Aero 320: Numerical Methods

Homework 7

Name: ....

Due: November 27, 2013

NOTE: All problems, unless explicitly asked to write a code, are to be done by hand (with the help of a calculator) but **you need to show all the steps**. Turn in a hard copy of your HW stapled with this as cover sheet with your name written in the above field. Submit your HW by Wednesday midnight at Room 201, Reed McDonald Building. Late submissions or failure to submit in the required format will receive no credit.

Problem 1

Application of numerical differentiation: heat equation $(10+10+10+5 = 35$ **points)**

Please review Lab 17 before you start this problem. Consider the numerical approximation of the heat equation that we derived in Lab 17, Problem 1(c). In this exercise, we will numerically solve that equation. To do this, first notice that the boundary conditions can be written in discrete form:

$$\begin{aligned} u_{i,0} &= \phi\left(x_i\right), \qquad i = 0, 1, \ldots, M, \\ u_{0,k} &= u_{M,k} = 0, \qquad k = 1, 2, \ldots, N. \end{aligned}$$

(a) Using the above information, rewrite the approximated heat equation derived in Lab 17, Problem 1(c), in matrix-vector form

$$\begin{pmatrix} u_{1,k+1} \\ u_{2,k+1} \\ \vdots \\ u_{M-1,k+1} \end{pmatrix} = \begin{pmatrix} a & b & 0 & 0 & 0 & \ldots & 0 \\ b & a & b & 0 & 0 & \ldots & 0 \\ 0 & b & a & b & 0 & \ldots & 0 \\ 0 & 0 & b & a & b & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & b & a \end{pmatrix} \begin{pmatrix} u_{1,k} \\ u_{2,k} \\ \vdots \\ u_{M-1,k} \end{pmatrix}.$$

Find a and b in the above matrix, in terms of λ (we defined λ in Lab 17).

(b) Assume $\alpha = 1$, $L = 1$, $\Delta x = 0.1$, $\Delta t = 0.001$, $T = 1$. (There was a typo in Lab 17:

Δk in page 1 should be Δt) The attached MATLAB code `HeatEqnFTCS.m` iteratively solves the FTCS equation derived in Lab 17, Problem 1(c), assuming the heat distribution at time $t = 0$ as $u(x,0) = \phi(x) = \sin\left(\frac{\pi x}{L}\right)$. The exact analytical solution is $u(x,t) = \sin\left(\frac{\pi x}{L}\right)\exp\left(-\frac{\alpha\pi^2 t}{L^2}\right)$. Run the MATLAB code that plots both analytical and exact solutions for different time instants. At any time, which location of the rod is most hot? Give intuitive reasons in support of your answer.

(c) What happens to the temperature distribution along the rod, in general, as time passes by? What will happen to the temperature distribution if we wait long time ($T \to \infty$)? Explain.

(d) For our derivative approximations in the heat equation, we had $O(\Delta t)$ error for time derivative, and $O\left((\Delta x)^2\right)$ error for spatial derivative. What kind of errors are these? Instead of *forward difference in time, central difference in space (FTCS)* approximation, if we instead use *backward difference in time, central difference in space (BTCS)* approximation, then will your order of total error change? Why/why not?

Solution

(a) In Lab 17, we showed that the FTCS approximation of the heat equation becomes

$$u_{i,k+1} \quad = \quad (1-2\lambda)\,u_{i,k} + \lambda\,(u_{i+1,k} + u_{i-1,k})\,, \qquad \text{where } \lambda = \frac{\alpha^2\Delta t}{(\Delta x)^2}.$$

