RIKEN R-CCS

RIKEN Center for Computational Science

ANNUAL REPORT

FY2020

RIKEN R-CCS Research Activities

RIKEN
R-CCS
RIKEN
Center for
Computational Science

RIKEN R-CCS
Annual Report FY2020
R-CCS Research Activities

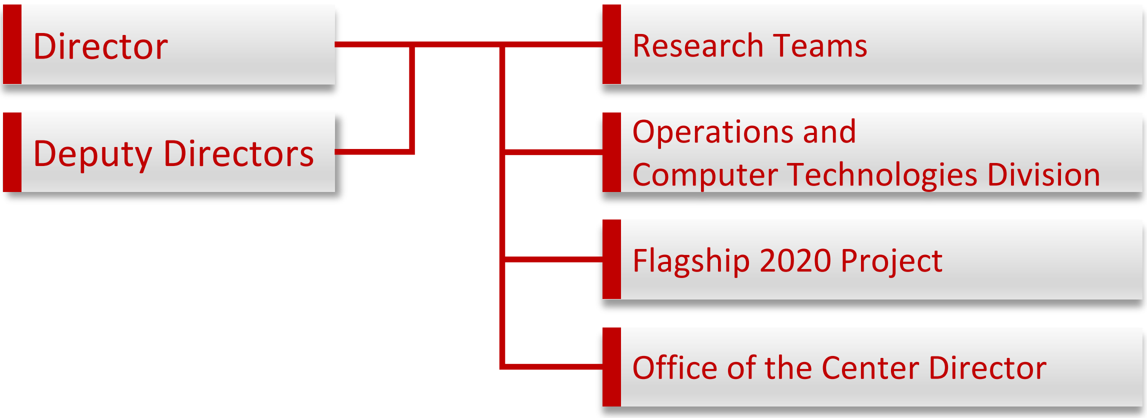
Preface

RIKEN Center for Computational Science (R-CCS) is one of the 13 research centers of Riken, as well as the national center for supercomputing in Japan, developing and hosting the national flagship supercomputer, the current incarnation *Fugaku* having been launched on March 9th 2021,. The research objectives of R-CCS) are threefold: one is to target high performance computation itself as a scientific objective, or namely, "Science of Computing"; another is to apply the enormous computational power thus obtained to solve difficult scientific problems, or namely "Science by Computing"; finally, to collaborate with other scientific disciplines that contribute to advances both sciences, or namely, "Science for Computing". Our goal is to be recognized as one of the global leadership research centers to advance high-end computational science in this regard, advancing computing towards important science and societal objectives.

In particular, R-CCS aims to focus on forefront of supercomputing, involving convergence of first-principle simulation, data, and articial intelligence (AI). Computational science employs multitudes of methodologies to essentially recreate various phenomena as *digital twins*. For example, we model a phenomenon by a set of physical/mathematical formulas, make them computable with world-leading algorithms on supercomputers such as Fugaku, enabling direct, first-principle 'simulation' of the phenomenon as a digital twin; alternatively, we could train an AI with measured or simulated data to create a digital-twin predictor/simulator in tandem with the first-principle simulations; finally, we could analyze the massive data measured on a phenomenon by scientific instruments, and further extrapolate future trends by assimilating with the digital twins on supercomputers. By investigating the most advanced hardware, systems, software, algorithms, and applications to enable such convergence on supercomputers, we will push the forefronts of mankind's knowledge and help to achieve its SDGs goals, as well as to disseminate the results for supercomputing to be the centerpieces of the so-called 'Society 5.0'.

Moreover, innovative Information Technologies (IT) R&D to advance supercomputing is not only applicable to itself, but rather are the bleeding-edge technologies to advance the entire IT as a whole, from Clouds to the Edge, and thus will contribute to the massive improvement for digital transformation of the society as a whole, to achieve the SDGs/Society5.0 goals. But this is becoming increasingly difficult with the so-called end of the Moore's Law approaching. We investigate advancement of both traditonal, massively parallel supercomputing as well as new computing paradigms such as quantum computing, neuromorphic computing, etc., and their combinations We are already in collaboration with other leadership centers of the world in this regard, and continue to establish new collaborations, to play a central role in advancing IT for the society.

R-CCS Organization



Director

Deputy Directors

Research Teams

Operations and Computer Technologies Division

Flagship 2020 Project

Office of the Center Director

Research Teams

- System Software Research Team
- Programming Environment Research Team
- Processor Research Team
- Large-scale Parallel Numerical Computing Technology Research Team
- Field Theory Research Team
- Discrete Event Simulation Research Team
- Computational Molecular Science Research Team
- Computational Materials Science Research Team
- Computational Biophysics Research Team
- Particle Simulator Research Team
- Computational Climate Science Research Team
- Complex Phenomena Unified Simulation Research Team
- Next Generation High Performance Architecture Research Team
- High Performance Big Data Research Team
- Data Assimilation Research Team
- Computational Disaster Mitigation and Reduction Research Team
- Computational Structural Biology Research Team
- High Performance Artificial Intelligence Systems Research Team

Operations and Computer Technologies Division

- Facility Operations and Development Unit
- System Operations and Development Unit
- Application Tuning Development Unit
- HPC Usability Development Unit
- Advanced Operation Technologies Unit

Flagship 2020 Project

- System Software Development Team
- Architecture Development Team
- Application Development Team
- Co-Design Team

Contents

Part I

Research Division

Chapter 1

System Software Research Team

1.1 Members

Yutaka Ishikawa (Team Leader)

Atsushi Hori (Senior Scientist)

Masamichi Takagi (Senior Scientist)

Balazs Gerofi (Research Scientist)

Takahiro Ogura (Research & Development Scientist)

1.2 Overview of Research Activities

The system software research has been conducted in cooperation with the System Software Development Team in the Flagship 2020 project. The team focuses on the research and development of an advanced system software stack not only for the "K" computer but also for toward exa-scale computing including Fugaku.

We have been mainly focusing on scalable high performance communication and file I/O libraries and/or middlewares. The former research topics, sharing virtual address space and IHK/McKernel light-weight kernel, have been almost taken over by the System Software Development Team, but the research results are shown here.

1.3 Research Results and Achievements

1.3.1 IHK/McKernel

This Section describes activities related to IHK/McKernel. Specifically, we discuss development items related to Fugaku and results from large-scale evaluation comparing Oakforest-PACS and Fugaku supercomputers.

1.3.1.1 Fugaku Development

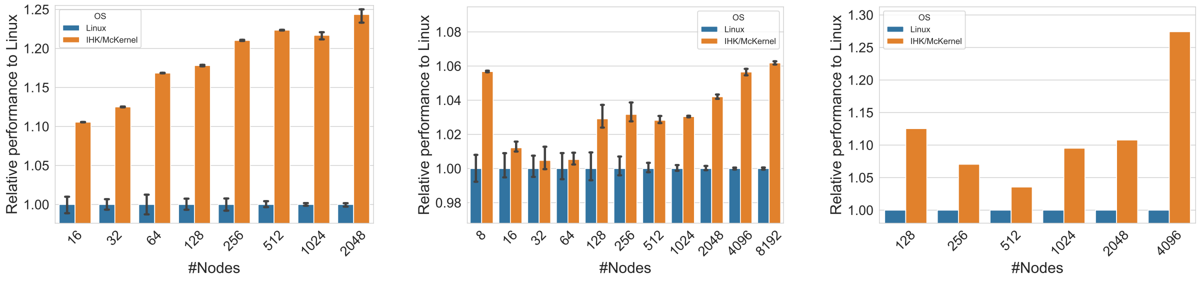
Most of the fiscal year 2020 has been spent on polishing the McKernel port to the ARM instruction set architecture. Additionally, we have integrated IHK/McKernel into Fugaku's containerized runtime as well as into the batch job submission system. Some of the major development items are as follows.

- **Fujitsu Job Submission System:** Fugaku runs a proprietary job scheduler developed by Fujitsu. As opposed to Oakforest-PACS, where booting IHK/McKernel entails nothing more than calling a few privileged mode scripts in the prologue and epilogue of a particular job, on Fugaku there is a much tighter integration between IHK/McKernel and the Fujitsu environment. This is primarily due to the unique features of the Fugaku platform, e.g, the hardware barrier, the way how process placement is performed, its interaction with MPI. One may consider the LWK as a plugin replacement to the `cgroup` facility of the Linux kernel with the important addition of its ability of kernel level specialization. In combination with containers, which enable customization of user-space components, the multi-kernel plus container approach enables specialization of the entire software stack.

- **Tofu PicoDriver:** Finally, another notable extension to McKernel is the Tofu PicoDriver. At high level, the Tofu network's system programming interface provides similar abstractions to that of Infiniband or Intel's OmniPath, but at the implementation level there are many subtle differences. For example, the registration of the so called STAGs, a concept similar to the Infiniband `verbs` layer's memory registration is performed through `ioctl()` calls into the Tofu driver. Because by default this is offloaded to Linux in our multi-kernel framework, it introduces additional latency. To eliminate such overhead we have developed a similar split driver infrastructure to that of OmniPath's PicoDriver describe in last fiscal year's report. We note that all of our experiments have been conducted using this capability.

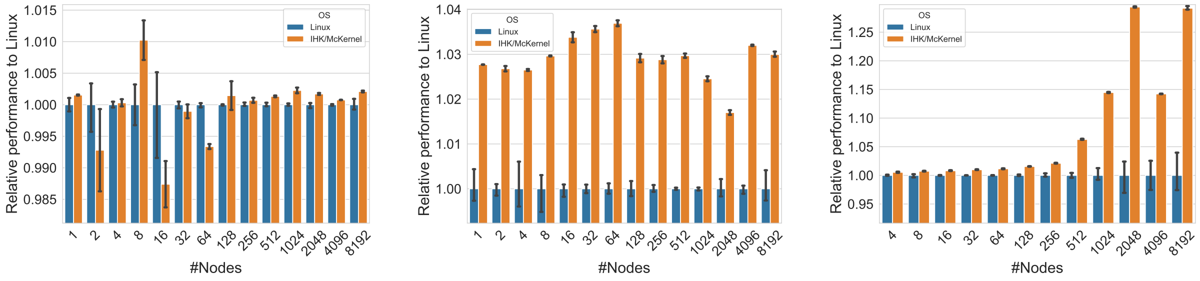
1.3.1.2 Fugaku vs. Oakforest-PACS Evaluation

This section compares the performance of IHK/McKernel with Linux on two supercomputers. We use LQCD and GAMERA, which are two of the priority target applications of the Fugaku development project. Both of these have highly optimized versions for both target platforms that entail substantial code changes, but the different versions of the applications address the same science problem. In addition we use GeoFEM, for which we had a highly optimized version for Oakforest-PACS, but it also has a few minor tweaks to support efficient execution on Fugaku.



(a) LQCD (b) GeoFEM (c) GAMERA

Figure 1.1: Application results on Oakforest-PACS



(a) LQCD (b) GeoFEM (c) GAMERA

Figure 1.2: Application results on Fugaku

Figure 1.1 and Figure 1.2 summarize application level results comparing OFP and Fugaku on LQCD, GeoFEM and GAMERA. Similarly to the CORAL applications, McKernel on OFP consistently outperforms Linux on these applications as well. Results are shown in Figure 1.1. Although we have results for only up to 2k nodes on LQCD, we observe an increase in McKernel's performance gain as we scale to larger node counts reaching close to 25% at 2k nodes. For GeoFEM, we have full-scale OFP measurements with McKernel outperforming Linux by up to 6% at the entire machine. To much of our surprise we also find a significant amount of variation (indicated by large error bars) across measurements even on McKernel, although we believe this could be related to the fact that different measurements run on different nodes, which is particularly true for smaller node counts. With respect to GAMERA, we have limited data points for showing error bars on all scales and decided to show only the mean performance. As seen, McKernel outperforms Linux by over 25% on half-scale of the OFP supercomputer.

Fugaku results are shown in Figure 1.2. We observe substantially lower performance gain with McKernel over Linux on the Fugaku machine, although arguably 8k compute nodes on Fugaku is still relatively small scale when considering the overall system's node count. LQCD performs almost identical on the two operating systems. For GeoFEM, we observe an average of 3% performance improvement over Linux when running on McKernel and the tendency indicates the gain would possibly remain constant even if we scaled further out. Only for GAMERA we see an increase in performance gain reaching up to 29% on McKernel when running on 8k compute nodes. We did investigate the GAMERA results further and found that McKernel performs significantly better in the first step (out of three) of the application, which may be related to some sort of overhead in initialization on Linux. Had the application run further steps we would likely see McKernel's advantage decrease. Unfortunately, we had a very narrow window of opportunity for pinpointing where the performance difference stems from, but we did observe faster RDMA registration in McKernel due to the LWK integrated Tofu driver, which we suspect as one of the main contributors to the performance improvement.

A paper summarizing this study, titled *Linux vs. Lightweight Multi-kernels for High Performance Computing: Experiences at Pre-Exascale* has been published at the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC'21).

1.3.2 Cross-Stack Dynamic Resource Management

This piece of work has been a collaborative effort with participants from CEA, RIKEN, Lawrence Livermore National Laboratory, Intel Corp., Inria and the University of Bordeaux. We have been focusing on emerging workloads on supercomputing platforms. These workloads are pushing the limits of traditional high-performance computing software environments. Multi-physics, coupled simulations, big data processing and machine learning frameworks, asynchronous background threads, and multi-component workloads pose serious challenges to system and application developers. At the heart of the problem is the lack of cross-stack coordination to enable flexible resource management among multiple runtime components.

As part of this collaboration, we analyzed seven real applications that represent emerging workloads and illustrate the scope and magnitude of the problem. From these applications, we extracted several themes which highlight requirements for resource managers of next generation systems. Using these requirements, we then proposed a general, cross-stack coordination layer and outlined its functionality. We also demonstrated the benefits of our approach through a number of elements that address pieces of the overall problem and discussed how these initial efforts may fit into the proposed, general solution.

The proposed framework provides a cross-stack coordination layer in charge of mapping any runtime component to the available hardware resources. A paper, titled *Application-Driven Requirements for Node Resource Management in Next-Generation Systems* has been published at the International Workshop on Runtime and Operating Systems for Supercomputers (ROSS), held in conjunction with the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC'20).

1.3.3 Sharing Virtual Address Space

The two most common parallel execution models are multiprocess (MPI) and multithread (OpenMP). The multiprocess model allows each process to own a private address space, and the multithreaded model shares all address space by default. In the multiprocess model inter-process communication is inefficient because a process cannot access data owned by the other processes. In the multithread model threads share the same virtual address space and can access the all data which may incur lock contention overhead. Thus, both models have advantage and disadvantage. We propose a new implementation of the third model, called *Process-in-Process (PiP)* to take the best of two worlds, mutiprocess and multithread. In PiP, processes are mapped into a single virtual address space, but each process still owns its private storage. The idea of address-space sharing is not new. What makes PiP unique, however, is that its design is completely in user space, making it a portable and practical approach for large supercomputing systems.

To demonstrate the performance benefit of PiP, MPICH (3.4.1) was modified to spawn PiP tasks instead of creating Linux processes in the original MPICH. Although the shared address space which PiP provides allows to potentially improve MPI application performance in various ways, the MPICH modification here is minimized to compare the bare performance. Figure 1.3 shows the relative performance ratios on various HPC applications (HPCCG, miniGhost, LULESH2.0, miniMD, miniAMR, and mpiGraph) running on one node, 32 processes, of Oakforest-PACS. In this figure, the original MPICH performance was the base. *PiP-Shmem* is

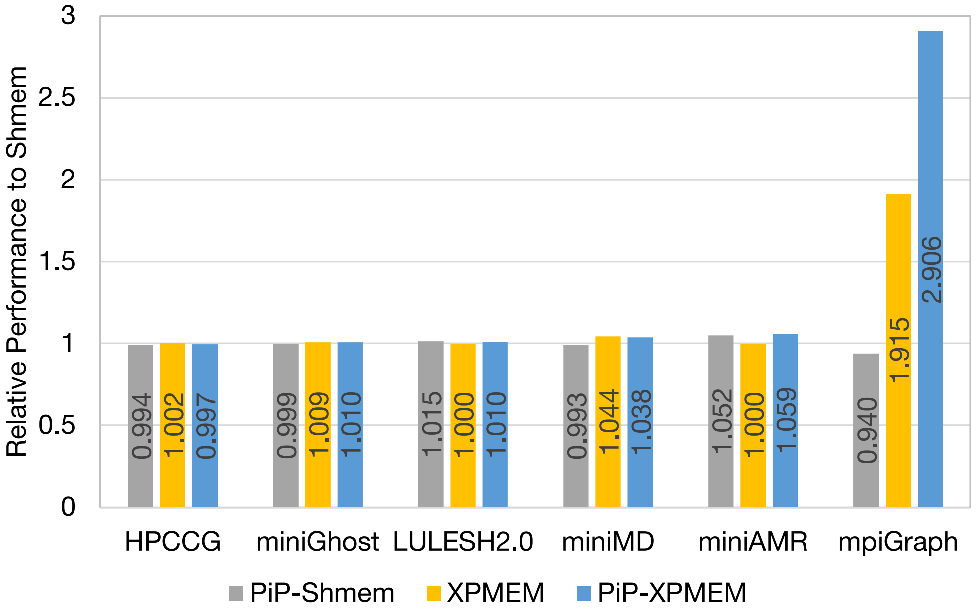


Figure 1.3: Comparing HPC Application single-node performance

PiP-aware version of MPICH, *XPMEM* is XPMEM-enabled MPICH and *PiP-XPMEM* is another XPMEM-enabled MPICH in which XPMEM functions are implemented by using PiP. The most notably the performance of mpiGraph with PiP-XPMEM is almost 3x to the original MPICH and 1.5x to the XPMEM-enabled MPICH.

1.3.3.1 CEA-Riken Collaboration

CEA has been developing MPC which has the same goal to implement shared address space as PiP does. PiP's approach is to implement a user-level library while MPC's approach is to implement another programming language based on the multithread model. It is very natural to collaborate with the other since both have the same goal.

As described above, PiP has succeeded to implement BLT and ULP on PiP. CEA is very interested in those idea and they are working to have the similar functionalities of PiP's BLT and ULP in their MPC implementation.

1.3.3.2 DOE-MEXT Collaboration

One of the biggest issues of an MPI implementation is the progress handling. In a naive MPI implementation, progress is only handled when calling an MPI function. This implementation may incur longer latency because the incoming messages are not processed when receiver process is busy for computation and having no chance to call any MPI functions. One way to solve this problem, having an additional thread which is responsible the progress handling.

In last fiscal year, it was proven that communication workloads of processes running on the same node can be balanced by stealing communication workloads of the other processes by using PiP (CAB-MPI), resulting more efficient communication performance. This year, the similar technique was applied to the asynchronous progress implementation. By implementing progress thread as a PiP task, the PiP task can steal the progress workloads of the other MPI processes running on the same node (Daps). Since most MPI implementations are based on the process model, the shared address space nature of PiP plays an very important role in both of communication workload stealing and progress workload stealing. The similar technique can be found in Casper by having asynchronous *process* to handle and balance the progress of one-sided communications. Daps, however, can handle any communication workloads in addition to the one-sided communications.

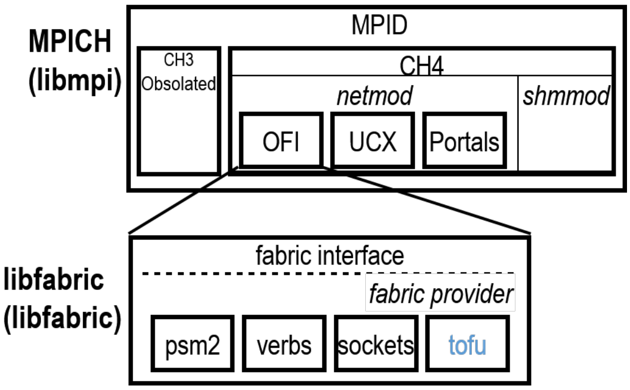


Figure 1.4: MPICH Software Architecture

1.3.4 MPICH-Tofu

We have been implementing an MPI library, based on MPICH developed by Argonne National Laboratory, for the Fugaku supercomputer under the DoE/MEXT collaboration in order to provide MPICH-based MPI for Fugaku.

The MPICH implementation has two layers, so-called MPID and device layers as shown in Figure 1.4. Basically, the MPID layer implements all the MPI functionalities using the device interface. The device layer CH4 is the latest device implementation. It consists of network and shared memory implementation layers, named "netmod" and "shmmod," respectively. The netmod layer consists of OFI (OpenFabrics Interfaces), UCX (Unified Communication X), and Portal network drivers. We support the OFI driver for the Tofu-D implementation. The "tofu" provider in OFI is our Tofu-D implementation. The almost same capabilities but much simpler user API is also provided, named "UTF."

The implementation of MPICH for Tofu-D is named MPICH-Tofu. The source code has been available in the following git repository: https://github.com/yutaka-ishikawa/mpich-tofu/wiki

1.4 Schedule and Future Plan

This team is terminated in FY2020.

1.5 Publications

1.5.1 Articles/Journal

[1] Atsushi Hori, Emmanuel Jeannot, George Bosilca, Takahiro Ogura, Balazs Gerofi, Jie Yin, Yutaka Ishikawa, An international survey on MPI users, Parallel Computing (submitted)

1.5.2 Conference Papers

[2] Balazs Gerofi, Kohei Tarumizu and Lei Zhang and Takayuki Okamoto and Masamichi Takagi and Shinji Sumimoto and Yutaka Ishikawa, "Linux vs. Lightweight Multi-kernels for High Performance Computing: Experiences at Pre-Exascale", IEEE/ACM International Conference for High Performance Computing, Networking, Storage, and Analysis (SC) 2021

[3] Edgar A Leon, Balazs Gerofi, Julien Jaeger, Guillaume Mercier, Rolf Riesen, Masamichi Takagi, Brice Goglin, "Application-Driven Requirements for Node Resource Management in Next-Generation Systems", International Workshop on Runtime and Operating Systems for Supercomputers (ROSS), held in conjunction with the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC), 2020

1.5.3 Posters

1.5.4 Invited Talks

1.5.5 Oral Talks

1.5.6 Software

- McKernel
  https://github.com/ihkmckernel
  https://ihkmckernel.readthedocs.io/ja/latest/

- PiP
  https://github.com/procinproc/procinproc.github.io

- MPICH-Tofu
  https://github.com/yutaka-ishikawa/mpich-tofu/wiki

- DTF
  https://github.com/maneka07/DTF

1.5.7 Patents

Chapter 2

Programming Environment Research Team

2.1 Members

Mitsuhisa Sato (Team Leader)

Yuetsu Kodama (Senior Research Scientist)

Hitoshi Murai (Research Scientist)

Miwako Tsuji (Research Scientist)

Masahiro Nakao (Research Scientist)

Jinpil Lee (Postdoc Researcher)

Tetsuya Odajima (Postdoc Researcher)

Itaru Kitayama (Technical Staff)

Masahiro Yasugi (Senior Visiting Researcher)

Hitoshi Sakagami (Senior Visiting Researcher)

Brian Wylie (Visiting Researcher)

Christian Feld (Visiting Researcher)

Hidetoshi Iwashita (Visiting Researcher)

Hiroko Takahashi (Assistant)

2.2 Overview of Research Activities

In order to exploit full potential computing power of large-scale parallel system to carry out advanced computational science, efficient and productive parallel programming models are required to coordinate these processors to perform scientific computing. Our team conduct researches and developments on parallel programming models and language to exploit full potentials of large-scale parallelism in the large-scale parallel system and increase productivity of parallel programming.

In 2020FY, in order to archive these objectives above, we carried out the following researches:

(1) We continued working on the development and improvement of XcalableMP (XMP) programming languages. XcalableMP is a directive-based language extension, designed by XcalableMP Specification Working Group (XMP Spec WG) including some members from our team as a community effort in Japan. In this year, we designed the XMP APIs for a compiler-free approach. And, we made a prototype implementation of XcalableMP for C++.

(2) Since the Fugaku is a large-scale multicore-based system, we are investigating programming models for manycore-based parallel systems as XcalableMP 2.0. We focus especially on the integration of dynamic tasking with PGAS programming model. In this year, we continued the design of programming models for task parallelism and PGAS, and designed an OpenMP task runtime system for batched kernel APIs for efficient heterogeneous computing.

(3) As a part of Flagship 2020 project, we carried out several performance evaluation of Fugaku and A64FX manycore processor using UK benchmarks, open source HPC software and SPEC benchmarks. We evaluated the performance of the breadth-first search (BFS) benchmark using the full system of Fugaku for Graph500 competition in collaboration with Kyushu University and Fixstars Corporation. As a results, we won the first position of Graph500 BFS in June/November of 2020.

(4) We conducted several collaborations on the performance evaluation with JSC, University of Tsukuba, and other groups. In this year, we had many intensive discussions with Prof. Boku's team of University of Tsukuba, and we continued the performance study of Fugaku on NEST brain simulator developed by JSC, using Scalasca.

Due to COVID-19's situation, we canceled promotion activities to disseminate our software, XcalableMP.

2.3 Research Results and Achievements

2.3.1 Performance Evaluation of Fugaku and A64FX manycore processor

We carried out the performance evaluation of Fugaku and A64FX manycore processor to validate our codesign effort for designing the system. The results were presented in Cluster 2020, SC20 and IEEE Micro (accepted).

2.3.1.1 Performance for Scientific Workload

The CloverLeaf hydrodynamics mini-app, taken from UK Mini-apps, solves Euler's equations of compressible fluid dynamics, under a Lagrangian-Eulerian scheme, on a two-dimensional spatial regular structured grid. The benchmark is a typical stencil code and is classified as a memory bandwidth-bound program. Figure 2.1 (A) shows the performance of CloverLeaf on A64FX (2.0GHz, single socket) in comparison with the dual sockets of Xeon Gold 6126 (Skylake, 2.6 GHz, 12 cores/socket) and Cavium Thunder X2 (TX2, 2.0 GHz, 28 cores/sockets). One MPI process with up to 12 threads in OpenMP runs on each CMG in the case of A64FX and each socket in case of Skylake and TX2. In the figure, the performance is shown relative to a single core of A64FX, changing the number of MPI processes and the number of cores for OpenMP in each MPI process. The performance of one chip of A64FX is better than that of 4 sockets of other processors thanks to the high memory bandwidth of HBM2. Figure 2.1 (B) shows the scalability of CloverLeaf using Fugaku up to 2048 nodes with one and two-dimensional distributions in a flat-MPI model. It shows good scalability in the 2D distribution.

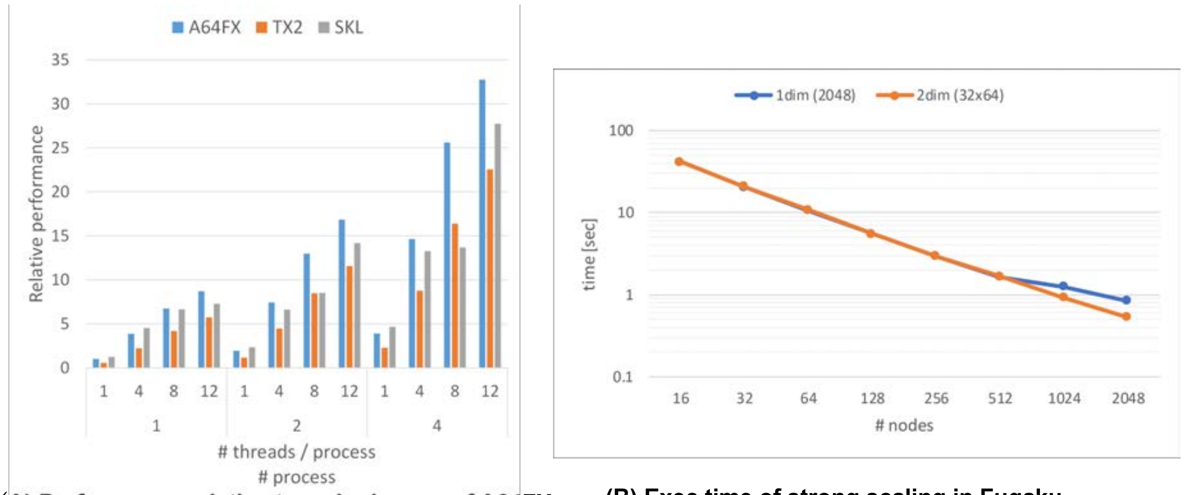
The results of other UK benchmarks were reported in [3].

Several open-source scientific applications are ported and evaluated on A64FX. Figure 2.2 shows the execution time and the average power of the A64FX (2.2GHz, single socket) relative to the dual-sockets of Xeon Platinum 8268 (CascadeLake, 2.90 GHz, 24 cores/socket) for open-source HPC applications shown in the figure. The results demonstrate that the power consumption of A64FX is about half that of the Intel Xeon's in most of these applications, while the performance of the A64FX is comparable to that of Intel.

2.3.1.2 SPEC benchmarks

SPEC benchmarks are well-known benchmarks suites for the evaluation of various kinds of computer systems. Table 2 shows the results for SPEC CPU (int) and SPEC OMP. For our evaluation, Speed (an index for evaluating the performance of single task) and Base (metrics when the compile option common to all benchmarks) were taken. For all benchmarks, A64FX runs at 2.0GHz (normal mode). ”Xeon” used for the SPEC CPU is Platinum 8168(Skylake), 2.7GHz, 24cores x 2 chip, turbo on. “Xeon” used for SPEC OMP is Platinum 8280(Cascade Lake), 2.7GHz, 28cores x 1chip, hyperthread on (56threads), turbo on.

The benchmark programs of SPEC CPU (int) run by a single thread. As shown in the results, the performance of A64FX is about one-quarter of the performance of the Xeon processor. The reason for the low single thread integer performance is that the SIMD rate is low in SPEC CPU (int) and the frequency and the O3 resource are limited for the throughput-oriented architecture of A64FX. As for SPEC OMP, the performance



(A) Performance relative to a single core of A64FX

(B) Exec time of strong scaling in Fugaku
(The input data is "clover_bm2048_shot.in")

Figure 2.1: Performance of CloverLeaf

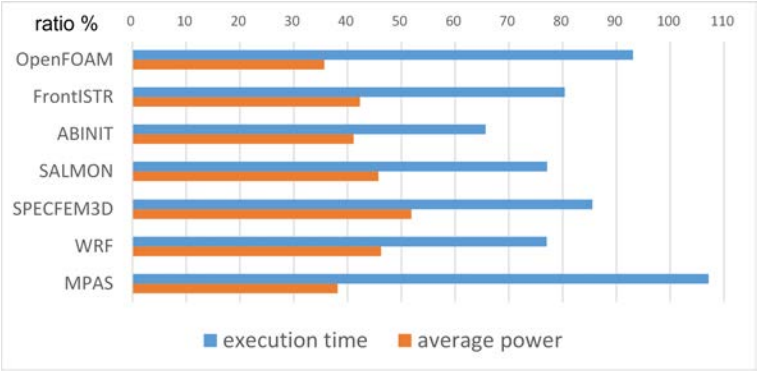


Figure 2.2: Execution time and average power of open-source applications using a single chip of A64FX (relative to a dual socket Intel Xeon)

of A64FX using 48 threads is about 65% of the performance of the Xeon processor using 56 threads (28 cores). For some programs such as 363.swim and 370.mgrid, A64FX archives extremely good performance thanks to the high memory bandwidth of the HBM2. Note that, although the performance of 350.md is very bad, the performance improvement has been confirmed by manual source code tuning such as in-lining and loop unrolling.

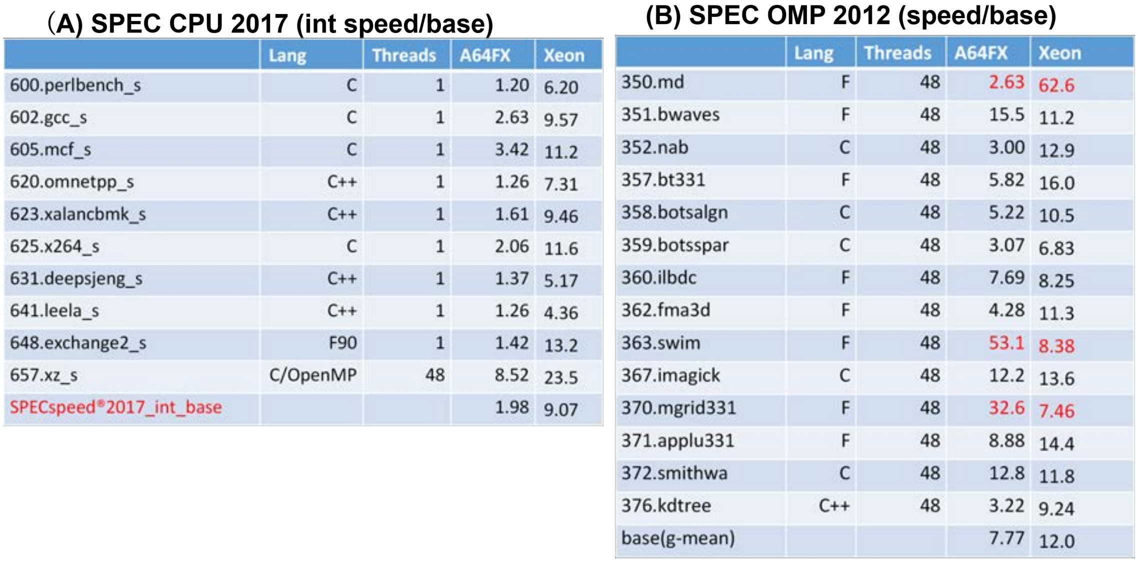
Other SPEC results of the A64FX are published in [4].

2.3.1.3 Effect of Eco Mode and Boost Mode

Eco-mode is useful to save the power for memory-intensive applications since this mode disables one SIMD arithmetic unit out of two by inactivating the stable operation for the stopped unit. Figure 2.2 shows the effect of the eco mode using the stream benchmark and the boost mode by dgemm operations. As shown in (A), the power is reduced by 15% to 25% compared to normal mode (no eco mode) while the performance is almost the same as that in normal mode. Note that the throughput with 24 threads hits 80% of peak memory bandwidth and the power also increase up to 24 threads. As shown in (B), while the performance of dgemm in boost mode is 10% better than that in normal mode (no boost mode), the power increases by 13.7%. As a result, the power efficiency decreases by 3.3%. More detailed analysis of the processor power consumption is described in [12]

2.3.2 Graph 500 Benchmark for Fugaku

We evaluated the performance of the breadth-first search (BFS) benchmark using the full system of Fugaku for Graph500 competition in collaboration with Kyushu University and Fixstars Corporation. As a results, we won



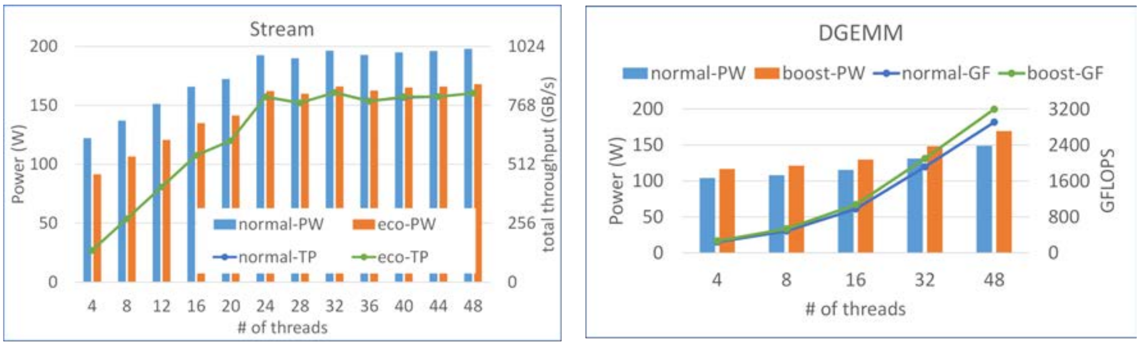
(A) SPEC CPU 2017 (int speed/base)

	Lang	Threads	A64FX	Xeon
600.perlbench_s	C	1	1.20	6.20
602.gcc_s	C	1	2.63	9.57
605.mcf_s	C	1	3.42	11.2
620.omnetpp_s	C++	1	1.26	7.31
623.xalancbmk_s	C++	1	1.61	9.46
625.x264_s	C	1	2.06	11.6
631.deepsjeng_s	C++	1	1.37	5.17
641.leela_s	C++	1	1.26	4.36
648.exchange2_s	F90	1	1.42	13.2
657.xz_s	C/OpenMP	48	8.52	23.5
SPECspeed®2017_int_base			1.98	9.07

(B) SPEC OMP 2012 (speed/base)

	Lang	Threads	A64FX	Xeon
350.md	F	48	2.63	62.6
351.bwaves	F	48	15.5	11.2
352.nab	C	48	3.00	12.9
357.bt331	F	48	5.82	16.0
358.botsalgn	C	48	5.22	10.5
359.botsspar	C	48	3.07	6.83
360.ilbdc	F	48	7.69	8.25
362.fma3d	F	48	4.28	11.3
363.swim	F	48	53.1	8.38
367.imagick	C	48	12.2	13.6
370.mgrid331	F	48	32.6	7.46
371.applu331	F	48	8.88	14.4
372.smithwa	C	48	12.8	11.8
376.kdtree	C++	48	3.22	9.24
base(g-mean)			7.77	12.0

Table 2.1: Results of SPEC benchmarks of A64FX processors (compared with Xeon)



(A) Triad of Stream benchmark

(B) Dgemm: Matrix-Matrix multiplication

Table 2.2: Effect of eco mode and boost-mode: performance and power consumption

the first position of Graph500 BFS in June/November of 2020.

While there is a growing demand for the fast processing of large-scale graphs in various fields, it is difficult to scale up graph processing performance on large-scale distributed memory systems due to their irregular computations. Against this background, Graph500, a ranking system used to evaluate largescale graph processing performance on supercomputer systems, has been releasing new rating lists twice-yearly since 2010. Our efforts have focused on improvements to breadth-first search (BFS), one of the kernels used in the Graph500 benchmark, resulting in the K computer being ranked first in the world. In this paper, we report on our efforts to further tune and evaluate the performance of BFS using the supercomputer Fugaku, which is the successor of the K computer. More specifically, we report on evaluating BFS performance for a large-scale graph consisting of about 2.2 trillion vertices and 35.2 trillion edges using all 158,976 Fugaku nodes. The obtained results show we achieved 102,956 giga-traversed edges per second (GTEPS), which is 3.3 times the K computer's performance. These results also demonstrate the high versatility of the supercomputer Fugaku because its irregular computations dominate BFS.

We implemented the parallel Hybrid-BFS algorithm for Fugaku. In the parallel Hybrid-BFS for distributed memory systems, the adjacency matrix A is assigned to the processes divided into two dimensions (R rows and C columns). A process $P(i, j)$ has information on a partial adjacency matrix $A_{i,j}$. The adjacency matrix A is symmetric because the graph in Graph500 is an undirected graph. If there is a non-zero element in the k column l row of the adjacency matrix A, it means that the k and l vertices are adjacent to each other. We performed the following optimization for Fugaku:

Optimization of the number of processes per node

We examined the optimum number of processes assigned to one node. The evaluation uses 1, 2, or 4 processes per node (denoted 1ppn, 2ppn, and 4ppn, respectively). The number of threads in each process is 48, 24, or 12. When the process shape is R x C, the job shape (Y , X) is (R, C) for 1ppn, (R, C/2) for 2ppn, and (R/2, C/2) for 4ppn. We found that that the performance difference becomes smaller as the number of nodes increases. Fig. 13 shows the time ratio of each BFS process for 1ppn and 4ppn. As the number of nodes increases, the rate of communication increases 1ppn has a smaller rate of communication than 4ppn. If the number of nodes is increased further, the communication ratio will increase. Thus, we select 1ppn, which can bring out the full communication performance.

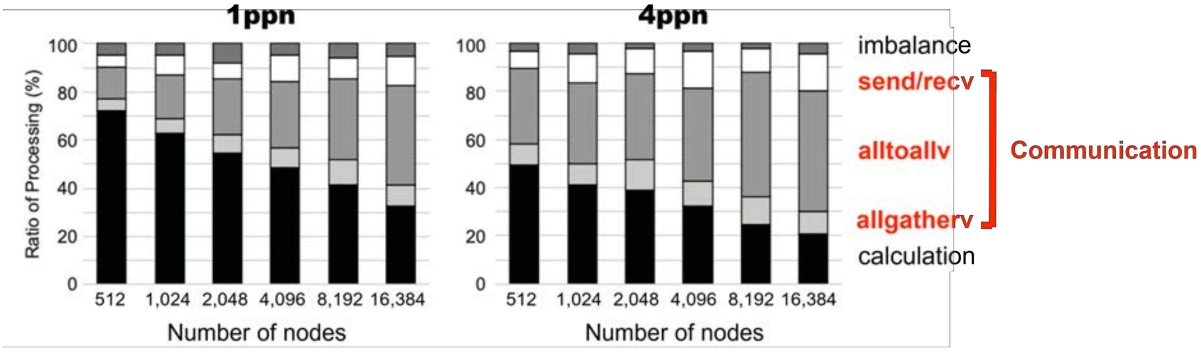


Figure 2.3: Break-down of processing time of 1ppn and 4ppn

Use of Eager method

In the point-to-point communication of most MPI implementations, the Eager and Rendezvous methods are implemented. Fujitsu MPI library on Fugaku can set the threshold for switching between Eager and Rendezvous methods We measured the performance when the Eager method is used for all sendrecv communications. Fig. 14 also shows the results for 1ppn. The switching threshold should be adjusted so that all sendrecv communications will use the Eager method. We changed the threshold to 512 Kbytes from default value to use Eager method. Since Fugaku's compute node has 32 Gbytes memory, the threshold is relatively small.

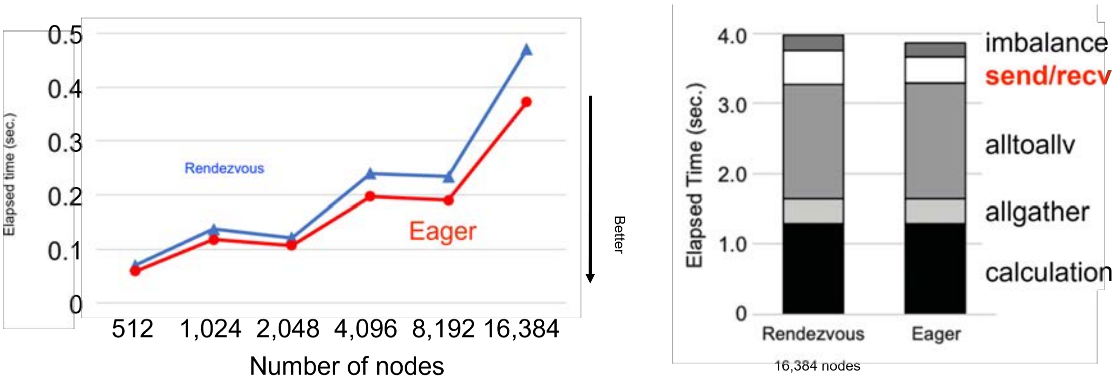


Figure 2.4: Comparison of Rendezvous and Eager

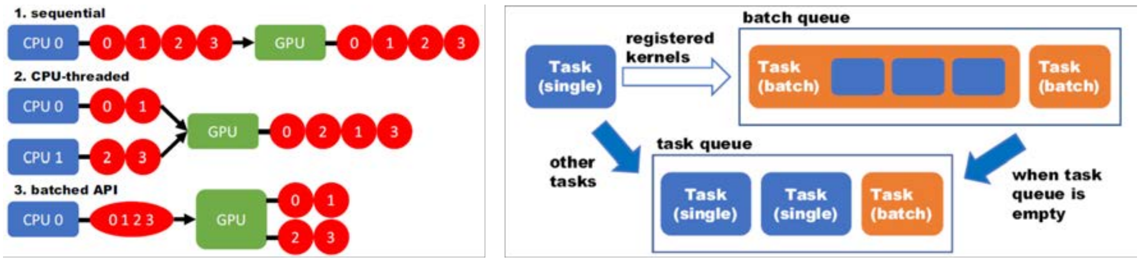
Six-dimensional process mapping

The size of six axes in Fugaku network is (X, Y, Z, a, b, c) = (23, 24, 23, 2, 3, 2) It is desirable that the values of R and C process grid of BFS are close We assign the processes to (R, C) = (XY, Zabc) = (552, 288) Since neighborhood communication occurs in BFS, we assign the processes physically next to each other in row/column dimension

2.3.3 OpenMP Task Generation for Batched Kernel APIs

The demand for calculating many small computation kernels is getting significantly important in the HPC area not only for the traditional numerical applications but also recent machine learning applications. While many-core accelerators such as GPUs are power-efficient compute platforms, a large amount of code modification is required. Batched kernel APIs such as batched BLAS can schedule numerical kernels efficiently on the target hardware while it still needs manual code modification.

We propose a code translation technique to generate batched kernel APIs in a high-level programming model. We use OpenMP task parallelism to specify dependency among numerical kernels. The user adds the task directives to specify tasks so that the compiler can recognize numerical kernels. The compiler detects conventional numerical kernels in the code and creates a unique batch ID for each kernel. When the task runtime detects tasks with the same batch ID, they are merged into a batch. Figure 2.5 shows the choices of APIs for GPU and our design of OpenMP task runtime for batched APIs.



(A) Choice of runtime-APIs for GPUs (B) Task Runtime design for batched APIs

Figure 2.5: Task Runtime for Batched Kernel APIs

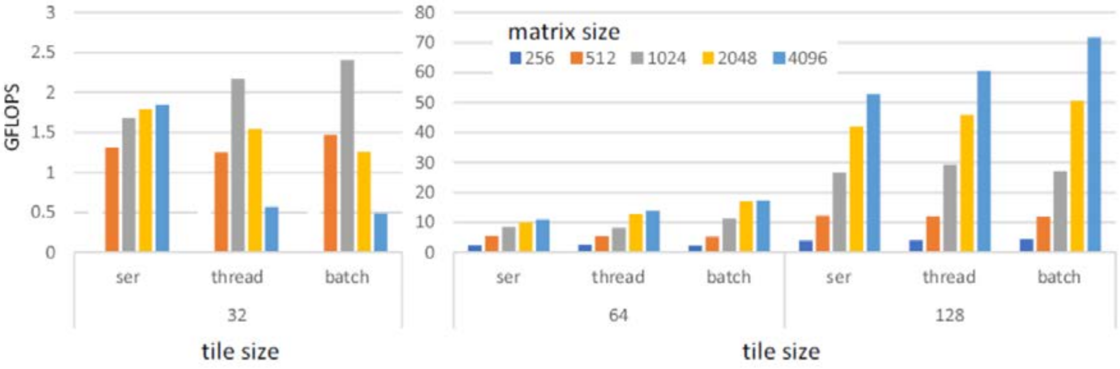


Figure 2.6: Performance of Blocked Cholesky Decomposition

The current implementation supports NVIDIA GPUs and batched BLAS in cuBLAS. DGEMM kernels can be detected and translated into batched DGEMM. A trivial DGEMM loop and blocked Cholesky decomposition code are used for performance evaluation. The evaluation result shows that batched DGEMM improves the performance when the matrix size is small and the number of DGEMM kernels is large. As shown in Figure 2.6, the time for DGEMMs in blocked Cholesky decomposition is 4 times faster than sequential execution when using batched DGEMM (40964096 matrix, tile size 128), however the overall performance is improved 36% because of task/batch management overhead.

2.3.4 Design of XcalableMP API: Compiler-free Approach

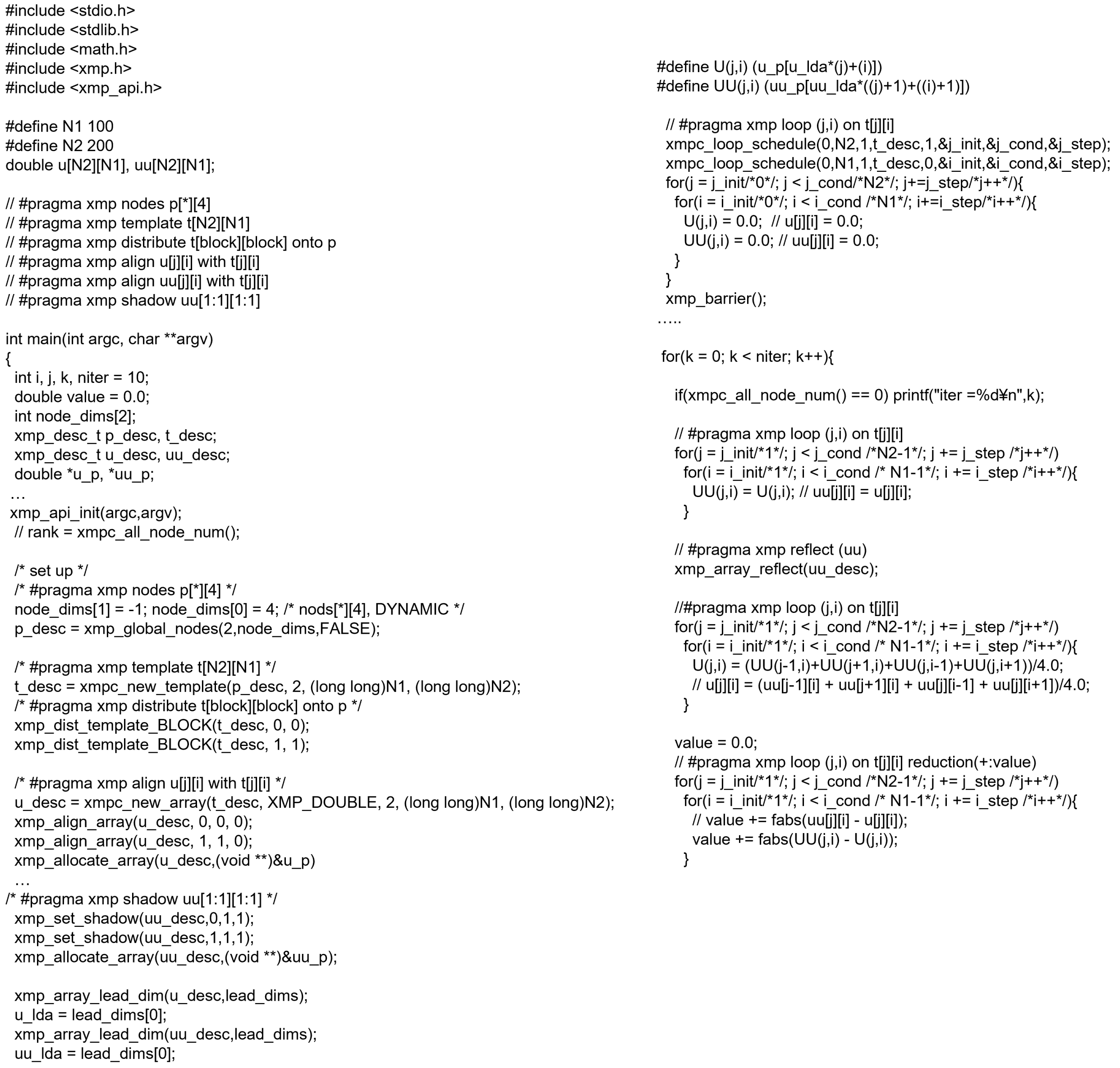
Although many PGAS languages such as UPC and Chapel, CAF, have proposed, it is hard to say that they are full accepted by the community of parallel programming. Recently, the libraries supporting the PGAS model, such as OpenShmem, GlobalArray, even MPI3 RMA, are getting popular for programming some specific applications. Furthermore, many C++ template-based design for PGAS, such as UPC++, DASH, are proposed as a compiler-free approach, as C++ template provides powerful abstraction mechanism. This approach may increase portability, clean separation from base compiler optimization, but a problem is that it is sometimes hard to debug in C++ template once a programmer writes wrong programs.

The approach of extending the language given by the support of the compiler, the compiler-approach, may give:

- A new language, or language extension provides easy-to-use and intuitive feature resulting in better productivity.
- This approach enables the compiler analysis for further optimization, such as removal of redundant sync and selection of efficient communication.

In reality, the compiler-approach is not easy to be accepted for deployment and supports in many sites, resulting in the failure of wide dissemination.

We designed the library interface for XcalableMP programming model, XMP API, which is aiming to provide the most equivalent programming functions by the set of libraries. Figure 2.7 shows an example of XMP API program for laplace explicit solver benchmark.



```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <xmp.h>
#include <xmp_api.h>

#define N1 100
#define N2 200
double u[N2][N1], uu[N2][N1];

// #pragma xmp nodes p[*][4]
// #pragma xmp template t[N2][N1]
// #pragma xmp distribute t[block][block] onto p
// #pragma xmp align u[j][i] with t[j][i]
// #pragma xmp align uu[j][i] with t[j][i]
// #pragma xmp shadow uu[1:1][1:1]

int main(int argc, char **argv)
{
  int i, j, k, niter = 10;
  double value = 0.0;
  int node_dims[2];
  xmp_desc_t p_desc, t_desc;
  xmp_desc_t u_desc, uu_desc;
  double *u_p, *uu_p;
 …
 xmp_api_init(argc,argv);
  // rank = xmpc_all_node_num();

  /* set up */
  /* #pragma xmp nodes p[*][4] */
  node_dims[1] = -1; node_dims[0] = 4; /* nods[*][4], DYNAMIC */
  p_desc = xmp_global_nodes(2,node_dims,FALSE);

  /* #pragma xmp template t[N2][N1] */
  t_desc = xmpc_new_template(p_desc, 2, (long long)N1, (long long)N2);
  /* #pragma xmp distribute t[block][block] onto p */
  xmp_dist_template_BLOCK(t_desc, 0, 0);
  xmp_dist_template_BLOCK(t_desc, 1, 1);

  /* #pragma xmp align u[j][i] with t[j][i] */
  u_desc = xmpc_new_array(t_desc, XMP_DOUBLE, 2, (long long)N1, (long long)N2);
  xmp_align_array(u_desc, 0, 0, 0);
  xmp_align_array(u_desc, 1, 1, 0);
  xmp_allocate_array(u_desc,(void **)&u_p)
  …
 /* #pragma xmp shadow uu[1:1][1:1] */
  xmp_set_shadow(uu_desc,0,1,1);
  xmp_set_shadow(uu_desc,1,1,1);
  xmp_allocate_array(uu_desc,(void **)&uu_p);

  xmp_array_lead_dim(u_desc,lead_dims);
  u_lda = lead_dims[0];
  xmp_array_lead_dim(uu_desc,lead_dims);
  uu_lda = lead_dims[0];
```

```
#define U(j,i) (u_p[u_lda*(j)+(i)])
#define UU(j,i) (uu_p[uu_lda*((j)+1)+((i)+1)])

  // #pragma xmp loop (j,i) on t[j][i]
  xmpc_loop_schedule(0,N2,1,t_desc,1,&j_init,&j_cond,&j_step);
  xmpc_loop_schedule(0,N1,1,t_desc,0,&i_init,&i_cond,&i_step);
  for(j = j_init/*0*/; j < j_cond/*N2*/; j+=j_step/*j++*/){
    for(i = i_init/*0*/; i < i_cond /*N1*/; i+=i_step/*i++*/){
      U(j,i) = 0.0;  // u[j][i] = 0.0;
      UU(j,i) = 0.0; // uu[j][i] = 0.0;
    }
  }
  xmp_barrier();
…..

 for(k = 0; k < niter; k++){

    if(xmpc_all_node_num() == 0) printf("iter =%d¥n",k);

    // #pragma xmp loop (j,i) on t[j][i]
    for(j = j_init/*1*/; j < j_cond /*N2-1*/; j += j_step /*j++*/)
      for(i = i_init/*1*/; i < i_cond /* N1-1*/; i += i_step /*i++*/){
        UU(j,i) = U(j,i); // uu[j][i] = u[j][i];
      }

    // #pragma xmp reflect (uu)
    xmp_array_reflect(uu_desc);

    //#pragma xmp loop (j,i) on t[j][i]
    for(j = j_init/*1*/; j < j_cond /*N2-1*/; j += j_step /*j++*/)
      for(i = i_init/*1*/; i < i_cond /* N1-1*/; i += i_step /*i++*/){
        U(j,i) = (UU(j-1,i)+UU(j+1,i)+UU(j,i-1)+UU(j,i+1))/4.0;
        // u[j][i] = (uu[j-1][i] + uu[j+1][i] + uu[j][i-1] + uu[j][i+1])/4.0;
      }

    value = 0.0;
    // #pragma xmp loop (j,i) on t[j][i] reduction(+:value)
    for(j = j_init/*1*/; j < j_cond /*N2-1*/; j += j_step /*j++*/)
      for(i = i_init/*1*/; i < i_cond /* N1-1*/; i += i_step /*i++*/){
        // value += fabs(uu[j][i] - u[j][i]);
        value += fabs(UU(j,i) - U(j,i));
      }
```

Figure 2.7: An example of XMP API (Laplace Explicit Solver)

We made a prototype implementation of XcalableMP for C++ using the LLVM tools libraries. We will integrate it to XMP API using advanced template libraries.

2.4 Schedule and Future Plan

We are working on the next version, XcalableMP 2.0, for cutting-edge high performance systems with manycore processors by multitasking with integrations of PGAS model and synchronization models for dataflow/multitasking executions. In this new programming model, the execution of the program is decomposed into several tasks executed according the dependency between tasks. This model enables less overhead of synchronization eliminating expensive global synchronization, overlap between computation and communication in manycore, and light-weight communication by RDMA in PGAS model. Especially for Fugaku, this model allows to exploit parallelism for both tasks and SIMD. We have a plan to implement OpenCL and oneAPI for Fugaku.

The task-based programming recently supported in OpenMP 4.0 enables to expose a lot of parallelism by executing several tasks of the program in the form of task-graph. To accelerate the task-based parallel program by accelerators such as GPU and FPGA, it is useful for some tasks frequently executed in parallel to be offloaded to accelerators as an asynchronous task executed by accelerators. The next step will be that this global task parallel programming model is extended to tasks offloaded to accelerators attached to each node in accelerated clusters.

Exploration of new high-performance architectures from programing model's point of view is an important challenge. Future parallel architecture will be more heterogenous having many kinds of accelerators and devices attached to the nodes and directly connected between accelerators by some dedicated interconnect. To program such a complex and heterogenous parallel system, the global task parallel programming model will give a flexible and decomposable model to exploit such heterogenous high performance architecture.

2.5 Publications

2.5.1 Articles/Journal

[1] Hitoshi Murai, Mitsuhisa Sato: Design and evaluation of efficient global data movement in partitioned global address space. Parallel Comput. 96: 102624 (2020)

2.5.2 Conference Papers

[1] Masahiro Nakao, Koji Ueno, Katsuki Fujisawa, Yuetsu Kodama, Mitsuhisa Sato: Performance Evaluation of Supercomputer Fugaku using Breadth-First Search Benchmark in Graph500. CLUSTER 2020: 408-409
[2] Yuetsu Kodama, Tetsuya Odajima, Eishi Arima, Mitsuhisa Sato: Evaluation of Power Management Control on the Supercomputer Fugaku. CLUSTER 2020: 484-493.
[3] Tetsuya Odajima, Yuetsu Kodama, Miwako Tsuji, Motohiko Matsuda, Yutaka Maruyama, Mitsuhisa Sato: Preliminary Performance Evaluation of the Fujitsu A64FX Using HPC Applications. CLUSTER 2020: 523-530.
[4] Yuetsu Kodama, Masaaki Kondo, Mitsuhisa Sato: Evaluation of SPEC CPU and SPEC OMP on the A64FX. CLUSTER 2021: 553-561

2.5.3 Invited Talks

[1] Mitsuhisa Sato, The Supercomputer "Fugaku" and Software, programming models and tool, Extreme-scale Scientific Software Stack E4S Forum 2020, Sep 24th 2020,
[2] Mitsuhisa Sato, 様々な演算加速機構へのオフロード記述するタスク並列モデル FPGA-HPC-WS2021, 筑波大、2月26日 2021年

2.5.4 Software

- Omni XcalableMP compiler ver. 1.3.3, Nov. 27, 2020 (registered as an R-CCS-supported software) `https://omni-compiler.org/`

Chapter 3

Processor Research Team

3.1 Members

Kentaro Sano (Team Leader)

Tomohiro Ueno (Postdosctoral Researcher)

Takaaki Miyajima (Postdosctoral Researcher)

Jens Christoph Huthmann (Postdosctoral Researcher)

Atsushi Koshiba (Postdosctoral Researcher)

Boma A. Adhi (Postdosctoral Researcher)

Mitsuo Kiyono (Research Assistant)

Hiroyuki Takizawa (Senior Visiting Researcher, Tohoku University)

Yuichiro Shibata (Senior Visiting Researcher, Nagasaki University)

Masahiro Iida (Senior Visiting Researcher, Kumamoto University)

Yasushi Inoguchi (Senior Visiting Researcher, Japan advanced institute of science and technology)

Hideki Takase (Visiting Researcher, Kyoto University)

Kazuya Tanigawa (Visiting Researcher, Hiroshima City University)

Kojima Takuya (Intern, and Research Parttimer, Keio University)

Kouki Watanabe (Student Trainee, Tohoku University)

Satoshi Kaneko (Student Trainee, Tohoku University)

Shunsuke Into (Student Trainee, Kumamoto University)

Yasuhiro Nakahara (Student Trainee, Kumamoto University)

Hiroki Tada (Student Trainee, Japan advanced institute of science and technology)

Shota Fukui (Student Trainee, Nagasaki University)

Tatsuma Mori (Student Trainee, Nagasaki University)

Daiki Furukawa (Student Trainee, Nagasaki University)

Keigo Motoyoshi (Student Trainee, Nagasaki University)

Ryota Miyagi (Student Trainee, Kyoto University)

Tomoko Nakashima (Assistant)

Keiko Inaba (Assistant)

3.2 Overview of Research Activities

3.2.1 Aim of Team

The processor research team explores data-driven parallel computing models and high-performance computer architectures in order to establish next-generation computing technologies which are promising and necessary as next-generation computing technologies in the forthcoming post-Moore era. Especially we are researching and developing reconfigurable high-performance computing hardware, which are based on the data-flow computing model, its system including hardware and software stacks, and application-specific hardware designs of FPGA-based custom computing machines. Some of them are conducted as a research aiming at advancement of the supercomputer Fugaku as its functional extension with FPGA cluster.

So far, planar lithography-scaling known as the Moore's law has continuously improved the performance and/or performance per power of computer systems. However, it is now foreseen for the Moore's law to end sooner or later in the next decade, and then we will be no longer able to rely on two-dimensional scaling of CMOS devices. In the "post-Moore" era, transistor integration, power consumption per transistor, and relative latency of data movement to the transistor's switching speed cannot be sufficiently improved. Consequently, it is predicted that the conventional computer architectures would not be able to further increase the performance and performance per power at a level that we have done so far with the Moore's law.

Accordingly, we will need to more efficiently and effectively utilize available hardware resources, i.e., transistors on chips, to achieve target performance. In particular, the conventional many-core architectures and large-scale systems based on the parallel computing model and global synchronization will be confronted with limitation in increasing computing performance due to:

1) dark silicon problem where most of transistors cannot be utilized due to the upper limit of on-chip power consumption. Even if we could have more transistors integrated on a chip, we cannot use most of them if power consumption per transistor does not decrease,

2) relatively-increasing latency to transistor's switching speed. Due to the increasing latency, we can no longer shorten cycle time to update memory elements for computing and control a data path based on a previous operation, and synchronization time among a large number of physically-distributed processors,

3) inefficient data-movement among on-chip cores via a memory subsystem or through a global network in a system, and

4) a large overhead in global synchronization for large-scale parallel computation which is affected by relatively-increasing delay of data transfer through a system-wide network.

That is, the existing approaches/architectures are not designed to scale the performance under these critical conditions. For example, the von-Neumann architecture is based on two cycles of "memory-element update" and "control" which cannot be accelerated any more, and therefore parallel processing is introduced as pipelining, super-scalar, and many cores as well as latency-hiding techniques of memory hierarchy with cache memories, speculative execution with branch prediction, and simultaneous multi-threading. These additional mechanisms make semiconductor resource utilization much worse for target processing and computing. In addition, the global barrier in parallel computation degrades overall performance as more nodes are utilized.

We believe that the reconfigurable computing with custom hardware of spatially-mapped algorithms (spatial computing) is promising to efficiently utilize hardware resources for target computation while such architectural redundancy as is seen in a field-programmable gate array (FPGA) device is also suitable for the dark silicon problem. The spatial computing with the data-driven model (or data-flow model) allows us to remove or reduce the cycles in data-paths. By spreading a sequence of operations onto space as a data-driven hardware, we can avoid instruction execution cycles with memory-element update and control so that we can increase a computing throughput naturally with a pipelining techniques and fine-grain parallelism.

The data-flow or task-flow approach can also make it easier to avoid the global synchronization in large-scale parallel computing. If we automatically schedule and control task execution based on task-flow with task dependency, we can efficiently execute tasks without global synchronization when they become ready to be executed. Under these consideration, we are researching the data-driven spatial computing which is essentially necessary for future computer architectures in the post-Moore era.

Since researches on these themes require broad range of expertise, we are collaborating with other research teams in R-CCS, universities in Japan (Tohoku university, University of Tsukuba, Nagasaki university, Kumamoto university, Hiroshima city university, Kyoto university, and Japan advanced institute of science and technology; JAIST), and a research institute out of Japan, such as Argonne national laboratory, US.

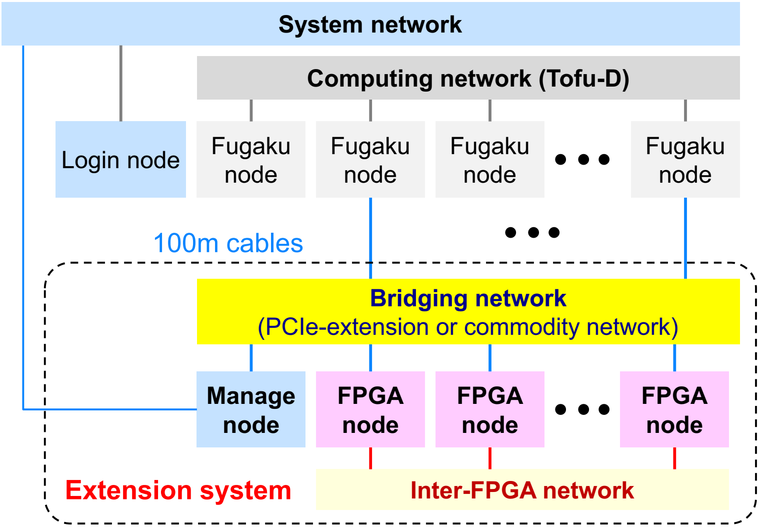


Figure 3.1: Architecture for functional extension of Fugaku.

3.2.2 Overview of Research Activities

In the fiscal year of 2020, we have conducted the following researches for the aim of our team.

1. Elastic and scalable system for high-performance reconfigurable computing (ESSPER)
2. FPGA-based applications
3. Initial design and simulation of Riken CGRA (coarse-grained reconfigurable array)

The background, motivation, objectives, and achievement of each research subject are described in the next section.

3.3 Research Results and Achievements

3.3.1 Elastic and scalable system for high-performance reconfigurable computing (ESSPER)

3.3.1.1 Overview

In order to clarify the technical issues and solutions for functional extension of the supercomputer Fugaku to give an additional value to it, we have developed the Elastic and scalable system for high-performance reconfigurable computing (ESSPER), which is an experimental cluster system of FPGAs (Field-Programmable Gate Arrays). For functional extension of Fugaku, we assume that FPGA, which is a circuit-reconfigurable semiconductor, is promising as an extension device connected by PCI interface of Fugaku. We studied its architecture, a hardware and software system, and operations with applications.

FPGA contains logic elements composed of look-up tables, on-chip memories, arithmetic units named DSP blocks, and I/O blocks, as well as many wiring and switch blocks to connect them arbitrarily. Since FPGAs can construct any circuits using those hardware resources, they are expected to realize high-performance and highly-efficient computation by constructing optimized circuits and memory systems, tailored for computational problems especially that general-purpose microprocessors are not good at. Therefore, in addition to the original use cases of FPGAs, such as emulation of glue logic and VLSI, and hardware in cell-phone base stations, FPGAs have been investigated as a device to build high-performance reconfigurable systems as an acceleration mechanism for processing that is insufficient for CPUs.

In particular, in recent years, FPGAs, which are manufactured with the most advanced semiconductor technology and combined with the most advanced memory such as HBM2, are becoming larger and more powerful than ever before. The adoption of FPGAs in data centers and industrial applications is increasing especially for high-speed and low-power processing of massive search, image processing, and machine learning. In addition, attempts are being made to build HPC systems for applications of scientific computing.

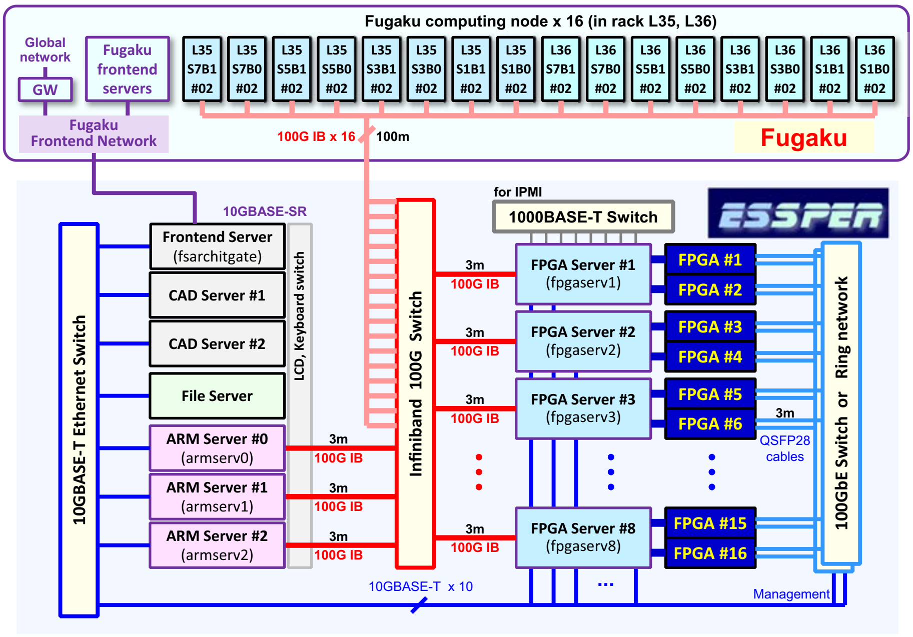


Figure 3.2: Organization of FPGA Cluster ”ESSPER” and its connection to Fugaku.

3.3.1.2 Architecture and system design

Since we have a constrain of space and power in the Fugaku rack for extension, we adopted the architecture of Fig.3.1 where an extension system is installed out of Fugaku, and connected with Fugaku by using a bridging network. As the bridging network, we can consider a network dedicated to extend PCI-Express slots or a commercially-available network such like Infiniband network. Due to flexibility, portability, and sustainability of the technology, we decided to use a commercially-available network and develop a software-bridged driver for FPGAs.

Based on the architecture, we have developed an experimental system of FPGA cluster, named ESSPER (Elastic and scalable system for high-performance reconfigurable computing), which is shown in Fig.3.2. ESSPER has eight x86-CPU servers, each of which hosts two FPGA cards. The FPGA card has Intel Stratix 10 FPGA 2800SX, which is a large-scale and high-end FPGA device fabricated in Intel’s 14nm technology. FPGAs are directly connected with their dedicated network of 100Gbps. We are researching two different types of inter-FPGA networks: the direct network with a ring topology, and the indirect network with 100Gbps Ethernet switches. In the system, we also have several servers such as CAD servers, and ARM CPU servers.

The FPGA host servers are connected to the EDR Infiniband switch with 100Gbps optical cables. We connected ESSPER with Fugaku with the Infiniband switch and sixteen 100m optical cables. Fig.3.3 is the photograph of Fugaku racks and the installed ESSPER.

3.3.1.3 System stack of ESSPER

We have also developed the system stack of ESSPER as shown in Fig.3.4. We have implemented a system-on-chip (SoC) for the FPGA, referred to as *AFUShell*, where the memory interfaces, the PCI-Express interfaces, and the network interfaces are implemented as well as the crossbar and DMA modules. We can embed our own application modules in the AFUShell. On the top of the AFUShell, we have an open-source PCI-Express driver for the FPGA, OPAE (Open Programmable Acceleration Engine), which is developed by Intel. Above OPAE, we developed the DMA library and R-OPAE (remote OPAE). R-OPAE is the software bridge used to remotely use the FPGA cards from Fugaku via Infiniband network. R-OPAE is composed of a R-OPAE daemon running on FPGA host servers, and R-OPAE library which is called by a user program running on Fugaku. We transplanted 99 % of OPAE APIs to R-OPAE except mmap related functions. By linking the R-OPAE library instead of OPAE library, we can access AFUShells on remote FPGAs as if the FPGAs are locally installed in Fugaku computing nodes.

On the top of OPAE and R-OPAE, we have also developed the afushell class, which provides a hardware abstraction layer for user programs. When we use APIs of OPAE or R-OPAE directly, we need to write many

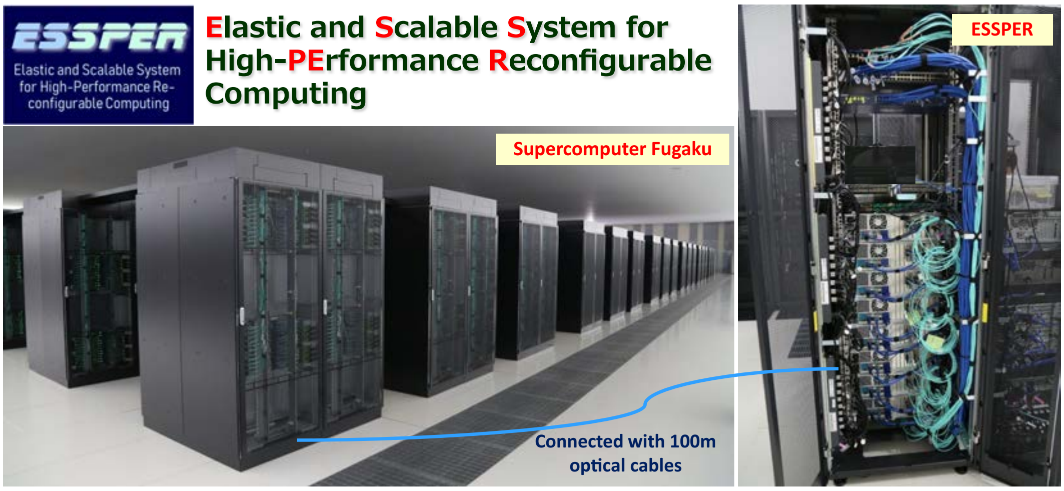

ESSPER
Elastic and Scalable System for High-Performance Re-configurable Computing
Elastic and Scalable System for High-PErformance Reconfigurable Computing
Supercomputer Fugaku
ESSPER
Connected with 100m optical cables

Figure 3.3: Photographs of ESSPER and Fugaku.

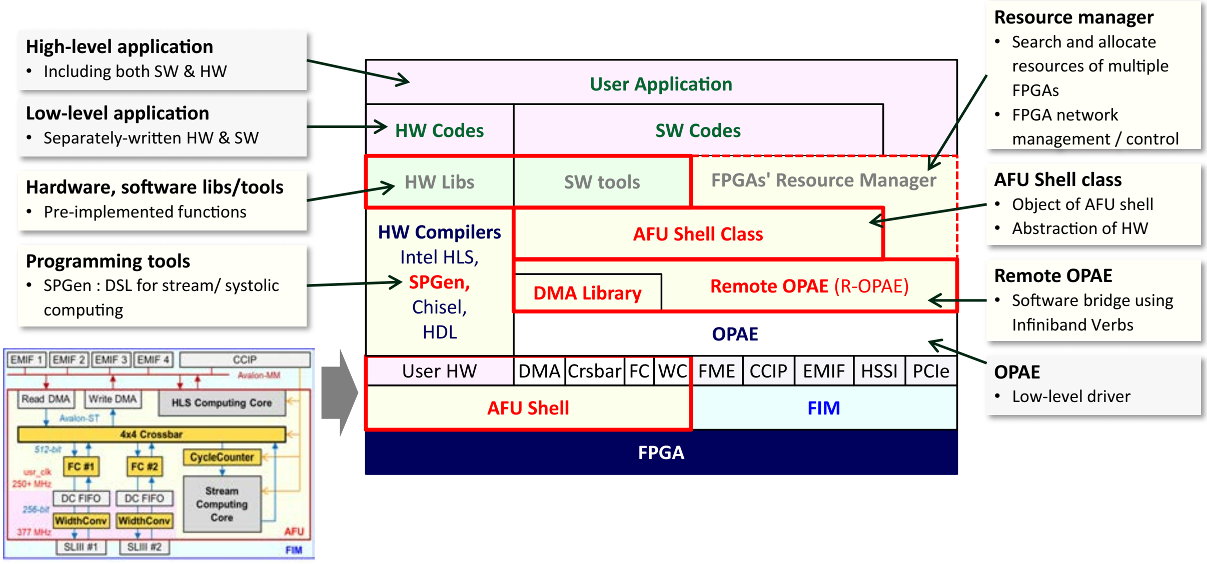


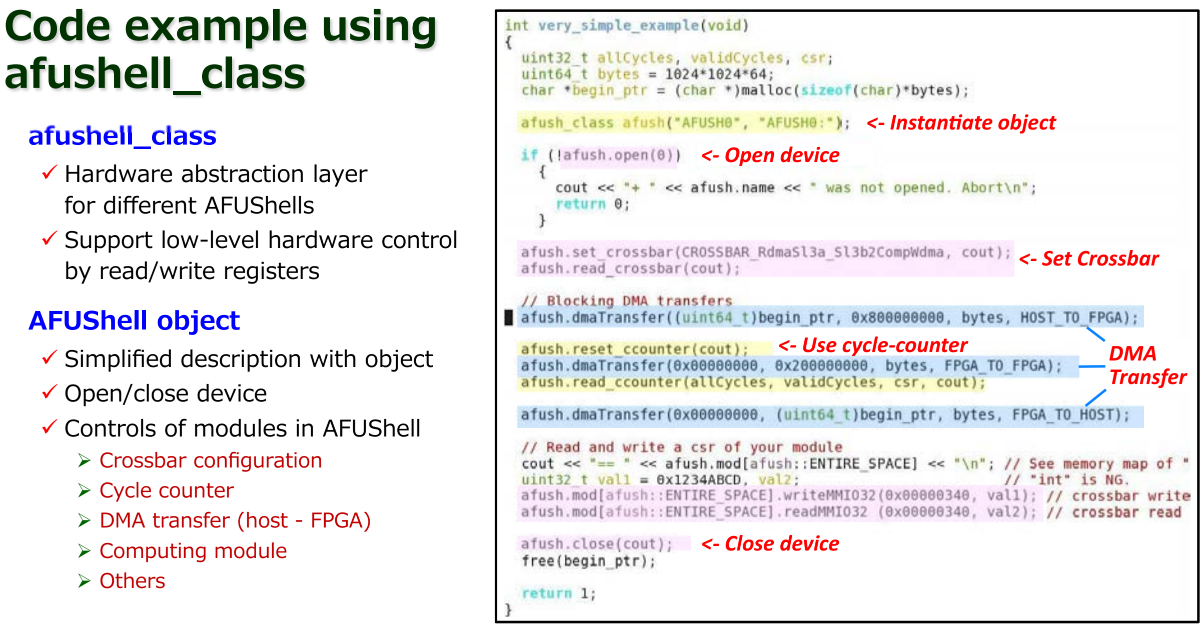
Figure 3.4: System stack of ESSPER.

lines for operations such as searching devices or resources, opening them, controlling modules in AFUShell, and so on. In addition, if we use similar but different AFUShell implemented for another FPGA card, we need to change parameters such as the addresses of embedded hardware modules in the AFUShell. In order for users to easily write their programs with an appropriate level of abstraction, the afushell class provides an object of AFUShell with its abstracted methods for control. Fig.3.5 shows an example code using the object of the afushell class, where we can open the AFUShell, set crossbar, and execute DMA transfers by using a small number of lines of simple descriptions easy to understand.

Moreover, by switching a mode from OPAE to R-OPAE, the same user code can be compiled for R-OPAE library. Although the example of Fig.3.5 does not contain a code to program FPGAs with bitstreams of circuit configuration, we can do it for both OPAE and R-OPAE. As far as I know, it is the first case that we can program FPGAs in a software running on a huge-scale supercomputer such as Fugaku.

3.3.1.4 Verification of task off-loading from Fugaku to ESSPER

We verified that we can successfully off-load tasks to FPGAs in a software running on Fugaku computing nodes. For this, we have implemented and embedded an example application module into the AFUShell, and wrote and compiled a user code such like Fig.3.5 linked with R-OPAE library, where we use the AFUShell module on remote FPGAs. We could successfully program an FPGA with a bitstream file of about 100MB for about two



Code example using afushell_class

afushell_class

- ✓ Hardware abstraction layer for different AFUShells
- ✓ Support low-level hardware control by read/write registers

AFUShell object

- ✓ Simplified description with object
- ✓ Open/close device
- ✓ Controls of modules in AFUShell
  - ➢ Crossbar configuration
  - ➢ Cycle counter
  - ➢ DMA transfer (host - FPGA)
  - ➢ Computing module
  - ➢ Others

```
int very_simple_example(void)
{
  uint32_t allCycles, validCycles, csr;
  uint64_t bytes = 1024*1024*64;
  char *begin_ptr = (char *)malloc(sizeof(char)*bytes);

  afush_class afush("AFUSH0", "AFUSH0:");

  if (!afush.open(0))
    {
      cout << "+ " << afush.name << " was not opened. Abort\n";
      return 0;
    }

  afush.set_crossbar(CROSSBAR_RdmaSl3a_Sl3b2CompWdma, cout);
  afush.read_crossbar(cout);

  // Blocking DMA transfers
  afush.dmaTransfer((uint64_t)begin_ptr, 0x800000000, bytes, HOST_TO_FPGA);

  afush.reset_ccounter(cout);
  afush.dmaTransfer(0x00000000, 0x200000000, bytes, FPGA_TO_FPGA);
  afush.read_ccounter(allCycles, validCycles, csr, cout);

  afush.dmaTransfer(0x00000000, (uint64_t)begin_ptr, bytes, FPGA_TO_HOST);

  // Read and write a csr of your module
  cout << "== " << afush.mod[afush::ENTIRE_SPACE] << "\n"; // See memory map of "
  uint32_t val1 = 0x1234ABCD, val2;                        // "int" is NG.
  afush.mod[afush::ENTIRE_SPACE].writeMMIO32(0x00000340, val1); // crossbar write
  afush.mod[afush::ENTIRE_SPACE].readMMIO32 (0x00000340, val2); // crossbar read

  afush.close(cout);
  free(begin_ptr);

  return 1;
}
```

Figure 3.5: Porgraming with afushell class.

seconds.

3.3.1.5 Evaluation evaluation of data-transfer between Fugaku to ESSPER

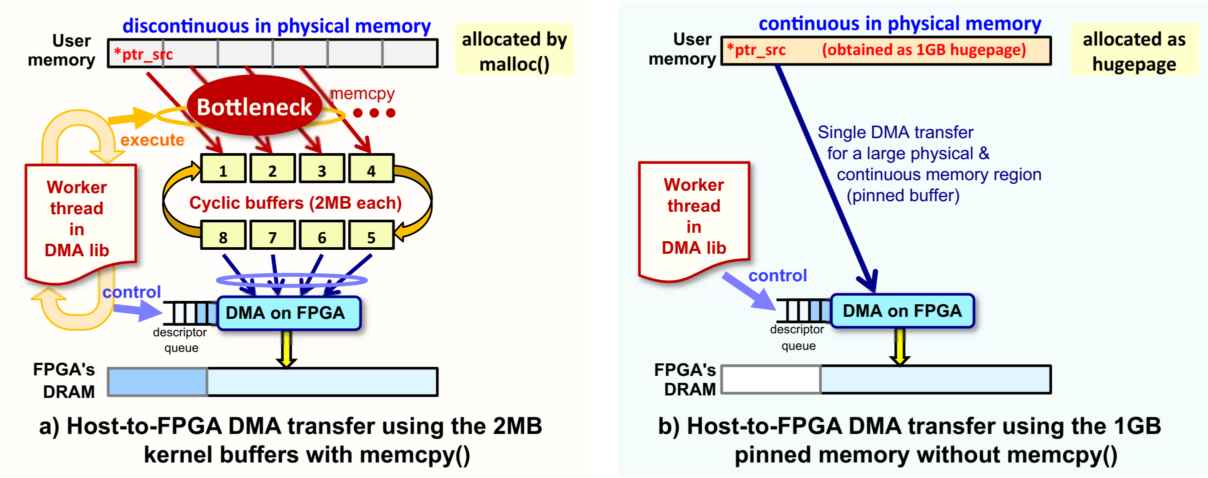
Then we tested and evaluated the data transfer between a CPU and the DDR4 memory of the FPGA card for OPAE or R-OPAE. For local data-transfer using OPAE, we have implemented the two types of data transfer shown in Fig.3.6. The first one of Fig.3.6a uses 2MB buffers in the kernel memory space, and memcpy() function to copy the chunks of data in the user program's memory region to the buffers, in the case of the host-to-FPGA transfer. This is necessary because the DMA modules on FPGA can only access the physical memory space of the host CPU, instead of the virtual memory space for an individual user process. However, the user program's memory regions in the virtual memory space are discontinuous, and distributed in the CPU's physical memory space by a unit of a 4KB page size. Therefore, for the DMA modules to continuously read and write some size of data, we need to prepare some size of buffers which are allocated continuously in the physical memory space, and copy data from the user program's regions to the buffer before the DMA module reads them.

This memory-to-buffer copy is typically implemented by memcpy() function and controlled by a worker thread which is dedicated to control and monitor the DMA module. Even if the program is well optimized, the memcpy for the buffers becomes a bottleneck which easily limits the data-transfer performance. If only a single core is used for memcpy(), only a portion of the full bandwidth given by the multiple memory channels is available, and therefore the throughput of memcpy can be lower than than that of the PCI-Express.

To avoid the problem, we implemented the second mode of Fig.3.6b, where we use a 1GB hugepage which is pinned in the physical memory space as a continuous region. By passing its pointer to the user program instead of using malloc(), the user program and the DMA modules on FPGA can share the same continuous memory region. As a result, we eliminate the bottleneck of memcpy(), and the DMA modules directly read and write the entire memory region with a single data-transfer command.

Figs.3.7a and b show the performance of local data transfer between the host CPU and FPGA on the FPGA server using OPAE. The horizontal axis is the transfer data size, and the vertical axis is the effective data transfer bandwidth. The data size is varied from 256 Bytes to 1 GByte. In the figure, "x86, pinned" indicates memcpy using the pinned memory buffer of Fig.3.6b. "x86, memcpy", "x86, sse2", and "x86, ymm" are of Fig.3.6a, which use the memcpy() function of a standard library in Linux, the memcpy() function implemented with the SSE2 instructions of x86 CPUs, and the memcpy() function implemented with YMM registers of x86 CPUs.

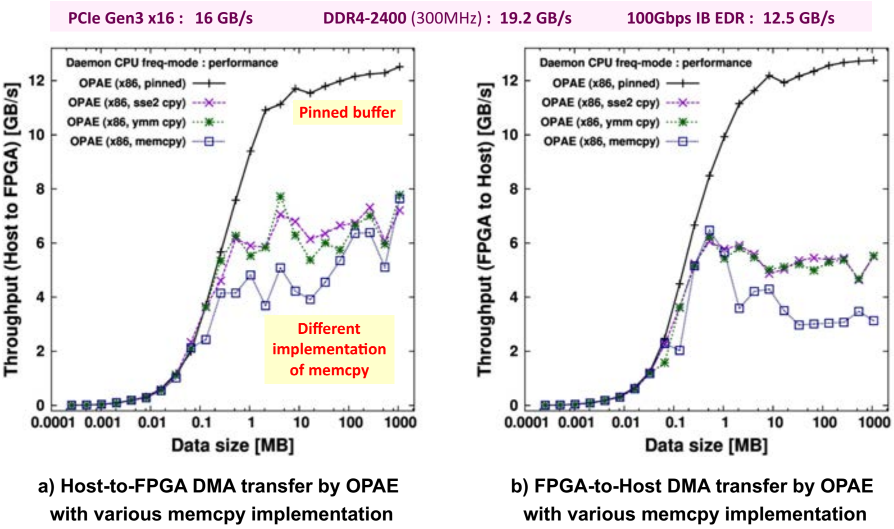
The higher effective bandwidth as the data size increases is due to the inevitable overhead in controlling the DMA module for data transfer and in detecting the end of the DMA transfer by interrupts. The pinned mode achieved a maximum effective bandwidth of 12.5GB/s for a 1GB data transfer. This is 78 % of the theoretical



a) Host-to-FPGA DMA transfer using the 2MB kernel buffers with memcpy()

b) Host-to-FPGA DMA transfer using the 1GB pinned memory without memcpy()

Figure 3.6: DMA Transfer mechanism with and without kernel buffers and memory copy.



a) Host-to-FPGA DMA transfer by OPAE with various memcpy implementation

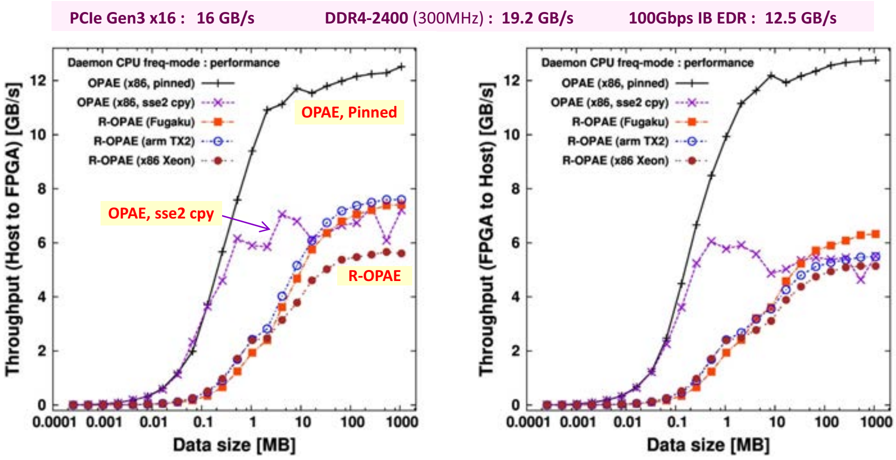
b) FPGA-to-Host DMA transfer by OPAE with various memcpy implementation

Figure 3.7: DMA Transfer performance using OPAE for the various modes.

peak bandwidth of 16 GB/s for PCIe Gen3 x16, which is sufficiently high accounting for the encoding and other overheads in PCIe. On the other hand, the transfers using kernel buffers result in an effective bandwidth of up to 7 to 8 GB/s for transfers from the host CPU to the FPGA and 3.5 to 6 GB/s for transfers from the FPGA to the host CPU, which is 2/3 and 1/2 of the bandwidth by the pinned mode, respectively. In particular, up to a data size of 512 KByte, their performance is comparable to that of using a the pinned mode, and beyond that point, the bandwidth does not increase smoothly but fluctuates significantly. In addition, it was confirmed that the bandwidth was higher when memory copy was optimized by SSE2 and YMM than when memcpy() function was used.

3.3.1.6 Comparison of data-transfer performance for OPAE and R-OPAE

Figs.3.8a and b show the performance of data transfer using OPAE or R-OPAE, from the host CPU to the FPGA and from the FPGA to the host CPU, respectively. For R-OPAEs, the effective bandwidths via the 100 Gb/s Infiniband EDR switch are shown for Fugaku node (Fugaku), the server with ThunderX2 CPU from Cavium (arm TX2), and the FPGA server with x86 CPU from Intel (x86 Xeon). In the case of transfer by R-OPAE, the RDMA transfer in the Infiniband network and the DMA transfer on the FPGA are overlapped to reduce the transfer time. The transferred data is divided into 2MB segments to pipeline the RDMA transfer in the Infiniband and the DMA transfer on the FPGA. In addition, in the FPGA server, the Infiniband network interface card is implemented to write the received data directly into a buffer in the kernel memory space. For



a) Host-to-FPGA DMA transfer by OPAE or R-OPAE

b) FPGA-to-Host DMA transfer by OPAE or R-OPAE

Figure 3.8: DMA Transfer performance using OPAE and R-OPAE.

this reason, the DDR4 memory bandwidth problem of the OPAE memory copy version of the DMA transfer library is mitigated. In addition, a 100-meter optical cable is used between Fugaku nodes and the Infiniband switch, and a 3-meter optical cable is used for all other connections. Since 5 nsec of delay is introduced per 1-meter optical cable, an additional 485 nsec of delay is introduced in the case of transfer with Fugaku nodes.

The theoretical peak bandwidth of the 100Gb/s EDR Infiniband network is 12.5 GB/s. As shown in Fig.3.8, the data transfer bandwidth from the R-OPAE host CPU to the FPGA by Fugaku and ARM server increases with the data size, reaching approximately 7.5 GB/s for a 1 GB transfer. This is 60 % of the peak bandwidth of the Infiniband network, which can be considered as an acceptable performance. In particular, although the effective bandwidth increases slowly with the data size, the maximum reachable effective bandwidth is comparable to the DMA transfer performance of OPAE with the memcpy mode. This means that practical data-transfer performance can be achieved even in the case of R-OPAE.

The slightly lower transfer bandwidth of Fugaku than that of the ARM server is considered due to the delay caused by the 100m cable. On the other hand, the transfer performance of R-OPAE by the x86 server shows a similar trend, but the performance improvement slows down when the transfer data size exceeds 2 MByte, and the effective transfer bandwidth remains at about 5.6 GB/s even in the case of 1 GByte transfer. The cause of this is still under investigation. The transfer bandwidth from the FPGA to the host CPU in Fig.3.8b shows a similar trend to Fig.3.8a. The reason why the bandwidth is generally lower than that from the host to the FPGA is that the R-OPAE client reads data from the server on which the R-OPAE daemon operates. In general, reading data is more susceptible to the latency of the communication channel than writing data, and thus the effective bandwidth is likely to decrease.

3.3.2 FPGA-based applications

ESSPER aims to off-load tasks that many-core processors are not good at to FPGAs to speed up them. In order to achieve this, it is necessary to find an application or a kernel which is advantageous for processing speed in implementing as a dedicated circuit.

Potentially advantageous problems for FPGAs are considered to have the following characteristics:

1. computational problems that can be accelerated by hardware algorithms with high parallelism, locality, and regularity, such as pipelining, systolic arrays, data flow, and stream processing,

2. computational problems that use special arithmetic operations or special precision, which the microprocessor is not equipped with a hardware arithmetic unit for,

3. computational problems that can be accelerated by highly efficient data access, movement, and reuse with a dedicated memory system,

4. computational problems that require low latency processing, such as feedback control,

Table 3.1: Candidate problems that are potentially advantageous for FPGA-based computing.

Problems (programming)	Acceleration by hardware algorithms	Special operations, special precision	Efficient access, reuse, and movement of data	Beneficial by low-latency processing	Strong scaling
Stencil computing (SPGen)	◎		○		○
N-body (SPGen)	◎		○		○
3D FFT (OpenCL/IP core)	◎		◎		○
Graph processing (Intel HLS, Nymble)	△	○	△		△
Genome data processing (OpenCL)	◎	◎	◎		○
ML inference (OpenCL,RTL,etc)	○	◎	○		○
ML training (OpenCL,RTL,etc)	○	◎	○	○	○
Data-base processing* (OpenCL,etc)	○	◎	○		○

◎:Very promising, ○:Promising, △:Need investigation, none:not promising *reg-exp match, sort, nearest neighbor search

5. computational problems that can be strongly scaled by highly efficient communication and synchronization using a dedicated network between FPGAs.

Table 3.1 summarizes the above types of computational problems. Designing and implementing a dedicated circuit specialized and optimized for each computational problem is a more difficult task than software programming because it requires exploration of the hardware architecture. Therefore, we focused on and implement a limited number of problems for which we have good ideas of the hardware architecture and optimization.

