---

Department of Scientific Computing
Written Preliminary Examination
Spring 2010

January 5, 2010

---

*Instructions:*

Solve 10 of the 11 questions as completely as you can. Only the first 10 questions will be counted if you do all 11 question.

All questions are weighted equally.

All parts to each question are weighted equally unless indicated in which case the percentage is listed in parentheses.

If you use any sources on the web, please list them.

The exam is due back to Cecilia Farmer no later than 4:00 pm on Monday, January 11, 2010; no exceptions allowed.

Put your Student ID on each page of your exam. When you turn in your exam, put a cover page with your name and Student ID; staple your exam to this cover page and do not put your name on each page. Following this instruction will allow the initial grading to be blind.

---

1. *Ordinary differential equations*

Consider the following ODE for the flow of current $i$ in an electrical circuit with inductance $L$, capacitance $C$, and resistance $R$.

$$L\frac{di}{dt} + Ri + \frac{q}{C} = E(t)$$

where $q$ is the charge on the capacitor, $i = dq/dt$, and $E(t) = E_0 \sin(\omega t)$, is the input voltage supply. The initial conditions are $q(t = 0) = 0$, and $i(t = 0) = 0$.

1. For $R = 0$, the equation for $q(t)$ can be solved analytically, with the solution being of the form:

   $$q(t) = a_1 \sin\left(\frac{t}{\sqrt{LC}}\right) + a_2 \sin(\omega t)$$

   Find $a_1$ and $a_2$ (10%).

2. For $L = 1$, $C = 0.25$, $E_0 = 1$, $R = 0$, and $\omega^2 = 3.5$, plot the analytical solution obtained above (all numbers are in consistent SI units). Solve the ODE using forward Euler and explicit fourth order Runge-Kutta methods (you may write your own routine or use an existing one) and plot $q(t)$ for $t = [0, 100]$, using a timestep $\Delta t = 0.1$ (40%).

3. Compute and plot the global error for the two numerical solutions above, as a function of $t$. Comment on the observed behavior, particularly as it related to the choice of $\Delta t$ (20 %).

4. Can the ODE become stiff for some values of $R$, $L$, $C$, $E_0$, or $\omega$? Compute the eigenvalues of the Jacobi matrix to help you decide. Based on your analysis, do you think L-stable methods are preferable for this problem? (30%)

2. *Monte Carlo and quadrature*

On a $N$-dimensional unit hypercube on $[0,1]^N$, define the mean square distance from the origin $\langle r^2 \rangle$ as:

$$\langle r^2 \rangle = \int_V d\mathbf{r}(\mathbf{r}\cdot\mathbf{r}) = \int_0^1 dx_1 \int_0^1 dx_2 \cdots \int_0^1 dx_N (x_1^2 + x_2^2 + \cdots + x_N^2)$$

where $\mathbf{r} = (x_1, x_2, \cdots x_N)$ is a point inside the $N$-dimensional hypercube. We want to use this integral to compare the behavior of quadrature and Monte Carlo schemes as a function of $N$.

1. Show that the integral can be evaluated analytically to $\langle r^2 \rangle = \frac{N}{3}$. (10 %)
2. Discretize each of the $N$ dimensions into $n+1$ evenly spaced grid points, so that $\Delta x_m = 1/n$ for $m = 1, \ldots, N$. Use the multidimensional analog of the simple rectangle rule

   $$\int\limits_{x_0}^{x_0+\Delta x} f(x)dx \approx f(x_0)\Delta x$$

   to obtain an approximate estimate $I_Q$ for the integral of $\langle r^2 \rangle$, and hence the error (absolute difference between the true and numerical solutions) as a function of the total number of integrand evaluations. (60%)
3. The error for a Monte Carlo integration of the same problem can be shown (you don't have to derive this) to be:

   $$\epsilon_{MC} = \sqrt{\frac{4N}{45 n_{tot}}}$$

   where $n_{tot}$ is the total number of randomly chosen N-dimensional vectors at which the integrand is evaluated. Using this and previous results, compare the efficiency (number of integrand evaluations) of the simple quadrature and simple MC scheme as a function of dimension $N$, for a few acceptable values of the tolerance $\epsilon$.

