Parallel Computing Team Project Proposal

Kevin Cheek, Yandong Wang, Ziyan Zhou

Summary

RIT has approximately 16,000 current students and offers roughly 3,000 sections of courses per quarter. The problem of generating a comfortable final exam schedule for all students becomes fitting 3,000 final exams into 20 time slots during the finals week. The number of possible permutations, 20^3000, is astronomical. We hope to show that a parallel approach will offer a significant performance benefit over a sequential approach for this problem.

The final exam scheduler is an application that will to compute an optimal final exam schedule for RIT students. The program will use a list of course offered by RIT and cross reference it with the class registration of students. In order to find the best schedule, we will devise a ranking algorithm to score each schedule on various criteria and the resultant schedule should be close to an optimal solution.

Goals

Our first project goal is to use parallel computing to explore as many exam schedules as possible in a fixed amount of time, we will then compare the results with that of the sequential program to determine if a size up advantage exists.

Our second project goal is to use parallel computing to locate an optimal exam schedule in an extremely small data set, then compare the results with that of a sequential program. We would like to show that parallel computing can provide a speed up in performance.

Program Description

The program will consist of two general parts. The first part is a procedure for generating candidate schedules based on input courses. The second is a ranking algorithm we will define to rate each candidate schedule generated. The schedule with the best ranking will be the optimal schedule. For the candidate generation algorithm, a full permutation of all possible schedules is only feasible if the input data set is small. The total number of such permutations is (# of time slots)^(# of courses). When the number of courses is large, a pseudo genetic approach will be used for schedule generation. By using a genetic algorithm the number of schedules to examine is reduced an opti.mAl schedule can be discovered after a limited number of generations.

In the sequential brute force approach, the program will first ask the generator for a candidate schedule. The program will then rank each generated schedule according to the ranking algorithm and if the score is better than the current best that new schedule will be considered the optimal. The generator will then issue another schedule and process is repeated until no more exist. In the sequential genetic approach we will have a fixed set of schedules which will be examined and manipulated to determine the optimal schedule.

In the parallel brute force approach, the generator will operate on each node of the cluster and provide candidate schedules from a divided permutation set. In the genetic algorithm approach, each node will run a single generation of candidate schedules. The same ranking procedure is run in the sequential and parallel programs, and an optimal schedule is kept separately on each node. When the program terminates, all the local optimal schedules are compared and the best of those will be chosen.

Citations

Yiyuan Gong, Morikazu Nakamura, Shiro Tamaki. *"Parallel Genetic Algorithms on Line Topology of Heterogeneous Computing Resources"*. GECCO '05: In Proceedings of the 2005 conference on Genetic and evolutionary computation. June 2005 <http://delivery.acm.org.ezproxy.rit.edu/10.1145/1070000/1068239/p1447-gong.pdf?key1=1068239&key2=1505929221&coll=portal&dl=ACM&CFID=14598444&CFTOKEN=88370104>

Michael O.Neary, Peter Cappello. "*Advanced eager scheduling for Java-based adaptively parallel computing*". In Proceedings of the 2002 joint ACM-ISCOPE conference on Java Grande. Pages: 56 - 65 Year of Publication: 2002 ISBN:1-58113-599-8. <http://delivery.acm.org.ezproxy.rit.edu/10.1145/590000/583817/p56-neary.pdf?key1=583817&key2=5455929221&coll=portal&dl=ACM&CFID=14599103&CFTOKEN=39297937>

Jing Tang, Meng Hiot Lim, Yew Soon Ong and Meng Joo Er. "*Parallel Memetic Algorithm with Selective Local Search for Large Scale Quadratic Assignment Problems*". International Journal of Innovative Computing, Information and Control Volume 2, Number 6, December 2006 1399-1416 <http://www.ijicic.org/05-056.pdf>