

Performance optimization of WEST and Qbox on Intel Knights Landing

Huihuo Zheng¹, Christopher Knight¹, Giulia Galli^{1,2},
Marco Govoni^{1,2}, and Francois Gygi³

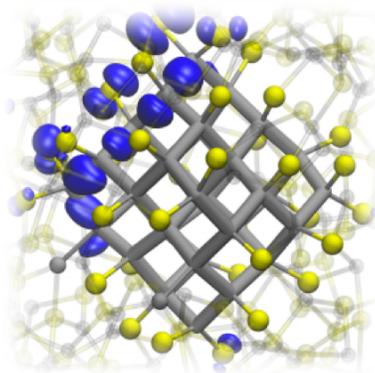
¹Argonne National Laboratory

²University of Chicago

³University of California, Davis

September 27th, 2017

ALCF Theta Early Science Project: First-Principles Simulations of Functional Materials for Energy Conversion

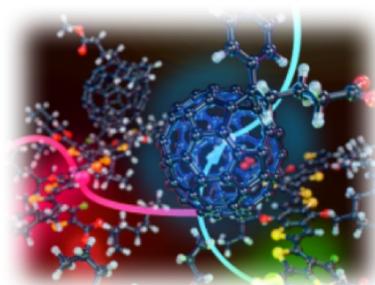
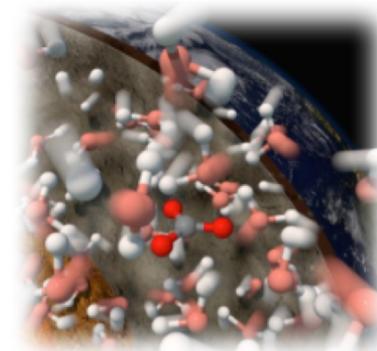


Embedded nanocrystal

T. Li, Phys. Rev. Lett. **107**, 206805 (2011)

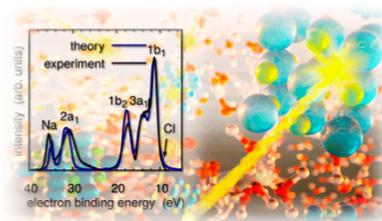
Matters at extreme conditions

D. Pan et al Proc. Nat'l Acad. Sci. **110**, 6250 (2013)



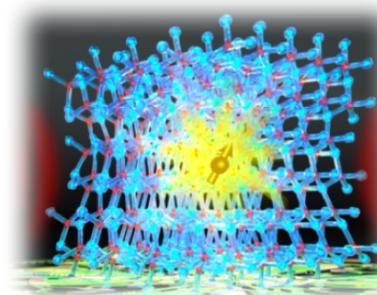
Organic photovoltaics

M. Goldey Phys. Chem. Chem. Phys., Advance Article (2016)



Aqueous solution

A. Gaiduk et al., J. Am. Chem. Soc. Comm. (2016)



Quantum information

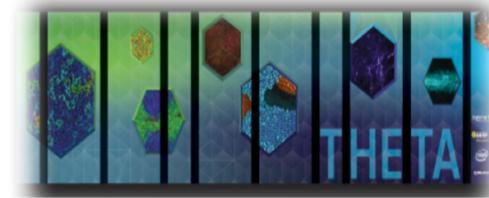
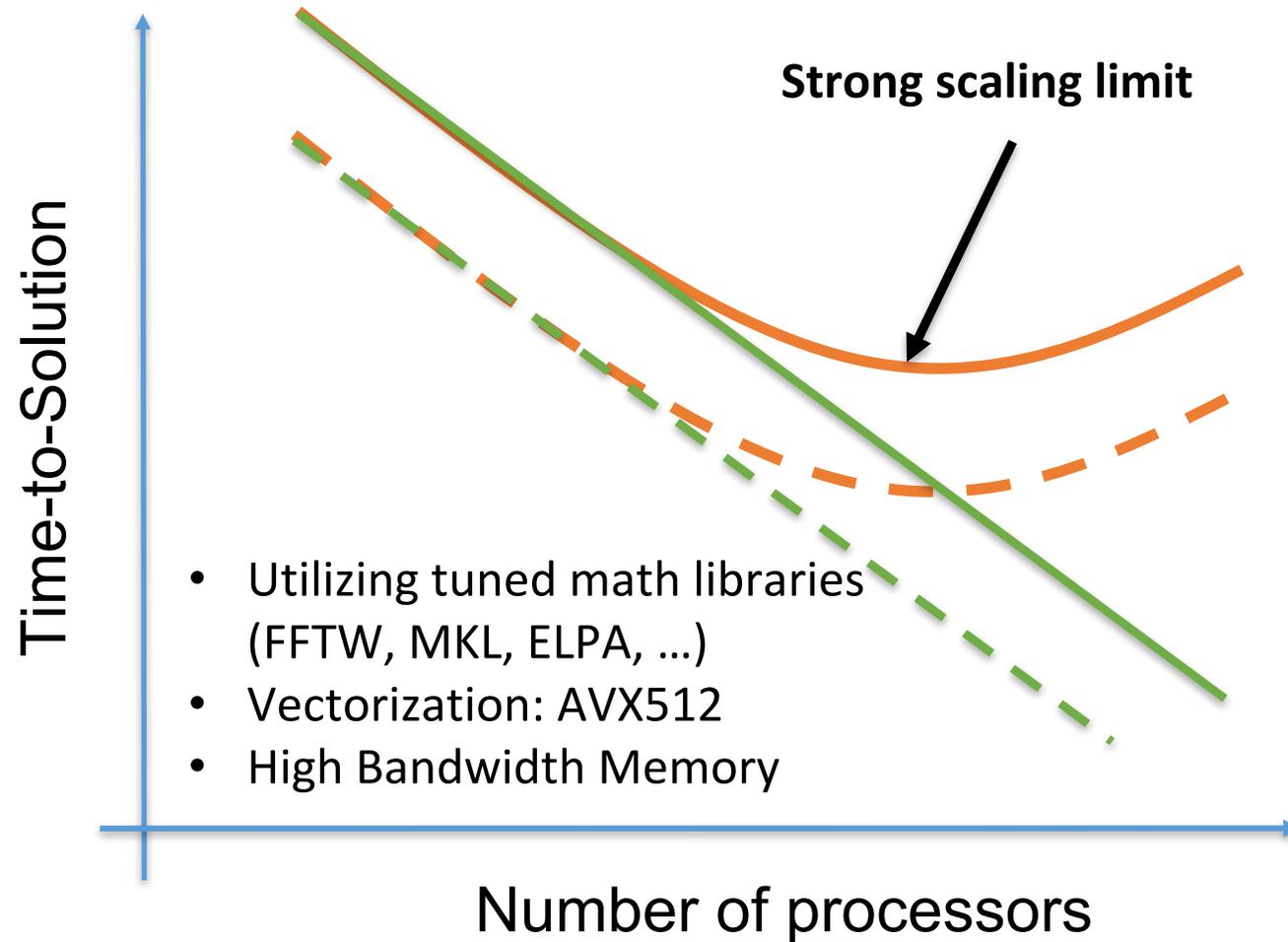
H Seo, Sci Rep. 2016; 6: 20803.

