Electron Dynamics Simulation with Time-Dependent Density Functional Theory on Large Scale Many-Core Systems

Yuta Hirokawa (Graduate School of Systems and Information Engineering, University of Tsukuba, Japan) Advisor: Prof. Taisuke Boku (University of Tsukuba, Japan)

筑波大学 University of Tsukuba
University of Tsukuba Center for Computational Sciences

1. Introduction

1. The many-core processor (such as Xeon Phi, GPU) Clusters are now in operation around the world.
2. This computing efficiency is difficult to achieve due to the different characteristics of advanced processors such as Intel Xeon.
3. We port ARTED, an electron dynamics simulator, to the Xeon Phi (Knights Corner, KNC), SPARC64 XIfx (Fujitsu FX100 system), and Knights Landing (KNL).
4. Our challenge was to optimize the application on the several many-core systems.

2. ARTED: Electron Dynamics Simulator

ARTED is an electron dynamics simulator with Time-Dependent Density Functional Theory (TDDFT), which is developed by Center for Computational Sciences, University of Tsukuba.
The original target system of the application is K computer at RIKEN AICS.
The application are published to Github as open-source software.
➜ https://github.com/ARTED/ARTED

1. It solves the electron wave function from TD-Kohn-Sham Equation.
2. Time-development part dominates the computation, especially the Hamiltonian computation of electron consumes over 80% of execution time.
3. An optimization target is 25-points stencil computation in the Hamiltonian, for the stencil computaton, performing NKxNB times of computation with size NL. (Fig. 1)

We implement and optimize the explicit vectorization of stencil by five SIMD instruction sets.
512-bit SIMD:
  Intel Initial Many-Core Instruction (for KNC) and AVX-512 (for KNL)
256-bit SIMD:
  Intel AVX, AVX2 (for Xeon CPU) and HPC-ACE2 (for SPARC64 XIfx)

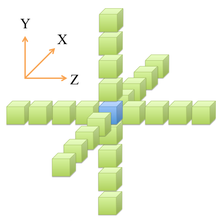

Fig. 1: The memory access pattern of 25-points stencil computation.

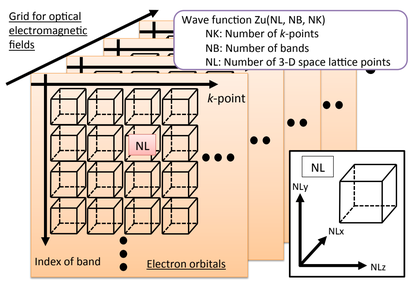

Fig. 2: The schematic picture of computation fields at ARTED with wave function.

The below two materials are frequently simulated with ARTED.
1. Si (large thread parallelism, short vector parallelism)
2. SiO_2 (small thread parallelism, large vector parallelism)

Table 1: The material parameters.

Material	NK	NB	NL	# of thread parallelism
Si	24^3	16	16x16x16	24^3 x 16
SiO_2	4^3	48	20x36x52	4^3 x 48

3. Optimization of The Stencil Computation for wide SIMD Processors

Compiler automatic vectorization (with Fortran90)

We modified some parts of code to be bottlenecks of the performance.

1. Remove the indirect reference that is used when they get the neighbor points.
2. In the direct reference, it issues the remainder operation because it has periodic boundary condition. We use the logical AND operation or the remainder table on behalf of the operation.

Explicit vectorization (with C)

We implemented the explicit vectorization for fitting to the compiler vectorization behavior.

1. Use the direct reference to help the compiler vectorization.
2. Omit the multiplication of complex values when an operand is constant value.
3. Remove the Gather Intrinsics under the Z-direction (unit-strided) memory access of which performance is not enough.
   - We implement unit-strided memory access with aligned load + bit-shift Intrinsics.

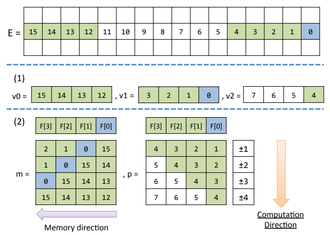

Fig. 3: The schematic picture of optimized unit-strided memory access under 512-bit SIMD.

4. Preliminary Evaluation of The Communication Performance each System

KNC cluster is a heterogeneous system that equips Xeon Phi card and Xeon CPU socket. We should be use the Symmetric mode execution when utilizing the all system resources.

MPI_Allreduce must be performed with size of NL (Fig. 3).
In KNC and Symmetric mode, the communication cost increases.
- Intel MPI can be configured to the collective algorithms.
- 4-d tree algorithm reduces latency at Symmetric mode.

However, time-development computation time is milliseconds level, it is negligible.

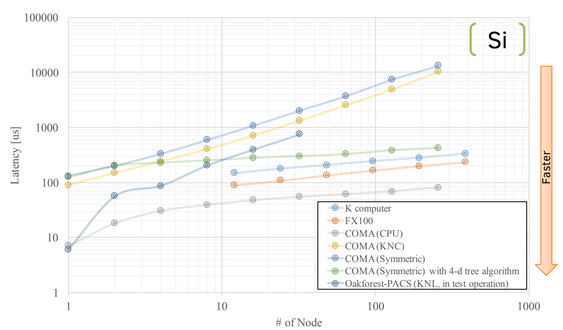

Fig. 3: MPI_Allreduce latency of the each systems. (# of element is 16^3)

5. Performance Evaluation

1. We use the KNC cluster COMA at University of Tsukuba for the performance evaluation. COMA is the largest KNC cluster in Japan with 1 PFLOPS of theoretical peak performance.
2. Fujitsu FX100 at Nagoya University equips 32+2 core processor (SPARC64 XIfx). The processor has two 256-bit SIMD units with FMA in each core. It has 3.2 PFLOPS of theoretical peak performance.
3. We use large-scale Knights Landing (KNL, Xeon Phi 7250, 68 cores) system "Oakforest-PACS" at JCAHPC (Joint Center for Advanced HPC), Japan (in test operation, early results).

