FYS3150

Fredrik WIlhelm Holmen

November 14, 2013

Analytical solution

The diffusion equations is given as:

$$\frac{\partial u}{\partial t} = D\frac{\partial^2 u}{\partial x^2}$$

We want the boundary conditions equal to 0, so we define a new function

$$v(x,t) = u(x,t) - u_s(x,t)$$

and solve the diffusion equation for this function. This equation can be solved by separation of variables. Setting $v(x,t) = X(x)T(t)$ gives:

$$\frac{dT}{dt}\frac{1}{T(t)}\frac{1}{D^2} = \frac{1}{X(x)}\frac{d^2X}{dx^2}$$

Both LHS and RHS must be equal to the same constant for this equation to hold at all times. We get:

$$\frac{dT}{dt}\frac{1}{T(t)}\frac{1}{D^2} = -\lambda^2 \text{and } \frac{1}{X(x)}\frac{d^2X}{dx^2} = -\lambda^2$$

The soltion for the time equation is

$$T(t) = e^{-D^2\lambda^2 t}$$

while the spacial solution is

$$X(x) = Asin(\lambda x) + Bcos(\lambda x)$$

Setting $D = 1$. By the boundary conditions $v(0,t) = v(1,t) = 0$ we see that $B = 0$ and that $\lambda = n\pi$. So we get

$$v(x,t) = \sum_{n=1}^{\infty} C_n sin(n\pi x)e^{-n^2\pi^2 t}$$

The initial condition is given on the form $Ax + b$, so $u_s(x) = 1 - x$ fullfills the boundary conditions. We have defined $v(x,t) = u(x,t) - u_s(x)$ so $u(x,t) = v(x,t) + u_s(x)$,

$$u(x,t) = 1 - x + \sum_{n=1}^{\infty} C_n sin(n\pi x)e^{-n^2\pi^2 t}$$

To find the C_ns, I solve the equation for time equal to 0. I know that $u(x,0) = 0$ for $0 < x < 1$, so

$$v(x,0) = \sum_{n=1}^{\infty} C_n sin(n\pi x) = x - 1$$

Using the fourier expansion we see that

$$C_n = \frac{-2}{\pi n}$$

and we get the full solution

$$v(x,t) = \sum_{n=1}^{\infty} \frac{-2}{\pi n} sin(n\pi x)e^{-n^2\pi^2 t}$$

Since this should be solved for $u(x,t)$, we get

$$u(x,t) = \sum_{n=1}^{\infty} \frac{-2}{\pi n} sin(n\pi x)e^{-n^2\pi^2 t} + x - 1$$

Algorithms

I will use three different schemes to solve the diffusion equation. First I have implemented the explicit scheme, then the implicit scheme, and at last I will use Crank-Nicolson.

The explicit scheme

The Explicit scheme is a very straight-forward coding of the following equation.

$$v_{i,j+1} = \alpha v_{i-1,j} + (1-2\alpha)v_{i,j} + \alpha v_{i+1,j}$$

where

$$\alpha = \frac{\Delta t}{\Delta x^2}$$

and the error going as

$$O(\Delta t, \Delta x^2)$$

One can see that without boundary and initial conditions, this equation is unsolvable, but since we have the conditions given, we do not need to compute for $j = 0$ and $i = 0$ or $i = j$.

$$\text{boundary conditions: } v(0) = v(L) = 0$$

Since we are solving for $v(x,t)$, we need to use the initial state given from the analytical section above:

$$v(x,0) = x - 1, \text{ for } 0 < x < 1 \text{ and } v(0,0) = v(1,0) = 0$$

The Implicit scheme

The implicit scheme is obtained by using the backward formula for the first derivative. This will result in the matrix-vector multiplication:

$$Av_j = v_{j-1}$$

where A is a tridiagonal matrix with off-diagonal elements $-\alpha$ and diagonal elements $1+2\alpha$

I solve this equation by using the solver from project 1 by setting v_0 by the initial conditions and the boundary conditions. Then I solve the equation N_t times to .

