CSE Ph.D. Qualifying Exam, Fall 2010

- You should choose two areas to work on. Each area consists of four problems, and you should choose three of them.
- Show all your work and write in a readable way.
- Write your student identifier (get from proctor) on each page, number the pages, and label clearly each question that you answer (area and number) i.e., Data Analysis 2 (a).
- Avoid using computers, internet, phones, or any other assistance besides textbooks and printed notes.

Data Analysis

1. Consider a random vector $X = [X_1, \ldots, X_N]^T \in R^N$, each component of X takes p possible discrete values.

   (a) Given the joint distribution $p(X)$, what is the computational complexity to compute the marginal $p(X_i, X_j)$ for some i and j with $1 \leq i < j \leq N$?

   (b) Suppose the joint distribution $p(X)$ has the following factorized form

   $$p(X) = \frac{1}{Z}\psi_{1,2}(X_1, X_2)\psi_{2,3}(X_2, X_2)\cdots\psi_{N-1,N}(X_{N-1}, X_N)$$

   where Z is a normalizing constant, and each factor $\psi_{i,i+1}(X_i, X_{i+1})$ can be represented by a $p \times p$ matrix. Derive a fast algorithm to compute the marginal $p(X_i, X_j)$ for some i and j with $1 \leq i < j \leq N$? What is the computational complexity of your algorithm? Describe your algorithm in terms of linear algebra operations such as matrix-vector multiplication etc.

   (c) (**Optional.**) Derive a fast algorithm to compute *all* the marginals $p(X_i, X_j)$ for *all* i and j with $1 \leq i < j \leq N$?

   Note. Please derive your algorithm from scratch.

2. (a) Derive the bias-variance decomposition of the mean squared error (MSE).

   (b) Consider a random variable X that takes value 1 with probability θ and takes value 0 otherwise. Assuming we have n iid samples, derive the MLE for θ, its bias, and its variance. Under what conditions will the MSE converge to 0?

   (c) How is the Bayes classification rule defined? Write its form in the case of the 0/1 loss and binary classification. How would it change for other loss functions (loss a for false positive and loss b for false negative).

3. You are in charge of a very important application - medical diagnosis. You are predicting whether a patient has a certain disease or not based on some measurements, i.e. each data point is a patient. Your boss says to be sure that we are not mis-estimating the generalization accuracy of the classifier, you should do the following: hold out half of the patients, never to be seen in training; train your model using the other half of the patients, and report the accuracy of the model on the held-out set. Cross-validation or any other mechanism for selecting parameters can be used on the training set.

   We would like to have the best estimate of the classiier's accuracy on future patients. Is this a good approach? If not, what is wrong with it, and what is a better approach? If so, justify the approach.

4. Consider a support vector machine with the Gaussian (RBF) kernel. List its parameters. In today's current practice, which objective function(s) is/are minimized to find these parameters? Is there just one, or are there more than one? Why? Is this good? Speculate on how today's current practice could be improved upon.

Discrete Algorithms

1. In computational biology, DNA can be represented as a sequence of characters drawn from an alphabet of four letters, A, C, T, and G, representing the four nucleotides. Given two sequences S_1 and S_2 of n and m characters, respectively, describe what is meant by a local alignment. Given a similarity score of +2, a mismatch penalty of -1, and a gap score of 0, give an efficient sequential algorithm to compute the score of the best local alignment between S_1 and S_2 . What is the asymptotic complexity of your algorithm? What are the space requirements? Suppose now that you are given a multi-core processor with p cores (with $1 < p < \min(n, m)$), design and analyze a multicore algorithm for sequence similarity problem using local alignments that scales with the number of cores. Describe in detail whether or not your algorithm is cache-friendly.

2. Given an undirected, connected, sparse graph $G = (V, E)$ with $n = |V|$, $m = |E|$ and an average vertex degree (m/n) of O(1), give an algorithm to find a spanning tree of G starting from vertex $s \in V$. A spanning tree is a subset of $m = n - 1$ edges that form a tree of all of the n vertices in the graph such that no cycles (or loops) are formed.

   (a) What data structures are selected and why?

   (b) What is the complexity of this algorithm?

   (c) Describe the performance one would expect from an implementation of this algorithm on a 32-bit uniprocessor computer (e.g. your PC), assuming $n < 100,000$.

   (d) How much memory (as a function of n and m) is required?

