Fugaku: the First 'Exascale' Supercomputer - Past, Present and Future.

Satoshi Matsuoka, Director R-CCS / Prof. Tokyo Inst. Tech.

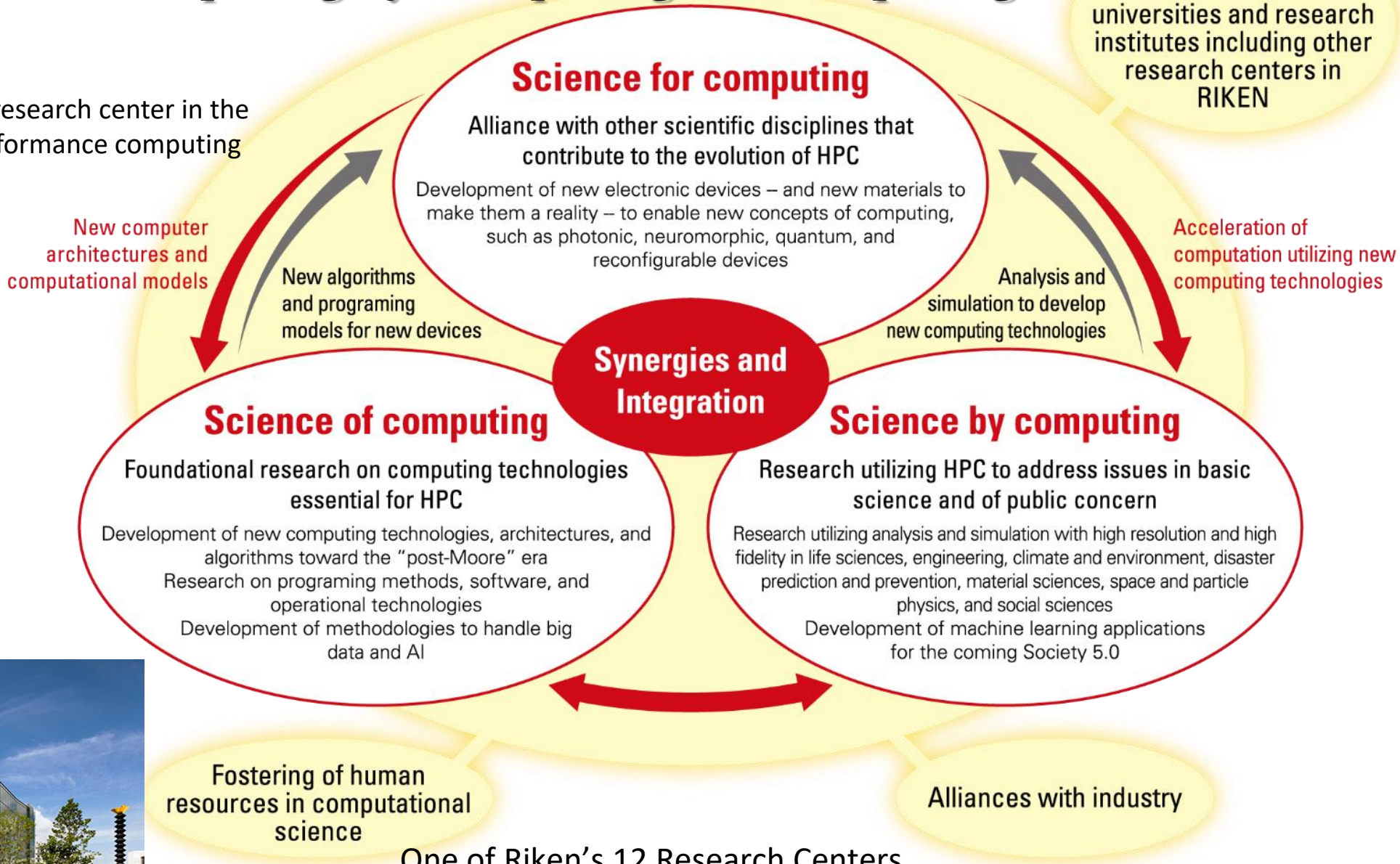
Cluster 2020 Keynote Presentation

15 Sep. 2020

Science of Computing by Computing for Computing

R-CCS

International core research center in the science of high performance computing (HPC)



One of Riken's 12 Research Centers
18 Research Teams + 4 Operations Teams > 200 FTEs

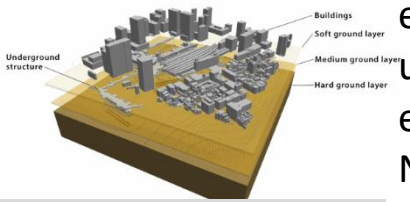
III Achievements

First 'Exascale' SC: Fugaku R&D

Fugaku/post- K rack, CPU (prototype) ©Fujitsu

- R&D started since the genesis of the center, to **design & build world's first "exascale" supercomputer**
- Fugaku construction official approval to build by the government on Nov. 2018
- **Installation start Dec. 2019, operations to start early 2021**

Groundbreaking Apps: E.g., Large-scale Earthquake disaster simulation w/HPC & AI



Modeling of bldgs., underground structures, with geological layers ©Tokyo Univ.

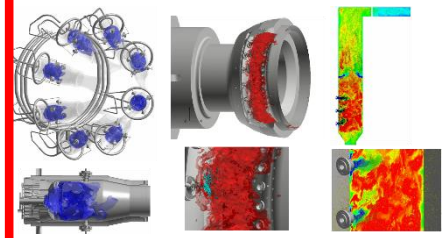
Ultra large scale earthquake simulation using AI for massive scale earthquake in Tokyo area. Nominated as **finalist of Gordon bell 2018**. Winning **SC16, 17 Best Poster Award.**

Top-Tier Awards HPC

High Speed algorithm for analyzing social networks won **Best paper Award** at HiPC. Contribution of Dr. Matsuoka has awarded at **ACM HPDC 2018 Achievement Award** also **Asia HPC Leadership Award** at supercomputing Asia 2019.

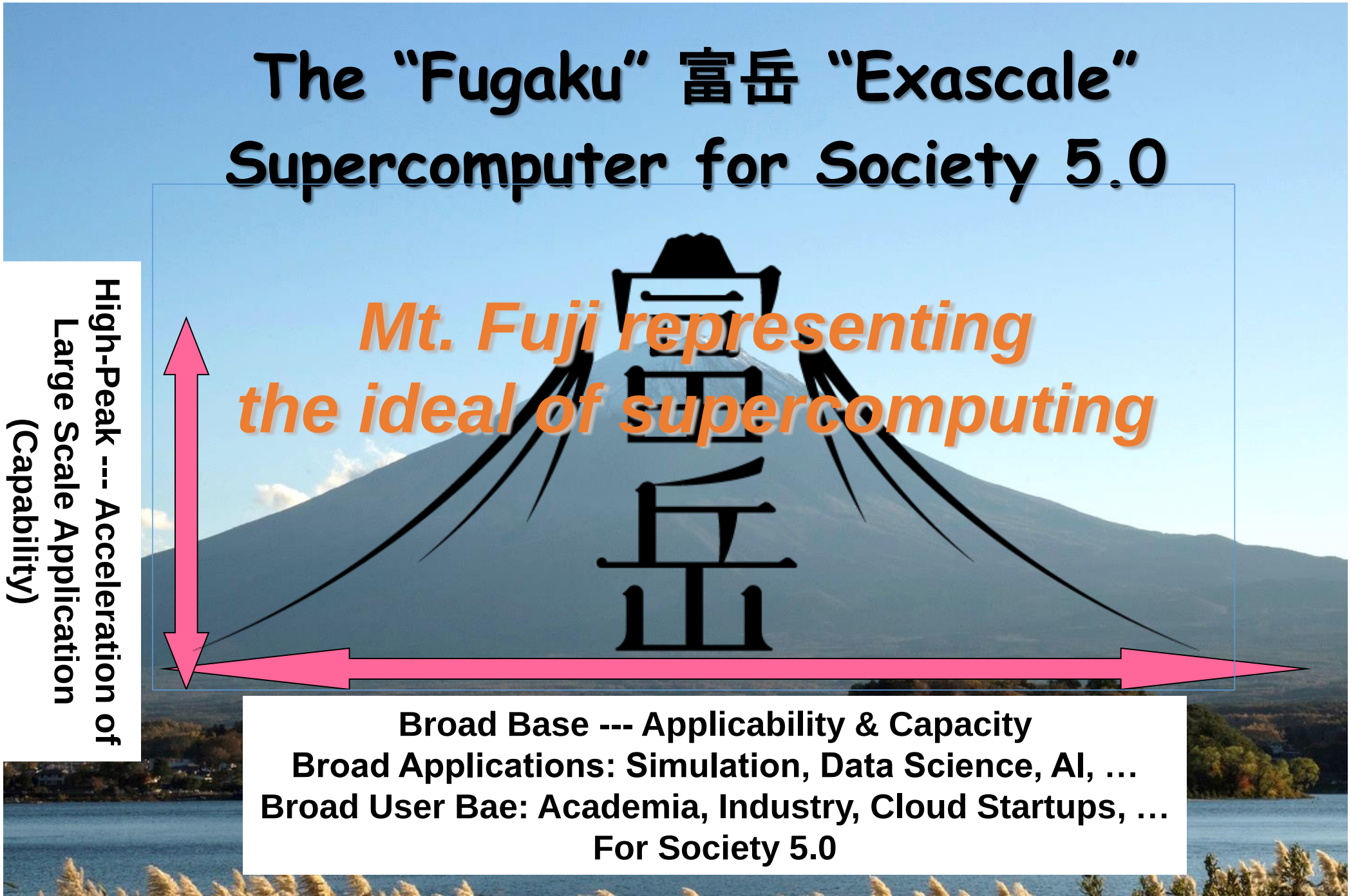
Industry outreach e.g. RIKEN Combustion System CAE Consortium

RIKEN Consortium to develop next generation CAE for combustors has established with **11 industries and 9 academia** members. It would expand the users of **R-CCS Software** as well as **Fugaku**.



Simulation CAE examples

The "Fugaku" 富岳 "Exascale" Supercomputer for Society 5.0



High-Peak --- Acceleration of Large Scale Application (Capability)

Broad Base --- Applicability & Capacity
Broad Applications: Simulation, Data Science, AI, …
Broad User Bae: Academia, Industry, Cloud Startups, …
For Society 5.0

Fugaku: Largest & Fastest Supercomputer Ever

'Applications First' R&D Challenge--- High Risk "Moonshot" R&D

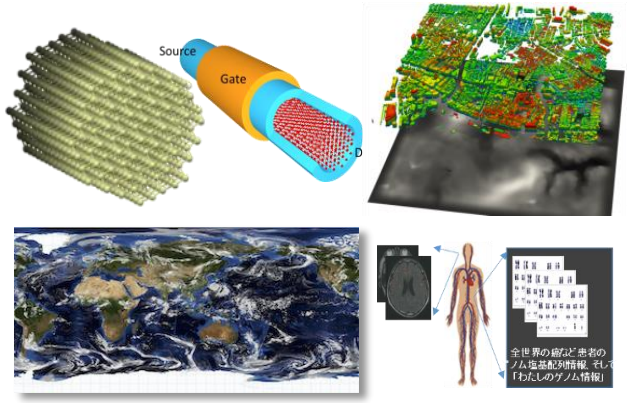
- **A new high performance & low power Arm A64FX CPU co-developed by Riken R-CCS & Fujitsu along with nationwide HPC researchers as a National Flagship 2020 project**

- 3x perf c.f. top CPU in HPC apps
- 3x power efficiency c.f. top CPU
- General purpose Arm CPU, runs same program as Smartphones
- Acceleration features for AI

"Moonshot" R&D Target

- **Fugaku x 2~3 = Entire annual IT in Japan**

	Smartphones		Servers (incl. IDC)		**Fugaku**		K Computer
Untis	20 million ~annual shipment in Japan	=	300,000 (~annual shipment in Japan	=	1 (160K nodes)		Max 120
Power (W)	10W×2,000万台= 200MW	=	600-700W×30万台= 200MW (incl cooling)	>>	30MW (very low)		15MW (less than 1/10 efficiency c.f. Fugaku)



- **Developed via extensive co-design**

"Science of Computing"

By Riken & Fujitsu & HPCI Centers, etc., Arm Ecosystem, Reflecting numerous research results

"Science by Computing"

"9 Priority Areas" to develop target applications to tackle important societal problems

Brief History of R-CCS towards Fugaku

July 2010
RIKEN AICS established
August 2010
HPCI Project start
September 2010
K computer installation start
First meeting of SDHPC (Post-K)

April 2012
Post-K Feasibility Study start
3 Arch Teams and 1 Apps Team
June 2012
K computer construction complete
September 2012
K computer production start
November 2012
ACM Gordon bell Award

April 2018
AICS renamed to RIKEN R-CCS.
Satoshi Matsuoka becomes new Director
Aug 2018
Arm A64fx announce at Hotchips
Oct 2018
NEDO 100x processor project start
Nov 2018
Post-K Manufacturing approval by Prime Minister's CSTI Committee

April 2020
COVID19 Teams Start
May 2020
Fugaku Shipment Complete
June 2020
Fugaku #1 on Top 500 etc.

2006 2010 2011 2012 2013 2014 2018 2019 2020

January 2006
Next Generation Supercomputer Project (K Computer) start

June 2011
K #1 on Top 500
November 2011
K #1 on Top 500 > 10 Petaflops
ACM Gordon Bell Award
End of FY 2011 (March 2012)
SDHPC Whitepaper

End of FY2013 (Mar 2014)
Post-K Feasibility Study Reports

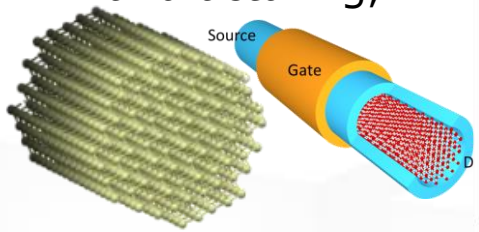
April 2014
Post-K project start
June 2014
K #1 on Graph 500

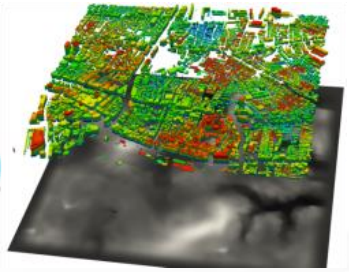
March 2019
Post-K Manufacturing start
May 2019
Post-K named "Supercomputer Fugaku"
July 2019
Post-Moore Whitepaper start
Aug 2019
K Computer shutdown
Dec 2019
Fugaku installation start

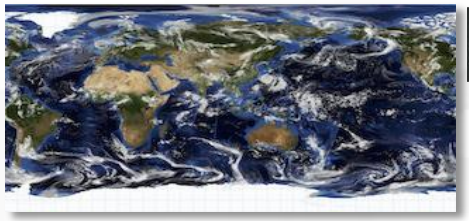
“Applications First” Co-Design Activities in Fugaku

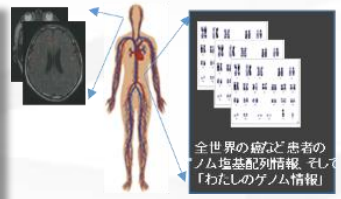
Science by Computing

・9 Priority App Areas: High Concern to General Public: Medical/Pharma, Environment/Disaster, Energy, Manufacturing, …









Select representatives fr
om 100s of applications
signifying various compu
tational characteristics

Design systems with param
eters that consider various
application characteristics

Science of Computing

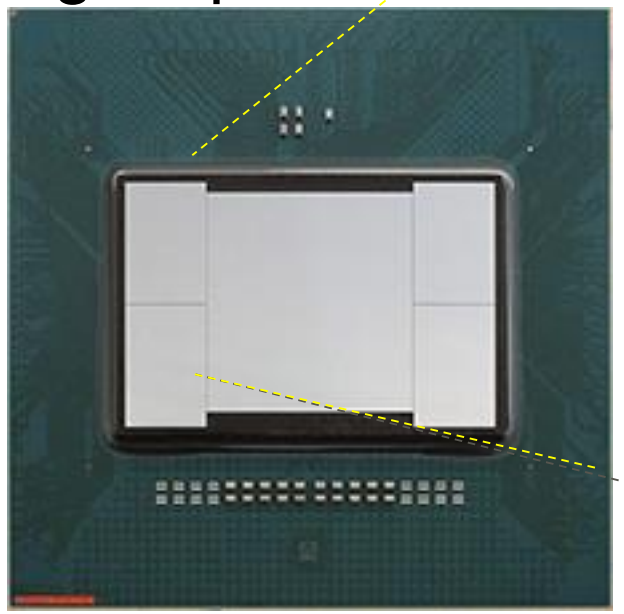
Riken R-CCS & Fujitsu

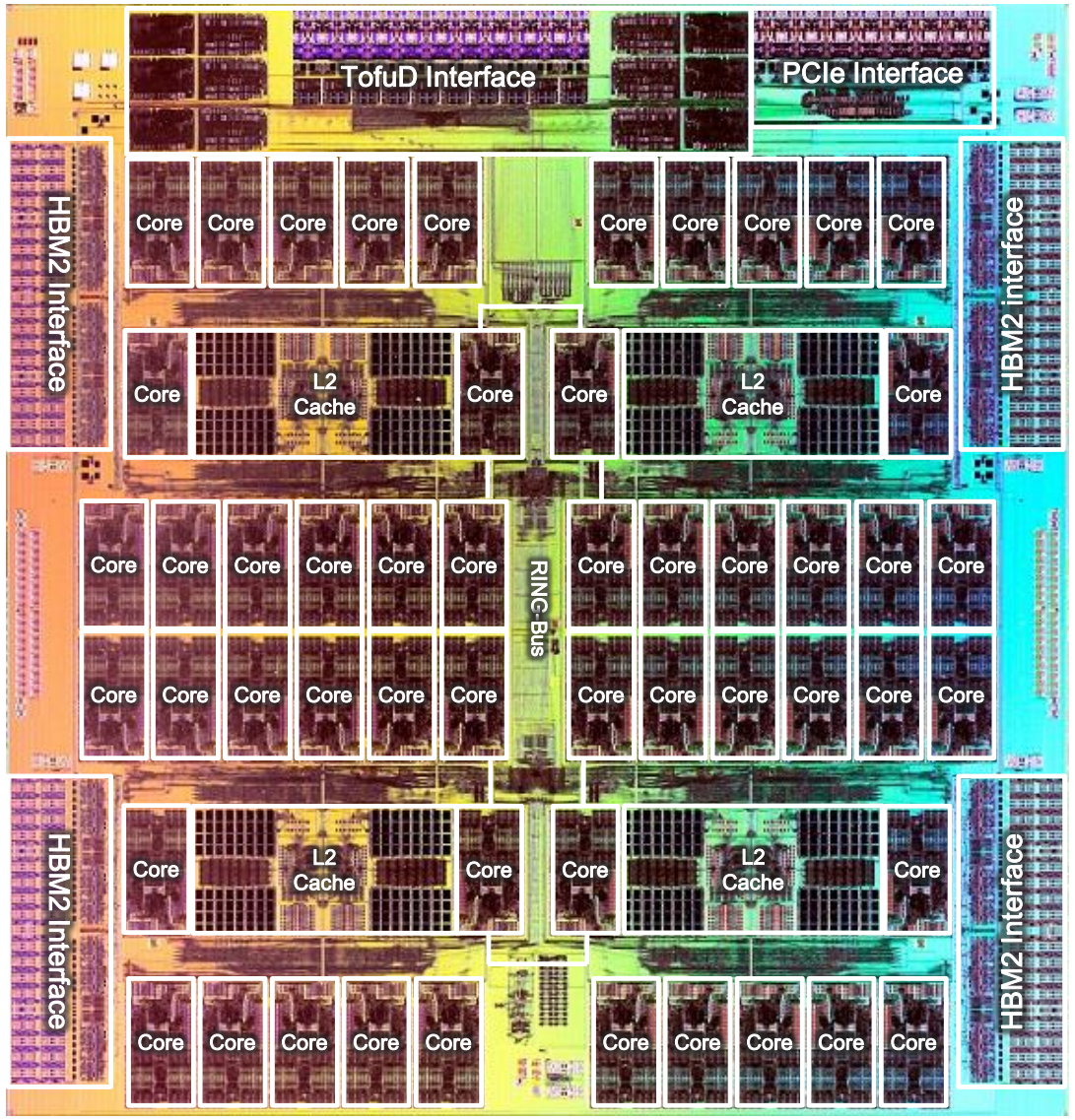
- **Extremely tight collabrations between the Co-Design apps centers, Riken, and Fujitsu, etc.**
- **Chose 9 representative apps as “target application” scenario**
- **Achieve up to x100 speedup c.f. K-Computer**
- **Also ease-of-programming, broad SW ecosystem, very low power, …**

A64FX Leading-edge Si-technology

FUJITSU

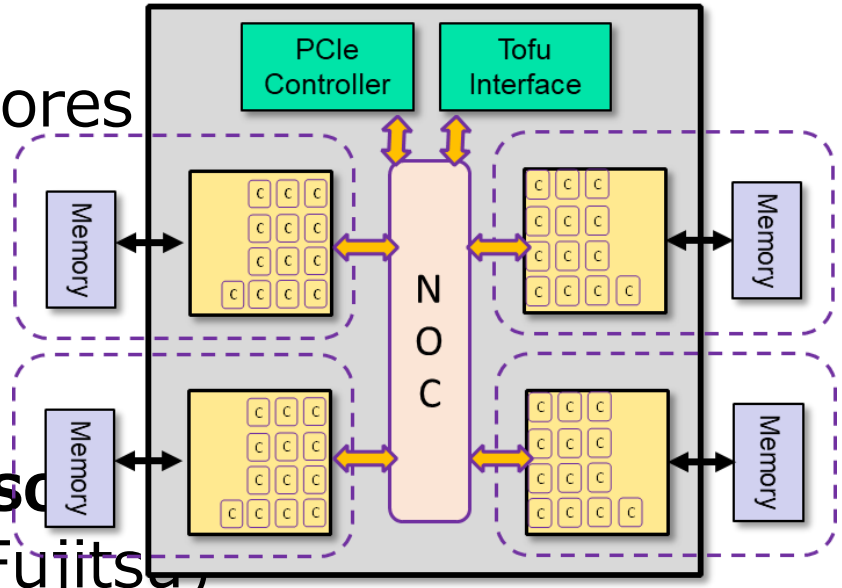
- TSMC 7nm FinFET & CoWoS
  - Broadcom SerDes, HBM I/O, and SRAMs
  - 8.786 billion transistors
  - 594 signal pins





Fugaku's FUjitsu A64fx Processor is…

- **an Many-Core ARM CPU…**
  - 48 compute cores + 2 or 4 assistant (OS) cores
  - Brand new core design
  - Near Xeon-Class Integer performance core
  - ARM V8 --- 64bit ARM ecosystem
  - Tofu-D + PCIe 3 external connection
- **…but also an accelerated GPU-like processo**
  - SVE 512 bit x 2 vector extensions (ARM & Fujitsu)
    - Integer (1, 2, 4, 8 bytes) + Float (16, 32, 64 bytes)
  - Cache + memory localization (sector cache)
  - HBM2 on package memory – Massive Mem BW (Bytes/DPF ~0.4)
    - Streaming memory access, strided access, scatter/gather etc.
  - Intra-chip barrier synch. and other memory enhancing features



- **GPU-like High performance in HPC especially CFD-- Weather & Climate (even with traditional Fortran code) + AI/Big Data**

A64FX Chip Overview

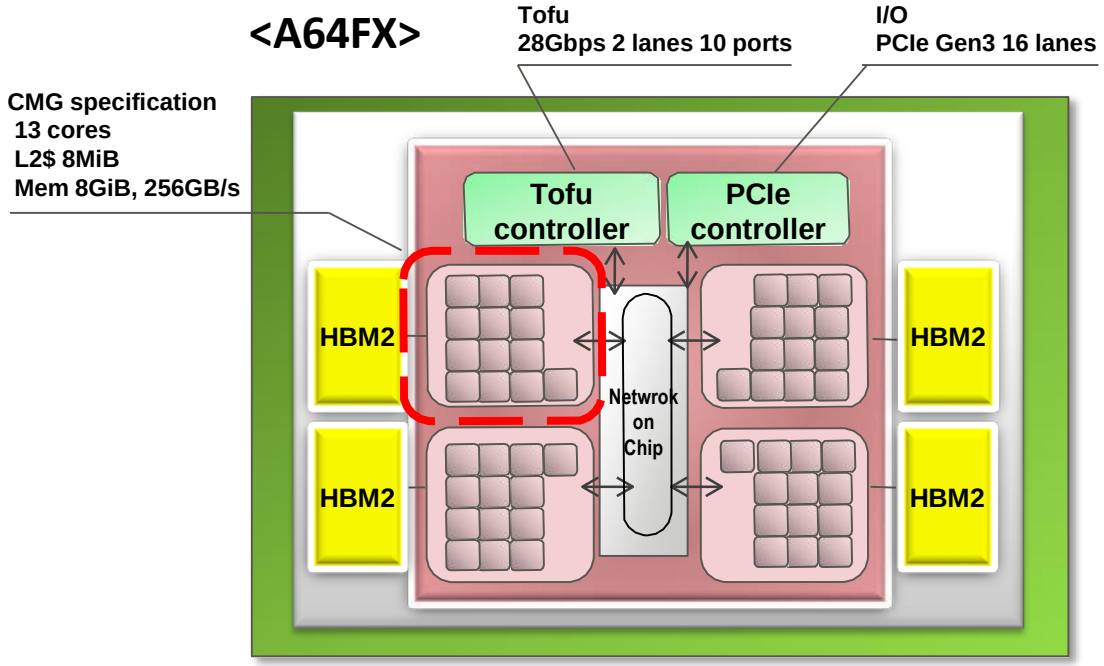
FUJITSU

Architecture Features

- Armv8.2-A (AArch64 only)
- SVE 512-bit wide SIMD
- 48 computing cores + 4 assistant cores*

*All the cores are identical

- HBM2 32GiB
- Tofu 6D Mesh/Torus
  28Gbps x 2 lanes x 10 ports
- PCIe Gen3 16 lanes



7nm FinFET

- 8,786M transistors
- 594 package signal pins

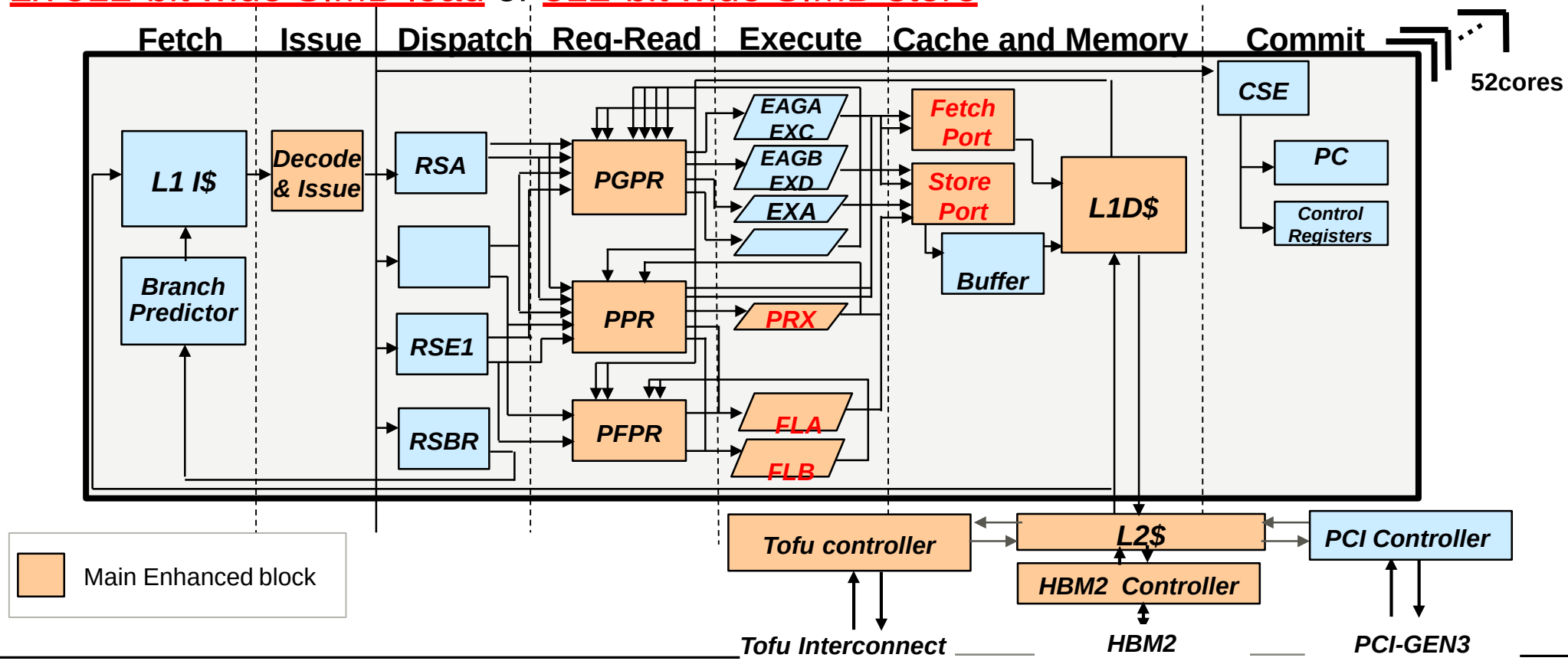
Peak Performance (Efficiency)

- >2.7TFLOPS (>90%@DGEMM)
- Memory B/W 1024GB/s (>80%@Stream Triad)

	A64FX (Post-K)	SPARC64 XIfx (PRIMEHPC FX100)
ISA (Base)	Armv8.2-A	SPARC-V9
ISA (Extension)	SVE	HPC-ACE2
Process Node	7nm	20nm
Peak Performance	>2.7TFLOPS	1.1TFLOPS
SIMD	512-bit	256-bit
# of Cores	48+4	32+2
Memory	HBM2	HMC
Memory Peak B/W	1024GB/s	240GB/s x2 (in/out)

A64FX Core Pipeline

- A64FX enhances and inherits superior features of SPARC64
  - Inherits superscalar, out-of-order, branch prediction, etc.
  - Enhances SIMD and predicate operations
    - 2x 512-bit wide SIMD FMA + Predicate Operation + 4x ALU (shared w/ 2x AGEN)
    - 2x 512-bit wide SIMD load or 512-bit wide SIMD store

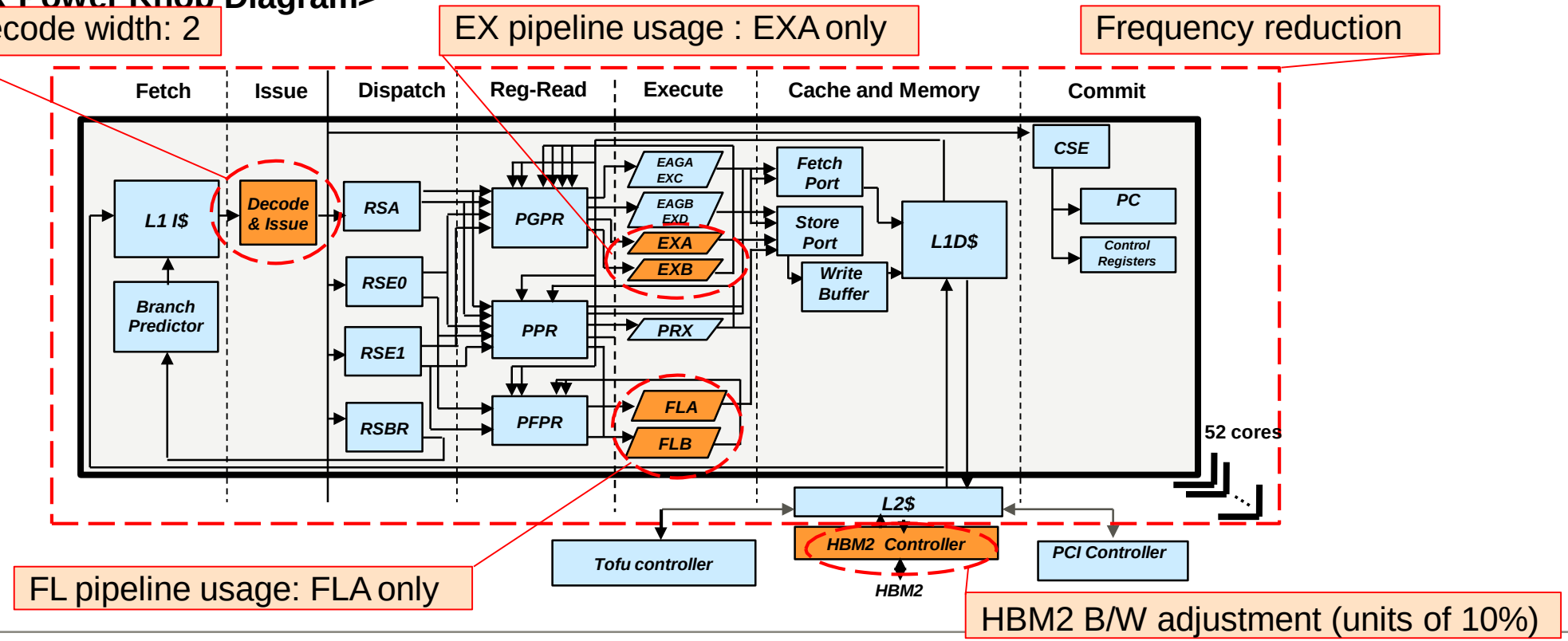


Power Management (Cont.)