At the present stage, we have been working on the following applications to design and implement their dedicated hardware for FPGAs.

1. stream processing of stencil and many-body problem computations,
2. 3D FFT,
3. genome data processing
4. Database processing (graph breadth-first search)

3.3.3 Initial design and simulation of Riken CGRA (coarse-grained reconfigurable array)

In this research project, we have been exploring a new non-Neumann-type computer architecture and its programming method, in order to establish a new architecture that can improve the computing performance by exploiting the data transfer capability which is expected to increase even after the end of Moore's law. For the goal, it is important for us to find an appropriate architecture that can overcome the limitations of such conventional Neumann-type processors as many-core CPUs which is based on sequential execution of instructions and its thread-based parallelization. We believe that one of the promising ones is a new non-Neumann-type architecture where we can unfold and map algorithms onto a data-flow circuit to execute computation.

The coarse-grained reconfigurable array (CGRA) satisfies the requirements while it has been researched and developed mainly targeting embedded use cases as a substitute of a DSP. In our research, we research an appropriate architecture and a programming model of CGRA for HPC. We explore the design space of CGRAs for HPC, design and implement a compiler for CGRA, and evaluate HPC benchmarks with the compiler to find an appropriate architecture.

In this fiscal year, we have updated the design of Riken CGRA that we had made so far, and evaluated it with several benchmarks with various parameter configurations.

In this fiscal year, we modified and improved the cycle-accurate CGRA simulator developed in the previous year. The RTL description of CGRA is simulated by GHDL, and the results are recorded in a log file. The CGRA RTL description is simulated using GHDL, and the results are recorded in a log file, and the waveforms of the necessary signals can be saved in a vcd file, which can be displayed using a waveform display tool such as gtkviewer to check the RTL operation. For computation performed in CGRA simulations, input data and output data (calculation results) can be passed via files for an external memory such as DDR4 memory to be read and written for computation. The correctness of computation can be confirmed by checking the consistency with reference results given by a file.

In the CGRA simulation, we were planning to explore the design space defined by the following parameters:

1. Numerical data type and width
2. SIMD width/vector length
3. Functional heterogeneity in CGRA
4. Network topology in CGRA
5. Types of external memory

Since 1, 3, and 4 require significant modification of the RTL description, we fixed them to the following settings and conducted the exploration.

1. Numerical data types and widths: single precision floating point numbers

3. Heterogeneity of functions: load & store element (LSE) and processing element (PE)

4. On-chip network topology: directly connected network with 8 adjacent tiles

The SIMD width was changed to 1, 2, 4, etc. by setting parameters. The external memory type can be either DDR4 memory, HBM2 memory, or STT(spin transfer torque)-RAM. In addition, the length of the FIFO queue in the PE can be changed to the n-th power of 2, where n is 2 to 6. The FIFO queue is a buffer to absorb the difference in the path length of the input data in a PE. When the data flow graph (DFG) of a computation is arranged and routed in a PE array, the input path difference at each PE may become large.

In addition, since mapping DFGs of computational problems to CGRAs is a combinatorial problem that is difficult to solve manually, we have developed a tool for arranging and wiring DFGs to CGRAs. That is a DFG place-and-route tool for CGRA. When a DFG is given, the place-and-route can be performed automatically by specifying various measures such as the difference in the path length per PE and the maximum path length.

Using the CGRA simulator, we executed a one-dimensional FFT and a 3x3 convolutional arithmetic kernel (Conv 3x3) on a 9x9 PE array to verify the operation and evaluate the performance. During this process, various bugs were found and they were fixed. As a result, it was possible to run simulations for different SIMD widths, memory types, FIFO lengths, and input data sizes. We ran simulations with these different parameters and started to evaluate the number of cycles and the computational throughput. At this point, the performance characteristics of the computational throughput differ greatly depending on the type of memory used.

In the future, we plan to conduct further explorations on the above parameters after confirming the validity of the simulation results. We will also evaluate the improved architecture by considering architectural changes, especially in terms of array size, on-chip network topology, and programmable buffer tiles for data reuse in CGRA. Furthermore, we would like to modify the front-end of the CGRA compiler to enable place-and-route of the DFGs made from loop descriptions written in C or C++ to CGRA, and evaluate various benchmark programs.

3.4 Schedule and Future Plan

In addition to the researches and development conducted in FY2020, we are planning the following researches for the next fiscal year.

1. Research and development of FPGA-based system hardware for the virtual circuit switching network (VCSN) connecting FPGAs. We have been developing a system-on-chip on an FPGA, called *AFU Shell*, which support fundamental data-movement for PCI-Express interface of a host CPU, DDR4 memory interfaces on an FPGA cards, and 100Gbps high-speed serial connection for inter-FPGA networks. We will implement and evaluate the AFU Shell for VCSN of 100bps Ethernet to connect FPGAs with an indirect network topology. It provides a virtualized circuit switching mechanism on the top of the packet switching mechanism of 100Gbps Ethernet.

2. Research and development of system software for FPGA cluster. We will develop a resource manage software for FPGA resources in the cluster, which allows us to exclusively utilize a part of FPGA resources in a system with configuration of VCSN connection for the allocated FPGAs on a request from a user program.

3. Research on the FPGA cluster system "ESSPER" as a functional extension testbed of Fugaku. We will evaluate performance and capability of task off-loading from Fugaku to FPGAs, to understand subjects to be tackled for such kind of co-operation of different but combined architectures for HPC. We will also research system software to program and off-load tasks from the computing nodes of Fugaku to FPGAs in the functional extension part.

4. Research on FPGA-based applications. We are continuouly developing and evaluating applications and benchmarks for FPGA cluster such as 3D FFT, stream computing of Fluid simulation by multi-pipelining, breadth first search of a graph, Bayesian network analysis, and so on.

5. Further exploration of CGRAs, and development of its place-and-route compiler. We will extend CGRA architectures for representative HPC workloads and evaluate their performance with micro kernels. We will develop an FPGA-based overlay of our CGRA archiecture so that we can emulate it with FPGAs. We will also continuously develop a CGRA compiler, especially for its C/C++ frontend.

3.5 Publications

3.5.1 Articles/Journal

[1] Artur Podobas, Kentaro Sano, and Satoshi Matsuoka, "A Survey on Coarse-Grained Reconfigurable Architectures from a Performance Perspective," IEEE Access, Vol.8, pp.146719-146743, DOI:10.1109/ACCESS.2020.3012084, 2020.

3.5.2 Conference Papers

[2] Jens Huthmann, Artur Podobas, Lukas Sommer, Andreas Koch, and Kentaro Sano, "Profiling and Visualizing Performance of FPGAs in High-Performance Computing Environments," Proceedings of IEEE International Conference on Cluster Computing (CLUSTER), pp.371-380, DOI: 10.1109/CLUSTER49012.2020.00047, 2020.

[3] Jens Huthmann, Lukas Sommer, Artur Podobas, Andreas Koch, and Kentaro Sano, "OpenMP Device Offloading to FPGAs using the Nymble Infrastructure," Proceedings of the 16th International Workshop on OpenMP (IWOMP), Vol.12295, pp.265-279, 2020.

[4] Artur Podobas, Kentaro Sano, and Satoshi Matsuoka, "A Template-based Framework for Exploring Coarse-Grained Reconfigurable Architectures," Proceedings of the 31st IEEE International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp.1-8, DOI: 10.1109/ASAP49362.2020.00010, 2020.

[5] Norihisa Fujita, Ryohei Kobayashi, Yoshiki Yamaguchi, Tomohiro Ueno, Kentaro Sano, and Taisuke Boku, "Performance Evaluation of Pipelined Communication Combined with Computation in OpenCL Programming on FPGA," Proceedings of the Tenth International Workshop on Accelerators and Hybrid Exascale Systems (AsHES 2020), pp.450-459, 2020.

[6] Antoniette Mondigo, Tomohiro Ueno, Kentaro Sano, and Hiroyuki Takizawa, "Comparison of direct and indirect networks for high-performance FPGA clusters," Applied Reconfigurable Computing. Architectures, Tools,

and Applications (ARC 2020), Lecture Notes in Computer Science, Vol.12083, pp.314-329, DOI:10.1007/978-3-030-44534-8_24, 2020.

3.5.3 Invited Talks (keynote, plenary talk, invited talk, panelist position talk)

[7] 佐野 健太郎, ”FPGAクラスタ試作システム ESSPER とそのFPGA間通信機構,” FPGA-HPC-Symposium2021, Feb 26, 2021. (Online)

[8] Kentaro Sano, ”FPGA Cluster System for High-Performance Reconfigurable Computing,” Intel FPGA Technology Day 2020 (IFTD), Asia Pacific region and North America region, 18- Nov, 2020. (Online)

[9] 佐野 健太郎, ”リコンフィギャラブル高性能計算の実現に向けたFPGAクラスタシステム開発,” Intel FPGA Technology Day 2020 (IFTD), Japan region, 18- Nov, 2020. (Online)

[10] 佐野 健太郎, ”FPGAクラスタによるカスタム高性能計算の現状と課題,” 日本応用数理学会 2020年度年会, Sep 8-10, 2020. (Online)

[11] 佐野 健太郎, ”FPGAを用いたカスタム高性能計算システムと自動チューニングへの期待,” 第23回AT研究会オープンアカデミックセッション(ATOS23), July 14, 2020. (Online)

3.5.4 Other Publication Articles

[12] 佐野 健太郎, ”スパコンはどこまで速くできるのか？新しいアーキテクチャの可能性を追求する” 計算科学の世界20号, インタビュー記事, 2020.

[13] Roman Iakymchuk, ..., Kentaro Sano, et. al., ”Workshop on Large-scale Parallel Numerical Computing Technology (LSPANC 2020) HPC and Computer Arithmetic toward Minimal-Precision Computing,” 13 pages, https://arxiv.org/abs/2004.04628, 2020.

3.5.5 Oral Talks (with non-reviewed papers)

[14] 佐野健太郎, 上野知洋, 宮島敬明, JensHuthmann, 小柴篤史, ”ESSPER: 高性能計算のためのスケーラブルかつ柔軟なFPGAクラスタシステムの開発”, 電子情報通信学会リコンフィギャラブルシステム研究会 信学技法, Vol.120, No.339, RECONF2020-59, pp.7-12, Jan 25-26, 2021.

[15] 宮島敬明, 上野知洋, 佐野健太郎, ”高性能計算のための高速フーリエ変換のFPGA実装と評価”, 電子情報通信学会リコンフィギャラブルシステム研究会 信学技法, Vol.120, No.339, RECONF2020-61, pp.19-24, Jan 25-26, 2021.

[16] 多田 大希, 上野 知洋, 小柴 篤史, 佐野 健太郎, 河野 隆太, 井口 寧, ”FDTD法による音響シミュレーションのためのストリーム計算ハードウェアの設計と評価”, 電子情報通信学会リコンフィギャラブルシステム研究会 信学技法, Vol.120, No.339, RECONF2020-60, pp.13-18, Jan 25-26, 2021.

[17] 佐野 健太郎, ”FPGAクラスタによるカスタム高性能計算の現状と課題,” 日本応用数理学会 2020年度年会講演予稿集, 2 pages, 2020.

[18] 小柴篤史, 上野知洋, 佐野 健太郎, ”Stratix 10 FPGAクラスタにおける格子ボルツマン法のパイプライン並列化と性能評価,” 電子情報通信学会リコンフィギャラブルシステム研究会 信学技法, Vol.120, No.168, pp.7-12, Sep 10-11, 2020.

3.5.6 Patents

[19] 共同出願特許: 富士通株式会社, 理化学研究所（佐野 健太郎, 上野 知洋）, ”受信レート監視によるネットワーク輻輳制御方式”, Patent applied in Sep, 2020.

Chapter 4

Large-scale Parallel Numerical Computing Technology Research Team

4.1 Members

Toshiyuki Imamura (Team Leader)

Yiyu Tan (Research Scientist, affiliating Next Generation High Performance Architecture Research Team)

Daichi Mukunoki (Research Scientist)

Shuhei Kudo (Postdoctoral Researcher)

Takeshi Terao (Postdoctoral Researcher)

Takuya Ina (Technical Staff)

Tetsuya Sakurai (Senior Visiting Researcher, University of Tsukuba)

Daisuke Takahashi (Senior Visiting Researcher, University of Tsukuba)

Franz Franchetti (Senior Visiting Researcher, Carnegie Mellon University)

Mitsuo Yokokawa (Seniort Visiting Researcher, Kobe University)

Sarah Huber (Visiting Researcher, Bergische Universtät Wuppertal)

Martin Galgon (Visiting Researcher, Bergische Universtät Wuppertal)

Ryuki Shimotori (Student Trainee, University of Yamanashi)

Tomoya Shimizu (Student Trainee, Kobe University)

Hotaka Yagi (Student Trainee, Tokyo University of Science)

Tatsuya Mitsuda (Intern, University of Kyushu)

Sameer Deshmukh (Intern, Tokyo Institute of Technology)

Thomas Spendlhofer (Intern, Tokyo Institute of Technology)

Aya Motohashi (Assistant)

4.2 Overview of Research Activities

The Large-scale Parallel Numerical Computing Technology Research Team conducts research and development of numerical software for the national flagship systems, K computer and the supercompter Fugaku. In particular, we are focusing on significant technical issues when we face at extensive computing, such as, large-scale, highly parallel and high-performance. In general, simulation programs require various numerical techniques to solve systems of linear equations, to solve eigenvalue problems, to compute and solve non-linear equations, and to do fast Fourier transforms. From a mission critical point of view, it is natural for us to develop and deploy highly-parallelized and scalable numerical software integrated over a software framework dedicated on Fugaku and other HPCI systems as well, but not limited on a target platform as one of the advantage of Fugaku is known as ARM eco-system. It comprises above-mentioned software components (numerical software) in order to run a specific simulation code which come from scientific and engineering domain problems. Beside, the Fugaku-related issues have been supposed to be our challenging works even though unexplored or conventional such as communication reducing and avoidance, cooperation with advanced devices, fault detection and recovery, and accuracy-hard problems.

We mainly focused on the critical issues on last paragraph, which were initiated when the team established. Since 2016, we have added some extensional themes to our research directions as recipes for long-term research goals;

1. investigation from conventional to unexplored numerical fields,
2. precision-aware computing and numerical reproducibility, and
3. utilization of emerging devices or information processing circuits.

The most of our research areas in FY2019-2020 are not clearly distinguishable into the above categories, but the research themes conducted during FY2020-2021 belong to one or more of them. In particular, outstanding achievements in our research include several studies to verify the arithmetic potential of low-precision arithmetic such as an FP16 format in Fugaku, and significant achievements in content related to arithmetic systems, such as techniques to emulate higher-precision arithmetic by utilizing low-precision or normal arithmetic.

4.3 Research Results and Achievements

4.3.1 Parallel 3D-FFT collaborated with DNS Fugaku project

In the super-ultra scale DNS simulation conducted by Yokokawa-Ishihara-Kanada et al., it is essential to perform a fast 3D-FFT on the grid scale of $24,576^3$ or larger using the supercomputer Fugaku. We have been conducting a preliminary study using the K computer as a pilot platform towards Fugaku, and a preliminary study on pipelined FFT for multiple physical variables has been investigated since 2017 collaborated with Porf. Yokokawa and Mr. Aoki, one of our former Student trainees. As a result of the research, besides the regular physical field variables used in the DNS code, there are additional variables for alias processing; meanwhile, the well-balanced system configuration on the K computer yielded an excellent performance of the pipelined FFT on the K computer (ParCo2019 Imamura et al.). Thus, pipeline processing has been recognized as a promising approach on the K computer and comparable systems, hopefully Fugaku. On the other hand, for effective implementation of pipelining, it is indispensable that the execution time of each asynchronously executed task is nearly or almost uniform. Supposed that the cost model of the 3D-parallel FFT is defined as $T_{\mathrm{3DFFT}} = T_{\mathrm{FFT(x)}} + C_{\mathrm{xtoy}} + T_{\mathrm{FFT(y)}} + C_{\mathrm{ytoz}} + T_{\mathrm{FFT(z)}}$. The computational cost of the stationary state of the pipelined 3D-FFT is roughly obtained as $T_{\mathrm{3DFFT\_steady}} = 3\max(C_{\mathrm{atob}}, T_{\mathrm{FFT(a')}})$. In the case of the K computer, the time required for FFT ($T_{\mathrm{FFT(a')}}$) and the data transfer time (C_{atob}) were almost equivalent or comparable to overlap each other. When updating from the K computer to the Fugaku computer, the processor performance was improved about 30 times from 128 GFLOPS to 3.4 TFLOPS, and the network performance (injection bandwidth per node) was improved only 2 times from 20 GB/s to 40.8 GB/s. The transfer cost C becomes dominant relatively 15 fold. In another viewpoint, potentially 93% of the processors are expected to be in an idle state on Fugaku.

We have kicked off a new practical study to reduce communication overhead for FFT on Fugaku. We have conducted a preliminary three-way investigation; 1) the performance of FFT kernels on various platforms, 2) performance and topology correlation with a multi-dimensional configuration of the Tofu-D interconnect, and 3) a rank map reconstruction or a space-filling curve mapping proposed by Kuroda et al. (High-Performance Computing Symposium, HPCS2013). These are still ongoing research topics, and the results cannot be mentioned in this report, but the performance of the FFT kernel test on an NEC's vector processor and the correlation with I/O have been published externally as one of the preliminary experiments Fugaku (IPSJ SIGHPC report[11]). As the nature of the butterfly operations in FFT, the global data exchange is essential, and communication is unavoidable, determining the actual cost of parallel FFT. From the HPC point of view, it is desirable to use approximation techniques such as asynchronous optimization for latency reduction and compression of data transmission.

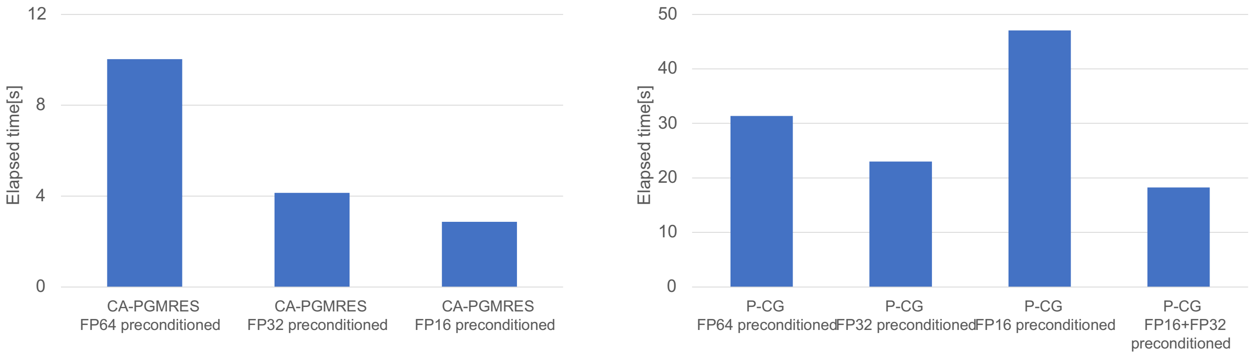


Figure 4.1: The different convergence properties in the computing time between CA-PGMRES on a plasma simulation (left) and P-CG on a multi-phase flow simulation (right) according to floating-point formats

4.3.2 Mixed-precision acceleration of preconditioned CG method for an ill-conditioned CFD problem

The significant part of the multi-phase flow simulation is to solve a vast ill-conditioned matrix, which comes from the inhomogeneous density distribution of the target fluid system having quite an extensive dynamic range beyond 10^4. Thus, the Krylov subspace method dominates more than 90% of the total computational time. A leading method for optimizing the Krylov subspace method is known to be advanced and fast preprocessing, and we attempted to take advantage of lower-precision arithmetic for preprocessing of the communication-avoiding GMRES (CA-GMRES) method in plasma turbulence calculations as one of the acceleration methods, on which we have been collaborating for a couple of years with Japan Atomic Energy Agency, JAEA (refer to our outreach histories, for examples, Idomura et al. at ScalA17, Scala18, and Ali et al. at ScalA19).

The summary of the FY2020-2021 collaboration contains two key points, and we published them as our latest results at the SC20 technical session [5]. First, we have introduced a preconditioner based on FP16 ('float16' so-called half precision) to Krylov subspace methods, reducing the memory transactions. By appropriately normalizing matrices and vectors to be representable in an FP16 format with a narrow dynamic range, memory access is reduced, and the speedup is achieved on Fugaku, a general-purpose CPU environment, and Summit, a GPU environment. At the same time, we confirmed that lower-precision arithmetic incurs poor accuracy, and it degrades convergence in practice. Figure 4.1 illustrates the fact that is quite different from our previous experiment in plasma turbulent simulations, in which the degradation of convergence is not a significant concern because solving a suitable condition matrix converges without preprocessing. We can observe that the behavior differs in CA+PGMRES+FP16 shown in the left and PCG+FP16 shown in the right.

Thus, secondly, to overcome the accuracy and numerical instability issues, we have developed a mixed-precision preconditioning method that stores data in the FP16 format in the main memory, then internally performs the FP32 operations, and eventually returns the results to the main memory after the truncation to FP16. As a result, we have achieved high speed and sustained better convergence by reducing the number of memory accesses. As the rightmost bar in Figure 4.1 shows, we have successfully achieved the desired numerical convergency and performance with FP16+FP32 mixed-precision calculations.

4.3.3 Analysis and development of the HPL-AI benchmark code

HPL-AI is a new benchmark program initiated in November 2019, allowing to incorporate AI-envisioning accelerating units and lower-precision arithmetic such as FP32, FP16, and so on. As a naming of the benchmark, it inherits the fair numerical benchmark concept from HPL and has introduced care of the gap between lower-precision algorithms and the conventional ones concerning the accuracy of the output result. Thus, the main components of the HPL-AI benchmark are the LU factorization with lower-precision arithmetic and the Iterative Refinement part with respect to the FP64 accuracy comparably equivalent to the original HPL benchmark.

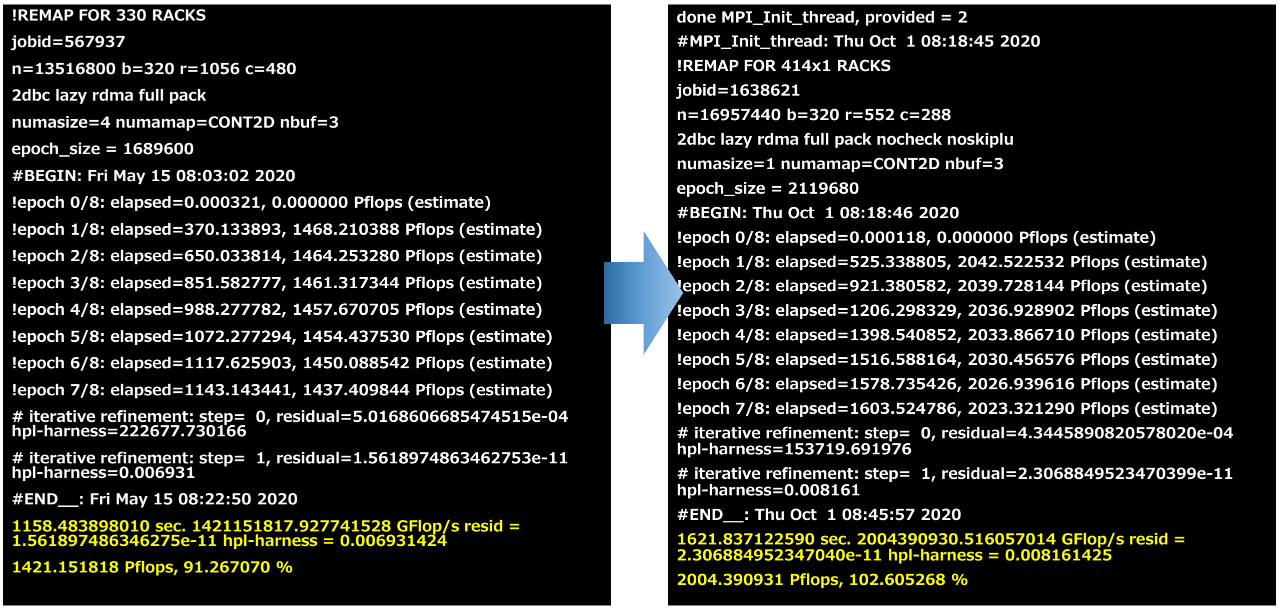
The primary difficulties in low-precision arithmetic are the removal of numerical overflows and the underflow and the continuation of stable and correct calculations within the weak representation ability of the data format. In the last year, we have analyzed the HPL-AI matrix itself and investigated the stability of the LU factorization and the specialization of the IR method from the viewpoint of numerical properties. Moreover, we have been developing the code using the system from the Fugaku pilot system to the Fugaku system with limited

```
subroutine MPLU(n,b,A)
integer, intent(IN)  :: n              ! matrix size
integer, intent(IN)  :: b              ! block size
real(4), intent(OUT) :: A(b,b,n/b,n/b) ! result in FP32
integer :: i, j, k, nb
real(4) :: s, is
real(2) :: B(b,b,n/b,n/b), L(b,b,n/b) U(b,b,n/b)
nb = n / b
B(:,:,1:nb,1:nb) = 0
s = sqrt(n*sqrt(n))
is = 1. / s
do k=1,nb
   ! static scaling and lazy initialization
   A(:,:,k:nb,k) = MG(b,k,nb,k,k+1) - is*real(B(:,:,k:nb,k),4)
   A(:,:,k,k+1:nb) = MG(b,k,k+1,k+1,nb) - is*real(B(:,:,k,k+1:nb),4)
   A(:,:,k,k) = LU(A(:,:,k,k))
   if(k.eq.nb) exit
   A(:,:,k+1:nb,k) = TRSM("R","U","N","N",A(:,:,k,k),A(:,:,k+1:nb,k))
   A(:,:,k,k+1:nb) = TRSM("L","L","N","U",A(:,:,k,k),A(:,:,k,k+1:nb))
   L(:,:,k+1:nb) = real(s*A(:,:,k+1:nb,k),2)
   U(:,:,k+1:nb) = real(A(:,:,k,k+1:nb),2)
   do j=k+1:nb
      do i=k+1:nb
         B(:,:,i,j) = B(:,:,i,j) + L(:,:,i)*U(:,:,j)
      end do
   end do
end do
end subroutine
```

Figure 4.2: A pseudo-program of the mixed-precision LU factorization (`MPLU`) with lazy initialization, static scaling and on-the-fly matrix generation techniques. `real(x,r)` refers to the explicit data conversion of `x` to the `r`-byte real format, `MG` generates part of the input matrix, `TRSM` corresponds to the BLAS's xTRSM, and `LU` factorizes a small part of the matrix.

access in the early installation period. In our code development, several techniques are employed such as three numerical precision types simultaneously (FP64(**double**: double precision) - FP32 (**float**: single precision) - FP16 (**float16**: half precision)), called mixed-precision computing as presented in Figure 4.2. Specifically, seven proposed technique underlie the success of 2 EFlop/s benchmark, namely, single iteration iterative refinement scheme (SIIR), lazy initialization, static-scaling, on-the-fly matrix generation presented in Figure 4.2. Besides, double-decker layout, packed data-layout, and hardware-offloaded collective communication are significant for the parallel implementation (reported in [3], [18], [20], [21], [27] and [30]).

The first phase of the code development was completed in May 2020, and we were awarded first place in the top500 BOF session at ISC20 with 1.4 EFLOPS by running on a 5/6 Fugaku system in normal mode (2.0GHz). The record overtook the Summit system, which recorded the first world record in November 2019. Following ISC20, we successfully extended the performance record to 2.0 EFLOPS in SC21 by running the whole Fugaku system in boost mode (2.2GHz), see Figure 4.3. In addition, the energy consumption during this benchmark showed an similar behavior comparable to the regular HPL benchmark on the Fugaku, which spent approximately 29–30 MW [21]. Other major benchmark records of four crown achievements have been also reported in [6]. The same code was also used on the Flow system at Nagoya University, which also recorded an excellent performance (over 90% of the peak) and was awarded 5th place at that time (`https://hpl-ai.org/doc/results`). Although the benchmarking was started less than a year ago and the number of participating systems is still small, the advantages of our code are incredibly high. However, our advantage is limited on the A64FX architecture and necessary to extend other major platforms, where another 16bit format (**bfloat16**) is available.



```
!REMAP FOR 330 RACKS
jobid=567937
n=13516800 b=320 r=1056 c=480
2dbc lazy rdma full pack
numasize=4 numamap=CONT2D nbuf=3
epoch_size = 1689600
#BEGIN: Fri May 15 08:03:02 2020
!epoch 0/8: elapsed=0.000321, 0.000000 Pflops (estimate)
!epoch 1/8: elapsed=370.133893, 1468.210388 Pflops (estimate)
!epoch 2/8: elapsed=650.033814, 1464.253280 Pflops (estimate)
!epoch 3/8: elapsed=851.582777, 1461.317344 Pflops (estimate)
!epoch 4/8: elapsed=988.277782, 1457.670705 Pflops (estimate)
!epoch 5/8: elapsed=1072.277294, 1454.437530 Pflops (estimate)
!epoch 6/8: elapsed=1117.625903, 1450.088542 Pflops (estimate)
!epoch 7/8: elapsed=1143.143441, 1437.409844 Pflops (estimate)
# iterative refinement: step=  0, residual=5.0168606685474515e-04
hpl-harness=222677.730166
# iterative refinement: step=  1, residual=1.5618974863462753e-11
hpl-harness=0.006931
#END__: Fri May 15 08:22:50 2020
1158.483898010 sec. 1421151817.927741528 GFlop/s resid =
1.561897486346275e-11 hpl-harness = 0.006931424
1421.151818 Pflops, 91.267070 %
```

```
done MPI_Init_thread, provided = 2
#MPI_Init_thread: Thu Oct  1 08:18:45 2020
!REMAP FOR 414x1 RACKS
jobid=1638621
n=16957440 b=320 r=552 c=288
2dbc lazy rdma full pack nocheck noskiplu
numasize=1 numamap=CONT2D nbuf=3
epoch_size = 2119680
#BEGIN: Thu Oct  1 08:18:46 2020
!epoch 0/8: elapsed=0.000118, 0.000000 Pflops (estimate)
!epoch 1/8: elapsed=525.338805, 2042.522532 Pflops (estimate)
!epoch 2/8: elapsed=921.380582, 2039.728144 Pflops (estimate)
!epoch 3/8: elapsed=1206.298329, 2036.928902 Pflops (estimate)
!epoch 4/8: elapsed=1398.540852, 2033.866710 Pflops (estimate)
!epoch 5/8: elapsed=1516.588164, 2030.456576 Pflops (estimate)
!epoch 6/8: elapsed=1578.735426, 2026.939616 Pflops (estimate)
!epoch 7/8: elapsed=1603.524786, 2023.321290 Pflops (estimate)
# iterative refinement: step=  0, residual=4.3445890820578020e-04
hpl-harness=153719.691976
# iterative refinement: step=  1, residual=2.3068849523470399e-11
hpl-harness=0.008161
#END__: Thu Oct  1 08:45:57 2020
1621.837122590 sec. 2004390930.516057014 GFlop/s resid =
2.306884952347040e-11 hpl-harness = 0.008161425
2004.390931 Pflops, 102.605268 %
```

Figure 4.3: Output logs of the HPL-AI benchmark in June 2020 (left) and November 2020 (right)

4.3.4 Iterative refinement approach towards the next generation Eigenvalue solver

Cuppen's divide-and-conquer method is one of the fast and practical approaches for the eigenvalue problem for symmetric tridiagonal matrices. We focused on the iterative refinement method (Ogita, and Aishima, JJIAM 2018) to improve eigenvalue problems and proposed a mixed-precision algorithm [28]. Our basic idea of the mixed-precision is that; First, by using low-precision arithmetic or low-accuracy calculations, we compute all eigenpairs of a rank-one perturbation problem, $D + \rho z z^\top$. After that, we apply the iterative refinement method to improve the accuracy ($\|AX - X\Lambda\|$) and orthonormality ($\|X^T X - I\|$) with the extension of regular/higher precision arithmetic. Here, it is known that the calculation of the secular equation becomes a bottleneck on the eigenvalue solver with the DC method. Although the iterative refinement is not a cheaper calculation than the calculation of the secular equation, we expect a good balance between both of them following the sparsity of the rank-one matrix, and they can omit. In addition, iterative refinement has stronger scalability since the main cost is matrix multiplications potentially acceleratable up to the theoretical peak performance.

The DC method can be extend to the solver for a band matrix as we implemented on our EigenExa solver [33]. Here, the DC method with a band matrix can be pointed out in its computational cost and accuracy as well. For the first issue of computational speed, the proposed method calculates eigenpairs using the low-precision DC method. In addition, the iterative refinement can specify user's demanding accuracy, which depends on the number of iterations. From both perspectives, the algorithm proposing has a trade-off property. However, from other incredible achievements of the iterative refinement approach, the approach to generate an initial set using low-precision arithmetic in a short time and to improve iteratively is crucial in the future. We must further improve the method from the HPC viewpoint and develop a standard programming framework for distributed parallel systems towards the achievement of higher usability on the supercomputer Fugaku.

4.3.5 Accurate and reproducible numerical linear algebra

Floating-point operations with finite precision cause two issues due to rounding errors: loss of accuracy and loss of reproducibility. For accuracy, 64-bit floating-point format (FP64) is commonly used in numerical computations, but in recent years, 32- and 16-bit formats are also used for further speedup and energy efficiency. On the other hand, 128-bit format (FP128) was defined in IEEE standard in 2008 for computations that require high accuracy. Reproducibility refers to the ability to obtain the bit-wise identical result when computing on the same input multiple times. Since floating-point operations are non-associative, when the degree of parallelism is changed in a parallel computation, the computation result can change at the level of rounding error due to a change in the computation order. This can be a barrier to quality assurance and debugging of programs. The issues of accuracy and reproducibility may become more serious as rounding errors accumulate in large-scale calculations, and methods and software have been developed to address these problems.

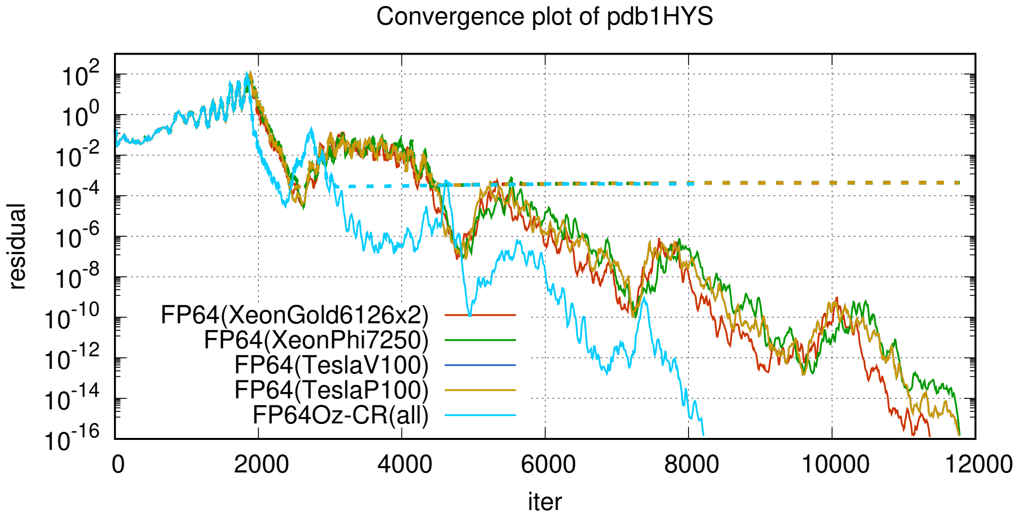


Figure 4.4: Convergence plot (at every 10 iterations) of matrix "pdb1HYS" from SuiteSparse Matrix Collection. The results of FP64Oz-CR on four platforms are shown with one line as they are identical. Solid lines show the relative residual $||\boldsymbol{r}_i||/||\boldsymbol{b}||$ and dotted lines show the relative true residual $||\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_i||/||\boldsymbol{b}||$.

Our team has so far developed a BLAS implementation that can ensure bit-wise reproducibility named OzBLAS. It utilizes the Ozaki scheme, which is an accurate matrix multiplication algorithm based on the error-free transformation of matrix multiplication. In this fiscal year, we extended the implementation of OzBLAS to the Conjugate Gradient (CG) method and developed accurate and reproducible CG solvers on CPUs and GPUs [10]. Figure 4.4 shows an example of the evaluation conducted in this study. The standard FP64 solver ("FP64") shows different convergence histories on four different platforms, while our implementation ("FP64Oz-CR") shows an identical convergence history and converges with fewer iterations thanks to its accurate operations. However, the effect of accurate operations depends on the problem. The cost of performing accurate and reproducible operations also depends on the problem, but in our evaluation using 8 matrices from SuiteSparse Matrix Collection, we observed overheads of 5.5 to 37 times compared to the standard implementations of FP64 using the vendor's BLAS on CPUs and GPUs.

As another extension of OzBLAS, in the last fiscal year we proposed the mixed-precision Ozaki scheme to compute the matrix multiplication of FP64 (DGEMM) using Tensor Cores' FP16/FP32 mixed-precision operations on NVIDIA GPUs. This fiscal year, we presented the result at an international conference, and the paper was published [2] (see the last fiscal year's Annual Report for details). In this fiscal year, as an extension of this work, we proposed a method to compute the matrix multiplication of FP128 using DGEMM [12]. FP128 is not implemented in hardware on most processors including x86, but the software emulation is available on some compilers such as GCC. Our method partially uses the FP128 emulation, but as most of the computation can be performed with DGEMM, it can achieve much faster performance than that of matrix multiplication performed using only the FP128 emulation. This study will be published in an international conference paper next fiscal year.

4.3.6 FPGA-based matrix multiplier with advanced parallelization

Matrix multiplication requires computer systems have huge computing capability and data throughput as problem size is increased. In this research, a matrix multiplier with systolic architecture is designed using OpenCL programming language and implemented using the FPGA board DE10-Pro, which contains a latest Intel FPGA Stratix 10 and 8 GB DDR memory. Table 4.1 shows the hardware utilization of the FPGA-based matrix multiplier in the case of different problem sizes and block sizes. Data are single-precision and data vectorization is 8. The systolic array and data vectorization determines the utilization of DSP blocks while the block size of matrices A and B affects the consumed block RAMs. When the dimension of systolic array is increased, more DSP blocks are required. And the more RAM blocks are utilized as the block size of matrices A and B is increased. Table 4.2 presents the computational throughput and clock frequency of the FPGA-based matrix multiplier. As the dimension of the systolic array is increased, although more DSP blocks involve in computation, the clock

Table 4.1: Hardware utilization

Matrix scale		Systolic array	Block size		Hardware utilization		
A	B		A	B	Logic utilization	DSP blocks	RAM blocks
4096×4096	4096×3584	32×14	1024×128	128×448	364317 (39%)	3584(62%)	2798(24%)
8192×8192	8192×7168	32×14	1024×128	128×448	364317(39%)	3584(62%)	2798(24%)
8192×8192	8192×7168	32×14	1024×256	256×448	361590(39%)	3584(62%)	3394(29%)
8192×8192	8192×7168	32×14	1024×512	512×448	372298(40%)	3584(62%)	4617(39%)
4096×4096	4096×4096	32×16	1024×256	256×512	397647(43%)	4096(71%)	3730(32%)
8192×8192	8192×8192	32×16	1024×256	256×512	397647(43%)	4096(71%)	3730(32%)
4096×4096	4096×4608	32×18	1024×256	256×576	463017(50%)	4608(80%)	4036(34%)

Table 4.2: Computation throughput (TFLOPs)

Matrix scale		Systolic array	Block size		Clock frequency (MHz)	Computation throughput
A	B		A	B		
4096×4096	4096×3584	32×14	1024×128	128×448	420	2.42
8192×8192	8192×7168	32×14	1024×128	128×448	420	2.49
8192×8192	8192×7168	32×14	1024×256	256×448	415	2.42
8192×8192	8192×7168	32×14	1024×512	512×448	419	2.37
4096×4096	4096×4096	32×16	1024×256	256×512	342	2.32
8192×8192	8192×8192	32×16	1024×256	256×512	342	2.38
4096×4096	4096×4608	32×18	1024×256	256×576	345	2.67

frequency of the system drops significantly. When the systolic array is 32×18, the system achieves the highest computational throughput, which is about 2.67 TFLOPs.

4.3.7 Other initiative works and activities

The following studies were also conducted in FY2019-2020, and we published papers and reports:

- Some review or summary talks for numerical libraries on Fugau or other modern systems ([1], [9], [16], [17], [22], [24]),
- Collaboration with large-scale parallel application users ([4]),
- Joint work on the DL4Fugaku project ([21], and MLPerf benchmark report at SC2020).
- Acceleration study by using FPGA devices ([19], [23]),
- Numerical verification, validation, and accurate methods for numerical linear algebra ([7], [8], [25]), and
- A strongly related work to the publish of the white paper towards a post-Fugaku computer system on the NGACI (Next-Generation Advanced Computing Infrastructure) project, `https://sites.google.com/view/ngaci/home`.

4.4 Schedule and Future Plan

The most significant task between FY2019 and FY2020 was to complete the development of the supercomputer Fugaku, and we recognized that almost all tasks were successfully carried out. As reported previously in the annual reports in FY2018-2019 and FY2019-2020, we have updated the research milestone on the short-term and long-term. We recognize that the must-do-topics are CA and precision issues, and their significance and our approaches should be re-mentioned again here as for our gentle reminder as every year we do in the annual reports.

- **Communication avoiding algorithm:**

  We continue to investigate the communication-avoiding algorithms and methods in any view of numerical linear algebra. The technologies will be applied to the algorithms of the existing CA-XX linear solvers, both dense and sparse eigenvalue solvers, multi-dimensional FFT routines, and so on. Furthermore, it is not just a simple matter of reducing the number and amount of communications, but it is also necessary to address the insufficient capacity of wideband memory and the ability to cover the transfer between upper memory hierarchies through analytical compression techniques or a newly-introduced representation method for simple vector and tensor formats to be represented.

- **Precision-and-power aware computing:**

  We have studied high/mixed/reduced-precision numerical software. Recently, reduced-precision computing enforced by deep learning becomes a significant driving force on the industry/consumer's market and the HPC community. What is more, another critical issue that came from power-saving was pointed out in the annular report last year. We will corporate with the stochastic and approximate approach and arbitrary precision arithmetic with the help of a reconfigurable device to reduce the cost of floating-point operations and the total volume of the required hardware and energy consumption.

In addition to the issues mentioned above, various international endeavors have been identified with the completion of the Fugaku project. For example, it is urgent to validate the numerical software developed in different parts of the world and evaluate the standard software's performance as the responsibility of the world's leading-edge computer institutions. In particular, the software products developed under the ECP initiatives in the US, SLATE, PETSc, kokkos, and FFT-X (SPIRAL) have be actively deployed on Fugaku within the framework of MoU between DOE-MEXT, and through joint research agreements between CMU and R-CCS, respectively.

Furthermore, we also need to promote the long-term-ranged fundamental research on numerical algorithms that can be employed on practical quantum devices, circuits, or realistic quantum computers, which will be available for ten or more years in the field of quantum computer science. In particular, it is thought to be one of the missions to explore the possibility of so-called difficult class numerical computation, such as complicated constrained optimization with real-time restriction, on post-silicon generation computers by making full use of the emulator on Fugaku and the quantum computer programming environment developed by related organizations.

4.5 Publications

4.5.1 Articles/Journal/Conference Papers

[1] Roman Iakymchuk, Daichi Mukunoki, Artur Podobas, Fabienne Jézéquel, Toshiyuki Imamura, Norihisa Fujita, Jens Huthmann, Shuhei Kudo, Yiyu Tan, Jens Domke, Kai Torben Ohlhus, Takeshi Fukaya, Takeo Hoshi, Yuki Murakami, Maho Nakata, Takeshi Ogita, Kentaro Sano, and Taisuke Boku: "While Paper from Workshop on Large-scale Parallel Numerical Computing Technology (LSPANC 2020): HPC and Computer Arithmetic toward Minimal-Precision Computing," arXiv:2004.04628, hal-02536316, https://hal.archives-ouvertes.fr/hal-02536316, Apr. 2020.

[2] Daichi Mukunoki, Katsuhisa Ozaki, Takeshi Ogita, and Toshiyuki Imamura: "DGEMM using Tensor Cores, and Its Accurate and Reproducible Versions," ISC High Performance 2020, Lecture Notes in Computer Science, Vol. 12151, pp. 230-248, Jun. 2020, DOI: https://doi.org/10.1007/978-3-030-50743-5_12.

[3] Shuhei Kudo, Keigo Nitadori, Takuya Ina, and Toshiyuki Imamura: "Implementation and Numerical techniques for One Eflop/s HPL-AI benchmark on Fugaku," 2020 IEEE/ACM 11th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA), 2020, pp. 69–76, DOI: https://doi.org/10.1109/ScalA51936.2020.00014.

[4] Hisashi Yashiro, Koji Terasaki, Yuta Kawai, Shuhei Kudo, Takemasa Miyoshi, Toshiyuki Imamura, Kazuo Minami, Hikaru Inoue, Tatsuo Nishiki, Takayuki Saji, Masaki Satoh, and Hirofumi Tomita: "A 1024-member ensemble data assimilation with 3.5-km mesh global weather simulations," SC '20: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Article No. 1, pp. 1–10, November 2020, DOI: https://doi.org/10.1109/SC41405.2020.00005, (Gordon Bell Finalist paper).

[5] Yasuhiro Idomura, Takuya Ina, Yussuf Ali, and Toshiyuki Imamura: "Acceleration of fusion plasma turbulence simulations using mixed-precision communication-avoiding Krylov method," SC '20: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, Article No. 93, pp. 1–13, November 2020, DOI: https://doi.org/10.1109/SC41405.2020.00097.

[6] 石川 裕, 佐藤 三久, 今村 俊幸, 児玉 祐悦, 工藤 周平, 似鳥 啓吾, 伊奈 拓也, 中尾 昌広, 上野 晃司, 藤澤 克樹, 清水 俊幸, 三吉 郁夫, 三輪 英樹, 細井 聡: 「スーパーコンピュータ「富岳」4冠達成」, "Feat of Winning

Four Major Benchmarks on Supercomputer Fugaku,” 電子情報通信学会誌, Vol. 103, No. 12, pp. 1217–1220, 2020-12-01, https://www.journal.ieice.org/summary.php?id=k103_12_1217&year=2020&lang=J.

[7] 原山 越幸, 工藤 周平, 椋木 大地, 今村 俊幸, 高橋 大介: 「オーバー・アンダーフローを抑えた高精度かつ高速な2ノルム計算手法」, 情報処理学会研究報告: ハイパフォーマンスコンピューティング(HPC), Vol. 2020-HPC-177, No. 8, pp. 1–9, 2020-12-14, Permalink: http://id.nii.ac.jp/1001/00208770/.

[8] 寺尾 剛史, 尾崎 克久, 荻田 武史, 今村 俊幸: 「最近点丸めのみを用いた実対称行列に対する標準固有値問題の精度保証法」, 情報処理学会研究報告: ハイパフォーマンスコンピューティング(HPC), Vol. 2020-HPC-177, No. 16, pp. 1–8, 2020-12-14, Permalink : http://id.nii.ac.jp/1001/00208778/.

[9] Fabienne Jézéquel, Stef Graillat, Daichi Mukunoki, Toshiyuki Imamura, and Roman Iakymchuk: “Can we avoid rounding-error estimation in HPC codes and still get trustful results?” Proc. 13th International Workshop on Numerical Software Verification 2020 (NSV 20), Lecture Notes in Computer Science, Vol. 12549, pp. 163-177, Dec. 2020, DOI: https://doi.org/10.1007/978-3-030-63618-0_10.

[10] Daichi Mukunoki, Katsuhisa Ozaki, Takeshi Ogita, and Roman Iakymchuk: “Conjugate Gradient Solvers with High Accuracy and Bit-wise Reproducibility between CPU and GPU using Ozaki scheme,” Proc. The International Conference on High Performance Computing in Asia-Pacific Region (HPCAsia 2021), pp. 100-109, 2021, DOI: https://doi.org/10.1145/3432261.3432270.

[11] 武中 裕次郎, 横川 三津夫, 石原 卓, 小松 一彦, 小林 広明, 今村 俊幸, 清水 智也: 「非圧縮性乱流DNSコードに現れる高速フーリエ変換のSX-Aurora TSUBASAにおける性能評価」, 情報処理学会研究報告: ハイパフォーマンスコンピューティング(HPC), Vol. 2021-HPC-178, No. 21, pp. 1–9, 2021-03-08, Permalink: http://id.nii.ac.jp/1001/00209917/.

4.5.2 Oral talks and Poster presentations

[12] 椋木大地, 尾崎克久, 荻田武史: 「尾崎スキームを用いたbinary128による4倍精度行列積」, 日本応用数理学会2020年度年会講演予稿集, Sep. 10, 2020 (in Japanese).

[13] 工藤 周平, 似鳥 啓吾, 今村 俊幸, 伊奈 拓也: 「HPL-AI行列の観察とベンチマークプログラムの実装について」, 日本応用数理学会 2020年度年会

[14] 椋木大地, 尾崎克久, 荻田武史: binary128 に対する尾崎スキーム行列積, 第4回精度保証付き数値計算の実問題への応用研究集会 (NVR 2020), online, Nov. 28-28, 2020 (in Japanese).

[15] Daichi Mukunoki, DGEMM using Tensor Cores and OzBLAS: BOS session 11th Joint Laboratory for Extreme Scale Computing (JLESC) Workshop, online, Sep. 8, 2020.

[16] Imamura Toshiyuki, Minimal-Precision Computing: BOS session at 1th Joint Laboratory for Extreme Scale Computing (JLESC) Workshop, online, Sep. 8, 2020.

[17] 今村 俊幸: 「「富岳」がもたらすエクサスケール数値計算環境、現状と課題」, 日本応用数理学会 第12回 三部会連携「応用数理セミナー」, 2020年12月23日

[18] Shuhei Kudo, Keigo Nitadori, Takuya Ina, Toshiyuki Imamura: Prompt report on Exa-scale HPL-AI benchmark, IEEE Cluster 2020, Kobe (poster)

[19] Yiyu Tan, Masaaki Kondo, Toshiyuki Imamura: An FPGA-based Sound Field Rendering System, IEEE Cluster 2020, Kobe (poster).

[20] Toshiyui Imamura: HPL-AI benchmark on Fugaku, The 3rd R-CCS International Symposium, February 2021.

[21] Akiyoshi Kuroda, Toshiyuki Imamura, Ikuo Miyoshi, Kazuo Minami and Satoshi Matsuoka: Evaluation of Power Consumption of MLPerf HPC by Comparison HPL-AI on the Supercomputer Fugaku, The 3rd R-CCS International Symposium, Feb. 15, 2021 (poster).

[22] Toshiyuki Imamura, Yusuke Hirota, Takuya Ina, Shuhei Kudo, Takeshi Terao, Katsuhisa Ozaki and Takeshi Ogita: EigenExa and related eigensolver projects – high performance to numerical verification, The 3rd R-CCS International Symposium, Feb. 15, 2021 (poster).

[23] Yiyu Tan, Toshiyuki Imamura and Masaaki Kondo: Design and Implementation of FPGA-based High-order FDTD Method for Room Acoustics, The 3rd R-CCS International Symposium, Feb. 15, 2021 (poster).

[24] Daichi Mukunoki, Katsuhisa Ozaki, Takeshi Ogita, Toshiyuki Imamura, Roman Iakymchuk: High-Precision, Accurate, and Reproducible Linear Algebra Operations using Ozaki Scheme, The 3rd R-CCS International Symposium, Feb. 15, 2021 (poster).

[25] Tekeshi Terao: Verification Method for Eigenvalue Problems without Directed Rounding, 12th JLESC workshop, Feburary 24-26, 2021, http://icl.utk.edu/jlesc12/

[26] Daichi Mukunoki, Katsuhisa Ozaki, Takeshi Ogita, Toshiyuki Imamura: DGEMM using Tensor Cores, SIAM Conference on Computational Science and Engineering (CSE21), online, Mar. 4, 2021.

[27] Toshiyuki Imamura, Yusuke Hirota, Takuya Ina: Highly-Scalable Parallel Eigensolver on Emerging Exascale Systems, SIAM Conference on Computational Science and Engineering (CSE21), online, Mar. 1, 2021.
[28] Takeshi Terao, Toshiyuki Imamura: A Mixed Precision Divide-and-Conquer Method for Eigenvalue Problems, SIAM Conference on Computational Science and Engineering (CSE21), online, Mar. 3, 2021.
[29] Toshiyuki Imamura: Development of EigenExa from K to Fugaku, and beyond Fugaku, The 4th meeting for application code tuning on A64FX computer systems, March 17, 2021, https://www.hpci-office.jp/pages/e_meeting_A64FX_210317
[30] Toshiyuki Imamura: HPL-AI Benchmark on Fugaku, 2021 Conference on Advanced Topics and Auto Tuning in High-Performance Scientific Computing, March 19-20, 2021, https://sites.google.com/site/atathpsc/

4.5.3 Award

[31] Ranked #1 in the HPL-AI benchmark, top500 in June 2020.
[32] Ranked #1 in the HPL-AI benchmark, top500 in November 2020.

4.5.4 Software (released as of April 2021)

[33] EigenExa, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/eigenexa/.
[34] KMATH_EIGEN_GEV, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/kmath-eigen-gev/.
[35] KMATH_RANDOM, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/kmath-random/.
[36] KMATHLIB_API, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/kmathlib-api.
[37] KMATH_FFT3D, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/kmath-fft3d/.
[38] ASPEN.K2, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/aspen-k2/.
[39] MUBLAS-GEMV, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/mublas/.
[40] Batched BLAS, https://www.r-ccs.riken.jp/labs/lpnctrt/projects/batchedblas/.

Chapter 5

Field Theory Research Team

5.1 Members

Yasumichi Aoki (Team Leader)

Issaku Kanamori (Research Scientist)

Yoshifumi Nakamura (Research Scientist)

Sinya Aoki (Senior Visiting Scientist)

Shoji Hashimoto (Senior Visiting Scientist)

C. J. David Lin (Visiting Scientist)

Atsuki Hiraguchi (Intern)

Yumeno Kusuhara (Assistant)

– close collaborators in R-CCS –

Keigo Nitadori (Technical Scientist at Operations and Computer Technologies Division)

5.2 Overview of Research Activities

Field theory research team performs researches related with the numerical computation of quantum field theory (QFT) in elementary particle and nuclear physics. The quantum field theory is the framework of quantum theory combined with Einstein's special relativity to describe the physical properties of elementary particles. The standard model (SM) of particle physics, which is scripted with QFT, represents the state-of-the-art understanding of the most basic physical law of the elements of matters in this world. This almost perfect law of nature is still incomplete in several reasoning: it does not explain the origin of dark matter; it does not explain the spectrum of elementary particles; it is not natural that highest energy scale in SM is seventeenth order of magnitude lower the Plank scale; etc. These motivate searches of new physics in various theoretical and experimental directions. Among these precision tests of SM and seeking answers of these questions in extensions of SM require precise information of the quantum chromo dynamics (QCD) which governs the strong interaction in SM. The involved QCD dynamics cannot be solved by hand. Computational approach with lattice QCD, discretized version of the QCD on the continuous space-time, is powerful and most effective. After the first lattice QCD simulation about 40 years ago, its technique has become sophisticated and matured to date by tremendous efforts. Realistic simulations with various parameters tuned as in the nature are becoming possible by use of supercomputer like K using a conventional version of the lattice QCD formulation. The required precision to many interesting quantities for the test of the standard model, however, has not been reached yet. In this situation, the breakthrough could be obtained by using methods which are ideal but computationally demanding. We believe using the next generation supercomputers like Fugaku with related developments makes this happen.

To maximize the power of the next generation supercomputers, multifaceted improvements are needed. First, algorithmic development for simulation technique and analysis of the large data set will be more than useful. Second, given the idea in algorithms, efficiency in the future HPC environment need to be maximized. The team conducts researches in such directions.

5.3 Research Results and Achievements

5.3.1 Lattice QCD codes and algorithms

Algorithm and code development of Lattice QCD (LQCD) for the supercomputer Fugaku is one of the most important missions of the team. The QCD Wide SIMD (QWS) Library is produced in the Flagship 2020 Project and open for public at github [30] in FY2019. QWS provides libraries for most demanding numerical efforts in the Wilson fermion simulations. Namely the linear solvers for Wilson types are supported. The QWS library would be the most effective for simulations using Wilson-type fermions on Fugaku. With further critical assessment it achieved [38+]× speed up compared with the K supercomputer for the target application Wilson fermion solver. The related presentation and publications include Ref. [15,16,19-22,25,26,28].

In the particle physics applications, domain-wall fermions (DWFs), a most efficient chiral fermion formulation to date, are the promising framework for the Fugaku era. DWFs are variants of the Wilson fermions. It is a five dimensional Wilson fermion with a special boundary condition imposed in the fifth direction, which effectively makes domain walls in the both ends of the fifth direction. The four-dimensional physical degree of freedom is projected out from the wall positions. Each four-dimensional slice of the DWF at a fixed fifth dimensional position is a four dimensional Wilson fermion. The 5D DWF is superficially a connection of many 4D Wilson fermions. Eventually QWS may be used as a high speed library to handle the 4D Wilson part there. For that preparation in the upper level LQCD packages from which QWS is called is needed.

The LQCD packages Bridge++ and Grid are the candidates for the use of DWF applications. This year with the supercomputer Fugaku made available for our usability research, test and development of Grid have been underway. Grid was tested using Intel Skylake and Knights Landing processors in past and showed effectiveness in DWF simulations. Implementation for the ARM/SVE specification of A64FX CPU on Fugaku has been done by the group of the University of Regensburg. Using that branch of Grid, porting of the code to Fugaku has been worked out with a help from Regensburg and then some benchmark tests have been performed.

The left Figure 5.1 shows the strong scaling of conjugate gradient (CG) solver of Möbius DWF. As usual in the lattice computation, the weak scaling is good, which is shown in the right figure. The newly developed rankmap, which can be used to optimize the mapping of physical system onto the allocated Fugaku nodes, works quite well.

One of the QCD package with Japanese initiative is Bridge++. Hybrid Monte Carlo for two-flavor QCD with domain-wall fermion algorithm was implemented in Bridge++ and tested at K-supercomputer last year. Developments for Fugaku is now underway for fermion discretization including DWF. Activities in these developments are reported in Ref. [15,28,29].

Large volume QCD simulations for Fugaku may suffer from extra slowing down. Multigrid algorithm is a promising method if implemented in efficient way. Applying it to domain-wall fermions may require special treatment as the physical light degree of freedom is not realized as simple as that of the Wilson fermion. Some development along this line is reported in the meeting of the Japan Physical Society [29].

HPC-Phys workshops [1] are domestic meeting aiming to encourage exchanging ideas for high-performance computing especially research and developments for large scale numerical simulations in fundamental physics. This workshop in series are supported by the Joint Institute for Computational Fundamental Science (JICFuS). This year four workshops are held online hosted by R-CCS and organized by our team members together with the members in JICFuS. We plan to continue this activity taking main role of the organization in the future as well.

5.3.2 Finite Temperature QCD with domain wall fermions

99% of the mass of the visible universe (and our body as well) is made of the energy dynamically generated by QCD interaction. The underlying mechanism of mass generation is the chiral symmetry breaking (Nambu-Goldstone mechanism): the chiral symmetry is spontaneously broken in the present universe. One can think of what happens if we unwind the history of the universe and with the temperature high enough, then the chiral

[1] https://hpc-phys.kek.jp/

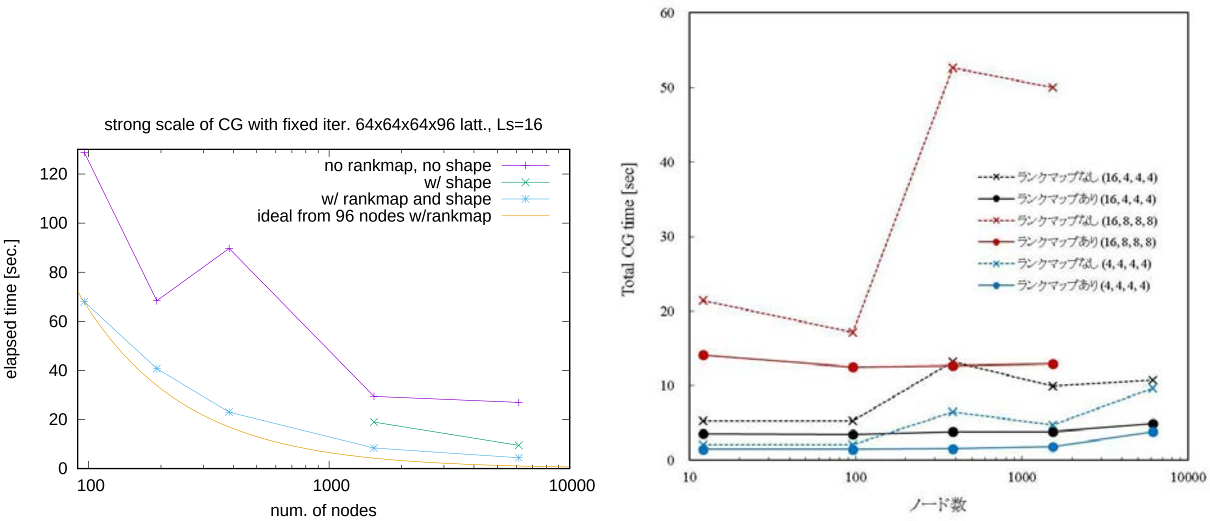


Figure 5.1: Strong (left) and weak (right) scaling of conjugate gradient (CG) of DWF using Grid on Fugaku. In the right figure the horizontal axis is number of nodes. There measurement with (ランクマップあり) and without (ランクマップなし) rankmap. Effectiveness of the rankmap is confirmed in these benchmarks.

symmetry should recover. If there has been a phase transition or just a crossover is a prime question whose answer would have had great influence on the history of the universe and the fate of the matters inside that. Furthermore, understanding the nature of the transition is necessary to judge the fate of one of the promising dark matter scenarios.

There is a long history of the research of QCD at finite temperature theoretically and experimentally. For theory side, as the strong coupling QCD is the target, numerical computation with lattice QCD is most useful. Because the target physics question is an involved interplay of the symmetry and its spontaneous breaking, we need the formulation which respects the symmetry in the first place. The lattice formulation which preserves exact "chiral symmetry", thus being called as chiral fermions, is in our hands. However, it requires tremendous computational effort compared to the conventional lattice formulations, such as Wilson or staggered fermions, which respects only a part of the full chiral symmetry or does not at all. On the supercomputer Fugaku, we aim to conduct computations using chiral fermions and by that we can get rid of the compromises which we were not able to.

Along this line we employ domain-wall fermions as a practically best implementation of the chiral fermions to date. Best approximation of the real world where six quark flavors exist is realized by neglecting the quarks heavier than the dynamical QCD scale. In this way 2+1 flavor simulation with degenerate u and d quarks and a slightly heavier s quark are the main target application to date. Before stepping to the 2+1-flavor DWF simulation, we conducted a series of 2-flavor simulations as benchmarks to the 2+1 flavor simulations. The 2-flavor system provides an approximation to 2+1 flavor world and actually known to be a very good approximation for low temperature systems. It is also expected to share crucial dynamical properties for finite temperatures near phase transition as well. This study is done in the JLQCD collaboration with a support of the priority issue No. 9 to be tackled by the supercomputer Fugaku to bridge to the 2+1 flavor simulations. There susceptibilities of topological charge, axial $U(1)$, by carefully treat the eigen modes of DWF as well as the overlap operator which is an exact chirally symmetric formulation. An enhanced symmetry is discussed using the composite particle spectrum as well. The related results are presented in Ref. [6] and [13].

Next step is to perform the realistic 2+1 flavor DWF simulations. A pilot study employing one lattice spacing, which is the coarsest among the JLQCD zero-temperature lattices, but quite fine for the finite temperature simulations to date, is conducted using Oakforest PACS in JCAHPC, Grand Chariot and Polaire at Hokkaido University through the MEXT Program for Promoting Researches on the Supercomputer Fugaku (PPR-Fugaku) - Simulation for basic science: from fundamental laws of particles to creation of nuclei. Fig. 5.2 shows the topological susceptibility as a function of quark mass. Left panel is the new result with 2+1-flavor simulations, which is compared with the 2-flavor result on the right. The temperature is approximately same for both and higher than the chiral phase transition temperature. Similar behavior is observed for both cases, indicating the small strange quark effect.

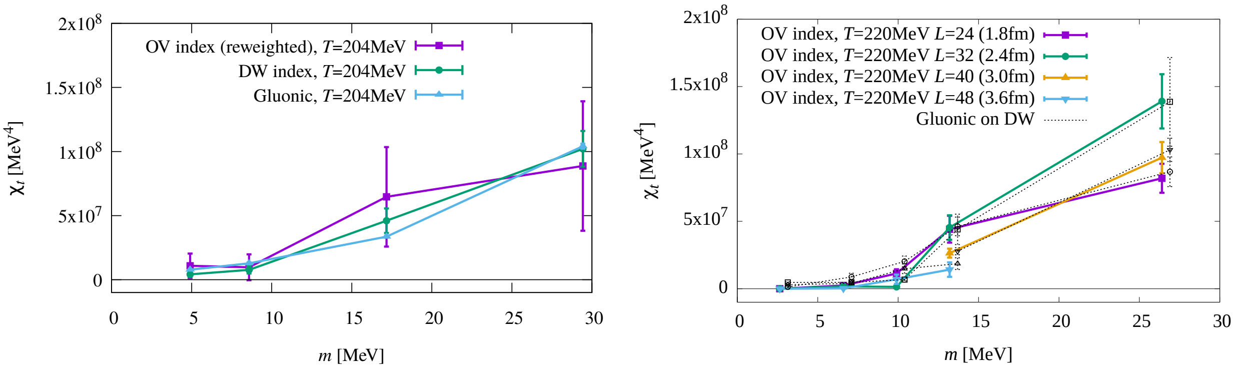


Figure 5.2: Topological susceptibility from 2+1-flavor (left) and 2-flavor DWF simulations. Figures from ref. [23]

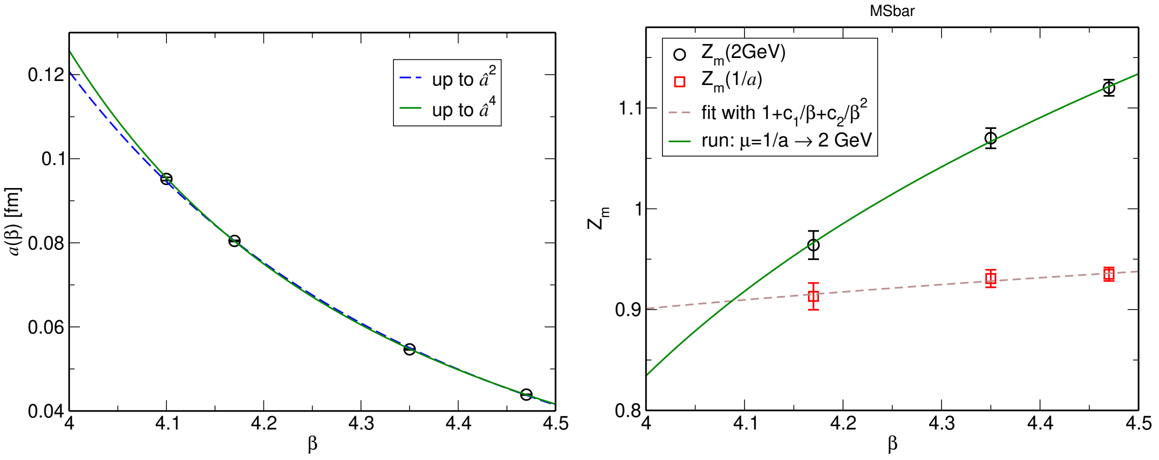


Figure 5.3: Lattice spacing a as a function of the gauge coupling β obtained for our DWF simulations (left) and the parameterization of the quark mass renormalization factor using a non-perturbative renormalization and the four-loop anomalous dimension, to be used to obtain the line of constant physics shown in Fig. 5.4.

Up to now, our investigation of finite temperature QCD has been conducted by fixing the temperature and varying the quark mass, while in the real world masses are kept fixed and only the temperature changes. As the setting of simulations, latter is convenient to see how one physical system reacts against the change of the temperature. To simulate such a situation we need to understand how to tune the quark masses as functions of the gauge coupling β through the lattice spacing $a(\beta)$ which is related to the temperature through $T = 1/(N_t a)$, where N_t is the number of lattice sites in the temporal direction. For that the lattice spacing needs to be parameterized as a function of β first. These parameterization is done by analyzing the zero temperature hadron spectrum. This will lead to the so-called line of constant physics specially determined for our DWF. The information is used to conduct the new 2+1 flavor, finite temperature QCD simulations with DWF in the next fiscal year on Fugaku through PPR-Fugaku.

To discuss the possible QCD phases at finite temperature the Columbia plot, in which the horizontal and vertical axes indicate the degenerate up/down and strange quark masses respectively (left panel of Fig. 5.5.). Recently, quite a few studies are shedding light on this plot, which is being one of the hot topics in the finite temperature lattice QCD studies. Two regions of interest in the plot are that near the physical point and the diagonal line near the chiral limit (origin). The aforementioned 2+1-flavor study would eventually unfold the phase structure near the physical point. The structure along the diagonal line (so-called 3-flavor system) is still undetermined yet.

We are trying to solve the 3-flavor phase structure using DWF. Through a series of parameter search studies conducted on Ito subsystem at Kyushu University through HPCI as well as on the HOKUSAI BigWaterfall, we have captured a signal of possibly a cross-over at high temperature and large quark mass region. The result is shown in the right panel of Fig. 5.5. One can read off the temperature and quark mass at this peak in the

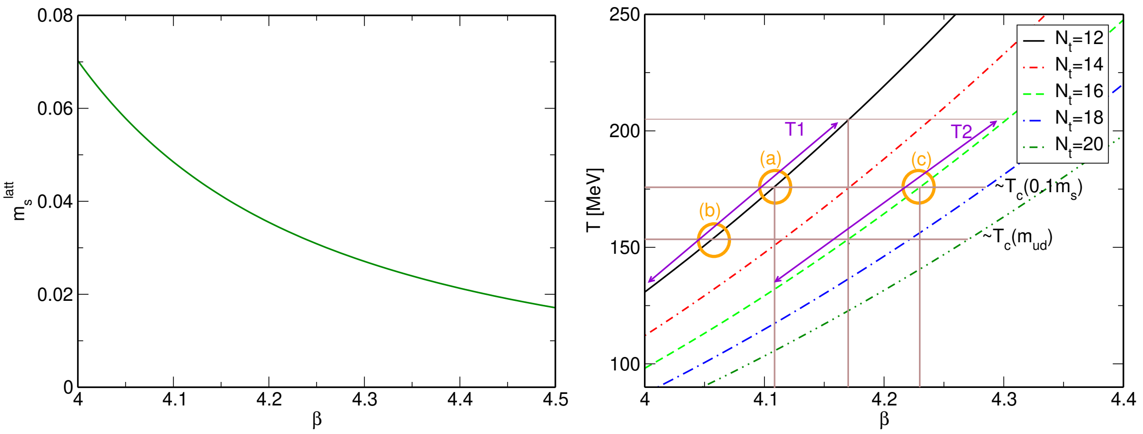


Figure 5.4: Line of constant physics showing the lattice strange quark mass as a function of the gauge coupling β (left) and temperature-β relation with possible target prime range of simulations to be performed through PPR-Fugaku.

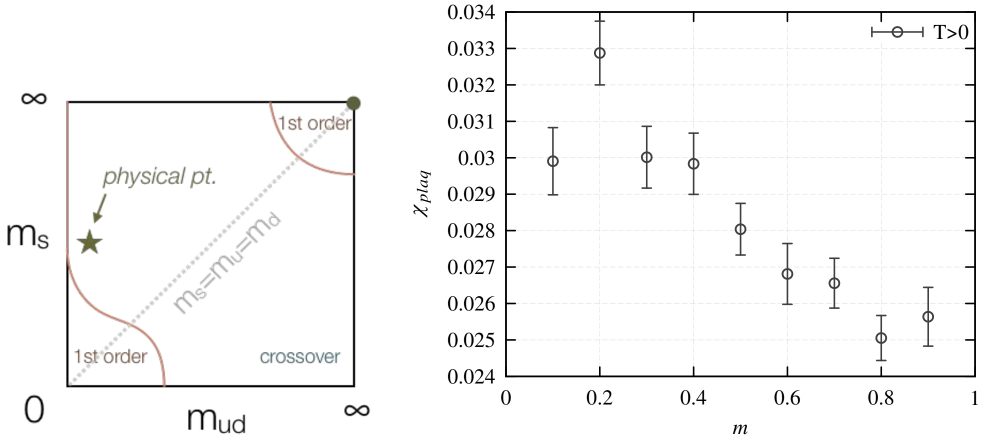


Figure 5.5: Conventional draw of the Columbia plot (left) and the plaquette susceptibility of the $N_t = 6$ lattice at $\beta = 4.0$ along the diagonal line. A peak structure indicate possible crossover.

physical units from the 2+1-flavor line of constant physics. The temperature is about 260 MeV and the quark mass is about 3 times the strange mass. Tracing such a peak towards smaller quark mass and lower temperature will eventually unfold the phase structure near the origin, which we plan to do in FY2021.

5.3.3 Miscellaneous LQCD developments

There are LQCD related publications reported by members of the team with external collaborations. These works serve as possible seeds for future high performance computing on supercomputer Fugaku and beyond that.

Study of the high temperature nature of the 2-flavor QCD is extended and reported on the fate of the $U(1)_A$ symmetry and extended symmetry of the meson sector($SU(2)_{CS}$ and $SU(4)$) [6][13].

One of the promising application of DWF is to simulate the B meson decays. A study focusing on the $B \to D^{(*)} l\nu$ form factors, which would provide crucial information to the precision study of Cabibbo-Kobayashi-Maskawa matrix. Preliminary results with three different fine lattice spacings using Oakforest-PACS through HPCI are reported [10][18].

Proton decay matrix elements and nucleon isovector couplings are computed using an improved Wilson fermion formulation with the non-perturbative renormalization employed for the operators. These works are one in the PACS collaboration. The preliminary results are reported in Refs. [5][14].

Issaku Kanamori on the pion distribution amplitude using the heavy quark operator product expansion [4]. They also reported on the SU(2) gauge, many-flavor system [12].

Yoshifumi Nakamura published on works using Wilson fermions for QCD+QED simulations [1] as well as on the Compton amplitudes [2]. He also studied the K_{l3} semi-leptonic kaon decay in the PACS Collaboration [3].

5.4 Schedule and Future Plan

The chiral fermion simulations of lattice QCD for the supercomputer Fugaku will eventually use 2+1 flavor (including the strange quark) physical point for heavy flavor physics targeting B mesons and the other precision tests of the standard model (SM). The same setting will also be used to explore the phase diagram of QCD at finite temperature which could end up simulating lighter quarks than physical. The Fugaku implementation of the existing algorithms are being developed and will further be extended to several LQCD packages in the future. Extended simulations to finer lattice, larger volume, and lighter quarks from the current ones, which would be the next stage on Fugaku as well as the post-Fugaku machines benefit from speedup with algorithmic development and tuning. There existing speedup techniques such as AMA or multigrid algorithm may serve as improvements. Developments in the methods beyond the conventional Monte-Carlo based on the molecular dynamics may find its use in the future simulations.

5.4.1 Algorithm and code development for the chiral fermion simulation in QCD for Fugaku

The chiral fermion simulations of two-flavor and 2+1-flavor QCD for now Grid, the code set developed mainly by Edinburgh University. Implementation for the ARM/SVE specification of A64FX CPU on Fugaku has been done by the group of the University of Regensburg. Through the discussion with the researchers in Regensburg a first series of DWF simulation has been made possible on Fugaku and it was useful for the stress test of Fugaku. Yet another source package called Bridge++, Japanese initiative, is also a promising framework. We will be implementing some applications of DWF there.

The large volume and light quark simulations are computationally challenging. The AMA technique, being developed and tested in last year, will be useful. Independent of this technique, implementing and accelerating Multi-Grid algorithm would be a promising direction. In the lattice simulations, parameters not directly related with the physical setting, like the AMA and multi-grid specific ones or like tunable parameters in the hybrid Monte-Carlo potentially provide large rooms of improvement. There, optimizations assisted with deep learning could bring a significant speed up.

5.4.2 QCD phase diagram with chiral fermion simulations

As a test bed of new code development and new algorithms the degenerate three flavor simulation of QCD using chiral fermions can be used. The simulation is simpler than physical, non-degenerate, 2+1 flavor simulation, thus good for tests with situation close to the final target. But, it is also interesting on its own physical perspective. The phase diagram with respect to the change of the degenerate quark mass is one of the important boundary of 2+1 flavor QCD phase diagram. Recent results reported using conventional fermions suffer from huge discretization error. Use of the chiral fermions, thus being able to simulate with correct counting of the light degree of freedom, is expected to solve this problem. We started an experimental simulations with small size on existing HPC resources and found a tail of the phase transition (possibly crossover) first time using the chiral fermions. Towards realistic simulations with larger volume and lighter quark mass, Fugaku-scale supercomputer is needed. Tests of the algorithm and codes for Fugaku is now complete, we will be conducting a large scale simulation in the coming year. The knowledge acquired through these studies provide a base of the simulation of more challenging 2+1 flavor, where we would find further rooms for development on the algorithms and codes.

5.4.3 Collaboration on physics applications

We have a close collaboration with the new "Program for Promoting Researches on the Supercomputer Fugaku" (Simulation for basic science: from fundamental laws of particles to creation of nuclei) from this year. It will be continued for three years. We started the simulation on Fugaku in the end of the first year and full scale simulations will be performed in the FY 2021 and 2022. Through this collaboration, we aim to maximize the scientific output of the issues for intensity frontier subjects including flavor physics, as well as for the QCD phase by cutting edge developments.

5.5 Publications

5.5.1 Articles/Journal

[1] NPLQCD and QCDSF Collaborations: S.R. Beane, W. Detmold, R. Horsley, M. Illa, M. Jafry, D.J. Murphy, Y. Nakamura, H. Perlt, P.E.L. Rakow, G. Schierholz, P.E. Shanahan, H. Stüben, M.L. Wagman, F. Winter, R.D. Young, J.M. Zanotti, "Charged multihadron systems in lattice QCD+QED", Phys.Rev.D 103 (2021) 5, 054504.

[2] K.U. Can, A. Hannaford-Gunn, R. Horsley, Y. Nakamura, H. Perlt, P.E.L. Rakow, G. Schierholz, K.Y. Somfleth, H. Stüben, R.D. Young, J.M. Zanotti, "Lattice QCD evaluation of the Compton amplitude employing the Feynman-Hellmann theorem", Phys.Rev.D 102 (2020) 114505.

[3] PACS Collaboration: Junpei Kakazu, Ken-ichi Ishikawa, Naruhito Ishizuka, Yoshinobu Kuramashi, Yoshifumi Nakamura, Yusuke Namekawa, Yusuke Taniguchi, Naoya Ukita, Takeshi Yamazaki, Tomoteru Yoshié, "K_{l3} form factors at the physical point on a (10.9 fm)^3 volume", Phys.Rev.D 101 (2020) 9, 094504.

5.5.2 Conference Papers

[4] William Detmold, Anthony V. Grebe, Issaku Kanamori, C.-J. David Lin, Santanu Mondal, Robert J. Perry, Yong Zhao, "A Preliminary Determination of the Second Mellin Moment of the Pion's Distribution Amplitude Using the Heavy Quark Operator Product Expansion", APLAT 2020 [arXiv:2009.09473].

[5] Yasumichi Aoki, Yoshinobu Kuramashi, Eigo Shintani, Natuki Tsukamoto (PACS Collaboration), "Proton decay matrix elements with physical quark masses", PoS LATTICE2019 (2020) 141.

[6] JLQCD Collaboration: Kei Suzuki, Sinya Aoki, Yasumichi Aoki, Guido Cossu, Hidenori Fukaya, Shoji Hashimoto, Christian Rohrhofer, "Axial U(1) symmetry and mesonic correlators at high temperature in $N_f = 2$ lattice QCD", PoS LATTICE2019 (2020) 178.

[7] R. Horsley, T. Howson, W. Kamleh, Y. Nakamura, H. Perlt, P.E.L. Rakow, G. Schierholz, H. Stüben, R.D. Young, J.M. Zanotti, "Determining the glue component of the nucleon", PoS LATTICE2019 (2020) 220.

[8] A. Hannaford-Gunn, R. Horsley, Y. Nakamura, H. Perlt, P.E.L. Rakow, G. Schierholz, K. Somfleth, H. Stüben, R.D. Young, J.M. Zanotti "Scaling and higher twist in the nucleon Compton amplitude", PoS LATTICE2019 (2020) 278.

[9] Antonio González-Arroyo, Issaku Kanamori, Ken-Ichi Ishikawa, Kanata Miyahana, Masanori Okawa, Ryoichiro Ueno, "Towards higher order numerical stochastic perturbation computation applied to the twisted Eguchi-Kawai model", PoS LATTICE2019 (2020) 030.

[10] JLQCD Collaboration: T. Kaneko, Y. Aoki, G. Bailas, B. Colquhoun, H. Fukaya, S. Hashimoto, J. Koponen, "$B \to D^{(*)} l\nu$ form factors from lattice QCD with relativistic heavy quarks", PoS LATTICE2019 (2020) 139.

[11] Y. Nakamura, G. Schierholz, "Does confinement imply CP invariance of the strong interactions?" PoS LATTICE2019 (2020) 172.

[12] Issaku Kanamori, C.-J. David Lin, "Chiral Condensate and Susceptibility of SU(2) $n_f = 8$ Naive Staggered System", PoS LATTICE2019 (2020) 034.

[13] C. Rohrhofer, Y. Aoki, G. Cossu, H. Fukaya, C. Gattringer, L.Ya. Glozman, S. Hashimoto, C.B. Lang, K. Suzuki, "Symmetries of the Light Hadron Spectrum in High Temperature QCD" PoS LATTICE2019 (2020) 227.

[14] Natsuki Tsukamoto, Yasumichi Aoki, Ken-Ichi Ishikawa, Yoshinobu Kuramashi, Eigo Shintani, Shoichi Sasaki, Takeshi Yamazaki, "Nucleon isovector couplings from 2+1 flavor lattice QCD at the physical point", PoS LATTICE2019 (2020) 132.

5.5.3 Posters

[15] I. Kanamori, S. Aoki, Y. Aoki, T. Aoyama, T. Doi, S. Hashimoto, K.-I. Ishikawa, T. Kaneko, H. Matsufuru, T. Nagai, Y. Nakamura, "Preparation of LQCD code for Fugaku", the 3rd R-CCS International Symposium (Feb. 15-16, 2021, Kobe, Japan, online).

[16] Y. Nakamura (collaboration with LQCD working group in FS2020 project), "Benchmarking QCD Wide SIMD Library (QWS) on Fugaku", the 3rd R-CCS international symposium (Feb. 15-16, 2021, Kobe, Japan, online).

[17] Y. Nakamura, Y. Aoki, I. Kanamori, “Finite temperature phase transition for three flavor QCD with domain wall fermions”, the 3rd R-CCS international symposium (Feb. 15-16, 2021, Kobe, Japan, online).

5.5.4 Invited Talks

[18] Y. Aoki, “Heavy quark physics, codes and machines”, EXALAT - Lattice Field Theory at the Exascale (June 15-17, 2020, UK/online).
[19] Y. Nakamura, “Supercomputer Fugaku and code development”, スーパーコンピュータ「富岳」とコード開発, The Physical Society of Japan 2021 Annual (76th) Meeting (Mar. 12-15, 2021, online).
[20] Y. Nakamura, “LQCD tuning on Fugaku”, 2nd meeting for application code tuning on A64FX computer systems (Dec. 23, 2020, RIKEN R-CCS and RIST, online).

5.5.5 Oral Talks

[21] Y. Nakamura, “Supercomputer Fugaku and QCD Wide SIMD Library (QWS) on Fugaku”, Asia-Pacific Symposium for Lattice Field Theory (APLAT 2020) (Aug. 4-7, 2020, online).
[22] I. Kanamori, “Implementation of neighboring communication in QWS”, Asia-Pacific Symposium for Lattice Field Theory (APLAT 2020) (Aug. 4-7, 2020, online).
[23] S. Aoki, Y. Aoki, H. Fukaya, S. Hashimoto, I. Kanamori, T. Kaneko, Y. Nakamura, C. Rohrhofer, K Suzuki, “Axial U(1) anomaly in 2+1 flavor QCD at high temperature”, 2020 Autumn Meeting of Physical Society of Japan (Sep. 14-17, 2020, online).
[24] Y. Nakamura, Y. Kuramashi, H. Ohno, S. Takeda, “Study of QCD with finite temperature”, 12th symposium on Discovery, Fusion, Creation of New Knowledge by Multidisciplinary Computational Sciences (Oct. 6, 2020, online).
[25] Y. Aoki, “Lattice QCD simulations in the Fugaku era”, R-CCS Cafe (Oct. 5, 2020, Kobe, Japan, online).
[26] I. Kanamori, Y. Nakamura, K. Nitadori, M. Tsuji, Y Mukai, I. Miyoshi H. Matsufuru, K.-I. Ishikawa, “Acceleration of communication with low latency uTofu interface in LQCD application” 第177回ハイパフォーマンスコンピューティング研究発表会 (Dec. 21-22, 2020, online).
[27] Y. Aoki “QCD相構造 (QCD phase structure)”, 「富岳で加速する素粒子・原子核・宇宙・惑星」シンポジウム (Jan. 28-29, 2021, online).
[28] I. Kanamori, “富岳での格子QCDコードの整備状況 (Current Status of LQCD code for Fugaku)”, 「富岳で加速する素粒子・原子核・宇宙・惑星」シンポジウム (Jan. 28-29, 2021, online).
[29] K.-I. Ishikawa, I. Kanamori, H. Matsufuru, “Implementation of multigrid solvers on multi-platform with general code Bridge++”, JPS 76th Annual Meeting, The Physical Society of Japan (Mar. 12-15, 2021, online).

5.5.6 Software

[30] Y. Nakamura, Y. Mukai, K.-I. Ishikawa, I. Kanamori, “QCD Wide SIMD Library (QWS)”, `https://github.com/RIKEN-LQCD/qws`

Chapter 6

Discrete-Event Simulation Research Team

6.1 Members

Nobuyasu Ito (Team Leader)

Toshiaki Iitaka(Research Scientist)

Yohsuke Murase(Research Scientist)

Naoki Yoshioka (Research Scientist)

Daigo Umemoto (Guest Researcher)

Tomio Kamada (Guest Researcher)

Hiroyasu Inoue (Guest Researcher)

6.2 Research Activities

Discrete-event simulation comprise various kinds of models, for example, particles, agents, automata, games and so on, and their applications are from material and biomedical sciences to ecological and environmental problems. Furthermore, computer systems executing various simulations including discrete-event ones are also described as discrete-event systems and threrefore design and construction of new computer requires discrete-event simulations. Discrete-event simulation research team(DESRT), unique research hub of discrete-event treatment in the Riken Center for Computational Science, has been challenging simulations for natural and social phenomena, and for future computer. Social designs and controls have becoming the more interesting target since so-called the "big data sciences" became popular. Continuous simulations are also becoming building block of discrete-event simulations. The DESRT aims to cultivate such applications of supercomputers.

One characteristic feature of discrete-event simulations is, of course, their discrete nature. Interpolation of two results from two parameter sets is not straightforward, but verification using results of simulations for parameter sets between them is expected. Discrete-event models often comprise with tens, hundreds, thousands, or more parameters which are often, again, discrete. So necessary number of simulations usually very large, and exhaustive parameter search in brute force way is not feasible even with a big supercomputer. One promising way is to use a Bayesian way: do best of what we observed. So-called the AI methods like multivariate statistical analysis, agent-based modeling, genetic method, and machine learning are expected to be powerful for the purposes.

Activities of the DESRT in the year 2020 are the following:

- simulation analysis of social phenomena: the covid-19 pandemic and its economic influence
- development and support of job-management application software named OACIS and CARAVAN
- game theoretic analysis for social dilemma

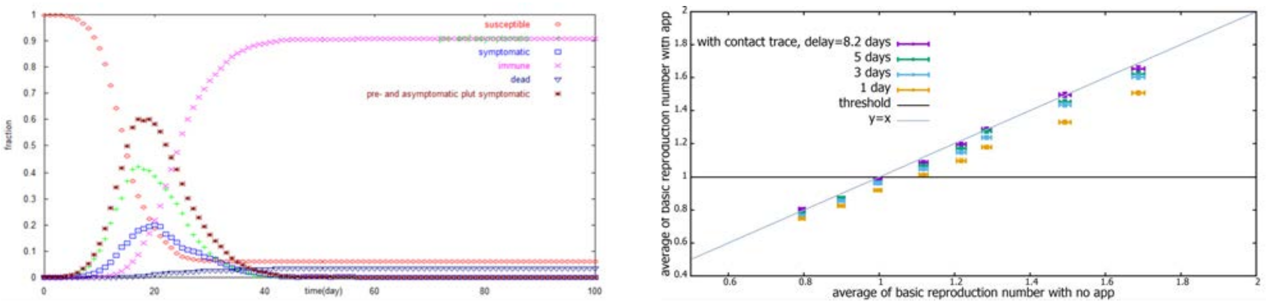


Figure 6.1: (Left)Typical behavior of Covid-19 infection clusters is shown for 1000 agents and $R_0 = 2.0$. Horizontal axis shows days after first infected agent apprears. Vertical axis shows fractions of S-state agents(red), IP and IA (light green), IS(blue), R(violet), D(black), and infected agents(brown). (Right)Relation between the reproduction number with and without contact-tracing applications.Horizontal axis shows basic reproduction number without application. Vertical axis shows effective reproduction number with applications of several delay days from symptom to alert.

- simulation analysis of nonlinear nonequilibrium phenomena and extreme substances: simulation of extreme substances
- development an application software for quantum-computer simulation

Some of their highlight achievements are described briefly in the following subsections.

6.2.1 Simulation of the covid-19 spread

Pandemic of the Covid-19 disease revealed that global economy and society is still with the risk of new pathogen and new disease as it has been always the case throughout of human history. Though discovery of antibiotics, development of medical science and public hygiene since last century, we are still lacking definitive countermeasure. Present society has a distinctive feature which had not existed previously. It is the information and communication technology (ICT). Giga-bit telecommunication, Petaflops processing for Exabyte data is now part of our society and daily life, and it is expected to be a new countermeasure for the disease propagation.

The Covid-19 challenge with the Fugaku covers from molecular to social level. As one of the social-level projects, analysis of the Covid-19 propagation in our society started. It comprises with simulation of disease propagation and "big data" analysis of human mobility. For these purpose, two application libraries have been developed mainly for the Fugaku using C++ with openMP and MPI for parallelization: one is a disease-spreading library named "DisSim"(Disease Simulator) and the other is a GPS-data mining library named "GPSMiner"(GPS-data miner), which are described in the following.

DisSim simulates propagation of infectious disease and state of each person using an agent-based model. Each model agent takes one of the following six states:

- S: susceptible state without infection nor immunity against the pathogen
- IP: pre-symptomatic incubation state after infection
- IS: symptomatic state after infection and incubation period
- R: recovered and immune state after infection
- IA: asymptomatic state after infection
- D: dead state

Transition conditions and model parameters are determined based on medical data and sensitivity analyses[18-20].

Simulation analysis of infection-cluster dynamics are in Fig.6.1(Left), and suppression effect to infection spread using contact-tracing applications are in Fig. 6.1(Right).

Modern ICT provides a new approach to diagnose current social state: so-called the "Big-data mining", that is, various data accumulated through various ICT devices and communication path. Mobile terminals

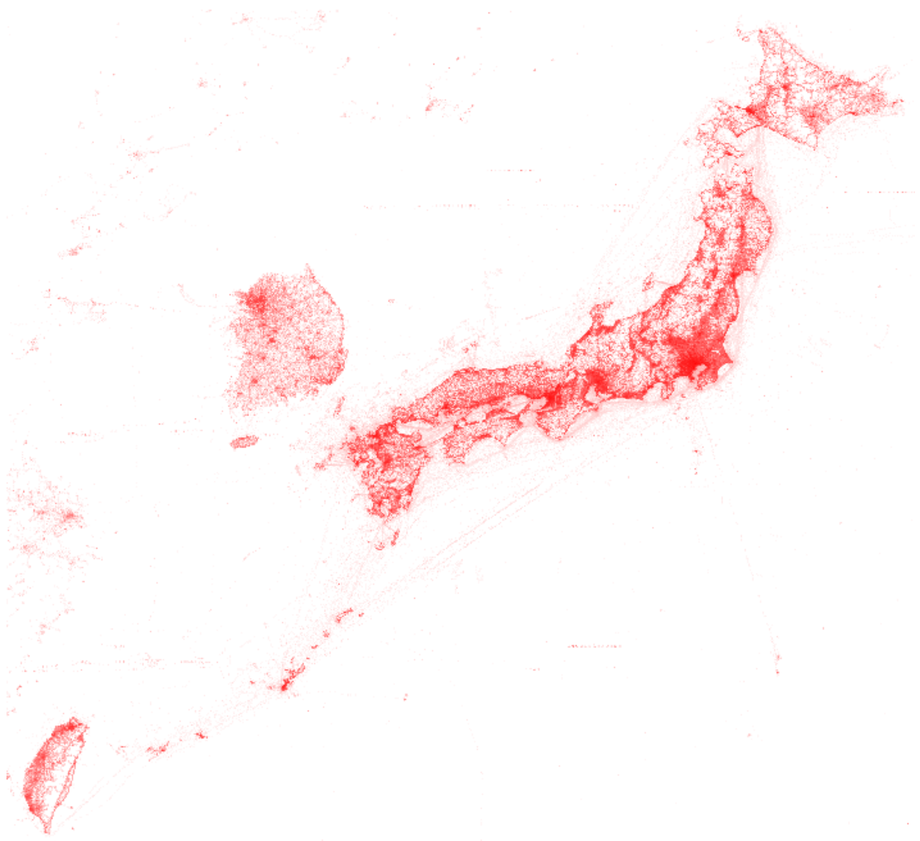


Figure 6.2: Distribution of GPS records for a month in 2020 is plotted using GPSMiner. Totally about 30 billion records are accumulated in the third-level area-code-wisely. Red color strength is proportionally determined to logarithm value of degree in each mesh. Human distribution draws not only cities in eastern asian countries, but also their relations like transportation trails.

and individual data through them are typical. Especially, location data from global positioning system(GPS) are useful to analyze human mobility, and therefore infectious disease spreading. Appropriate preparation and organization from such original big data is important to use them. For example, daily size of available GPS data from mobile terminal in Japan is 100GB or more with billions of records. Just to scan the data is already some work, and massive parallel computer is necessary to analyze over annual-time scale using 10TB or more data. GPSMiner is a software library to treat original GPS or other big data Fugaku and other massive-parallel computer. Its major functions are sorting records and connected cluster analysis using linkage information extracted from records. So far it is organized to treat hundreds billions of on-memory records of the total size of tens TB using thousands nodes of Fugaku. To hold each record, a C++ class named “record” is implemented and used. Current GPSMiner is designed for a specific GPS data record, and the record class is designed to treat the specific data structure. Each recod holds some ID tags, space-time coordinate and its resolution, and its size is about 100B. Usually 100 million or less records are assigned to each node of Fugaku with 32GiB so that each node holds two or more set of records for analysis. Using 3,072 nodes, totally 147,456cores and 96TiB memory, of a connectivity shape of 16 x 12 x 16 on Fugaku, initial data loading of about 0.4 trillion records stored in about 300 thousands files totally about 40TB takes about 2,000 seconds, and sorting and cluster analysis takes about 100 to 200 second. Writing all records to files takes about 2,000 seconds. Geographic information system(GIS) of GPSMiner supports Japanese area mesh code system(JIS X 0410). Spatial resolution of GPS coordinate, latitude and longitude, in each record is not uniform, but it depends strongly on local environment. GPSMiner has a function to make geographic distribution using an available data suggesting their accuracy, and Japanese area mesh code system. Graphics support function of GPSMiner uses postscript format. Examples of visualization are in Fig.6.2.

