M344 - ADVANCED ENGINEERING MATHEMATICS

Lecture 17: Numerical Solution of Partial Differential Equations

If it is not possible to solve a partial differential equation by separation of variables, or some other analytic method, the solution can always be approximated by numerical methods. The particular methods used depend on the type of the equation, and the method described in this lecture is used for parabolic p.d.e.s. such as the one-dimensional heat equation.

Numerical Approximations to Partial Derivatives

To solve partial differential equations numerically, the basic idea is to partition both the x-axis and the t-axis, and approximate the partial derivatives of the unknown function in terms of *differences* of function values at nearby points in the resulting 2-dimensional grid. We will first define numerical derivatives for functions of a single variable, but since partial derivatives are just ordinary derivatives with respect to one of the variables, it is easy to see how these formulas extend to partial derivatives. The difference formulas that we will use are described below.

For a function f of a single variable x, if Δx is a *small* number, we can approximate the first derivative of f by

$$f'(x) \approx \frac{f(x+\Delta x) - f(x)}{\Delta x}. \tag{1}$$

This is called a **forward difference approximation** to $f'(x)$. If $f(x)$ is assumed to be an analytic function of a single variable, we know that the Taylor Series about any point x is

$$f(x+\Delta x) = f(x) + f'(x)\Delta x + \frac{f''(x)}{2!}(\Delta x)^2 + \cdots \frac{f^{(n)}(x)}{n!}(\Delta x)^n + \cdots . \tag{2}$$

Using a bit of algebra allows us to write

$$\frac{f(x+\Delta x) - f(x)}{\Delta x} = f'(x) + \Delta x\left(\frac{f''(x)}{2!} + \Delta x\frac{f'''(x)}{3!} + \cdots\right) = f'(x) + \mathcal{O}(\Delta x),$$

where, for any integer n, $\mathcal{O}((\Delta x)^n)$ is read as "on the order of $(\Delta x)^n$"; that is, becomes approximately equal to a constant times $(\Delta x)^n$ as $\Delta x \to 0$. Using this notation, we can say that the forward difference approximation (1) to $f'(x)$ has error $\mathcal{O}(\Delta x)$. If Δx is small the error will be small, and it will tend to zero like a constant times Δx as $\Delta x \to 0$.

A better approximation to $f'(x)$ can be obtained by using what is called a **central difference approximation**:

$$f'(x) \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}. \tag{3}$$

This is obtained by subtracting the Taylor Series

$$f(x-\Delta x) = f(x) - f'(x)\Delta x + \frac{f''(x)}{2!}(\Delta x)^2 - \cdots + (-1)^n \frac{f^{(n)}(x)}{n!}(\Delta x)^n + \cdots$$

from the Taylor Series (2) for $f(x+\Delta x)$. This gives

$$f(x+\Delta x) - f(x-\Delta x) = 2f'(x)\Delta x + 2\frac{f'''(x)}{3!}(\Delta x)^3 + \cdots,$$

and dividing by $2\Delta x$,

$$\frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x} = f'(x) + \frac{f'''(x)}{3!}(\Delta x)^2 + \mathcal{O}((\Delta x)^4).$$

Solving this for $f'(x)$, we see that the central difference approximation has error $= \mathcal{O}((\Delta x)^2)$. As Δx gets small, this gets small much faster than the error in the forward difference approximation.

The second derivative of a function of one variable can be approximated by

$$f''(x) = \frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{(\Delta x)^2}. \tag{4}$$

This is called the **central difference approximation** to $f''(x)$. It is obtained by adding the Taylor series for $f(x+\Delta x)$ and $f(x-\Delta x)$:

$$f(x+\Delta x) + f(x-\Delta x) = 2f(x) + 2\frac{f''(x)}{2}(\Delta x)^2 + 2\frac{f^{(4)}(x)}{24}(\Delta x)^4 + \cdots.$$

Then subtracting $2f(x)$ and dividing by $(\Delta x)^2$, we have

$$\frac{f(x+\Delta x) - 2f(x) + f(x-\Delta x)}{(\Delta x)^2} = f''(x) + \frac{f^{(4)}(x)}{12}(\Delta x)^2 + \mathcal{O}((\Delta x)^4),$$

so that the central difference approximation for $f''(x)$ can be seen to have an error $= \mathcal{O}((\Delta x)^2)$.