The Crank-Nicolson scheme The Crank-Nicolson scheme is a combination of the implicit and the explicit schemes. This scheme is shortly written as

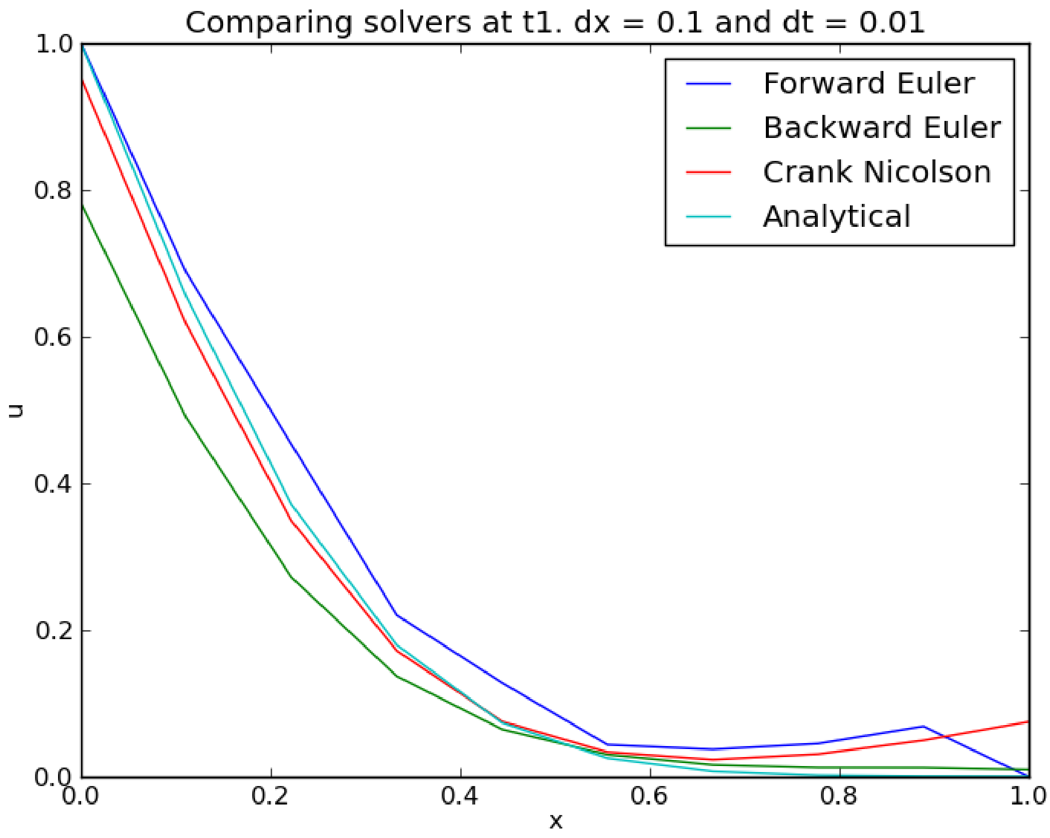
$$(2I + \alpha B)V_j = (2I - \alpha B)V_{j-1}$$

