Objectives:

Introduce the basic concepts needed to solve a partial differential equation using finite difference methods.

Discuss basic time integration methods, ordinary and partial differential equations, finite difference approximations, accuracy.

Show the implementation of numerical algorithms into actual computer codes.

Outline

- Solving partial differential equations
  - Finite difference approximations
  - The linear advection-diffusion equation
  - Matlab code
  - Accuracy and error quantification
  - Stability
  - Consistency
  - Multidimensional problems
  - Steady state

Derive a numerical approximation to the governing equation, replacing a relation between the derivatives by a relation between the discrete nodal values.

The Time Derivative
The Time Derivative is found using a FORWARD EULER method. The approximation can be found by using a Taylor series

\[ f(t + \Delta t) = f(t) + \frac{\partial f(t)}{\partial t} \Delta t + \frac{\partial^2 f(t)}{\partial t^2} \Delta t^2 + \cdots \]

Solving this equation for the time derivative gives:

\[ \frac{\partial f(t)}{\partial t} = \frac{f(t + \Delta t) - f(t)}{\Delta t} - \frac{\partial^2 f(t)}{\partial t^2} \Delta t + \cdots \]

Apply this to the ODE from last lecture:

The original equation at time level \( n \) is:

\[ \frac{df}{dt}(n) = g_n \]

And using the approximation just derived for the time derivative results in:

\[ \frac{f^{n+1} - f^n}{\Delta t} = g^n + O(\Delta t) \]

which is exactly the first order forward Euler method.

We will use the model equation:

\[ \frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2} \]

to demonstrate how to solve a partial equation numerically.

Although this equation is much simpler than the full Navier Stokes equations, it has both an advection term and a diffusion term.

Before attempting to solve the equation, it is useful to understand how the analytical solution behaves.
When using FINITE DIFFERENCE approximations, the values of \( f \) are stored at discrete points.

\[
\frac{f(x+h) - f(x)}{h} \quad \frac{f(x) - f(x-h)}{h}
\]

The derivatives of the function are approximated using a Taylor series.

### The Spatial First Derivative

Start by expressing the value of \( f(x+h) \) and \( f(x-h) \) in terms of \( f(x) \):

\[
f(x+h) = f(x) + \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \cdots
\]

\[
f(x-h) = f(x) - \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} - \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \cdots
\]

Subtracting the second equation from the first:

\[
f(x+h) - f(x-h) = 2\frac{\partial f(x)}{\partial x} h + \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \cdots
\]

### The Spatial Second Derivative

Start by expressing the value of \( f(x+h) \) and \( f(x-h) \) in terms of \( f(x) \):

\[
f(x+h) = f(x) + \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} + \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \cdots
\]

\[
f(x-h) = f(x) - \frac{\partial f(x)}{\partial x} h + \frac{\partial^2 f(x)}{\partial x^2} \frac{h^2}{2} - \frac{\partial^3 f(x)}{\partial x^3} \frac{h^3}{6} + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \cdots
\]

Adding the second equation to the first:

\[
f(x+h) + f(x-h) = 2f(x) + \frac{\partial^2 f(x)}{\partial x^2} h^2 + \frac{\partial^4 f(x)}{\partial x^4} \frac{h^4}{24} + \cdots
\]
Using our shorthand notation gives:

\[ f_j^n = f(t, x_j) \]
\[ f_{j+1}^{n+1} = f(t + \Delta x, x_j) \]
\[ f_{j+1}^n = f(t, x_j + h) \]
\[ f_{j-1}^n = f(t, x_j - h) \]

\[ \left( \frac{\partial f}{\partial x} \right)_j = \frac{f_{j+1}^n - f_{j-1}^n}{2h} + O(h^2) \]
\[ \left( \frac{\partial f}{\partial t} \right)_j = \frac{f_{j+1}^{n+1} - 2f_j^n + f_{j-1}^n}{h^2} + O(h^2) \]

Solving for the new value and dropping the error terms yields:

\[ f_{j+1}^{n+1} = f_j^n - \frac{U\Delta t}{2h} (f_{j+1}^n - f_{j-1}^n) + \frac{D\Delta t}{h^2} (f_{j+1}^n - 2f_j^n + f_{j-1}^n) \]
Thus, given \( f \) at one time (or time level), \( f \) at the next time level is given by:

\[
f_{j}^{n+1} = f_{j}^{n} - \frac{UM}{2h}(f_{j+1}^{n} - f_{j-1}^{n}) + \frac{DM}{h}((f_{j+1}^{n} - 2f_{j}^{n} + f_{j-1}^{n})
\]

The value of every point at level \( n+1 \) is given explicitly in terms of the values at the level \( n \).