<http://qboxcode.org/>; <http://west-code.org/>; <http://www.quantum-espresso.org/>

M. Govoni, G. Galli, *J. Chem. Theory Comput.* 2015, 11, 2680–2696

P. Giannozzi, et al *J.Phys.:Condens.Matter*, 21, 395502 (2009)

Optimization focus



3,624 KNL nodes, 9.65petaFLOPS

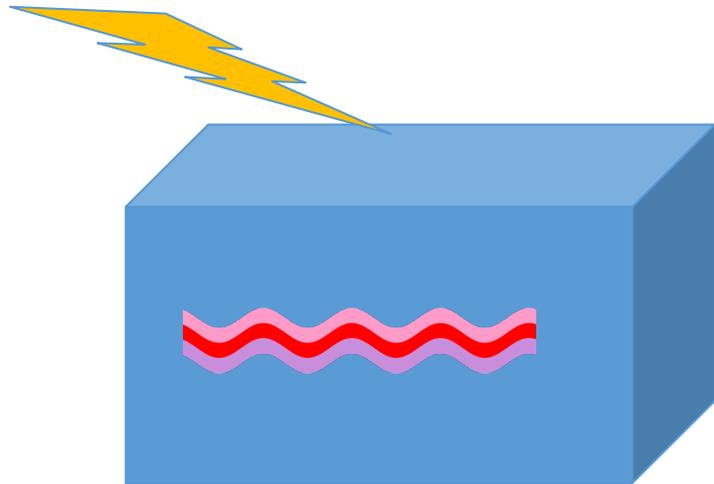
- Adding extra layers of parallelization -> increase intrinsic scaling limit
- Reducing communication overhead to reach the intrinsic limit



Outline

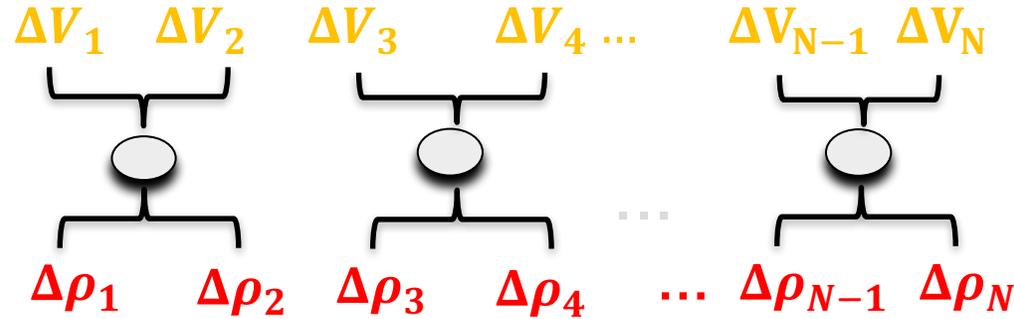
- WEST – additional layers of parallelization
 - Band parallelization of Sternheimer equation
 - Task group parallelization to fit 3D FFTs within single KNL node to reduce communication overheads and take advantage of HBM
- Qbox – reduce communication overheads of dense linear algebra with on-the-fly data redistribution
 - Gather & scatter remap
 - Transpose remap
- Conclusions and insights

Linear response theory



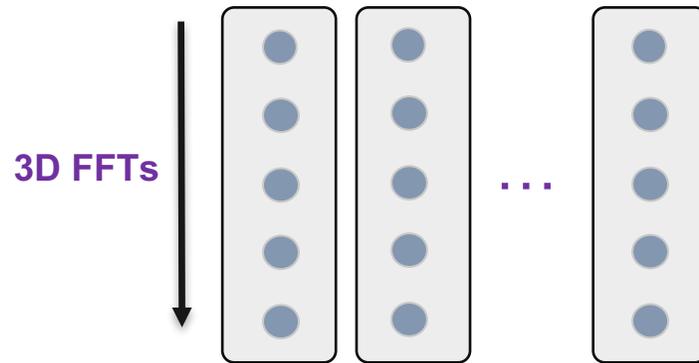
$$\Delta\rho = \chi \Delta V_{pert}$$

Electronic density Response function Perturbation potential



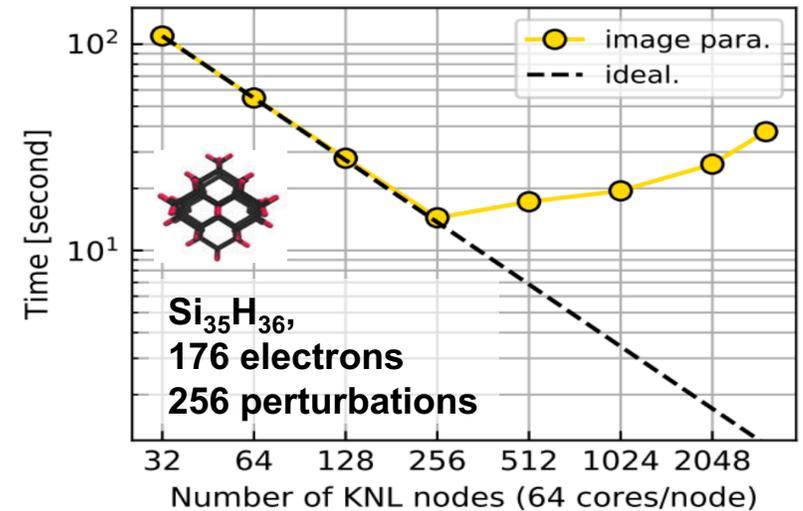
Massively parallel by distributing perturbations

Parallelization scheme (image groups & plane wave)

