Recent Developments in the ELSI Infrastructure for Large-Scale Electronic Structure Theory

Victor Yu[1], William Dawson[2], Alberto García[3], Ville Havu[4], Ben Hourahine[5], William Huhn[1], Mathias Jacquelin[6], Weile Jia[6,7], Murat Keçeli[8], Raul Laasner[1], Yingzhou Li[9], Lin Lin[6,7], Jianfeng Lu[9], Jose Roman[10], Álvaro Vázquez-Mayagoita[8], Chao Yang[6], and Volker Blum[1]

[1]Department of Mechanical Engineering and Materials Science, Duke University, USA
[2]RIKEN Center for Computational Science, Japan
[3]Institut de Ciència de Materials de Barcelona, Spain
[4]Department of Applied Physics, Aalto University, Finland
[5]Department of Physics, University of Strathclyde, Scotland
[6]Computational Research Division, Lawrence Berkeley National Laboratory, USA
[7]Department of Mathematics, University of California at Berkeley, USA
[8]Argonne Leadership Computing Facility, Argonne National Laboratory, USA
[9]Department of Mathematics, Duke University, USA
[10]Departament de Sistemes Informàtics i Computació, Universitat Politècnica de València, Spain

Kohn-Sham Density-Functional Theory (KS-DFT)

Kohn-Sham equation

$$\hat{h}_{KS}\psi_l = \epsilon_l \psi_l$$

$\hat{h}_{KS}$: Kohn-Sham Hamiltonian
ψ_l: Kohn-Sham orbitals
ϵ_l: Eigen-energies

Basis set expansion

$$\psi_l(\mathbf{r}) = \sum_j c_{lj}\varphi_j(\mathbf{r})$$

ψ_l: Kohn-Sham orbitals
$\varphi_j(\mathbf{r})$: "Basis" functions
c_{lj}: Expansion coefficients

Eigenvalue problem

$$\mathbf{Hc} = \boldsymbol{\epsilon}\mathbf{Sc}$$

$\mathbf{H}$: Hamiltonian matrix
$\mathbf{S}$: Overlap matrix
$\mathbf{c}$: Eigenvectors
$\boldsymbol{\epsilon}$: Eigenvalues

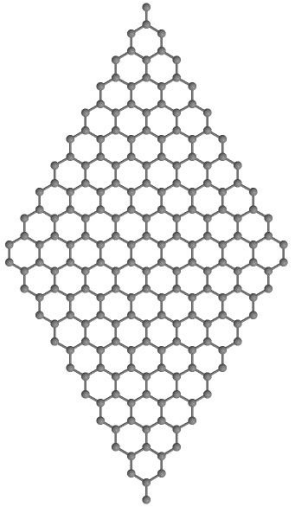
Target: Total energy is a functional of electron density: $E = E[n]$

Nonlinear optimization

- Hamiltonian $\mathbf{H}$ depends on electron density n
- Electron density n depends on wavefunctions $\{\psi_l\}$
- Wavefunctions $\{\psi_l\}$ depend on Hamiltonian $\mathbf{H}$

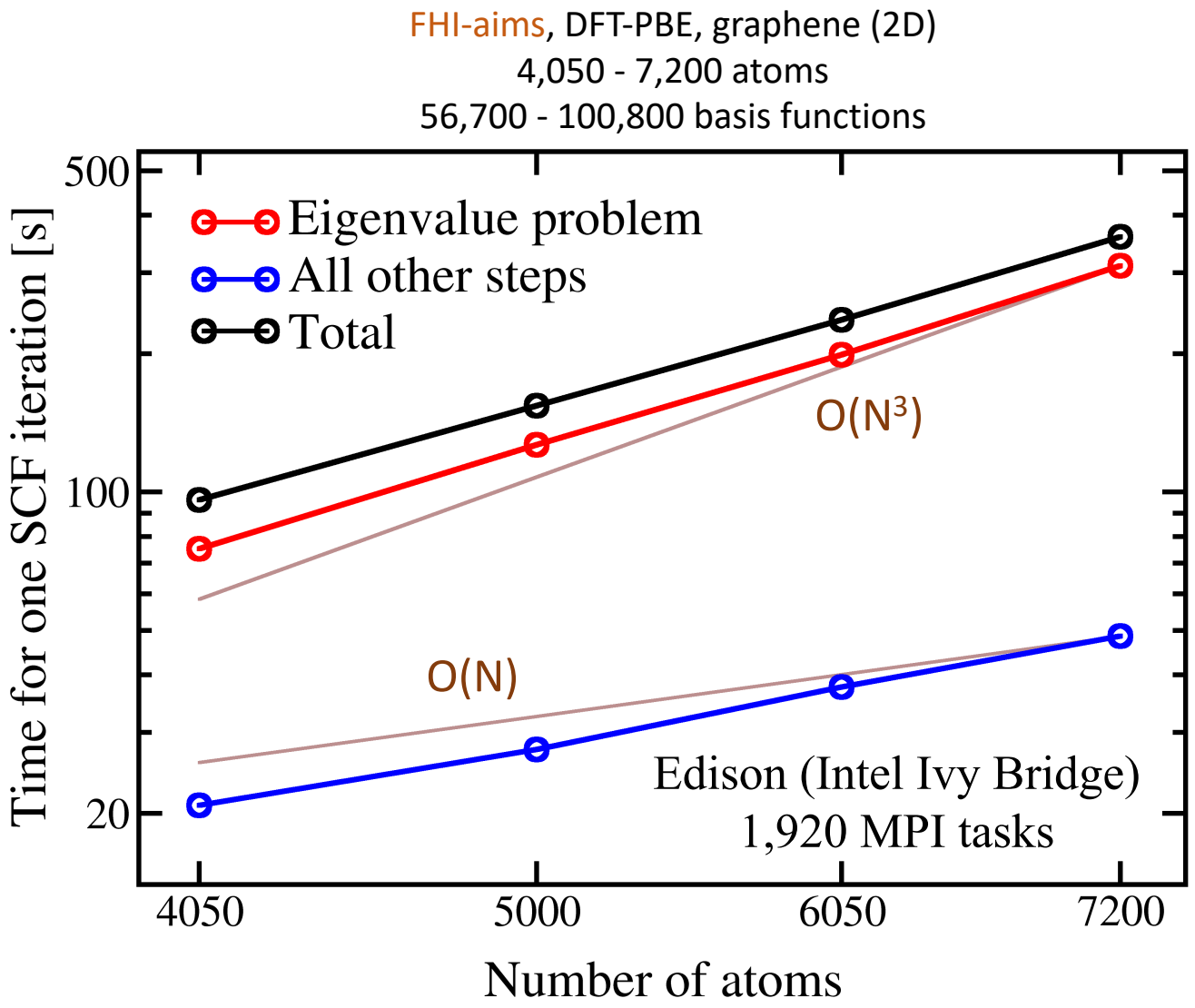
Solution: Repeatedly update electron density n towards self-consistency

“Cubic Wall” in KS-DFT

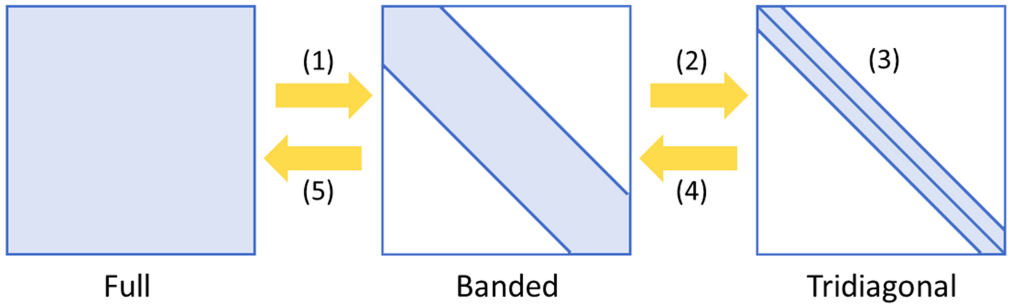


- Diagonalization with a dense eigensolver: $O(N^3)$
- Other steps: O(N) (achievable with localized basis)

