QCD Wide SIMD Library (QWS) for Fugaku

Yoshifumi Nakamura
RIKEN, R-CCS
Fugaku QCD coding workshop
12/12/2019

Disclaimer

The software used for the evaluation, such as the compiler, is still under development and its performance, which is obtained by "performance estimation tool" and even actual execution on a prototype machine, may be different when the supercomputer Fugaku starts its operation.

QCD Wide SIMD Library (QWS)

- **Is the lattice QCD simulation program library for Wide SIMD width**
- **Is written in C, C++ (mostly C)**
- **Has been developed by Y.N. since 2014, starting for a benchmark program for "Post-K" supercomputer processor in Flagship 2020 project, FS2020 project**
- **Now optimized for Fugaku (Post-K)**
  - Clover Wison Dirac operator
  - Even-odd preconditioned Dirac matrix
  - SAP (Schwarz Alternating Procedure) domain decomposition for full Dirac matrix
  - Double, float, half precision
  - Conjugate gradient (CG), shifted CG, BiCGstab
- **Download and copying**
  - QWS will be free software under a BSD-like License
  - Will appear at https://github.com/RIKEN-LQCD

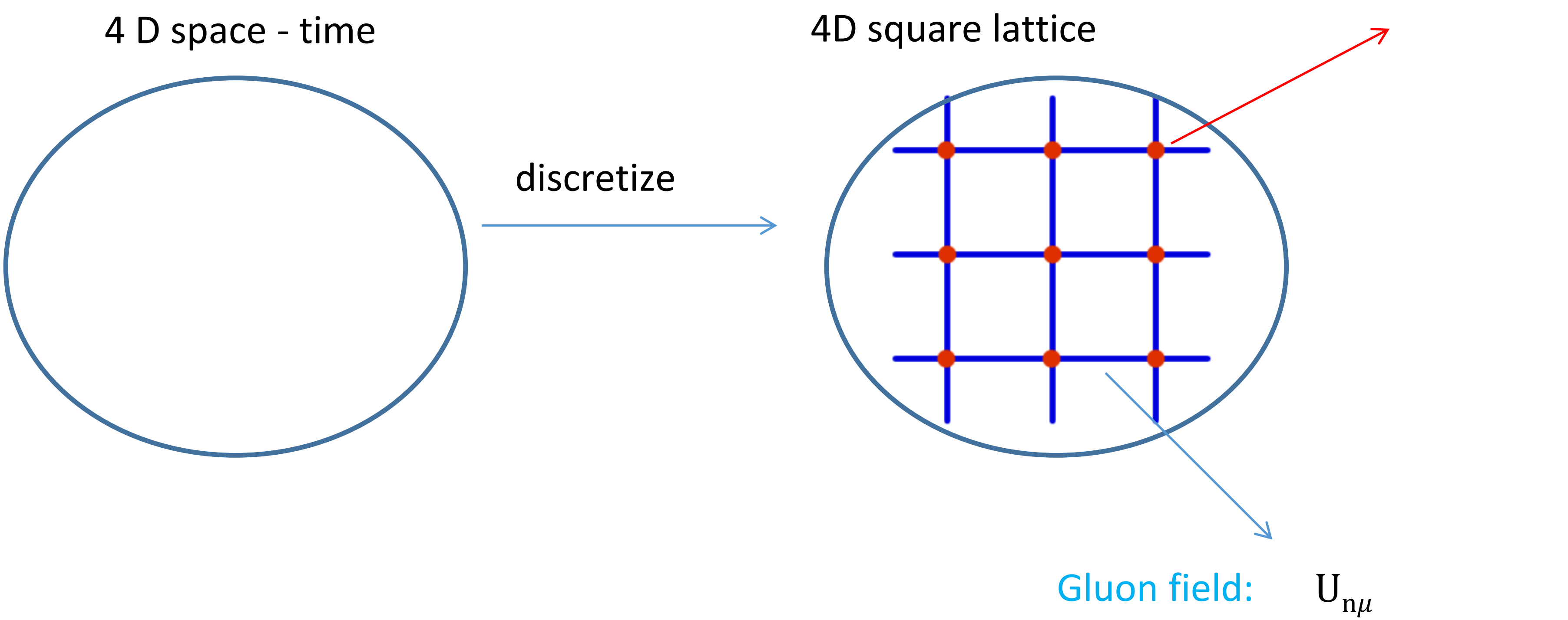
Contents

- Brief introduction to Lattice QCD simulation
- LQCD working group history in FS2020
- LQCD (QWS) tuning for Fugaku
- LQCD (QWS) status on Fugaku
- LQCD working group plan in FS2020
- Summary