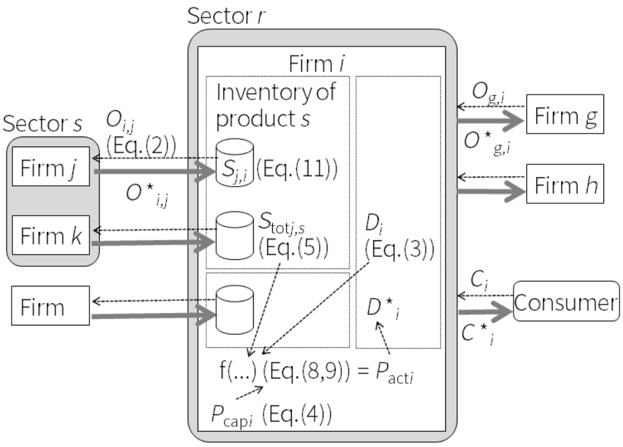


Figure 6.3: Overview of the agent-based model. Products flow from left to right, whereas orders flow in the opposite direction. The equation numbers correspond to those in [1].

6.2.2 Economic influence of the covid-19

To prevent the spread of COVID-19 (SARS-CoV-2) in 2020, many governments have imposed "lockdowns" on cities, regions, or entire countries, shutting down or shrinking economic and social activities. As of March, 2021, while some countries and regions have lifted lockdowns, others continue or have resumed lockdowns because of the second and third waves of the pandemic. These lockdowns obviously reduce the economic activity of regions under lockdowns. In addition, the negative economic effects of lockdowns diffuse through supply chains, i.e., supplier-client relationships among firms, to other regions that are not necessarily locked down. When a firm shrinks its production due to lockdown, its client firms suffer from reduced production because of a lack of supply of intermediate goods and services. Its suppliers also reduce production because of demand shortages. One important aspect not fully examined in the literature is interactions of lockdowns in different regions and countries and the resulting need for policy coordination, although interregional and international policy coordination is discussed in some other contexts. This study investigates the effect of regional coordination of lockdowns and lifts of them. To do so, we conduct a simulation analysis applying an agent-based model of production to actual supply chain data on 1.6 million firms in Japan.

We employ the dynamic agent-based model. A brief explanation of the model is as follows: firms utilize various intermediates as inputs and provide a sector-specific product to client firms and final consumers. Firms have an inventory of intermediates from suppliers. However, they have no inventory of completed product, and the completed product is immediately delivered to clients. In addition, there are no price or market mechanisms. An overview of the model is depicted in Figure 6.3.

Regarding the coordinations of the lockdowns, we found (1) coordinated lockdowns in multiple regions can reduce economic loss and (2) the effect of coordination increases according to the economic connections between the regions. The greater the number of links between two regions is, the less the economic loss. The links have a stronger correlation than the geodesic distance or the GRP sum of the regions. Regarding the lifting of the lockdowns, on the other hand, we found that a region's upstreamness, intensity of loops, and supplier substitutability in supply chains with other regions largely determine the economic effect of the lockdown in the region. In particular, when a region lifts its lockdown, its economic recovery substantially varies depending on whether it lifts the lockdown alone or together with another region closely linked through supply chains. These results indicate that the economic effect produced by exogenous shocks in a region can affect other regions and therefore this study proposes the need for inter-region policy coordination to reduce economic loss due to lockdowns.

6.2.3 Job management applications, OACIS and CARAVAN

A characteristic feature of discrete-event models is their large and complex parameter spaces, and they often show highly nonlinear and qualitatively distinct behaviors with different parameters. Supercomputers are promising tool to help us overcome such difficulty: their performance with extreme parallelism allows us to simulate a huge number of simulation runs exploring the parameter spaces. While the parallelization of such explorations in the parameter spaces is trivially possible, handling a lot of jobs efficiently in a traceable way has its own

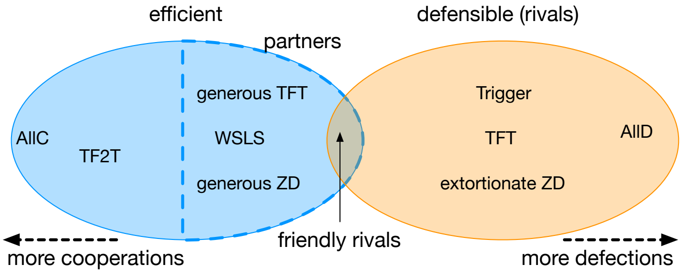


Figure 6.4: A schematic diagram of the strategy space. Strategies that tend to cooperate (defect) are shown on the left (right). The blue (red) ellipse represents a set of partner (rival) strategies. We found that the intersection, called 'friendly rivals', indeed exists and shows excellent performance in evolutionary games.

challenge in supercomputing when the number of required jobs is huge. OACIS and CARAVAN are open-source software frameworks to overcome these challenges being developed by our team. They have different use cases and purposes. For the details, refer to their release pages on Github[33,34].

In FY2020, we released an update of OACIS including a couple of UI improvements and bug fixes. CARAVAN was largely rewritten in C++, which used to be written in X10, for better compatibility and maintainability. In addition to these updates, support for supercomputer Fugaku has been newly added to these frameworks.

6.2.4 Game theoretic analysis for social dilemma

The tragedy of the commons or the social dilemmas, the discrepancy between the societal optimums and the individual optimums, are ubiquitous in our society when organizing collective action toward a common goal. Prisoner's Dilemma and the public goods game are the most fundamental theoretical framework which have been intensively studied for decades. According to the recent understanding of the Prisoner's Dilemma, most of well-known strategies fall into one of the classes called "partners" and "rivals". When acting as a partner, a player aims at mutual cooperation and punishes the co-player's defection from it. As a rival, on the other hand, the player aims at a payoff higher than or equal to the co-player's, regardless of the co-player's strategy. A schematic diagram of the strategy space is shown in Fig. 6.4. A natural question would be on the possibility that a single strategy is both a partner and a rival simultaneously. Let us call such a strategy a 'friendly rival' hereafter. If such a strategy exists, mutual cooperation is realized with assuring that the player will never be beaten by any kind of opponents. However, the construction of a friendly rival is not easy since the requirements for being a partner and for being a rival are seemingly contradictory therefore it has long been believed that such a strategy does not exist.

We tackled this problem [6] by exhaustively searching the memory-3 strategy space, which amounts to $2^{64} \approx 1.84 \times 10^{19}$ strategies in total, by using K-computer. We developed a graph-theoretic algorithm to comprehensively search this space and succeeded in finding that there indeed exist about seven trillions friendly rivals in this space. Although its fraction to the entire strategy space is tiny, the space indeed contains diverse friendly-rival strategies. Among these strategies, we discovered a strategy which is fairly easy to interpret, named 'CAPRI'. By investigating this strategy closely, we also succeeded in generalizing the strategy to general n-player public goods game and verifying the strategies indeed qualifies as a friendly rival by supercomputer Fugaku [7]. To our knowledge, this is one of the first applications of a supercomputer to a problem in game theory, in which analytical calculations have been the main stream traditionally. We believe that these researches will open up the application of supercomputers to game theory.

6.2.5 Simulation of extreme substances

Pressure and temperature are two critical thermodynamic variables controlling the stability and chemical reactivity of materials. Temperature can be easily tuned from ambient to absolute zero or to a few thousand Kelvin. In comparison, nowadays pressure can be changed over ten orders of magnitude from ultrahigh vacuum (10^{-16} GPa) to the pressure at the center of the Earth (350 GPa) in the laboratory. Thus, the energy barriers associated with structural and chemical transformations can be overcome by manipulating the two variables leading to the creation of new materials with exotic behaviors and functionalities. One would speculate on the potential impact on materials research and the investigation of the properties of matters under extreme

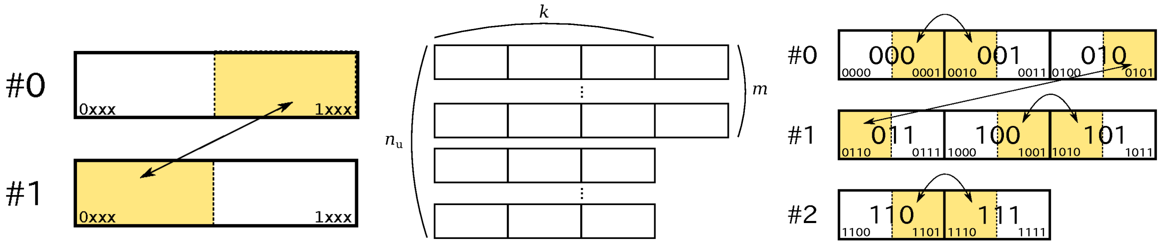


Figure 6.5: (Left) Schematic picture of MPI communications if the number of MPI processes is two, as an example the number of MPI processes is a power of two. Swap a global qubit and a most significant local qubit. (Center) Schematic picture of *unit* of MPI processes. Each unit has n_{u} MPI processes, and m processes of them have $k+1$ *data blocks*. (Right) MPI communications using unit method in the case a least significant unit qubit is swapped to a most significant local qubit.

conditions, such as mineral physics and chemistry under extreme pressure of the Earth. However, formidable theoretical and technical challenges remain to realize the expectation. We have made several fundamental contributions, through computations and theory developments, to the progress in this area. Our research addresses two fundamental issues: development of a theoretical framework to explain and predict the crystal and electronic structures and to elucidate the chemistry of materials under high pressure and high temperature. The knowledge is essential to guide formulating strategies for synthesis of new materials and, equally important, to understand the physics and chemistry of Earth that directly affects our environment.

Knowledge of the macroscopic physical properties of silicate melt (magma) under high temperature and high pressure is extremely important for understanding the phenomena of terrestrial planets such as the formation of the early Earth, the transport of volatile elements, and volcanic eruptions. We study the influence of the microscopic structures (structural factors, radial distribution function, coordination number, network topology, etc.) of basalt melts and basalt glasses, which are related to the history of earth formation, on their macroscopic physical properties (density, diffusivity, viscosity, thermal conductivity, etc.), paying particular attention to the medium-range structure, trace elements, and entropy effect (temperature effect).

Transport properties like diffusivity and viscosity of melts dictated the evolution of the Earth's early magma oceans. We reported the structure, density, diffusivity, electrical conductivity and viscosity of a model basaltic ($Ca_{11}Mg_7Al_8Si_{22}O_{74}$) melt from first-principles molecular dynamics calculations at temperatures of 2200 K (0 to 82 GPa) and 3000 K (40–70 GPa). A key finding is that, although the density and coordination numbers around Si and Al increase with pressure, the Si–O and Al–O bonds become more ionic and weaker. The temporal atomic interactions at high pressure are fluxional and fragile, making the atoms more mobile and reversing the trend in transport properties at pressures near 50 GPa. The reversed melt viscosity under lower mantle conditions allows new constraints on the timescales of the early Earth's magma oceans and also provides the first tantalizing explanation for the horizontal deflections of superplumes at 1000 km below the Earth's surface [8].

6.2.6 Simulation of quantum computer

Quantum information processing is expected to be a next breakthrough beyond current computers. Noisy Intermediate-Scale Quantum (NISQ) devices such as 50-qubit quantum computer of IBM and 72-qubit quantum computer of Google have been developed recently. In order to evaluate the complexity of new quantum algorithms and validate those NISQ devices, classical simulation of quantum circuits has a crucial role. We have been developing full-state qunatum circuit simulator `braket` [35]. It was succeeded to simulate quantum circuits upto 45 qubits in K computer if using two double precision floating-point numbers to represent a complex number[1]. In this year, in addition to porting `braket` to Fugaku, we have modified (1) operations of gates represented by diagonal matrices not to use MPI communications, and also been modifying (2) to support the case the number of MPI processes is not a power of two.

Let us denote a state $|\Psi\rangle = \sum_n a_n |n\rangle$, where $n = n_N \cdots n_1$ in binary representation and $|n\rangle = |n_N\rangle_N \otimes \cdots \otimes |n_1\rangle_1$. In the massively parallel simulation of quantum circuits, we have to consider two kinds of qubits; *logical* qubits $\left(|n_j\rangle_j\right)$ and *physical* qubits $\left(|\tilde{n}_j\rangle_j\right)$, where σ is a permutation and $\tilde{n}_j = n_{\sigma^{-1}(j)}$. While logical qubits

[1] H. De Raedt, *et al.*, Comp. Phys. Commun. **237**, 47 (2019).

correspond to real qubits, physical qubits represent a MPI rank and an index of data array where the coefficient a_n exists. We divide physical qubits into M *global qubits* $|\tilde{n}_{L+M} \cdots \tilde{n}_{L+1}\rangle = |\tilde{n}_{L+M}\rangle_{L+M} \otimes \cdots \otimes |\tilde{n}_{L+1}\rangle_{L+1}$ and $L = N - M$ *local qubits* $|\tilde{n}_L \cdots \tilde{n}_1\rangle = |\tilde{n}_L\rangle_L \otimes \cdots \otimes |\tilde{n}_1\rangle_1$. For a given n, a coefficient a_n exists l-th element of data array in MPI process with rank g, where $g = \tilde{n}_{L+M} \cdots \tilde{n}_{L+1}$ and $l = \tilde{n}_L \cdots \tilde{n}_1$. We need MPI communications if a gate is applied to global qubit(s), then swap the global qubit(s) and most significant local qubit(s). For example, as shown in Fig. 6.5 (Left), if there are only two MPI processes and a single-qubit gate is applied to a global qubit, we swap the second half of data array in the MPI process with rank $r = 0$ and the first half of data array in the MPI process with rank $r = 1$, and update a permutation

$$\sigma \mapsto \begin{pmatrix} N & N-1 & N-2 & \cdots & 1 \\ N-1 & N & N-2 & \cdots & 1 \end{pmatrix} \cdot \sigma. \tag{6.1}$$

After MPI communications, we make gate operations to the swapped local qubit(s).

Quantum gates applied to n qubits can be represented by $2^n \times 2^n$ matrices. There are a few important gates represented by diagonal matrices; *e.g.*, controlled phase-shift gate $R(\theta)\big(a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle\big) = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + \mathrm{e}^{\mathrm{i}\theta} a_{11}|11\rangle$. We have modified operations for those gates in `braket` because we don't have to swap global qubit(s) and local qubit(s) and therefore we can skip MPI communications.

The above method of MPI communication requires the number of MPI processes is a power of two 2^M. However, because the number of computing nodes in most of supercomputers including Fugaku is not a power of two, we cannot use computing resources fully in the method. Moreover, there is unused space in memory because the size of data array is also limited to a power of two 2^L. In order to overcome these limits, we have developed novel method to use $2^M n_\mathrm{u}$ MPI processes, where n_u is an odd number. As shown in Fig. 6.5 (Center), we introduce *unit* that contains n_u MPI processes, and assume we totally have 2^M units in simulation. Each unit has 2^K *data blocks* with size 2^L. In each unit, m processes have $k+1$ data blocks, and $n_\mathrm{u} - m$ processes have k data blocks; that is, $kn_\mathrm{u} + m = 2^K$. We divide physical qubits into M global qubits $|\tilde{n}_N \cdots \tilde{n}_{N-M+1}\rangle = |\tilde{n}_N\rangle_N \otimes \cdots \otimes |\tilde{n}_{N-M+1}\rangle_{N-M+1}$, K *unit qubits* $|\tilde{n}_{L+K} \cdots \tilde{n}_{L+1}\rangle = |\tilde{n}_{L+K}\rangle_{L+K} \otimes \cdots \otimes |\tilde{n}_{L+1}\rangle_{L+1}$, and $L = N - K - M$ local qubits $|\tilde{n}_L \cdots \tilde{n}_1\rangle = |\tilde{n}_L\rangle_L \otimes \cdots \otimes |\tilde{n}_1\rangle_1$. For a given n, a coefficient a_n exists l-th element of u-th data data block in g-th unit, where $g = \tilde{n}_N \cdots \tilde{n}_{N-M+1}$, $u = \tilde{n}_{L+K} \cdots \tilde{n}_{L+1}$, and $l = \tilde{n}_L \cdots \tilde{n}_1$. We need MPI communications if a gate is applied to not only global qubit(s) but also unit qubit(s). For example, if there are only one unit with three MPI processes and a single-qubit gate is applied to a least significant unit qubit, we swap data as shown in Fig. 6.5 (Right). We have implemented this "unit method" for gates represented by non-diagonal matrices in `braket`. Implementing "unit method" for gates represented by diagonal matrices is more complicated, and therefore it will be achieved in the next fisical year.

6.3 Schedule and Future Plan

From the research activities of DESRT so far, following problems are becoming clearer:

1. Simulation models, typically social ones, comprise with large input parameters and output numbers, and their behaviors are strongly nonlinear with various regimes.

2. Social "big data" are often not big enough to picture details. It is clearly observed from our multivariate analysis of the traffic data. Thousands of samples are necessary to get minor traffic factors, but such repetitions are not expected in the real traffic. Weather, economics, calendar, accidents and other factors varies every day.

3. Applications of multiscale social and economic models and simulations.

4. Bridging a gap between physical and social phenomena.

5. Bayesian AI will be useful to cultivate diverse behaviors of social phenomena.

6. Quantum information processing will become in near future.

Aiming for realizing these issues, simulation challenges with the Fugaku are to be pursued collaborating with global researchers, especially in young generation.

6.4 Publications

6.4.1 Articles

[1] Hiroyasu Inoue, Yohsuke Murase, Yasuyuki Todo, “Do economic effects of the anti-COVID-19 lockdowns in different regions interact through supply chains?”, COVID Economics, 56, pp.157-194, 2020.
[2] Hiroyasu Inoue, and Yasuyuki Todo, “The propagation of the economic impact through supply chains: The case of a mega-city lockdown to prevent the spread of COVID-19”, PLOS ONE, 15(9): e0239251, 2020.
[3] Hazem Krichene, Hiroyasu Inoue, Takashi Isogai, and Abhijit Chakraborty, “A model of indirect losses from negatives shocks in production and finance”, PLOS ONE 15(9): e0239293, 2020.
[4] Shunki Takami, Masaki Onishi, Itsuki Noda, Kazunori Iwata, Nobuhiro Ito, Takeshi Uchitane, Yohsuke Murase “Infrastructure in Assessing Disaster-Relief Agents in the RoboCupRescue Simulation.” In: Lee R. (eds) Computational Science/Intelligence and Applied Informatics. CSII 2019. Studies in Computational Intelligence, vol 848. Springer, Cham (2020)
[5] Yohsuke Murase, Seung Ki Baek “Automata representation of successful strategies for social dilemmas” Scientific Reports, 10, 13370 (2020)
[6] Yohsuke Murase, Seung Ki Baek “Five rules for friendly rivalry in direct reciprocity” Scientific Reports, 10, 16904 (2020)
[7] Yohsuke Murase, Seung Ki Baek “Friendly-rivalry solution to the iterated n-person public-goods game” Plos Computational Biology, 17(1): e1008217 (2021)
[8] A. Majumdar, M. Wu, Y. Pan, T. Iitaka, and J. S. Tse, ”Structural dynamics of basaltic melt at mantle conditions with implications for magma oceans and superplumes”, Nature Communications 11, 4815 (2020).

6.4.2 Invited talks

[9] 伊藤伸泰「「富岳」による社会シミュレーション」進化計算シンポジウム2020(2020 年12 月18-20 日、オンライン)
[10] Hiroyasu Inoue and Yasuyuki Todo, “The propagation of the economic impact through supply chains: The case of a mega-city lockdown to contain the spread of Covid-19”, OECD NAEC Lab workshop, OECD, Paris, 2021/2/23.
[11] Hiroyasu Inoue, “The economic effect of the restriction by Japanese government under COVID-19”, The 3rd R-CCS International Symposium, RIKEN, 2021/1/15.
[12] Hiroyasu Inoue, “The propagation of the economic impact through supply chains: The case of a mega-city lockdown against the spread of COVID-19”, Mason Online Pandemic MODeling Forum, 2020/6/5.
[13] Hiroyasu Inoue and Yasuyuki Todo, “The propagation of the economic impact through supply chains: The case of a mega-city lockdown against the spread of COVID-19”, Modelling the Spread and Impact of the Coronavirus, the Graz Schumpeter Centre of the University of Graz, 2020/12/9.
[14] Y. Murase and S.K. Baek “Five rules for friendly rivalry in direct reciprocity” RHINO2020

6.4.3 Oral talks

[15] 伊藤伸泰・吉岡直樹・遠藤彰・黒川真理子「COVID19 感染クラスターの形成とコンタクトトレースによる拡大抑止のシミュレーション解析」日本物理学会2017 年秋季大会（2020 年９月８-11 日、オンライン）11pL2-13
[16] Nobuyasu Ito and Naoki Yoshioka, ”Simulation of quantum computer on the Fugaku computer”, 32nd IUPAP Conference on Computational Physics(CCP2021) (August 2 - 5, 2021, Online)
[17] 伊藤伸泰「パンデミック現象および対策のシミュレーション解析」第10回理研未来戦略室フォーラム(2020 年５月26 日、オンライン）
[18] Nobuyasu Ito, ”Agent-based simulation of infection clusters of the COVID-19 on the Fugaku computer”, 34th Workshop ”Recent Developments in Computer Simulation Studies in Condensed Matter Physics” (February 22-25, 2021, online)
[19] Nobuyasu Ito, “Covid-19 disease and social simulation with the Fugaku supercomputer”, 26th International Symposium on Artificial Life and Robotics(AROB26th 2021) and 6th International Symposium on BioComplexity(ISBC6)
[20] Naohiro Tsuzu, Naoki Yoshioka and Nobuyasu Ito, “An effect of a contact trace application when the

reproduction number is around 1.0”, 26th International Symposium on Artificial Life and Robotics(AROB26th 2021) and 6th International Symposium on BioComplexity(ISBC6) (January 21 - 23, 2021, Online)
[21] Nobuyasu Ito, ”Agent-based simulation of COVID-19 infection cluster”, 2020 RIKEN-NRC HPC Workshop(Online,October 20-21, 2020)
[22] Nobuyasu Ito, ”Social simulation of the COVID-19 disease”, Chesapeake Large-Scale Analytics ”Virtual” Conference (On-line, October 6-7,2020)
[23] Nobuaysu Ito, Naoki Yoshioka, Akira Endo and Mariko Kurokawa, ”Agent-based simulation of COVID19 infection cluster and its deterrence with contact trace”, NETSCI2020(On-line September 21-25,2020)
[24] Nobuyasu Ito, Akira Endo, Mariko Kurokawa and Naoki Yoshioka“Simulation challenge to covid19 spread”, Roles of heterogeneity in non-equilibrium collective dynamics(RHINO 2020)(On-line, August 18-19, 2020)
[25] Yohsuke Murase, Seung Ki Baek, “Five rules for friendly rivalry in iterated Prisoner’s Dilemma” Conference on Complex Systems 2020
[26] J. Kertesz, Y. Murase, J. Torok, H.-H. Jo, K. Kaski, “Selection of communication channels and the related sampling bias for mapping out the network of social interactions” NetSci2020
[27] Naoki Yoshioka, Hajime Inaoka, Nobuyasu Ito, Fengping Jin, Kristel Michielsen, Hans De Raedt, “Simulation of quantum circuits in classical supercomputers”, ADAC Applications Monthly Seminars.
[28] Hiroyasu Inoue, Yohsuke Murase, and Yasuyuki Todo, “The impact of supply-chain networks on interactions between the anti-COVID-19 lockdowns in different regions”, AROB 2021, Oita (Online), 2021/1/21-23.
[29] Hiroyasu Inoue, Yohsuke Murase, and Yasuyuki Todo, “The economic effect of the restriction by Japanese government under COVID-19”, NetSci 2020, Rome (Online), 2020/9/17-25.

6.4.4 Other activities

[30] 伊藤伸泰「スーパーコンピュータ『富岳』によるコロナパンデミック予測」理化学研究所と産業技術総合研究所が共同でお送りする公開講演会「ウィズ・アフターコロナ研究の最前線〜社会変容にどう立ち向かうか？〜」（2020年12月2日、オンライン）
[31] 伊藤伸泰「シミュレーションで探る新型コロナ対策」自然と科学の情報誌ミルシル No.6 (2020年11月) p.6-8
[32] 伊藤伸泰「新型コロナウイルスの感染シミュレーション」理化学研究所一般公開（2020年10月31日、オンライン）
[32] 伊藤伸泰「世界最高速のスーパーコンピュータ「富岳」で人類の夢をかなえる」聖路加国際大学看護学研究科(2021年7月10日）

6.4.5 Software

[33] OACIS (v3.8.0), https://github.com/crest-cassia/oacis
[34] CARAVAN, https://github.com/crest-cassia/caravan
[35] braket, https://github.com/naoki-yoshioka/braket

Chapter 7

Computational Molecular Science Research Team

7.1 Members

Takahito Nakajima (Team Leader)

William Dawson (Research Scientist)

Stanislav Kedžuch (Research Scientist)

Nobuki Inoue (Postdosctoral Researcher)

Takahide Matsuoka (Postdosctoral Researcher)

Eisuke Kawashima (Postdosctoral Researcher)

Kai Guther (Postdosctoral Researcher)

Kimihiko Hirao (Guest Senior Researcher)

Kizashi Yamaguchi (Guest Senior Researcher)

Muneaki Kamiya (Guest Researcher)

Takashi Kawakami (Guest Researcher)

Jong-Won Song (Guest Researcher)

Bun Chan (Guest Researcher)

7.2 Overview of Research Activities

7.2.1 Development of Original Molecular Theory

An atomic- and molecular-level understanding of drug actions and the mechanisms of a variety of chemical reactions will provide insight for developing new drugs and materials. Although a number of diverse experimental methods have been developed, it still remains difficult to investigate the state of complex molecules and to follow chemical reactions in detail. Therefore, a theoretical molecular science that can predict the properties and functions of matter at the atomic and molecular levels by means of molecular theoretical calculations is keenly awaited as a replacement for experiment. Theoretical molecular science has recently made great strides due to progress in molecular theory and computer development. However, it is still unsatisfactory for practical applications. Consequently, our main goal is to realize an updated theoretical molecular science by developing a molecular theory and calculation methods to handle large complex molecules with high precision under a variety of conditions. To achieve our aim, we have so far developed several methods of calculation. Examples include a way for resolving a significant problem facing conventional methods of calculation, in which the calculation volume increases dramatically when dealing with larger molecules; a way for improving the precision of calculations in molecular simulations; and a way for high-precision calculation of the properties of molecules containing heavy atoms such as metal atoms.

7.2.2 Quantum Chemistry Software NTChem and Nekir

Quantum chemistry software comprises immensely useful tools in material and biological science research. Widely diverse programs have been developed in Western countries as Japan has lagged; in fact, only a few programs have been developed in Japan. The mission of our research team is to provide K and the successor Fugaku computer users with a high-performance software for quantum molecular simulation. In the early stage of the K computer project, no quantum chemistry software was available for general purpose and massively parallel computation on the K computer because not every program was designed for use on it. Therefore, we have chosen to develop a new comprehensive *ab initio* quantum chemistry software locally—NTChem. NTChem is completely new software that implements not only standard quantum chemistry approaches, but also original and improved theoretical methods that we have developed in our research work. The main features of the current version, NTChem2013, are the following:

1. Electronic structure calculation of the ground state of atoms and molecules based on Hartree–Fock, HF, and density functional theory, DFT, methods.

2. Linear- or low-scaling DFT: Gaussian and finite-element Coulomb (GFC) resolution-of-the-identity (RI) DFT, pseudospectral DFT/HF, and dual-level DFT.

3. Low-scaling self-consistent field, SCF, calculation using diagonalization-free approaches: purification density matrix, pseudodiagonalization, and quadratic convergence SCF.

4. Excited-state DFT calculation: time-dependent DFT, TDDFT, and transition potential, DFT-TP.

5. Accurate electron correlation methods for ground and excited states: Møller–Plesset perturbation theory, coupled-cluster, CC, theory, and quantum Monte Carlo, QMC.

6. Massively parallel computing on the K computer and Intel-based architectures: HF, DFT, resolution-of-the-identity second-order Møller–Plesset, RI-MP2, and QMC.

7. Two-component relativistic electronic structure calculation with spin-orbit interactions: first-, second- and third-order Douglas–Kroll methods; zeroth- and infinite-order regular approximations; and relativistic scheme for eliminating small components.

8. Model calculations for large molecular systems: quantum mechanics/molecular mechanics, QM/MM, and our own N-layered integrated molecular orbital and molecular mechanics, ONIOM.

9. Calculation of solvation effects: conductor-like screening model, COSMO, which is interfaced with the HONDO program; averaged solvent electrostatic potential/molecular dynamics; and QM/MM-MD.

10. Efficient calculation for chemical reaction pathway by using nudged elastic band, NEB.

11. *Ab initio* molecular dynamics, AIMD, calculation.

12. Calculation of magnetic properties: nuclear magnetic resonance, NMR, chemical shifts, magnetizabilities, and electron paramagnetic resonance, EPR, g tensors.

13. Population analysis: Mulliken and Löwdin, and natural bond orbital (NBO) analyses (interfaced with NBO 6.0).

14. Orbital interaction analysis: maximally interacting orbital and paired interacting orbital methods.

We have also been developing Nekir, an NTChem variant package with enhanced distributed memory parallelization and load balancing. The packages have been ported to the Fugaku supercomputer, and prepared for practical.

7.3 Research Results and Achievements

7.3.1 Quantum Chemistry Softwares NTChem and Nekir

NTChem and Nekir were modified for the Fujitsu compilers on Fugaku supercomputer; users can use the programs by spack package management tool.

Benchmark of NTChem and Nekir was taken on Fugaku: single point energies of layered graphenes were evaluated at HF/cc-pVTZ and LC-wPBE/cc-pVTZ level of theory; a hydrogen-terminated graphene, $C_{150}H_{30}$, has 4920 molecular orbitals. NTChem successfully calculates single-layer graphene, but it runs out of memory for larger systems, whereas Nekir is able to compute four-layered graphene and shows good strong scaling up to 4096 nodes because of its distributed memory parallelization strategy (Figure 7.1).

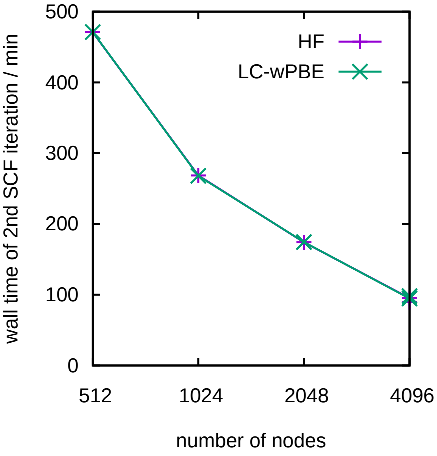


Figure 7.1: Timing of Nekir on Fugaku supercomputer: single point calculation at HF/cc-pVTZ and LC-wPBE/cc-pVTZ level of theory. The system is four-layered graphenes, $C_{600}H_{120}$, with 19680 atomic orbitals.

7.3.2 Flexibilities of wavelets as a computational basis set for large-scale electronic structure calculations

The BigDFT project started in 2005 with the aim of testing the advantages of using a Daubechies wavelet basis set for Kohn-Sham density functional theory with pseudopotentials. This project led to the creation of the BigDFT code, which employs a computational approach with optimal features for flexibility, performance and precision of the results. In particular, the employed formalism has enabled the implementation of an algorithm able to tackle DFT calculations of large systems, up to many thousands of atoms, with a computational effort which scales linearly with the number of atoms. In this work we recall some of the features that have been made possible by the peculiar properties of Daubechies wavelets. In particular, we focus our attention on the usage of DFT for large-scale systems. We show how the localised description of the KS problem, emerging from the features of the basis set, are helpful in providing a simplified description of large-scale electronic structure calculations. We provide some examples on how such simplified description can be employed, and we consider, among the case-studies, the SARS-CoV-2 main protease.

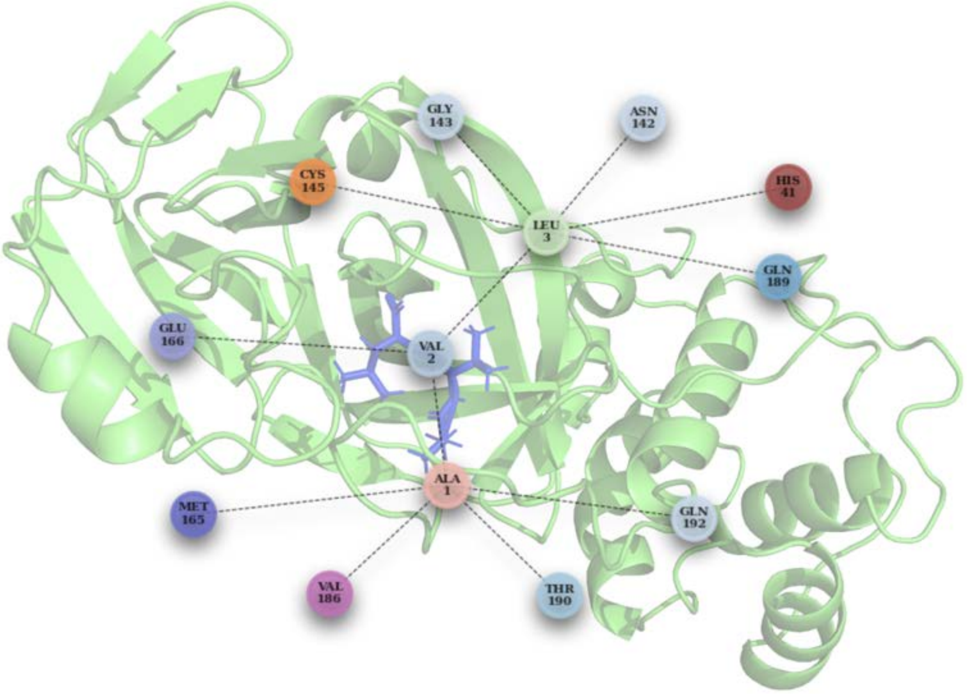


Figure 7.2: an interaction graph of the N3 inhibitor in complex with the main protease of SARS-CoV-2 (PDB: 6LU7). Interactions were computed by post-processing large-scale DFT calculations performed with BigDFT using the Fragment Bond Order tool implemented in the PyBigDFT package.

7.3.3 Multi-Scale Simulation to Predict Biodegradability of Plastics

Biodegradable plastics are attracting attention to reduce environmental load and achieve Sustainable Development Goals, SDGs. Biodegradable aliphatic polyesters—polymers or copolymers of hydroxyalkanoates, or copolymers of diols and dicarboxylic acids—ultimately mineralized into carbon dioxide and water by microbial metabolism. Degree of degradation depends on such environmental conditions as temperature, humidity, oxygen concentration, pH, and microbial communities in compost, soil, or seawater; and on form of a biodegradable plastic product—film, pellet, bar, or cup.

We are developing theoretical tools to estimate biodegradability of plastics to offer design guideline. A kinetic Monte Carlo, KMC, simulator is implemented to estimate time change of polymer bulk under degradation. Polymers were modeled as self-avoiding chains, and a site of a system is occupied by either a polymer segment or water. Elementary processes, i.e., scission and dissolution of polymer, were stochastically simulated: their rates were modeled as step or ramp functions of number of coordinated water and length of the polymer.

Time change of total polymer volume is shown in Figure 7.3. The simulation box is composed of $100 \times 100 \times 6$. Initially polymers with length of 99 were aligned along x-axis, and the polymer bulk occupied the region of $[0, 100) \times [0, 100) \times [0, 6)$; erosion starts from the top surface. The scission and dissolution rates were modeled as step functions: the non-zero values are respectively represented as r_{sci} and r_{sol}, and l_{sol} is the threshold polymer length for dissolution. Degradation is diffusion- or reaction-limited, depending on the relative values of the rates.

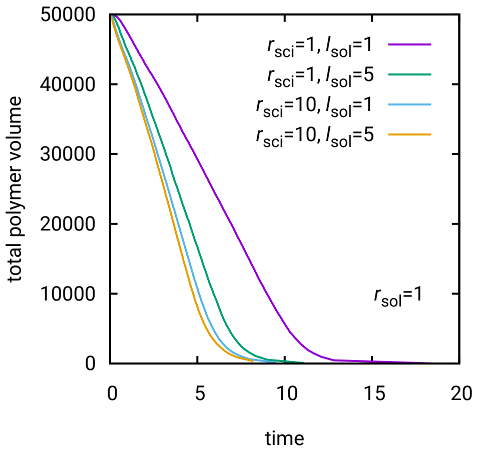


Figure 7.3: Time change of total polymer volume during degradation.

7.3.4 Development of QED-MO method

Relativistic effects are very important for systems containing heavy atoms, and it is necessary to treat them based on the relativistic molecular orbital (MO) methods. However, relativistic quantum mechanics, which is the basic theory of the relativistic MO method, is theoretically incomplete, and has historically evolved into quantum electrodynamics (QED). In other words, it is desirable that the relativistic molecular orbital method be rewritten to a QED-level theory as well, and we have been working to develop a theory that makes this possible.

We had already shown that the frequency-dependent Breit interaction can be derived from the Schrieffer-Wolff transformation of the QED interaction in the interaction representation, and used this to formulate a formalism for the QED-MO method that incorporates the QED effect in the form of an electron-electron interaction.

In this formalism, the QED effect is incorporated as a kind of electron correlation and can be treated as just a two-electron integral.

Therefore, existing electron correlation programs can be used to calculate the QED effect. However, the calculation of the integral of the frequency-dependent Breit interaction, which is necessary to perform the QED-MO method, requires a calculation cost on the order of the sixth power of the basis functions, and improving faster this calculation has been an issue. This kinds of integral can be calculated using Obara-Saika method by changing the ordinary molecular incomplete gamma function (MIGF) to one corresponding to the frequency-dependent Breit interaction. The MIGF is generally a special function defined in integral form, and it is too computationally expensive to evaluate this integral in MO calculations by numerical integration. The MIGF for the usual Coulomb interaction between electrons has single variable T, determined by the distance between the centers of the basis functions and the exponent of the Gaussian function, and can be fast evaluation by keeping the function values on grids, and by Taylor expansion from the grid closest to T. However, the MIGF for the frequency-dependent Breit interaction has two variables, T, and β which is determined from the photon frequency. which is determined by the photon frequency. Here, it is difficult to use the Taylor expansion for the β direction as in the T direction, and the integral of the definition required to obtain the value on the grid is not available when β is large. It is difficult to evaluate numerically the integral of the definition when β is large.

We have developed a new series expansion that converges well for the T direction, and combined it with spline interpolation of the values on the grid on the β-axis created by using an arbitrary-precision library exflib, to enable fast evaluation of the MIGF for frequency-dependent Breit interactions. The asymptotic expansions that we have already developed can be used for large T and large β, respectively. Figure 7.4 (a) and (b) show the values of MIGF estimated by our method for the frequency-dependent Gaunt term and the frequency-dependent gauge term. It is found that the absolute values of both terms decrease in proportion to β^{-2} when β is large. We also found that the usual gaunt and gauge terms always correspond to the case where $\beta = 0$, and thus greatly overestimate the interaction when β is large.

The methods for faster evaluation of the two-electron integral of the frequency-dependent Breit interaction developed in this study will contribute to lowering the computational cost of the QED-MO method and will enable the accurate inclusion of QED effects in molecular systems.

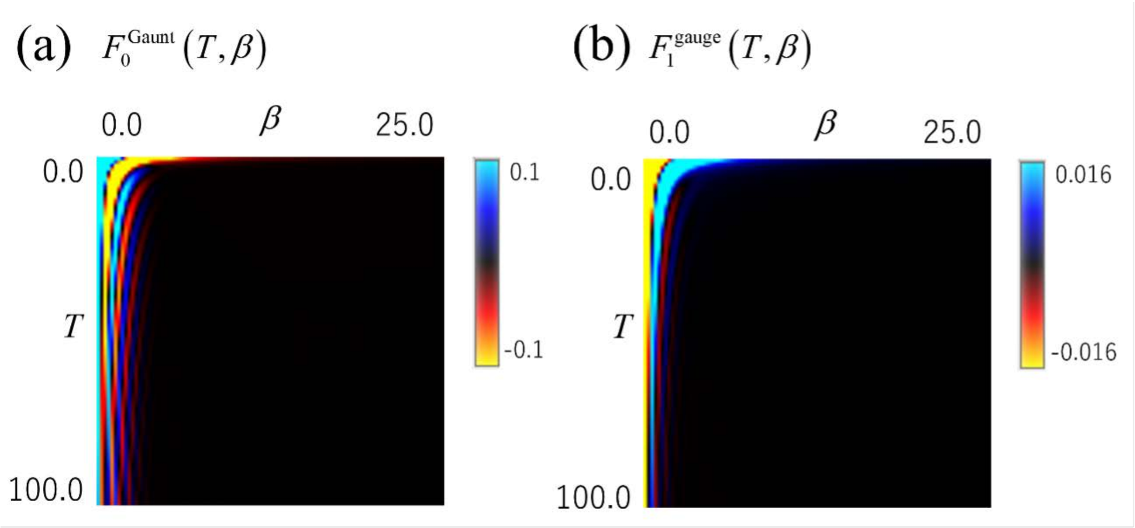
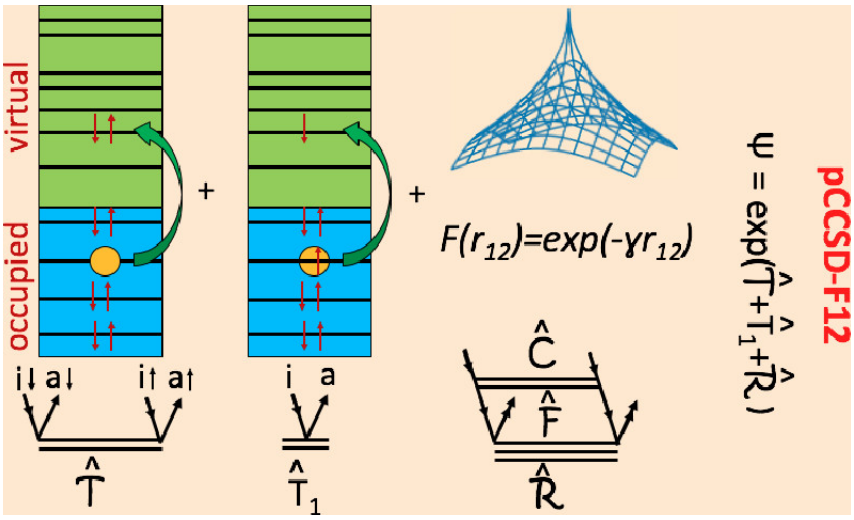


Figure 7.4: The lowest order molecular incomplete gamma functions for the frequency-dependent Breit interaction

7.3.5 Doubly Occupied Pair Coupled Cluster F12 Approach

Inspired by reports of the good performance of the doubly occupied pair coupled cluster (pCCD) theory in describing static electron correlation, we have introduced and implemented a variant thereof that includes single excitations and explicitly treats the dynamic electron correlation using the F12 methodology (pCCSD-F12). This drastically reduces the computation scaling with respect to the standard method using the full double-excitation operator (CCSD-F12). Slater-type geminals as a correlation factor, together with fixed cusp conditions, were used, which is known as the SP-ansatz. For sample model systems, we have investigated the performance of reference states constructed from either canonical or localized molecular orbitals. Finaly, the employment of Brueckner orbitals has been tested, which causes the single excitations to naturally vanish from the wave function expansion (B-pCCD-F12). Our test systems include different-sized rings of hydrogen atoms and dissociation curves for small molecules such as HF, N2, and CO2; and comparison with CCSD-F12 is presented for a series of reaction enthalpies.



7.3.6 Molecular Design for Solar Cell Materials

Spiro-OMeTAD, a hole-transporting material (HTM) for perovskite solar cell, is known for its high power conversion efficiency (PCE). Its PCE for $(FAPbI_3)_{0.92}(MAPbBr_3)_{0.08}$ perovskite is 23.4%, and the estimated cost for synthesizing Spiro-OMeTAD is approximately \$274/g. Although materials with cheaper synthesizing cost and fairly high PCE have been previously reported (X60: \$120/g, Py-C: \$192/g, etc), more cost-effective materials with higher PCEs are yet to be searched.

We trained deep neural network (DNN) model, which predicts PCEs of HTMs with molecular descriptors provided as inputs. Furthermore, we conducted Bayesian optimization by evaluating the acquisition function with Gaussian process regression (GPR). We employed discrete particle swarm optimization (DPSO) method to optimize the vast chemical space.

We constructed learning models to predict the PCEs of candidate HTMs, generated from fragments of known HTMs. For training data, we collected 400 entries of experimental data of HTMs for perovskite solar cells. We employed HTMs (170 molecules), active layers (54 compounds, band gaps, valence band maximum, conduction band minimum), dopants, co-dopants, active area, and PCEs as input. 170 HTM molecules have been decomposed into three fragments, and the candidate HTMs have been generated from these fragments. The molecular descriptors have been calculated for each fragment with Mordred, and the top 10% of the sensitivity analysis was included in explanatory variables. Furthermore, we prepared Coulomb matrix calculated using Mordred's Matrix aggregating method for each fragment and whole molecule, and the quantum descriptors (HOMO, LUMO, total energy, electronic energy, heat of formation, dispersion energy, dipole moment) for each fragment using NTChem. We constructed a PLSR, SVM, and kNN learning model as the stacking base layers using experimental data, molecular descriptors (whole molecules and fragments, quantum descriptor (fragments), and Coulomb matrix (whole and fragments) as explanatory variables. The predicted PCE of the constructed stacking base layers is added to the explanatory variables, and the learning model of the stacking base layers is further constructed. By repeating this, 30 Stacking base layers were prepared. A DNN model is constructed by adding the predicted values of the stacking base layers to the explanatory variables. Candidate molecules were selected by a GPR model under typical experimental conditions. As a virtual experiment, the energy conversion efficiency of the candidate molecule was estimated from the DNN model. By repeating the virtual experiment, the GPR model was improved and the optimal candidate molecules have been selected.

The chemical space approximately spans among 32,294,400 combinations of molecules, making it difficult to comprehensively search this chemical space. We adopt DPSO and explore the huge chemical space. DPSO method was applied with the PCEs given by the model as the target function. When the number of fragments is n and the number of locations of the fragments is N, the coordinates of the particles are defined as $n \times N$ bit matrix. The coordinates were updated probabilistically, and the probability function was evaluated by the softmax function so that one fragment was selected for each location.

Twenty calculations were performed and a total of 5793 molecules were selected as candidates. It was shown that the molecules with a structure having a triarylamine and long alkyl-chain have been predicted to show high PCEs.(Figure 7.5)

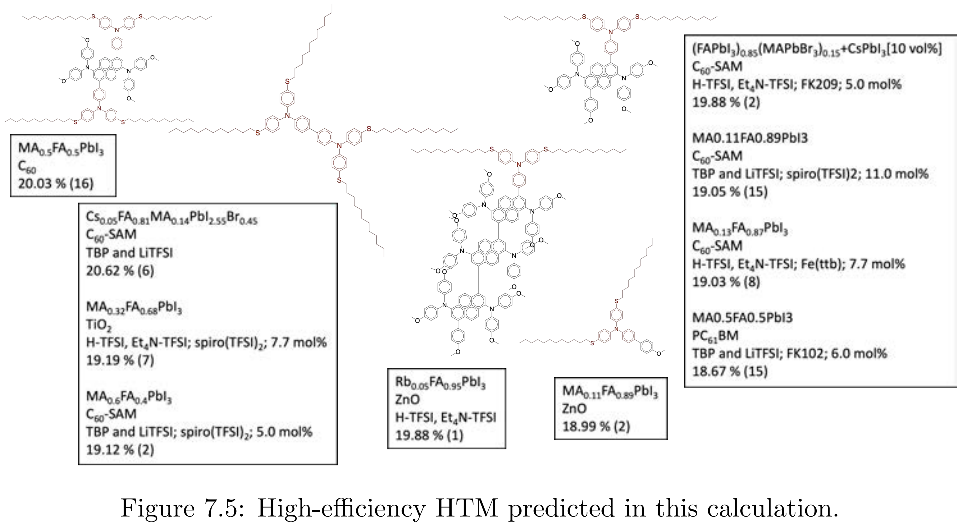

$MA_{0.5}FA_{0.5}PbI_3$
C_{60}
20.03 % (16)
$Cs_{0.05}FA_{0.81}MA_{0.14}PbI_{2.55}Br_{0.45}$
C_{60}-SAM
TBP and LiTFSI
20.62 % (6)
$MA_{0.32}FA_{0.68}PbI_3$
TiO_2
H-TFSI, Et_4N-TFSI; spiro$(TFSI)_2$; 7.7 mol%
19.19 % (7)
$MA_{0.6}FA_{0.4}PbI_3$
C_{60}-SAM
TBP and LiTFSI; spiro$(TFSI)_2$; 5.0 mol%
19.12 % (2)
$Rb_{0.05}FA_{0.95}PbI_3$
ZnO
H-TFSI, Et_4N-TFSI
19.88 % (1)
$MA_{0.11}FA_{0.89}PbI_3$
ZnO
18.99 % (2)
$(FAPbI_3)_{0.85}(MAPbBr_3)_{0.15}$+$CsPbI_3$[10 vol%]
C_{60}-SAM
H-TFSI, Et_4N-TFSI; FK209; 5.0 mol%
19.88 % (2)
MA0.11FA0.89PbI3
C_{60}-SAM
TBP and LiTFSI; spiro(TFSI)2; 11.0 mol%
19.05 % (15)
$MA_{0.13}FA_{0.87}PbI_3$
C_{60}-SAM
H-TFSI, Et_4N-TFSI; Fe(ttb); 7.7 mol%
19.03 % (8)
MA0.5FA0.5PbI3
$PC_{61}BM$
TBP and LiTFSI; FK102; 6.0 mol%
18.67 % (15)

Figure 7.5: High-efficiency HTM predicted in this calculation.

7.3.7 Development of nonadiabatic couplings of TDDFT/TDA

The understanding of the electronic state in the electronically excited state caused by light absorption is very important in the conversion of light energy and photomaterials, such as in plant photosynthesis and solar cells, as well as in the expression process of many optical functions. Reactions in such excited states often involve transitions between excited states through conical intersections between adiabatic excited states, and non-adiabatic couplings are an important computational quantity in describing such transitions.

Time-dependent density functional theory (TDDFT) is becoming a popular methodology for calculating excited states due to its reasonable cost and relatively high accuracy. However, it is difficult to uniquely define the excitation wavefunction of TDDFT because, strictly speaking, TDDFT depends on the electron density rather than the wavefunction.

In this year, we developed and implemented nonadiabatic couplings for obtaining excitation spectra and excitation energies of various molecules.

7.3.7.1 Implementation of nonadiabatic coupling of TDDFT/TDA method by numerical differentiation

In this study, we first implemented a method using numerical differentiation. This method is equivalent to taking the numerical derivative of the wave function at two points, but by modifying the mathematical expression, we can avoid calculating the determinant that is required when the definition is left unchanged, and thus reduce the increase in the computational cost even when the system becomes large [1].

7.3.7.2 Implementation of analytical nonadiabatic coupling of TDDFT/TDA method

Although various implementations of non-adiabatic couplings in the TDDFT method have been proposed, in this study, we implement a nonadiabatic coupling [2] for TDDFT/TDA that corrects for the electron translation factors (ETFs) that break the translational symmetry. This formulation was adopted because translational symmetry is very important when non-adiabatic couplings are used in molecular dynamics such as surface hopping. Similar to the development of the excitation energy gradients in TDDFT, these were derived using the Lagrangian and adapted for memory-separated parallel TDDFT, and the Z-vector calculation, which is necessary to avoid explicit differentiation of the molecular orbital coefficients, was performed using the Liner-Response TDDFT solver. The parallelized program was developed.

[1] H. Song, S. A. Fischer, Y. Zhang, C. J. Cramer, S. Mukamel, N. Govind, and S. Tretia J. Chem. Theory Comput. 16, 6418 (2020)

[2] X. Zhang and J. M. Herbert, J. Chem. Phys. 141, 064104 (2014)

7.3.8 Designing a bioremediator: mechanistic models guide cellular and molecular specialization

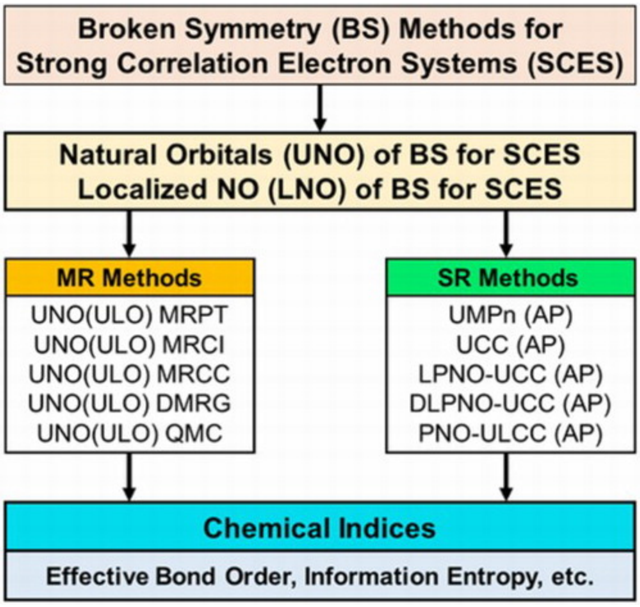
Bioremediators are cells or non-living subcellular entities of biological origin employed to degrade target pollutants. Rational, mechanistic design can substantially improve the performance of bioremediators for applications, including waste treatment and food safety. We highlight how such improvements can be informed at the cellular level by theoretical observations especially in the context of phenotype plasticity, cell signaling, and community assembly. At the molecular level, we suggest enzyme design using techniques such as Small Angle Neutron Scattering and Density Functional Theory. To provide an example of how these techniques could be synergistically combined, we present the case-study of the interaction of the enzyme laccase with the food contaminant aflatoxin B1. In designing bioremediators, we encourage interdisciplinary, mechanistic research to transition from an observation-oriented approach to a principle-based one.

7.3.9 NWChem: Past, present, and future

Specialized computational chemistry packages have permanently reshaped the landscape of chemical and materials science by providing tools to support and guide experimental efforts and for the prediction of atomistic and electronic properties. In this regard, electronic structure packages have played a special role by using first-principle-driven methodologies to model complex chemical and materials processes. Over the past few decades, the rapid development of computing technologies and the tremendous increase in computational power have offered a unique chance to study complex transformations using sophisticated and predictive many-body techniques that describe correlated behavior of electrons in molecular and condensed phase systems at different levels of theory. In enabling these simulations, novel parallel algorithms have been able to take advantage of computational resources to address the polynomial scaling of electronic structure methods. In this paper, we briefly review the NWChem computational chemistry suite, including its history, design principles, parallel tools, current capabilities, outreach, and outlook.

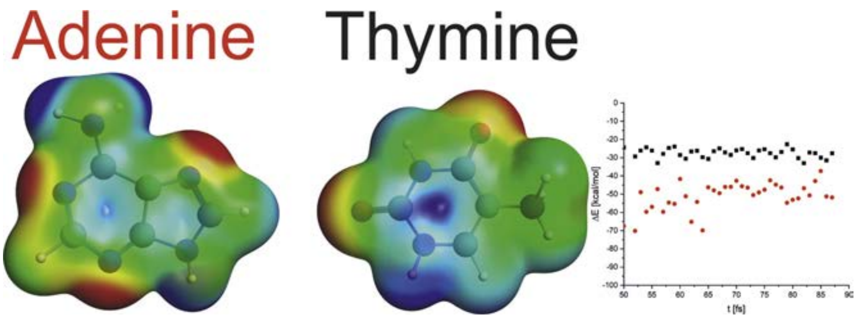
7.3.10 Domain-based local pair natural orbital CCSD(T) calculations of strongly correlated electron systems: Examination of dynamic equilibrium models based on multiple intermediates in S_1 state of photosystem II

Domain-based local pair natural orbital (DLPNO) coupled cluster single and double (CCSD) methods with perturbative triples (T) correction with NormalPNO were used to compute energies for twelve different S_1 structures of the $CaMn_4O_5$ cluster in the oxygen evolving complex (OEC) of photosystem II (PSII). The DLPNO-CCSD(T0) calculations with TightPNO for the important six structures among them revealed that the right (R)-opened S_{1XYZW} structures were more stable than the corresponding left (L)-opened structures ($X=O_{(5)}$, Y=W2, Z=W1, and $W=O_{(4)}$) of $CaMn_4O_5$. The three different S_1 structures belonging to the R-opened type (S_{1acca}, S_{1bbca}, and S_{1abcb}, where O^{2-} = a, OH^- = b and H_2O = c) were found nearly degenerated in energy, indicating the possibility of the coexistence of different structures in the S_1 state. The DLPNO-CCSD(T0) calculations with TightPNO supported the proposal of a dynamic equilibrium model based on the multi-intermediate structures for the S_1 state, which is also in agreement with EPR and other experimental and hybrid DFT computational results. Implications of the computational results are discussed in relation to scope and applicability of NormalPNO and TightPNO for the CCSD(T0) calculations of strongly correlated electron systems such as 3d transition-metal complexes.


Broken Symmetry (BS) Methods for Strong Correlation Electron Systems (SCES)
Natural Orbitals (UNO) of BS for SCES
Localized NO (LNO) of BS for SCES
MR Methods
UNO(ULO) MRPT
UNO(ULO) MRCI
UNO(ULO) MRCC
UNO(ULO) DMRG
UNO(ULO) QMC
SR Methods
UMPn (AP)
UCC (AP)
LPNO-UCC (AP)
DLPNO-UCC (AP)
PNO-ULCC (AP)
Chemical Indices
Effective Bond Order, Information Entropy, etc.

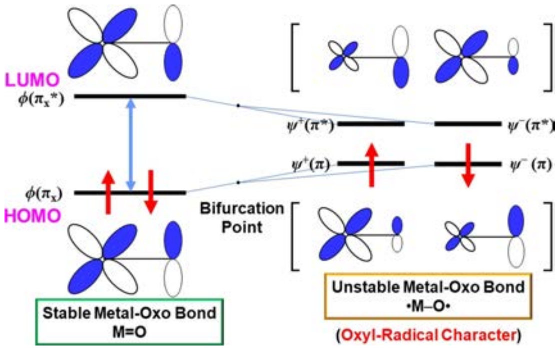
7.3.11 A comparison of the hydrogen bond interaction dynamics in the adenine and thymine crystals: BOMD and spectroscopic study

In this work we present the comparison study of Adenine and Thymine crystals based on the hydrogen bond dynamics. The ab initio molecular dynamics have been used as the base for the further studied interactions observed inside crystals. The generated power spectra, as well as the fluctuation of the interaction energies, showed large differences between hydrogen bond networks in the considered crystals. The analysis of intermolecular interactions have been done base on the reactivity descriptors as well frontiers orbitals along trajectories. The main results showed that in adenine crystals the intermolecular interactions have three directions and fluctuate, while in the thymine crystal have only two directions and are weak but stable. These results explain also on the difference between adenine and thymine melting temperature.



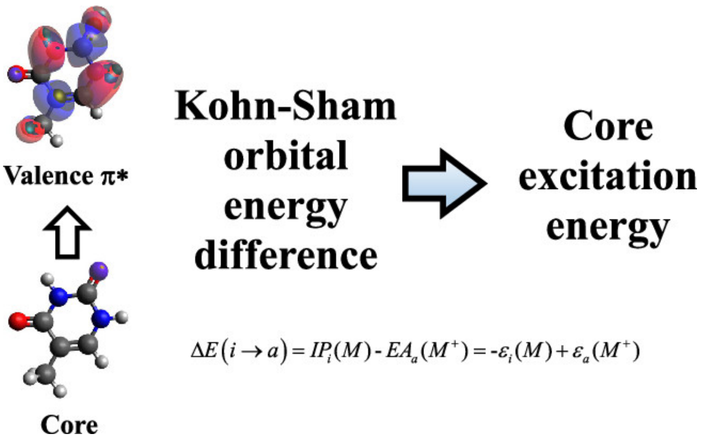
7.3.12 Development of broken-symmetry (BS) methods in chemical reactions. A theoretical view of water oxidation in photosystem II and related systems

Fundamental concepts and basic theories for the broken-symmetry (BS) methods have been reviewed in relation to theoretical elucidation and understanding of the mechanism for water oxidation in the oxygen evolving complex (OEC) of photosystem II (PSII). The HOMO-LUMO mixings by the BS method have provided the BS orbitals which are mainly localized on the metal and oxygen sites of the high-valent transition-metal oxo ($M = O$) bonds, respectively. The oxyl-radical character ($\cdot M - O\cdot$) is responsible for radical reactivity such as the radical coupling in accord with various experimental results. The Lewis acids play important roles for reduction of the oxyl-radical character, indicating the participation of the water-coordinated Ca(II) ion of the $CaMn_3O_4$ cubane to stabilization of the $Mn(V) = O \cdots Ca(II)$ bond for essentially non-radical reactions. Several chemical indices have been calculated to elucidate the radical character for quantitative purpose. Implications of the computational results are discussed in relation to possible mechanisms of water oxidation in OEC of PSII.



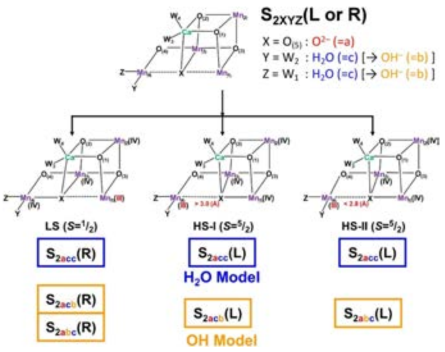
7.3.13 Core-Level Excitation Energies of Nucleic Acid Bases Expressed as Orbital Energies of the Kohn–Sham Density Functional Theory with Long-Range Corrected Functionals

The core electron binding energies (CEBEs) and core-level excitation energies of thymine, adenine, cytosine, and uracil are studied by the Kohn–Sham (KS) method with long-range corrected (LC) functionals. The CEBEs are estimated according to the Koopmans-type theorem for density functional theory. The excitation energies from the core to the valence π^* and Rydberg states are calculated as the orbital energy differences between core-level orbitals of a neutral parent/cation and unoccupied π^* or Rydberg orbitals of its cation. The model is intuitive, and the spectra can easily be assigned. Core excitation energies from oxygen 1s, nitrogen 1s, and carbon 1s to π^* and Rydberg states, and the chemical shifts, agree well with previously reported theoretical and experimental data. The straightforward use of KS orbitals in this scheme carries the advantage that it can be applied efficiently to large systems such as biomolecules and nanomaterials.


Valence π*
Kohn-Sham orbital energy difference
Core excitation energy
$\Delta E(i \rightarrow a) = IP_i(M) - EA_a(M^+) = -\varepsilon_i(M) + \varepsilon_a(M^+)$
Core

7.3.14 Relative stability among intermediate structures in S_2 state of $CaMn_4O_5$ cluster in PSII by using hybrid-DFT and DLPNO-CC methods and evaluation of magnetic interactions between Mn ions

Full geometry optimization of all S_2 intermediates in OEC (oxygen-evolving complex) of PSII (photosystem II) were carried out by using UB3LYP-D3/Def2-TZVP with COSMO solvation effects. Our detail calculations yielded meta-stable structures of six (=3×2 (HS (high-spin), IS (intermediate-spin)) intermediates for $W_1 = W_2 = H_2O$ and eight (=4×2 (HS, IS)) intermediates for $W_1 = OH^-$ or $W_2 = OH^-$, and were named to H_2O and OH models, respectively. In the next step, relative stability among these intermediate structures were investigated by hybrid-DFT and DLPNO-CC methods. UB3LYP methods show that right (R)-opened structures (open-cubane) are more stable than left (L)-opened structures (closed-cubane) by about 3.5kcal/mol, though decreasing of DFT-weights suppresses such energy gaps. DLPNO-CCSD(T0) methods promote stabilization of (L)-structures and finally reproduce near degeneration or more stable (L)-structure. All pattern of spin configurations in four Mn(III) and Mn(IV) ions were assumed and BS (broken-symmetry) solutions were successfully obtained to find the most stable spin structures. This complete sets of all spin conformations enabled us evaluate sets of effective exchange integrals J as magnetic coupling parameters. The calculated J values for the spin Hamiltonian elucidated one $g2$ ($S = 1/2$) and two $g4$ ($S = 5/2$) molecular structures in the S_2 state in accord with the recent EXAFS results.



7.3.15 Attenuation of Redox Switching and Rectification in Azulenequinones/Hydroquinones after B and N Doping: A First-Principles Investigation

The redox switching of doped 1,5-azulenequinones/hydroquinones wired between gold electrodes is investigated using density functional theory and the nonequilibrium Green's function. Their electronic transport properties when separately doped with nitrogen and boron as well as co-doping of these atoms are examined. The results illustrate a significant enhancement of the current at low bias voltage in some of the 12 doped studied systems, leading to "switching on" the transmission, where the greatest switching ratio is 18. These systems also exhibit a modest rectification in which the greatest rectification ratio is 4. The significance of the position of the doped atom and the functional group on the switching behavior is analyzed through the transmission spectra and molecular orbitals. The present study broadens knowledge of organic redox switching bringing in potential diverse options for future molecular electronic circuit components.

7.4 Schedule and Future Plan

Profiling shows that the bottleneck of NTChem and Nekir on Fugaku is subroutines that calculate two-electron integrals. We plan to take further benchmarks and profiling to inspect bottlenecks in detail and improve performance.

7.5 Publications

7.5.1 Articles/Journal

[1] M. Zaccaria, W. Dawson, V. Cristiglio, M. Reverberi, L. E. Ratcliff, T. Nakajima, L. Genovese, and B. Momeni Curr. Opin. Biotechnol., 62, 98–105 (2020).

[2] E. Aprà, E. J. Bylaska, W. A. de Jong, N. Govind, K. Kowalski, T. P. Straatsma, M. Valiev, H. J. J. van Dam, Y. Alexeev, J. Anchell, V. Anisimov, F. W. Aquino, R. Atta-Fynn, J. Autschbach, N. P. Bauman, J. C. Becca, D. E. Bernholdt, K. Bhaskaran-Nair, S. Bogatko, P. Borowski, J. Boschen, J. Brabec, A. Bruner, E. Cauët, Y. Chen, G. N. Chuev, C. J. Cramer, J. Daily, M. J. O. Deegan, T. H. Dunning Jr., M. Dupuis, K. G. Dyall, G. I. Fann, S. A. Fischer, A. Fonari, H. Früchtl, L. Gagliardi, J. Garza, N. Gawande, S. Ghosh, K. Glaesemann, A. W. Götz, J. Hammond, V. Helms, E. D. Hermes, K. Hirao, S. Hirata, M. Jacquelin, L. Jensen, B. G. Johnson, H. Jónsson, R. A. Kendall, M. Klemm, R. Kobayashi, V. Konkov, S. Krishnamoorthy, M. Krishnan, Z. Lin, R. D. Lins, R. J. Littlefield, A. J. Logsdail, K. Lopata, W. Ma, A. V. Marenich, J. Martin del Campo, D. Mejia-Rodriguez, J. E. Moore, J. M. Mullin, T. Nakajima, D. R. Nascimento, J. A. Nichols, P. J. Nichols, J. Nieplocha, A. Otero-de-la-Roza, B. Palmer, A. Panyala, T. Pirojsirikul, B. Peng, R. Peverati, J. Pittner, L. Pollack, R. M. Richard, P. Sadayappan, G. C. Schatz, W. A. Shelton, D. W. Silverstein, D. M. A. Smith, T. A. Soares, D. Song, M. Swart, H. L. Taylor, G. S. Thomas, V. Tipparaju, D. G. Truhlar, K. Tsemekhman, T. Van Voorhis, Á. Vázquez-Mayagoitia, P. Verma, O. Villa, A. Vishnu, K. D. Vogiatzis, D. Wang, J. H. Weare, M. J. Williamson, T. L. Windus, K. Woliński, A. T. Wong, Q. Wu, C. Yang, Q. Yu, M. Zacharias, Z. Zhang, Y. Zhao, and R. J. Harrison J. Chem. Phys., 152, 184102 (2020).

[3] L. E. Ratcliff, W. Dawson, G. Fisicaro, D. Caliste, S. Mohr, A. Degomme, B. Videau, V. Cristiglio, M. Stella, M. D'Alessandro, S. Goedecker, T. Nakajima, T. Deutsch, and L. Genovese J. Chem. Phys., 152, 194110 (2020).

[4] K. Miyagawa, T. Kawakami, Y. Suzuki, H. Isobe, M. Shoji, S. Yamanaka, M. Okumura, T. Nakajima, and K. Yamaguchi Mol. Phys., 118, e1666171 (2020).

[5] M. Z. Brela, O. Klimas, M. Boczar, T. Nakajima, and M. J. Wójcik Spectrochim. Acta A Mol. Biomol. Spectrosc., 237, 118398 (2020).

[6] K. Yamaguchi, H. Isobe, M. Shoji, K. Miyagawa, S. Yamanaka, T. Kawakami, and T. Nakajima J. Photochem. Photobiol. A Chem., 402, 112791 (2020).

[7] S. Kedžuch, J. Šimunek, M. Veis, and J. Noga J. Chem. Theory Comput., 16, 12, 7372–7380 (2020).

[8] K. Hirao, T. Nakajima, B. Chan, J.-W. Song, and H.-S. Bae J. Phys. Chem. A, 124, 10482–10494 (2020).

[9] K. Miyagawa, T. Kawakami, Y. Suzuki, H. Isobe, M. Shoji, S. Yamanaka, M. Okumura, T. Nakajima, and K. Yamaguchi J. Photochem. Photobiol. A Chem., 405, 112923 (2021).

[10] E.-A. Haidar, S. A. Tawfik, C. Stampfl, K. Hirao, K. Yoshizawa, T. Nakajima, K. A. Soliman, and A. M. El-Nahas Adv. Theory Simul., 4, 2000203 (2021).

7.5.2 Invited Talks

[1] W. Dawson, "Complexity Reduction: how BigDFT basis set provides insights on electronic structure calculations of macromolecular systems", Max Webinars, November 12th 2020.

Chapter 8

Computational Materials Science Research Team

8.1 Members

Seiji Yunoki (Team Leader)

Yuichi Otsuka (Research Scientist)

Shigetoshi Sota (Research Scientist)

Tomonori Shirakawa (Research Scientist)

Hiroshi Ueda (Research Scientist)

Sandro Sorella (Guest Researcher)

Takami Toyama (Guest Researcher)

Nauta Takemori (Guest Researcher)

Masako Hirata (Assistant)

Keiko Matsuoka (Assistant)

8.2 Overview of Research Activities

Strongly correlated quantum materials show great promise for next-generation electronic applications. In order to accelerate the development of functional strongly correlated quantum materials, a reliable theory with good predictability is required. However, the strong interactions that take place in this class of materials do not allow us to apply the traditional band theory based on the density functional theory, which played a major role in the advance of today's electronic technology based on semiconductors.

Consequently, we are developing large-scale numerical simulations for strongly correlated quantum systems, including strongly correlated quantum materials, where the many-body interactions are essential to induce novel phenomena and properties. We are interested particularly in the quantum Monte Carlo (QMC) method, the density matrix renormalization group (DMRG) method, and the tensor network method to simulate not only the ground state but also the dynamics (thermodynamics, excitation dynamics, and real time dynamics). We have established a platform for advanced research of strongly correlated quantum systems by developing state-of-the-art simulations.

8.3 Research Results and Achievements

8.3.1 Large-scale QMC simulations for interacting fermions

We develop a QMC method, which is one of the most reliable and efficient techniques for Hubbard-type lattice models of interacting electrons. Typical target systems we aim are of the order of 10,000 electrons unless the notorious minus-sign problem occurs. One of the main focuses in our QMC project is to clarify quantum criticality of quantum phase transitions in strongly-correlated electrons with high accuracy, which would be impossible without the power of supercomputers.

We have implemented a highly efficient QMC code based on the auxiliary field scheme for lattice fermion systems at zero temperature. Since numerical calculations involved in this formulation are mostly linear algebraic procedures such as matrix-matrix products and numerical orthogonalizations, we can take advantage of the highly optimized numerical library on supercomputers such as supercomputer Fugaku to calculate physical observables with a high degree of accuracy on unprecedentedly large systems.

In this fiscal year, we have investigated a strongly-correlated topological state of matter: The higher-order topological Mott insulator (HOTMI). This recently found phase originates from nontrivial topological properties and correlation effects, the two major subjects in modern condensed matter physics, and therefore has been attracting increasing interest. We examined if and how the HOTMI emerges in three spatial dimensions. The Hamiltonian we have studied reads

$$\mathcal{H} = \mathcal{H}_t^{\triangle} + \mathcal{H}_t^{\bigtriangledown} + \mathcal{H}_U, \tag{8.1}$$

with

$$\mathcal{H}_t^{\gamma} = -t_{\gamma} \sum_{i,j\in\gamma} \sum_{\alpha,\beta=\uparrow,\downarrow} \left(c_{i\alpha}^{\dagger} \sigma_{\alpha\beta}^{z} c_{j\beta} + \mathrm{h.c.} \right) \tag{8.2}$$

and

$$\mathcal{H}_U = U \sum_{i} \left(n_{i\uparrow} - \frac{1}{2} \right) \left(n_{i\downarrow} - \frac{1}{2} \right), \tag{8.3}$$

where $c_{i\alpha}^{\dagger}$ is a creation operator of an electron with spin α $(=\uparrow,\downarrow)$ at site i, $n_{i\alpha} = c_{i\alpha}^{\dagger} c_{i\alpha}$ is a number operator, and $\sigma_{\alpha\beta}^{z}$ is the z-component of Pauli matrix. In the kinetic part $\mathcal{H}_t^{\gamma}$ with $\gamma = \triangle$ and $\bigtriangledown$, $t_{\triangle}$ and $t_{\bigtriangledown}$ denotes the transfer integrals for the intra and inter unit cell, respectively [see Fig. 8.1(a)]. Their relative ratio is parameterized by ϕ with $0 \le \phi \le 1/2$ as $t_{\triangle} = t \sin(\phi\pi)$ and $t_{\bigtriangledown} = t\cos(\phi\pi)$. The Hubbard term of Eq. (8.3) represents the repulsive (> 0) on-site interaction.

Thanks to our highly-developed QMC code, we have performed large scale simulations for the model of Eq. (8.1) on the three dimensional pyrochlore lattice [Fig. 8.1(a)] with size up to 500 sites at finite temperatures. As shown in Figs. 8.2(a) and (b), the edge (boundary) states are gapped in the charged sector, while the spin-only excitations are observed. To examine these gapless modes, we calculate "site-resolved" charge compressibility and spin susceptibility defined as $\kappa_{\mathrm{c}}(i) = \frac{\partial\langle n_{i\uparrow}+n_{i\downarrow}\rangle}{\partial\mu}$ and $\kappa_{\mathrm{s}}(i) = \frac{\partial\langle n_{i\uparrow}-n_{i\downarrow}\rangle}{\partial h}$ with μ (h) being the chemical potential (magnetic field). As shown in Fig. 8.2(c), $\kappa_{\mathrm{c}}(i)$ for $U = 0$ exhibits peaks at four site locations that are the isolated corners in the decoupled limit ($\phi = 0$). This is the expected behavior of the third-order topological insulator in three dimensions. When the interaction U is included, the peaks in $\kappa_{\mathrm{c}}(i)$ immediately disappear; on the other hand, the peaks in $\kappa_{\mathrm{s}}(i)$ remain and even develop for $U > 0$. This clearly shows that the gapless spin excitations appear around the $(d-3)$-dimensional boundary, namely the corners, which can also be observed from the enhancement of the local magnetic moments as shown in Fig. **??**(a). We also investigate the fate of the HOTMI by changing ϕ at the fixed value of $U = 3$. The temperature-dependnece of the charge compressibility [Fig. 8.3(a)] and the spin susceptibility [Fig. 8.3(b)] for the system under the periodic boundary conditons show that, in the bulk, only the spin gap closes at a critical point of $\phi_{\mathrm{c}} \simeq 0.16$. This is ascribed to the phase transition from the HOTMI to the usual Mott insulator. Indeed, the peak structure in the site-resolved spin susceptibility, characteristic to the HOTMI, seems to vanish at the same point as shown in fig. 8.3(c).

8.3.2 Massively parallel DMRG algorithms for quantum many-body systems

The DMRG method is recognized as one of the most efficient numerical methods to investigate quantum many-body systems. We are developing four types of massively parallel DMRG programs, 2D-DMRG, DDMRG, paraDMRG, and QUARTZ. The 2D-DMRG is developed for the ground state calculations on two-dimensional strongly correlated quantum systems by introducing long-range interactions that corresponds to our desired lattice shape of the traditional one-dimensional DMRG algorithm. The DDMRG is developed for quantum dynamics of strongly correlated quantum systems, which uses the kernel polynomial method to calculate excited

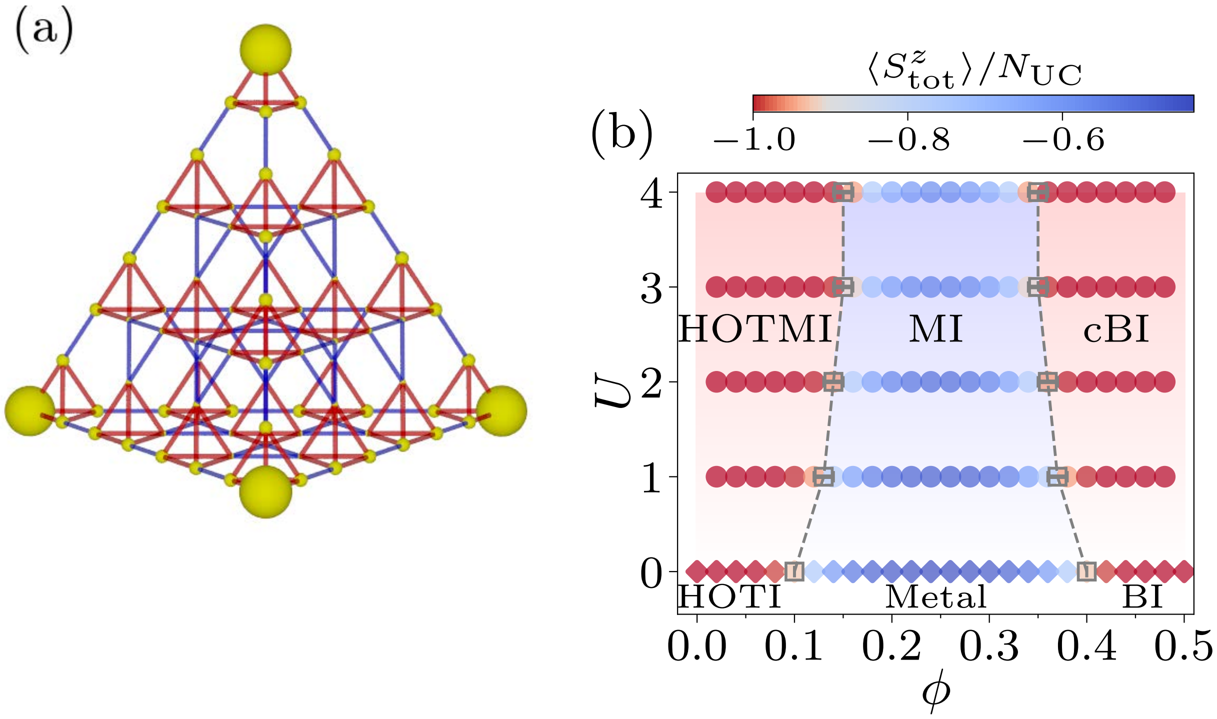


Figure 8.1: (a) Pyrochlore lattice for $L = 4$ with the open boundary conditions. The radius of the yellow spheres shows to what extent the local moment is enhanced by the the Coulomb interaction for $U = 1$ and $\phi = 0.08$ at $T = 0.08$. (b) Ground-state phase diagram.

states and time-evolving states. The paraDMRG is developed for the full-CI calculations in *ab initio* simulations. The QUARTZ is developed to simulate quantum computers. Our developed massively parallel DMRG algorithm has achieved the extremely high peak performance ratio of 73.6% when 82,488 nodes are used on K computer, which is about 7.8 PFLOPS. We are now optimizing our developed DMRG programs for supercomputer Fugaku.

In this fiscal year, we have developed a new method "infinite dynamical DMRG for two-dimensional systems". In this method, we introduce a ladder system corresponding to our desired two-dimensional system. Then we expand the ladder system by the infinite DMRG algorithm as in the one-dimensional system. Here, we employ not only the ground state but also the excited states as the target states of the DMRG, i.e., the multi-target procedure of the usual dynamical DMRG method. Our developed infinite dynamical DMRG algorithm for two-dimensional systems does not require the finite DMRG algorithm which is usually necessary to converge the DMRG calculations. For the application of the DMRG calculation to high-dimensional systems, the finite DMRG algorithm is quite important to introduce long-range interactions corresponding to a desired lattice shape, and consumes most of the computational time. In our developed DMRG algorithm, the DMRG calculation can be performed by introducing only nearest neighbor interactions as in the one-dimensional system. Thus, our developed infinite dynamical DMRG method is able to give accurate results without the finite DMRG algorithm, similar to the one-dimensional cases (Fig. 8.4). Our developed method can reduce the computation time comparing with the previous dynamical DMRG method for two-dimensional systems. Thus, we can simulate larger two-dimensional systems that are difficult to be treated by the previous dynamical DMRG method. In this new method, we have to treat large degrees of freedom, since the number of added sites increases because of introducing a ladder system. The degree of freedom increases exponentially with the number of added sites. The degree of freedom for these added sites can be easily parallelized, since the operators of added sites are given by sparse matrices. Therefore, our developed infinite dynamical DMRG method are suitable for large scale simulations.

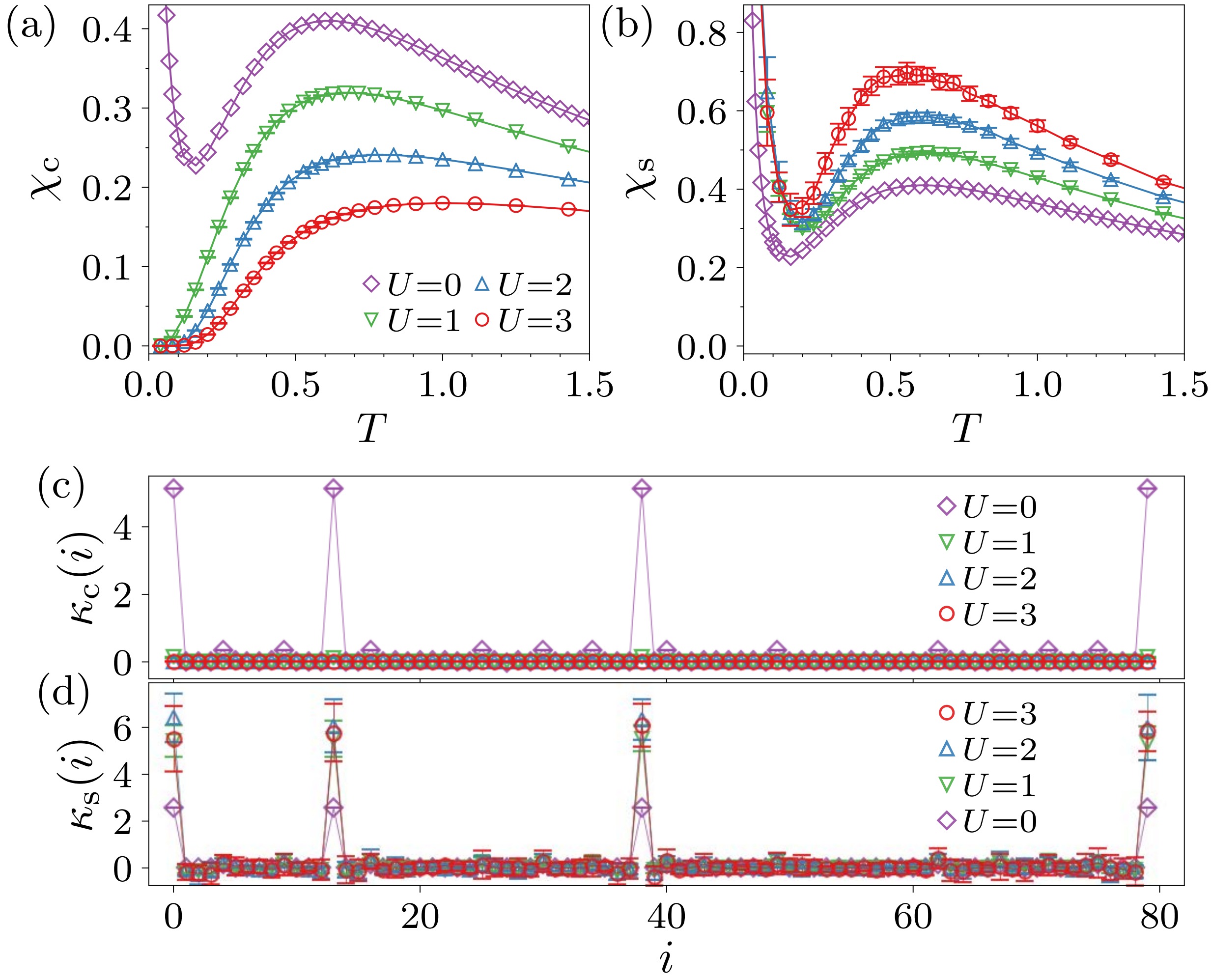


Figure 8.2: Temperature dependence of (a) charge compressibility χ_c and (b) spin susceptibility χ_s at $\phi = 0.08$ for $L = 5$ under the OBC. Site-resolved (c) charge compressibility $\kappa_c(i)$ and (d) spin susceptibility $\kappa_s(i)$ for the system of $L = 4$ under the OBC at $\phi = 0.08$ and $T = 0.08$.

8.3.3 Unitary dynamics of quantum many-body systems

The numerical simulation for unitary dynamics of quantum many-body systems is one of the important subjects to bridge the experimental observation and the theoretical prediction. Moreover, those are also promising fields to find new phenomena which can be controlled artificially. Indeed, we have found a new phenomenon in the last fiscal year: the photo-induced η pairings. In addition, one of the applications by using controlled dynamics includes quantum computing, which has been attracted much attention not only in fundamental scientific fields but also in industrial fields.

In the last fiscal year, we have developed a numerical technique to perform the numerically exact simulation in the complex systems composed of the quantum spins and fermion bath system [F. Lange, S. Ejima, T. Shirakawa, S. Yunoki, and H. Fehske, Journal of the Physical Society of Japan **89**, 044601/1-6 (2020)] by extending the method originally proposed by two of our members [T. Shirakawa and S. Yunoki, Physical Review B **90**, 195109 (2014)]. In this fiscal year, we applied this method to study the spin current dynamics in the spin systems coupled to fermions with Rashba spin-orbit coupling, which finds important applications in the field of spintronics. We then found that injection of the spin current to the Rashba system can induce the current vortex, as shown in Figure 8.5. In the continuum limit, this new phenomenon found by our numerical simulation can be understood as a conversion from the spin-angular momentum to the orbital-angular momentum due to the conservation law of the total angular momentum in the direction perpendicular to the Rashba surface.

Motivated by recent progress of quantum technologies making small-scale programmable quantum computing possible, in this fiscal year, we have also studied a discretized quantum adiabatic process for a one-dimensional

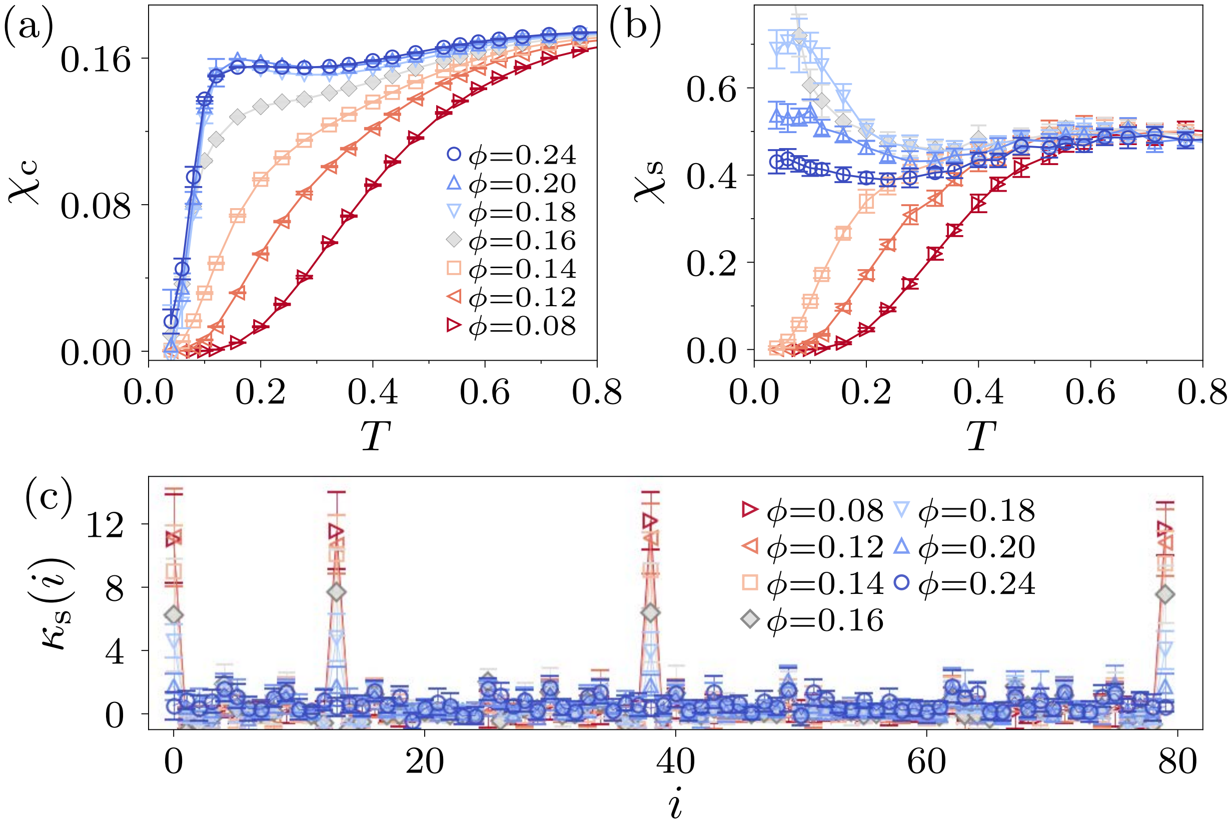


Figure 8.3: Temperature dependence of (a) charge compressibility χ_c and (b) spin susceptibility χ_s for the system of $L = 5$ under the PBC at $U = 3$. (c) Site-resolved spin susceptibility $\kappa_s(i)$ for $L = 4$ under the OBC at $U = 3$ and $T = 0.08$.

free-fermion system described by a variational wave function, i.e., a parametrized quantum circuit. In this circuit ansatz, the wave function comprises M layers of two elementary sets of time-evolution operators, each set being decomposed into commutable local operators acting on neighboring sites. We then found that the exact ground state is reached by applying the layers of time-evolution operators as many as a quarter of the system size, implying that at least in this case, the state is exactly prepared in a quantum circuit with linear depth. This is the minimum number M_B of layers set by the limit of speed, i.e., the Lieb-Robinson bound, for propagating quantum entanglement via the local time-evolution operators.

However, parameterized quantum circuit's wave functions have not been accurate enough for the strongly correlated quantum systems in general, preventing the application to the real problem. To improve the accuracy, we have introduced a general framework for symmetry-projection in quantum computing. Consequently, we found drastic improvements in the ground state energies. In addition, we also showed that the introduction of the symmetry-projection is helpful to obtain the low-lying excited state.

8.4 Schedule and Future Plan

8.4.1 Large-scale QMC simulations for interacting fermions

The graphene is a very good conductor, which is of benefit in many applications. On the other hand, it is also highly desired to introduce a gap in the graphene from the view point of the theory and applications. In the two-dimensional Dirac electrons, which is the basic model of the graphene, it has been theoretically established that the strong on-site Coulomb interactions give rise to the Mott transition inducing the charge gap. Consequently, extensive experimental efforts have been devoted to realize the Mott insulator by stretching the graphene which controls the effective strength of the Coulomb interactions. Recently, the first-principles quantum Monte Carlo calculations for these experiments have revealed that there exists a Peierls insulating

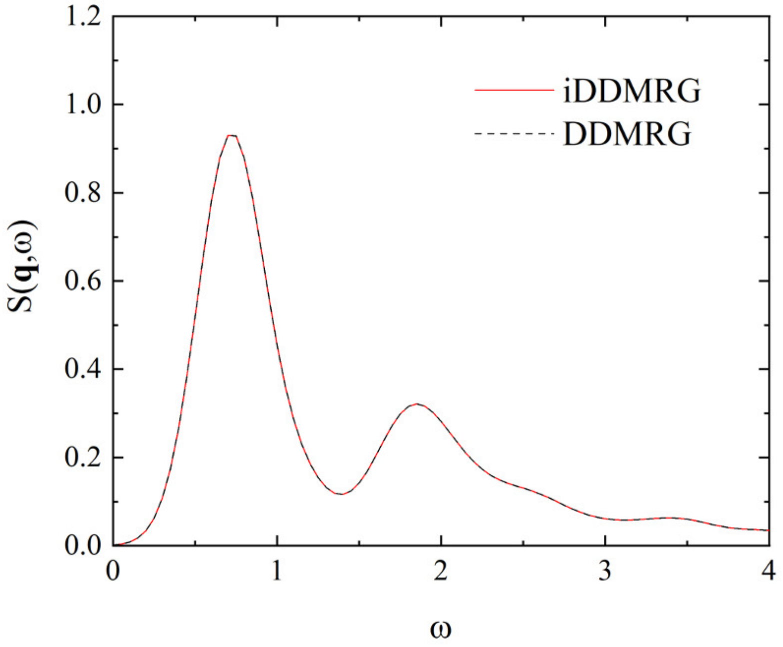


Figure 8.4: The dynamical DMRG results of the dynamical spin correlation function of the spin $S = 1/2$ antiferromagnetic Heisenberg model on 4×6 trianglar lattice. The Red solid line shows the result by our developed infinite dynamical DMRG method. The black dashed line shows the results of the previous dynamical DMRG method introducing long-range interactions.

states with lattice deformation in the vicinity of the Mott insulator. We plan to investigate this possible Peierls phase on the basis of the Hubbard model on the honeycomb lattice taking into account electron-lattice couplings by the auxiliary field quantum Monte Carlo method, which can more exactly study the correlation effects than the first-principles quantum Monte Carlo calculations. We also plan to explore the possibilities of the valence bond solid (VBS) phase and the resonating valence bond (RVB) state in this model.

8.4.2 Massively parallel DMRG algorithms for quantum many-body systems

We will continue to optimize our massively parallel DMRG programs and perform calculations effectively and efficiently on supercomputer Fugaku. In addition, we will develop our DMRG programs to perform calculations which were difficult to perform on K computer. One of the promising applications using supercomputer Fugaku is finite temperature DMRG calculations for high-dimensional systems. We have already developed finite DMRG method using the kernel polynomial method, which can be employ for higher-dimensional systems. Although the finite DMRG method for higher-dimensional systems requires a huge amount of computational costs, it is expected to be practical by using the computational power of supercomputer Fugaku. Also, we will expand the application of our developed DMRG programs for not only condensed matter physics but also the other research field such as quantum information. Our DMRG programs can simulation quantum computing with having more qubits than the exact simulation (i.e., the state vector method) cannot treat and evaluate any physical quantities and information of quantum states. We expect that these simulations should be useful for the development of quantum devices as well as quantum algorithms.