Substituting $i = 1, 2, \ldots, M-1$, and utilizing the boundary conditions given in the question, we get the matrix vector equation

$$\begin{pmatrix} u_{1,k+1} \\ u_{2,k+1} \\ \vdots \\ u_{M-1,k+1} \end{pmatrix} = \begin{pmatrix} 1-2\lambda & \lambda & 0 & 0 & 0 & \ldots & 0 \\ \lambda & 1-2\lambda & \lambda & 0 & 0 & \ldots & 0 \\ 0 & \lambda & 1-2\lambda & \lambda & 0 & \ldots & 0 \\ 0 & 0 & \lambda & 1-2\lambda & \lambda & \ldots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \ldots & 0 & 0 & \lambda & 1-2\lambda \end{pmatrix} \begin{pmatrix} u_{1,k} \\ u_{2,k} \\ \vdots \\ u_{M-1,k} \end{pmatrix}.$$

Comparing the above matrix-vector equation with the one given in the question, we observe that $a = 1-2\lambda$, and $b = \lambda$.

(b) From the plot generated by running the code `HeatEqnFTCS.m`, the midpoint of the rod at $\frac{L}{2}$, seems to be the hottest location for all times $k = 0, 1, \ldots, N$. The intuitive reasoning lies in the boundary condition, which says that both ends of the rod (location $x = 0$ and $x = L$) are kept at zero temperature for all times, i.e., $u_{0,k} = u_{M,k} = 0$ for all $k = 0, 1, 2, \ldots, N$. Hence, temperature increases as we go away from each end. Farther the location is from one end, higher the temperature. The midpoint is the farthest location from both ends, hence the answer.

NOTE: If you don't like "hand waving explanations" like above, then it is possible to derive the same conclusions from precise mathematical calculations. At any fixed time t, maximum temperature occurs at x such that

$\frac{\partial}{\partial x}u(x,t) = 0 \Rightarrow \exp\left(-\frac{\alpha\pi^2 t}{L^2}\right)\left(\frac{\partial}{\partial x}\sin\left(\frac{\pi x}{L}\right)\right) = 0 \Rightarrow \frac{\pi}{L}\cos\left(\frac{\pi x}{L}\right) = 0 \Rightarrow \cos\left(\frac{\pi x}{L}\right) = 0 = \cos\left(\frac{\pi}{2}\right) \Rightarrow \frac{\pi x}{L} = \frac{\pi}{2} \Rightarrow x = \frac{L}{2}$. You may check that this corresponds to a maximum, not a minimum, by verifying that the second derivative is negative at $x = \frac{L}{2}$.

(c) From the plot, as time increases, the temperature distribution becomes more flat. In other words, at each location in the rod, the temperature at next time $k+1$, is lower than the temperature at previous time k. This means that the rod, in general, cools down. You may ask: why does the rod cool down? You may have learnt in thermodynamics that heat flow happens if there is a temperature difference. Because of our boundary condition, both ends were kept at zero temperature i.e. $u_{0,k} = u_{M,k} = 0$, but our initial condition was that temperature was maximum at the middle $u_{i,0} = \sin\left(\frac{\pi x_i}{L}\right), \quad i = 0, 1, \ldots, M$, and cooler near the ends. Hence, the rod dissipates heat (fancy way of saying "cools down") until it reaches thermal equilibrium, which is when each point of the rod achieves the same temperature as the ends, namely zero temperature.

So as $T \to \infty$, $u_{i,\infty} = 0$ for all $i = 0, 1, 2, \ldots, M$. In other words, if we wait infinite time, then the rod achieves thermal equilibrium, and temperature at each point in the rod will be equal to zero.

NOTE: If you don't like "hand waving explanations" like above, then it is possible to derive the same conclusions from precise mathematical calculations. For each location x, we get $\lim_{t\to\infty} u(x,t) = \sin\left(\frac{\pi x}{L}\right)\lim_{t\to\infty}\exp\left(-\frac{\alpha\pi^2 t}{L^2}\right) = \sin\left(\frac{\pi x}{L}\right) \times 0 = 0$.

(d) These are truncation errors. See Lab 16 solutions for details.

