

# Spring 2000 CDA 4101 Homework 8

This homework is not due. It is for studying purposes.

## Reading

Chapter 9.

## Problem 1

Consider the Omega network in the notes constructed from  $4 \times 4$  cross bars using 4-way shuffles. Describe the local routing algorithm that determines the path of each message through the crossbars.

Determine whether or not some common permutations can be passed through the network without contention, e.g., the identity, a left- or right-end-around shift, reversal, etc.

## Problem 2

Suppose you have  $p$  processors in a shared memory programming / physically shared memory / physically centralized memory environment that does not use caches. Suppose the only synchronization mechanism available is a barrier. Describe how you would implement the summation

$$s = \sum_{i=1}^p x(i)$$

How many barriers are executed? Assuming that each processor allows only one outstanding request to memory at any given time estimate the amount of time this code would take to execute. How many loads from memory are required to accomplish the task? Estimate the speedup that you would expect over a single processor (again without a cache). Suppose a cache was available on the single processor, estimate the speedup on the parallel processor without caches over the single processor with a cache.

## Problem 3

Amdahl's law is applicable to parallel processing as well as vector processing. In this case the fast and slow modes of processing are parallel ( $p$  processors) and scalar (one processor) respectively. In fact, however, there are more than two modes. Suppose you are given,  $\alpha_i$ , the fraction of work that can be done using  $i$  processors for  $i = 1, \dots, p$ . The fraction  $\alpha_1$  is the sequential fraction and  $\alpha_p$  is the fully parallel fraction. Derive a more complicated version of Amdahl's law to predict the execution time and speedup on a parallel machine given the  $\alpha_i$ .

## **Problem 4**

Describe a scenario where false sharing on a coherent cache-based shared memory machine causes performance difficulties and a method of removing or mitigating the performance degradation.

## **Problem 5**

Describe how you would implement a runtime dynamic scheduling library for a parallel loop on a machine that has lock/unlock synchronization. Assume worker tasks are executing on each processor and at the time a do loop with independent iterations is encountered the loop iteration bounds are made available to the processors as well as the code constituting the body of the loop. The runtime library must perform the dynamic scheduling of the iterations of the loop and determine when the loop processing is completed (after which one processor will continue sequential execution).

Identify potential performance bottlenecks and strategies to mitigate or eliminate them.