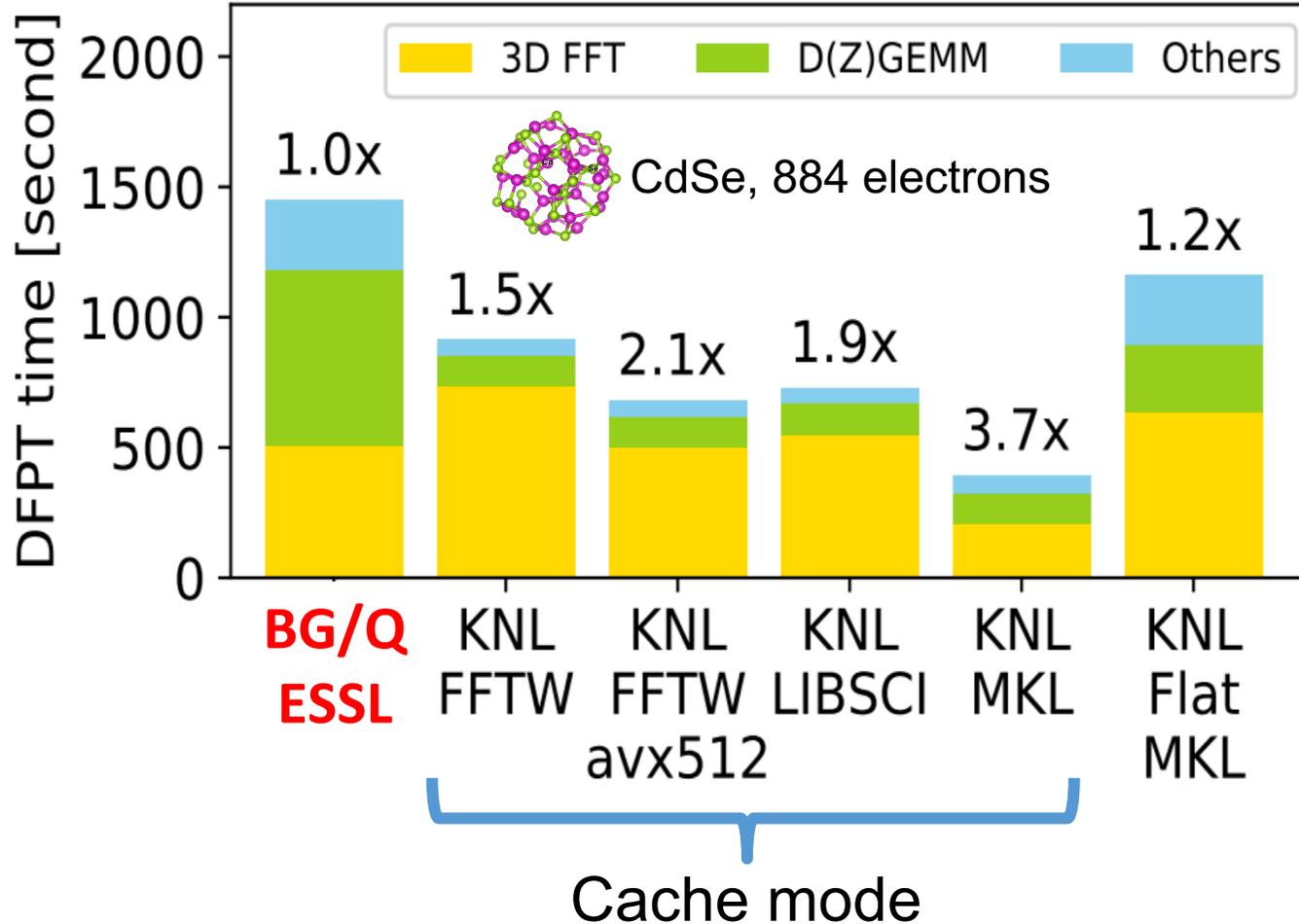


Intrinsic strong scaling limit
 $nproc \sim N_{pert} \times N_z$

Efficient strong scaling up to the intrinsic limit then stop scaling

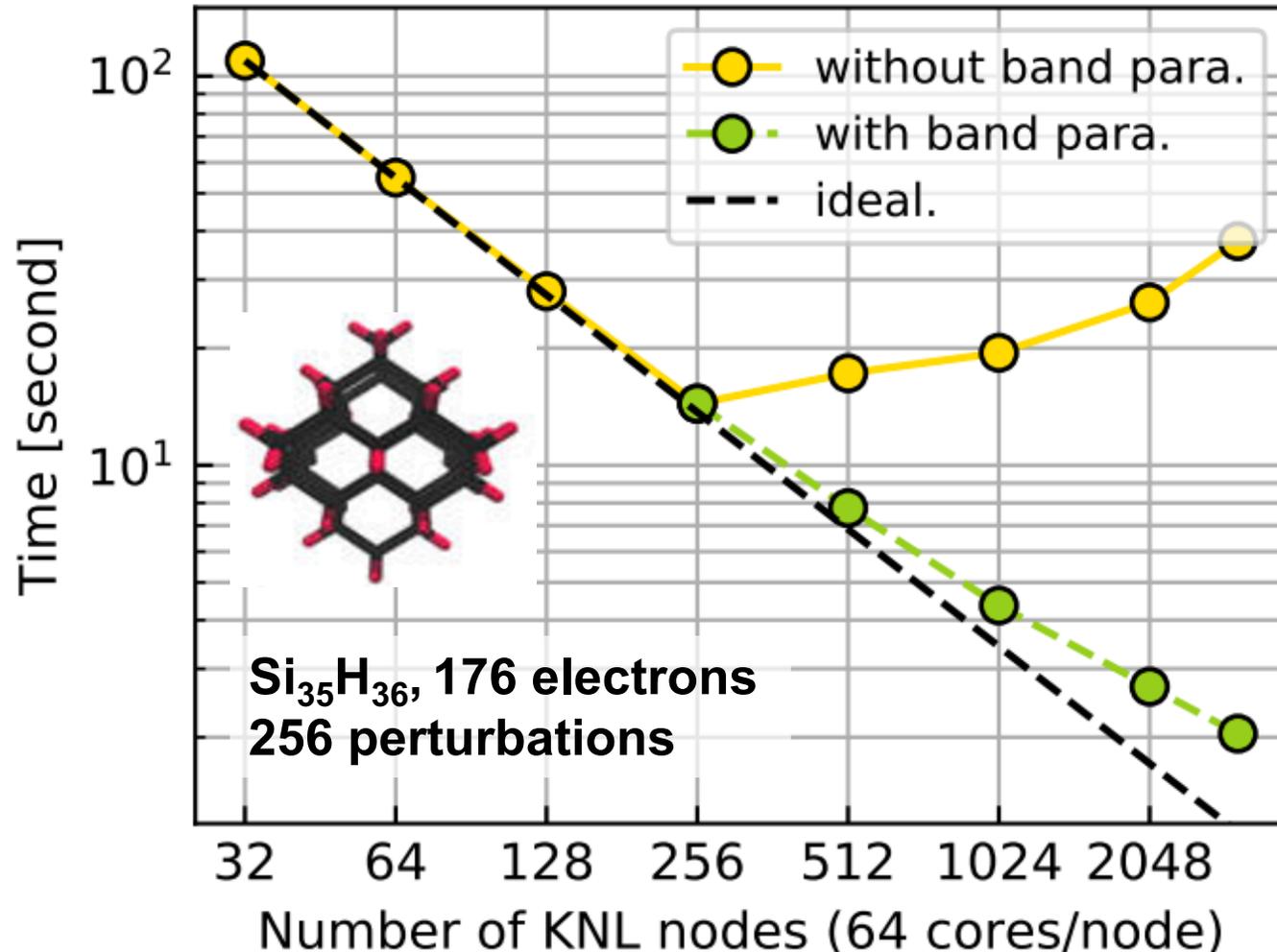


Single perturbation runtime (4BG/Q vs 1KNL)