No, the order of total error will not change. That is because both forward difference and backward difference approximation of $\frac{\partial u}{\partial t}$ have truncation errors $O(\Delta t)$. Hence both FTCS and BTCS approximations for heat equation have total truncation error equal to $O(\Delta t) + O\left((\Delta x)^2\right)$.

Problem 2

More on numerical differentiation $(3+5+5+10+(2+2+2+1) = 30$ **points)**

Consider $f(x) = \ln x$. We want to compute $f'(x)$ at $x = 3$, using different numerical methods.

(a) Find the exact expression for $\frac{d}{dx}\ln x$. Evaluate your answer numerically at $x = 3$ to find $f'_{\text{exact}}(3)$. Keep up to 8 significant digits.

(b) Compute the *second order central difference* approximation $f'_{h_1}(3)$ with step size $h_1 = 0.4$. Keep up to 8 significant digits.

(c) Compute the *second order central difference* approximation $f'_{h_2}(3)$ with step size $h_2 = \frac{h_1}{2} = 0.2$. Keep up to 8 significant digits.

(d) Using your answers in part (b) and (c), compute the (second order) *Richardson extrapolation*

$f'_{\text{Richardson}}(3)$. Keep up to 8 significant digits.

(e) Consider your answer in part (a) to be the true/exact value of the derivative at $x = 3$. Compute the *absolute* and *relative errors* in the answers of part (b), (c) and (d). Which one seems to be the best algorithm to numerically compute the derivative?

Solution

(a) $\frac{d}{dx} \ln x = \frac{1}{x}$. Hence, $f'_{\text{exact}}(3) = \frac{1}{3} \approx 0.33333333$ (up to 8 significant digits).

(b) $f'_{h_1}(3) = \frac{\ln(3+0.4) \;-\; \ln(3-0.4)}{2 \times 0.4} = \frac{\ln(3.4) \;-\; \ln(2.6)}{0.8} \approx 0.33532998$ (up to 8 significant digits).

(c) $f'_{h_2}(3) = \frac{\ln(3+0.2) \;-\; \ln(3-0.2)}{2 \times 0.2} = \frac{\ln(3.2) \;-\; \ln(2.8)}{0.4} \approx 0.33382848$ (up to 8 significant digits).

(d) $f'_{\text{Richardson}}(3) = f'_{h_2}(3) \;+\; \frac{1}{2^2-1}\left(f'_{h_2}(3) - f'_{h_1}(3)\right) \approx 0.33332798$ (up to 8 significant digits).

(e)

Approximation	Absolute error	Relative error
part (b)	$\lvert 0.33333333 - 0.33532998 \rvert = 0.00199665$	$\frac{\lvert 0.33333333 - 0.33532998 \rvert}{\lvert 0.33333333 \rvert} = 0.0059899501$
part (c)	$\lvert 0.33333333 - 0.33382848 \rvert = 0.00049515$	$\frac{\lvert 0.33333333 - 0.33382848 \rvert}{\lvert 0.33333333 \rvert} = 0.00148545$
part (d)	$\lvert 0.33333333 - 0.33332798 \rvert = 0.00000535$	$\frac{\lvert 0.33333333 - 0.33332798 \rvert}{\lvert 0.33333333 \rvert} = 0.00001605$

Looking at the errors, we conclude that the second order Richardson extrapolation in part (d) performs better than central difference approximations in part (b) and (c).

Problem 3

Numerical integration **$(5+5+5+5+(2+2+2+2+1+6) = 35$ points)**

Let $f(x) = \frac{1}{1+x^2}$. In this exercise, we want to numerically compute $\int_{-1}^{1} f(x)\, dx$.

(a) Compute the integral using *three point Simpson's method.*

(b) Compute the integral using *five point Simpson's method.*

(c) Partition the interval $[-1, 1]$ into 2 subintervals: $[-1, 0]$ and $[0, 1]$. For this partition, compute the integral using *trapezoid method.*

(d) Using the same partition as in part (c), compute the integral using *midpoint method.*

(e) The exact value of our integral is $\frac{\pi}{2}$. Compute the *absolute* and *relative errors* in your answers in part (a), (b), (c), (d). Among the methods in (a), (b), (c), (d), which method performs better than which? Why?

