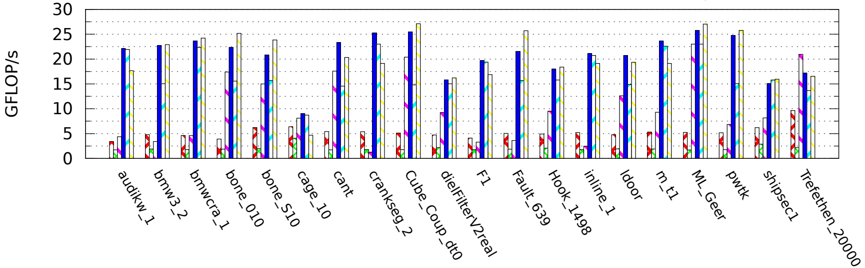
Hybrid Multi-Elimination ILU Preconditioners on GPUs

May 9, 2014

Numerical Linear Algebra on GPUs

- outstanding GFLOP rates for compute-intensive operations
- high memory bandwidth for aligned memory access
- poor performance for random data access

GPUs work great for dense LA - but what about sparse?



Numerical Linear Algebra on GPUs

- outstanding GFLOP rates for compute-intensive operations
- high memory bandwidth for aligned memory access
- poor performance for random data access

GPUs work great for dense LA - but what about sparse?

- iterative solvers for sparse systems usually memory-bound
- algorithms run at fraction of peak performance
- idea: use dense routines for sparse LA?
  - no way to target the full-size problem
  - solve small dense problem as preconditioner

Multi-Elimination Idea

- sparse linear system $Ax = b$
- use independent set ordering for permutations and decompose A

$$PAP^T \equiv \begin{pmatrix} D & F \\ E & C \end{pmatrix} = \begin{pmatrix} I & 0 \\ ED^{-1} & I \end{pmatrix} \times \begin{pmatrix} D & F \\ 0 & \hat{A} \end{pmatrix}$$

with D diagonal or easy to invert and

$$\hat{A} = C - ED^{-1}F$$

Y. Saad. Ilum: A multi-elimination ilu preconditioner for general sparse matrices. SIAM J. Sci. Comput, 17:830–847, 1999.

Multi-Elimination Idea

- sparse linear system $Ax = b$
- use independent set ordering for permutations and decompose A

$$PAP^T \equiv \begin{pmatrix} D & F \\ E & C \end{pmatrix} = \begin{pmatrix} I & 0 \\ ED^{-1} & I \end{pmatrix} \times \begin{pmatrix} D & F \\ 0 & \hat{A} \end{pmatrix}$$

with D diagonal or easy to invert and

$$\hat{A} = C - ED^{-1}F$$

- $\hat{A}$ has lower dimension but fill-in
- do it again? *use threshold to control fill-in*
- recursive application generates a sequence

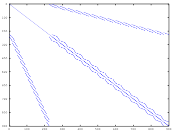
Y. Saad. Ilum: A multi-elimination ilu preconditioner for general sparse matrices. SIAM J. Sci. Comput, 17:830–847, 1999.