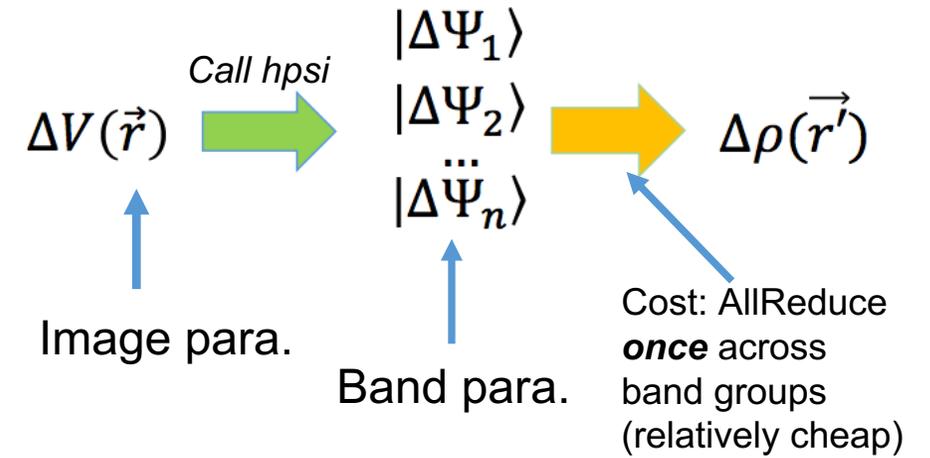


- 80% of runtime is spent in external libraries
- 3.7x speedup from BG/Q(ESSL) to KNL(MKL)
- High-bandwidth memory on Theta critical for performance (e.g. 3D FFTs): 3.1x speedup

Improvement of strong scaling by band parallelization of Sternheimer equation



Sternheimer equation

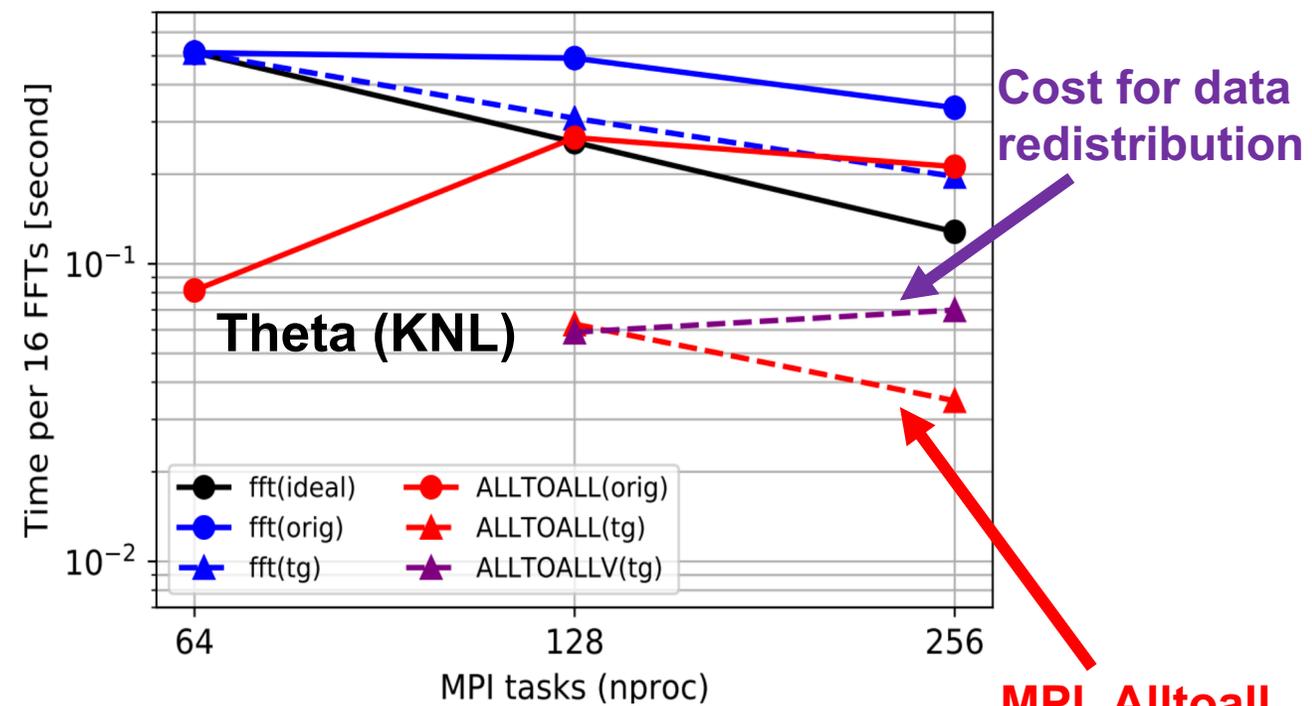
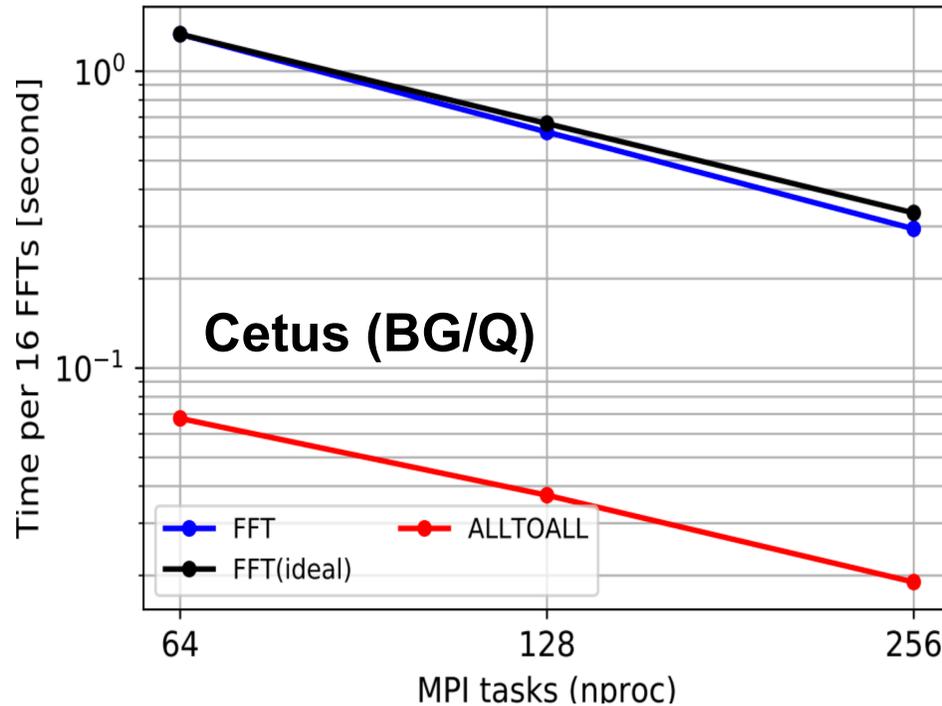


Increased parallelism by arranging the MPI ranks in a 3D grid (perturbations & bands & FFT)

New intrinsic strong scaling limit:

$$nproc = N_{pert} \times N_{band} \times N_z$$

Improving strong scaling of 3D FFTs using task group parallelization when a band group spreads across multiple nodes