8.4.3 Quantum computing for quantum many-body systems

As demonstrated in the research results and achievements in this fiscal year, the numerical simulation for quantum dynamics of quantum many-body systems are promising fields not only to find new phenomenona which can be controlled artificially but also for the application to the quantum technologies which are raplidly developing in these years. Motivated by these research activities, in the next fiscal year, we plan to investigate the

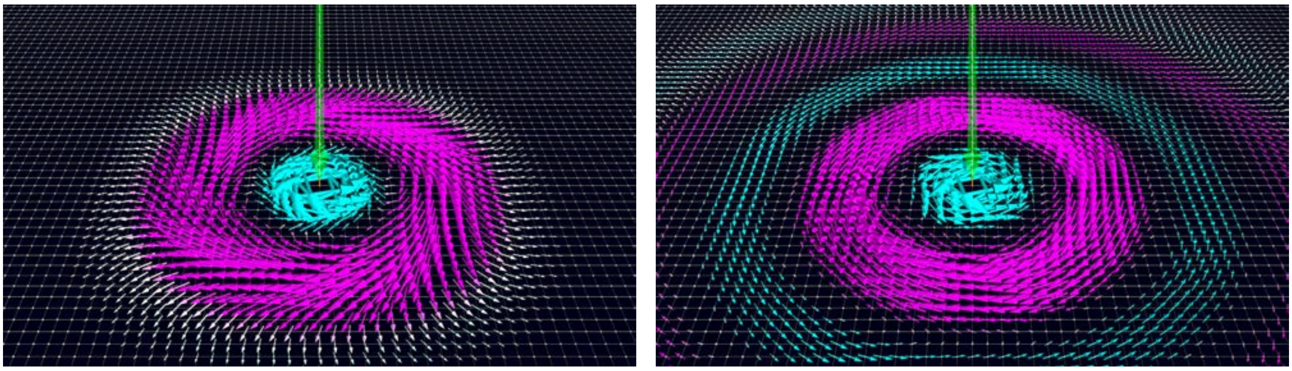


Figure 8.5: Numerical results of current vortex induced by injection of spin current (green arrow). Magenta and cyan arrows indicate the clockwise and counter-clockwise components of local current, respectively.

quantum-classical hybrid algorithm for the quantum many-body systems based on the simulation techniques of the quantum computating developed in our team. Especially, we are interested in the way to design the appropriate circuit depending on the problem by using the idea for the tensor network methods.

8.5 Publications

8.5.1 Articles/Journal

[1] M. Fujihala, K. Morita, R. Mole, S. Mitsuda, T. Tohyama, S. Yano, D. Yu, S. Sota, T. Kuwai, A. Koda, H. Okabe, H. Lee, S. Itoh, T. Hawai, T. Masuda, H. Sagayama, A. Matsuo, K. Kindo, S. Ohira-Kawamura, and K. Nakajima, "Gapless spin liquid in a square-kagome lattice antiferromagnet", Nature Communications **11**, 1–7 (2020).

[2] N. C. Costa, K. Seki, S. Yunoki, and S. Sorella, "Phase diagram of the two-dimensional Hubbard-Holstein model", Communication Physics **3**, 80/1–6 (2020).

[3] T. Kaneko, S. Yunoki, and A. J. Millis, "Charge stiffness and long-range correlation in the optically induced η-pairing state of the one-dimensional Hubbard model", Physical Review Research **2**, 032027(R)/1–5 (2020).

[4] K. Seki and S. Yunoki, "Emergence of a thermal equilibrium in a subsystem of a pure ground state by quantum entanglement", Physical Review Research **2**, 043087/1–19 (2020).

[5] K. Seki, T. Shirakawa, and S. Yunoki, "Symmetry-adapted variational quantum eigensolver", Physical Review A **101**, 052340/1–15 (2020).

[6] K. Sasaki, T. Sugimoto, T. Tohyama, and S. Sota, "Magnetic excitations in magnetization plateaus of a frustrated spin ladder", Physical Review B **101**, 144407/1–9 (2020).

[7] T. Shirakawa, S. Miyakoshi, and S. Yunoki, "Photoinduced η pairing in the Kondo lattice model", Physical Review B **101**, 174307/1–12 (2020).

[8] K. Shinjo, S. Sota, S. Yunoki, and T. Tohyama, "Characterization of photoexcited states in the half-filled one-dimensional extended Hubbard model assisted by machine learning", Physical Review B **101**, 195136/1–10 (2020).

[9] K. Seki and S. Yunoki, "Thermodynamic properties of an $S = 1/2$ ring-exchange model on the triangular lattice", Physical Review B **101**, 235115/1–16 (2020).

[10] R. Fujiuchi, T. Kaneko, K. Sugimoto, S. Yunoki, and Y. Ohta, "Superconductivity and charge density wave under a time-dependent periodic field in the one-dimensional attractive Hubbard model", Physical Review B **101**, 235122/1–7 (2020).

[11] B.-H. Kim, S. Sota, T. Shirakawa, S. Yunoki, and Y.-W. Son, "Proximate Kitaev system for an intermediate magnetic phase in in-plane magnetic fields", Physical Review B **102**, 140402(R)/1–7 (2020).

[12] Y. Otsuka, K. Seki, S. Sorella, and S. Yunoki, "Dirac electrons in the square-lattice Hubbard model with a d-wave pairing field: The chiral Heisenberg universality class revisited", Physical Review B **102**, 235105/1–11 (2020).

[13] H. Ueda, K. Okunishi, K. Harada, R. Krčmár, A. Gendiar, S. Yunoki, and T. Nishino, "Finite-m scaling analysis of Berezinskii-Kosterlitz-Thouless phase transitions and entanglement spectrum for the six-state clock model", Physical Review E **101**, 062111/1–7 (2020).

[14] H. Ueda, K. Okunishi, S. Yunoki, and T. Nishino, “Corner transfer matrix renormalization group analysis of the two-dimensional dodecahedron model”, Physical Review E **102**, 032130/1–8 (2020).
[15] F. Lange, S. Ejima, T. Shirakawa, S. Yunoki, and H. Fehske, “Block-Lanczos density-matrix renormalization-group method approach to spin transport in Heisenberg chains coupled to leads”, Journal of the Physical Society of Japan **89**, 044601/1–6 (2020).
[16] T. Tohyama, S. Sota, and S. Yunoki, “Spin dynamics in the t-t’J model: Dynamical density-matrix renormalization group study”, Journal of the Physical Society of Japan **89**, 124709/1–7 (2020).
[17] F. Lange, S. Ejima, J. Fujimoto, T. Shirakawa, H. Fehske, S. Yunoki, and S. Maekawa, “Generation of current vortex by spin current in Rashba systems”, Physical Review Letters **126**, 157202/1–6 (2021).
[18] K. Seki and S. Yunoki, “Quantum power method by a superposition of time-evolved states”, PRX Quantum **2**, 010333/1–45 (2021).
[19] T. Shirakawa, K. Seki, and S. Yunoki, “Discretized quantum adiabatic process for free fermions and comparison with the imaginary-time evolution”, Physical Review Research **3**, 013004/1–32 (2021).
[20] K. Shinjo, S. Sota, and T. Tohyama, “Effect of phase string on single-hole dynamics in the two-leg Hubbard ladder”, Physical Review B **103**, 035141/1–12 (2021).
[21] J. Fujimoto, F. Lange, S. Ejima, T. Shirakawa, H. Fehske, S. Yunoki, and S. Maekawa, “Spin-charge conversion and current vortex in spin-orbit coupled systems”, AIP Materials **9**, 060904/1–11 (2021).

8.5.2 Invited Talks

[1] S. Yunoki, “Photoinduced superconductivity by η pairs in a Mott insulator”, 2nd International Workshop on Theoretical Developments and Experimental Progresses in Quantum Matter – Emergent Phenomena, August 24–28 (2020), Shanghai (China), Online.
[2] S. Yunoki, “Quantum computing & materials science”, NRC-RIKEN HPC Workshop, October 20-21 (2020), Online.

8.5.3 Oral Talks

[1] S. Sota and S. Yunoki, “DMRG calculation of random Ising models and comparison with D-Wave”, Autumn Meeting of Physics Society of Japan, September 8–11 (2020), Online.
[2] K. Shinjo, S. Sota, and T. Tohyama, “Time-dependent DMRG study of optical conductivity of ladder lattice extended Hubbard model”, Autumn Meeting of Physics Society of Japan, September 8–11 (2020), Online.
[3] Y. Otsuka, T. Yoshida, K. Kudo, S. Yunoki, and Y. Hatsugai, “Higher-order topological Mott insulators on the pyrochlore lattice”, The 76th Annual Meeting of Physics Society of Japan, March 12–15 (2021), online.
[4] K. Shinjo, Y. Tamaki, S. Sota, and T. Tohyama, “Time-dependent density-matrix renormalization group study on the optical conductivity of two-dimensional Hubbard model”, The 76th Annual Meeting of Physics Society of Japan, March 12–15 (2021), Online.

8.5.4 Posters

[1] S. Sota, T. Tohyama, T. Shirakawa, and S. Yunoki, “Dynamical DMRG study of excitation dynamics of triangular lattice antiferromagnetic Heisenberg models”, The 7th HPCI project report meeting, Octorber 29–30 (2020), Online.
[2] S. Sota, T. Tohyama, and S. Yunoki, “Spin dynamics in cuprate superconductors: Dynamical DMRG study of dynamical spin structure factor in the $t - t' - J$ model”, The 7th HPCI project report meeting, Octorber 29–30 (2020), Online.

8.5.5 Software

[1] S. Sota, K. Morita, H. Matsueda, S. Yunoki, and T. Tohyama, “DDMRG (Dynamical DMRG)”, R-CCS software.
[2] S. Sota, S. Yunoki, and T. Tohyama, “2-D DMRG”, R-CCS software.
[3] S. Sota, Y. Imamura, T. Nakajima, S. Yunoki, and T. Tohyama, “paraDMRG”, R-CCS software.
[4] S. Sota and S. Yunoki, “QUARTZ”, R-CCS software.

Chapter 9

Computational Biophysics Research Team

9.1 Members

Yuji Sugita (Team Leader (concurrent))*

Jaewoon Jung (Research Scientist (concurrent))**

Chigusa Kobayashi (Research Scientist)

Koichi Tamura (Special Postdoctoral Researcher)***

Ai Shinobu (Postdosctoral Researcher)

Cheng Tan (Postdosctoral Researcher)

Hiromi Kano (Assistant)

Mitsunori Ikeguchi (Guest Researcher)****

Michael Feig (Guest Researcher)*****

Takao Yoda (Guest Researcher)******

Naoyuki Miyashita (Guest Researcher)*******

* Chief Scientist in RIKEN CPR; Team leader in RIKEN BDR
** Senior Technical Scientist in RIKEN CPR
*** Special Postdoctoral Fellow in RIKEN CPR
**** The main affiliation is Yokohama City University
***** The main affiliation is Michigan State University
****** The main affiliation is Nagahama Bio Institute.
******* The main affiliation is Kindai University.

9.2 Overview of Research Activities

We have developed GENESIS (Generalized-ensemble simulation system) software for high-performance molecular dynamics (MD) simulations of various chemical and biological systems. GENESIS allows us to perform multi-scale MD simulations using various molecular models including quantum mechanics/molecular mechanics (QM/MM) for accurate chemical interaction, atomistic models for accurate molecular interactions, and coarse-grained (CG) models for large-scale biological phenomena. The new version of GENESIS (2.0beta or later) has been highly optimized to Fugaku or other computers by introducing hardware-aware multi-kernels. Each kernel is specifically optimized to the individual processor. Since the multi-kernels are embedded in the new version of GENESIS, users just need to compile it with a standard way.

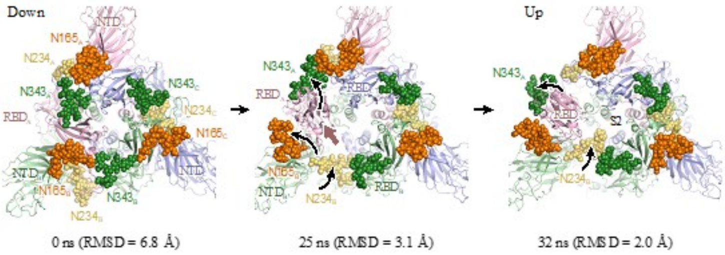


Figure 9.1: Three glycans attached at N165 (orange), N234 (yellow), and N343 (green) have important roles on the stabilization of the RBD domain in spike protein. The structures were taken from one of the target MD simulations starting from the down toward the up structure. Adapted from Biophys. J 120,1060 (2020) with permission of Biophysical Journal, Copyright 2020.

In this fiscal year, the pandemic of SARS-CoV-2 virus caused large social problems in all over the world. We aimed to contribute to resolve the big social problems by using the new version of GENESIS on Fugaku supercomputer. Although Fugaku was under development in the beginning of the fiscal year, we carried out atomistic MD simulations of spike protein on the surface of SARS-CoV-2 using GENESIS 2.0beta on Fugaku. In the MD simulations, we could understand key interactions in the structural conformations discussed below. We also developed various schemes in CG models and atomics MD simulation methods. The new releases of GENESIS (1.6 and 2.0beta) include many advanced computational methods for chemical and biological simulations and both are available as free software under the license of LGPLv2.

9.3 Research Results and Achievements

9.3.1 1. Conformational dynamics of Spike protein on the surface of SARS-Cov-2 using Fugaku

Spike protein on the surface of SARS-CoV-2 binds its receptor binding domain (RBD) to a receptor (Angiotensin-converting ensyme2) when the virus invades a human cell. It is an important target protein for antibodies in human immune system so that we need to understand the structure-dynamics relationship in the protein. From early 2020, structural analyses by cryo-electron microscopy (cryo-EM) have shown that the protein has up- or down-form structures in the RBD regions. There are many Asn residues on the surface of spike protein, which are glycosylated to intervene from the human immune system. However due to their structural flexibility, their atomic structures are determined by cryo-EM and their functional roles have been largely unknown.

We performed 1 microsecond MD simulations of the down- and up-forms of a spike protein with glycans in solution on Oakforest-PACS in JCAHPC and on Fugaku. In the simulations, we identified special glycan-attached residues stabilizing the structure of RBDs. Importantly, the up or down forms are stabilized by different glycans (Fig. 9.1), suggesting that the interaction switch is one of the key mechanisms for the down-to-up conformational changes. Also, we pointed out the importance of electrostatic interactions in the center of spike protein in the down form. Although it is known as an inactive form, the electrostatic repulsions existing in the conformation suggests the possibility of large conformational flexibility of the RBD regions regardless of the binding of its ligands. To explore the conformational space of spike protein more efficiently, we employed gREST (generalized Replica Exchange with Solute Tempering) method, which is one of the enhanced conformational sampling algorithms developed in our group, in the next simulations of spike proteins. So far, we have performed three different gREST simulations: gREST_Down starting from the down form with glycans, gREST_Up from the up form with glycans, gREST_Down w/o glycan from the down without glycans. We are now analyzing these simulation results to shed more light on the molecular mechanisms underlying the down-to-up transitions in spike protein, which could be useful for the development of anti-virus drugs or antibodies.

9.3.2 Implementation of residue-level coarse-grained models in GENESIS

Compared with the classical all-atom molecular dynamics (MD) simulations, coarse-grained (CG) models can simulate large biomolecule complexes at longer time scales. We implemented the latest popular residue-level CG models (Fig. 9.2A) into GENESIS to provide the biophysical society a versatile framework for running CG MD

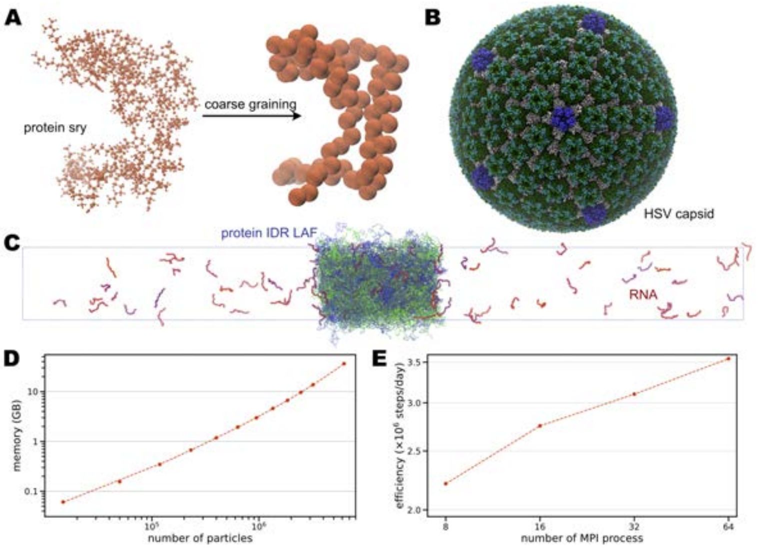


Figure 9.2: Coarse-grained simulations with GENESIS atdyn. (A) Coarse-graining of the protein,SRY, from atomistic structure (left) into the one-bead-per-amino acid CG structure (right). (B) CG simulation of the HSV capsid using the AICG2+ model. The system consists of 1,687,980 CG particles in total. (C) CG simulation of the phase behavior of protein IDR LAF and RNAs. The total number of CG particles is 57,300. (D) Memory benchmark of GENESIS CG simulations. (E) CPU benchmark of GENESIS CG simulations. The benchmark runs were carried out on the RIKEN Hokusai BWMPC. For all the CPU benchmark, OpenMP thread number of 5 was used.

simulations. These models include the atomic interaction-based CG model version 2+ (AICG2+) for protein, the 3-site-per-nucleotide (3SPN) model for DNA, the structure-based model for RNA, and the hydrophobicity scale (HPS) or Kim-Hummer (KH) model for the intrinsically disordered regions (IDR) in proteins and RNAs. We considered the Debye-Hückel type electrostatics and excluded volume potentials for the general non-specific inter-molecular interactions. As for the sequence-specific recognition between protein and DNA, we employed the position-weight-matrix-complex-structure (PWMcos) model. We have implemented and benchmarked all the models mentioned above in the "atdyn" program of the GENESIS package (Fig. 9.2D and 9.2E). We then tested our program on the structure of the HSV capsid (Fig. 9.2B), which is one of the largest systems so far simulated with the residue-level CG simulations to our knowledge. We also validated our implementation by simulating the condensation of the mixture of the protein IDR LAF and RNAs (Fig. 9.2C). We then utilized GENESIS and the CG IDR models to study the phase behaviors of the heat-resistant obscure (Hero) proteins. We also used GENESIS CG simulations to investigate the liquid-liquid phase separation (LLPS) facilitated gene regulation by transcription factors.

9.3.3 Development of molecular dynamics integration enabling a large time step

9.3.3.1 Development of group pressure and temperature

In MD simulations, we update positions and momenta in the equation of motion using a numerical integration scheme with a time step of δt. Generally, δt is limited to 1 - 2 fs to reproduce the fastest motion such as vibrational motions in a molecule. Otherwise, it is not possible to simulate molecular systems such with high stability and accuracy. Recently, we found that larger values of δt can lose the accuracy in MD simulations under the constant temperautre and pressure conditions. The numerical problems mainly result from inaccurate temperature and pressure esitmations. The novel algorithm developed by us provide more accurate evaluations of temperature and pressure and thereby resolve the numerical problems caused with larger time steps. However, it requires iterations when we perform MD simulations with constraints, which increases computational costs significantly. To overcome the problem in the method, we further developed group-based temperature and pressure (group T/P) evaluations in this fiscal year. It increases the accuracy by negelecting the high-frequency vibrational motions of hydrogen atoms in pressure/temperature evaluations. It also increases the overall performance by avoiding iterations in thermostat and barostat updates. From various tests, we found that it conserves

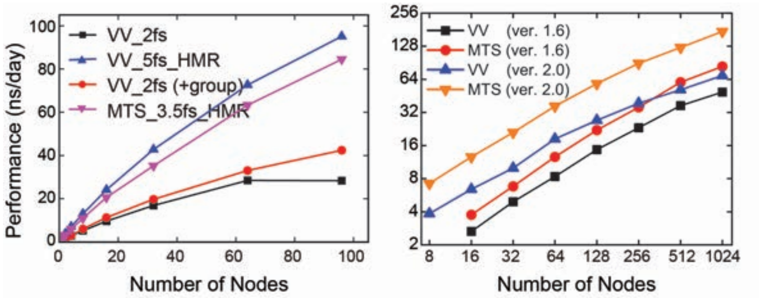


Figure 9.3: Benchmark perofrmance of 1.5 million atoms sytem according to the time step and group group T/P (left) and the performance comparison between GENESIS 2.0beta and GENESIS 1.6 (right).

physical properties up to $\delta t = 5$ fs for conventioanl integrator and $\delta t = 3.5$ fs for multiple time step integrator, respectively.

9.3.3.2 Development of new hydrogen mass repartitioning scheme

Hydrogen mass repartitioning (HMR) scheme is widely used to increase a time step in MD simulations. In HMR, the mass of hdyrogen atoms is increased to prevent high-frequency motion while the the mass of non-hydrogen atom is decreased to conserve the total molecular mass. Usually, the mass of hydrogen atoms is scaled by the factor of 3, which is shown to be stable up to 4 fs. We, however, found that this can introduce unstable structures of five- or six-membered ring molecules when 5 fs time step is used. To solve the problem, we suggested to scale the mass of hydrogen by the factor of 2 for the five- or six-membered rings and 3 (for CHARMM force field) or 2.5 (for AMBER force field) otherwise. We also designed a heterogeneous hydrogen mass scaling where hydrogen mass repartition is not applied to rigid water molecules. With these, kinetic physical properties are conserved more than the usually adopted one.

The new integration schemes were validated by testing various biolgoical systems, including soluble/membrane proteins and lipid bilayers. In all biological systems, we could reproduce the same physical properties even increasing the time step from 2 fs to 3.5 fs (for fast motion) and 7.0 fs (for slow motion). We also confirmed that the new integration scheme increases the speed more than 2-fold. This integration is available in GENESIS 2.0beta, which increases the performance of MD simulations more than three times compared to the existing one (GENESIS 1.6) (Fig. 9.3).

9.4 Schedule and Future Plan

In 2021, we plan to develop an efficient parallelization scheme of CG MD based on domain decomposition scheme. Based on the development, we will investigate cellular scale biological phenomena that has not been accessible up to now.

Using Fugaku supercomputer and GENESIS 2.0beta, we will continue the simulation of spike protein on the surface of SARS-CoV-2. Also, several membrane proteins will be studied using multi-scale MD simulations. We also aim to study new biological problems, such as protein/DNA interactions in nucleosomes or chromatin, protein/protein interactions in membrane less organelle. Intrinsically disordered proteins are known to be important in many biological phenomena, which are also the target of our research.

9.5 Publications

9.5.1 Articles/Journal

[1] Jaewoon Jung, Chigusa Kobayashi, Kento Kasahara, Cheng Tan, Akiyoshi Kuroda, Kazuo Minami, Shigeru Ishiduki, Tatsuo Nishiki, Hikaru Inoue, Yutaka Ishikawa, Michael Feig, and Yuji Sugita, New parallel computing algorithm of molecular dynamics for extremely huge scale biological systems, Journal of Computational Chemistry, vol. 42, 231-241 (2021).

[2] Takahru Mori, Jaewoon Jung, Chigusa Kobayashi, Hisham M. Dokainish, Suyong Re, and Yuji Sugit, Elucidation of Interactions Regulating Conformational Stability and Dynamics of SARS-CoV-2 S-Protein, Biophsyical Journal, vol. 120, 1061-1070 (2021).
[3] Jaewoon Jung and Yuji Sugita, Group-based evaluation of temperature and pressure for molecular dynamics simulation with a large time step, Journal of Chemical Physics, vol. 153, 234115 (2020).
[4] Cheng Tan, Jaewoon Jung, Chigusa Kobayashi, and Yuji Sugita, A singularity-free torsion angle potential for coarse-grained molecular dynamics simulations, Journal of Chemical Physics, vol. 153, 044110 (2020).
[5] Yasuhiro Matsunaga and Yuji Sugita, Use of single-molecule time-series data for refining conformational dynamics in molecular simulations, Current Opinion in Structural Biology, vol. 61, 153-159 (2020).

9.5.2 Invited Talks

[6] Yuji Sugita, 富岳を用いた新型コロナウイルス表面のスパイクタンパク質動的構造予測, 近畿化学協会コンピュータ化学部会, Online, March, 2021.
[7] Yuji Sugita, 新型コロナウイルス表面のタンパク質動的構造予測, HPCIフォーラム, Online, March, 2021.
[8] Yuji Sugita, Replica-exchange simulations on the conformational dynamics of spike protein on the surface of SARS-CoV-2, Molecular Basis of Proteinopathies, Online, February, 2021.
[9] Yuji Sugita, Intrinsic Conformational Flexibility of SARS-CoV-2 Spike Protein Simulated on Fugaku, 2021 R-CCS Symposium, Online, February, 2021.
[10] Yuji Sugita, Multi-scale MD simulations using GENESIS on Fugaku, The 1st Fugaku Bio-supercomputing Workshop on Cellular-Scale Molecular Dynamics Simulations, Online, January, 2021.
[11] Yuji Sugita, スーパーコンピュータが明らかにする蛋白質の動的構造と機能, 理研・東北大学連携シンポジウム「計測科学が拓く生命科学の新展開」, Online, November, 2020.
[12] Yuji Sugita, 計算機シミュレーションで細胞の中を観る, Online, November, 2020.
[13] Jaewoon Jung, Development of GENESIS on Fugaku supercomputer and its application of Spike protein on the surface of SARS-CoV-2 in solution, HPCI計算科学フォーラム, Online, November, 2020.
[14] Yuji Sugita, 細胞環境はどのようにタンパク質の構造・ダイナミクス・機能に影響を与えるか？, Online, October, 2020.
[15] Yuji Sugita, All-atom molecular dynamics simulations of spike protein on the surface of SARS-CoV-2 in solution, RIKEN-NRC HPC Workshop, October, 2020.
[16] Yuji Sugita, 新型コロナウイルス 表面のタンパク質動的構造予測, 第9回JCAHPCセミナー, Online, October, 2020.
[17] Koichi Tamura, Theretical Study on the Transport Cycle of the Heme ABC Transporter BhuUV-T, 第58回生物物理学会年会, Online, September, 2020.
[18] Yuji Sugita, Use of enhanced conformational sampling for the analysis of protein-ligand binding processes, Telluride Summer Research Conference, Online, August, 2020.

9.5.3 Posters

[19] Jaewoon Jung, 小林千草, 笠原健人, Cheng Tan, Michael Feig, 杉田有治, New parallel computing algorithm of molecular dynamics for extremely huge scale biological systems, 第33回 分子シミュレーション討論会, online, December, 2020.
[20] Cheng Tan, Jaewoon Jung, Chigusa Kobayashi, Yuji Sugita, A singularity-free torsion angle potential for coarse-grained molecular dynamics simulations, 第58回日本生物物理学会年会, Online, September, 2020.

9.5.4 Software

[21] Molecular dynamics and modeling software GENESIS, `https://www.r-ccs.riken.jp/labs/cbrt` (version 1.5, 1.6, and 2.0beta)

Chapter 10

Particle Simulator Research Team

10.1 Members

Junichiro Makino (Team Leader)

Masaki Iwasawa (Research Scientist)

Daisuke Namekata (Postdoctoral Researcher)

Miyuki Tsubouchi (Technical Staff)

10.2 Overview of Research Activities

We are developing particle-based simulation software that can be used to solve problems of vastly different scales.

Simulation schemes for hydrodynamics and structural analysis can be divided into grid-based and particle-based methods. In grid-based methods, the computational region is mapped to regular or irregular grids. Continuous distributions of physical values are represented by discrete values at grid points, and the governing partial differential equation is approximated to a set of finite difference equations.

In the case of the particle-based methods, physical values are assigned to particles, while the partial differential equation is approximated by the interactions between particles.

Both methods are widely used, and they have their advantages and disadvantages. The computational cost of grid-based schemes is generally lower than that of particle-based methods with similar number of freedoms. Thus, if an near-uniform grid structure is appropriate for the problem to be solved, grid-based methods perform better.

The advantage of the particle-based methods comes from the fact that they use "Lagrangian" schemes, in which the particles move following the motion of the fluid in the case of the CFD calculation. In the case of grid-based methods, we generally use "Eulerian" schemes, in which the grid points do not move.

There are three points in which the Lagrangian schemes are better than Eulerian schemes. One is that the Lagrangian schemes are, to some extent, adaptive to the requirement of the accuracy, since when a low-density region is compressed to become high density, Second one is that the timestep criteria are quite different. In the case of the Lagrangian schemes, the timestep is determined basically by local sound velocity, while in the Eulerian scheme by global velocity. Thus, if a relatively cold fluid is moving very fast, the timestep for the Eulerian schemes can be many orders of magnitude shorter than that for Lagrangian schemes. Finally, in the case of fast-moving low-temperature fluid, the required accuracy would be very high for Eulerian scheme, since the error comes from the high velocity, while that error would be transferred to internal energy of the fluid element which is much smaller than that of the kinetic motion.

Of course, there are disadvantages of Lagrangian schemes. The primary one is the difficulty of construction of such schemes in two or higher dimensions. In the case of one-dimensional calculation, it is easy to move grid points following the motion of the fluid, but in two or higher dimensions, the grid structure would severely deform if we let the grid points follow the flow. Thus, we have to reconstruct the grid structure every so often. This requirement causes the program to become complex. Moreover, reconstruction of the grid structure (so called remeshing) means we lose numerical accuracy.

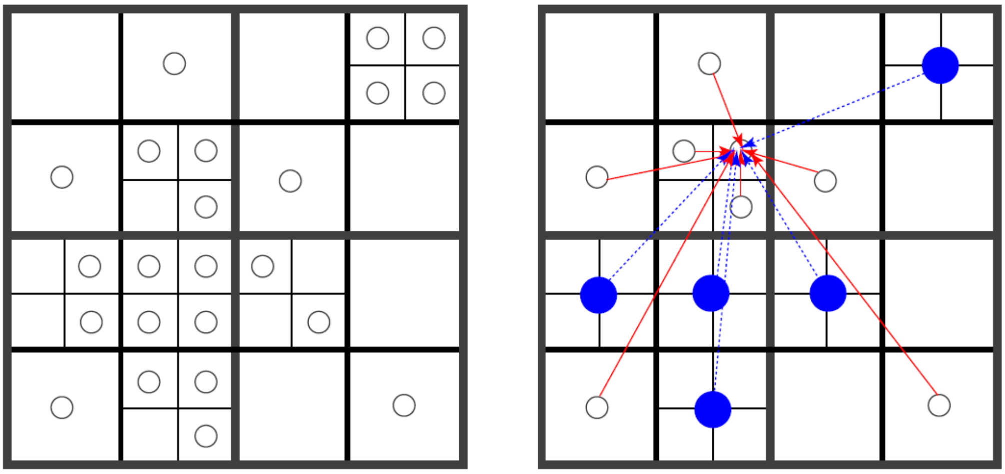


Figure 10.1: Basic idea of tree algorithm

Particle-based methods "solve" this difficulty by not requiring any mesh. In particle-based methods, particles interact with its neighboring particles, not through some connection through grid, but through distance-dependent kernel functions. Thus, there is no need of remeshing. As a result, particle-based schemes are simple to implement, and can give reasonable results even when the deformation is very large. Another important advantage is that it is relatively easy to achieve high efficiency with large-scale particle-based simulation.

In the case of grid-based schemes, in order achieve some adaptivity to the solution, we have to use either irregular grid or regular grid with adaptive mesh refinment. In both cases, adaptivity breaks the regularity of the mesh structure, resulting in non-contiguous access to the main memory. In the case of the particle-based schemes, it does require some irregular memory access, but it is relatively straightforward to make good use of spacial locality, and thereby achieving high efficiency. Similarly, very high parallel performance can be achieved.

However, it has its own problems. In the case of the SPH method, it has been known that the standard scheme cannot handle the contact discontinuity well. It also require rather strong artificial viscosity, which results in very low effective Reynolds number.

Thus, in many fields of computational sciences, many groups are working on implementation of high-performance particle-based simulation codes for their specific problem.

One serious problem here is that, high-performance, highly-parallel simulation codes for particle-based simulations are becoming more and more complex, in order to make full use of modern supercomputers. We need to distribute particles to many computing nodes in an appropriate way, so that the communication between nodes is minimized and at the same time near-optimal load balance is achieved. Within each nodes, we need to write an efficient code to find neighbor particles, rearrange data structure so that we can make good use of the locality, make good use of multiple cores and SIMD units within each core.

Even for the case of very simple particle-particle interaction such as the Lenard-Jones potential or Coulomb potential, the calculation code tends to be very large, and since the large fraction of the code is written to achieve a high efficiency on a specific architecture, it becomes very hard to port a code which is highly optimized to one architecture to another architecture.

Our goal is to develop a "universal" software that can be applied to a variety of problems whose scales are vastly different. In designing such universal software, it is important to ensure that it runs efficiently on highly parallel computers such as the K computer. Achieving a good load balance for particle-based simulations is a difficult task, since using a regular spatial decomposition method causes severe load imbalance, though this works well for grid-based software. Consequently, we have developed an adaptive decomposition method that is designed to work in a way that the calculation time on each node is almost the same, resulting in the near-optimal load balance.

The strategy to develop such a universal software is as follows.

We first construct an highly parallel and very efficient implementation of the TreePM algorithm for gravitational N-body problem. This is actually not a completely new implementation, but the GreeM code developed by researchers of the Strategic Program for Innovative Research (SPIRE) Field 5 "The origin of matter and the universe. In collaboration with the Field 5 researchers, we improve the efficiency of the code and study the issues of the data structure, domain decomposition, load balance strategy etc.

In the second stage, we will develop a prototype of the parallel particle simulation platform. We will design the platform so that it can be used for multiple physical systems. In practice, we consider the following three applications as the initial targets.

1. Gravitational N-body simulation

2. Smoothed Particle Hydrodynamics

3. Molecular Dynamics

In the meantime, we will also investigate the way to improve the performance and accuracy of the current particle-based algorithms for hydrodynamics.

10.3 Research Results and Achievements

10.3.1 High-performance gravitational N-body solver.

We use the TreePM algorithm as the basic method for the evaluation of gravitational interaction between particles. TreePM is a combination of the tree method and the P^{3}M (particle-particle particle-mesh) scheme. Figure 1 shows the basic idea of the tree algorithm. The space is divided into a hierarchical octree structure (quadtree in the figure). Division is stopped when a cell contains one or no particle. When we calculate the force on a particle, we evaluate the force from a group of particles, with size larger for more distant particles. In this way, we can reduce the calculation cost from O(N2) to O(N log N).

The tree algorithm is widely used, but when the periodic boundary condition is applied, we can actually use a more efficient efficient scheme, since we can calculate the long-range, periodic term using FFT. The P^{3}M scheme has been used for such problem, but it has the serious problem that when the density contrast becomes high, the calculation cost increases very quickly. The TreePM scheme solves this difficulty by using the tree algorithm to evaluate the forces from nearby particles. Even when there are very large number of neighbor particles, the calculation cost does not increase much, since the calculation cost of the neighbor force is proportional to the logarithm of the number of neighbors.

In order to map the problem to the distributed-memory parallel computer such as the K computer, we adopted the approach to divide the space into domains and assign particles in one domain to one calculation node. We used the orthogonal recursive multisection method developed by the team leader some years ago. It is the generalization of the orthogonal recursive bisection (ORB), which has been widely used in many parallel implementations of the tree algorithm.

With ORB, we recursively divide space into two halves, each with the same number of particles. An obvious disadvantage of the ORB approach is that it can utilize the computing nodes of integral powers of two. Thus, in the worst case we can use only half of the available nodes.

The difference between the multisection method and the ORB is that with the multisection method we allow the divisions to arbitrary number of domains, instead of bisection. This would allow too many possible divisions. In our current implementation, we limit the number of levels to three, and make the numbers of divisions at all levels as close as possible. Thus, our domain decomposition is topologically a simple three-dimension grid. This fact makes the multisection method well suited to the machines with the 3D torus network like the K computer.

We have developed a "reference code" for gravitational N-body simulation on the K computer. This code is fairly well optimized for the K computer, and shows quite good scalability for even for relatively small-size problems. The asymptotic speed per timestep for large number of nodes is around 7ms. This speed is comparable to that of highly optimized molecular dynamics codes on K, even though our code is designed to handle highly inhomogenous systems.

We used this code as the reference implementation for more generalized particle simulation platform which will be described in the next subsection.

10.3.2 Particle Simulation Platform.

In FY 2014, We have completed and released Version 1.0 of the particle simulation platform, which we call FDPS (Framework for Developing Particle Simulator). In FY 2015, we have applied a number of improvements to FDPS.

The basic idea of FDPS is that the application developer (or the user) specified the way the particles interact with each other, and the rest is taken care by FDPS. Here, "the rest" includes domain decomposition and re-distribution of particles, evaluation of interactions between particles, including those in different domains (different MPI processes, for example).

In practice, there are many additional details the user should give. Consider a relatively simple case of particles interacting with softened 1/r potential. There are a number of small but important points one has to decide on. For example, what algorithm should be used for the interaction calculation? Even if we limit the possibilities to reasonably adaptive schemes for open boundary problems, we have the choice between Barnes-Hut tree and FMM. For both algorithms, there are many different ways to parallelize them on distributed-memory parallel computers. Also, there are infinitely many variations for the time integration schemes.

The base layer of FDPS offers the domain decomposition based on the recursive multisection algorithm, with arbitrary weighting function for the load balancing. It also offers the parallel implementation of interaction calculation between particles.

The domain decomposition part takes the array of particles on each node as the main argument. It then generates an appropriate domain for each node, redistribute particles according to their locations, and returns.

The interaction calculation part takes the array of particles, the domain decomposition structure, and the specification of the interaction between particles as main arguments. The actual implementation of this part need to take into account a number of details. For example, the interaction can be of long-range nature, such as gravity, Coulomb force, and interaction between computational elements in the boundary element method (BEM). In this case, the user should also provide the way to construct approximations such as the multipole expansion and the way to estimate error. The interaction might be of short-range nature, with either particle-dependent or independent cutoff length. In these cases, the interaction calculation part should be reasonably efficient in finding neighbor particles.

We have successfully implemented all of these functionalities in FDPS version 1.0. (https://github.com/FDPS/FDPS). Using FDPS, a gravitational N-body simulation code can be written in 120 lines, and that code is actually fully scalable even to full-node runs on K computer. For SPH calculations, we have also achieved similar scaling.

FDPS is implemented as a class template library in C++ language. It receives the class definition of particles and a function (or multiple functions in the case of complex interactions) to evaluate the interaction between particles. When a user program is compiled with the FDPS library, the class template is instantiated with the user-specified definition of the particle class. Thus, even though the FDPS library functions are generic ones not specialized to a particular definition of particles, it behaves as if it is a specialized one.

The measured performance of applications developed using FDPS is quite good. Both for gravity-only calculation and SPH calculation, weak-scaling performance is practically perfect, up to the full-node configuration of K computer. Moreover, the measured efficiency, in terms of the fraction of the peak floating-point performance, is also very high. It is around 50% for gravity-only calculation. For SPH calculations, at the time of writing the performance is around 10%.

In FY 2015, we have extended FDPS in several important directions. The first one is the improvement of the strong scaling. The algorithm used for the domain decomposition contains one serial bottleneck. The "sampling" algorithm used in FDPS 1.0 works well only when the average number of particles per MPI process is significantly larger than the total number of MPI processes. We developed a new parallel algorithm, in which $O(p^{1/3}$ MPI processes are used to decompose the computational domain. Here p is the total number of MPI processes. Thus now the requirement for the number of particle is relaxed from larger than p to larger than $p^{2/3}$. Now we can achieve pretty good performance for around 1 billion particles, on the full nodes of K computer. Previously we need near 100 billion particle to achieve good efficiency.

The second one is the addition of new interface method to interaction calculation function, which allows efficient use of accelerator hardware such as GPGPU or Intel MIC. In order to achieve high performance on accelerators, it is important to pass a large chunk of work at one time. In order to achieve this goal, in the current version of FDPS the CPU creates the list of multiple interaction lists, and send all of them at once so that the overhead of the initialization of the accelerator would not become a bottleneck. This interface has been tested on NVIDIA GPGPUs as well as the PEZY-SC processor.

In FY 2016, we have released FDPS 3.0. The most important new feature of this release is the interface to application programs written in Fortran. FDPS itself is implemented using C++. The reason why we adopted C++ is to use its "template" functions. Using templates, we can write library functions which accept user-defined data types as template arguments. This means we can effectively generate "specialized" libraries for user-specified particle data types, without knowing the data types beforehand.

In FY 2017, we have released several new versions of FDPS, up to 4.0a. There are a number of improvements, mostly for improved performance, For example, we have implemented the reuse of the interaction list. In the case of the calculation of short-range interactions, the neighbor-list method or so-called bookkeeping method has been used in many applications. On the other hand, to our knowledge, such a method has not been applied to Barnes-Hut treecode or FMM. There is no fundamental difficulty in combining the two methods, and the reason why such a combination has not tried before is probably it was not really necessary. The calculation cost of constructing the tree structure and traversing the tree to construct the interaction lists is, int the case of Barnes-Hut algorithm, a small fraction of the total calculation cost. Thus, it is usually unnecessary to try to reduce the cost of the tree construction and tree traversal.

However, some of recent high-performance computers have rather extreme ratios in various aspects, and thus require the performance improvements which were not necessary. One example is the Sunway SW26010. It's architecture is rather extreme in two aspects. First, its "core group" consists of one MPE (management processing element) and 64 CPEs (computing processing elements). MPE has data cache and runs the primary thread, and CPEs do not have cache. Thus, it is difficult and very time-consuming to develop the program which runs on CPEs, in particular for complex operations like tree construction. On the other hand, MPE is very slow compared to CPE, and thus we need to minimize the computational work of MPE.

Another aspect is the rather low memory bandwidth. The B/F number of SW26010 is around 0.03, which is around 1/15 of that of K computer. Thus, in order to achieve reasonable performance on SW26010, we need to minimized the main memory access per timestep.

The reuse of the interaction list turned out to be very effective on SW26010 and other machines with relatively low memory bandwidth, such as NVIDIA P100/V100 and PEZY-SC2.

In FY 2018, we have improved the API of FDPS and also worked on further performance improvement. Concerning the improvement of API, we added the API in the C language in FDPS 5.0. Before 5.0, user programs should be written either C++ or Fortran. Although these two languages cover a fair fraction the needs of of HPC users, it is clearly desirable to have API in C language, since that would allow users to write programs not only in C languages but also in any other languages with FFI (foreign function interface), since FFI is usually defined in C. The C language API works much in the same way as Fortran API.

In FY 2019, we have worked mainly on new functionalities for FDPS. The first one is the support of PM^3 (Particle-Mesh Multipole) method. As its name suggests,PM^3 method is a combination of Particle-Mesh method and Fast Multipole method. In the Particle-Mesh method, long-range and short-range interactions are separated using smooth splitting function, and the long-range interaction is calcualted using FFT. In the case of PM^3 method, Long-range interaction is defined as the interactions calculated using the tree higher than a certain level in FMM. Then, instead of using hieratchical FMM method, the long-range interaction is obtained as the convolution of the muitipole expansions and Green's function, and then FFT is used to evaluate convolution. Compared to the traditional particle-mesh (or even particle-mesh Ewald) method, PM^3 method can achieve much higher accuracy with significantly smaller amount of communication, and thus suited to large-scale parallel machines with relative weak network. We also worked on the optimization of FDPS on Fugaku. Compared to K, Fugaku has relatively weak network and CPU core with large latency for arithmetic operations. Thus, bottlenecks appear in several unexpected places. We have implemented various new algorithms to remove these bottlenecks.

In FY 2020, we have worked mainly to improve the performance of FDPS on Fugaku. One important feature of Fugaku is its deep arithmetic pipeline (9 stages) and relatively weak OOO resources. These features made it difficult to achieve high efficiency in particle-particle interaction kernels, which are usually the most expensive part of particle-based simulations. In order to solve this issue, we have developed PIKG (Particle Interaction Kernel Generator, `https://github.com/FDPS/PIKG`. PIKG allows application programmers to write interaction kernels in high-level, simple DSL, from which the software generates optimized code for various platforms, including Intel x86 processors with AVX2 or AVX512 extensions, the A64fx processor with SVE, and NVIDIA GPUs with Cuda. For A64fx, the generated code is written fully using SVE intrinsics and more importantly include compiler pragmas for loop unrolling and loop fission. Thus, PIKG-generated kernels achieve quite high efficiencies (up to 30%). Unfortunately, even with such techniques the efficiency of Fugaku on compute-intensive kernels is not as high as that have been achieved on other platforms, in particular K computer, because of the difference in microarchitecture as well as instruction-set architecture. It will be important to utilize such experiences for the design of next-generation processors.

10.4 Schedule and Future Plan

We plan to improve the performance of FDPS further in FY 2021. The main issue will be the performance tuing on Fugaku supercomputer. Other issues include new scheme for the handling or periodic coordinates and domain decomposition optimized to disk- and ring-shaped systems. The particle simulator research team will end by FY 2021, but we will continue the development, maintenance and user support of FDPS elsewhere.

10.5 Publications

10.5.1 Articles/Journal

[1] Iwasawa, M., D. Namekata, K. Nomura, Tsubouchi, and J. Makino, *Extreme-scale particle-based simulations on advanced HPC platforms*, CCF Transactions on High Performance Computing, 2020, 13, https://doi.org/10.1007/s42514-020-00020-1.
[2] Hernandez, D. M., S. Hadden, and J. Makino, *Are long-term N-body simulations reliable?*, Monthly Notices of the Royal Astronomical Society, 2020, **493**, 1913-1925.
[3] Wang, L., K. Nitadori, and J. Makino, *A slow-down time-transformed symplectic integrator for solving the few-body problem*, Monthly Notices of the Royal Astronomical Society, 2020, **493**, 3398-3411.
[4] Wang, L., M. Iwasawa, K. Nitadori, and J. Makino, *PETAR: a high-performance N-body code for modelling massive collisional stellar systems*, Monthly Notices of the Royal Astronomical Society, 2020, **497**, 536-555.

10.5.2 Software

[5] FDPS github.com/FDPS
[6] Formura github.com/Formura

10.5.3 Patents

Chapter 11

Computational Climate Science Research Team

11.1 Members

Hirofumi Tomita (Team Leader)

Yoshiyuki Kajikawa(Senior Research Scientist)

Seiya Nishizawa (Research Scientist)

Sachiho Adachi (Research Scientist)

Tsuyoshi Yamaura (Research Scientist)

Kenta Sueki (Postdosctoral Researcher)

Toshiki Matsushima (Postdosctoral Researcher)

Yuta Kawai (Postdosctoral Researcher)

Tomoro Yanase (Junior Research Associate)

Hiroaki Miura (Guest Researcher)

Yoshiaki Miyamoto (Guest Researcher)

Yousuke Sato (Guest Researcher)

Shinichiro Shima (Guest Researcher)

Kazuyoshi Kikuchi (Guest Researcher)

Ryuji Yoshida (Guest Researcher)

11.2 Overview of Research Activities

The primary aim of Computational Climate Science Research Team is to indicate a direction of future climate modeling with a reliable suggestion for the age of high-performance computers. For this purpose, we intend to construct a basic library, in which model components and their numerical method are inter-exchangeable among multiple models. This work contributes directly to the climate modeling community for the enhancement of fast output/outcome creation. Using the library, we also develop a more advanced climate model with new techniques that is necessary for efficient climate simulation. We promote to develop them, considering the following issues; the readability of code, its convenience for users, and traceability of computational results.

The second aim is to pursue the high efficiency of the climate model in massively parallel computers. Computational meteorology and climatology always require large-scale numerical experiments. However, it is recently pointed out that conventional computer architecture with the straightforward extension of existing simulation codes would be a limitation in getting higher performance. From the viewpoint of hardware and climate modeling, we consider the numerical method, algorithm, and their implementation to large-scale computers, cooperating with research teams of computational and computer sciences in R-CCS. The third aim is to apply our model to address the meteorological/climatological problems, collaborating with outside research institutes. Climate studies are widely spread from the basic understanding of phenomena to the assessment of the environment.

The former is more scientific and the latter is more practical for the requirement of the society. Several issues we currently focus on are as follows; feedback mechanism between cloud, aerosol, and radiation, theory for the moist process linked to the turbulence process, comprehensive understanding of our earth's particularity by investigating other planets, and so on. The assessment of future environmental change for disaster prevention at the regional scale level is also one of our targets.

In this fiscal year, we improved the basic library for more reliable simulation results. We also evaluated characteristics of moist convections in high-resolution simulations, especially their dependence on simulation resolution.

11.3 Research Results and Achievements

11.3.1 Research and development of SCALE

Porting to Fugaku : So far, SCALE has been developed for computer systems with Fujitsu SPARC architecture and x86 architecture such as K computer. The development for compilation and execution environment in Fugaku using the open software of Spack was a crucial issue. Since SCALE uses netCDF, it is necessary to compile and execute using Spack. However, a problem here was that the compile/execution procedure changes depended on changes in the include file and library paths that accompany the update of the Spack itself and the version of the provided package. In general, it is difficult for the users to deal with them. Therefore, for users' convenience, we have developed a framework that can appropriately handle different versions. This work was imported into the SCALE repository with the documentation for the procedure for compiling and executing.

Optimization : The execution performance of SCALE measured by as on Fugaku was just 5 % against the theoretical peak performance of floating-point arithmetic. Compared to the 10% on the K computer, it was found that the execution efficiency was significantly degraded. As a result of optimization, the performance was improved up to 7 % against peak performance. The main optimizations are as follows:

1. SIMD: Eliminate the dependency in the innermost loop by changing the array and loop structure
2. Loop splitting: Splitting a huge loop to reduce register usage and enable software pipelining by the compiler.
3. Reduction of floating-point arithmetic amount: Change the loop structure and make the number of intermediate variables be small to reduce the number of calculations such as power and logarithm calculation.
4. Inline expansion: Manually expand the part that the compiler could not expand inline.
5. Change thread parallelism scheduling: Review the thread parallel scheduling in loops with a biased amount of calculations.

This information was published at the 1st A64FX Tuning Technology Research Meeting, held on 9th December in 2020 by RIKEN and RIST.

Solving system problems : During the SCALE improvement, we discovered multiple system malfunctions and cooperated with the support desk to contribute to the improvement of Fugaku. Some of these bugs were fatal. One is causing the program to hang the SCALE for more than a certain amount of time. The timing at which the program stopped was different for each execution, and we spent a lot of time identifying the problem. Finally, we found out that there was a problem with the MPI library itself and reported it to the support desk. Thus, we also contributed widely to the sophistication of the MPI library.

In this fiscal year, we concentrated the development and work necessary to carry out SCALE in Fugaku was carried out. As a result, SCALE has been available in Fugaku, and this model is currently being used in the

Fugaku result accelarattion program and some Fugaku research projects in HPCI. On the other hand, there are still challenges in the development and improvement of SCALE on Fugaku, such as more efficient execution. It is necessary to continue research and development for expanding and enhancing the use of SCALE.

11.3.2 The Hyogo-Kobe COE establishment project

We continued to drive the subject "Computational Research on Estimation to Complex Disaster Risk for Better Urban Planning" in Hyogo-Kobe COE establishment project started in FY2018. In this fiscal year, we published a review paper on the method of regional climate assessment using the regional climate model. A brief description will be given below. Dynamic downscaling (DDS) is one of the numerical experimental methods for predicting regional-scale climate and understanding the mechanism of climate change. In the general DDS method, the climate data calculated by the parent model (often, the global climate model: GCM) is given as the boundary condition of the child model (regional climate model: RCM) for estimating the regional climate. Limiting the computational domain allows us to conduct high-resolution computations with the same computational resources with more sophisticated schemes.

On the other hand, neither GCM nor RCM can perfectly reproduce the actual atmospheric conditions, so the atmospheric conditions calculated by the DDS method include model bias. Observed values can be obtained by targeting past climates, so data assimilation would reduce model bias. However, when predicting future climate, the model bias is unavoidable because the future atmospheric conditions are unknown. In particular, since RCM is constrained by the GCM climate as boundary conditions, model bias derived from GCM is an important issue in future climate prediction. With the above background as a motivation, many modified DDS methods (MDDS) have been proposed to reduce model bias in future climate prediction and to understand the factors of future climate change.

The calculated climatic conditions in the DDS method are constrained by boundary conditions, while we can actively use this property. By devising boundary conditions, it is possible to quantitatively estimate the sensitivity and response to constraints. Many MDDS methods suggest ways to modify the boundary conditions to suit their purpose. Even if the purpose is the same, the method of generating boundary conditions is different. The method used depends on the climatic characteristics of the target area and how we do interpret the impact of constraints on the calculation results. In this study, we proposed a new concept for understanding the DDS method, considering the following three points. First, RCM calculation results with a constraint can be divided into three components: "large-scale mean state", "large-scale perturbation component", and "other small-scale components" (Nishizawa et al. 2018; Adachi et al.2017). We then mathematically analyzed the model biases contained in the RCM results by dividing them into GCM-derived biases in the constraints and RCM-dependent biases. Third, based on these mathematical concepts, the DDS methods proposed so far are categorized in terms of how to handle model bias (Fig. 11.1). The climatology conditions calculated by the conventional DDS method are highly biased in both large-scale mean conditions and perturbation components (Fig. 11.1 (a)). The modified DDS method, on the other hand, attempts to reduce such model bias. As a result, the expected calculation results are distributed near the actual atmospheric conditions (Fig. 11.1 b, c, d). In the DDS method shown in Fig. 11.1 (b, c), the calculation results are distributed around the constraints. However, with the method in Fig.11.1(d), the calculation result is between the constraint and the actual atmospheric conditions. This difference is due to the following reasons: The method in Fig. 11.1 (b, c) assumes that large mean and perturbation components are calculated under the influence of constraints, while the method in Fig. 11.1 (d) that simply corrects the mean component bias is expected to correct both the large mean and perturbed components of the calculation. Journal reviewers appreciate this reconstruction of DDS methods.

As successive research, we are investigating the non-linear effects based on our FSCC concept in FY2020. First, a $\Delta C - \Delta P - \Delta cp$ scatters diagram was proposed to analyze the characteristics of the nonlinear effect (Fig. 1). In this figure, the horizontal axis represents the change in the average state of large-scale atmospheric conditions (ΔC), and the vertical axis represents the change in the perturbation component of large-scale atmospheric conditions (ΔP). The non-linear effect (Δcp) is indicated by the color of the topological space with these axes. The non-linear effect is obtained by $\Delta cp = \Delta - \Delta C - \Delta P$. Where Δ is the region's climatic response to large-scale changes in atmospheric conditions. We found that the non-linear effect intends to suppress the linear summation of each factor change effect. We also found that this tendency is strong for the precipitation by extreme events. This achievement is now prepared to submit to a journal.

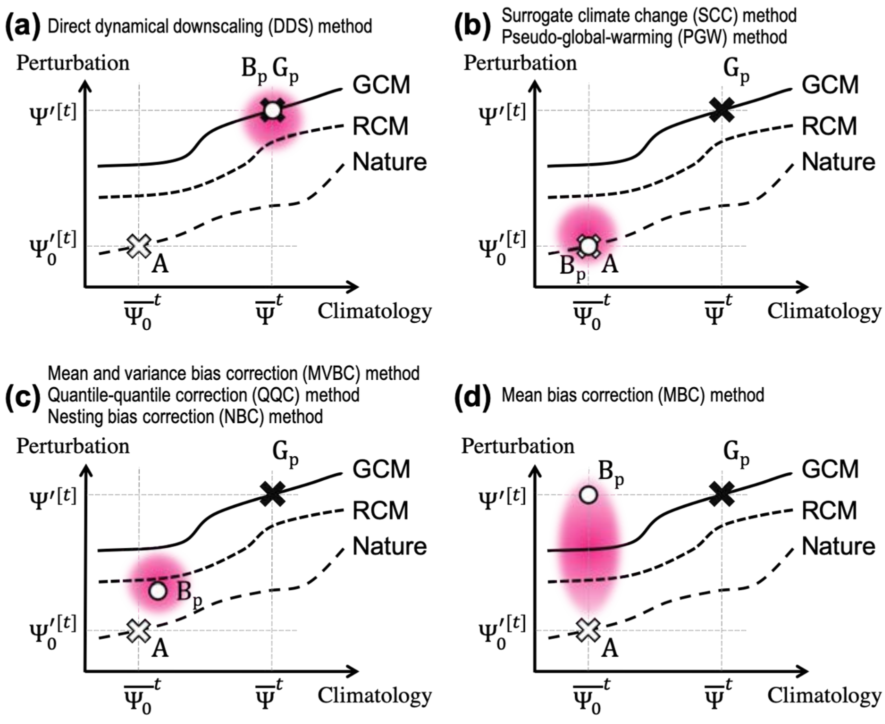


Figure 11.1: Schematic of the bias correction methods in a climatology-perturbation phase space. The horizontal and vertical axes represent the climatology and perturbation components with a large spatial scale, respectively. The solid, short dashed, and long dashed lines illustrate the state curve corresponding to the stable relation between climatology and perturbation components in a GCM, an RCM, and nature (reanalysis data), respectively. The white and black crosses illustrate the large-scale true state and the large-scale state simulated by a GCM, respectively. The white circle represents the constraint condition driving the downscaling simulations. The red areas show the expected position of RCM output, which is projected on the phase space. This figure is from Figure 4 in Adach & Tomta 2020, *Journal of Geophysical Research*)

11.3.3 Self-organization of moist convection in the idealized radiative-convective equilibrium simulation

Atmospheric moist convection plays significant roles in the earth's weather and climate system: it not only causes clouds and precipitation at a local scale but also organizes into a hierarchical structure extending to a global scale. Despite its importance being widely recognized, the representation of cloud-related processes is still insufficient and brings large uncertainties in climate model simulations. Recently, toward deepening basic understanding of the role of clouds, a numerical simulation framework of idealized climate, called radiative-convective equilibrium (RCE), is spotlighted in the climate modeling community. In the RCE simulation, it is known that moist convection can be organized into an aggregated cloud system even under a uniform boundary condition; that process is called convective self-aggregation (CSA) and is recognized as a key for the relationship of clouds and climate. Previous studies have shown that the CSA occurred if the simulation domain was larger than 200–300 km to a horizontal extent, which suggests the existence of critical length for the CSA onset. On the other hand, the CSA onset also depends on the simulation resolution; only in the low-resolution simulation CSA have been found to spontaneously occur.

In FY2018, to investigate the characteristic length of CSA onset, we conducted systematic cloud-resolving simulations, with a scope covering the horizontal domain size and resolution, by using SCALE on the supercomputers including the K-computer. As a result, we updated an RCE regime diagram compared to Muller and Held (2012, MH12), The main conclusions are summarized as follows: (1) CSA is unlikely to occur with a high resolution or small domain (the lines II and III, and MH12 reference line). (2) CSA occurs in sub-kilometer high-resolution cloud-resolving simulations with domain sizes larger than 500 km (line I).

In this fiscal year, a CSA onset mechanizm was further investigated and this result, as shown in Fig. 11.2, togerther with the above coclusion was published (Yanase et al. 2020).

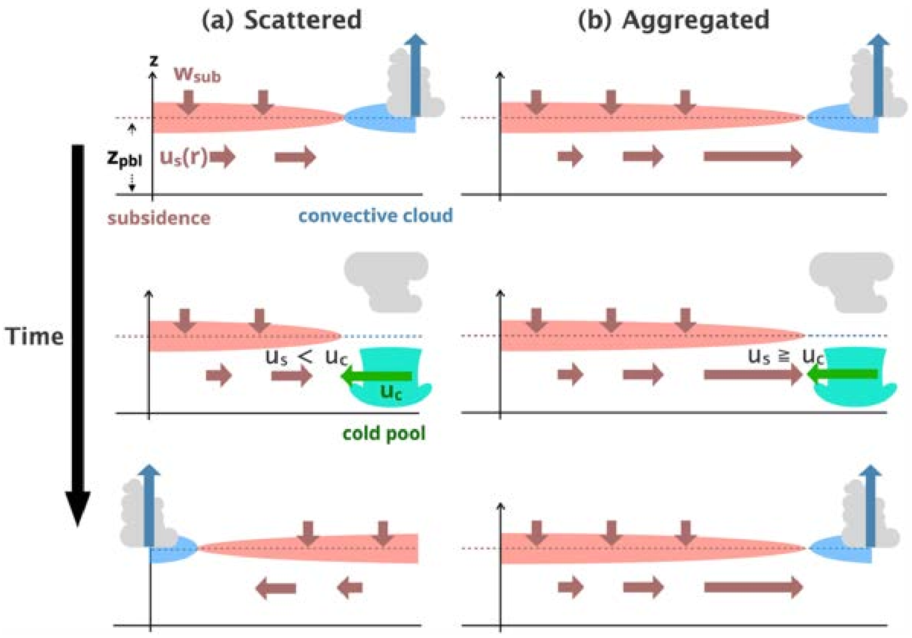


Figure 11.2: Schematic of the competition between the two opposite feedbacks by horizontal divergent flows in a subsidence area and a convective area accompanied by temporal evolution. (a) Scattered case and (b) aggregated case. This figure is from Figure 4 in Yanase et al. 2020, *Geophysical Research Letter*)

11.4 Schedule and Future Plan

The basic library SCALE is being advanced. At the moment, major processes such as dynamics and physics have already been comprehensively developed. In the future, we will pursue ease of use for external users. The advanced climate models based on the schemes continue to be improved in principle and provide a direction for future climate models. The key to advancement is the sub-grid scale models. A more accurate representation of cloud physics requires a turbulence model that takes into account moist processes. The combination of a turbulence-aware LES and Super Droplet Method (SDM) for cloud microphysics will be a useful tool for solving this issue. As well as developing accurate schemes for high-resolution simulations, there is also a need to improve parameterizations for coarser-resolution simulations. Data science techniques are useful for this purpose, including optimization of model parameters in the parameterizations and development of AI-based parameterization generated using a high-resolution simulation database.

Estimating model parameters using Data assimilation will be one major research axis. In fact, we have begun research on parameter estimation using data assimilation. Currently, we are conducting basic research on this topic while focusing on cloud microphysical processes under ideal experimental settings. In the future, we would like to establish such an automatic tuning technology.

One of our primary missions is to provide tools for the analysis of simulation results as well as improve the computational and physical performance of simulations. This is critical for accelerating scientific and social outcomes. Specifically, we will parallelize major analysis programs and make them suitable for massively parallel computer systems. In addition, we will continuously develop a short- and medium-range numerical weather forecasting system in collaboration with Data Assimilation Research Team. LETKF is an important application for data assimilation. Combining our models with LETKF, which combines high-resolution and large-scale ensemble simulations and observational big data, will enable us to take a new stage in numerical weather forecasting, especially for sudden and heavy rainfall. This has already begun as an applied research project under the AIP project and Program for Promoting Researches on the Supercomputer Fugaku (Issue 04).

11.5 Publications

11.5.1 Articles/Journal

[1] Miyamoto, Y., S. Nishizawa, and H. Tomita (2020): Impacts of Number of Cloud Condensation Nuclei on Two-Dimensional Moist Rayleigh Convection, J. Meteorol. Soc. Japan, 98(2), 437-453

[2] Fudeyasu, H., R. Yoshida, M. Yamaguchi, H. Eito, C. Muroi, S. Nishimura, K. Bessho, Y. Oikawa, and N. Koide (2020): Development Conditions for Tropical Storms over the Western North Pacific Stratified by Large-Scale Flow Patterns, Journal of the Meteorological Society of Japan, 2020 Volume 98 Issue 1 Pages 61-72, 2020-004

[3] Morrison, H., M. Lier-Walqui, A. M. Fridlind, W. W. Grabowski, J. Y. Harrington, C. Hoose, A. Korolev, M. R. Kumjian, J. A. Milbrandt, H. Pawlowska, D. J. Posselt, O. P. Prat, K. J. Reimel, S. Shima, B. Diedenhoven, and L. Xue(2020): Confronting the challenge of modeling cloud and precipitation microphysics, Journal of Advances in Modeling Earth Systems, Volume12 (8)

[4] Adachi, S. A. and H. Tomita(2020): Methodology of the constraint condition in dynamical downscaling for regional climate evaluation: A review, JGR Atmospheres, Vol. 125, Issue 11, e2019JD032166

[5] Yanase, T., S. Nishizawa, H. Miura, T. Takemi, and H. Tomita (2020): New Critical Length for the Onset of Self-Aggregation of Moist Convection, Geophysical Research Letters, Volume47, Issue16,

[6] Shima, S., Y. Sato, A. Hashimoto, and R. Misumi (2020): Predicting the morphology of ice particles in deep convection using the super-droplet method, Geoscientific Model Development, Volume 13, issue 9, 4107–4157

[7] 末木健太, 栃本英伍 竜巻の発生環境 (2020), 気象研究ノート, 第243号、P.47-70

[8] Fujita, K., K. Koyoma, K. Minami, H. Inoue, S. Nishizawa, M. Tsuji, T. Nishiki, T. Ichimura, M. Hori, and L. Maddegedara (2020): High-fidelity nonlinear low-order unstructured implicit finite-element seismic simulation of important structures by accelerated element-by-element method, Journal of Comutational Science, Volume 49, 101277

[9] Fudeyasu, H., K. Yoshida, and R. Yoshida (2020): Future changes in western north Pacific tropical cyclone genesis environment in high-resolution large-ensemble simulations, MDPI Oceans, 1(4), 355-368

[10] Sato, Y., Y. Miyamoto, and H. Tomita(2020), Lightning frequency in an idealized hurricane-like vortex from initial to steady-state using a coupled meteorological and explicit bulk lightning model, Monthly Weather Review, Volume 149: Issue 3, Page(s): 753–771

[11] Takayabu, I., R. Rasmussen, E. Nakakita, A. Prein, H. Kawase, S.-I. Watanabe, S. A. Adachi, T. Takemi, K. Yamaguchi, Y. Osakada, and Y.-H. Wu, Summary of the 4th international convection-permitting Modeling workshop for climate research, GEWEX News, 31(1) (2020)

[12] Miyamoto, Y., Y. Sato, S. Nishizawa, H. Yashiro, T. Seiki, and A. T. Noda (2020), An energy balance model for low-level clouds based on a simulation resolving mesoscale motions, J. Met. Soc. Japan, 98, 987-1004

11.5.2 Conference Papers

[13] Yashiro, H., K. Terasaki, Y. Kawai, S. Kudo, T. Miyoshi, T. Imamura, K. Minami, H. Inoue, T. Nishiki, T. Saji, M. Satoh, and H. Tomita (2020): ”A 1024-Member Ensemble Data Assimilation with 3.5-Km Mesh Global Weather Simulations” in SC20: International Conference for High Performance Computing, Networking, Storage and Analysis (SC), Atlanta, GA, US, 2020 pp. 1-10. (Gordon Bell Finalist)

11.5.3 Posters

[14] Ichimura, T., K. Fujita, K. Koyama, R. Kusakabe, K. Minami, H. Inoue, S. Nishizawa, M. Tsuji, T. Nishiki, M. Hori, L. Maddegedara, N. Ueda: Fast scalable implicit solver with convergence of physics-based simulation & data-driven learning: toward high-fidelity simulation with digital twin city, SC20, online, 2020/11/19.

[15] 梶川義幸, 山浦剛: Topographic and resolution impact on the monsoon rainfall in dynamical downscaling simulation, JPGU-AGU Joint Meeting 2020, オンライン, 12-16 July 2020.

[16] 山浦剛, : 浮動小数点演算誤差を利用したアンサンブル予測, 日本気象学会2020年度秋季大会, オンライン, 2020/10/27-30.

[17] Yamaura, T.: The ensemble weather forecasting by using the error of floating-point numbers, American Geophysical Union Fall Meeting 2020, Online San Francisco, USA, 2020/12/01-17.

11.5.4 Invited Talks

11.5.5 Oral Talks

[18] 河合佑太, 富田浩文: ラージエディシミュレーションにおける力学コアの高精度化の必要性に関する考察, 日本気象学会2020年度春季大会, 誌上開催, 2020/5/22.

[19] 末木健太, 山浦剛, 西澤誠也, 富田浩文: Ensemble Kalman Filterを用いた雲微物理スキームのパラメータ推定, 気象学会2020年度春季大会, 誌上開催, 2020年5月22日.

[20] 河合佑太, 富田浩文: ラージエディシミュレーションに必要とされる大気力学コアの数値精度に関する研究, JpGU-AGU Joint Meeting 2020: Virtual, オンライン開催, 2020/7/14

[21] 足立幸穂, 富田浩文: 地域気候変化の評価のための力学的ダウンスケーリング手法の本質, 第61回大気環境学会年会, 誌上開催, 2020/9/14-10/4

[22] 河合佑太, 富田浩文: ラージエディシミュレーションで必要とされる力学コアの離散精度に関する研究, 日本気象学会2020年度秋季大会, オンライン開催, 2020 年 10 月 28 - 30 日

[23] 柳瀬友朗, 西澤誠也, 三浦裕亮, 竹見哲也, 富田浩文: 湿潤対流の自己集合化の発生に関する新たな臨界長さ, 第22回非静力学モデルに関するワークショップ, オンライン, 2020年11月11日-12日

[24] Yanase, T., S. Nishizawa, H. Miura, T. Takemi, and H. Tomita: New Critical Length Scale for the Onset of Self-Aggregation of Moist Convection, JpGU - AGU Joint Meeting 2020, Online, 2020/07/15.

[25] Nishizawa, S.: Our model development activities and prospects in cloud resolving simulations, International Workshop Convection-Permitting Modeling for Climate Research Current and Future Challenges, Online, 2020/09/02. (Oral)

[26] Yanase, T., S. Nishizawa, H. Miura, T. Takemi, H. Tomita: New Critical Length for the Onset of Self-Aggregation of Moist Convection, AGU Fall Meeting 2020, Online, San Francisco, USA, 2020/12/01-17

[27] 末木健太 : 竜巻を生ずる台風の構造・環境場について, 竜巻シンポジウム - 藤田哲也博士生誕100年を記念して - , オンライン開催, 2021年3月11日

11.5.6 Software

[28] SCALE, Library and environment for Meteorological and Climate calculations,urlhttps://scale.riken.jp/.

11.5.7 Patents

Chapter 12

Complex Phenomena Unified Simulation Research Team

12.1 Members

Makoto Tsubokura (Team Leader)

Keiji Onishi (Researcher)

Rahul Bale (Researcher)

Kazuto Ando (Technical Staff)

Hsueh-Jui Lu (Student Trainee)

Tokimasa Shimada (Student Trainee)

Yuta Natsume (Student Trainee)

Ryoichi Kurose (Visiting Researcher)

Nobuyuki Oshima (Visiting Researcher)

Shigeru Okazawa (Visiting Researcher)

Akiyoshi Iida (Visiting Researcher)

Huilai Zhang (Visiting Researcher)

Chung-Gang Li (Visiting Researcher)

Leif Niclas Jansson (Visiting Researcher)

Koji Nishiguchi (Visiting Researcher)

Yuji Wada (Visiting Researcher)

12.2 Overview of Research Activities

The objective of our research team is to propose a unified simulation method of solving multiple partial differential equations by developing common fundamental techniques such as the effective algorithms of multi-scale phenomena or the simulation modeling for effective utilization of the massively parallel computer architecture. The target of the unified simulation is supposed to be complex and combined phenomena observed in manufacturing processes in industrial cycles. Our final goal is to contribute to enhance Japanese technological capabilities and industrial process innovation through the high-performance computing simulation.

Most of the complex flow phenomena observed in manufacturing processes are relating to or coupled with other physical or chemical phenomenon such as turbulence diffusion, structure deformation, heat transfer, electromagnetic field or chemical reaction. While computer simulations are rapidly spreading in industry as useful engineering tools, their limitations to such coupled phenomena have come to realize recently. This is because of the fact that each simulation method has been optimized to a specific phenomenon and once two or more solvers of different phenomena are coupled for such a complicated target, its computational performance is seriously degraded. This is especially true when we utilize a high-performance computer such as Fugaku. In such a situation, in addition to the fundamental difficulty of treating different time or spatial scales, interpolation of physical quantities like pressure or velocity at the interface of two different phenomena requires additional computer costs and communications among processor cores. Different mesh topology and hence data structures among each simulation and treatment of different time or spatial scales also deteriorate single processor performance. We understand that one of the keys to solve these problems is to adopt unified structured mesh and data structure among multiple simulations for coupled phenomena. As a candidate of unified data structure for complicated and coupled phenomena, we focused on the building-cube method (BCM) proposed by Nakahashi [1].

In summary, the overview of Research Activities in FY2020 was as followings.

In the research activity of the Fugaku COVID-19 Project: prediction and countermeasure for virus droplet infection under the indoor environment, we have successfully modeled the flow and droplets and successfully assessed the risk of droplet infection in various environments. We have shown that in an outdoor environment, the risk of infection changes significantly depending on the wind direction.

In the live house analysis, the ventilation volume was only about half of the design value, and it was shown that the indoor ventilation environment may not be evaluated by empirical rules alone. Face mask analysis showed that the mask had a certain effect on the suppression of splashing and protection from inhalation. Various analyzes of the transport sector have been able to quickly provide an indication of ventilation performance in different environments according to social demands.

In the development and analysis of the multi-physics complex fluid-solid simulation framework, we have modified the conventional full Eulerian methods, the velocity gradient tensor is used to compute solid deformation. It is difficult to compute deformation of solid accurately near the interface, where the velocity between fluid and solid changes drastically. So, in order to overcome this problem, we have introduced Reference Map Technique(RMT) into our Eulerian formulation for fluid-structure interaction problems using Lagrangian marker particles. It is confirmed that the proposed method can obtain good agreement results through a benchmark problem.

In sibilant noise analysis as an application of the framework, by analyzing the real human vocal tract replica with complicated shapes, the underlying insights necessary to design dental prostheses has been provided for the production of sibilant fricatives.

In the research of speed-up and scale-up of fundamental methodology targeting Fugaku system, due to the tuning results, the peak performance ratio has improved from the original 7.03% to 13.95% for 1 CMG. And the weak scaling performance of 27,648 nodes per node was 91.07%, it was confirmed that the scale of calculation can be expanded to at least 21.7 billion cells and 4.5 PFLOPS.

In the basic research for constructing a flow solution method using a neural network, we have successfully constructed a contraction model of the three-dimensional flow field around the cylinder using the mode division by CNN and the contraction model simulation by LSTM. We performed and evaluated the methods by large-scale distributed learning on Fugaku.

[1] Nakahashi, K., “Building-cube Method for Flow Problems with Broadband Characteristic Length,” In: Armfield S., Morgan R., and Srinivas K., editors. Computational Fluid Dynamics. (Springer, 2002, 2003), pp.77-81.

12.3 Research Results and Achievements

12.3.1 Fugaku COVID-19 Project: Prediction and Countermeasure for Virus Droplet Infection under the Indoor Environment

Virus droplet infection caused by sneezing, coughing, or talking is strongly influenced by the flow, temperature, and humidity of the air around an infected person and potential victims. Especially in the case of the new coronavirus, the possibility of aerosol infection by atomized droplets is suggested in addition to the usual droplet infection. Because smaller aerosol particles drift in the air for a longer time, it is imperative to predict the scattering route and to estimate how surrounding airflow affects the infection so that the risk of droplet infection can be properly assessed, and effective measures to reduce infection can be proposed. In this project, massively parallel coupling simulation of virus droplet scattering, with airflow and heat transfer under the indoor environment such as inside a commuter train, offices, classrooms, and hospital rooms will be conducted. By taking into account the characteristics of the virus, its infection risk of virus droplets is assessed under various conditions. Then countermeasures to reduce the risk are proposed from a viewpoint of controlling the airflow. This project is a collaboration with RIKEN, Kyoto Institute of Technology, Kobe University, Osaka University, Toyohashi University of Technology, and Kajima Corporation. Complex Unified Simulation framework called CUBE, developed at RIKEN R-CCS and implemented on the supercomputer Fugaku is mainly used, which will be the world-largest and highly accurate virus droplet simulation ever conducted.

12.3.1.1 Modeling of droplet dispersion and evaporation and its applications

(a) Governing Equations

The numerical formulation employed for modeling droplet dispersion in this work comprises a combination of a Lagrangian and an Eulerian frame of reference. The discretized equations of mass, momentum and energy conservation reside on the Eulerian meshed. The equations for conservation of evaporated phase of liquid droplets and species of the gas phase are also solved on the Eulerian mesh. The liquid droplet dynamics equations are solved on the Lagrangian frame. Details of the liquid droplet model will be presented in Section 12.3.1.1. The conservation equations in compact vector notation are given by

$$\frac{\partial \mathrm{U}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{S}, \tag{12.1}$$

where the vector U represents the primitive flow variables and F holds the convective and diffusive terms, and they are given below [1].

$$\mathrm{U} = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ \rho e \\ \rho Y_k \end{pmatrix}, \quad F_i = \begin{pmatrix} \rho u_i \\ \rho u_i u_1 + P\delta_{i1} - \mu A_{i1} \\ \rho u_i u_2 + P\delta_{i2} - \mu A_{i2} \\ \rho u_i u_3 + P\delta_{i3} - \mu A_{i3} \\ \rho(\rho e + P)u_i - \mu A_{ij} u_j + q_i \\ \rho u_i Y_k - \rho \hat{u}_i^k Y_k \end{pmatrix}. \tag{12.2}$$

Here, the density of the gas is represented by ρ and the viscosity is given by μ. u, e and P represent the velocity, the total specific energy and the pressure, respectively. Species mass fraction and diffusion velocities of the k^{th} species are given by Y_k and $\hat{u}_i^k$, respectively. (u_1, u_2, u_3) are the components of the velocity field u along the principal directions $1, 2, 3$. Definition of the total specific energy is given by $e = \frac{P}{\gamma - 1} + \frac{1}{2} u_i u_i$ where γ is the ratio of the gas specific heat capacities. The heat flux q is given by $\mathrm{q} = -\lambda \nabla T$, in which $T\&\lambda$ represent temperature and thermal diffusivity, respectively. The diffusion velocity $\hat{\mathrm{u}}^k Y_k$ of the k^{th} species is defined in terms of the species diffusivity D_k, the relationship is given by $\hat{\mathrm{u}}^k Y_k = D_k \nabla Y_k$.

The source term in the governing equation arises due to the coupling with the liquid droplet model. The evaporation, heat exchange and momentum of the liquid droplets give rise to source terms to the mass, momentum, energy and droplet vapour species. The source term vector is given by $\mathrm{S} = (\mathrm{S}_\rho, \mathrm{S}_{\rho \mathrm{u}_1}, \mathrm{S}_{\rho \mathrm{u}_2}, \mathrm{S}_{\rho \mathrm{u}_3}, \mathrm{S}_{\rho e}, \mathrm{S}_{\rho \mathrm{Y}_\mathrm{k}})$. The terms S_ρ, $S_{\rho \mathbf{u}}$ and $S_{\rho e}$ are the source terms to the mass, momentum and energy equations. Of the species source terms $S_{\rho Y_k}$, the non-droplet vapor species are zero. These terms will be discussed in detail in the following section.

(b) Droplet Model

The single droplet model has been widely used in the literature for modeling the evaporation and dispersion of liquid droplets. In this approach, the evaporation and transport of each droplet are tracked separately. While the droplet-droplet interactions are most often ignored, the model focuses primarily on the interaction of each droplet with the bulk gas phase and solid surface/geometries[2]. The equations governing the transport, dispersion and settling of the droplets are presented below.

$$\frac{d\mathrm{x}_d}{dt} = \mathrm{u}_d, \tag{12.3}$$

$$\frac{d\mathrm{u}_d}{dt} = \frac{3C_d}{4d_d}\frac{\rho}{\rho_d}(\mathrm{u}-\mathrm{u}_d)\,|\mathrm{u}-\mathrm{u}_d|, \tag{12.4}$$

$$C_d = \begin{cases} \frac{24}{Re_d}\left(1+1/6Re_d^{2/3}\right) & Re_d < 1000, \\ 0.424 & Re_d > 1000, \end{cases} \tag{12.5}$$

$$Re_d = \frac{\rho(\mathrm{u}-\mathrm{u}_d)d_d}{\mu}. \tag{12.6}$$

In the above equations, u_d is the droplet velocity, x_d is the droplet position, the droplet drag coefficient C_d is expressed a function of the droplet Reynolds number Re_d and d_d is the droplet diameter. While a general expression for droplet drag coefficient has been presented above, in the application of interest in this work, i.e. sputum droplet dispersion, the droplet Reynolds numbers for the majority of the droplets are expected to be within 1000. The evaporation of the droplet is modeled by the energy balance equation to track the droplet surface temperature and the evaporation mass flux equation to track the mass of the droplet, which are given below.

$$\frac{dT_d}{dt} = \frac{Nu}{3Pr}\frac{c_p}{c_l}\frac{f_1}{\tau_d}(T-T_d) + \frac{1}{m_d}\left(\frac{dm_d}{dt}\right)\frac{L_V}{c_{p,d}}, \tag{12.7}$$

$$\frac{dm_d}{dt} = \dot{m}_d. \tag{12.8}$$

Details of the various parameters involved in the droplet model can be found in the work of Bale et al.[3].

(c) Boundary and initial conditions

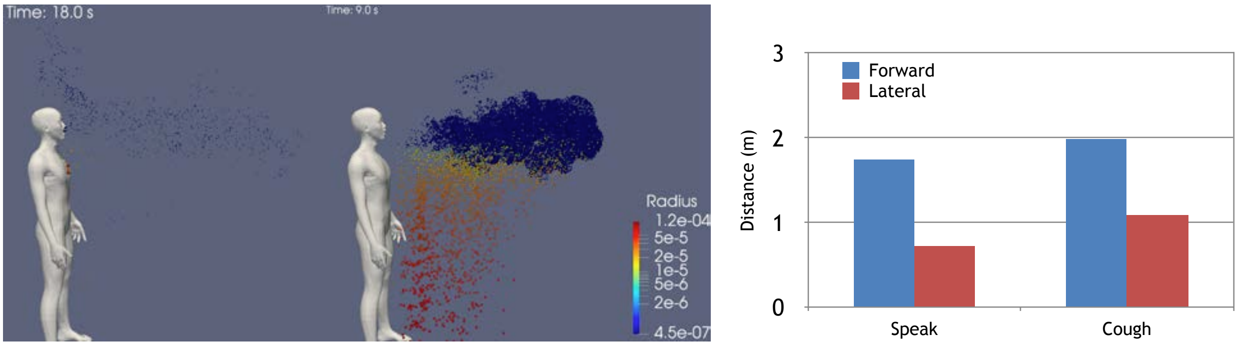
For all cases presented in this work the ambient condition of the air was approximately set to STP conditions: Humidity, Temperature and pressure were set to 60% relative humidity, 297 K and 1 atm, respectively. Stagnant ambient air is used as an initial condition for the simulations. The initial conditions of the droplets at the time of injection during a coughing event is as follows. The initial velocity, density, temperature of the droplets were chosen to be 0 m/s, 1000 kg/m^3 and 308 K. For the details of the flow velocity and flow rate, droplets size distribution during cough and speech the reader is referred to Bale et al. [3] in which these conditions are elucidated in depth.

A circular mouth of area 4 cm^2 [4] is used to model the mouth opening for cough simulations. For speaking simulations the mouth area was taken to be 6 cm^2. The mouth model is placed approximately 1 cm in front of the mouth of the human models in all the simulations presented in this work. The cough or speech flow boundary condition is imposed through the mouth model. The droplets are randomly injected 3 grid cells in front of the mouth model with zero initial velocity. Relative humidity of 90% and temperature of 307 K was specified for the flow emanating from the mouth model to account for the higher humidity and temperature of the air exhaled through mouth and nose.

12.3.1.2 Cough and speech in a quiescent environment

The first case we consider is the investigation of the dispersion of droplets during cough and speech in an environment in which the ambient fluid is motionless. The setup consists of a person in a standing pose subject modelled to cough and speak. The height of the human model is 1.7m. The human model is placed in a computational domain measuring $16 \times 16 \times 16$ m^3 with the feet of the human model coinciding with the centroid of the computational domain. A no-slip and isothermal boundary condition with $T = 307$ K is imposed on the human model through IB modelling. The slip boundary condition is applied to the computational domain boundaries. In the region where the cough flow emanates and where the droplets are injected into the domain, a mesh resolution is 0.5 mm, which is rapidly decreased to 4 mm. The 4 mm mesh spacing is extended over a distance of 1m in front of the mouth in the expected path of the sputum droplets. The mesh is made up of 4593 cubes and 18,812,928 cells.

A visualization of the droplet dispersion after a cough and during speech is shown in Fig. 12.1a. It is seen that large droplets start settling under the influence of gravity and don't travel very far in both cases. However,



(a) (b)

Figure 12.1: a)Left: Dispersion of sputum droplets after 18 s of speech. Right: Dispersion of droplet after a cough. (b)Distance traversed by the droplets 18s after the initiation of cough and speak expiration events in forward and lateral direction.

medium and small-sized droplets manage to travel longer distances. Particularly, the aerosolized droplets are not influenced by the flow and travel as far as the flow carries them. In Fig. 12.1b, the maximum distance traversed by the droplet in the front and lateral directions is presented. For the speaking case, the droplets traverse about 1.74 m and 0.71 m along with the front and lateral directions. In the cough case, these values are 1.98 m and 1.09m in the front and lateral directions.

12.3.1.3 Droplet dispersion in an outdoor environment

We consider the dispersion of droplets in a real-world outdoor environment for the second application. The setup consists of a group of 10 people huddled around a barbecue table. One out of the 10 individuals assumed to be infected with a disease, such as COVID-19, that can be transmitted through respiratory droplets. The dimensions of the table between the human models and the distance between the models are shown in the figure. All the human models are identical and measure 1.7 m in height. A relatively large computational domain is employed to model the effects of an outdoor environment. The computational domain extents are −250 m to 250 m along the x, and y directions, and from −50 m to 200 m along the z-direction. The vertical direction is along the z-axis with gravity acting along the negative z-axis. The centroid of the computational domain coincides with the feet of the infected person. A rectangular planar geometry measuring 50 × 50 m^2 with its normal along the z-axis is used to model the ground. The centre of the ground geometry is aligned with the centroid of the computational domain. At the location of the mouth of the infected person, the mesh spacing is 1 mm which progressively increases to 16 mm in the spatial region around the table. The mesh spacing on all the human geometries and the table is 16 mm. At all other regions, the mesh is progressively coarsened to contain the overall mesh size. The mesh was made up of 9552 cubes and 39,124,992 cells.

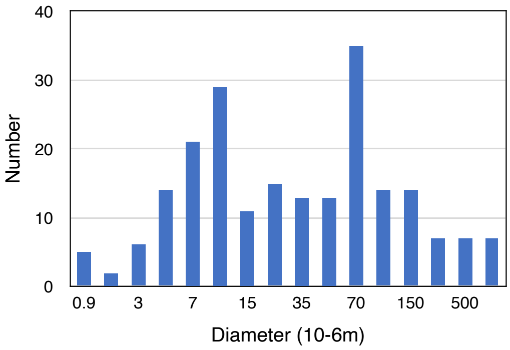
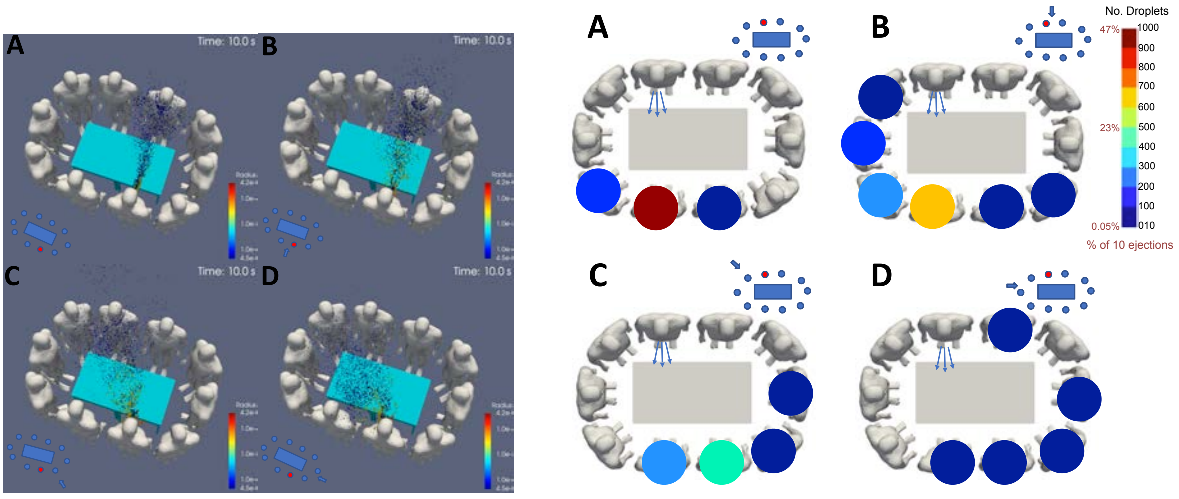


Figure 12.2: Droplet diameter distribution for speech at louder volume in outdoor environment.

In an outdoor barbecue type environment, individuals would likely tend to speak with a louder voice. A consequence of this is that the overall flow generated at the mouth is likely to be higher. To account for this we increase the flow rate for the speaking case by 50 percent. Furthermore, a tendency commonly observed when speaking loudly is also an increase in the number and size of the sputum droplets being ejected. The droplet diameter distribution model is shown in Fig. 12.2.



(a) (b)

Figure 12.3: (a)Visualization of droplet dispersion under A) no wind, and wind velocity B) $\mathbf{u} = (0, 0.5, 0)$ m/s, C) $\mathbf{u} = (-\sqrt{1/2}, \sqrt{1/2}, 0)$ m/s, D) $\mathbf{u} = (-0.5, 0, 0)$ m/s. (b) Quantitative evaluation of risk of each individual around the infected person.

The droplet dispersion was investigated under the conditions of no wind and varying wind direction: A) no wind, B) $\mathbf{u} = (0, 0.5, 0)$ m/s, C) $\mathbf{u} = (-\sqrt{1/2}, \sqrt{1/2}, 0)$ m/s, and D) $\mathbf{u} = (-0.5, 0, 0)$ m/s. Before the initialization of the speaking model in the simulation, the simulation was carried out for a duration of 5s to develop the flow due to natural convection around the human model. Thereafter, the speaking model was initialized during which the droplets are continuously injected according to the droplet model until the end of the simulation. A visualization of the droplet dispersion under different wind conditions is shown in Fig. 12.3a. The direction of droplet dispersion is strongly dependent on the wind direction. And, depending on the direction of the wind, the individuals who might be exposed to the virus-laden droplets varies.

To assess the risk of infection of an individual upon exposure to the droplets, the total number of droplets that might be within the vicinity of an individual was measured. For this, the breathing zone around the head of each human subject was defined. The total number of droplets within the breathing zone of each individual was measured at an interval of 1s after the initialization of the speaking model for a duration of 20s (i.e. 20 measurements in total). The maximum value of these measurements for each subject was quantified as the risk. This quantity is presented in Fig. 12.3b. In case A, only three subjects directly in front of the infected person are at risk of infection. In case B, the presence of the wind tends to disperse the droplets laterally, increase the number of subjects being exposed to the droplets. In cases C and D, due to the change in the direction of the wind, the subjects who are at risk of infection change accordingly.

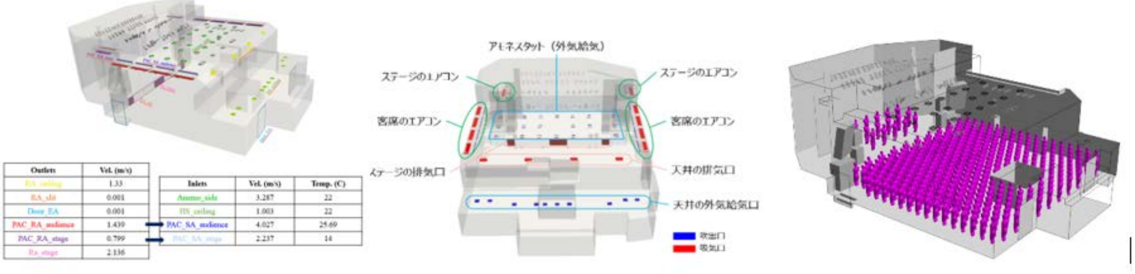
12.3.1.4 An estimation of the indoor air quality for the live house

Estimating the indoor air quality to prevent the spread of viruses and other infections is in urgent need after the Coronavirus pandemic in March 2020. Due to the development of High Performance Computing (HPC), CFD has been widely applied as a major tool to investigate the indoor airflow for our living environments such as the office, music hall, or shopping mall. To estimate indoor air quality, the interaction between heat and fluid flow should be taken into consideration. Besides the well-awarded heat source in the indoor such as the human body, the significant heat transfer for example the gas stove, the LED light, and the heater make the interaction more complicated. To accurately capture the airflow, a compressible solver, which can deal with the density variation caused by the significant heat transfer, is necessary.

Therefore, an estimation framework based on the compressible solver in CUBE is developed in this research. The case of the live house has been studied to investigate the feasibility of the current framework for practical applications.

In order to estimate the ventilation volume in the room, the transport equation for scalar c was incorporated into the governing equation. For the initial condition of c, 1 is adopted, which means the air quality of the indoor environment is polluted. On the other hand, c for the fresh air from outdoor is set as 0, which means the air quality is totally clean. Besides, the portion of the fresh air adding to the ventilation system or the air conditioner can also be fixed to mimic the effect of the filter and using the scalar equation to calculate the transient state development of the air quality.

The air quality inside a live house is numerically predicted between with 360 audiences situations. The physical model and boundary setting are shown in Fig 12.4. To conduct the simulation for the whole building inlcuding the effect of the natural convection from the human bodies, the total grid number of around 135 million is used and the time duration reaches more than 5 minutes to develop the flow field to reach the quasi steady-state.



Outlets	Vel. (m/s)
RA_ceiling	1.33
RA_slit	0.001
Door_EA	0.001
PAC_RA_audience	1.439
PAC_RA_stage	0.799
Ra_stage	2.136

Inlets	Vel. (m/s)	Temp. (C)
Anemo_side	3.287	22
HS_ceiling	1.003	22
PAC_SA_audience	4.027	25.69
PAC_SA_stage	2.237	14

Figure 12.4: Computational setting and physical model

Fig 12.5 shows the isothermal surface of 24 °C contoured with the magnitude of the velocity, and the slice with the velocity and temperature contour. The full development of the flow can be clearly observed. The pattern of the airflow from the air conditioner (PAC-RA-audience and PAC-RA-stage) and the anemostat in the audience region (Anemo-side), can also be identified. The cold flow in the audience regions from the air conditioner is slightly lifted by the natural convection from the human body and the light. Because the number of the people inisde the live hourse is more than 360, the effect of the natural convection becomes very strong, which also indicates that the flow from the anemostat is difficult to reach the ground.

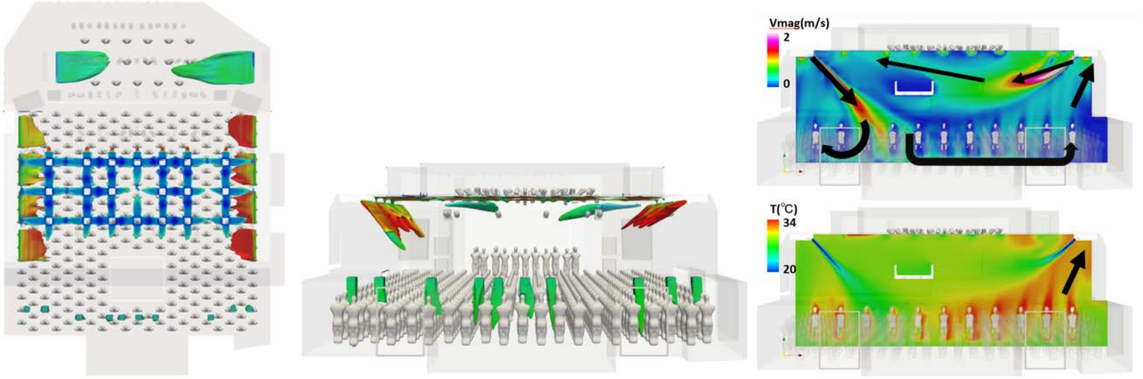


Figure 12.5: Flow field

To compare the different situations between with and without audiences, the case with no people insdie is also conducted. Fig 12.6 shows the comparison of the scalar c between with audience and without audience at t=100s and t=200s. 0 in the coloar bar represents the fresh air and 1 represents the dirty air. The fresh air is only from the ceiling inlet (HS-ceiling) and the anemostat (Anemo-side). Thus, the air in the rear of the live house close to the ceiling inlet is cleaned earlier than that in other places. On the other hand, there is no inlet for the fresh air on the stage so it takes the longest time to clean the air. Compared with no audience case, because of the strong effect of the natural convection from the human body and the light, the heat convection will be enhanced and the air quality will be cleaner. However, the air quality near the stage still can not be improved.

