

# Characteristics of advection-diffusion equation

Fangbo Wang

September 2017

The advection-diffusion equation is a well-known PDE which can capture advection and diffusion effects that are popular in the physical world. It can be used in fluid mechanics (Navier-Stokes equation), stochastic differential theory (Fokker-Planck equation), finance modeling (Black-Scholes equation), etc. The equation can be written as below:

$$\frac{\partial f}{\partial t} = N_1 \frac{\partial f}{\partial x} + N_2 \frac{\partial^2 f}{\partial x^2} \quad (1)$$

Where  $N_1$  and  $N_2$  are the advection and diffusion coefficients, respectively.

There are some interesting characteristics for advection-diffusion equation which we will explore below in an probabilistic context. Assume we have an initial condition of a standard normal probability density function. The boundary is set to be  $[-20 \ 20]$  for numerical computation which is actually infinite, and also an reflecting boundary is employed [1].

Assume  $timestart = 0, dt = 0.01, N_1 = 100, N_2 = 1$  as the standard parameters.

First of all, we can change the advection coefficient to see its effect. As can be seen in Fig. 1, positive  $N_1$  would shift to right and negative  $N_1$  would shift to left. The advection coefficient can be understood as the moving velocity of a particle. The moving distance is 5 with velocity 500 and time 0.01 second as can be seen on the bottom right plot.

Similarly, the effect of diffusion coefficient can be shown in Fig. 2. With positive diffusion coefficient, the shape of the function would diffuse to have a larger spreadness. However, negative diffusion coefficient which indicating concentration would render the equation ill-posed and difficult to solve the problem. It can be categorized as a backward heat equation. It's common

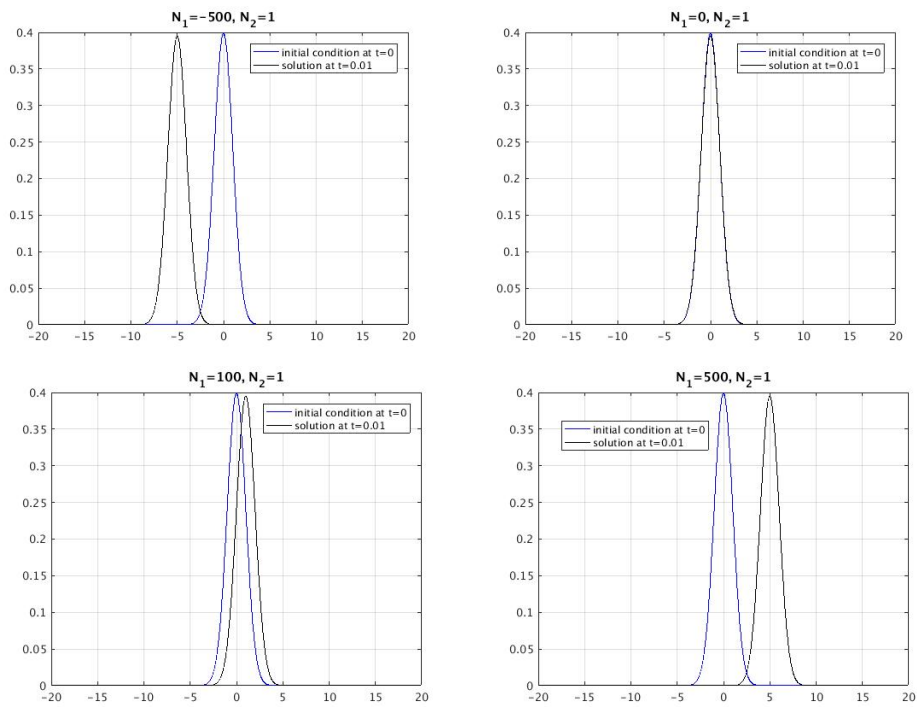


Figure 1: Effects of changing advection coefficient.

to see high frequency oscillations in the solution, that's because the negative diffusion process behaves like differentiation to make the solution coarser than initial condition. However, positive diffusion process behaves like integration to make the solution smoother than initial condition [3]. Numerous techniques have been proposed to overcome this ill-posed problem, and regularization is one of them, see ref [2].

Furthermore, negative diffusion coefficient is an inverse process which might have multiple possible solutions. Only equation with very small negative diffusion coefficient would be solved. Regarding this specific problem, the smallest possible value of  $N_2$  is -0.2. Solution with negative diffusion coefficient would have smaller standard deviation due to concentration effect (standard deviation=0.998).