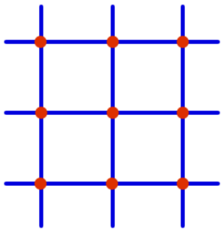
Brief introduction to Lattice QCD simulation

Lattice QCD

QCD on the lattice



Lattice QCD action



Field

- quark field: color × spinor / site → 12 complex numbers
- gluon field: SU(3) matrix / link → 9 complex numbers

Lagrangian

$$L = \sum_i \bar{\psi}_i D(m_i) \psi_i - \frac{1}{4} F^a_{\mu\nu} F^{a\mu\nu} \,, \quad F^a_{\mu\nu} \;:\; \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + i g f^{abc} A^b_\mu A^c_\nu$$

$$D(m_i) \;:\; \gamma_\mu (i\partial_\mu + g A^a_\mu T^a) - m_i$$

Dirac operator

Input parameters：
1 coupling constant, 6 quark masses

Similar Lagrangian as QED

Lattice QCD fermion action

Grassmann integral

fermion field(Grassmann number) $\rightarrow$ pseudo-fermion field(usual number)

$$\int d\bar{\psi}_i d\psi_i \quad \exp(-\sum_{i=1}^{2} \bar{\psi}_i D(m_i)\psi_i) = \det D(m_1) \det D(m_2)$$

determinant

when $m = m_1 = m_2$

$$\det D(m)^2 = \int d\phi^\dagger d\phi \quad \exp(-\phi^\dagger \left[D^\dagger(m) D(m)\right]^{-1} \phi)$$

Numerical cost to calculate determinant
Is similar as all eigen value calculation (～N^3)

Change to inversion problem by auxiliary field

Dominant part of QCD simulation is solving this inversion

Lattice QCD kernel

$$Ax = b$$

- Krylov subspace method (iterative method)
- 4-dimensional square lattice

  $$V = L_x \times L_y \times L_z \times L_t$$
- parallelization
  - Domain decomposition : sub lattice / MPI rank
  - Mainly nearest neighbor interaction (communication)

Weak scaling is very good
Strong scaling is problematic

Dirac operator multiplication (matrix vector multiplication)

$$A\phi = \sum_{x=1}^{L_x \times L_y \times L_z \times L_t} (\phi_x - \kappa\eta_x)\,, \qquad \eta_x = \sum_{\mu=1}^{4} \left[(1-\gamma_\mu)U_{x,\hat\mu}\phi_{x+\hat\mu} + (1+\gamma_\mu)U^\dagger_{x-\mu,\hat\mu}\phi_{x-\hat\mu}\right]$$

Algorithm parameter for comp. and comm. per node

$1320 \times l^4$ Flops : 480MADDS, 96 MULS, 264 ADDS, FMA theoretical best performance is $\sim 78\%$

$192 \times l^3$ words : node-node communication (send/recv of $\pm$xyzt for spin projected η field)

Depends on fermionic action
(this is simplest example)

Overlapping communication time

Network bandwidth $B \sim \frac{0.15}{l} \frac{words}{flops} F$

Once we decide action, know ideal machine

F: performance / node, l: lattice size / node (for 1 direction)

Supercomputers for LQCD

name	Develop / appear	Peak performance
Columbia(USA)	1985–1989	0.25–16 GFlops
GF11(USA)	1983–1992	11 GFlops
QCDPAX(Tsukuba)	1989	14 GFlops
APE(Italy)	1986–1988	0.25–1 GFlops
ACPMAPS(USA)	1991–1993	5–50 GFlops
APE100(Italy)	1994	100 GFlops
CP-PACS(Tsukuba)	1996	614 GFlops
QCDSP(USA)	1995–1998	600 GFlops
APEMille(Italy)	2000	1 TFlops
APENEXT(Italy)	2006	12 TFlops
QCDOC(USA)	2005	10 TFlops
PACS-CS(Tsukuba)	2006	14 TFLops
QPACE(EU)	2009	200 TFlops
QPACE2(Germany)	2015	310 Tflops
QPACE3(Germany)	2016-2017	1.7 Pflops
QPACE4(Germany)	??	?? Pflops

