Introduction to Parallel Computation
Exercise 1: Automatic Parallelization

Wolfgang Schreiner

November 4, 1997

Write on the Convex SPP parallel programs in Fortran *and* C solving the "All Pairs Shortest Path" problem. Use the enclosed sequential programs as the starting point and apply the parallelization features of the Convex SPP Fortran and C compilers[1].

1. Compile the sequential programs with `-O2`. Measure the execution times for matrix dimensions 256 *and* 512 and take these times as the base times of all your comparisons with the parallel programs.

2. Compile the programs with `-O3` without changes and measure the execution times for both input sizes and 1, 2, 4, 8, 16 processors. Analyze the messages generated by the compiler (`-or all`) and explain them in detail. Use the CXpa performance profiler (`-pa`) to analyze the performance of your program using

   - "Profile Select" to analyze all regions and collect all metrices,
   - "Subcomplex Selection" to choose subcomplex "parallel" for execution,
   - "2D Profile" and "3D Profile" to obtain graphical results.

   Determine which fraction of execution time was spent in which part (thread/function/loop) of the program, document your findings, and compare the C program with the Fortran program.

3. Try to improve the programs (wherever appropriate) by rewriting the sequential code and recompiling them with `-O3` *without* explicit pragmas or parallelization constructs. Proceed in the analysis as for the previous case.

[1] Both are located in `/usr/convex/bin`, take care not to call the HP compilers.

4. Try to improve the programs (wherever appropriate) by using pragmas and/or explicit parallelization constructs and/or compilation flags. recompiling them with `-O3`. Proceed in the analysis as for the previous case.

5. Write a report documenting your results including

   - the changed parts of your source code,
   - analysis of the compiler reports,
   - execution times (numerical tables and graphical diagrams),
   - speedups (numerical tables and graphical diagrams),
   - efficiencies (numerical tables and graphical diagrams),
   - analysis of the program behavior and performance,
   - any other findings, problems, comments, etc.

Deliver your preliminary report by the first deadline shown below; be prepared to present and to discuss it in class. Based on our discussions, overwork your programs and deliver the final report by the second deadline shown below. *Both reports will be used for grading.*

First Report and Presentation: November 24
Final Report and Presentation: December 9