CDFCI: High-Performance Parallel Software for Eigenvalue Problems in Many-Body Schrödinger Equation

Yuejia Zhang

Fudan University

March 2026

Background and Challenges

- In quantum mechanics, the Hamiltonian operator $\hat{\boldsymbol{H}}$ represents the total energy of the system, including the contributions of kinetic energy and potential energy.
- The evolution of the state of the system over time (according to the Schrödinger picture)

$$i\hbar\frac{\partial}{\partial t}\Psi(t) = \hat{\boldsymbol{H}}\Psi(t). \tag{1}$$

- The spectrum of the Hamiltonian operator determines the energy level structure and dynamic behavior of the system

$$\hat{\boldsymbol{H}}\Psi = E\Psi. \tag{2}$$

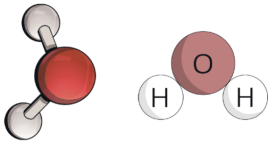
- The essential difficulty of the many-body problem: no analytical solution for $n_{\text{elec}} > 1$. The size of the problem grows exponentially with n_{elec}.

Numerical Discretization of Schrödinger Equation

- Let's assume $\Psi \in L^2(\mathbb{R}^{3n_{\text{elec}}}, \mathbb{C})$ for now. ($||\Psi||_{L^2} = 1$)
- **Finite Element Method** with naive grid $10 \times 10 \times 10$: $10^{3n_{\text{elec}}}$ grid points.
- **Spectral Method** with (truncated) one-electron basis functions $\{\varphi_p\}_{p=1}^{10}$:

$$L^2(\mathbb{R}^{3n_{\text{elec}}}, \mathbb{C}) = \bigotimes_{i=1}^{n_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C}),$$

$10^{n_{\text{elec}}}$ many-body basis functions.



System-Level Approximations

- Under Born–Oppenheimer approximation, the Hamiltonian operator with n_{nuc} nuclei and n_{elec} electrons is

$$\hat{\boldsymbol{H}} = -\frac{1}{2}\sum_{i=1}^{n_{\text{elec}}} \nabla_i^2 + \sum_{i=1}^{n_{\text{elec}}} V_{\text{ext}}(r_i; \{R_I\}_{I=1}^{n_{\text{nuc}}}) + \sum_{i<j}^{n_{\text{elec}}} \frac{1}{\|r_i - r_j\|}.$$

- **Hartree–Fock method**: mean-field approximation, neglecting electron correlation.
- **Density Functional Theory**: reformulate the problem in terms of electron density, with approximated exchange-correlation functionals.

$$\hat{\boldsymbol{H}} \approx \sum_{i=1}^{n_{\text{elec}}} \left(-\frac{1}{2}\nabla_i^2 + V_{\text{ext}}(r_i; \{R_I\}_{I=1}^{n_{\text{nuc}}}) + V_{\text{approx}}(r_i) \right).$$

Dealing with Electron Correlation — Full Configuration Interaction (FCI)

- Work in space $\bigwedge^{n_{\text{elec}}} L^2(\mathbb{R}^3, \mathbb{C})$ to ensure anti-symmetry of the wavefunction (Pauli exclusion principle).
- From a truncated one-electron basis set $\{\chi_p\}_{p=1}^{n_{\text{orb}}}$, the basis functions of the many-body space are **Slater determinants** (or, **anti-symmetrized tensor products**):

$$|D_i\rangle = |\chi_{p_1}\chi_{p_2}\cdots\chi_{p_n}\rangle = \chi_{p_1} \wedge \chi_{p_2} \wedge \cdots \wedge \chi_{p_n},$$

  with $n = n_{\text{elec}}$ and $1 \le p_1 < p_2 < \cdots < p_n \le n_{\text{orb}}$.
- FCI variational space dimension: $N_{\text{FCI}} = \binom{n_{\text{orb}}}{n_{\text{elec}}}$.
- Schrödinger equation can be transformed to FCI eigenvalue problem

$$H\mathbf{c} = E_0\mathbf{c}, \quad H \in \mathbb{R}^{N_{\text{FCI}} \times N_{\text{FCI}}}, \quad \mathbf{c} \in \mathbb{R}^{N_{\text{FCI}}}.$$