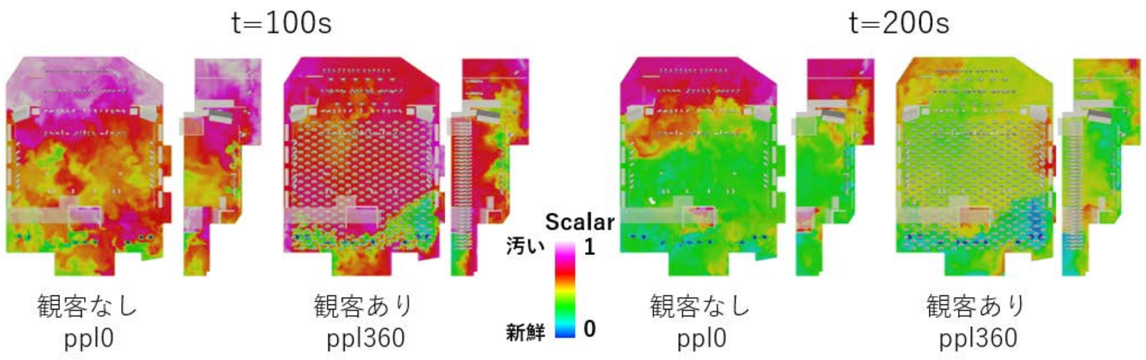


Figure 12.6: The scalar distribution

With the time history of the scalar, the decay of the scalar can be predicted and the ventilation flow rate can be numerically estimated. Fig 12.7 shows the time history for both cases.The distribution of the scalar can be approximated using an exponential curve. According to the exponential curves, the ventilation flow rates of audience and no audience cases are 11606(m^3/hr) and 13552(m^3/hr), respectively. Compared with the design ventilation flow rate, which is 17450 (m^3/hr), the performance of 66.5% in the practical operation estimated by the current framework is shown. On the other hand, when the live event is held and there are 360 audiences inside the live house, the performance of 77.66% can be achieved. The ventilation flow rate for the second case can be forward divided by the total number of audience 360, the ventilation flow rate for each people, 37.64 m3/hr, can be estimated. Based on the regulation for the building, the necessary ventilation flow rate is 30 m3/hr, it can be known that the current design of the live house can obey the law.

The framework developed in this study can be applied to practical situations to numerical estimate the indoor air quality. Especially for the event with crowded people such as a live band, the experiment is hard to be conducted and the real sitiutation is also far from the original design. Thus, the imperial experience couldn't be applied in such situations. With the current framework, the problem can be solved and the simulation is closer to the real situations.

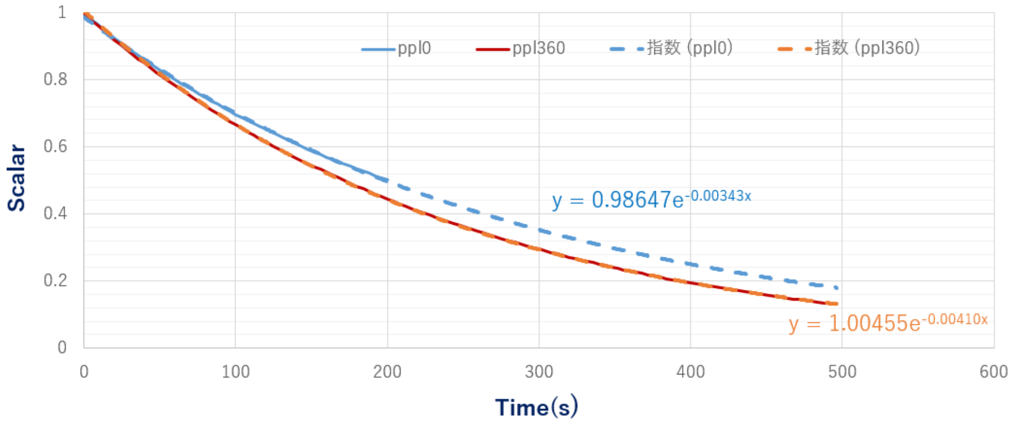


Figure 12.7: The time history of the scalar

12.3.1.5 Face mask simulation for efficiency study towards droplet infection

Accumulation of numerical research on the virus infection prevention effect of face masks, which is a basic protective device for people, is not sufficient. If we can show some index for this problem, which is still controversial in social life, it will be beneficial to society. In this research, the protective efficiency against droplets with face masks is analyzed using the topology-free immersed boundary method (IBM) [5] developed by this team. Since the complex flow path shapes around the face mask are difficult to reproduce with a computational grid using conventional methods, the method is advantageous to perform three-dimensional multiphase flow simulation on this topic.

First, we investigated the blocking effect under exhalation events of the single coughing condition. Figure 12.8 shows a volume-rendered visualization (Color indicates flow velocity) and particle distribution (Color indicates wall flag) of the three-dimensional flow field with a mask. The collection rate of particles captured by the face mask surface was modeled using probabilistic treatment using the measured transmissive rate in the experiment for each surface material. There are two types of flow: a flow that leaks from the gap between the face and the mask surface, and a flow that permeates through the pressure loss from the entire mask surface. The gap between the face and the mask forms a three-dimensional complex shape, and these are partially overlapped and narrow passed, however the calculation method used in this study does not cause any problems by the complexity of the flow channel. The initial particles placed in front of the mouth are diffused by the flow every moment. These are separated into those captured by the mask surface (stick), penetrate the surface (penetrate), and leak through the gap (fly). Some particles stuck on the face and someones flew to the region surrounded by the mouth and mask.

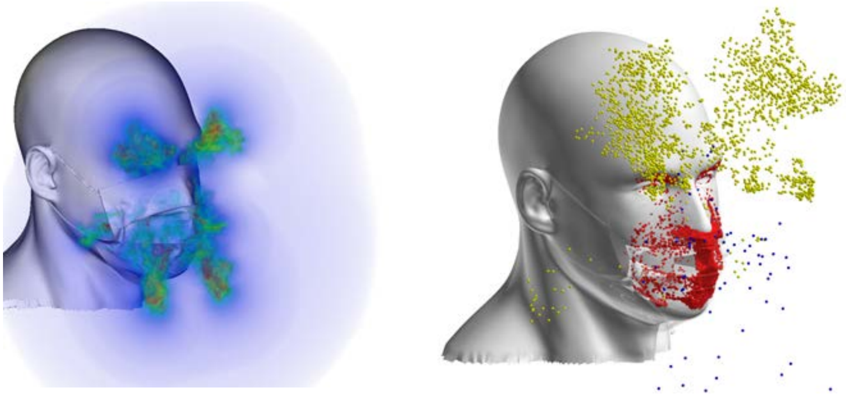


Figure 12.8: A volume-rendered 3D visualization(left) and a spray particles distribution colored by flag (yellow: fly, blue: penetrate, red: stick) (right).

From the states of the particles, the droplet transmission rate is calculated by counting each particle's numbers. Figure 12.9 compares the droplet collection rate (inverse of the transmission rate) for varias materials. The handmade mask is made of one layer of 100% polyester cloth, T-shirt/Y-shirt masks are made from T-shirt/Y-shirt material (mixed cotton and polyester) and the shape is the same as the surgical mask. The gauze mask and the N95 mask has their own shape.

In general, a surgical mask has about 60 to 70% of particles are collected as a whole. Compared with the cloth mask, the surgical mask has a high collection rate in large size particles. However, since the cloth mask has less leakage from the gap, the almost same efficiency has been obtained around the aerosol particle sizes. There is a large difference in performance depending on the material, i.e. pressure loss and transmittance, there is also a sensitivity to the mask shape, i.e. the gap size with the face. In the evaluation by volume ratio of droplets, the contribution of large particles is dominant, resulting in the collection rate of almost 100% in most cases. To limit the measurement range within the fine particles ($< 10\mu m$) of interest in COVID-19, the volume collection rate can be obtained as well as the particle number correction rate.

Next, we investigated the blocking effect under inhalation events. As an initial distribution, particles of 0.1 to 10 μm are evenly arranged around the face, and the number of particles that have invaded the trachea is counted by deep breathing for 6 seconds (2 s exhalation, 2 s inhalation, then 2 s exhalation). The efficiency was evaluated comparing with the result of the no mask case and performed under the conditions of double mask cases. Figure 12.10 shows the particle distribution after 6.0 s. In the figure, only the particles that have reached the trachea are visualized. Figures 12.11 show the mask collection efficiency calculated from the number of particles inhaled or exhaled by coughing. The "surgical_urethane" represents the case where the surgical mask is attached to the inside and the urethane mask is attached to the outside, and it shows the best collection rate of 95% or more. The outer urethane mask crushes the gaps in the inner surgical mask, especially on the side of the nose, the particles leaking from the gap have been suppressed. Compared to the "surgical (fit)" case in which only the surgical mask is fitted perfectly, the same performance is obtained even with the single mask configuration.

The results of this research have been shown in various media reports. We think it was an opportunity to

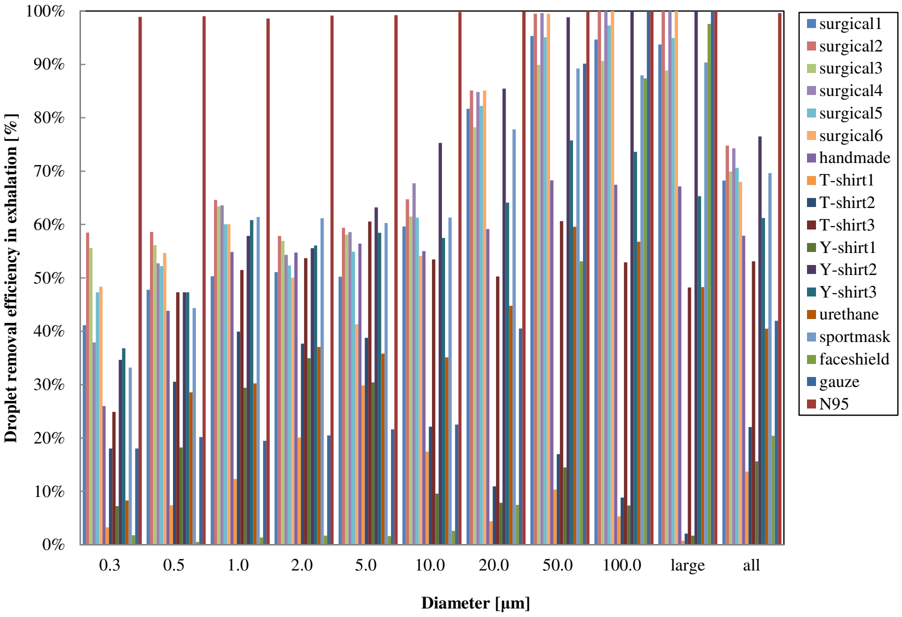


Figure 12.9: Comparison of the collection rates for each droplet size in exhalation.

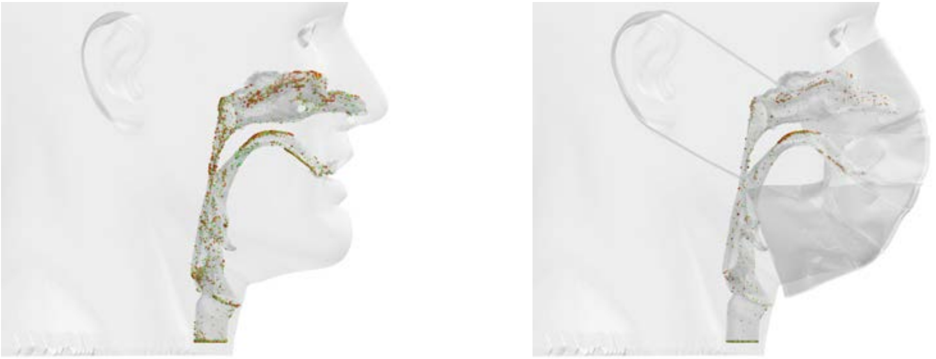


Figure 12.10: Droplet distribution after 6.0s colored by diameter: (left) no mask (right) surgical mask(fit).

deepen people's understanding of face masks. It would be honored if this could contribute to epidemiological measures to reduce the risk of infection.

12.3.1.6 Assessment of ventilation and droplet dispertion in the transport sector

Regarding the spread of aerosol droplets, which is one of the main transmission routes by SARS-CoV-2, attention is focused on the importance of ventilation in the transportation sector, which is a closed indoor environment. In this study, in response to requests from related organizations, we conducted ventilation and droplet diffusion analysis in traveling trains, taxis, and ambulances utilizing the topology-free immersed boundary method [5].

First, we investigated changes in ventilation volume due to the opening of windows on running commuter train. The train speed was 80 km/h and the window opening size is changed from 5 cm to 20 cm (4 places on the left and right side). The interaction between the air inside the vehicle and the outside air is calculated directly. Figure 12.12 shows a visualization of the typical instantaneous flow field and the content of fresh air rate. By solving the scalar distribution of substance transport, the ventilation volume can be evaluated from the history of scalar concentration. Figure 12.13 shows the results of scalar concentration obtained in 5 minutes. It shows that the ventilation volume increased almost in proportion to the window opening size.

Furthermore, the ventilation volume when the opening and closing of the entrance / exit door (about 30 seconds) was reproduced by the moving boundary. The result shows that it corresponded to the ventilation volume of about 5 cm window open case. This fact shows that sufficient ventilation was obtained on routes that

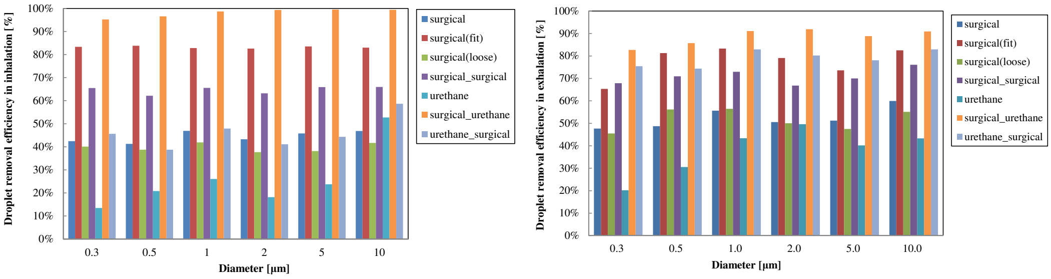


Figure 12.11: Comparison of the collection rates via droplet size in double-mask configurations (left) inhalation (right) exhalaton.

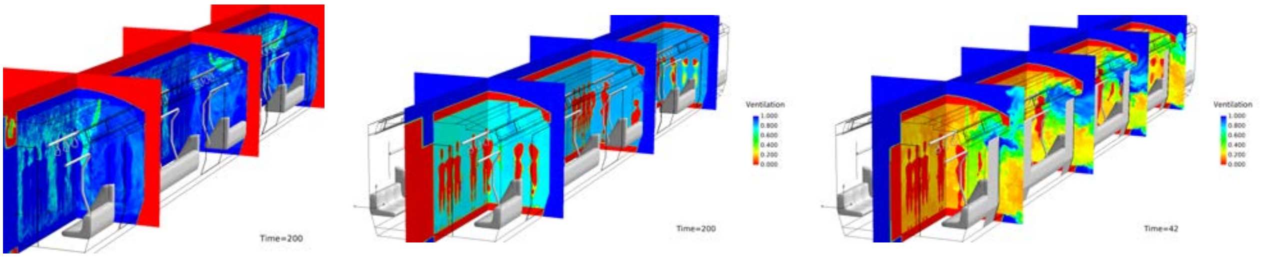


Figure 12.12: A typical visualization of the instantaneous velocity field(left) and a scalar concentration for window open 20 cm (middle) a scalar concentration for door open / close (right).

train stopped frequently, and vice varsa. Comparing the conditions in which the train has many passengers or sparse, the ventilation volume was slightly smaller due to the expansion of the occupied volume integral by the human body, however no significant difference in both cases.

Next, we investigated the ventilation performance inside the taxi cab and the ambulance. The effect of measures against droplet scattering by air-conditioning, opening windows or partitioning. This research is being carried out with the cooperation of the Ministry of Land, Infrastructure, Transport and Tourism, Toyota Motor Corporation, the Kobe Municipal Fire Department and Toyota C&D Co., Ltd. The vehicle speed is 40 km/h, and several peoples including the driver or patient are on board. The air conditioner set is premised on the introduction of outside air. The window openings were assumed to be set as 5 cm.

Figure 12.14 shows an example of the scalar distribution of contaminated air due to ventilation and the results of quantitative evaluation of the actual ventilation volume. It shows the mechanical ventilation effect of the air conditioner is dominant. The effect of increasing ventilation volume by opening windows differs depending on the vehicle shape, vehicle speed, window opening size and opening locations.

Figure 12.15 shows the analysis results of droplet scattering immediately after the patient coughed once and when using the curtain. Large droplets fell immediatelly and the aerosol diffuses slowly throughout the cabin room. In general, the strong current of the air conditioner resulted in the rapid diffusion of aerosols throughout the vehicle. Discharge by separator / curtain or window opening has a certain effect, however it was the most reasonable to encourage occupants to wear masks, as which can reduce the amount of particles emitted to about 30

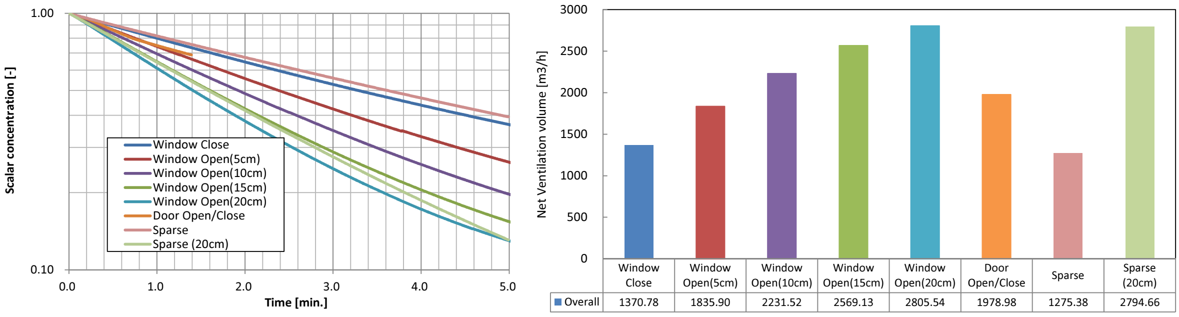


Figure 12.13: A history of scalar concentration (left) and a obtained ventilation volume for each cases (right).

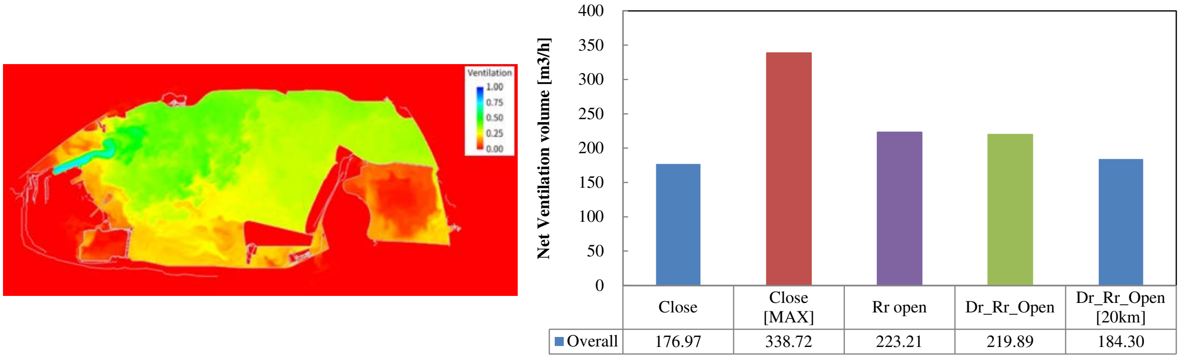


Figure 12.14: A typical visualization of a scalar concentration (left) and a obtained ventilation volume for each cases (right).

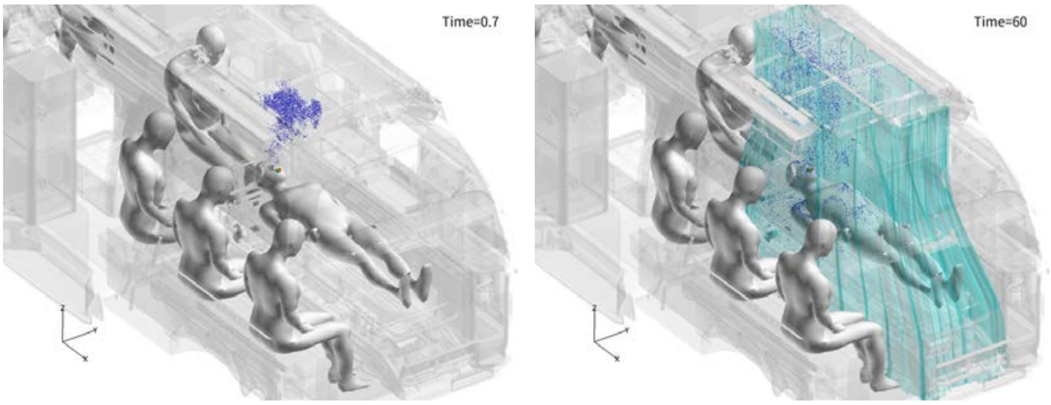


Figure 12.15: A snapshot a spray droplet distribution immediately after the patient's coughing (left) and after 60 s using separator curtain (right).

These results were able to summarize in a short period from the start of the initiative. We believe that we could disseminate information to society quickly and we could contribute to society's demand. In addition, we could show the superiority of applying our framework to complex problems.

12.3.1.7 References

[1] T. Poinsot, D. Veynante, Theoretical and numerical combustion, RT Edwards, Inc., 2005.

[2] T.Kitano,J.Nishio,R.Kurose,S.Komori,Effectsofambientpressure,gastemper- ature and combustion reaction on droplet evaporation, Combustion and Flame 161 (2) (2014) 551–564.

[3] R. Bale, C.-G. Li, M. Yamakawa, A. Iida, R. Kurose, M. Tsubokura, A scalable framework for numerical simulation of combustion in internal combustion engines, in: Proceedings of the Platform for Advanced Scientific Computing Conference, 2021, pp. 1–10. https://doi.org/10.1145/3468267.3470575

[4] J. Gupta, C.-H. Lin, Q. Chen, Flow dynamics and characterization of a cough, Indoor air 19 (6) (2009) 517–525.

[5] Keiji Onishi, Makoto Tsubokura, "Topology-free immersed boundary method for incompressible turbulence flows: An aerodynamic simulation for 'dirty' CAD geometry", Comput. Methods Appl. Mech. Engrg. Vol.378, 113734, (2021).

12.3.2 Development and analysis of the multi-physics complex fluid-solid simulation framework

12.3.2.1 Implementation of Reference Map Technique into Eulerian formulation for fluid-structure interaction problems using Lagrangian marker particles

Reference Map Technique(RMT)[1] is a computational method of solid deformation with a full Eulerian method. In the RMT, the Reference Map(the initial position vector of solid based on the reference configuration) is used to compute the deformation gradient tensor of solid. Thus, the velocity gradient tensor is not required to compute solid deformation with the RMT. However, in the original RMT[1], the advection equation of the Reference Map is used to update the spatial distribution of the Reference Map and the numerical dissipation of the Reference Map is unavoidable.

We have already developed the Eulerian method for fluid-structure interaction problems using Lagrangian marker particles to avoid numerical dissipation. Thus, in this fiscal year, we implemented an Eulerian method for fluid-structure interaction problems using Lagrangian marker particles with the Reference Map Technique without using the advection equation.

In our method, the mixture equation of continuity and the mixture equation of motion[2] are used to compute the motion of incompressible fluids and incompressible solids.

$$\nabla \cdot \boldsymbol{v}_{\mathrm{mix}} = 0$$

$$\frac{\partial \rho_{\mathrm{mix}} \boldsymbol{v}_{\mathrm{mix}}}{\partial t} + \nabla \cdot (\rho_{\mathrm{mix}} \boldsymbol{v}_{\mathrm{mix}} \otimes \boldsymbol{v}_{\mathrm{mix}}) = \nabla \cdot \boldsymbol{\sigma}_{\mathrm{mix}} + \rho_{\mathrm{mix}} \boldsymbol{b}$$

Here, $\boldsymbol{v}_{\mathrm{mix}}$ means the mixture velocity, $\boldsymbol{\rho}_{\mathrm{mix}}$ means the mixture density, $\boldsymbol{\sigma}_{\mathrm{mix}}$ means the mixture stress, and $\boldsymbol{b}$ is the body force. Figure 12.16 shows the flowchart of the computational method on an Eulerian mesh and Lagrangian marker particles. Based on our previous method, the governing equations of motion and spatial derivatives are computed on the Eulerian mesh and Lagrangian marker particles represent the solid region and carry solid internal variables(including Reference Map).

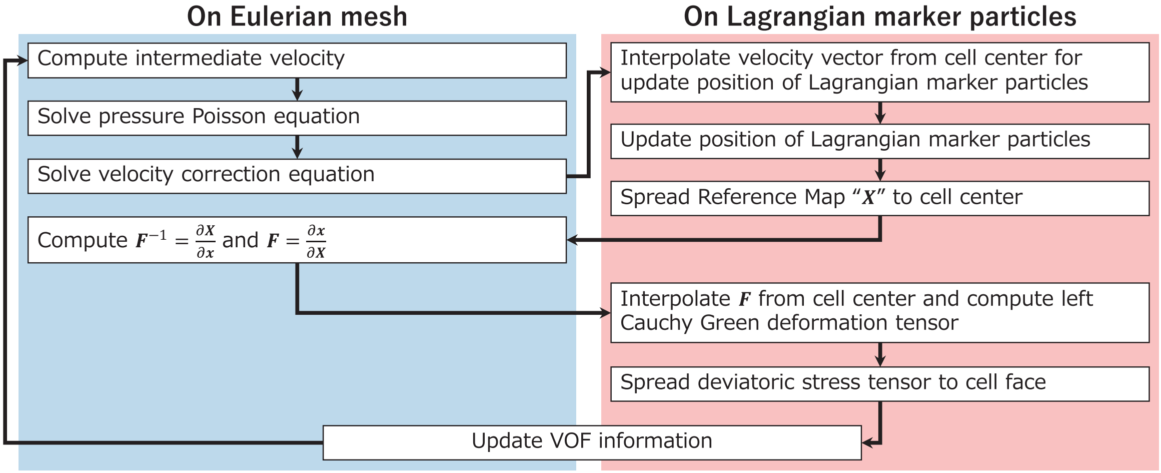

On Eulerian mesh
On Lagrangian marker particles
Compute intermediate velocity
Solve pressure Poisson equation
Solve velocity correction equation
Compute $\boldsymbol{F}^{-1} = \frac{\partial \boldsymbol{X}}{\partial \boldsymbol{x}}$ and $\boldsymbol{F} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}}$
Interpolate velocity vector from cell center for update position of Lagrangian marker particles
Update position of Lagrangian marker particles
Spread Reference Map "$\boldsymbol{X}$" to cell center
Interpolate $\boldsymbol{F}$ from cell center and compute left Cauchy Green deformation tensor
Spread deviatoric stress tensor to cell face
Update VOF information

Figure 12.16: Flowchart of proposed method in one computational step

12.3.2.2 Verification analysis by the benchmark problem

In order to confirm the validity of the proposed method, the benchmark problem[3] was conducted with the proposed method. The dimensionless material properties are given in Table 12.1 and the initial geometry is shown in Figure 12.17 At $t = 0$, the top cavity wall starts to move horizontally with velocity $v_x = 1.0$, and non-slip boundary conditions are imposed on the other walls.

Figure 12.18 shows the results of the proposed method using 128×128 computational cells(uniformly divided) and the reference solution[3]. From this figure, it is sufficient to use a 128×128 mesh for the proposed method to achieve good agreement with the reference solution.

12.3.2.3 Numerical investigation of effects of incisor angle on production of sibilant /s/

In this study, simulations of a realistic vocal tract geometry with different incisor angles was conducted to investigate the effect of the incisor inclination angle on the sibilant sound. The geometry of a vocal tract replica was constructed from CT images of a 32-year-old Japanese male who self-reported no speech disorders. To investigate the effects of the incisor angle, the incisor angle was varied from 0° to 30°.

Fig. 12.19 shows the normalized instantaneous velocity magnitude and root mean square (RMS) value of the velocity fluctuations of the 0° and 30° models on the mid-sagittal plane. The instantaneous velocities of

Table 12.1: Material properties of the benchmark problem

Solid: incompressible viscoelastic solid solid	
Mass density ρ_s	1.0
Shear elastic modulus G	0.1
Viscosity μ	0.01
Fluid: incompressible Newtonian fluid	
Mass density ρ_f	1.0
Viscosity μ	0.01

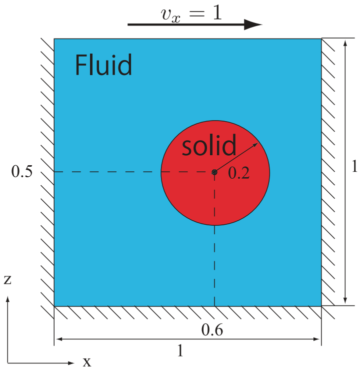


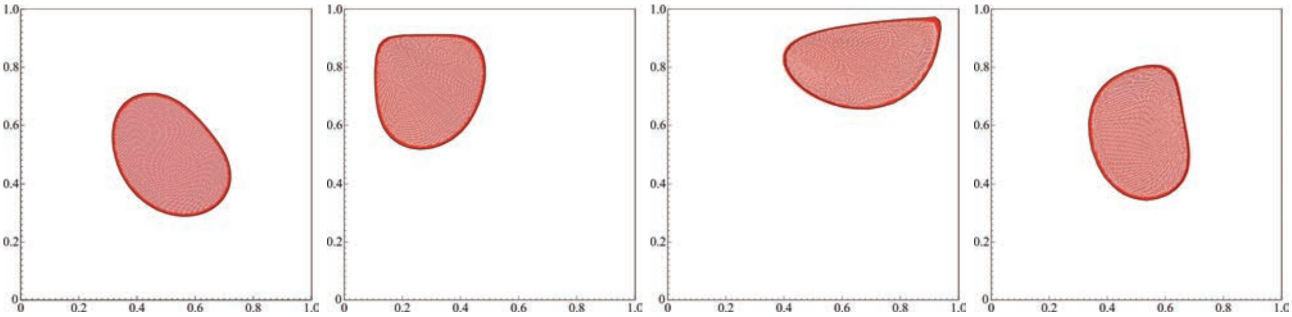
Figure 12.17: Initial geometry of the benchmark problem

both cases are accelerated at the narrow channel between the tongue and the hard palate (the sibilant groove). Downstream of the sibilant groove, the flow became turbulent at the region between the teeth and the lower lip as a result of the jet flow leaving the sibilant groove. However, because the exit of the sibilant groove became wider in superior direction with the inclined incisors, the reduction of the occlusion made the mainstream flow faster and the flow reached more distant positions. Hence, the high RMS region moved from the cavity between the lower incisor and the lip to the tip of the lower lip with increasing incisor angle. Meanwhile, the turbulence intensity was not affected by the raised incisor angle.

The far-field SPL spectra for different angles are investigated. Since the general audible frequency range for human speech is up to 9kHz, and the characteristic peak of the sibilant sound at around 4 kHz was observed for all teeth angles, these sounds generated by all the cases can be characterized as sibilant fricative /s/. However, the changes of incisor angle cause the amplitudes from 8 kHz to 12 kHz be decreased. This decreasing amplitudes in the mid-frequency led to a smaller noise band range and it might affect the recognition of the /s/ sound.

To identify the cause of the different SPL values for 8 kHz to 12 kHz between 0° and 30° in Fig. 12.20 , the positions of the potential sound sources for the 0° and 30° models were calculated. The magnitudes of the velocity fluctuations at specific frequencies (5 kHz and 10 kHz) were calculated via FFT on each grid. The magnitudes of the velocity fluctuations inside the vocal tract for the 0° model along the mid-sagittal plane are shown in Fig. 12.20 (a) at 5 kHz (x3 = 1.1 mm) and Fig. 12.20 (b) at 10 kHz (x3 = -7.7 mm), respectively. And the contours for the 30° model are at Fig. 12.20 (c) 5 kHz (x3 = 1.1 mm) and Fig. 12.20 (d) 10 kHz (x3 = -4.5 mm). At 5 kHz, the maximum value is located behind the incisor, which is the position jet flow emerged for both angles. Conversely, the maximum value at 10 kHz appeared at the cavity between the teeth and the lower lip for the 0° model, which corresponds to the exit of the jet flow generated by the gap between the teeth. On the other hand, the flow of the 30° model at 10 kHz passed along the surface of the incisor and the maximum value of the velocity fluctuations appeared above the lower lip.

Accordingly, it is clearly that the inclined angle of the incisor will change the flow configuration and shifted the position of the potential sound source, thereby affect the far-field sound spectrum. Specifically, if the



a) $t = 1.17$ b) $t = 3.52$ c) $t = 5.86$ d) $t = 8.20$

Figure 12.18: Comparison of the solid motion and deformation between the reference and proposed method

sound source position was located closer to the exit of the vocal tract, i.e., the lips, the source didn't couple strongly with the resonator, and no significant frequency could be caught at far-field. Because the flow channel downstream of the sibilant groove became wider when the incisor angle was increasing from 0° to 30°, the large velocity fluctuation region was shifted and the amplitude of the far-field sound around 10 kHz was reduced. These results provide the underlying insights necessary to design dental prostheses for the production of sibilant fricatives.

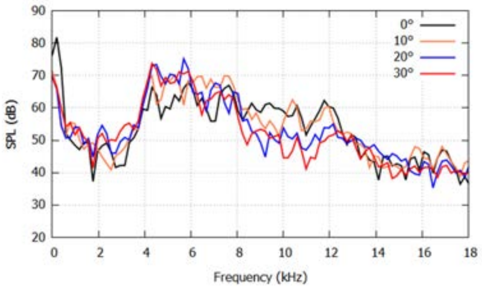


Figure 12.19: Sound pressure level (SPL) spectrum with incisor angles from 0° to 30°.

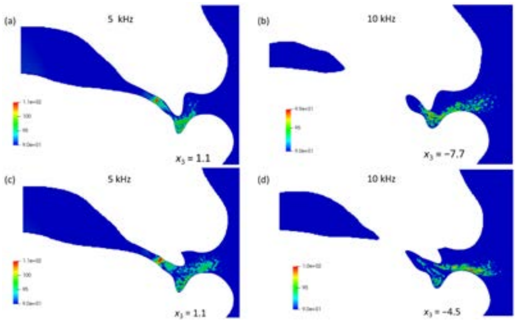


Figure 12.20: Fast Fourier transforms of the velocity fluctuations at (a, c) 5 kHz and (b, d) 10 kHz for the (a, b) 0° and (c, d) 30° models.

12.3.2.4 References

[1] K. Kamrin, C. H. Rycroft, J. C. Nave, Reference map technique for finite-strain elasticity and fluid-solid interaction. Journal of the Mechanics and Physics of Solids, 60(11), (2012), 1952-1969.
[2] K. Nishiguchi, R. Bale, S. Okazawa, M. Tsubokura, Full Eulerian deformable solid-fluid interaction scheme based on building-cube method for large-scale parallel computing. International Journal for Numerical Methods in Engineering, 117(2), (2019), 221-248.
[3] H. Zhao, J. B. Freund, R. D. Moser, A fixed-mesh method for incompressible flow-structure systems with finite solid deformations. Journal of Computational Physics, 227(6), (2008), 3114-3140.

12.3.3 Speed-up and scale-up of fundamental methodology targeting Fugaku system

12.3.3.1 Performance evaluation and tuning of CUBE on Fugaku

For the problem idealized for the performance benchmark, the result of executing CUBE on Fugaku and measuring its performance is shown. (Using a compressible flow solver).

The execution performance when the compressible solver of CUBE is operated on a single node of Fugaku is shown. As a problem for performance benchmarks, we deal with cavity flow in a region that extends long in the y-axis direction. This is set so that the area division by each MPI process is ideal (one-dimensional).

Table 12.2 shows the performance of the entire program (time integration loop). The performance of one node (4 MPI process) is 3.59 times that of 1 CMG (1 MPI process), which is generally better than the ideal performance improvement (4 times). The double-precision floating-point arithmetic performance is 59.96GFLOPS for 1 CMG and 215.84GFLOPS for 1 node, which are 7.09% and 6.38% of the peak performance, respectively, which are high values for CFD application performance.

Table 12.2: Time integration loop performance.

	Execution time	Calc. performance	Mem. throughput
	[seconds]	[GFLOPS]	[GB/sec.]
1 CMG	38.40	59.96 (7.09%)	48.44 (18.92%)
1 node	10.67	215.84 (6.38%)	171.90 (16.78%)

Table 12.3 shows the computational performance per CMG (1 process) of the high-cost viscous term kernel part and its upper limit estimate. The upper limit estimate was calculated based on the ratio of the amount of memory access of the source and system to the number of operations (Byte per Flop). Due to the tuning results, the peak performance ratio has improved from the original 7.03% to 13.95%. Estimates suggest a further 2.7% improvement.

Table 12.3: Viscosity term calculation performance measured value and upper limit estimated value (when running 2.0 GHz).

Original	Tuning	Estimated upper limit
53.87 GFLOPS (7.03%)	106.75 GFLOPS (13.95%)	128.07 GFLOPS (16.67%)

Table 12.4 shows the computational performance per CMG (1 process) of the high-cost convection term kernel part and its upper limit estimate. For the convection term, this kernel is not a memory throughput bottleneck because the ratio of the source memory access to the number of operations (Byte per Flop) is below the system value. Therefore, it is not possible to calculate the performance upper limit estimated value by Byte per Flop, but a significant performance improvement can be expected by computing performance tuning such as software pipelining and SIMD.

Table 12.4: Convection term calculation performance measured value and upper limit estimated value (when running 2.2GHz).

Original	Tuning	Estimated upper limit
82.23 GFLOPS (10.10%)	123.98 GFLOPS (14.70%)	- (-%)

Evaluate CUBE's weak scaling using idealized questions for benchmarking.

The cavity flow in the region extending in the y-axis direction per node is treated as a benchmark. When measuring scaling, the area is extended in the y-axis direction according to the number of nodes. As a result, this problem is divided into areas only in the y-axis direction, and communication is performed in the same one-dimensional pattern regardless of the number of nodes. In addition, it is guaranteed that the process responsible for the adjacent subregions will be mapped to the adjacent nodes in the space of TofuD's interconnect TofuD.

Table 12.5 shows the results of measuring the weak scaling from 1 node to 27,648 nodes of Fugaku using the above problem. Looking at the execution time, it is 27.24 seconds for 27,648 nodes compared to 25.48 seconds for one node, and the increase in execution time is suppressed to about 10%. As a result, the scaling of 27,648 nodes per node is 91.07%, which is a good value. In addition, it was confirmed that the scale of calculation can be expanded to at least 21.7 billion cells and 4.5 PFLOPS.

Table 12.5: Weak scaling results of ideal problem.

Nodes	Cells	Exec. time [seconds]	FLOPS [TFLOPS]	Scaling
1	786,432	25.48	0.18 (5.38%)	100.00%
256	201,326,592	26.94	44.00 (5.09%)	94.57%
512	402,653,184	26.92	88.09 (5.09%)	94.67%
1,024	805,306,368	26.86	176.55 (5.10%)	94.87%
4,096	3,221,225,472	27.05	701.37 (5.07%)	94.22%
8,192	6,442,450,944	27.07	1,401.68 (5.06%)	94.15%
16,384	12,884,901,888	27.24	2,785.88 (5.03%)	93.56%
27,648	21,743,271,936	27.98	4,575.74 (4.90%)	91.07%

The results of executing CUBE on Fugaku and measuring its performance for vehicle models are shown. (Using an incompressible fluid solver)

As evaluations of parallel performance, strong scaling (increasing the number of nodes by fixing the problem size) and weak scaling (increasing the number of nodes while fixing the problem size per node) were evaluated.

The result of weak scaling is shown. A vehicle model (130 million cells) was used for the calculation. The mesh is subdivided step by step so that the number of cells is eight times larger, and at the same time, the number of nodes is increased.

Table 12.6 shows the results of evaluating the change in elapsed time when the number of nodes is increased for the time integration loop (scaling is better as the elapsed time does not increase). The scaling at 51,200 nodes based on 100 nodes was 74%, and good results were obtained as the measurement results of the actual problem using the hierarchical grid.

Table 12.6: Weak scaling results of vehicle model

Nodes	Cubes	Cells	Time	Scaling
100	32,341	132,468,736	22.92 s	100.00%
800	260,331	1,066,315,776	32.63 s	70.23%
6,400	2,082,648	8,530,526,208	33.46 s	68.49%
51,200	16,661,184	68,244,209,664	30.927 s	74.11%

12.3.3.2 Basic research for constructing a flow solution method using a neural network

The ultimate goal is to build a contraction model of the flow field around the vehicle, and as a preliminary step, we are working on the construction of a contraction model of the three-dimensional flow field around the cylinder. Below, we show the mode division by CNN and the contraction model simulation by LSTM that we performed by large-scale distributed learning on Fugaku.

In this research, based on the neural network (MD-CNN-AE[1]) with an autoencoder type structure devised by the research group of Professor Koji Fukagata of Keio University, we made changes to apply it to 3D flow field data.

Figure 12.21 shows how to parallelize the learning process of the neural network. It is a method that combines the following two parallelization schemes (hybrid parallelism).

1. The weights and calculations of the encoder section and the weights and calculations of the branched parts for each mode of the decoder section are distributed to separate MPI processes (model parallel).
2. The training data is distributed to separate MPI processes, and the amount of weight updates calculated by each person is reduced (averaged) by communication (data parallelism).

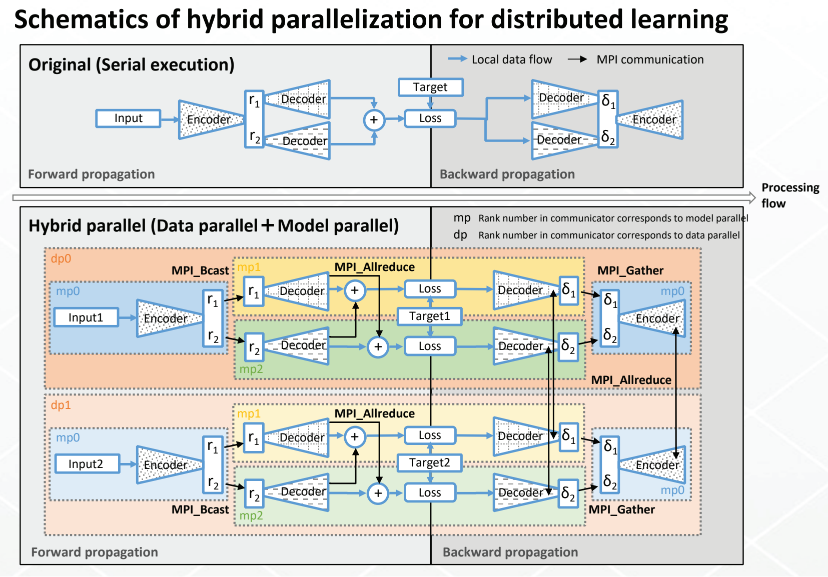


Figure 12.21: Parallelization of learning process.

LSTM, which is a type of RNN (Recurrent Neural Network), is a network that predicts the time change of latent vector by learning the time series of latent vector that is output when flow field data is given as input to the trained encoder. Implemented by (Long Short-Term Memory). Configure a network that takes the past 20 steps as input and outputs the past 19 steps + the next 1 step. As the network structure, we adopted a method of arranging LSTMs with different numbers of features in the Hidden state in parallel.

The result of the reduction simulation using the LSTM which learned the latent vector obtained by the mode division using CNN is shown.

Figure 12.22 shows a snapshot of the flow field u 500 seconds after the start of the simulation. Compared with the result of the high-precision model, the complicated flow structure of the wake of the cylinder cannot be reproduced in both the number of modes 2 and the number of modes 20, but the number of modes 20 has less phase shift.

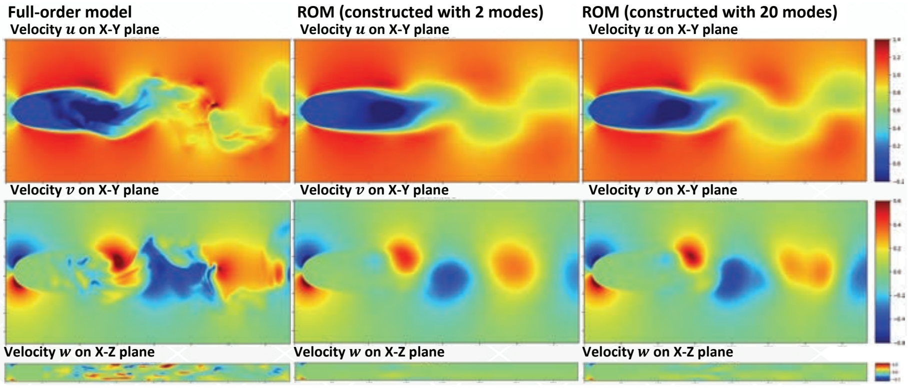


Figure 12.22: Snapshot of flow velocity u on the X-Y plane.

Figure 12.23 shows a snapshot of the Q-value isosurface 50 seconds after the start of the simulation. Compared with the result of the high-precision model, the complicated vortex structure of the wake of the cylinder cannot be reproduced in both the number of modes 2 and the number of modes 20. In addition, there is no clear difference between the vortex structures of mode numbers 2 and 20. This is considered to indicate that even 20 modes are significantly insufficient to reproduce the complicated vortex structure of the high-precision model.

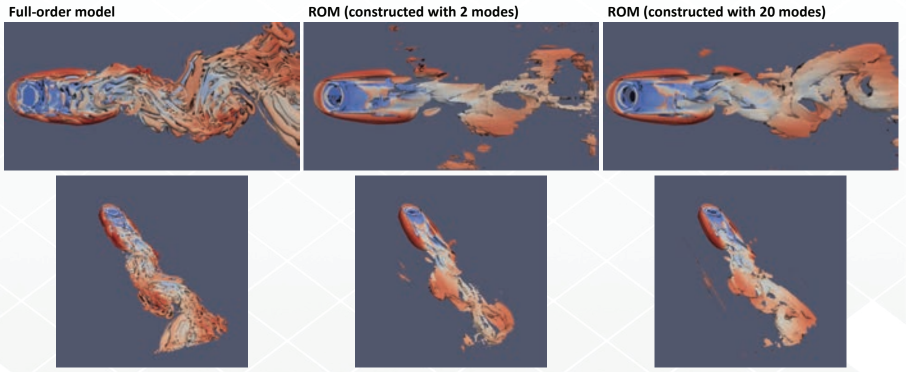

Full-order model
ROM (constructed with 2 modes)
ROM (constructed with 20 modes)

Figure 12.23: Q-value isosurface snapshot.

12.3.3.3 References

[1] T. Murata, K. Fukami, and K. Fukagata, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," J. Fluid Mech. 882, A13 ,2020.

12.4 Schedule and Future Plan

In the research and development of unified simulation framework 'CUBE' from 2012, the following goals has been achieved:

1. Construction and development of the simulation technology for bringing out the performance of Fugaku/K-computer

2. Validation and practical usage of industrial applications such as vehicle aerodynamics, aeroacoustics, unsteady motion aerodynamics, engine combustion, city area wind environment analysys, sport CFD, droplet dispersion analysis, and structure analysis

3. Preparation of the simulation technologies of HPC toward EXA-scale

In the long-term objectives, the following target is considered:

1. Establishment of the research and development center for industrial simulation technology

2. Contribution to computer science by expanding the developed simulation technology to different fields

In terms of the deployment of the framework, we have successfully applied to the flow and droplet dispersion analysis for the Fugaku COVID-19 Project: prediction and countermeasure for virus droplet infection under the indoor environment. The key technology was computational availability for complex shapes, which in a short period of time carried out a huge number of study cases and were recognized for our contribution to social epidemiological activities.

In terms of fluid-structure interaction analysis, we are going to conduct basic verifications and validation of our proposed method through several benchmark problems, and we are planning to propose a new method to impose velocity boundary conditions for arbitrary positions and shapes in a computational domain with our Eulerian method.

In terms of developing surrogate modeling, we have successfully constructed a contraction model of the three-dimensional flow field around using the mode division by CNN. By accumulating basic research, we plan to improve prediction accuracy and increase the scale of learning.

From collaboratetive companies, including companies through the activities of the "Consortium for Next Generation Automotive CAE using HPC" and "Consortium for Next Generation Combustion CAE using HPC" organized by RIKEN, voices of surprises and expectation for software development have been received. We are planning to continue development for practical application in near future.

12.5 Publications

12.5.1 Articles/Journal

[1] Keiji Onishi, Makoto Tsubokura, "Topology-free immersed boundary method for incompressible turbulence flows: An aerodynamic simulation for 'dirty' CAD geometry", Comput. Methods Appl. Mech. Engrg. Vol.378, 113734, (2021).

[2] Bale R., Patankar N. A., Jansson N., Onishi K., & Tsubokura M. (2020). Stencil Penalty approach based constraint immersed boundary method. Computers & Fluids, 200, 104457.

12.5.2 Conference Papers

[1] Bale R., Wang W. H., Li C. G., Onishi K., Uchida K., Fujimoto H., Kurose R., & Tsubokura M. (2020, June). A Scalable Framework for Numerical Simulation of Combustion in Internal Combustion Engines. In Proceedings of the Platform for Advanced Scientific Computing Conference (pp. 1-10)

[2] Cho Y., Bale R., ODA T., & Oshima N., Application of Compressible Immersed Boundary Method on a Flow of a Fuel-Injector of an Aircraft Engine, The Proceedings of Mechanical Engineering Congress, Japan, J05112, 2020

[3] Kajimoto H., Oyama S., Bale R., Yamamoto K., & Tsubokura M., Aerodynamics simulation framework for Ski-jumping take-off and its application to various jumper, Nagare : journal of Japan Society of Fluid Mechanics F09-1, 2020

[4] Bale R., Desai S., LI CG., Iida A., Yamakawa M., Kurose R., & Makoto T., Numerical simulation of evaporation and dispersion of cough droplets in the air, Nagare: Journal of Japan Society of Fluid Mechanics. D01-03, 2020

[5] Koji Nishiguchi, Tokimasa Shimada, Masafumi Otaka, Christian Peco, Shigenobu Okazawa, Makoto Tsubokura, Voxel-Based Fluid-Structure Interaction Method for Nonlinear Biomaterials Involving Large Deformation, Proceedings of 14th World Congress on Computational Mechanics (WCCM) ECCOMAS Congress, paper ID: 1795, (2020).

[6] Masafumi Otaka, Koji Nishiguchi, Tokimasa Shimada, Makoto Tsubokura, Hirofumi Sugiyama, Shigenobu Okazawa, Large Deformation Solid Dynamics Using Marker Particles Based on Eulerian Formulation, Proceedings of 14th World Congress on Computational Mechanics (WCCM) ECCOMAS Congress, paper ID: 4673, (2020).

[7] Koji Nishiguchi, Tokimasa Shimada, Masafumi Otaka, Shigenobu Okazawa, Makoto Tsubokura, Eulerian finite volume formulation using reference map technique for structure analysis, In: Proceedings of the 25th Conference on Computational Engineering and Science, paper No. F-06-03, (2020).

[8] Tokimasa Shimada, Koji Nishiguchi, Shigenobu Okazawa, Makoto Tsubokura, Eulerian Unified Formulation for Fluid-Structure Interaction Problems with Using Reference Map Technique, In: Proceedings of the 25th Conference on Computational Engineering and Science, paper No. D-03-02, (2020).

[9] Yuji Wada, Tokio Ogawa, Tokimasa Shimada, Koji Nishiguchi, Masafumi Otaka, Shigenobu Okazawa, Makoto Tsubokura, Structural topology optimization under particle-in-cell method using hierarchical Cartesian mesh, In: Proceedings of the 25th Conference on Computational Engineering and Science, paper No. A-06-01, (2020).

[10] Masafumi Otaka, Koji Nishiguchi, Tokimasa Shimada, Makoto Tsubokura, Hirofumi Sugiyama, Shigenobu Okazawa, Development of Large Deformation Elastoplastic Analysis by using Marker Particles based on Fixed Cartesian Mesh, In: Proceedings of the 25th Conference on Computational Engineering and Science, paper No. D-07-04, (2020).

12.5.3 Oral Talks

[1] Kazuto Ando, Kiyoshi Kumahata, Kazuo Minami, Keiji Onishi, Li Chung-Gang, Makoto Tsubokura, and Jun Ikeda, Performance evaluation and tuning of compressible fluid solver of CUBE on Fugaku, 34th Symposium on Computational Fluid Dynamics (Web, Dec. 21-23, 2020), Paper C03-3.

[2] Kazuto Ando, Keiji Onishi, Rahul Bale, Makoto Tsubokura, Akiyoshi Kuroda, and Kazuo Minami, Distributed Learning for Three-dimensional Flow Field Mode Decomposition on Fugaku, 34th Symposium on Computational Fluid Dynamics (Web, Dec. 21-23, 2020), Paper F09-1.

[3] Koji Nishiguchi, Tokimasa Shimada, Masafumi Otaka, Shigenobu Okazawa, Makoto Tsubokura, Eulerian finite volume formulation using hierarchical Cartesian mesh for multi-material vehicle structures, Proceedings of the 3rd International Conference on Computational Engineering and Science for Safety and Environmental Problems (COMPSAFE2020), Kobe(Online), (2020).

[4] Tokimasa Shimada, Koji Nishiguchi, Shigenobu Okazawa, Makoto Tsubokura, Full Eulerian formulation using Lagrangian marker particles for fluid-structure interaction problem, Proceedings of the 3rd International Conference on Computational Engineering and Science for Safety and Environmental Problems (COMPSAFE2020), Kobe(Online), (2020).

[5] Masafumi Otaka, Koji Nishiguchi, Tokimasa Shimada, Makoto Tsubokura, Hirofumi Sugiyama, Shigenobu Okazawa, Large deformation solid dynamics using Marker particles based on Eulerian formulation, Proceedings of the 3rd International Conference on Computational Engineering and Science for Safety and Environmental Problems (COMPSAFE2020), Kobe(Online), (2020).

[6] Yuji Wada, Tokimasa Shimada, Koji Nishiguchi, Masafumi Otaka, Shigenobu Okazawa, Makoto Tsubokura, Topology optimization under structural Lagrangian marker particle method using hierarchical Cartesian mesh, Proceedings of the 3rd International Conference on Computational Engineering and Science for Safety and Environmental Problems (COMPSAFE2020), Kobe(Online), (2020).

12.5.4 Software

[1] CUBE, ver1.4, Published: March 18, 2021, dedicated for the automotive aerodynamics working-group members of the consortium for next-generation automobiles CAE utilizing HPC.

Chapter 13

Next Generation High Performance Architecture Research Team

13.1 Members

Masaaki Kondo (Team Leader)

Yiyu Tan (Research Scientist)

13.2 Overview of Research Activities

The next generation high performance architecture research team is conducting research and development of a next-generation high-performance computer architecture. Currently, we are mainly focusing on non-von Neumann architectures such as systolic arrays and neuromorphic computers based on the latest advances in device technologies, architectures that can integrate next generation non-volatile memories and/or various types of accelerators into a general-purpose processor, the advancement of scientific simulations by accelerating machine learning computations, and hybrid computing architectures that combine the benefits of quantum computing and classical computing. We are also performing detailed co-design evaluations of the computer architectures noted above as well as the co-design evaluations of algorithms that take advantage of them on the supercomputers K and Fugaku.

Another important aspect of designing future high-performance systems is power consumption. Power consumption is a prerequisite design constraint for developing exascale or next-generation computer systems. In order to maximize effective performance within a given power constraint, we need a new system-design concept in which the system's peak power is allowed to exceed maximum power provisioning using adaptively controlling power knobs incorporated in hardware components so that effective power consumption is maintained below the power constraint. This concept is recently known as hardware overprovisioning. In such systems, it is indispensable to allocate the power budget adaptively among various hardware component such as processors, memories, and interconnects, or among co-scheduled jobs, instead of fully utilizing all available hardware resources. We are researching strategies to improve the power efficiency and total system throughput for future hardware overprovisioned supercomputer systems.

In this fiscal year, we have conducted several researches including system performance analysis using performance counter, domain-specific architectures based on systolic array for sound field rendering, architecture considerations for a next-generation supercomputer system.

13.3 Research Results and Achievements

13.3.1 System performance analysis using performance counter

Most HPC applications achieves only a small fraction of the peak performance of processors. Although emerging processors greatly increase their peak computational performance, they also increase the complexity of performance optimization. In order to enable users to collect an application's performance metrics, such as cache misses and TLB misses, performance monitoring counters have been widely applied to monitor different events that happen during instruction execution of a processor. These performance metrics can help identify possible performance bottlenecks of an application and provide valuable clues to programmers and computer architects for performance optimization. In this research, performance counter was introduced to gather performance information and characterize application performance. During characterization, the top-down analysis method (Figure 13.1) was applied, in which execution time of an application was categorized into Front Bound, Bad Speculation, Backend Bound, and Retiring. Ten performance counter events were monitored and the measurement functions based on the PAPI (version 6.0) were developed. The ECP Proxy-apps 3.0 was used as a benchmark to be analyzed on a desktop machine with 512 GB DDR4-2933 RAMs and an Intel Xeon Gold 6212U processor running at 2.4 GHz. The compiler is GNU 4.8.5 and Intel Parallel Studio XE 2020.

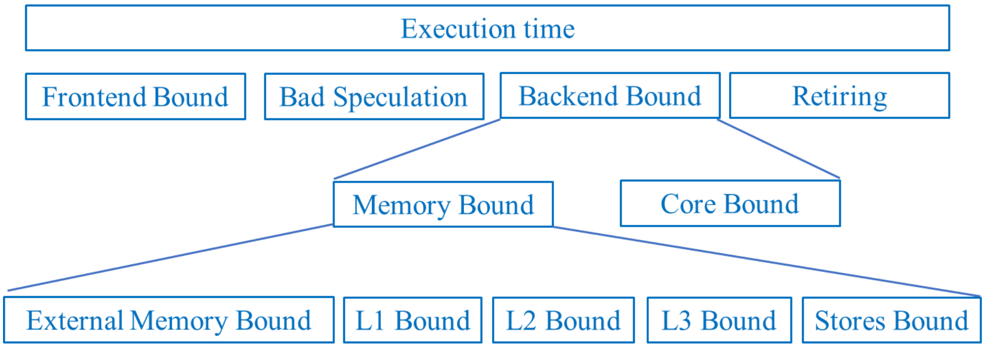


Figure 13.1: Top down analysis

Figure 13.2 presents the breakdown of execution time of each benchmark program. As shown in Fig. 2, it is easy to characterize performance bottleneck of each benchmark program, and then performance optimization techniques may be applied on such characterization.

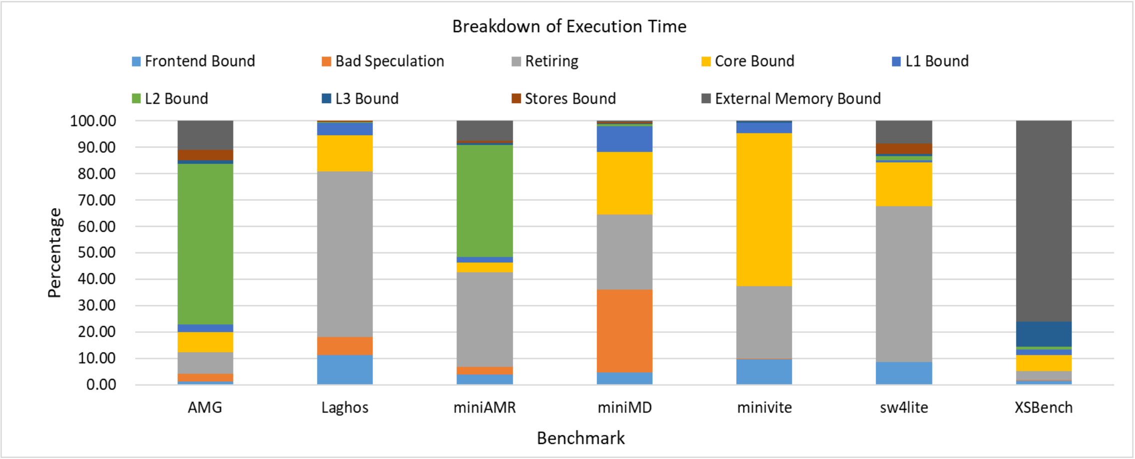
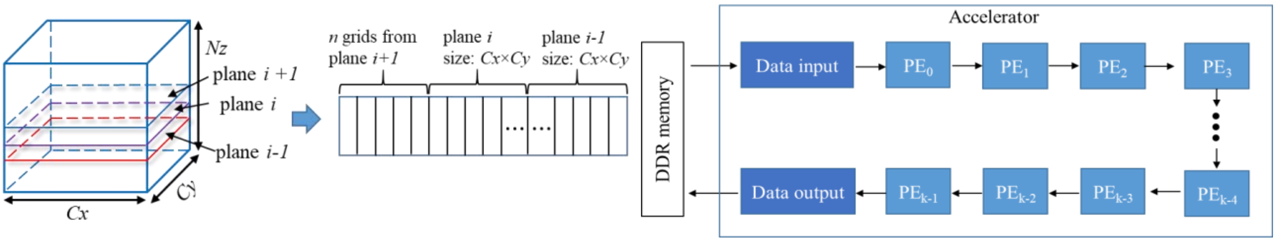


Figure 13.2: Breakdown of execution time

13.3.2 FPGA-based acceleration of FDTD sound field rendering

Finite difference time domain (FDTD) schemes are widely applied to analyse sound propagation, but are memory-intensive as sound space is increased. In this research, the high-order FDTD algorithm was proposed to reduce memory requirement in the sound rendering, and an FPGA-based acceleration system with the proposed algorithm was designed and evaluated, in which the spatial blocking and temporal blocking were applied to reduce required memory bandwidth and reuse data, respectively. As shown in Figure 13.3(a), spatial blocking is employed to reduce the required external memory bandwidth and size of on-chip buffers. A large sound space with M × N × K grids is decomposed into small sub-places (blocks) and a sub-place is further partitioned into x-y layers along the z dimension. Each sub-place has Nx × Ny × K grids. Calculating sound pressures of grids on the plane i requires sound pressures of grids on three adjacent planes (planes i-1, i, and i+1). In order to efficiently utilize the external memory bandwidth and speed up computation, data accesses are coalesced and computation is vectorized simultaneously by loop unrolling. Furthermore, shift registers are used as on-chip buffers to exploit the regular memory access pattern. Data values on the planes i-1 and i are firstly streamed into the right shift register from external memory. If N grids are computed in parallel, along with N values on the layer i+1 are read into the shift register, the computing unit fetches related data values from the shift register in parallel through their fixed offsets to the computed nodes, and computes sound pressures of N grids. Then the shift register is shifted right by N data slots, another N new data are streamed into the head of the shift register from external memory. Sound pressures of another N grids are computed. Such procedure is iterated until all grids are computed.

Figure13.3(b) shows the system diagram with temporal blocking, in which several processing elements (PEs) are cascaded to compute sound pressures of grids in a same spatial block at continuous time steps. A PE calculates sound pressures at a time step, and computed results are sent to the neighbor PE, where sound pressures at next time step are computed. Such computation procedure is repeated until the final PE is ended. Therefore, accesses to external memory are reduced and computation is speeded up because sound pressures of a spatial block at several time-steps are computed concurrently.



(a) Spatial blocking (b) System diagram with temporal blocking

Figure 13.3: Spatial and temporal blocking

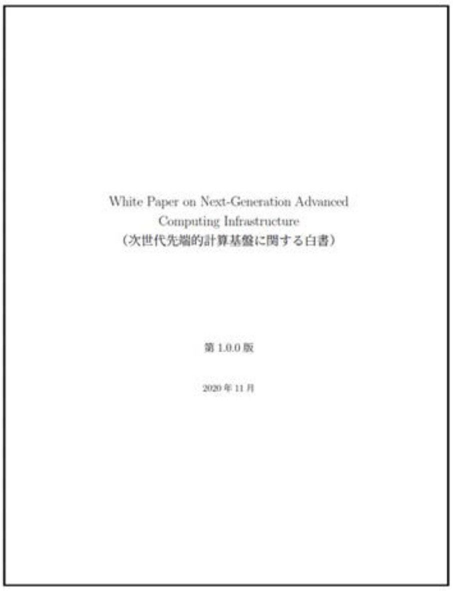
Table 13.1 presents the rendering time per time step taken by the FPGA-based accelerator on the DE10-Pro FPGA card and software simulation performed on a desktop machine with 512 GB RAMs and an Intel Xeon Gold 6212U processor running at 2.4 GHz in the case of a three-dimensional shoebox with dimension being 16m×8m×8m. The FPGA-based system outperforms the software simulation by 11, 13 and 18 times in second-order, 4th-order, and 6th-order, respectively. The related results were presented as posters on the IEEE Cluster 2020, USE 2020, and R-CCS International Symposium. Some other results will be published by the Journal of the Audio Engineering Society.

Table 13.1: Rendering time per time step

Orders	FPGA	Software Simulation
2nd	0.0486	0.5363
4th	0.0333	0.4458
6th	0.0238	0.4437

13.3.3 Considerations of next-generation supercomputer systems

Supercomputer Fugaku began operation in 2020. The starting point of Fugaku development was the community-wide activities called ”Strategic Development of High-Performance Computers (SDHPC)” project which actively worked from year 2010 to 2013. In the SDHPC project, a white paper of ”The report for future HPCI technical development” was published, which addressed several architecture candidates as well as research roadmaps for future supercomputer systems. Having thought for sustainable evolution of high-performance computing and its system environment, we expect a new wave of supercomputing such as further tight collaboration between scientific simulation and AI/Big-data as well as a new application domain of Society 5.0, but we need to address several research challenges, for example, preparing the end of moore's law. Therefore, we launched a new community-wide activities called NGACI (Next-Generation Advanced Computing Infrastructure) to discuss prospected next-generation system images, technical challenges to realize them, and research opportunities for future high-performance computing infrastructure. We published a white paper to summarize the discussion in the NGACI community.



White Paper on Next-Generation Advanced
Computing Infrastructure
（次世代先端的計算基盤に関する白書）

第 1.0.0 版

2020 年 11 月

Figure 13.4: NGACI White paper (https://sites.google.com/view/ngaci/home)

13.4 Schedule and Future Plan

In order to achieve further performance improvement for next generation HPC systems in post-Moore era, it is necessary to explore various types of devices, hardware architectures, system software/programming models, and algorithms that may contribute to the future system designs. We need to evaluate and analyze huge amount of possible scenarios varying the architectural parameters on wide variety of underlying system architecture. In future, We will focus on the following topics.

(1) Performance model of HPC benchmark applications. We will analyze HPC benchmark applications, including performance requirement and execution characteristics, and develop a performance model or performance simulation environment to evaluate wide variety of future HPC architectures.

(2) Architecture for HPC. Different architectures for HPC will be explored, including systolic array, SIMD, CGRA. Their performance model will be developed and compared, and new architecture will be investigated for HPC.

(3) Computing with high-performance and low-power devices. we will consider next-generation high-performance and low-power computing paradigms and architectures for post-Moore era, and continue to study an ultra-high-performance accelerator system with an emerging devices.

13.5 Publications

13.5.1 Articles/Journal

[1] Yiyu Tan, Toshiyuki Imamura, and Masaaki Kondo, "FPGA-based Acceleration of FDTD Sound Field Rendering", accepted by the Journal of the Audio Engineering Society.

13.5.2 Conference Papers

[2] Yuan He, Jinyu Jiao, Thang Cao and Masaaki Kondo, ”Energy-Efficient On-Chip Networks through Profiled Hybrid Switching”, The 30th ACM Great Lakes Symposium on VLSI (GLSVLSI 2020), pp241-246, Sep. 2020.
[3] Motoki Sakurai, Yosuke Ueno, Masaaki Kondo, ”Path Planning and Moving Obstacle Avoidance with Neuromorphic Computing”, 2021 IEEE International Conference on Intelligence and Safety for Robotics (IEEEISR 2021), March 2021.

13.5.3 Posters

[4] Yiyu Tan and Toshiyuki Imamura, “An FPGA-based Sound Field Rendering System”, IEEE Cluster, Kobe, Japan, September 2020.
[5] Yiyu Tan, Toshiyuki Imamura, and Masaaki Kondo, “Design and Implementation of High-order FDTD Method for Room Acoustics”, the 41st Symposium on Ultrasonic Electronics, Osaka, Japan, November 2020.
[6] Yiyu Tan, Toshiyuki Imamura, and Masaaki Kondo, “Design and Implementation of FPGA-based High-order FDTD Method for Room Acoustics”, the 3rd R-CCS International Symposium, Feb. 2021.

13.5.4 Invited Talks

[7] 近藤正章, ”NGACI: 次世代先端的計算基盤の開発に向けたコミュニティ活動”, 第20回PCクラスタシンポジウム, 2020年12月.
[8] 近藤正章, ”NGACIにおける検討状況の紹介”, 次世代コンピューティング・フォーラム, 2021年2月.

Chapter 14

High Performance Big Data Research Team

14.1 Members

Kento Sato (Team Leader)

Jens Domke (Postdoctoral Researcher)

Jian Guo (Postdoctoral Researcher)

Takaaki Fukai (Postdoctoral Researcher)

Takashi Shimokawabe (Visiting Scientist)

14.2 Overview of Research Activities

High Performance Big Data Research Team at RIKEN R-CCS researches and develops system software for large-scale HPC (High Performance Computing) systems such as the Supercomputer Fugaku. Especially, we study state-of-the-art techniques for convergence of HPC, AI and Big data technologies as well as fundamental R&D in HPC. To achieve the goal, we develop system software to accelerate deep learning and big data processing on large-scale HPC systems, i.e., (HPC for AI/BD). We also apply AI and Big data processing techniques to resolve several technical challenges in large-scale HPC systems. We also study techniques to design next-generation large-scale HPC systems.

The research topics include (but not limited to): (1) Scalability and acceleration of big data processing with massively parallel I/O by making use of next-generation storage and file systems; (2) Development of massively parallel algorithms and programming models for the next-generation non-volatile memory and deeply hierarchical memory/storage architectures; (3) Research and development for big data collection, transfer, accumulation, management, and utilization for data science; (4) Development of system software for integrating applications (Society 5.0 simulation as well as system operation), artificial intelligence training, and training data collection; (5) Scalability and acceleration of large-scale deep learning and inference; (6) Scalability and acceleration of high-reliability technologies such as checkpointing with large-scale I/O; (7) Architecture exploration for the development of next generation large-scale computers; (8) Development of tools to support application development and execution environments for large-scale computers; (9) Any other research and development related to high performance computing.

In FY2020, our team made several achievements which include: (1) Unprecedented performance achievement on DL4Fugaku in MLPerf HPC; (2) Optimizing Asynchronous Multi-Level Checkpoint/Restart Configurations with Machine Learning; (3) High-Performance Routing with Multipathing and Path Diversity in Supercomputers and Data Centers; (4) Improved failover for HPC interconnects through localised routing restoration; (5) Measurement of I/O performance for distributed deep neural networks on Fugaku.

14.3 Research Results and Achievements

14.3.1 DL4Fugaku: Deep learning for Fugaku

Large-scale deep learning has emerged as an essential machine learning approach for many research challenges such as image classification, speech recognition and many others. Fast and large-scale deep learning enables us to train neural networks with more training data in shorter time. Supercomputer Fugaku is expected to enable high performance computing for deep learning since A64FX, which is a general-purpose processor equipped in Fugaku, provides high-speed half-precision floating point (FP16) and 8-bit integer (INT8) operations for matrix multiplications and high bandwidth HBM2 memory (1,048 GB/sec) for convolutions. Also, Fugaku interconnects employ the next-generation ToFu interconnects (ToFuD) for gradient reduction operations. However, to make use of Fugaku/A64FX hardware performance, tuning software stacks from deep learning frameworks to low-level numerical libraries is indispensable.

To achieve fast and scalable deep learning in Fugaku, we launched a new project, DL4Fugaku (Deep learning for Fugaku). The goals of the projects are (1) performance analysis and tuning of deep learning frameworks and low-level numerical libraries used by the frameworks; (2) Reliable deployment of large-scale deep learning environments; (3) Enhancement of the usability for production use in Fugaku. We organized a project team for DL4Fugaku from PIs and researchers in the application development unit, the high-performance AI system research team, the high-performance big data research team and the large-scale parallel numerical computing technology research team under collaboration with industry, academia and government; AIST, ARM, Cybozu, Fujitsu laboratories, Fujitsu limited, Linaro and Toky Tech. To facilitate the logistics and accelerate the software development, RIKEN R-CCS signed MOU with Fujitsu ltd. for further collaboration on the DL4Fugaku project.

In FY2021, our core project team (RIKEN R-CCS, Fujitsu and AIST) achieved unparalleled speed on the MLPerf HPC machine learning on Japanese Supercomputer Systems, ABCI and Fugaku (`https://www.fujitsu.com/global/about/resources/news/press-releases/2020/1119-02.html`). This article summarizes our achievement as *"Fujitsu, the National Institute of Advanced Industrial Science and Technology (AIST), and RIKEN today announced a performance milestone in supercomputing, achieving the highest performance and claiming the ranking positions on the MLPerf HPC benchmark(1). The MLPerf HPC benchmark measures large-scale machine learning processing on a level requiring supercomputers, and the parties achieved these outcomes leveraging approximately half of the "AI-Bridging Cloud Infrastructure" ("ABCI") supercomputer system, operated by AIST, and about 1/10 of the resources of the supercomputer Fugaku, which is currently under joint development by RIKEN and Fujitsu. Utilizing about half the computing resources of its system, ABCI achieved processing speeds 20 times faster than other GPU-type systems. That is the highest performance among supercomputers based on GPUs, computing devices specialized in deep learning. Similarly, about 1/10 of Fugaku was utilized to set a record for CPU-type supercomputers consisting of general-purpose computing devices only, achieving a processing speed 14 times faster than that of other CPU-type systems. The results were presented as MLPerf HPC v0.7 on November 18th (November 19th Japan Time) at the 2020 International Conference for High Performance Computing, Networking, Storage, and Analysis (SC20) event, which is currently being held online".*

14.3.2 Optimizing Asynchronous Multi-Level Checkpoint/Restart Configurations with Machine Learning

Current petascale systems in High-performance computing (HPC) deal with enormous workloads. Such workloads require a massive number of components simultaneously. Though the HPC systems are built using highly reliable components, with the sheer number of components, there is an increasing frequency of component failures, resulting in degradation of system and application reliability. To reliably run applications on such large-scale systems, a common technique is checkpoint/restart (CR) in which the system writes a snapshot of the application's state at fixed intervals to persistent storage, which is called a checkpoint. Application states can later be restored to the last saved checkpoint in case of a failure. Though CR is useful for large scale systems, its overhead can be an enormous challenge on large-scale systems.

One of approaches to reduce the overhead of CR is to determine the optimal checkpoint interval and checkpoint count. Poorly determined checkpoint interval makes system resilience worse. There are two approaches for obtaining the optimal checkpoint interval and checkpoint count values for any given configuration, namely the modeling approach and the simulation approach. The modeling approach mainly formulates an analytical solution to obtain optimal values, whereas the simulation approach runs the application across multiple failures to check different scenarios for obtaining the optimal values.

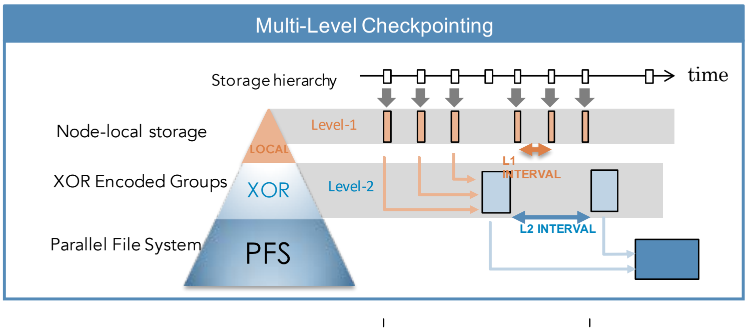


Figure 14.1: Multi-level checkpointing

In simple checkpoint models with synchronous checkpointing, where the checkpointing process mainly comprises of compute, checkpoint, and recovery state in a serial manner without other concurrent operations in the background. such modeling is beneficial in formulating an analytical solution for the optimal interval. However, with fast local storage becoming commercially available, asynchronous multi-level checkpointing (Async-MLC) has become a common approach for efficient checkpointing. As shown in Figure 14.1, in Async-MLC, compute, checkpoint and recovery occur asynchronously from the application computation. This allows most of the checkpoint operations to happen in the background without delaying the critical compute operations in the application, thus minimizing the checkpoint overhead. This is particularly attractive when a parallel file system (PFS) is equipped with multi-level storage devices, which allows the checkpointing operations to happen on lower-level storage devices while the application I/O continues with upper-level storage devices.

Although introduction of multi-level checkpointing has greatly improved its performance over the simple model, however without the optimal interval and checkpoint count configuration, the performance of the system will still be degraded due to the checkpoint overhead. To obtain the optimal configuration of checkpoint interval and checkpoint count in MLC, the modeling approach is ineffective as it faces significant difficulty to formulate an analytical solution unless we simplify the model and/or make strong assumptions. Another approach is simulation. It is very effective in determining the optimal checkpoint configuration, however, it is very slow as it runs thousands of scenarios before obtaining the optimal values and this makes the simulation approach impractical for real world usage. In this work, we obtained the optimal checkpoint configuration for a given HPC system using the effectiveness and accuracy of the simulation approach while reducing the time taken by simulation to obtain the optimal result. We achieved this by combining machine learning methods with the simulation approach where we applied our machine learning models on existing simulated data to learn from the existing data and obtain the optimal checkpoint configuration with minimal error. The major contributions of this work are:

- Development of multi-level checkpoint simulator to replicate the behavior of real world large scale systems.

- Novel machine learning (ML) approach that achieve convergence of traditional ML (Random forest) and state-of-the-art ML (NN), i.e. daisy chaining.

- Novel pre-processing for neural network (NN) to optimize CR i.e., parameter reduction. (Parameter correlation analysis is universal to other optimization area in general. Our work gives one instance in CR)

- Quantitative evaluation: Random Forest v.s. LightGBM v.s. Baseline NN v.s. NN after daisy chaining v.s. NN after parameter reduction with daisy chaining

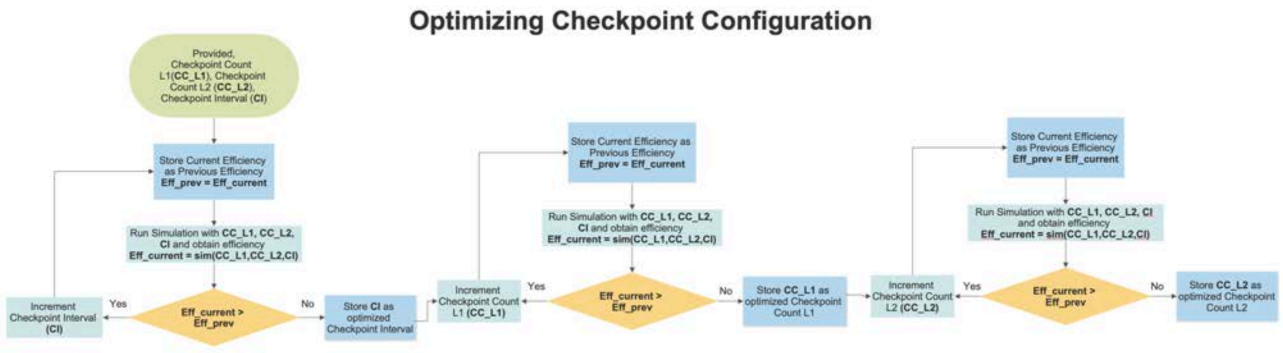

Optimizing Checkpoint Configuration
Provided, Checkpoint Count L1(CC_L1), Checkpoint Count L2 (CC_L2), Checkpoint Interval (CI)
Store Current Efficiency as Previous Efficiency Eff_prev = Eff_current
Run Simulation with CC_L1, CC_L2, CI and obtain efficiency Eff_current = sim(CC_L1,CC_L2,CI)
Eff_current > Eff_prev
Yes
Increment Checkpoint Interval (CI)
No
Store CI as optimized Checkpoint Interval
Increment Checkpoint Count L1 (CC_L1)
Store Current Efficiency as Previous Efficiency Eff_prev = Eff_current
Run Simulation with CC_L1, CC_L2, CI and obtain efficiency Eff_current = sim(CC_L1,CC_L2,CI)
Eff_current > Eff_prev
Yes
No
Store CC_L1 as optimized Checkpoint Count L1
Increment Checkpoint Count L2 (CC_L2)
Store Current Efficiency as Previous Efficiency Eff_prev = Eff_current
Run Simulation with CC_L1, CC_L2, CI and obtain efficiency Eff_current = sim(CC_L1,CC_L2,CI)
Eff_current > Eff_prev
Yes
No
Store CC_L2 as optimized Checkpoint Count L2

Figure 14.2: Optimization of Multi-level checkpointing with neural networks

With our approach and design optimizations, we show that our models can predict the optimized parameter values when trained with the simulation approach. We also showed that using more advanced deep neural network techniques can improve the performance of neural network over the machine learning models by up to 50%. which can further be used in future research works to optimize the checkpoint restart configuration of large scale systems.

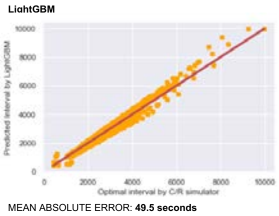


Figure 14.3: Prediction accuracy of LGBM

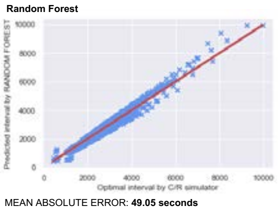


Figure 14.4: Prediction accuracy of random forest

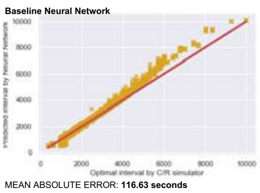


Figure 14.5: Prediction accuracy of simple neural network

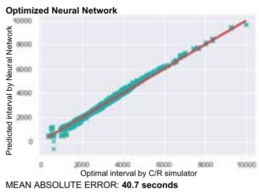


Figure 14.6: Prediction accuracy of optimized neural network

14.3.3 High-Performance Routing with Multipathing and Path Diversity in Supercomputers and Data Centers

The recent line of research into topology design focuses on lowering network diameter. Many low-diameter topologies such as Slim Fly or Jellyfish that substantially reduce cost, power consumption, and latency have been proposed, see Figure 14.7. A key challenge in realizing the benefits of these topologies is *routing*. On one hand, these networks provide shorter path lengths than established topologies such as Clos or torus, leading to performance improvements. On the other hand, the number of shortest paths between each pair of endpoints is much smaller than in Clos, but there is a large number of non-minimal paths between router pairs. This hampers or even makes it impossible to use established multipath routing schemes such as ECMP.

Many networking architectures and routing protocols have been developed. They offer different forms of support for multipathing, they are related to different parts of various networking stacks, and they are based on miscellaneous classes of simple routing building blocks or design principles. To propel research into future developments in the area of high-performance routing, we present the first analysis and taxonomy of the rich landscape of multipathing and path diversity support in the routing designs in supercomputers and data centers. We identify basic building blocks, see Table 14.1, we crystallize fundamental concepts, we list and categorize existing architectures and protocols, and we discuss key design choices, focusing on the support for different forms of multipathing and path diversity. Our analysis can be used by network architects, system developers, and routing protocol designers who want to understand how to maximize the performance of their developments in the context of bare Ethernet, full TCP/IP, or InfiniBand and other HPC stacks.

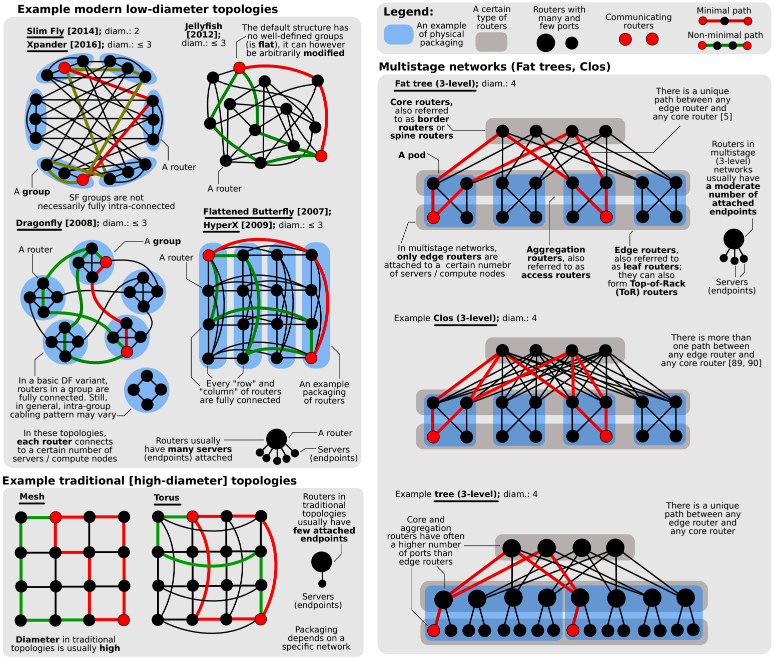

Example modern low-diameter topologies
Slim Fly [2014]; diam.: 2
Xpander [2016]; diam.: ≤ 3
A router
A group
SF groups are not necessarily fully intra-connected
Jellyfish [2012]; diam.: ≤ 3
The default structure has no well-defined groups (is flat), it can however be arbitrarily modified
A router
Dragonfly [2008]; diam.: ≤ 3
A router
A group
In a basic DF variant, routers in a group are fully connected. Still, in general, intra-group cabling pattern may vary
Flattened Butterfly [2007];
HyperX [2009]; diam.: ≤ 3
A router
Every "row" and "column" of routers are fully connected
An example packaging of routers
In these topologies, each router connects to a certain number of servers / compute nodes
Routers usually have many servers (endpoints) attached
A router
Servers (endpoints)
Example traditional [high-diameter] topologies
Mesh
Torus
Routers in traditional topologies usually have few attached endpoints
Servers (endpoints)
Diameter in traditional topologies is usually high
Packaging depends on a specific network
Legend:
An example of physical packaging
A certain type of routers
Routers with many and few ports
Communicating routers
Minimal path
Non-minimal path
Multistage networks (Fat trees, Clos)
Fat tree (3-level); diam.: 4
Core routers, also referred to as border routers or spine routers
A pod
There is a unique path between any edge router and any core router [5]
Routers in multistage (3-level) networks usually have a moderate number of attached endpoints
In multistage networks, only edge routers are attached to a certain numebr of servers / compute nodes
Aggregation routers, also referred to as access routers
Edge routers, also referred to as leaf routers; they can also form Top-of-Rack (ToR) routers
Servers (endpoints)
Example Clos (3-level); diam.: 4
There is more than one path between any edge router and any core router [89, 90]
Example tree (3-level); diam.: 4
Core and aggregation routers have often a higher number of ports than edge routers
There is a unique path between any edge router and any core router

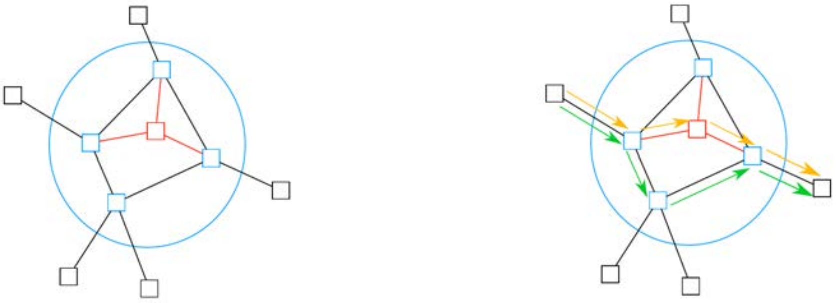
Figure 14.7: Illustration of network topologies related to the routing protocols and schemes considered in this work. Red color indicates an example shortest path between routers. Green color indicates example alternative non-minimal paths. Blue color illustrates grouping of routers.

Routing Scheme (Name, Abbreviation, Reference)	Related concepts	Stack Layer	Features of schemes						Additional remarks and clarifications
			SP	NP	MP	DP	ALB	AT	
General routing building blocks (classes of routing schemes)									
Simple Destination-based routing	Ⓡ	L2, L3	🖒	👍*	✖	✖	✖	🖒	*Care must be taken not to cause cyclic dependencies
Simple Source-based routing (SR)	Ⓡ	L2, L3	🖒	🖒	👍*	👍*	✖	🖒	Source routing is difficult to deploy in practice, but it is more flexible than destination-based routing. *As endpoints know the physical topology, multipathing should be easier to realize than in destination routing.
Simple Minimal routing	Ⓟ	L2, L3	🖒	✖	✖	✖	✖	🖒	Easy to deploy, numerous designs fall in this category
Specific routing building blocks (concrete protocols or concrete protocol families)									
Equal-Cost Multipathing (ECMP)	Ⓡ Ⓛ	L3	🖒	✖	🖒	✖	✖	🖒	In ECMP, all routing decisions are local to each switch.
Spanning Trees (ST)	Ⓟ	L2	👍*	👍*	✖	✖	✖	🖒	*ST protocol offers shortest paths only within one spanning tree.
Packet Spraying (PR)	Ⓛ	L2, L3	🖒	✖	🖒	✖	✖	🖒	One selects output ports with round-robin or randomization.
Virtual LANs (VLANs)	Ⓟ	L2	👍*	👍*	✖	✖	✖	🖒	*VLANs by itself does not focus on multipathing, and it inherits spanning tree limitations, but it is a key part of multipathing arch.
IP Routing Protocols	Ⓡ	L2, L3	🖒	✖	✖	✖	✖	🖒	Examples are OSPF, IS-IS, EIGRP.
Location–Identification Separation (LIS)	Ⓡ	L2, L3	🖒*	🖒*	✖*	✖*	✖	🖒	*LIS by itself does not focus on multipathing and path diversity, but it may facilitate developing a multipathing architecture.
Valiant load balancing (VLB)	Ⓡ Ⓟ Ⓛ	L2, L3	✖	🖒	✖	✖	✖	🖒	—
UGAL	Ⓡ Ⓟ Ⓛ	L2, L3	🖒	🖒	👍*	✖	🖒	🖒	UGAL means Universal Globally-Adaptive Load balanced routing.
Network Address Aliasing (NAA)	Ⓛ	L3, subnet	🖒*	🖒*	🖒*	🖒*	🖒*	🖒	NAA is based on IP aliasing in Ethernet networks and virtual ports via LID mask control (LMC) in InfiniBand. *Depending on how a derived scheme implements it.
Multi-Railing	Ⓟ	L2, L3, subn.	🖒*	🖒*	🖒	🖒*	🖒*	🖒	*Depending on how a derived scheme implements it.
Multi-Planes	Ⓟ	L2, L3, subn.	🖒*	🖒*	🖒	🖒	🖒*	🖒	*Depending on how a derived scheme implements it.

Table 14.1: **Comparison of simple routing building blocks (often used as parts of more complex routing schemes; see Table 2 of publication). Rows are sorted chronologically. We focus on how well the compared schemes utilize path diversity. "Related concepts"** indicates the associated routing concepts described in paper (Ⓟ **Path selection**, Ⓡ **Routing** itself, and Ⓛ **Load balancing**). **"Stack Layer"** indicates the location of each routing scheme in the TCP/IP or InfiniBand stack. **SP**, **NP**, **MP**, **DP**, **ALB**, and **AT** illustrate whether a given routing scheme supports various aspects of path diversity. Specifically: **SP**: A given scheme enables using arbitrary **shortest** paths. **NP**: A given scheme enables using arbitrary **non-minimal** paths. **MP**: A given scheme enables **multipathing** (between two hosts). **DP**: A given scheme considers **disjoint** paths. **ALB**: A given scheme offers **adaptive load balancing**. **AT**: A given scheme works with an **arbitrary topology**. 🖒: A given scheme does offer a given feature. 👍: A given scheme offers a given feature in a limited way. ✖: A given scheme does not offer a given feature. *Explanations in remarks.

14.3.4 Improved failover for HPC interconnects through localised routing restoration

With the growth in size of high performance computing systems and interconnects, the runtime performance of routing algorithms for arbitrary topologies becomes a problem in some cases. Namely, recalculation of the routing information after network component failure is one, as the time required to calculate a new routing using a typical routing algorithm directly translates to system downtime. We introduce a failover strategy that aims to improve the runtime performance in such cases by using the original routing information and repairing it instead of recalculating it from zero, see Figure 14.8. It achieves this while preserving deadlock-freedom, which is usually an essential property in HPC interconnects, tries to optimise the load balancing of the new routes as much as possible, and does so without the need for additional virtual channels, the number of which is typically limited by the underlying network hardware.

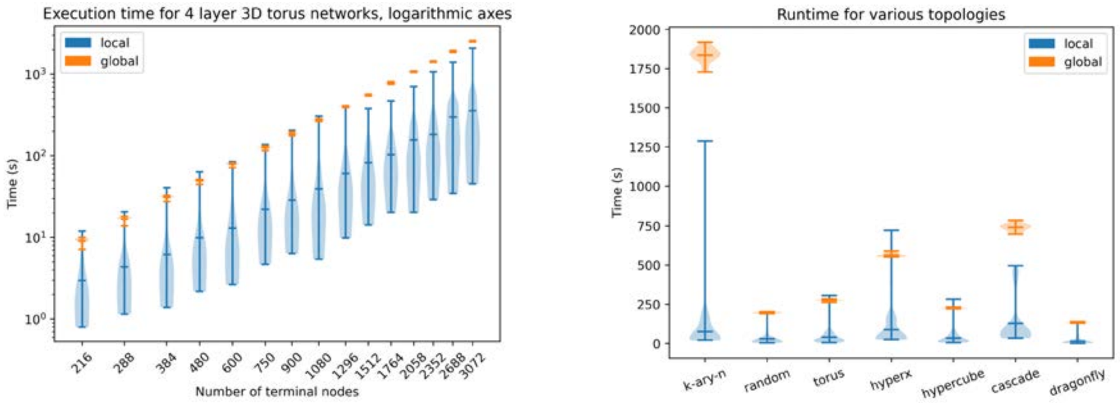


(a) Hardware failure (red) causing network outage

(b) Failover through localised routing restoration

Figure 14.8: Novel, 4-step failover approach: (1) Identify a Rerouting Area (RA) encompassing failure; (2) Find affected existing routes; (3) Find paths within the RA; and (4) Repair the original routes which are inside RA

In particular, it achieves this by choosing a recalculation area which completely encompasses the failure, and finding paths between endpoints only locally within that subset. Subsequently, it identifies affected paths from the original routing and repairs them locally in the recalculation area. Due to the need to preserve deadlock freedom in a network that already has dependency constraints, this approach is based on the Nue routing algorithm, which is equipped for the task by its nature of directly considering dependencies while calculating paths. Finally, we demonstrate the improvements by comparing the runtime performance of our approach and recalculation from zero using a typical algorithm on a multitude of different topology types and sizes, see Figure 14.9.



(a) Approx. the same scaling properties as Nue routing, but yielding 5-fold performance increase

(b) Routing time improvements vary by topology type, but more than 5-fold for each one

Figure 14.9: Application of routing failover approach to various topologies with artificially injected errors

14.3.5 Measurement of I/O performance for distributed deep neural networks on Fugaku

Today, deep learning is an essential technology for our life. For example, deep learning contributes to high-quality machine translation, image classification, and recognition, used in mobile devices and auto-motives. Deep learning is also indispensable for science applications.

