

Numerical Solver Method

Jeffrey Rutledge

The function titled `CrankNicolson` is the solver method intended for use, and explained here. The other functions use different schemes.

Problem

The PDE this solver numerically approximates is,

$$\frac{\partial c_g}{\partial t} = D_e \frac{\partial^2 c_g}{\partial z^2} - V_e \frac{\partial c_g}{\partial z}$$

where c_g is the concentration of the tracer in the gas phase, z is the depth under the surface, D_e is the effective diffusion constant, V_e is the effective velocity, and t is time. The boundary at the surface, $c_g(t, z = 0)$, is the measured atmospheric concentration of the tracer gas. The initial boundary, $c_g(t = 0, z)$, and the boundary at the max depth of 200 meters, $c_g(t, z = 200)$, are both assumed to be zero.

Solution

The PDE is approximated using a scheme similar to the Crank Nicolson method, an implicit finite difference scheme. Let u_j^m be the approximation of $c_g(t, z)$ at $t = m\Delta t$ and $z = j\Delta z$. The approximations used are forward difference in time,

$$\frac{\partial c_g}{\partial t} \approx \frac{u_j^{m+1} - u_j^m}{\Delta t},$$

and central differences in position averaged over the $(m + 1)$ th time step and the m th time step,

$$\frac{\partial^2 c_g}{\partial z^2} \approx \frac{1}{2} \left(\frac{u_{j+1}^{m+1} - 2u_j^{m+1} + u_{j-1}^{m+1}}{\Delta x^2} + \frac{u_{j+1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2} \right)$$

and

$$\frac{\partial c_g}{\partial z} \approx \frac{1}{2} \left(\frac{u_{j+1}^{m+1} - u_{j-1}^{m+1}}{2\Delta x} + \frac{u_{j+1}^m - u_{j-1}^m}{2\Delta x} \right).$$

When substituted into the original equation these approximations yield the equation,

$$\frac{u_j^{m+1} - u_j^m}{\Delta t} = \frac{1}{2} \left(D_e \frac{u_{j+1}^{m+1} - 2u_j^{m+1} + u_{j-1}^{m+1} + u_{j+1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2} - V_e \frac{u_{j+1}^{m+1} - u_{j-1}^{m+1} + u_{j+1}^m - u_{j-1}^m}{2\Delta x} \right).$$

This can be and rearranged into the equation,

$$\begin{aligned} (-2D_e - V_e\Delta x)u_{j-1}^{m+1} + 4 \left(\frac{\Delta x^2}{\Delta t} + D_e \right) u_j^{m+1} + (-2D_e + V_e\Delta x)u_{j+1}^{m+1} = \\ (2D_e + V_e\Delta x)u_{j-1}^m + 4 \left(\frac{\Delta x^2}{\Delta t} - D_e \right) u_j^m + (2D_e - V_e\Delta x)u_{j+1}^m. \end{aligned} \tag{1}$$

This gives a system of equations that may be solved using a tridiagonal matrix.

Accuracy

The depth approximations are both central differences, which are second order, so their error is $O(\Delta x^2)$. The time approximation is a forward difference which is only first order, but because the depth approximations are averaged over the $(m+1)$ th and the m th time steps the error in time is $O(\Delta t^2)$. Thus the approximation is second order in the time and depth steps.

Stability

To use von Neumann stability we will make the substitution,

$$u_j^m = Q^m e^{ij\Delta x k},$$

where Q^m is the magnitude at the m th time step, k is frequency, and i is $\sqrt{-1}$.

Substituting this into the equation (1),

$$\begin{aligned} (-2D_e - V_e\Delta x)Q^{m+1}e^{i(j-1)k} + 4\left(\frac{\Delta x^2}{\Delta t} + D_e\right)Q^{m+1}e^{ijk} + (-2D_e + V_e\Delta x)Q^{m+1}e^{i(j+1)k} = \\ (2D_e + V_e\Delta x)Q^m e^{i(j-1)k} + 4\left(\frac{\Delta x^2}{\Delta t} - D_e\right)Q^m e^{ijk} + (2D_e - V_e\Delta x)Q^m e^{i(j+1)k}. \end{aligned}$$

This can be simplified into,

$$Q^{m+1} = Q^m a,$$

where a is the amplification factor,

$$a = \frac{(2D_e + V_e\Delta x)e^{-i\Delta x k} + (2D_e - V_e\Delta x)e^{i\Delta x k} + 4\frac{\Delta x^2}{\Delta t} - 4D_e}{(-2D_e - V_e\Delta x)e^{-i\Delta x k} + (-2D_e + V_e\Delta x)e^{i\Delta x k} + 4\frac{\Delta x^2}{\Delta t} + 4D_e}.$$

Now we can reduce this further using Euler's formula to,

$$a = \frac{4D_e \cos \Delta x k - 2V_e \Delta x i \sin \Delta x k + 4\frac{\Delta x^2}{\Delta t} - 4D_e}{-4D_e \cos \Delta x k + 2V_e \Delta x i \sin \Delta x k + 4\frac{\Delta x^2}{\Delta t} + 4D_e}$$

and finally to,

$$a = \frac{1 - \frac{D_e \Delta t}{\Delta x^2}(1 - \cos \Delta x k)}{1 + \frac{D_e \Delta t}{\Delta x^2}(1 - \cos \Delta x k)} - i.$$

From here it can be seen that the magnitude of a will always be less than or equal to 1. Thus this method is unconditionally stable.