Multi-Elimination Idea

GR_30_30

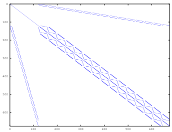
level 4

$n = 900$, $nnz = 7744$



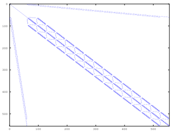
level 3

$n = 675$, $nnz = 9809$



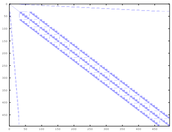
level 2

$n = 555$, $nnz = 10989$



level 1

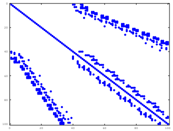
$n = 495$, $nnz = 14585$



NOS4

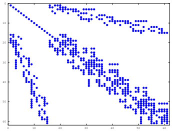
level 4

$n = 100$, $nnz = 594$



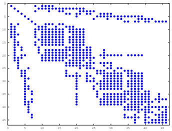
level 3

$n = 61$, $nnz = 599$



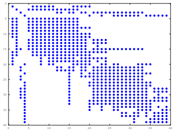
level 2

$n = 64$, $nnz = 676$



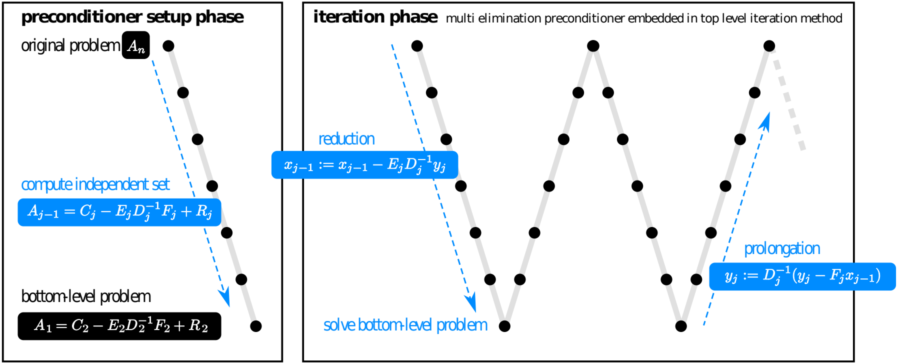
level 1

$n = 39$, $nnz = 709$

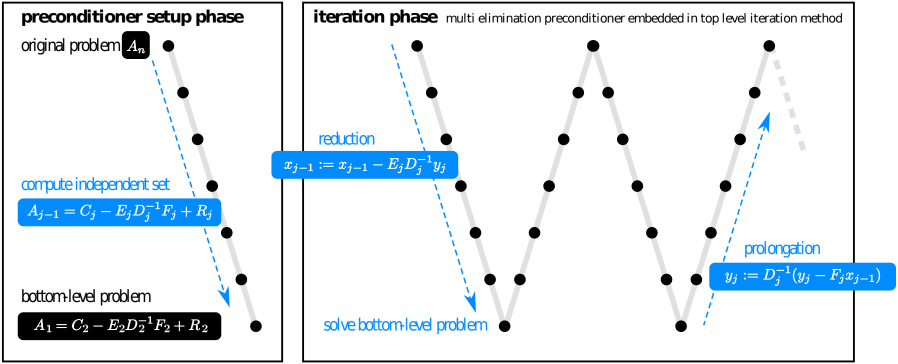


UFMC; `http://www.cise.ufl.edu/research/sparse/matrices/`

Multi-Elimination Preconditioning

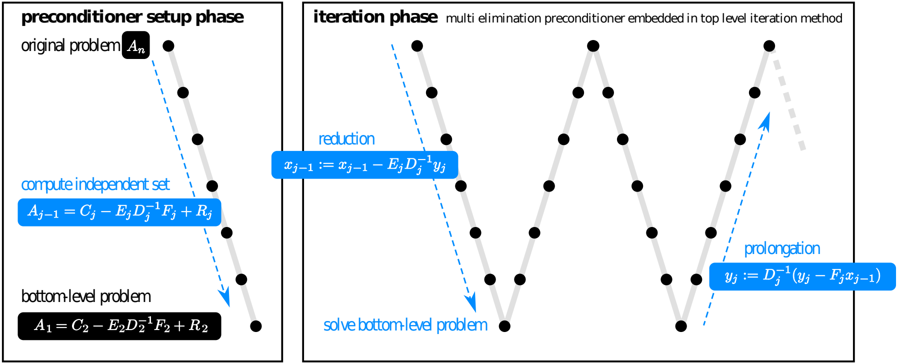


Bottom-level Solver



- compute LU factorization in preconditioner setup phase
- apply forward/backward triangular solves in iteration phase

Mixed Precision Concept

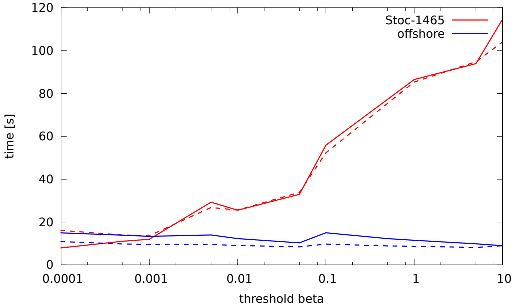


- concept is an approximation with accuracy depending on threshold
- leverage superior single-precision performance of GPUs