- "Power knob" for power optimization
  - A64FX provides power management function called "Power Knob"
    - Applications can change hardware configurations for power optimization

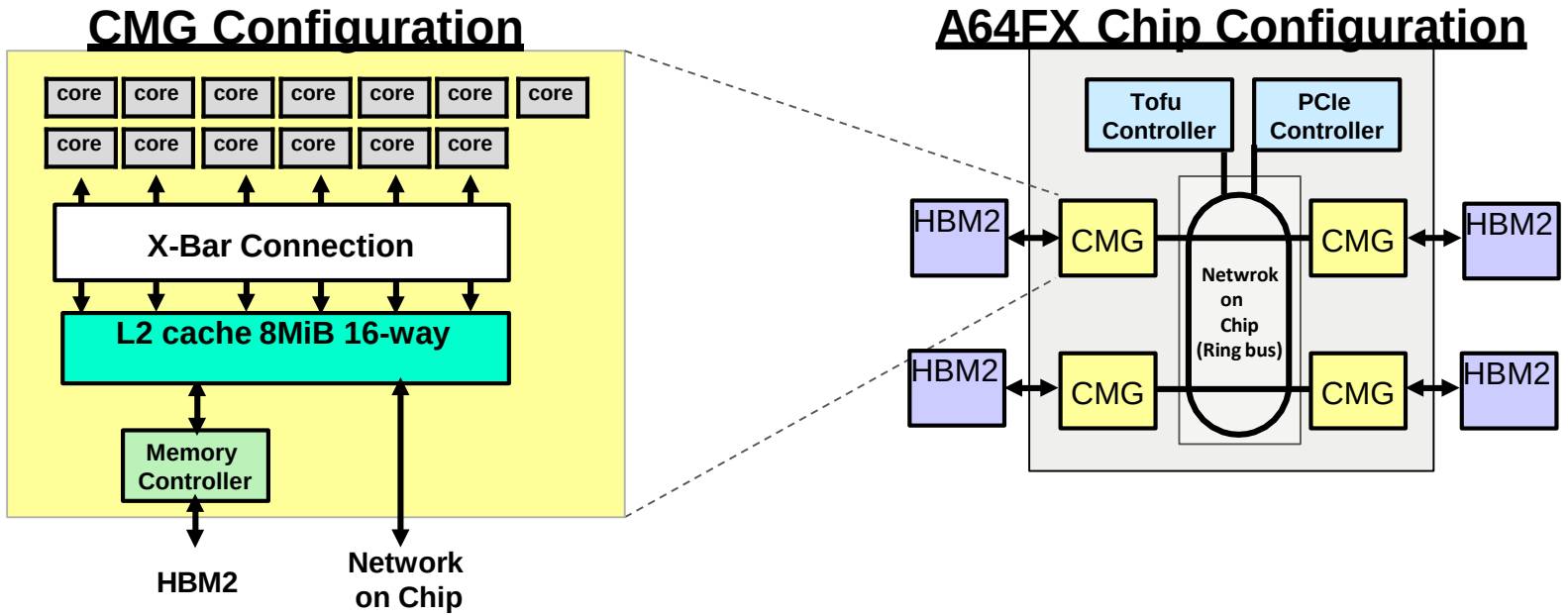
→ Power knobs and Energy monitor/analyzer will help users to optimize power consumption of their applications

<A64FX Power Knob Diagram>



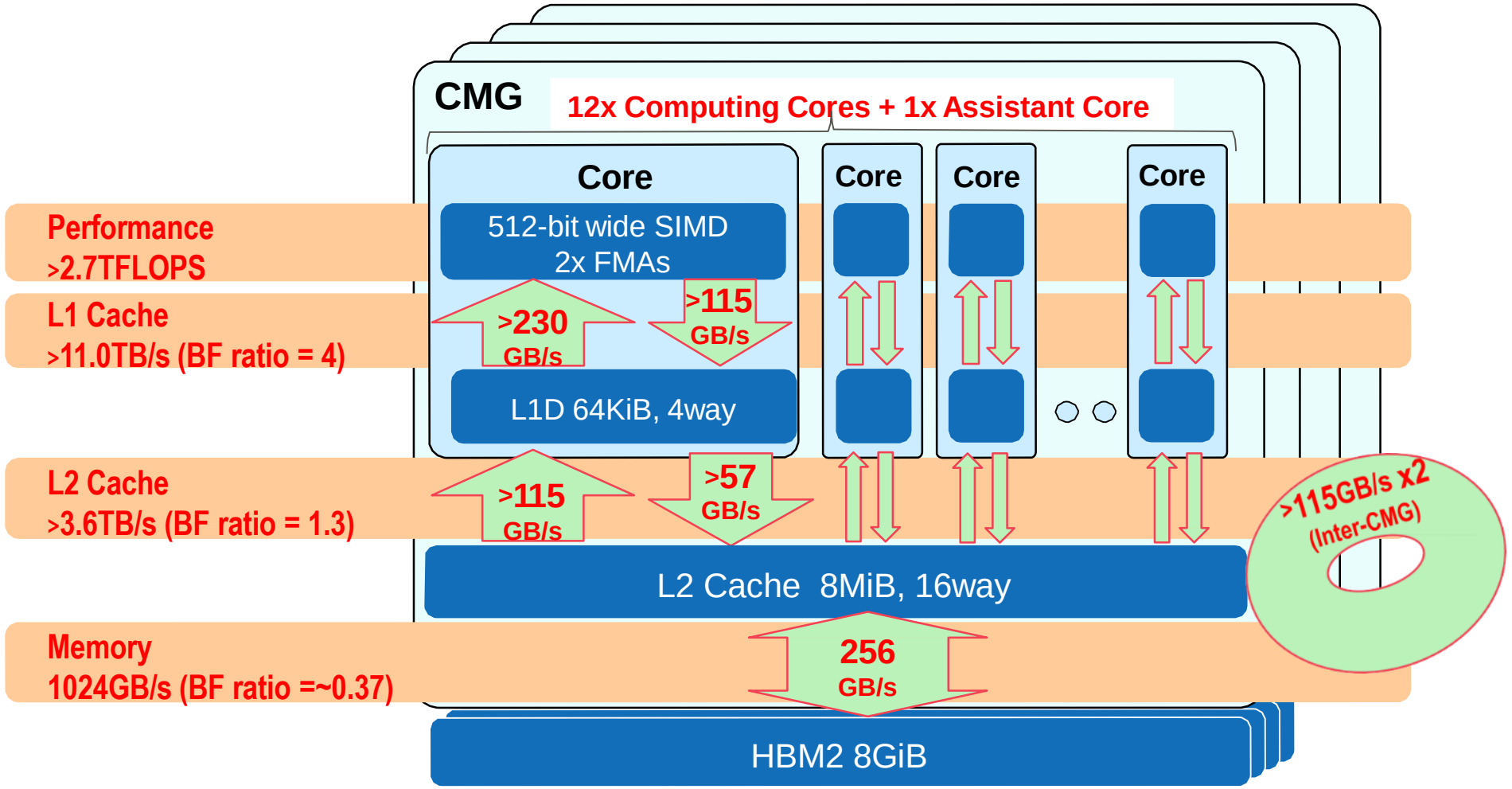
Many-Core Architecture

- A64FX consists of four CMGs (Core Memory Group)
  - A CMG consists of 13 cores, an L2 cache and a memory controller
    - One out of 13 cores is an assistant core which handles daemon, I/O, etc.
  - Four CMGs keep cache coherency by ccNUMA with on-chip directory
  - X-bar connection in a CMG maximizes high efficiency for throughput of the L2 cache
  - Process binding in a CMG allows linear scalability up to 48 cores
- On-chip-network with a wide ring bus secures I/O performance



High Bandwidth

- Extremely high bandwidth in caches and memory
  - A64FX has out-of-order mechanisms in cores, caches and memory controllers. It maximizes the capability of each layer's bandwidth



Fujitsu Mission Critical Technologies

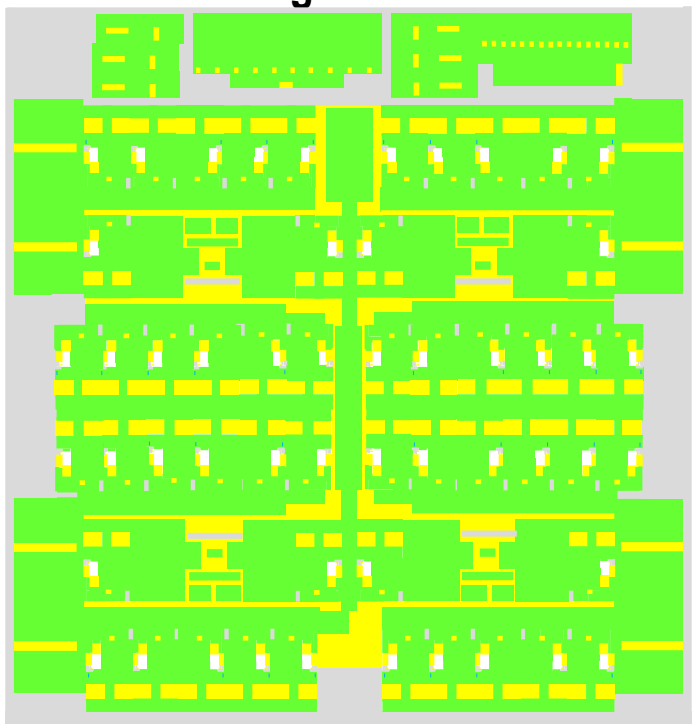
FUJITSU

- Large systems require extensive RAS capability of CPU and interconnect
- A64FX has a mainframe class RAS for integrity and stability.
  It contributes to very low CPU failure rate and high system stability
  - ✓ ECC or duplication for all caches
  - ✓ Parity check for execution units
  - ✓ Hardware instruction retry
  - ✓ Hardware lane recovery for Tofu links
  - ✓ **~128,400 error checkers** in total

<A64FX RAS Mechanism>

Units	Error Detection and Correction
Cache (Tag)	ECC, Duplicate & Parity
Cache (Data)	ECC, Parity
Register	ECC (INT), Parity(Others)
Execution Unit	Parity, Residue
Core	Hardware Instruction Retry
Tofu	Hardware Lane Recovery

<A64FX RAS Diagram>

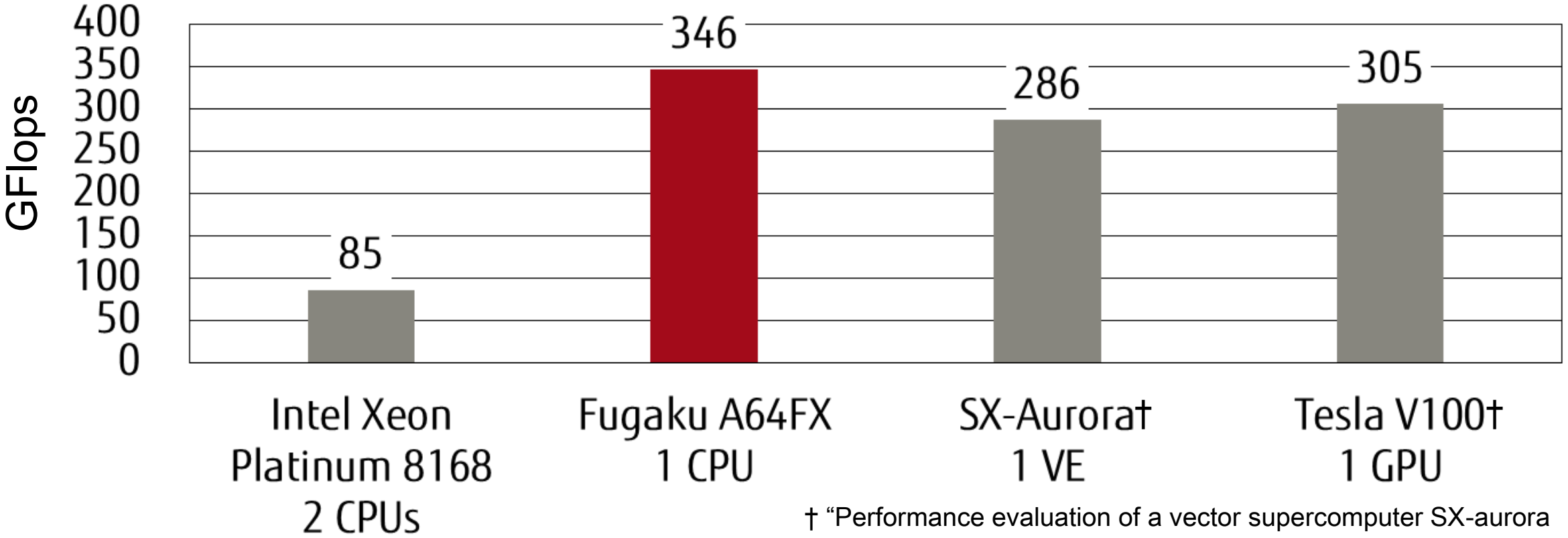


Green: 1 bit error Correctable
Yellow: 1 bit error Detectable
Gray : 1 bit error harmless

“Fugaku” CPU Performance Evaluation (2/3)

FUJITSU

- Himeno Benchmark (Fortran90)
  - Stencil calculation to solve Poisson’s equation by Jacobi method

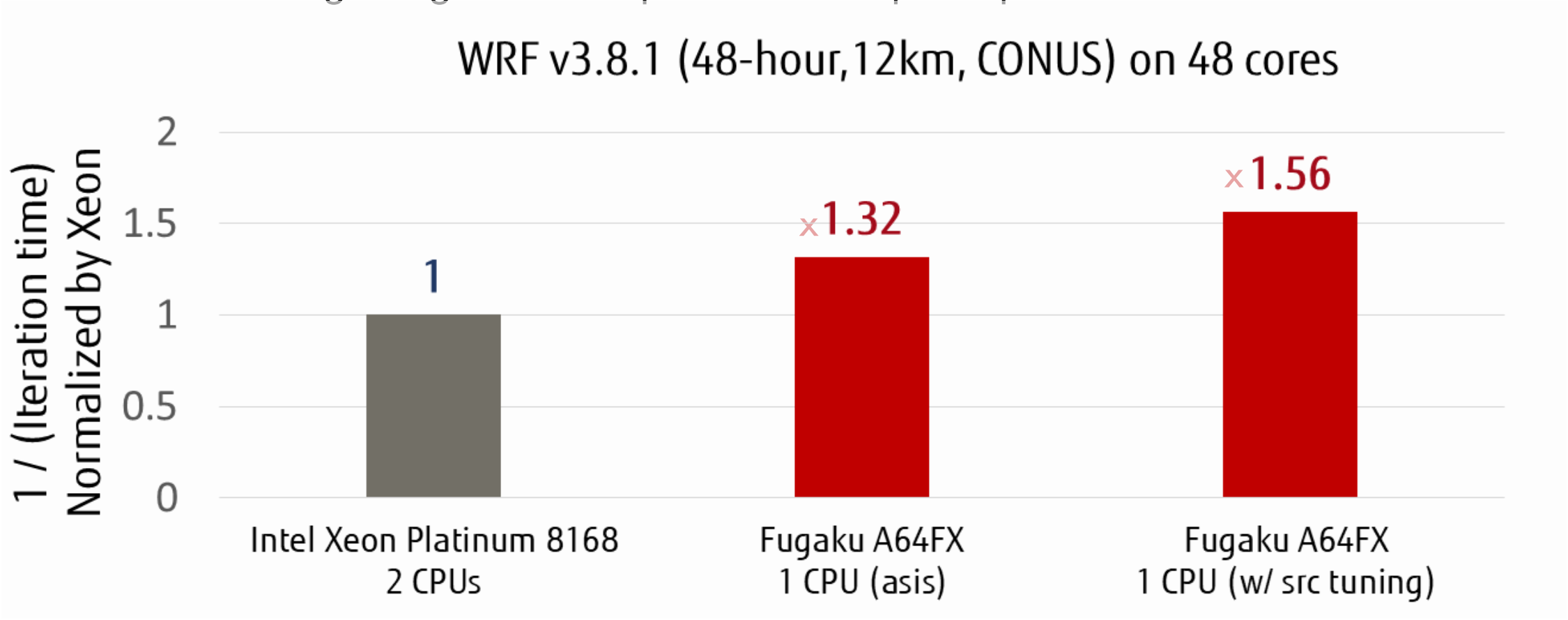


† “Performance evaluation of a vector supercomputer SX-aurora TSUBASA”, SC18, https://dl.acm.org/citation.cfm?id=3291728

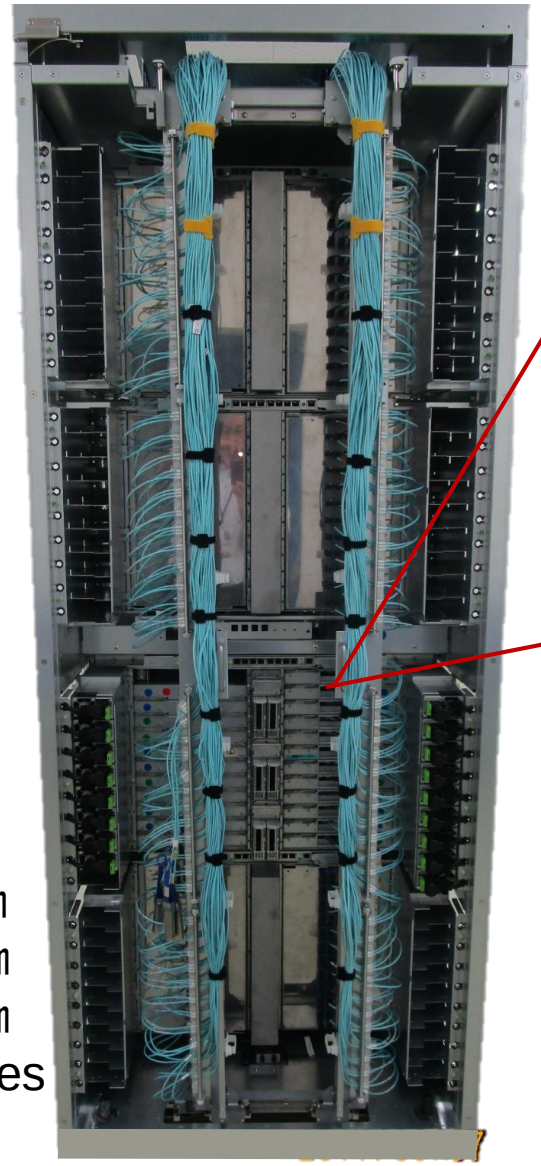
“Fugaku” CPU Performance Evaluation (3/3)

FUJITSU

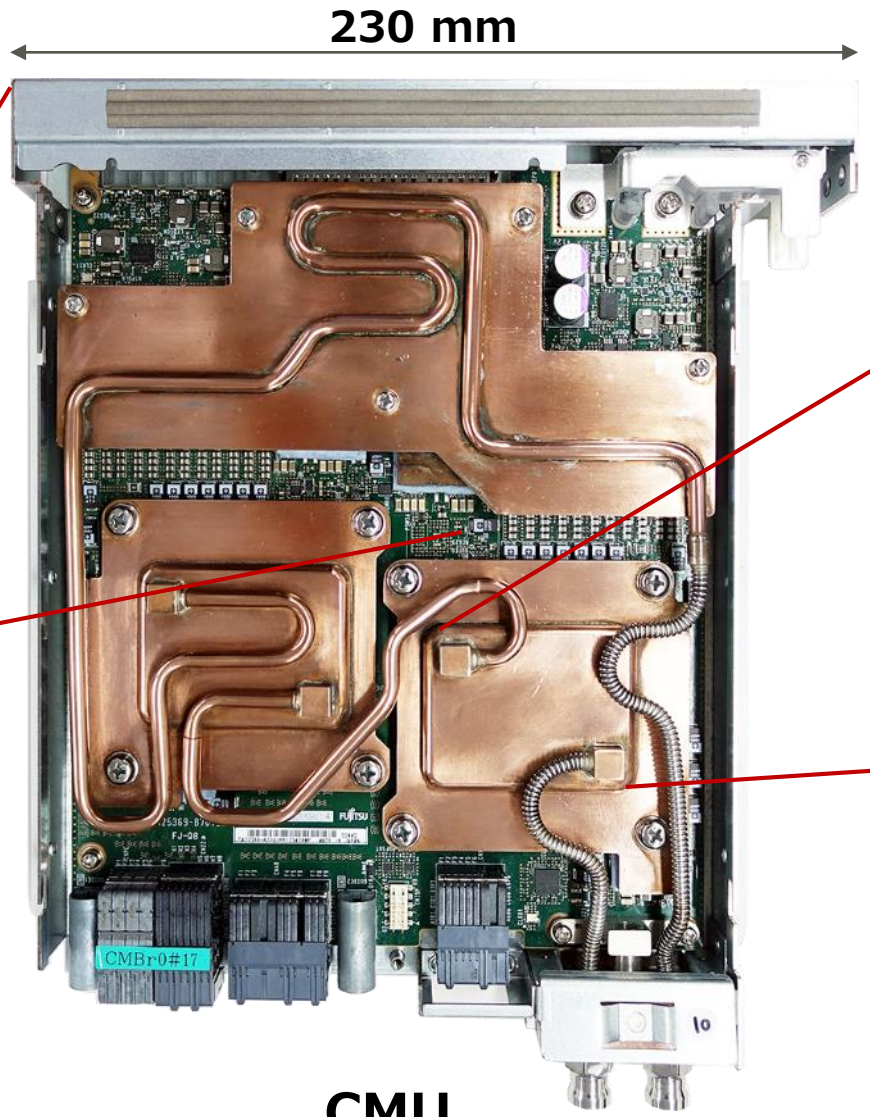
- WRF: Weather Research and Forecasting model
  - Vectorizing loops including IF-constructs is key optimization
  - Source code tuning using directives promotes compiler optimizations



Post-K Chassis, PCB (w/DLC), and A64fx CPU Package



W 800mm
D1400mm
H2000mm
384 nodes



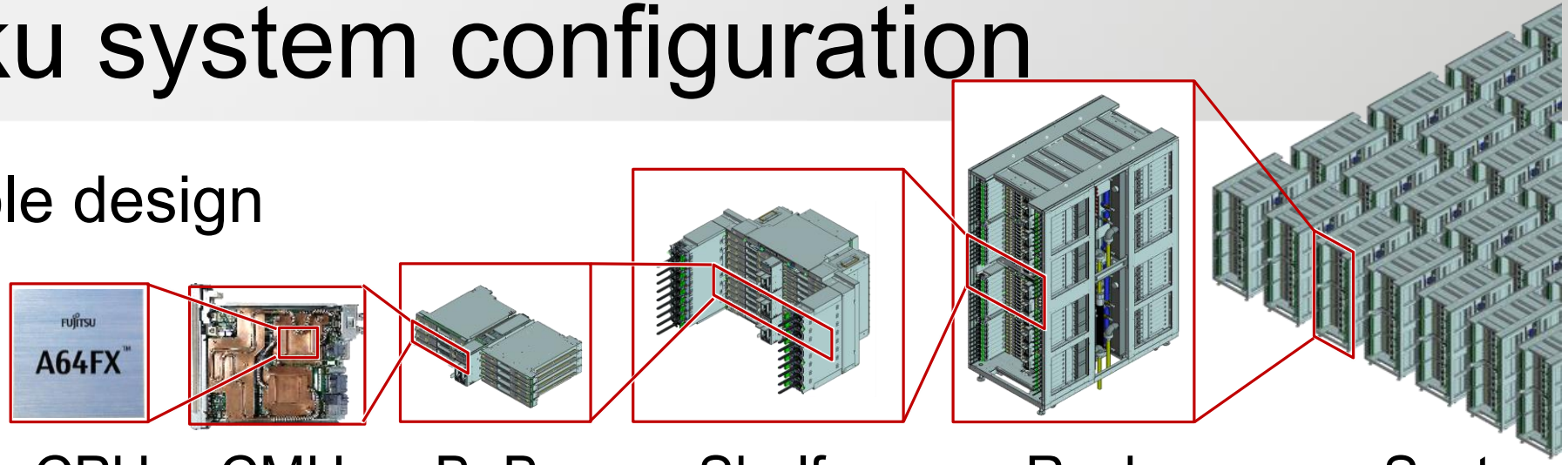
CMU

CPU Package

A0 Chip Booted in June
Undergoing Tests
B0 underway

Fugaku system configuration

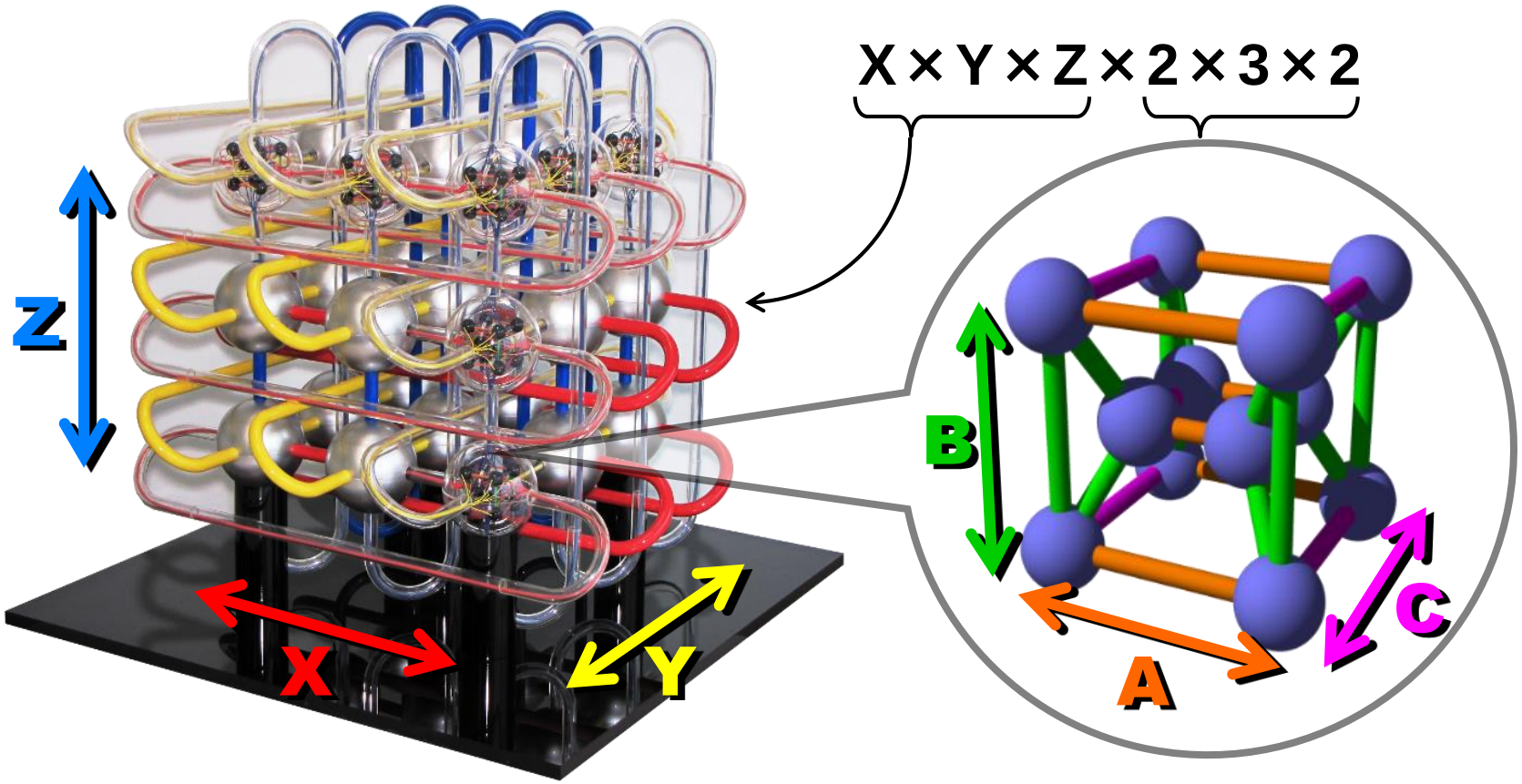
- Scalable design



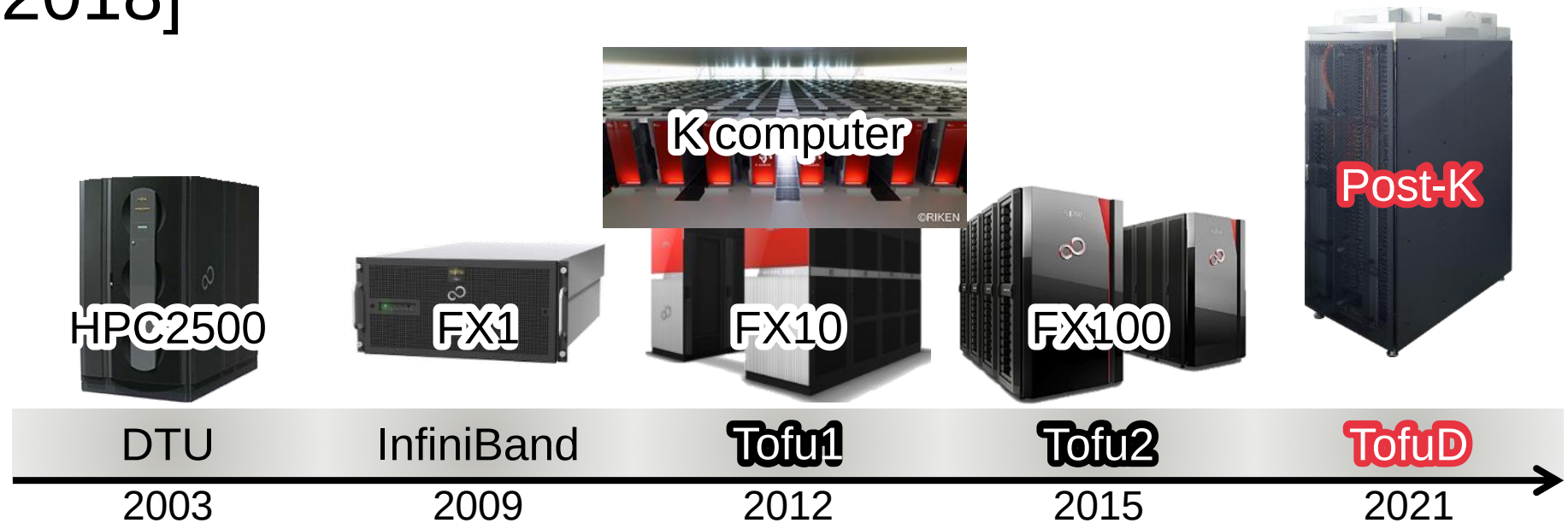
Unit	# of nodes	Description
CPU	1	Single socket node with HBM2 & Tofu interconnect D
CMU	2	CPU Memory Unit: 2x CPU
BoB	16	Bunch of Blades: 8x CMU
Shelf	48	3x BoB
Rack	384	8x Shelf
System	150k+	As a Fugaku system

6D Mesh/Torus Network

- Six coordinate axes: X, Y, Z, A, B, C
  - X, Y, Z: the size varies according to the system configuration
  - A, B, C: the size is fixed to 2 × 3 × 2
- Tofu stands for “torus fusion”: (X, Y, Z) × (A, B, C)