3. *Orthogonal matrices and transformations*

Let $Q$ be an $n \times n$ (real) *orthogonal* matrix. Recall that an $n \times n$ matrix $Q$ is orthogonal provided $QQ^T = Q^TQ = I$ where $I$ is the $n \times n$ identity matrix. The corresponding linear transformation is called orthogonal.

a. Use this definition to show that the columns of $Q$, denoted $\vec{q}_i$, form an orthonormal basis for $\mathbb{R}^n$.

b. What do you know about the eigenvalues of $Q$? Prove your result and explicitly give the general form of an eigenvalue of $Q$.

c. Show that an orthogonal transformation preserves the Euclidean length of a vector, and that it also preserves the angle between two vectors.

d. Sometimes we want to project one space onto another one. Suppose, for example, that we want to project vectors in $\mathbb{R}^n$ onto $S = \text{span}\{\vec{\phi}_1, \vec{\phi}_2, \dots, \vec{\phi}_p\}$ where $p < n$ and the $\vec{\phi}_i$ are orthonormal vectors in $\mathbb{R}^n$.
(i) Determine the matrix $P$ which accomplishes this projection.
(ii) Is this projection matrix $P$ always orthogonal? If not, when is it guaranteed to be orthogonal?
(iii) Explicitly give the matrix which projects vectors in $\mathbb{R}^4$ onto

$$S = \text{span}\left\{ \begin{pmatrix} -4 \\ 0 \\ 0 \\ 3 \end{pmatrix}, \begin{pmatrix} 0 \\ -1 \\ 1 \\ 0 \end{pmatrix} \right\}$$

e. A *rotation matrix* is a special type of orthogonal matrix which is very useful in areas such as computer graphics. A rotation matrix is an orthogonal matrix whose determinant is one.
(i) Show that the composition of two rotation matrices is a rotation matrix.
(ii) Explicitly give the rotation matrix that takes an image displayed on a computer screen and flips it over (from right to left) and then the axis towards us is tilted 30°. Assume that the image is displayed on a screen in the $xy$ plane and the $z-$plane is orthogonal to the screen where the $x$-plane is horizontal and the $y$ plane is vertical.

4. *Singular value decomposition*

Let $A$ be an $m \times n$ real matrix. Then the SVD guarantees the existence of the decomposition $A = U\Sigma V^T$ where $U$ is an $m \times m$ orthogonal matrix, $\Sigma$ is an $m \times n$ diagonal matrix and $V$ is an $n \times n$ orthogonal matrix.

a. Let

$$A = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$$

Compute the eigenvalues and eigenvectors of $AA^T$ and $A^TA$; use them to determine the SVD of $A$. Explain your reasoning.

b. If we have the SVD for $A$, then can we use it to determine the SVD of (i) $\alpha A$ for a nonzero constant $\alpha$, (ii) $A^T$, (iii) $A + I$ and (iv) the pseudo-inverse of $A$. If so, how is the SVD of $A$ modified to form the new SVD? If we can't modify the SVD of $A$ to determine the desired result, explain why.

c. The matrix

$$B = \begin{pmatrix} 0 & A^T \\ A & 0 \end{pmatrix}$$

is clearly a symmetric matrix and thus can be made similar to a diagonal matrix. Use the SVD of $A$ to find the matrices which diagonalize $B$. What is the rank of $B$?

d. Let $C$ be a $p \times q$ matrix and $A$ an $m \times n$ matrix. Suppose we want to find an orthonormal basis for the intersection of the null space of $A$ and the null space of $C$. Describe how a straightforward application of the SVD can be used for this; explain clearly why this gives the desired result.

e. Let

$$A = \begin{pmatrix} 1 & 3 & 4 & 0 & 8 \\ 2 & 1 & 0 & 6 & 0 \\ 3 & 4 & 2 & 1 & 1 \end{pmatrix}$$

The SVD of $A$ can be used to find the "best" rank $p$ matrix approximation to $A$. Explain what "best" means in this situation. Find it for $p = 1$ and $p = 2$ and compare with $A$ by determining the relative error.

5. *Computational Fourier analysis*

(a) The function $f(x)$ has a period of 2 and is sampled with $N = 64$ grid points on an interval of length $A = 2$. These samples are used as input to the Discrete Fourier Transform (DFT) in order to approximate the Fourier coefficients of $f$. Use the reciprocity relations in part (c) below to answer the following questions:
(i) What are the minimum and maximum frequencies that are represented by the DFT?
(ii) Do these minimum and maximum frequencies change if the same function is sampled, with $N = 64$, on an interval of length $A = 4$?

