Homework Project #2
Math 365, Spring 2013

Due Wednesday February 27th

All assignments are to be done in Matlab. It is recommended that you use the Matlab “Publish” command to present your work. This assignment is out of 100 points, but you will notice that you can only get 90 points if you do not do any of the Bell and Whistle (B&W) problems. Each Bell and Whistle is worth 5 points (unless otherwise stated). The following concepts are covered in this assignment.

• Introduction to computational performance.


2. Timing a linear solve. (30 points) In this problem, you will investigate the cost of solving linear systems and compare your results to the theoretical computational cost we have discussed in class. To do this problem, you will need the following information about the laptop or PC you are working on.

• The amount of RAM in your computer. A typical value is 4 GB (gigabytes).
• The clock speed of your PC or laptop. A typical value is 2.5 GHz (gigahertz).

(a) Compute a maximum matrix size. Keeping in mind the total amount of RAM that you have available on your machine, determine the maximum memory you can comfortably use to store a single Matlab array. For example, if you have 4 gigabytes RAM, you might be able to comfortably store and work with a matrix that is 100 megabytes. Call this memory size “Mb”. Next, use the following information to compute a maximum matrix dimension $N$ that you can store in $Mb$ megabytes of memory.

• A megabyte has 1024 kilobytes
• A kilobyte is 1024 bytes
• A floating point number is 8 bytes.
• An $N \times N$ matrix contains $N^2$ floating point numbers.

Call the $N$ you compute ‘$n_{max}$’ ($N_{max}$).

(b) Estimate your machine’s flop rate. Create two random matrices $A$ and $B$ each of size $N_{max} \times N_{max}$. Using the Matlab functions tic and toc, determine how much time (seconds) it takes to compute the product $AB$. Determine the number of floating point operations (additions and multiplications) it takes to compute the $N_{max} \times N_{max}$ matrix-matrix product. Use this number to estimate the number of floating point operations per second (‘flops’) your computer can carry out. Call this flop rate ‘flops’.

Compare this number to the theoretical number obtained using your computer’s clock speed. The following information might be useful.

• A gigahertz is $10^9$ hertz.
• A hertz is equal to one clock cycle.
• A typical microprocessor (CPU) can compute 4 flops per clock cycle.

If you have a dual or quad core machine, you may also want to investigate whether Matlab is making use of multiple cores.

(c) Time the LU decomposition. Create an integer sequence of values $N$ between, say, $N = 100$ and the $N_{max}$ you found above. Loop over the entries $N_i$ of the sequence. In each pass through the loop, create a random matrix $A$ of size $N_i \times N_i$. Then, using tic and toc, time how long it takes to compute the LU decomposition of $A$. Store this time as the $i^{th}$ entry in a vector ‘lu_times’. Plot the time values you found above versus the $N_i$ in your sequence of $N$ values. On the same set of axis, plot the curve of the theoretical time estimated by the operation count...
we discussed in class. Be sure to use the flop rate ‘flops’ you computed above to get a time (in seconds) from an operation count. The two plots should be very close. Be sure to label your plots. Use the Matlab legend command to identify each curve that you get.

(d) Compute timing results from the following solution methods. For each solution method, also include a theoretical flop count curve on the graph. In each case, you may use an arbitrary (e.g. random) right hand side vector b. Make a separate set of graphs for each set of timings.

i. Time the forward solve and back solve. Using the resulting L and U that you get from the decomposition above, obtain timings for the combined forward and back solves needed to solve the system Ax = b using an LU decomposition. Use the backslash operator.

ii. Time the a solve using the backslash operator. Solve the system using x = A\b.

iii. Time a solve using inv(A). Time the cost of using the matrix inverse to solve the linear system as x = inv(A)*b.

(e) A log-log plot: Put the results of your four sets of timings (LU, forward/back solve, backslash, inv(A)) and theoretical curves above on a single loglog plot. Label each set of timings using a legend.

(f) Bells and Whistles. (5 points) Visit the website http://www.top500.org and find the list of top 10 supercomputers in the world today. Choose one of these top ten and answer the following questions.

- What size matrix can comfortably (e.g. using 10% of available memory) fit into the memory available on the supercomputer you chose?

Include any calculations you arrive at in your Matlab script, and explain any of your reasoning.

3. Temperature conductivity. (30 points) Imagine that we have a metal rod of length L (m) whose temperature at each end is held fixed at 20°C and which is being heated by a flame held at the midpoint. We can approximate the steady state temperature at equally spaced points along the rod using the following model. We divide the rod into N intervals of equal length. This partitioning of the rod gives us N + 1 equally spaced points

\[ x_j = h j, \quad h = \frac{L}{N}, \quad j = 0, 2, \ldots N \]

The steady state temperature \( T_j (K) \) at interior points \( x_j \) in the rod can be modeled as

\[ T_j = \frac{T_{j+1} + T_{j-1}}{2} + h^2 f(x_j), \quad T_j = 1, 2, \ldots, N - 1 \]

The flame \( f(x) \) is given by

\[ f(x) = \frac{S}{2\rho c_p \beta} \exp \left( - \frac{(x - L/2)^2}{\varepsilon} \right) \]

where

- \( S \) is the heating rate for the flame \((W/m^3)\)
- \( \beta \) is the thermal diffusivity of the metal \((m^2/s)\)
- \( c_p \) is the heat capacity of the metal \((J/(kg \cdot K))\)
- \( \rho \) is the density of the metal \((kg/m^3)\)
- \( \varepsilon \) is a scaling factor specifying the flame width \((m)\)

Set up a linear system to solve for the steady state temperature distribution in the rod of length \( L = 1 \). Solve the system for a thermal conductivity of \( \beta = 10^{-3} \), heat rate \( S = 1 \times 10^4 \), heat capacity of \( c_p = 200 \), density of \( \rho = 10 \) and a scaling factor \( \varepsilon = 5 \times 10^{-2} \). Compute the result at least at 200 points on the rod. Plot the results and be sure to add labels and a legend to your graph.

Hint: You might it useful to use the diag command to set up the matrix needed for this problem.
Answer the following question: How close to the center of the rod could you put your hand without getting burned? Indicate on the graph where this point is.

**Bells and Whistles.** (10 points) Suppose the rod is made of steel. What flow rate would you need to melt the rod? Do a web search to find the relevant parameters for a steel rod. Choose a fuel (propane, natural gas, methane) and determine the flow rate needed to get the heating $S$ needed to melt the rod.