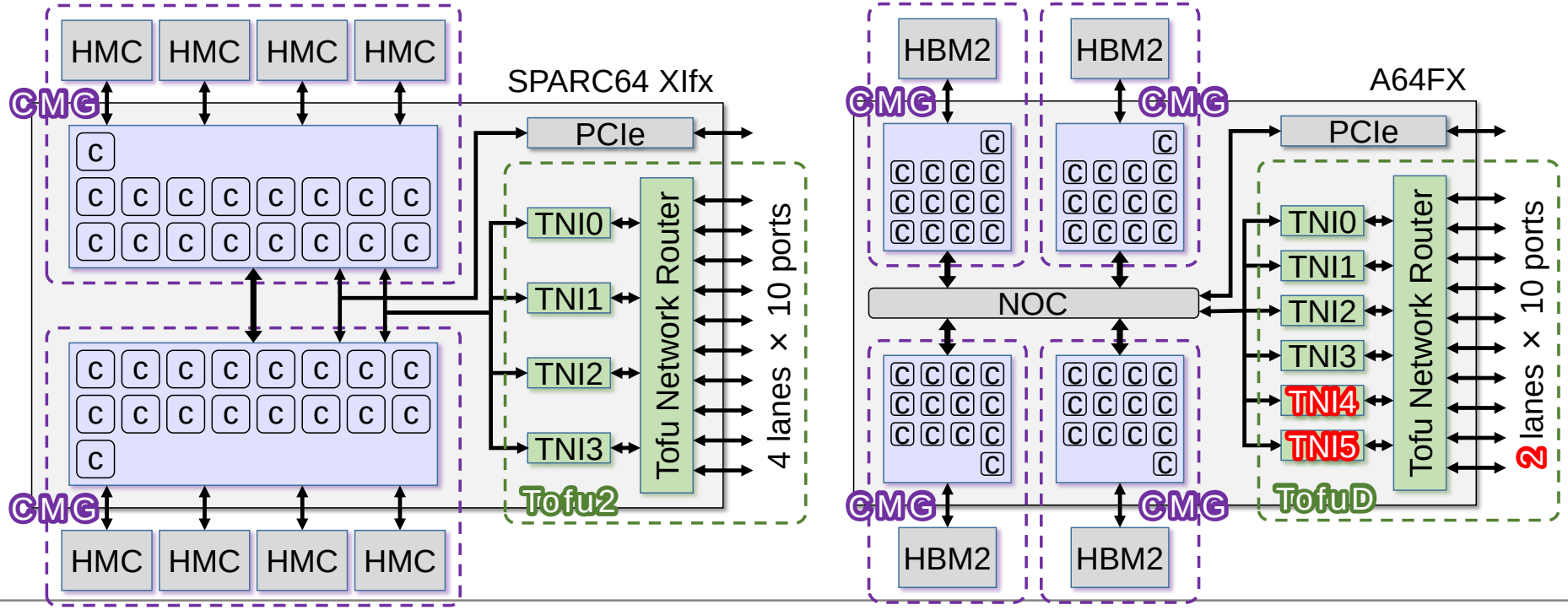
Fujitsu Ew6-D Tofu Network – Tofu D [Ajima et. al., IEEE Cluster 2018]



- The Tofu interconnect (Tofu1) for the K computer
  - Highly-scalable and fault-tolerant 6D mesh/torus network
- The Tofu interconnect 2 (Tofu2) for FX100 machines
- The Tofu Interconnect D (TofuD) for the post-K machine
  - High “density” of node: integrate more resources into a smaller node
  - Fault resilient of network: “dynamic” packet slicing for packet transfer

Higher-density Node Configuration

- The CPU is smaller and the off-chip channels are halved
  - The number of 3D-stacked memories was halved from 8 to 4
  - Each Tofu link was reduced from 4 lanes to 2 lanes
- More resources are integrated into the CPU
  - The number of CPU Memory Groups (NUMA nodes) doubled from 2 to 4
  - The number of Tofu Network Interfaces increased from 4 to 6



Put Latencies

- 8B Put transfer between nodes on the same board
  - The low-latency features were used

	Communication settings	Latency
Tofu1	Descriptor on main memory	1.15 µs
	Direct Descriptor	0.91 µs
Tofu2	Cache injection OFF	0.87 µs
	Cache injection ON	0.71 µs
TofuD	To/From far CMGs	0.54 µs
	To/From near CMGs	0.49 µs

0.20 µs

0.22 µs

- Tofu2 reduced the Put latency by 0.20 µs from that of Tofu1
  - The cache injection feature contributed to this reduction
- TofuD reduced the Put latency by 0.22 µs from that of Tofu2

Injection Rates per Node

FUJITSU

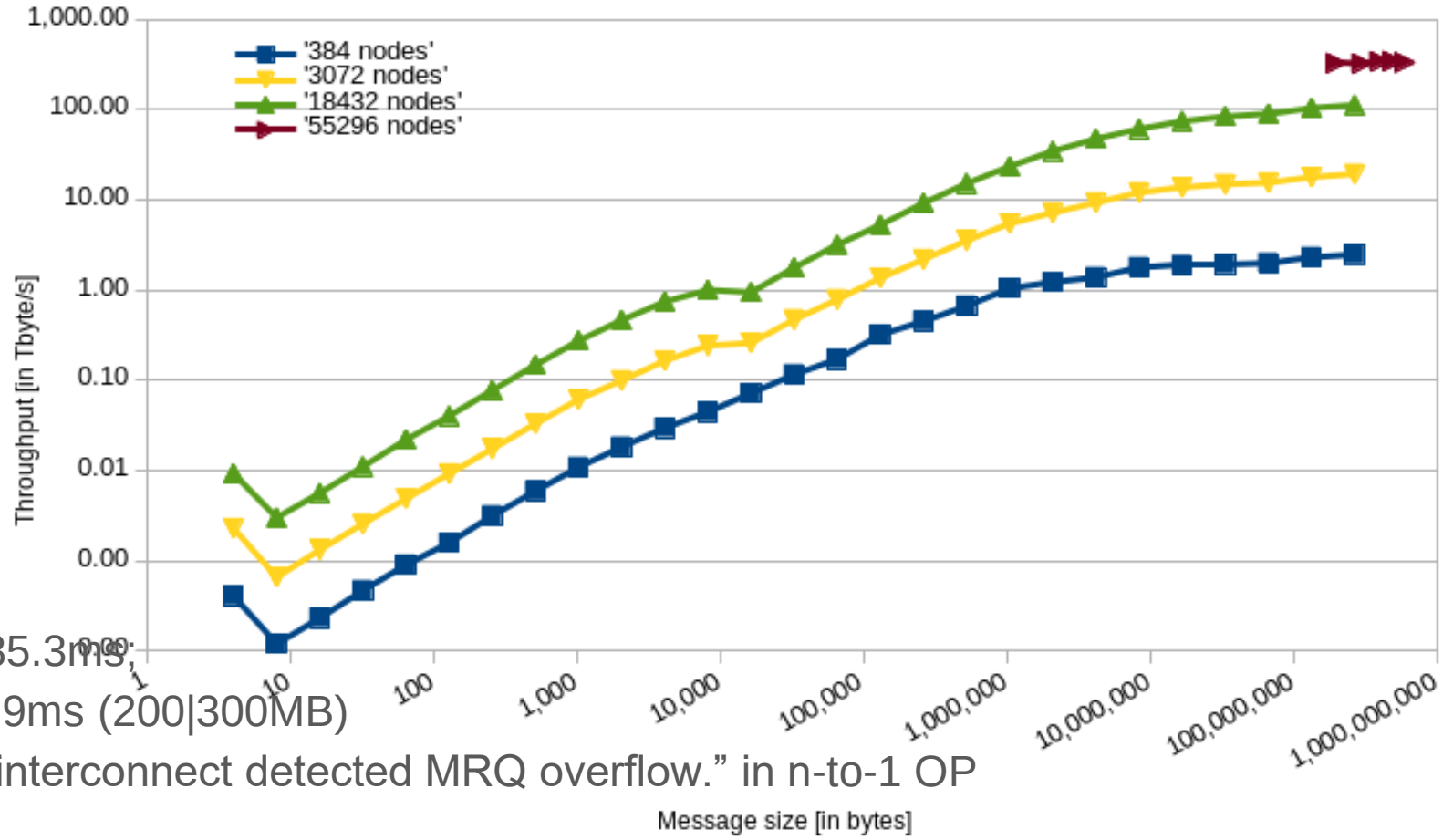
- Simultaneous Put transfers to multiple nearest-neighbor nodes
  - Tofu1 and Tofu2 used 4 TNIs, and TofuD used 6 TNIs

	Injection rate	Efficiency
Tofu1 (K)	15.0 GB/s	77 %
Tofu1 (FX10)	17.6 GB/s	88 %
Tofu2	45.8 GB/s	92 %
TofuD	38.1 GB/s	93 %

- The injection rate of TofuD was approximately 83% that of Tofu2
- The efficiencies of Tofu1 were lower than 90%
  - Because of a bottleneck in the bus that connects CPU and ICC
- The efficiencies of Tofu2 and TofuD exceeded 90 %
  - Integration into the processor chip removed the bottleneck

TOFU Allreduce Throughput on K

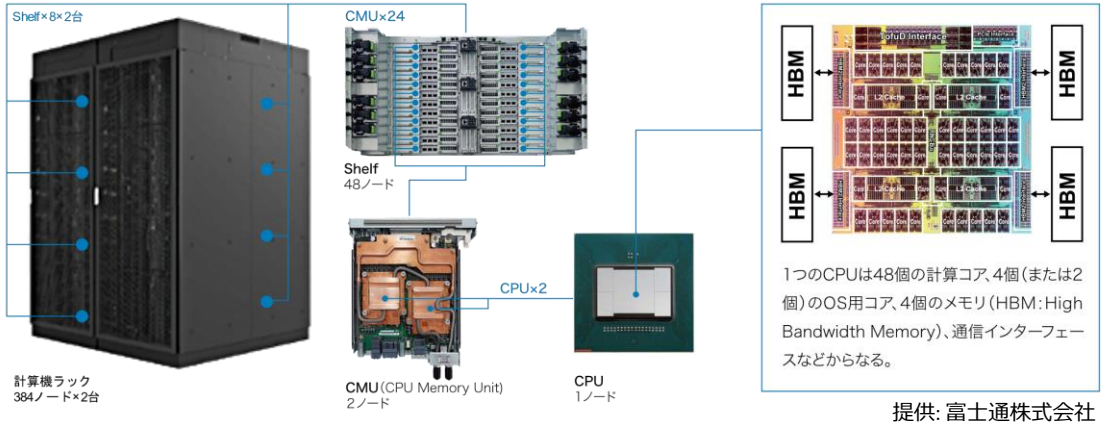
- Allreduce (& Bcast, Barrier, Reduce) nearly independent of #nodes ✓
- >340 TByte/s (for 400MB msg size) ✓
- Per 256MiB FP32 MPI_SUM allreduce:
  384 nodes: 83.3ms; 3072: 85.3ms;
  18k: 87.4ms; 55k: 66.6|100.9ms (200|300MB)
- x Full-K IMB crashed: “Tofu interconnect detected MRQ overflow.” in n-to-1 OP



Fugaku Total System Config & Performance

- **Total # Nodes: 158,976 nodes**
  - 384 nodes/rack x 396 (full) racks = 152,064 nodes
  - 192 nodes/rack x 36 (half) racks = 6,912 nodes

  c.f. K Computer 88,128 nodes
- **Theoretical Peak Compute Performances**
  - Normal Mode (CPU Frequency 2GHz)
    - 64 bit Double Precision FP: 488 Petaflops
    - 32 bit Single Precision FP: 977 Petaflops
    - 16 bit Half Precision FP (AI training): 1.95 Exaflops
    - 8 bit Integer (AI Inference): 3.90 Exaops
  - Boost Mode (CPU Frequency 2.2GHz)
    - 64 bit Double Precision FP: 537 Petaflops
    - 32 bit Single Precision FP: 1.07 Exaflops
    - 16 bit Half Precision FP (AI training): 2.15 Exaflops
    - 8 bit Integer (AI Inference): 4.30 Exaops
- **Theoretical Peak Memory Bandwidth: 163 Petabytes/s**



提供: 富士通株式会社

- C.f. K Computer performance comparison (Boost)
  - 64 bit Double Precision FP: 48x
  - 32 bit Single Precision: 95x
  - 16 bit Half Precision (AI training): 190x
    - K Computer Theoretical Peak: 11.28 PF for all precisions
  - 8 bit Integer (AI Inference): > 1,500x
    - K Computer Theoretical Peak: 2.82 Petaops (64 bits)
  - Theoretical Peak Memory Bandwidth: 29x
    - K Computer Theoretical Peak: 5.64 Petabytes/s

Fugaku is a Year's worth of IT in Japan

	Smartphones		IDC Servers incl Clouds		Fugaku		K Computer
Units	**20 million** (~annual shipments in Japan)	=	**300,000** (~annual shipments in Japan)	=	**1**		**30~100**
Power	10W×20 mil = **200MW**	=	600-700W x 30K= **200MW** (incl cooling)	> >	**30MW**		**15MW**
CPU ISA System SW	**Arm iOS/ Android Linux**		**x86/Arm Linux (Red Hat etc.)/Win**		**Arm Linux (Red Hat etc.)**		**Sparc Proprietary Linux Low generality**
AI Acceleration	**Custom ASIC Inference Only**		**Gen. Purpos Accelerator e.g. GPU**		**Gen. CPU SVE instructions**		**None**

Green500, Nov. 2019

A64FX prototype –
Fujitsu A64FX 48C 2GHz
ranked #1 on the list

768x general purpose A64FX CPU w/o accelerators

- 1.9995 PFLOPS @ HPL, 84.75%
- 16.876 GF/W
- Power quality level 2

The GREEN 500

HOME GREEN500 LISTS RESOU

Home / Lists / November 2019

NOVEMBER 2019

- The most energy-efficient system and No. 1 on the Green500 is a new Fujitsu A64FX prototype installed at Fujitsu, Japan. It achieved 16.9 GFlops/Watt power-efficiency during its 2.0 Pflop/s Linpack performance run. It is listed on position 160 in the TOP500.
- In second position is the NA-1 system, a PEZY Computing / Exascaler Inc. system which is currently being readied at PEZY Computing, Japan for a future installation at NA Simulation in Japan. It achieve 16.3 GFlops/Watt power efficiency. It is on position 421 in the TOP
- The No 3 on the Green500 is AiMOS, a new IBM Power systems at the Rensselaer Polytechnic Ins Computational Innovations (CCI), New York, USA. It achieved 15.8 GFlops/Watt and is listed at pos

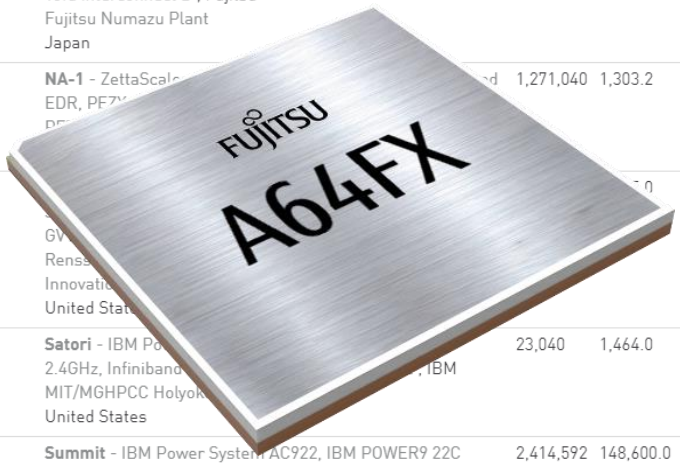

富岳 省エネ世界一
スパコン「京」の後継
富士通と理化学研究所は18日、スーパーコンピューター「京」の後継となる「富岳」の試作機が、スパコンの省エネ性能のランキング「グリーン500」で世界1位になったと発表した。消費電力1ワット当たり毎秒16.8億回の計算を達成した。
最新鋭の中央演算処理装置（CPU）＝イメージ図・富士通提供＝を768個搭載し、装置間の通信を効率的にするソフトを開発したことが奏功した。
富岳は2021年ごろの運用開始を予定している。

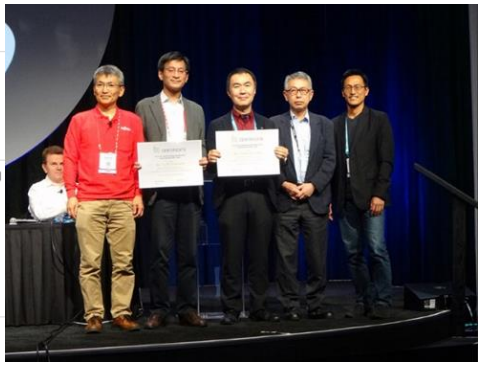
Green500 List for November 2019

Listed below are the November 2019 The Green500's energy-efficient supercomputers ranked from

Note: Shaded entries in the table below mean the power data is derived and not meassured.

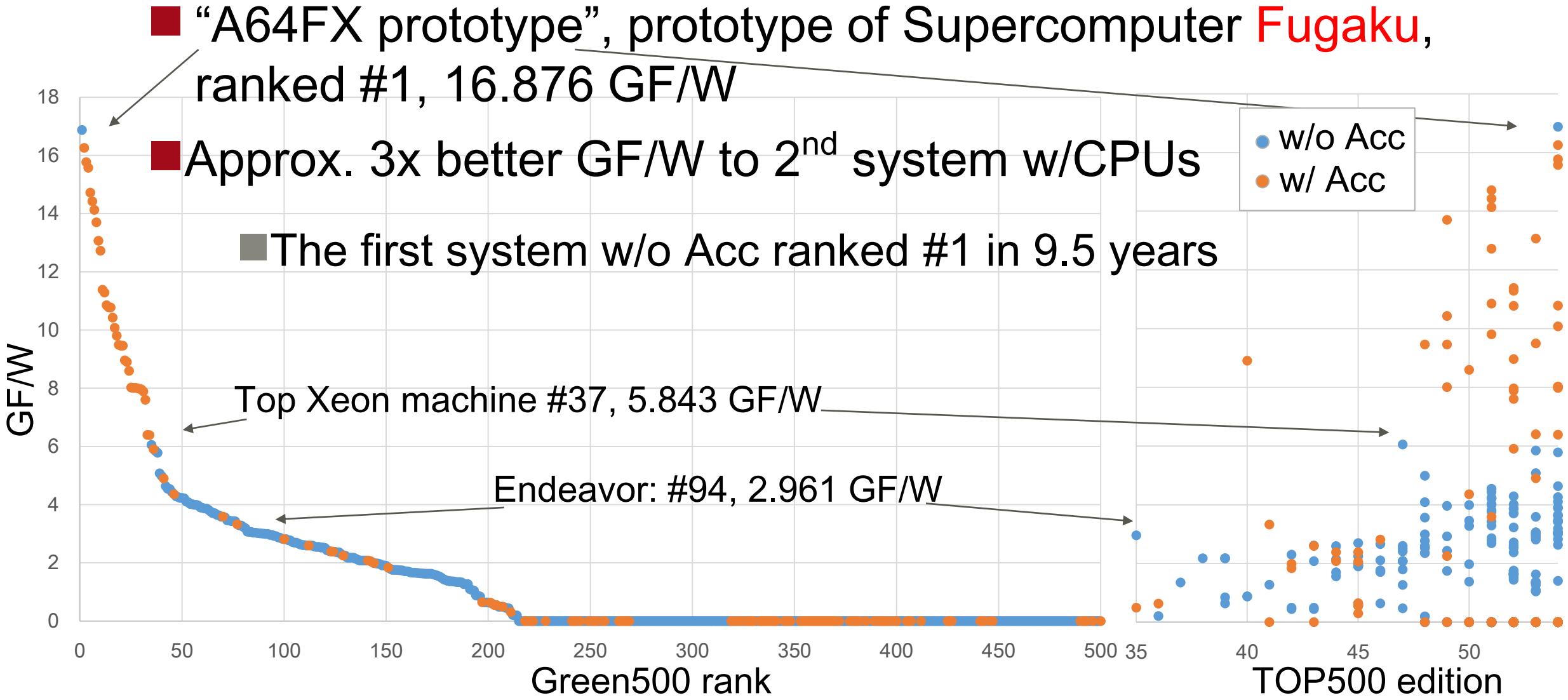
Rank	TOP500 Rank	System	Cores	Rmax (TFlop/s)	Power (kW)	Power Efficiency (GFlops/watts)
1	159	**A64FX prototype** - Fujitsu A64FX, Fujitsu A64FX 48C 2GHz, Tofu interconnect D , Fujitsu Fujitsu Numazu Plant Japan	36,864	1,999.5	118	16.876
2	420	**NA-1** - ZettaScal... EDR, PEZ...	1,271,040	1,303.2	80	
3	24	... GV... Rens... Innovati... United Stat...			510	
4	373	**Satori** - IBM Po... 2.4GHz, Infiniband... MIT/MGHPCC Holyok... United States	23,040	1,464.0	94	
5	1	**Summit** - IBM Power System AC922, IBM POWER9 22C 3.07GHz, NVIDIA Volta GV100, Dual-rail Mellanox EDR	2,414,592	148,600.0	10,096	14,719





SC19 Green500 ranking and 1st appeared TOP500 edition

FUJITSU



Fugaku Performance Estimate on 9 Co-Design Target Apps

Performance target goal

- ✓ 100 times faster than K for some applications (tuning included)
- ✓ 30 to 40 MW power consumption

Peak performance to be achieved

	PostK	K
Peak DP (double precision)	>400+ Pflops (34x +)	11.3 Pflops
Peak SP (single precision)	>800+ Pflops (70x +)	11.3 Pflops
Peak HP (half precision)	>1600+ Pflops (141x +)	--
Total memory bandwidth	>150+ PB/sec (29x +)	5,184TB/sec

Geometric Mean of Performance Speedup of the 9 Target Applications over the K-Computer

> 37x+

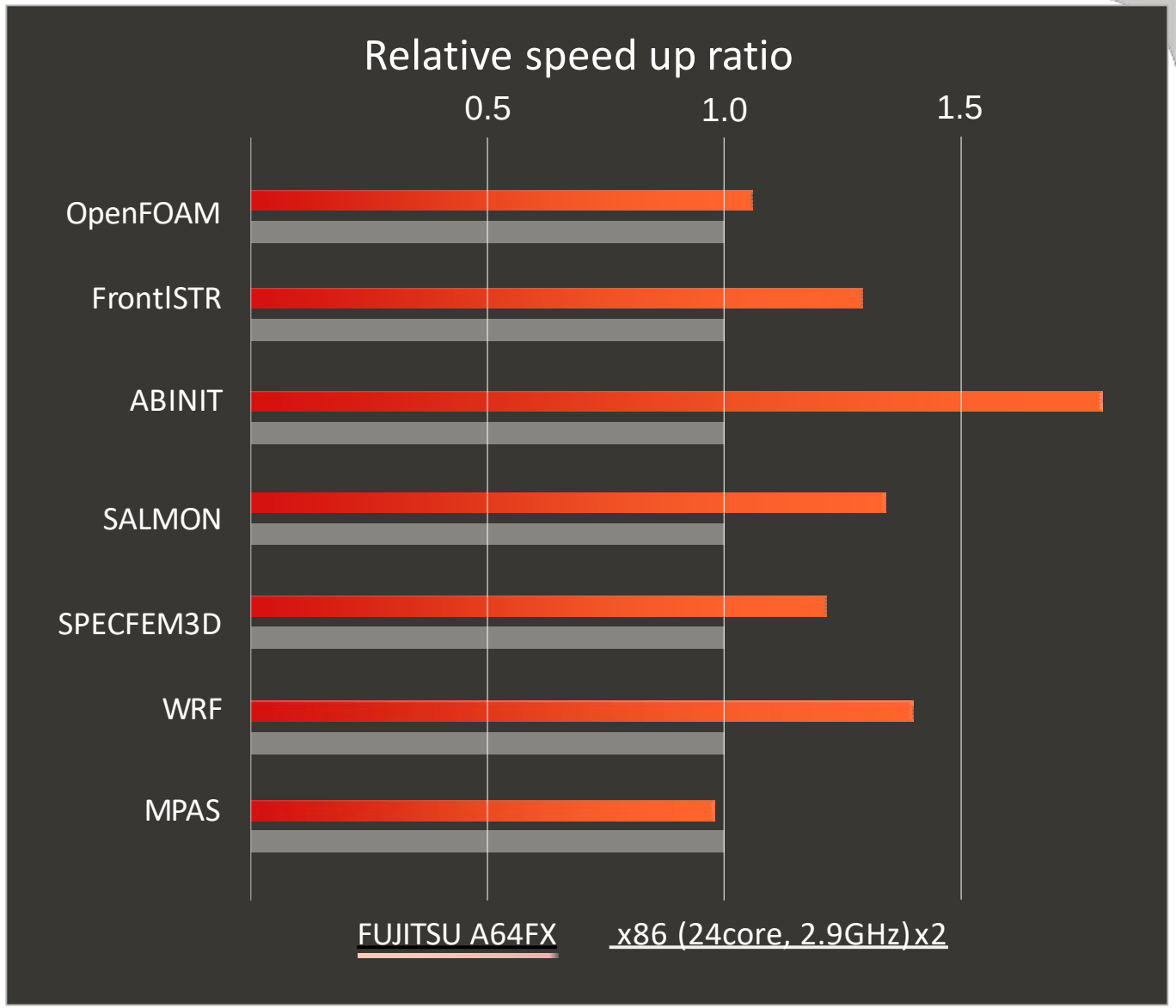
As of 2019/05/14

Category	Priority Issue Area	Performance Speedup over K	Application	Brief description
Health and longevity	1. Innovative computing infrastructure for drug discovery	**125x +**	GENESIS	MD for proteins
	2. Personalized and preventive medicine using big data	8x +	Genomon	Genome processing (Genome alignment)
Disaster prevention and Environment	3. Integrated simulation systems induced by earthquake and tsunami	45x +	GAMERA	Earthquake simulator (FEM in unstructured & structured grid)
	4. Meteorological and global environmental prediction using big data	**120x +**	NICAM+ LETKF	Weather prediction system using Big data (structured grid stencil & ensemble Kalman filter)
Energy issue	5. New technologies for energy creation, conversion / storage, and use	40x +	NTChem	Molecular electronic simulation (structure calculation)
	6. Accelerated development of innovative clean energy systems	35x +	Adventure	Computational Mechanics System for Large Scale Analysis and Design (unstructured grid)
Industrial competitiveness enhancement	7. Creation of new functional devices and high-performance materials	30x +	RSDFT	Ab-initio simulation (density functional theory)
	8. Development of innovative design and production processes	25x +	FFB	Large Eddy Simulation (unstructured grid)
Basic science	9. Elucidation of the fundamental laws and evolution of the universe	25x +	LQCD	Lattice QCD simulation (structured grid Monte Carlo)

A64FX CPU performance evaluation for real apps

FUJITSU

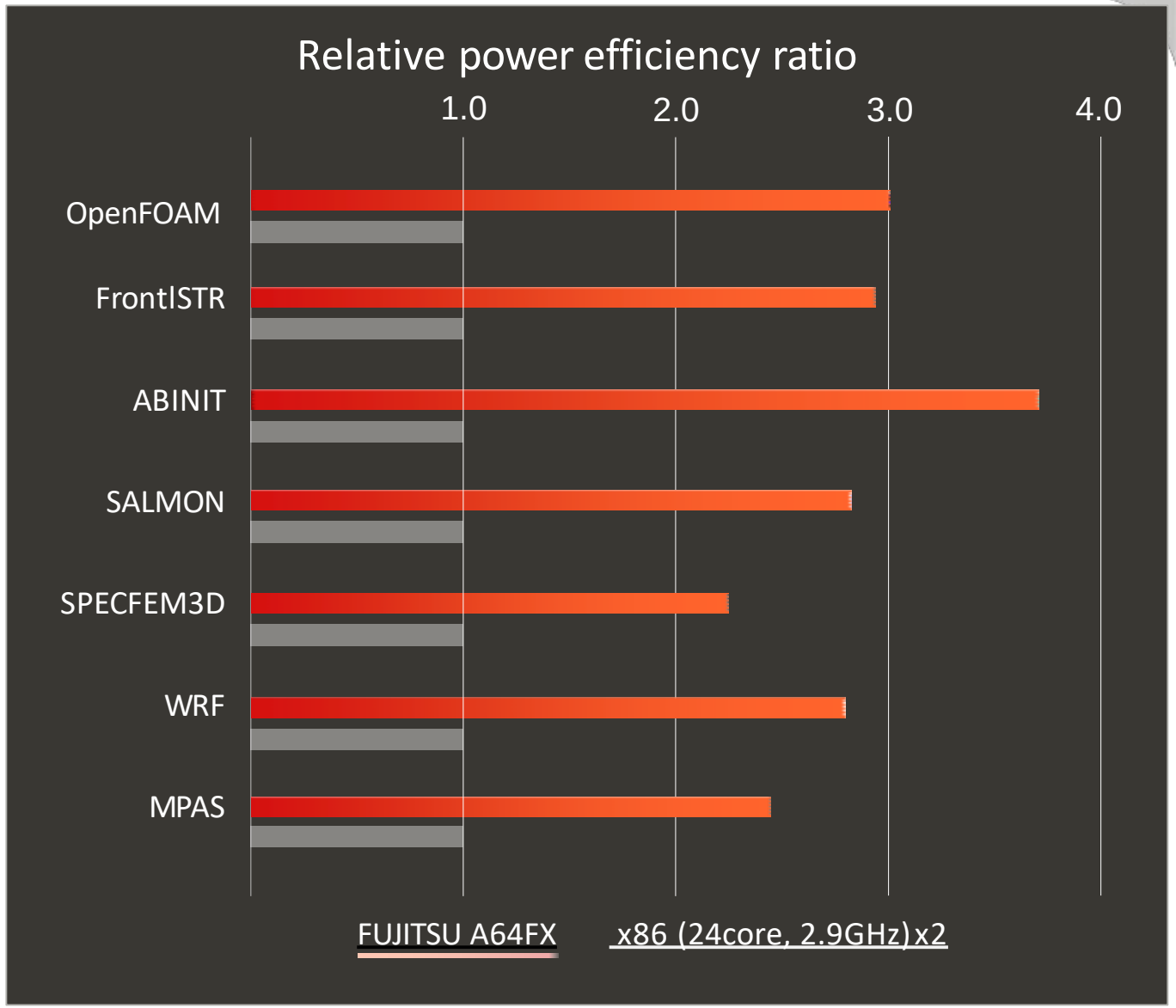
- Open source software, Real apps on an A64FX @ 2.2GHz
- Up to 1.8x faster over the latest x86 processor (24core, 2.9GHz) x 2, or 3.6x per socket
- High memory B/W and long SIMD length of A64FX work effectively with these applications



A64FX CPU power efficiency for real apps

FUJITSU

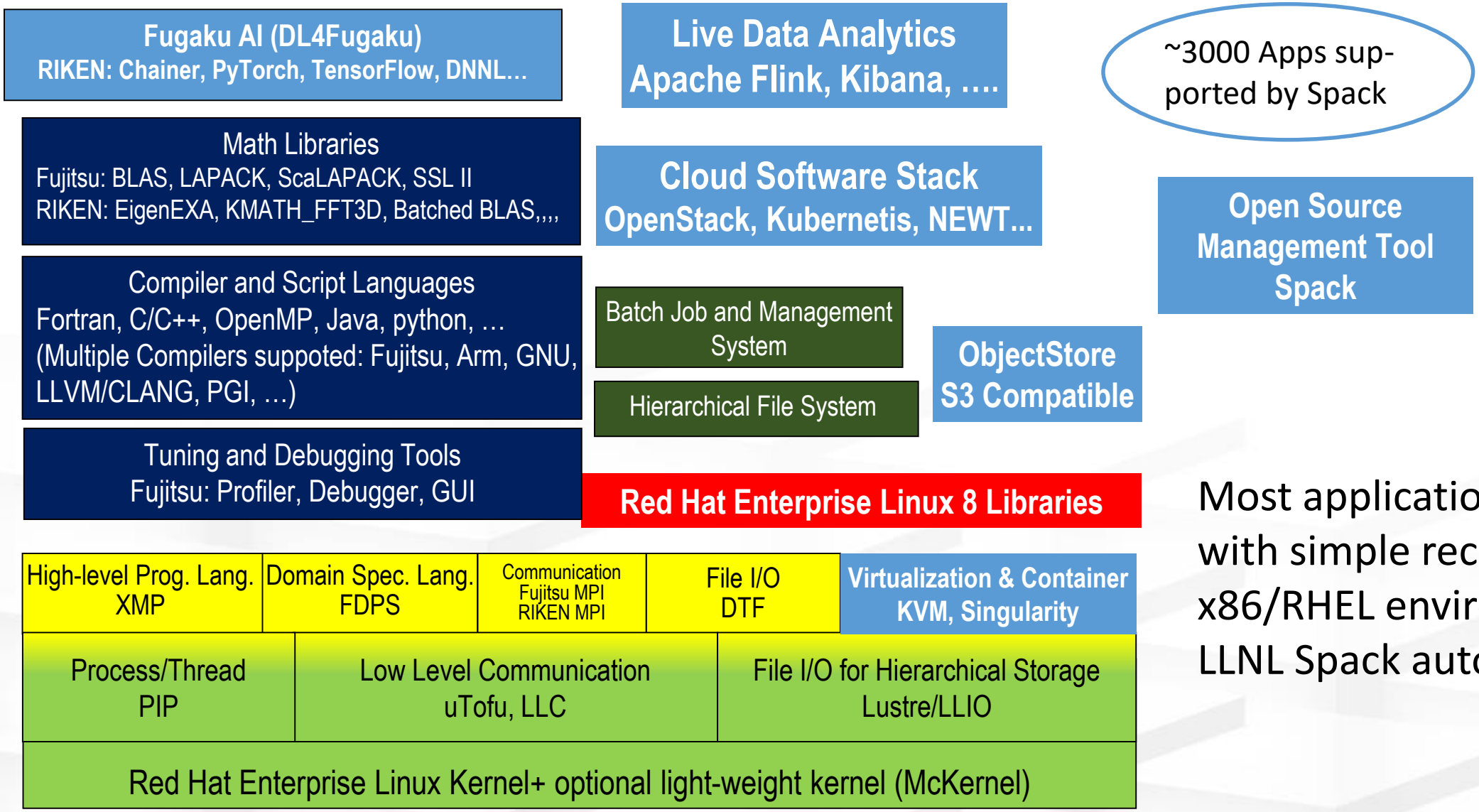
- Performance /Energy consumption on an A64FX @ 2.2GHz
- Up to 3.7x more efficient over the latest x86 processor (24core, 2.9GHz) x2
- High efficiency is achieved by energy-conscious design and implementation



Fugaku Deployment Status (July 2020)

- Pipelined manufacturing, installation, and bringup, first rack shipped on Dec 3 2019.
- All racks on the floor May 13, 2020(!)
- 2020 early users, incl. COVID-19 apps running already
- Open to international users through HPCI, general allocation April 2021 (application starting Sept. 2020) (does NOT need to involve a Japanese PI)
- Also some internal test nodes (Apr 2020) and allocations (Apr. 2021) are available for R-CCS

Fugaku / Fujitsu FX1000 System Software Stack



Most applications will work with simple recompile from x86/RHEL environment.
LLNL Spack automates this.