   (e) What do you expect dominates the running time of the implementation?

   (f) Assume now that you wish to find a spanning tree on a graph with a billion vertices. Please identify strategies you could employ to solve this large problem.

3. It is often useful to partition graphs for various computational science and engineering applications. For example, given a graph $G = (V, E)$, we wish to partition the vertices into k sets such that each set contains about n/k vertices, and the total number of edges cut is minimized. Please describe two heuristics for partitioning graphs when the vertices have nodal coordinates. Describe how these approaches work for graph bisection ($k = 2$) versus multilevel partitioning (e.g., for $k = 16$). Give an example of a graph topology that works well for each of the heuristics, and an example of a topology that does not work well. Explain what is meant by coarsening a graph in multilevel partitioning, and give an example of an algorithm that could coarsen a graph without nodal coordinates.

4. In the RAM model, a balanced binary tree is often held in an array data structure of n elements where node is two children are held in locations $2i$ and $2i + 1$. Compare the time complexity of searching for a leaf in this tree using the RAM model and using the cache-oblivious model with block size B and memory size M , and $M \ll n$. If there is a more effective cache-oblivious data structure for this problem, describe in detail the layout and the new cost for performing a search.

High Performance Computing

HPC

1. **Balanced system design.** Consider the manycore processor-based machine shown in Figure 1. This machine has the following components:

   - p cores, *each* capable of a maximum floating-point performance of C flops per second. That is, the peak of the entire processor is $p \cdot C$ flops per second.
   - A shared cache of size M words.
   - An infinite-capacity main memory.
   - A single communication channel between the cache and main memory. The maximum bandwidth of this channel is β words per second. However, the width of this channel is L words, meaning data can only be transferred in packets of size exactly L words coming from *consecutive* addresses. Suppose the cost of sending such a packet is $\alpha + L/\beta$.

   Next, consider a computation for which we know only the following two characteristics. First, it performs f flops. Secondly, it transfers $g_L(M)$ words between main memory and cache; however, we only know this value when $L = 1$, i.e., we only know $g_1(M)$. Answer the following questions about this computation running on the given machine.

   (a) (25%) Let T_{comp} be the time required to perform just the flops, ignoring communication / data transfer. What is the minimum possible value for T_{comp}? What is its maximum possible value?

   (b) (25%) Let T_{comm} be the time required to do just the memory-cache transfers, independent of doing any flops. What is its minimum possible value, in terms of $g_1(M)$? What is its maximum possible value, again in terms of $g_1(M)$?

   (c) (25%) We say that the overall system, consisting of the computation *and* the machine, is *balanced* if $T_{\text{comm}} \leq T_{\text{comp}}$. Derive a sufficient condition for the system to be balanced.

   (d) (25%) Suppose we have a computation for which $g_1(M) = \frac{f}{\log M}$. A computer architect approaches you and proposes a new system in which the number of cores doubles to $\hat{p} = 2 \cdot p$. Suggest to this architect at least two ways in which he or she might maintain the system balance under such a change. That is, how might he or she adjust the other machine parameters to maintain the balance condition you derived in part (c)? Of these suggestions, which would you prefer and why? (The latter question does not have a single "correct" answer, but aims to see your thinking.)

2. **Distributed memory K-means clustering.** Suppose we wish to split a set of n input points, into k clusters (disjoint subsets). Denote the input points by $x_1, \ldots, x_n \in \mathbb{R}^d$, and associate center points, $c_1, \ldots, c_k \in \mathbb{R}^d$,

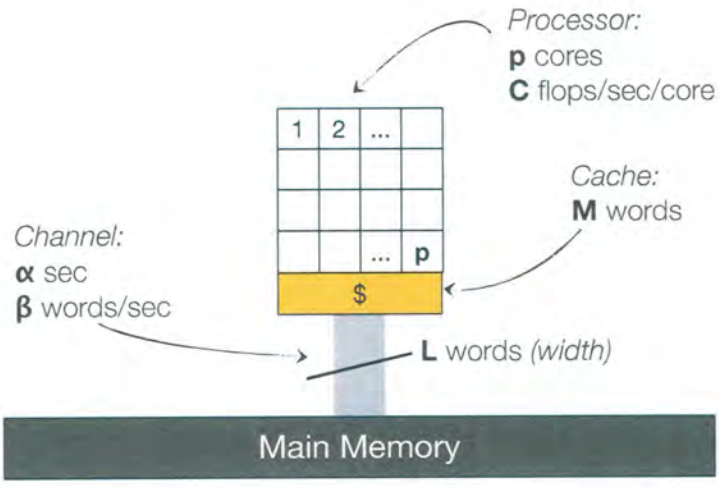

Figure 1: Manycore processor-based machine for the Q1 on balanced systems.

with the clusters. Let $D(x, y)$ be the scalar function that computes the distance between two points, x and y.

