Lecture 2 Numerical methods for fractional diffusion-reaction equations

Dr Qianqian Yang

AMSI Winter School 2019

In the first lecture we learned how the fractional diffusion equation can be derived as the continuum limit of a random walk with "long jumps". That is, a random walk where the jump size is drawn from a probability distribution with infinite variance. Actually, this is just one kind of fractional diffusion equation. Other variants are possible, and correspond to different assumptions about the jump lengths. For example, if the jump length distribution is not symmetric about zero, different kinds of "one-sided" fractional operators can be defined. We will concentrate on the symmetric case, which as we saw, leads to the fractional Laplacian operator $(-\Delta)^{\alpha/2}$. So far we only considered this operator in one dimension, and on an infinite domain. Its definition follows the usual convention for computing a function of a self-adjoint ("positive definite") operator. That is, it's defined by its action in Fourier space

$$\mathcal{F}\{(-\Delta)^{\alpha/2}u(x)\} = |k|^{\alpha}\mathcal{F}\{u(x)\}.$$

In fact, if we interpret $\mathcal{F}$ as the n-dimensional Fourier transform, and x and k as vectors in $\mathbf{R}^n$ with $|k|$ being the usual Euclidean norm, this definition holds for any number of space dimensions. So there is nothing much more to say about that.

What about on finite domains? On finite domains we also have boundary conditions, which must somehow be part of the definition of the operator. There is lots of current research on the fractional Laplacian on finite domains, with various choices of boundary conditions. Here we will focus on the simplest case: one-dimensional domain $[0, L]$ with homogeneous Dirichlet boundary conditions.

In this case, we can use the spectral decomposition of the Laplacian $(-\Delta)$ to define its fractional power. To find the eigenvalues and eigenfunctions of the Laplacian, we need to solve the equation

$$(-\Delta)\varphi = \lambda\varphi$$

subject to the homogeneous Dirichlet boundary conditions

$$\varphi(0) = \varphi(L) = 0.$$

Here λ is the eigenvalue, and φ is the eigenfunction. The solutions are found to be

$$\lambda_n = \frac{n^2\pi^2}{L^2} \text{ and } \varphi_n(x) = \sin\left(\frac{n\pi x}{L}\right) \text{ for } n = 1, 2, \ldots$$

Before defininining the fractional Laplacian, let's review some key points about this spectral decomposition. Since the operator is self-adjoint, the eigenfunctions are *orthogonal*:

$$\int_0^L \varphi_m(x)\varphi_n(x)dx = 0,\ m \neq n$$

and any reasonable function $u(x)$ defined on $[0, L]$ can be expressed in terms of these eigenfunctions as

$$u(x) = \sum_{n=1}^{\infty} c_n \varphi_n(x)$$

with the coefficients c_n found by integration of u against the eigenfunctions

$$c_n = \frac{2}{L}\int_0^L u(x)\,\varphi_n(x)dx.$$

So, the effect of the Laplacian $(-\Delta)$ on a function u is seen to be

$$(-\Delta)u(x) = (-\Delta)\sum_{n=1}^{\infty} c_n \varphi_n(x) = \sum_{n=1}^{\infty} c_n (-\Delta)\varphi_n(x) = \sum_{n=1}^{\infty} c_n \lambda_n \varphi_n(x)$$

The ***fractional*** **Laplacian** on finite domain is then *defined* by

$$(-\Delta)^{\alpha/2}u(x) = \sum_{n=1}^{\infty} c_n {\lambda_n}^{\alpha/2} \varphi_n(x).$$

Notice that this definition depends on the particular eigenfunctions φ_n, and so it's specific to the choice of boundary conditions. Other boundary conditions (like Neumann conditions) would have different φ_n.

Analytical solution for the fractional diffusion equation

We remember from lecture 1 there was no analytical solution for the fractional diffusion equation

$$\partial u(x,t)/\partial t = -D(-\Delta)^{\alpha/2}u(x,t)$$

on an infinite domain, with initial condition $u(x,0) = \delta(x)$. The solution is known in Fourier space though, and we recognised it as the characteristic function of the α-stable distribution. We won't go any further with this idea.