Table 3: Evaluation Environment

	K-computer	FX100	COMA	Oakforest-PACS (test operation)
# of Node	192 (System total: 88128)	192 (System total: 2880)	128 (System total: 393)	32
CPU	Fujitsu SPARC64 VIIIfx	Fujitsu SPARC64 XIfx	Intel E5-2670v2 (Ivy-Bridge)	Intel Xeon Phi 7250 (KNL)
			Xeon Phi 7110P (KNC) x2 / Node	
# of Cores / Node	8 cores	32 compute cores + 2 assistant cores	20 (10 cores x2, IVB) + 120 (60 cores x2, KNC)	68 cores (up to 64 cores)
Memory / Node	16 GB	32 GB	64 GB	MCDRAM: 16 GB (MCDRAM only)
Interconnect	Tofu Interconnect	Tofu Interconnect 2	InfiniBand FDR Connect-X3 with OFEF 1.5.4-1	Intel Omni-Path Architecture
Compiler and MPI	Fujitsu K-1.2.0-20-1	Fujitsu 2.0.0	Intel 16.0.2 and Intel MPI 5.1.3	Intel 16.0.4 and Intel MPI 5.1.3

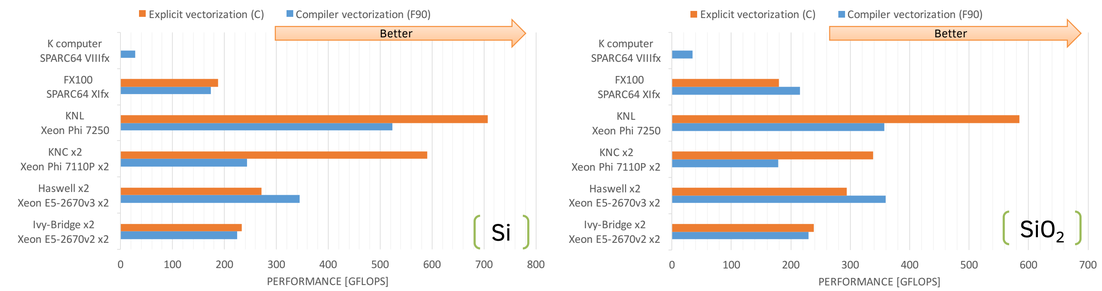

Fig. 4: The single-processor performance of the stencil computation.

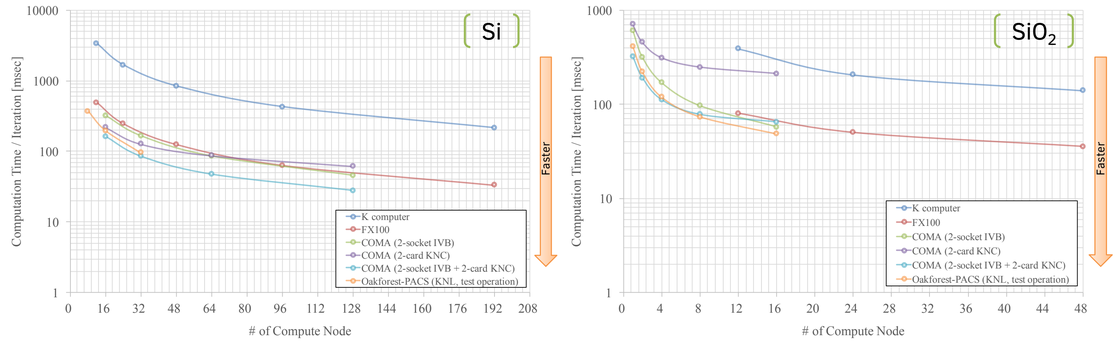

Fig. 5: Time-development loop computation time. For Symmetric mode (2-socket IVB + 2-card KNC), We apply load-balancing between Xeon and Xeon Phi by varying wave pace assignment.

6. Conclusion and Future works

1. KNC is 2.53x faster than Ivy-Bridge CPU with explicit vectorization, and 1.71x performance of Haswell CPU in Si case.
2. We port ARTED code with Symmetric mode execution with IVB and KNC to improve the performance per node up to 1.97.
3. Latest processors performance is not enough from the theoretical peak performance, so we will further optimize it as the future work.
4. In early results, we implement and evaluate the next generation Xeon Phi processor called Knights Landing. KNL achieved over 3x the stencil performance compared with KNC for SiO_2 case. (up to 23.2 % performance efficiency for Si case)
   - ■ AVX-512 code can be easily implemented from IMCI code with preprocessor directives.
5. We will implement and evaluate the NVIDIA GPU version in collaborative research with NVIDIA.

◆ This research was supported by Interdisciplinary Computational Science Program in CCS, University of Tsukuba, Japan.
◆ This work was supported in part by MEXT as a social and scientific priority issue (Creation of new functional devices and high-performance materials to support next-generation industries) to be tackled by using post-K computer.
◆ A part of this research is supported by the Japan Science and Technology Agency's (JST) CREST program.
◆ A part of this research is based on Oakforest-PACS system operated at JCAHPC (Joint Center for Advanced HPC) under cooperation by Information Technology Center at the University of Tokyo and Center for Computational Sciences at University of Tsukuba.
◆ This is collaborative research with Center for Computational Sciences, University of Tsukuba, Japan. (The author would you like to thank Dr. Shunsuke A. Sato, and Prof. Kazuhiro Yabana)