New PRIMEHPC Lineup

FUJITSU

PRIMEHPC FX1000

Supercomputer optimized for large scale computing

High Scalability | High Density | Superior power efficiency

A64FX processor
384 nodes/Rack
Tofu-D Interconnect

PRIMEHPC FX700

Supercomputer based on

Ease to use | Installation

A64FX Processor
8 nodes/2U Rackmount

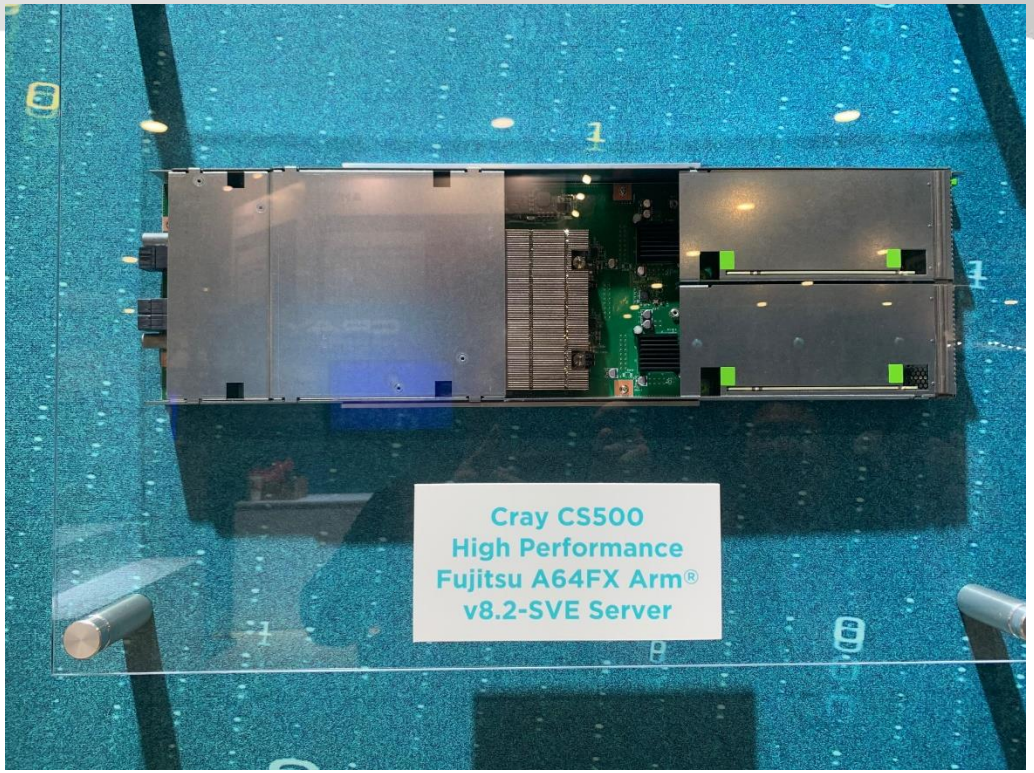
The HPE/Cray CS500 - Fujitsu A64FX Arm-based Server

FUJITSU

- Cray Fujitsu Technology Agreement
- Supported in Cray CS500 infrastructure

- Full Cray Programming Environment (note: HPE product, support only by HPE)

- Leadership performance for many memory intensive HPC applications, e.g., weather
- GA in mid'2020
- A number of adoptions
  US: Stony Brook, DoE Labs, etc.
  Multiple yet-to-be-named EU centers



Fugaku / FX1000 / FX700 Commercial Apps

(as of June 2020)

Available soon

Engineering (Structural analysis, Fluid dynamics and Electronics)

- LS-DYNA (by Ansys, Inc.)
- Poynting (by Fujitsu Limited)

Chemistry*

- Amber
- Gaussian16 (by Gaussian, Inc.)

*Collaboration with Australian National University

In a research & development phase

- ADVENTURECluster (by Allied Engineering Co.)
- CONVERGE CFD SOFTWARE (by Convergent Science)
- Marc (by MSC Software Ltd.)
- VASP
- Altair Radioss™ (by Altair Engineering, Inc.)
- HELYX (by ENGYS Ltd. & VINAS Co., Ltd.)
- scFLOW (by Software Cradle Co., Ltd.)
- VPS (PAM-CRASH) (by ESI Group)
- Ansys FLUENT (by Ansys, Inc.)
- JMAG (by JSOL Corporation)
- Simcenter STAR-CCM+ (by Siemens Industry Software Inc.)

Cloud Service Providers Partnership

https://www.r-ccs.riken.jp/library/topics/200213.html (in Japanese)

Action Items

- Cool Project name and logo!
- Trial methods to provide computing resources of Fugaku to end-users via service providers
- Evaluate the effectiveness of the methods quantitatively as possible and organize the issues
- The knowledges gained will be feedbacked to scheme design of Fugaku by the government

FUJITSU

shaping tomorrow with you

TOP500 & HPCG

2020.6.21

FUJITSU LIMITED

Results of ISC2020 Fugaku Rankings at a Glance

FUJITSU

- Fugaku ranked #1 by large margin in ALL performance benchmarks

Benchmark	Unit	#1	Score	#2	Score	#1/#2
TOP500	PFLOPS	Fugaku	416	Summit	148.6	2.80
HPCG	PFLOPS	Fugaku	13.4	Summit	2.93	4.57
HPL-AI	EFLOPS	Fugaku	1.42	Summit	0.55	2.58
Graph500	GTEPS	Fugaku	70,980	TaihuLight	23,756	2.99

Note: all the benchmarks on Fugaku were not conducted on the full machine due to time constraints; subsequent submissions may further improve performance in the future

Fugaku and A64FX Greenness on TOP500

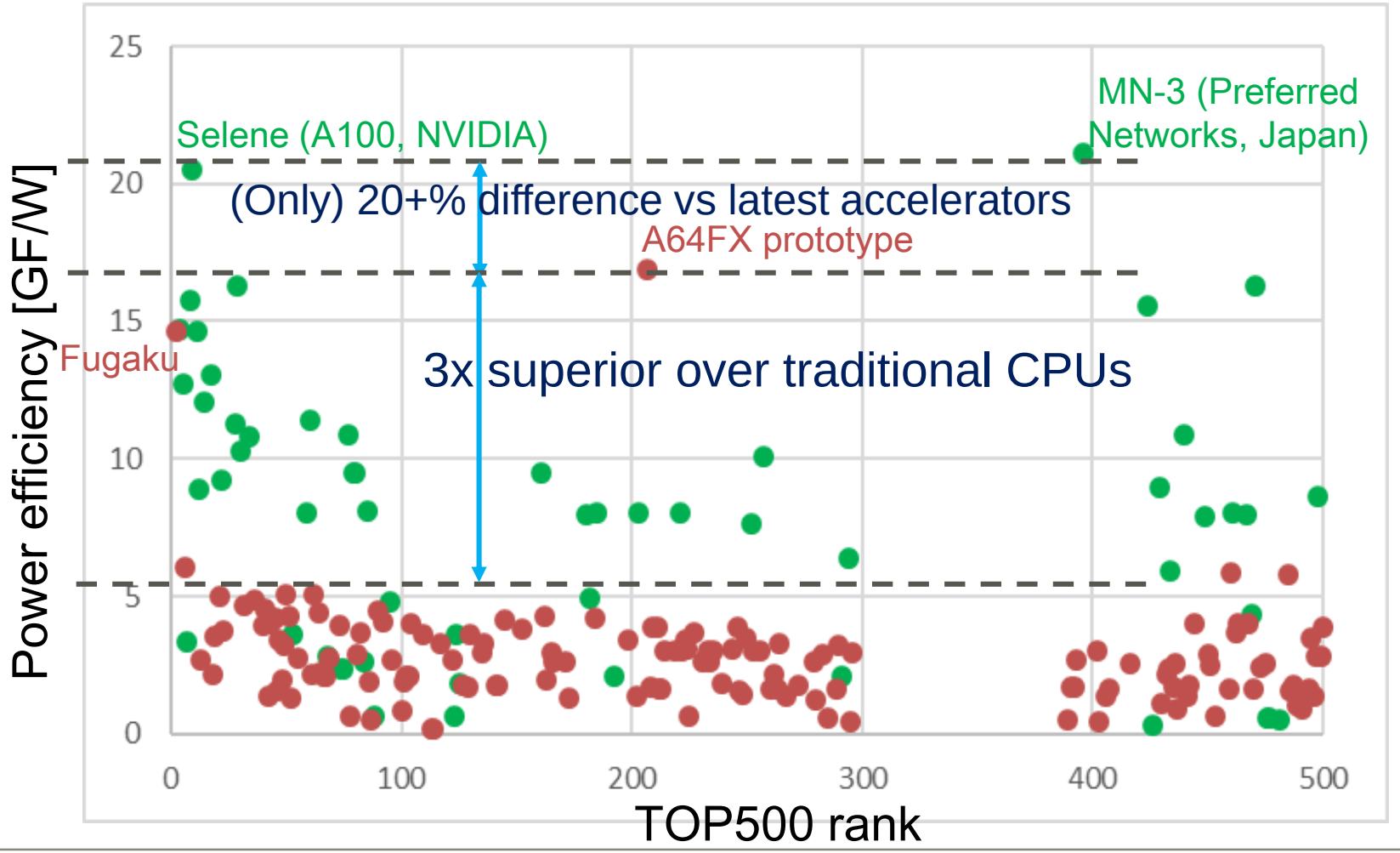
FUJITSU

- Power efficiency in GF/W, **w/ ACC** and **w/o ACC**

Way for perf.	Accelerator core	CPU core
GF/W improvement	Easier	Not easy
Apps development	Not easy	Easier
Apps domains	Narrow	Wider

Fugaku's choice

A64FX demonstrating power efficiency comparable to latest accelerators, and 3x superiority c.f. traditional CPUs

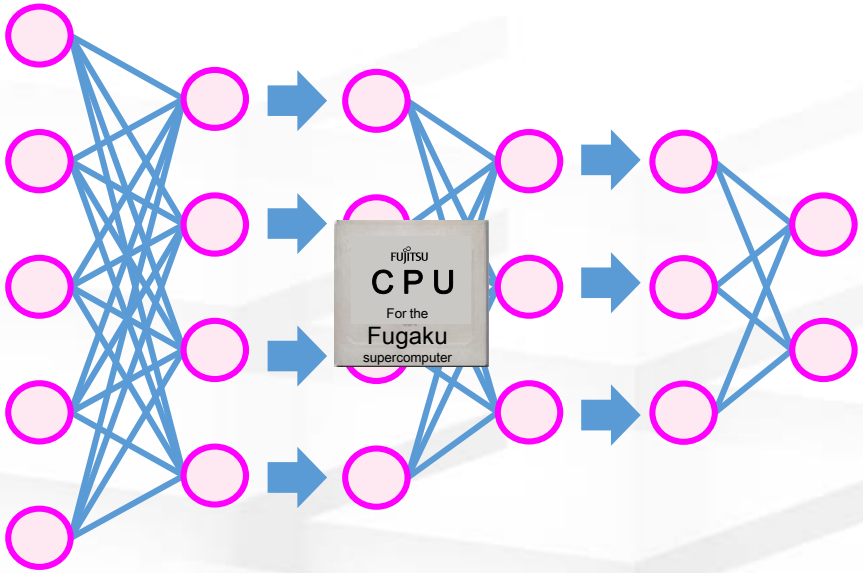


Massive Scale Deep Learning on Fugaku

Fugaku Processor is AI-DL ready

- High perf FP16&Int8
- **High mem BW for convolution**
- **Built-in scalable Tofu network**

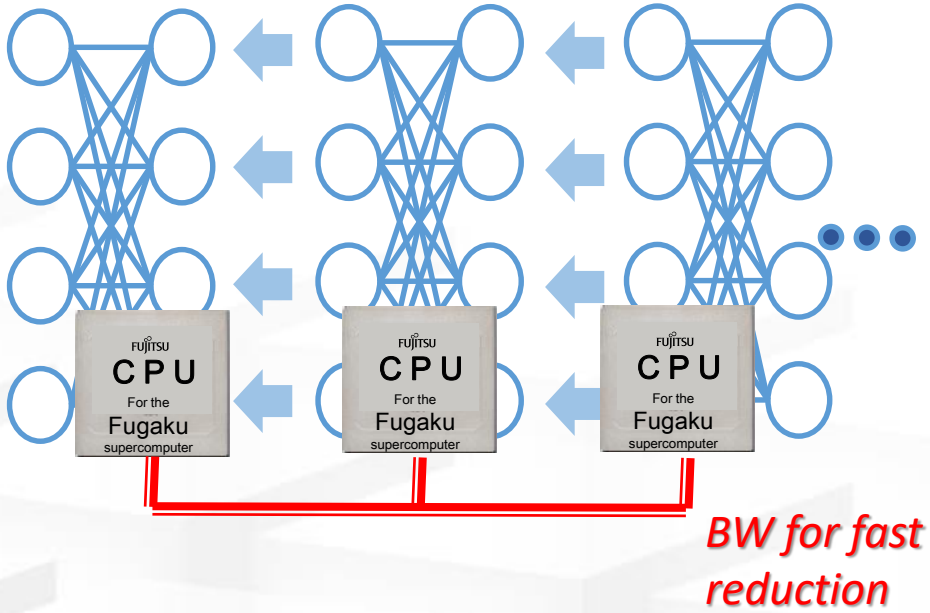
High Performance DNN Convolution



Low Precision ALU + High Memory Bandwidth + Advanced Combining of Convolution Algorithms (FFT+Winograd+GEMM)

Unprecedened DL scalability

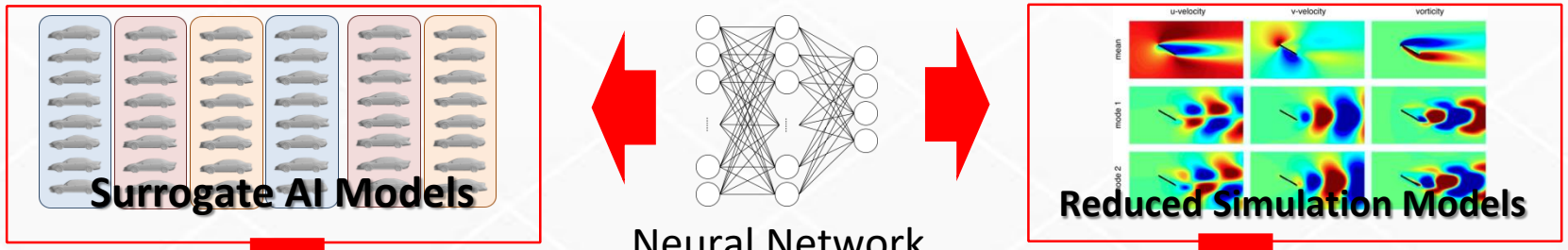
High Performance and Ultra-Scalable Network for massive scaling model & data parallelism



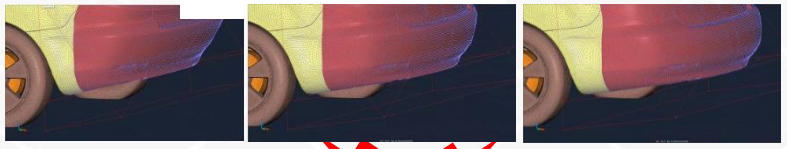
Ultra Scalability of Data/Model Parallelism
Scalability shown to 20,000 nodes
HPL-AI to beyond exaflops

HPC and AI Convergence for Society 5.0 Manufacturing [Tsubokura et. al., R-CCS]

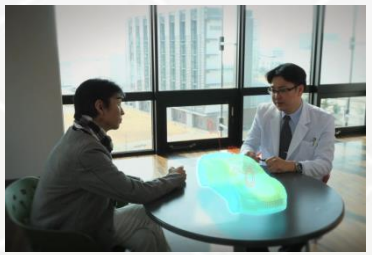
- **Combining ML/Deep Learning, Data Assimilation, Multivariate Optimization with Simulation for new generation manufacturing**
- **Use output of high-resolution simulation data to train AI**
  - Construct AI surrogate model training on simulation data, allowing real-time CFD to facilitate designer-engineer collaboration, multivariate design optimization, etc.
  - Use NN to derive reduced simulation model, allowing digital twin in cyberspace corresponding to entities in physical space for real-time interactions



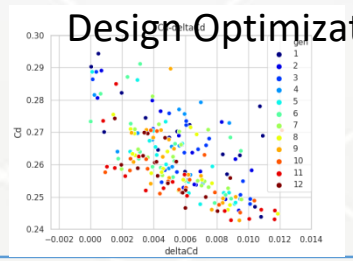
Real-Time CFD



Designer-Engineer Collaboration



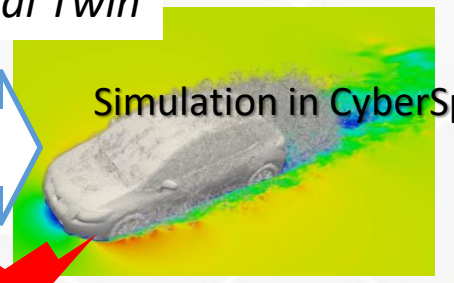
Multivariate Design Optimization



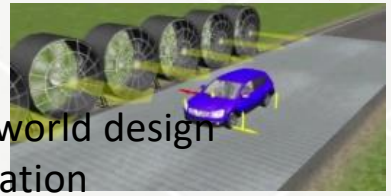
Digital Twin

Physical Space

Simulation in CyberSpace



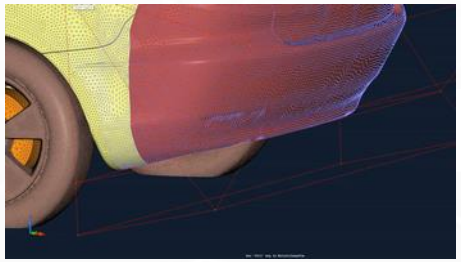
Real-world design evaluation

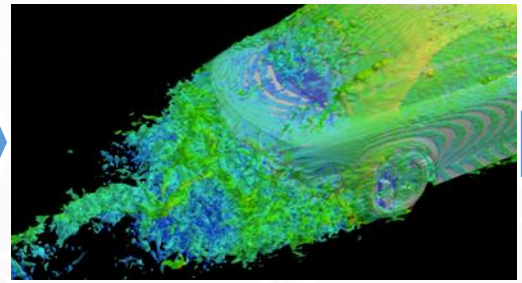


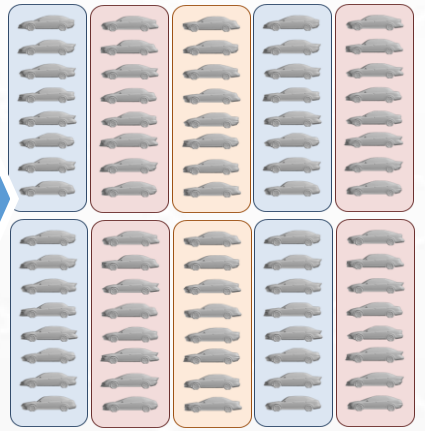
Auto-driving AI Training

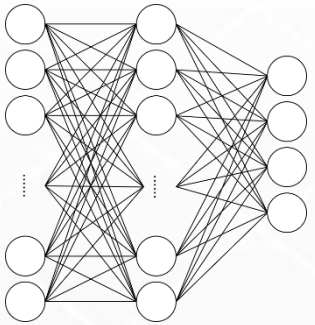
Example: Automotive Multivariate CFD Design using HPC & AI

- **Train AI with Simulation Inputs to create AI Surrogate Model**
  - Hundreds of CFD simulations on variable car designs to generate training input data
  - Surrogate NN: body design input => Aerodynamic properties



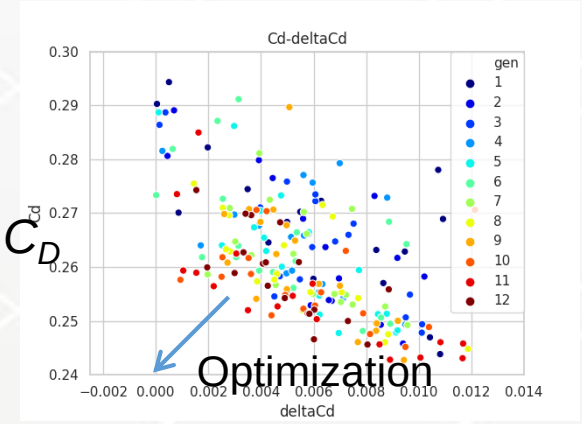






Max epoch	10000
Batch size	100

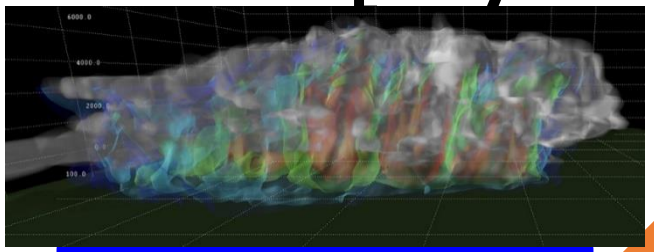
- **Multivariate optimization w/genetic algorithm**
  - Derive multiple optimal car designs rapidly



Fuel Efficient Design

Wind shear Resistant Design

Revolutionizing Weather Prediction for Tokyo Olympics [Miyoshi et. al., R-CCS]



1-h-lead downpour forecast
refreshed every 30 seconds
at 100-m mesh

BDA BIG DATA ASSIMILATION

Cyberspace

synchronize
predict & control

Real world

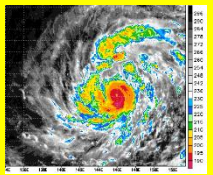
Simulation

1101011
1101011
1101011

Big Data Assimilation

Data Assimilation

Big Data



AI x HPC

Sensing

Nature

Prepared for weather

IoT

Human society and economy

Large Scale Public AI Infrastructures in Japan

	Deployed	Purpose	AI Processor	Inference Peak Perf.	Training Peak Perf.	HPL-AI Perf	Top500 Perf/Rank	Green500 Perf/Rank
Tokyo Tech. TSUBAME3	July 2017	HPC + AI Public	NVIDIA P100 x 2160	45.8 PF (FP16)	22.9 PF / 45.8PF (FP32/FP16)		8.125 PF #22	13.704 GF/W #8
U-Tokyo Reedbush-H/L	Apr. 2018 (update)	HPC + AI Public	NVIDIA P100 x 496	10.71 PF (FP16)	5.36 PF / 10.71PF (FP32/FP16)		(Unranked)	(unranked)
U-Kyushu ITO-B	Oct. 2017	HPC + AI Public	NVIDIA P100 x 512	11.1 PF (FP16)	5.53 PF/11.1 PF (FP32/FP16)		(Unranked)	(Unranked)
AIST-AIRC AICC	Oct. 2017	AI Lab Only	NVIDIA P100 x 400	8.64 PF (FP16)	4.32 PF / 8.64PF (FP32/FP16)		(Unranked)	(Unranked)
Riken-AIP Raiden	Apr. 2018 (update)	AI Lab Only	NVIDIA V100 x 432	54.0 PF (FP16)	6.40 PF/54.0 PF (FP32/FP16)		1.213 PF #462	(Unranked)
AIST-AIRC ABCI	Aug. 2018	AI Public	NVIDIA V100 x 4352	544.0 PF (FP16)	65.3 PF/544.0 PF (FP32/FP16)		19.88 PF #8	14.423 GF/W #6
NICT & Sakura Internet	Summer 2019	AI Lab Only	NVIDIA V100 x 1700	~210 PF (FP16)	~26 PF/~210 PF (FP32/FP16)		4.128 #51 3.712 #58	(Unranked)
C.f. US ORNL Summit	Summer 2018	HPC + AI Public	NVIDIA V100 x 27,000	3,375 PF (FP16)	405 PF/3,375 PF (FP32/FP16)	445 PF (FP16)	143.5 PF #1	14.719 GF/W #5
Riken R-CCS Fugaku	Summer 2020	HPC + AI Public	Fujitsu A64fx x 158,976	4,300 PO (Int8)	1070PF/2150PF (FP32/FP16)	1420PF (FP16)	415.5PF #1 Jun2020	16.876 GF/W #1 Nov2019 (proto)

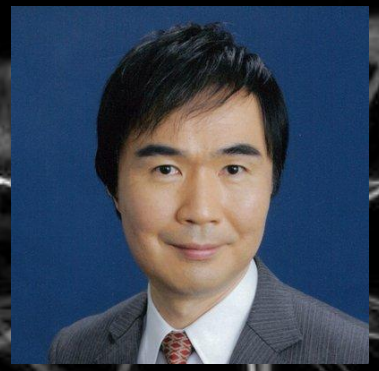
Inference 838.5PF
Training 86.9 PF

vs. Summit
Inf. 1/4
Train. 1/5

X2.58 Summit HPL-AI (equiv. ~70K GPUs)

High Performance Artificial Intelligence Systems

Team leader



Satoshi Matsuoka

Full-time research staff

Jun Igarashi

Aleksandr Drozd

Emil Vatai

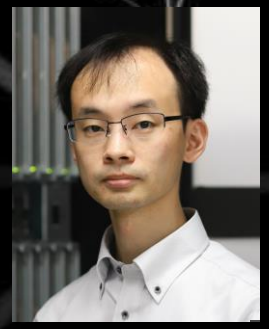
Shweta Salaria

Visiting researchers

+ more joining in a couple of weeks

Toshio Endo

Mohamed Wahib



Akihiro Nomura

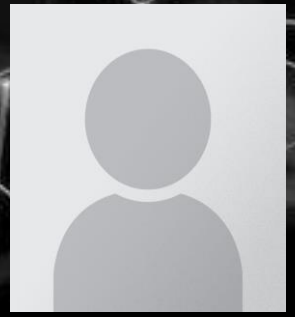
Peng Chen (pending)

Students at Tokyo Tech

Mateusz Bysiek

Lingqi Zhang

Yosuke Oyama

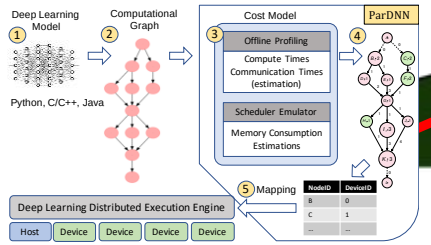


Ryan Barton

Challenges in Scaling DNN Training to 100K Nodes on Fugaku

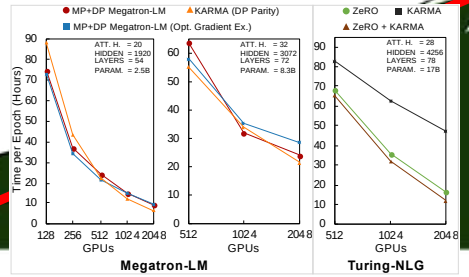
- **High performance in basic DNNL kernels (Convolution, etc.)**
  - Hardware: A64FX ARM SVE w/ low precision vectorized ops (FP16, INT8, etc.)
  - Software: (1) Aarch64+SVE port of Intel DNNL/OneAPI, (2) MoCUDA-CUDA emulation on Aarch64+SVE
  - Automated optimized kernel selection: μCUDNN, etc.
- **High performance in Data Parallel reduction**
  - HW: optimized reduction on Tofu-D network
- **Dealing with memory capacity limitations for large networks and large training datasets (e.g. 3D)**
  - (1) Efficient combination (model + data) parallelism, (2) KARMA: optimized scheduling of out-of-core memory and network offload for data parallelism
- **Convergence problem of effective large batch size, esp. SGD**
  - (1) traditional method: SGD with various tricks e.g., batchnorm, learning rate adjustment
  - (2) model + data parallelism to effectively reduce data parallelism
  - (3) K-FAC: advanced second-order method adapted to massive parallelism

Exploring and Merging Different Routes to O(100,000s) Nodes Deep Learning



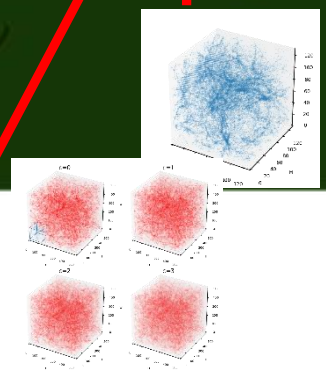
Non-intrusive graph-based partitioning strategy for large DNN models achieving superlinear scaling [1]

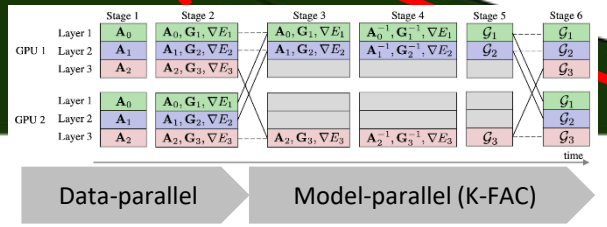
AIST, Koc U.



KARMA: Out-of-core distributed training (pure data-parallel) outperforming SoTA NLP models on 2K GPUs [2]

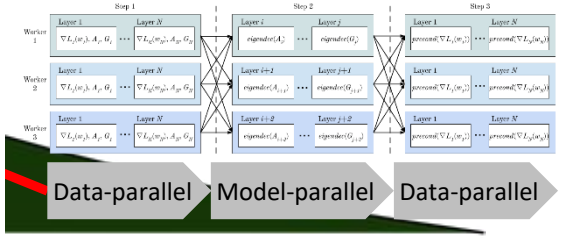
AIST, Matsuoka-lab, RIKEN





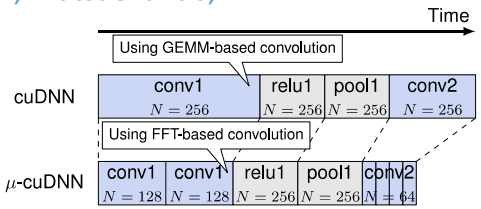
A model-parallel **2nd-order method (K-FAC)** trains ResNet-50 on **1K GPUs** in 10 minutes [4]

TokyoTech, NVIDIA, RIKEN, AIST



Layer-wise distribution and inverse-free design further accelerate K-FAC [5]

UT Austin, UChicago, ANL

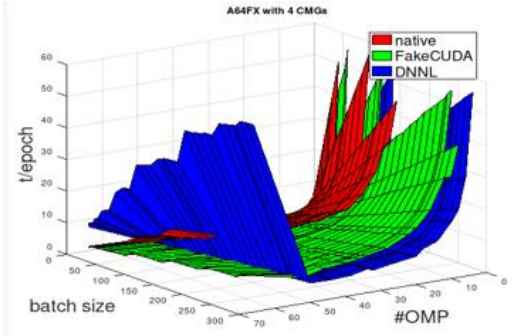


Model-parallelism enables 3D CNN training on **2K GPUs** with 64x larger spatial size and better convergence [3]

Matsuoka-lab, LLNL, LBL, RIKEN

Layer-wise loop splitting accelerates CNNs [6]

Matsuoka-lab, ETH Zurich



MocCUDA: Porting CUDA-based Deep Neural Network Library to A64FX and (other CPU arch.)

