The University of Texas at Austin
McKetta Department of Chemical Engineering
Cockrell School of Engineering

A22.6

A Space-filling Curve Based Grid Partitioning to Accelerate Real-space Pseudopotential Density Functional Theory Calculations

Ariel Biller[1], **Kai-Hsin Liou[2]**, Deena Roller[1], Leeor Kronik[1], James Chelikowsky[2]

[1] Weizmann Institute of Science

[2] The University of Texas at Austin

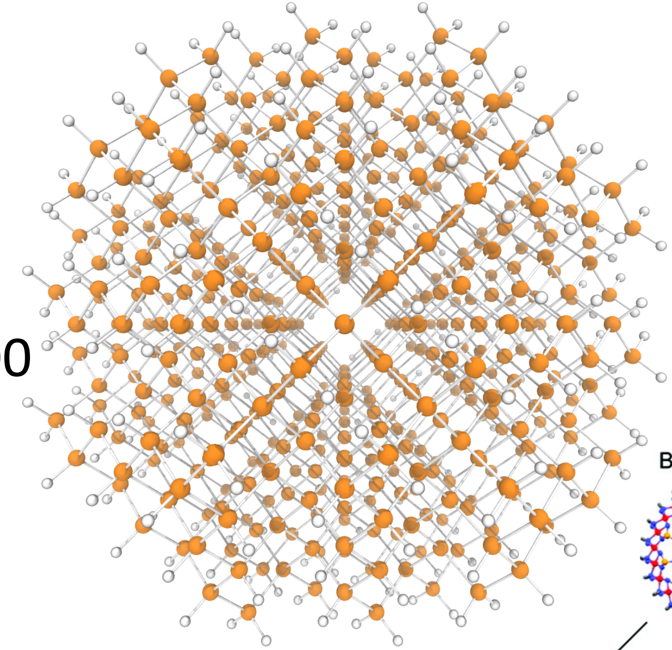
TEXAS
The University of Texas at Austin

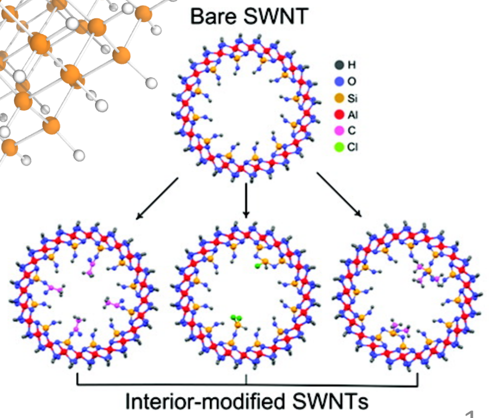
מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

APS March Meeting
March 2, 2020

Motivation

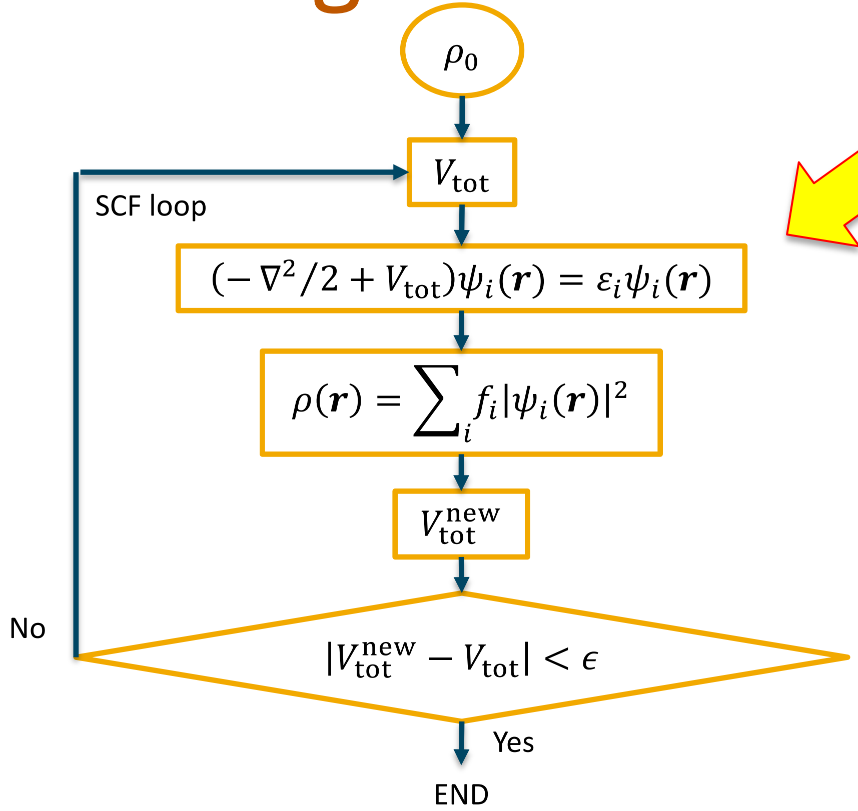
- Silicon nanocrystals
  - Optoelectronics
  - 10 nm in diameter ~ 30,000 atoms
- Inorganic nanotubes
  - Catalysis
  - Water desalination
  - 20 nm in length ~ 5,000 atoms





