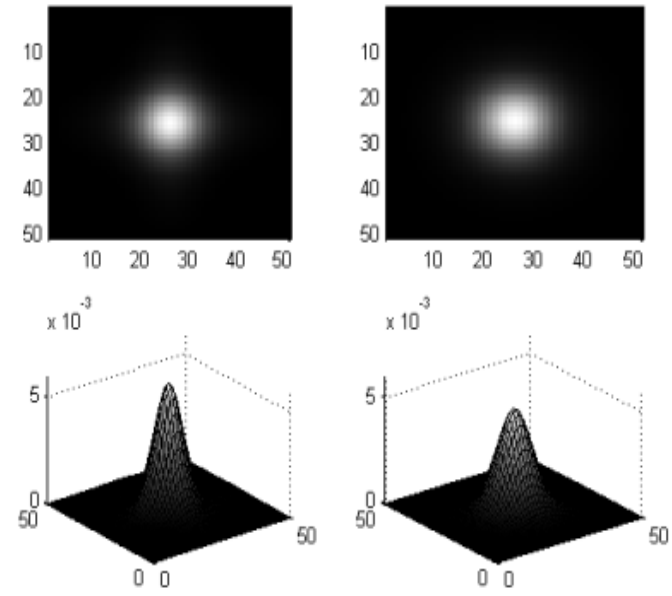

FIG. 5.3. *Example 1 - t=20s, left 4 directions, right 17 directions*

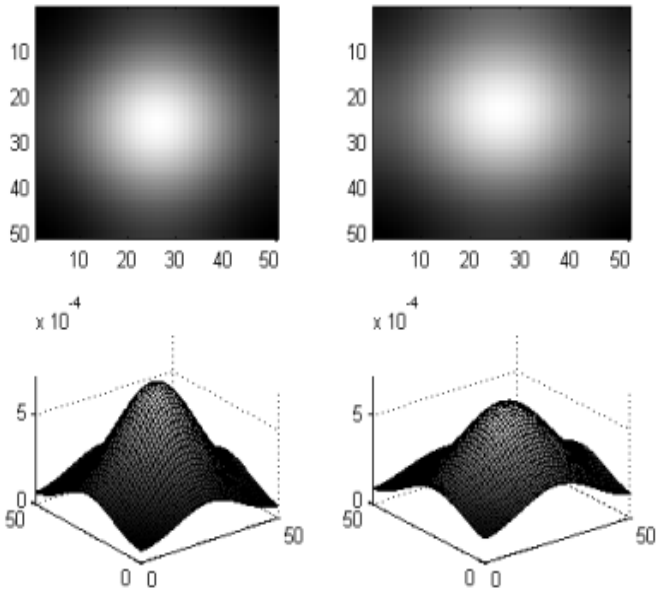

FIG. 5.4. *Example 1 - t=100s, left 4 directions, right 17 directions*

5.5. Example 2 - Flow through pipe. The second example is going to be the simple approximation of contaminant flow in a pipe We are going to solve (5.3) with $\alpha = 1.5$ on $\Gamma = \langle 0, 50\rangle \times \langle 0, 20\rangle$ with a grid of $n_x = 50$ points on the x axis and $n_y = 20$ points on the y axis, thus $\Delta x = \Delta y = 1$. We use a timestep of $\Delta t = 1$. The diffusion coefficient is uniformly equal to 1, i.e. $d = 1$ and there is a advective flow from left to right, i.e. $v_x = 0.2, v_y = 0$. The boundary conditions used are Neumann boundary conditions on the right side, i.e. $\frac{\partial c}{\partial x}(R, y, t) = 0$ The source of the contaminant is the left side which has a non-zero Dirichlet boundary condition, i.e. $c(L, y, t) = 1$. The upper and lower boundaries have zero Dirichlet boundary conditions. 7 directions will be used, i.e. $n_\varphi = 7$. We will consider in each case a uniform measure $m_d(\varphi_i) = \frac{1}{n_\varphi}$. The initial condition in this case is $c(x, y, 0) = 0$. The result is shown on Figure 5.5.

6. Acknowledgement. The work was partly supported by the project No. MSM 6840770010 "Applied Mathematics in Technical and Physical Sciences" of the Ministry of Education, Youth and Sports of the Czech Republic.

REFERENCES

[1] D.A. BENSON, *The Fractional Advection-Dispersion Equation - Development and Application* , disertation thesis, University of Nevada, Reno (1998).

[2] D.A. BENSON, S.W. WHEATCRAFT, AND M.M. MEERSCHAERT, *Application of a Fractional Advection-Dispersion Equation*, Water Resources Reasearch, **36**, no. 6 (2000), 1403–1412.

[3] D.A. BENSON, S.W. WHEATCRAFT, AND M.M. MEERSCHAERT, *The Fractional-Order Governing Equation of Lévy Motion*, Water Resources Reasearch, **36**, no. 6 (2000), 1413–1423.

[4] M. CIESIELSKI, AND J. LESZCZYNSKI, *Numerical Treatment of an Initial-Boundary Value Problem for Fractional Patrial Differential Equations*, Signal Processing **86** (2006), 2619–2631.