On a finite domain, we *can* find the analytical solution, for arbitrary initial condition, by using the eigenfunction expansion.

We write the solution $u(x,t)$ in terms of the eigenfunctions with time-varying coefficients

$$u(x,t) = \sum_{n=1}^{\infty} c_n(t)\,\varphi_n(x).$$

Substituting in the PDE, we simplify

$$\sum_{n=1}^{\infty} \left({c_n}'(t) + D {\lambda_n}^{\alpha/2} c_n(t) \right) \varphi_n(x) = 0$$

and find the ODE for the coefficients

$${c_n}'(t) + D{\lambda_n}^{\alpha/2} c_n(t) = 0.$$

The general solution is

$$c_n(t) = c_n(0)\, \exp(-D {\lambda_n}^{\alpha/2} t).$$

To match an arbitrary initial condition

$$u(x,0) = f(x)$$

requires

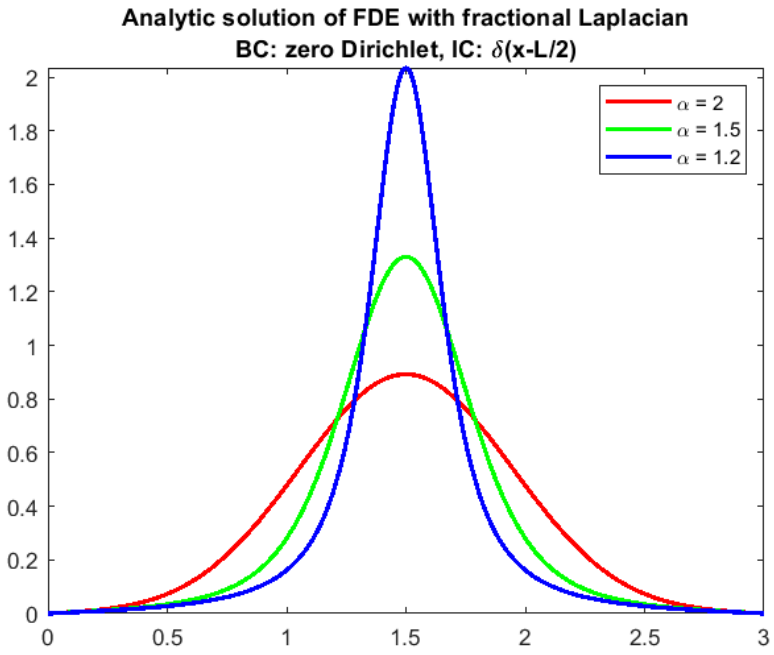
$$c_n(0) = \frac{2}{L} \int_0^L f(x)\, \varphi_n(x) dx$$

and so the solution to the PDE is

$$u(x,t) = \sum_{n=1}^{\infty} c_n(0)\, \exp(-D {\lambda_n}^{\alpha/2} t)\, \varphi_n(x).$$

Let's try an example. We'll take the interval $[0, 3]$ with homogeneous Dirichlet boundary conditions, and initial condition $u(x,0) = \delta(x - L/2)$. The diffusivity is $D = 1$ and we'll plot solutions at time $t = 0.1$ for different values of α.

```
Lec2_Analytic_Solution_FDE
```



You can see the different solution profiles for various values of α, but all solutions are forced to zero on the boundary. If you wanted to derive this problem using a random walk model like in lecture 1, it would involve "killing" the particles once they reached $x = 0$ or $x = L$. We won't down this path though.

It's also worth noting that the solutions for different α values aren't directly comparable even if they use the same value of D. That's because the *dimensions* of D depend on α. For the equation $\partial u(x,t)/\partial t = -D(-\Delta)^{\alpha/2} u(x,t)$ to be dimensionally correct, D must have dimensions of $\text{length}^{\alpha}/\text{time}$. This was also clear from our derivation in lecture 1, where D was defined as the limiting value of $\frac{C_{\alpha} {x_m}^{\alpha}}{\Delta t}$ (remember that x_m has dimensions of length).

