LBL Updates

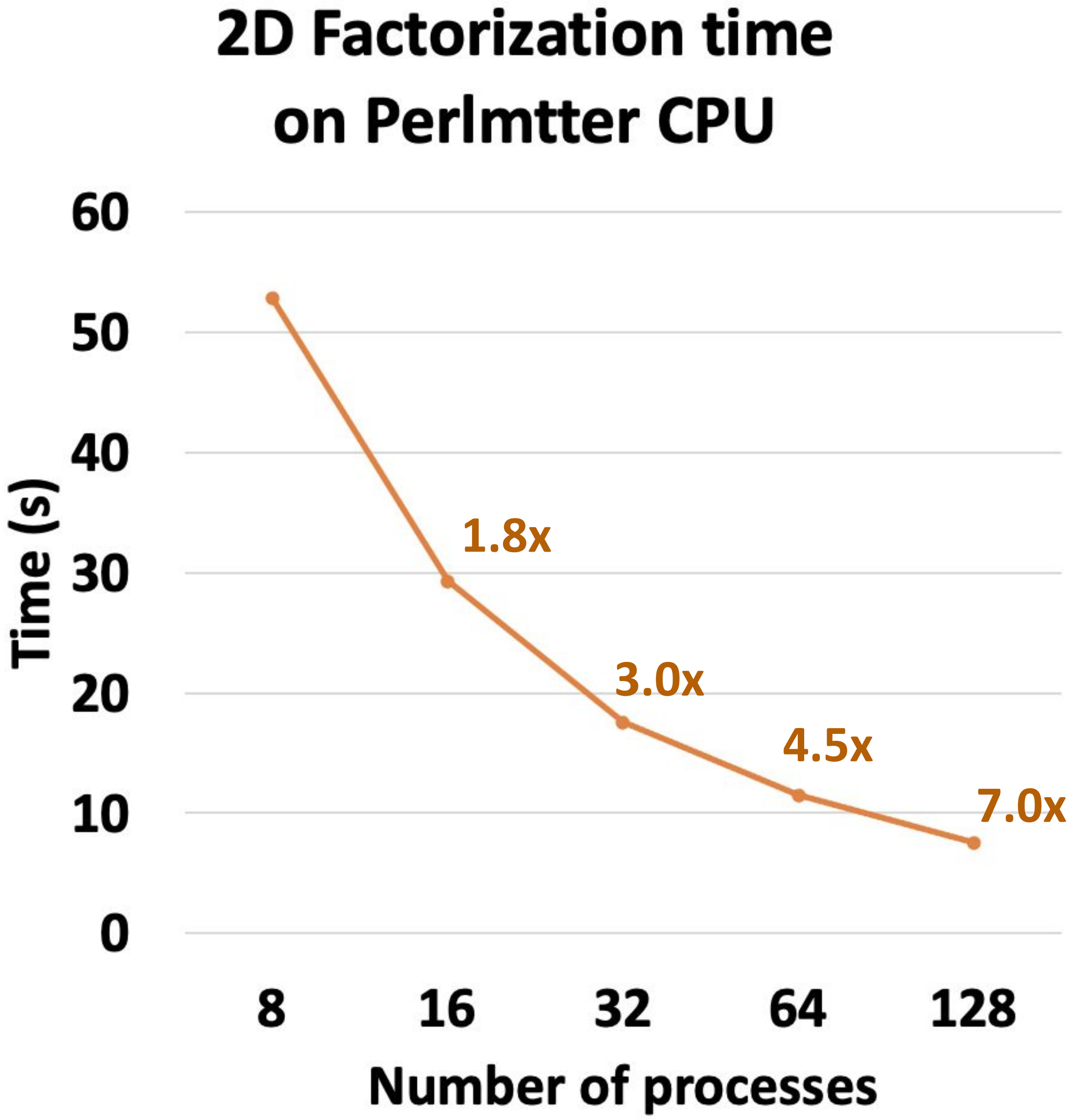
November, 2022

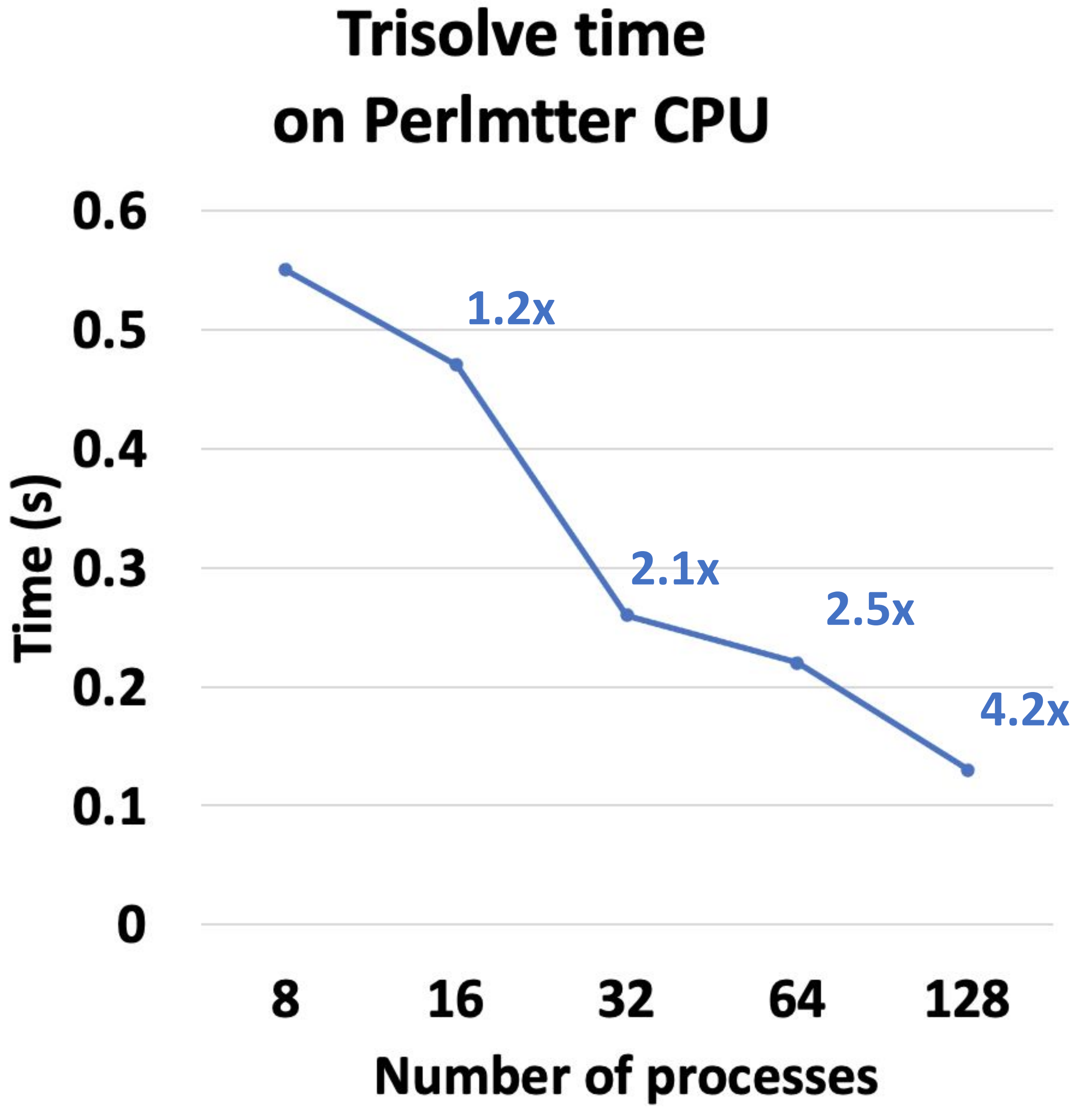
Topics

- Scaling Experiments on Perlmutter CPU
- One-sided Communication for Solvers
- M3D Benchmarking
- Batch 1D Toroidal Solves (Hans)
- Q&A

superLU with M3DC1 128K matrix Scaling Experiments on Perlmutter CPU

Matrix size min_mn 1,781,784
Nonzeros in L 1,920,066,272
Nonzeros in U 1,919,417,190
nonzeros in L+U 3837701678