N: Number of atoms



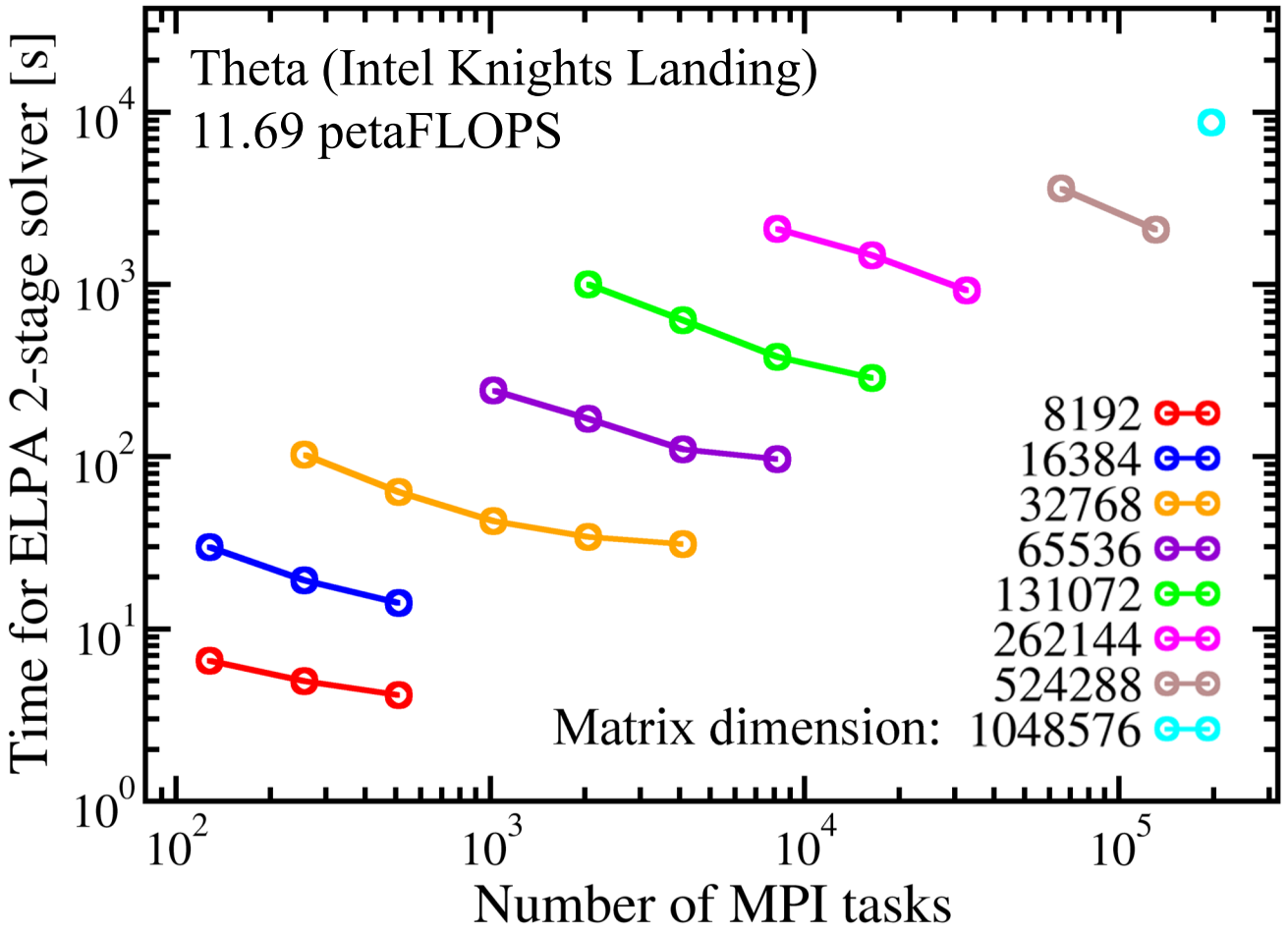
Conventional Solution: Diagonalization



Auckenthaler et al., Parallel Comput. 2011
Marek et al., J. Phys. Condens. Matter 2014
http://elpa.mpcdf.mpg.de

- ELPA eigenvalue solver library: 2-stage diagonalization
- Highly efficient for solving moderately-sized eigenproblems on HPC systems

Performance of ELPA 2-stage solver with AVX512 kernel
Benchmark: Álvaro Vázquez-Mayagoitia, ANL



Eigensolvers and Density Matrix Solvers

Diagonalization (wavefunctions)

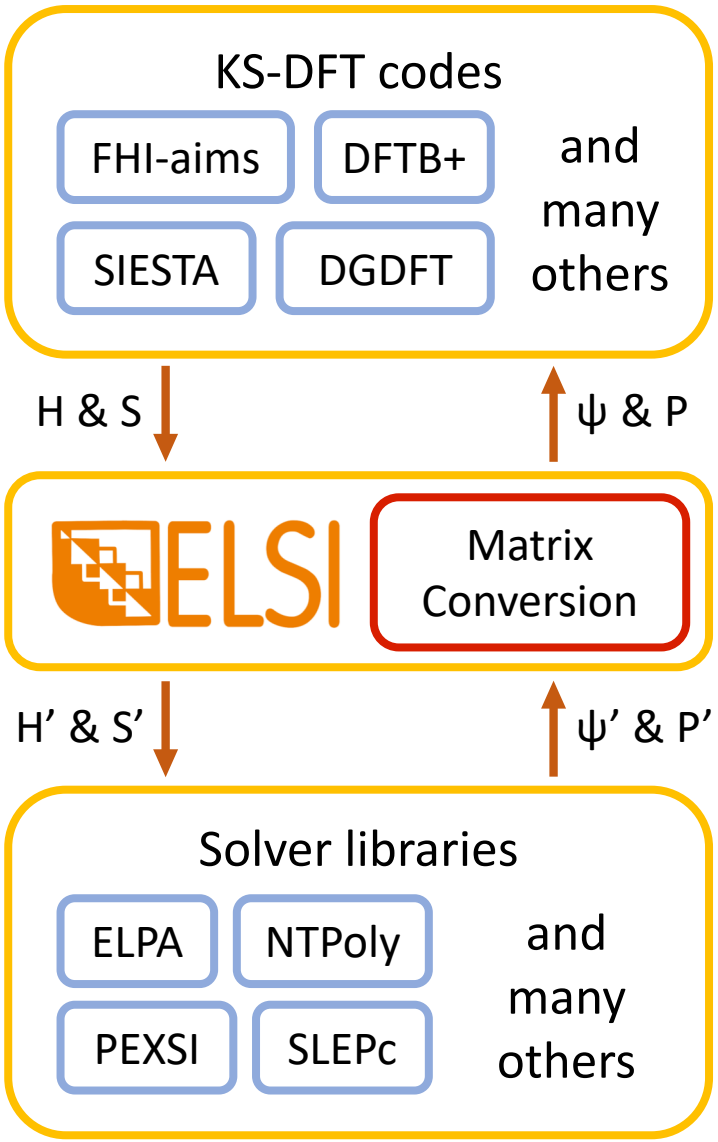
- Efficient for small-to-medium-sized problems
- Applicable to metals, semiconductors, insulators
- $O(N^3)$ time and $O(N^2)$ memory

... and many methods in between ...
(crossover unclear)

Traditional O(N) methods (density matrix)

- Lower scaling factor and memory consumption
- Larger prefactor, overhead for small problems
- Difficulty in treating metallic systems

ELSI: Connection between KS-DFT Codes and Solvers



Yu et al., Comput. Phys. Commun. 2018
http://elsi-interchange.org
http://git.elsi-interchange.org/elsi-devel/elsi-interface

- Unified interface connecting KS-DFT codes and solvers
- Fast, automatic matrix format conversion and redistribution
- Complexity in solver selection for different problems

Benchmark Methods

All-electron KS-DFT
Numeric atom-centered orbitals
Sparse matrices
Blum et al., Comput. Phys. Commun. 2009

Pseudopotential KS-DFT
Numeric atom-centered orbitals
More sparse matrices
Soler et al., J. Phys.: Condens. Matter 2002

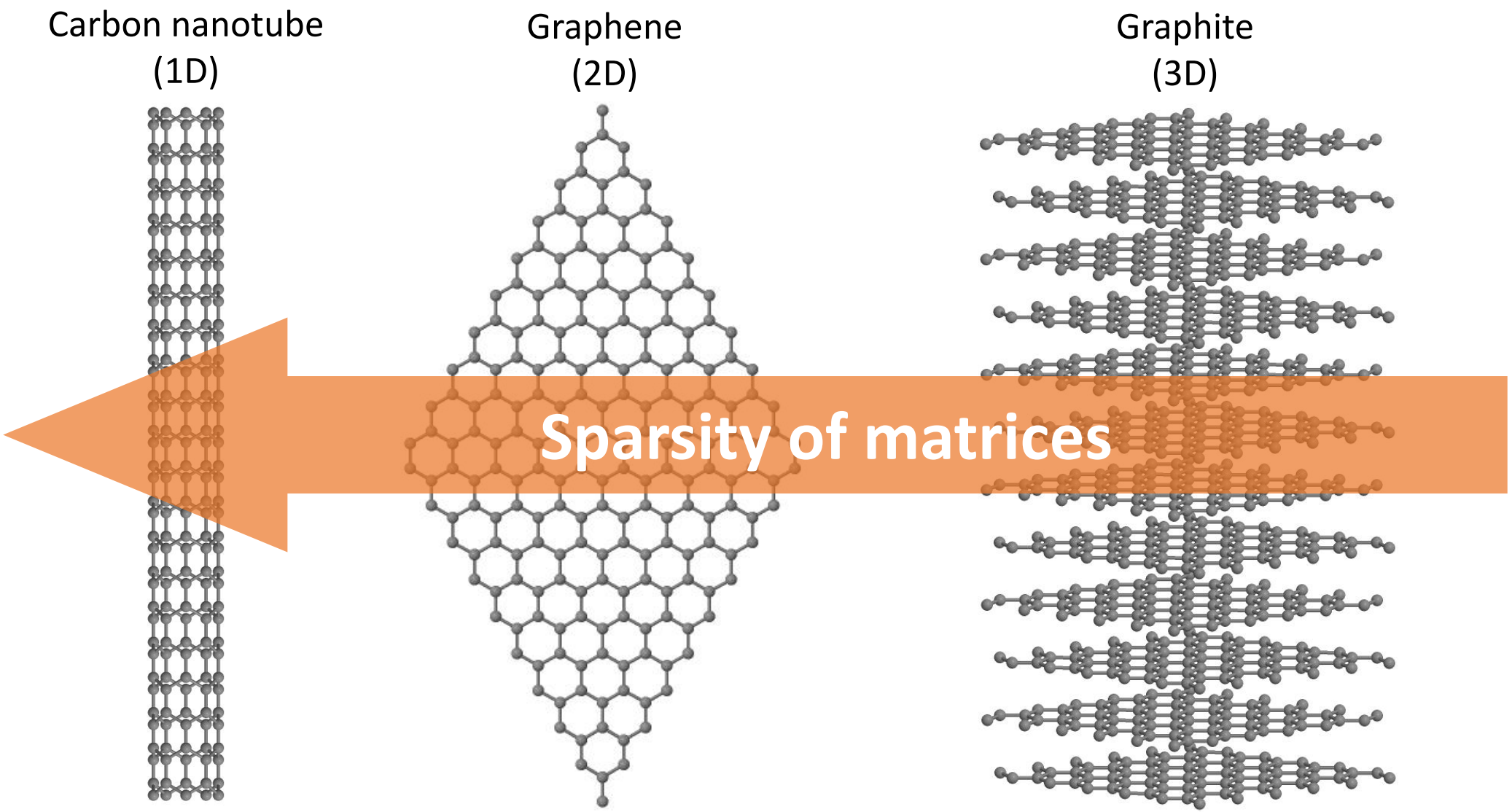
Semi-empirical tight-binding
Atomic orbitals
Highly sparse matrices
Aradi et al., J. Phys. Chem. A 2007

Edison supercomputer
http://www.nersc.gov/edison

Cray XC30
Intel Ivy Bridge

2.57 Petaflops
5,586 compute nodes
134,064 processing cores
(24 cores per node)

Benchmark Methods



Benchmark Methods

- All benchmarks were performed on the Edison supercomputer at the National Energy Research Scientific Computing Center (Berkeley, CA, USA).