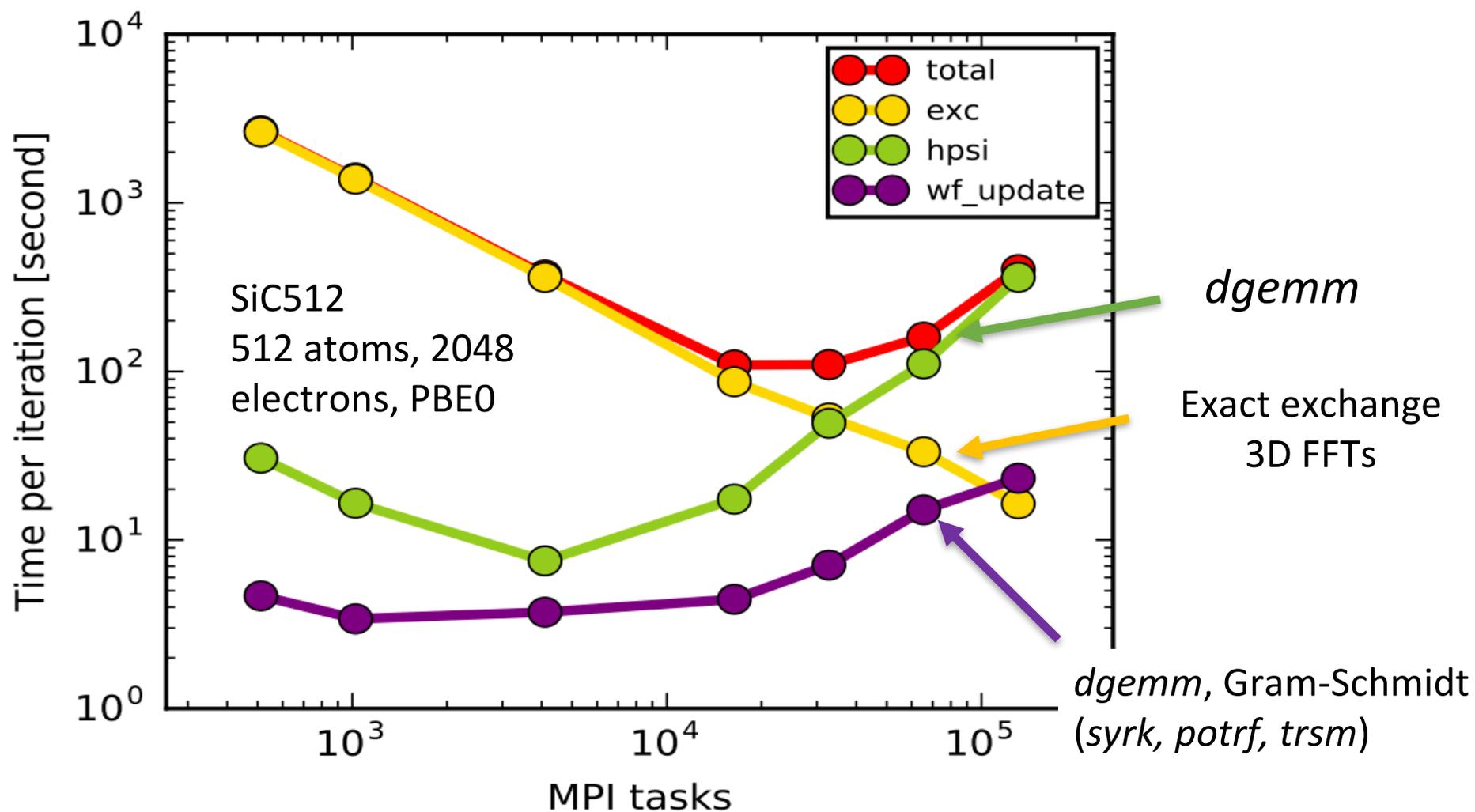
Strong scaling of 3D FFT (plane and pencil decomposition) on Cetus and Theta using 256x256x256 FFT grid and FFTXlib in QE

Small 3D FFTs do not scale well across multiple KNL nodes because of internode communication overheads relative to shared-memory MPI. Task groups (tg) redistribute complete wave functions to separate nodes to simultaneously compute multiple 3D FFTs.