Consider the following procedure to compute a clustering.

1: Initially, generate k random centers, $c_1, \ldots, c_k$, within the bounding box of all points.
2: **repeat**
3: Assign each point x_i to the nearest cluster. Denote this nearest cluster by an integer label, L_i. That is, the point x_i is assigned to the cluster whose center is c_{L_i}.
4: For each cluster, recompute its center.
5: **until** no cluster assignments (labels) have changed.

Answer the following questions.

(a) (60%) Give an efficient *distributed memory* algorithm for this procedure. Assume a single-program multiple data (SPMD) programming model, with MPI-like communication primitives, e.g., point-to-point sends and receives; collective operations, such as reductions, all-reductions, gathers/scatters; and barriers. You may assume there are p processes and that the n points are initially evenly distributed among them, with p evenly dividing n. If you use reductions, be sure to clearly state the reduction operator(s).

(b) (30%) Analyze the time per iteration of your algorithm in the latency-bandwidth communication model, for n initial points and p processes. That is, suppose the cost of sending a message containing k bytes is $\alpha + k/\beta$, where α is the *latency* of sending a message (in units of time) and β is the *bandwidth* (in units of words per unit time). If you use any collectives operations, be sure to state your assump-

tions about their cost.

(c) (10%) Compute the speedup of your algorithm relative to the sequential procedure.

3. **Work and depth (span).** Consider the following *parallel* quicksort algorithm, written in a Cilk-like multithreaded nested-parallel shared-memory programming model.

```
1: procedure QUICKSORT(X, i, j)
2: if i < j then
3:     q ← PARTITION(X, i, j)    /* Selects a pivot value, X[q] */
4:     spawn QUICKSORT(X, i, q − 1)
5:     spawn QUICKSORT(X, q + 1, j)
6:     sync
7: end if
```

Answer the following questions.

(a) (50%) Give an efficient parallel algorithm for PARTITION. Assume the following keywords for expressing parallelism: **spawn**, **sync**, and **parallel-for** (parallel for loops, where all iterations are independent).

(b) (20%) Argue the correctness of your PARTITION algorithm.

(c) (30%) Analyze your PARTITION algorithm. Specifically, assume the multithreaded directed acyclic graph (DAG) model in which you compute the *work* and *depth* (or *critical path* or *span*), given n input elements. If your algorithm requires auxiliary storage, state its requirements. If necessary, assume the input element values are uniformly distributed.

4. **Work, depth (span), and storage.** This question is conceptual: while it does not require you to prove or calculate anything, it does require you to interpret some analysis results and try to apply them.

Consider a computation in the multithreaded directed acyclic graph (DAG) model. In particular, suppose the computation has work T_1, which matches the best sequential algorithm, and depth (or span) T_∞. Further, suppose the sequential algorithm requires storage S_1 to execute.

Now suppose you are given two schedulers for executing this computation. Both schedulers produce *time-optimal* schedules, in the sense that they *both* guarantee that the execution time T_p on p processors will be:

$$T_p \leq \frac{T_1}{p} + T_\infty \tag{1}$$

However, these schedulers differ in critical ways:

- The first scheduler, Scheduler A, has these two key properties: (i) it is a distributed scheduler, rather than a centralized one, and (ii) it requires storage S_p on p processors in the amount of $S_p = p \cdot S_1$.

- The second scheduler, Scheduler B, has these two key properties: (i) it is centralized, rather than being distributed, and (ii) it requires storage S_p on p processors in the amount of $S_p = S_1 + O(p \cdot T_\infty)$.

Answer the following questions about these two schedulers.

(a) (50%) Based on the information given, discuss the trade-offs between the two scheduling approaches. Under what conditions might one be better than the other?

(b) (50%) One key difference between the two are there different storage requirements. Does this difference tell you anything about which one might have, say, better locality? Why or why not?

