

Extra Credit Project: N-Queens

Parallel Computing

Due on Reading day (Dec 8th), no extension will be granted.

1 N-Queens

The N-Queens problem is a classic. On an N by N chess board, one can set N Queens on the board so that no queen is able to capture any other using the standard chess queen moves: that is to say, for a queen, no other queen is in the same row, the same column and the same diagonal. A valid placement of the queens is a solution of the N-Queens problem. Feel free to consult Wikipedia's page for the 8-queens problem: https://en.wikipedia.org/wiki/Eight_queens_puzzle

The problem to solve here is not to find a solution of N-queens, but to count how many solutions exist. Usually recursive algorithm with backtracking are used.

The problem is to provide parallel implementation of this algorithm and to evaluate their speedup.

Work to accomplish:

- provide a shared-memory parallel implementation of N-Queens.
- provide a distributed-memory parallel implementation of N-Queens.
- provide a GPU parallel implementation of N-Queens.
- provide strong scaling experiments for different N

The work will be judged based on the performance achieved and the largest N solved. You can turn in only some of the implementations if you wish for partial credit.

2 Some hints

N-Queens is a classical problem for backtracking. You can find some discussion of it in the classic algorithm book "Introduction to the design and analysis of algorithms" by Anany Levitin, in the backtracking chapter.

For shared memory system, OpenMP tasking is probably the simpler approach. For distributed memory, probably master-worker is the simpler approach.