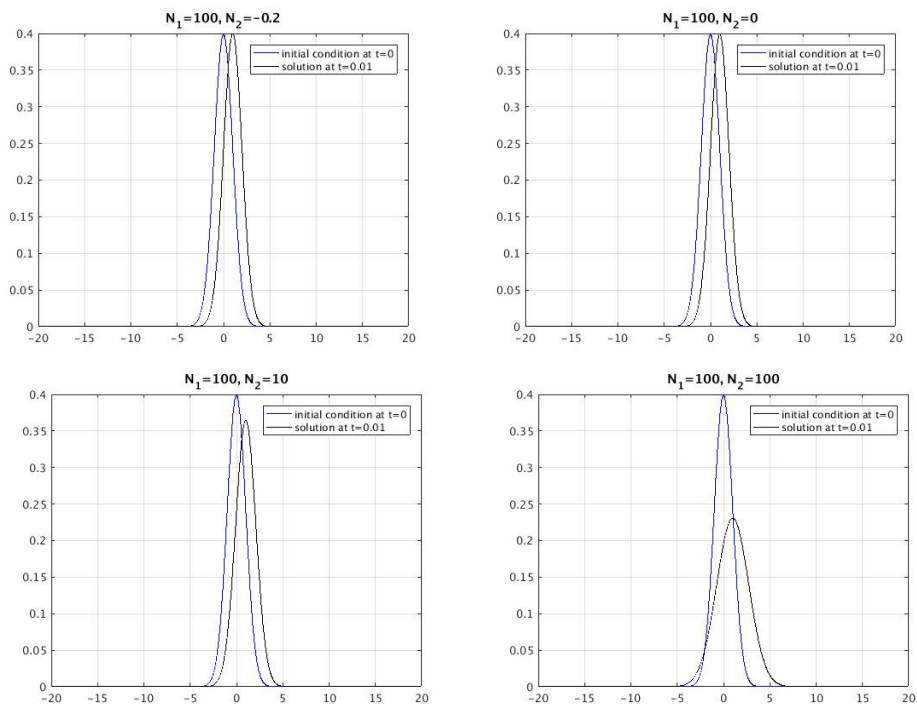


Figure 2: Effects of changing diffusion coefficient.

Furthermore, the effect of changing starting time is shown in Fig. 3. It is observed that starting time doesn't have influence on the result.

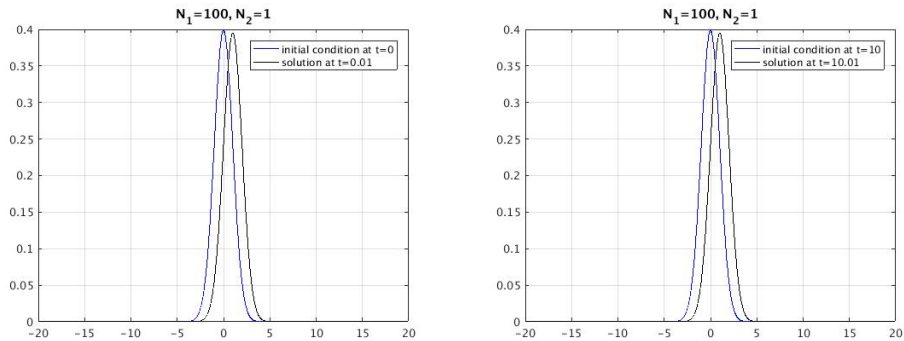


Figure 3: Effects of changing starting time.

The effect of changing the incremental time is shown in Fig. 4. It is seen that longer incremental time would both advect and diffuse the function along time.

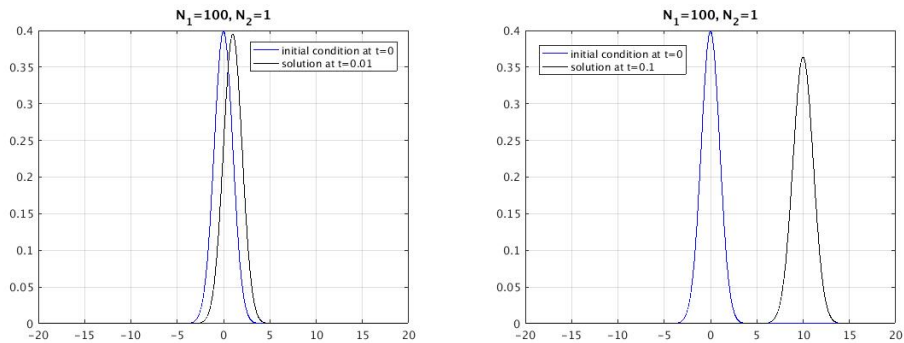


Figure 4: Effects of changing incremental time.

Below is the main.m script file.

```

clc; clear;
leftbound=-20; rightbound=20; meshpoint=2000;
stressmesh=linspace(leftbound, rightbound, meshpoint);
% standard normal assumption;
u00=exp(-stressmesh.*stressmesh/2)/sqrt(2*pi);

```

```

timestart=10; dt=0.01;
N1=100; N2=1;

[sol]=solve_FPK(u00, N1, N2, timestart, dt, leftbound, rightbound, meshpoint);

plot(stressmesh, u00, 'b'); hold on;
plot(stressmesh, sol, 'k'); grid on;
legend('initial condition at t=0', 'solution at t=0.01' )
title('N_1=100, N_2=1')

```

Below is the solve\_FPK function file

```

function [sol]=solve_FPK(u00, N1, N2, timestart, timeincrement_small, leftbound,
                        rightbound, meshpoint)

    x = linspace(leftbound, rightbound, meshpoint);
    t = linspace(timestart, timestart+timeincrement_small, 101);
    m = 0;
    solution = pdepe(m,@pdex1pde,@pdex1ic,@pdex1bc,x,t);
    sol=solution(end,:);

    function [c,f,s] = pdex1pde(x,t,u,DuDx)
        c = 1;
        f = N2*DuDx;
        s = -N1*DuDx;
    end

    function u0 = pdex1ic(x)
        u0=interp1( linspace(leftbound, rightbound, meshpoint), u00, x );
    end

    function [pl,q1,pr,qr] = pdex1bc(xl,ul,xr,ur,t)
        pl = N1*ul;
        q1 = -1;
        pr = N1*ur;
        qr = -1;
    end

end

```

## References

- [1] Arezoo Sadrinezhad, Kallol Sett and S. I. Hariharan. Efficient solution algorithms for multiaxial probabilistic elasto-plastic constitutive simulations of soils. *Int J Numer Anal Mech Geomech.* 2017; 0:1-21.
- [2] Chuli Fu, Xiangtuan Xiong and Zhi Qian. Fourier regularization for a backward heat equation. *Journal of Mathematical Analysis and Applications.* 2007; 331:471-480.
- [3] Lloyd N. Trefethen and Kristine Embree. The PDE Coffee Table Book. Chapter 3. Backward heat equation...ill-posed problems and regularisation. <https://people.maths.ox.ac.uk/trefethen/pdectb.html>. 2011.