

High Performance Computing for Lattice QCD

Hideo Matsufuru (KEK/SOKENDAI)



葛飾北斎 富嶽三十六景《深川万年橋下》

Hokusai, "Fukagawa Mannen-bashi shita" in Fugaku Thirty-six scenery

Nonperturbative Analysis of Quantum Field Theory and its Applications

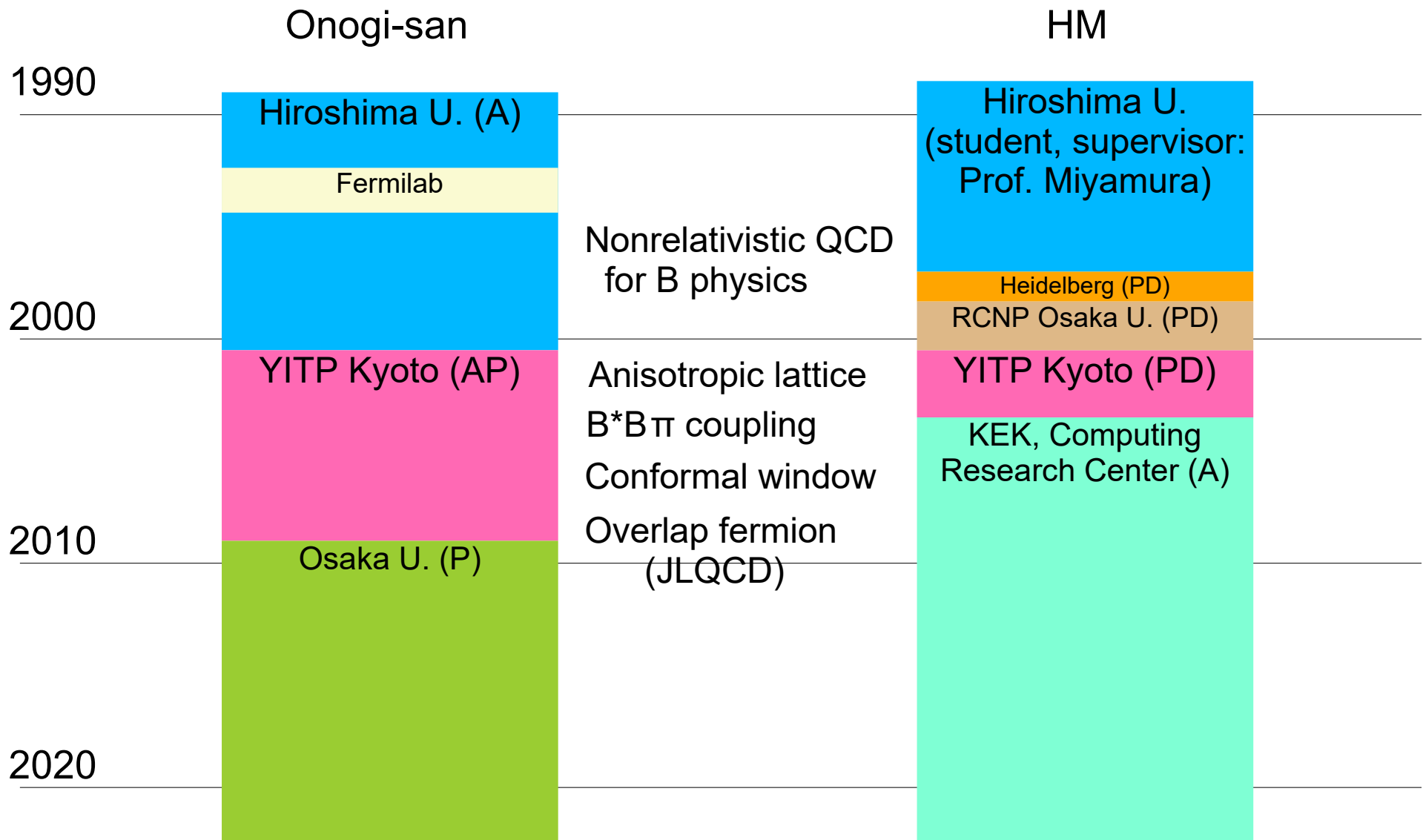
22 September 2022, Osaka University, Japan