Matrix size min_mn 1,781,784
Nonzeros in L 1,920,066,272
Nonzeros in U 1,919,417,190
nonzeros in L+U 3837701678

One-sided Communication for Solvers

- Network is noisy, nodes are quirky.

Speedup Onesided vs. Twosided solve						
	1node	2nodes	4nodes	8nodes	16nodes	32nodes
1core/node		1.00	1.02	1.17	1.39	0.95
2cores/node	0.94	0.98	1.28	1.38	0.95	0.99
4cores/node	1.15	1.27	1.34	0.93	0.98	0.94
8cores/node	1.36	1.41	0.95	0.97	1.00	
16cores/node	1.47	0.94	0.95	0.96		
32cores/node	0.91	1.04	0.97			
64cores/node	0.98	0.93				
128cores/node	0.95					

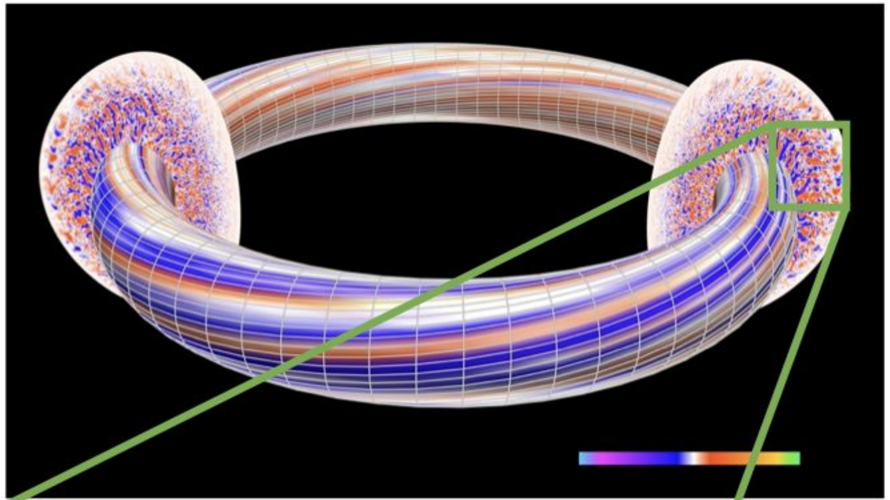
M3D Benchmarking

- Performance trend mismatch between M3DC1 and standalone superLU
- Benchmarking the solve part in M3DC1 via PETSc interface

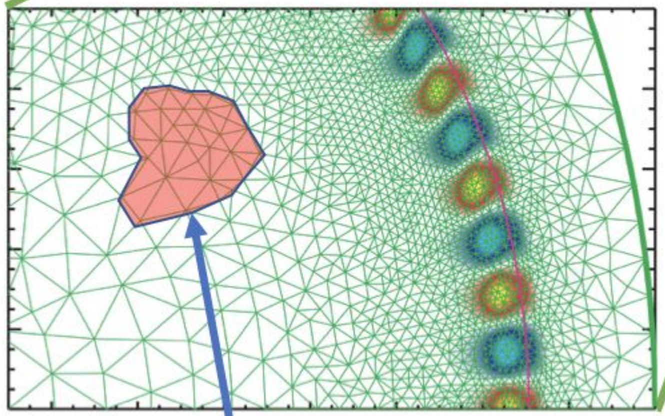
Cori Haswell	Factorization			Trisolve		
processes/plane	64	128	256	64	128	256
standalone superLU (one plane)	13.5s	13.3s	12.5s	0.19s	0.12s	0.10s
M3DC1 (time/count) 2 planes reported from PETSc	7.59s	6.89s	9.24s	0.09s	0.09s	0.10s

Batch 1D Toroidal Solves for C1 (Hans Johansen, hjohansen@lbl.gov)

Domain partitioning / system assumptions?



