Matrix-free algorithms for Kohn-Sham DFT calculations on distributed CPU architectures

Gourab Panigrahi, Phani Motamarri (PI)

MATRIX Lab, Department of Computational and Data Sciences
Indian Institute of Science, Bengaluru

FINITE-ELEMENT DISCRETIZED MATRIX VECTOR MULTIPLICATION

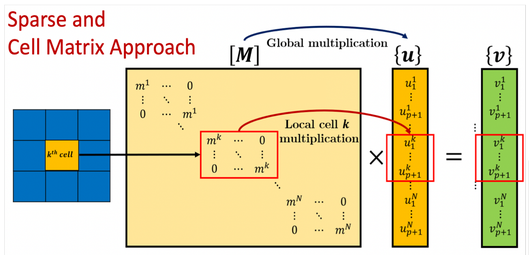

Figure 1. Sparse and Cell-Matrix Approach

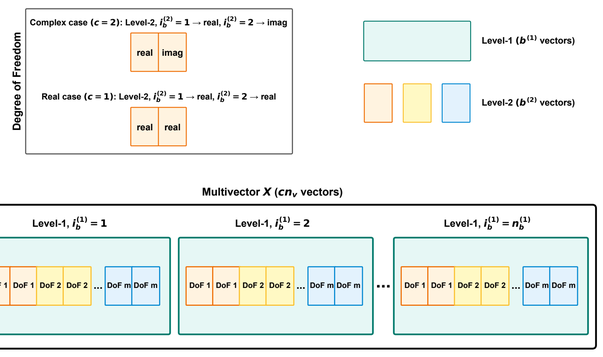

Figure 2. Multilevel Batched Layout

Avoid global assembly of matrices and evaluate matrix-vector products at FE cell level.

- **Cell-Matrix Approach:**
  - *Local dense matrix multiplications*, *High arithmetic intensity* (straightforward strided batched GEMM calls by vendor optimized libraries)
  - *Storage* and *access cost* of cell-matrix (every cell) and cell-level vectors (**u** and **v**)
- **Matrix-Free Approach:**
  - *Reduced floating point ops* and *lesser memory footprint*
  - Efficient than cell-matrix approach for single vector
- **Challenge for multivectors:** Matrix-Free has *low arithmetic intensity* due to sequence of small tensor contractions, hence *non-trivial* to extend to multivectors. (Current state-of-the-art deals with single/small number of vectors only)

Goal: Propose and implement a computationally efficient and scalable matrix-free algorithm to compute FE discretized matrix-multivector products arising in large scale eigenvalue problems with application to quantum modeling of materials using Kohn-Sham Density Functional Theory (DFT) on multi-node CPU architectures.

KOHN-SHAM DFT EIGENVALUE PROBLEM

Kohn-Sham DFT Equation (All-electron GGA)

$$\mathcal{H}\psi_j(\mathbf{x}) = \left(-\frac{1}{2}\nabla^2 + \mathcal{V}_L(\rho) + \mathcal{V}_G(\rho, \nabla\rho, \boldsymbol{R})\right)\psi_j(\mathbf{x}) = \lambda_j\psi_j(\mathbf{x}), \qquad \rho(\mathbf{x}) = \sum_j |\psi_j(\mathbf{x})|^2 \quad \forall \mathbf{x} \in \Omega$$

Introducing the finite-element (FE) discretization of the above PDE using $\psi_j^h(\mathbf{x}_q) = \sum_I N_I(\mathbf{x}_q) U_{Ij}^h$, we have

$$\mathbf{AU} := (\mathbf{T} + \mathbf{L} + \mathbf{G})\mathbf{U} = \mathbf{MU\Lambda}$$

where the matrices **T**, **L** and **M** are the FE stiffness, weighted mass and mass matrices respectively.

$$T^{IJ} = \int \frac{1}{2}\nabla N_I(\mathbf{x}) \cdot \nabla N_J(\mathbf{x}) d\mathbf{x} \qquad L^{IJ} = \int \mathcal{V}_L(\rho) N_I(\mathbf{x}) N_J(\mathbf{x}) d\mathbf{x}$$
$$G^{IJ} = \int \mathcal{V}_G(\rho, \nabla\rho) \cdot (\nabla N_I(\mathbf{x}) N_J(\mathbf{x}) + N_I(\mathbf{x}) \nabla N_J(\mathbf{x})) d\mathbf{x} \qquad M^{IJ} = \int N_I(\mathbf{x}) N_J(\mathbf{x}) d\mathbf{x}$$