Contents

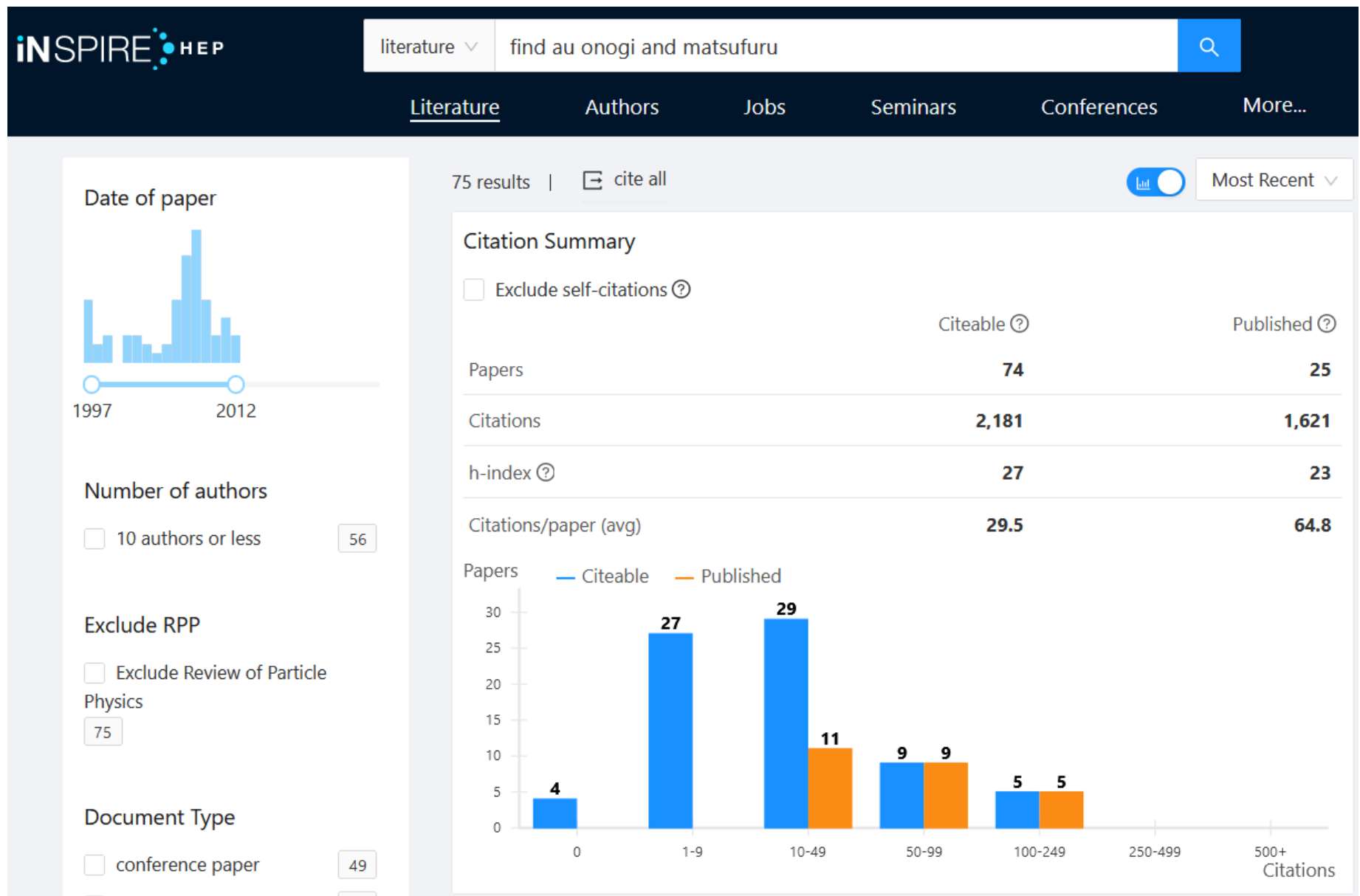
Congratulations for Onogi-san's 60th birthday!

- Onogi-san&!
- Lattice QCD and high performance computing
- Development of Bridge++ code set

Onogi-san&!



Onogi-san&!



Onogi-san&

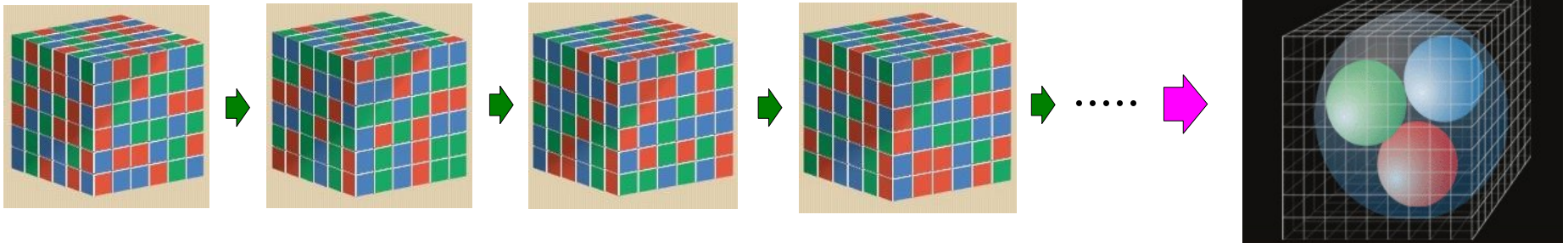
(private photos were removed)

Onogi-san&

(private photos were removed)

Lattice QCD simulations

- Numerical simulations of lattice QCD
 - Gauge and fermion fields on 4D Euclidean lattice
 - Generation of “gauge configurations”
 - Measurement of generated configurations



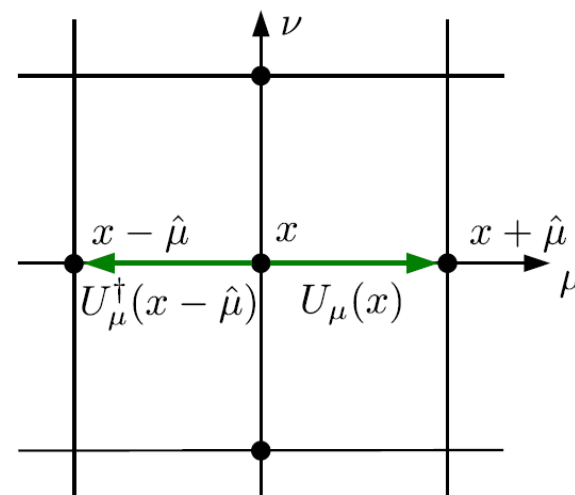
- Hybrid Monte Carlo algorithm
 - At each step of molecular dynamical evolution, a linear equation for fermion matrix must be solved
 - Large sparse complex matrix → iterative solver
 - rank: 3 (color) x 4 (spinor) x #site ($O(10^6)$)
- As decreasing quark mass, iterative solver becomes increasingly time consuming → **Bottleneck of lattice QCD simulations**

Lattice QCD simulations

- Example of fermion matrix: Wilson fermion

$$D_{x,y} = \delta_{x,y} - \kappa \sum_{\mu} \{ (1 - \gamma_{\mu}) U_{\mu}(x) \delta_{x+\hat{\mu},y} + (1 + \gamma_{\mu}) U_{\mu}(x - \hat{\mu}) \delta_{x-\hat{\mu},y} \}$$

- $U_{\mu}(x)$: 3x3 complex matrix (gauge field)
- γ_{μ} : 4x4 matrix (permutation of components)
- κ : hopping parameter (related to quark mass)
- **Coupling with nearest neighbor sites**
- Chiral symmetry is explicitly broken