Finite difference approximations by Taylor expansion

Approximating a partial differential equation
Elementary Numerical Analysis: Finite Difference Approximations-III

Grétar Tryggvason
Spring 2010

Computational Fluid Dynamics I

Outline

- Solving partial differential equations
- Finite difference approximations
- The linear advection-diffusion equation
- Matlab code
- Accuracy and error quantification
- Stability
- Consistency
- Multidimensional problems
- Steady state

Computational Fluid Dynamics I

Evolution for
\[ U=1; \]
\[ D=0.05; \]
\[ k=1 \]
\[ N=21 \]
\[ \Delta t=0.05 \]

Exact
Numerical

Accuracy

It is clear that although the numerical solution is qualitatively similar to the analytical solution, there are significant quantitative differences.

The derivation of the numerical approximations for the derivatives showed that the error depends on the size of \( h \) and \( \Delta t \).

First we test for different \( \Delta t \).

Number of time steps = \( T/\Delta t \)
How accurate solution can we obtain?

Take
\[ \Delta t = 0.0005 \]
and
\[ N = 200 \]

Quantifying the Error
Order of Accuracy

Examine the spatial accuracy by taking a very small time step, \( \Delta t = 0.0005 \) and vary the number of grid points, \( N \), used to resolve the spatial direction.

The grid size is \( h = L/N \) where \( L = 1 \) for our case.
If the error is of second order:
\[ E = Ch^2 = C \left( \frac{1}{N} \right)^2 \]

Taking the log:
\[ \ln E = \ln \left( \left( \frac{1}{N} \right)^2 \right) \]
\[ = \ln C - 2 \ln \left( \frac{1}{N} \right) \]

On a log-log plot, the \( E \) versus \( (1/N) \) curve should therefore have a slope -2.

Why is does the error deviate from the line for the highest values of \( N \)?
As long as accuracy is reasonable, integration at larger time steps is more efficient and desirable.

Can we increase the time step indefinitely?

Let's repeat the 1-D advection-diffusion equation with large time steps.

Use $\Delta t = 0.2$, instead of $\Delta t = 0.05$

Ordinary Differential Equation

Stability
Take:
\[
\frac{df}{dt} = -f
\]
with initial condition \( f(0) = 1 \)

The exact solution is
\[
f(t) = e^{-t}
\]

Forward Euler
\[
f^{n+1} = f^n - f^n \Delta t \rightarrow f^{n+1} = (1 - \Delta t) f^n
\]

\[
\left| \frac{f^{n+1}}{f^n} \right| \leq 1 \quad \text{only if} \quad \Delta t \leq 2
\]

Backward Euler
\[
f^{n+1} = f^n - f^{n+1} \Delta t \rightarrow \frac{f^{n+1}}{f^n} = \frac{1}{1 + \Delta t}
\]

\[
\left| \frac{f^{n+1}}{f^n} \right| \leq 1 \quad \text{for all} \quad \Delta t
\]

However
\[
f^{n+1} = (1 - \Delta t) f^n = (1 - \Delta t)^2 f^{n-1}
\]

\[
= \ldots \ldots = (1 - \Delta t)^n f^1
\]

Obviously, \( f \) oscillates unless \( \Delta t \leq 1 \)

Consider the 1-D advection-diffusion equation:
\[
\frac{\partial f}{\partial t} + U \frac{\partial f}{\partial x} = D \frac{\partial^2 f}{\partial x^2}
\]

In finite-difference form:
\[
\frac{f_j^{n+1} - f_j^n}{\Delta t} + U \frac{f_{j+1}^n - f_{j-1}^n}{2h} = D \frac{f_{j+1}^{n+1} - 2 f_j^{n+1} + f_{j-1}^{n+1}}{h^2}
\]

Look at the evolution of a small perturbation
\[
f_j^0 + \delta_j
\]
The evolution of the perturbation is governed by:
\[
\frac{e_{j+1}^n - e_j^n}{\Delta t} + U \frac{e_{j+1}^n - e_{j-1}^n}{2h} = \frac{D e_{j+1}^n - 2e_j^n + e_{j-1}^n}{h^2}
\]