[1] D.-Y. Kang, J. Zang, C. Jones, S. Nair, Single-Walled Aluminosilicate Nanotubes with Organic-Modified Interiors, *J Phys Chem C* **115**, 7676–7685 (2011).

Solving the Kohn–Sham Equations



Key to large systems:

An efficient eigensolver

Observation: intermediate SCF iterations require only converged charge density, not individual wfns

Chebyshev-filtered Subspace Iteration (CheFSI)

1. Filtering
2. Orthonormalization
3. Rayleigh–Ritz method

Many MATVECs Are Needed for Filtering

- Sparse matrix-vector multiplications in PARSEC
- Cost ~ O(*N s m*)

  N: # of grid points
  s: # of states
  m: degree of the filter

- Faster and better scalability would be nice 🙂

Algorithm 1 Chebyshev filtering

```
procedure W = ChebyFilter(H, V, m, λ_F, λ_ub, λ_lb)
    e = (λ_ub − λ_F)/2
    c = (λ_ub + λ_F)/2
    σ = e/(c − λ_lb)
    τ = 2/σ
    W = (HV − cV)(σ/e)
    for i = 2 → m do
        σ_new = 1/(τ − σ)
        W_t = (HV − cV)(2σ_new/e) − (σσ_new)V
        V = W
        W = W_t
        σ = σ_new
```

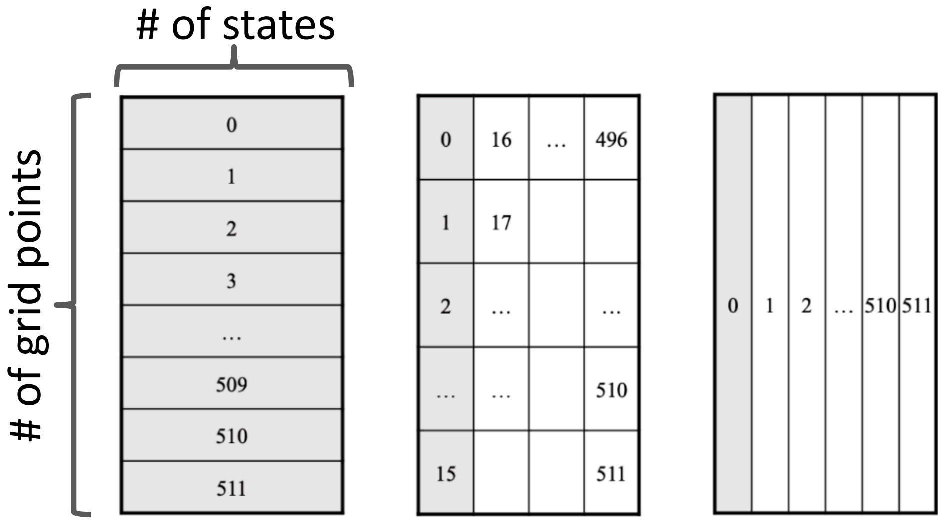
[1] PARSEC (**P**seudopotential **A**lgorithm for **R**eal-**S**pace **E**lectronic Structure **C**alculation), https://real-space.org/