- Compute nodes (24 CPU cores) were fully exploited by launching 24 MPI tasks per node. No OpenMP was employed.

- Third-party libraries and tools used in these benchmarks include
  - Compilers: Intel 18.0.1
  - MPI: Cray MPICH 7.6.2
  - BLAS/LAPACK/ScaLAPACK: Intel MKL 2018.1
  - ELPA: 2016.11.001
  - NTPoly: 1.3
  - PEXSI: 1.0.0
  - SLEPc: 3.8.3

- Benchmarks reported here are not official benchmarks of the above third-party libraries. Performance may differ when running different versions of code on different computers.

Pole Expansion and Selected Inversion (PEXSI)

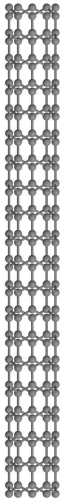
$$\mathbf{P} = \sum_{l} \mathrm{Im}\left(\frac{w_l}{\mathbf{H} - (z_l + \mu)\mathbf{S}}\right)$$

$\mathbf{P}$: Density matrix
$\mathbf{H}$: Hamiltonian matrix
$\mathbf{S}$: Overlap matrix
z_l: Shift (pole)
w_l: Weight
μ: Chemical potential

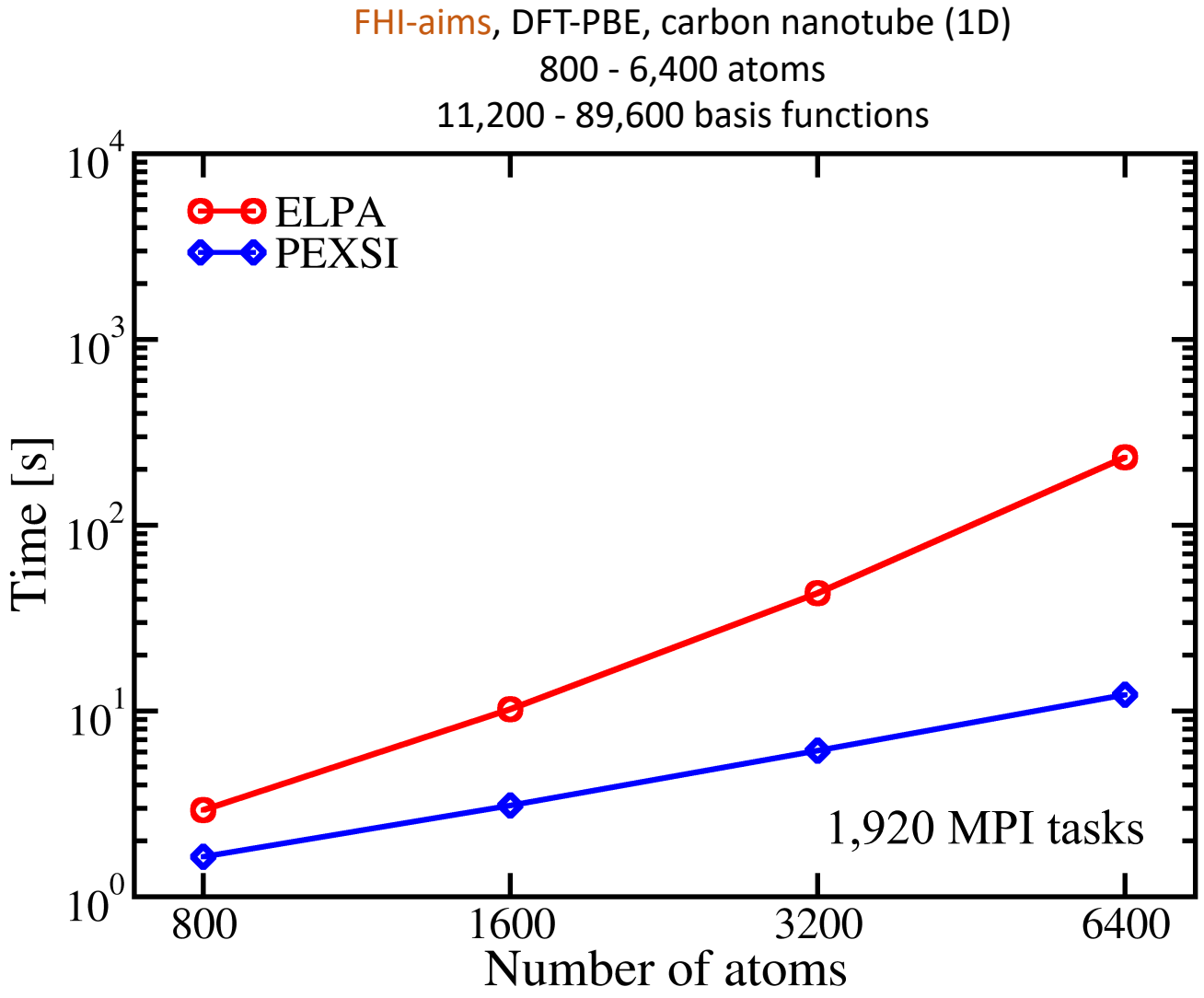
Lin et al., J. Phys. Condens. Matter 2013
Lin et al., J. Phys. Condens. Matter 2014
Jia and Lin., J. Chem. Phys. 2017
http://pexsi.org

- Selected inversion: Evaluate selected elements of $(\mathbf{H} - (z_l + \mu)\mathbf{S})^{-1}$
- Computational cost (semilocal XC):
  - 1D system: O(N)
  - 2D system: $O(N^{1.5})$
  - 3D system: $O(N^2)$
- No dependence on band gap
- Highly scalable: Poles can be evaluated independently in parallel

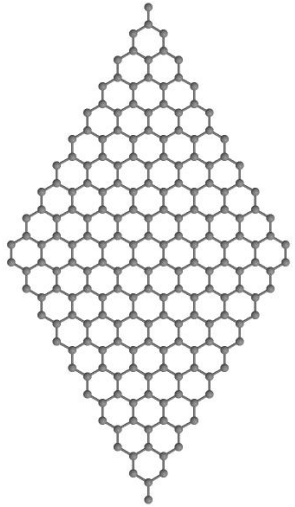
ELPA vs. PEXSI: 1-Dimensional FHI-aims Models



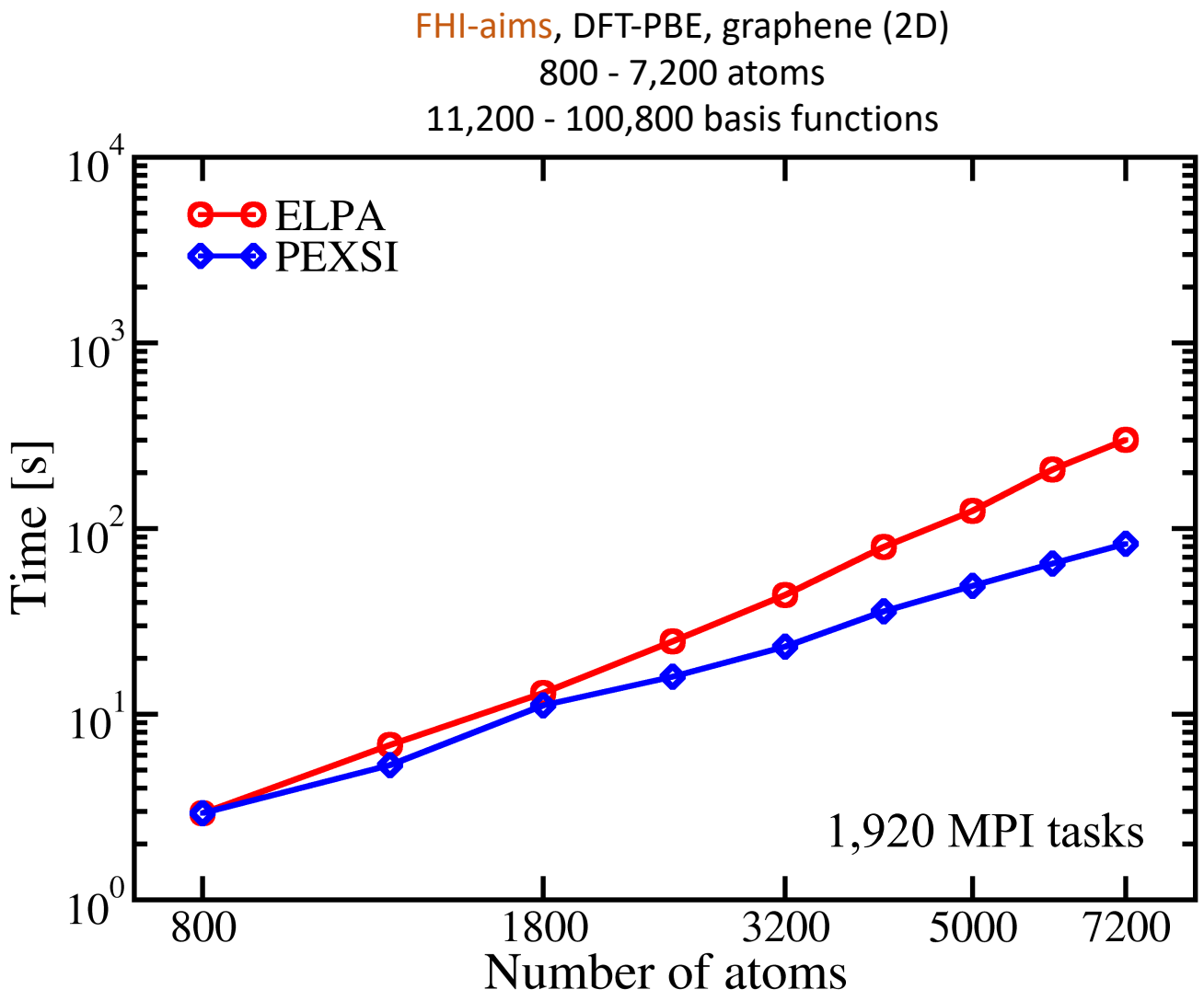
- Sparsity: 96% - 99% zeros
- Theoretical scaling:
  - ELPA: $O(N^3)$
  - PEXSI: O(N) for 1D systems
- PEXSI favorable for 1D systems