Write the error as a wave (expand as a Fourier series):
\[
e_j^n = e^{-ikx_j}
\]

Dropping the subscript
\[
e_j^n = e_j e^{ikx_j}
\]

The error at node \( j \) is:
\[
e_j^n = e_j e^{ikx_j}
\]

The error at \( j+1 \) and \( j-1 \) can be written as
\[
e_{j+1}^n = e_k e^{ikx_{j+1}} = e_k e^{ik(x_j + h)}
\]
\[
e_{j-1}^n = e_k e^{ikx_{j-1}} = e_k e^{ik(x_j - h)}
\]

Dividing by the error amplitude at \( n \):
\[
e_j^n = \frac{e_{j+1}^n - e_{j-1}^n}{\Delta t} + \frac{U e_{j+1}^n - e_{j-1}^n}{2h} = \frac{D e_{j+1}^n - 2e_j^n + e_{j-1}^n}{h^2}
\]

The equation for the error is:
\[
\frac{e_{j+1}^n - e_j^n}{\Delta t} + U \left( e_j e^{ikx_j} - e_j e^{-ikx_j} \right) = \frac{D e_j^n}{h^2} \left( e_j e^{ikx_j} - 2 + e^{-ikx_j} \right)
\]

Solving for the ratio of the errors:
\[
\frac{e_{j+1}^n}{e_j^n} = 1 - \frac{U \Delta t}{2h} \left( e^{ikx_j} - e^{-ikx_j} \right) + \frac{D \Delta t}{h} \left( e^{ikx_j} - 2 + e^{-ikx_j} \right)
\]

The ratio of the error amplitude at \( n+1 \) and \( n \) is:
\[
\frac{e_{j}^{n+1}}{e_j^n} = 1 - 4 \frac{D \Delta t}{h^2} \sin^2 \frac{kx_j}{2} - i \frac{U \Delta t}{h} \sin k h
\]

Stability requires that
\[
\left| \frac{e_{j}^{n+1}}{e_j^n} \right| < 1
\]

Since the amplification factor is a complex number, and \( k \), the wave number of the error, can be anything, the determination of the stability limit is slightly involved.

We will look at two special cases: (a) \( U = 0 \) and (b) \( D = 0 \)
(a) Consider first the case when $U = 0$, so the problem reduces to a pure diffusion
\[ \epsilon^{n+1}/\epsilon^n = 1 - 4DM \Delta t h^2 \sin^2 k h/2 \]
Since $\sin^2() \leq 1$, the amplification factor is always less than 1, and we find that it is bigger than -1 if
\[-1 \leq 1 - 4DM \Delta t h^2 \leq 1 \]
\[ DM h^2 \leq 1/2 \]

(b) Consider now the other limit where $D = 0$ and we have a pure advection problem.
\[ \epsilon^{n+1}/\epsilon^n = 1 - iU \Delta t h \sin kh \]
Since the amplification factor has the form $1+i()$, the absolute value of this complex number is always larger than unity and the method is unconditionally unstable for this case.

For the general case we must investigate the stability condition in more detail. We will not do so here, but simply quote the results:
\[ \Delta D/\Delta t \leq 1 \]
\[ U^2 \Delta t D \leq 2 \]

Notice that high velocity and low viscosity lead to instability according to the second restriction.

For a two-dimensional problem, assume an error of the form
\[ \epsilon_{ij}^{n+1} = \epsilon_{ij} e^{i(kx_i + ly_j)} \]
A stability analysis gives:
\[ DM/\Delta t h^2 \leq 1/4 \]
\[ (U|+|V|+|W|)^2 \Delta t D \leq 4 \]

For a three-dimensional problem we get:
\[ DM/\Delta t h^2 \leq 1/6 \]
\[ (U|+|V|+|W|)^2 \Delta t D \leq 8 \]

Convergence – the solution to the finite-difference equation approaches the true solution to the PDE having the same initial and boundary conditions as the mesh is refined.

**Lax’s Equivalence Theorem**
Given a properly posed initial value problem and a finite-difference approximation to it that satisfies the consistency condition, stability is the necessary and sufficient condition for convergence.

Stability – Now you know!

**Summary**
Introduced a formal method to examine whether a given finite difference approximation is stable or not

Introduced Lax’s equivalent theorem.