COMP90025 Parallel and Multicore Computing

Project 3

Aaron Harwood and Lachlan Andrew

School of Computing and Information Systems
The University of Melbourne

2020 Semester II

Summary and Tasks

- In this project you should work individually.
- You are required to pick a problem of your choice to parallelize.
- The problem should be one that requires some amount of message passing at each step, i.e. a loosely synchronous or synchronous problem. Some examples, such as N-body Problem, have been given in the lecture slides. You can choose one of the examples if you wish.
- You are required to write an MPI±OpenMP/GPU program that can run on multiple nodes of the cluster, using multiple cores per node, for your selected problem. If you wish to use GPU nodes in the cluster then please contact us.
- Write up to 2000 words (not including pseudo-code, figures, tables or references) that outlines your problem and how you achieved parallelism/high performance. Include tables and/or charts of your own measurements that support your discussion. You should show the speedup that you achieved for a range of different numbers of cores/nodes. This can be measured either with respect to a given sequential algorithm or to your parallel algorithm deployed on 1 core.

Example problems

While some problems may be easier to parallelize than others, they also may not give you as much to write about. Please ask if you are having trouble picking a problem to work on. Here is a selection of problems that other students have worked on in the past: Viterbi Algorithm, Vertex Cover, Edge Detection, Ant colonies, Collaborative Filtering, Cracking RSA Key, Fast Fourier Transform, Global Illumination, Goldbach's Conjecture, Heat Distribution, Longest Common Subsequence, Louvain Modularity, N Queens Puzzle, N-Body Problem, Random Forrests, Sequence Alignment, Travelling Salesman.

Report

Use the ACM conference proceedings template style for your report: `https://www.acm.org/publications/proceedings-template`. Use headings:

Title and author at the top of the first page in the ACM conference format.

Abstract about 150 words. Provide a high level problem definition and motivation, and then show your main results achieved.

Keywords put relevant keywords.

Introduction about 500 words. Discuss the problem and motivation for parallelization. Include a subsection for related work, that cites existing approaches to address the parallelization of the problem.

Parallel Algorithm about 500 words. Discuss the algorithm that you have implemented. Give pseudo code. Justify your choice of parallelization platform.

Experiments about 600 words. Discuss the experiments (speedup tests) that you have conducted and show results in the form of tables and/or charts.

Conclusion about 250 words. Provide a statement overall of the effectiveness of your approach. What do we learn from your results and what could be further investigated?

References at least 10 references.

Assessment

- Project 2 is worth **20%** of your total assessment. It is individual work.
- Assessment of the report (8/20) is based on the level of details and presentation.
- Assessment of the program (12/20) is based on your parallel techniques employed/investigated.

Submission

- Submit a PDF of your report (use PDF only, no other format will be assessed) via LMS on or before **Saturday 31st October**.
- As well you will need to submit your program. You can submit this as a ZIP archive or TAR.GZ archive only, via LMS. The archive should contain your program, instructions for compiling it or a Makefile, and the slurm script(s) that you used to run your experiments. We should be able to reproduce your results by testing your program on Spartan ourselves, if we choose to.