Kohn-Sham DFT Equation (Pseudopotential GGA)

$$\mathcal{H}\psi_j(\mathbf{x}) = \left(-\frac{1}{2}\nabla^2 + \mathcal{V}_L(\rho) + \mathcal{V}_G(\rho, \nabla\rho, \boldsymbol{R})\right)\psi_j(\mathbf{x}) + \int \mathcal{V}_{nl}(\mathbf{x}, \mathbf{y}, \boldsymbol{R})\psi_j(\mathbf{y}) d\mathbf{y} = \lambda_j\psi_j(\mathbf{x})$$

Finite-element discretized equations:

$$\mathbf{AU} := (\mathbf{T} + \mathbf{L} + \mathbf{G} + \mathbf{V}_{nl})\mathbf{U} = \mathbf{MU\Lambda}$$
$$V_{nl}^{IJ} = \sum_a \int \chi^a(\mathbf{x}) N_I(\mathbf{x}) d\mathbf{x} \;\; \Delta^a \int \chi^a(\mathbf{y}) N_J(\mathbf{y}) d\mathbf{y}$$

COMPUTATIONAL METHODOLOGY FOR $\mathbf{A}^e\mathbf{U}^e$: REAL ARITHMETIC

- **Cell-Matrix approach ($\mathbf{A}^e\mathbf{U}^e$):** Evaluation of the element-level matrices $\mathbf{T}^e$, $\mathbf{L}^e$ and $\mathbf{G}^e$:

$$\mathbf{T}^e = \begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix}^T \begin{bmatrix}\mathcal{G}_{00} & \mathcal{G}_{01} & \mathcal{G}_{02}\\ \mathcal{G}_{10} & \mathcal{G}_{11} & \mathcal{G}_{12}\\ \mathcal{G}_{20} & \mathcal{G}_{21} & \mathcal{G}_{22}\end{bmatrix} \begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix}, \qquad \mathbf{G}^e = \left(\begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix}^T \begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix} + \begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix}^T \begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix} \right),$$
$$\mathbf{L}^e = \mathbf{N}^T\mathbf{V}_L\mathbf{N}$$

where $N_{QI} = \hat{N}_I(\hat{\mathbf{x}}_Q)$ and $D_{QI}^s = \hat{\nabla}\hat{N}_I(\hat{\mathbf{x}}_Q) \cdot \hat{\boldsymbol{n}}_s$ are $n_q^3 \times n_p^3$ matrices representing the FE cell-level basis function values and the basis function gradients respectively at Gauss quadrature points, with $\hat{\boldsymbol{n}}_s$, $s = 0, 1, 2$ representing the unit vector along the s^{th} axis, and $\mathbf{V}_L^{QQ} = \mathcal{V}_L \det\mathbf{J}^e w_Q\Big|_{\hat{\mathbf{x}}_Q}$, $\mathcal{G}_{sd}^{QQ} = \frac{1}{2}\left[(\mathbf{J}^e)^{-1}(\mathbf{J}^e)^{-T}\right]_{sd} \det\mathbf{J}_e w_Q\Big|_{\hat{\mathbf{x}}_Q}$ are $n_q^3 \times n_q^3$ matrices representing the mapping to the reference domain ($\mathbf{J}_e$ is the jacobian matrix of the map from Ω^e to $\hat{\Omega}$.).