1. Assume each poloidal plane has parallel decompositions consisting of (connected) subset of FEM nodes
2. 1D solve in toroidal direction has fragmented data:
   - 2x2 block tridiagonal periodic (Jardin write-up)
   - Data is distributed in subsets of toroidal slices
   - Each "line solve" has different non-trivial matrix entries (metrics, dx, velocity, etc.?)



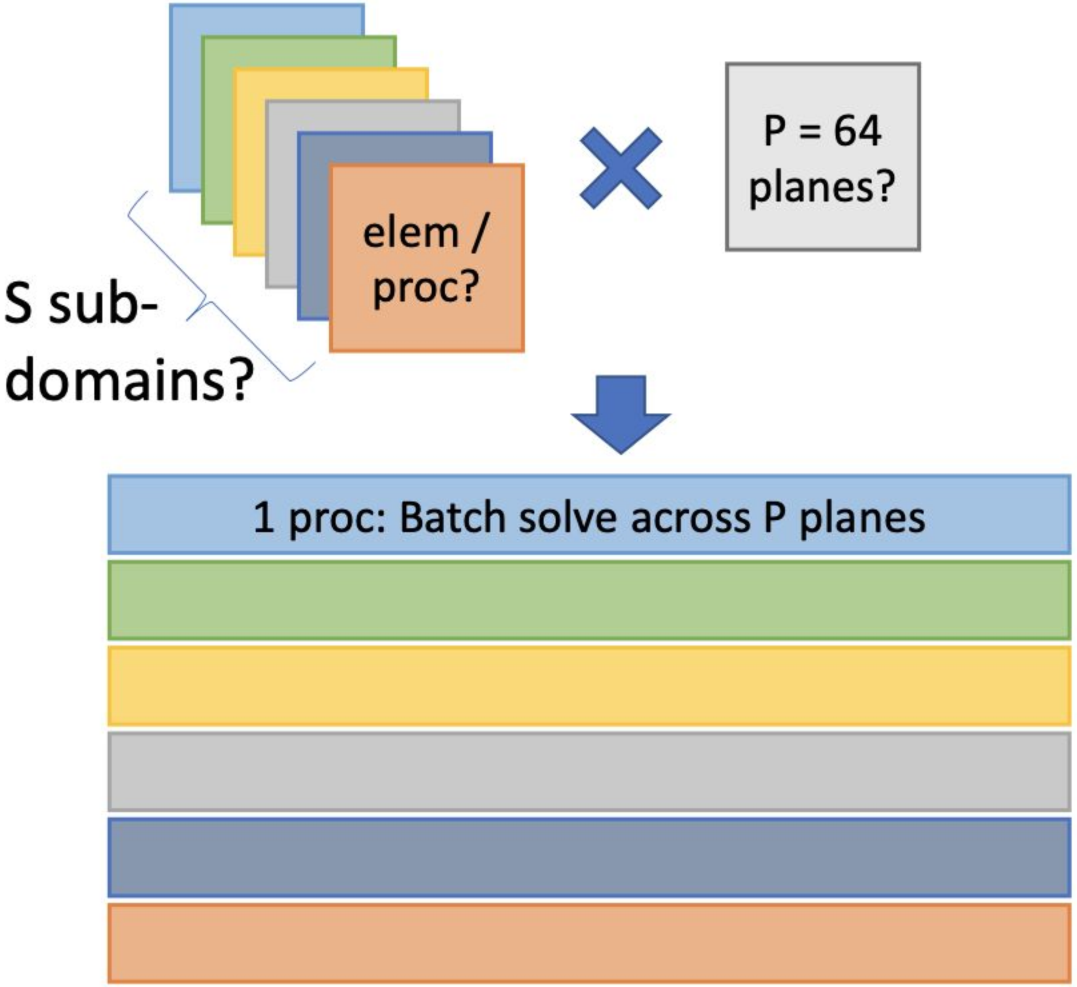
1 toroidal plane subdomain for node parallel partition?