(b) The function $f(x)$ is zero for $|x| > 5$. The DFT will be used to approximate the Fourier transform of $f$ at discrete frequencies up to a maximum frequency of 100 periods per unit length. Use the reciprocity relations in part (c) below to select a grid in the spatial domain that will provide this resolution. Specify the minimum $N$ and the maximum $\Delta x$ that will do the job.

(c) The above problems require knowledge of the reciprocity relations

$$A\Omega = N \quad \text{and} \quad \Delta x \Delta \omega = 1/N$$

where $A$ and $\Omega$ are the sampling intervals in the $x$ (real) and $\omega$ (frequency) domains and $\Delta x$ and $\Delta \omega$ are the corresponding grid sizes in these domains, respectively. Derive the reciprocity relations.

Here, assume that N-point DFT is defined by

$$F_k = \sum_{n=-N/2+1}^{N/2} f_n \mathrm{e}^{2\pi i k n/N}$$

and that the sampling intervals are $[A/2, A/2]$ and $[-\Omega/2, \Omega/2]$.

(d) Compute the DFT of $f(x)$ defined by $f(x) = \cos(x)$ for $-1 \le x \le 1$ and $f(x) = 0$ otherwise. Use $N = 64$. Do not include numerical data as part of your answer; rather plot the magnitude of $F_k$ as a function of $\omega_k$ over the appropriate range of $\omega$ and include this graph. Outline the steps.

6. *Partial differential equations: finite element method*

Consider the diffusion equation

$$u_t = \alpha u_{xx}$$

with the initial and boundary conditions

$$u(x,0) = g(x), \quad u(0,t) = u_L, \quad u(1,t) = u_R.$$

The function $g(x)$ is prescribed over the interval $0 < x < 1$, and $\alpha$, $u_L$ and $u_R$ are constants and $\alpha > 0$.

(a) The backward-time difference scheme can be used to convert the above initial-boundary value problem into a two-point boundary value problem (BVP) at every time step. Carry out the details of this step and develop this BVP. (20%)

(b) Develop a complete piecewise-linear Galerkin-type finite element scheme to solve the resulting boundary value problem derived in part (a). (70%)

(c) Comment on the numerical stability of the backward-time finite element scheme developed in (a) and (b) above (10%)

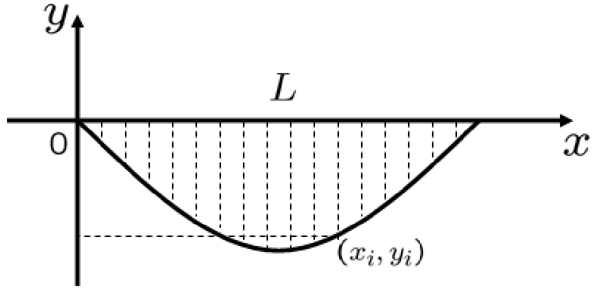


Figure 1: Discretization of catenary.

---

7. *Optimization*

The catenary is the shape of a hanging chain or cable that is only supported at its ends and acted on only by its own weight. We can use the optimization techniques to draw the catenary numerically.

As illustrated in Figure 1, a cable with uniform density is hanging at two ends: $(0, 0)$ and $(L, 0)$. And we suppose the total length of this cable is $S$ and $S > L$. We divide the interval between the two ends by $0 = x_0 < x_1 < ... < x_i < ... < x_{n+1} = L$ equally. For $x_i = \frac{i}{n+1}L$, the $y$ coordinate of the corresponding point on the catenary is denoted by $y_i$. And we approximate this catenary by a piecewise linear function with nodal points $(x_i, y_i)$.

1. Suppose the whole weight of the cable is $\rho S$, where $\rho$ is the density constant, please give the formula of the total gravity potential energy of the whole cable $P(y_1, ..., y_n)$ as a function of $y_1, ..., y_n$.

2. We can get the shape of the cable by minimizing the gravity potential energy $P(y_1, ..., y_n)$. But we still need a constraint for this problem. Please write down the equation of the constraint.

3. Please explain how you will solve this constrained optimization problem by the penalty method.

4. Please explain how you will choose the initial value of $y_i's$ and how to calculate the steepest descent direction.

5. Which optimization algorithm will be the best to use to solve the penalty problem? Why?

---

8. *Linear programming*

Consider the following linear program:

Max $x_2$, such that $x_1 + x_2 \leq 10$, $-5 \leq x_1 \leq 5$. (Note: no constraint specified for $x_2$ for this problem.)