Exponential Growth of FCI Problem Size

Table: Different Molecule Systems and Storage cost

Molecule	Basis	Electrons	Spin–Orbitals	Dimension	Memory
H_2O	cc-pVDZ	10	48	$\sim 10^8$	~ 1 GB
N_2	cc-pVDZ	14	56	$\sim 10^{11}$	~ 1 TB
N_2	cc-pVTZ	14	120	$\sim 10^{16}$	~ 100 PB
Cr_2	Ahlrichs	48	84	$\sim 10^{22}$	-

Observations (Good news!)

1. We know every entry of the Hamiltonian matrix H explicitly and the sparsity pattern *a priori*!
2. Entry: $H_{ij} = H_{ji} = \langle D_i|\hat{\boldsymbol{H}}|D_j\rangle$. For off-diagonals $|D_i\rangle \neq |D_j\rangle$,
   - If $|D_i\rangle = a_r^\dagger a_p |D_j\rangle$, $H_{ij} = \langle r|\hat{\boldsymbol{H}}|p\rangle + \sum_k \langle rk||pk\rangle$.
   - If $|D_i\rangle = a_r^\dagger a_s^\dagger a_p a_q |D_j\rangle$, $H_{ij} = \langle rs||pq\rangle$.
   - Otherwise, $H_{ij} = 0$.

   Consequence: H has $O(n_{\text{elec}}^2 n_{\text{orb}}^2)$ entries per row.
3. Eigenvector $\mathbf{c}$ of a few low-lying eigenvalues usually sparse (in the sense of truncation).

Idea: we can select an index set of "important" coefficients $\mathcal{I}$ and compute the lowest eigenvalues of $H_{\mathcal{I},\mathcal{I}}$.
Question: How to select $\mathcal{I}$?

Existing Methods

- **Selected CI**: Pick the important configurations based on perturbation theory estimates.

$$c_i^{(1)} = \frac{\langle D_i|\hat{\boldsymbol{H}}|\Psi_0\rangle}{E_0 - \langle D_i|\hat{\boldsymbol{H}}|D_i\rangle}, \qquad \Delta E^{(2)} = \sum_{i\notin\mathcal{V}} \frac{\left|\langle D_i|\hat{\boldsymbol{H}}|\Psi_0\rangle\right|^2}{E_0 - \langle D_i|\hat{\boldsymbol{H}}|D_i\rangle}.$$

  1. From selected configurations, generate new ones by single/double excitations.
  2. Sort by perturbation estimates, and select the top ones.
  3. Diagonalize the Hamiltonian in the selected space.
  4. Repeat until convergence/out of memory.

- ... and other FCI methods such as **DMRG** or **FCIQMC**.

Coordinate Descent FCI (CDFCI)[1]

Coordinate gradient descent method + new way of formalizing the problem.
Consider the unconstrained minimization problem

$$\min_{\mathbf{c}\in\mathbb{R}^{N_{\mathrm{FCI}}}} f(\mathbf{c}) = \min_{\mathbf{c}} \|H + \mathbf{c}\mathbf{c}^{\mathsf{T}}\|_F^2.$$

- Gradient $\nabla f(c) = 4H\mathbf{c} + 4(\mathbf{c}^{\mathsf{T}}\mathbf{c})\mathbf{c}$.
- Hessian $\nabla^2 f(c) = 4H + 8\mathbf{c}\mathbf{c}^{\mathsf{T}} + 4(\mathbf{c}^{\mathsf{T}}\mathbf{c})I$.
- Stationary points: 0 and $\pm\sqrt{-\lambda_k}\mathbf{v}_k$ for any $\lambda_k < 0$.
- If $\lambda_1 < \lambda_2$ and $\lambda_1 < 0$, only two local minimizers $\pm\sqrt{-\lambda_1}\mathbf{v}_1$,which are also global minimizers, strongly convex in a neighborhood!

