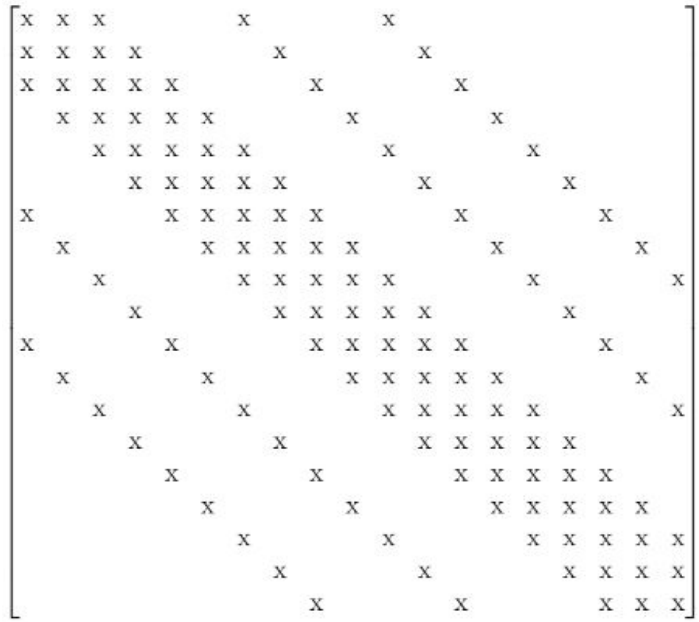
TEXAS
The University of Texas at Austin

Parallelism study of iterative linear solvers using Regent and CUDA

EE382N-20 - Parallelism and Locality

Grant Guglielmo, Ashwin Krishnan,
Manish Ravula, Nicholas Deak



Presentation Outline

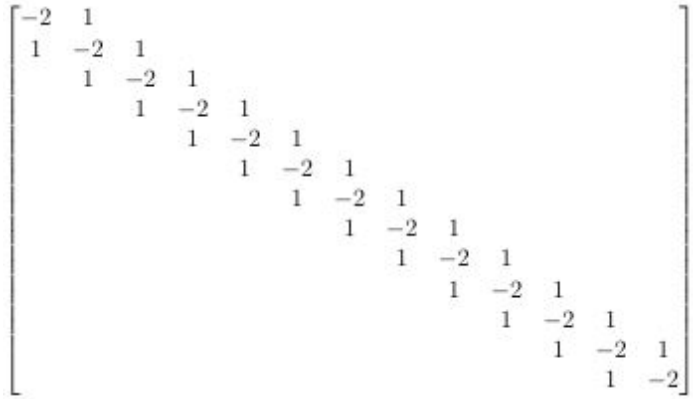
TEXAS
The University of Texas at Austin

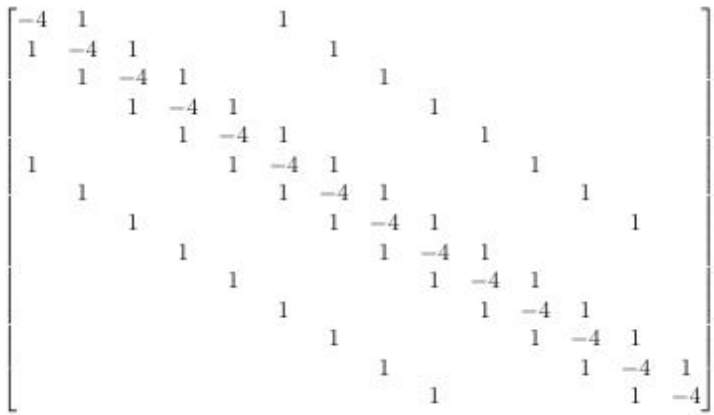
- Problem motivation and background
- CUDA implementation results
- Regent implementation results, and comparison to CPU performance
- Results discussion and conclusions

Sparse Linear Systems

TEXAS
The University of Texas at Austin

- Linear systems arise naturally when discretizing PDEs to solve numerically
- Systems are usually quite sparse in nature, and follow certain patterns
- Direct solvers are generally of complexity $O(n^3)$, which can be cost prohibitive for large systems
- Additionally, direct solve storage requirements also become an issue, as factors of a sparse matrix are generally non-sparse
- Iterative solvers are crucial to handling these types of problems