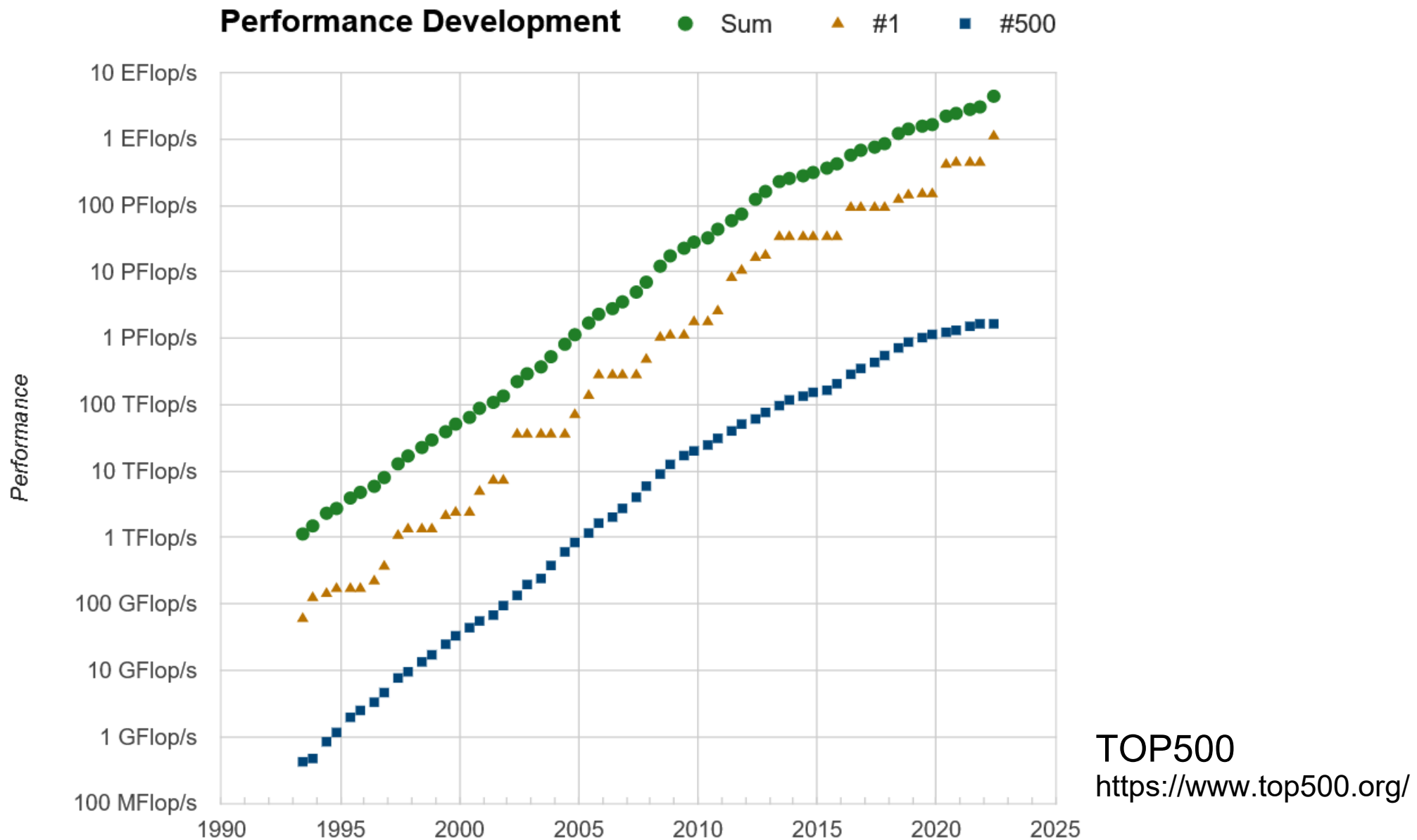
- There is variety of fermion matrices

- Clover fermion (improved Wilson)
- Staggered fermion (less cost, remnant chiral symmetry)
- Overlap/Domain-wall fermion (good chiral symmetry, high cost)

Lattice QCD and HPC

- Quantitative calculation requires large-scale simulations
 - For such as flavor physics and QCD phase diagram
 - Employing latest supercomputer/architecture
 - Leading development of machines (QCDPAX, CP-PACS, QCDOC, etc.)
- As computers develop, possible subjects have been extended
 - As computational power increases 1000 times, qualitatively different study becomes possible
 - Quenched simulations
 - dynamical simulations
 - physical quark masses
 - large volume for nucleon interaction

High Performance Computing



High Performance Computing

- **Development of architecture**
 - Parallel clusters: distributed memory nodes bound by fast interconnect
 - Vector processor (vector instruction/registers)
 - Multi-core with shared memory (multi-thread)
 - Hybrid parallelization with distributed/shared memory
 - Accelerators (CELL B.E., GPU, PEZY-SC)
 - SIMD (Single Instruction/Multiple Data) processors (Intel Xeon, Xeon-Phi, A64FX)