Example 1 *Consider the function* $f(x) = ln(x)$*. We know from Calculus that* $f'(x) = \frac{1}{x}$ *and* $f''(x) = -\frac{1}{x^2}$*; therefore,* $f'(2) = 0.5$ *and* $f''(2) = -0.25$*. Using the forward difference approximation (1), with* $\Delta x = 0.1$*, the derivative is approximated by*

$$f'(2) \approx \frac{f(2+0.1) - f(2)}{0.1} = \frac{\ln(2.1) - \ln(2)}{0.1} \approx 0.4879.$$

This is not a very good approximation to $f'(2) = 0.5000$*. However, if we use* $\Delta x = 0.001$*, it becomes*

$$f'(2) \approx \frac{f(2.001) - f(2)}{0.001} \approx 0.49988$$

which is much better. Using the central difference approximation (3), with $\Delta x = 0.1$*,*

$$f'(2) \approx \frac{f(2.1) - f(1.9)}{2(0.1)} \approx 0.500417$$

which is nearly as accurate as the forward difference approximation with $\Delta x = 0.001$*. This means that our choice of approximation method can cut the number of steps needed in a solution by* $1/100$*.*

To examine the accuracy of the central difference approximation to the second derivative of $\ln(x)$ *at* $x = 2$*, using the larger value* $\Delta x = 0.1$*,*

$$f''(2) \approx \frac{\ln(2.1) - 2\ln(2) + \ln(1.9)}{(0.1)^2} \approx -0.25031.$$

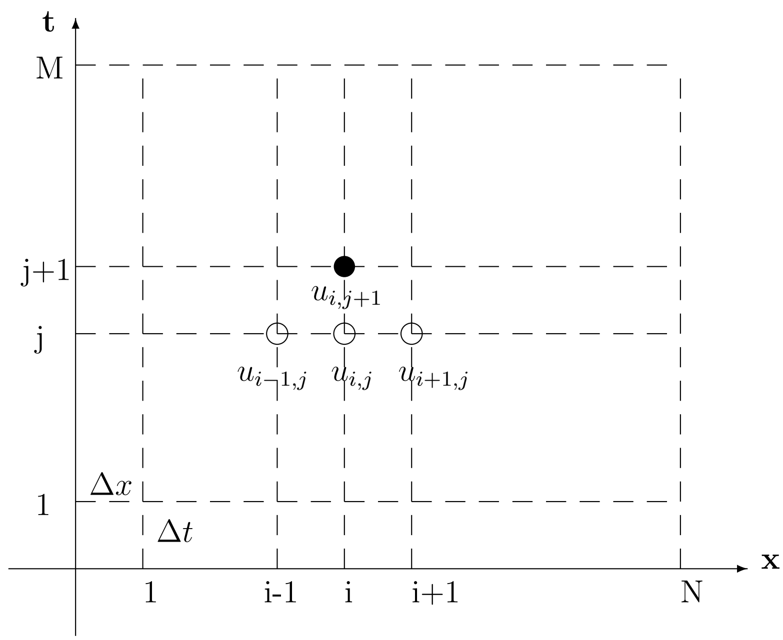
This is very close to the exact value $-\frac{1}{4}$*.*

One thing to be careful of, in approximating numerical derivatives: if you take Δx too small, round-off error can give meaningless results. Consider, for example, the computation of $\frac{f(2+\Delta x) - f(2-\Delta x)}{2\Delta x}$ when Δx is **very** small. Depending on the number of significant digits stored by the calculator or computer, if Δx is so small that $f(2 + \Delta x)$ and $f(2 - \Delta x)$ are the same to that many digits, you will end up with the meaningless value $f'(2) = 0$.

Numerical Solution of the Heat Equation

Consider the 1-dimensional heat equation $u_t = \alpha^2 u_{xx}, \;\; u(x,0) = f(x)$ for $0 < x < L$. Make a 2-dimensional grid with x on the horizontal axis and t on the vertical axis, as shown below. If you want to find an approximation to $u(x,t)$ for $0 \le x \le L$ and $0 \le t \le T$, then let $\Delta x = \frac{L}{N}$ and $\Delta t = \frac{T}{M}$ for some large integers N and M. This partitions the two axes to give a rectangular grid. The points on the grid can be indexed by $0 \le i \le N$ along the x-axis, and $0 \le j \le M$ along the vertical axis, where we can write $x_i = i \cdot \Delta x$ and $t_j = j \cdot \Delta t$. Using a forward difference approximation for $u_t(x,t)$ and a central difference approximation for $u_{xx}(x,t)$, the partial difference equation becomes

$$\frac{u(x,t+\Delta t) - u(x,t)}{\Delta t} = \alpha^2 \left(\frac{u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)}{(\Delta x)^2} \right). \tag{5}$$



The reason for NOT using a central difference approximation for u_t is that we have only the initial condition $u(x,0) = f(x)$ with which to begin our calculation. The central difference formula (3), applied to the function $u_t(x,t)$, requires knowing u at two different values of t in order to compute u_t at a third value of t.