[1] Z. Wang, Y. Li, J. Lu, J. Chem. Theory Comput., 2019.

CDFCI Framework

Initialize $\mathbf{c}^{(0)}$, $\mathbf{b}^{(0)} = H\mathbf{c}^{(0)}$.
For iteration $\ell = 1, 2, \ldots$

1. Select coordinate $i^{(\ell)} = \arg\max_i |\nabla_i f(\mathbf{c}^{(\ell-1)})|$.
2. Find stepsize by exact line search $\alpha^{(\ell)} = \arg\min_\alpha f(\mathbf{c}^{(\ell-1)} + \alpha \mathbf{e}_{i^{(\ell)}})$.
3. Update $\mathbf{c}^{(\ell)} = \mathbf{c}^{(\ell-1)} + \alpha^{(\ell)} \mathbf{e}_{i^{(\ell)}}$, $\mathbf{b}^{(\ell)} = \mathbf{b}^{(\ell-1)} + \alpha^{(\ell)} H_{:,i^{(\ell)}}$.

Remark: Gradient
$\nabla f(\mathbf{c}) = 4H\mathbf{c} + 4\mathbf{c}^\mathsf{T}\mathbf{c}\mathbf{c} = 4\mathbf{b} + 4\mathbf{c}^\mathsf{T}\mathbf{c}\mathbf{c}$.

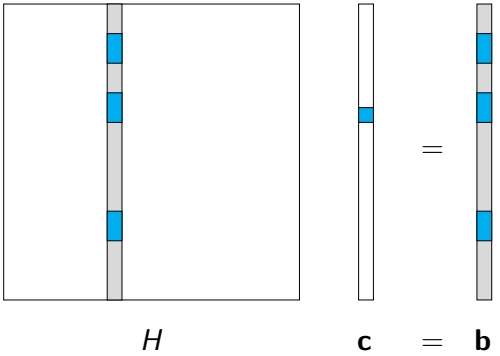


Figure: Update for one coordinate.

CDFCI Framework - for Two Coordinates?

Initialize $\mathbf{c}^{(0)}$, $\mathbf{b}^{(0)} = H\mathbf{c}^{(0)}$.
For iteration $\ell = 1, 2, \ldots$

1. Select coordinate
   $i^{(\ell)} = \arg\max_i |\nabla_i f(\mathbf{c}^{(\ell-1)})|$,
   $j^{(\ell)} = \arg\max_{j \neq i^{(\ell)}} |\nabla_j f(\mathbf{c}^{(\ell-1)})|$.
2. Find stepsize
   $\alpha^{(\ell)} = \arg\min_\alpha f(\mathbf{c}^{(\ell-1)} + \alpha \mathbf{e}_{i^{(\ell)}})$,
   $\beta^{(\ell)} = \arg\min_\beta f(\mathbf{c}^{(\ell-1)} + \beta \mathbf{e}_{j^{(\ell)}})$.
3. Update
   $\mathbf{c}^{(\ell)} = \mathbf{c}^{(\ell-1)} + \alpha^{(\ell)} \mathbf{e}_{i^{(\ell)}} + \beta^{(\ell)} \mathbf{e}_{j^{(\ell)}}$,
   $\mathbf{b}^{(\ell)} = \mathbf{b}^{(\ell-1)} + \alpha^{(\ell)} H_{:,i^{(\ell)}} + \beta^{(\ell)} H_{:,j^{(\ell)}}$.