- **Matrix-free approach ($\mathbf{A}^e\mathbf{U}^e$):** The element level matrix multi-vector products are evaluated *on-the-fly* by taking advantage of the tensor structured nature of the FE basis functions and the quadrature rules instead of precomputing $\mathbf{K}^e$, $\mathbf{V}_l^e$, etc.

$$\begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix} = \begin{bmatrix}\tilde{\mathbf{D}}_0\\ \tilde{\mathbf{D}}_1\\ \tilde{\mathbf{D}}_2\end{bmatrix}\mathbf{N}; \quad \mathbf{A}_0 = \mathbf{NU}^e \equiv (\mathbf{N}^{1D} \otimes \mathbf{N}^{1D} \otimes \mathbf{N}^{1D} \otimes \boldsymbol{I})\boldsymbol{\mathcal{U}}^e, \quad \mathbf{A}_1 = \begin{bmatrix}\tilde{\mathbf{D}}_0\\ \tilde{\mathbf{D}}_1\\ \tilde{\mathbf{D}}_2\end{bmatrix}\mathbf{A}_0, \quad \mathbf{A}_2 = \begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix}^T \mathbf{A}_1,$$

$$\mathbf{A}^e\mathbf{U}^e = \mathbf{A}_3 = \mathbf{N}^T\left[\begin{bmatrix}\tilde{\mathbf{D}}_0\\ \tilde{\mathbf{D}}_1\\ \tilde{\mathbf{D}}_2\end{bmatrix}^T \left(\begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix}\mathbf{A}_0 + \begin{bmatrix}\mathcal{G}_{00} & \mathcal{G}_{01} & \mathcal{G}_{02}\\ \mathcal{G}_{10} & \mathcal{G}_{11} & \mathcal{G}_{12}\\ \mathcal{G}_{20} & \mathcal{G}_{21} & \mathcal{G}_{22}\end{bmatrix}\mathbf{A}_1\right) + \mathbf{A}_2 + \mathbf{V}_L\mathbf{A}_0\right].$$

The computational complexity of evaluating the products $\tilde{\mathbf{D}}_k\mathbf{U}^e$ and $\mathbf{N}^{1D}\mathbf{U}^e$ can be further reduced by utilizing the symmetry relations of the Gauss Quadrature points and the finite-element nodes, $\tilde{\mathbf{D}}_{k,i,j} = -\tilde{\mathbf{D}}_{k,n_q-i,p-j}$ and $\mathbf{N}_{i,j}^{1D} = \mathbf{N}_{n_q-i,p-j}^{1D}$.

KOHN-SHAM DFT EIGENVALUE PROBLEM: COMPLEX ARITHMETIC

Kohn-Sham DFT Equation (Pseudopotential GGA for Periodic Systems)

$$\mathcal{H}\psi_j(\mathbf{x}) = \left(-\frac{1}{2}\nabla^2 + \mathcal{V}_L(\rho) + \mathcal{V}_G(\rho, \nabla\rho, \boldsymbol{R}) - i\boldsymbol{k}\cdot\nabla + \frac{1}{2}|\boldsymbol{k}|^2\right)\psi_j(\mathbf{x}) + \int \mathcal{V}_{nl}(\mathbf{x}, \mathbf{y}, \boldsymbol{R})\psi_j(\mathbf{y}) d\mathbf{y} = \lambda_j\psi_j(\mathbf{x})$$

Finite-element discretized equations:

$$\mathbf{AU} := (\mathbf{T} + \mathbf{L} + \mathbf{G} + \mathbf{V}_{nl} - i\mathbf{K})\mathbf{U} = \mathbf{MU\Lambda}$$
$$\mathbf{K}^{IJ} = \int \mathcal{V}_K \cdot (N_I(\mathbf{x})\nabla N_J(\mathbf{x})) d\mathbf{x}$$

Matrix-free approach ($\mathbf{A}^e\mathbf{U}^e$):

$$\mathbf{K}^e = \begin{bmatrix}\mathbf{V}_0^K\\ \mathbf{V}_1^K\\ \mathbf{V}_2^K\end{bmatrix}^T \begin{bmatrix}\mathbf{D}_0\\ \mathbf{D}_1\\ \mathbf{D}_2\end{bmatrix}, \qquad \mathbf{A}_0 = \mathbf{NU}^e \equiv (\mathbf{N}^{1D} \otimes \mathbf{N}^{1D} \otimes \mathbf{N}^{1D} \otimes \boldsymbol{I})\boldsymbol{\mathcal{U}}^e, \qquad \mathbf{A}_1 = \begin{bmatrix}\tilde{\mathbf{D}}_0\\ \tilde{\mathbf{D}}_1\\ \tilde{\mathbf{D}}_2\end{bmatrix}\mathbf{A}_0, \qquad \mathbf{A}_2 = \begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix}^T \mathbf{A}_1,$$