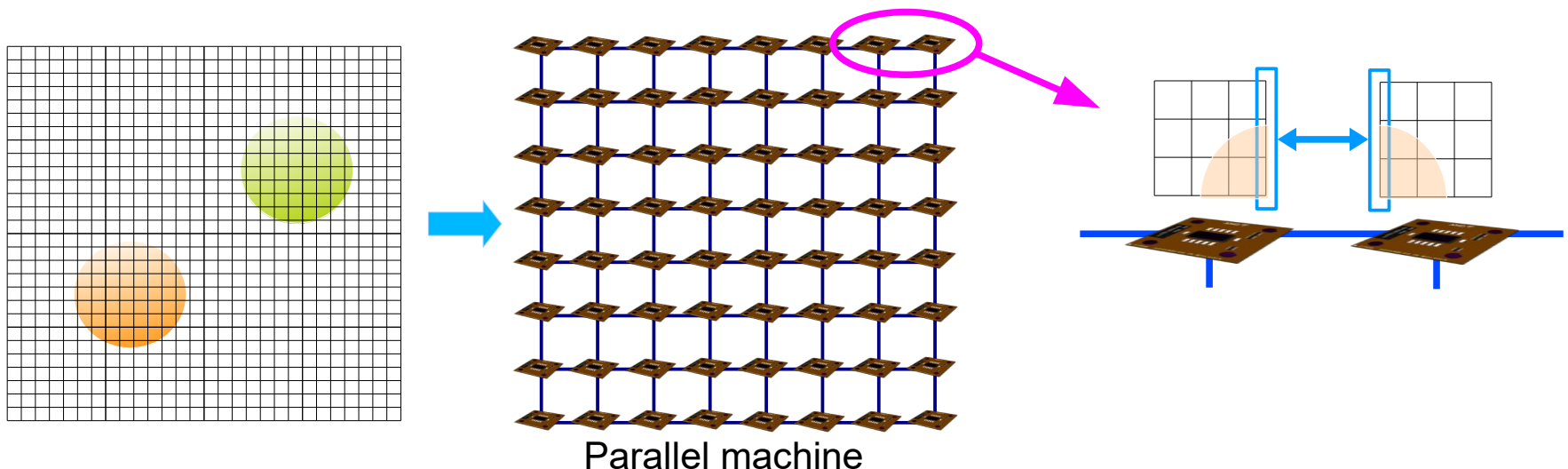
Massive Parallel

- Parallel systems

- My first supercomputer:
Intel Paragon at Hiroshima Univ. (56 nodes, 4.2 GFlops)
- Distributed memory
- Communication with message passing (MPI)



<https://fukuyama.hiroshima-u.ac.jp/super/insamz.html>



Massive Parallel



Blue Gene/L @KEK



Blue Gene/Q @KEK



Oakforest-PACS @JCAHPC

<https://www.cc.u-tokyo.ac.jp/supercomputer/ofp/service/>

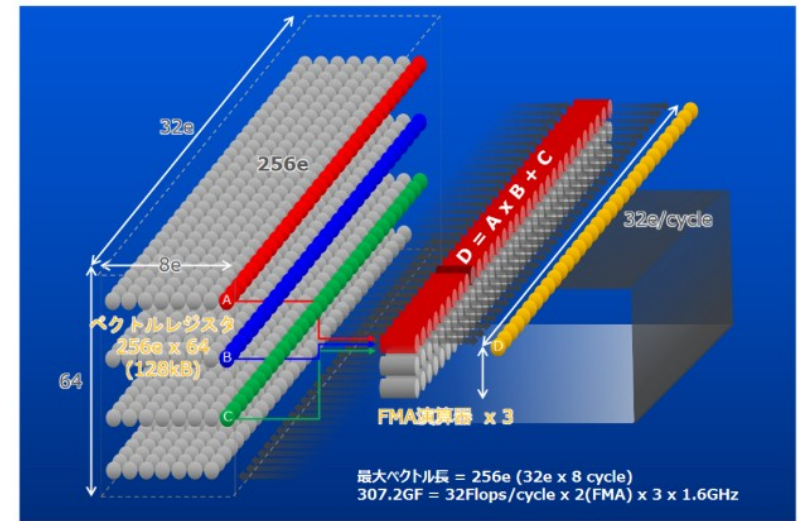


Fugaku @RIKEN CCS

<https://www.r-ccs.riken.jp/en/fugaku/>

Vector Processor

- Vector processor
 - Vector instruction
 - Large vector registers
 - Pipelined arithmetic operations
 - High memory bandwidth
 - Latest architecture:
NEC SX-Aurora's vector length = 512



https://www.pccluster.org/ja/event/2017/12/pcccSymp17_nec.pdf



NEC SX-5 @Osaka Univ.

<http://www.hpc.cmc.osaka-u.ac.jp/sx-5/>



NEC SX-9 @Osaka Univ.