Strategies

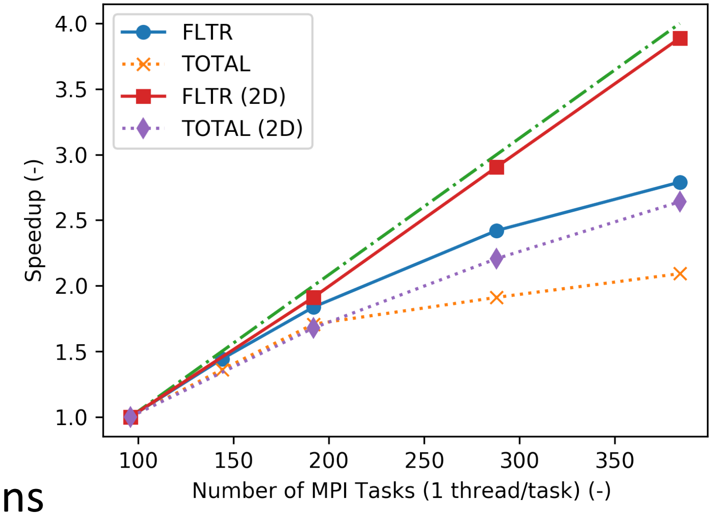
1. Two-dimensional process grid
2. Space-filling curve based grid partitioning
3. Bigger data exchange unit

Two-dimensional Process Grid

Providing flexibility on parallelizing over rows and states



Different process grids on top of the wave functions



Space-filling Curves

- One-dimensional representation of multi-dimensional space
- Self-similarity
- We use Hilbert space-filling curves

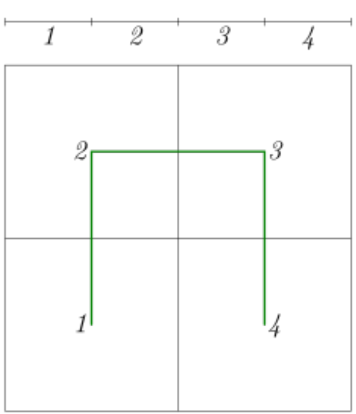


Fig. 1.

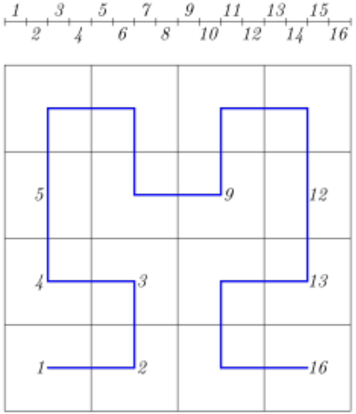


Fig. 2.