$$\mathbf{A}_3 = \begin{bmatrix}\mathbf{V}_0^K\\ \mathbf{V}_1^K\\ \mathbf{V}_2^K\end{bmatrix}^T \mathbf{A}_1, \qquad \mathbf{A}^e\mathbf{U}^e = \mathbf{A}_4 = \mathbf{N}^T\left[\begin{bmatrix}\tilde{\mathbf{D}}_0\\ \tilde{\mathbf{D}}_1\\ \tilde{\mathbf{D}}_2\end{bmatrix}^T \left(\begin{bmatrix}\mathbf{V}_0^G\\ \mathbf{V}_1^G\\ \mathbf{V}_2^G\end{bmatrix}\mathbf{A}_0 + \begin{bmatrix}\mathcal{G}_{00} & \mathcal{G}_{01} & \mathcal{G}_{02}\\ \mathcal{G}_{10} & \mathcal{G}_{11} & \mathcal{G}_{12}\\ \mathcal{G}_{20} & \mathcal{G}_{21} & \mathcal{G}_{22}\end{bmatrix}\mathbf{A}_1\right) + \mathbf{A}_2 + \mathbf{V}_L\mathbf{A}_0 - i\mathbf{A}_3\right].$$

HARDWARE-AWARE IMPLEMENTATION : TENSOR CONTRACTIONS

CPU implementation strategy

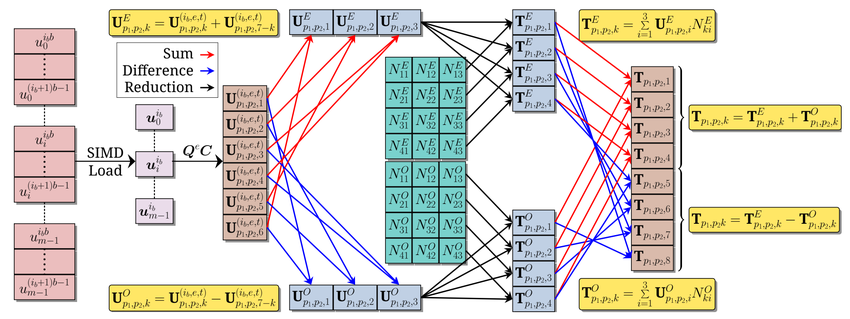

Figure 3. Pictorial depiction of tensor contractions done on CPUs using the even-odd decomposition strategy. The extraction and first tensor contraction steps of evaluation of $\mathbf{A}^{(e)}\mathbf{U}^{(i_b,e,t)}$ are depicted for the case of $n_p = 6$ and $n_q = 8$. Each block in **U** represents n_p^2 sized array of AVX-512 doubles.