Modeling and Simulation Exam Questions

1. Discrete Event Simulation

Consider a simulation of the transportation network of a major urban region such as the Atlanta metropolitan area. Sketch the design of a discrete event simulation model for the simulation by characterizing the system state, key parameters and required data, and some of the event types that you anticipate would be required. Give an example of a question researchers might pose regarding the transportation system where the discrete model is more appropriate than a continuous simulation of the same phenomenon. Explain why the discrete model is more appropriate. Cite all sources you use in coming up with the model.

2. Continuous Simulation

Consider again a simulation of the transportation network of a major metropolitan area, however, this time describe what a continuous simulation model of the network might look like. Describe the key state variables, parameters and required data, and the form of the equations that might be used. Give an example of a simulation question where the continuous model might be more appropriate to use than the discrete model, and explain why. Cite all sources you use in coming up with the model.

3. Random Number Generation

Consider the generation of random numbers following an exponential distribution with some known mean. Give three reasons why the rejection-acceptance method would *not* be the preferred approach for generating random numbers for this distribution. What is the preferred approach? Assume a program is available to generate uniformly distributed random numbers between 0 and 1.

4. Simulation Data Structures

Design a data structure that you would use to implement the input queue (including both processed and unprocessed events), output queue, and state queue for a Time Warp parallel simulator. Assuming a distributed-memory machine architecture and message-based communications, show the data structure implemented on each processor. You should assume there are multiple (possibly a large number) of logical processes mapped to each processor, and the number of events assigned to each LP may be very large. Justify your choice of data structures that you select. You should maximize the efficiency of your design for the operations that will be required.

CSE/Numerical Computing Qualifying Exam – Fall 2010

Show all your work and write in a readable way.

1. **[Basic linear algebra]**

   (a) *(2 pnts)* **[Norms]** Let $A \in \mathbb{C}^{m\times n}$. Is $\|A\| = \max_{i,j}(|A_{ij}|)$ a valid norm? Prove your result.

   (b) *(4 pnts)* **[Condition number]** Let $A \in \mathbb{R}^{n\times n}$ be given by

$$A = \begin{bmatrix} a_1 & 0 & \dots & \dots & \dots & \dots \\ -1 & a_2 & 0 & \dots & \dots & \dots \\ 0 & -1 & a_3 & 0 & \dots & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \dots & \dots & 0 & -1 & a_{n-1} & 0 \\ \dots & \dots & \dots & 0 & -1 & a_n \end{bmatrix}.$$

   Assume that $0 < \{a_i\}_{i=1}^n < 1$. Give a lower bound for the $\|\cdot\|_2$-norm condition number of A as a function of the elements in the diagonal of A. (Hint: consider Ae_1 and $A^{-1}e_1$, where $e_1 = \{1, 0, \dots, 0\}^T$.)

   (c) *(4 pnts)* **[Eigenvalues]** Let A be a real symmetric matrix. Consider the following algorithm for the maximum eigenvalue λ_1 of the matrix A (assume no multiple eigenvalues):

$$\lambda_1^{m+1} = \frac{z_m^T A z_m}{z_m^T z_m}, m \geq 0, \quad \text{with} \quad z_{m+1} = \frac{Az_m}{\|Az_m\|_\infty},$$

   given an initial guess z_0. Show that the convergence of the scheme is given by

$$|\lambda_1^{m+1} - \lambda_1| = O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2m}\right),$$

   where λ_2 is the second largest eigenvalue of A. Would you use this or the Power method to compute the maximum eigenvalue λ_1 of a real symmetric matrix?

   (d) *(2 pnts)* **[SVDs]** Let $A \in \mathbb{C}^{m\times n}$. Assume that you have black box routine `[Q,S]=eig(B)` that can compute and return the eigenvalues S and eigenvectors Q of a complex Hermitian matrix B. Describe an algorithm in terms of pseudocode to compute the singular value decomposition (SVD) of A using `eig()`.

2. [**Linear systems**]

   (a) *(2 pnts)* [**Rank defficient linear systems**] Let $K \in \mathbb{R}^{m\times n}$, with $m < n$. Given $b \in \mathbb{R}^m$ consider the linear system $Kx = b$. (a) Does it have a solution? (b) If yes, is the solution unique. (c) Describe three different algorithms that can be used to solve for x.