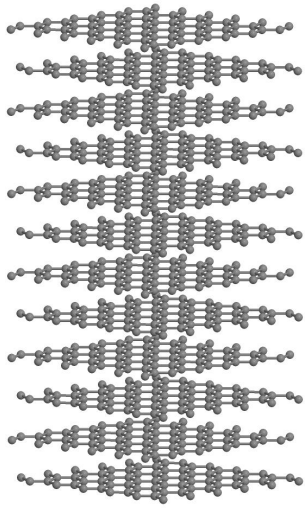
ELPA vs. PEXSI: 2-Dimensional FHI-aims Models



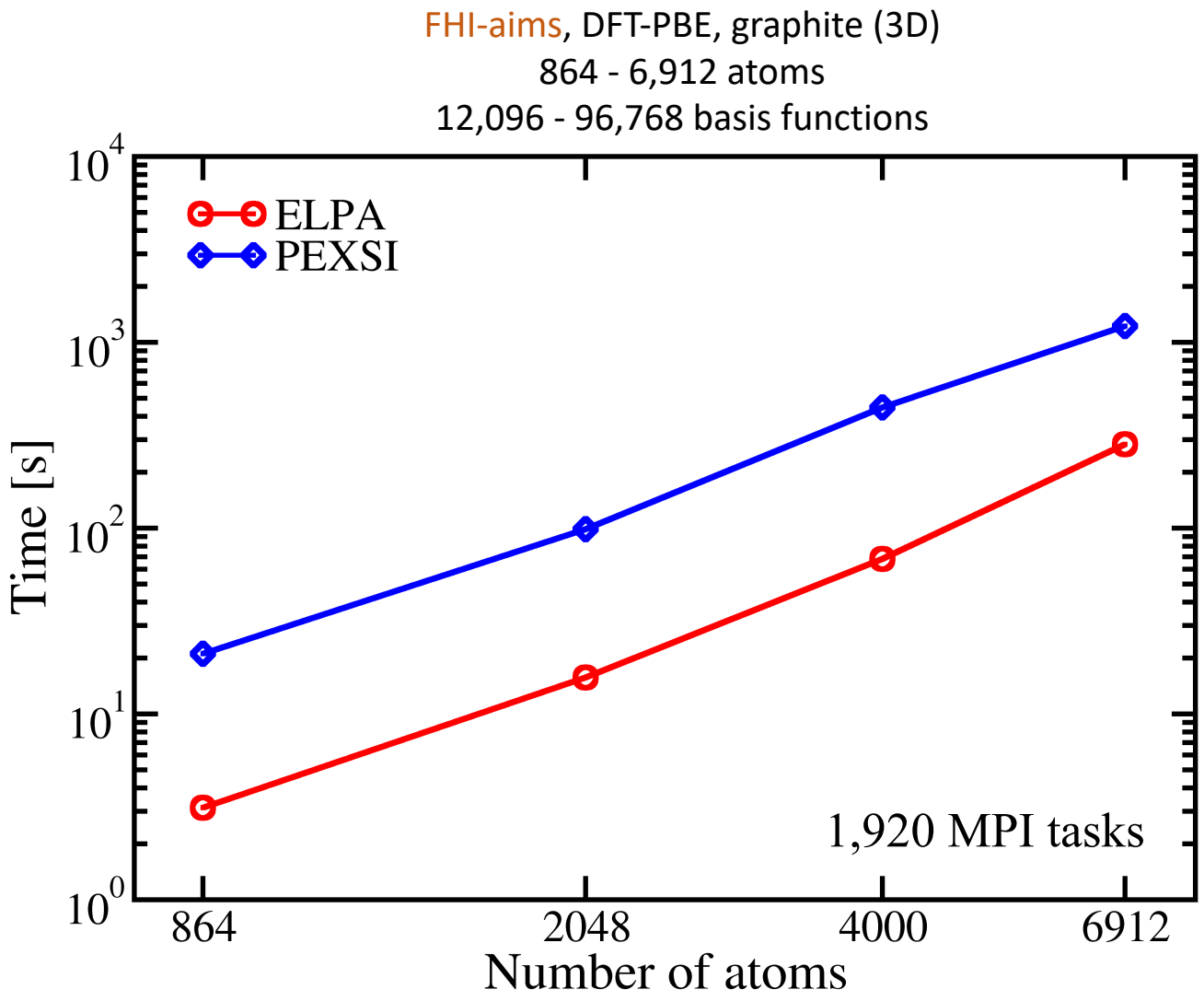
- Sparsity: 91% - 99% zeros
- Theoretical scaling:
  - ELPA: $O(N^3)$
  - PEXSI: $O(N^{1.5})$ for 2D systems
- Crossover: 800 atoms



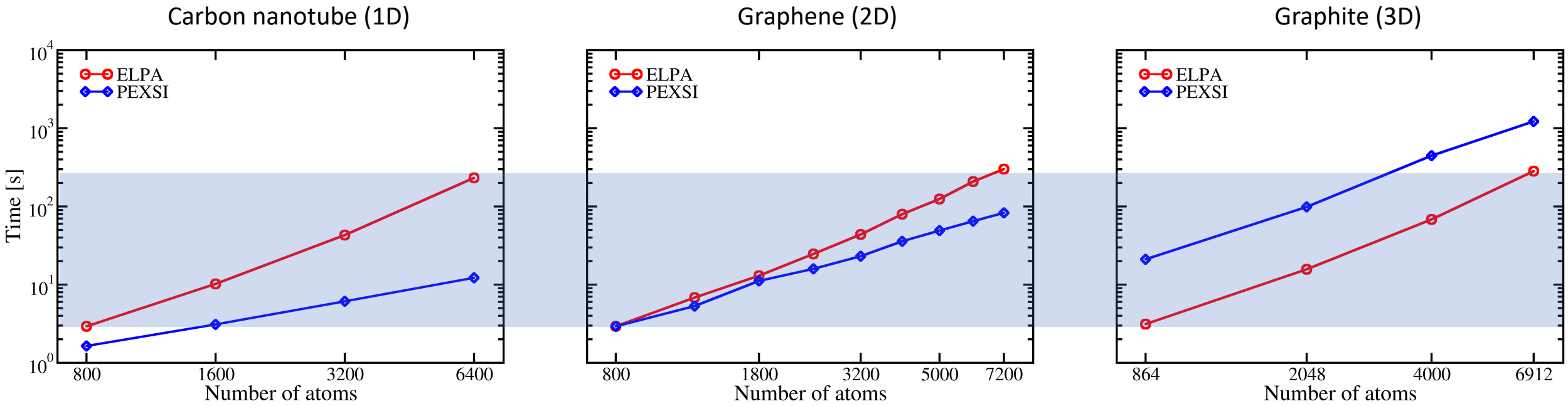
ELPA vs. PEXSI: 3-Dimensional FHI-aims Models



- Sparsity: 74% - 90% zeros
- Theoretical scaling:
  - ELPA: $O(N^3)$
  - PEXSI: $O(N^2)$ for 3D systems
- ELPA favorable for 3D bulk systems



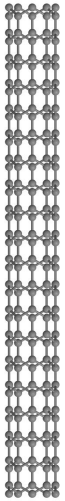
ELPA vs. PEXSI: Dimensionality of Systems



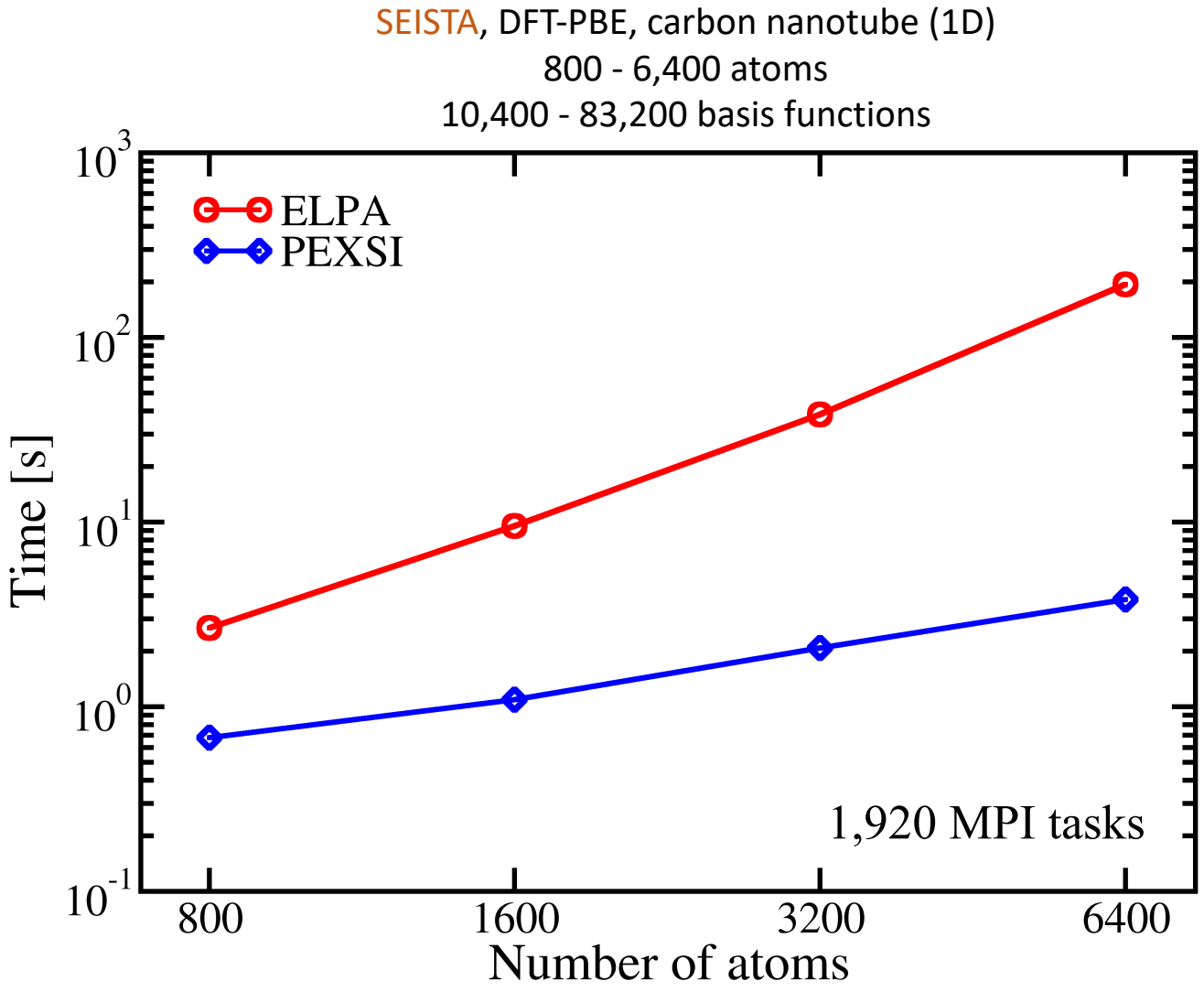
All calculations: FHI-aims, DFT-PBE, 1,920 MPI tasks