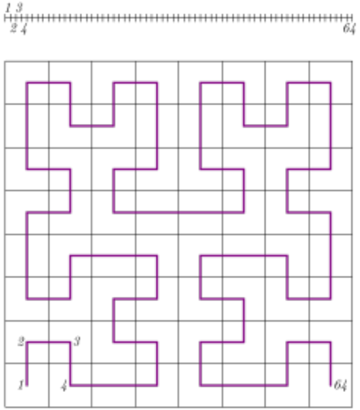
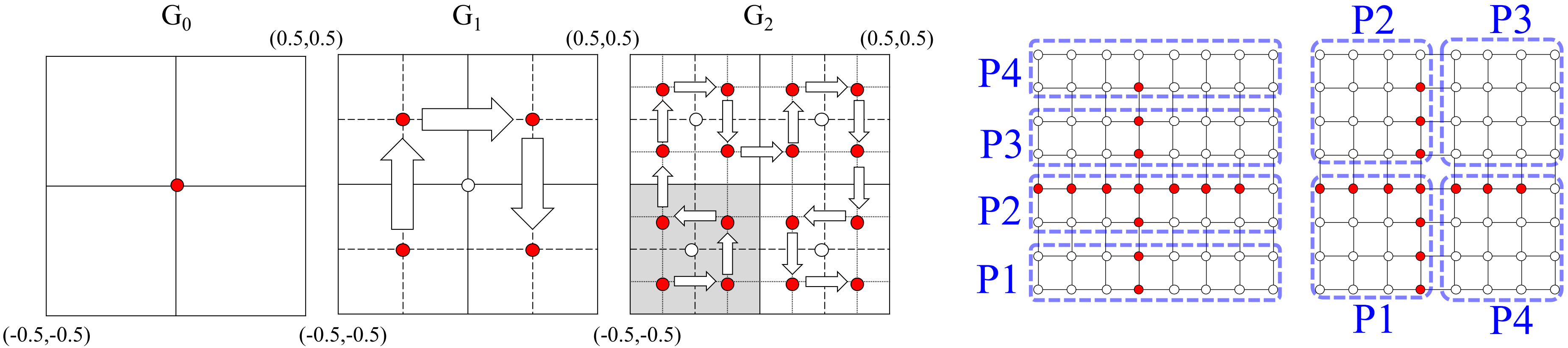


Fig. 3.

From Wikipedia

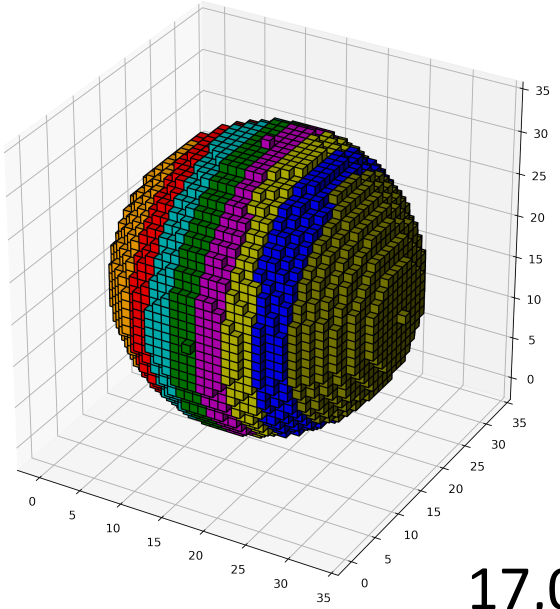
Space-filling Curve Based Grid Partitioning

- A natural way for domain decomposition
- Good locality of grid points.
- Less and more balanced communication between domains