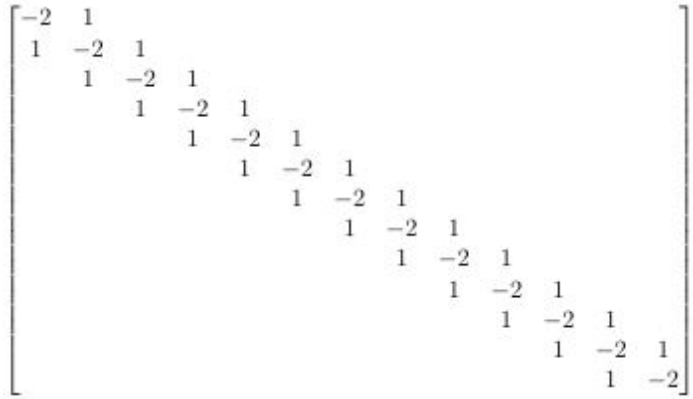


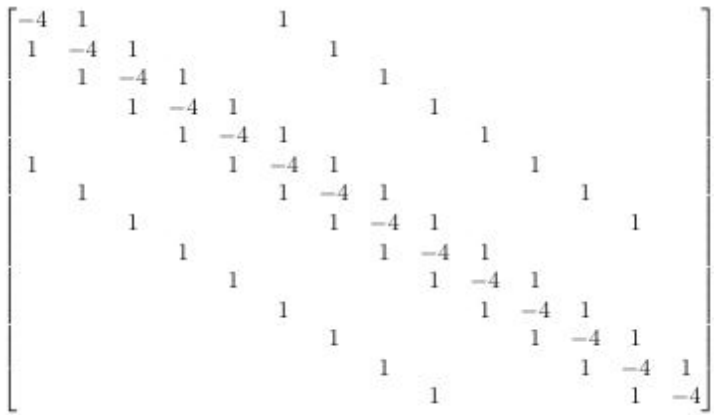


Sparse Linear Systems

TEXAS
The University of Texas at Austin

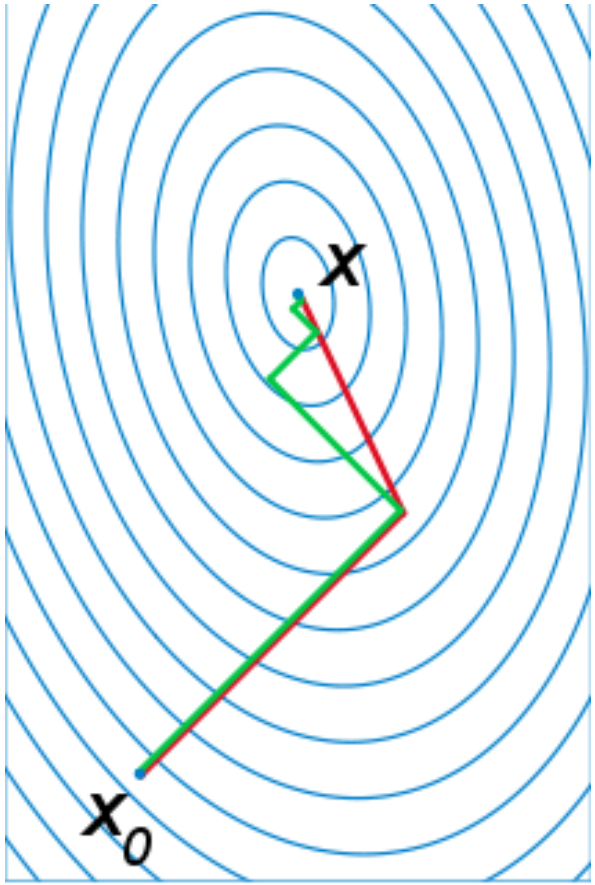
- Linear systems arise naturally when discretizing PDEs to solve numerically
- Systems are usually quite sparse in nature, and follow certain patterns
- Direct solvers are generally of complexity $O(n^3)$, which can be cost prohibitive for large systems
- Additionally, direct solve storage requirements also become an issue, as factors of a sparse matrix are generally non-sparse
- **Iterative solvers are crucial to handling these types of problems**





Conjugate Gradient Method

- Common iterative method used for solving sparse linear systems
- Designed to work on SPD matrices, but can be expanded to work on more general cases
- Can be thought of as solving a minimization problem $q(x) = \frac{1}{2} \cdot Ax - x \cdot b$
- search directions and an optimal step size are chosen at each iteration, until $||Ax - b||_2$ falls below some threshold



CUDA Results

Regent Implementation

Conclusions