LQCD working group history in FS2020

History of LQCD in FS2020 (2014FY-1)

- **2014/10**
  - Fujitsu and RIKEN began basic design work
  - 1st Co-design meeting
  - LQCD preceded others for performance estimation
- **2014/11**
  - Performance estimation by roofline model
    - CG, mult-shifted CG, and BiCGStab with Wilson & clover fermion
  - starting QWS
  - 1st LQCD (ALP9) (sub-)working group

History of LQCD in FS2020 (2014FY-2)

- **2014/12**
  - Optimization of QWS, CCS-QCD
  - Discussing problem size / process
  - Consideration of global reduction
- **2015/01**
  - Consideration of Memory (for new memory configuration)
- **2015/02**
  - Consideration of core memory group (CMG)
- **2015/03**
  - Started consideration OS jitter
  - Started communication performance estimation by using LDDHMC and K

History of LQCD in FS2020 (2015FY-1)

- **2015/04**
  - Measuring baseline time for target problem size on K to estimate post-K's performance speedup over K
  - Domain wall fermion and nuclear force calculation
- **2015/05**
  - Estimation energy-efficiency (flops/watt)
  - MPI process rank mapping
- **2015/06**
  - File IO check, performance measurement on Haswall
- **2015/07**
  - Performance estimation and consideration for new system configurations

History of LQCD in FS2020 (2015FY-2)

- **2015/10**
  - Making code for performance simulator
- **2015/12**
  - Analysis for results by performance simulator
- **2016/02**
  - Performance estimation and consideration for new memory type

History of LQCD in FS2020 (2016FY-1)

- **2016/04**
  - testing with performance simulator
  - Considering (1 MPI proc)/4CMS, proc/node
- **2016/05**
  - Improving thread imbalance
  - Implement explicit prefetch
- **2016/06**
  - Testing loop fission
- **2016/07**
  - Removing integer register spill/fill
- **2016/08**
  - Expanding OMP Parallel region

History of LQCD in FS2020 (2016FY-2)

- **2016/09**
  - Explicit prefetch all regions
- **2016/11**
  - LDDHMC double buffering code
  - uTofu (low level communication library) sample code
- **2016/12**
  - Improving rank map search program
- **2017/01**
  - Testing some (local problem size) / process
- **2017/02**
  - Communication estimation
- **2017/03**
  - Tuning clover mult

History of LQCD in FS2020 (2017FY)

- **2017/04**
  - Tuning clover mult
- **2017/06**
  - Performance estimation with FP16
- **2017/11**
  - performance measurement on Skylake
- **2018/02**
  - Vector load tuning by Arm C Language Extensions (ACLE)

History of LQCD in FS2020 (2018FY)

- **2018/06**
  - Redefine performance estimation regions
  - Estimation SU(3) reconstruction performance
  - FP16 code by half-precision floating-point library
- **2018/09**
  - Testing FP16 on realistic lattices
- **2018/11**
  - Performance estimation and consideration for new system configuration
- **2018/12**
  - Double buffering test (2D Poisson's equation)
- **2019/02**
  - Results on prototype

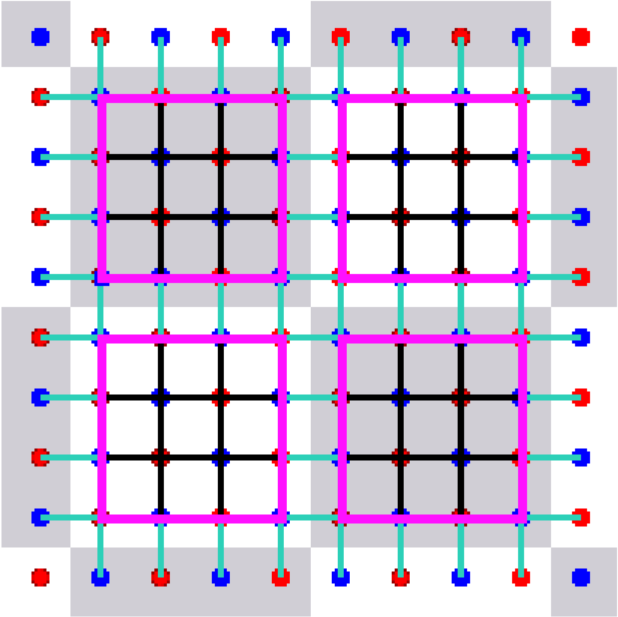
History of LQCD in FS2020 (2019FY)