   (b) *(2 pnts)* [**QR**] Let $A \in \mathbb{R}^{m\times n}$ and assume that $v \neq 0$ satisfies $\|Av\|_2 = \sigma_n(A)\|v\|_2$. Let Π be a permutation such that if $\Pi v = w$, then $|w_n| = \|w\|_\infty$. Show that if $A\Pi = QR$ is the QR factorization of $A\Pi$, then $R_{nn} \leq \sqrt{n}\sigma(A)$. Thus, there exist always a permutation Π such that the QR factorization of $A\Pi$ "displays near rank deficiency.

   (c) *(2 pnts)* [**Error estimates**] Suppose you solved a linear system $Ax = b$ by a backward stable algorithm on a machine with $\epsilon_{\text{machine}} = 10^{-8}$. The relative error in the computed solution $\tilde{x}$ is $\frac{\|\tilde{x}-x\|}{\|x\|} = 10^{-3}$ for a given A and b. Now suppose you compute the solution of the same problem with $\epsilon_{\text{machine}} = 10^{-16}$. Derive an estimate for $\frac{\|\tilde{x}-x\|}{\|x\|}$.

   (d) *(4 pnts)* [**Iterative methods**] Let A be a real symmetric positive definite operator in $\mathbb{R}^n$. (1) State the damped Jacobi iterative method and specify how certain parameter(s) of the method can be chosen to guarantee convergence. (2) Given an optimal choice for this parameters, What is the convergence rate of the method?

   (e) *(2 pnts)* [**Iterative methods**] Which Krylov iterative solver would you choose for linear system $Ax = b$ for the case that A t is a (a) symmetric positive definite matrix; (b) symmetric indefinite matrix; (c) symmetric positive semidefinite matrix; and (d) an unsymmetric matrix.

3. [**Nonlinear problems and Fourier Transforms**]

   (a) *(4 pnts)* [**Newton's method**] Let $Ax + (xx^T)w = b$ be a nonlinear system for the vector $x \in \mathbb{R}^n$, given known vectors w, b and known matrix A. (a) State the Newton method for this problem (derive the Jacobian). (b) Is the Newton method for this problem guaranteed to converge? (c) If not, under what conditions we can have convergence? (d) Give one algorithm that can be used to guarantee convergences to a local solution.

   (b) *(4 pnts)* [**Fast Fourier Transform**] Describe how you would use the Fast Fourier Transform to upsample by a factor of 2 a 2π-periodic function, which was sampled

on a equispaced grid. The work should be $O(n \log n)$, where n is the number of samples.

(c) *(4 pnts)* **[Approximation]** Let $\{f_i\}_{i=1}^N$ be the *noisy* samples of an unknown function $f(x) : [0,1] \to \mathbb{R}$ at distinct and arbitrarily distributed (but sorted) points $\{x_i\}_{i=1}^N$.

(a) Describe the construction of a first-order accurate approximation scheme for $f(x)$ at some arbitrary point x. (b) Describe a third-order scheme. (c) How would you account for the presence of noise in the data?

4. **[Applications]**

(a) *(6 pnts)* **[Dynamical systems]** Consider a two-dimensional dynamical system given by

$$\begin{pmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{pmatrix} = \begin{pmatrix} -2 & 1 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}, \; 0 < t \leq 10 \quad \text{and} \quad \begin{pmatrix} x_1(0) \\ x_2(0) \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \tag{1}$$

where $\dot{z}(t)$ is defined as $\frac{dz(t)}{dt}$ for any given function $z(t)$.

i. Derive an expression for the exact solution of (1).

ii. Is this system stable or unstable?

iii. Calculate the largest discrete time step δt for both the forward and backward Euler methods for which the methods are numerically stable when applied to (Eq. 1).

(b) *(6 pnts)* **[Optimization]** Consider the following quadratic program:

$$\min_x \frac{1}{2} x^T A x - b^T x \quad \text{subject to} \quad Bx = 0, \tag{2}$$

where $x, b \in \mathbb{R}^n$, $A \in \mathbb{R}^{n \times n}$, and $B \in \mathbb{R}^{m \times n}$, with $m < n$; A, B, b are assumed to be known.

i. After introducing Lagrange multipliers p, state the first-order optimality conditions for Equation (2).

ii. State the second-order sufficient conditions for Equation (2).

iii. In order to solve for x, one has to 'solving $Kv = g$ for v, where

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

Show that K is an indefinite matrix.