<http://www.hpc.cmc.osaka-u.ac.jp/sx-9/>

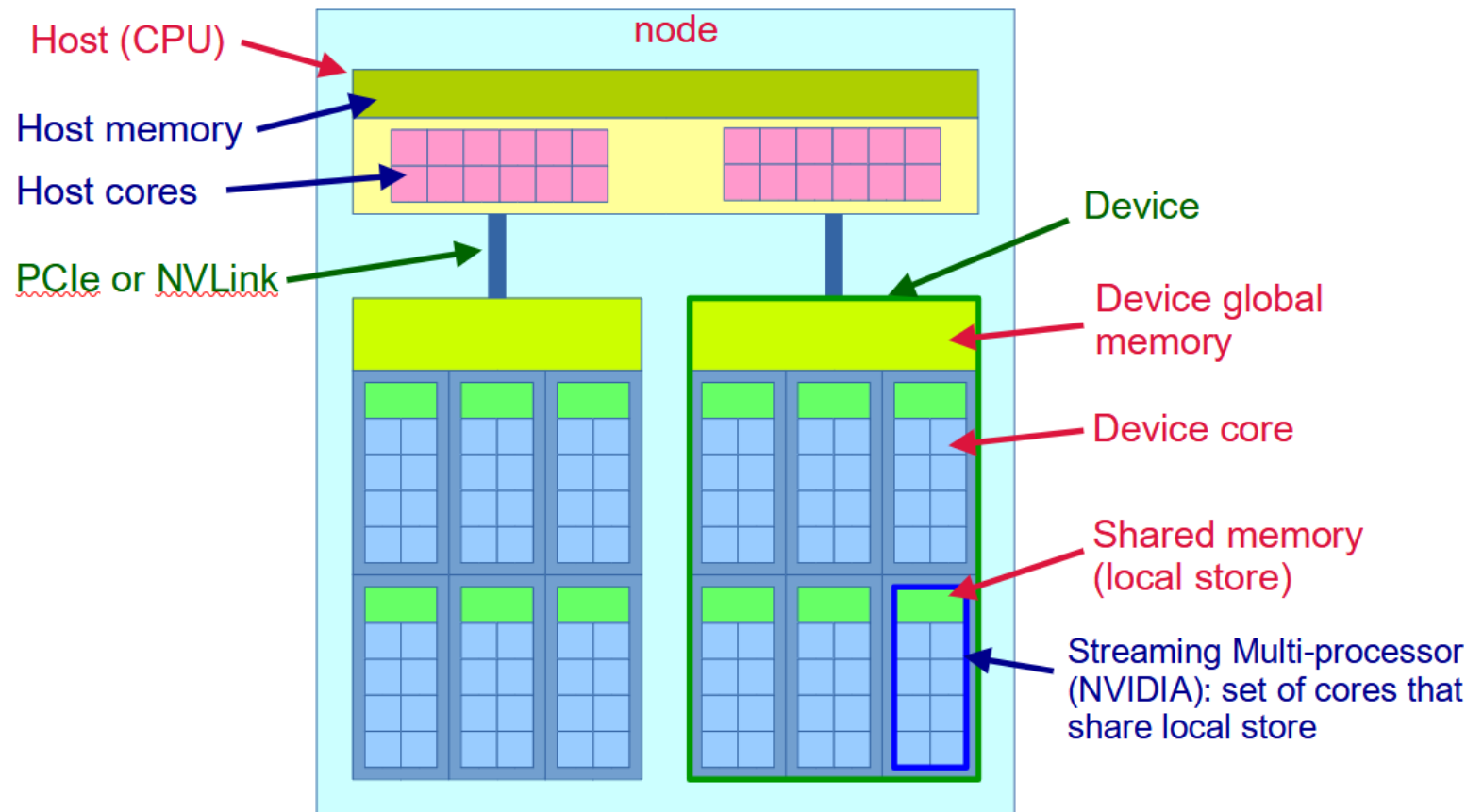


NEC SX-Aurora @KEK

Accelerators

- Accelerator

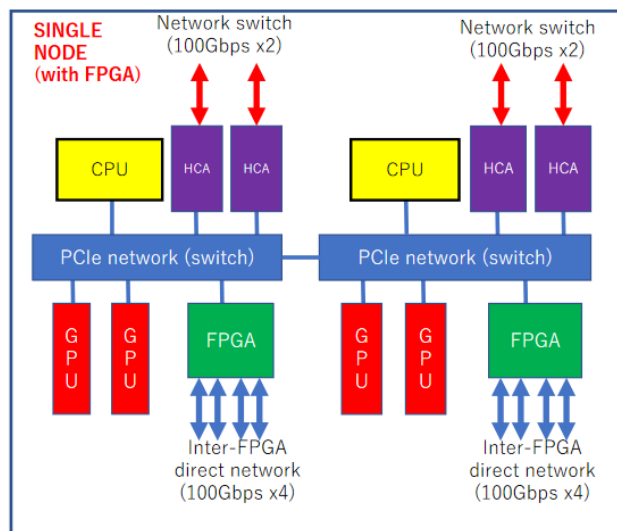
- Bottleneck tasks are offloaded to accelerator devices
- Many-core architecture in device
- Data transfer between host and device is bottleneck



Accelerators



Cygnus @U. CCS, Tsukuba



<https://www.ccs.tsukuba.ac.jp/wp-content/uploads/sites/14/2021/07/Cygnus.pdf>



Sui ren2 @KEK

Lattice QCD and HPC

- While fast computers are welcome, we also need a code that makes use of new architecture
 - Distributed parallelization → MPI (Message Passing Interface)
 - Shared memory parallelization → OpenMP
 - Offloading to accelerators → OpenACC, OpenCL, CUDA
 - Vector processor → change of data layout
 - SIMD processor → change of data layout, intrinsics, inline assembly
 - Optimization techniques: unrolling, prefetch, etc.
 - Demand of general purpose code set
 - Common basis for code development
 - Readable and extensible framework/toolkit
- Bridge++ project
- Launched as a project of 2008 Grant-in-Aid for Scientific Research on Innovative Areas “Research on the Emergence of Hierarchical Structure of Matter by *Bridging* Particle, Nuclear and Astrophysics in Computational Science” conducted by Sinya Aoki

Bridge++

Lattice QCD code Bridge++

- General purpose code set for simulations of lattice gauge theory
- C++, object-oriented design
- Development policy
 - Readable: for beginners
 - Extendable: for testing new ideas
 - Portable: works on many machines
 - Practically enough high performance
- History
 - Project launched in October 2009, first public release in July 2012
 - Latest version: 1.6.1 (15 June 2021)
 - Now preparing for release of ver.2.0



Y.Akahoshi et al., J. Phys.: Conf. Ser. 2207, 012053 (2021)

Bridge++

- **Current members**

S. Aoki (YITP), T. Aoyama (ISSP Tokyo U.), I. Kanamori (Riken R-CCS),
K. Kanaya (U. Tsukuba), H. Matsufuru (KEK), Y. Namekawa (Hiroshima U.)
H. Nemura (RCNP)

- **With memories of Yusuke Taniguchi**

(private photos are removed)

Bridge++

- **Implementation (ver.1.x)**
 - Parallelized by MPI (possible to replace with a low-level library)
 - Multi-threaded by OpenMP
 - Algorithms are generally implemented making use of polymorphism
 - Guided by Design Patterns
- **Examples of implemented algorithms**
 - Hybrid Monte Carlo with arbitrary nested integrators, RHMC
 - Fermion operators with link smearing (APE, HYP)
 - Many linear equation solvers, Implicitly restarted Lanczos eigensolver
 - Gradient flow

Restriction

- Fixed data layout
- Double precision
- Requirement of optimization for each architecture