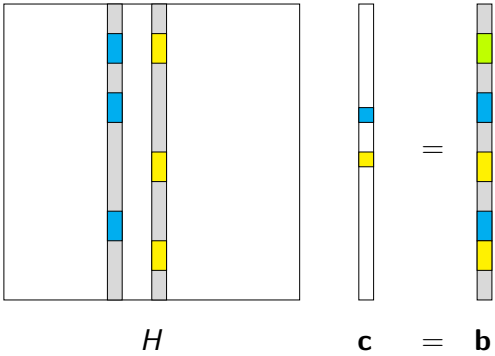


Figure: Update for two coordinates.

CDFCI Framework - Exact Line Search?

Initialize $\mathbf{c}^{(0)}$, $\mathbf{b}^{(0)} = H\mathbf{c}^{(0)}$.
For iteration $\ell = 1, 2, \ldots$

1. Select coordinate
   $i^{(\ell)} = \arg\max_i |\nabla_i f(\mathbf{c}^{(\ell-1)})|$,
   $j^{(\ell)} = \arg\max_{j \neq i^{(\ell)}} |\nabla_j f(\mathbf{c}^{(\ell-1)})|$.
2. Find stepsize $\alpha^{(\ell)}, \beta^{(\ell)} =$
   $\arg\min_{\alpha,\beta} f(\mathbf{c}^{(\ell-1)} + \alpha \mathbf{e}_{i^{(\ell)}} + \beta \mathbf{e}_{j^{(\ell)}})$.
3. Update
   $\mathbf{c}^{(\ell)} = \mathbf{c}^{(\ell-1)} + \alpha^{(\ell)} \mathbf{e}_{i^{(\ell)}} + \beta^{(\ell)} \mathbf{e}_{j^{(\ell)}}$,
   $\mathbf{b}^{(\ell)} = \mathbf{b}^{(\ell-1)} + \alpha^{(\ell)} H_{:,i^{(\ell)}} + \beta^{(\ell)} H_{:,j^{(\ell)}}$.

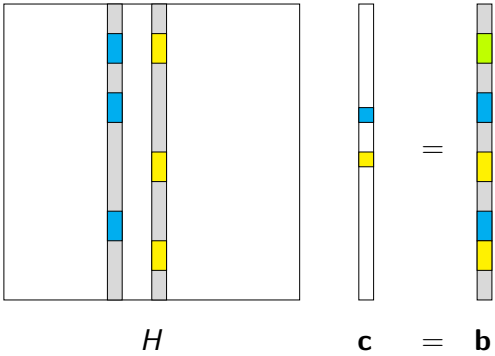


Figure: Update for two coordinates.

Add a Scalar γ for Exact Line Search

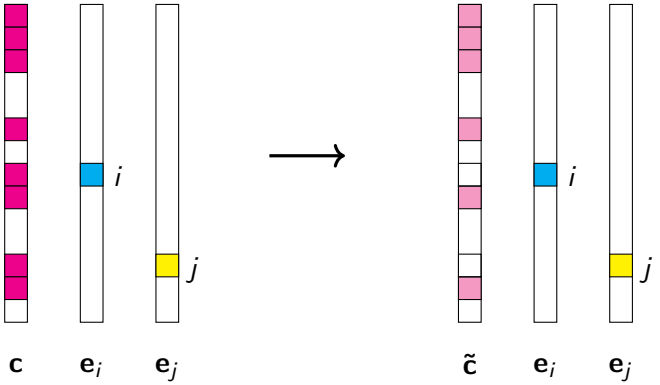
Modify the minimization problem from $\min_{\alpha,\beta} f(\mathbf{c} + \alpha\mathbf{e}_i + \beta\mathbf{e}_j)$ to

$$\min_{\gamma,\alpha,\beta} f(\gamma\mathbf{c} + \alpha\mathbf{e}_i + \beta\mathbf{e}_j) = f(\begin{bmatrix}\mathbf{c} & \mathbf{e}_i & \mathbf{e}_j\end{bmatrix}\begin{bmatrix}\gamma\\ \alpha\\ \beta\end{bmatrix})$$

$$= \left\| H + \begin{bmatrix}\mathbf{c} & \mathbf{e}_i & \mathbf{e}_j\end{bmatrix}\begin{bmatrix}\gamma\\ \alpha\\ \beta\end{bmatrix}\begin{bmatrix}\gamma & \alpha & \beta\end{bmatrix}\begin{bmatrix}\mathbf{c}^T\\ \mathbf{e}_i^T\\ \mathbf{e}_j^T\end{bmatrix} \right\|_F^2 .$$