To solve more complex problems with deep learning, both sizes of training datasets and neural networks are increasing. For training models with large datasets and networks, a single computer is not enough. Therefore, distributed deep neural network (DDNN) training, training models by multiple computers, is necessary.

For large-scale DDNN training, HPC clusters are a promising computation environment. Therefore, our next-generation supercomputer, Supercomputer Fugaku, is also expected to be used for the DDNN training. Our project named DL4Fugaku is working on DDNN training on Fugaku and achieved the 2nd highest score at the previous MLPerf HPC.

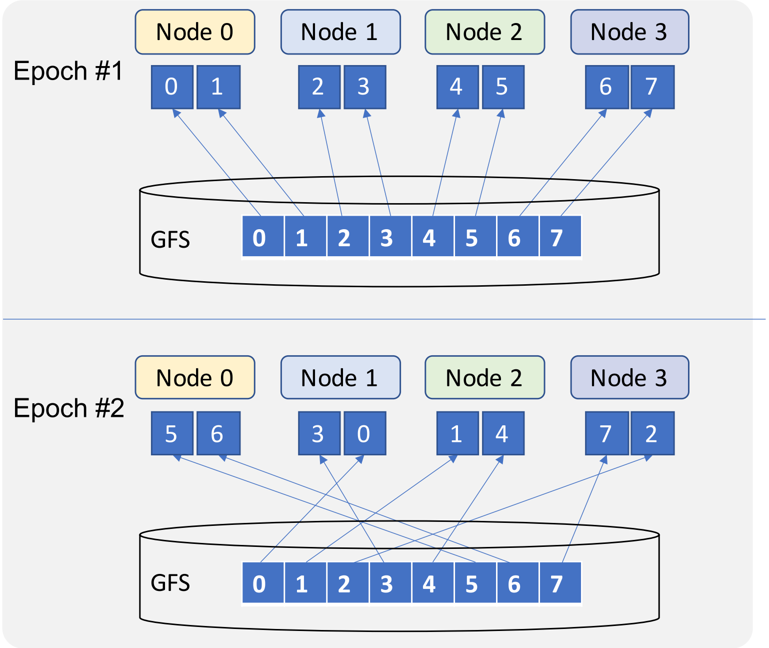


Figure 14.10: Accesses to samples by nodes in distributed deep learning workloads (The blue boxes indicate samples, and their numbers indicate sample indices.)

In large-scale DDNN, I/O performance is a challenging problem. In deep learning, the training program should randomly change the input order of the training samples to the network for each epoch. Therefore, as shown in Figure 14.10, each sample is accessed once an epoch and the order of accessing them is random. So there is no locality of space and time, and general cache mechanisms do not work well. On the other hand, it is impractical that each node keeps the whole dataset. Therefore, the large-scale DDNN training applications suffer performance overhead because of the large number of remote accesses on loading data.

Several researchers revealed the bottleneck of I/O performance and proposed methods to improve I/O performance on their clusters. Existing works also show that using remote memory as buffers for the shuffle improves the data load performance. However, Fugaku has the following different points from those clusters.

- Fugaku consists of a large number of small compute nodes (48 cores and 32 GiB HBM2 memory without accelerator), unlike clusters that introduce GPUs.
- The compute nodes are connected through ToFuD six dimension mesh-torus network.

Therefore, we want to reveal how the data loading performance with the remote memory scales on Fugaku. Additionally, we want to know if there are any optimization opportunities for the I/O workload. To do this, we measured the I/O performance of the workload with remote memory on Fugaku.

For an early-stage experiment, we implemented a benchmark to measure the data loading performance from the remote memory in DDNN workload. The benchmark program splits the dataset into the same number of chunks as that of nodes, then each node loads one of the chunks and puts it on the memory. On loading the samples, the node loads them from the own memory if they are on there. Otherwise, the node loads samples from the other nodes serving them. The nodes transferred samples by the functions Isend() and Irecv() provided by mpi4py module. Instead of all-reduce operations, the benchmark just synchronized all nodes on each iteration. To measure only the data loading performance, the benchmark did not execute calculations such as forward, backward, and optimization.

We set up the experiments assuming MLPerf HPC CosmoFlow workload, a large-scale CNN workload. The file size of a sample was 16 MiB, and the number of samples per node was 32. Therefore, the total number of the samples was 32 × the number of the node. The local batch size was 1, and the global batch size was the same as the number of nodes. The benchmark executed 100 epochs for each measurement. We measured the execution time with 8, 16, 48, 96, 192, 384 nodes.

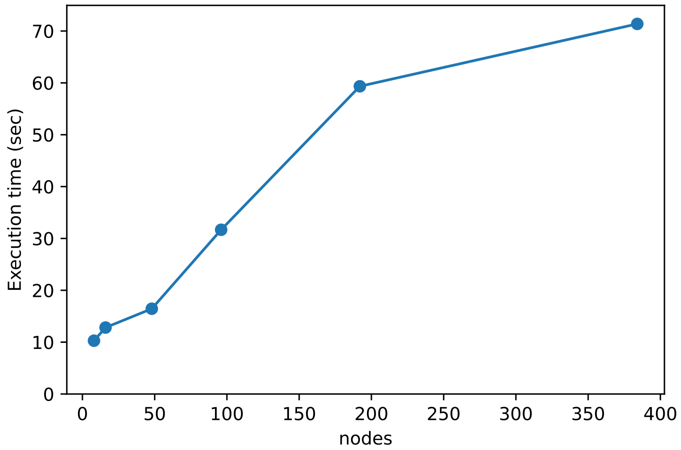


Figure 14.11: Data loading time from remote memory

Figure 14.11 is the early stage results of the measurement. Each value is an average of 10 times measurement. In this experiment, ideally, the execution time would be constant because the number of samples in each node was constant. However, the measurement result shows that the execution time was increasing with the number of nodes.

We think that increasing the farthest network distance results the performance degradation. Therefore, we expect it will improve the data loading performance that multiple nodes serve a sample so that each node can load it from the nearer node. In other words, making replicas of the dataset on the memory of the nodes may improve the data loading performance. The future work is more analysis of this measurement result and proposing the optimization method for data loading in DDNN workload on massive parallel HPC clusters according to the finding from the analysis.

14.4 Schedule and Future Plan

In FY2021, we will continuously work on HPC-for-AI, AI-for-HPC and many other researches and developments for HPC. These research topic include: (1) Fast and scalable parallel I/O by taking advantage of next-generation memory (e.g., Non-volatile memory) in big data processing and machine learning; (2) Scalable checkpointing for fault tolerance by taking advantage of next-generation memory (e.g., Non-volatile memory); (3) Scalable algorithms for deeply hierarchical memory and storage architecture; (4) Fast data transfer technique for multi-petabyte of big data on high-speed network; (5) Other research and software development related to big data, machine learning and I/O.

14.5 Publications

14.5.1 Articles/Journal

1. M. Besta, J. Domke, M. Schneider, M. Konieczny, S.D. Girolamo, T. Schneider, A. Singla, T. Hoefler, ”High-Performance Routing with Multipathing and Path Diversity in Supercomputers and Data Centers,” IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 4, pp. 943-959, 2021

14.5.2 Conference Papers

2. T. Dey, K. Sato, B. Nicolae, J. Guo, J. Domke, W. Yu, F. Cappello, K. Mohror, ”Optimizing Asynchronous Multi-level Checkpoint/Restart Configurations with Machine Learning,” in Proceedings of the 2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), May 2020. (co-located with 34th IEEE IPDPS)

3. M. Wahib, H. Zhang, T.T. Nguyen, A. Drozd, J. Domke, L. Zhang, R. Takano, S. Matsuoka, ”Scaling Distributed Deep Learning Workloads beyond the Memory Capacity with KARMA,” in Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC ’20, (Piscataway, NJ, USA), IEEE Press, Nov. 2020

14.5.3 Posters

4. T. Fukai, K. Sato, “Measurement of I/O performance for distributed deep neural networks on Fugaku” Poster presented at The 3rd R-CCS International Symposium (RCCS-IS3), Kobe, Japan, Feb. 2021.

5. I.R. Ivanov, J. Domke, A. Nomura, T. Endo, ”Improved failover for HPC interconnects through localised routing restoration” Poster presented at The 3rd R-CCS International Symposium (RCCS-IS3), Kobe, Japan, Feb. 2021.

14.5.4 Invited Talks

6. K. Sato, “High Performance System Software Enabling Convergence of AI, Big data and HPC”, ADAC9 Workshop, Sept. 2020

14.5.5 Oral Talks

7. K. Sato, “End-to-End Optimization for Big Data Processing From Data Sources To Fugaku”, Oct. 2020

8. K. Sato, ”High Performance System Software for Big data processing” in CEA-RIKEN Workshop, Oct. 2020

9. B. Nicolae, K. Sato, “Towards fine-tuning of multi-level checkpointing using machine learning: The case of VeloC” in JLESC Workshop, Sept. 2020

10. K. Sato, “DL4Fugaku: AI frameworks on Fugaku” in JLESC Workshop, Sept. 2020

11. J. Domke, ”Matrix Engines for HPC: A Paragon of Performance or Grasping at Straws?” in ASTAR IHPC ACRC-RIKEN-CREST Deep workshop, Feb. 2021

12. J. Domke, ”MocCUDA: Running CUDA codes on Fugaku” in 12th JLESC Workshop, Feb. 2021

Chapter 15

Data Assimilation Research Team

15.1 Members

Takemasa Miyoshi (Team Leader)

Koji Terasaki (Research Scientist)

Shigenori Otsuka (Research Scientist)

Ting-Chi Wu (Research Scientist)

Takumi Honda (Special Postdoctoral Researcher)

Kohei Takatama (Postdoctoral Researcher)

James Taylor (Postdoctoral Researcher)

Arata Amemiya (Postdoctoral Researcher)

Maha Mdini (Postdoctoral Researcher)

Shun Ohishi (Postdoctoral Researcher)

Yasumitsu Maejima (Postdoctoral Researcher)

Jianyu Liang (Postdoctoral Researcher)

Hideyuki Sakamoto (Technical Staff)

Qiwen Sun (Junior Research Associate)

Iyan Mulia (Research Scientist) *concurrent

Ken Furukawa (Postdoctoral Researcher) *concurrent

Tianfeng Hou (Postdoctoral Researcher) *concurrent

John C. Wells (Senior Visiting Scientist)

Shu-Chih Yang (Senior Visiting Scientist)

Juan Ruiz (Visiting Scientist)

Yohei Sawada (Visiting Scientist)

Pierre Tandeo (Visiting Scientist)

Keichi Kondo (Visiting Scientist)

Atsushi Okazaki (Visiting Scientist)

Shunji Kotsuki (Visiting Scientist)

Hazuki Arakida (Visiting Scientist)

Hironori Arai (Visiting Scientist)

Tobias Necker (Visiting Scientist)

Shlok Mohta (Student Trainee/Visiting Technician)

Kota Takeda (Intern)

Noboru Isobe (Intern)

Futo Tomizawa (Intern)

Minglu Zhao (Intern)

Andrew Pensoneault (Student Trainee)

Cheng Da (Student Trainee)

Paula Maldonado (Student Trainee)

Kenta Kurosawa (Student Trainee)

Yukie Komori (Assistant)

Saeko Imano (Assistant)

Aki Mukunoki (Assistant)

Sakiko Tashiro (Assistant)

15.2 Overview of Research Activities

Data Assimilation Research Team (DA Team) was launched in October 2012 and is composed of 29 research and technical staff including 12 visiting members as of March 2021. Data assimilation is a cross-disciplinary science to synergize computer simulations and real-world data, using statistical methods and applied mathematics. As computers become more powerful and enable more precise simulations, it will become more important to compare the simulations with actual observations. DA Team performs cutting-edge research and development on advanced data assimilation methods and their wide applications, aiming to integrate computer simulations and real-world data in the wisest way. Particularly, DA Team tackles challenging problems of developing efficient and accurate data assimilation systems for “big simulations” with real-world “big data” from various sources including advanced sensors. The specific foci include 1) theoretical and algorithmic developments for efficient and accurate data assimilation, 2) data assimilation methods and applications by taking advantage of powerful supercomputers and “big data” from new advanced sensors, and 3) exploratory new applications of data assimilation in wider simulation fields. These advanced data assimilation studies will enhance simulation capabilities and lead to a better use of supercomputer “Fugaku.”

In FY2020, we continued on the ongoing data assimilation research in the following aspects: 1) theoretical research on challenging problems, 2) leading research on meteorological applications, 3) optimization of computational algorithms, and 4) exploratory research on wider applications. We also explored close collaborations with several research teams within the R-CCS. We have made substantial progress on the following research items:

Press releases

- The impact of oversampled Geostationary Precipitation Radar (GPR) observations on typhoon forecasts was evaluated. (1 paper published in FY2021, 1 press relase)
- Ensemble Kalman filter experiments were conducted with an extended SEIR model for COVID-19. (1 press release in FY2021)
- The SCALE-LETKF “Big Data Assimilation” system with the Multi-Parameter (MP)-PAWR for a 30-second update, 500-m mesh, 30-minute-lead localized rainfall forecast with 50 ensemble size was developed. A real-time demonstration was conducted on Oakforest-PACS during 25 August-7 September, 2020. (1 RIKEN News article)

Theoretical research

- Non-Gaussian PDF in DA was investigated using the Lorenz-63 3 variable model.
- A quasi-Monte method was used to accelerate spin-up of particle filter.
- A particle filter was applied to the cellular automata of 3 state sheep model.
- Well-posedness of higher-order elliptic equations with dynamic boundary conditions was investigated (i.e., mathematical analysis of partial differential equations). (1 paper published)
- Mathematical justification of the hydrostatic approximation in the primitive equations was investigated.
- Mathematical analysis of data assimilation (nudging) of the primitive equations was conducted.
- Weight structure of the Local Ensemble Transform Kalman Filter (LETKF) was investigated with a simplified atmospheric general circulation model (GCM). (1 paper published)
- A local particle filter (LPF) was developed and tested with a simplified GCM.
- Model acceleration using machine learning was investigated. Convolutional neural networks were applied to a quasi-geostrophic model.
- Precipitation nowcasting was investigated using residual neural networks, and the system was applied to the Phased Array Weather Radar (PAWR) data.
- Convolutional neural networks-based forecast verification algorithm was investigated.
- A local particle filter with additive inflation was implemented with the Lorenz 96 model.
- A Kalman filter with cross-correlation between forecast and observation errors was developed for Lorenz 96 model.
- The local particle filter with Gaussian mixture was investigated using the Lorenz-96 model.
- Statistical forecasts for downscaling and bias correction on the Lorenz-96 system was investigated.
- Control simulation experiment with the Lorenz-63 3-variable model was performed. (1 paper published in FY2021)
- Control simulation experiment with the Lorenz-96 40-variable model was performed.
- Non-Gaussian measure in Gaussian filtering problem was investigated.
- Model bias correction using a machine learning method (Long Short Term Memory, LSTM) was explored with the Lorenz96 model.

Leading research on meteorological applications

- The SCALE-LETKF real-time system was ported to Fugaku to conduct a 30-minute lead localized rainfall forecast with 1000 ensemble size.
- Data analysis study was conducted using global reanalysis data on the intra-seasonal variability of the Asian monsoon (Tibetan) anticyclone. (1 paper published)
- Predictability of the July 2018 heavy rain event in Japan associated with typhoon Prapiroon and southern convective disturbances was investigated.
- Comparison between lightning flash and infrared radiance observations was conducted with idealized observing system simulation experiments.
- Assimilation of Himawari-8 infrared radiances in a heavy precipitation event in Taiwan was performed.
- An advanced operation system was developed for hydroelectric dam using machine learning.
- Predictability of the July 2020 heavy rainfall event was investigated with the SCALE-LETKF. (1 paper published in FY2021)
- Non-Gaussian PDF in DA was investigated using 1000-member, 1-km-resolution assimilation experiments with assimilation windows ranging from 30 seconds to 5 minutes and assimilating PAWR observations in the SCALE-LETKF system. (1 paper published in FY2021)
- Conventional radar data assimilation experiments were conducted using the SCALE-LETKF system and RELAMPAGO field campaign observations. The performance of the analysis and short range forecasts was analyzed.
- Conventional radar data assimilation experiments were conducted using the WRF-LETKF system and RELAMPAGO field campaign observations. The performance of the analysis and short range forecasts was analyzed.
- EFSO was implemented to SCALE-LETKF and the impact of surface observations on rainfall forecast was evaluated by EFSO.
- An observation operator of the lightning data assimilation was improved.
- Impacts of rich PAWR data on the forecasts for the July 2020 heavy rainfall event were investigated by 1-km-mesh, 30-second-update SCALE-LETKF.
- A control system experiment for a heavy rainfall event in Hiroshima was conducted.

- Distributions and convergence of forecast variables in a 1000-member convection-permitting ensemble were investigated.
- Vertical localisation for convective-scale data assimilation using a 1000-member ensemble was investigated.
- Correlation structure and optimal localisation of visible and infrared satellite observations were investigated.
- Predictability of a coupled General Circulation Model (GCM) was investigated using an atmospheric model SPEEDY and an ocean model NEMO for online data assimilation of paleoclimate.
- Impact of frequency of Tropical Cyclone (TC)-vital assimilation cycle for reproducing Typhoon Soudelor (2015) in SCALE-LETKF was investigated.
- The three-dimensional precipitation nowcasting system was continuously operated with the PAWR at NICT Kobe and the MP-PAWR at Saitama.
- A machine-learning system for the three-dimensional precipitation nowcasting system was developed with Convolutional LSTM. The use of future data from numerical weather prediction (NWP) improved the prediction accuracy.
- A quality control system for MP-PAWR observations in shadow areas behind mountains and buildings was developed.
- GPM/DPR-sensed heavy ice precipitation was compared with a 3.5-km NICAM simulation to design better cloud microphysics.
- GPM/DPR is assimilated to estimate cloud microphysics model parameter.
- Advantage of hybrid background error covarinace is investigated with SPEEDY-LETKF. (1 paper published)
- A machine learning approach to the observation operator for satellite radiance data assimilation was investigated. (1 paper submitted)
- The predictability of the heavy rainfall event in Kyushu in 2020 was investigated with 56-km resolution and 1,024-member NICAM-LETKF system.
- Sensitivity tests to optimize the inflation parameter was investigated with the treatment of observation error covariance matrix.
- Development of EFSO for the latest version of NICAM-LETKF was started.
- A control system experiment for a typhoon was conducted with NICAM-LETKF.
- Impact of incremental analysis updates and covariance inflation methods was investigated using an LETKF-based ocean data assimilation system.
- Impact of an adaptive observation error inflation method was investigated using an LETKF-based ocean data assimilation system.
- Simulating rapid water level change in Lake Biwa due to Typhoon Jebi (2018) was investigated by using a regional ocean model (ROMS).
- Bio-chemical simulation in Tokyo Bay for predicting blue-green algae bloom was investigated by using a regional ocean model (ROMS).
- A nowcasting system was developed for blue green algae bloom in Lake Kinneret based on a particle tracking method.

Computational optimization

- The NICAM-LETKF system was developed for “Fugaku” as a target application in collaboration with the Computational Climate Science Research Team and FS2020. (Selected as Gordon Bell Finalist)
- Direct GPM/DPR assimilating capability was implemented into a global numerical weather prediction model using NICAM-LETKF. A prototype with GPM DPR assimilation capability using the Joint Simulator is made available for the latest NICAM-LETKF system.

Wider applications

- Bibliometric analysis on mathematics with classification tree was conducted. (1 paper accepted)
- Particle filter experiments with discrete extended SEIR model for COVID-19 were performed.
- Venus data assimilation was investigated.
- Particle filter experiments were conducted, aiming to improve springback estimation after plate processing.

Several achievements are selected and highlighted in the next section.

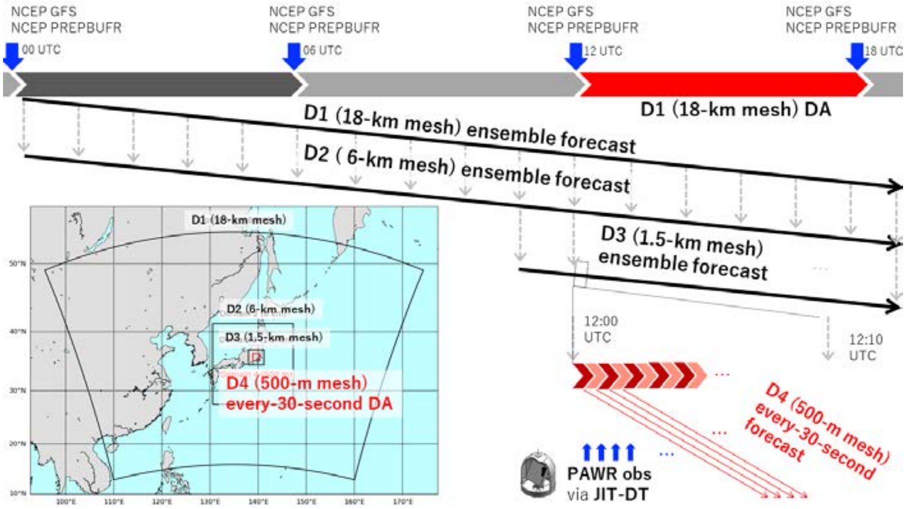


Figure 15.1: Workflow of the real-time prediction system.

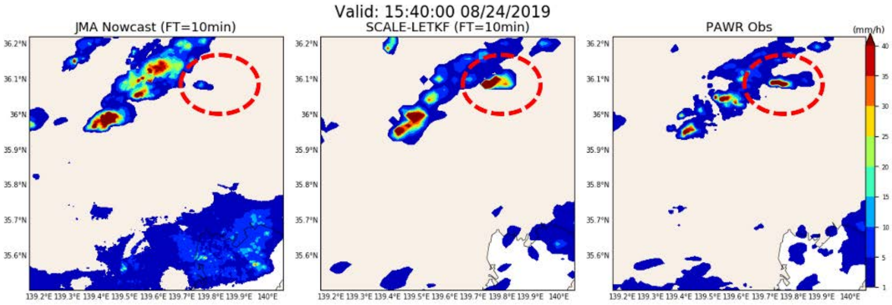


Figure 15.2: An example of 10-minute precipitation forecast at 15:40 UTC, 24 August, 2019. Color shadings represent rain rate. Left: Japan Meteorological Agency's operational precipitation nowcast. Middle: Prediction by the SCALE-LETKF. Right: PAWR observation.

15.3 Research Results and Achievements

15.3.1 "Big Data Assimilation" for predicting sudden severe rainstorms

This research aims to perform demonstration experiments of an innovative "Big Data Assimilation" system realizing 30-second-update 30-minute-lead sudden torrential rain forecast by fully taking advantage of the phased array weather radar, geostationary weather satellite Himawari-8, and Oakforest-PACS. In FY2020, a real-time demonstration was performed on Oakforest-PACS during 25 August-7 September, 2020, and the forecasts were disseminated via RIKEN's website and a smartphone application by MTI Ltd. This research also aims to propose requirements of the computational power and communication speed for a possible future operational implementation, and to guide future research and development of observing systems, data assimilation, and numerical weather prediction models.

Following the achievements in past years, we performed data assimilation experiments using the SCALE-LETKF, phased-array weather radar (PAWR), and Himawari-8 geostationary weather satellite. A workflow for real-time 30-second-update analysis with a 500-m mesh and subsequent 30-minute forecast with SCALE-LETKF assimilating the PAWR was implemented (Fig. 15.1), including real-time data transfer from PAWR, four nested domains with the outer-most domain of 6-hourly update and 18-km mesh, and dissemination of the forecast on a webpage and a smartphone app. We conducted detailed analyses on the forecast accuracy for several rain cases and investigated optimal parameters to improve the forecast. Figure 15.2 indicates that the SCALE-LETKF can predict intensification of a convective cell (middle panel, red circle) during the forecast experiment for 24 August, 2019, while Japan Meteorological Agency's nowcast (left) misses the intensification of the same convection.

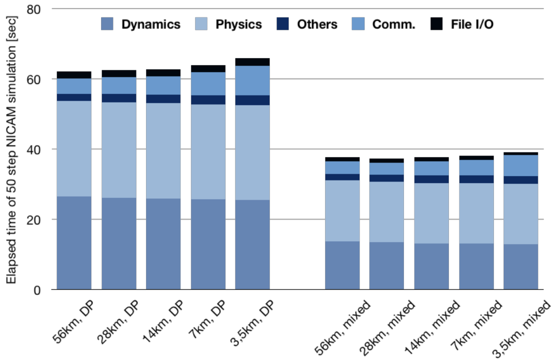


Figure 15.3: Weak scaling test for a single member NICAM simulation. Adopted from Yashiro et al. (2020).

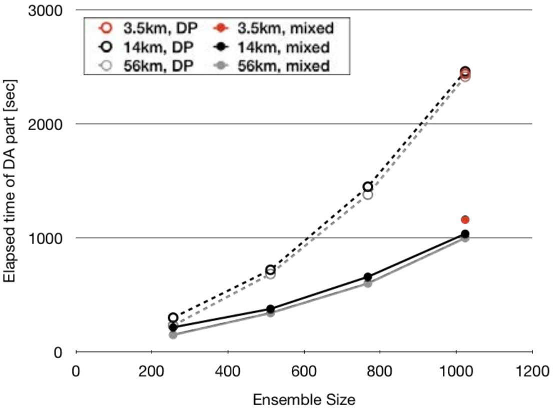


Figure 15.4: Elapsed time of the DA part in the NICAM-LETKF. Adopted from Yashiro et al. (2020).

15.3.2 A 3.5-km resolution and 1024-member numerical weather prediction with Fugaku

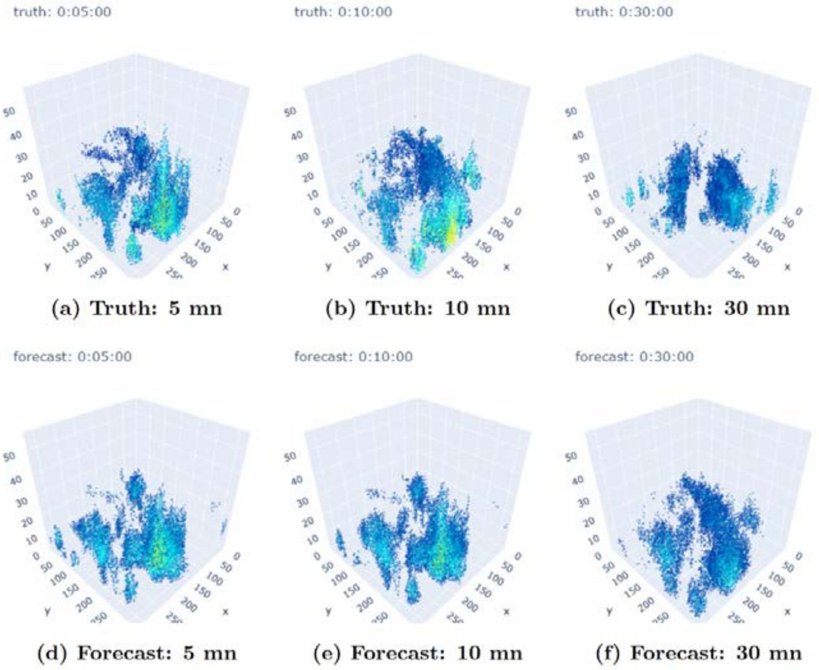
The NICAM-LETKF system has been developed in the codesign project so that we can effectively perform high resolution and big ensemble numerical weather prediction with Fugaku. We aimed at getting 100 times performance on Fugaku compared with K computer. We redesigned the NICAM-LETKF system to solve the bottleneck of memory-intensive applications and file input/output (I/O). The NICAM-LETKF system was redesigned to read and write files so that it can use a local file system consisted of a solid-state drive. For computational aspect, we found out “hot spots” in whole source codes by measuring the elapse time. These hot spots were optimized considering the architecture of Fugaku. Finally, we succeeded running 3.5-km horizontal resolution and 1,024-member data assimilation computation using 131,072 nodes (6,291,456 cores) of Fugaku (Figs. 15.3–15.4). Based on the result, we were selected as the finalist of the Gordon Bell Prize at the International Conference for High Performance Computing, Networking, Storage and Analysis (SC20) (Yashiro et al. 2020).

15.3.3 3D Precipitation Nowcasting: RESNet applied to Highly Dense PAWR Data

Sudden heavy rain may lead to disasters like flooding and loss of life and property. To reduce the risk, predicting sudden downpours is of key importance. However, predictability of such events is limited to only for a very short range within an hour or shorter because of their abruptness. In this case nowcasting is an effective approach. Detecting sudden heavy rain even 10 minutes before it occurs can reduce the damage drastically. Precipitation nowcasting is the process of short-range prediction based on observation data. In the case of sudden rainfalls, this process is difficult due to the fast evolution of the rain and its chaotic nature. Therefore, we need innovative techniques. The novel PAWR offers dense 3D images of reflectivity every 30 seconds. We took advantage of this big data to perform nowcasting using neural networks (NN). Our objective is to predict rain evolution within a short time interval with a fair accuracy.

We use Residual Neural Network (RESNet) to compress the images and extract information relevant for the prediction. Next, we use a Convolutional Neural Network (CNN) to make the prediction. Our first results show that we can predict precipitations up to 30 minutes, with the threat score going from 0.85 at 5 minutes to 0.55 at 30 minutes. Figure 15.5 shows an example of the outputs. The use of the RESNet allowed to alleviate the memory load and the computational complexity of the prediction. Moreover, training the RESNet and the CNN jointly reduced immensely the prediction noise in non-precipitation regions and improved the accuracy in precipitation regions.

At this stage, we have proved the possibility of performing precipitation nowcasting using NNs and we have studied the predictability range. The next steps would include training the model using a larger dataset and tuning the model to improve the prediction accuracy. Moreover, our system fails to predict some case of rapid precipitation evolution. We would identify these cases and explore advanced techniques to address them.



(a) Truth: 5 mn (b) Truth: 10 mn (c) Truth: 30 mn

(d) Forecast: 5 mn (e) Forecast: 10 mn (f) Forecast: 30 mn

Figure 15.5: Nowcasting Example.

15.4 Schedule and Future Plan

DA Team aims to explore the frontier of large-scale DA problems, and to be a world's leader in DA research. With the goals in mind, we plan to continue working on the three fundamental foci: 1) theoretical and algorithmic developments for efficient and accurate DA (core focus), 2) DA methods and applications by taking advantage of the Japan's flagship supercomputer and "Big Data" from new advanced sensors (lead focus), and 3) exploratory new applications of DA in wider simulation fields (pioneer focus). We have very strong projects in weather forecast applications, and we will enhance the leading research to the world's top level. Also, we will pioneer new application fields that have direct connection with societal benefits. For direct benefit to society, "real-time" application is essential with efficient computational algorithms, which are also an important aspect of our research.

"Big Data Assimilation" (BDA) is one of the major activities that we have developed in the past years. Real-time demonstration of the 30-second update, 500-m mesh, 30-minute lead forecast was conducted for the first time in the world. The real-time forecast experiments showed promising results, but the physical performance for the 30-minute forecast of precipitation patterns can be improved. We will continue to work on the development of the BDA system to further improve the computational and physical performances using "Fugaku."

Beyond the direct future of the BDA effort, we can extend the idea of BDA to a broader perspective: integration of "Big Data" and "Big Simulation." New sensors provide orders of magnitude more data, and simulations become more precise. Collaborative work with computer scientists will be essential to utilize the complex high-performance computer systems effectively. In addition, dense sensor data tend to have complicated error structures such as correlated errors, and the proper treatment is necessary to fully utilize the "Big Data." The current DA methods usually assume no observation error correlation. Based on our previous theoretical research on the observation-error correlations, we plan to develop methods to consider the observation error correlations in realistic NWP applications.

Treating the model errors and non-Gaussian probability distribution has been grand challenges in DA. "Big Ensemble Data Assimilation" with the largest-ever 10240 samples was a milestone providing fundamental data to investigate the non-Gaussian probability distribution. We have developed expertise and exclusive dataset to tackle these challenges. We have been pioneering a new implementation of Local Particle Filter (LPF) in collaboration with German Weather Service. We will continue the LPF development toward realistic applications.

We started to investigate a new research area, weather controllability beyond predictability. If we can modify the intensity, timing, and range of the extreme weather that leads to disasters, it may be possible to avoid direct damage or and greatly reduce damage. Under the Moonshot R&D MILLENNIA Program, in FY2020, we conducted a basic survey, theoretical investigations of controllability of chaotic dynamical systems, and a series of control system experiments.

DA is a cross-disciplinary science based on statistical mathematics and dynamical systems theory. In addition, DA connects simulations and real-world data. Therefore, it is naturally beneficial to enhance close collaborations with experts in mathematics, sensor technology, and various application fields. The current weather-forecast projects involve active collaborations with observation experts. In addition, we have been teaching semester-long courses on DA under the MACS (MAthematics-based Creation of Science Program), Graduate School of Science, Kyoto University, and this will lead to more cross-disciplinary collaborations. Moreover, TL Miyoshi has joint appointments at RIKEN iTHEMS (Interdisciplinary Theoretical and Mathematical Sciences Program) and RIKEN CPR (Cluster for Pioneering Research), and these will enhance broader collaborations. Collaborations with other R-CCS Research Teams will also be beneficial. The challenges for future DA systems require cross-disciplinary collaborations to most effectively use the massive supercomputers with more heterogeneous architecture design. Further, RIKEN's Engineering Network already gave us opportunities to collaborate among different disciplines among RIKEN centers. RIKEN started a new pioneering project titled "Prediction for Science" in April 2020 for 5 years. The CPR Prediction Science Laboratory is a sister laboratory of DA Team and pioneers a new science of prediction as the 5th paradigm fusing the computational science (3rd science) and the data science (4th science) through tight collaborations within and beyond RIKEN. We also started collaborations with industry partners for more direct benefit to society. Enhancing the broader collaborations is a key to success, i.e., to make a new scientific movement across the borders through DA as an innovation hub, to establish DA as a new scientific paradigm, and to change the world through innovation and education of DA.

15.5 Publications

15.5.1 Awards

[1] Takemasa Miyoshi, Commendation by the Prime Minister for Disaster Prevention

[2] Shun Ohishi, Young Author Award, Oceanographic Society of Japan, "Frontolysis by surface heat flux in the eastern Japan Sea: importance of mixed layer depth"

15.5.2 Articles/Journal

[1] Chang, C., S. G. Penny, and S. Yang, 2020: Hybrid gain data assimilation using variational corrections in the subspace orthogonal to the ensemble. *Mon. Wea. Rev.*, **148**, 2331–2350.

[2] Hsiang-Wen Cheng, Shu-Chih Yang, Yu-Chieng Liou, Ching-Sen Chen, 2020: An investigation of the sensitivity of predicting a severe rainfall event in northern Taiwan to the upstream condition with a WRF-based radar data assimilation system, *SOLA*, **16**, 97–103.

[3] Wu, P., S. Yang, C. Tsai, and H. Cheng, 2020: Convective-scale sampling error and its impact on the ensemble radar data assimilation system: a case study of a heavy rainfall event on 16 June 2008 in Taiwan. *Mon. Wea. Rev.*, **148**, 3631–3652.

[4] Tandeo, P., P. Ailliot, M. Bocquet, A. Carrassi, T. Miyoshi, M. Pulido, and Y. Zhen, 2020: A review of innovation-based methods to jointly estimate model and observation error covariance matrices in ensemble data assimilation. *Mon. Wea. Rev.*, **148**, 3973–3994.

[5] Sawada, Y., 2020: Machine learning accelerates parameter optimization and uncertainty assessment of a land surface model, *J. Geophys. Res. Atmos.*, **125(20)**, e2020JD032688.

[6] Kotsuki, S., Pensoneault, A., Okazaki, A. and Miyoshi, T., 2020: Weight structure of the local ensemble transform Kalman filter: a case with an intermediate AGCM. *Q. J. R. Meteorol. Soc.*, **146(732)**, 3399–3415.

[7] Amemiya, A. and Sato, K., 2020: Characterizing quasi-biweekly variability of the Asian monsoon anticyclone using potential vorticity and large-scale geopotential height field. *Atmos. Chem. Phys.*, **20**, 13857–13876.

[8] Tomita, H., M. F. Cronin, and S. Ohishi, 2021: Asymmetric air-sea heat flux response and ocean impact to synoptic-scale atmospheric disturbances observed at JKEO and KEO buoys, *Scientific Reports*, **11**, 469(2021).

[9] Furukawa, K., Kajiwara, N., 2020: Maximal Lp–Lq regularity for the quasi-steady elliptic problems. Journal of Evolution Equations. `https://doi.org/10.1007/s00028-020-00638-2`

[10] Miyoshi, T., S. Kotsuki, K. Terasaki, S. Otsuka, G.-Y. Lien, H. Yashiro, H. Tomita, M. Satoh, and E. Kalnay, 2020: Precipitation ensemble data assimilation in NWP models. In: Levizzani V., Kidd C., Kirschbaum D., Kummerow C., Nakamura K., Turk F. (eds) *Satellite Precipitation Measurement.* Advances in Global Change Research, 69, Springer, 983–991.

[11] Mulia, I. E. and Satake, K., 2021: Synthetic analysis of the efficacy of the S-net system in tsunami forecasting. *Earth, Planets and Space*, **73**, 36, `https://doi.org/10.1186/s40623-021-01368-6`

[12] Taylor, J., Honda, T., Amemiya, A., Maejima, Y., and Miyoshi, T., 2020: Predictability of the July 2020 heavy rainfall with the SCALE-LETKF. *SOLA*, **17**, 48–56.

15.5.3 Conference Papers

[1] H. Yashiro, K. Terasaki, Y. Kawai, S. Kudo, T. Miyoshi, T. Imamura, K. Minami, H. Inoue, T. Nishiki, T. Saji, M. Satoh, and H. Tomita, “A 1024-Member Ensemble Data Assimilation with 3.5-Km Mesh Global Weather Simulations,” in SC20: International Conference for High Performance Computing, Networking, Storage and Analysis (SC), Atlanta, GA, US, 2020 pp. 1–10, `https://dl.acm.org/doi/pdf/10.5555/3433701.3433703`

15.5.4 Invited Talks

[1] Takemasa Miyoshi, Weather Predictability and Data Assimilation: Perspectives Toward Prediction and Control in Agriculture, LSSE2020, onlilne, 2020/4/24.

[2] 三好建正、ビッグデータ同化：ゲリラ豪雨予測から予測科学へ、JST未来社会創造事業ワークショップ 数理科学がドライブする未来、オンライン, 2020/6/3.

[3] Kotsuki S., Pensoneault A., Okazaki A. and Miyoshi T.: Weight Structure of the Local Ensemble Transform Kalman Filter: A Case with an Intermediate AGCM. JpGU-AGU Joint Meeting 2020, online, 2020/7/13.

[4] Kotsuki, S., Miyoshi, T., Kondo, K. and Potthast, R.: A Local Particle Filter and Its Gaussian Mixture Extension: Experiments with an Intermediate AGCM. RIKEN Data Assimilation Seminar, online, 2020/9/11.

[5] Keiichi Kondo, Shunji Kotsuki, Takemasa Miyoshi, “A local particle filter based on non-Gaussian statistics using an intermediate AGCM,” DA seminar, online, 2020/9/11.

[6] Takemasa Miyoshi (RIKEN),Big Data, Big Computation, and Machine Learning in Numerical Weather Prediction, Workshop on Data Assimilation and Uncertainty Quantification at the exascale, online, 2020/9/24.

[7] Takemasa Miyoshi, Big Data, Big Computation, and Machine Learning in Numerical Weather Prediction,Virtual Event: ECMWF-ESA Workshop on Machine Learning for Earth System Observation and Prediction, online, 2020/10/6.

[8] 三好 建正, ビッグデータとスーパーコンピュータによる豪雨予測 —世界最先端「ビッグデータ同化」の気象予測研究—,第25回日本難病看護学会 第8回日本難病医療ネットワーク学会合同学術集会, オンライン, 2020/11/20.

[9] Takemasa Miyoshi, Predicting Sudden Local Storms by 30-second-update NWP Using Phased Array Weather Radar, KU-ITB Biweekly Webinar Series, online, 2020/11/27.

[10] Takemasa Miyoshi,T. Honda, A. Amemiya, S. Otsuka, Y. Maejima, J. Taylor, H. Tomita, S. Nishizawa, K. Sueki, T. Yamaura, Y. Ishikawa, S. Satoh, T. Ushio, K. Koike, E. Hoshi, and K. Nakajima, Big Data Assimilation: Real-Time Demonstration Experiment of 30-s-Update Forecasting in Tokyo in August 2020, American Meteorological Society 101st Annual meeting, online, 2021/1/12.

[11] 三好建正, ビッグデータ同化 ゲリラ豪雨予測から、予測科学へ, JST/CRDSセミナー「数学と科学、工学の協働に関する連続セミナー」 第13回「シミュレーションとデータ科学」, オンライン, 2021/1/27.

[12] Takemasa Miyoshi, Big Data, Big Computation, and Machine Learning in Numerical Weather Prediction, AI Chair OceaniX Webinars, IMT-Atlantique & RIKEN Online Joint Seminar Series (Jointly with Data Assimilation Seminar Series)2021, online, 2021/2/17.

[13] 三好建正,データ同化と気象予測の展望、JST未来社会創造事業ワークショップ、「次世代情報社会の実現」領域　R03重点公募テーマ検討ワークショップ、オンライン, 2021/2/23.

[14] Takemasa Miyoshi, Fusing Big Data and Big Computation in Numerical Weather Prediction, Climate Research with HPC Forum, SupercomputingAsia 2021, online, 2021/3/4.

[15] Serge RICHARD, Qiwen SUN, Bibliometric analysis on mathematics, 3 snapshots: 2005, 2010, 2015, Himeji conference on partial differential equations, online, 2021/3/5.

[16] 寺崎康児、三好建正、全球水平解像度56km・1024メンバーのNICAM-LETKFを用いた令和2年7月豪雨実験、第54回メソ気象研究会、オンライン, 2021/3/8.

15.5.5 Oral Talks

[1] 三好 建正, "Enhancing Precipitation Prediction Algorithm by Data Assimilation of GPM Observations," 「NICAM-LETKFと衛星観測データ同化に関する打ち合わせ」第20回会合, オンライン, 2020/4/22.

[2] Takemasa Miyoshi, Shunji Kotsuki, Koji Terasaki, Shigenori Otsuka, Ying-Wen Chen, Kaya Kanemaru, Masaki Satoh, Hisashi Yashiro, Hirofumi Tomita, Keiichi Kondo, Kozo Okamoto, Eugenia Kalnay, and Takuji Kubota, Enhancing Precipitation Prediction Algorithm by Data Assimilation of GPM Observations, EGU2020, online, 2020/5/7.

[3] Necker, T., Weissmann, M., Ruckstuhl, Y., Ruiz, J., Miyoshi, T., and Anderson, J.: Sampling errors on convective scales: What can we learn from a 1000-member ensemble?, EGU General Assembly 2020, online, 2020/5/8.

[4] Takemasa Miyoshi, Takumi Honda, Shigenori Otsuka, Arata Amemiya, Yasumitsu Maejima, Yoshihiro Ishikawa, Hiromu Seko, Yoshito Yoshizaki, Naonori Ueda, Hirofumi Tomita, Yutaka Ishikawa, Shinsuke Satoh, Tomoo Ushio, Kana Koike, and Yasuhiko Nakada, Big Data Assimilation: Real-time Workflow for 30-second-update Forecasting and Perspectives toward DA-AI Integration, EGU2020, online, 2020/5/8.

[5] 寺崎康児、三好建正、観測ビッグデータ同化に向けたマイクロ波サウンダデータの水平観 測誤差相関を考慮したデータ同化、日本気象学会2020年度春季大会、予稿集のみ, 2020/5/19.

[6] 本田匠, 佐藤陽佑, 三好建正, 雷観測のデータ同化へ向けた基礎的な調査, 日本気象学会2020年度春季大会, 予稿集のみ, 2020/5/21.

[7] 前島康光、牛尾知雄、三好建正、雷観測データBOLTの同化に向けた観測演算子の設計調査、日本気象学会2020年度春季大会、予稿集のみ, 2020/5/23.

[8] H.Arai, W.Takeuchi, K. Oyoshi, K. Terasaki, T. Miyoshi, H.Yashiro, H. Phan, L.D. Nguyen, T. Fumoto, K. Inubushi, T.L. Toan. Approach toward global methane emission using GOSAT data and a bottom-up method based on SAR data in tropical rice cropping wetlands. IWGGMS-16, EUMETSAT, online, 2020/6/2-5.

[9] Qiwen Sun, Serge Richard, Takemasa Miyoshi, SIIR Model and Parameter Estimation, Prediction Science, covid-19 meeting, online, 2020/6/10.

[10] Takemasa Miyoshi, Takumi Honda, Shigenori Otsuka, Arata Amemiya, Yasumitsu Maejima, Yoshihiro Ishikawa, Hiromu Seko, Yoshito Yoshizaki, Naonori Ueda, Hirofumi Tomita, Yutaka Ishikawa, Shinsuke Satoh, Tomoo Ushio, Kana Koike, Yasuhiko Nakada, Big Data Assimilation: Real-time Workflow for 30-second-update Forecasting and Perspectives toward DA-AI Integration, JpGU-AGU Joint Meeting 2020, online, 2020/7/12.

[11] Takemasa Miyoshi, Shunji Kotsuki, Keiichi Kondo, Roland Potthast, Local Particle Filter Implemented with Minor Modifications to the LETKF Code, JpGU-AGU Joint Meeting 2020, online, 2020/7/13.

[12] H.Arai, W.Takeuchi, K. Oyoshi, K. Terasaki, T. Miyoshi, H.Yashiro, H. Phan, L.D. Nguyen, T. Fumoto, K. Inubushi, T.L. Toan. Global methane emission quantification based on GOSAT data and a bottom-up method based on SAR data in tropical rice paddies. Ozflux 20years anniversary. Australia, online, 2020/7/16-17.

[13] Takemasa Miyoshi, “Enhancing Precipitation Prediction Algorithm by Data Assimilation of GPM Observations,” 「NICAM-LETKFと衛星観測データ同化に関する打ち合わせ」第21回会合, オンライン, 2020/7/30.

[14] 大石俊，日原勉，相木秀則，石坂丞二，宮澤泰正，可知美佐子，三好建正「アンサンブルカルマンフィルタを用いた海洋データ同化システムの開発」，第24回海洋データ同化夏の学校，オンライン, 2020/8/6.

[15] Takemasa Miyoshi, Real-time Demonstration of Big Data Assimilation in Numerical Weather Prediction, The 197th R-CCS Café, online, 2020/9/7.

[16] Ken Furukawa, Maximal Regularity and Partial Differential Equations, iThems Math Seminar, online, 2020/9/8.

[17] Qiwen Sun, Serge Richard, Takemasa Miyoshi, Understand the infectious disease model of covid-19 by using Ensemble Kalman Filter, Prediction Science COVID-19 Meeting, online, 2020/9/9.

[18] 雨宮新、局地的豪雨のリアルタイム実証実験、「富岳」成果創出加速プログラム「防災・減災に資する新時代の大アンサンブル気象・大気環境予測」第1回シンポジウム～豪雨・台風の高精度な予測をめざして～、オンライン, 2020/9/26.

[19] 三好建正, ゲリラ豪雨予測のリアルタイム実証実験, 第9回JCAHPCセミナー（第4回OFP利活用報告会）「人類と地球を護るスーパーコンピューティング」, online, 2020/10/15.

[20] 雨宮新、Shlok Mohta、 三好建正、 機械学習を用いたモデルバイアス補正：Lorenz96モデル実験（続報）、日本気象学会2020年度秋季大会、オンライン, 2020/10/28.

[21] 本田匠, 雨宮新, 大塚成徳, Guo-Yuan Lien, James Taylor, 前島康光, 西澤誠也, 山浦剛, 末木健太, 富田浩文, 佐藤晋介, 石川裕, 小池佳奈, 星絵理香, 三好建正, “SCALE-LETKF による 30 秒更新 30 分予報のリアルタイム実験,” 日本気象学会2020年度秋季大会, オンライン, 2020/10/28.

[22] 本田匠, 佐藤陽祐 , 三好建正, “静止衛星による雷観測データ同化の観測システム シミュレーション実験,” 日本気象学会2020年度秋季大会, オンライン, 2020/10/30.

[23] Koji Terasaki, Takemasa Miyoshi, Including the horizontal observation error correlation in the assimilation of AMSU-A data, ECMWF/EUMETSAT NWP SAF Workshop on the treatment of random and systematic errors in satellite data assimilation for NWP, online, 2020/11/4.

[24] 寺崎康児, 三好建正, 全球水平解像度56km・1024メンバーのNICAM-LETKFを用いた令和2年7月豪雨の事例実験. 第22回非静力学モデルに関するワークショップ. オンライン, 2020/11/11.

[25] 雨宮新,本田匠, 三好建正, 2020年夏のリアルタイム実証実験における埼玉MP-PAWR30秒同化システム開発, 第22回非静力学モデルに関するワークショップ, オンライン, 2020/11/12.

[26] 近藤圭一, 三好建正 ”非ガウス分布の定量的評価による非ガウスデータ同化,” 第22回非静力学モデルに関するワークショップ, オンライン, 2020/11/12.

[27] 30 秒ごとに更新するゲリラ豪雨予報−首都圏でのリアルタイム実証実験−,三好 建正，本田 匠，雨宮 新，大塚 成徳，前島 康光，James Taylor，富田 浩文，西澤誠也，末木 賢太，山浦 剛，石川 裕，佐藤 晋介，牛尾 知雄，小池 佳奈，星 絵里香，中島研吾、第22回非静力学モデルに関するワークショップ,オンライン, 2020/11/12.

[28] Yasumitsu Maejima, Impacts of dense surface observations on predicting a torrential rainfall event on September 9 and 10, 2015 in Ibaraki and Tochigi prefectures., The 202nd R-CCS Café, online, 2020/11/13.

[29] Tianfeng Hou: Efficient probabilistic assessment of building performance: sequential Monte Carlo and decomposition methods, iTHEMS Math Seminar, online, 2020/11/13.

[30] Koji Terasaki, Takemasa Miyoshi, Including the horizontal observation error correlation in the assimilation of AMSU-A data, Data assimilation seminar, online, 2020/11/20.

[31] 大石俊，日原勉，相木秀則，石坂丞二，宮澤泰正，可知美佐子，三好建正，アンサンブルカルマンフィルタを用いた海洋データ同化システムの力学的非平衡の改善，日本海洋学会2020年度秋季大会、オンライン, 2020/11/27.

[32] 雨宮新, β面浅水系モデルを用いたアジアモンスーン高気圧の季節内変動の研究, 日本海洋学会2020年度秋季大会、オンライン, 2020/11/27.

[33] Tobias Necker, “Sampling errors and observation impact: What can we learn from a convective-scale 1000-member ensemble?”, IMGW Master Seminar, online, 2020/12/9.

[34] Takemasa Miyoshi RIKEN Enhancing Precipitation Prediction Algorithm by Data Assimilation of GPM Observations, The Joint PI Meeting of JAXA Earth Observation Missions FY2020, online, 2020/12/21.

[35] 大石俊，日原勉，相木秀則，石坂丞二，宮澤泰正，可知美佐子，三好建正，アンサンブルカルマンフィルタを用いた海洋データ同化システムの力学的非平衡の改善，研究集会「宇宙地球環境の理解に向けての統計数理的アプローチ」，オンライン, 2020/12/25.

[36] Qiwen Sun, Takemasa Miyoshi, Serge Richard, Ensemble Kalman filter experiments with an extended SIR model for COVID-19, The 205th R-CCS Cafe, online, 2021/1/21.

[37] Takemasa Miyoshi, Innovating “Big Data Assimilation” technology for revolutionizing very-short-range severe weather prediction, CREST International Symposium on Big Data Application, online, 2021/1/23.

[38] James Taylor, Atsushi Okazaki, Moeka Yamaji, Takuji Kubota, Riko Oki, Takemasa Miyoshi “Oversampling Reflectivity Observations from a Geostationary Precipitation Radar Satellite: Impact to Typhoon Forecasts within a Perfect Model OSSE Framework”, ISDA, online, 2021/2/5.

[39] Qiwen Sun, Takemasa Miyoshi, Serge Richard, Ensemble Kalman filter experiments with an extended SIR model for COVID-19, The 12th RIKEN-Kyoto University data assimilation workshop, online, 2021/2/10.

[40] Takemasa Miyoshi, Fugaku’s Illuminating a Path to the Future of Numerical Weather Prediction, Session11Distinguished Achievements in AI, Big Data and Simulations supporting Society5.0, The 3rd R-CCS International Symposium, online, 2021/2/16.

[41] 寺崎康児、三好建正、全球水平解像度56km・1024メンバーのNICAM-LETKFを用いた令和2年7月豪雨実験、第2回富岳成果創出加速プログラムテーマ1サブテーマA研究会、オンライン, 2021/2/18.

[42] Takemasa Miyoshi, Big data assimilation and AI: Creating new development in real-time weather prediction, ERCIM-JST Joint Symposium on Big Data and Artificial Intelligence, online, 2021/2/18.

[43] 大石俊，日原勉，相木秀則，石坂丞二，宮澤泰正，可知美佐子，三好建正「アンサンブルカルマンフィルタを用いた海洋データ同化システムの力学的非平衡の改善」，第11回データ同化ワークショップ，オンライン, 2021/2/24.

[44] 近藤圭一, 三好建正 “背景誤差の非ガウス性定量化による非ガウスデータ同化手法,” 第2回先端的データ同化と巨大アンサンブル手法に関する研究会, オンライン, 2021/3/9.

[45] 雨宮新, 三好建正, “1000メンバーの18km解像度SCALE-LETKFを用いたアンサンブルサイズ依存性調査,” 第2回先端的データ同化と巨大アンサンブル手法に関する研究会, オンライン, 2021/3/9.

[46] Jianyu (Richard) Liang: A Machine Learning Approach To The Observation Operator For Satellite Radiance Data Assimilation, 金星大気の観測・シミュレーション・データ同化に関する研究会(Workshop on Venus Atmosphere), online, 2021/3/10.

[47] 小槻峻司, 寺崎康児, 佐藤正樹, 三好建正: GPM DPRデータ同化によるNICAM雲微物理パラメータ推定、GPMおよび衛星シミュレータ合同研究集会、オンライン, 2021/3/15.

[48] 寺崎康児、三好建正、全球水平解像度56km・1024メンバーのNICAM-LETKFを用いた令和2年7月豪雨実験、「富岳」成果創出加速プログラム防災・減災に資する新時代の大アンサンブル気象・大気環境予測　令和2年度成果発表会、オンライン, 2021/3/16.

[49] 三好建正、ゲリラ豪雨予測のリアルタイム実験、「富岳」成果創出加速プログラム　防災・減災に資する新時代の大アンサンブル気象・大気環境予測　第1回成果発表会、オンライン, 2021/3/16.

[50] Maha Mdini, Shigenori Otsuka, Takemasa Miyoshi, "Accelerating Climate Model Computation by Neural Networks." IMT-Atlantique & RIKEN Online Joint Seminar Series, online, 2021/3/18.

[51] Arata Amemiya and Takemasa Miyoshi, "Connecting Data Assimilation and Neural ODEs." IMT-Atlantique & RIKEN Online Joint Seminar Series, online, 2021/3/31.

15.5.6 Posters

[1] 三好建正、ゲリラ豪雨予測のリアルタイム実証実験、学際大規模情報基盤共同利用・共同研究拠点（JHPCN)第12回 シンポジウム、オンライン, 2020/7/9.

[2] Honda, T., Y. Sato, T. Miyoshi, Exploring the potential of assimilating lightning flash observations with an ensemble Kalman filter, JpGU-AGU Joint Meeting 2020, online, 2020/7/12.

[3] Y. Maejima, Tomoo Ushio, Takemasa Miyoshi, Toward assimilation of dense and frequent 3-D lightning location data on a severe local rainfall, JpGU-AGU Joint Meeting2020, online, 2020/7/12.

[4] M. Mdini, S. Otsuka, T. Miyoshi, Precipitation Nowcasting Based on Convolutional Neural Networks, JPGU-AGU Joint Meeting 2020, online, 2020/7/12.

[5] Shigenori Otsuka, Yasumitsu Maejima, Pierre Tandeo, Takemasa Miyoshi, Toward an integrated NWP-DA-AI system for precipitation prediction, JpGU-AGU Joint Meeting 2020, online, 2020/7/12.

[6] Koji Terasaki and Takemasa Miyoshi, "Accounting for the horizontal observation error correlation of satellite radiances in data assimilation," JpGU, online, 2020/7/12.

[7] Sawada, Y., Machine learning accelerates parameter optimization and uncertainty assessment of an ecosystem model, JpGU-AGU joint meeting, online, 2020/7/13.

[8] S. Ohishi, T. Hihara, H. Aiki, J. Ishizaka, Y. Miyazawa, M. Kachi, and T. Miyoshi, Development of an ensemble Kalman filter-based regional ocean data assimilation system, JpGU-AGU Joint Meeting 2020, online, 2020/7/13.

[9] James Taylor, Atsushi Okazaki, Takumi Honda, Shunji Kotsuki, Moeka Yamaji, Takuji Kubota, Riki Oki, Toshio Iguchi, Takemasa Miyoshi, Evaluating the impact of precipitation radar observations from a geostationary satellite on typhoon forecasts, JpGU-AGU Joint Meeting 2020, online, 2020/7/15.

[10] Kotsuki S., Terasaki K., Satoh M. and Miyoshi T.: Ensemble-Based Data Assimilation of GPM/DPR Reflectivity into the Nonhydrostatic Icosahedral Atmospheric Model NICAM. JpGU-AGU Joint Meeting 2020, online, 2020/7/16.

[11] James Taylor, Guo-Yuan Lien, Shinsuke Satoh, Takemasa Miyoshi, The Use of Dual Phased Array Radar Observations on Short-Range Convective Weather Forecasts, JpGU-AGU Joint Meeting 2020, online, 2020/7/16.

[12] Arata Amemiya, Shlok Mohta and Takemasa Miyoshi, Application of the Long-Short Term Memory neural networks to model bias correction: idealized experiments with the Lorenz-96 model. ECMWF-ESA Workshop on Machine Learning for Earth System Observation and Prediction, online, 2020/10/5.

[13] Maha Mdini, Shigenori Otsuka, Takemasa Miyoshi, “Accelerating Climate Model Computation by Neural Networks: A Comparative Study,” ECMWF-ESA Workshop on Machine Learning for Earth System Observation and Prediction, online, 2020/10/5.

[14] Shigenori Otsuka, Yasumitsu Maejima, Pierre Tandeo, and Takemasa Miyoshi, Toward an integrated NWP-DA-AI system for 30-second-update precipitation prediction. ECMWF-ESA Workshop on Machine Learning for Earth System Observation and Prediction, online, 2020/10/5.

[15] 前島康光, 三好建正 “EFSOを用いた稠密地上観測データ同化のインパクト評価,” 日本気象学会2020年度秋季大会, WF-14, オンライン, 2020/10/30.

[16] 大塚成徳, 前島康光, Pierre Tandeo, 三好建正, 深層学習と数値天気予報の融合による降水予測に向けて. 日本気象学会2020年度秋季大会, オンライン, 2020/10/30.

[17] 三好建正, ゲリラ豪雨予測を目指した「ビッグデータ同化」の研究. 第7回「京」を中核とするHPCIシステム利用研究課題 成果報告会, オンライン, 2020/10/30.

[18] James Taylor, Arata Amemiya, Takumi Honda, Yasumitsu Maejima, Takemasa Miyoshi, Optimizing the localization scale for a convective-scale ensemble radar data assimilation system, AMS 101st Annual Meeting, online, 2021/1/11.

[19] Jianyu Liang, Koji Terasaki and Takemasa Miyoshi, “A purely data-driven approach to satellite simulator in numerical weather prediction”, the 3rd R-CCS International Symposium, online, 2021/2/15.

[20] Shun Ohishi, Tsutomu Hihara, Hidenori Aiki, Joji Ishizaka, Yasumasa Miyazawa, Misako Kachi and Takemasa Miyoshi, “Development of an ensemble Kalman filter-based regional ocean data assimilation system,” the 3rd R-CCS International Symposium, online, 2021/2/15.

[21] Maha Mdini, Takemasa Miyoshi and Shigenori Otsuka, “Accelerating Climate Model Computation by Neural Networks: A Comparative Study”, the 3rd R-CCS International Symposium, online, 2021/2/15.

[22] Koji Terasaki, and T. Miyoshi, A 1024-Member Data Assimilation and Forecast Experiment with NICAM-LETKF Using Fugaku: A Heavy Rainfall Event in Kyushu in July 2020, the 3rd R-CCS International Symposium, online, 2021/2/15.

[23] Yasumitsu Maejima, Tomoo Ushio and Takemasa Miyoshi, “Toward assimilation of dense and frequent 3-D lightning location data for severe thunderstorm forecast”, the 3rd R-CCS International Symposium, online, 2021/2/15.

[24] James Taylor, Takumi Honda, Arata Amemiya and Takemasa Miyoshi, “Optimizing the localization scale for a convective-scale ensemble radar data assimilation system”, the 3rd R-CCS International Symposium, 2021/2/15.

[25] Qiwen Sun, Takemasa Miyoshi, Serge Richard, Ensemble Kalman filter experiments with an extended SIR model for COVID-19, the 3rd R-CCS International Symposium, online, 2021/2/15.

[26] Tianfeng Hou, Staf Roels, Hans Janssen The use of proper orthogonal decomposition for the Monte Carlo based uncertainty analysis, the 3rd R-CCS International Symposium, online, 2021/2/15.

[27] Shigenori Otsuka, Yasumitsu Maejima, Pierre Tandeo, and Takemasa Miyoshi, Toward an integrated NWP-DA-AI system for precipitation prediction, the 3rd R-CCS International Symposium, online, 2021/2/15.

[28] Arata Amemiya, Shlok Mohta, and Takemasa Miyoshi, Application of the Long-Short Term Memory neural networks to model bias correction: idealized experiments with the Lorenz-96 model, the 3rd R-CCS International Symposium, online, 2021/2/15.

[29] Kohei Takatama, John C. Wells and Takemasa Miyoshi, “Simulating rapid water level decrease of Lake Biwa due to Typhoon Jebi (2018)”, the 3rd R-CCS International Symposium, online, 2021/2/15.

[30] Takumi Honda, Yousuke Sato and Takemasa Miyoshi, “Potential Impacts of Lightning Flash Observations on Numerical Weather Prediction with Explicit Lightning Processes,” the 3rd R-CCS International Symposium, online, 2021/2/15.

[31] 寺崎康児、三好建正、全球水平解像度56km・1024メンバーのNICAM-LETKFを用いた令和2年7月豪雨実験、「富岳」成果創出加速プログラム防災・減災に資する新時代の大アンサンブル気象・大気環境予測　令和2年度成果発表会、オンライン, 2021/3/16.

[32] 前島康光, 牛尾知雄, 三好建正, "雷観測データBOLTの同化に向けた観測演算子の設計調査," 「富岳」成果創出加速プログラム防災・減災に資する新時代の大アンサンブル気象・大気環境予測　令和2年度成果発表会, オンライン, 2021/3/16.

[33] 雨宮新, 本田匠, 大塚成徳, Guo-Yuan Lien, James Taylor, 前島康光, 西澤誠也, 山浦剛, 末木健太, 富田浩文, 佐藤晋介, 石川裕, 三好建正, "2020年夏季のOakforest-PACS部分占有利用による30秒更新30分降水予報のリアルタイム実験," 「富岳」成果創出加速プログラム防災・減災に資する新時代の大アンサンブル気象・大気環境予測　令和2年度成果発表会, オンライン, 2021/3/16.

[34] James Taylor, Optimizing the localization scale for a convective-scale ensemble radar data assimilation system, 「富岳」成果創出加速プログラム防災・減災に資する新時代の大アンサンブル気象・大気環境予測　令和2年度成果発表会, オンライン, 2021/3/16.

15.5.7 Press release / News

[1] RIKEN and JAXA collaborate to offer real-time rainfall forecasts, `https://www.riken.jp/en/news_pubs/news/2020/20200820_2/index.html`, 2020/8/20.

[2] 30秒ごとに更新するゲリラ豪雨予報 —首都圏でのリアルタイム実証実験を開始—, `https://www.riken.jp/pr/news/2020/20200821_1/index.html`, 2020/8/21.

[3] スーパーコンピュータ「富岳」を利用した史上最大規模の気象計算を実現 —スパコン×シミュレーション×データ科学の協働が切り開く未来の気象予報—, `https://www.riken.jp/pr/news/2020/20201120_2/index.html`, 2020/11/20.

Chapter 16

Computational Disaster Mitigation and Reduction Research Team

16.1 Members

Satoru Oishi (Team Leader)

Muneo Hori (Senior Visiting Scientist)

Hideyuki O-tani (Research Scientist)

Yasuyuki Nagano (Senior Visiting Scientist)

Masaaki Yabe (Senior Visiting Scientist)

Tsuyoshi Ichimura (Visiting Scientist)

Lalith Maddegedara (Visiting Scientist)

Kohei Fujita (Visiting Scientist)

Jian Chen (Visiting Scientist)

Kazuki Yamanoi (Visiting Scientist)

Hiroki Motoyama (Visiting Scientist)

Tomohide Takeyama (Visiting Scientist)

16.2 Overview of Research Activities

Computational Disaster Mitigation and Reduction Research Team aims to develop advanced large-scale numerical simulations of natural disasters such as earthquakes, tsunami, floods, and inundation, for Kobe City and other urban areas in Hyogo Prefecture. Besides developing a sophisticated urban area model and new codes for numerical simulation, the team seeks to bridge science and the local government for disaster mitigation and reduction.

Computational Disaster Mitigation and Reduction Research Team is also conducting to integrate all kinds of geohazards, water hazards, and related hazards. Demand for natural disaster simulations increased related to a growing number of disasters in recent years. Therefore, we are developing appropriate sets of programs that meet the demand of calculations. Computational Disaster Mitigation and Reduction Research Team is dealing with the following four kinds of research topics.

Urban model development: Research for urban hazards requires urban models that represent cities' structure and shape in numerical form. However, it takes a very long time to create urban models consisting of buildings, foundations, and infrastructures like bridges, ports, and roads in an ordinary way. Therefore, it is indispensable to invent methods that automatically construct urban models from exiting data that is ill-structured. Computational Disaster Mitigation and Reduction Research Team developed Data Processing Platform (DPP) for such purpose. Using DPP, construction of a national-wide urban model and 3D model construction from engineering drawings are achieved.

Development of large-scale analysis methods for numerical simulation of urban areas under earthquakes [5]: High-fidelity urban earthquake simulation involves huge computational cost and thus efficient use of HPC systems are required. By extending the convergence of physics-based simulation and data-driven learning in the HPC resources, we designed an algorithm that benefits from the strengths of both physics based simulation and data-driven learning to accelerate solvers. The developed solver achieved 15.2-fold speedup over the conventional method PCGE with 96.4% size-up scalability up to 24,576 nodes (1,179,648 cores) of Fugaku on an urban earthquake problem.

Development of simulation-based assessment for debris-flow: We have conducted debris-flow and flood simulations for the Hiroshima prefecture, which was affected by heavy rainfall disaster in July, 2018. Employing the simulation results as training data of the machine learning method, we have developed a rapid damage detection method from the satellite imaginary. We also started to add a rainfall-runoff process to the debris-flow simulation.

Development of an HPC enhanced 1:1-scale simulator of national economy for simulating post-disaster economy: To seamlessly forecast the impact of major natural disasters on the national economy, we developed an HPC enhanced code capable of simulating the national economy in 1:1 scale. The computational performance of the code is improved to simulate several hundred million agents in a few minutes, using several hundred CPU cores. Analyzing the data from e-Stat, the Bank of Japan, and other government institutions, we identify the parameters necessary for simulating Japanese economy, and validated the numerical model by reproducing economy-wide and economic-sector-wide indices for the period from 2011 to 2019. Further, we simulated an earthquake disaster in Osaka bay area, estimated the economic losses due to the disaster, and forecasted the post-disaster economy under an assumed simple recovery plan.

16.3 Research Results and Achievements

16.3.1 Urban model development

In this year, we re-developed sets of programs for creating three dimensional urban model automatically. Those programs, namely Data Processing Platform (DPP) has been re-designed and implemented as Data Processing Platform Version 2 (DPP2). This program gives thematic elements on two dimensional computer aided design data (2D-CAD data) according to their levels of recognition, contexts of 2D-CAD data which come from arrangement of lines, figures and types of 2D-CAD data. After 2020 re-designing process, it has natural language interface to show the status of recognition. Fig. 16.1 shows the result of three dimensional model of bridge from 2D-CAD data. The programs utilize rule-base processes for recognizing elements on the drawings in order to adopting the contexts and levels of recognition. However, the programs also utilize other kinds of process for recognizing elements, then it is possible to develop each software for many kinds of drawings using division of programming the software.

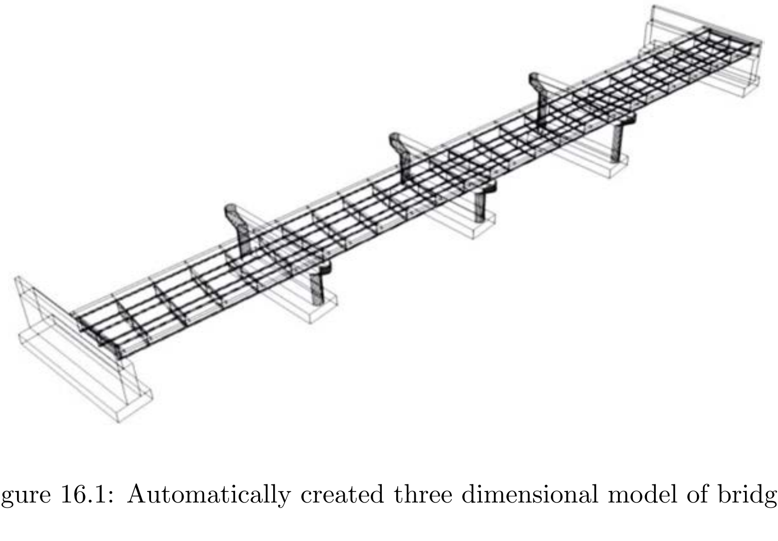


Figure 16.1: Automatically created three dimensional model of bridge.

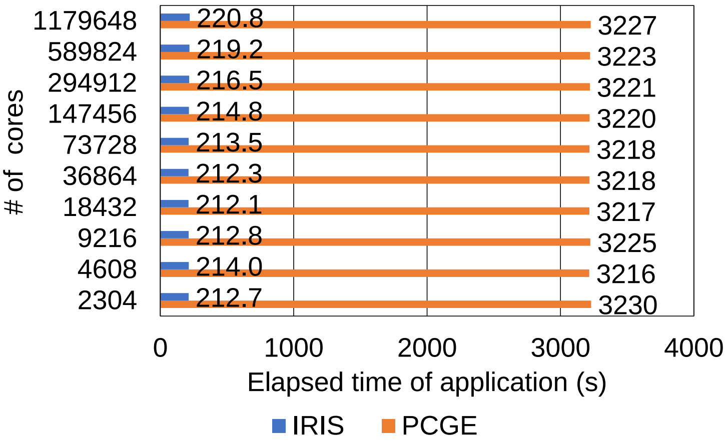


Figure 16.2: Weak scaling of the developed finite-element solver (IRIS) and standard solver (PCGE) on Fugaku.

16.3.2 Development of large-scale analysis methods for numerical simulation of urban areas under earthquakes [5]

In this research, we attempt to reduce the huge analysis cost involved in high-fidelity urban earthquake simulations resolving aboveground and underground structures and local ground structures. The urban earthquake problem is a nonlinear time evolution problem from a mathematical viewpoint, and from a computational science/computer science viewpoint, the random memory access dominated low-order unstructured finite element analysis used in solving this problem is challenging for attaining performance. In this research, a new approach is developed in merging high-performance computing (HPC)-enhanced physics based simulations with data-driven learning. Here we propose a fast and scalable implicit solver algorithm that uses data generated while conducting physics-based modeling for data-driven learning.

For the challenging urban earthquake problem, we showed that the developed method significantly reduces analysis cost compared to conventional solvers: The developed solver achieved 15.2-fold speedup over the conventional method PCGE with 96.4% size-up scalability up to 24,576 nodes (1,179,648 cores) of Arm SVE CPU-based Fugaku (Fig. 16.2). The performance of the developed solver was 12.7% of FP64 peak on 48 nodes of Fugaku and 12.8% of FP64 peak on 48 nodes of an Intel Xeon-based (Cascade Lake) supercomputer Oakbridge-CX, which is high for the low-order unstructured finite element method involving random access-dominated sparse matrix operations and global communication. Furthermore, the developed solver attained a 10.3-fold speedup over SC14 Gordon Bell Prize finalist solver GAMERA on a super-high resolution analysis of urban model shown in Fig. 16.3. We can see that complex response reflecting the underground structure as well as overground structure is obtained by this fully coupled simulation.

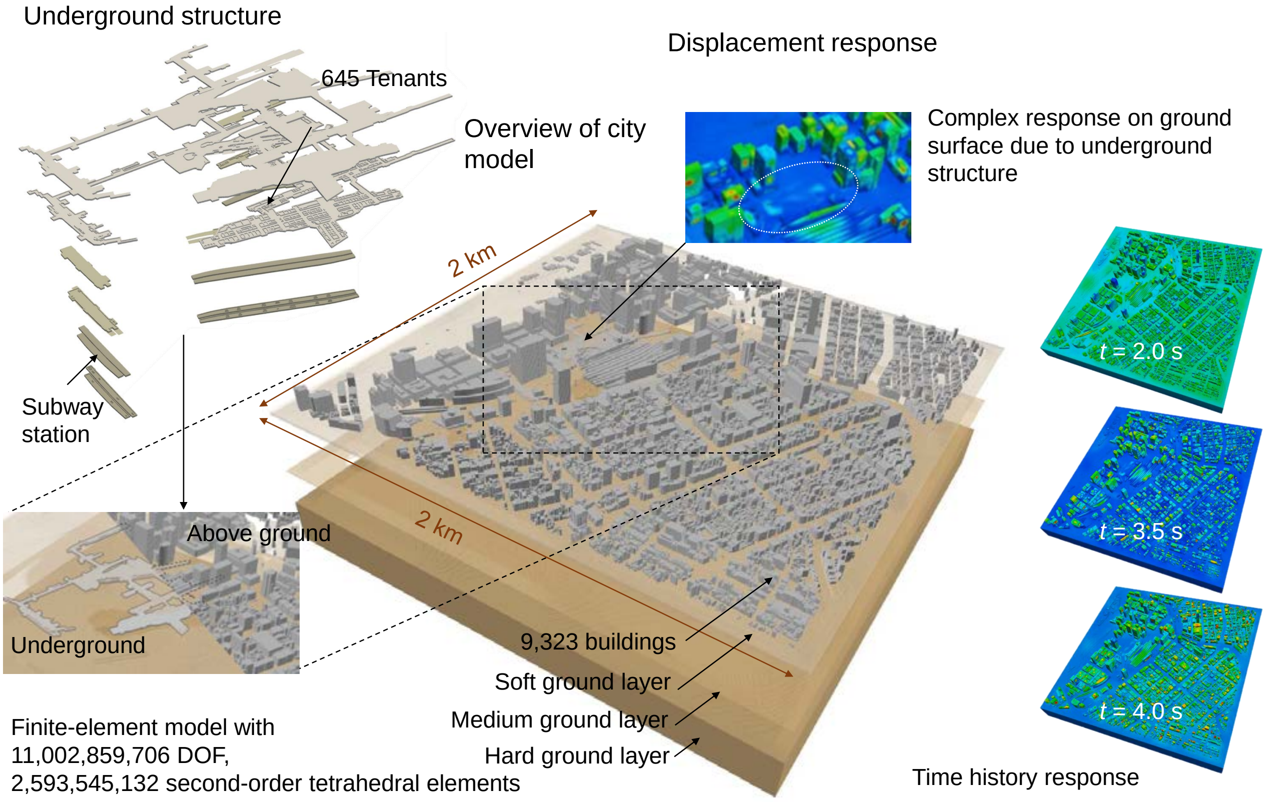
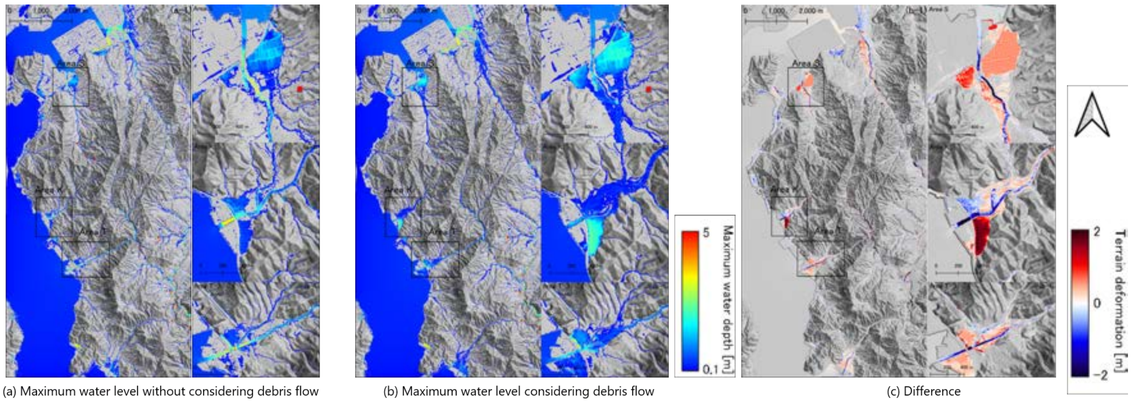


Figure 16.3: Problem setting and results of high-fidelity urban earthquake simulation based on unstructured implicit finite-element method. The 2 × 2 km domain was modeled with 0.5-m tetrahedral elements with three soil layers, 9323 buildings, and an underground complex with 645 tenants. The complex nonlinear seismic response was computed as a result of the three-dimensional ground structure and building configuration.



(a) Maximum water level without considering debris flow

(b) Maximum water level considering debris flow

(c) Difference

Figure 16.4: Rainfall runoff simulation results for 2018 flood in Hiroshima pref. (a) Maximum water level using the original DEM. (b) Maximum water level using the DEM after debris flow. (c) Difference of (b) from (a).

16.3.3 Debris flow and flood simulation

Debris flow is a phenomenon that develops from landslides and is a catastrophic hazard that can cause human damages. An important characteristic of the debris flow during heavy rainfall is increasing the following water inundation by changing the topography in and around the river by sediment deposition. To assess such risks, we have conducted the continuous simulation of rainfall-runoff and inundation by using the deformed DEM by debris flow simulation. The target is Saka town, Hiroshima prefecture, which was affected by a heavy rainfall disaster that happened in Jul. 2018. By considering the topographic change due to debris flow, the increase of maximum water level appeared in rainfall-runoff and inundation simulation, see Fig, 16.4. Currently, the simulation for rainfall-runoff and inundation is separated from the debris-flow simulation. Therefore, the current method does not treat the timing of debris flow nor sediment transport by the flood. To solve this problem, we have started to integrate the rainfall-runoff process into the debris-flow simulation.

Since 2020, we have been developing a rapid damage detection technique using the results of debris-flow simulations as training data for machine learning, in collaboration with the RIKEN AIP Center. The paper proposing the methodology (see Fig. 16.5) was accepted to IEEE Transactions on Geoscience and Remote Sensing.

16.3.4 Development of an 1:1-scale economic simulator for disaster applications

Wide-spread damages to infrastructures caused by natural disasters can inflict long term economic losses. Complicated dependencies of businesses on infrastructures and lifelines demands fine grained simulations of economy considering availability of resources (e.g. repair crews, transportation, electricity, gas, water, access to market, etc.) to each individual firm. Although Agent Based Economic Models (ABEMs) are capable of including all the complexities involved, the existing computer implementations are not capable of simulating more than several 10 million agents, and their matrix based designs makes it difficult to include dependencies on lifelines, etc. In order to address these, we developed a flexible and HPC enhanced ABEM capable of simulating several hundred millions of agents in a few ten minutes utilizing only a small amount of computer resources. As an example, a single period of a 330 million agents (equivalent to Euro-zone) can be simulated within 150s using 64 CPU cores. The need of only a small computational resources enables 1:1 scale Monte-Carlo simulations of large economies to take the involved uncertainities into account.

Analyzing the data from various government resources like, the Bank of Japan, e-Stat, etc., we identified the necessary parameters to model Japanese economy, and validated the model by reproducing nation-wide and economic-sector-wide observations for the period from 2015 to 2020. As seen in Fig. 16.6 and Fig. 16.7, the simulation results are in good agreement with the observations, which validates the developed model. Further, we simulated the Japanese economy in the aftermath of a potential earthquake disaster in the Osaka bay area caused by the Nankai through mega earthquake scenario proposed by the Cabinet office of Japan. Ground shaking and earthquake induced damages to 1.8 million buildings of the Osaka bay area are estimated using IES, corresponding repair cost and recovery time of each building are estimated using and PACT (Performance Assessment Calculation Tool developed by FEMA, USA) , and used in the ABEM to analyze their impact on

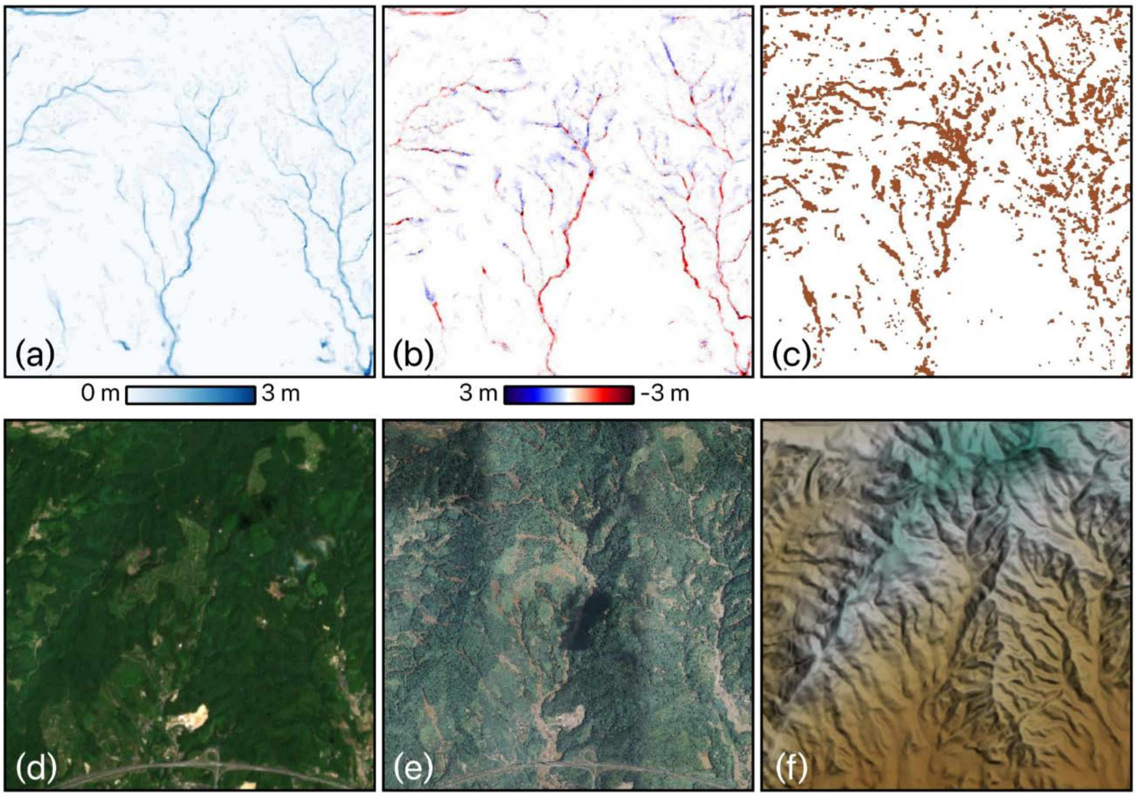


Figure 16.5: The overview of the proposed framework; (a)maximum water level and (b) topographic change estimated from the (c) binary change map derived from (d) predisaster and (e) postdisaster images together with (f) DEM [1].

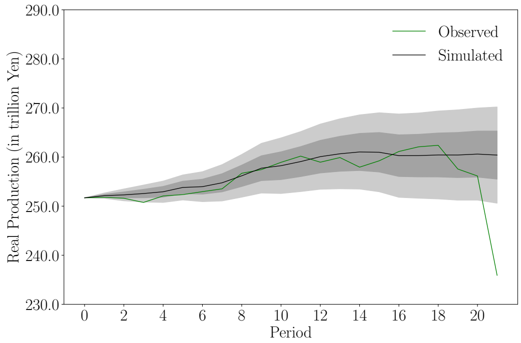


Figure 16.6: National economy

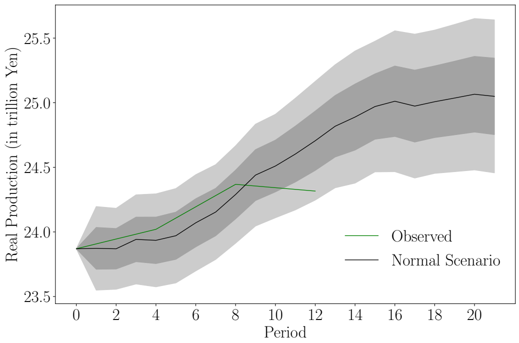


Figure 16.7: Commerce sector

Figure 16.8: Simulated quarterly real-production of national economy and commerce-sector of Japan from 2015 to 2020

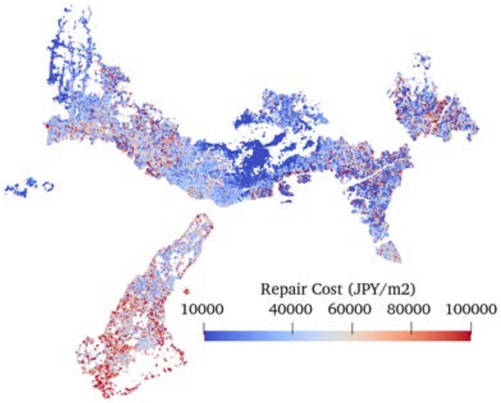


Figure 16.9: Repair cost per unit floor area

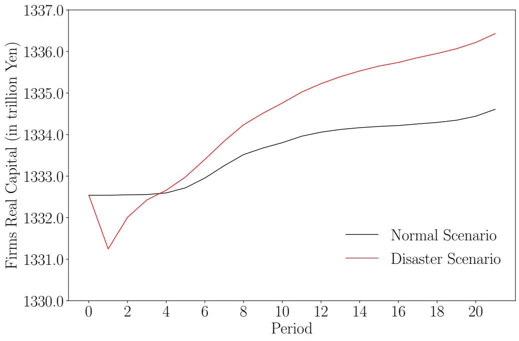


Figure 16.10: Firms' real capital

Figure 16.11: Estimated repair cost for a potential earthquake in Osaka bay area, and the forecasted firms' real capital under a simple recovery plan.

the economy. Figure 16.9 and Fig. 16.10 shows the repair cost and the forecast of firms' real capital for 20 quarters, under the assumed simple recovery plan.

16.4 Schedule and Future Plan

1. Developing a national-wide real-time disaster simulation: We will enhance the automation of DPP2 to collect real-time seismic information and to perform an automated disaster simulation in a certain target area. This will reveal the extent to which speeding up is required for real-time characteristics.

2. Construction of templates for high fidelity models of highway network: In the template-based methodology, we need to ready the templates in beforehand, and the quality and quantity of templates will be critical to the output model.

3. Developing an integrated simulation scheme for rainfall-runoff and debris-flow.

4. Sensitivity analysis of the multiple parameters related to soils, sediments, and landslide inputs by using the integrated model to establish an optimized prediction scheme.

5. Several scenario analysis of mulit-hazard disaster in Kinki region including economic impact of the disasters.

16.5 Publications

16.5.1 Articles/Journal

[1] Yokoya, N., Yamanoi, K., He, W., Baier, G., Adriano, B., Miura, H., and Oishi, S., “Breaking limits of remote sensing by deep learning from simulated data for flood and debris flow mapping,” IEEE Transactions on Geoscience and Remote Sensing, 2020. (doi: 10.1109/TGRS.2020.3035469.).
[2] Gill, A., Lalith, M., Poledna, S., Hori, M., Fujita, K., and Ichimura, T., “High-Performance Computing Implementations of Agent-Based Economic Models for Realizing 1:1 Scale Simulations of Large Economies,” in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 8, pp. 2101-2114, 1 Aug. 2021
[3] Maddegedara, L., Gill, A., Poledna, S., Hori, M., Inoue H., Noda T., Toda K., and Ichimura T., “Distributed Memory Parallel Implementation of Agent Based Economic Models”, Lecture Notes in Computer Science, 2019

16.5.2 Conference Papers

[4] Gill, A., Maddegedara L., Poledna, S., Hori, M., Ichimura, T., and Fujita, K., “Development of High-performance Computing Extension for an Agent Based Economic Model for Fine-grained Post-disaster Economy Simulations.”, Conference Proceedings, The 17th World Conference on Earthquake Engineering, 17WCEE, 2020
[5] Ichimura, T., Fujita, K., Koyama, K., Kusakabe, R., Minami, K., Inoue, H., Nishizawa, S., Tsuji, M., Nishiki, T., Hori, M., Maddegedara, L., and Ueda, N., “Fast Scalable Implicit Solver with Convergence of Physics-Based Simulation and Data-Driven Learning: Toward High-Fidelity Simulation with Digital Twin City”, Research Poster, SC20 The International Conference for High Performance Computing, Networking, Storage, and Analysis, 2020.

16.5.3 Invited Talks

16.5.4 Oral Talks

[6] Yamanoi, K., and Oishi, S., “Multi-case Simulation on Water- and Sediment-Related Disasters for Estimating the Uncertainty of Damage Prediction Due to Input Conditions”, COMPSAFE 2020, (Kobe, December 10,2020)
[7] Gill, A., Maddegedara L., Poledna, S., Hori, M., Ichimura, T., and Fujita, K., “A Scalable High-Performance Computing Implementation of Macroeconomic Agent- Based Models”, The Workshop on Economic Science with Heterogeneous Interacting Agents, 29th June, 2021, on-lines
[8] Gill, A., Maddegedara L., Poledna, S., Hori, M., Ogawa, Y., Akiyama, Y., Sekimoto, Y., “Unification of Integrated Earthquake Simulator and Agent Based Economic Model to Estimate Impacts of an Earthquake Disaster”, 11th March, 2020
[9] Maddegedara L., Gill, A., Poledna, S., Hori, M., Inoue, H., Noda, T., Toda, K., and Ichimura, T., “Distributed Memory Parallel Implementation of Agent Based Economic Models, International Conference on Computational Sciences”, Faro, Portugal, 12-14 June, 2019

16.5.5 Software

16.5.6 Patents

Chapter 17

Computational Structural Biology Research Team

17.1 Members

Florence Tama (Team Leader)

Osamu Miyashita (Senior Scientist)

Bhaskar Dasgupta (Research Scientist)

Miki Nakano (Post-doctoral Researcher)

Sandhya Tiwari (Post-doctoral Researcher – April - February)

Sriram Raghavan (Post-doctoral Researcher – September - March)

Yumeno Kusahara (Assistant)

17.2 Overview of Research Activities

Biological molecular complexes of proteins and RNAs are of great interest in the area of molecular biology, as they are responsible for core biological functions such as cell replication, gene transcription, protein synthesis, regulation of cellular transport, and numerous others. For their functional processes, these molecular systems often undergo large conformational transitions. Therefore, the characterization of dynamical structures of these macromolecular complexes is crucial to understand their functional mechanisms and plays an important role in the development of new drugs to treat human disease.

Experimentally, X-ray crystallography has been the primary tool to study protein conformations, providing high-resolution structures. Cryo-electron microscopy (EM) is becoming an important technique due to the development of experimental apparatus as well as data analysis software. Although the structural resolution of the cryo-EM data tends to be lower than atomic level, it has revealed critical information on the structure and dynamics of large biological molecules. More recently, efforts such as in RIKEN/SPring-8 have focused on developing intense X-ray free-electron laser (XFEL) light sources, which offer a new possibility to image single biological macromolecules. Using extremely strong X-ray pulse, XFEL enables single-shot observation of the samples to perform time-resolved data collections. In addition, since crystallization is not necessary for such a structure analysis, it would be possible to investigate the structure of biomolecules under various physiological conditions or to observe elementary steps of a biochemical function. However, with the current experimental technology, it cannot achieve atomic-level resolution as obtained by X-ray crystallography. In addition, high-speed atomic force microscopy (HS-AFM) is another technique that is seeing a growing number of biological applications. This technique enables observation of biological molecular complexes in motions in a near-native environment, providing unique information regarding functional dynamics. However, due to limitations with the manufacturing process of the observation probes, the structural resolution of the images is relatively low. In this way, new experimental techniques are being developed to reveal the structural information of important biomolecules. Yet, each experimental technique has its own strengths and weaknesses, and thus integration of computational simulations with experimental data is beneficial to obtain new and detailed information on the structure and dynamics of biological molecules.

Computationally, we have been working on the development of algorithms and software to construct structural models from low-resolution experimental data such as from cryo-EM using molecular dynamics simulations. Experimentally known atomic structures can be used as complementary structural information. In addition, even when the structures of the molecules are unknown, atomic models can be predicted using homology modeling and ab initio predictions, which can also be used for the modeling of new biomolecular complexes. Such hybrid approaches need to be extended further to integrate various available experimental data to determine the structures of important biomolecular complexes.

The ultimate line of our interdisciplinary research is the development and applications of computational tools, through high-performance computing, to integrate experimental data as obtained from various techniques such as X-ray crystallography, cryo-EM, high-speed atomic force microscopy, and XFEL with molecular modeling and simulations to acquire knowledge on the structure of physiologically important protein complexes that are unattainable with existing individual experimental techniques.

17.3 Research Results and Achievements

17.3.1 Development of computational tools for analyzing X-ray free electron laser data

The recent development of intense X-ray free-electron laser (XFEL) light sources offers a new possibility to image single biological macromolecules. Since crystallization is not necessary for such structure analysis, it would be possible to investigate the structure of macromolecular complexes under various physiological conditions. Furthermore, very short (fs) but strong X-ray pulses enable the observation of elementary steps of a biochemical function. SPring-8/SACLA in Japan is one of the pioneering XFEL facilities in the world.

As XFEL experiments are very recent and still undergoing further development for the routine study of biological molecules, computational algorithms and tools to understand and analyze experimental data need to be developed simultaneously. With the current experimental technology, an atomic level resolution such as obtained by X-ray crystallography cannot yet be achieved. Therefore, computational approaches play important roles in data analysis and structural modeling. One focus of our research is the development of such computational tools. We are tackling these issues from multiple angles described below.

17.3.1.1 3D Reconstruction from XFEL diffraction patterns – Applications to experimental data

We have been conducting joint research with Professor Yoshinori Nishino's research group at Hokkaido University to demonstrate 3D structure reconstruction from single-particle XFEL experimental data using the data analysis algorithms we have been developing. Pulsed coherent X-ray solution scattering (PCXSS), which is being developed by the Nishino group, is a unique method that can observe samples in aqueous solution using XFEL, and with the continuous improvement of the device, it is becoming possible to observe nano-scale samples.