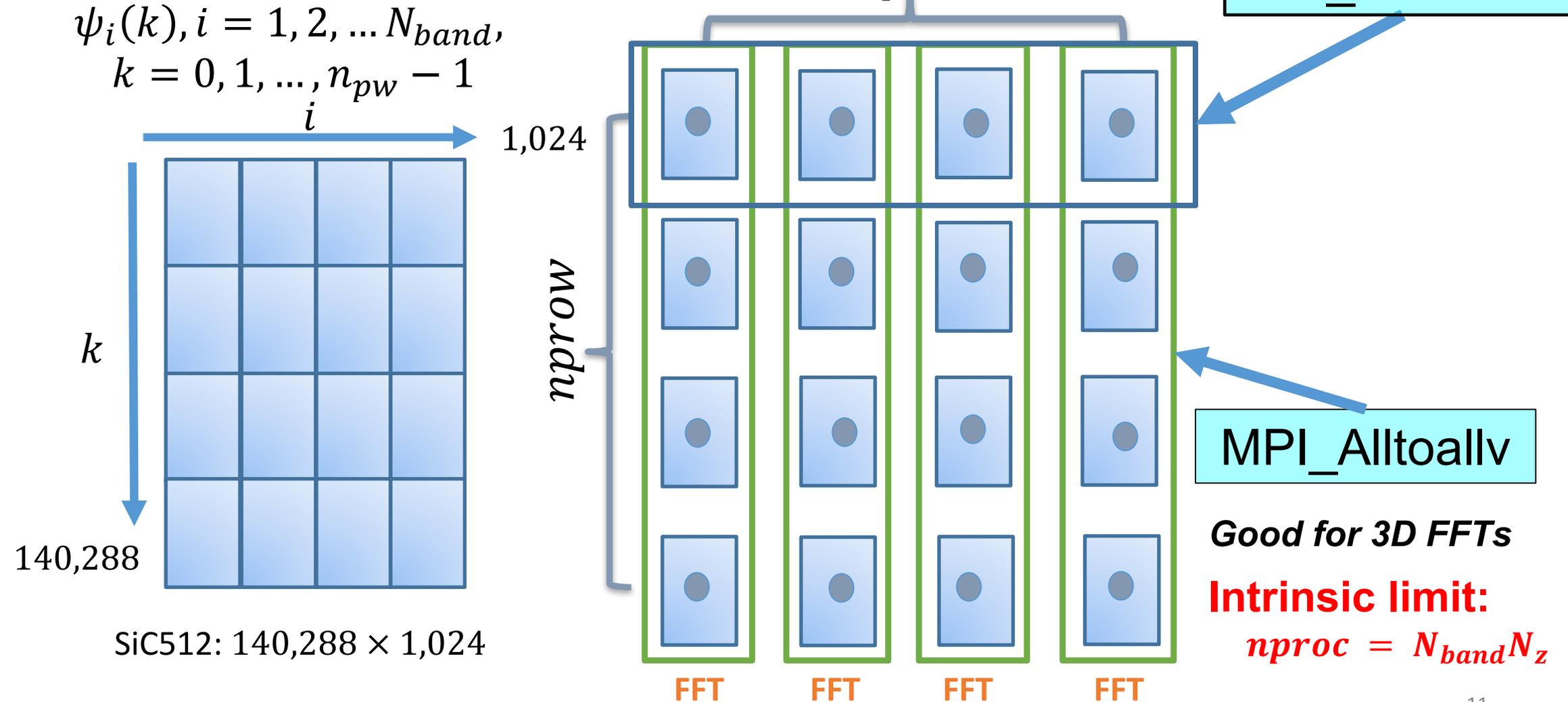
Outline

- WEST – additional layers of parallelization
 - Band parallelization of Sternheimer equation
 - Task group parallelization to fit 3D FFTs within single KNL node to reduce communication overheads and take advantage of HBM
- Qbox – reduce communication overheads of dense linear algebra with on-the-fly data redistribution
 - Gather & scatter remap
 - Transpose remap
- Conclusions and insights

Strong scaling of Qbox for hybrid-DFT calculations

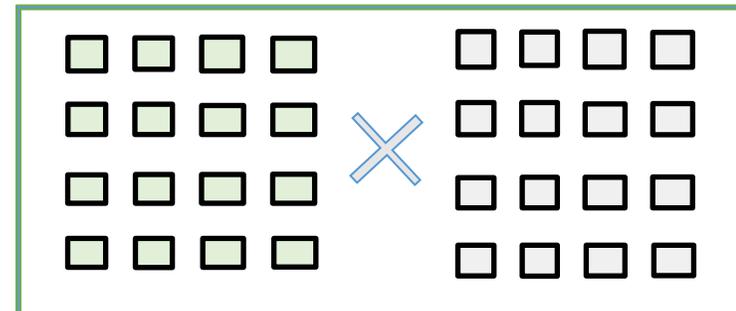
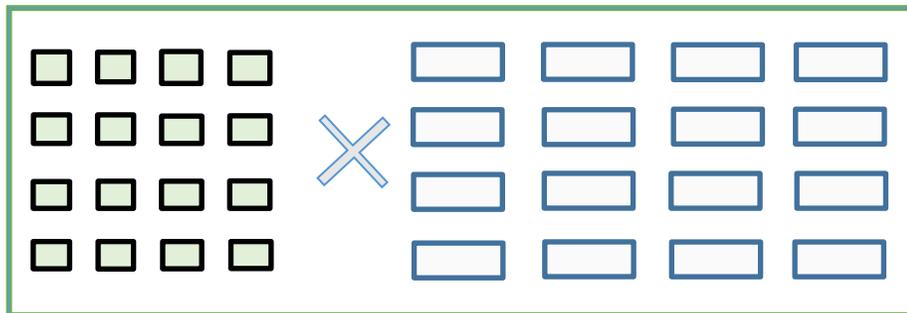
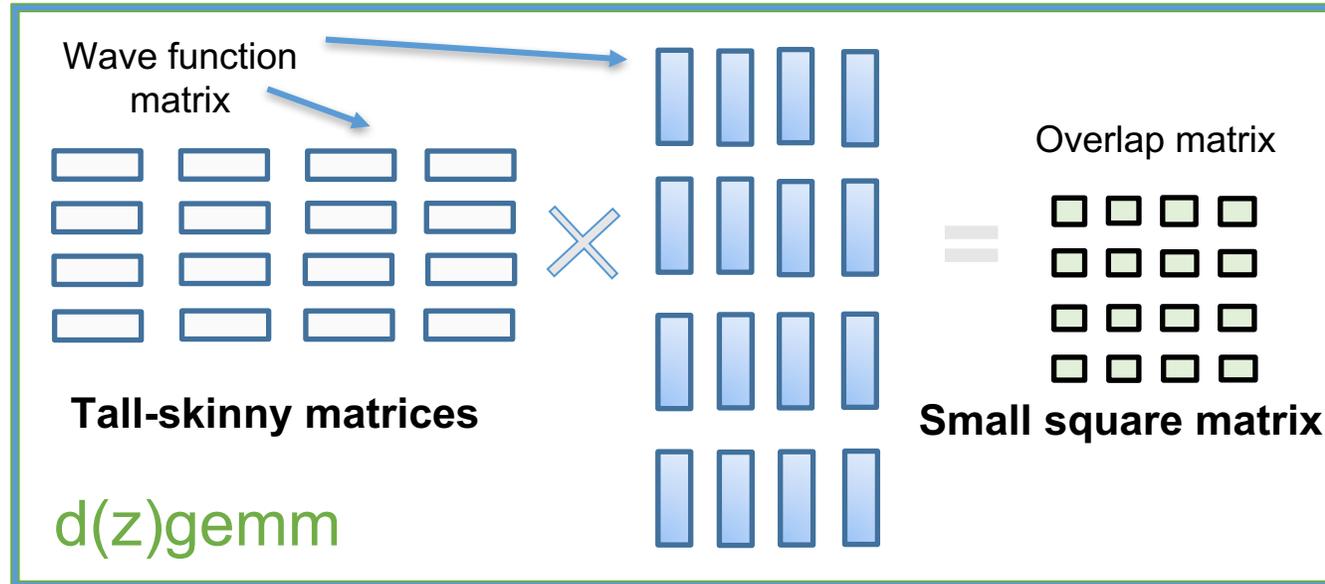
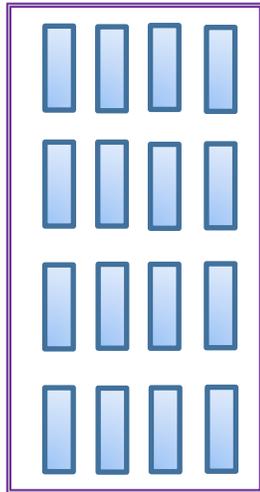


Data layout: block distribution of wave functions to 2D process grid



Poor scalability of ScaLAPACK for tall-skinny matrices and small square matrices due to communication overheads