A lot of methods can be used to figure out the answer. But here you are required to use the simplex method.

1. Write this linear program into standard form.
2. Solve this problem by simplex method.

9. *Parallel programming*

Suppose you are solving the time dependent 1D heat equation

$$\frac{\partial H}{\partial t} - k\frac{\partial^2 H}{\partial x^2} = f(x,t)$$

for the values $H(x,t)$ over a range $t_0 <= t <= t_1$ and space $x_0 <= x <= x_1$, given $k$, an inital value function $H(x,t_0)$, boundary condition functions $H(x_0,t)$ and $H(x_1,t)$, and the heat source function $f(x,t)$.

We discretize the problem by choosing $N$ equally spaced spatial nodes, and $M$ equally spaced time points. We plan to use MPI to solve this problem using $P$ nodes. Since $N = P * Q$, we plan to make each processor responsible for a set of $Q$ connected nodes, maintaining the current value of $H$ at each node, and computing the value at the next time step.

The value at the next time step is estimated using a four point stencil

$$\frac{H(i,j+1) - H(i,j)}{\Delta t} - k\frac{H(i-1,j) - 2H(i,j) + H(i+1,j)}{\Delta x^2} = f(i,j),$$

which can be rewritten as an explicit formula for the solution $H(i,j+1)$ at the new time in terms of the known values $H(i-1,j)$, $H(i,j)$ and $H(i+1,j)$ at the current time.

1. Describe, in words or in pseudocode, the communication that an arbitrary processor $P_i$ must now carry out, before it can begin the next time step. In particular, explain *who* processor $P_i$ must talk to (which other processors), and *what* information processor $P_i$ must send or receive.

2. What is different about the communication patterns for the first and last processors?

10. *Numerical differentiation*

All calculations should be in double precision. Please provide all codes with your solution.

Consider the function $f(x) = e^x$.

1. (30 points) Consider the following approximation to the derivative:

$$f'(x) \approx \frac{\alpha f(x+2h) + \beta f(x+h) + \gamma f(x)}{h}$$

   as a function of $\alpha$, $\beta$, $\gamma$. Provide the details of the derivation. Compute (theoretically) the parameters $\alpha$, $\beta$, $\gamma$ to maximize the accuracy of the derivative. Estimate the value of $h$ that maximizes the accuracy of the derivative evaluated at an arbitrary point $x$.

2. (20 points) If one writes

$$f'_{approx} = f'_{exact} + O(h^n),$$

   what is $n$?

3. (40 points) Write a computer program that approximates the $L_2$ norm of the error as a function of $h$ in the interval $x \in [0, 2]$; draw a plot of the L2 norm as a function of $h$. The plot should consider values of $h$ from $h = 10^{-8}$ to $h = 10^{-1}$. Use log scales if appropriate. Use the values of $\alpha$, $\beta$, $\gamma$ found above.

4. (10 points) Is there a minimum error found in the plot above? If yes, please explain why there is a minimum.

Hint:

$$||f||_2 = \left[ \int_{0,2} f^2 \, dx \right]^{1/2}$$

11. *Data structure*

Assume that one is running an AMR (Adaptive Multi Resolution) code with upwards of a million grids (grid sizes ranging from $2 \times 2 \times 2$ to $20 \times 20 \times 20$. More than one grid might have the same area. Each grid has a structure such as:

```
struct Grid {
  int nx, ny;   // number of points in the i and j directions
  float ***x;   // x(i,j,k), i = 0,...,nx-1, j=0,...,ny-1,
            // k=0,...,nz-1
  float ***y;   // y(i,j,k)
  int area;   // area covered by the grid (precomputed once
            // the grid is known)
  ...           // additional variables, not of relevance here
};

int nb_grids;
Grid* gridList; // list of all the grids.
```

Answer the following questions:

1. Given an area $A$, find out how many grids have an area smaller or equal than $A$. Write a function that accomplishes this task in $O(1)$ operations. Create the structure above with $n$ grids, with integer areas that are random between 0 and 100. The other elements of the structure need not be considered.

   ```
   int function nbSmallerThan(float area) ();
   ```

   Implement this function using a data structure that will return the result in $O(1)$ operations. Of course, you might have to first do some preprocessing.

2. What is the asymptotic cost of the preprocessing?

3. How often must the function be called (using big $O$ notation) to make the cost of preprocessing asymptotically negligible?