- ELPA not dependent on dimensionality
- PEXSI favorable for low-dimensional (sparse) systems

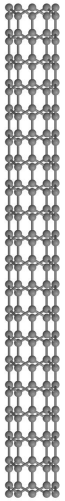
ELPA vs. PEXSI: 1-Dimensional SIESTA Models



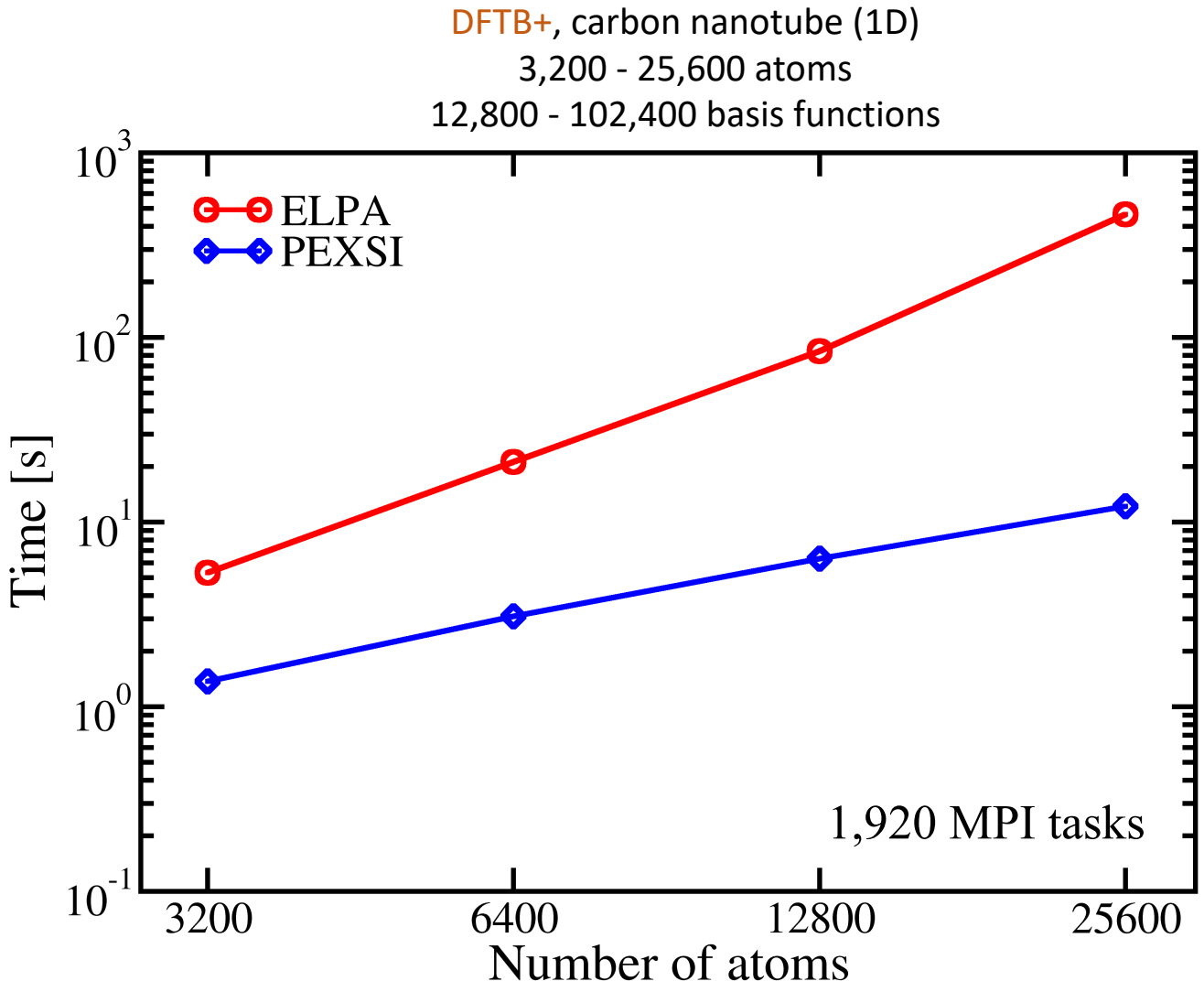
- Sparsity: 97% - 99% zeros
- Same conclusion in FHI-aims, SIESTA, and DFTB+



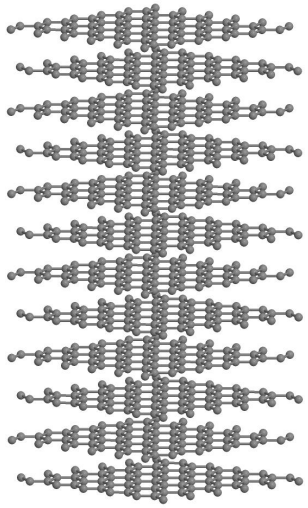
ELPA vs. PEXSI: 1-Dimensional DFTB+ Models



- Sparsity: > 99% zeros
- Same conclusion in FHI-aims, SIESTA, and DFTB+



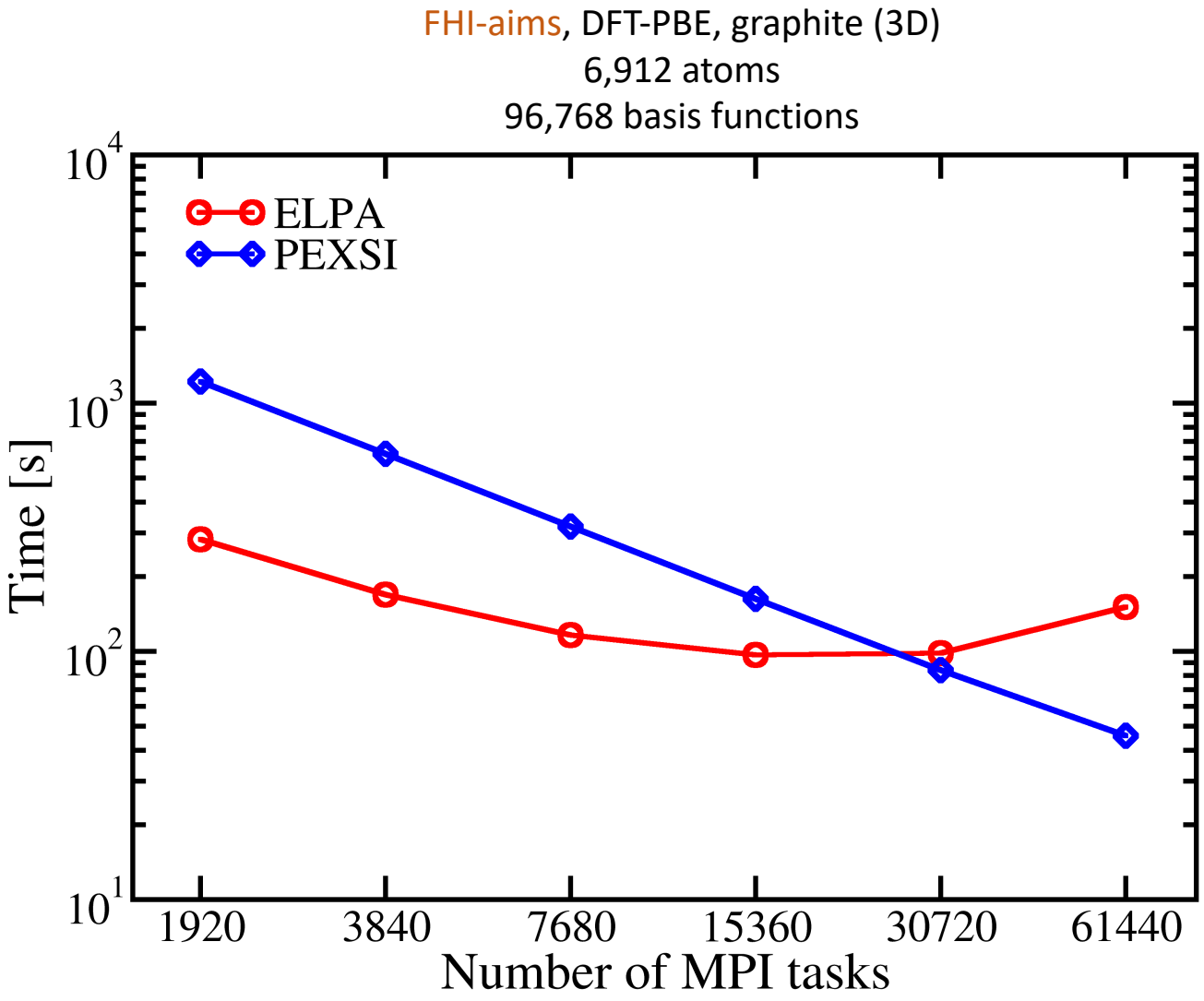
ELPA vs. PEXSI: Parallel Scalability



- Sparsity: 90% zeros

- ELPA: Scales up to ~20k tasks
- PEXSI: Almost ideal scaling

- ELPA: General 3D bulk systems
- PEXSI: Low-dimensional systems with a large number of processors