Gram-Schmidt



Reducing communication overheads from ScaLAPACK with “gather & scatter” remap

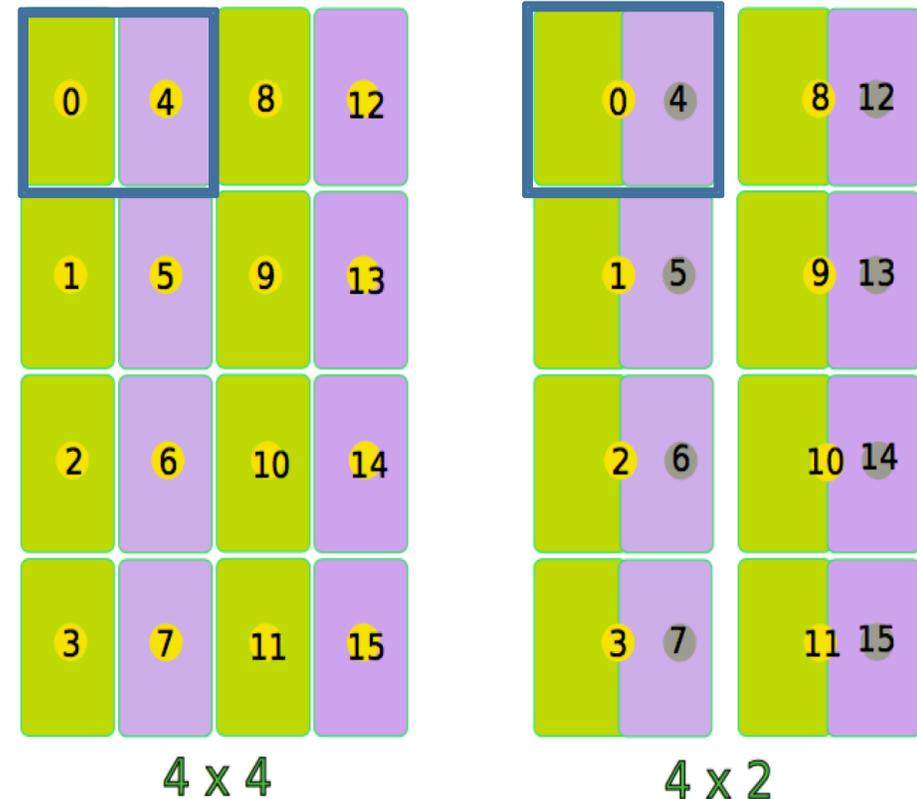
Solution: creating a context with fewer columns and on-the-fly data redistribution

- Compute 3D FFTs on original grid
- Gather data to smaller grid
- Run ScaLAPACK on smaller grid
- Scatter data back to original grid

The remap communication pattern only involves procs within same row or column.

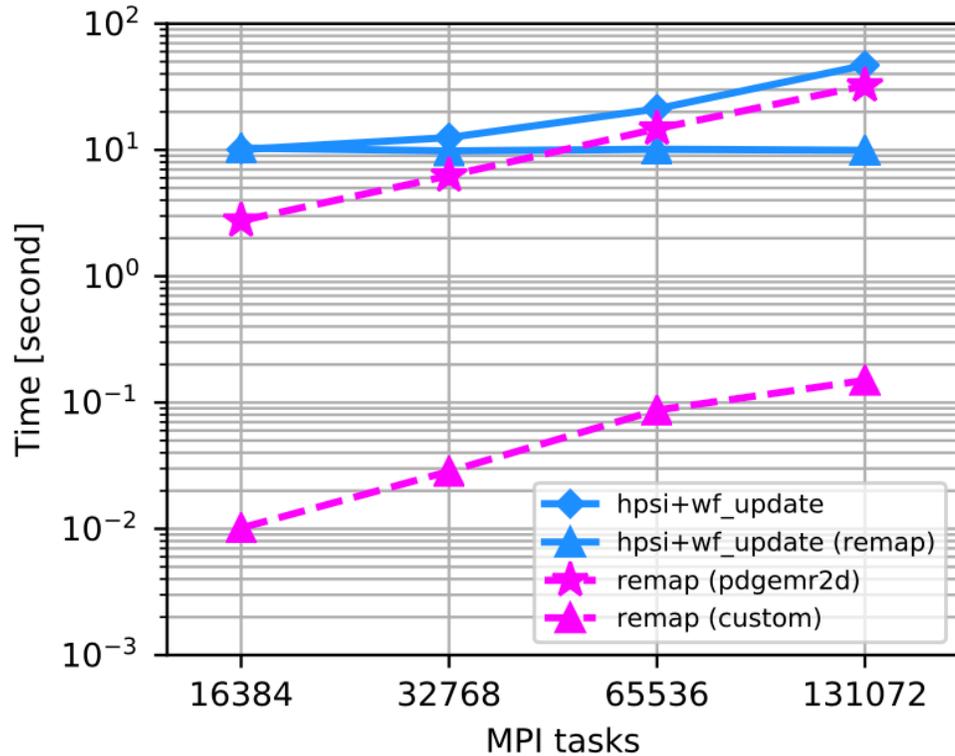
Key: remap communication time needs to be small.

Gather & scatter communication pattern



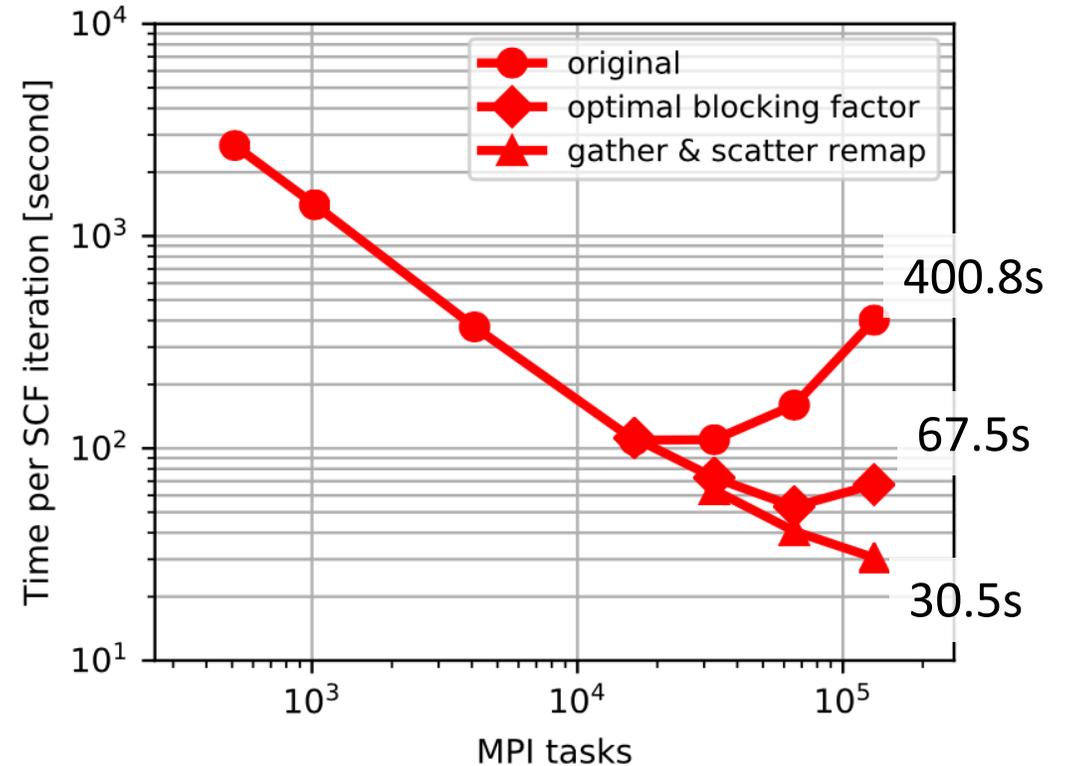
Schematic of gather & scatter remap, gray processes are idle during ScaLAPACK computation

Improvement of strong scaling using “gather & scatter” remap



hpsi + wf_update time remains minimal relatively flat with remap, and the **remap time (custom)** is two orders of magnitude smaller than ***hpsi + wf_update* time**.