- **2019/04**
  - Merging FP16 and double buffering to latest version
  - Fixing compiling and executing bugs for several environments and lattice and process setups
- **2019/10**
  - Testing FP16 on prototype
- **2019/11**
  - Testing uTofu + double buffering

LQCD (QWS) tuning for Fugaku

Optimization for quark solver

- **Target problem size is 192^4**
- **Single precision BiCGstab solver (Dx=b)**
  - Evaluation region in FS2020 project
  - Clover Wilson Dirac operator (D)
  - Schwarz Alternating Procedure (SAP) preconditioning
    - Jacobi inversion for inside domain Dirac operator



12 tuning regions in single prec. BiCGstab

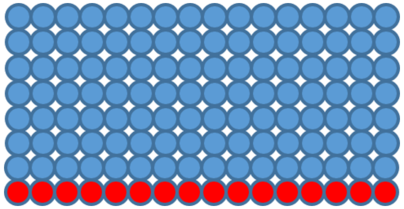
- **5 computation regions**
  - jinv_ddd_in_s (Jacobi inversion of inside domain D)
  - ddd_in_s (inside domain D mult)
  - ddd_out_pre_s (preprocess of boundary D mult)
  - ddd_out_pos_s (postprocess of boundary D mult)
  - other_calc (other calculations, e.g. axpy)
- **7 communication regions**
  - Irecv (starting receiving buffer)
  - Isend (starting sending buffer)
  - Recv_wait (waiting receiving buffer)
  - Send_wait (waiting sending buffer)
  - global reductions
    - 1 time 1 float
    - 1 time 2 float
    - 2 times 3 floats

Avoiding memory bandwidth

- **Putting all evaluation region on L2 by using full system**
  - Single precision BiCGstab with SAP on 192^4 requires 2TB < system L2 cache size
  - 150k+ nodes / system
  - L2 cache/node
    - 32 MB
    - 3.6+ TB/s
  - Memory/node
    - 32 GB
    - 1024 GB/s

Data layout for wide SIMD

- **Continuous access except for x-direction**
  - Fugaku(FP64):[nt][nz][ny][nx/2/8][3][4][2][8]
  - Fugaku(FP32):[nt][nz][ny][nx/2/16][3][4][2][16]
  - Fugaku(FP16):[nt][nz][ny][nx/2/32][3][4][2][32]
  - cf. K:[nt][nz][ny][nx][3][4][2]