Matrix Orthogonalization

Construct $\begin{bmatrix} \tilde{\mathbf{c}} & \mathbf{e}_i & \mathbf{e}_j \end{bmatrix}$, where $\|\tilde{\mathbf{c}}\|_2 = 1$, $(\tilde{\mathbf{c}}, \mathbf{e}_i) = 0$, $(\tilde{\mathbf{c}}, \mathbf{e}_j) = 0$.



Add γ and $\tilde{\mathbf{c}}$ for Exact Line Search

Modify the minimization problem from $\min_{\alpha,\beta} f(\mathbf{c} + \alpha\mathbf{e}_i + \beta\mathbf{e}_j)$ to

$$\min_{\gamma,\alpha,\beta} f(\gamma\tilde{\mathbf{c}} + \alpha\mathbf{e}_i + \beta\mathbf{e}_j) = f(\begin{bmatrix}\tilde{\mathbf{c}} & \mathbf{e}_i & \mathbf{e}_j\end{bmatrix}\begin{bmatrix}\gamma\\ \alpha\\ \beta\end{bmatrix})$$

$$= \left\| H + \begin{bmatrix}\tilde{\mathbf{c}} & \mathbf{e}_i & \mathbf{e}_j\end{bmatrix}\begin{bmatrix}\gamma\\ \alpha\\ \beta\end{bmatrix}\begin{bmatrix}\gamma & \alpha & \beta\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{c}}^{\mathsf{T}}\\ \mathbf{e}_i^{\mathsf{T}}\\ \mathbf{e}_j^{\mathsf{T}}\end{bmatrix} \right\|_F^2$$

$$= \left\| \underbrace{\begin{bmatrix}\tilde{\mathbf{c}}^{\mathsf{T}}\\ \mathbf{e}_i^{\mathsf{T}}\\ \mathbf{e}_j^{\mathsf{T}}\end{bmatrix} H \begin{bmatrix}\tilde{\mathbf{c}} & \mathbf{e}_i & \mathbf{e}_j\end{bmatrix}}_{\in \mathbb{R}^{3\times 3}} + \begin{bmatrix}\gamma\\ \alpha\\ \beta\end{bmatrix}\begin{bmatrix}\gamma & \alpha & \beta\end{bmatrix} \right\|_F^2 + \text{constant.}$$

Extension to Multi Coordinate Descent FCI

- Select a set of coordinates $I = \{i_1, \ldots, i_k\}, 1 \leq i_j \leq N_{\text{FCI}}$ based on gradient $\nabla f(\mathbf{c}) = 4H\mathbf{c} + 4\mathbf{c}^{\mathsf{T}}\mathbf{c}\mathbf{c}$.
- Denote $\mathcal{E}_I = [e_{i_1}, \ldots, e_{i_k}] \in \mathbb{R}^{N_{\text{FCI}} \times k}$.
- The update is given by

$$\mathbf{c} \leftarrow \gamma\mathbf{c} + \mathcal{E}_I\mathbf{a}.$$

- The values of γ and $\mathbf{a}$ are given by the eigenvector of

$$\begin{bmatrix} \tilde{\mathbf{c}}^{\mathsf{T}} \\ \mathcal{E}_I^{\mathsf{T}} \end{bmatrix} H \begin{bmatrix} \tilde{\mathbf{c}} & \mathcal{E}_I \end{bmatrix} \in \mathbb{R}^{(k+1)\times(k+1)}$$

corresponding to the minimal eigenvalue $\lambda_{\min}$, which is the current energy estimate.
Subspace Methods with Rayleigh–Ritz projection!