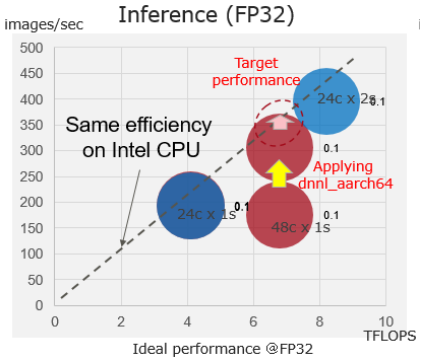
RIKEN, Matsuoka-lab, AIST

Merging Theory and Practice

Engineering for Performance Foundation

Porting High Performance CPU-based Deep Neural Network Library (DNNL) to A64FX chip

Fujitsu, RIKEN, ARM

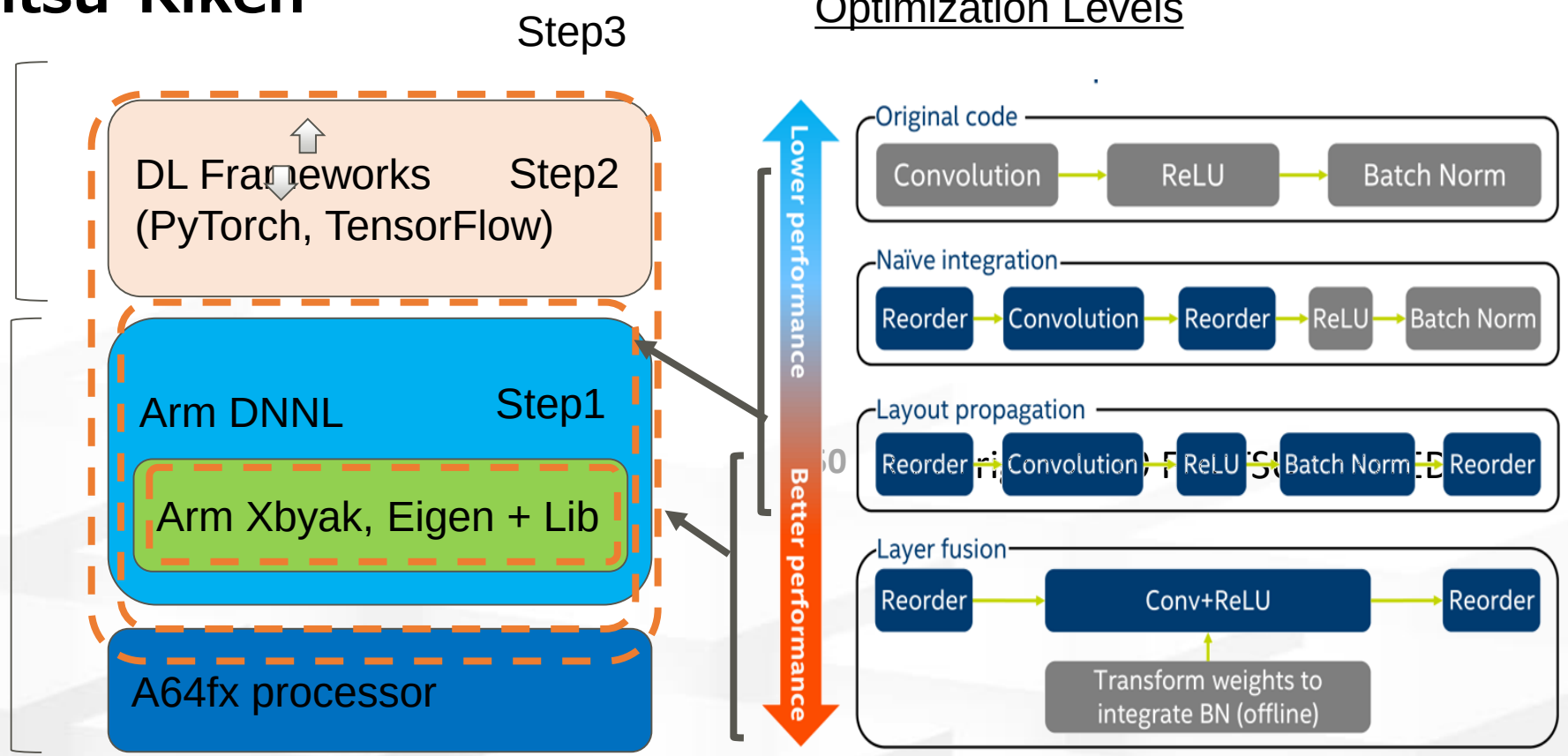


[1] M. Fareed et al., "A Computational-Graph Partitioning Method for Training Memory-Constrained DNNs", Submitted to PPoPP21
[2] M. Wahib et al., "Scaling Distributed Deep Learning Workloads beyond the Memory Capacity with KARMA", ACM/IEEE SC20 (Supercomputing 2020)
[3] Y. Oyama et al., "The Case for Strong Scaling in Deep Learning: Training Large 3D CNNs with Hybrid Parallelism," arXiv e-prints, pp. 1–12, 2020.
[4] K. Osawa, et al., "Large-scale distributed second-order optimization using kronecker-factored approximate curvature for deep convolutional neural networks," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2019-June, pp. 12351–12359, 2019.
[5] J. G. Pauloski, Z. Zhang, L. Huang, W. Xu, and I. T. Foster, "Convolutional Neural Network Training with Distributed K-FAC," arXiv e-prints, pp. 1-11, 2020.
[6] Y. Oyama et al., "Accelerating Deep Learning Frameworks with Micro-Batches," Proc. IEEE Int. Conf. Clust. Comput. ICCC, vol. 2018-September, pp. 402–412, 2018.

Fujitsu-Riken-Arm joint effort on AI framework development on SVE/A64FX

- **MOU Signed Fujitsu-Riken Nov. 25, 2019**

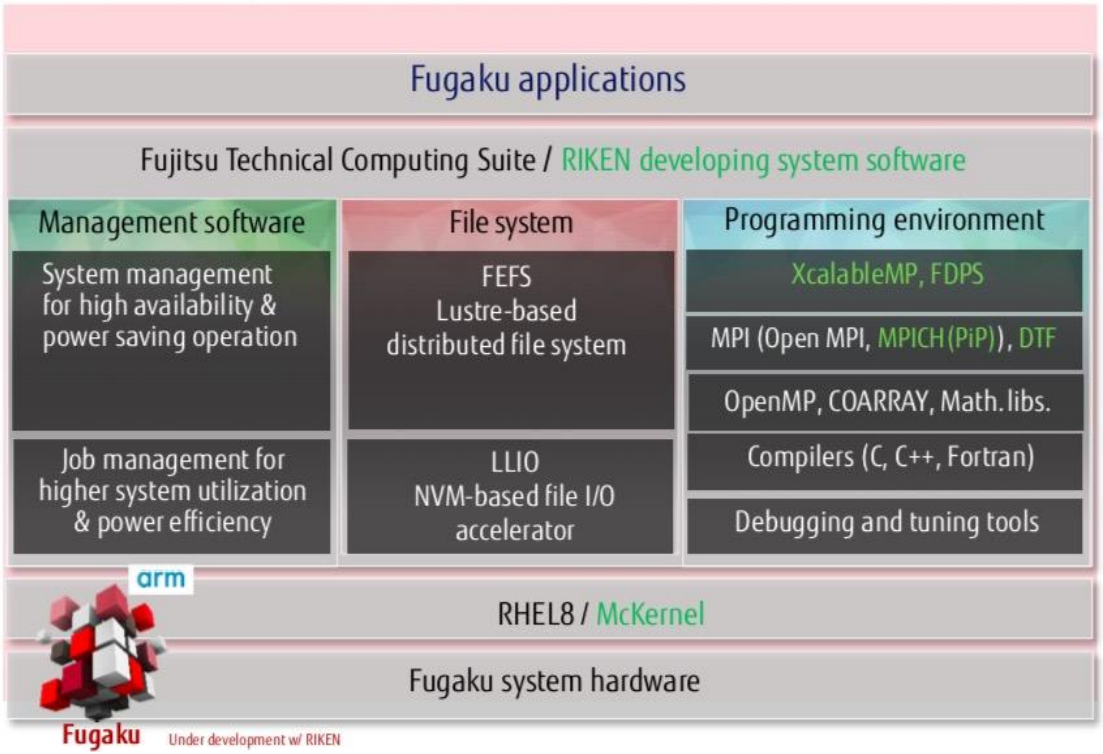
- **Also w/Arm**
- **1st release May 2020**
  - First ver. optimized for inference
  - Next ver. training optimization

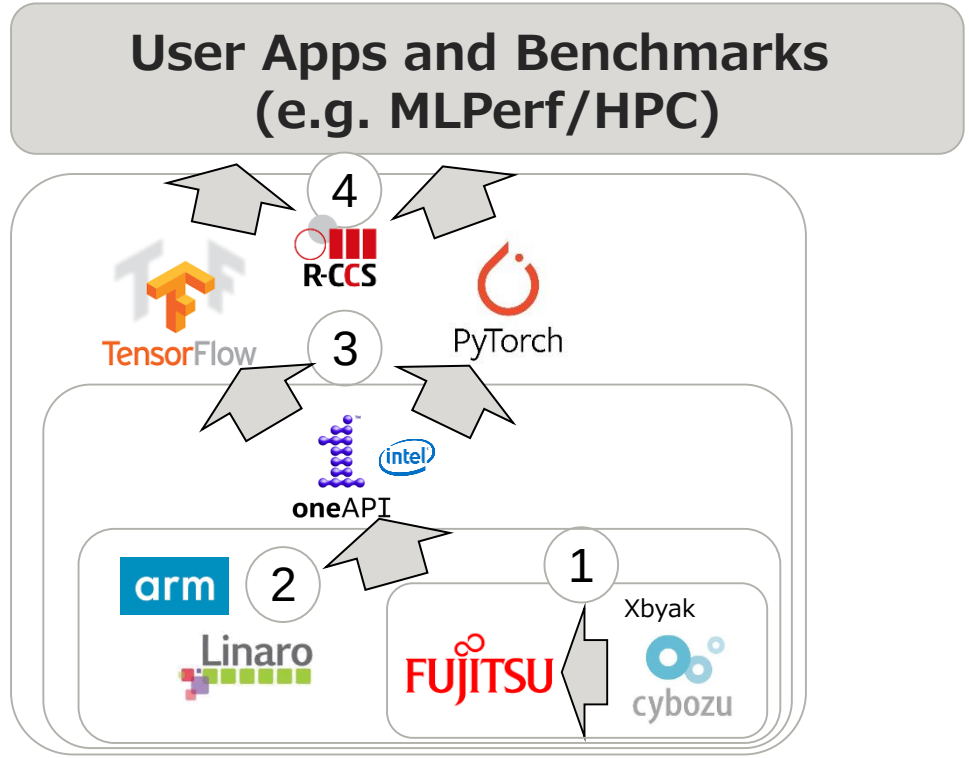


Exaops of sim, data, and AI on Fugaku and Cloud

AI Software Stack

- Fujitsu + Riken AI Framework Development on Fugaku / A64FX
  - ① Collaboration w/Cyborz co. to port & tune Xbyak+OneDNN
  - ② ARM, Linaro collaboration to upstream to OneDNN
  - ③ Upstream Aaarch64/OneDNN to DL framwork
  - ④ Benchmark & Parallelization

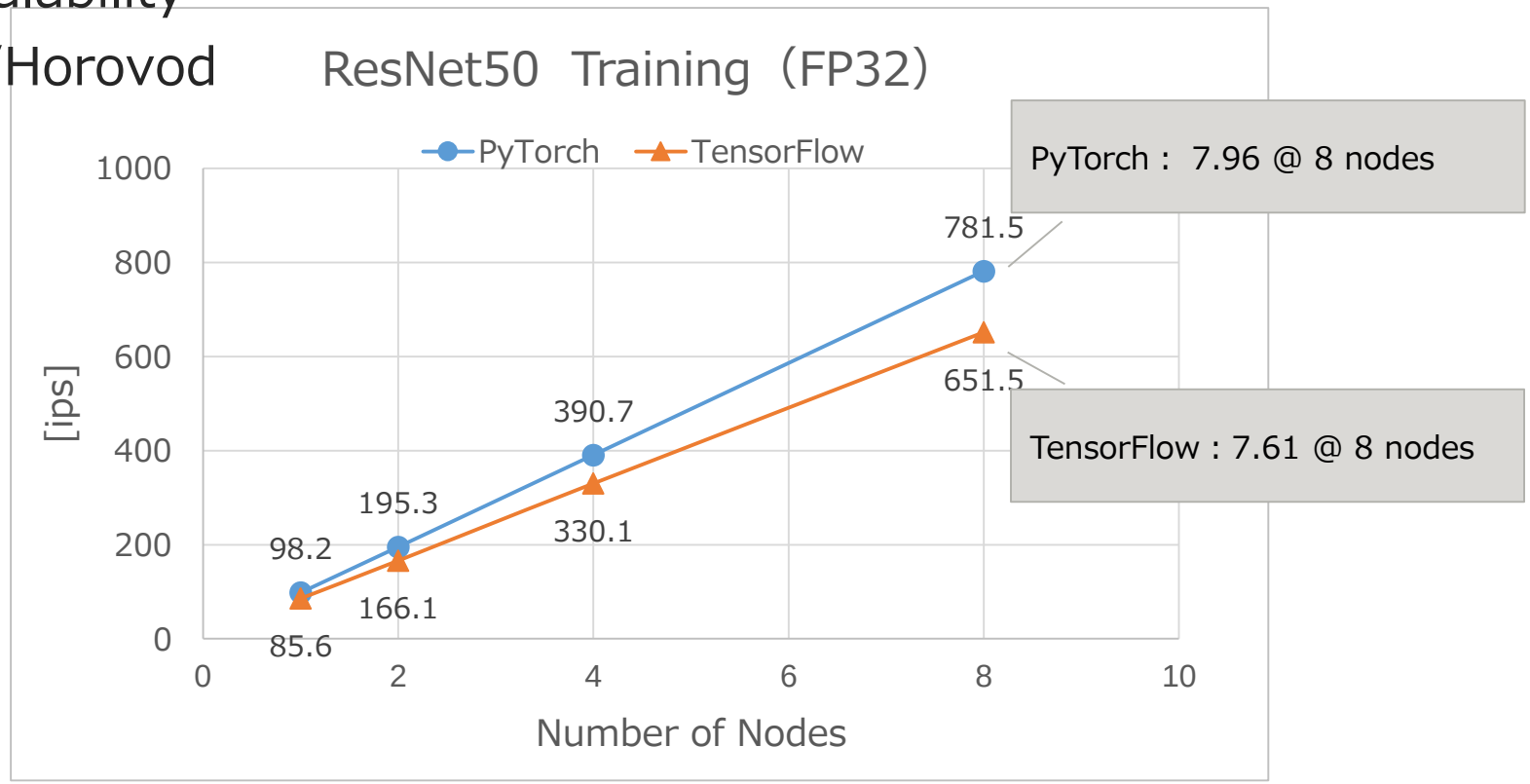




Fugaku DNNL ResNet50 Multi-Node Training (FP32)

FUJITSU

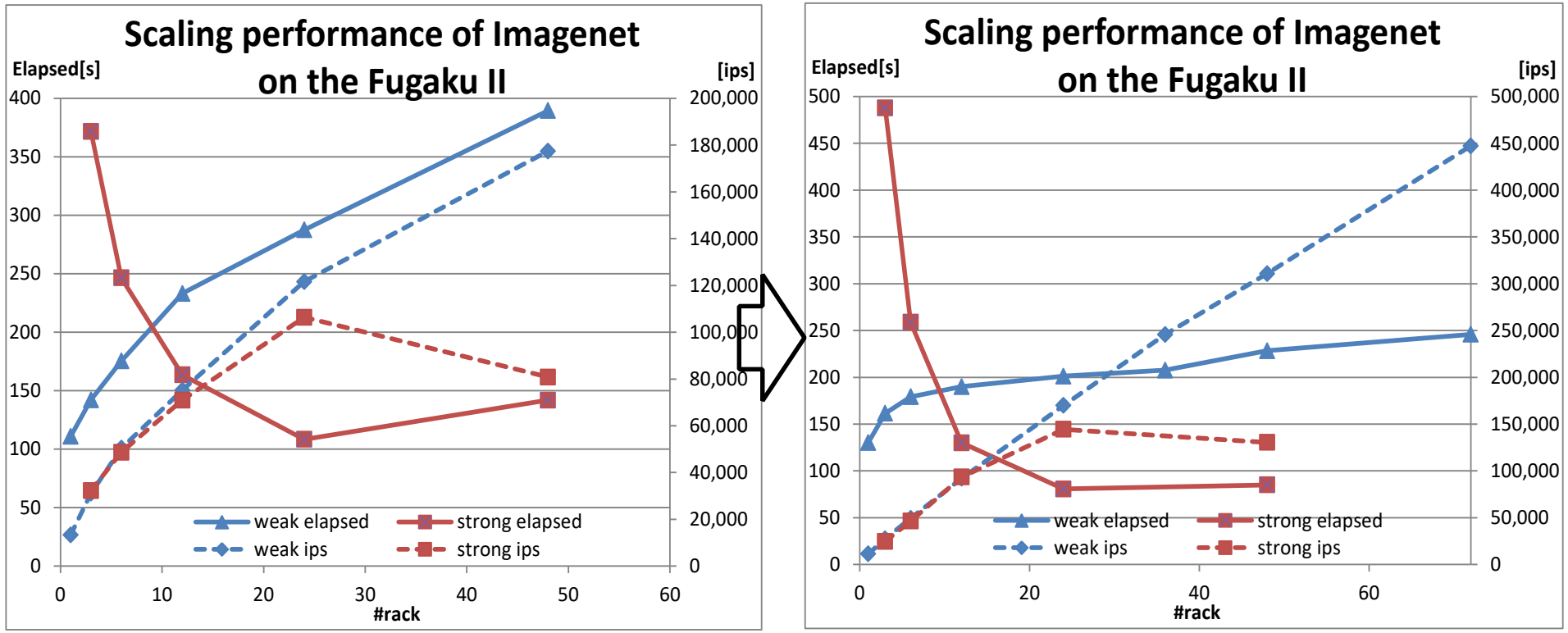
- Training scalability on TensorFlow and PyTorch
  - Comparable per-core performance to Skylake
  - Expect similar ips/W c.f. V100
  - Good linear scalability
  - Tofu-D + MPI/Horovod



Good scalability

Chainer on Fugaku (April 2020)

- **Scaling Result**
  - **448,048[ips]/72rack (27,648 nodes)**
    - JOB runtime = 910.0s
    - Pseudo staging I/O optimization = 997.0[s]
    - Further scaling is possible with I/O optimization



Deep Learning Performance Benchmarking

Modular and expandable:

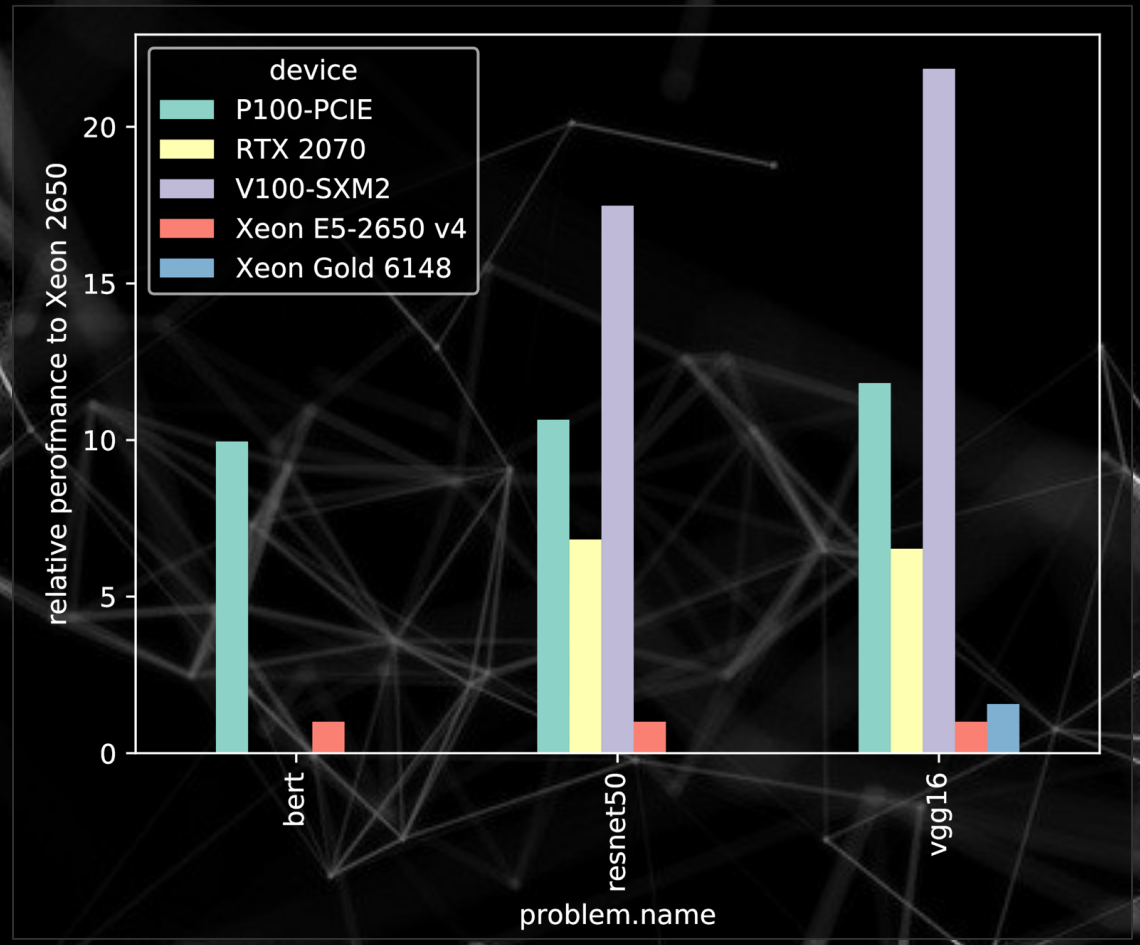
- Hardware: benchmark different hardware configs
- Backend: swap out low-level libraries (e.g. BLAS)
- Frameworks: PyTorch, Tensorflow etc.
- Well-defined APIs and architecture
- Support for training and inference

Support popular models:

- *MLPerf and MLPerf HPC tasks*: ResNet, DeepLAB, BERT etc
- Kernel level: DeepBench specs (2D convolutions, GEMM) and more
- Small versions (real and synthetic) dataset to be usable on slower devices

Versatile tool:

- Detailed human & machine readable JSON logs
- Out-of-the-box analysis and visualisation
- Easy comparison of HW and implementations



http://benchmarker.blackbird.pw/

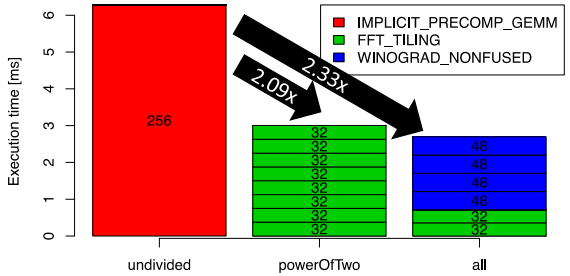
In collaboration with AIST, Tokyo Tech

Selecting the Optimal Convolution Kernel

- **NEW! Micro Batching: Tokyo Tech. and ETH [Oyama, Tan, Hoefler & Matsuoka, IEEE Cluster 2019]**
  - Use the “micro-batch” technique to select the best convolution kernel
    - Direct, GEMM, FFT, Winograd
    - Optimize both speed and memory size
  - On high-end GPUs, in many cases Winograd or FFT chosen over GEMM
    - They are faster but use more memory
  - Currently implemented as cuDNN wrapper, applicable to all frameworks
  - For Post-K, (1) Winograd/FFT are selected more often, and (2) performance will be similar to GPUs in such cases

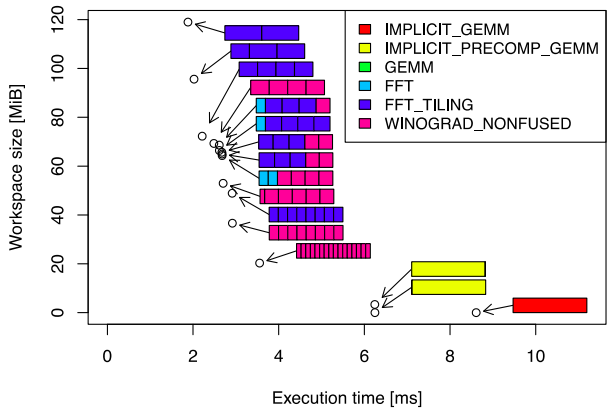
Evaluation: WR using Dynamic Programming

- µ-cuDNN achieved **2.33x** speedup on forward convolution of AlexNet conv2



`cudnnConvolutionForward` of AlexNet conv2 on NVIDIA Tesla P100-SXM2
Workspace size of 64 MiB, mini-batch size of 256
Numbers on each rectangles represent micro-batch sizes

Evaluation: WD using Integer LP



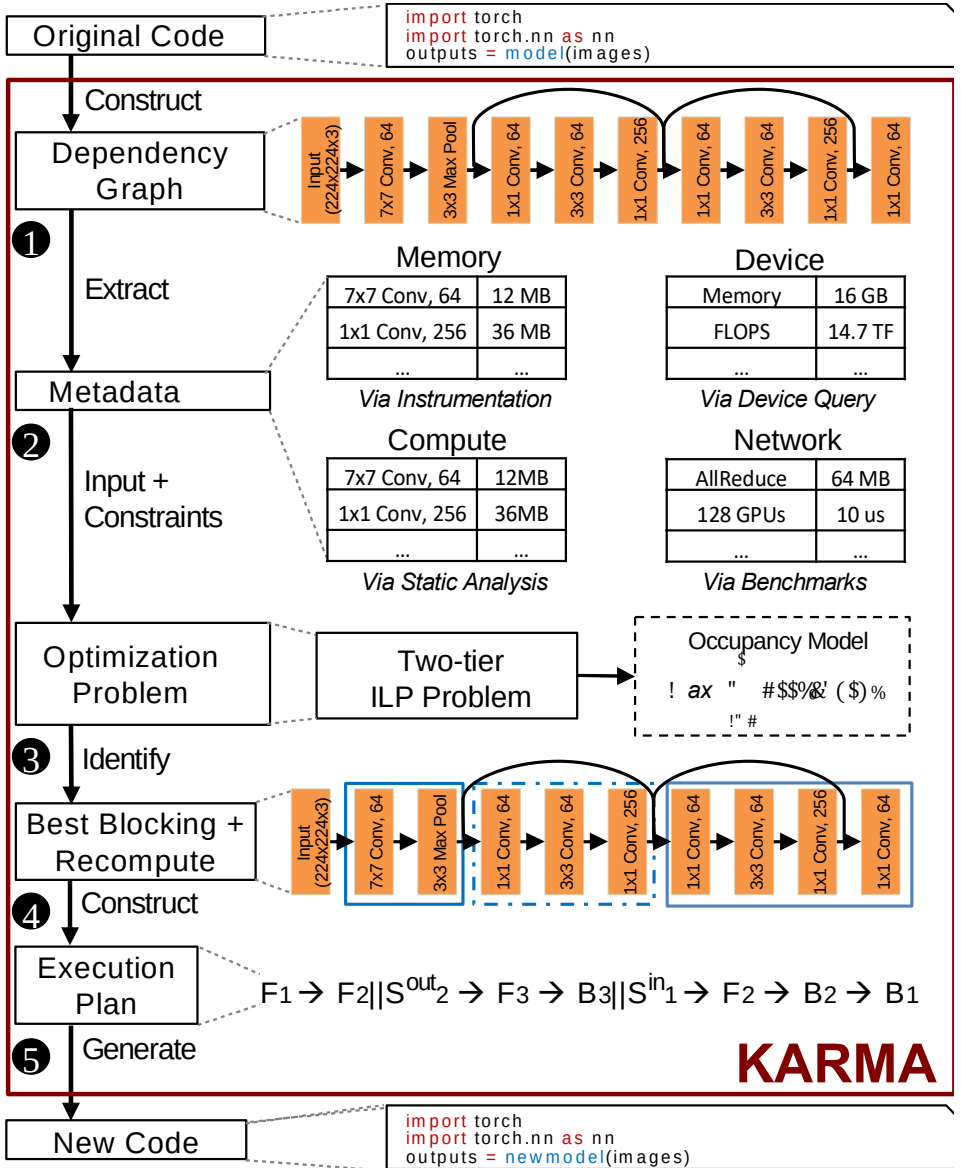
A desirable configuration set of AlexNet conv2 (Forward)
Mini-batch size of 256, P100-SXM2
Each bar represents proportion of micro-batch sizes and algorithms

Scaling Distributed Deep Learning Workloads beyond the Memory Capacity with KARMA

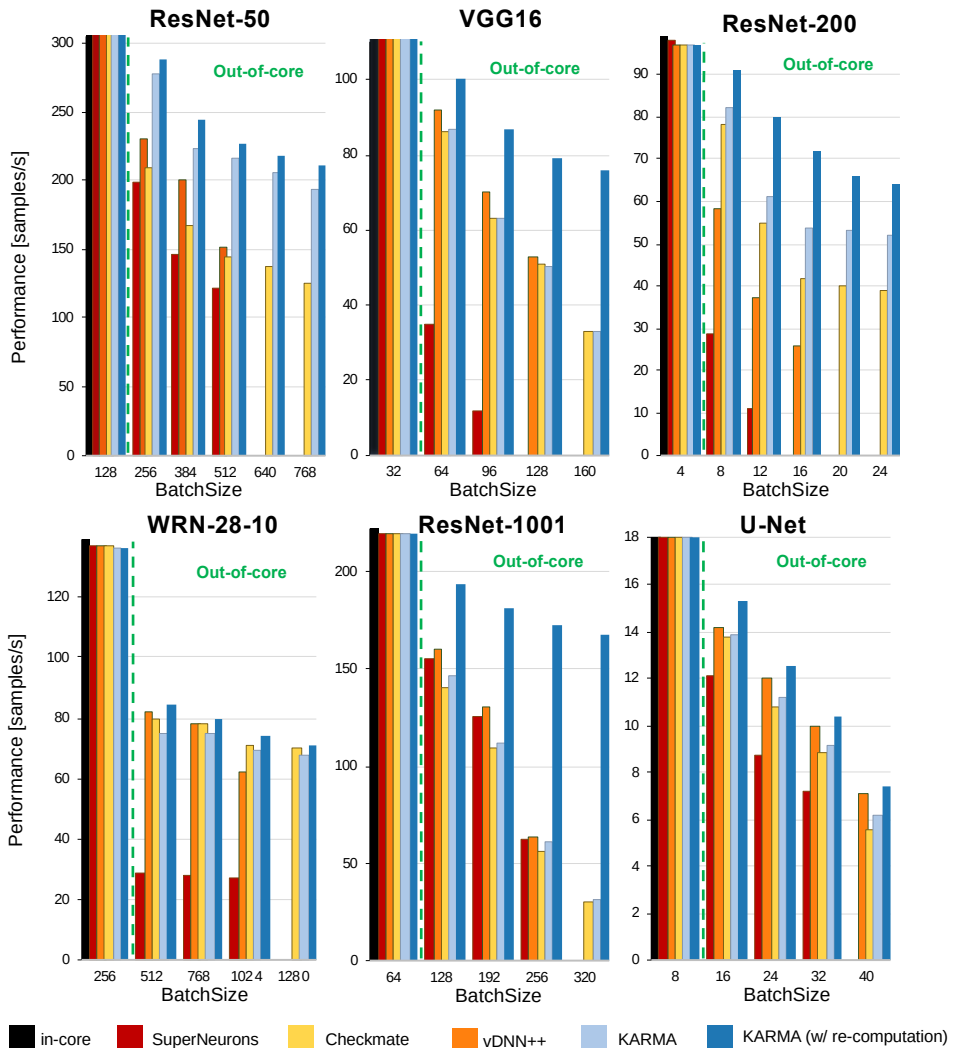
Mohamed Wahib[1,2], Haoyu Zhang[4], Truong Thao Nguyen[2], Aleksandr Drozd[1], Jens Domke[1], Lingqi Zhang[3], Ryousei Takano[2], Satoshi Matsuoka[1,3]

1 RIKEN-CCS 2 AIST (OIL) 3 TokyoTech 4 miHoYo Inc

- New strategy for Out-of-Core
  - Keep enough concurrency for GPU to be busy
  - Capacity-based interleaved with recompute
- 1.52x over state-of-the-art (single GPU)
- First out-of-core to support multi-GPU
  - Heterogeneity and careful orchestration
- Outperforming DP+MD with out-of-core
  - Experiments with up to 2,048 GPUs

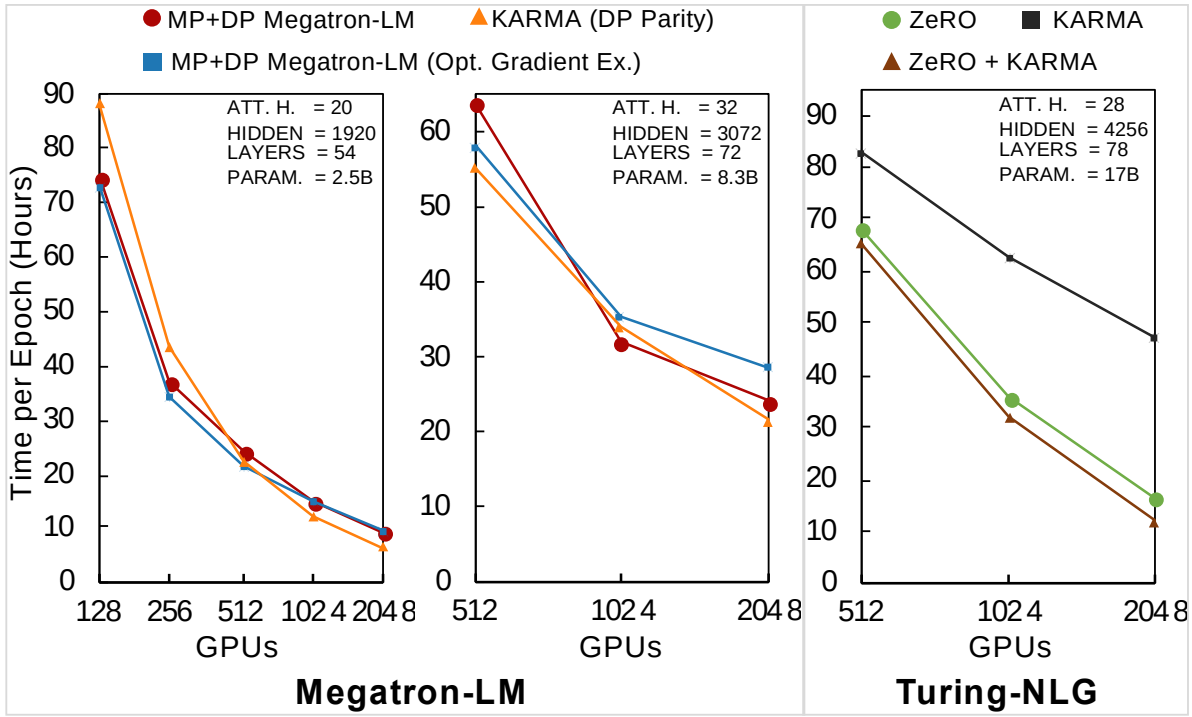


Results



Performance using a V100 SMX2 (16GB) GPU. For all figures, only the first reported mini-batch size (x-axis) fits in memory

- KARMA gives better performance on parity
  - KARMA has fewer communication rounds (larger mini-batch)



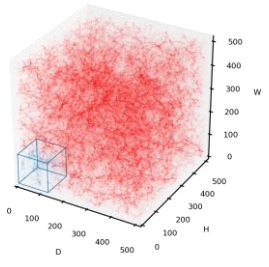
We compare using the same number of GPUs. Megatron-LM: we compare the original data-/model-parallel hybrid, the original plus our optimized phased gradient exchange, and data parallel KARMA. Turing-NLG: we compare the hybrid ZeRO reference implementation, data parallel KARMA, and KARMA used on top of data parallel ZeRO. The mini- batch size in KARMA is multiplied by the model parallel factor of the original implementation

Toward Training a Large 3D Cosmological CNN with Hybrid Parallelization

Yosuke Oyama, Naoya Maruyama, Nikoli Dryden, Erin McCarthy, Peter Harrington, Jan Balewski, Satoshi Matsuoka, Peter Nugent, and Brian Van Essen

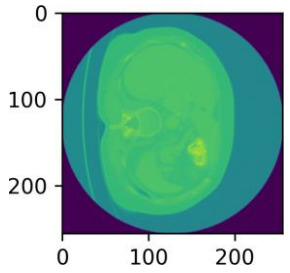
- **Background: Hybrid-parallelism has various advantages over data-parallelism or model-parallelism for deep learning**
  - **3 Huge models can be trained by partitioning each sample into multiple GPUs**
- **Target applications:**

CosmoFlow (regression)



A 4×512^3 cube requires 8 GPUs

The 3D U-Net (segmentation)



A 256^3 cube requires 16 GPUs

- **Proposal: Extend Livermore Big Artificial Neural Network Toolkit (LBANN) to perform hybrid-parallel training of 3D CNNs**
  - **Introduce spatial partitioning and in-memory caching for data I/O**
  - **Identify and optimize computational bottlenecks for 3D CNNs**
  - **Propose a performance model to reveal the scalability with hybrid-parallelism**

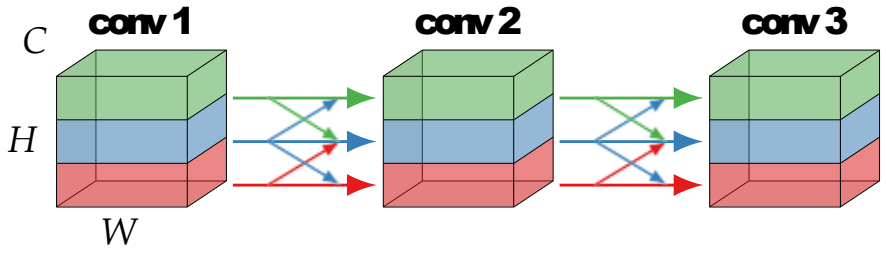


Figure: Spatial partitioning.