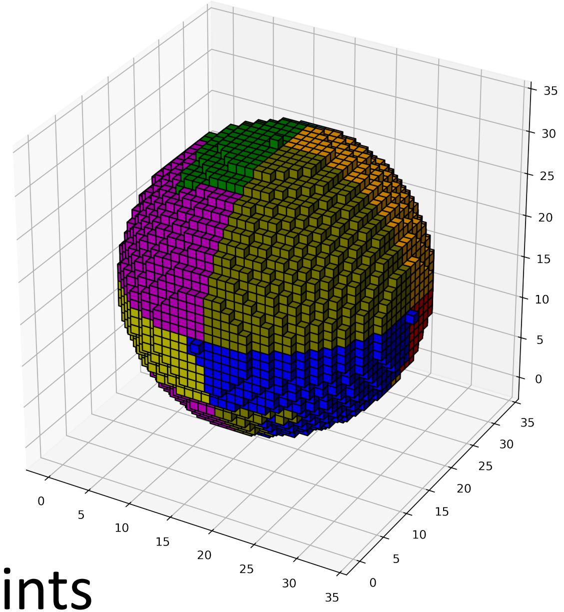


Simple vs SFC-based Grid Partitioning

Simple



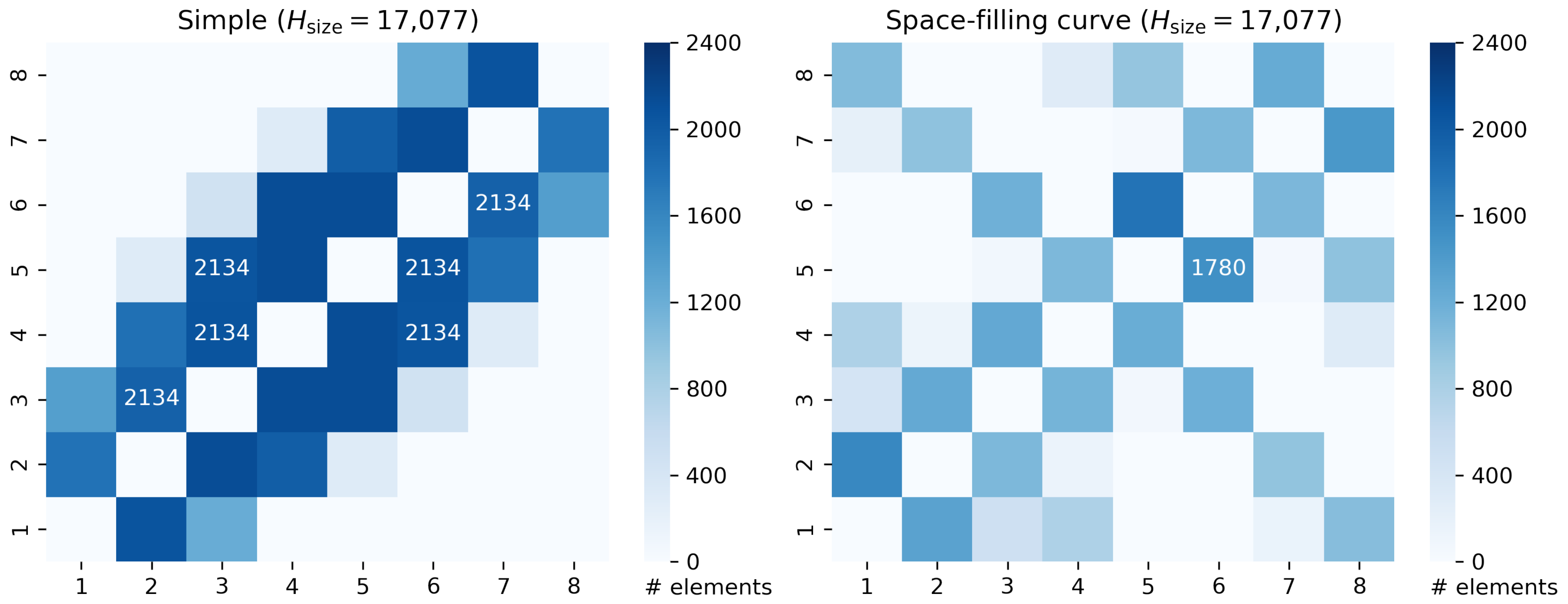
Space-filling curve



17,077 grid points

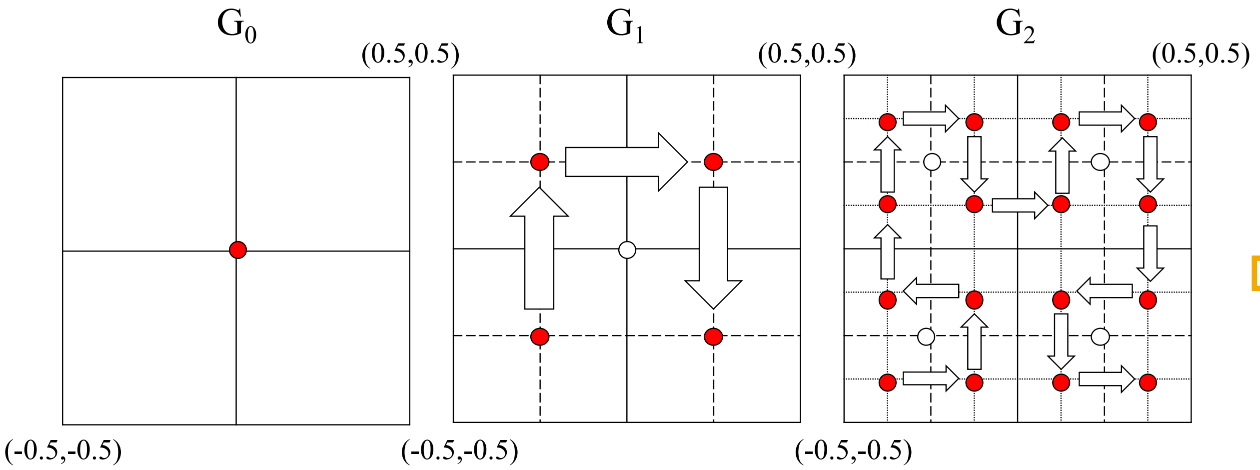
8 MPI ranks

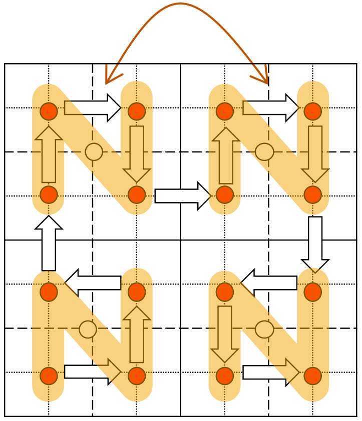
Communication Pattern



Bigger Data Exchange Units

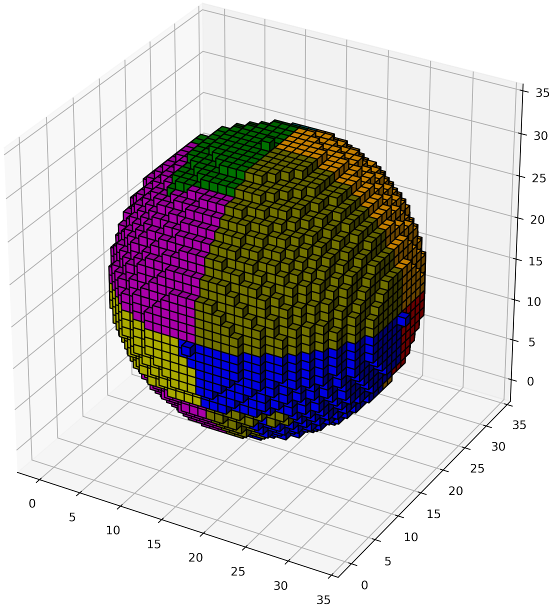