Compression Strategy

- Compression occurs when updating $\mathbf{b} = H\mathbf{c}$:

$$\mathbf{b} \leftarrow \gamma \mathbf{b} + H\mathcal{E}_I \mathbf{a},$$

$$b_i \leftarrow \gamma b_i + \sum_{j \in I} H_{ij} a_j.$$

- Update $H_{ij}a_j$ is discarded if $\mathbf{c}_i = 0$ and $|H_{ij}a_j| < \tau$.
- Not affecting eigenvalue estimator

$$\mathrm{RQ}(\mathbf{c}) = \frac{\mathbf{c}^\mathsf{T} H \mathbf{c}}{\mathbf{c}^\mathsf{T}\mathbf{c}} = \frac{\mathbf{c}^\mathsf{T}\mathbf{b}}{\mathbf{c}^\mathsf{T}\mathbf{c}}.$$

- Tolerance τ balances between memory-cost and accuracy.

Parallel Implementation Details

- Shared memory parallelism based on OpenMP.
- Sparse vectors $\mathbf{c}$ and $\mathbf{b} = H\mathbf{c}$ are stored in a concurrent hash table[2] that supports parallel read/write.
- Two levels of parallelism: row-wise and column-wise.

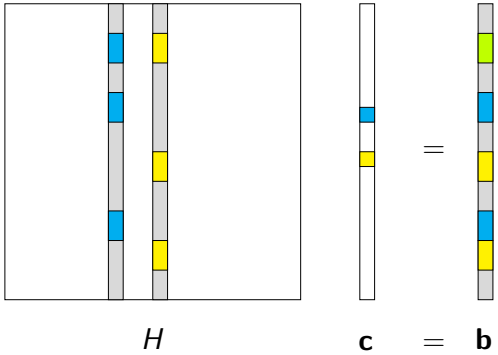
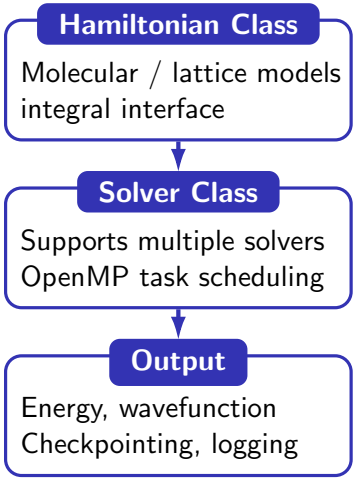


Figure: Update for two coordinates.

[2] https://github.com/efficient/libcuckoo

CDFCI Software Package

- Written in **C++17**, modular design with **OpenMP** parallelization
- Scalable to determinant spaces with **tens of millions of basis functions**
- Dynamic task scheduling and load balancing with a multi-solver framework
- Supports excited states calculations (xCDFCI) and single-particle orbitals compression (OptOrbFCI)
- **Open-source software**: reproducible code and reliable results



Numerical Results

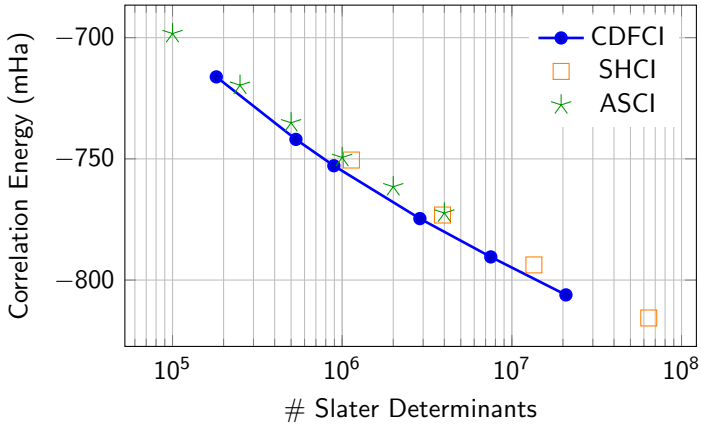


Figure: Correlation energy vs. number of Slater determinants for the benzene benchmark. CDFCI results are compared with reference data from SHCI and ASCI.