Custom remap function is 1000x faster than ScaLAPACK’s pdgemr2d.



Improvement of Qbox’s strong scaling after optimizations; runtime of improves from ~400 to ~30s per SCF iteration (13x speedup) on 131,072 ranks for 2048 electrons.

Reducing communication overheads from ScaLAPACK by “transpose” remap

Problem of “gather & scatter”:

Idle processes.

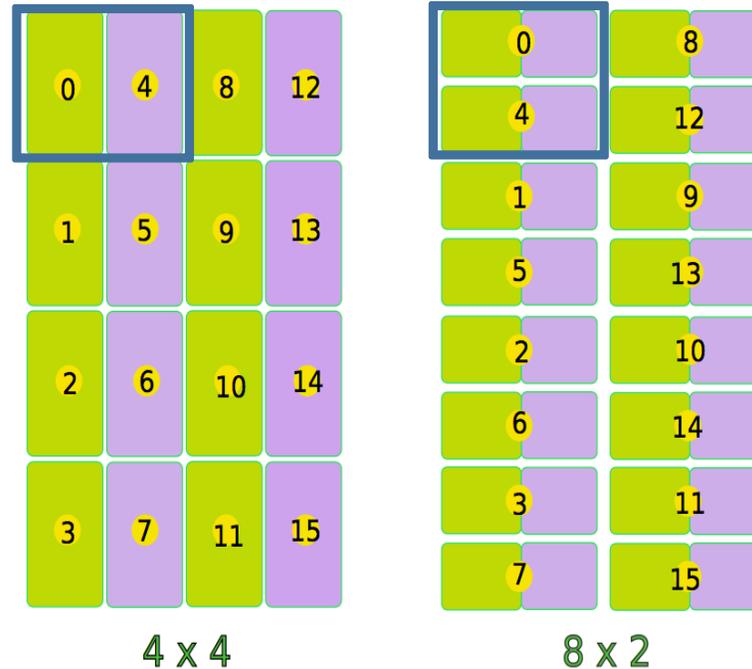
How to utilize them? Assign idle processes to active columns.

Transpose remap:

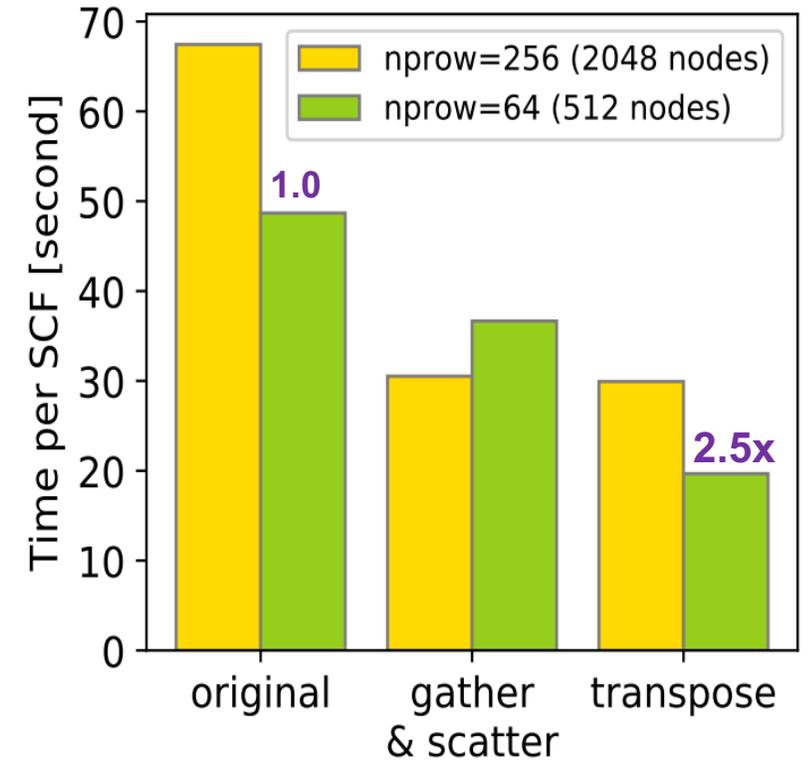
- Perform 3D FFTs in the original context.
- Transfer data through a series of **local regional transposes**
- Run ScaLAPACK in the new context

Key concept for remap: creating different contexts that are optimal for different kernels redistributing the data on-the-fly

Transpose communication pattern



Process rearrangement and data movement of transpose remap



Improvement of runtime by remap methods

$$(1) \text{ npcol}' = \frac{\text{npcol}}{8}, \text{ nprow}' = \text{nprow}$$

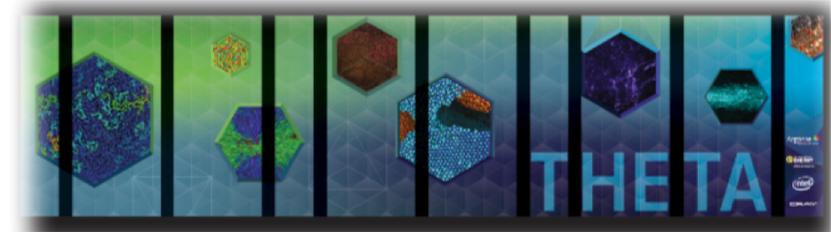
$$(2) \text{ npcol}' = \frac{\text{npcol}}{8}, \text{ nprow}' = 8 \times \text{nprow}$$

Conclusion and Insights

- Band parallelization reduces the internode communication overhead and improves strong scaling of WEST up to $N_{\text{FFT}}N_{\text{pert}}N_{\text{band}}$ cores.
- Optimal remapping of data for matrix operations in Qbox reduces ScaLAPACK communication overhead at large scale, and makes hybrid- DFT calculation scale to $N_{\text{FFT}}N_{\text{band}}$ cores.
- Given the increased computational performance relative to network bandwidths, it is crucial to reduce and/or hide inter-node communication costs.
- Guiding principles for developing codes in many-core architecture:
 - 1) Parallelizing independent, fine-grain units of work, reducing inter-node communication, and maximizing utilization of on-node resources.
 - 2) Optimizing communication patterns for performance critical kernels with on-the-fly data redistribution and process reconfiguration.

Acknowledgement

- This research is part of Theta Early Science Project at Argonne Leadership Computing Facility, which is a DOE Office of Science User Facility under Contract DE-AC02-06CH11357.
- This work was supported by MICCoM, as part of Comp. Mats. Sci. Program funded by the U.S. DOE, Office of Science, BES, MSE Division.



U.S. DEPARTMENT OF
ENERGY

Office of Science

MICCoM