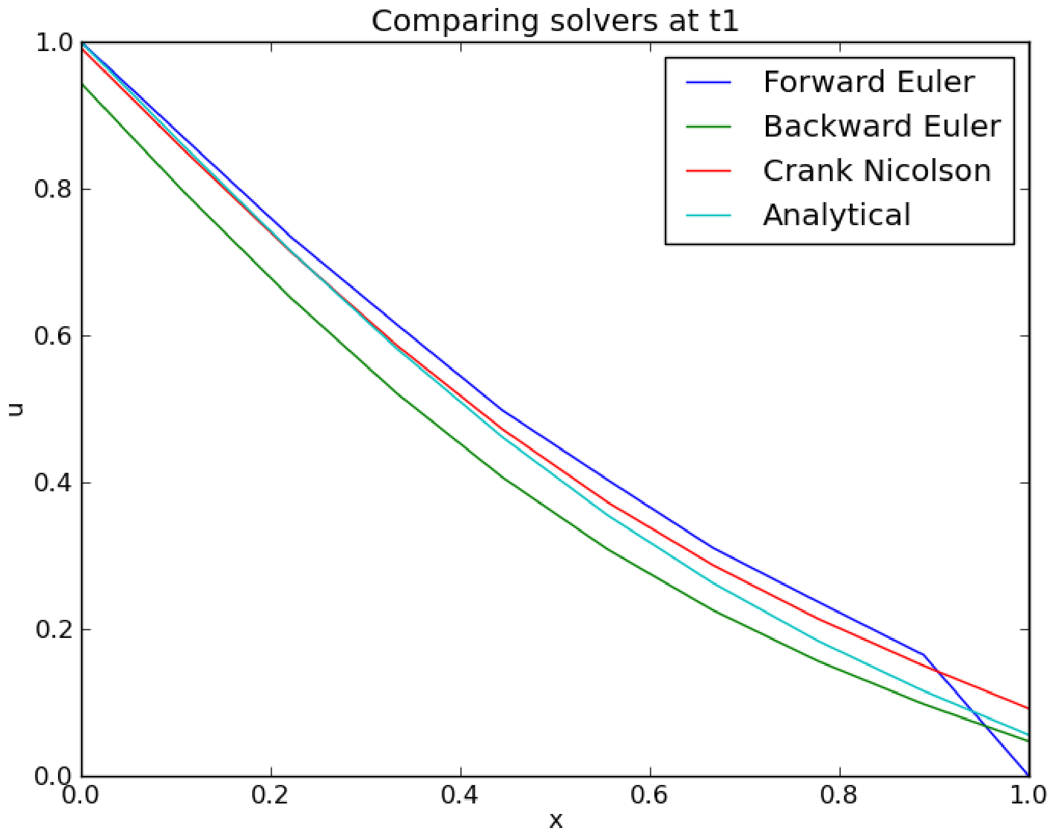
Numerical Notes

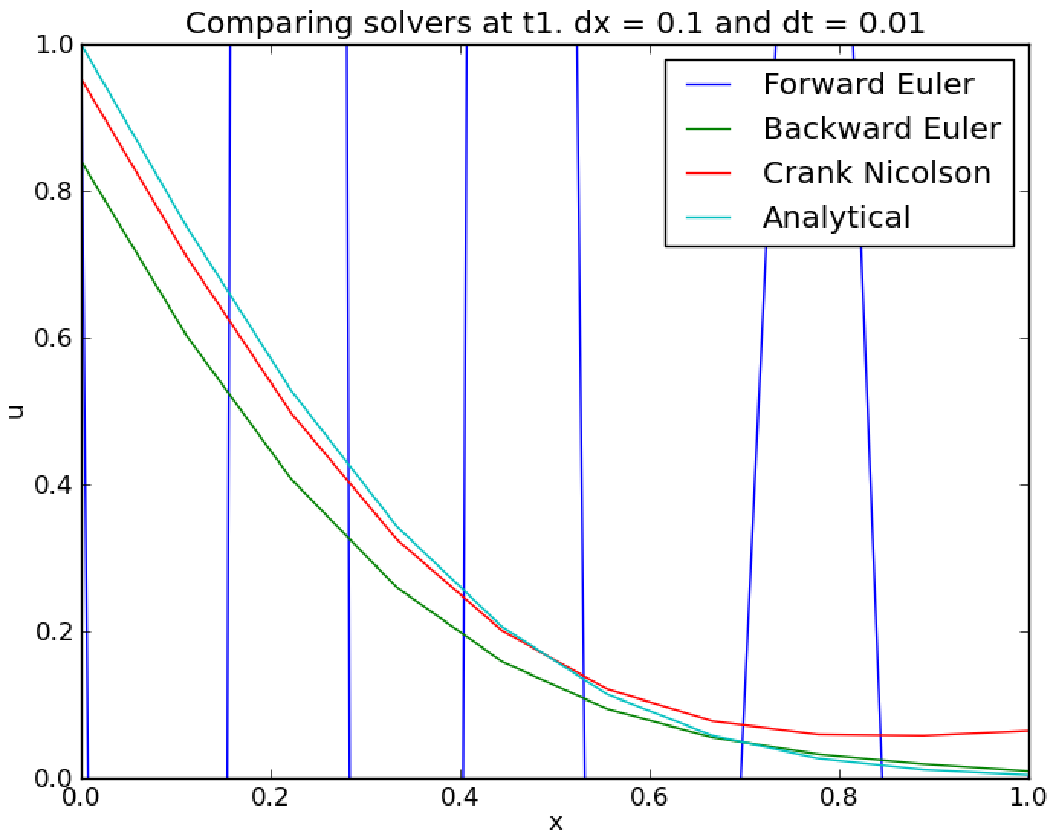
When using $\Delta x = \frac{1}{10}$ and $\Delta t = \frac{1}{100}$, the Explicit scheme will go berzerk and give horrible results. This is because this scheme is unstable for $\Delta t \geq \frac{1}{2}\Delta x^2$. So when increasing Δt to $\Delta t = \frac{1}{200}$ I get stable results.

None of the numerical approximations manage to overlap with the analytical solution, even though Crank-Nicolson is pretty close. The error for Crank-Nicolson is smaller because the error goes as Δt^2 and Δx^2, but for explicit and implicit scheme, the error goes as Δx^2 and Δt^2.

Results and plots




Comparing solvers at t1
Forward Euler
Backward Euler
Crank Nicolson
Analytical
u
x
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0


Comparing solvers at t1. dx = 0.1 and dt = 0.01
Forward Euler
Backward Euler
Crank Nicolson
Analytical
u
x
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0