Numerical scheme for fractional diffusion-reaction equations

Now we will move on to fractional diffusion-reaction equations by including a nonlinear source term

$$\partial u/\partial t = -D(-\Delta)^{\alpha/2} u + g(u).$$

The analytical solution is not available any more. Instead we need to derive a numerical method. The idea is to build a matrix approximation to the Laplacian operator $-\Delta$. Let's call it $\mathbf{A}$. We can use finite differences, or any similar method. Then the PDE is discretised in space by

$$d\mathbf{u}/dt = -D\mathbf{A}^{\alpha/2}\mathbf{u} + g(\mathbf{u}).$$

Remember that fractional powers of a matrix are defined through its spectral decomposition (by finding its eigenvalues and eigenvectors). So this formulation is the discrete equivalent of our fractional diffusion-reaction equation.

We will discretise in time in the simplest way, treating the diffusion term implicitly and the source term explicitly. (If you haven't studied numerical methods for PDEs before, the reason to treat the diffusion term implicitly is so the method produces reliable answers whatever time step size τ is used.)

$$\frac{\mathbf{u}^n - \mathbf{u}^{n-1}}{\tau} = -D\mathbf{A}^{\alpha/2}\mathbf{u}^n + g(\mathbf{u}^{n-1}).$$

Rearrange to get

$$\mathbf{u}^n = (\mathbf{I} + D\tau\mathbf{A}^{\alpha/2})^{-1}(\mathbf{u}^{n-1} + \tau g(\mathbf{u}^{n-1})).$$

Since the operator $(-\Delta)$ is self-adjoint, the matrix $\mathbf{A}$ will be symmetric positive definite, and so its diagonalisation is just

$$\mathbf{A} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^T$$

where $\mathbf{P}$ has the orthonormal eigenvectors as columns, and the matrix of eigenvalues $\mathbf{\Lambda} = \mathrm{diag}(\lambda_i)$. Then the matrix power $\mathbf{A}^{\alpha/2}$ can be computed as

$$\mathbf{A}^{\alpha/2} = \mathbf{P}\mathbf{\Lambda}^{\alpha/2}\mathbf{P}^T.$$

From this diagonalisation, the solution $\mathbf{u}^n$ can be calculated as

$$\mathbf{u}^n = \mathbf{P}\,\mathrm{diag}\left(\frac{1}{1 + D\,\tau\,\lambda_i^{\alpha/2}}\right)\mathbf{P}^T(\mathbf{u}^{n-1} + \tau g(\mathbf{u}^{n-1})).$$

So now we have a numerical scheme based on the diagonalisation to advance the solution in time.

A fast algorithm for computing $\mathbf{u}^n$

On a uniform grid with N divisions, for the interval $[0, L]$, the finite difference matrix is just $\mathbf{A} = \mathrm{tridiag}(-1, 2, -1)/h^2 \in \mathbf{R}^{N-1\times N-1}$, where $h = L/N$ is the spacing between nodes. This matrix has some very nice properties for our numerical method. In fact, we already know its diagonalisation. The eigenvalues are

$$\lambda_i = \frac{4}{h^2}\sin^2\left(\frac{\pi i}{2N}\right)$$

and the entries of the eigenvector matrix are

$$p_{ij} = \sqrt{\frac{2}{N}}\sin\left(\frac{\pi ij}{N}\right)$$

The connection between these formulas, and the eigenvalues and eigen*functions* of the operator $(-\Delta)$ are interesting, but we won't go further into that. Let's take these formulas as known. Let $\mathbf{v} = \mathbf{u}^{n-1} + \tau g(\mathbf{u}^{n-1})$ at each time step. Then computing $\mathbf{u}$ at each time step would be

$$\mathbf{u} = \mathbf{P}\,\mathrm{diag}\left(\frac{1}{1 + D\,\tau\,\lambda_i^{\alpha/2}}\right)\mathbf{P}^T\mathbf{v}$$

which is really the 3 steps:

1. Let $\mathbf{w} = \mathbf{P}^T\mathbf{v}$
2. Scale $\mathbf{w}$ element-wise by the values $1/(1 + D\tau\lambda_i^{\alpha/2})$
3. Let $\mathbf{u} = \mathbf{P}\mathbf{w}$