- Processing multiple grid points at the same time
- Better cache reuse and helping vectorization





Different Block Sizes

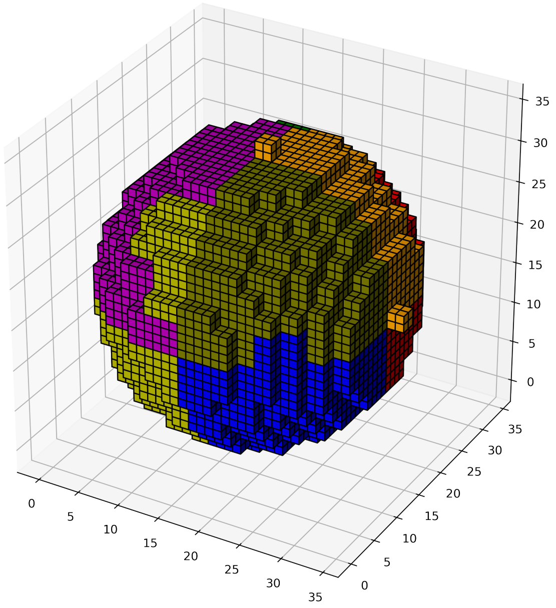
1×1×1



17,077 grid points

2×2×2

8 grid points per block



19,568 grid points

Preliminary Results

1. 2D process grid
2. Space-filling curve based grid partitioning
3. Bigger data exchange units