Shift-and-Invert Parallel Spectral Transformation in SLEPc

$$\mathbf{Hc} = \epsilon\mathbf{Sc}$$

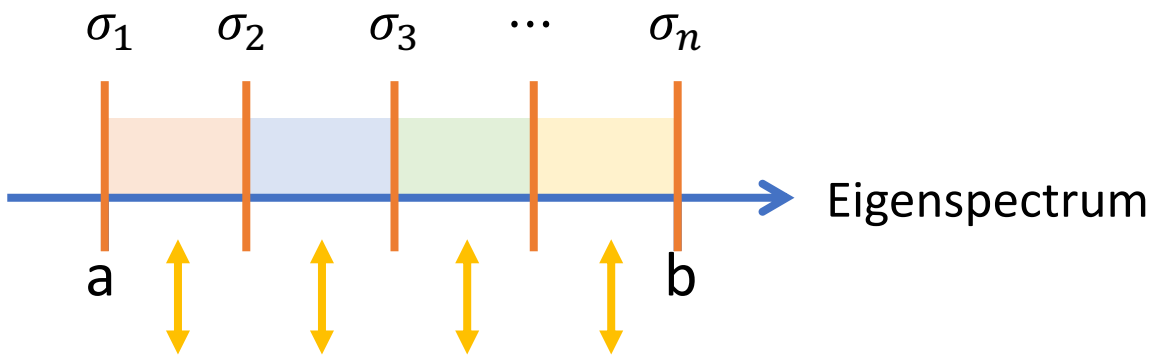
shift

$$(\mathbf{H} - \sigma\mathbf{S})\mathbf{c} = (\epsilon - \sigma)\mathbf{Sc}$$

invert

$$\frac{\mathbf{S}}{\mathbf{H} - \sigma\mathbf{S}}\mathbf{c} = \frac{\mathbf{1}}{\epsilon - \sigma}\mathbf{c}$$

$$\tilde{\mathbf{H}}\mathbf{c} = \tilde{\epsilon}\mathbf{c}$$

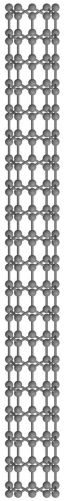


0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

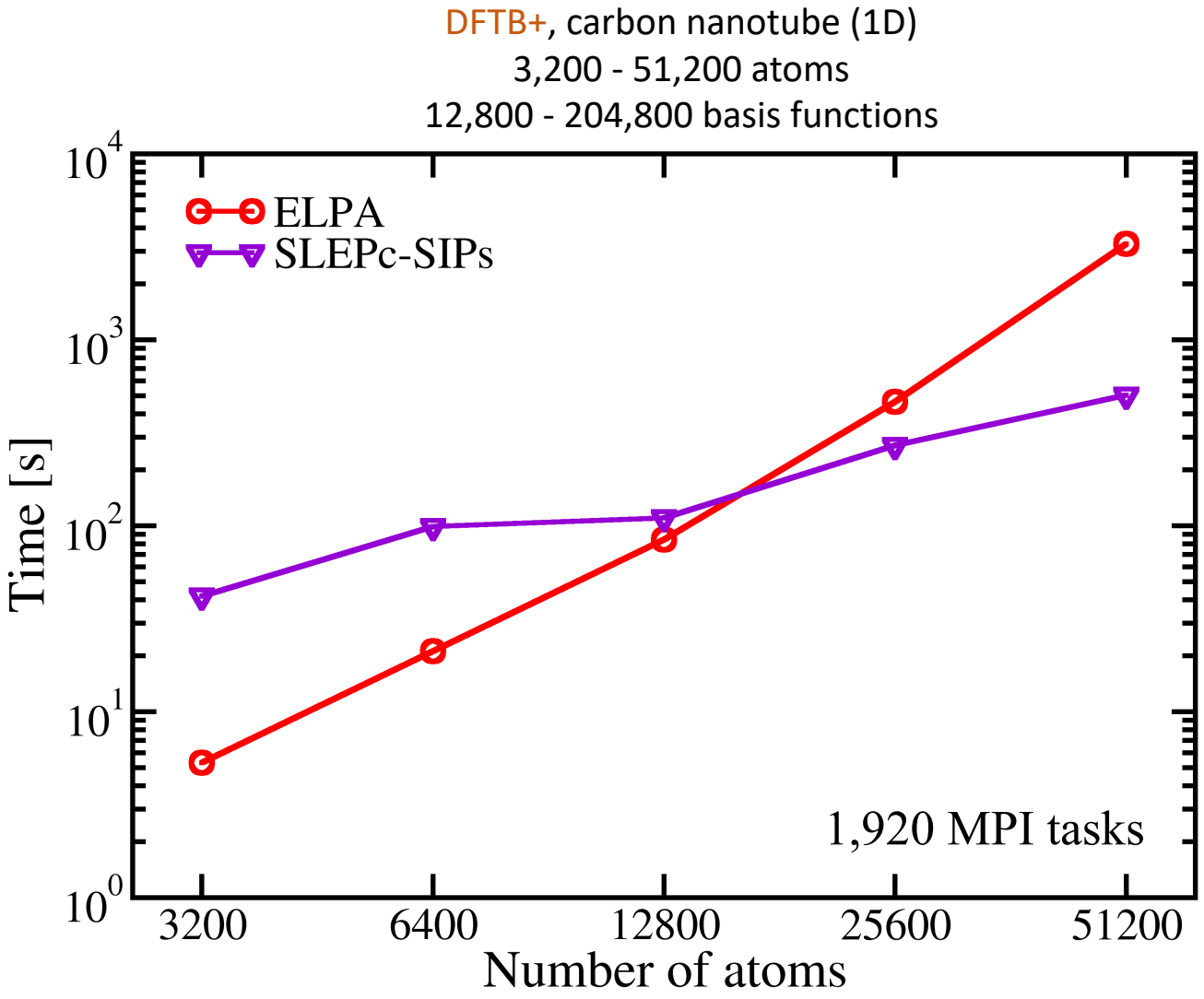
Processors

Hernandez et al., ACM T. Math. Software 2005
Campos and Roman, Numer. Algorithms 2012
Keçeli et al., J. Comput. Chem. 2016
http://slepc.upv.es

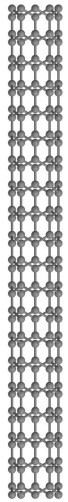
ELPA vs. SLEPc-SIPs: 1-Dimensional DFTB+ Models



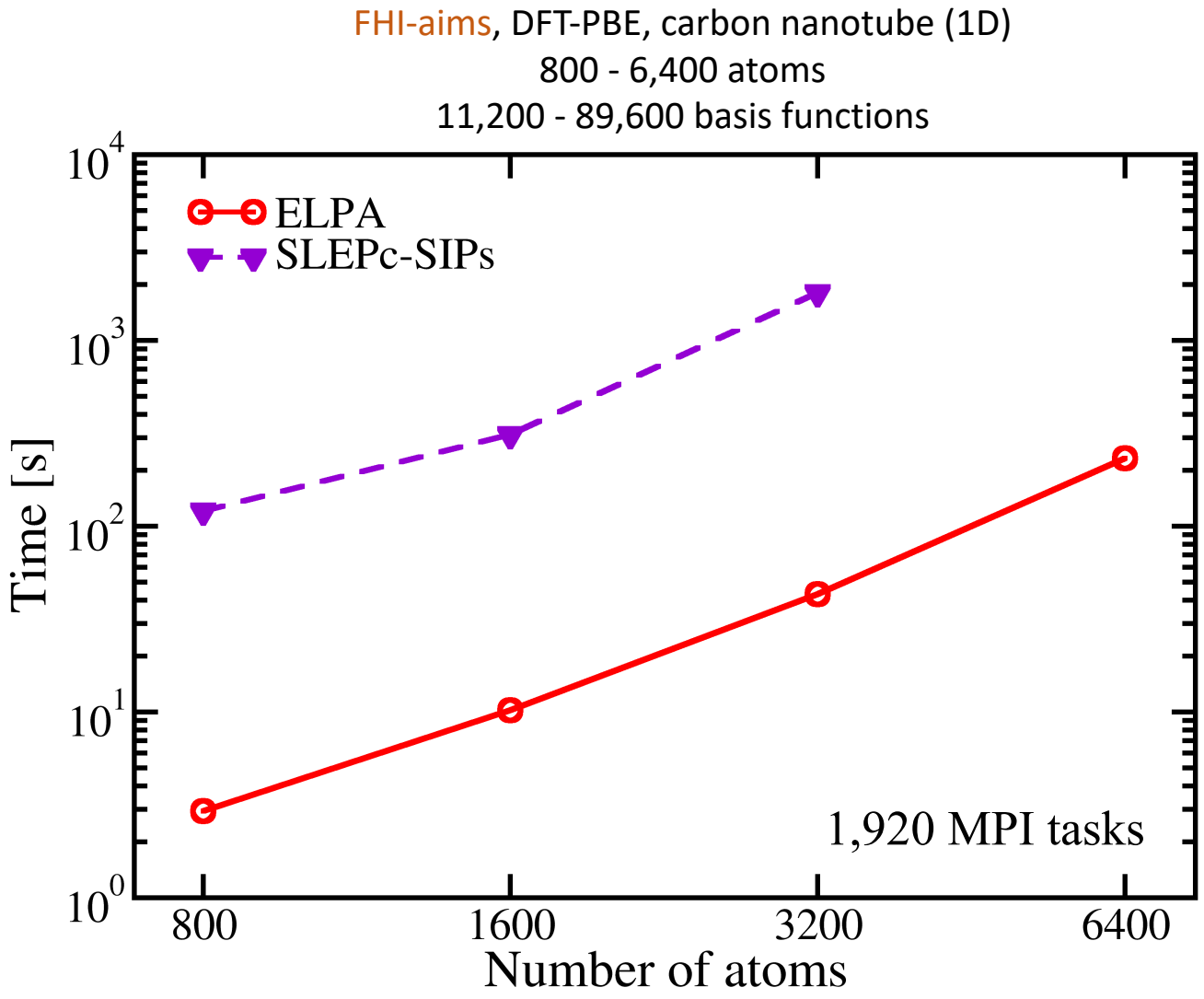
- Sparsity: > 99% zeros
- SLEPc-SIPs: Load balance across slices (MPI tasks) matters a lot