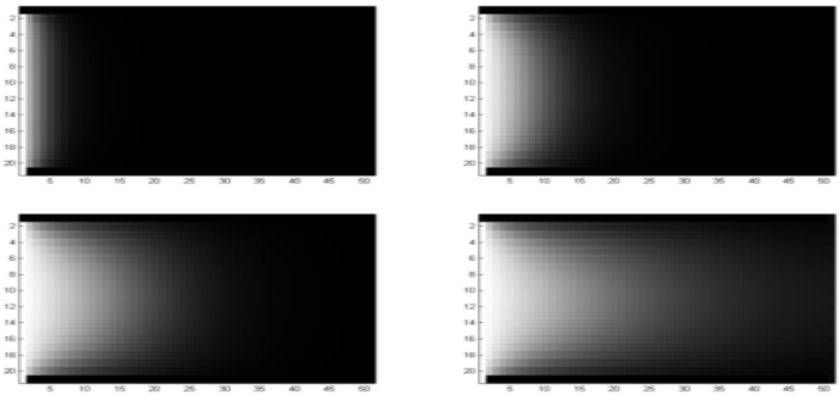

FIG. 5.5. *Example 2 - from left to right, top to bottom: t=5s,t=20s,t=50s,t=100s*

[5] Z. DENG, V.P. SINGH, AND L. BENGTSSON, *Numerical Solution of Fractional Advection-Dispersion Equation*, Journal of Hydraulic Engineering (2004), 422–431.

[6] V.J. ERVIN, AND J.P. ROOP, *Variational Solution of Fractional Advection Dispersion Equations on Bounded Domains R^d*, 2005, preprint.

[7] V.J. ERVIN, AND J.P. ROOP, *Variational Solution to the Fractional Advection Dispersion Equation*, Numer. Meth. P.D.E, to appear 2006, preprint.

[8] G.J. FIX, AND J.P. ROOP, *Least Squares Finite Element Solution of a Fractional Order Two-Point Boundary Value Problem*, Comput. Math. Appl. **48** (2004), 1017–1033.

[9] R.F. HOLUB, M. BENEŠ, AND HONEYMAN, B.D., *Interfacial Phenomena in Fluid Dynamics: Linking Atomistic and Macroscopic Properties: Can They Explain the Transport Anomalies?* in Beneš, M., Mikyška, J. and Oberhuber, T., editors, Proceedings of the Czech Japanese Seminar in Applied Mathematics. Prague: Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague, 2005. ISBN 80-01-03181-0, pages 57–62.

[10] M.M. MEERSCHAERT, AND CH. TADJERAN, *Finite Difference Approximations for Fractional Advection-Dispersion Flow Equations*, Journal of Computational and Applied Mathematics **172** (2004), 65–77.

[11] M.M. MEERSCHAERT, AND CH. TADJERAN, *Finite Difference Approximations for Two-Sided Space-Fractional Partial Differential Equations*, Applied Numerical Mathematics **56** (2006), 80–90.

[12] M.M. MEERSCHAERT, H. SCHEFFLER, AND CH. TADJERAN, *Finite Difference Methods for Two-Dimensional Fractional Dispersion Equation*, Journal of Computational Physics **211** (2006), 249–261.

[13] M.M. MEERSCHAERT, D.A. BENSON, AND B. BAUMER, *Multidimensional Advection and Fractional Dispersion*, Physical Review E, **59**, no. 5 (1999), 5026–5028.

[14] K. MILLER, AND B. ROSS, *An Introduction to the Fractional Calculus and Fractional Differential Equations*, Wiley, New York (1993).

[15] F. LIU, V. ANH, AND I.TURNER, *Numerical Solution of the Space Fractional Fokker-Planck Equation*, Journal of Computational and Applied Mathematics **166** (2004), 209–219.

[16] V.E. LYNCH, B.A. CARRERAS, D. DEL-CASTILLO-NEGRETO, K.M. FERREIRA-MEJIAS, AND H.R. HICKS, *Numerical Methods for the Solution of Partial Differential Equations of Fractional Order*, Journal of Computational Physics **192** (2003), 406–421.

[17] K.B. OLDHAM, AND J. SPANIER, *The Fractional Calculus*, Academic Press, New York (1974).

[18] I. POLUBNY, *Fractional Differential Equations*, Academic Press, New York (1999).

[19] S. SAMKO, A KILBAS, AND O. MIRICHEV, *Fractional Integrals and Derivatives: Theory and Applications*, Gordon and Breach, London (1993).

[20] R. SCHUMER, D.A. BENSON, S.W. WHEATCRAFT, AND M.M. MEERSCHAERT, *Eulerian Derivation of Fractional Advection-Dispersion Equation*, Journal of Contaminant Hydrology **48** (2001), 69–88.

[21] S. SHEN, AND F. LIU, *Error Analysis of an Explicit Finite Difference Approximation for the Space Fractional Diffusion Equation with Insulated Ends*, preprint, 2004.

[22] Ch. Tadjeran, M.M. Meerschaert, and H. Scheffler, *A Second-Order Accurate Numerical Approximation for the Fractional Diffusion Equation*, Journal of Computational Physics **213** (2006), 205–213.
[23] Ch. Tadjeran, and M.M. Meerschaert, *A Second Order Accurate Numerical Method for the Two-Dimensional Fractional Diffusion Equation*, preprint 2006.
[24] S.B. Yuste, and L. Acedo, *An Explicit Finite Difference Method and a New Von Neumann-Type Stability Analysis for Fractional Diffusion Equations*, SIAM J. Numer. Anal. **42**, No. 5 (2005), 1862–1874.
[25] S.B. Yuste, *Weighted Average Finite Difference Methods for Fractional Diffusion Equations*, Journal of Computational Physics **216** (2006), 264–274.