

Out-of-Core Cholesky Factorization Algorithm on GPU and the Intel MIC Co-processors

Ben Chan (Chinese University of Hong Kong)

Nina Qian (Chinese University of Hong Kong)

Mentors: Ed D'Azevedo (ORNL)

Shiquan Su (UTK)

Kwai Wong (UTK)

10th July 2013

Cholesky Factorization

Similar to LU factorization

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 5 & 4 \\ 0 & 4 & 13 \end{pmatrix} \begin{matrix} \nearrow \\ \searrow \end{matrix} \begin{matrix} \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 4 & 4 \\ 0 & 0 & 9 \end{pmatrix} & L*U \\ \begin{pmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 0 & 2 & 3 \end{pmatrix} \begin{pmatrix} 1 & 1 & 0 \\ 0 & 2 & 2 \\ 0 & 0 & 3 \end{pmatrix} & U = L' \end{matrix}$$

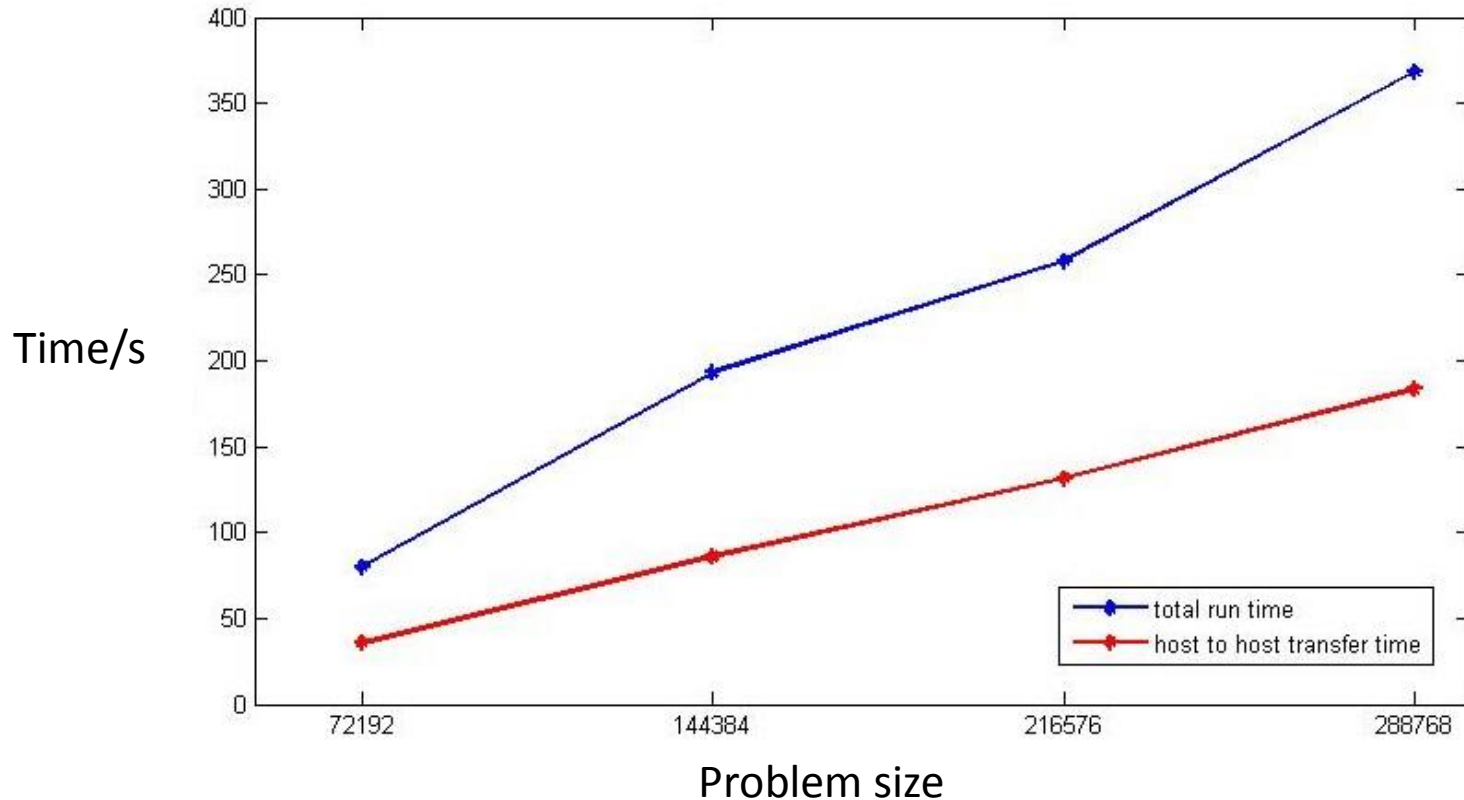
- LU works on a wider variety of matrices
- Cholesky factorization on symmetric positive-definite (SPD) matrices only

“Out-of-Core”

- Core – computing unit of the algorithm
GPU or MIC
high FLOPS, low storage
- Data stored outside of the core
- Solve larger problems quickly!

Performance Issue with GPU

4 test cases on Keeneland



~160 GFLOPS/PROC

Nvidia M2090 GPU peak performance: 665 GFLOPS

- Performance of algorithm constrained by the hardware architecture
- Quantify data transfer amount

Application: Large scale radiosity problem

- Calculate view factor between any two surfaces in space to form VF matrix (view3d program)
 - Need to separate obstruction-existing case
- Factorize VF matrix by using OOC algorithm
 - Transform VF matrix to SPD first

$$VF = \begin{pmatrix} F_{11} & \dots & F_{1n} \\ \vdots & \ddots & \vdots \\ F_{n1} & \dots & F_{nn} \end{pmatrix}$$



$$G = \begin{pmatrix} \frac{A_1}{\phi_1} - A_1 F_{11} & -A_2 F_{12} & \dots & -A_n F_{1n} \\ -A_1 F_{21} & \frac{A_2}{\phi_2} - A_2 F_{22} & \dots & -A_n F_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ -A_1 F_{n1} & -A_2 F_{n2} & \dots & \frac{A_n}{\phi_n} - A_n F_{nn} \end{pmatrix}$$

Which is a SPD matrix used to solve Boltzmann's Law

Application: Large scale radiosity problem

- Already done:
 - Serial code (Walton, 2008) → GPU-based parallel code (Hu, 2012)
- Goal:
 - Improve detecting obstruction part
 - Transfer to Intel MIC co-processor instead of GPU
 - Also need to transfer the OOC algorithm to MIC

Out-of-Core Cholesky Factorization Algorithm on GPU and the Intel MIC Co-processors

Ben Chan (Chinese University of Hong Kong)

Nina Qian (Chinese University of Hong Kong)

Mentors: Ed D'Azevedo (ORNL)

Shiquan Su (UTK)

Kwai Wong (UTK)

10th July 2013