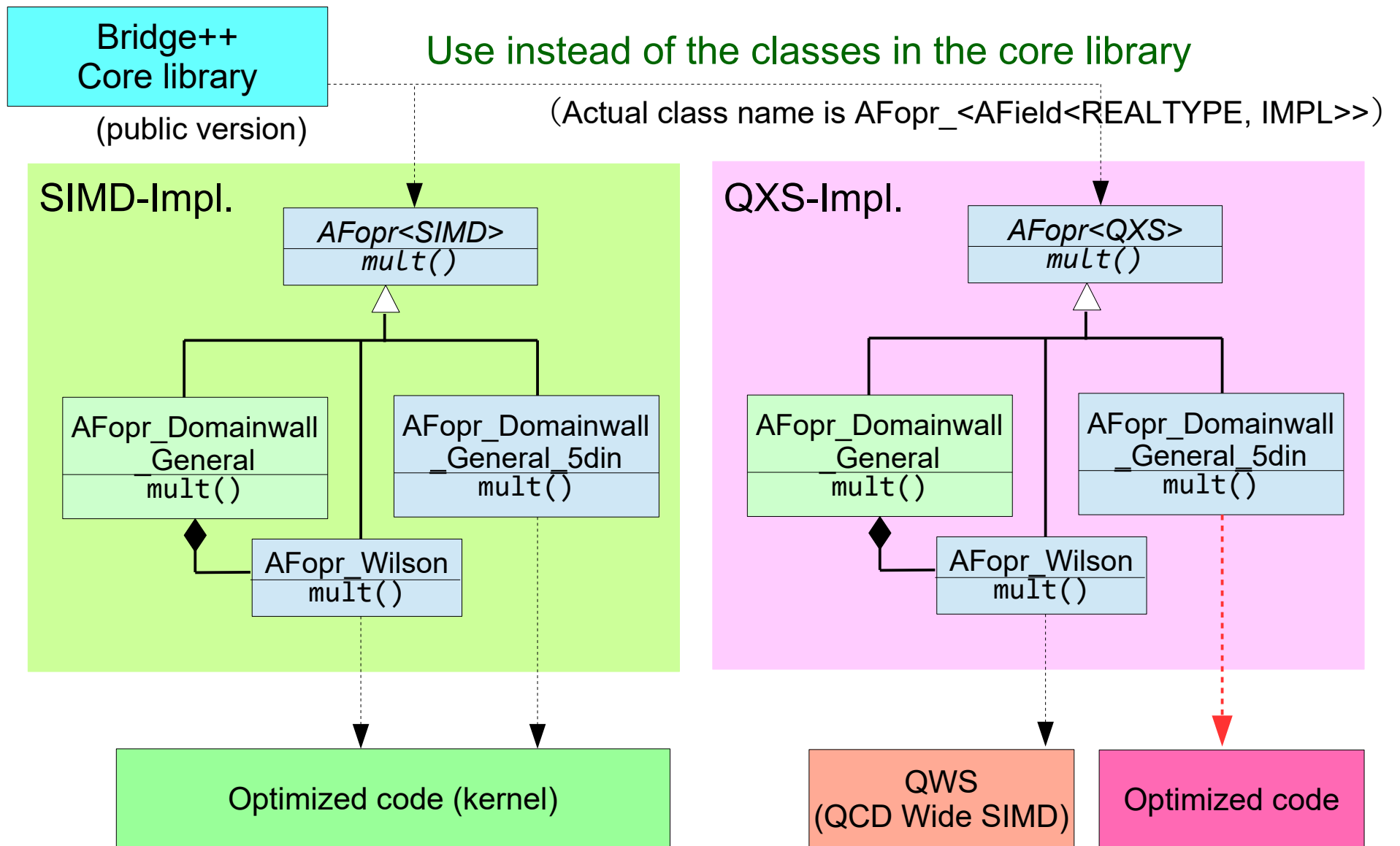
Extended Bridge++

Extended Bridge++ framework:

Core library + *extension (“alternative”)*

- **Bridge++ core library**
 - Original Bridge++ code (while still actively developed)
 - Used as a firm framework and general purpose tool set
 - Kept working on arbitrary system
- **Extension (“alternative” branches)**
 - Arbitrary data type and layout
 - Machine-specific implementation is easily incorporated
 - Exploit C++ template keeping the class structure of the core library
 - **To be publicly released VERY SOON as ver.2.0**
- **Class for field object: employing C++ template**
 - `Field` → `AField<REALTYPE, IMPL>` (IMPL is defined by enum)
 - e.g. `AField<double, SIMD>`

Extended Bridge++



Extended Bridge++

Development of extended Bridge code

- SIMD architecture (Intel AVX-512)
 - Optimization with AVX-512 intrinsics and manual prefetching
 - I.Kanamori and H.Matsufuru, LNCS 10962 (2018) 456-471
- QXS: A64FX (another SIMD) architecture
 - Optimized with ACLE intrinsics
 - Another data layout for SIMD
 - Making use of QWS (QCD Wide SIMD) library developed in Co-design
- Accelerators (GPU)
 - Currently using OpenACC
 - Cf. OpenCL and OpenACC
 - S.Motoki et al., Proc. Comp. Sci. 29 (2014) 1701
 - H.Matsufuru et al., Proc. Comp. Sci. 51 (2015) 1313
 - Cf. Pezy-SC with OpenCL
 - T. Aoyama et al. Proc. Comp. Sci. 80 (2016) 1418
- Vector processor (NEC SX-Aurora)

Status

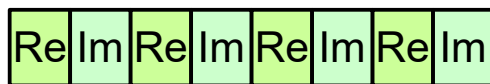
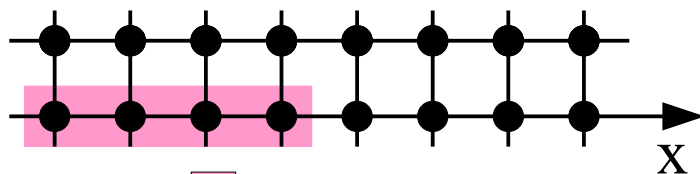
- Fermion operators for each architecture
 - Major fermion operators are implemented
 - QWS (QCD Wide SIMD) library for A64FX can be called from QXS branch
 - Developed in FS2020 Co-Design (mainly by Y. Nakamura)
 - Solver/eigensolver are incorporated with C++ template

	Accel	Vector	SIMD	QXS	QWS
Wilson (lex)	○	○	○	○	
(eo)	○		○	○	◎
Clover (lex)	○		○	○	
(eo)	○		○	○	◎
Domainwall(5din) (lex)			○	○	
(eo)				○	
Staggered (lex)	○	○		○	
(eo)	○			○	