mCDFCI vs CDFCI

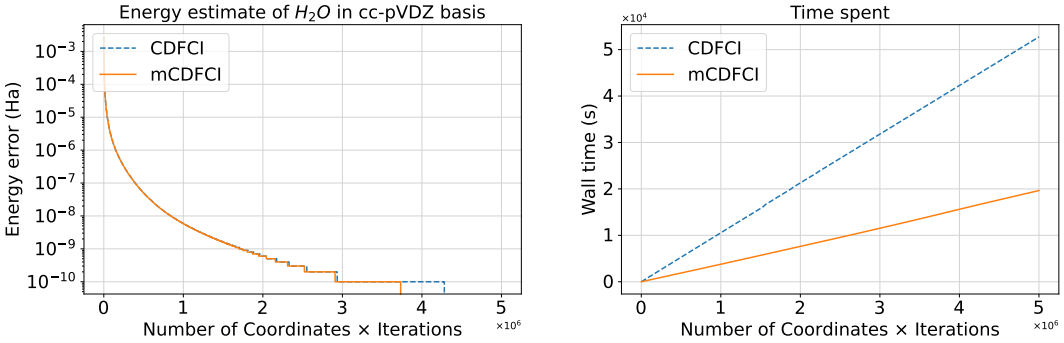


Figure: Speedup of mCDFCI compared with CDFCI, both in 64 threads and $k = 64$.

Numerical Results - Parallel Scaling

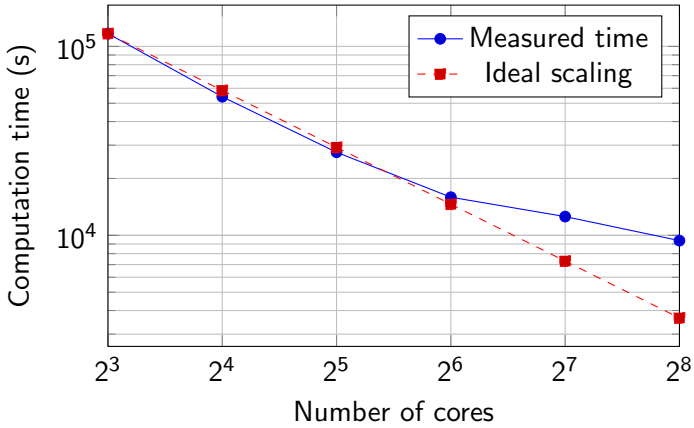


Figure: Strong scaling of CDFCI for the N_2/cc-pVQZ system. Experiments were conducted on a system equipped with two AMD EPYC 9754 processors (128 cores each) and 1.5 TB of memory.

Thanks for Your Attention!

I'd love to hear any questions or suggestions!

This is joint work with Dr. Zhe Wang, Prof. Jianfeng Lu and Prof. Yingzhou Li.
Contact: `yuejiazhang21@m.fudan.edu.cn`

Related work to this talk:

- **CDFCI Core Algorithm**: Z. Wang, Y. Li, J. Lu, JCTC, 15(6), 2019.
- **The Parallel Version**: Y. Zhang, W. Gao, Y. Li, JCTC, 21(5), 2025.
- **Other methods implemented in the software package**:
  - Y. Li, J. Lu, JCTC, 16(10), 2020.
  - Z. Wang, Z. Zhang, J. Lu, Y. Li, JCTC, 19(21), 2023.