We have been analyzing the XFEL diffraction data of gold nanoparticles (truncated triangular bipyramid) with a size of about 50 nanometers, aiming to reconstruct the three-dimensional structure of the sample. Previously we had developed algorithms for selecting data that can be used for three-dimensional structure reconstruction from the raw experimental XFEL diffraction patterns which includes not only clean hit patterns but also no-hit or multiple-sample hits patterns. In addition, we have developed a new approach to estimate background signals in the measured diffraction patterns using the data that are categorized as no-hit patterns during the filtering procedure. Then, the selected hit diffraction patterns are assembled into 3D space using our algorithms, with an additional improvement to take into account the fluctuation of diffraction intensity strength resulting from beam intensity fluctuations. As the last step, the model of nano-particle can be obtained from the obtained 3D diffraction intensity volume through the phase-recovery procedure. We have discovered that by subtracting background signals, which are estimated by the filtering method, from 3D diffraction intensity volume, the 3D real space model obtained from phase recovery improves significantly showing a clear truncated triangular bipyramid shape. In addition, we have improved the phase recovery procedure by incorporating the information of particle size, which can be estimated from a few strong diffraction patterns showing clear hexagonal symmetry, as the initial support. Combining these improvements, we have successfully demonstrated that our algorithm can reconstruct precise 3D structures with the correct size (Figure 17.1).

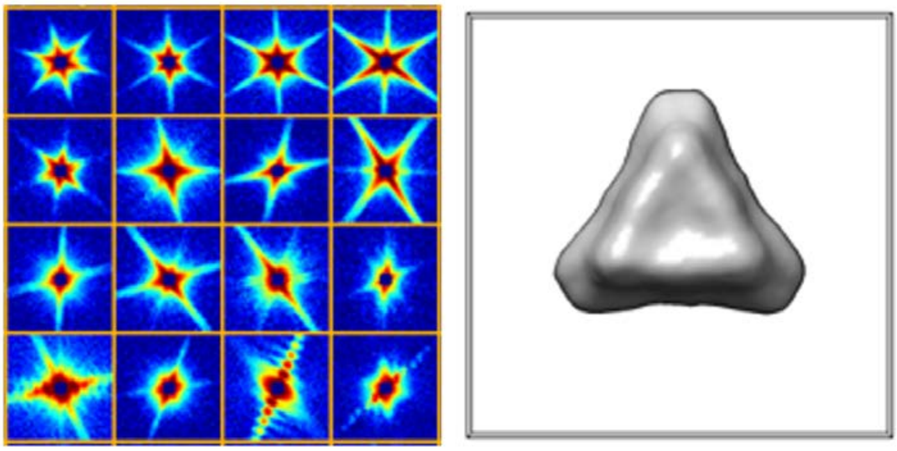


Figure 17.1: (a) Examples of experimental data. (b) A 3D structure of gold nanoparticle using newly developed data filtering and 3D reconstruction approach.

17.3.1.2 Quick "idea generator" from 2D XFEL data of biological systems using structural databases

Data analysis for XFEL data remains challenging. XFEL diffraction pattern is an unintuitive representation of the projection image of the sample in Fourier space. For biological systems, the current standard approach to reconstruct a real-space image of the sample, phase recovery, often fails due to the low diffraction power of biological samples. On the other hand, a wealth of knowledge of structural information from various experiments, assembled in multiple databases, is available. Therefore, we are developing a new hybrid approach to interpret diffraction patterns that utilizes image analysis within a database search. More specifically, for a given set of XFEL diffraction patterns, we identify plausible structures via searching a database of possible 3D low-resolution shapes. Such tools could be used as an idea generator which serves as a starting point for further analyses.

In the developed approach, we first prepare the database of molecular shapes and simulated XFEL single-particle diffraction patterns expected from these structures (Figure 17.2A). We built a library of diffraction patterns simulated from 1,628 structure models obtained from cryo-electron microscopy (EM) in which the models with similar shapes are already removed to avoid redundancy. Then, we established an algorithm to compare the input XFEL diffraction patterns against the patterns in the library. Since the XFEL diffraction pattern is in Fourier space, low-frequency (center) regions of diffraction patterns reflect the overall shapes of the molecule, while details of the molecular structure are represented in high-frequency (outer) regions. In addition, diffraction patterns from biomolecules contain a limited amount of signals due to the weak diffraction intensity, particularly being weak at high and strong at low wavenumber pixels, requiring careful selection of the matching region when aligning the diffraction images in Fourier space. We observed that choosing the best strategy to define matching regions on the diffraction patterns is critical for identifying correctly matching diffraction patterns. Thus, we have developed an algorithm, in which the approximate size of the sample in the input image is estimated via fitting against a theoretical model, spherical form factors, and used to estimate the appropriate regions within each pattern that should be used for comparison (Figure 17.2B). The significance of the matches is evaluated by assessing the distributions of the matching scores converted to z-scores; if the top-ranking matching patterns have high z-scores, the proposed matching 3D models are more reliable for the given set of input diffraction patterns. While increasing the number of input diffraction patterns improved the matches in some cases, we found that the resulting matches are more dependent on the uniqueness or complexity of the shape as captured in the individual input diffraction patterns, and the availability of a similar 3D biological shape in the search library. Nevertheless, the features of the shape in the input images are still captured within the top matching hits (Figure 17.2C). The protocol could be useful for finding candidate models for a limited number of low-resolution XFEL diffraction patterns, even when insufficient for 3D reconstruction, performing a quick exploration of new data upon collection, and the analysis of the conformational heterogeneity of the particle of interest as captured within the diffraction patterns (S. P. Tiwari, F. Tama, O. Miyashita, *J Chem Inf Model*, 2021).

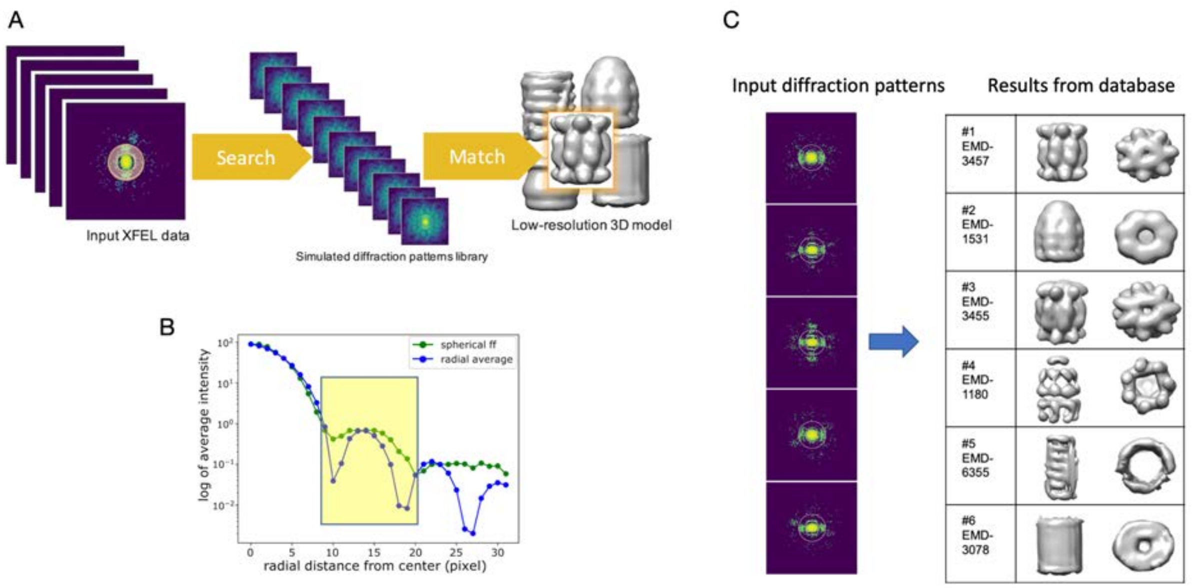


Figure 17.2: (A) The conceptual scheme of the proposed XFEL diffraction pattern analysis using databases. From known 3D models, expected XFEL diffraction patterns are simulated and assembled to form a library. For a given set of XFEL diffraction patterns, similar patterns in the library are identified using the proposed pattern image comparison algorithm. Finally, the corresponding matched 3D structures are proposed as the most likely 3D structures observed in the input XFEL diffraction patterns. (B) An example result of matching region selection. A theoretical model of the spherical form factor is fitted against the experimental data, radial average, to identify the best region for matching (colored in yellow). (C) An example result of the proposed approach. Using five input XFEL diffraction patterns, the molecular structures that are most consistent with the inputs are automatically selected and ranked.

17.3.1.3 Computational analysis with XFEL time-resolved serial femto-second crystallography data

X-ray free electron laser generates photon pulses which are extremely strong and very short in time (¡ 10 femto-second), and thus enables obtaining a time-resolved snapshot of molecules in motion. Another important technique is serial-femtosecond crystallography (SFX), in which a large number of X-ray diffraction patterns are obtained from multiple micro-crystals and assembled to obtain atomic-resolution structures of biomolecules. Combining these techniques, time-resolved (TR) SFX is a revolutionary technique that can provide high-resolution information on biomolecules in motions. However, several limitations remain. First, practically only a limited amount of experimental data can be collected, i.e., the structures only at a limited number of observation time-points can be obtained. In addition, due to the method using protein crystals, the obtained X-ray diffraction patterns reflect the average structure of the biomolecule in motion. Thus, molecular dynamics simulation can play an important role to provide complementary information with atomic details.

Structure modeling from TR-SFX data is complex, involving a large amount of data with strong noise, and many procedures rely on manual processing of the data by expert researchers. Thus, we have been developing algorithms to aid such structure modeling procedures by providing automated computational tools that can construct the time-resolved structural models using molecular dynamics simulations. In this approach, structural modeling from TR-SFX data is commonly performed by visually inspecting the Fourier difference electron density map, which represents the change in the electron density between two observation time points. We have adapted flexible fitting algorithms for cryo-EM electron density maps to use the TR-SFX difference map as the target information. Using this approach, starting a conformation at one-time point, the conformational model corresponding to another time point can be automatically obtained (Figure 17.3). We have also demonstrated this approach for XFEL experimental data on bacteriorhodopsin conformational changes obtained at SACLA.

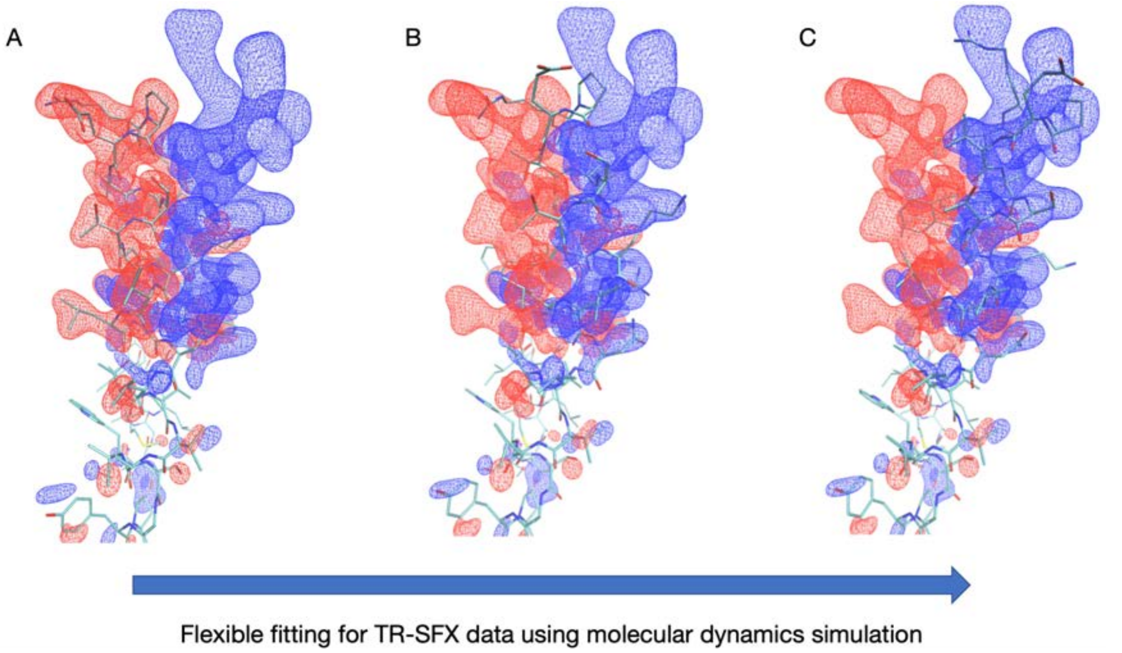

A
B
C
Flexible fitting for TR-SFX data using molecular dynamics simulation

Figure 17.3: Molecular dynamics flexible fitting using difference electron density map. (A) The initial protein structure (stick model) and the synthetic difference electron density map; negative densities are in red and positive densities are in blue. (B) An intermediate structure during the flexible fitting. (C) A resulting protein conformation after the flexible fitting. The structure is in agreement with the positive electron densities, while there are no structural changes where the electron density difference is small.

17.3.2 Structure modeling from atomic force microscopy images

Atomic Force Microscopy (AFM) is an experimental technique that enables the observation of biomolecules in near-native conditions. AFM uses a mechanical device (cantilever) to trace the molecular surface and provides images of biomolecules at nanometer resolution. In particular, high-speed AFM experiments produce a series of images following live dynamics of biomolecules, which may provide a wealth of information regarding protein dynamics and functions. However, the information in the data is very limited; due to the size of AFM probes, the resolution is very low and only 2-dimensional information on the molecular surface can be obtained. As information on three-dimensional (3D) structures and dynamics is critical for studies on biomolecular functions, computational tools to analyze HS-AFM data could further enhance its utility for application to biomolecules.

We have been developing algorithms to recover 3D information from 2D AFM images by computational modeling. The AFM image includes only a low-resolution representation of a molecule; therefore, we represent the structures by a coarse-grained model (Gaussian mixture model). Using Monte-Carlo sampling, candidate models that are in better agreement with the experimental AFM images can be generated and used for further interpretation of the experimental data.

We applied our algorithm to the experimental data of a protein complex, ClpB, obtained by the Uchihashi group, Nagoya University. ClpB belongs to the cellular disaggregase machinery involved in rescuing misfolded or aggregated proteins during heat or other cellular shocks (Figure 17.4A). HS-AFM experiment on ClpB revealed four predominant conformational classes that correlate with its functional dynamics. Since AFM images provide only partial structural information regarding the molecular surface, computational modeling of three-dimensional (3D) structures of these conformations should help interpret dynamical events related to ClpB functions. We performed reconstructions of 3D models of ClpB from HS-AFM images in different conformational classes and obtained models that reflect conformational variety embedded within the AFM images (Figure 17.4B, C). From these reconstructed 3D models, we described, in terms of relative domain arrangement, the different types of ClpB oligomeric conformations observed by HS-AFM experiments.

This study demonstrates that such details of information, necessary for annotating the different conformational states involved in the ClpB function, can be obtained by combining HS-AFM images, even with limited resolution, and computational modeling (B. Dasgupta, O. Miyashita, T. Uchihashi, F. Tama, *Front Mol Biosci*, 2021).

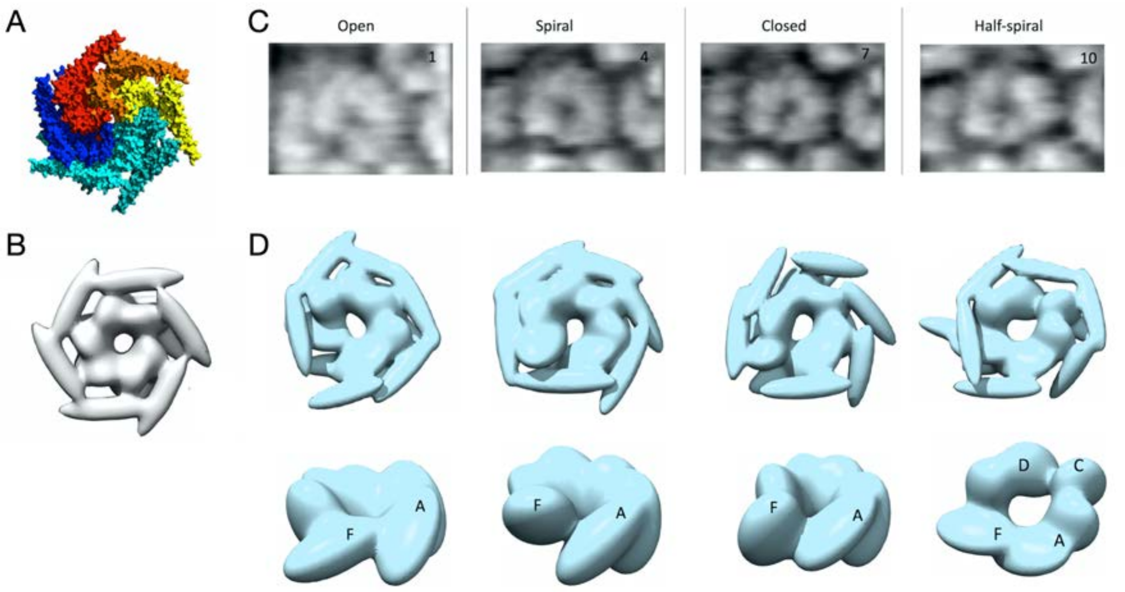


Figure 17.4: Overview of the modeling results with ClpB AFM experimental data. (A) A conformation of ClpB at atomic level details. (B) The coarse-grained representation of ClpB using a Gaussian Mixture Model. (C) Examples of HS-AFM experimental data that are representing 4 different conformational states. (D) Examples of 3D structural models corresponding to the experimental images obtained by Monte-Carlo sampling of the Gaussian kernels are optimized so that the agreement with the target AFM experimental image is maximized.

17.4 Schedule and Future Plan

The importance of cryo-EM experiments is continuously increasing and it is quickly becoming an essential tool for studying biomolecular complexes. New XFEL facilities are getting into operation every year in the world, providing opportunities for new experiments. The amount of data from these experiments will continue to grow in numbers and analysis of such big datasets will increase the necessity of high-performance computing. Time-resolved experiments using high-speed AFM and XFEL serial femtosecond crystallography also provide additional information on molecular movies. Our goal is to utilize high-performance computation, such as Fugaku, to break the limitation of current data processing and further improve the hybrid computational modeling approaches to obtain a new level of structural information of biological complexes from various experimental data. In particular, we will focus on the development of algorithms and software to analyze large datasets to obtain not only structural models but also dynamical information which is essential for the understanding of the mechanisms of biomolecular functions. For example, current cryo-EM experiments rather focus on obtaining a high-resolution 3D structural model; however, the original cryo-EM images include the information on conformational dynamics and heterogeneity of the system. Therefore, we aim to develop tools to extract such new information from experimental data using computer simulations. With these techniques, through collaborations with experimental groups, we aim to contribute to the structural biology community.

17.5 Publications

17.5.1 Articles/Journal

[1] Dasgupta, B., Miyashita, O., Uchihashi, T., Tama, F. Reconstruction of Three-Dimensional Conformations of Bacterial ClpB from High-Speed Atomic-Force-Microscopy Images. Front Mol Biosci 8, 704274 (2021).

[2] Tiwari, S. P., Tama, F., Miyashita, O. Protocol for Retrieving Three-Dimensional Biological Shapes for a Few XFEL Single-Particle Diffraction Patterns. J Chem Inf Model 61, 4108-4119 (2021).

[3] Kolarski, D., Miller, S., Oshima, T., Nagai, Y., Aoki, Y., Kobauri, P., Srivastava, A., Sugiyama, A., Amaike, K., Sato, A., Tama, F., Szymanski, W., Feringa, B. L., Itami, K., Hirota, T. Photopharmacological Manipulation

of Mammalian CRY1 for Regulation of the Circadian Clock. J Am Chem Soc 143, 2078-2087 (2021).
[4] Tiwari, S. P., Chhabra, S., Tama, F., Miyashita, O. Computational Protocol for Assessing the Optimal Pixel Size to Improve the Accuracy of Single-particle Cryo-electron Microscopy Maps. J Chem Inf Model 60, 2570-2580 (2020).
[5] Miller, S., Srivastava, A., Nagai, Y., Aikawa, Y., Tama, F., Hirota, T. Structural differences in the FAD-binding pockets and lid loops of mammalian CRY1 and CRY2 for isoform-selective regulation. Proc Natl Acad Sci U S A 118, e2026191118 (2021).
[6] Srivastava, A., Tiwari, S. P., Miyashita, O., Tama, F. Integrative/Hybrid Modeling Approaches for Studying Biomolecules. J Mol Biol 432, 2846-2860 (2020).

17.5.2 Book Chapter

[7] Bhaskar, D., Gert-Jan, B., Narutoshi, K., Dynamical methods to study interaction in proteins facilitating molecular understanding of cancer, Handbook of Oxidative Stress and Cancer: Mechanistic Aspects, Edited By: Chakrabarti S, in Springer, in production

17.5.3 Posters

[8] Bhaskar Dasgupta, Miyashita Osamu, Tama Florence, Modelling multiple types of three-dimensional hexameric conformations of bacterial heat-shock protein ClpB from low-resolution AFM image, The 3rd R-CCS International Symposium, 2021/2/15, online
[9] Nakano Miki, Tama Florence, Miyashita Osamu Requirement for 3D-reconstruction of biomolecule structure from single-particle analysis using X-ray free-electron laser diffraction images The 3rd R-CCS International Symposium, 2021/2/15, online
[10] Sandhya P. Tiwari, Tama Florence, Miyashita Osamu Retrieving potential three-dimensional biological shape matches from a small number of two-dimensional single-particle XFEL diffraction patterns 2021/2/15, online
[11] Nakano Miki, Tama Florence, Miyashita Osamu, Simulation study of three-dimensional reconstruction of ribosome from Xray free-electron laser diffraction patterns, 20th Annual Meeting of the Protein Society Meeting of Japan, 2020/7/28, online
[12] Sandhya P. Tiwari Tama Florence, Miyashita Osamu Retrieving potential three-dimensional biological shape matches from a small number of two-dimensional single-particle XFEL diffraction patterns, 20th Annual Meeting of the Protein Society Meeting of Japan 2020/7/28, online

17.5.4 Invited Talks

[13] Osamu Miyashita, “Hybrid approach for Structural Biology: Integration of Simulations and Experimental Data” University of Hyogo the 1st Pico-biology workshop, 2020/11/26, Online

17.5.5 Oral Talks

[14] Sandhya Tiwari, Florence Tama, Osamu Miyashita, “Protocol for Assessing the Optimal Pixel Size to Improve the Accuracy of Single-Particle Cryo-Electron Microscopy Maps” Chem-Bio Informatics Society(CBI) Annual Meeting 2020, 2020/10/27-30, Online

Chapter 18

High Performance Artificial Intelligence Systems Research Team

18.1 Members

Satoshi Matsuoka (Team Leader)

Jun Igarashi (Senior Scientist)

Balazs Gerofi (Senior Scientist)

Aleksandr Drozd (Research Scientist)

Shweta Salaria (Postdoctoral Researcher)

Emil Vatai (Postdoctoral Researcher)

Toshio Endo (Senior Visiting Scientist)

Mohamed Wahib (Visiting Scientist)

Miquel Pericas (Visiting Scientist)

Anna Rogers (Visiting Scientist)

Akihiro Nomura (Visiting Scientist)

Tokyo Tech Matsuoka lab students

18.2 Overview of Research Activities

The High Performance Artificial Intelligence Systems Research Team focuses on convergence of HPC and AI, namely high performance systems, software, and algorithms research and development for artificial intelligence / machine learning.

In collaboration with other research institutes in HPC and AI-related research in Japan as well as globally it seeks to develop next-generation AI technology that will utilize state-of-art HPC facilities, including Fugaku.

For the time being until Mar 31, 2021, team's research work is being primarily sponsored by the JST-CREST "DEEP" AI Project "Fast and cost-effective deep learning algorithm platform for video processing in social infrastructure".

18.3 Research Results and Achievements

18.3.1 Brain simulation

A human-scale whole-brain simulation is an ultimate tool to study the interaction of neurons in the human brain. However, it has been difficult to realize it due to the lack of computational resources and simulators to use the resource efficiently. We developed a new data structure for a spiking neural network simulator, MONET, to utilize computational features of Fugaku, the wide SIMD, and many-core architecture. The latest version of the MONET with the new data structure and previously proposed parallelization method showed excellent performance in the weak scaling test (See Figure 18.1). The MONET performed a simulation of a human-scale brain model consisting of the cerebral cortex, cerebellum, and thalamus for one second of biological time with less than 20 seconds. The MONET realized 41-162 times faster simulation on Fugaku than K in terms of cerebral cortex and cerebellum simulations. These results demonstrate that human-scale whole brain simulations can realize on exascale a supercomputer with our proposed method.

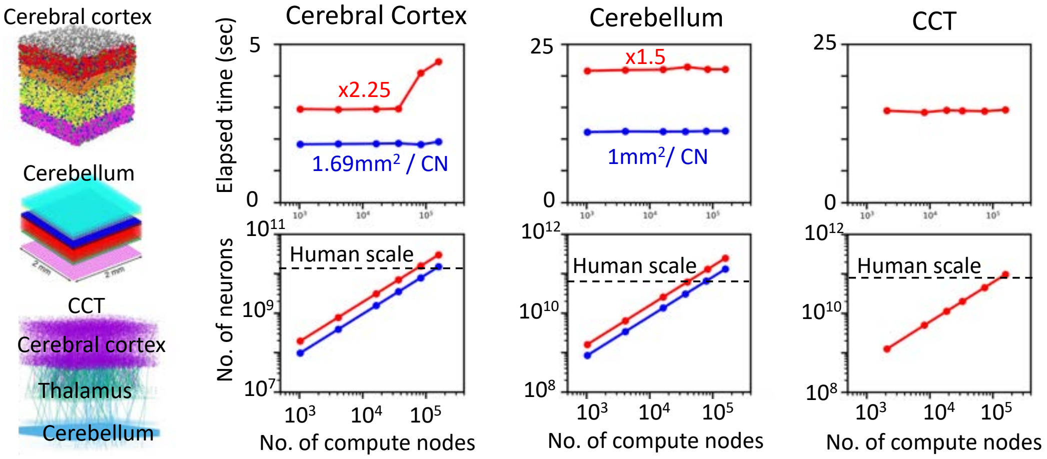


Figure 18.1: Weak scaling performance tests of spiking neural network simulations of cerebral cortex, cerebellum, and CCT using MONET on Fugaku

18.3.2 A Computational-Graph Partitioning Method for Training Memory-Constrained DNNs

Many state-of-the-art Deep Neural Networks (DNNs) have substantial memory requirements. Limited device memory becomes a bottleneck when training those models. In this work We propose *ParDNN*, an automatic, generic, and non-intrusive partitioning strategy for DNNs that are represented as computational graphs (See Figure 18.4). *ParDNN* decides a placement of DNN's underlying computational graph operations across multiple devices so that the devices' memory constraints are met and the training time is minimized. *ParDNN* is completely independent of the deep learning aspects of a DNN. It requires no modification neither at the model nor at the systems level implementation of its operation kernels. *ParDNN* partitions DNNs having billions of parameters and hundreds of thousands of operations in seconds to few minutes. Our experiments with TensorFlow on 16 GPUs demonstrate efficient training of 5 very large models while achieving superlinear scaling for both the batch size and training throughput. In Figure 18.3 (a), *ParDNN* shows a substantial improvement on 2 GPUs and superlinear speedups up to 4 GPUs for all the models. The sharp performance increase happens because, in addition to the parallelism introduced by adding more GPUs, pushing larger batches while doubling the number of GPUs, improves the device utilization considerably.

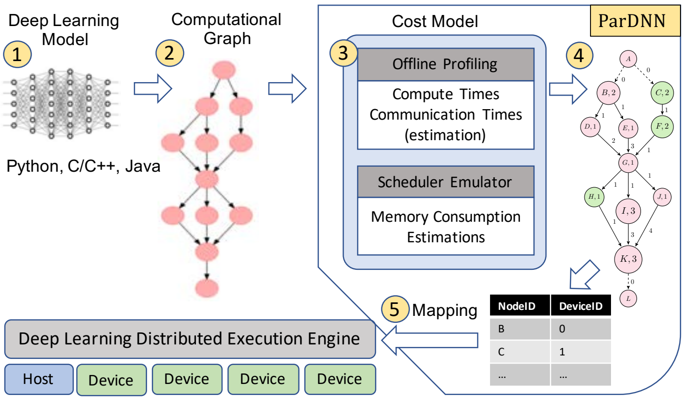
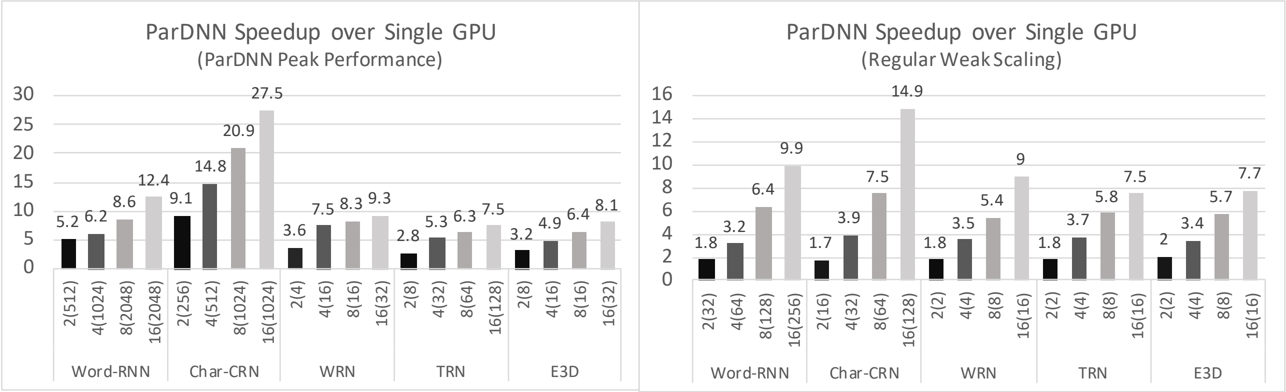


Figure 18.2: *ParDNN* Overview



(a)

(b)

Figure 18.3: *ParDNN* speedup over a single GPU using 2, 4, 8 and 16 GPUs. X-axis: Number of GPUs (Batch Size)

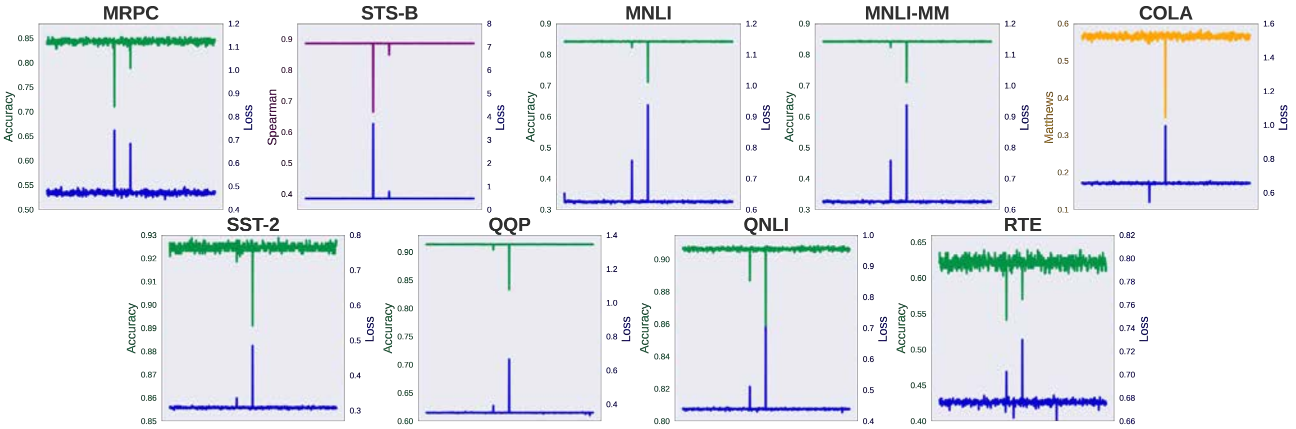
18.3.3 Improving design and evaluation of large language models

Team members continued to improve design of language models and their evaluation methods. To support this line of research, jointly with involved collaborators we have successfully applied for the HPCI Fugaku Preliminary Use Project (Startup Preparation) ”Training Novel Types of Large-Scale Language Models: Preparation”, adopted for the period from 2020/10/01 to 2021/03/31. The preparation phase project was fallowed but the regular application, which was successfully accepted for the FY 2021.

Within the NLP line of research we have made some advances in analysis methods as well as understanding training dynamics of neural language models. While multiple studies have shown that Transformers are remarkably robust to pruning in general, we demonstrate that pre-trained Transformer encoders are surprisingly fragile to the removal of a very small number of features in the layer outputs (¡0.0001% of model weights). In case of BERT and other pre-trained encoder Transformers, the affected component is the scaling factors and biases in the LayerNorm. The outliers are high-magnitude normalization parameters that emerge early in pre-training and show up consistently in the same dimensional position throughout the model. We show that disabling them significantly degrades both the MLM loss and the downstream task performance. This effect is observed across several BERT-family models and other popular pre-trained Transformer architectures, including BART, XLNet and ELECTRA; we also show a similar effect in GPT-2.

18.4 Schedule and Future Plan

Supercomputer Fugaku is now in production stage, but the software ecosystem for deep learning on this platform still has potential for improvement, and this is one for the objectives set for FY 2021. Additionally, early



,

Figure 18.4: Performance of BERT-base on GLUE tasks when one LayerNorm weight at a time is disabled throughout the model. Dimensions are shown on the X axis. Loss (blue) and accuracy, or correlation coefficients, where applicable (other colors) are shown.

exploratory work on design of the next generation machine had been started. We plan to develop simulation methodologies to evaluate not yet existing hardware across a wide verity of benchmarks, including deep learning models and kernel.

In the field of language models developing we intend to further scale up pretraining process, as well as to add more downstream tasks to the evaluation pipeline. We are developing a software solution to simplify training and evaluation of language models at large scale and plan to release it in FY 2021.

18.5 Publications

18.5.1 Articles/Journal

[1] Yosuke Oyama, Naoya Maruyama, Nikoli Dryden, Erin McCarthy, Peter Harrington, Jan Balewski, Satoshi Matsuoka, Peter Nugent, Brian Van Essen, The Case for Strong Scaling in Deep Learning: Training Large 3D CNNs with Hybrid Parallelism, Special Section on Parallel and Distributed Computing Techniques for AI, ML, and DL, In IEEE Transactions on Parallel and Distributed Systems (TPDS), vol. 32, (2021), pp. 1641-1652

[2] Martin Schlueter, Mehdi Neshat, Mohamed Wahib, Masaharu Munetomo, Markus Wagner, GTOPX Space Mission Benchmarks, ELSEVIER SoftwareX, Volume 14, (2021), 100666

[3] Fareed Qararyah, Mohamed Wahib, Doga Dikbayır, Mehmet Esat Belviranli, Didem Unat. A Computational-Graph Partitioning Method for Training Memory-Constrained DNNs, Elsevier Parallel Computing, Vol 104-105, (2021), 102792

18.5.2 Conference Papers

[4] Lingqi Zhang, Wahib Mohamed, Haoyu Zhang, Matsuoka Satoshi, A Study of Single and Multi-device Synchronization Methods in Nvidia GPUs, In 2020 International Symposium on Parallel and Distributed Processing (IPDPS), Online, Originally New Orleans, LA, USA

[5] Martin Schlueter, Mohamed Wahib, and Masaharu Munetomo, New state-of-the-art Results on ESA's Messenger Space Mission Benchmark, In PDPTA20 (The 26th Int'l Conf on Parallel and Distributed Processing Techniques and Applications), Online, Originally Las Vegas, NV, USA

[6] Mohamed Wahib, Haoyu Zhang, Truong Thao Nguyen, Aleksandr Drozd, Jens Domke, Lingqi Zhang, Ryousei Takano, Satoshi Matsuoka, Scaling Deep Learning Workloads Beyond Memory Capacity, SC20 (The International Conference for High Performance Computing, Networking, Storage, and Analysis), Online, Originally Dallas, TX, USA

[7] Jens Domke, Emil Vatai, Aleksandr Drozd, Peng Chen, Yosuke Oyama, Lingqi Zhang, Shweta Salaria, Daichi Mukunoki, Artur Podobas, Mohamed Wahib, Satoshi Matsuoka, Matrix Engines for HPC: A Performance Study from the Applications Perspective, In 35th IEEE International Parallel & Distributed Processing Symposium (IPDPS 2021), Online, Originally Portland, OR, USA

[8] Truong Thao Nguyen, Mohamed Wahib, System Co-design for Large-Scale Training of Distributed Deep Learning, In The 21th IEEE/ACM International Symposium on Cluster, Cloud and Internet Computing (CCGrid 2021), Online, Melbourne, Victoria, Australia

[9] Albert Khaira, Truong Thao Nguyen, Leonardo Bautista Gomez, Ryousei Takano, Rosa Badia, Mohamed Wahib, An Oracle for Guiding Large-Scale Model/Hybrid Parallel Training of Convolutional Neural Networks, In Proceedings of the 30th International Symposium on High-Performance Parallel and Distributed Computing (HPDC21), June 2020

[10] Peng Chen, Mohamed Wahib, Xiao Wang, shinichiro takizawa, Takahiro Hirofuchi, Ogawa Hirotaka, Satoshi Matsuoka, Performance Portable Back-projection Algorithms on CPUs: Agnostic Data Locality and Vectorization Optimizations, In ICS '21: Proceedings of the ACM International Conference on SupercomputingJune 2021 (ICS21), Online, June 2021

[11] Peng Chen, Mohamed Wahib, Xiao Wang, Takahiro Hirofuchi, Hirotaka Ogawa, Ander Biguri, Richard Boardman, Thomas Blumensath, Satoshi Matsuoka, Scalable FBP Decomposition for Cone-Beam CT Reconstruction, In SC '21: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, November 2021

[12] Prajjwal Bhargava, Aleksandr Drozd, Anna Rogers, Generalization in NLI: Ways to [Not] Go Beyond Simple Heuristics, In Proceedings of the Second Workshop on Insights from Negative Results in NLP (Insights 2021), November 2021

[13] Tadashi Yamazaki, Jun Igarashi, Hiroshi Yamaura, Human-scale Brain Simulation via Supercomputer: A Case Study on the Cerebellum, In Neuroscience, 462, 235–246.

[14] Hiroshi Yamaura, Jun Igarashi, Tadashi Yamazaki, Simulation of a Human-Scale Cerebellar Network Model on the K Computer, Frontiers in Neuroinformatics, 03 April 2020

18.5.3 Posters

18.5.4 Invited Talks

18.5.5 Oral Talks

[16] Peng Chen, Mohamed Wahib, shinichiro takizawa, Takahiro Hirofuchi, Ogawa Hirotaka, Satoshi Matsuoka, Efficient FDK Algorithms on SIMD-accelerated Processors, SWoPP 2020, Online

[17] 野村 哲弘, 遠藤 敏夫, 三浦 信一, 朝倉 博紀, 越野 俊充, 草間 俊博, TSUBAME3のインタラクティブ利用の利便性向上にむけた取り組み, SWoPP 2020, Online

18.5.6 Software

Benchmarker: We have released our implementation of a solution to automate benchmarking of various deep learning and related (GEMM, allreduce etc) kernels as an open-source software labelled *benchmarker*. It is currently hosted at the following URL: `https://github.com/undertherain/benchmarker`

18.5.7 Patents

Part II

Operations and Computer Technologies Division

Chapter 19

Facility Operations and Development Unit

19.1 Members

Toshiyuki Tsukamoto (Unit Leader)

Satoshi Matsushita (Technical Staff)

Katsuyuki Tanaka (Technical Staff)

Fumio Tsuda (Technical Staff)

Makoto Igasaki (Technical Staff)

Hiroshi Shibata (Technical Staff)

19.2 Overview of Research Activities

Our facilities possess multiple features not found at other supercomputer sites. These features include an expansive and pillar-free computer room, a power supply system that consists of a co-generation system (CGS) and a high-speed current-limiting circuit breaker without uninterruptible power supply (UPS), distribution boards installed not on the computer-room walls but under a raised floor, extremely quiet and high-efficiency air conditioners, and a water-cooling system for the CPUs featuring precise temperature control.

To ensure stable operation of the supercomputer and its peripherals, the facility operations and development unit (FODU) of the operations and computer technologies division, RIKEN R-CCS, is responsible for the operation and enhancement of the facilities. Furthermore, FODU conducts research on the advanced management and operations of the R-CCS facilities.

One of the most serious problems is the rapid and substantial increase in the electricity prices since 2011. Therefore, we are investigating the most suitable driving conditions to allow the R-CCS facilities to achieve effective cost reductions.

Another problem is the increased power consumption by R-CCS. The use of electricity by R-CCS is strictly limited by a contract between R-CCS and the local electric supply company. However, in the early stage of operation, the facility's power consumption exceeded the contract limit. This is important because the company requires us to accept an increase in the upper/lower power limit, which amounts to an increase in the electricity cost. To prevent this, we have investigated methods to control the power consumption of the supercomputer using emergency job stopping together with the system operations and development unit and the application tuning development team of the operations and computer technologies division, RIKEN R-CCS.

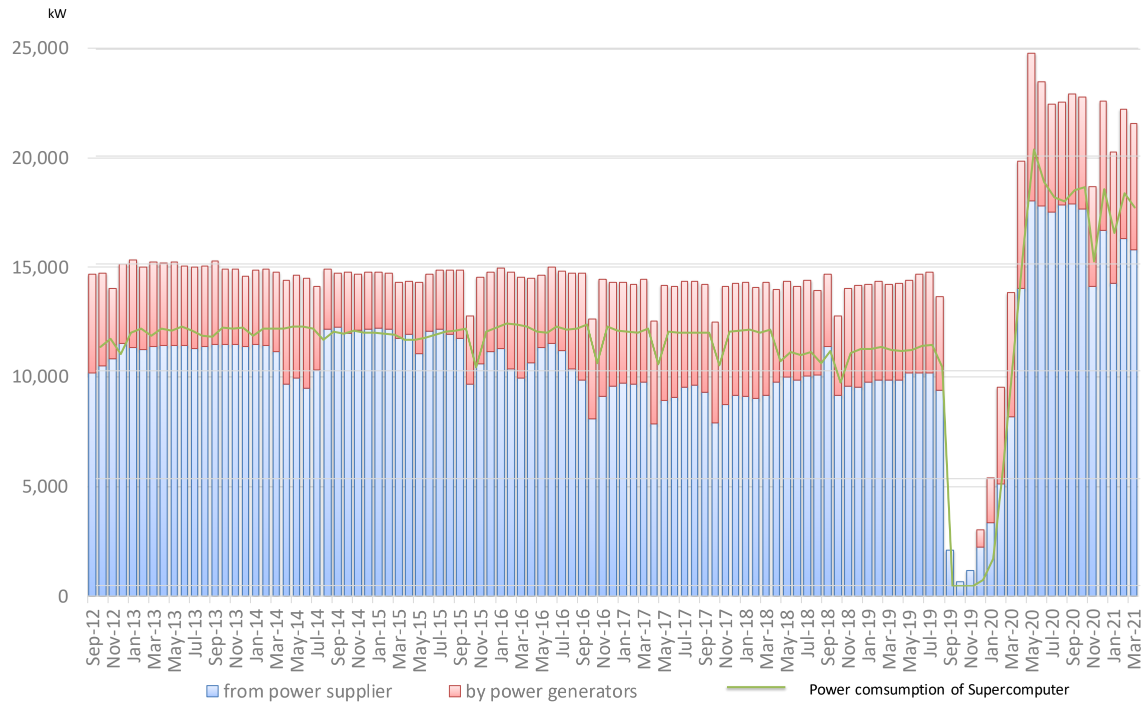


Figure 19.1: Monthly power supply and power consumption of supercomputer.

19.3 Research Results and Achievements

19.3.1 Optimum operation of electric power

Figure 19.1 shows the monthly total power supply and power consumption of our supercomputer(K and Fugaku) from September 2012 to March 2021. The power supply consists of commercial power purchased from a supply company and power generated by CGS.

The R-CCS power consumption is nearly synchronized with that of our supercomputer. From October 2012 to August 2019, the power consumption of R-CCS is nearly 14,000 kW on average, and the power consumption of the K Computer accounts for approximately 80% (11,000 kW) of the total consumption of R-CCS.

As shown in Figure 19.1, the electric power supply of R-CCS consists of commercial and CGS power. There are two CGS systems at R-CCS, and they are used in turn for two weeks at a time. Therefore, at least one CGS is always in use. From October 2012 to August 2019, commercial electric power is contractually set at approximately 12,500 kW, and the power consumption was approximately 11,000 kW (annual average), which corresponds to approximately a 90% load factor.

From December 2019 to May 2020, Fugaku was installed and conducted high load tests. After that, we built and tested the operation system, and started operation in March 2021.

To minimize the cost, we try to optimize the ratio of the commercial and CGS electricity.

To investigate the optimized conditions that minimize the sum of the electricity and gas cost, we determined the costs of several ratios of commercial electricity to CGS electricity. We also constructed a model to describe the energy flow of the electric power supply and the cooling system. Then, we performed computer simulations using the model and the actual operating data. In the near future, we intend to identify the cost-optimized conditions that contribute to reducing costs.

19.3.2 Improvements to the power usage effectiveness (PUE)

We have continued to work on improvements for the effective use of electricity. PUE is a well-known indicator of the effectiveness of electricity use.

To improve the PUE, we have attempted to optimize the operation of the cooling equipment (e.g., chillers and air-handlers) since FY2013.

Figure 19.2 indicates the change in the annual average power consumption of the K Computer (including

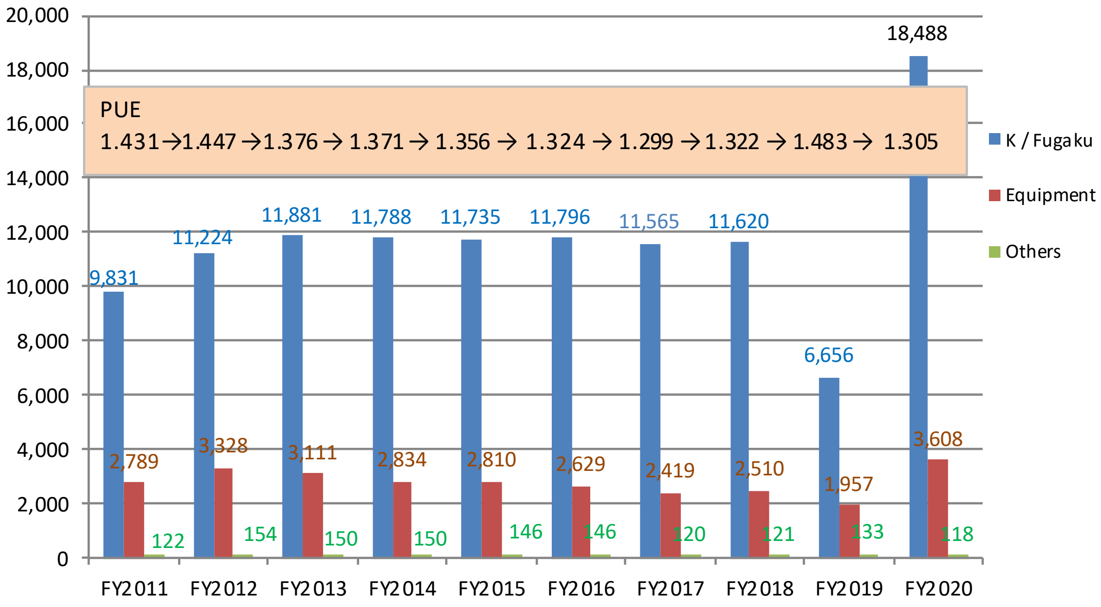


Figure 19.2: Trend in the annual average electric power consumption.

the peripheral devices) and the cooling equipment. Since FY2013, the power consumption of the K Computer has been nearly flat at approximately 11,800 kW; however, the power consumption of the equipment decreased gradually from FY2013 to FY2017. Accordingly, the PUE of R-CCS improved to 1.322 in FY2018 from 1.447 in FY2012, contributing to the reduction in the electricity cost.

In FY2013, we reduced the electricity cost of the air conditioners by reducing the number of working air conditioners. The total cooling performance was maintained by lowering the air temperature. We achieved a reduction in the power consumption of 217 kW.

In FY2014, we focused on the fault-tolerance feature of the air-conditioning equipment. Each air conditioner has two motors for fault tolerance. We found that if one of the two motors could be stopped, the airflow could be maintained at approximately 60%. Therefore, we reduced the power consumption by a further 277 kW in FY2014 and by 24 kW in FY2015.

In FY2016, we worked on improving the efficiency of the cooling tower. As a result, we achieved a reduction of 181 kW in the power consumption.

In FY2017, we focused on optimizing the operation control of the refrigerator using a heat storage tank. This reduced the power consumption by 210 kW.

In FY2018, we carried out the overhaul of a cooling tower and a large number of pumps. However, we were not able to operate CGS by the effective output because the electric equipment disorder caused by the typhoon occurred. Therefore, we were not able to improve PUE of FY2018 from last year.

The operation of K Computer was completed in August 2019, and the installation of Fugaku started in December. All installations were completed in May 2020, introduction tests and operation environment construction were conducted, and the service was put into service in March 2021.

19.3.3 Facility Expansion Work

This work was carried out between January 2019 and March 2020. We have greatly enhanced the power supply and cooling capacity for the Fugaku installation. Specifically, we doubled the number of 6000V/200V transformers, and tripled the cooling capacity per rack. In addition, the necessary chillers were added.

Since the work was to be carried out in parallel with the operation of the K Computer until the end of August 2019, we had to be very careful not to affect our operation. In addition, a strict schedule was implemented to remove all the K Computers by the end of September and to complete the under-floor facility work so that Fugaku could be installed in December.

In 2020, the power consumption increased due to the installation test of Fugaku. We conducted cooling capacity tests and evaluations to establish a reliable operating method.

19.4 Schedule and Future Plan

We will continue to improve the advanced management and operation of the R-CCS facilities and contribute to the user service of our supercomputer. We will work on reducing costs by investigating and applying the most suitable driving conditions to all the electric power supply and cooling equipment. Further, we will improve the electric power and cooling control of the entire R-CCS facility with the system operations and development unit to minimize the electric power consumption.

19.5 Publications

19.5.1 Conference Papers

[1] Masaaki Terai, Fumiyoshi Shoji, Toshiyuki Tsukamoto and Yukihiro Yamochi, “A Study of Operational Impact on Power Usage Effectiveness using Facility Metrics and Server Operation Logs in the K Computer”, EEHPCSOP 2020 short paper

19.5.2 Oral Talks

[2] Keiji Nonose, Yuhei Seki, Iwao Hasegawa, Hajime Naemura, Satoshi Matsushita, Toshiyuki Tsukamoto, “Facility Renovation and Operational Verification for Supercomputer Fugaku Part1 Overview of Facility Renovation to increase equipment capacity”, Proceedings of the 53th Japanese Joint Conference on Air-conditioning and Refrigeneration(Tokyo). (In Japanese)
[3] Yuhei Seki, Keiji Nonose, Iwao Hasegawa, Toshiyuki Tsukamoto, Hajime Naemura, “Facility Renovation and Operational Verification for Supercomputer Fugaku Part2 TAB of Cooling Performance under the Test Load of Fugaku”, Proceedings of the 53th Japanese Joint Conference on Air-conditioning and Refrigeneration(Tokyo). (In Japanese)

Chapter 20

System Operations and Development Unit

20.1 Members

Atsuya Uno (Unit Leader)

Yuichi Tsujita (Senior Technical Scientist)

Hitoshi Murai (Research & Development Scientist)

Fumio Inoue (Research & Development Scientist)

Katsufumi Sugeta (Technical Staff)

20.2 Overview of Research Activities

Supercomputer Fugaku, a distributed-memory parallel computer comprising 158,976 computing nodes, played a central role in the High Performance Computing Infrastructure (HPCI) initiative granted by the Ministry of Education, Culture, Sports, Science and Technology. The HPCI has achieved the integrated operation of the supercomputer Fugaku and other supercomputer centers in Japan. Further, it has enabled seamless access from user machines to a cluster of supercomputers that includes the supercomputer Fugaku. Moreover, the HPCI has provided large-scale storage systems that are accessible from all over Japan.

The production of the supercomputer Fugaku was started in April 15, 2019 and transportation of the real machine to RIKEN started on December 3, 2019 and ended on May 13, 2020. The installation adjustment by Fujitsu was done until the end of December 2020. While in the period of the installation adjustment, part of the machine was offered to the respective users of the Gordon Bell Prize Challenge, the MEXT's Program for Promoting Researches on the Supercomputer Fugaku, and researches for COVID-19. The shared use of the supercomputer Fugaku has started since March 9, 2021.

While in the period of the installation adjustment, the system operations and development (SOD) unit prepared for the shared use of the supercomputer Fugaku. SOD studied the system configuration for the shared use, including the system configuration, the job scheduling, the file system, the user environment and etc.

In the fiscal year 2020 (JFY2020), we primarily implemented improvements to the following operational issues:

- I/O Characterization using System Operation Log
- Job Classification about Electric Power Consumption using System Operation Log
- Load-Balancing in I/O Workload on the Second-Layer Storage System of the Fugaku
- Support for power resource management functions on Fugaku

20.3 Research Results and Achievements

The installation of all computer racks completed at May 13, 2020, and the installation adjustment was conducted until the end of December 2020. Some of compute nodes were provided to the respective users as the evaluation environment in the trial phase, and the shared use started at March 9, 2021. The number of available compute nodes and available period during the evaluation environment in the trial phase were unstable, and figure 20.2 shows the node usage and a number of available nodes in JFY2021. The shared use started on March 9, 2021, so the operation period of the shared use is about 1 month. Figure 20.2 shows the resource usage details per day, and there was no major failure in this period.

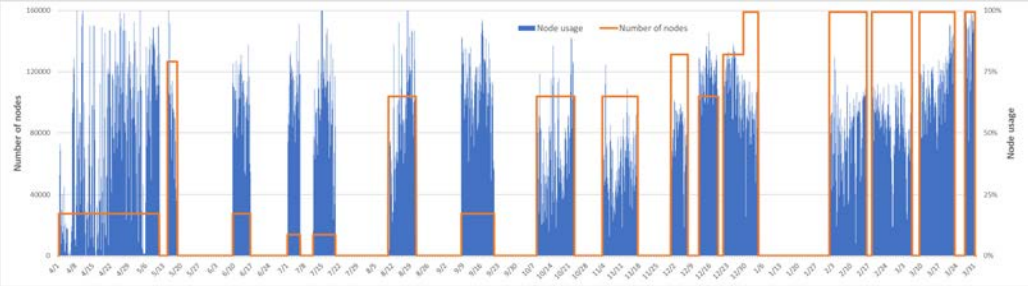


Figure 20.1: Node usage and number of available compute nodes in JFY2020

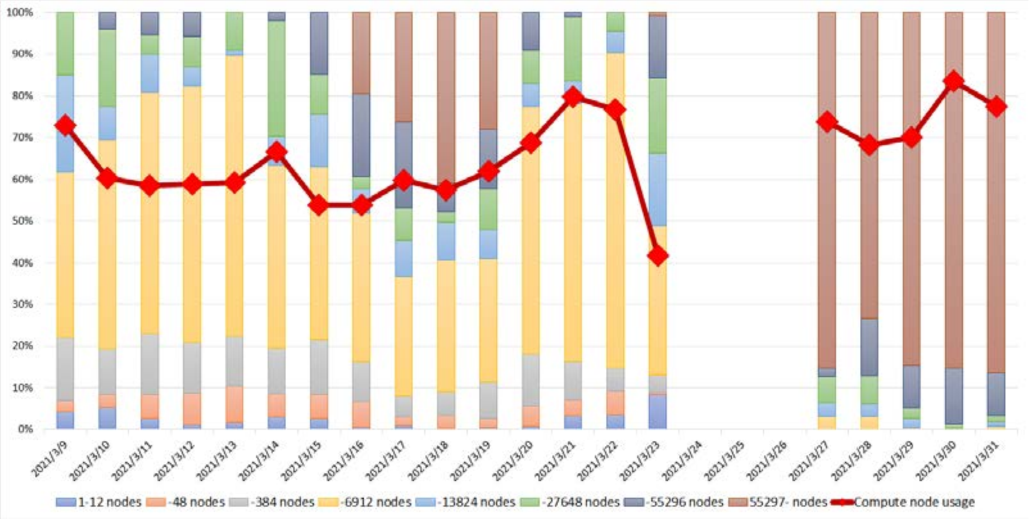


Figure 20.2: Resource usage in JFY2020

There was a system maintenance from March 23 to March 27, the Gordon Bell Prize Challenge using half of all the nodes from March 16 to March 19, and the Gordon Bell Prize Challenge using all the nodes from March 27 to March 31 respectively. The average system usage is approximately 65.2%.

20.3.1 Activities for Stable System Operation

20.3.1.1 I/O Characterization using System Operation Log

We have already built an I/O characterization analysis framework to characterize I/O activities of applications at the K computer in the last fiscal year. The analysis framework utilized log information about I/O activities of the file system and Tofu links among I/O nodes, which were collected from the K computer with the help of job information such as used compute nodes stored in a PostgreSQL database.

In this fiscal year, we have conducted further analysis about another I/O benchmark run that we did in the last few months of the K computer operation. Through this study, we have found several distinctive differences in activities in each I/O subsystem component in terms of I/O optimization configuration at I/O benchmark run. We expect that similar analysis framework would be useful in the Fugaku operation.

20.3.1.2 Job Classification about Electric Power Consumption using System Operation Log

We have conducted machine learning study about job classification in terms of electric power consumption and file I/O activities using system operation log to exploit new insights in the forthcoming Fugaku operation. In this study, we have examined possibility of job classification in electric power consumption based on not only computing activities such as CPU and memory utilization but also file I/O activities using the logs of executed jobs. We have shown effectiveness of the job classification using the *RandomForestClassifier* among several classifier models provided from *scikit-learn* Python library.

20.3.1.3 Load-Balancing in I/O Workload on the Second-Layer Storage System of the Fugaku

One of the important issues about the filesystem operation is providing stable I/O operations for users under a variety of I/O workloads. In order to mitigate I/O contention on the second-layer storage system of the Fugaku due to heavy I/O tasks, we have introduced load-balancing scheme on meta-data servers and object storage servers of the storage system using the QoS mechanism of the FEFS. Especially, we have newly adopted the fairshare control mechanism on the QoS, where each task on a server cannot exceed the predefined maximum number of I/O threads according to the fairshare control. As a result, other light-weight tasks can proceed their operations without waiting long time until I/O threads are given.

20.3.1.4 Support for power resource management functions on Fugaku

In K computer, resource management was performed on the basis of “node hours” defined as the product of job execution time and the number of nodes. In Fugaku, power consumption becomes one of important problems related to system operation. So, we have extended the resource management function to handle the amount of resources in any manner from which the information needed for calculating charges can be acquired, and the power resources are added as one of users’ resources. We also provide a tool that displays the power consumption of each job. With this resource management, users can run jobs until either computational or power resources are exhausted.

20.4 User support

The number of projects and the number of registered users were approximately 260 and 2300 respectively at the end of March 2021. The number of active daily users was approximately 270 in March 2021.

We supported users through the support desk and provided them with technical information regarding the supercomputer Fugaku, information regarding its system environment, system tools, compilers and libraries. In addition, we performed user registration, failures investigation, software installation, and etc. We offered our consulting services together with members in the operation and computer technology division. Figure 20.3 presents details of the issues addressed in March 2021, and the number of issues was 96. The most common issues were related to the user environment such as the user registrations and expanding the disk space, followed by the job executions.

20.5 Schedule and Future Plan

In this fiscal year, the installation adjustment was conducted until the end of December 2020, and the shared use started on March 9, 2021. While in the period of the installation adjustment, part of the machine was offered to the respective users and we prepared for the shared use.

The operation of the supercomputer Fugaku has just started, we will work for the stable operation of the supercomputer Fugaku with the experience we’ve gained through the operation of the K computer.

20.6 Publications

20.6.1 Articles/Journal

[1] Shigeto Suzuki, Michiko Hiraoka, Takashi Shiraishi, Enxhi Kreshpa, Takuji Yamamoto, Hiroyuki Fukuda, Shuji Matsui, Masahide Fujisaki, Atsuya Uno, Power Prediction for Sustainable HPC, IPSJ Transactions on Advanced Computing System, vol.14, No.1, February 2021.

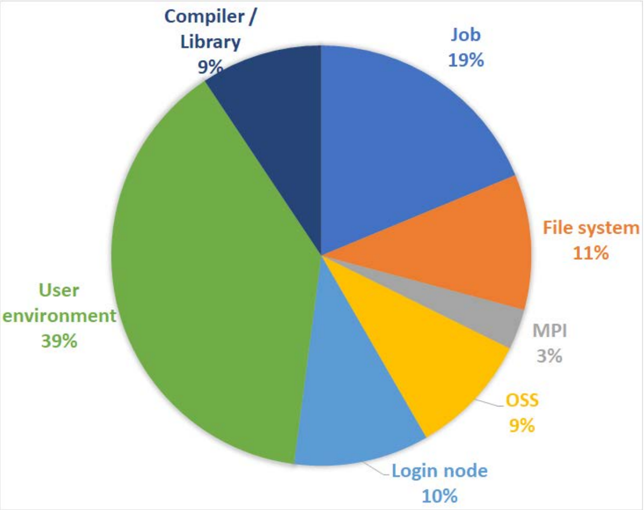


Figure 20.3: Breakdown of issues addressed in March JFY2020

20.6.2 Conference Papers

[2] Yuichi Tsujita, Yoshitaka Furutani, Hajime Hida, Keiji Yamamoto, Atsuya Uno, Characterizing I/O Optimization Effect Through Holistic Log Data Analysis of Parallel File Systems and Interconnects, LNCS 12321, Springer, pp. 177–190 2020.
[3] Yuichi Tsujita, Atsuya Uno, Ryuichi Sekizawa, Keiji Yamamoto, Fumichika Sueyasu, Job Classification Through Long-Term Log Analysis Towards Power-Aware HPC System Operation, Proc. of 29th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP), IEEE, pp. 26–34, March 2021

20.6.3 Posters

[4] Atsuya Uno, Consideration of Operation of the supercomputer Fugaku, The 3rd International Symposium poster, February 2021.

20.6.4 Invited Talks

[5] Atsuya Uno, Introduction of Supercomputer Fugaku, HPCSYSPROS20, SC20, Nov.24–25, 2020.

20.6.5 Oral Talks

[6] Yuichi Tsujita, Status of Lustre-Based Filesystem at the Supercomputer Fugaku, LUG Webinar, September 9, 2020
[7] Yuichi Tsujita, Storage System at the supercomputer Fugaku, JLUG2020, December 4, 2020

Chapter 21

Application Tuning Development Unit

21.1 Members

Kazuo Minami (Unit Leader)

Akiyoshi Kuroda (Research & Development Scientist)

Kiyoshi Kumahata (Research & Development Scientist)

Kazuto Ando (Technical Staff I)

Keigo Nitadori (Technical Scientist)

21.2 Overview of Research Activities

Toward completion of the supercomputer Fugaku, the collaboration between the system and the applications in use is the key to create innovative results. In order to maximize the performance and usability of both the system and such applications, the Application Tuning Development Unit conducted the following activities:

(1) Activities to establish a deep learning (DL) environment on the supercomputer Fugaku

(2) Performance evaluation and tuning of CUBE on Fugaku

(3) Basic research for constructing a flow solution method using a neural network

(4) Support of Fugaku early access program system

(5) Effort for the Fugaku supercomputer development

21.3 Research Results and Achievements

The application tuning development team has been conducting performance evaluation and enhancement aimed at popularizing the applications developed by the RIKEN Center of Computational Science (R-CCS) research team (i.e., R-CCS software) from applications on the K computer and the Fugaku supercomputer. By improving and enhancing the software, we expect industries and communities that have not used the R-CCS software to start using this software. Furthermore, along with the above activities, we are also trying to systemize performance optimization technology. Improving application performance will lead to a shortening of the elapsed time, which makes it possible to use more computational resources. This will help in the more effective utilization of resources. We kept these factors in mind while carrying out the following tasks.

21.3.1 Activities to establish a deep learning environment on the supercomputer Fugaku

Recently, machine learning, especially deep learning (DL), has become commonplace. Deep learning calculation consists of a framework for flexibly handling complex network structures and a library (DNNL) for computing typical calculations with high performance. In addition, applications and research using graphics processing units (GPUs) are advancing in DL science fields. The GPU is said to be essential for DL. However, surveys up to last year have shown that even massively parallel computers, such as the supercomputer Fugaku, can perform high-performance calculations by taking advantage of the massively parallel characteristics of the CPUs. Focusing on this point, we performed the following work this year, including the performance tuning of Chainer, which is a representative framework, with the goal of contributing to user convenience.

21.3.1.1 Maintenance of framework environment for the Fugaku

Aiming to promote the use of the Fugaku, we have developed the use environment for AI calculations, and provided the environment for the user of the Fugaku early access program. Last year, Chainer-4.5.0 which has been accelerated more than 30x on the K computer, also provided on the Fugaku as R-CCS software same as the time of operation of K computer. PyTorch-1.5.0, TwnaorFlow-2.1.0 was also developed and released on the Fugaku, by built-in the special high performance library oneDNN developed by the Fujitsu Research Institute.

21.3.1.2 Evaluation of parallel performance using Chainer

The high-performance deep learning training on the Fugaku requires massive parallelization. We evaluated the parallel performance of AI training on the Fugaku using Chainer which is accelerated in last year. It was found that communication and I/O become bottle neck at massive parallelization, although the calculation time is scaled using the weak scaling problem. The problem is because of MDS access concentration by import of Python modules which is composed of many files. Since LLIO is not available at the development stage of the Fugaku, we proposed the method of one file staging to the BIO, and this access concentration was avoided. The parallel performance is scalable until 72 racks and achieved 410,726 [ips (images per seconds)] [Figure 21.1]. This approach is provided as a standard usage for users of Fugaku early access program. Chainer, which achieved 55.2 [ips] last year, achieved 66 [ips] this year by calling tcmalloc under boost mode. This performance exceeds 44.7 to 63.6 [ips], which performance is the estimation by framework tuning using the Fortran kernel.

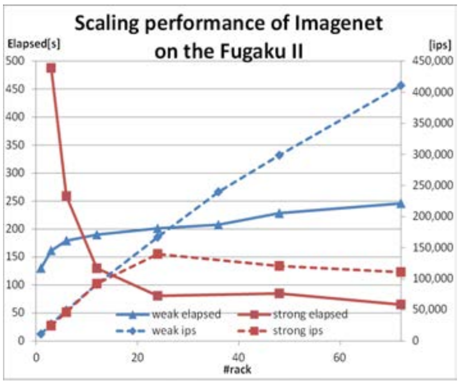
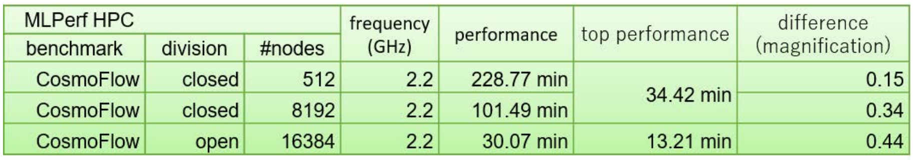


Figure 21.1: Massive parallel performance of Chainer.

21.3.1.3 Contribution to real application benchmark MLPerf HPC

We performed the evaluation and measurement of MLPerf HPC. The high performance library oneDNN developed by Fujitsu Laboratories was used to speed up major training calculation. Our unit optimized I/O and communication performance. For I/O relationships, the staging time was reduced to about 1/10 by using the SSD that can be shared in BOB instead of the /tmp on the SSD and LLIO. For the communication of the parallel models in BOB data sharing, we measured using the rank arrangement for simultaneously optimizing peer-to-peer communication in the BOB and global communication. CosmoFlow of a three-dimensional image analysis problem achieved the second-place performance though did not reach ABCI. It showed that AI of HPC was driven by Japan in both CPU performance and parallel performance [Table 21.1] of the CPU system. We could appeal the existence value of the Fugaku by showing parallel performance of 16,384 CPU.

Table 21.1: Summary of MLPerf HPC ranking performance.



MLPerf HPC			frequency (GHz)	performance	top performance	difference (magnification)
benchmark	division	#nodes				
CosmoFlow	closed	512	2.2	228.77 min	34.42 min	0.15
CosmoFlow	closed	8192	2.2	101.49 min		0.34
CosmoFlow	open	16384	2.2	30.07 min	13.21 min	0.44

21.3.2 Performance evaluation and tuning of CUBE on Fugaku

For the problem idealized for the performance benchmark, the result of executing CUBE on Fugaku and measuring its performance is shown. (Using a compressible fluid solver).

The execution performance when the compressible solver of CUBE is operated on a single node of Fugaku is shown. As a problem for performance benchmarks, we deal with cavity flow in a region that extends long in the y-axis direction. This is set so that the area division by each MPI process is ideal (one-dimensional).

Table 21.2 shows the performance of the entire program (time integration loop). The performance of one node (4 MPI process) is 3.59 times that of 1 CMG (1 MPI process), which is generally better than the ideal performance improvement (4 times). The double-precision floating-point arithmetic performance is 59.96GFLOPS for 1 CMG and 215.84GFLOPS for 1 node, which are 7.09% and 6.38% of the peak performance, respectively, which are high values for CFD application performance.

Table 21.2: Time integration loop performance.

	Execution time	Calc. performance	Mem. throughput
	[seconds]	[GFLOPS]	[GB/sec.]
1 CMG	38.40	59.96 (7.09%)	48.44 (18.92%)
1 node	10.67	215.84 (6.38%)	171.90 (16.78%)

Table 21.3 shows the computational performance per CMG (1 process) of the high-cost viscous term kernel part and its upper limit estimate. The upper limit estimate was calculated based on the ratio of the amount of memory access of the source and system to the number of operations (Byte per Flop). Due to the tuning results, the peak performance ratio has improved from the original 7.03% to 13.95%. Estimates suggest a further 2.7% improvement.

Table 21.3: Viscosity term calculation performance measured value and upper limit estimated value (when running 2.0 GHz).

Original	Tuning	Estimated upper limit
53.87 GFLOPS	106.75 GFLOPS	128.07 GFLOPS
(7.03%)	(13.95%)	(16.67%)

Table 21.4 shows the computational performance per CMG (1 process) of the high-cost convection term kernel part and its upper limit estimate. For the convection term, this kernel is not a memory throughput bottleneck because the ratio of the source memory access to the number of operations (Byte per Flop) is below the system value. Therefore, it is not possible to calculate the performance upper limit estimated value by Byte per Flop, but a significant performance improvement can be expected by computing performance tuning such as software pipelining and SIMD.

Evaluate CUBE's weak scaling using idealized questions for benchmarking.

The cavity flow in the region extending in the y-axis direction per node is treated as a benchmark. When measuring scaling, the area is extended in the y-axis direction according to the number of nodes. As a result,

Table 21.4: Convection term calculation performance measured value and upper limit estimated value (when running 2.2GHz).

Original	Tuning	Estimated upper limit
82.23 GFLOPS	123.98 GFLOPS	-
(10.10%)	(14.70%)	(-%)

this problem is divided into areas only in the y-axis direction, and communication is performed in the same one-dimensional pattern regardless of the number of nodes. In addition, it is guaranteed that the process responsible for the adjacent subregions will be mapped to the adjacent nodes in the space of TofuD's interconnect TofuD.

Table 21.5 shows the results of measuring the weak scaling from 1 node to 27,648 nodes of Fugaku using the above problem. Looking at the execution time, it is 27.24 seconds for 27,648 nodes compared to 25.48 seconds for one node, and the increase in execution time is suppressed to about 10%. As a result, the scaling of 27,648 nodes per node is 91.07%, which is a good value. In addition, it was confirmed that the scale of calculation can be expanded to at least 21.7 billion cells and 4.5 PFLOPS.

Table 21.5: Weak scaling results of ideal problem.

Nodes	Cells	Exec. time [seconds]	FLOPS [TFLOPS]	Scaling
1	786,432	25.48	0.18 (5.38%)	100.00%
256	201,326,592	26.94	44.00 (5.09%)	94.57%
512	402,653,184	26.92	88.09 (5.09%)	94.67%
1,024	805,306,368	26.86	176.55 (5.10%)	94.87%
4,096	3,221,225,472	27.05	701.37 (5.07%)	94.22%
8,192	6,442,450,944	27.07	1,401.68 (5.06%)	94.15%
16,384	12,884,901,888	27.24	2,785.88 (5.03%)	93.56%
27,648	21,743,271,936	27.98	4,575.74 (4.90%)	91.07%

The results of executing CUBE on Fugaku and measuring its performance for vehicle models are shown. (Using an incompressible fluid solver)

As evaluations of parallel performance, strong scaling (increasing the number of nodes by fixing the problem size) and weak scaling (increasing the number of nodes while fixing the problem size per node) were evaluated.

The result of weak scaling is shown. A vehicle model (130 million cells) was used for the calculation. The mesh is subdivided step by step so that the number of cells is eight times larger, and at the same time, the number of nodes is increased.

Table 21.6 shows the results of evaluating the change in elapsed time when the number of nodes is increased for the time integration loop (scaling is better as the elapsed time does not increase). The scaling at 51,200 nodes based on 100 nodes was 74%, and good results were obtained as the measurement results of the actual problem using the hierarchical grid.

Table 21.6: Weak scaling results of vehicle model

Nodes	Cubes	Cells	Time	Scaling
100	32,341	132,468,736	22.92 s	100.00%
800	260,331	1,066,315,776	32.63 s	70.23%
6,400	2,082,648	8,530,526,208	33.46 s	68.49%
51,200	16,661,184	68,244,209,664	30.927 s	74.11%

21.3.3 Basic research for constructing a flow solution method using a neural network

The ultimate goal is to build a contraction model of the flow field around the vehicle, and as a preliminary step, we are working on the construction of a contraction model of the three-dimensional flow field around the cylinder. Below, we show the mode division by CNN and the contraction model simulation by LSTM that we performed by large-scale distributed learning on Fugaku.

In this research, based on the neural network with an autoencoder type structure, we made changes to apply it to 3D flow field data.

Figure 21.2 shows how to parallelize the learning process of the neural network. It is a method that combines the following two parallelization schemes (hybrid parallelism).

1. The weights and calculations of the encoder section and the weights and calculations of the branched parts for each mode of the decoder section are distributed to separate MPI processes (model parallel).
2. The training data is distributed to separate MPI processes, and the amount of weight updates calculated by each person is reduced (averaged) by communication (data parallelism).

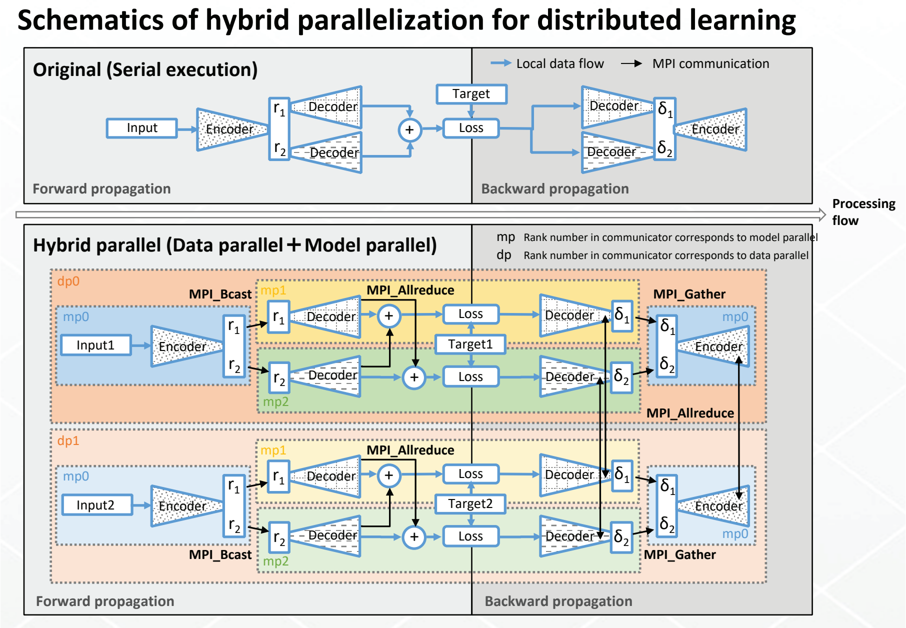


Figure 21.2: Parallelization of learning process.

LSTM, which is a type of RNN (Recurrent Neural Network), is a network that predicts the time change of latent vector by learning the time series of latent vector that is output when flow field data is given as input to the trained encoder. Implemented by (Long Short-Term Memory). Configure a network that takes the past 20 steps as input and outputs the past 19 steps + the next 1 step. As the network structure, we adopted a method of arranging LSTMs with different numbers of features in the Hidden state in parallel.

The result of the reduction simulation using the LSTM which learned the latent vector obtained by the mode division using CNN is shown.

Figure 21.3 shows a snapshot of the flow field u 500 seconds after the start of the simulation. Compared with the result of the high-precision model, the complicated flow structure of the wake of the cylinder cannot be reproduced in both the number of modes 2 and the number of modes 20, but the number of modes 20 has less phase shift.

Figure 21.4 shows a snapshot of the Q-value isosurface 50 seconds after the start of the simulation. Compared with the result of the high-precision model, the complicated vortex structure of the wake of the cylinder cannot be reproduced in both the number of modes 2 and the number of modes 20. In addition, there is no clear difference between the vortex structures of mode numbers 2 and 20. This is considered to indicate that even 20 modes are significantly insufficient to reproduce the complicated vortex structure of the high-precision model.

21.3.4 Operation and support of the Fugaku early access program

We supported for Fugaku early access program environment. In the first half it was in charge of compiler, numerical library, and language related failures, and corresponded to a large number of resulting abnormal disorders. In the second half it was in charge the problems of performance, numeric library, and AI Framework.

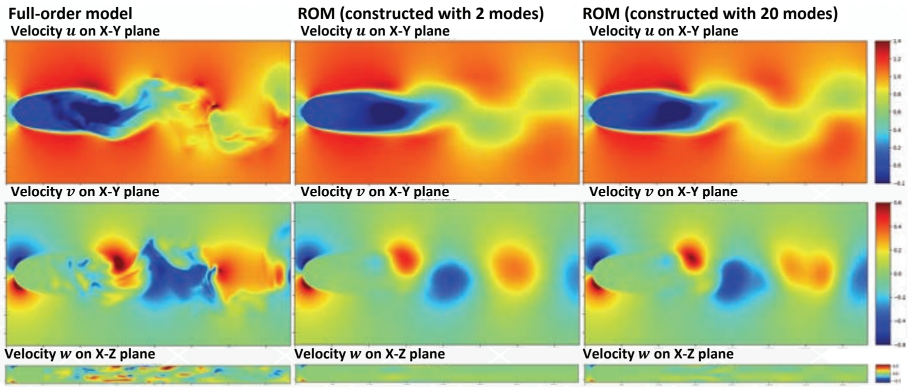


Figure 21.3: Snapshot of flow velocity u on the X-Y plane.

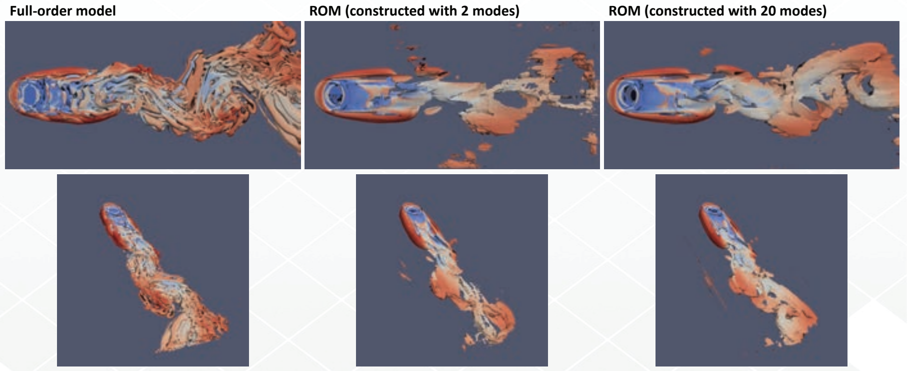

Full-order model
ROM (constructed with 2 modes)
ROM (constructed with 20 modes)

Figure 21.4: Q-value isosurface snapshot.

With regard to performance, the problem of memory leaks had a major impact. There is a report from the five applications of FFB, GKV, LQCD, RSDFT, GD5D from a long-time calculation of real problems. It was found that caused by Rendezvous communication, communicator generation destroy, MPI_IN_PLACE of the MPI_AllReduce, MPI_SEND_INIT.

21.3.5 Efforts in the development of the Fugaku supercomputer

The FLAGSHIP2020 (FS2020) Project is engaged in research and development for the Japanese national flagship supercomputer "Fugaku", which is a post K computer. As part of the FS2020 project, the application tuning development unit applied co-design of the application and Fugaku supercomputer system through the performance optimization and sophistication of target applications. In particular, our unit is responsible for these three applications.

- ADVENTURE: A structural analysis application based on the finite-element method in Priority Issue No. 6
- RSDFT: A first-principles material simulation and optimization application based on density functional theory, in Priority Issue No. 7
- FrontFlow/Blue: A fluid analysis application based on the finite-element method in Priority Issue No. 8

For these applications, we have paid significant attention to the following efforts:

- Application performance tuning
- Establishing methods for estimating application performance on the supercomputer Fugaku.
- Establishing test kernel codes for evaluating the CPU and system of Fugaku.

The following subsections describe application tuning.

21.3.5.1 Enhancement of the co-design application (ADVENTURE)

ADVENTURE, the structural analysis application based on the finite-element method, is composed of two major parts. One is a sub-domain solver, and the other is a coarse grid correction. The whole analysis domain is divided into multiple sub-domains. A sub-domain solver solves each sub-domain deformation. Coarse grid correction solves the inter-sub-domain relationship by representing a sub-domain as one point having six degrees of freedom. In both parts, a hot kernel is the multiplication of a dense matrix and a vector. Until last year, we continued to tune a hot kernel mainly by separating only the hot kernel source. In particular, we focused on the following issues:

- Avoiding unsuitable loop unrolling that caused cache thrashing to occur frequently
- Decreasing the number of data streams for effective cache memory utilization
- The essential procedure of multiplying a dense matrix and a vector is the taking the dot-product of two vectors. In order to apply SIMD instruction to the summation operation of the dot-product calculation effectively, we implemented a recursive summation operation and minimized the number of temporary arrays. Furthermore, we decreased the number of recursive summation operations to just one.

In previous years, the application used for evaluating performance on the Fugaku supercomputer was a limited version. For example, Fugaku omitted some functions and could only treat special input data. We could evaluate a full-function application. Using this version, a successful run of a sample problem, namely, a 4,096-node-sized problem (16,384 processes, 65,536 sub-domains), was conducted on the Fugaku. Subsequently, in order to validate the performance on the Fugaku, timer routines and tuned kernel codes were implemented. As a result, on Fugaku supercomputer, the target problem achieved over 60× faster performance than the K computer. In addition, power consumption was significantly lower than estimated previously.

21.3.5.2 Enhancement of the co-design application (RSDFT)

The main procedures in RSDFT are DGEMM, and collective communications are based on message passing interfaces (MPIs), such as MPI_Allreduce and MPI_Bcast. Owing to performance tuning conducted during K computer development, RSDFT achieved a performance efficiency of approximately 43.6% for the Si nano-wire problem and won the Gordon-Bell Award in 2011.

In this year, we conducted activities to achieve the target performance using Fugaku. In the execution of the target problem calculation, however the lack of memory was frequent, these problems could be avoided by some measures. By further performance tuning, finally it was achieved 37.5x the performance ratio of the K computer. The group communication performance by the representative process method confirmed that it was able to shorten to about 10There was no power gap that was concerned with EigenExa, and the maximum power was 29.9 [MW/sys].

21.3.5.3 Enhancement of the co-design application (FrontFlow/blue)

FrontFlow/blue is a fluid simulation code based on the finite-element method. Its major kernels are "kernels depending on memory bandwidth". The performance of these kernels is strongly dependent on the data transfer speed between the CPU and memory. Therefore, in order to tune the performance, it is necessary to obtain higher bandwidth rather than higher CPU calculation performance. In previous years, in order to increase the weak scaling performance on Fugaku, we performed a precise performance evaluation to spcify the main cause which prevent performance scaling in distributed parallel execution. Due to the pricise performance evaluation and tuning, this application indicated over 85% of weak scaling performance using 5,505,024 computational cores on Fugaku. Finally, the optimizations so far have been summarized in a paper and been selected as a finalist for ACM Gordon-Bell Prize.

21.4 Schedule and Future Plan

21.4.1 R-CCS software center activities

Activities related to the following two areas should be continued for use with the Fugaku supercomputer after the service of the K computer ends this summer. One is application improvements from the viewpoint of both performance and usability, and the other is activities to promote the advancement of a utilization environment for R-CCS software users via demonstrations, tutorials, documents, and other methods. Although we focused on five R-CCS software products including CUBE in our activities up to this point, it is undisputable that the focus could be expanded to additional software applications.

21.4.2 Activities to establish a DL environment on the Fugaku supercomputer

Under the policy of Society 5.0, in order for Japan to lead AI in HPC, we will make arrangements to promote the utilization of deep learning on the Fugaku. The following nine missions are listed. These cannot be realized by myself. It is carried out in collaboration with research teams, development teams, and other centers.

- Performance optimization: Continuous performance improvement of PyTorch, TensorFlow, ChainerK, ChainerX that have already been released. (always)

- Enhancement of Functions: Providing frameworks, tools, and libraries that incorporate the latest trends for various usage scenarios such as natural language processing, reinforcement learning, and object recognition. (always)

- Model parallelism: Support for providing model parallel network models that take advantage of the massively parallelism of the Fugaku. (always)

- Convenience: For user convenience, provide a construction environment with Spack and a massively parallel execution environment with a container environment. (1st year)

- System cooperation: Construction of a cooperative calculation environment by providing environments on various platforms centered on the R-CCS ARM cloud. (2nd year)

- Feedback: Enhance the knowledge gained from speeding up AI calculation to the current system and provide feedback to the operation / next system design. (always, long-term planning)

- Needs: Absorption of needs. By collaboration with other institutions (RIKEN AIP, other centers, research teams in other fields, etc.) as well as the Fugaku users, I will get the usage scene. (always)

- Support: Expansion of user support. Enlightenment through seminars. Provision of training networks according to the scientific usage scene. Development of usage scenes in collaboration with research teams and other centers. (always)

- Deployment: Cooperation in deployment to other HPC systems (long-term plan)

21.5 Publications

21.5.1 Articles/Journal

[1] Jaewoon Jung, Chigusa Kobayashi, Kento Kasahara, Cheng Tan, Akiyoshi Kuroda, Kazuo Minami, Shigeru Ishiduki, Tatsuo Nishiki, Hikaru Inoue, Yutaka Ishikawa, Michael Feig and Yuji Sugita, ”New parallel computing algorithm of molecular dynamics for extremely huge scale biological systems”, Journal of Computational Chemistry, Vol.42, Issue4, pp.231-241 (2021.02.05) https://doi.org/10.1002/jcc.26450 .

21.5.2 Conference Papers

[2] Chisachi Kato, Yoshinobu Yamade, Katsuhiro Nagano, Kiyoshi Kumahata, Kazuo Minami, Tatsuo Nishikawa, ”Toward Realization of Numerical Towing-Tank Tests by Wall-Resolved Large Eddy Simulation based on 32 billion grid Finite-Element”, Finalist of ACM Gordonbel Prize for SC20, (2020).

[3] Hisashi Yashiro, Koji Terasaki, Yuta Kawai, Shuhei Kudo, Takemasa Miyoshi, Toshiyuki Imamura, Kazuo Minami, Hikaru Inoue, Tatsuo Nishiki, Takayuki Saji, Masaki Satoh and Hirofumi Tomita, ”A 1024-member ensemble data assimilation with 3.5-km mesh global weather simulations”, Finalist of ACM Gordonbel Prize for SC20, (2020).

21.5.3 Posters

[4] Akiyoshi Kuroda, Toshiyuki Imamura, Ikuo Miyoshi, Kazuo Minami and Satoshi Matsuoka, ”The Evaluation of the Power Consumption of MLPerf HPC by Comparison HPL-AI”, The 3rd R-CCS International Symposium, Online, JAPAN (2021.02.15-16).

21.5.4 Invited Talks

[5] Akiyoshi Kuroda, ”What is the supercomputer Fugaku? Let’s actually use it.”, SEEDS Conference2020, Tokyo (Online), JAPAN (2020.10.11).

21.5.5 Oral Talks

[6] Kazuto Ando, Kiyoshi Kumahata, Kazuo Minami, Keiji Onishi, Li Chung-Gang, Makoto Tsubokura, and Jun Ikeda, Performance evaluation and tuning of compressible fluid solver of CUBE on Fugaku, 34th Symposium on Computational Fluid Dynamics (Web, Dec. 21-23, 2020), Paper C03-3.
[7] Kazuto Ando, Keiji Onishi, Rahul Bale, Makoto Tsubokura, Akiyoshi Kuroda, and Kazuo Minami, Distributed Learning for Three-dimensional Flow Field Mode Decomposition on Fugaku, 34th Symposium on Computational Fluid Dynamics (Web, Dec. 21-23, 2020), Paper F09-1.

21.5.6 Books

[8] Atsushi Nukariya, Kazutoshi Akao, Jin Takahashi, Naoto Fukumoto, Kentaro Kawakami, Akiyoshi Kuroda, Kazuo Minami, Kento Sato, Satoshi Matsuoka, ”HPC and AI Initiatives for Supercomputer Fugaku and Future Prospects”, Fujitsu Technical Review, Vol.2020, No.3, article 9, pp.1-6, ISSN 2435-6085 (2020.10.13).

Chapter 22

HPC Usability Development Unit

22.1 Members

Fumiyoshi Shoji (Unit Leader)

Masaaki Terai (Research & Development Scientist)

Hitoshi Murai (Senior Technical Scientist)

Jorji Nonaka (Technical Scientist)

Motohiko Matsuda (Technical Scientist)

Hiroshi Harada (Technical Scientist)

Tomohiro Kawanabe (Technical Scientist)

Chigusa Kobayashi (Technical Scientist)

Yoshifumi Nakamura (Technical Scientist)

Shigetoshi Sota (Technical Scientist)

Tsuyoshi Yamaura (Technical Scientist)

Hidetomo Kaneyama (Technical Staff I)

Megumi Takeuchi (Technical Staff I)

Chihiro Shibano (Temporary Staffing)

Kenji Ono (Senior Visiting Scientist)

Naohisa Sakamoto (Visiting Scientist)

Go Tamura (Student Trainee)

22.2 Overview of Research Activities

The HPC Usability Development Unit has covered a wide range of topics including software, service, analytics, and infrastructure, aiming to improve the overall usability of Fugaku, which was developed as the successor to the K computer. In addition, we also have some activities dedicated to one of the HPCI shared storage sites, located at the R-CCS, which is a commissioned project funded by the MEXT (Ministry of Education, Culture, Sports, Science and Technology). In FY 2020, prior to the start of the official service of Fugaku, we provided a pre-launch service period in parallel with the installation of the Fugaku equipment. Targeting the launch of the official service, we continued to push forward the already ongoing projects from the previous fiscal year and started several new projects to prepare a useful environment for the Fugaku users. Finally, they became available to the Fugaku users following the start of the official service in early March 2021.

The summary of our activities are listed below.

- **Open-Source Software (OSS) Install and Maintenance:** For better usability of the Fugaku environment, it was required to provide a set of open-source software (OSS). In FY 2020, we installed some of the requested OSS packages with Spack, which is a package management tool designed for large supercomputing centers.

- **Software Center:** We have been running the *RIKEN R-CCS Software Center* to support and promote a set of software, also called "R-CCS Software," developed at the R-CCS.

- **K Pre-Post Cloud (Data Analysis Server using Virtualization Technology):** Data analysis has become an important topic in the field of HPC/Data Centers. However, the existing data analysis servers in the K computer environment were disproportionately small compared to the K compute nodes. Nowadays, virtualization technology has sufficiently matured and can provide an ideal software environment for the users. Therefore, as a solution for accelerating data analysis without compromising the usability, we have investigated the use of OpenStack-based virtualization technology for data analysis and visualization, by deploying an experimental cloud testbed.

- **Oracle Cloud FastConnect Service:** We have developed a new secure and fast network infrastructure in the Fugaku and the HPCI shared storage, using the "Oracle Cloud FastConnect Service" under the collaboration with Oracle. This service enables the Fugaku users to transfer their data between the Fugaku/HPCI shared storage and their users' environment on the Oracle Cloud Infrastructure (OCI) as a public cloud service via Science Information NETwork (SINET).

- **HPCI Shared Storage:** The HPCI shared storage is a commissioned project, by MEXT, and is operated by two maintenance sites: RIKEN R-CCS and the University of Tokyo. To provide high availability and to avoid service interruption even in a critical situation (e.g. serious failure at one of the sites), we have operated in the data-duplication mode.

- **SFConnect: Research and Development of an Infrastructure for Collecting, Analyzing, and Utilizing Big Data through Collaboration of Large-Scale Research Facilities to Support the Society 5.0:** The purpose of this project is to establish a "Big Data Infrastructure" that enables data collection, analysis, and utilization between SPring-8/SACLA and Fugaku, as one of the world's most advanced large-scale research facilities that support Society 5.0. In FY 2020, we have collaborated with some R-CCS research teams and the RSC, to build a Data Analysis and Transfer infrastructure. It is planned to start providing this infrastructure service to the Fugaku/HPCI users in FY 2021.

- **ChOWDER (COoperative Workspace DrivER):** ChOWDER is a scalable tiled display management framework capable of displaying visualization results on high-resolution large screen area consisting of multiple tiled physical display devices. In addition, this framework has a remote collaboration feature to share the same contents on remotely distributed tiled display systems.

- **Large Data Visualization and Analysis:** HPC systems and facilities have been generating vast amounts of data in the form of simulation results as well as log data sets, and visualization has been widely recognized as an important tool for analyzing such large data sets. Since different visualization users usually have different needs and goals, we have tried to work closely with end users to co-develop the required visualization functionalities for scientific visualization (simulation results) or information visualization (log data). We have also investigated and evaluated some exisiting OSS visualization applications targeting the Fugaku users.

- **Operational Data Analysis for the Data Center Infrastructure:** Modern supercomputer system depends on various information technology (IT) components (e.g., servers, storage, and network systems) and facility equipment (e.g., power supply and cooling systems). Each system component constantly produces a huge amounts of time-series log/metric data to monitor and maintain itself. We have attempted to understand the complicated data center systems' behavior, based on evidence-based analysis with the log/metric data, to further improve our operations. The main goal of this trial is to feed the insights from the data analysis back to the real operation.

- **Workflow Management Software (WHEEL):** WHEEL is a workflow management software originally developed by the former Advanced Visualization Research Team, and has been continuously developed and improved to enable more efficient capacity computing capabilities. In FY 2020, we have enhanced the WHEEL features to improve its usability. We have also supported some trial evaluations at several research institute sites, and could verify its usefulness for the capacity computing workflows in combination with practical application software.

- **LQCD User Environment on Fugaku:** We have improved the usage environment of lattice quantum chromodynamics (LQCD) applications on Fugaku to study high energy physics.

- **Development of Massively Parallel Density Matrix Renormalization Group Method:** We are developing massively parallel density matrix renormalization group (DMRG) programs to study ground state properties and quantum dynamics of higher-dimensional quantum many-body systems.

- **Study on the Operational Impact of Hot Water Cooling:** Hot water cooling is widely accepted as an effective approach to improve the energy efficiency of modern HPC and Data Centers. However, it is also known that higher operating temperature may impact the CPU performance and power consumption. To better understand the pros and cons of the hot water cooling, we have studied the impact of the cooling water temperature on the performance and energy consumption by using a DIY evaluation system (HUD-Oden) as well as an actual HPC system (Oakforest-PACS) and its facility. The latter was done under the "Oakforest-PACS Large-Scale HPC Challenge" approved program, and with the collaboration of the JCAHPC staff.

Furthermore, we have other activities that have been conducted in collaboration with universities and research institutes, in various fields of science and technology, and what they have in common is the investigation and exploration on how to enhance the usability, and to improve the productivity of the supercomputer environment. Although we have a wide variety of missions, we can say that our common objective is to improve the usability of the Fugaku environment as well as of the HPCI shared storage. Under this main goal, we have investigated and explored new usage for Fugaku and the HPCI storage, trying to embrace not only the traditional HPC users but especially new users from various other fields. In the rest of this chapter, we will detail these aforementioned research and development projects carried out in FY 2020.