INTERNAL USE ONLY

Toward Training a Large 3D Cosmological CNN with Hybrid Parallelization

Yosuke Oyama, Naoya Maruyama, Nikoli Dryden, Erin McCarthy, Peter Harrington, Jan Balewski, Satoshi Matsuoka, Peter Nugent, and Brian Van Essen

- Evaluation (strong scaling): **Achieved 1.77x of speedup on 512 nodes (2048 GPUs) compared to 128 nodes when the mini-batch size (N) is 64**

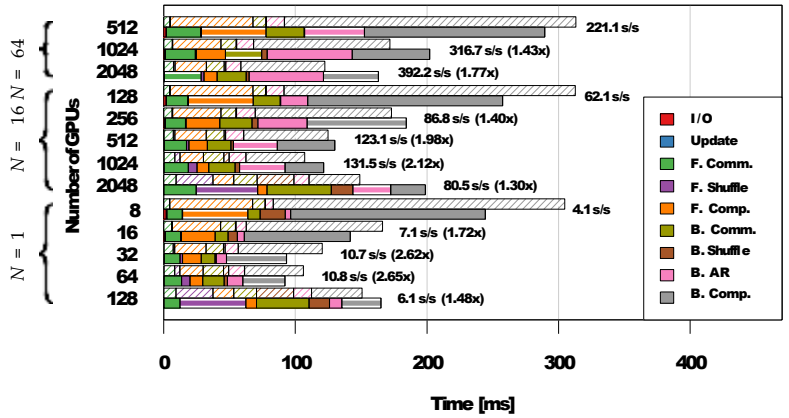


Figure: Strong scaling of the CosmoFlow network. Shaded bars show iteration time predicted by the performance model.

- Evaluation (weak scaling): **Achieved 147.31x of speedup on 2048 GPUs over 8 GPUs by exploiting hybrid-parallelism, even if layer-wise communication is introduced**

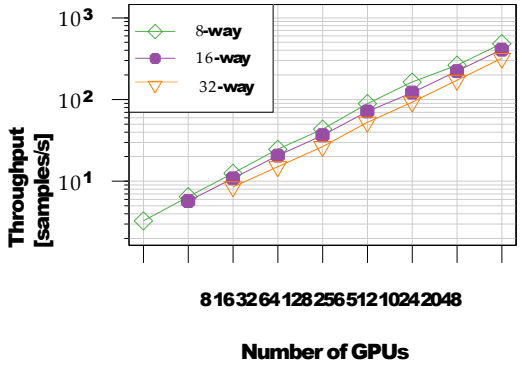
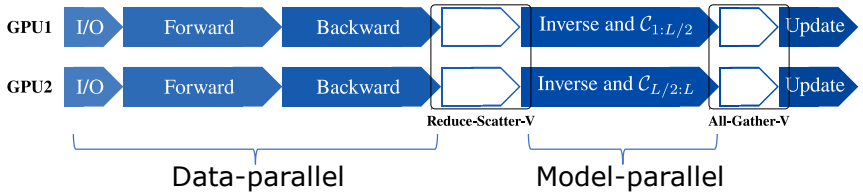


Figure: Weak scaling of the CosmoFlow network.

INTERNAL USE ONLY

Accelerating DL with 2nd Order Optimization and Distributed Training [Tsuji et al.] => *Towards 100,000 nodes scalability*

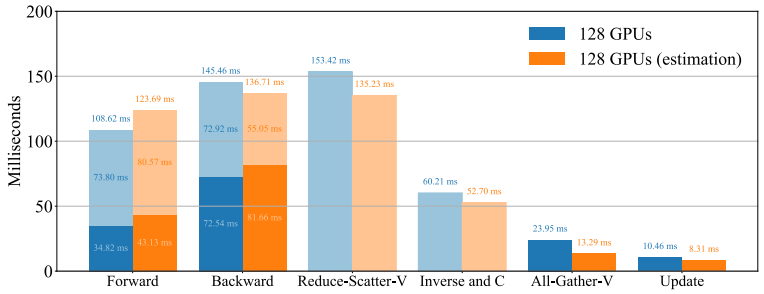
- Background
  - Large complexity of DL training.
  - Limits of data-parallel distributed training.
  - *> How to accelerate the training further?*
- Method
  - Integration of two techniques: 1) data- and model-parallel distributed training, and 2) K-FAC, an approx 2nd order optimization.
- Evaluation and Analysis
  - Experiments on ABCI supercomputer.
  - Up to 128K batch size w/o accuracy degradation.
  - Finish training in 35 epochs/10 min/1024 GPUs in 32K batch size.
  - A performance tuning / modeling.



Design our hybrid parallel distributed K-FAC

	Batch size	# Iterations	Accuracy
Goyal et al.	8K	14076	76.3%
Akiba et al.	32K	3519	75.4%
Ying et al.	64K	1760	75.2%
Ours	**128K**	**978**	**75.0%**

Comparison with related work (ImageNet/ResNet-50)



Time prediction with the performance model

Osawa et al., Large-Scale Distributed Second-Order Optimization Using Kronecker-Factored Approximate Curvature for Deep Convolutional Neural Networks, CVPR 2019

Optimizing Collective Communication in DL Training (3 of 3)

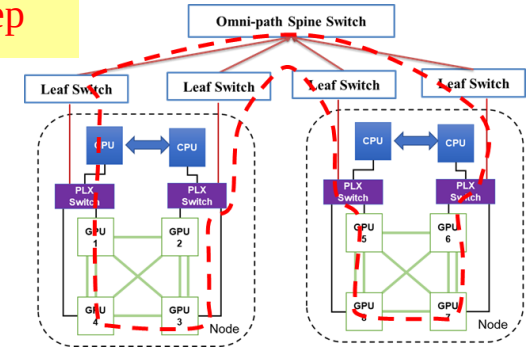
Proposal: Separate intra-node and inter-node comm. ➔ **multileader hierarchical algorithm**

- Phase 1: Intra-node reduce to the node leader
- Phase 2: Inter-node all-reduce between leaders
- Phase 3: Intra-node broadcast from the leaders

Key Results:

- Cut down the communication time up to **51%**
- Reduce the power consumption up to **32%**

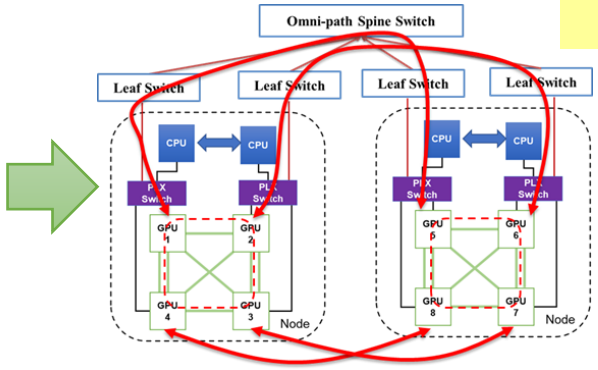
2(P-1) steps, send $\frac{N}{P}$ per step



Ring-based algorithm

- Good for large message size
- Worse with inter-node comm.

2($\frac{P}{k}$ -1) steps, $\frac{N\,(p-k)}{P\,k}$ per step



Multileader hierarchical algorithm

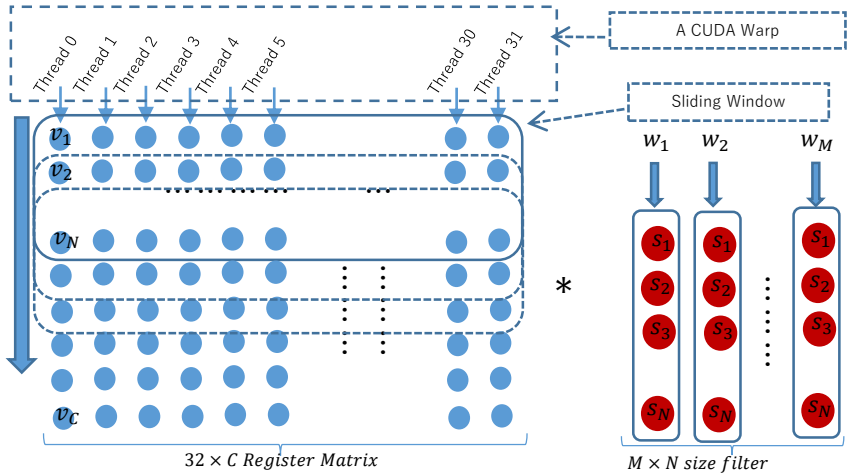
- Optimized for inter-node comm.

"Efficient MPI-Allreduce for Large-Scale Deep Learning on GPU-Clusters", Truong Thao Nguyen, Mohamed Wahib, Ryousei Takano, Journal of Concurrency and Computation: Practice and Experience (CCPE) , Accepted: to appear in 2019.10

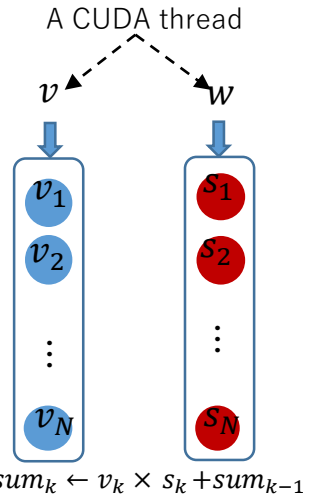
Pushing the Limits for 2D Convolution Computation On GPUs [1]

[To appear SC19]

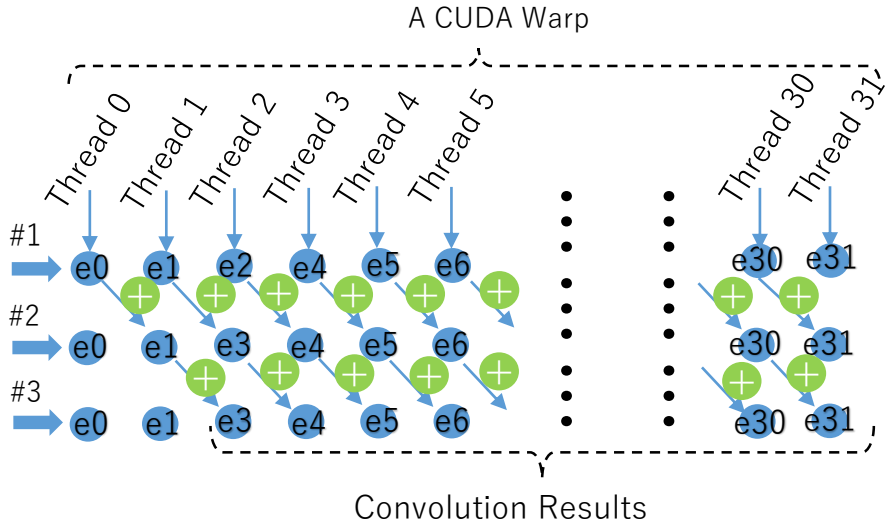
- **Background of 2D convolution**
  - Convolution on CUDA-enabled GPUs is essential for Deep Learning workload
  - A typical memory-bound problem with regular access
- **Method**



(1) Register Cache

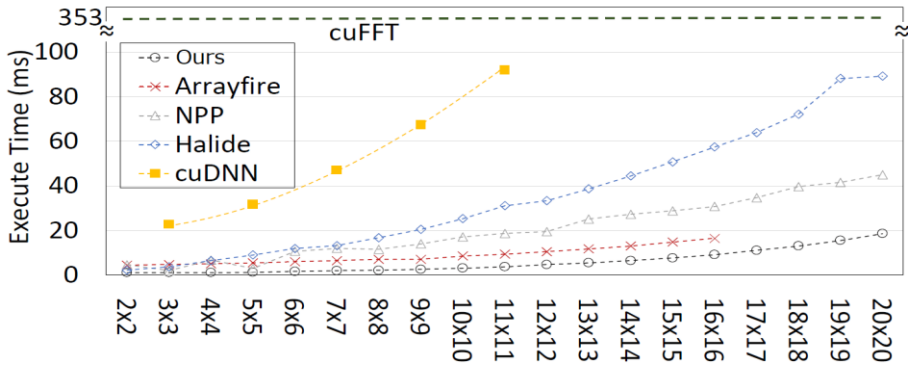


(2) Compute partial sums

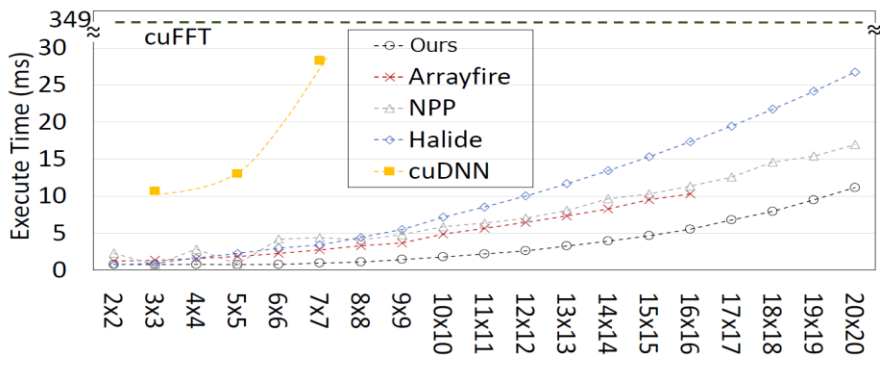


(3) Transfer partial sums

- **Evaluation**
  - a single Tesla **P100** and **V100** GPUs
  - Single precision



Evaluation on Tesla **P100** GPU

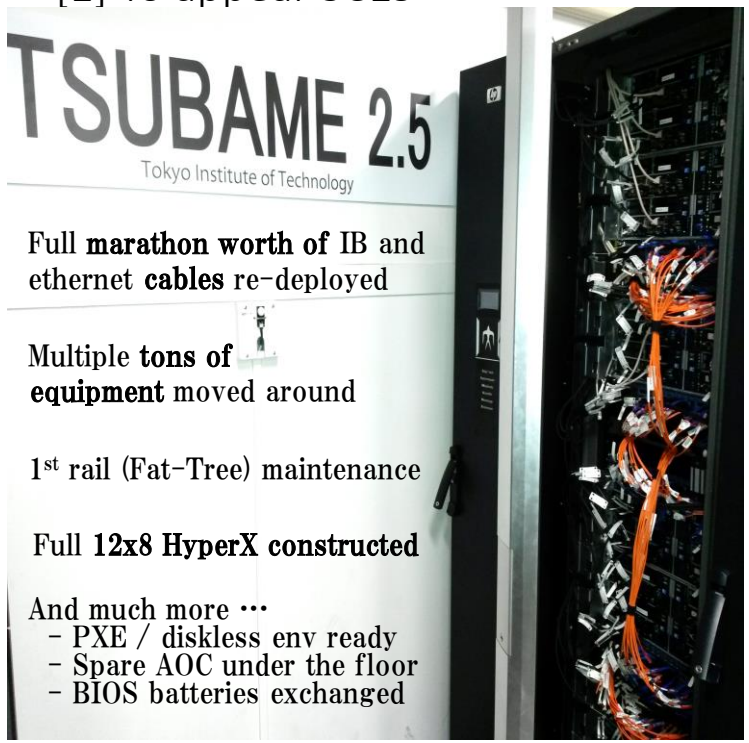


Evaluation on Tesla **V100** GPU

[1] Peng Chen, Mohamed Wahib, Shinichiro Takizawa, Satoshi Matsuoka. Pushing the Limits for 2D Convolution Computation On CUDA-enabled GPUs. 第163回ハイパフォーマンスコンピューティング研究会, Mar. 2018.

Evaluating the HyperX Topology: A Compelling Alternative to Fat-Trees?

[1] To appear SC19



➔ First large-scale 2.7 Pflop/s (DP)

ullation 2D HyperX:

- **24 racks** (of 42 T2 racks)
- 96 QDR switches (+ 1st rail)
- 1536 IB cables (720 AOC)
- **672 compute nodes**
- **57% bisection bandwidth**

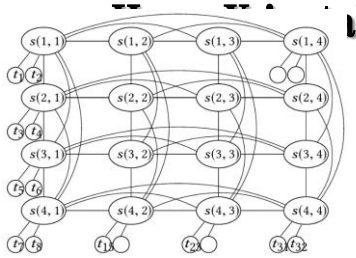
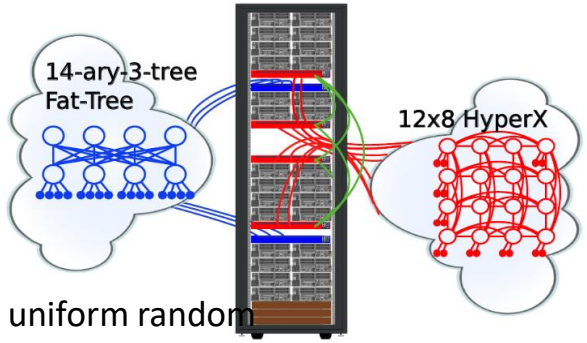


Fig.1: HyperX with n-dim. integer lattice (d_1,…,d_n) base structure fully connected in each dim.

1:1 comparison (as fair as possible) of 672-node 3-level Fat-Tree and 12x8 2D HyperX

- NICs of 1st and 2nd rail even on same CPU socket
- Given our HW limitations (few "bad" links disabled)



Advantages (over FT) assuming **adaptive routing (AR)**

- **Reduced HW cost** (AOC/switches) → similar perf.
- **Lower latency** when scaling up (less hops)
- **Fits** rack-based **packaging** model for HPC/racks
- **Only needs 50% bisection BW** to provide 100% throughput for uniform random

Q1: Will reduced bisection BW (57% for HX vs. ≥100%) impede Allreduce performance?

Q2: Mitigation strategies against lack of AR? (→ eg. placement or smart routing)

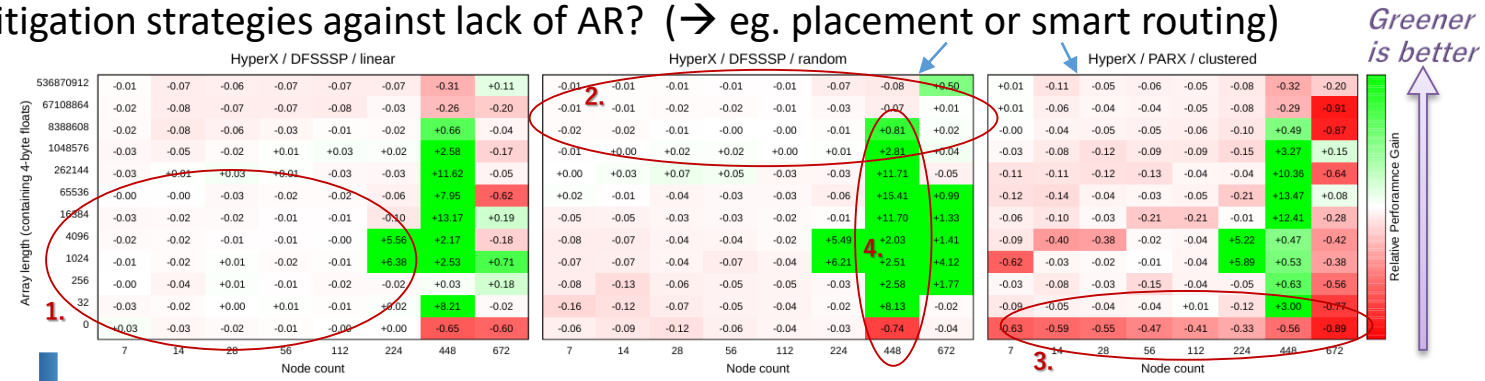


Fig.2: Baidu's (DeepBench) Allreduce (4-byte float) scaled 7→672 cn (vs. "Fat-tree / ftree / linear" baseline)

1. *Linear* good for small node counts/msg. size
2. *Random* good for DL-relevant msg. size (+/– 1%)
3. Smart routing suffered SW stack issues
4. FT + ftree had bad 448-node corner case

HyperX topology is promising and cheaper alternative to state-of-the-art Fat-Tree networks!

[1] Domke et al. "HyperX Topology: First at-scale Implementation and Comparison to the Fat-Tree" currently under review at SC'19 and HOTI'19

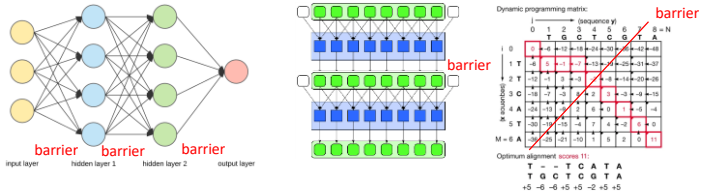
Funded by and in collaboration with Hewlett Packard Enterprise, and supported by Fujitsu, JSPS KAKENHI, and JSP CREST

A Study of Synchronization Methods in Modern GPUs[1]

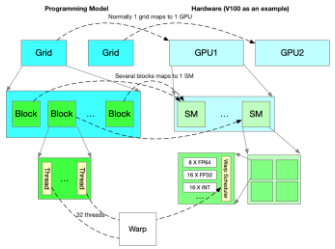
Lingqi Zhang[1] , Mohamed Wahib[2,3], Haoyu Zhang[4], Satoshi Matsuoka[2,1]

1 TokyoTech 2 RIKEN-CCS 3 AIST (OIL) 4 miHoYo Inc

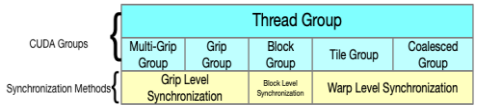
- Introduction
  - Different level of synchronization is necessary



  - Mainstream solution
    - Launch sequence of Kernels in single stream as implicit synchronization
    - Block synchronization (__syncthreads())
  - Nvidia proposed new synchronization methods (2017). Not yet being fully understood.



CUDA programming abstraction



CUDA synchronization hierarchy

- CPU-side implicit Barrier

Launch Type	Launch Overhead (ns)	Null Kernel Kernel Total Latency (ns)
Traditional	1081	8888
Cooperative	1063	10248
Cooperative Multi-Device	1258	10874

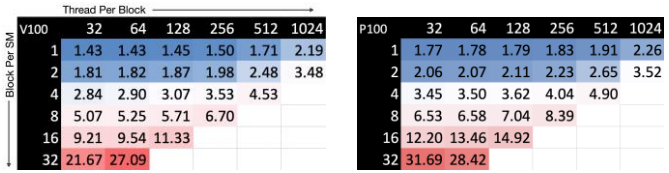
- GPU synchronization instructions Single Stream Processor

Type (group size)	Latency (cycle)		Throughput (sync/cycle)	
	V100	P100	V100	P100
Shuffle(Tile)(*)	22	31	0.928	0.642
block(warp))	22	218	0.475	0.091

- Guideline for device-wide barrier

Single Node	Grid Sync	• Iterative algorithms
	Implicit Barrier	• Only synchronize limited amount of
Multi Node	Grid Sync	• Iterative algorithms • Not know much about OpenMP and
	Implicit Barrier	• Only synchronize limited amount of

- GPU synchronization instructions Single Node



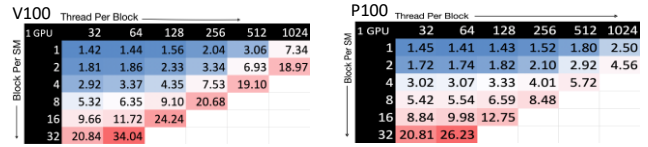
Thread Per Block

V100	32	64	128	256	512	1024
1	1.43	1.43	1.45	1.50	1.71	2.19
2	1.81	1.82	1.87	1.98	2.48	3.48
4	2.84	2.90	3.07	3.53	4.53	
8	5.07	5.25	5.71	6.70		
16	9.21	9.54	11.33			
32	21.67	27.09				

Block Per SM

P100	32	64	128	256	512	1024
1	1.77	1.78	1.79	1.83	1.91	2.26
2	2.06	2.07	2.11	2.23	2.65	3.52
4	3.45	3.50	3.62	4.04	4.90	
8	6.53	6.58	7.04	8.39		
16	12.20	13.46	14.92			
32	31.69	28.42				

Latency of grid synchronization (us)



V100 Thread Per Block

1 GPU	32	64	128	256	512	1024
1	1.42	1.44	1.56	2.04	3.06	7.34
2	1.81	1.86	2.33	3.34	6.93	18.97
4	2.92	3.37	4.35	7.53	19.10	
8	5.32	6.35	9.10	20.68		
16	9.66	11.72	24.24			
32	20.84	34.04				

Block Per SM

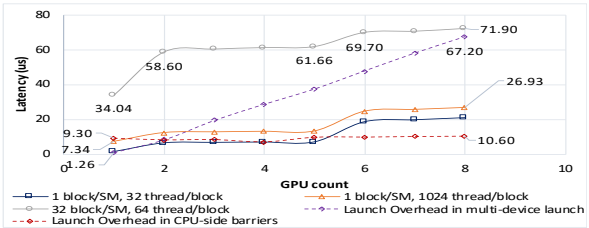
P100 Thread Per Block

1 GPU	32	64	128	256	512	1024
1	1.45	1.41	1.43	1.52	1.80	2.50
2	1.72	1.74	1.82	2.10	2.92	4.56
4	3.02	3.07	3.33	4.01	5.72	
8	5.42	5.54	6.59	8.48		
16	8.84	9.98	12.75			
32	20.81	26.23				

Block Per SM

Latency of multi-grid synchronization (us)

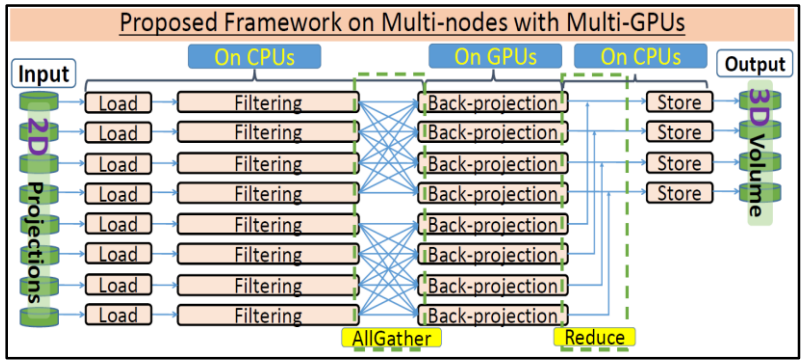
- GPU synchronization instructions Multi Node

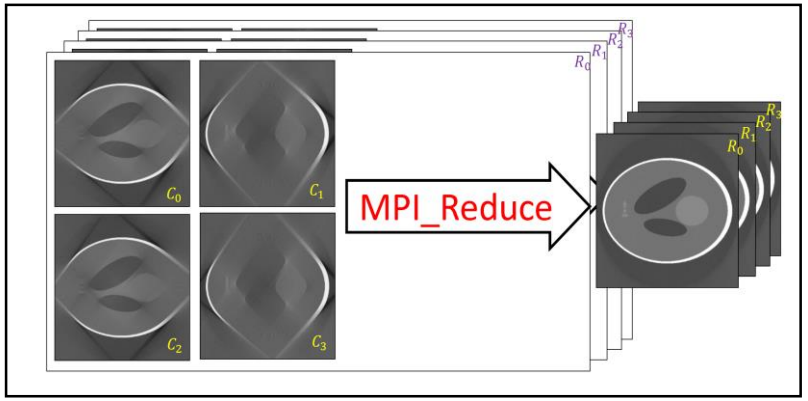


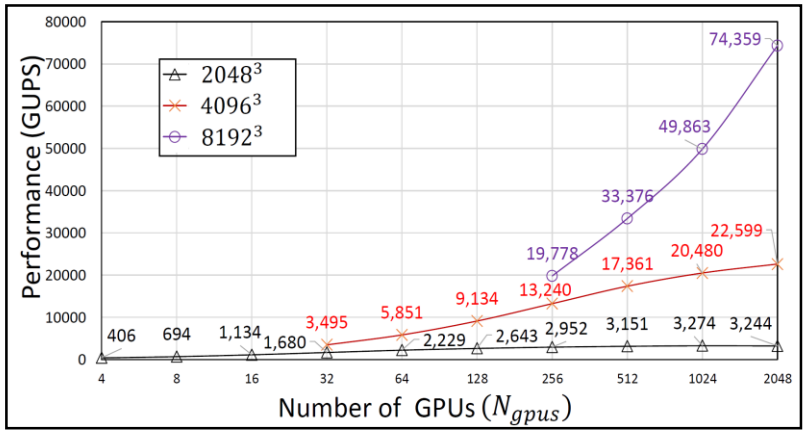
[1] Lingqi Zhang, Mohamed Wahib, Haoyu Zhang, Satoshi Matsuoka. A Study of Single and Multi-device Synchronization Methods in Nvidia GPUs. InProceedings of the IPDPS 2020.