	H_{size}	Time for filtering (sec)
Simple	2,225,667	56
SFC, 1x1x1	2,225,667	49
SFC, 2x2x2	2,287,936	34

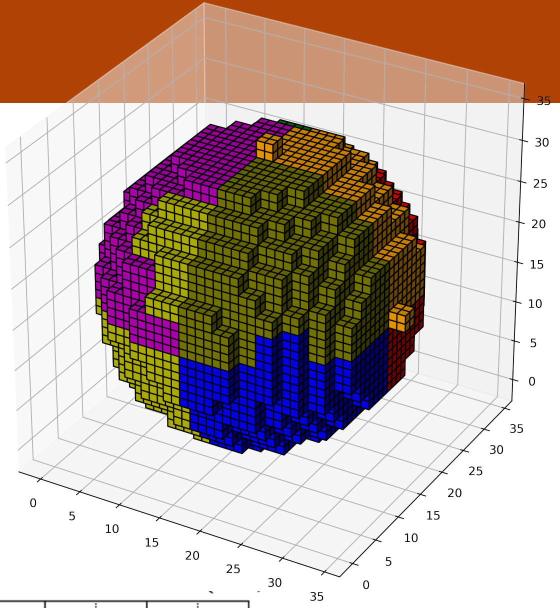
14% faster

44% faster

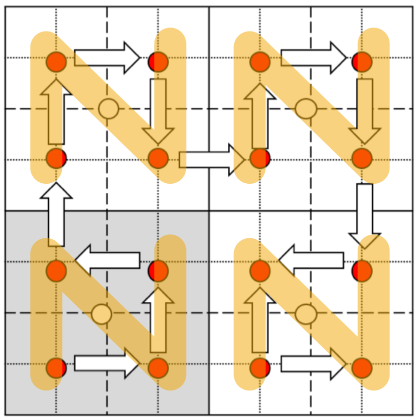
* Performed on Intel Xeon E5-2623 v3 3GHz, 8 physical cores

** Degree of the Chebyshev filters = 20

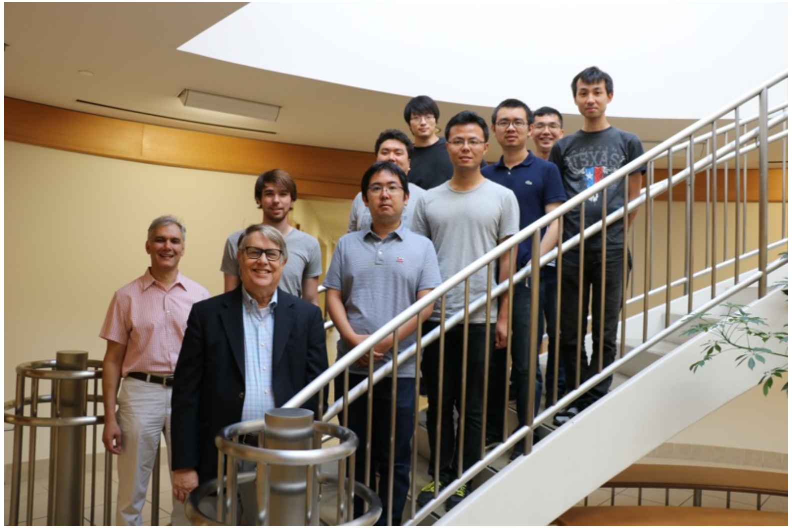
Summary



- Strategies to improve the **scalability** of the sparse matrix-vector multiplications
  - 2D process grid
  - Space-filling curve based grid partitioning
  - Bigger data exchange units
- **Reduced communication** between MPI ranks and better use of **vectorization**



Acknowledgements



Center for Computational Materials

Ariel Biller

U.S. DEPARTMENT OF
ENERGY

Office of Science

TACC

NERSC