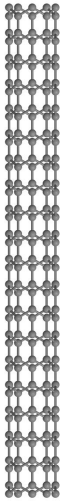
ELPA vs. SLEPc-SIPs: 1-Dimensional FHI-aims Models



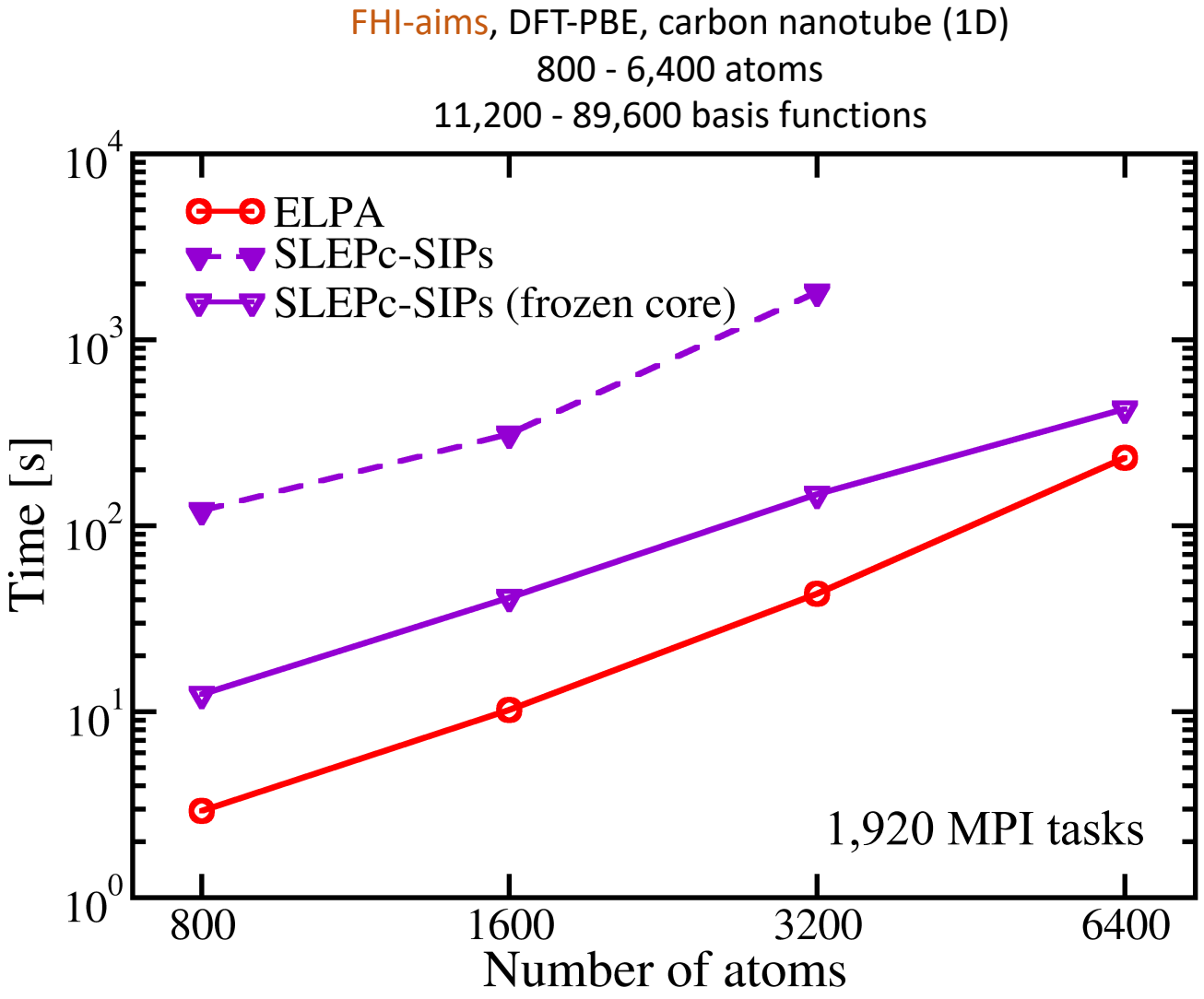
- Sparsity: 96% - 99% zeros
- SLEPc-SIPs: Not competitive due to poor load balance
- Carbon 1s orbitals clustered in a tiny energy interval cannot be efficiently partitioned



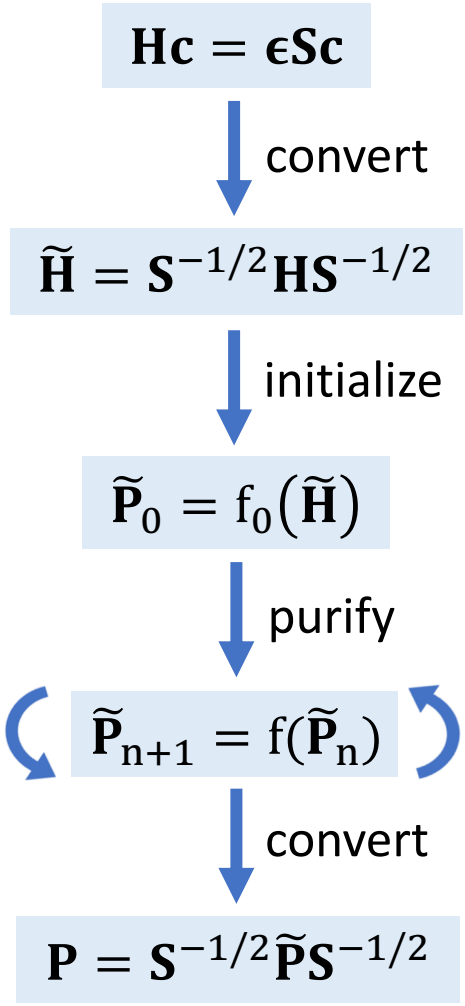
ELPA vs. SLEPc-SIPs: Frozen Core Approximation



- Frozen core approximation: “Freeze” inactive core states; solve valence states only
- SLEPc-SIPs can be ~ 8x faster due to an improved load balance



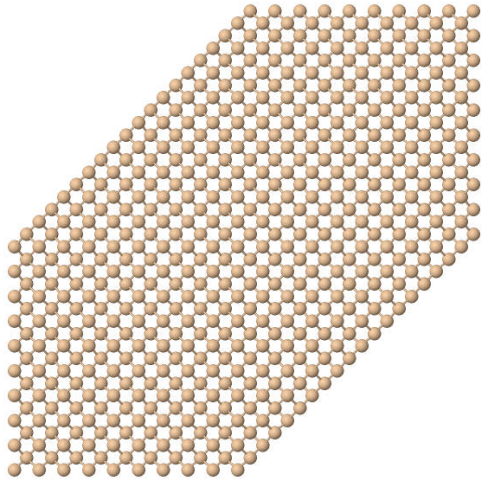
(Preliminary) Density Matrix Purification with NTPoly



- $\widetilde{\mathbf{P}}_{n+1} = \mathrm{f}(\widetilde{\mathbf{P}}_n)$ often matrix polynomial of order m
- $\mathbf{S}^{-1/2}$ and density matrix purification available in NTPoly, powered by its sparse matrix-matrix multiplication kernel
- Currently supported purification algorithms:
  - Canonical purification (m = 3)
  - Trace resetting purification (m = 2, 3, 4, ...)
  - Generalized canonical purification (m = 3)

Dawson and Nakajima, Comput. Phys. Commun. 2018
http://github.com/william-dawson/NTPoly
Palser and Manolopoulos, Phys. Rev. B 1998
Niklasson, Phys. Rev. B 2002
Truflandier et al., J. Chem. Phys. 2016