22.3 Research Results and Achievements

22.3.1 Open-Source Software (OSS) Install and Maintenance

For better usability of Fugaku, a set of open-source software (OSS) was selected to become available to the Fugaku users. In FY 2020, we installed this set of OSS packages with Spack, which is a package management tool designed for large supercomputing centers [15]. The selection was based on the requests we had received from the pilot users during the early access period of Fugaku, and most of the requested OSS packages were successfully installed and became available on Fugaku. This work was conducted in collaboration with Research Organization for Information Science and Technology (RIST). Table 22.1 shows the list of the packages available on the Fugaku compute nodes. We also worked on optimizing some of the packages for Fugaku. This work has contributed to the development of the Arm software ecosystem. From a mid- to long-term perspective, we plan to arrange an OSS-based software environment, as an alternative to the vendor-provided one, on the Fugaku.

In addition, we have a collaboration with the developer of Spack, namely, Lawrence Livermore National Laboratory, to enhance the support for Fugaku. We have also made efforts for porting the Score-P/Scalasca and TAU tools with the collaborations of JSC and the University of Oregon, respectively.

22.3.2 Software Center

We have been running the *RIKEN R-CCS Software Center*, since FY 2017, to support and promote the *R-CCS software*, which is a set of applications, libraries, programming tools, and other software developed at the R-CCS.

The major achievements of this Center, in FY 2020, are as follows:

- Deployment of some R-CCS software onto the Fugaku, which include:
  - Process-in-Process (PiP)

Table 22.1: List of packages available on the Fugaku compute nodes.

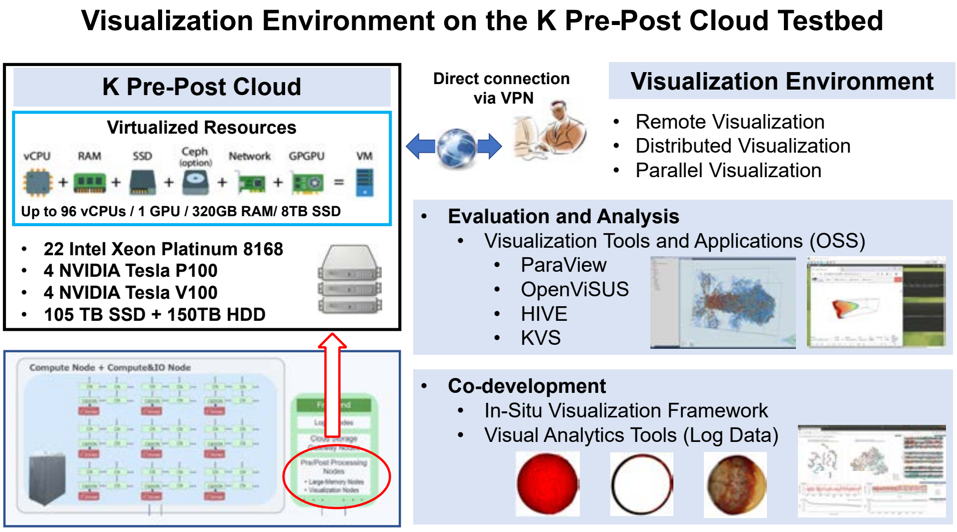
Name	Version
adios2	2.6.0
alps	2.3.0
bcftools	1.10.2
bedtools2	2.27.1
blitz	1.0.1
boost	1.74.0
bwa	0.7.17
cblas	2015-06-06
cmake	3.18.4
dssp	3.1.4
eigenexa	2.6
fftw	3.3.8
fujitsu-fftw	master
fujitsu-mpi	4.3.1
gcc	10.2.0
genesis	1.5.1
grads	2.2.1
gromacs	2021.1 / 2021
hdf5	1.10.7
htslib	1.10.2
julia	1.5.2
kokkos	3.1.01
lammps	20201029
lis	2.0.27
llvm	11.0.0
lz4	1.9.2
mapsplice2	2.2.1
metis	5.1.0
mpich-tofu	master
mptensor	0.3.0
n2p2	2.1.1
netcdf-c	4.7.4
netcdf-cxx	4.2
netcdf-cxx4	4.3.1
netcdf-fortran	4.5.3
netlib-lapack	3.8.0
netlib-scalapack	2.1.0
openblas	0.3.12
openfoam	2006
openfoam-org	8
openjdk	11.0.0-2020-01-01
parallel-netcdf	1.12.1
paraview	5.8.1
parmetis	4.0.3
petsc	3.14.1
pfapack	2014-09-17
picard	2.20.8
povray	3.7.0.8
process-in-process	2 / 3
py-dask	2.16.0
py-h5py	2.10.0
py-jupyterhub	1.0.0
py-keras	2.2.4
py-mpi4py	3.0.3
py-netcdf4	1.5.3
py-numpy	1.19.4
py-pydmd	0.3
py-pygps	1.3.5
py-pysam	0.15.2
py-scikit-learn	0.23.2
py-scipy	1.5.4
py-seaborn	0.9.0
py-xarray	0.14.0
quantum-espresso	6.5 / 6.6
r	4.0.3
raja	0.12.1
samtools	1.1
scale	5.4.3
screen	4.8.0
star	2.7.6a
tmux	3.1b
xios	2.5

  - MPICH on Tofu
  - Omni XcalableMP
  - EigenExa
  - GENESIS (GENeralized-Ensemble SImulation System)

- Arrangement of a development environment supporting CI/CD processes on Fugaku for R-CCS software (WIP).

22.3.3 K Pre-Post Cloud (Data Analysis Servers using Virtualization Technology)

The K Pre-Post Cloud was designed as a proof-of-concept environment, to enhance the pre/post data processing features including data analysis and visualization in both the K computer and Fugaku. As shown in the Fig. 22.1, this system consists of 11 compute nodes and some auxiliary servers. Each compute node has 48 physical cores with 384 GiB of RAM, and is connected to other nodes with a 25 GiB Ethernet network switch. In addition, the system has an external Ceph-based 150 TiB storage. Also, the system employed server virtualization technology using the OpenStack framework. That feature is able to install and uninstall any software on their own virtualized environment, including not only the OSS but even the operating system they want to use. Furthermore, some of the compute nodes has GPUs to accelerate the data processing capabilities.



Visualization Environment on the K Pre-Post Cloud Testbed

K Pre-Post Cloud

Virtualized Resources

Up to 96 vCPUs / 1 GPU / 320GB RAM/ 8TB SSD

- 22 Intel Xeon Platinum 8168
- 4 NVIDIA Tesla P100
- 4 NVIDIA Tesla V100
- 105 TB SSD + 150TB HDD

Direct connection via VPN

Visualization Environment

- Remote Visualization
- Distributed Visualization
- Parallel Visualization

- Evaluation and Analysis
  - Visualization Tools and Applications (OSS)
    - ParaView
    - OpenViSUS
    - HIVE
    - KVS

- Co-development
  - In-Situ Visualization Framework
  - Visual Analytics Tools (Log Data)

Figure 22.1: Overview of the K Pre-Post Cloud testbed and some usage examples.

The Fugaku development project conducted procurement and system implementation work for not only the main system but also a new pre/post environment, where the users are allowed to submit long-duration jobs using large memory as with the previous pre/post servers in the K computer. The new pre/post servers were scheduled for completion in FY 2021 after the development of the main system was completed. In FY 2020, while continuing the development for the Fugaku main system, we had started the pre-lunch service using part of the system. Therefore, to fill in the gaps between the K and Fugaku and support the migration, we opted to continue the K Pre-Post Cloud service and to provide a stable pre/post environment for the Fugaku users. Also, we had obtained the know-how and experience from our operations and fed those back into the development of the new pre/post servers.

The OpenStack framework installed in our environment will reach the end of life in FY 2021. While the K Pre-Post Cloud service provided various insights, the operations revealed issues regarding updating the system on a continuous basis due to the complexity of the configuration. Following the start of the official service of Fugaku, we also started to provide a new Fugaku pre/post environment, by using brand new servers, as a successor to the K Pre-Post Cloud service.

22.3.4 Oracle Cloud FastConnect Service

To take advantage of a public cloud service to improve the usability of Fugaku and the HPCI shared storage, we have developed a new secure and high-speed network infrastructure by using the "Oracle Cloud FastConnect Service" as shown in Fig. 22.2, under the collaboration with Oracle. This service was started in June 2020, and this infrastructure enables the Fugaku users to transfer their data, without any additional fee, between the Fugaku/HPCI shared storage and their users' environment on the Oracle Cloud Infrastructure (OCI) via Science Information NETwork (SINET) [20].

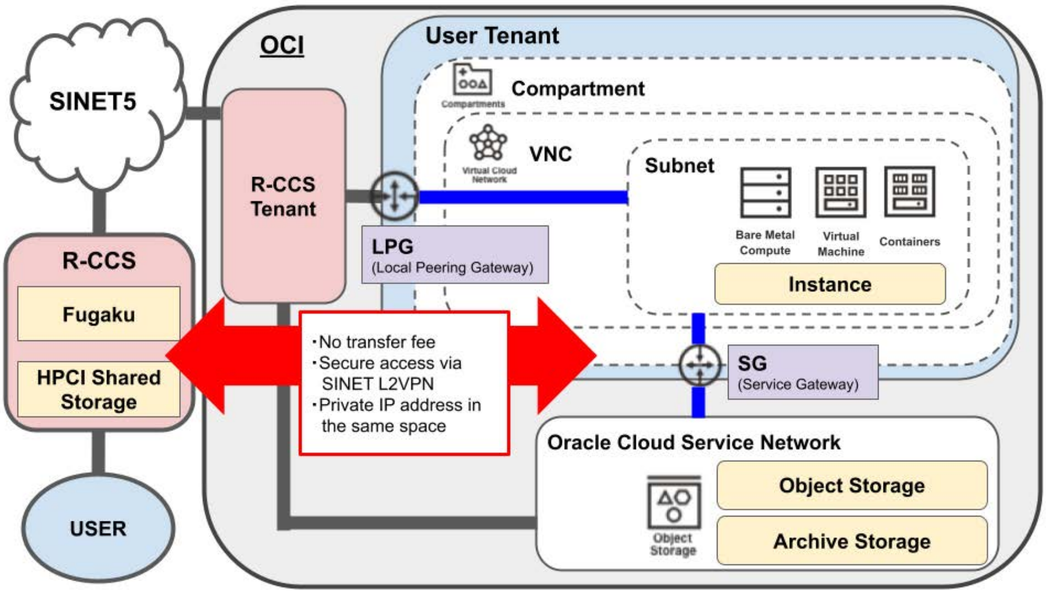


Figure 22.2: Overview of the Oracle Cloud FastConnect Service

Nowadays, various cloud services (e.g., data store, visualization, data sharing and publication) have matured enough to become commonplace. In the next decade, we expect that the HPC users will tend to utilize more and more such public cloud services as an important alternative to conduct their research efficiently. Therefore, one of the objectives for this infrastructure development was to promote the use of cloud services by the Fugaku users since it it is expected to improve the usability in the supercomputing environment. In addition, we expect to bring additional effects to the operations through the development. One of the expected effects is to reduce the cost of the cloud computing in terms of operations at the Fugaku and HPCI environment. In addition, it can be used as an engineering environment when the operators/developers attempt to consider or implement a new service.

In FY 2020, toward the launch of the official service, we have prepared an exhaustive documentation and held some study sessions to share the knowledge and know-how regarding the use of the service. Our activities included user support, organization of briefing sessions, and preparation of test environments to the candidate users. The service was mainly used as a multi-cloud test environment to address the Fugaku usage issues. Also, we held user briefings in July 2020 and December 2020 with the collaboration of Oracle staff. At these briefings, introductory hands-on session was carried out to explain how to use the cloud service, and to build a job system for visualization using the OCI Instance and Object Storage. Details of the service and briefings can be found at the Oracle Cloud FastConnect Service Website (`https://hudtech.r-ccs.riken.jp/ocisf/html_en/`).

22.3.5 HPCI Shared Storage

The operational goals of the HPCI shared storage system for FY 2020 was defined based on the operational results of the FY 2019, which had the following clear operational goals:

- 100% read-write service utilization
- No security incidents

- No data loss
- 1,000 days of continuous operation

Since the utilization rate in FY 2019 was 100%, we maintained the target of 100% utilization rate in FY 2020, and also maintained the long-term continuous operation target of 1,000 days. Figure 22.3 shows the monthly operation rate of the HPCI shared storage system in FY 2020, and we could successfully repeat the utilization rate of 100%, without any service outages during the regular operational period. Considering that the HPCI storage has been in continuous operation since October 10, 2018, thus it is scheduled to achieve the 1,000 days of continuous operation on July 7, 2021. As an additional functionality to the HPCI users, a Web-based dashboard, using Grafana, has become available for visualizing the operational service status [21] as shown in the Fig. 22.4. By accessing this dashboard, HPCI users can check the operational status of the HPCI storage and monitor the progress of data multiplexing in real time.

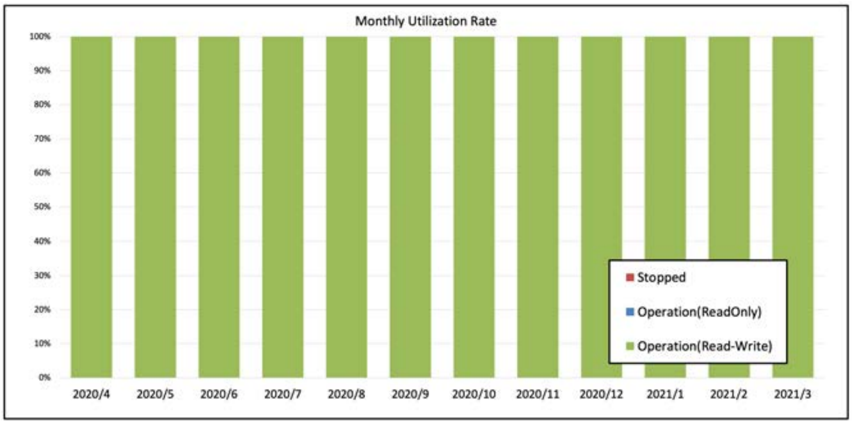
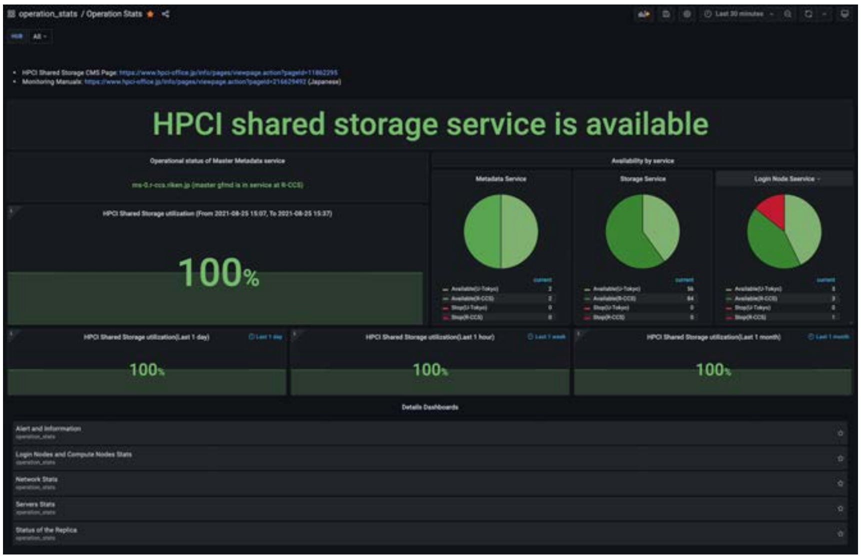


Figure 22.3: FY 2020 monthly utilization rate of the HPCI shared storage (monthly operation rate).


operation_stats / Operation Stats

- HPCI Shared Storage CMS Page: https://www.hpci-office.jp/info/pages/viewpage.action?pageId=11862295
- Monitoring Manuals: https://www.hpci-office.jp/info/pages/viewpage.action?pageId=216629492 (Japanese)

HPCI shared storage service is available

Operational status of Master Metadata service

ms-0.r-ccs.riken.jp (master gfmd is in service at R-CCS)

HPCI Shared Storage utilization (From 2021-08-25 15:07, To 2021-08-25 15:37)

100%

Availability by service

Metadata Service

	current
Available(U-Tokyo)	2
Available(R-CCS)	2
Stop(U-Tokyo)	0
Stop(R-CCS)	0

Storage Service

	current
Available(U-Tokyo)	56
Available(R-CCS)	64
Stop(U-Tokyo)	0
Stop(R-CCS)	0

Login Node Seervice

	current
Available(U-Tokyo)	3
Available(R-CCS)	3
Stop(U-Tokyo)	0
Stop(R-CCS)	1

HPCI Shared Storage utilization(Last 1 day) — Last 1 day

100%

HPCI Shared Storage utilization(Last 1 hour) — Last 1 week

100%

HPCI Shared Storage utilization(Last 1 month) — Last 1 month

100%

Details Dashboards

Alert and Information
operation_stats

Login Nodes and Compute Nodes Stats
operation_stats

Network Stats
operation_stats

Servers Stats
operation_stats

Status of the Replica
operation_stats

Figure 22.4: Grafana-based dashboard for visualizing the operational status of the HPCI shared storage.

In FY 2020, the COVID-19 pandemic has continued affecting the operation and service. In particular, some users were unable to obtain their GSI user certificates because it became difficult to verify their identity in person. Since the HPCI storage adopts GSI authentication, the GSI user certificate becomes indispensable to access the HPCI storage. To overcome this problem, we prepared another authentication mechanism of Gfarm, via shared key authentication, for those users unable to obtain their GSI user certificates. Obviously, this adoption of the shared key authentication has been previously approved by the HPCI Cooperative Service Committee. Preparing for the start of the official operation of the flagship supercomputer of the HPCI, Fugaku, we built the HPCI storage client environment on the Fugaku's login servers and cloud storage gateway. In addition, we have started to provide the HPCI storage resources to the Fugaku users since the start of the official operation.

We have also trying to contribute to the international HPC research community by expanding the Japanese HPCI resources and services, which is currently being developed and used only in Japan, to the ASEAN countries. The goal is to promote the use of HPCI computing resources, such as Fugaku, to the researchers in the ASEAN countries. In order to make the Japanese HPCI computational resources available to the ASEAN researchers, it was necessary to establish a data sharing environment. As an initial trial, Gfarm file system was employed for the data sharing between R-CCS and Singapore's A*STAR Computational Resource Centre (A*CRC), as shown in the Fig. 22.5. This Gfarm file system consists of two redundant metadata servers as well as 500 TB of storage space on both A*CRC and R-CCS sides, and the data are constantly replicated between the A*CRC and R-CCS. The Fugaku testbed at the R-CCS and login servers at the A*CRC are able to access this Gfarm file system. At the SUPERCOMPUTING ASIA 2021, held on March 2021, a practical demonstration was carried out in which the data written from the A*CRC was processed by the Fugaku testbed at the R-CCS, and the results were sent back to the A*CRC.

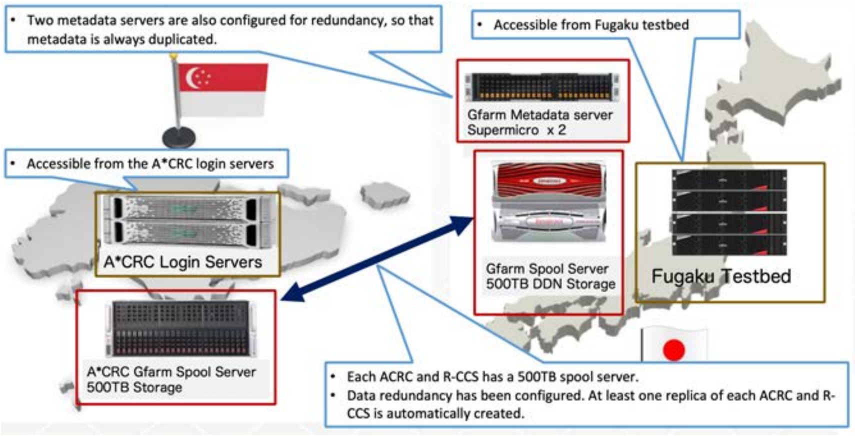

• Two metadata servers are also configured for redundancy, so that metadata is always duplicated.
• Accessible from Fugaku testbed
Gfarm Metadata server Supermicro x 2
• Accessible from the A*CRC login servers
A*CRC Login Servers
Gfarm Spool Server 500TB DDN Storage
Fugaku Testbed
A*CRC Gfarm Spool Server 500TB Storage
• Each ACRC and R-CCS has a 500TB spool server.
• Data redundancy has been configured. At least one replica of each ACRC and R-CCS is automatically created.

Figure 22.5: Overview of the constructed Gfarm file system between R-CCS and ACRC.

22.3.6 SFConnect: Research and Development of an Infrastructure for Collecting, Analyzing, and Utilizing Big Data through Collaboration of Large-Scale Research Facilities to Support the Society 5.0

22.3.6.1 Overview

The purpose of this project is to establish a "Big Data Infrastructure" that enables data collection, analysis, and utilization between SPring-8/SACLA and the supercomputer Fugaku, as one of the the world's most advanced large-scale research facilities that support Society 5.0. The SPring-8 (large synchrotron radiation facility) and SACLA (X-ray free electron laser facility) are positioned as leading-edge large-scale research facilities that support Society 5.0, and the construction of a new shared platform to enhance the maintenance capability and utilization of these facilities and their research and development is essential. In order to enhance the maintenance and utilization of these facilities, the construction of a new shared platform and its research and development are indispensable.

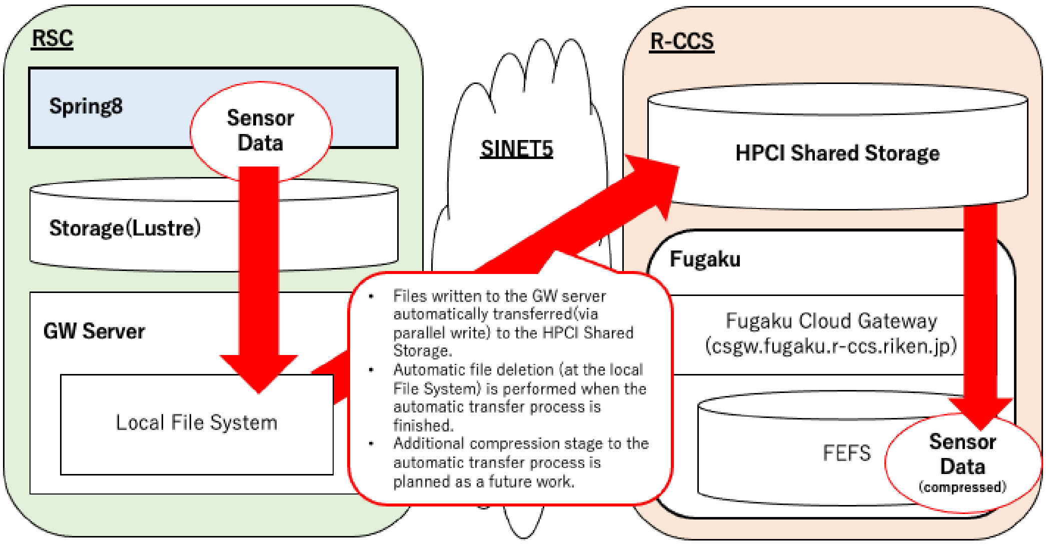
R-CCS research teams and Operation & Development units have collaborated to conduct various research and development activities to build this big data platform. (1) Data pre-processing platform: By performing data conversion and pre-processing at the hardware level using FPGAs, experimental data obtained from sensors can be efficiently stored in temporary storage. (2) Data compression and transfer infrastructure: Development of a data compression and transfer infrastructure for high-speed transfer of experimental data on storage to Fugaku. (3) Data analysis infrastructure: Development of an infrastructure (workflow tools and deep learning framework) to efficiently analyze the collected data on Fugaku. (4) Data utilization infrastructure: Development of a data utilization infrastructure to make extensive use of the collected primary data and analysis results. Finally, all of the above will be integrated into a big data infrastructure by conducting the necessary research and development for end-to-end big data processing from sensors to computers.

22.3.6.2 Achievements

In FY 2020, we developed a data analysis and transfer infrastructure with the collaborations of some R-CCS research teams and RSC teams as shown in Fig. 22.6. In this infrastructure, once a user stores data on the RSC gateway server, the gateway server automatically transfers the data in parallel to the HPCI Shared Storage. When this data transfer is completed, the data is automatically deleted from the gateway server. As a result, the user can then retrieve the data stored in the HPCI shared storage, via the cloud storage gateway (csgw.fugaku.r-ccs.riken.jp), for processing on the Fugaku. In FY 2021, We are planning to start this data transfer service, using this infrastructure, to the Fugaku/HPCI users.

Main research and development tasks include, but not limited to, following items:

- Development and test of the data analysis and transfer infrastructure;
- Development of data transfer method from the SPring-8 to Fugaku;
- Test of parallel data transfer mechanism from the RSC gateway servers to the HPCI Shared Storage;
- Preparation for shared key authentication in the HPCI Shared Storage;
- Research and development of compression methods using the ADIOS2 framework.



- Files written to the GW server automatically transferred(via parallel write) to the HPCI Shared Storage.
- Automatic file deletion (at the local File System) is performed when the automatic transfer process is finished.
- Additional compression stage to the automatic transfer process is planned as a future work.

Figure 22.6: Overview of the data analysis and transfer infrastructure between Spring-8 and Fugaku.

22.3.6.3 External collaborators

- Kento Sato (R-CCS High Performance Big Data Research Team)
- Kentaro Sano (R-CCS Processor Resarch Team)
- Masaaki Kondo (R-CCS Next Generation High Performance Architecture Research Team)
- Tomohiro Ueno (R-CCS Processor Resarch Team)
- Takaki Hatsui (RSC)
- Yasumasa Joti (RSC)

22.3.7 ChOWDER (COoperative Workspace DrivER)

In recent years, display wall technology becomes possible to build a large-scale tiled display system to create one large pixel space. That system connects multiple physical display devices onto a single PC by using high-performance graphics cards.

We are advancing the research and development of a scalable display system named ChOWDER (CO-operative Workspace DrivER). While common tiled display systems often have limitations on the maximum achievable resolution due to the hardware and its middleware, ChOWDER is a software-based system, and there is no logical upper limit regarding the maximum resolution it can provide. In addition, ChOWDER is a web-based system that uses only a web browser and provides high availability. Also, the ChOWDER's server works based on virtual pixel space called VDA (Virtual Display Area), and it enables the coexistence of tiled displays with physical display devices possessing different resolutions and aspect ratios. It is also possible to use the ChOWDER as a remote collaboration tool for displaying the same content on the tiled displays placed at multiple sites thanks to the web-based high availability functionality.

In FY 2020, we conducted research and development to display 3D Web geographic information system (GIS) data on a large-scale tiled screen using ChOWDER. To develop this framework, we used an open-source 3D Web GIS, iTowns, developed mainly by IGN in France. As a result, we succeeded in developing the distributed tiled display feature for the 3D Web GIS data on the ChOWDER system [25]. To verify that the newly developed functionality works, we overlaid user-defined data (polygons or point clouds) onto the GIS data, and obtained some verification examples by rendering atmospheric clouds taken by the meteorological satellite Himawari, provided by NICT, which was converted into three-dimensional geometric point cloud data on the ChOWDER system. This functionality is expected to be useful in the field of earth science.

22.3.8 Large Data Visualization and Analysis

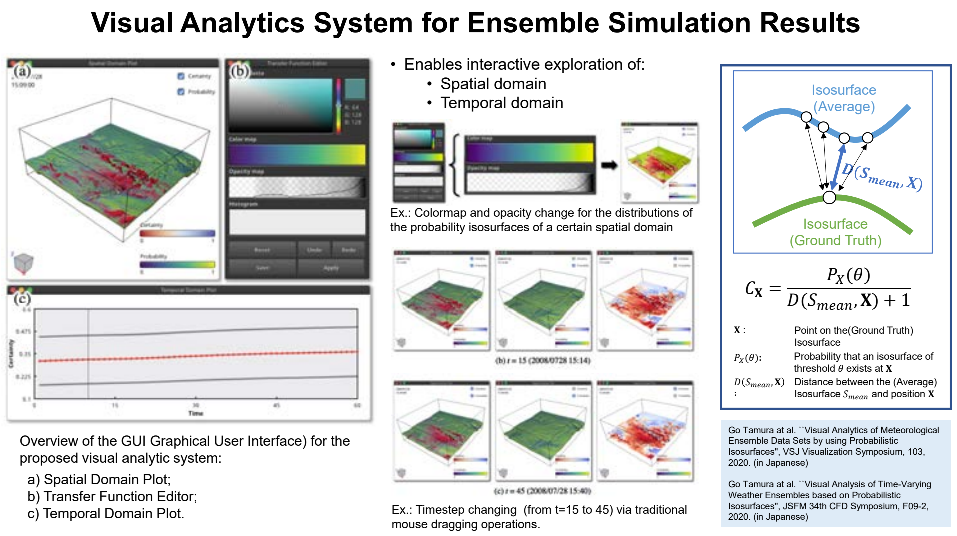
The K computer environment has imposed some difficulties for deploying traditional tools and applications for large data visualization and analysis, and we had some lessons learned from the visualization software development for the K computer [3]. Focusing the Fugaku environment, with attached pre-post processing nodes, we had utilized the *K Pre-Post Cloud Testbed*, shown in Fig. 22.1, for analyzing and evaluating existing visualization applications for *Scientific Visualization* (Simulation Data) and *Information Visualization* (Log Data) [23]. Among them, we can cite the Heterogeneously Integrated Visualization Environment (HIVE) framework [1], which belonged to a small group of applications capable of running on the K computer and its commercial versions (Fujitsu FX-10 and FX-100). In addition to the general-purpose visualization applications targeting the future Fugaku users, we have also worked on more specific visualization tools for *Scientific Visualization* and *Information Visualization* trying to meet the specific needs of the end users as detailed in the next subsections. Most of the developments have been done with the members of a visualization research group from Kobe University, which are also assigned as visiting researchers and student trainees of this Unit.

22.3.8.1 Scientific Visualization

We have worked on a visualization tool for analyzing ensemble weather simulation results, in collaboration with an internal computational climate scientist. We have focused on the existing "Probabilistic Isosurface" technique as an approach for efficiently analyze the spatio-temporal certainty distribution of the information in the ensemble space defined by meteorological ensemble simulation results in comparison with observational data. We developed a prototype visual analysis tool, shown in Fig. 22.7, capable of interactively visualize and explore the statistical certainty distributions within the ensemble domain results [16, 17, 24]. We have also worked with external domain experts, in the Computational Fluid Dynamics (CFD) field, for co-developing visualization-oriented processing modules for in-situ visualization. We have focused on methods for selecting important timesteps and viewpoints to minimize the turnaround visual analysis time and accelerate knowledge acquisition [18]. This work has been conducted as a part of the development of a "Smart In-Situ Visualization" framework supported in part by a KAKENHI Grant.

22.3.8.2 Information Visualization

We have worked, with a visualization research group from UCDavis (University of California, Davis), on visual analytics approaches for large time-varying and multi-variate data sets, focusing on the log data generated from HPC systems and facilities. We have participated in the co-development of visual analytics tools based on Dimensionality Reduction (DR) [2, 19] and Functional Data Analysis (FDA) [26]. One of the DR-based



Visual Analytics System for Ensemble Simulation Results

- Enables interactive exploration of:
  - Spatial domain
  - Temporal domain

Ex.: Colormap and opacity change for the distributions of the probability isosurfaces of a certain spatial domain

Ex.: Timestep changing (from t=15 to 45) via traditional mouse dragging operations.

Overview of the GUI Graphical User Interface) for the proposed visual analytic system:

a) Spatial Domain Plot;
b) Transfer Function Editor;
c) Temporal Domain Plot.

$$C_{\mathbf{X}} = \frac{P_X(\theta)}{D(S_{mean}, \mathbf{X}) + 1}$$

$\mathbf{X}$: Point on the(Ground Truth) Isosurface

$P_X(\theta)$: Probability that an isosurface of threshold θ exists at $\mathbf{X}$

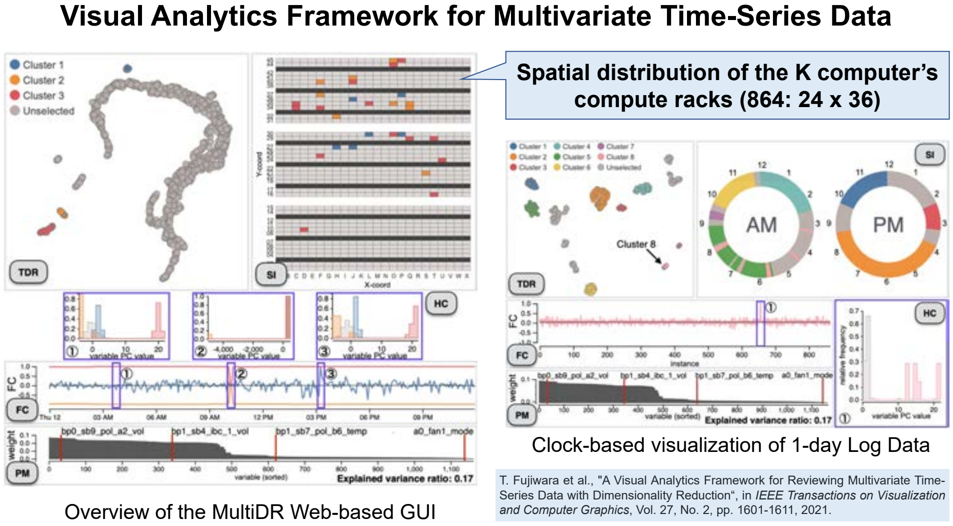
$D(S_{mean}, \mathbf{X})$: Distance between the (Average) Isosurface S_{mean} and position $\mathbf{X}$

Go Tamura at al. ``Visual Analytics of Meteorological Ensemble Data Sets by using Probabilistic Isosurfaces", VSJ Visualization Symposium, 103, 2020. (in Japanese)

Go Tamura at al. ``Visual Analysis of Time-Varying Weather Ensembles based on Probabilistic Isosurfaces", JSFM 34th CFD Symposium, F09-2, 2020. (in Japanese)

Figure 22.7: Overview of the visualization analytics tool for ensemble weather simulation results.

tools, named MultiDR, is shown in Fig. 22.8. For the development, we have used the sensor data sets from the K computer's compute racks (total of 864 racks), consisting of 1,163 measurements (e.g., CPU temperatures, circuit voltages, and cooling fan speeds) at every 5 minutes (1,440 sampling data per day) for each compute rack.



Visual Analytics Framework for Multivariate Time-Series Data

Spatial distribution of the K computer's compute racks (864: 24 x 36)

Overview of the MultiDR Web-based GUI

Clock-based visualization of 1-day Log Data

T. Fujiwara et al., "A Visual Analytics Framework for Reviewing Multivariate Time-Series Data with Dimensionality Reduction“, in *IEEE Transactions on Visualization and Computer Graphics*, Vol. 27, No. 2, pp. 1601-1611, 2021.

Figure 22.8: Overview of the visualization analytics tool for time-varying multivariate data sets.

22.3.9 Operational Data Analysis for the Data Center Infrastructure

Although the official service period of the K computer ended in 2019, part of the facility equipment has been enhanced and others are being reused as part of the infrastructure of the Fugaku successor system. Therefore, the understanding of the historical facility behavior of the K computer becomes highly valuable to ensure stable and energy-efficient operations during the next operational life cycle.

The K computer used two different types electrical energy sources. It used electricity purchased from a public utility company, and also used electricity generated by gas turbine power generators on premises. To evaluate the energy efficiency of the entire center, we used a modified power usage effectiveness (PUE) metric that considers different forms of energy sources, including other than that purchased from public utility companies, and the calculated values for this proposed metric, during the service period, was presented in the paper [6]. To analyze the effect of operational impact on the PUE, we used both the facility metrics and the server operation metrics extracted from the K computer logs. Applying the metric data to three case studies revealed that some maintenance operations can degrade the PUE. In particular, compared to emergency operations, annual maintenance operations tend to affect more the PUE metric. Finally, we presented some preliminary results that clearly indicate an operational issue regarding the gas co-generation system.

22.3.10 Workflow Management Software (WHEEL)

To easily handle many computational tasks on the HPC resources, we have developed a scientific workflow management system, named WHEEL (Workflow in Hierarchical distributEd parallEL) [27], jointly with Kyusyu University. WHEEL supports traditional HPC simulation workflow consisting of pre-processing, simulation, and post-processing stages, and can work on various HPC computing resources including not only the Fugaku but also other computing resources. In addition, since this system is a Web-based application written in JavaScript, it is able to run on common Web browsers available for Windows, macOS, and Linux. WHEEL employs ordinary SSH and login shell to communicate with remote computers when it performs job submitting, monitoring, and file transfer. Therefore, the additional middleware is not required on the remote computers. In addition, WHEEL has the iteration/conditional branch functionality (e.g., For, Foreach, While, and If), and a parameter study functionality that allows multiple simulations simultaneously with the given parameter range specified by the user. By employing both features, the users can construct more complex workflows.

In FY 2020, WHEEL has been enhanced to make it more convenient to be used on the Fugaku. Specifically, we have added new workflow component functions for Step Job and Bulk Job, which are provided by the job scheduler of Fugaku. Step Job is a function that controls the execution of subsequent jobs according to the completion status of the preceding job. By using this function, WHEEL does not need to interpret the results of the preceding job and make conditional branching, which contributes to shortening the execution time of the entire workflow process. Bulk Job is a function to submit the same type of job execution with different parameters in bulk. The same kind of processing can be performed by the Parameter Study function originally provided by WHEEL, however, by supporting Bulk Job, WHEEL does not need to manage the status of each job. In other words, it is no longer necessary to periodically inquire the status of each job to the job scheduler, which contributes to reducing the load on the job scheduler.

As for the use cases, the structural relaxation calculation of inorganic materials by VASP has been performed using the supercomputer ITO at Kyushu University, from FY 2019, and now with a more efficient workflow structure. The multi-objective optimization problem of rotating machines at the University of Tokyo has been carried out by changing the target computer to Fugaku, and since the use of WHEEL has reduced the burden on the analyst, two methods, Large Eddy Simulation (LES) and Reynolds Averaged Navier-Stokes (RANS), are planned to be run using the same WHEEL workflow for comparative analysis.

22.3.11 LQCD User Environment on Fugaku

We have developed "Lattice quantum chromodynamics simulation library for Fugaku and computers with wide SIMD" (QWS) and published it on GitHub repository [22]. In the benchmark test of QWS using the entire Fugaku system, 102 PFLOPS (single precision) and 10% execution efficiency (single-precision peak ratio) were achieved, proving that QWS can achieve high performance on Fugaku. It is the first lattice QCD quark solver in the world to record a performance of over 100 PFLOPS. This is also more than five times the performance of about 20 PFLOPS (2018 Gordon Bell finalist) measured on Sierra at Lawrence Livermore National Laboratory in 2018. Software development related to LQCD will continue to be carried out by the Software Development Technology Unit. The optimization method of the lattice QCD simulation code for A64FX was presented at the second meeting for application code tuning on A64FX computer systems. We contributed to the improvement of the tuning technology of A64FX users including Fugaku and concretely showed the tuning method and the reduction of execution time, and improved the motivation to use Fugaku.

We discovered problems of using Japan Lattice Data Grid (JLDG), which is a platform to share lattice QCD data in Japan (like HPCI Shared Storage), from Fugaku, and solved them. The storage capacity of JLDG has been very tight for the past few years. We have procured a file server (360 TB) at the R-CCS site. In addition, the existing 600 TB worth of storage was integrated into JLDG.

22.3.12 Development of Massively Parallel Density Matrix Renormalization Group Method

The degree of freedom of quantum many-body systems is exponentially increasing with increasing the system size. The density matrix renormalization group (DMRG) method is known as one of the most efficient schemes for one-dimensional quantum many-body systems. For the higher-dimensional systems, the DMRG method requires large computational resources to perform accurate calculations. The development of computer science enables us to perform the DMRG calculations on two-dimensional systems. We are developing a massively parallel DMRG method to perform not only ground state calculations, but also quantum dynamics simulations on huge scale computers, such as the supercomputer Fugaku.

Our developed massively parallel DMRG programs have fulfilled quite high efficiency, more than 70% for the peak performance of the K-computer. In FY 2020, we optimized our developed DMRG programs for the Fugaku. We have confirmed that our developed DMRG programs shows high efficiency on the Fugaku. Also, to improve convenience for the users of our developed DMRG programs, we integrate our developed DMRG programs into one package as "qNET." Using the qNET, we can perform ground state calculations, excitation dynamics calculations, and time-dependent simulations on many kinds of quantum many-body systems. The qNET will be open for the researchers of quantum many-body systems and relating research areas.

22.3.13 Study on the Operational Impact of Hot Water Cooling

The R-CCS water cooling facility was designed to provide chilled water, of around 15^oC, to remove the heat from the main components of the K computer's compute racks, and this approach will also continue on the Fugaku. Modern HPC and Data Centers have been employed the *Hot Water Cooling*, which was emerged as an effective method for improving the energy efficiency by minimizing, or even eliminating, the use of power hungry chillers. However, from the operational point of view, it becomes important to also understand the possible cons, such as increase in the power consumption and degradation of computational performance, brought by the higher operating temperature. For this purpose, we set up a Do It Yourself (DIY) hot water cooling evaluation environment, named *HUD-Oden*, based on a commercial circulating water bath (AS ONE NEXAS MCX-450), as shown in Fig. 22.9, for evaluating the power consumption and computational performance within the ASHRAE (American Society of Heating, Refrigerating and Air-Conditioning Engineers) Class W4 ($2^oC <$ Temp. $< 45^oC$) and W5 (Temp. $> 45^oC$) temperature range [5]. We could also evaluate on an actual HPC system (Oakforest-PACS) and its facility (Joint Center for Advanced High Performance Computing [JCAHPC] facility) under the "Oakforest-PACS Large-Scale HPC Challenge" approved program with the collaboration of the JCAHPC staff. By using lower (9^oC) and higher (18^oC) cooling water temperature other than the regular operational temperature (12^oC), we could observe considerable gain in the energy consumption on the HPC facility side [22]. On the other hand, we could also observed an increase in the number of nodes suffering from performance degradation on the HPC system side when using higher cooling water temperature [4].

22.3.14 Other Activities

In addition to those activities detailed in the previous subsections, we participated in a wide range of meetings and gatherings [7,8,9,10,11,12,13,14] related to the Fugaku operations and utilization, and have also contributed to the following projects:

- With the increase in storage capacity in the exascale era, file transfer from a user's local site to Fugaku takes much time than it used to be. We started to arrange the file transfer mechanism and infrastructure in the Fugaku environment.

- To share knowledge and actual usage regarding cutting-edge cloud resources in Bioinformatics and HPC, we started to collaborate with the Laboratory for Bioinformatics Research at the RIKEN Center for Biosystems Dynamics Research (BDR)

22.4 Schedule and Future Plan

In FY 2020, we started the early access program for the use of Fugaku, in April 2020, and the official service was started in March 2021, after the completion of the installation and commissioning. Fugaku has achieved the No.1 position in four major benchmark rankings for three times in a row, and has already been used to

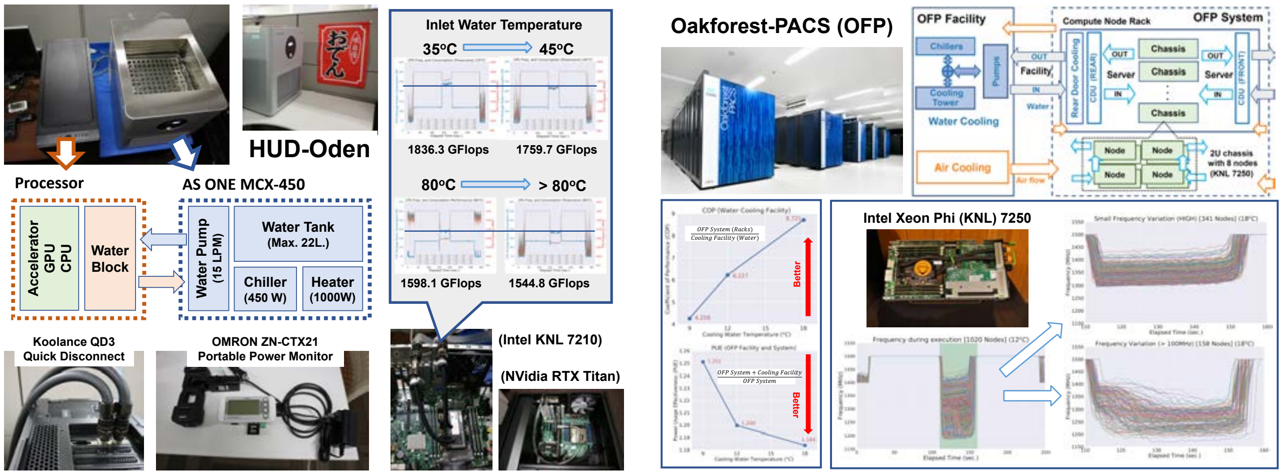

HUD-Oden
おでん
Processor
AS ONE MCX-450
Accelerator
GPU
CPU
Water Block
Water Pump (15 LPM)
Water Tank (Max. 22L.)
Chiller (450 W)
Heater (1000W)
Koolance QD3 Quick Disconnect
OMRON ZN-CTX21 Portable Power Monitor
Inlet Water Temperature
35°C
45°C
1836.3 GFlops
1759.7 GFlops
80°C
> 80°C
1598.1 GFlops
1544.8 GFlops
(Intel KNL 7210)
(NVidia RTX Titan)
Oakforest-PACS (OFP)
OFP Facility
Chillers
Cooling Tower
Pumps
Water Cooling
Air Cooling
OUT
Facility
IN
Water
Air flow
Compute Node Rack
OFP System
Rear Door Cooling
CDU (REAR)
CDU (FRONT)
Server
Chassis
Node
2U chassis with 8 nodes (KNL 7250)
COP (Water Cooling Facility)
OFP System (Racks) / Cooling Facility (Water)
Coefficient of Performance (COP)
Cooling Water Temperature (°C)
Better
PUE (OFP Facility and System)
OFP System + Cooling Facility / OFP System
Power Usage Effectiveness (PUE)
Intel Xeon Phi (KNL) 7250
Frequency during execution [1020 Nodes] (12°C)
Small Frequency Variation (HIGH) [341 Nodes] (18°C)
Frequency Variation (> 100MHz) [158 Nodes] (18°C)
Frequency (MHz)
Elapsed Time (sec.)

Figure 22.9: Analysis on the impact of the water cooling temperature to the computational performance and energy consumption using a DIY apparatus (HUD-Oden) and actual HPC system and its facility (JCAHPC Oakforest-PACS).

obtain many remarkable scientific results, including those fighting against COVID-19. For Fugaku to continue serving as an essential computational platform for many scientists and engineers, we must continuously update the operation and service of the Fugaku and its infrastructure to remain state-of-the-art. Our unit will continue developing and improving the provided services, especially those related to data storage systems, high-speed data transfer, scientific workflow tools, and visualization environments.

22.5 Publications

22.5.1 Articles

[1] Kenji Ono, Jorji Nonaka, Tomohiro Kawanabe, Masahiro Fujita, Kentaro Oku, Kazuma Hatta, "HIVE: A Cross-Platform, Modular Visualization Framework for Large-Scale Data Sets", Elsevier Future Generation Computer Systems, Vol. 112, pp. 875–883, 2020.

[2] Takanori Fujiwara, Shilpika, Naohisa Sakamoto, Jorji Nonaka, Keiji Yamamoto, Kwan-Liu Ma, "A Visual Analytics Framework for Reviewing Multivariate Time-Series Data with Dimensionality Reduction", IEEE Transactions on Visualization and Computer Graphics, Vol. 27, No. 2, pp. 1601–1611, 2021.

[3] Jorji Nonaka, Naohisa Sakamoto, "Lessons Learned from Large Data Visualization Software Development for the K computer", VisGap - The Gap between Visualization Research and Visualization Software (VisGap 2020), pp. 77–81, 2020.

[4] Jorji Nonaka, Toshihiro Hanawa, Fumiyoshi Shoji, "Analysis of Cooling Water Temperature Impact on Computing Performance and Energy Consumption", IEEE International Conference on Cluster Computing (CLUSTER), pp. 169–175, 2020.

[5] Jorji Nonaka, Fumiyoshi Shoji, "HUD-Oden: A Practical Evaluation Environment for Analyzing Hot-Water Cooled Processors", IEEE International Conference on Cluster Computing (CLUSTER), pp. 494–498, 2020.

[6] Masaaki Terai, Fumiyoshi Shoji, Toshiyuki Tsukamoto, Yukihiro Yamochi, "A Study of Operational Impact on Power Usage Effectiveness using Facility Metrics and Server Operation Logs in the K Computer", IEEE International Conference on Cluster Computing (CLUSTER), pp. 14–17, 2020.

22.5.2 Invited Talks

[7] Fumiyoshi Shoji, "New services and operation on Fugaku", 3rd R-CCS International Symposium, 2020.

[8] Fumiyoshi Shoji, "Facility and its operation for large scale supercomputer", R-CCS Cafe #195 part3, 2020.

[9] 庄司 文由, "「京」から「富岳」へ～運用とサービスの観点から～", 第23回PSEワークショップ, 2020.

[10] 庄司 文由, "「京」から「富岳」へ", 帝塚山大学付属中学校・高校向け講演会, 2020.

[11] 庄司 文由, “「富岳」における運用の考え方について”, 第九回アプリケーションソフトウェア利用環境整備 アドバイザリ WG 会合, 2020.

[12] 庄司 文由, “「富岳」の利用について ”, 第7回「京」を中核とするHPCIシステム利用研究課題 成果報告会, 2020.

[13] 庄司 文由, “「富岳」のプリポスト環境について”, スーパーコンピューティング技術産業応用協議会　広報・提言ＷＧ, 2020.

[14] Fumiyoshi Shoji, “Early Operation Experience on Fugaku”, 11th Energy Efficient HPC Working Group meeting Session 3: ARM Operational Experiences, 2020.

22.5.3 Oral Talks

[15] Hitoshi Murai, “Open-source Software on the supercomputer Fugaku”, OpenCAE and FrontISTR Joint Symposium 2020, 2020. (in Japanese)

[16] 田村剛，坂本 尚久，前島 康光, 野中 丈士, “確率的等値面を使った気象アンサンブルデータ向け視覚的分析”，第48回可視化情報シンポジウム, 2020.

[17] 田村剛，坂本 尚久，前島 康光, 野中 丈士, “確率的等値面技術をもとにした時系列気象アンサンブルデータ向け視覚的分析” ，第34回数値流体力学シンポジウム, 2020.

[18] 山岡 義明，坂本 尚久，野中 丈士，吉永 司，野崎 一徳，“In-situ 可視化向け適応的視点選択”，第34回数値流体力学シンポジウム, 2020.

[19] 藤田啓二郎，坂本尚久，野中丈士, “次元削減技術を用いた大規模ログデータ向け視覚的分析手法の検討”，核融合科学研究所（NIFS）先進的描画技術を用いた可視化情報の研究会, 2020.

[20] 金山秀智, 山本啓二，庄司文由, “SINET5を利用したスーパーコンピュータ富岳とクラウド間連携の試み”, 先端ネットワーク利用研究に関するワークショップ「ADVNET2020」, 2020.

[21] 金山秀智, 原田浩, 芝野千尋, 小瀬田勇, “Prometheus ではじめる Gfarm サービス監視”, Gfarmシンポジウム, 2020.

22.5.4 Posters

[22] Jorji Nonaka, Fumiyoshi Shoji, Toshihiro Hanawa, “Analysis on the Impact of the Cooling Water Temperature on the HPC System and Facility - Case Study: Oakforest-PACS System and Facility”, ISC 2020 Research Poster, 2020.

[23] Jorji Nonaka, Naohisa Sakamoto, Go Tamura, Masaaki Terai, “Large Data Visualization Environment on the K Pre-Post Cloud Testbed”, The 3rd R-CCS International Symposium, 2020.

[24] Go Tamura, Naohisa Sakamoto, Yasumitsu Maejima, Jorji Nonaka, “Probabilistic Isosurface based Visual Analytics System for Time Varying Ensemble Weather Simulation Data”, The 3rd R-CCS international Symposium, 2020.

[25] Tomohiro Kawanabe, Kazuma Hatta, Kenji Ono, “ChOWDER: A New Approach for Viewing 3D Web GIS on Ultra-High-Resolution Scalable Display”, IEEE International Conference on Cluster Computing (CLUSTER), pp.14–17, 2020.

22.5.5 Others

[26] Fnu Shilpika, Takanori Fujiwara, Naohisa Sakamoto, Jorji Nonaka, Kwan-Liu Ma, “A Visual Analytics Approach to Monitor Time-Series Data with Incremental and Progressive Functional Data Analysis,” arXiv:2011.13079 (`https://arxiv.org/abs/2011.13079v1`), 2020.

22.5.6 Software

[27] WHEEL: Scientific Workflow Management System. (`https://gitlab.com/aicshud/WHEEL`) (Currently managed as a private repository).

[28] RIKEN-LQCD: Lattice Quantum Chromodynamics Simulation Library for Fugaku and Computers with Wide SIMD (QWS) (`https://github.com/RIKEN-LQCD/qws`).

Chapter 23

Advanced Operation Technologies Unit

23.1 Members

Keiji Yamamoto (Unit Leader)

Masaaki Terai (Research & Development Scientist)

Shinichi Miura (Technical Scientist)

Hiroshi Harada (Technical Scientist)

Akihiro Nomura (Technical Staff)

Hidetomo Kaneyama (Technical Staff)

Mitsuo Iwamoto (Technical Staff)

23.2 Overview of Research Activities

The Advanced Operation Technologies Unit is engaged in research and development for advanced operation of the entire data center, including the supercomputer Fugaku and infrastructure such as power supplies, cooling facilities, and networks. In particular, we will develop a platform system for analyzing enormous operational data, which is also big data, improve operations based on knowledge obtained by the data analysis, research the new use of supercomputers such as virtualization and cloud computing. We will also collaborate with domestic and international HPC research institutions to research future supercomputers and data centers.

The summaries of our activities for FY2020 are listed below.

- **Operational Data Analytics Infrastructure:** Operational data analytics for Fugaku is an important activity for optimizing HPC operations and monitoring, alerting, failure prediction, and anomaly detection. We implemented a data collection infrastructure for Fugaku's log and metrics and visualized those data.

- **Fugaku Cloud Platform:** We called for collaborative research project proposals for a trial cloud-like use service using the computational resources of the supercomputer Fugaku. Seven collaborative research project proposals were started.

23.3 Research Results and Achievements

23.3.1 Operational Data Analytics Infrastructure

Operational data is very important. We need to operate the supercomputer based on operational data. Modern supercomputers are composed of many nodes, and the operational data is enormous. The supercomputer Fugaku has about 160 thousand compute nodes, and the logs outputted from all nodes may be several hundred gigabytes per day. We used these logs to identify the root cause when a failure occurs. If a batch job doesn't work correctly, we look at the file system metrics and network statistics in addition to the logs to investigate the root cause.

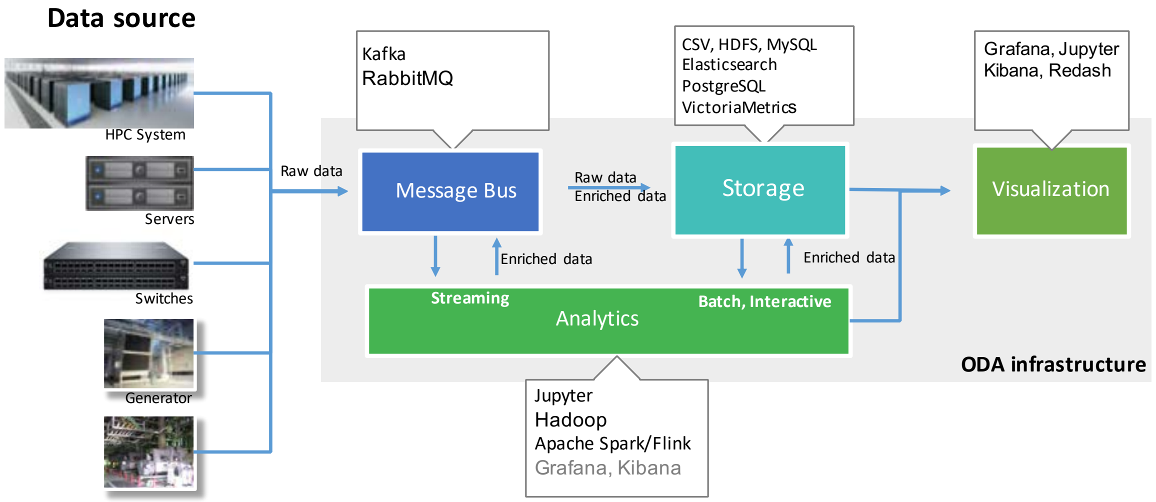


Figure 23.1: Operational Data Analytics Infrastructure

If we have a lot of standard operation data, we may detect anomaly data. Operational Data Analytics (ODA) systems collect and analyze monitoring data from HPC systems and their supporting infrastructure. A Large supercomputing center is consists of many compute nodes, servers, switches, and cooling systems. Each system component produces a huge number of time-series logs and metric data. We have built the ODA infrastructure from the K computer operation to collect these operational data through trial and error. In FY2020, we built a new ODA infrastructure for Fugaku.

Figure 23.1 shows outline of our ODA infrastructure. The main components are divided into four categories: Message bus, Storage, Analytics, and Visualization. The message bus component receives a large amount of data from a data source. The analytics component analyzes or transforms data from message bus or storage. The storage component includes simple POSIX file systems, database systems, and object storage. The visualization component visualizes raw or enriched data from storage.

Figure 23.2 shows data source of our HPC data center. We have two types of data sources. The HPC platform comprises information devices such as supercomputers, subsystems, storage systems, and network switches. Linux or SNMP daemon is running on these systems, and it is possible to collect logs and metrics without using any particular method. The data center infrastructure is a High-voltage substation, chiller, heat exchanger, air handler, and pumps for operating supercomputers. These infrastructures have sensors and are controlled by the building management system.

Figure 23.2 shows the website that visualizes the operational status for Fugaku. This site shows users the number of running jobs and the load on the parallel file system. We will continue to provide useful information for users in the future.

23.3.2 Fugaku Cloud Platform

While the K computer has been used as a science and technology infrastructure of the world's highest level and produced many excellent results, Fugaku is intended for producing maximum achievements through the expansion of its utilization in view of Society 5.0. For this, it is especially important to improve the convenience of its use, and the issue is thus one of the main focuses in its development. As part of this process, we will carry out a trial operation of Fugaku in the cloud-like use form in about FY2020 for verifying its effectiveness for the full-scale operation planned to start in FY2021. The results of this verification will be reflected in establishing a management system for the full-scale operation.

The cloud-like use here means a form of using Fugaku in which its computational resources are provided to its users as a computational environment through service providers. It is expected that the collaboration between RIKEN R-CCS, which has operation know-how of high-performance computers (HPC), and service providers having cloud technology and application service technology will improve the utility of Fugaku and expand its usership. At the same time, we intend to contribute to the large development of the IT industry by widely introducing into the cloud infrastructure not only the hardware and software technology of Fugaku, which is the world's leading-edge HPC system, but also its related technologies, such as foremost use services

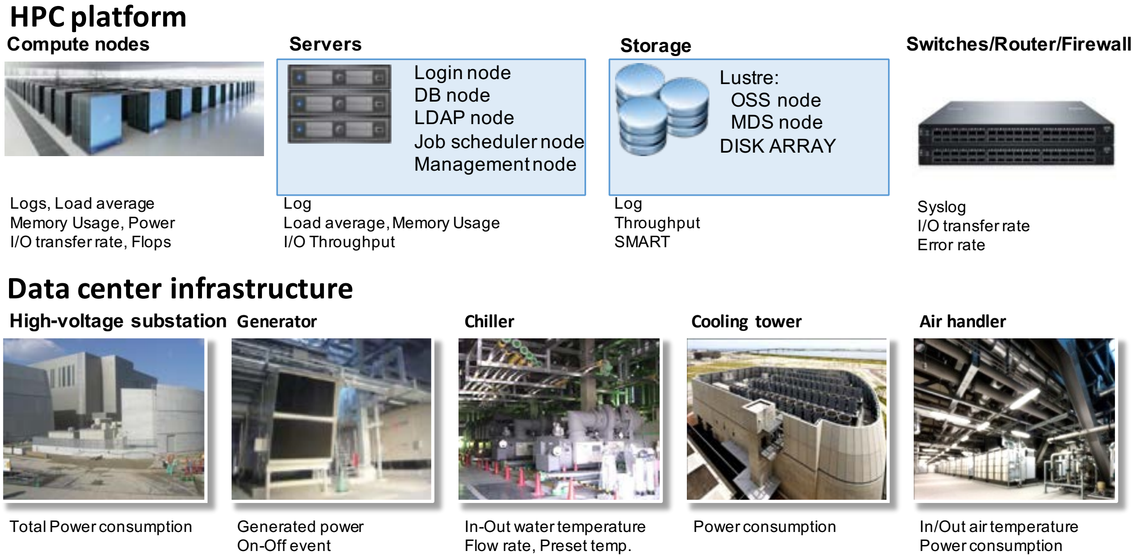


Figure 23.2: Data source of our HPC data center

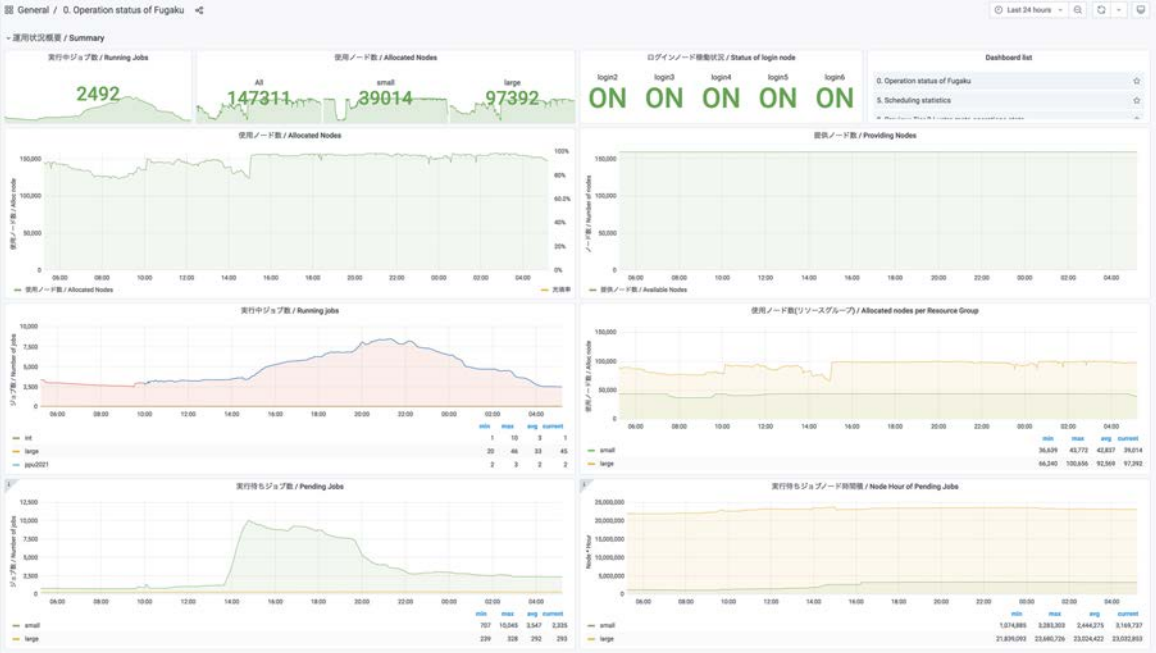

General / 0. Operation status of Fugaku
Last 24 hours
運用状況概要 / Summary
実行中ジョブ数 / Running Jobs
2492
使用ノード数 / Allocated Nodes
All
147311
small
39014
large
97392
ログインノード稼働状況 / Status of login node
login2
ON
login3
ON
login4
ON
login5
ON
login6
ON
Dashboard list
0. Operation status of Fugaku
5. Scheduling statistics
使用ノード数 / Allocated Nodes
提供ノード数 / Providing Nodes
実行中ジョブ数 / Running jobs
min max avg current
int 1 10 3 1
large 20 46 33 45
jppu2021 2 3 2 2
使用ノード数(リソースグループ) / Allocated nodes per Resource Group
min max avg current
small 36,639 43,772 42,837 39,014
large 66,240 100,656 92,569 97,392
実行待ちジョブ数 / Pending Jobs
min max avg current
small 707 10,045 3,547 2,335
large 239 328 292 293
実行待ちジョブノード時間積 / Node Hour of Pending Jobs
min max avg current
small 1,074,885 3,283,303 2,444,275 3,169,737
large 21,839,093 23,680,726 23,024,422 23,032,853

Figure 23.3: Operational status for Fugaku (https://status.fugaku.r-ccs.riken.jp/)

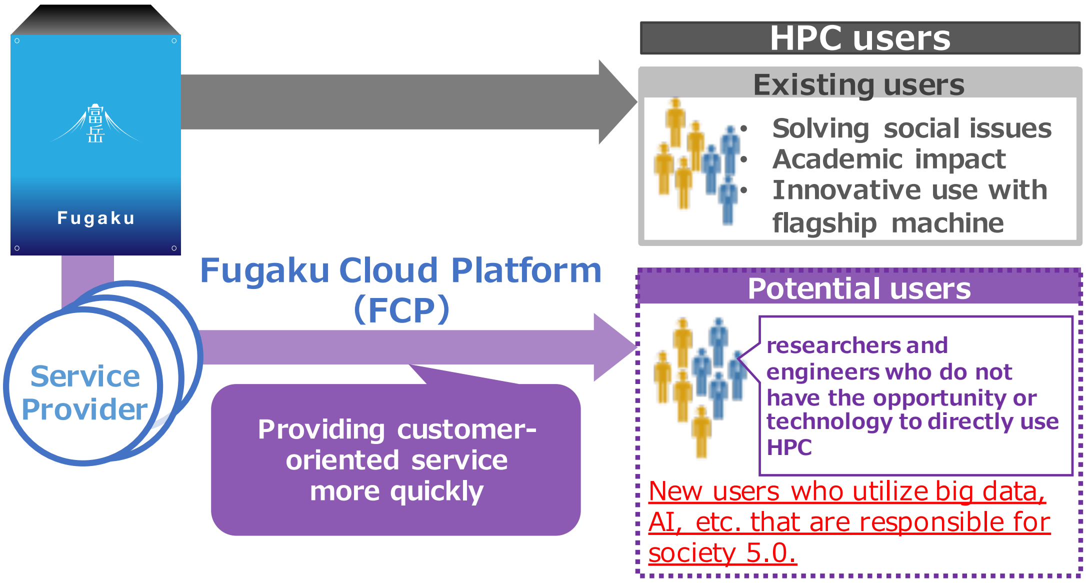

Fugaku
富岳
HPC users
Existing users
• Solving social issues
• Academic impact
• Innovative use with flagship machine
Fugaku Cloud Platform (FCP)
Service Provider
Providing customer-oriented service more quickly
Potential users
researchers and engineers who do not have the opportunity or technology to directly use HPC
New users who utilize big data, AI, etc. that are responsible for society 5.0.

Figure 23.4: Fugaku Cloud Platform

and operation technologies to be developed through its cloud-like use.

Figure 23.4 shows outline of Fugaku Cloud Platform. Whereas traditional HPC users carry out social and academic impact issues, in cloud-like use, Fugaku's computing resources are provided to potential users of researchers and engineers who have not had the opportunity or technology to use HPC through the mediation of HPC service providers directly.

This initiative will be implemented on a trial basis for two years, from April 2020 to March 2022. We will try a wide range of methods to provide computational resources to end-users and evaluate each effect as quantitatively as possible during this period. We solicited proposals for collaborative research projects on cloud-like use in FY2019 and started seven collaborative research projects from FY2020. In collaborative research projects, the services provided by each service provider can be broadly classified into the following two types.

Support and usage consulting service (support service)

Application usage service (SaaS)

The support service provides information on how to use Fugaku, installing and optimizing the application on Fugaku for users who have never used a supercomputer. In FY2020, we evaluated the application process and examined redundant parts and online application processes. In FY2021, we will implement online application processes and issue user accounts more quickly.

In SaaS (Software as a Service), a service provider implements and optimizes an application to Fugaku, and the user uses it through interfaces provided by the service provider. Users can use the applications provided by the service provider without any knowledge of Fugaku or HPC. In FY2020, we examined how to port applications to Fugaku, issue user accounts, authentication methods, billing management methods. In addition, we have implemented a web API that can operate Fugaku remotely. We plan to consider, implement and evaluate the introduction of Kubernetes, which is a container platform.

23.4 Schedule and Future Plan

We will continue to improve the advanced operation of our supercomputer. In FY2021, we will deploy the implemented service of Fugaku Cloud Platform to pilot users and evaluate it. In addition, we plan to implement API-based file operations (S3 compatible protocol) as a cloud like function. We will also collect performance metrics for all executed jobs on Fugaku and investigate their relationship to electric power.

23.5 Publications

23.5.1 Conference Papers

[1] Yuichi Tsujita, Atsuya Uno, Ryuichi Sekizawa, Keiji Yamamoto, Fumichika Sueyasu, “Job Classification Through Long-Term Log Analysis Towards Power-Aware HPC System Operation”, 2021 29th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)
[2] Takanori Fujiwara, Naohisa Sakamoto, Jorji Nonaka, Keiji Yamamoto, Kwan-Liu Ma, “A visual analytics framework for reviewing multivariate time-series data with dimensionality reduction”, IEEE Transactions on Visualization and Computer Graphics Vol 27, 1601–1611
[3] Yuichi Tsujita, Yoshitaka Furutani, Hajime Hida, Keiji Yamamoto, Atsuya Uno, “Characterizing I/O Optimization Effect Through Holistic Log Data Analysis of Parallel File Systems and Interconnects”, International Conference on High Performance Computing pp. 177–190

23.5.2 Posters

[4] Keiji Yamamoto, “Design of cloud-like functions on the supercomputer Fugaku”, The 3rd R-CCS International Symposium, Feb. 2021.

23.5.3 Oral Talks

[5] Keiji Yamamoto, “Cloud use project for supercomputer Fugaku”, PBS works 2020, Sep. 2020. (in Japanese)

Part III

Flagship 2020 Project

Chapter 24

Flagship 2020 Project

24.1 Members

Primary members are only listed.

24.1.1 System Software Development Team

Yutaka Ishikawa (Team Leader)

Masamichi Takagi (Senior Scientist)

Atsushi Hori (Senior Scientist)

Balazs Gerofi (Senior Scientist)

Takahiro Ogura (Research & Development Scientist)

Fumiyoshi Shoji (Research & Development Scientist)

Atsuya Uno (Research & Development Scientist)

Toshiyuki Tsukamoto (Research & Development Scientist)

Toyohisa Kameyama (Technical Staff I)

Jie Yin (Technical Staff I)

24.1.2 Architecture Development Team

Mitsuhisa Sato (Team Leader)

Yuetsu Kodama (Senior Scientist)

Miwako Tsuji (Research Scientist)

Masahiro Nakao (Research Scientist)

Jinpil Lee (Research Scientist)

Yutaka Maruyama (Research Scientist)

Tetsuya Odajima (Postdoctoral Researcher)

Hitoshi Murai (Technical Scientist)

Motohiko Matsuda (Technical Scientist)

Itaru Kitayama (Technical Staff)

Toshiyuki Imamura (Research Scientist)

Kentaro Sano (Research Scientist)

Table 24.1: Development Teams

Team Name	Team Leader
Architecture Development	Mitsuhisa Sato
System Software Development	Yutaka Ishikawa
Co-Design	Junichiro Makino
Application Development	Hirofumi Tomita

24.1.3 Application Development

Hirofumi Tomita (Team Leader)

Yoshifumi Nakamura (Research Scientist)

Soichiro Suzuki (Research & Development Scientist)

Kazunori Mikami (Research & Development Scientist)

Kiyoshi Kumahata (Research & Development Scientist)

Mamiko Hata (Technical Staff I)

Hiroshi Ueda (Research Scientist)

Naoki Yoshioka (Research Scientist)

Yiyu Tan (Research Scientist)

24.1.4 Co-Design

Junichiro Makino (Team Leader)

Masaki Iwasawa (Research Scientist)

Daisuke Namekata (Postdoctoral Researcher)

Kentaro Nomura (Research Associate)

Miyuki Tsubouchi (Technical Staff)

24.2 Overview of Research Activities

The Japanese government launched the FLAGPSHIP 2020 project [1] in FY 2014 whose missions are defined as follows:

- Building the Japanese national flagship supercomputer, the successor to the K computer, Fugaku supercomputer, and
- developing wide range of HPC applications that will run on the post K computer in order to solve the pressing societal and scientific issues facing our country.

RIKEN is in charge of co-design of the Fugaku supercomputer and development of application codes in collaboration with the Priority Issue institutes selected by Japanese government, as well as research aimed at facilitating the efficient utilization of the Fugaku supercomputer by a broad community of users. Under the co-design concept, RIKEN and the selected institutions are expected to collaborate closely. The official name of the Fugaku supercomputer was decided in May 2019.

As shown in Table 24.1, four development teams are working on Fugaku supercomputer system development with the FLAGSHIP 2020 Planning and Coordination Office that supports development activities. The primary members are listed in Section 24.1.

[1] FLAGSHIP is an acronym for Future LAtency core-based General-purpose Supercomputer with HIgh Productivity.

The Architecture Development team designs the architecture of the Fugaku supercomputer in cooperation with Fujitsu. And, the team designs and develops a productive programming language, called XcalableMP (XMP), and its compiler, and also specifies requirements of standard languages such as Fortran and C/C++ and mathematical libraries provided by Fujitsu.

The System Software Development team designs and specifies a system software stack such as Linux, MPI and File I/O middleware for the Fugaku computer in cooperation with Fujitsu and designs and develops multi-kernel for manycore architectures, Linux with light-weight kernel (McKernel), that provides a noise-less runtime environment, extendibility and adaptability for future application demands. The team also designs and develops a low-level communication layer to provide scalable, efficient and portability for runtime libraries and applications.

The Co-Design team leads to optimize architectural features and application codes together in cooperation with RIKEN teams and Fujitsu. It also designs and develops an application framework, FDPS (Framework for Developing Particle Simulator), to help HPC users implement advanced algorithms.

The Application Development team is a representative of nine institutions aimed at solving Priority Issues. The team figures out weakness of target application codes in terms of performance and utilization of hardware resources and discusses them with RIKEN teams and Fujitsu to find out best solutions of architectural features and improvement of application codes.

24.3 Target of System Development and Achievements in FY2020

The Fugaku's design targets are as follows:

- A one hundred times speed improvement over the K computer is achieved in maximum case of some target applications. This will be accomplished through co-design of system development and target applications for the nine Priority Issues.

- The maximum electric power consumption should be between 30 and 40 MW.

In FY2020, the installation of Fugaku hardware was completed in May. Fujitsu continued to test the Fugaku software stack. The system partially served the early-access program including projects to fight against the COVID-19 pandemic.

The major components system software is summarized as follows:

- Highly productive programming language, XcalableMP
  XcalableMP (XMP) is a directive-based PGAS language for large scale distributed memory systems that combine HPF-like concept and OpenMP-like description with directives. Two memory models are supported: global view and local view. The global view is supported by the PGAS feature, i.e., large array is distributed to partial ones in nodes. The local view is provided by MPI-like + Coarray notation. We finished the front-end for Fortran 2008 Standard for Omni XcalableMP compiler. The Omni XcalableMP compiler is already available in Fugaku. In 2020, we designed the XMP APIs for a compiler-free approach, which is reported in Section 2.3.4, and made a prototype implementation of XcalableMP for C++ for future extensions of XMP. And, we are working on the research for XcalableMP 2.0 which newly supports task-parallelism with the integration of PGAS models for distributed memory environment.

- Domain specific library/language, FDPS
  FDPS is a framework for the development of massively parallel particle simulations. Users only need to program particle interactions and do not need to parallelize the code using the MPI library. The FDPS adopts highly optimized communication algorithms and its scalability has been confirmed using the K computer.

- MPI + OpenMP programming environment
  The current de facto standard programming environment, i.e., MPI + OpenMP environment, is supported. Two MPI implementations have been developed. Fujitsu supports own MPI implementation based on the OpenMPI. RIKEN collaborates with ANL (Argonne National Laboratory) to develop MPICH, mainly developed at ANL, for Fugaku supercomputer.

- New file I/O middleware
  The Fugaku supercomputer does not employ the file staging technology for the layered storage system. The users do not need to specify which files must be staging-in and staging-out in their job scripts in the

Fugaku supercomputer environment. The LLIO middleware, employing asynchronous I/O and caching technologies, was designed by Fujitsu in order to provide transparent file access with better performance. Its scalability has been evaluated in FY2020.

- Application-oriented file I/O middleware
  In scientific Big-Data applications, such as real-time weather prediction using observed meteorological data, a rapid data transfer mechanism between two jobs, ensemble simulations and data assimilation, is required to meet their deadlines. A framework called Data Transfer Framework (DTF), based on PnetCDF file I/O library, that silently replaces file I/O with sending the data directly from one component to another over network was developed.

- Process-in-Process
  "Process-in-Process" or "PiP" in short is a user-level runtime system for sharing an address space among processes. Unlike the Linux process model, a group of processes shares the address space and thus the process context switch among those processes does not involve hardware TLB flushing.

- Multi-Kernel for manycore architectures
  Multi-Kernel, Linux with light-weight Kernel (McKernel) is being designed and implemented. It provides: i) a noiseless execution environment for bulk-synchronous applications, ii) ability to easily adapt to new/future system architectures, e.g., manycore CPUs, a new process/thread management, a memory management, heterogeneous core architectures, deep memory hierarchy, etc., and iii) ability to adapt to new/future application demands, such as Big-Data and in-situ applications that require optimization of data movement. The achievements in FY2020 is described in Section 1.3.1.

The architecture development team carries out the research on co-design tools as well as the design of the Fugaku supercomputer:

GEM5 processor simulator for the Fugaku processor
We are developing a cycle-level processor simulator for the Fugaku processor based on GEM-5, which is a general-purpose processor simulator commonly used for the processor architecture research. ARM provided us the source code of GEM-5 Atomic-model processor simulator for ARM v8 with Scalable Vector Extension (SVE). The Atomic model enables an instruction-level simulation. We deployed and tested it, and extend it for the cycle-level Out-Of-Order(O3) model processor simulator with the Fugaku hardware parameters. It enables the cycle-level performance evaluation of application kernels. This simulator was provided as the service of "Fugaku performance evaluation environment" for performance evaluation and tuning by potential Fugaku users.

Performance estimation tools for co-design study
We have tools for co-design study for future huge-scale parallel systems. The MPI application replay tool is a system to investigate a performance and behavior of parallel applications on a single node using MPI traces. SCAMP (SCAlable Mpi Profiler) is other system to simulate a large-scale network from a small number of profiling results.

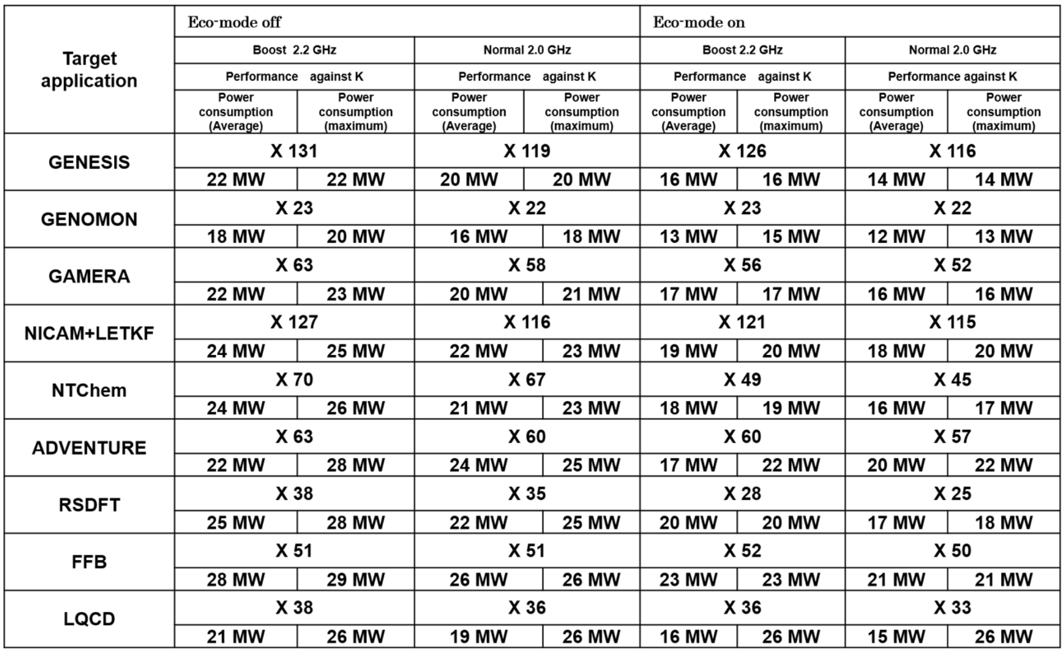
Study on performance metrics
We have been developing a new metric, called Simplified Sustained System Performance (SSSP) metric, based on a suite of simple benchmarks, which enables performance projection that correlates with applications.

In addition to co-design tools, we were working on the evaluation of compilers for ARM SVE. There are two kinds of compiler for ARM SVE: Fujitsu Compiler and ARM compiler. The Fujitsu compiler is a proprietary compiler supporting C/C++ and Fortran. The ARM compiler is developed by ARM based on LLVM. Initially, LLVM only supports C and C++, and supports Fortran recently by flang. We evaluated the quality of code generated by both compilers with collaboration of Kyoto University. And our team carried out several collaborations with ARM compiler team on LLVM.

In June 2020, the Fugaku achieved HPL performance of 415.53 PFLOPS, using 396 racks (152,064 nodes, approximately 95.6% of the entire system) with a computing efficiency ratio of 80.87%, and ranked as the 1st position of the Top 500 list. It also took the first place in the ranking of HPCG, achieving 13,400 TFLOPS (360 racks, 138,240 nodes, approximately 87% of the entire system), while claiming the first position in the HPL-AI ranking with 1.421 EFLOPS using 330 racks (126,720 nodes, approximately 79.7% of the entire system) HPL-AI is a new benchmark that takes into account the capabilities of single-precision and half-precision arithmetic logic units used in artificial intelligence. It has taken the top spot on the Graph500 list, a ranking of the world's fastest supercomputers on data-intensive workloads, with the performance of a breadth-first search of a large scale graph 70,980 GTEPS, using 92,160 nodes (approximately 58% of the entire system). In November 2020, Fugaku kept the first position of these 4 benchmarks by updating the performance, 442.01 PFLOPS (82.3%) for HPL, 16.00 PFLOPS (3.0%) for HPCG, 2.00 EFLOPS (93.2%) for HPL-AI and 102.95 Tteps for Graph500, using full system of Fugaku. The achievement of remarkable records in these rankings demonstrates the overall high performance of Fugaku for a wide range of workloads.

In 2020, we presented the paper about our codesign effort for Fugaku and A64FX processor in SC20[1]. We carried out several performance evaluation of Fugaku and A64FX manycore processor using UK benchmarks, open source HPC software and SPEC benchmarks, which is reported in Section 2.3.1. The detail of Graph500 benchmark is reported in Section 2.3.2.

Although the nine priority issues successfully finished at the end of March 2020, we have continued to concentrate on more accurate performance estimation regarding the nine target application with the cooperation of co-design members of the former nine priority issues and compiler and system software groups. This fiscal year, unlike the last year, the number of nodes used has gradually increased. As a result, performance estimation was performed on the full system. After we conducted the tunning and improvement of the application algorithm. almost all of the nine target applications can obtain their target performance within the power cap. Figure 24.1 gives the final performance result of target application target application performance on Fugaku. We confirmed that the actual performances for the almost applications exceeded the target performances and the power consumption of all running patterns was within the power cap.



Target application	Eco-mode off				Eco-mode on			
	Boost 2.2 GHz		Normal 2.0 GHz		Boost 2.2 GHz		Normal 2.0 GHz	
	Performance against K		Performance against K		Performance against K		Performance against K	
	Power consumption (Average)	Power consumption (maximum)	Power consumption (Average)	Power consumption (maximum)	Power consumption (Average)	Power consumption (maximum)	Power consumption (Average)	Power consumption (maximum)
GENESIS	X 131		X 119		X 126		X 116	
	22 MW	22 MW	20 MW	20 MW	16 MW	16 MW	14 MW	14 MW
GENOMON	X 23		X 22		X 23		X 22	
	18 MW	20 MW	16 MW	18 MW	13 MW	15 MW	12 MW	13 MW
GAMERA	X 63		X 58		X 56		X 52	
	22 MW	23 MW	20 MW	21 MW	17 MW	17 MW	16 MW	16 MW
NICAM+LETKF	X 127		X 116		X 121		X 115	
	24 MW	25 MW	22 MW	23 MW	19 MW	20 MW	18 MW	20 MW
NTChem	X 70		X 67		X 49		X 45	
	24 MW	26 MW	21 MW	23 MW	18 MW	19 MW	16 MW	17 MW
ADVENTURE	X 63		X 60		X 60		X 57	
	22 MW	28 MW	24 MW	25 MW	17 MW	22 MW	20 MW	22 MW
RSDFT	X 38		X 35		X 28		X 25	
	25 MW	28 MW	22 MW	25 MW	20 MW	20 MW	17 MW	18 MW
FFB	X 51		X 51		X 52		X 50	
	28 MW	29 MW	26 MW	26 MW	23 MW	23 MW	21 MW	21 MW
LQCD	X 38		X 36		X 36		X 33	
	21 MW	26 MW	19 MW	26 MW	16 MW	26 MW	15 MW	26 MW

Figure 24.1: The final performance result of target application. Each magnification value is the reduction rate of the elapsed time compared to when running on K computer.

24.4 International Collaborations

24.4.1 DOE/MEXT Collaboration

The following research topics were performed under the DOE/MEXT collaboration MOU.

- Efficient MPI for exascale
  In this research collaboration, the next version of MPICH MPI implementation, mainly developed by Argonne National Laboratory (ANL), has been cooperatively developed. The FY2016 achievements have been described in the previous section.

- Metadata and active storage
  This research collaboration, run by the University of Tsukuba as contract, studies metadata management and active storage.

- Storage as a Service
  This research collaboration explores APIs for delivering specific storage service models. This is also run by the University of Tsukuba.

- Parallel I/O Libraries
  This research collaboration is to improve parallel netCDF I/O software for extreme-scale computing facilities at both DOE and MEXT. To do that, the RIKEN side has designed DTF as described in the previous section.

- OpenMP/XMP Runtime
  This research collaboration explores interaction of Argobots/MPI with XcalableMP and PGAS models.

- Exascale co-design and performance modeling tools
  This collaborates on an application performance modeling tools for extreme-scale applications, and shared catalog of US/JP mini-apps.

- LLVM for vectorization
  This research collaboration explores compiler techniques for vectorization on LLVM.

- Power Monitoring and Control, and Power Steering
  This research collaboration explores APIs for monitoring, analyzing, and managing power from the node to the global machine, and power steering techniques for over-provisioned systems are evaluated.

- Performance Evaluation of Fugaku and Porting ECP software
  Since the beginning of 2021, the collaboration on the performance evaluation of Fugaku was started by providing the accounts for US collaborators. It was also expected to port ECP software to Fugaku for the enhancement of Fugaku's usability.

24.4.2 CEA

RIKEN and CEA, Commissariat *à* l'*é*nergie atomique et aux *é*nergies alternatives, signed MOU in the fields of computational science and computer science concerning high performance computing and computational science in January 2017. The following collaboration topics are selected:

- Programming Language Environment
- Runtime Environment
- Energy-aware batch job scheduler
- Large DFT calculations and QM/MM
- Application of High Performance Computing to Earthquake Related Issues of Nuclear Power Plant Facilities
- Key Performance Indicators (KPIs)
- Human Resource and Training

While continuing these topics in 2020, two new topics of particular relevance emerged from the evolving HPC landscape from 2019:

- big data and AI became increasingly important in many sectors and applications. The combination of big data and AI with numerical simulation and HPC became increasingly important
- ARM-based processors made significant breakthroughs in the HPC ecosystem, in particular with the Fujitsu A64FX processor co-designed by RIKEN.

24.5 Schedule and Future Plan

The public use of the Fugaku system has started and available since March 2021. This project was ended in FY2020.

24.6 Publications

24.6.1 Articles/Journal

[1] Z. R. Kordov, R. Horsley, Y. Nakamura, H. Perlt, P. E. L. Rakow, G. Schierholz, H. Stüben, R. D. Young, and J. M. Zanotti, “Electromagnetic contribution to $\Sigma - \Lambda$ mixing using lattice QCD+QED”, Phys. Rev. D 101, 034517 (2020)

24.6.2 Conference Papers

[1] Mitsuhisa Sato, Yutaka Ishikawa, Hirofumi Tomita, Yuetsu Kodama, Tetsuya Odajima, Miwako Tsuji, Hisashi Yashiro, Masaki Aoki, Naoyuki Shida, Ikuo Miyoshi, Kouichi Hirai, Atsushi Furuya, Akira Asato, Kuniki Morita, Toshiyuki Shimizu: Co-design for A64FX manycore processor and ”Fugaku”. SC 2020: 47
[2] 鈴木惣一朗, 伊東聰、酒井憲一郎、稲田由江、三吉郁夫、石川裕、宮野悟, 「ヒト全ゲノム解析プログラムGenomonのスーパーコンピュータ「富岳」向け最適化と性能評価」 研究報告ハイパフォーマンスコンピューティング, Vol.2021-HPC-178, No.18, P.1-9

24.6.3 Invited Talks

[1] Mitsuhisa Sato, The Supercomputer “Fugaku” and Arm-SVE enabled A64FX processor for energy-efficiency and sustained application performance, International Symposium on Parallel and Distributed Computing (ISPDC 2020).
[2] Mitsuhisa Sato, The Supercomputer Fugaku and Arm-SVE enabled A64FX processor for energy-efficiency and sustained application performance, Excalibur SLE 2020, July 15th 2020.
[3] Mitsuhisa Sato, The Supercomputer “Fugaku” and Arm-SVE enabled A64FX processor for energy-efficiency and sustained application performance, The Argonne Training Program on Extreme-Scale Computing (ATPESC) ATPESC 2020, July 27th 2020,
[4] Mitsuhisa Sato, The Supercomputer “Fugaku” and Arm-SVE enabled A64FX processor for energy-efficiency and sustained application performance, 第14回アクセラレーション技術発表討論会　2020年 9月 3日(木)
[4] Mitsuhisa Sato, The Supercomputer “Fugaku” and Arm-SVE enabled A64FX processor for energy-efficiency and sustained application performance, KSC 2020, September 25th 2020
[4] Mitsuhisa Sato, The Supercomputer “Fugaku” and A64FX Manycore Processor, CCS International Symposium 2020, U Tsukuba, 6th Oct. 2020,
[4] Mitsuhisa Sato, スーパーコンピュータ「富岳」の概要,第10回材料系ワークショップ, 2020年10月16日
[4] Mitsuhisa Sato, Performance Evaluation of The Supercomputer “Fugaku” and A64FX Manycore Processor ScalA workshop 2020, 12th Nov. 2020,
[4] Mitsuhisa Sato, スーパーコンピュータ「富岳」システム, 「富岳で加速する素粒子・原子核・宇宙・惑星」シンポジウム, 1月28日 2021年.
[4] Mitsuhisa Sato, スーパーコンピュータ「富岳」の開発とコデザイン, IEICE-ICT-Webinar 2021年 1月19日.

24.6.4 Oral Talks

[1] Yoshifumi Nakamura, “Supercomputer Fugaku and QCD Wide SIMD Library (QWS) on Fugaku” Asia-Pacific Symposium for Lattice Field Theory (APLAT 2020), Aug 4th, 2020 (online).

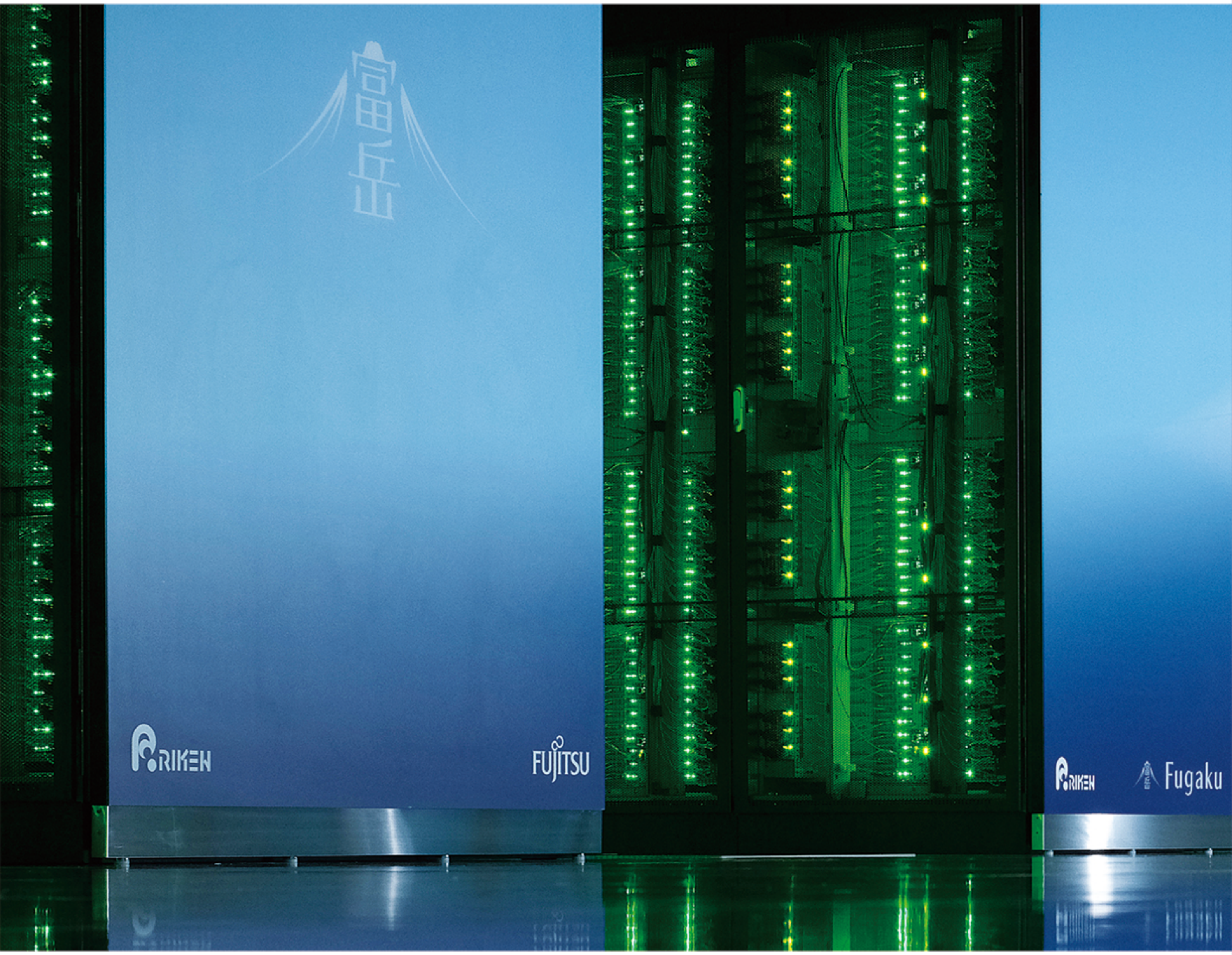
24.6.5 Software

24.6.6 Patents

RIKEN Center for Computational Science

ANNUAL REPORT

FY2020



7-1-26, Minatojima-minami-machi, Chuo-ku, Kobe, Hyogo 650-0047 Japan
https://www.r-ccs.riken.jp/en/

RIKEN 2021-054