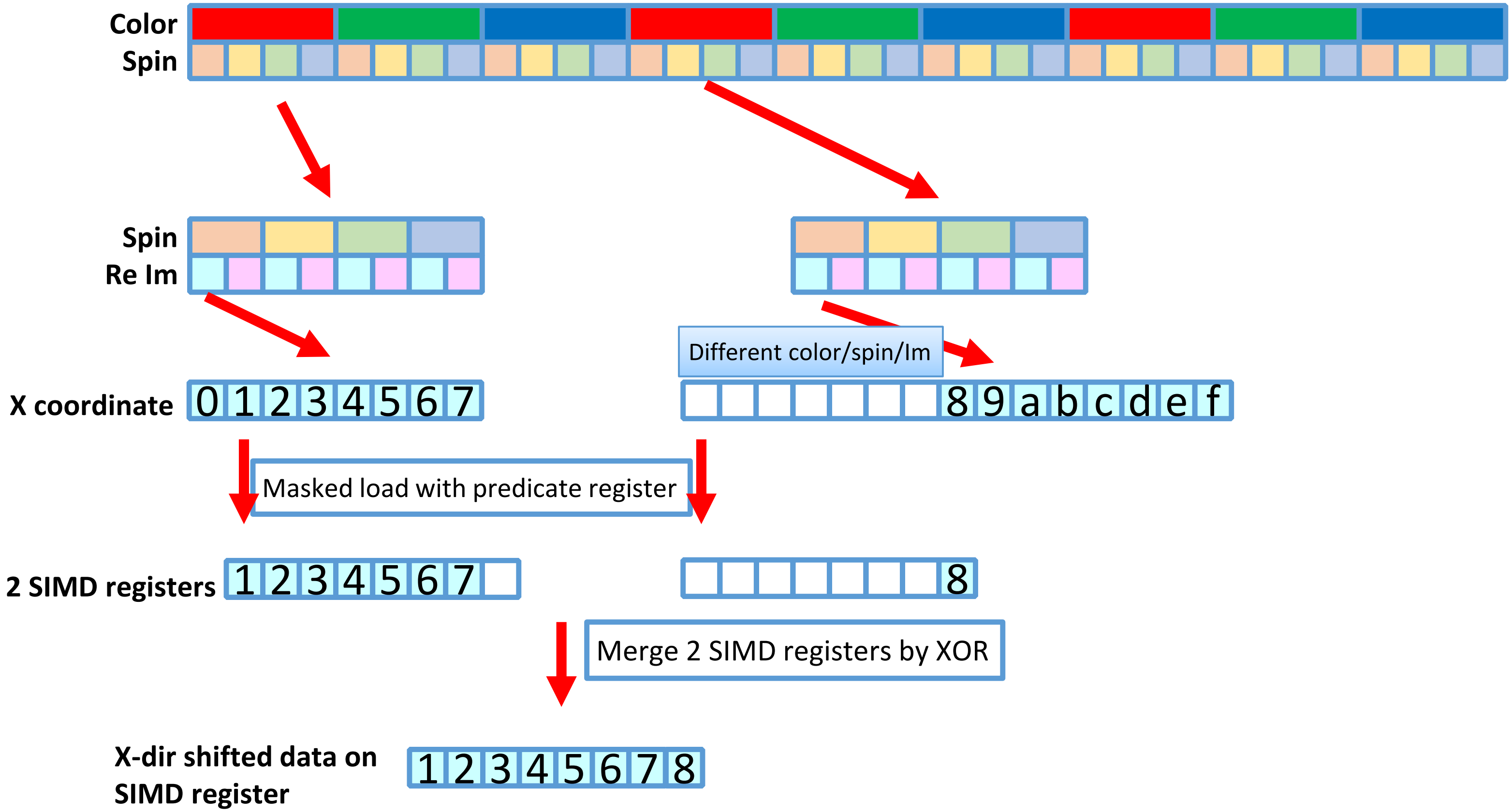


- **Separating real and imaginary part**
  - rrrrrrrriiiiiiii is better throughput than riririririririri
  - Complex number addition and multiplication (svcadd / svcmla) with rotation of 0, 90, 180, 270 degree are supported for FP64, FP32

Tuning by ACLE

Input spinor :[nt][nz][ny][nx/2/8][3][4][2][8] (double precision case for example)

X-direction shift for x-direction hopping term multiplication



Other tunings

- **OMP**
  - Parallel region expansion : Making omp parallel region is costly, must put “omp parallel” on higher level caller routines
  - Load balancing of threads after loop fission (obsoleted, simple multiple loop is faster in latest version)
- **Prefetching**
  - Explicitly, every 256 B for all arrays by building_prefetch()
  - No speed up by Gather prefetch of SVE
- **Instruction-level scheduling**
  - Clover mult
- **Process mapping search program**
  - 4D QCD processes to TofuD
  - Calculate stream for all possible rank maps and find the best process mapping
  - Tofu Network Interface (TNI) load balancing
  - Minimizing link stream
- **Removing temporal arrays**
- **Removing unnecessary horizontal addition in multi loop (will be fixed by new compiler)**
- **FP16 (on going)**
- **Double buffering (on going)**

LQCD (QWS) status on Fugaku

Performance estimation method

- **Single precision BiCGstab with SAP precondition using Jacobi iteration for inside domain Dirac matrix**
- **Estimating performance for 1 BiCGstab iteration by executing 500 iterations for each comp/comm region separately**
- **Comm/comp overlapping time is subtracted from total time**
- **Estimation for computation regions**
  - On 2 processes of 12 threads/process
    - 1 process/CMG on Fugaku prototype
    - 2 process/node on FX100
  - Estimation from FX100 to Fugaku : Performance-power estimation tool using Performance Analysis by Fujitsu's Profiler on FX100
- **Estimation for communication regions**
  - Allreduce : fast reduction mechanism up to 3 words by Tofu hardware
  - Neighboring communications
    - Irecv (starting receiving buffer) : estimated to be 0s by double buffering
    - Isend (starting sending buffer) : same as observed time on K (expected to be improved by uTofu)
    - Recv_wait (waiting receiving buffer) : link stream /(0.9*network bandwidth)
    - Send_wait (waiting sending buffer) : same as observed time on K (expected to be improved by uTofu)

