Structure-Aware Linear Solvers for Matrix-Free High-Order Finite-Element Simulations on GPUs

Exact fast diagonalization where the mesh is a tensor product, device-resident p-multigrid where it is not

Gourab Panigrahi | **Phani Motamarri** (advisor) | Department of Computational and Data Sciences, Indian Institute of Science, Bangalore
pgourab@iisc.ac.in | SC26 Research Posters

DFT-FE code | Website | LinkedIn

1. The problem: one system, solved every step

Implicit time stepping, pressure projection and electronic-structure (DFT) codes solve the same finite-element (FE) system at every step:

$$\mathbf{Au} = \mathbf{f}, \qquad \mathbf{A} = \mathbf{K}_{3D} + \alpha\,\mathbf{M}_{3D} \qquad (\alpha = 0 \text{ Poisson}, \alpha > 0 \text{ Helmholtz})$$

operator action **Au** → preconditioner $\mathbf{M}^{-1}r$ → global `Allreduce`

one PCG iteration, repeated 10^2 to 10^4 times

- Iteration count grows with FE order p and mesh grading
- Every iteration waits on a latency-bound global reduction

Goal: remove iterations, and with them the global reductions.

2. Approach: the mesh chooses the solver

Tensor-product nodes and a constant coefficient?

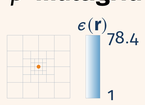
yes → Part I: fast diagonalization

uniform or graded tensor-product mesh

exact direct solve, **one apply**

138–4570× faster than converged Chebyshev–Jacobi PCG

no → Part II: p-multigrid



atom-refined octree, or variable coefficient

iterative, **p-robust**

5–7 iterations, 63–95× fewer than point preconditioners

Both matrix-free, fully on the GPU, chosen at setup.

Setting and acknowledgements

- **Polaris** A100 (ALCF), up to 128 GPUs, and **Frontier** MI250X (OLCF), up to 1024 GCDs
- up to 1.25×10^7 DoFs, $p = 2$ to 7, uniform, graded and atom-refined meshes
- Jacobi and Chebyshev–Jacobi PCG at DFT-FE tolerances, deal.II host p-multigrid

Supported by the *PMRF* (Ministry of Education, India), the *Google India Research Award 2023*, *ANRF ARG MATRICS* and *NSM*. **Computing:** *ALCF* (Polaris) and *OLCF* (Frontier), U.S. DOE Office of Science. **AI use:** parts of the poster text were drafted with Claude (Anthropic) [1] and verified by the authors, who are accountable for all content. [1] Anthropic, *Claude*, large language model, 2026, `anthropic.com/claude`.

3. Part I. Fast diagonalization: one exact apply

On a tensor-product mesh the operator is a sum of Kronecker products of *assembled* 1D stiffness and mass matrices:

$$\mathbf{A} = \mathbf{K}_x\otimes\mathbf{M}_y\otimes\mathbf{M}_z + \mathbf{M}_x\otimes\mathbf{K}_y\otimes\mathbf{M}_z + \mathbf{M}_x\otimes\mathbf{M}_y\otimes\mathbf{K}_z + \alpha\,\mathbf{M}_x\otimes\mathbf{M}_y\otimes\mathbf{M}_z$$

Three small 1D eigenproblems $\mathbf{K}_d\mathbf{S}_d = \mathbf{M}_d\mathbf{S}_d\mathbf{\Lambda}_d$, solved once, diagonalize every term:

$$\mathbf{A}^{-1} = \mathbf{S}\,\mathbf{D}^{-1}\mathbf{S}^T, \quad \mathbf{S} = \mathbf{S}_x\otimes\mathbf{S}_y\otimes\mathbf{S}_z, \quad d_{ijk} = \lambda_i^x + \lambda_j^y + \lambda_k^z + \alpha$$

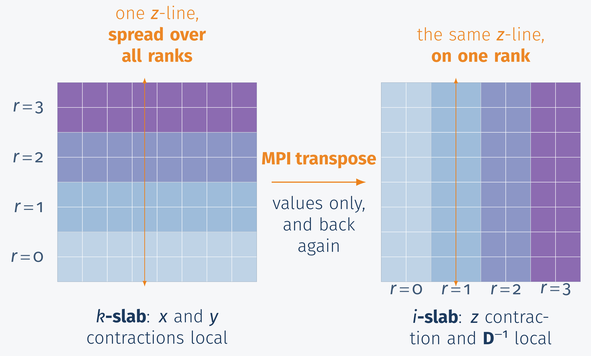
f → $\mathbf{S}_x^T$, $\mathbf{S}_y^T$, $\mathbf{S}_z^T$ **3 GEMMs** → $\div\, d_{ijk}$ **diagonal** → $\mathbf{S}_z$, $\mathbf{S}_y$, $\mathbf{S}_x$ **3 GEMMs** → **u**

0 iterations	**0** reductions	$O(N^{4/3})$ FLOPs per apply	**exact** up to round-off

Cost does not depend on the condition number of **A**. For fully periodic Poisson, setting $1/d_{000} = 0$ applies the pseudoinverse and keeps the Kronecker form.

4. Distributed FD: transpose the field, not the mesh

A contraction along z needs **whole z-lines** on one rank, so the field changes slab layout between contractions.