High-resolution Image Reconstruction on Supercomputers[1]

- **Background**
  - High-resolution Compute Tomography (CT) is widely used
    - Medical diagnosis, non-invasive inspection, reverse engineering
  - Computations are very intensive
    - Filtering computation, $O(Log(N)N^2)$
    - Back-projection, $O(N^4)$
  - High-resolution image is often required but not attainable
    - $(4K)^3$, $(6K)^3$, $(8K)^3$, etc.
- **Proposed algorithm**
  - Take advantage of the heterogeneity of GPU-accelerated supercomputer
    - GPUs are used for back-projection
    - CPUs are used for filtering computation
  - Design a parallel computing pipeline
  - Employ distributed system to tackle the out-of-core problem
  - Use advanced MPI for inter-node communication
- **Impact to the real-world applications**
  - Regardless of resolution of 3D images
  - Efficiently generate 3D images for many purposes
    - Training sample for Deep Learning
    - Improve image quality with iterative reconstruction [2]







[1] P. Chen, et al., "iFDK: A Scalable Framework for Instant High-resolution Image Reconstruction," SC19.
[2] W. Lin, et al. “DuDoNet: Dual Domain Network for CT Metal Artifact Reduction”. CVPR 2019

Vectorization and Maximization of Locality for Back-projection Algorithms[1] (1/2)

- Background of Computed Tomography (CT)
  - CT image is widely used in industrial, medical, and scientific fields
  - Back-projection is a fundamental computation in image reconstruction
    - Filtered Back-Projection (FBP)
    - Maximum Likelihood Estimation Method (MLEM)
    - Algebraic Reconstruction Technique (ART)
- Challenges
  - Intensive computation: $O(\log(N)N^2)$
  - Complex operations:
    - 3D projection computation
    - Bilinear interpolation
- Motivation
  - OpenCL is a platform portable interface and can generate vectorized codes
  - SIMD-accelerated processors are prevalent, e.g. X86 CPU, A64FX
- Solution
  - Improve the data locality by transposing the projection images and volume data
  - Reduce the arithmetic computation by geometrical characteristics
  - Optimize the bilinear interpolation in a general memory access fashion

Fig. 1: CT images.

Image source: https://the-imaging-centers.com/computed-tomography-ct-scan/

[1] **P. Chen, et al., "Ecient FDK Algorithms on SIMD-accelerated Processors," SWOPP2020.**

Vectorization and Maximization of Locality for Back-projection Algorithms (2/2)

- Evaluation environment
- Results

OpenMP-based implementation

Our OpenCL-based Implementations

Image Reconstruction problems

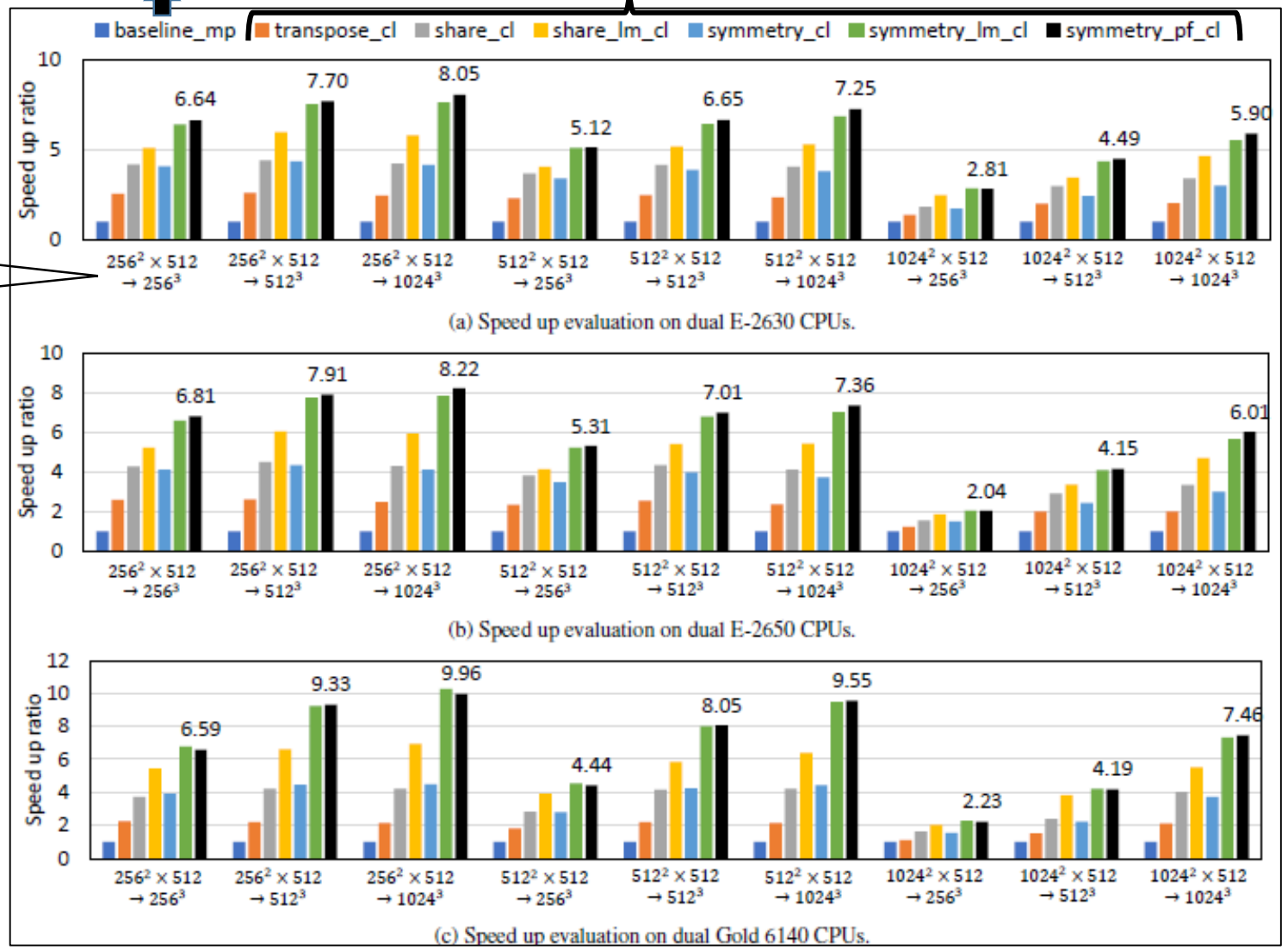
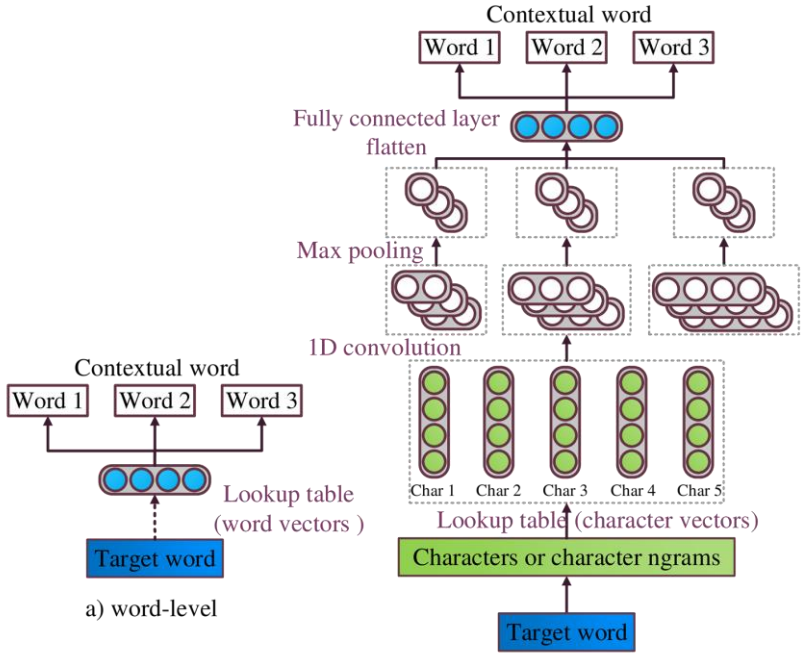


Table 1: Evaluation environment.

Intel CPU	E5-2630 v4	E5-2650 v3	Gold 6140
Cores	10	10	18
Threads/core	2	2	1
Frequency	2.2GHz	2.3GHz	2.3GHz
Sockets	×2	×2	×2
Memory	256GB	256GB	192GB
OS	CentOS 7.4	CentOS 7.4	CentOS 7.4

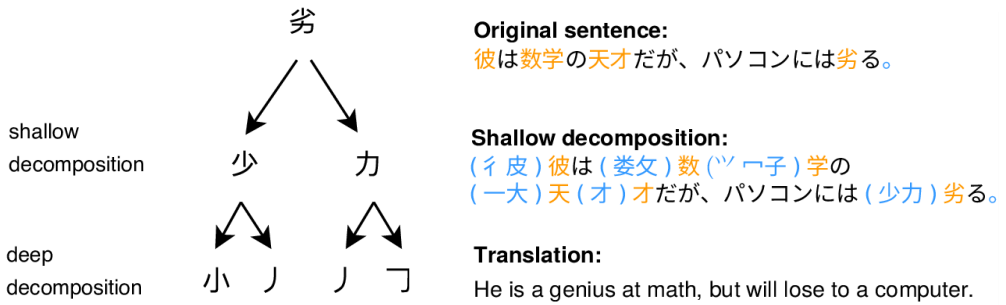
Advanced scalable NLP models [Drozd]



- Words come at different frequencies in texts
- With large batch size, frequent words are sampled many times per update
- This skews magnitudes of errors for different word embeddings
- Decreases convergence rate/ final accuracy

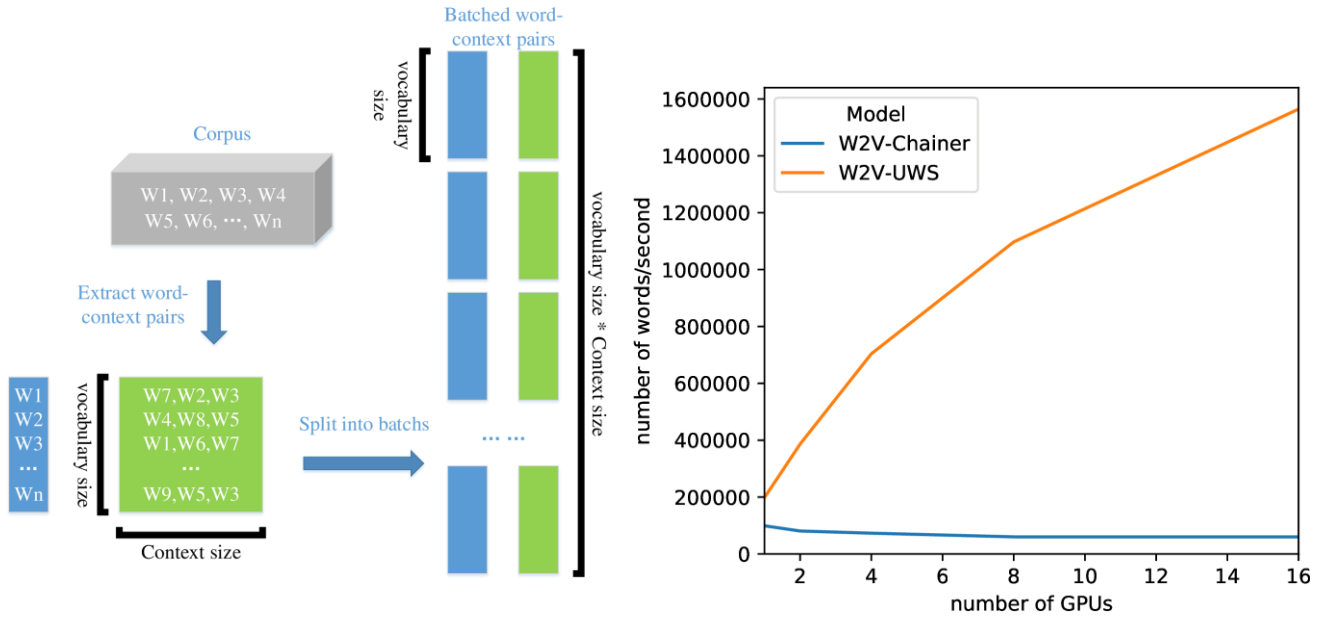
Mining samples with respect to word frequencies to address convergence issues on large batch

Neural net to compose subword elements



Sub-character models for Japanese characters

* in collaboration with U-Tokyo, U-Mass Lowell, Renmin University

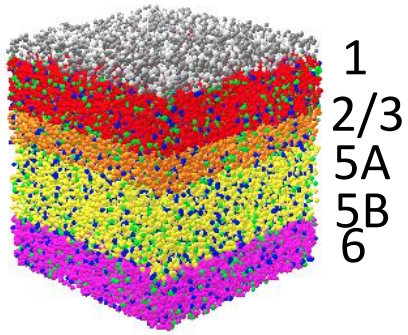


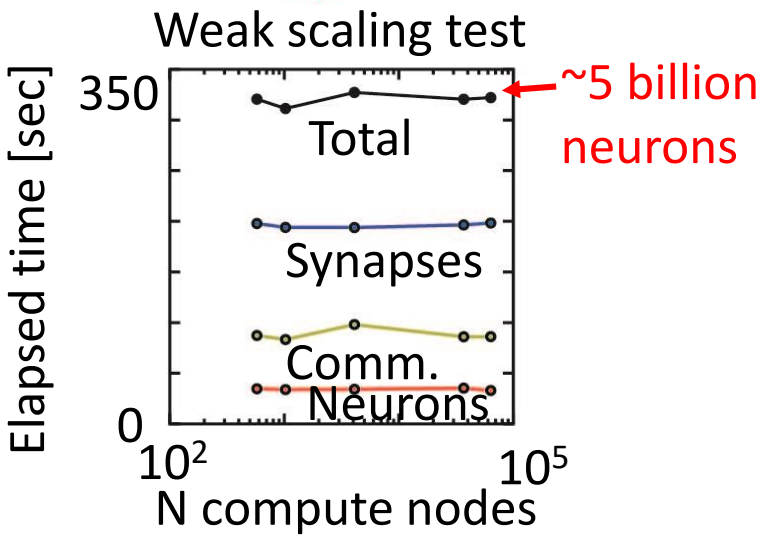
Large batch allows to scale efficiently

Large-scale simulations of the cortex and cerebellum on the K and Fugaku computers

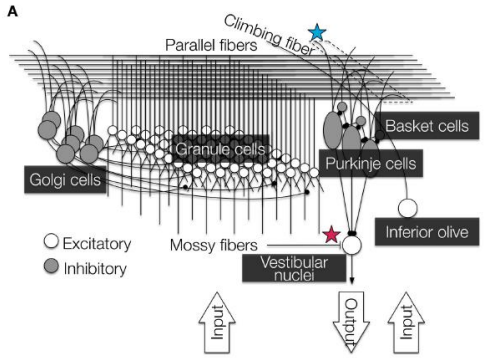
Jun Igarashi (Post-K Exploratory Challenge #4-1)

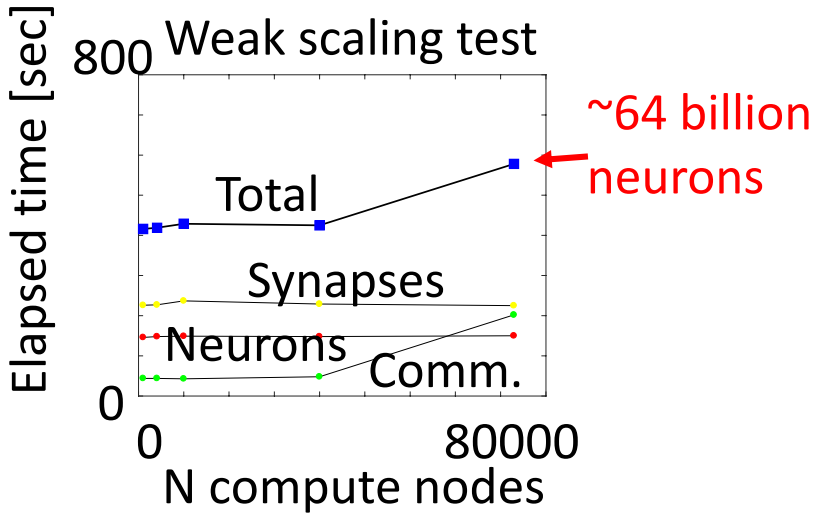
1/3 human-scale cortical simulation (Igarashi et al., 2019)



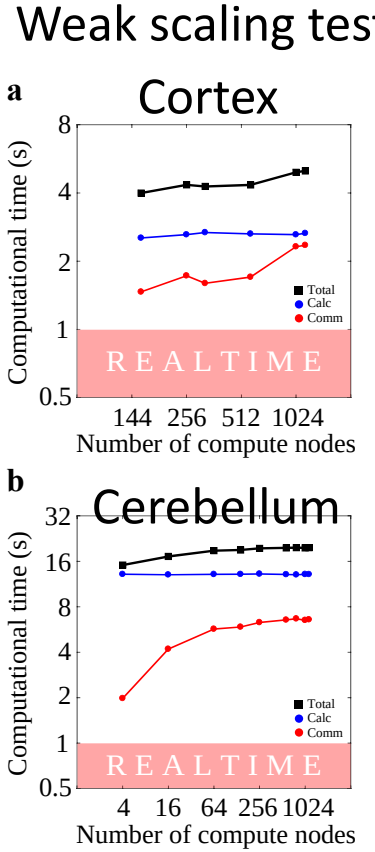


Human-scale cerebellar simulation (Yamaura et al., 2020)





Cortical & cerebellar simulations on the Fugaku test system

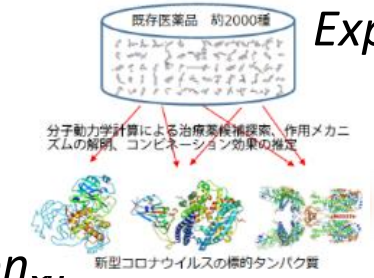


Our parallelization method using spatial partitioning method and communication frequency reduction utilizing signal delay realized excellent scaling performance and primate-scale simulations on the K (left and center). The proposed method showed 30~185 times faster simulations on the Fugaku test system than the K (right).

MEXT Fugaku Program: Fight Against COVID19

Fugaku resources made available a year ahead of general production (more research topics under international solicitation, joining US-lead COVID-19 High Performance Computing Consortium)

Medical-Pharma

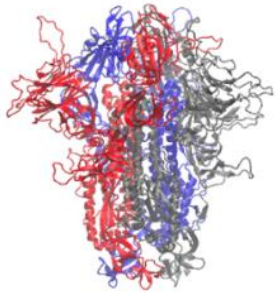


Exploring new drug candidates for COVID-19

Large-scale MD to search & identify therapeutic drug candidates showing high affinity for COVID-19 target proteins from 2000 existing drugs

(Yasushi Okuno, RIKEN / Kyoto University)

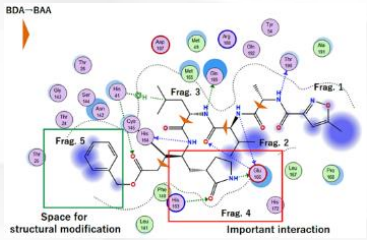
Prediction of conformational dynamics of proteins on the surface of SARS-Cov-2



GENESIS MD to interpolate unknown experimentally undetectable dynamic behavior of spike proteins, whose static behavior has been identified via Cryo-EM

(Yuji Sugita, RIKEN)

Fragment molecular orbital calculations for COVID-19 proteins



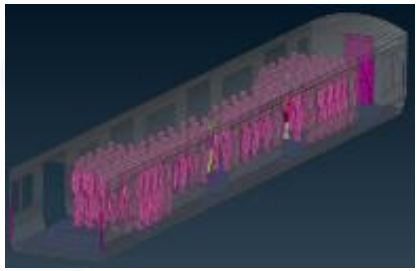
Large-scale, detailed interaction analysis of COVID-19 using Fragment Molecular Orbital (FMO) calculations using ABINIT-MP

(Yuji Mochizuki, Rikkyo University)

Societal-Epidemiology

Prediction and Countermeasure for Virus Droplet Infection under the Indoor Environment

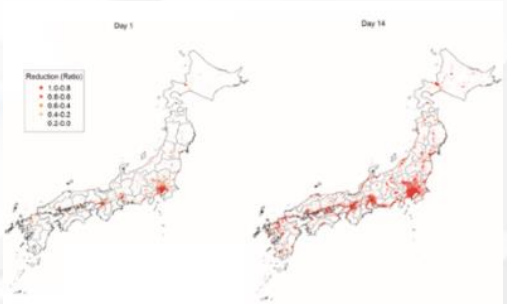
Massive parallel simulation of droplet scattering with airflow and hat transfer under indoor environment such as commuter trains, offices, classrooms, and hospital rooms



(Makoto Tsubokura, RIKEN / Kobe University)

Simulation analysis of pandemic phenomena

Combining simulations & analytics of disease propagation w/contact tracing apps, economic effects of lockdown, and reflections social media, for effective mitigation policies

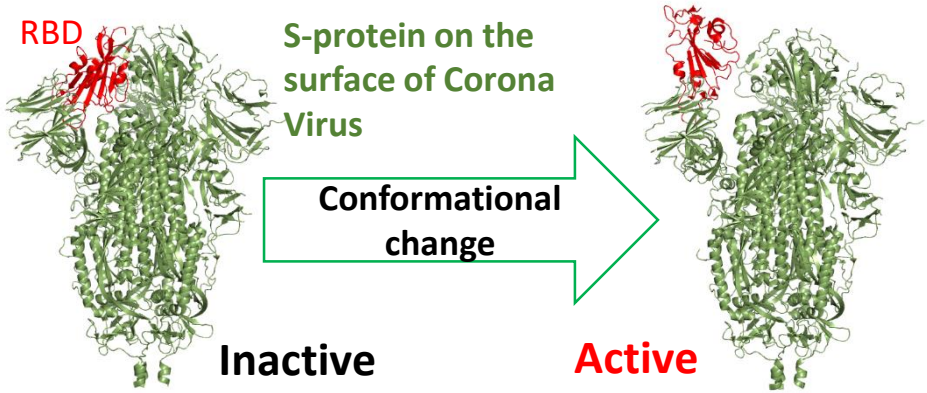


(Nobuyasu Ito, RIKEN)

Conformational Changes of S-protein in Corona Virus

(Yuji Sugita, Riken R-CCS)

Structures determined by cryo-EM (China, US)

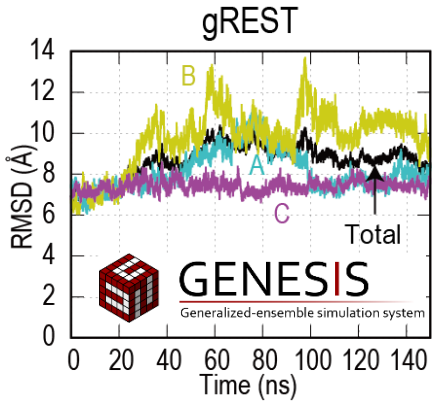


Closed (6VXX)

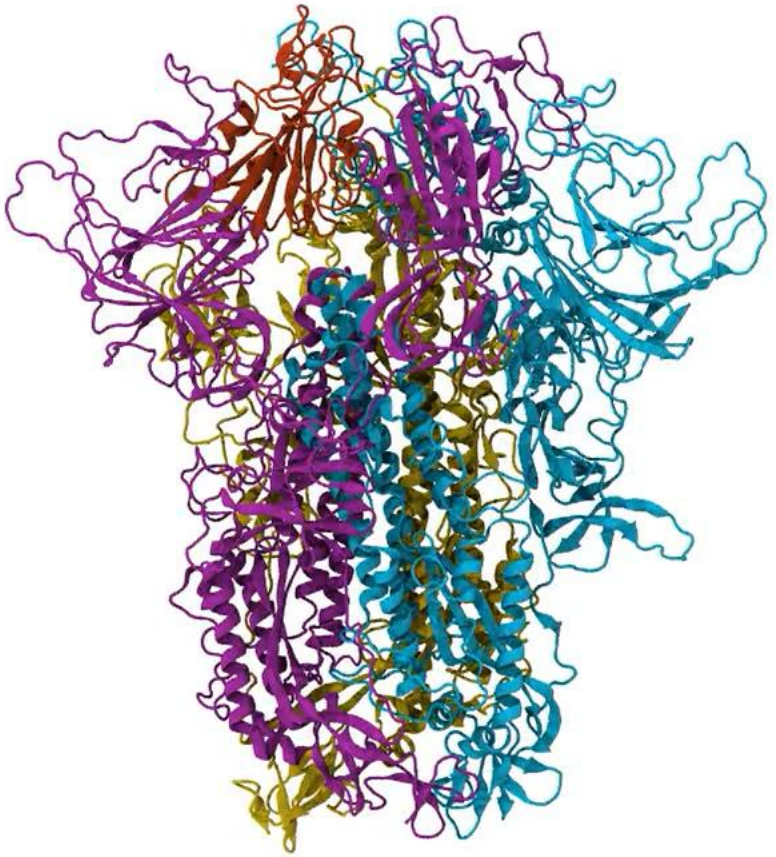
Up (6VYB)

gREST method in the GENESIS software allowed us to simulate conformational changes from inactive to active states.

The GENESIS software has been developed by RIKEN for K computer and Fugaku.



MD simulations on Fugaku (RIKEN)



Molecular mechanisms for the conformational changes of S-protein are necessary to develop drugs for Corona Virus.

Fragment molecular orbital calculations for COVID-19 proteins

Yuji Mochizuki* (Rikkyo Univ.), Shigenori Tanaka (Kobe Univ.), Kaori Fukuzawa (Hoshi Univ.)

Objective

A series of fragment molecular orbital (FMO) calculations are carried out on SARS-CoV-2/COVID-19 proteins, by using our ABINIT-MP program which is efficiently parallelized. Detailed analyses are then made.

Capacity computing

The first theme of this project has been set on the main protease (Mpro) in a capacity computing context. Based on the N3 ligand - Mpro comlex (PDB-id 6LU7), molecular dynamics (MD) simulations were performed to generate a number of sample structures. These structures with hydration waters (1700 fragments) were subjected to the FMO calculations at the second-order perturbation level, FMO-MP2/6-31G*. Typical timing for a single sample is only 0.6 h on a half rack (192 nodes). This kind of massive processing was difficult to do on the K-computer. About 2000 sample calculations were already completed, and statistical analyses on ligand - amino acid residue have been in progress. Importance of structural fluctuation is shown as follows.

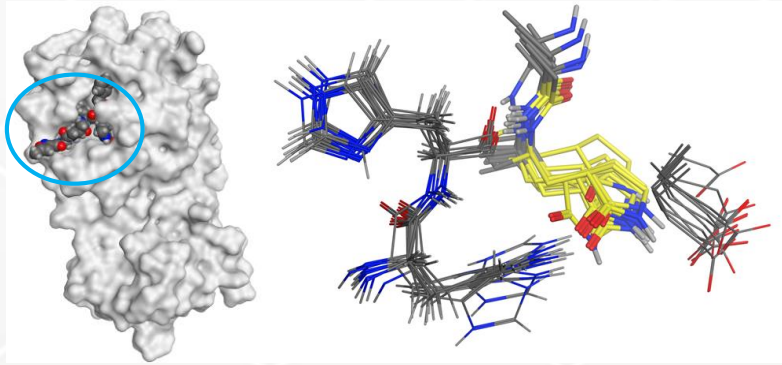


Fig.1. Left: N3 ligand – Mpro complex. Right: Superimposed MD structures of Fragment 4 of N3.

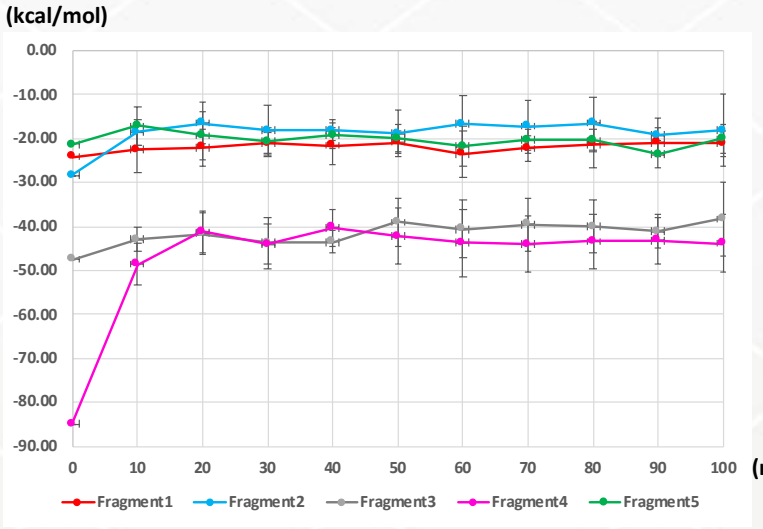


Fig.2. MD structure-based interaction energies of five Fragments in N3.

Capability computing

The second theme of this project concerns the spike protein whose number of amino acid residues are as many as 3300, and the corresponding jobs are processed rather in a capability computing context. Both closed and open forms of spike protein (PDB-id 6VXX and 6VYB, respectively) were calculated at the third-order perturbation level with a better basis set, FMO-MP3/cc-pVDZ. In fact, the enhanced computing power of Fugaku-computer (relative to K-computer) made this world largest FMO calculation possible. A job timing with 8 racks (3072 nodes) for 6VXX model was only 3.4 h. The difference in inter-unit (A-B, A-C and B-C) interaction energies between the closed and open forms is significant, and this difference should be correlated with the fact that the latter binds better with ACE2 (angiotensin-converting enzyme 2) of human cell surface. Similar difference is observed for the receptor binding domain (RBD) consistently. Brief summary of the present analyses (FMO-MP3/cc-pVDZ) is given below.

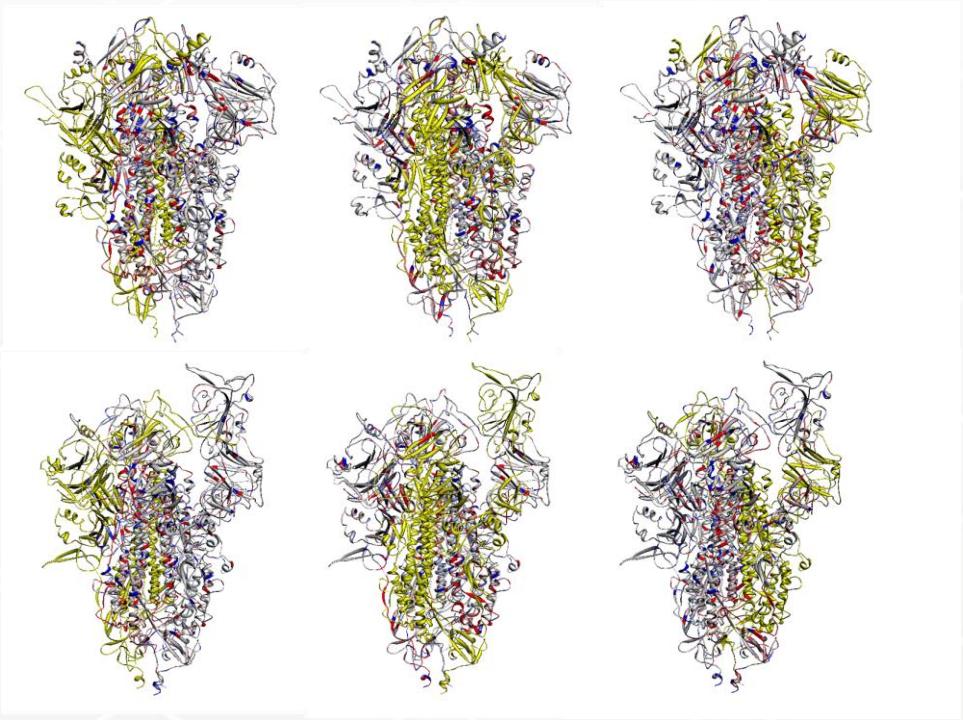


Fig.3. Visualized inter-unit interaction energies: upper (6VXX) & lower (6VYB).

Table 1. Job time (h) with 8 racks (3072 nodes).

	MP3/6-31G*	MP3/cc-pVDZ
6VXX (closed)	1.9	3.4
6VYB (open)	2.0	3.9

Table 2. Unit pair energies (kcal/mol).

Unit pair	6VXX (closed)	6VYB (open)
A-B	-2405.0	-1991.6
A-C	-2475.1	-2208.0
B-C	-2537.4	-1730.8

Table 3. RDB interaction energies (kcal/mol).

RBD	6VXX (closed)	6VYB (open)
A	-1636.4	-1142.1
B	-1695.1	-197.7
C	-1634.2	-1206.0

Members of our project

- Rikkyo University
  Yuji Mochizuki* (Prof.), Koji Okuwaki (Assist. Prof.), Ryo Hatada (M2), Kazuki Akisawa (B4)

- Hoshi University
  Kaori Fukuzawa (Ass. Prof.), Yusuke Kawashima (Assist. Prof.), Yuma Handa (D1)

- Kobe University
  Shigenori Tanaka (Prof.)