Compile

- Fugaku prototype
  - lang/fjcompiler20190731_04
    - Option for kernel codes, clover_s.cc, ddd_in_s.cc, ddd_out_s.cc

  -Kfast,restp=all,optmsg=2,ocl,preex,noprefetch,noswp

  -Knosch_pre_ra,nosch_post_ra -Knoeval

    - Option for others

  -Kfast,restp=all,optmsg=2,ocl,preex,noprefetch,noswp

Baseline on K

		elapse time [ms]	efficiency [%]
compotation total		27.99	34.8
	jinv_ddd_in_s	14.09	41.9
	ddd_in_s	6.52	44.3
	ddd_out_pre_s	0.95	12.6
	ddd_out_pos_s	3.84	16.9
	other_calc	2.56	7.1
cummunication total		2.66	
	irecv	0.45	
	isend	0.17	
	recv_wait	0.16	
	send_wait	0.17	
	reduc1 (2 calls)	0.18	
	reduc2 (2 calls)	0.85	
	reduc3 (1 call)	0.67	
total		30.65	31.8

192x192x192x192, 8x8x8x32/node(proc), 8 threads/node(proc)

Estimated communication time on Fugaku

		elapse time [ms]	
		32x6x6x2 (for tool)	32x6x4x3 (for prototype)
	irecv	0.00	0.00
	isend	0.13	0.13
	recv_wait	0.36	0.24
	send_wait	0.11	0.11
	reduc1 (1 call)	0.02	0.02
	reduc2 (1 call)	0.02	0.02
	reduc3 (2 calls)	0.04	0.04
cummunication total		0.68	0.56

192x192x192x192, 4 processes/node, 1 proc/CMG, 12 threads/process
On 147456 nodes (< 150K+ full system nodes)

Estimated performance by tool on Fugaku

32x6x6x2 by tool		elapse time [ms]	efficiency [%]
	jinv_ddd_in_s	0.47	21.5
	ddd_in_s	0.20	24.0
	ddd_out_pre_s	0.06	7.5
	ddd_out_pos_s	0.15	15.0
	other_calc	0.08	5.4
compotation total		0.96	18.8
communication total		0.68	
overlapped		0.20	
total		1.44	12.5

192x192x192x192, 4 processes/node, 1 proc/CMG, 12 threads/process
On 147456 nodes (< 150K+ full system nodes)

Performance-power estimation tool using performance analysis information on FX100
Linear corrections for individual times (commit times, L1/L2/mem/ wait times, and so on)
by SIMD, #cores, frequency, latencies,……., rations between K and Fugaku CPU

Performance estimation using prototype

method	mixed			prototype		
size/proc	32x6x6x2			32x6x4x3		
region	elaps[ms]	perf[%]	method	elaps[ms]	perf[%]	method
jinv_ddd_is_s	0.30		prototype			
ddd_in_s	0.13		prototype			
ddd_out_pre_s	0.06	7.5	tool			
ddd_out_pos_s	0.15	15.0	tool			
other_calc	0.08	5.4	tool			
all_calc				0.42		prototype
overlapped				0.13		prototype
computation	0.72		mixed	0.56		prototype
communication	0.68			0.56		
overlapped	0.13		prototype	0.13		prototype
total	1.27			0.99		

192x192x192x192,   4 processes/node, 1 proc/CMG,   12 threads/process
On 147456 nodes (< 150K+ full system nodes)
Peak performance ration (perf[%]) on prototype has not been confirmed yet

LQCD working group plan

- **2019FY**
  - Performance measurements
    - With FP16
    - With uTofu + double buffering
- **2020FY**
  - Performance confirmation on massively parallel runs

Summary

- **Briefly introduced Lattice QCD simulation**
- **History and plan of LQCD working grope in FS2020**
- **Tuning and status of LQCD (QWS) on Fugaku**

25x+ speedup over K

- **Missing part (out of evaluation region in FS2020)**
  - Force part
  - Gauge part
  - Measurements