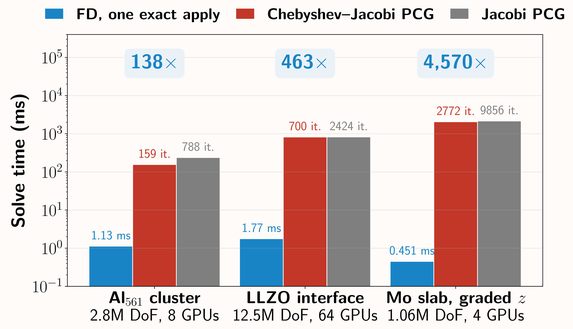


Colours are MPI ranks. **Four collectives** per apply, index maps built **once at setup**, $O(N/P)$ memory per rank.

GPU and MPI optimizations

1.18–1.28× CUDA graphs replay 12–16 launches as one	**37 → 1.31 ms** all-to-all, padded on node, variable between nodes	**0.86–0.97×** fused kernel vs cuBLAS, measured, not adopted

5. One exact apply against a converged PCG solve



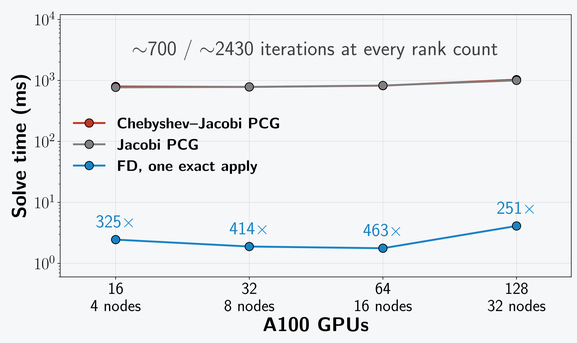
FE7 on Polaris A100s, PCG converged at DFT-FE's tolerances. FD residual 1.4×10^{-14} (Al_{561}) and about 10^{-10} (mixed boundaries). **The gain grows with the conditioning that slows CG, while FD's cost stays the same.**

Against the strongest baselines we have

138–4570× vs **Chebyshev–Jacobi PCG** 3 systems, Polaris	**14–140×** vs **our CJ-pMG solver** (Part II) uniform Dirichlet, Frontier

Where the mesh allows it, one exact apply is one to two orders of magnitude cheaper than the best iterative solve.

6. Strong scaling on a production interface

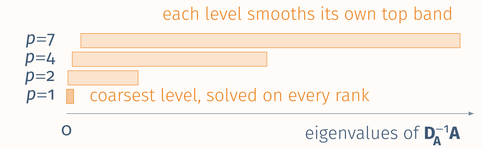


$Li_{272}La_{48}Zr_{32}O_{192}$ solid-electrolyte interface, 12.5M DoFs, FE7, Polaris. PCG iterations stay flat, so its time does not fall.

251–463× faster than Chebyshev–Jacobi PCG, 4 to 32 nodes	**22.9 → 2.8 MB** memory per rank, halving with every doubling	✓ Frontier matches all iterations, memory and residuals

7. Part II. Chebyshev–Jacobi p-multigrid

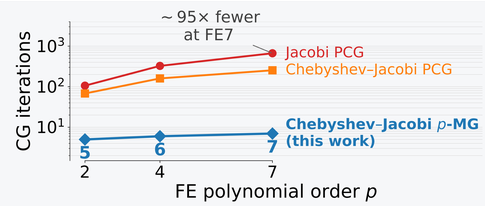
Keep what makes FD fast, **tensor-product kernels and no global reductions**, and give up only exactness.



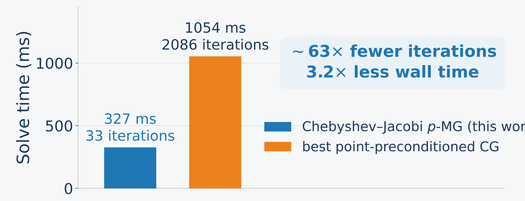
The smoother is a Chebyshev polynomial in $\mathbf{D}_\mathbf{A}^{-1}\mathbf{A}$: operator applies only, **no inner products**. Coarsening in p moves the missed modes into the next level's band.

$7 \to 4 \to 2 \to 1$ same cells, partition and ghosts at every level	**1** `Allreduce` 3 scalars per iteration, V-cycle on the GPU

8. p-robust, and fast where no direct solver exists



Al_{147} atom-refined octree with hanging nodes, $N \approx 6.1 \times 10^5$ at FE7, A100s. deal.II host p-multigrid takes 9 iterations at FE7.



Trp-cage protein, 304 atoms in implicit solvent, 11.4M DoFs, FE7, ladder $7 \to 4 \to 1$, 8 A100s. On Frontier it strong-scales to **1024 MI250X GCDs** at 5.5–5.9 ms per iteration.

9. Takeaways and limits

- **Tensor-product mesh:** one exact apply, no iterations, **14–140×** faster than our best iterative solve
- **Any other mesh:** **5–7** p-robust iterations with reduction-free smoothing
- **Limits:** FD needs a tensor-product mesh and is communication-bound below about 10^5 DoFs per rank. On solvation the coarsest level is too large for a dense solve, so the 3.2× time gain trails the 63× iteration drop
- **Next:** ghost exchange overlapped with the smoother, block solves inside the SCF cycle, FD as a filter preconditioner