Now the products $\mathbf{P}\mathbf{w}$ and $\mathbf{P}^T\mathbf{v}$ are quite interesting. Actually from the formula for p_{ij} it's clear that $\mathbf{P}$ is symmetric ($p_{ij} = p_{ji}$) , so $\mathbf{P} = \mathbf{P}^T$ and we might as well just look at $\mathbf{u} = \mathbf{P}\mathbf{w}$. Written out in full, the *i*th element in $\mathbf{u}$ is

$$u_i = \sqrt{\frac{2}{N}} \sum_{j=1}^{N-1} \sin\left(\frac{\pi ij}{N}\right) w_j$$

This calculation already has a name: it's called the *discrete sine transform*. It's closely related to the discrete *Fourier* transform, and indeed it can be computed using the famous **FFT** algorithm (Fast Fourier Transform) in only $O(N \log N)$ operations. Altogether, this means we have a wonderfully efficient algorithm to compute $\mathbf{u}$ at each time step as

$$\mathbf{u} = \mathrm{idst}\left(\mathrm{dst}(\mathbf{v})./(1 + D\tau\lambda_i^{\alpha/2})\right),$$

where dst and $idst$ are MATLAB's built-in functions for computing discrete sine transform and its inverse transform. For Neumann boundary conditions, the formulae for eigenvalues and eigenvectors of $\mathbf{A}$ can be found here, and the solution $\mathbf{u}$ can be computed using *discrete cosine transform*. dct and $idct$ are the MATLAB's functions for this purpose.

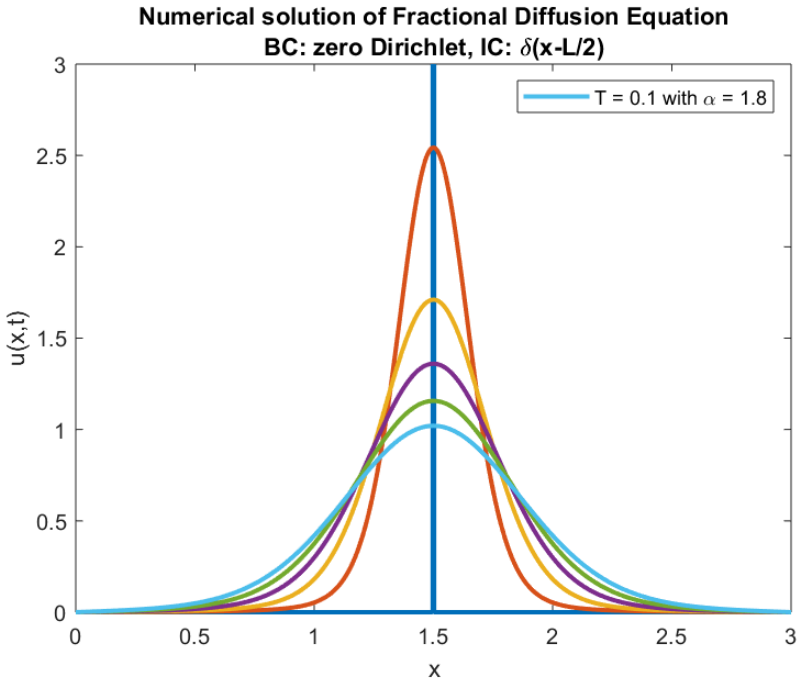
Now, Let's have a look at our MATLAB examples.