Threshold controlling the fill-in

- trade-off between fill-in and accuracy loss
- relate to the matrix entries
- self-adapting sequence-depth (require $n < 12,000$)

$$\tau = \beta \cdot \frac{\sum_{a_{ij} \neq 0} |a_{ij}|}{nnz}$$



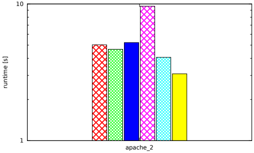
Multi-Elimination Preconditioned CG

		MCGS-GMRES		*double LU solve*		*single LU solve*	
matrix	l	# iters	time [s]	# iters	time [s]	# iters	time [s]
APACHE_2	14	293	7.61	292	3.65	293	3.04
ECOLOGY_2	14	780	8.74	779	16.39	779	10.98
G2_CIRCUIT	7	359	10.03	358	3.62	359	2.49
G3_CIRCUIT	11	511	5.84	511	5.32	512	5.42
LAPLACE_2D1M	13	338	4.37	338	3.52	338	3.41
OFFSHORE	2	1313	431.20	1323	15.02	1310	9.59
STOCF-1465	4	4399	825.22	4375	55.91	4395	52.06
THERMAL2	12	915	14.57	915	17.87	916	13.93

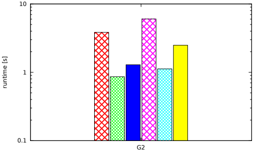
- single precision triangular solve has minor impact on CG iterations
- faster execution beneficial for performance

Hybrid Multi-Elimination Performance

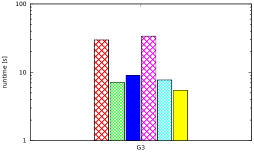
apache_2



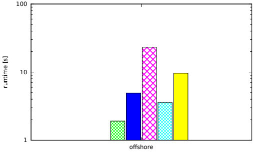
G2_circuit



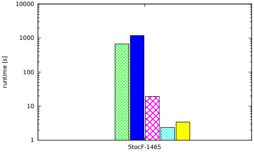
G3_circuit



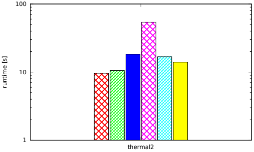
offshore



StocF-1465



thermal_2



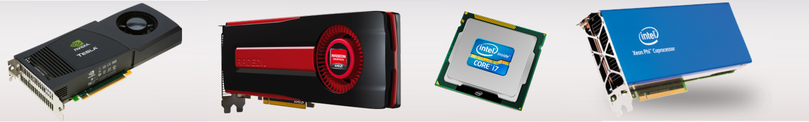
- competitive
- suitable for circuit simulation?

plain CG
Jacobi CG
multi-colored Gauss-Seidel CG
ILU0 CG
multi-colored ILU0 CG
mixed precision multi-elimination CG

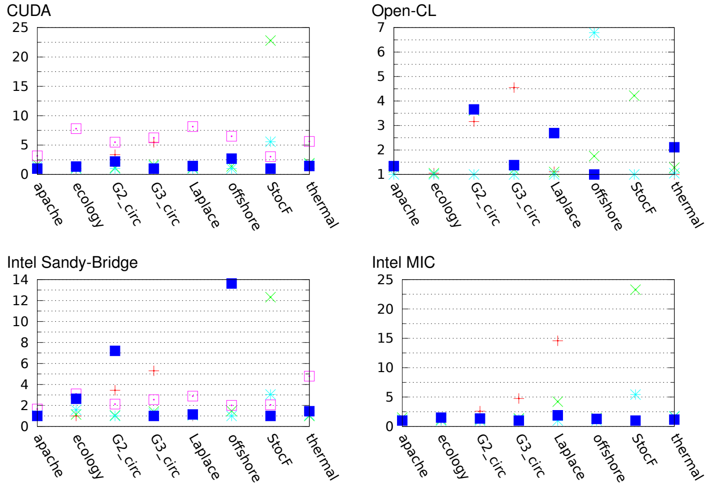
Multi-Elimination only for GPUs?

- cross-platform suitability of high interest
- non-streaming technology?

Acronym	System	Performance Peak	Memory	Bandwidth
ISB	2 × Intel Xeon E5-2670 (Sandy Bridge)	333 GFlop/s	65 GB	51 GB/s
K40	NVIDIA Tesla K40c	1,682 GFlop/s	12 GB	288 GB/s
AMD	AMD Radeon HD 7970 (Tahiti)	947 GFlop/s	3 GB	264 GB/s
MIC	Intel Xeon Phi 7110P	1,238 GFlop/s	16 GB	352 GB/s



Cross-Platform Performance



time normalized to fastest

Summary & Outlook

- multi-elimination on GPUs
- direct bottom level solver
- mixed precision concept
- competitive to established methods
- cross-platform performance

- how to choose threshold?
- non-linear problems?
- approximate (asynchronous) factorizations?