Data layout

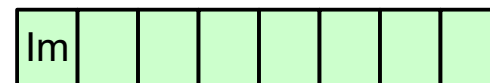
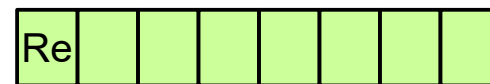
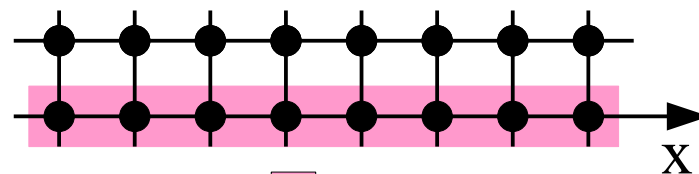
- Length of SIMD vector: 512 bit
 - VLEND=8 (double), VLENS=16 (float)
 - Restriction of local lattice size in x-direction
- SIMD implementation
 - VLEN/2 sites in x-direction are packed in a SIMD vector
- QWS/QXS implementation
 - QWS: VLEN sites in x-direction are packed in a SIMD vector
 - QXS: enabling 2D tiling in x-y plane

SIMD2 (lexical site ordering)



(double precision)

QWS (lexical site ordering)



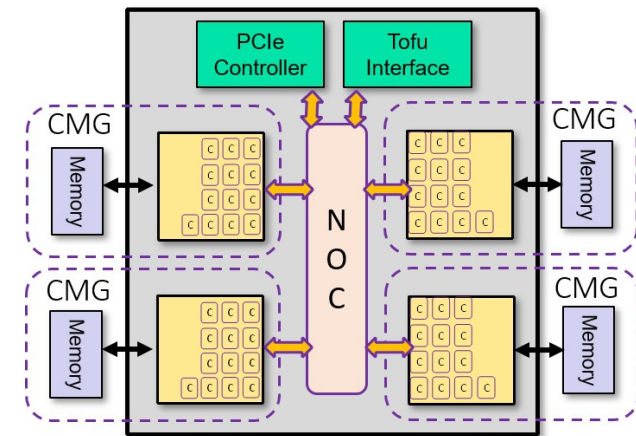
Convention

- Conversion is needed before and after callin QWS
 - Implemented in classes of QWS-impl.
 - Multiply γ_4 to spinor
 - Order of color/spinor/complex indices is changed

	QWS	Bridge++
VLEN	site(x)	complex * site(x)
spinor (even-odd)	v[Neo][Nt][Nz][Ny][Nxd] [Nc][Nd][Nri][VLEN]	v[Neo][Nt][Nz][Ny][Nxd] [Nd][Nc][VLEN]
Gauge (even-odd) U_{ab}	g[Neo][Nt][Nz][Ny][Nxd] [Ncb][Nca][Nri][VLEN]	g[Neo][Nt][Nz][Ny][Nxd] [Nca][Ncb][VLEN]
Nxd	Nx/VLEN	Nx/VLEN2 (VLEN2=VLEN/2)
gamma matrix (Dirac)	$\gamma_i^{(QWS)} = -\gamma_i^{(Bridge)}$, $\gamma_4^{(QWS)} = \gamma_4^{(Bridge)}$, $\gamma_5^{(QWS)} = \gamma_5^{(Bridge)}$	
Lattice size	Macro (#define)	Given at run-time

Fugaku: A64FX architecture

- ARM based processor by Fujitsu with Scalable Vector Extension
 - Architecture: Armv8.2-A SVE 512bit
 - Total peak performance 488 PFlops (2.0 GHz)
 - 158, 976 nodes
 - 4 CMGs (NUMA nodes), 48 cores for compute
 - **3.072 Tflops/node (DP), x2 (SP) in normal mode**
 - Memory: HBM2 32 GiB, 1024 GB/s
 - Tofu Interconnect D (28 Gbps x 2 lane x 10 port)