There is an alternative method, called the **Crank-Nicolson scheme**, which assumes that $\frac{u(x,t+\Delta t)-u(x,t)}{2\Delta t}$ represents a central difference around time $t + \frac{\Delta t}{2}$, and writes the right-hand side of the equation as an average of u_{xx} at times t and $t + \Delta t$. This method is complicated by the fact that at each time step it is necessary to solve a system of linear equations for u at each value of $x_i, 0 \leq i \leq N$. However, the Crank-Nicolson method has the advantage that the error on both sides of the difference equation is $= \mathcal{O}((\Delta x)^2)$; therefore, the time step Δt can be much larger. This significantly reduces the amount of calculation required.

From equation (5), we obtain the following formula for values of u at time $t + \Delta t$ entirely in terms of values of u at the previous time t:

$$u(x, t + \Delta t) = u(x, t) + \frac{a^2 \Delta t}{(\Delta x)^2} \left(u(x + \Delta x, t) - 2u(x, t) + u(x - \Delta x, t) \right). \quad (6)$$

If we let $u_{i,j}$ denote $u(x_i, t_j) \equiv u(i \cdot \Delta x, j \cdot \Delta t)$, equation (6) can be written in the form

$$u_{i,j+1} = u_{i,j} + C(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}), \quad (7)$$

where C is the constant $\frac{a^2 \Delta t}{(\Delta x)^2}$. There is a known condition on C, namely that $C \leq \frac{1}{2}$, in order for this method to be *stable*. If $C > \frac{1}{2}$, round-off errors will be magnified as t increases, and the numerical solution will show large meaningless oscillations. What this means practically is that if you want to have a finer grid in the x-direction, the number of steps in the t-direction needed to reach a given value of T must be much larger.

Since the initial function $f(x)$ determines the value of $u(x, 0)$ for any x, the values of $u_{i,0}$ can be stored, for $i = 0, 1, \cdots N$. Before using equation (7) to find the values of $u_{i,j}$ for any integer $j > 0$, the boundary conditions on u must first be used to assign values to $u_{0,j}$ and $u_{N,j}$. Note that this means we are able to solve the heat equation with arbitrary (possibly non-constant) boundary conditions $u(0, t) = T1(t)$ and $u(L, t) = T2(t)$. The following MAPLE program solves the heat equation $u_t = \alpha^2 u_{xx}$. An arbitrary initial function $u(x, 0) = f(x)$ and boundary conditions $u(0, t) = T1(t), u(L, t) = T2(t), t > 0$ can be entered by the user.

MAPLE solution of the one-dimensional heat equation

```
#Define the variables:
alsq:= ...  # thermal diffusivity
L:= ...  # length of the rod
Tmax:= ...  # value of t at which u(x,t) is to be plotted
N:= ...  # number of intervals on x-axis
M:= ...  # number of intervals on t-axis
#Define the initial and boundary functions:
f:=x-> ...  # initial temperature for x in [0,L]
T1:=t-> ...  # temperature u(0,t), t>0
T2:=t-> ...  # temperature u(L,t), t>0
dx:=L/N: dt:=Tmax/M:
C:=alsq*dt/(dx*dx); # if C > 0.5, increase M
for i from 0 to N do u[i,0]:=f(i*dx); od:
for j from 1 to M do
u[0,j]:=T1(j*dt); u[N,j]:=T2(j*dt);
for i from 1 to N-1 do
u[i,j]:=u[i,j-1]+C*(u[i+1,j-1]-2*u[i,j-1]+u[i-1,j-1]);
od; od:
#Plot u(x,t) at time t = M*dt = Tmax:
LP:=[]:  for i from 0 to N do LP:=[op(LP),[i*dx,u[i,M]]]; od:
with(plots):
pointplot(LP,title=cat(`temp at t = `,convert(Tmax,string)));
```