Solution

(a) $I_a = \frac{1-(-1)}{6}\left(f(-1)+4f\left(\frac{-1+1}{2}\right)+f(1)\right) = \frac{1}{3}\left(f(-1)+4f(0)+f(1)\right) = \frac{1}{3}\left(\frac{1}{2}+4\times 1+\frac{1}{2}\right) = \frac{5}{3}.$

(b)

$$\begin{aligned} I_b &= \frac{1-(-1)}{12}\left(f(-1)+4f\left(\frac{3\times(-1)+1}{4}\right)+2f\left(\frac{-1+1}{2}\right)+4f\left(\frac{(-1)+3\times 1}{4}\right)+f(1)\right) \\ &= \frac{1}{6}\left(f(-1)+4f\left(-\frac{1}{2}\right)+2f(0)+4f\left(\frac{1}{2}\right)+f(1)\right) \\ &= \frac{1}{6}\left(\frac{1}{2}+4\times\frac{4}{5}+2\times 1+4\times\frac{4}{5}+\frac{1}{2}\right) = \frac{47}{30}. \end{aligned}$$

(c) $I_c = \frac{1}{2}\times 1\times\left(f(-1)+f(0)\right)+\frac{1}{2}\times 1\times\left(f(0)+f(1)\right) = \frac{1}{2}\times\left(\frac{1}{2}+1\right)+\frac{1}{2}\times\left(1+\frac{1}{2}\right) = \frac{3}{2}.$

(d) $I_d = \frac{1-(-1)}{2}\left[f\left(\frac{-1+0}{2}\right)+f\left(\frac{0+1}{2}\right)\right] = 1\times\left[\frac{4}{5}+\frac{4}{5}\right] = \frac{8}{5}.$

(e)

Approximation	Absolute error	Relative error
part (a)	$\lvert\frac{\pi}{2}-\frac{5}{3}\rvert = 0.09587034$	$\frac{\lvert\frac{\pi}{2}-\frac{5}{3}\rvert}{\lvert\frac{\pi}{2}\rvert} = 0.061032954$
part (b)	$\lvert\frac{\pi}{2}-\frac{47}{30}\rvert = 0.0041296601$	$\frac{\lvert\frac{\pi}{2}-\frac{47}{30}\rvert}{\lvert\frac{\pi}{2}\rvert} = 0.0026290233$
part (c)	$\lvert\frac{\pi}{2}-\frac{3}{2}\rvert = 0.070796327$	$\frac{\lvert\frac{\pi}{2}-\frac{3}{2}\rvert}{\lvert\frac{\pi}{2}\rvert} = 0.045070341$
part (d)	$\lvert\frac{\pi}{2}-\frac{8}{5}\rvert = 0.029203673$	$\frac{\lvert\frac{\pi}{2}-\frac{8}{5}\rvert}{\lvert\frac{\pi}{2}\rvert} = 0.018591636$

Let us use the notation "method 1 > method 2" to mean that method 1 performs better than method 2. In our case, looking at the errors, we conclude that "five point Simpson's method (part (b)) > midpoint method (part (d)) > trapezoid method (part (c)) > three point Simpson's method (part (a))".

The results may seem unusual since in this case, midpoint method outperforms trapezoid method, and both of them outperforms Simpson's three point method (which is a higher order method). The reason for such interesting behavior can be understood by looking at the geometry of the these approximations. Run the attached MATLAB file `HW7Problem3e.m` to generate the plot in Fig. 1 that explains our numerical results. See the Fig. 1 caption for details.