$$\Phi^{n+1} = \Phi^{n} - \delta t\, V\left[\theta \frac{\partial \Phi^{n+1}}{\partial x} + (1-\theta)\frac{\partial \Phi^{n}}{\partial x}\right] + \delta t\, \alpha \left[\theta \frac{\partial^2 \Phi^{n+1}}{\partial x^2} + (1-\theta)\frac{\partial^2 \Phi^{n}}{\partial x^2}\right] - \delta t\, \varepsilon \left[\theta \frac{\partial^4 \Phi^{n+1}}{\partial x^4} + (1-\theta)\frac{\partial^4 \Phi^{n}}{\partial x^4}\right]$$

$$\left[\mathbf{M} + \delta t \theta \left[V\mathbf{N} + \alpha \mathbf{P} + \varepsilon \mathbf{Q}\right]\right] \bullet \mathbf{Y}^{n+1} = \left[\mathbf{M} - \delta t (1-\theta)\left[V\mathbf{N} + \alpha \mathbf{P} + \varepsilon \mathbf{Q}\right]\right] \bullet \mathbf{Y}^{n}$$

Approach / assumptions

1. Batch, block-tridiagonal solves are best solved in parallel
2. Consolidating data will reduce communication during solve
3. “Neighbor” comms are better than all-reduce or all-to-all
4. Load balancing to distribute all solves / comms, no idle procs

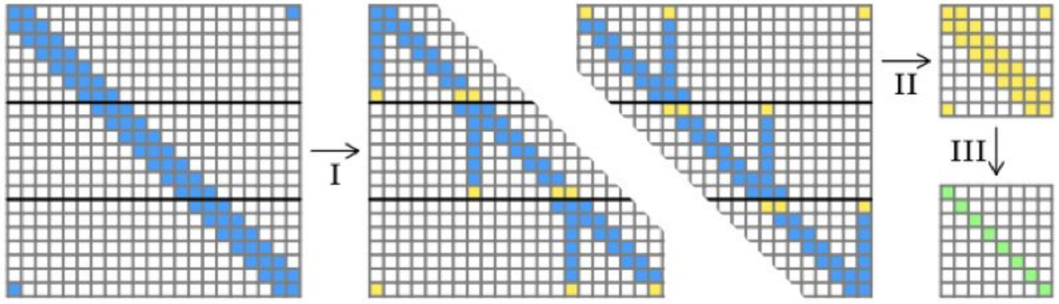


Combine two approaches

- Batch (each system is different) block tridiagonal solves (pivoting?):

Example of Problem Class Block is a system with $n_r = 1$, $N = 8$, $\hat{N} = 4$, and $n = 2$

$$AX = \left(\begin{array}{cc|cc|cc|cc} 13 & 15 & 29 & 31 & & & & \\ 14 & 16 & 30 & 32 & & & & \\ \hline 1 & 3 & 17 & 19 & 33 & 35 & & \\ 2 & 4 & 18 & 20 & 34 & 36 & & \\ \hline & & 5 & 7 & 21 & 23 & 37 & 39 \\ & & 6 & 8 & 22 & 24 & 38 & 40 \\ \hline & & & & 9 & 11 & 25 & 27 \\ & & & & 10 & 12 & 26 & 28 \end{array}\right) \begin{pmatrix} x_{0,0} \\ x_{1,0} \\ x_{2,0} \\ x_{3,0} \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \end{pmatrix} = D.$$



- Rank / system "consolidation" to remove communication in solve