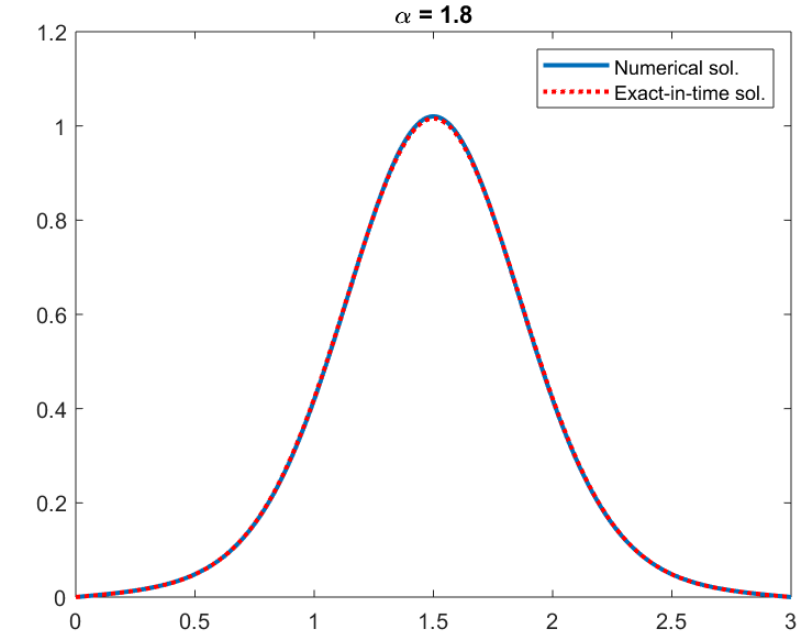
Example 1: Solve the fractional diffusion equation numerically on the interval $[0, 3]$ under zero Dirichlet boundary conditions and initial condition $u(x, 0) = \delta(x - L/2)$. The diffusivity $D = 1$ and the final time $T = 0.1$. Introduce a mesh with Ndivisions in space and n steps in time:

$x_i = ih,\ i = 1, 2, ..., N - 1$, where $h = L/N$, and

$t_j = j\tau,\ j = 0, 1, 2, ..., n$ where $\tau = T/n$.

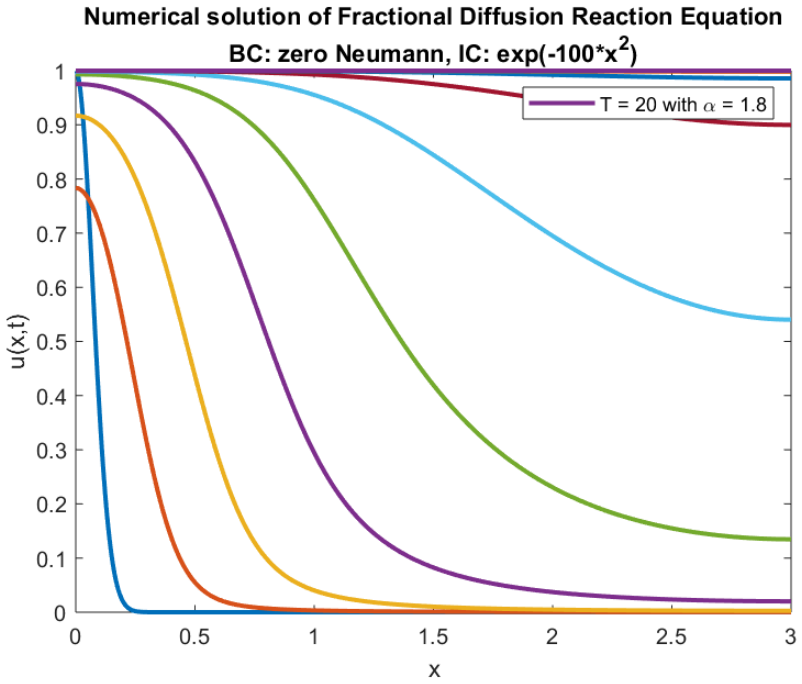
```
Lec2_Numerical_Solution_FDE_Dirichlet
```





Example 2: Solve the fractional diffusion-reaction equation with a Fisher source term $g(u) = u(1-u)$, under zero Neumann boundary condition and initial condition $u(x,0) = \exp(-100x^2)$. $D = 0.01$ and $T = 20$.

```
Lec2_Numerical_Solution_FDRE_Neumann
```



Conclusions for this lecture: If you're solving a 1D fractional diffusion-reaction problem, it is quite tempting to use this finite difference discretisation because of the fast algorithms for discrete sine and cosine transforms. It even generalises to higher dimensions. But if you need a nonuniform mesh, or if your domain is irregular, then the fast algorithms can't be used anymore.

In the next lecture we'll look at other ways of computing $\mathbf{u}^n$ that apply in more general situations.