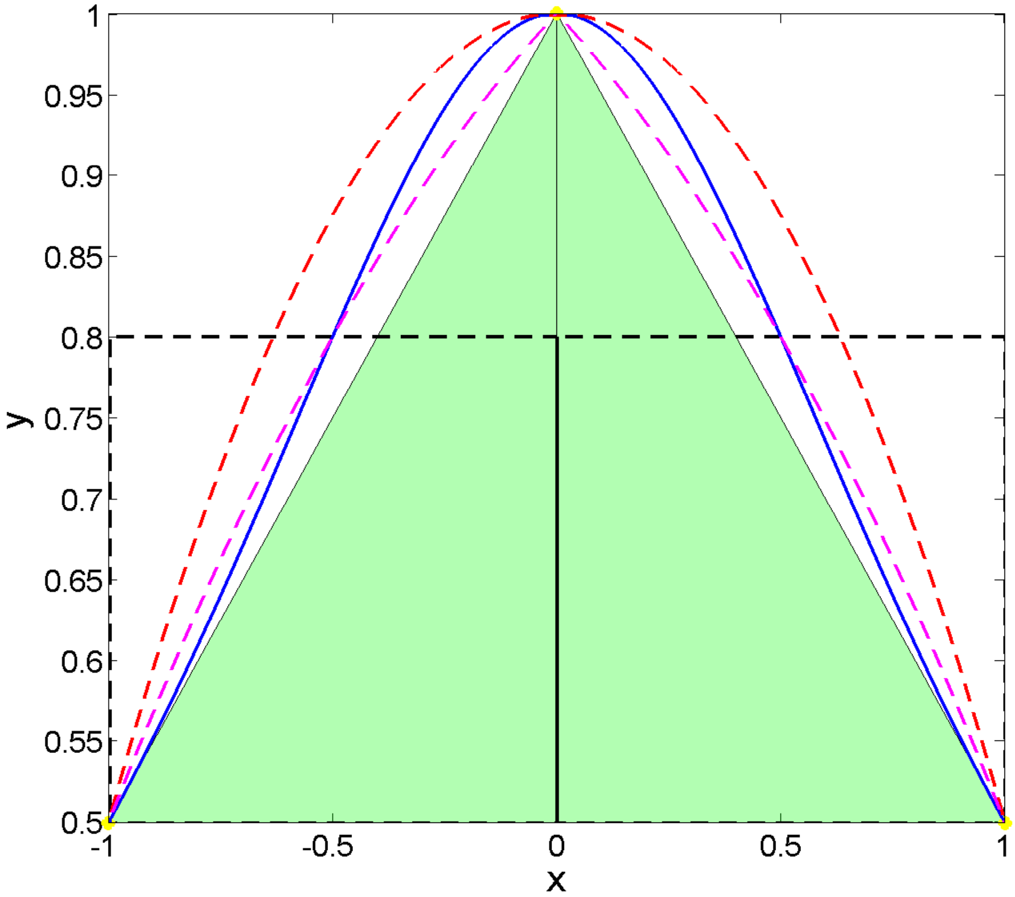

Figure 1: The function $f(x) = \dfrac{1}{1+x^2}$ is plotted in *blue solid line*. In *midpoint method*, the area under the curve $f(x)$ is approximated as the sum of the areas of two rectangles in *black dashed boundary*. Notice that due to symmetry, the amount of *overestimated area* is approximately same as the amount of *underestimated area*. Hence the midpoint method gives very good approximation of the actual area under the blue curve. However, in *trapezoid method*, the area is approximated as the sum of the areas of the two *green triangles*. As a result, we fail to account the two slices on either side: the area between the blue curve and the triangle boundaries. Hence, in this case, the trapezoid method performs worse than midpoint method. On the other hand, the *three point Simpson's method* fits a single parabola (*red dashed line*) and overestimates the area on either side. As a result, this method performs poorly in this case. However, in *five point Simpson's method*, two parabolas (shown in *dashed magenta*) are fit and we see that the amount of overestimated area is approximately compensated by the amount of underestimated area. Hence, this method performs best.