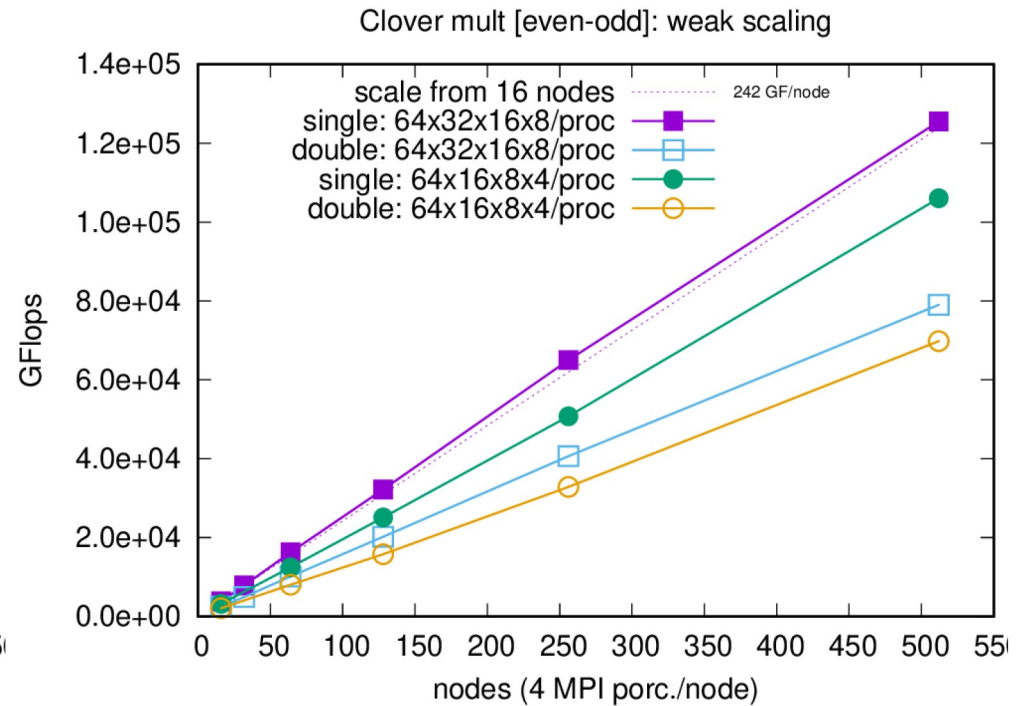
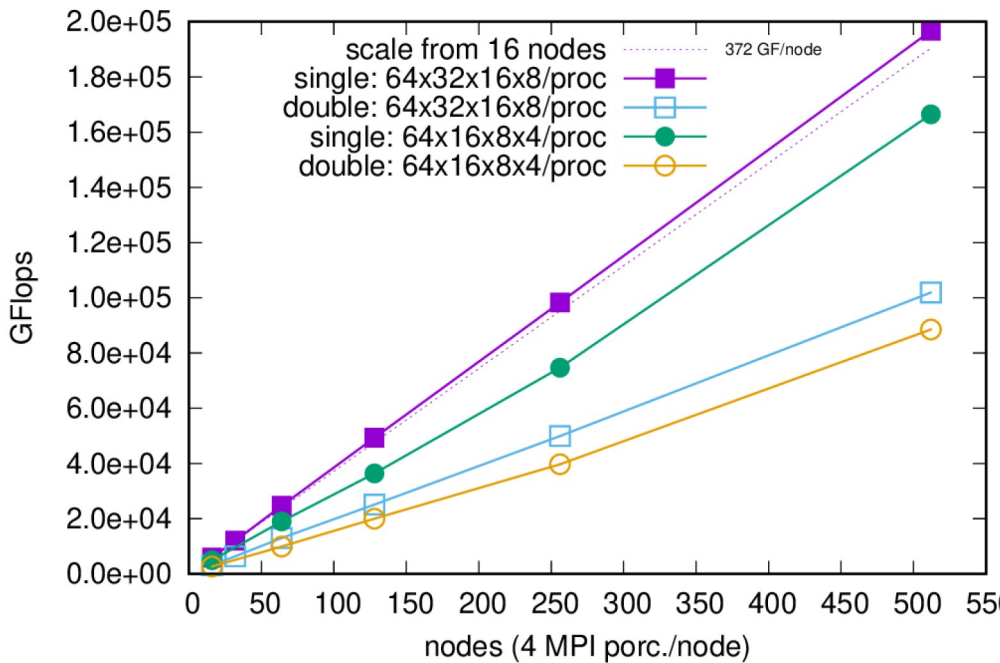


RIKEN CCS
<https://www.r-ccs.riken.jp/en/fugaku/>



Performance on Fugaku

- Performance on Fugaku (Poster by I. Kanamori in Lattice 2022)
 - Clover fermion matrix multiplication (Single precision)
 - Weak scaling plot

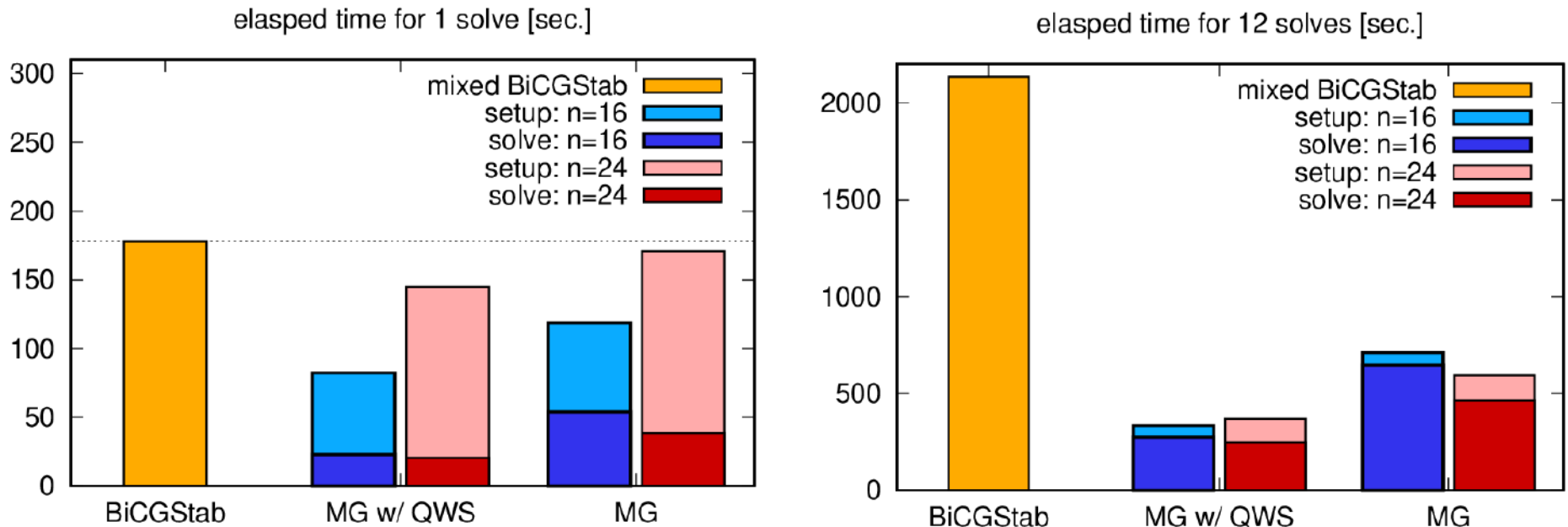


Performance on Fugaku

- Multi-grid solver for clover fermion

K-I. Ishikawa et al., PoS LATTICE2021(2022) 278

- Acceleration by coarse grained solver
- Smoother: multiplicative SAP, single prec.: from QWS



96^4 lattice, $M_\pi = 145$ MeV configuration [PACS] on 216 nodes

cf. LDDHMC (reimplementation of QWS for HMC) : 52 sec/solve

Summary

- Congratulations for Onogi-san's 60th birthday!
- Lattice QCD requires High Performance Computing
- Code should be optimized for new architecture
- Bridge++ ver.2.0 will be released soon
 - Code branch for Fugaku is included