Example 2 *We will solve the heat equation $u_t = 0.5u_{xx}$, with $u(x,0) = f(x) = 2x(10-x)$, holding the boundary temperatures constant at $u(0,t) = 60$ and $u(10,t) = -20$. This problem has already been solved by separation of variables, so we know that the series solution for $u(x,t)$ is*

$$u(x,t) = 60 - 8x + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi x}{10}\right) e^{-\frac{n^2\pi^2 t}{200}},$$

where

$$b_n = \frac{2}{10}\int_0^{10} \left(2x(10-x) - (60-8x)\right) \sin\left(\frac{n\pi x}{10}\right) dx.$$

Figure (1) shows the temperature in the rod at time $t = 2$. The MAPLE program was used to approximate the solution to this problem for three different values of M. The table below shows the numerical values of the temperature $u(x,2)$, for $x = 0, 2, \cdots, 10$ for these three cases. The values in Column 1 were obtained from the series solution, using enough terms in the series so that the answers were correct to the given number of decimal places. These are compared to the values obtained from the numerical solution for the three different values of Δt. In each case $\Delta x = 0.25$, but the number of steps in

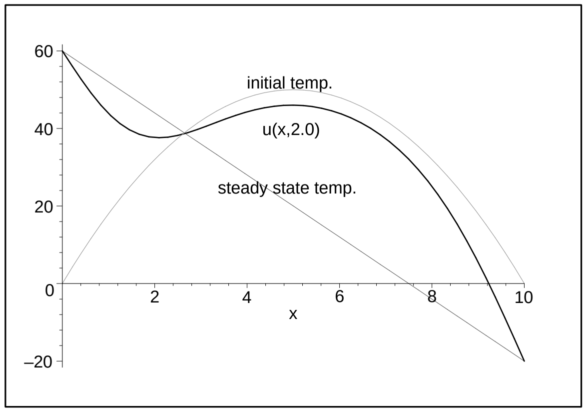


Figure 1: Temperature at times $t = 0, 2.0$, and ∞

the t-direction was 80 *in column 2,* 40 *in column 3 and* 20 *in column 4. The corresponding value of* $C = \frac{\alpha^2 \Delta t}{\Delta x}$ *increases to a value greater than* $\frac{1}{2}$ *in column 4, and it can be seen that the solution becomes unstable, causing the value of* $u(x, 2)$ *to oscillate wildly.*

Note that the numerical approximation in column 2 agrees with the series solution out to the first decimal place. This could be improved by using even larger values of M*.*

	Exact Soln.	*Numerical Solutions*		
x		$M = 80$ $C = 0.2$	$M = 40$ $C = 0.4$	$M = 20$ $C = 0.8$
0	*60*	*60*	*60*	*60*
2	*37.666*	*37.580*	*37.456*	*-611750*
4	*44.282*	*44.272*	*44.236*	*-5373*
6	*43.912*	*43.914*	*43.925*	*-4841*
8	*25.081*	*25.108*	*25.143*	*196440*
10	*-20*	*-20*	*-20*	*-20*

Numerical Solution of the Wave Equation

The wave equation can also be solved numerically, by writing the pde $u_{tt} = \beta^2 u_{xx}, \;\; u(x, 0) = f(x), \;\; u_t(x, 0) = g(x)$ for $0 < x < L$, as a difference equation

$$\frac{u(x, t + \Delta t) - 2u(x, t) + u(x, t - \Delta t)}{(\Delta t)^2} = \beta^2 \left(\frac{u(x + \Delta x, t) - 2u(x, t) + u(x - \Delta x, t)}{(\Delta x)^2} \right). \quad (8)$$

Note that the approximations on both sides of this difference equation are second order differences. This means that the error is on the order of Δx and Δt squared.

If this equation is solved for $u(x, t+\Delta t)$ in terms of values of u at previous times:

$$u(x,t+\Delta t) = 2u(x,t) - u(x,t-\Delta t)$$
$$+C\left(u(x+\Delta x,t) - 2u(x,t) + u(x-\Delta x,t)\right),$$

where $C = (\frac{\beta\Delta t}{\Delta x})^2$. If we again define a grid in (x,t)-space, with N intervals of length $\Delta x = L/N$ and M intervals in the t-direction, each of length $\Delta t = T_{max}/M$, then the difference equation can be written in the form:

$$\begin{aligned} u[i,j] &= 2u[i,j-1] - u[i,j-2] \\ &+ C\left(u[i+1,j-1] - 2u[i,j] + u[i-1,j-1]\right). \end{aligned} \tag{9}$$