PERFORMANCE BENCHMARKS ON FRONTIER, PRAVEGA AND FUGAKU

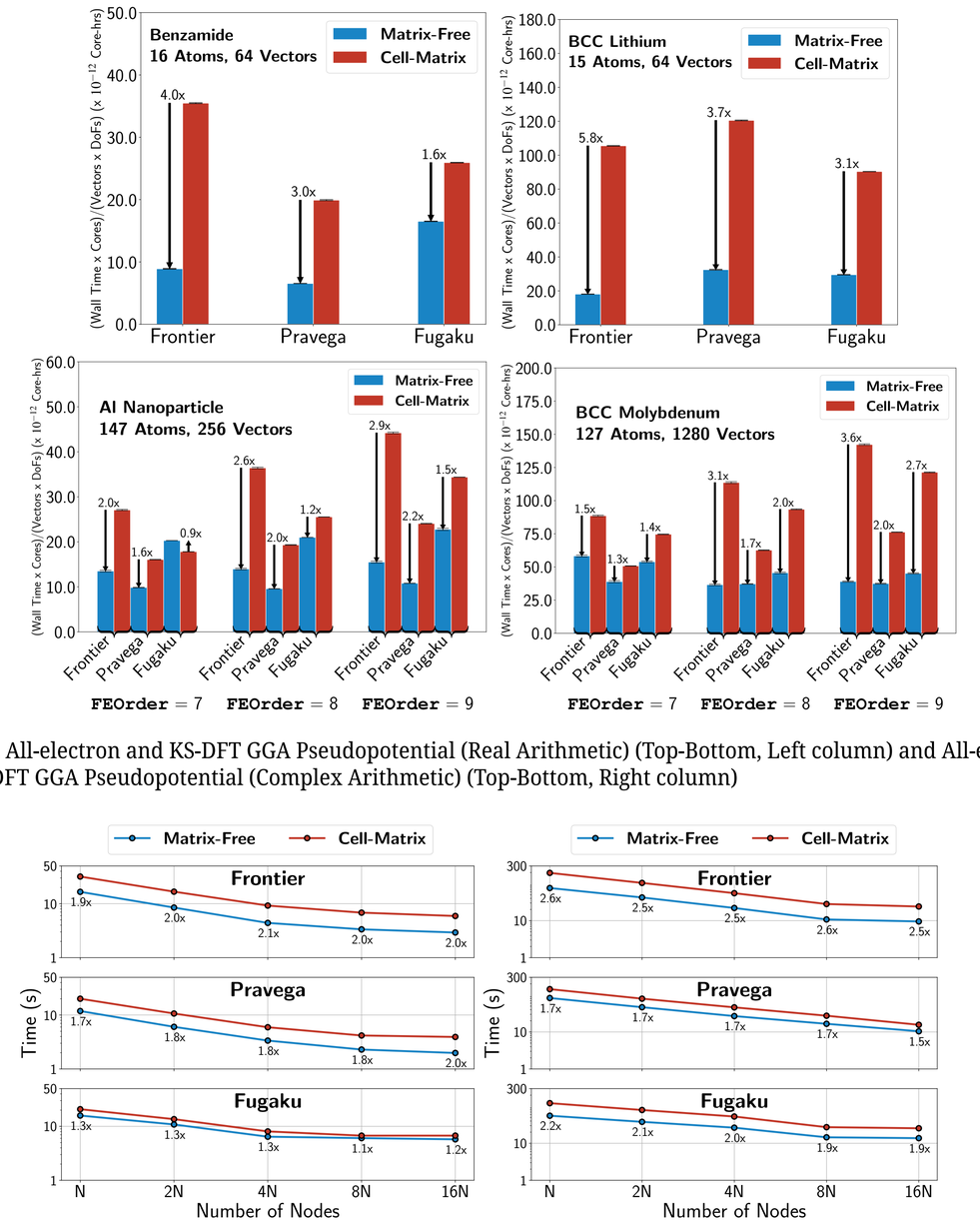

Figure 4. All-electron and KS-DFT GGA Pseudopotential (Real Arithmetic) (Top-Bottom, Left column) and All-electron and KS-DFT GGA Pseudopotential (Complex Arithmetic) (Top-Bottom, Right column)

Figure 5. Left: Al nanoparticle 561 atoms, Right: BCC molybdenum 1023 atoms with FEOrder = 8, FP32 precision on Frontier (Left: N = 4, Right: N = 12), Param Pravega (Left: N = 4, Right: N = 10) and Fugaku (Left: N = 70, Right: N = 200) supercomputers.

REFERENCES

[1] G. Panigrahi, N. Kodali, D. Panda, and P. Motamarri, "Fast hardware-aware matrix-free algorithms for higher-order finite-element discretized matrix multivector products on distributed systems," *Journal of Parallel and Distributed Computing*, vol. 192, p. 104925, 2024.

[2] S. Das, B. Kanungo, V. Subramanian, G. Panigrahi, P. Motamarri, D. Rogers, P. Zimmerman, and V. Gavini, "Large-scale materials modeling at quantum accuracy: Ab initio simulations of quasicrystals and interacting extended defects in metallic alloys," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '23, (New York, NY, USA), Association for Computing Machinery, 2023.

[3] G. Panigrahi and P. Motamarri, "Matrix-free algorithms for fast ab initio calculations on distributed cpu architectures using finite-element discretization," 2025.

ACKNOWLEDGEMENTS

Google Research · NSM National Supercomputing Mission (Infrastructure | Applications | R&D | HRD) · PMRF Prime Minister's Research Fellowship · Ministry of Education, Government of India · Oak Ridge National Laboratory · RIKEN