GPU-Enabled Steady-State Solution of Large Markov Models

Dr. Nick Dingle (Maths)

Joint work with Bruno Magalhaes and Dr. Will Knottenbelt (DoC, Imperial College)

Overview

- Continuous Time Markov Chains (CTMCs) are often solved to get steady-state probabilities.
- Accomplish this by solving a set of linear equations **Ax=b**, where **A** is a large and sparse matrix.
- Jacobi's method is a simple iterative technique that uses repeated matrix-vector multiplications.
- Maps nicely to CUDA programming model:
  - Each thread multiplies one row of **A** with **x** (SIMT).
  - Independence of updates to elements of **x** is key as can’t control order in which CUDA threads are executed.

Problems Overcome

- Limited amount of contiguous free memory (c. 20MB on 256MB GeForce 8600GTS) means can't store large matrices in their entirety
- Obvious solution: copy across a number of contiguous rows, do one Jacobi iteration with that portion and copy updated vector elements back.
- But main memory to GPU copies are time-consuming.
- Inspiration from asynchronous iterative solvers: perform multiple iterations with each portion of the matrix before copying vector elements back.
- Reduces number of copy operations but may require more iterations to converge.

Conclusions and Future Work

- Using a GPU we are able to compute CTMC steady-state probabilities up to 8 times faster than a CPU-only solver.
- We have implemented a parallel version that uses MPI to distribute work across a number of machines, each of which then runs the GPU-enabled solver.
- Want to investigate more efficient sparse matrix layouts, other iterative solution algorithms and extension to other CTMC measures.