The one major difference between the solution of this equation and the heat equation is the fact that it takes *two* initial conditions to specify a unique solution of the wave equation. This means that we have to use the condition $u_t(x,t) = g(x)$ to get values of $u[i,-1]$ which appear in the calculation of $u[i,1]$. We can use a $2nd$ order difference

$$\frac{u[i,1]-u[i,-1]}{2\Delta t} \approx \frac{u(x,\Delta t)-u(x,-\Delta t)}{2\Delta t} \approx g(x). \tag{10}$$

Values of $u[i,0]$ for $0 \leq i \leq N$ can be computed from the initial position function $f(x)$. Then equation (9) can be used, with equation (10) to write

$$u[i,1] = 2u[i,0] - (u[i,1] - 2\Delta t g(i*dx)) + C\left(u[i+1,0] - 2u[i,0] + u[i-1,0]\right).$$

Once the values of $u[i,1]$ are computed for all i, equation (9) can be used to find values for $j \geq 2$. At the beginning of each step in the t-direction, values of $u[0,j]$ and $u[N,j]$ must be set to correspond to the boundary conditions $u(0,t) = T_1(t), u(L,t) = T_2(t)$. Note that this means the boundary conditions can be arbitrary functions of t. This would make it possible, for example, to fasten the string at one end and move it in an arbitrary way at the other end.

The following MAPLE program can be used to solve the wave equation numerically.

MAPLE solution of the wave equation

```
#Define the variables:
bsq:= ...  # Tension/density
L:= ...  # length of the string
TMAX:= ...  # value of t at which u(x,t) is plotted
N:= ...  # number of intervals on x-axis
M:= ...  # number of intervals on t-axis
#Define the functions:
f:=x-> ...  # initial position of the string at t=0
g:=x-> ...  # initial velocity
T1:=t-> ...  # position of left end, u(0,t)
T2:=t-> ...  # position of right end, u(L,t)
dx:=L/N: dy:=TMAX/M: C:=bsq*dt*dt/(dx*dx); # need dt<dx/a
#Use f(x) to compute u(x,0):
for i from 0 to N do u[i,0]:=f(i*dx); od:
u[0,1]:=T1(dt):  u[N,1]:=T2(dt):
#Use g(x) to compute u(x,dt):
for i from 1 to N-1 do #use the initial velocity function
u[i,1]:=u[i,0]+dt*g(i*dx)+0.5*C*(u[i+1,0]-2.0*u[i,0]+u[i-1,0]);
od:
#Compute u(x,t) for t > 2:
for j from 2 to M do
u[0,j]:=T1[j*dt]; u[N,j]:=T2[j*dt];
for i from 1 to N-1 do
u[i,j]:=2*u[i,j-1]-u[i,j-2]+C*(u[i+1,j-1]-2*u[i,j-1]+u[i-1,j-1]);
od; od:
#Plot u(x,t) at time t = M*dt = TMAX:
L:=[]:  for i from 0 to N do L:=[op(L),[i*dx,u[i,M]]]; od:
with(plots):  pointplot(L,title=`temperature at t=Tmax`);
```

Practice Problems:

1. * For the function $f(x) = e^x$, approximate $f'(0)$ by the forward difference formula. Try several values of Δx. How small must Δx be to have the derivative exact to 4 decimal places? Repeat this for $f'(0)$ approximated by the central difference formula, and for $f''(0)$ approximated by the central difference formula.

2. * The temperature in a 5 meter rod satisfies the heat equation $u_t = 2.0u_{xx},\;\; u(x,0) = f(x) = 70$. Let $N = 50$ so that the approximate temperatures across the rod are computed at $x = 0, 0.1, 0.2, \cdots 5.0$. Choose non-constant temperature functions $T1$ and $T2$ for the two ends of the rod, and use the MAPLE program to plot the temperature at 5 or 6 different values of t. Explain in your own words what is happening at the two ends, and how it might produce the observed temperatures in the interior of the rod. Be sure to choose values of M that make $C < 0.5$. As a simple example, you might model an apartment where the temperature of two opposite windowed sides are affected by the position of the sun over a 24-hour period, and heat is only assumed to flow in one direction (between the two sides).

3. * Solve the wave equation $u_{tt} = 0.1u_{xx}$ with $u(x,0) = f(x) \equiv 0$ and $u_t(x,0) = g(x) \equiv 0$. The boundary functions are $T_1(t) \equiv 0$, and $T_2(t) = \sin(\pi t)$. Run the solution out to time $t = 6$. Plot the position of the string at times $t = 1, 2, \cdots, 6$.