- National Institute of Advanced Industrial Science and Technology (AIST)
  Yuto Komeiji (Principal Researcher)

- Foundation for Computational Science (FOCUS)
  Kota Sakakura (Manager)

- HPC Systems Inc.
  Hiromasa Watanabe (Manager)


Mochizuki


Okuwaki


Komeiji


Tanaka


Fukuzawa


Kawashima

Exploring new drug candidates for COVID-19 by "Fugaku"

RIKEN / Kyoto University　Yasushi OKUNO, Prof. PhD.

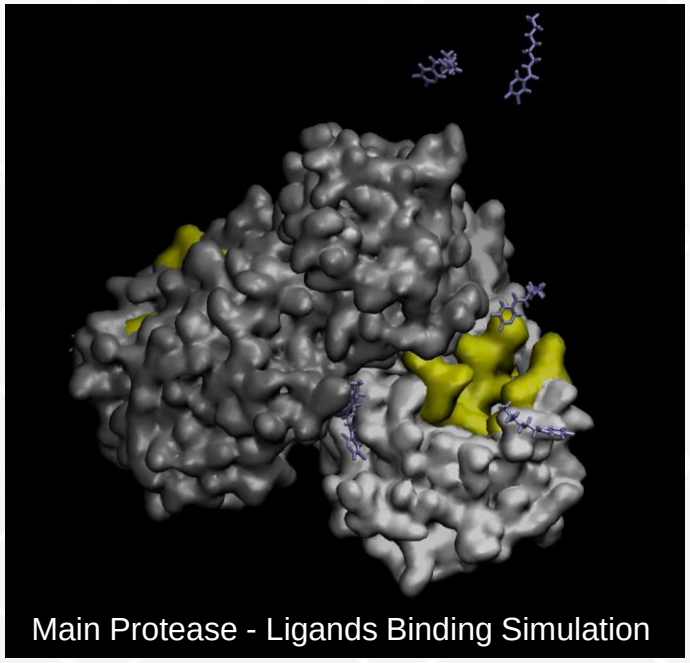
Research content:

Currently, clinical trials are underway in Japan and overseas to confirm the effects of existing drugs on COVID-19. Some reports have shown that the drug has shown efficacy through these clinical trials, but the number of cases has been small, and no effective therapeutic drug has yet been identified. Furthermore, due to the small number of drugs being tested, it is possible that none of the drugs have a definite effect.

Therefore, in this study, we performe molecular dynamics calculations using "Fugaku" to search and identify therapeutic drug candidates showing high affinity for the target proteins of COVID-19 from approximately 2,000 existing drugs that are not limited to existing antiviral drugs targeted in clinical trials.

Expected results:

- ✓ New therapeutic drug candidates other than those currently undergoing clinical trials can be discovered.
- ✓ Combination effects of multiple drugs can be estimated
- ✓ The molecular action mechanism of existing drugs currently undergoing clinical trials will be elucidated. In addition, these findings provide a clear direction for developing new drugs that go beyond the existing drugs.



Main Protease - Ligands Binding Simulation

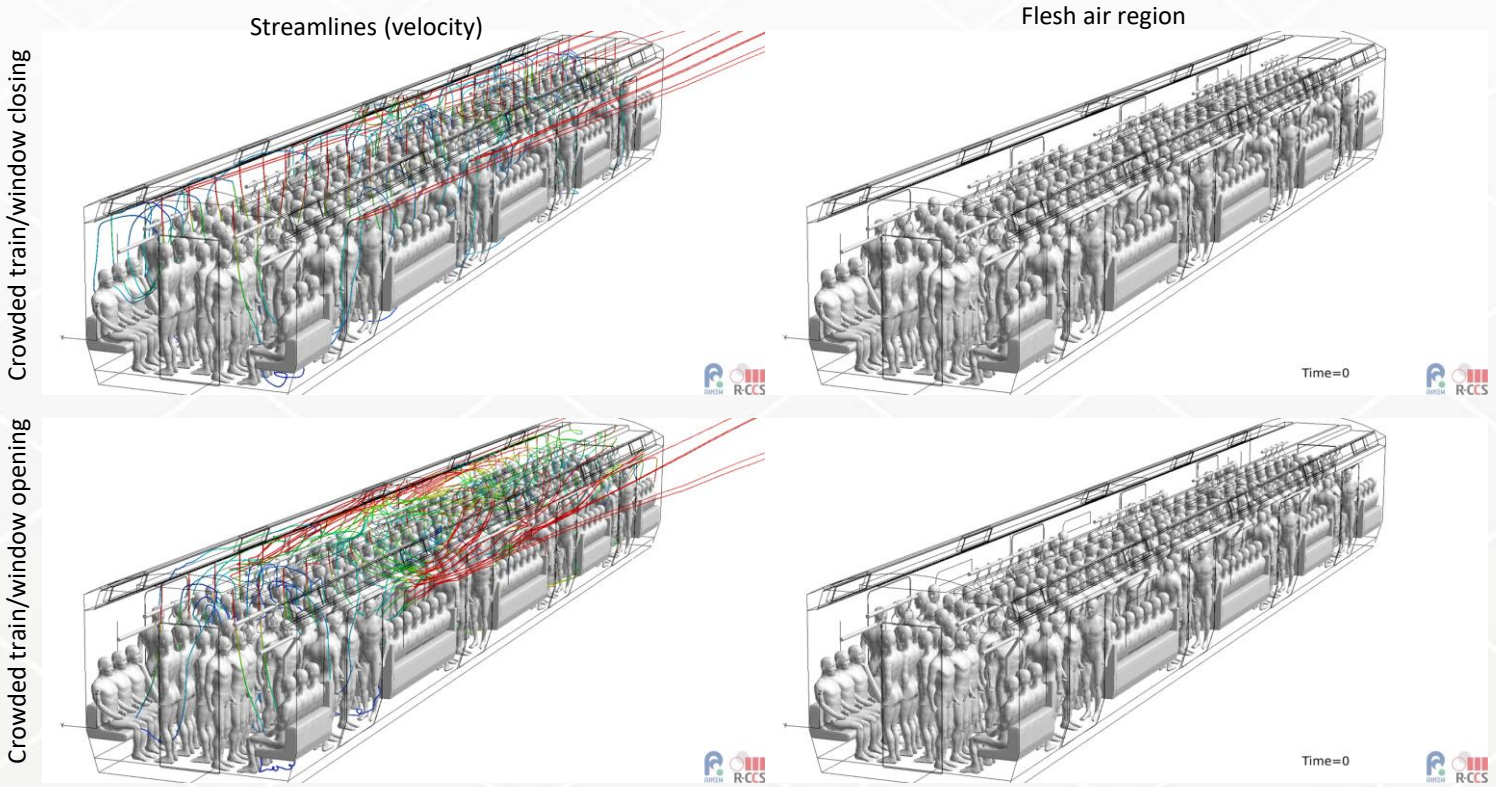
Prediction and Countermeasure for Virus Droplet Infection under the Indoor Environment

RIKEN R-CCS Makoto TSUBOKURA

Objective: Risk Assessment of Virus Droplets/Aerosol Infection
Targets: Train Cabin, Office, Hospital Room, Classroom
Methods: CFD with a Lagrangean droplet dispersion model including evaporation, reflection/attachment on the wall

Train Cabin

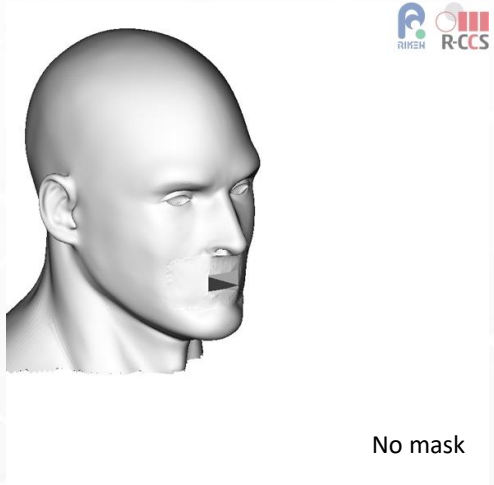
Train running 80km/h with/without four side windows opening/closing by solving air flows of both inside and outside the train.
Time for changing the inside air drastically decreased by four to five times by opening the windows.



マスク無し（動画）　　不織布マスク隙間あり（動画）　　不織布マスク隙間なし（動画）

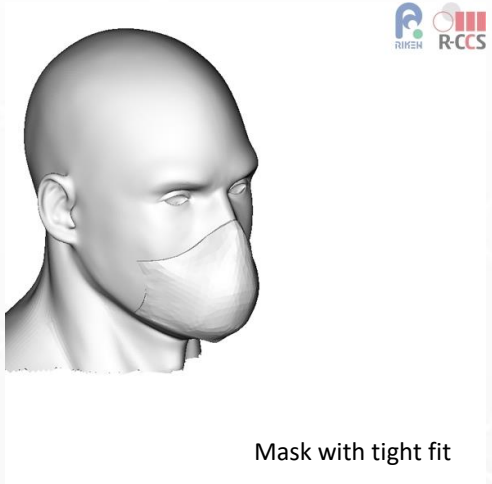
Office

Effect of mask and its fitting on the face in the case of cough.
Loose fit degrades the mask's performance by more than 40%



No mask

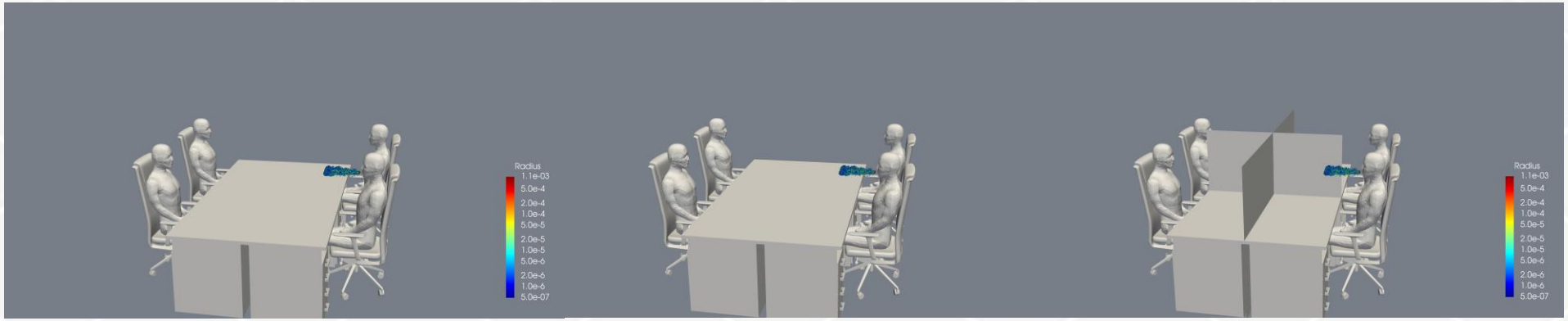
Mask with loose fit



Mask with tight fit

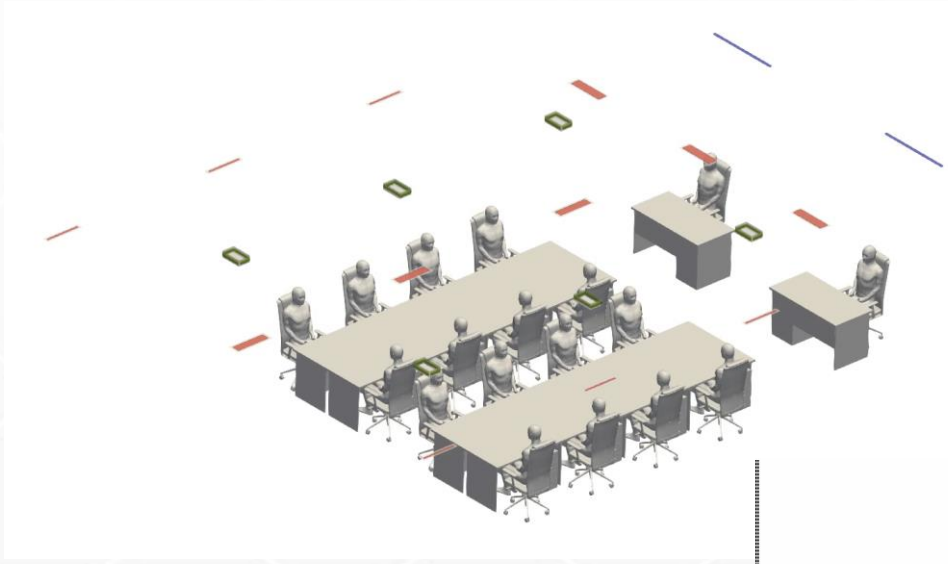
Four persons sitting in the office with/without partitions of 120cm and 140cm, one person cough and 50,000 droplets emitted in the air.
For the droplet infection, partition height of more than 140cm from the floor is effective, while for the aerosol infection, aerosol can easily go beyond the partition, thus another countermeasure is necessary.

No partition　　Partition, 120cm　　Partition, 140cm



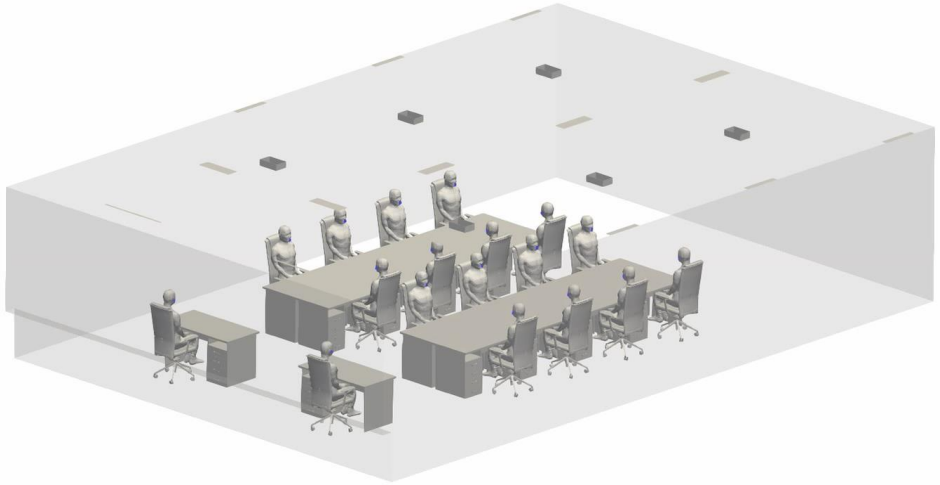
Office

Risk of aerosol infection in the small office with 18 persons.
Under the indoor environment with window closed, aerosol infection risk increases.



Airflow in the office
(Main flow driven by 8 AC and human temperature)

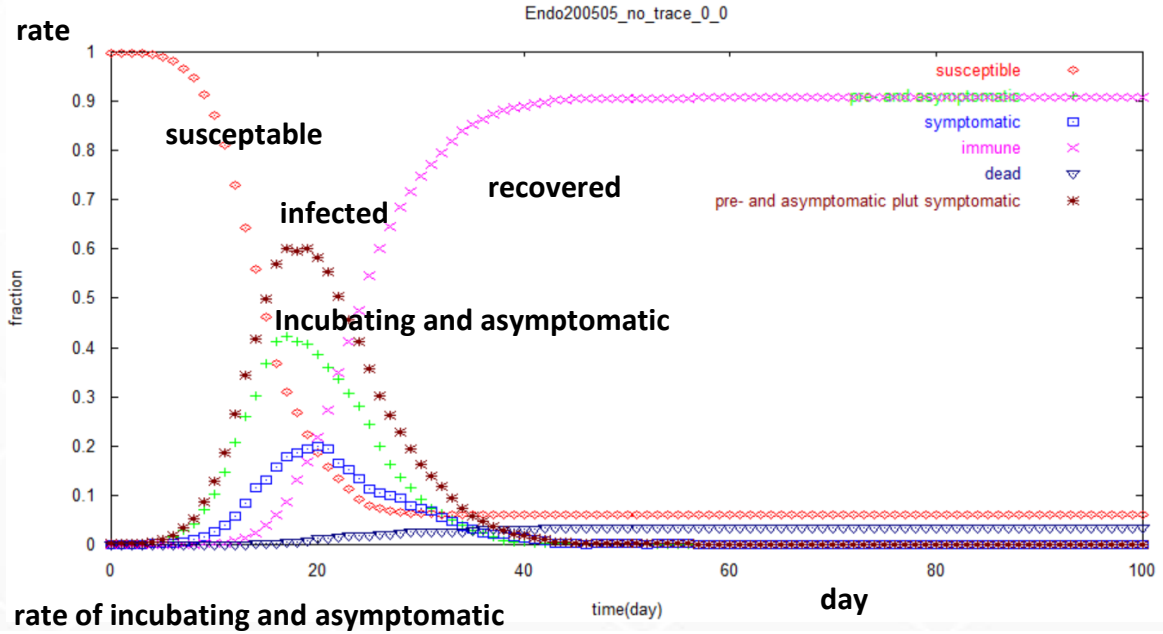
Virtual particles representing aerosol transmission



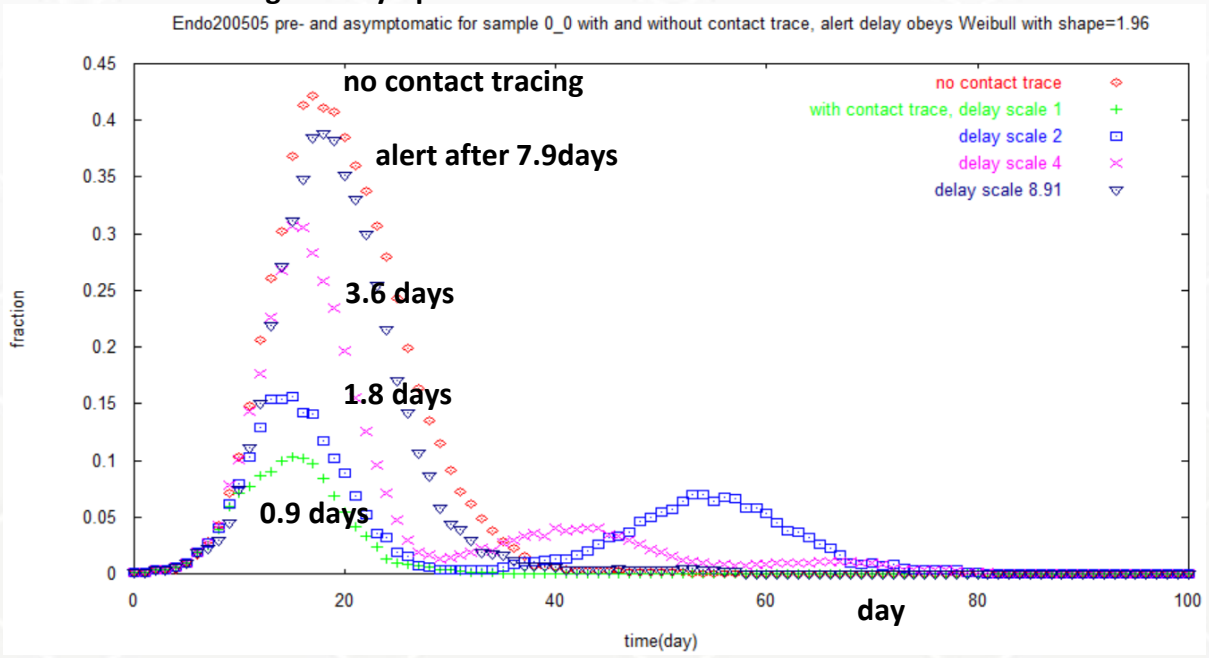
Infection dynamics and suppression with contact tracing
- simulation with agent-based model of the COVID19 infection -

RIKEN Nobuyasu Ito

rate



rate of incubating and asymptomatic



- 1,000 agents, contacting randomly within susceptible and presymptomatic agents.
- All except one are susceptible. The one has just infected at t=0, and shows symptom later.
- Basic reproduction number is set to be 2.5.
- Alert is issued after one becomes symptomatic and it is authorized. Time delay between start of symptom and authorization is assumed to obey Weibull distribution with shape 1.96 and various scale.
- Alert is for agents contacted in last 14 days.
- When an agent gets alert, it is isolated for the following 14 days.

←One set of agent parameters, named 0_0, without contact trace alert.

←Behavior of fractions of pre- and asymptomatic agents in cases of various conditions of contact trace. From top to bottom at the first peak around 15 – 20 days:

red ◇: no contact-trace alert

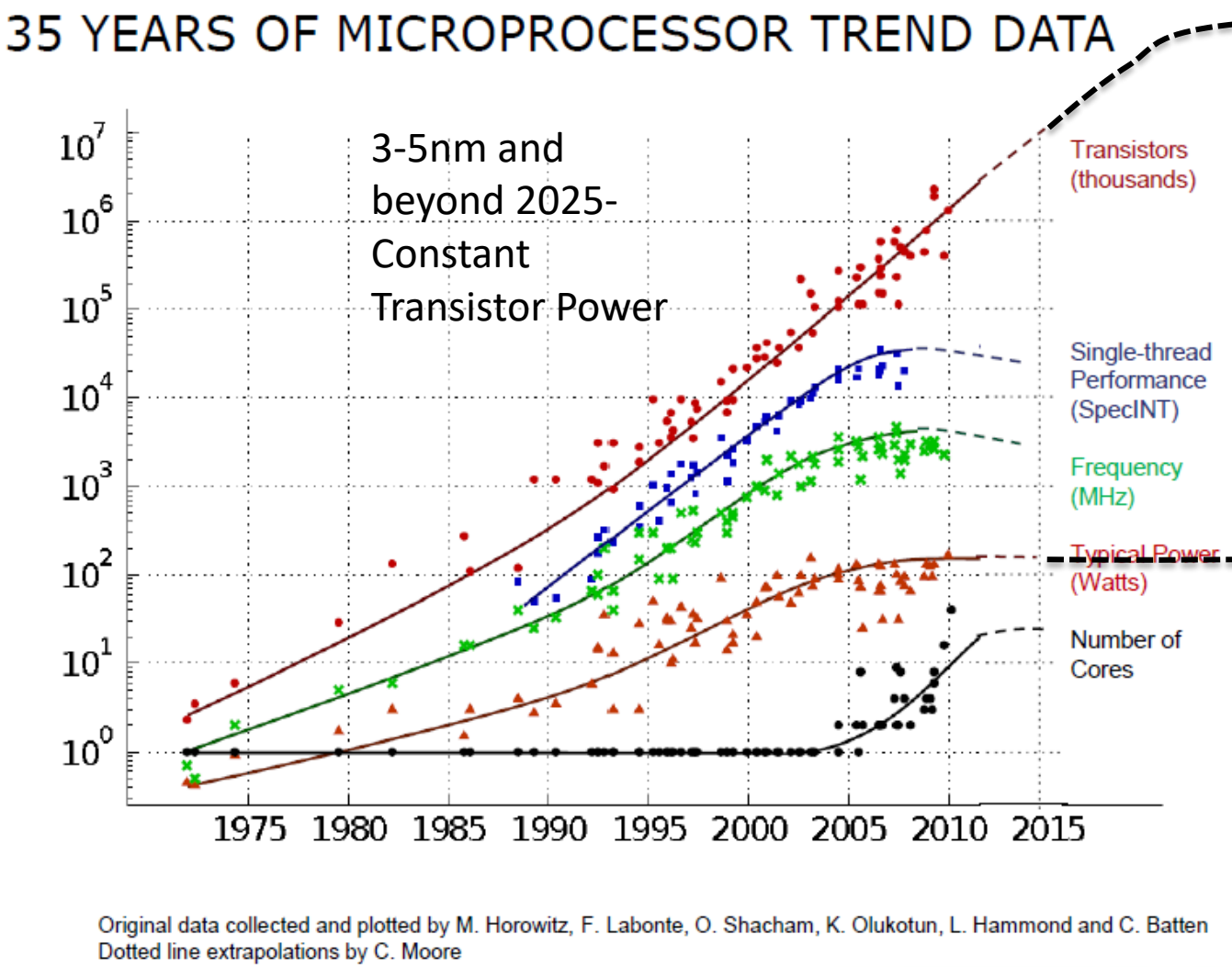
dark blue ▽: with contact-trace alert and delay time obeying Weibull with scale 8.91(7.9 days in average, current situation)

violet ×: with contact-trace alert and delay time obeying Weibull with scale 4 (3.55 days in average)

blue □: with contact-trace alert and delay time obeying Weibull with scale 2 (1.77 days in average)

right green ＋: with contact-trace alert and delay time obeying Weibull with scale 1 (0.89 days in average)

20 year Eras towards of End of Moore's Law



20-year Moore-Dennard Single Core ILP-Vector Killer-Micro Era

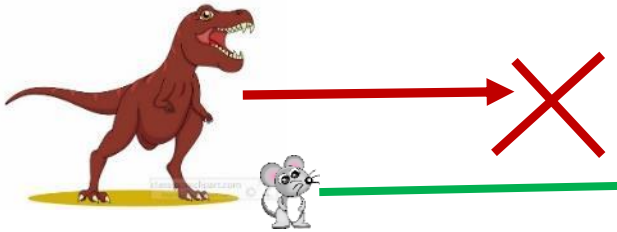
20 year Post-Dennard Many-Core Era

20-year Next-Gen Post-Moore era

- 1980s~2004 Dennard scaling, perf+ = single thread+ = transistor & freq+ = power+
- 2004~2015 feature scaling, perf+ = transistor+ = core#+, constant power
- 2015~2025 all above gets harder
- 2025~ post-Moore, **constant feature&power = flat performance**

Need to realize the next 20-year era of supercomputing

Many Core Era



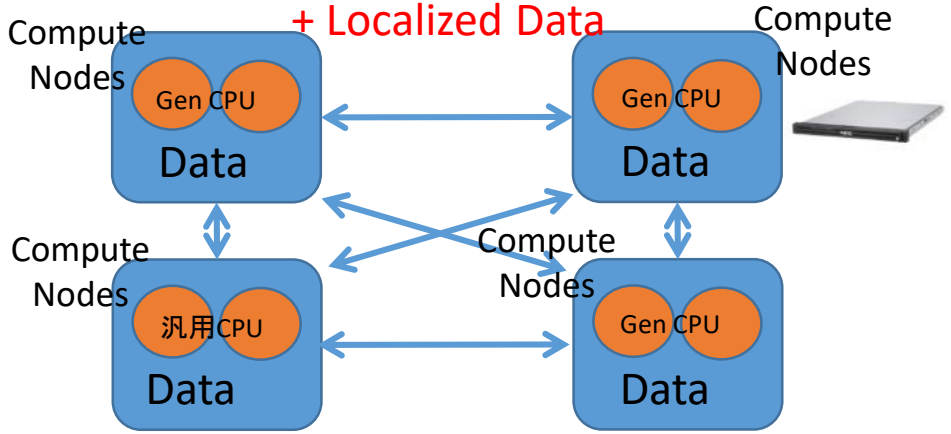
Post Moore
Cambrian Era

Flops-Centric Monolithic Algorithms and Apps

Flops-Centric Monolithic System Software

Hardware/Software System APIs
Flops-Centric Massively Parallel Architecture

Homogeneous General Purpose Nodes
+ Localized Data



Loosely Coupled with Electronic Interconnect

Transistor Lithography Scaling
(CMOS Logic Circuits, DRAM/SRAM)

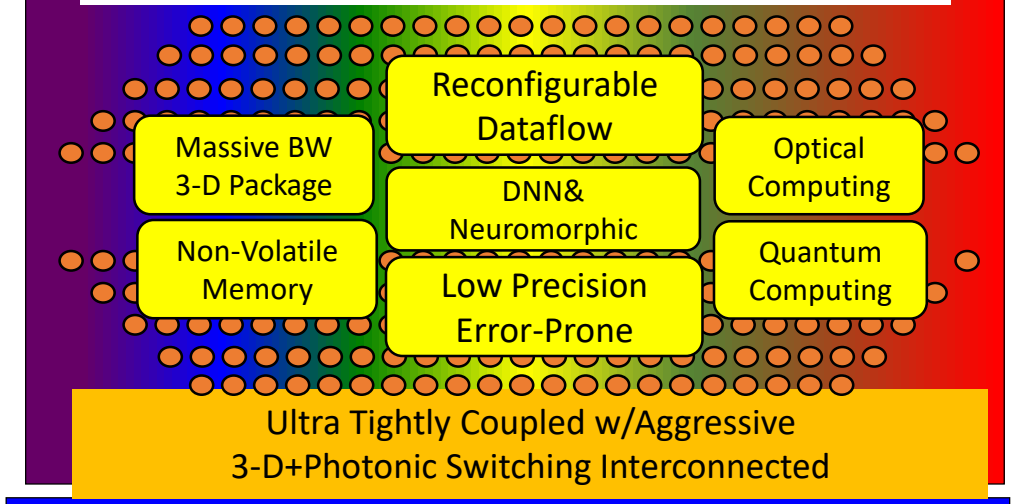
~2025
M-P Extinction Event

Cambrian Heterogeneous Algorithms and Apps

Cambrian Heterogeneous System Software

Hardware/Software System APIs
"Cambrian" Heterogeneous Architecture

Heterogeneous CPUs + Holistic Data



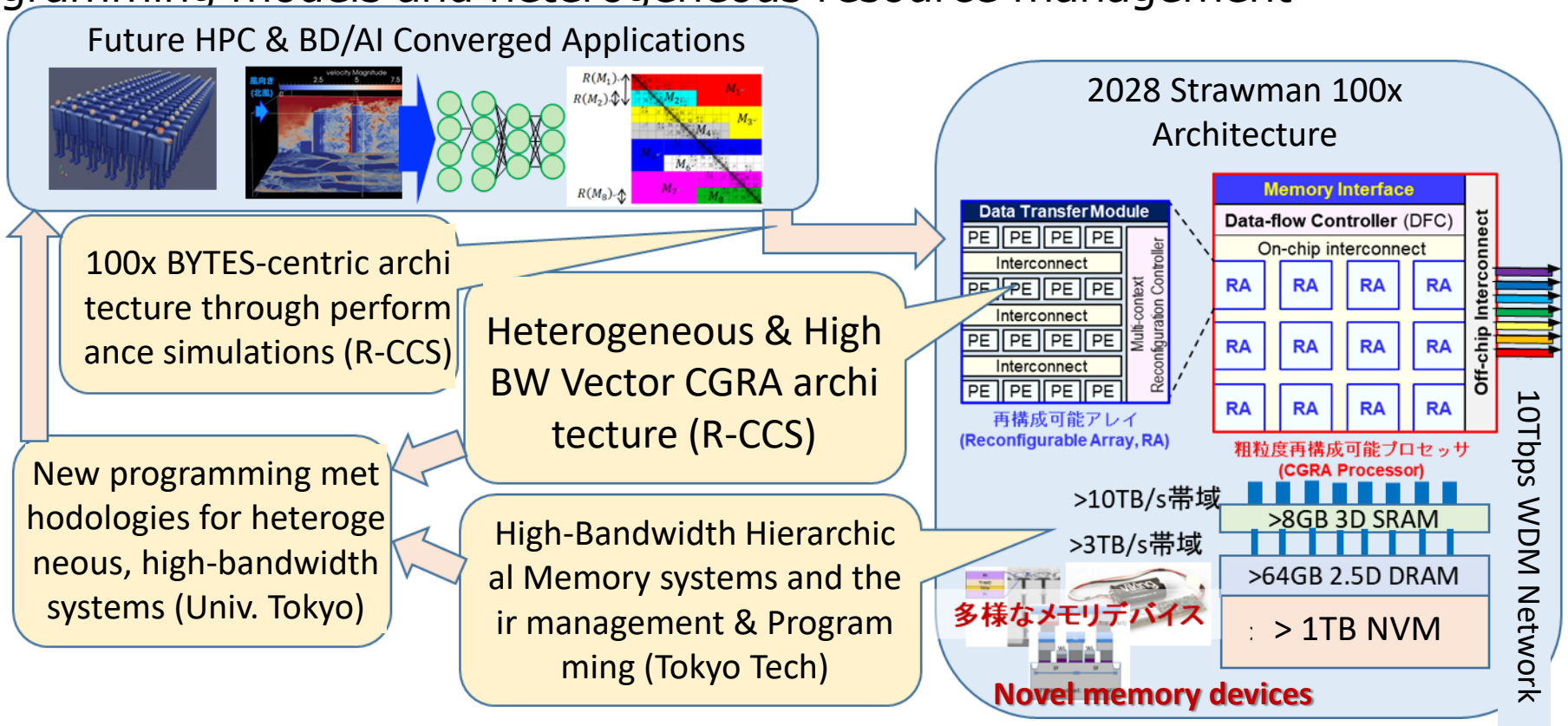
Ultra Tightly Coupled w/Aggressive
3-D+Photonic Switching Interconnected

Novel Devices + CMOS (Dark Silicon)
(Nanophotonics, Non-Volatile Devices etc.)

NEDO 100x Processor Project
Riken (R-CCS)/U-Tokyo/Tokyo Tech

Towards 100x processor in 2028

- Various combinations of CPU architectures, new memory devices and 3-D technologies
- Perf. measurement/characterization/models for high-BW intra-chip data movement
- Cost models and algorithms for horizontal & hierarchical data movement
- Programming models and heterogeneous resource management



12 Apr, 2019