Computation of Matrix Inverse Square Root

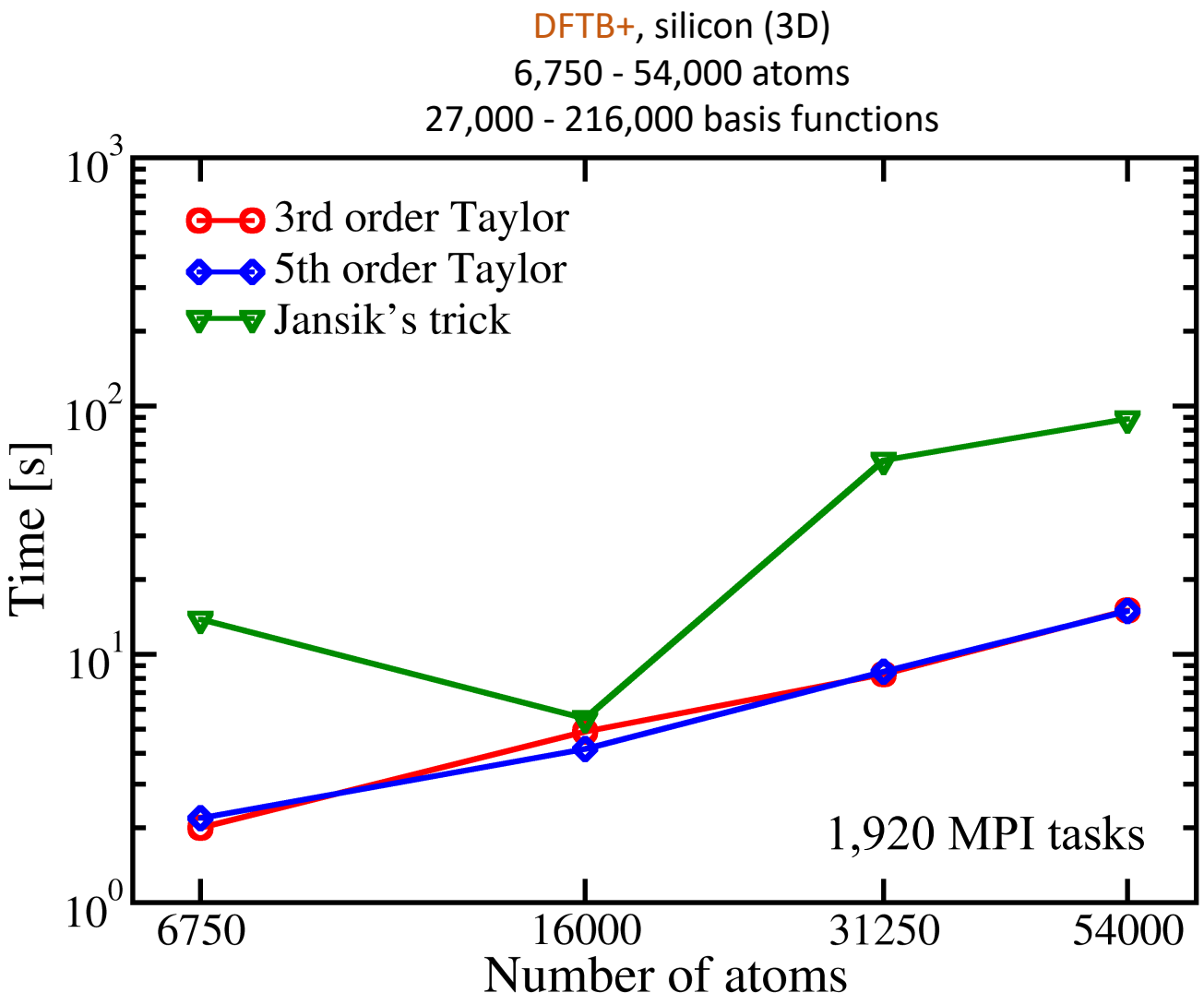


Newton-Schulz method + Taylor expansion

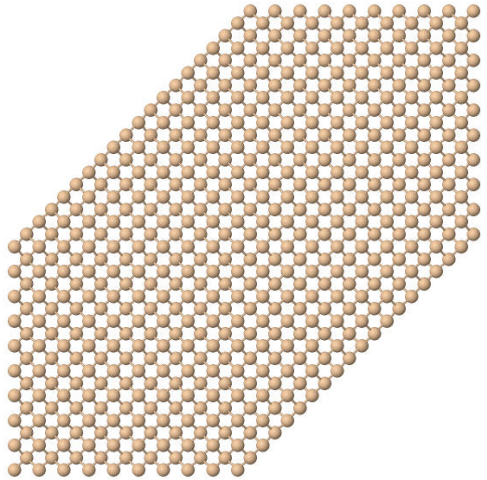
$$(\mathbf{I}-\mathbf{X})^{-1/2}=\mathbf{I}+\frac{1}{2}\mathbf{X}+\frac{3}{8}\mathbf{X}^2+\frac{5}{16}\mathbf{X}^3+\cdots$$

Convergence: $\|\mathbf{X}\|_2=\|\lambda\mathbf{S}-\mathbf{I}\|_2\leq 1$

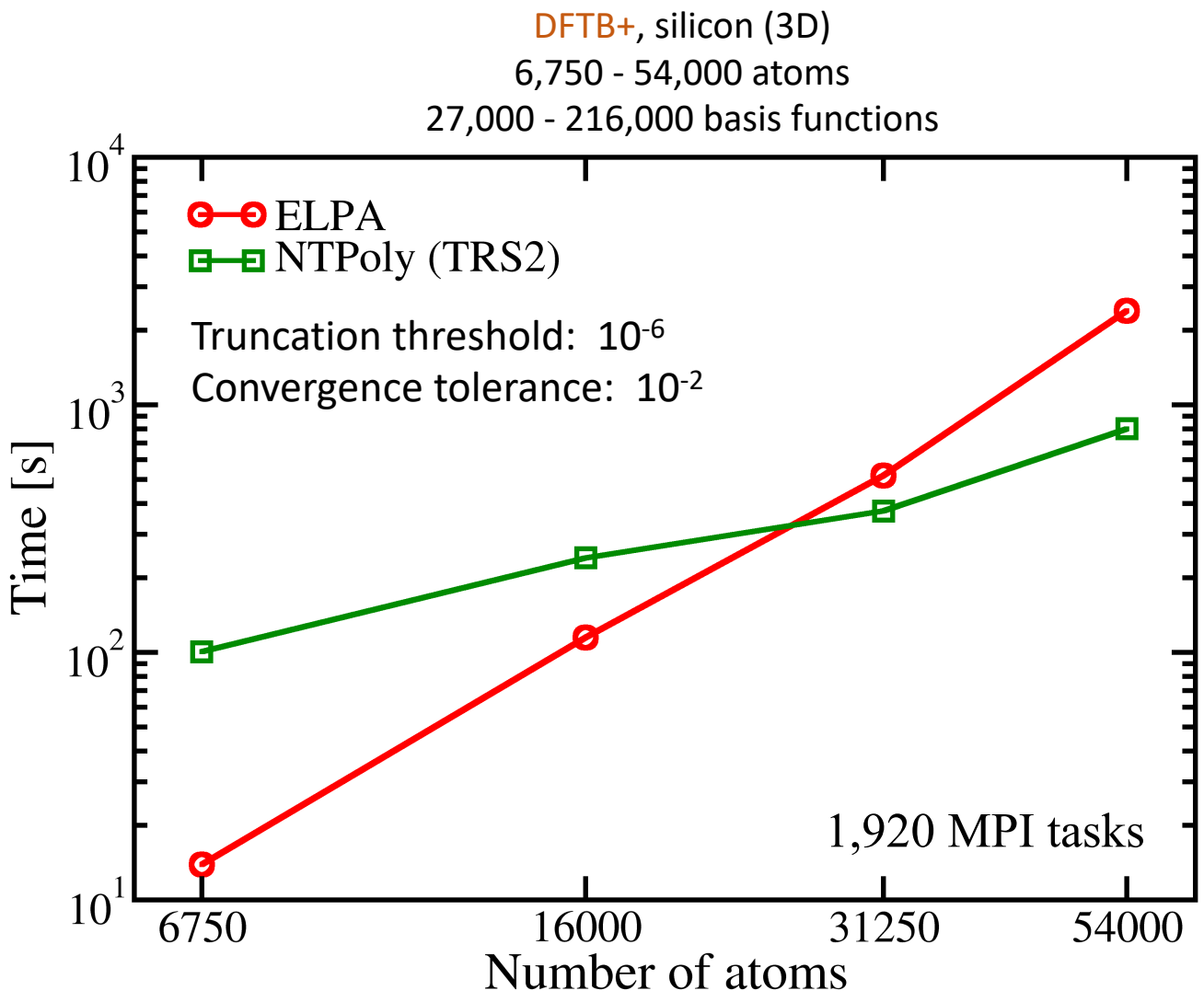
Higham, Numer. Algorithms 1997
Niklasson, Phys. Rev. B 2004
Jansík et al., J. Chem. Phys. 2007



ELPA vs. NTPoly: 3-Dimensional DFTB+ Models



- Sparsity: > 99% zeros
- Theoretical scaling:
  - ELPA: $O(N^3)$
  - NTPoly: $O(N)$
- Tests with other purification algorithms ongoing



Conclusions and Outlook

ELSI offers a seamless connection between electronic structure software and a variety of solver libraries.

- Solvers: ELPA, LAPACK, libOMM, NTPoly, PEXSI, SLEPc-SIPs
- Parallel matrix I/O, matrix visualization, chemical potential, ...

Based on our benchmarks and analysis, we recommend:

- An optimized dense eigensolver (ELPA) for small-to-medium-sized calculations;
- PEXSI for large, low-dimensional geometries;
- (Work-in-progress) Density matrix purification for large, bulk, gapped systems.

Future work directions include:

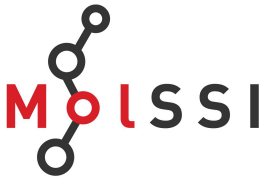
- More solvers: Iterative solvers (RCI), FEAST, Chebyshev filtering, and more!
- More users: Integration of ELSI into more electronic structure code projects.

Acknowledgments

http://elsi-interchange.org

Developers and contributors:

Volker Blum (Duke Univ.)
Danilo Brambila (FHI)
Christian Carbogno (FHI)
Fabiano Corsetti (QuantumWise)
William Dawson (RIKEN)
Alberto García (ICMAB-CSIC)
Stefano de Gironcoli (SISSA)
Ville Havu (Aalto Univ.)
Ben Hourahine (Strathclyde Univ.)
William Huhn (Duke Univ.)
Mathias Jacquelin (LBL)
Weile Jia (LBL)
Murat Keçeli (ANL)
Florian Knoop (FHI)
Björn Lange (Duke Univ.)
Raul Laasner (Duke Univ.)
Yingzhou Li (Duke Univ.)
Lin Lin (LBL)
Jianfeng Lu (Duke Univ.)
Wenhui Mi (Duke Univ.)
Stephan Mohr (BSC)
Jose Roman (UPV)
Ali Seifitokaldani (Duke Univ.)
Álvaro Vázquez-Mayagoitia (ANL)
Chao Yang (LBL)
Haizhao Yang (Duke Univ.)
Victor Yu (Duke Univ.)



ELSI is an NSF SI2-SSI supported software infrastructure project under Grant Number 1450280. Any opinions, findings, and conclusions or recommendations expressed here are those of the author(s) and do not necessarily reflect the views of NSF.