Algorithmic and Machine issues on the N_f =2+1 lattice QCD project by the PACS-CS collaboration

Ken-Ichi Ishikawa (Hiroshima Univ.)

For PACS-CS collaboration

Member: S. Aoki, N.Ishii, T.Ishikawa, N.Ishizuka, D.Kadoh, K.Kanaya, Y.Kuramashi, N.Tsutsui, M.Okawa, Y.Taniguchi, A.Ukawa, N.Ukita, T.Yoshie

1.Introduction

- June 27 2006 : PACS-CS system has been installed at *Center for Computational Sciences University of Tsukuba*
  - Project Leader: Akira Ukawa
  - System Development: T. Boku, M. Sato, D. Takahashi, O. Tatebe
  - For Computer, Materials, Life Science, Particle physics, Astrophysics, Biology …

- PACS-CS collaboration

  Physics plan

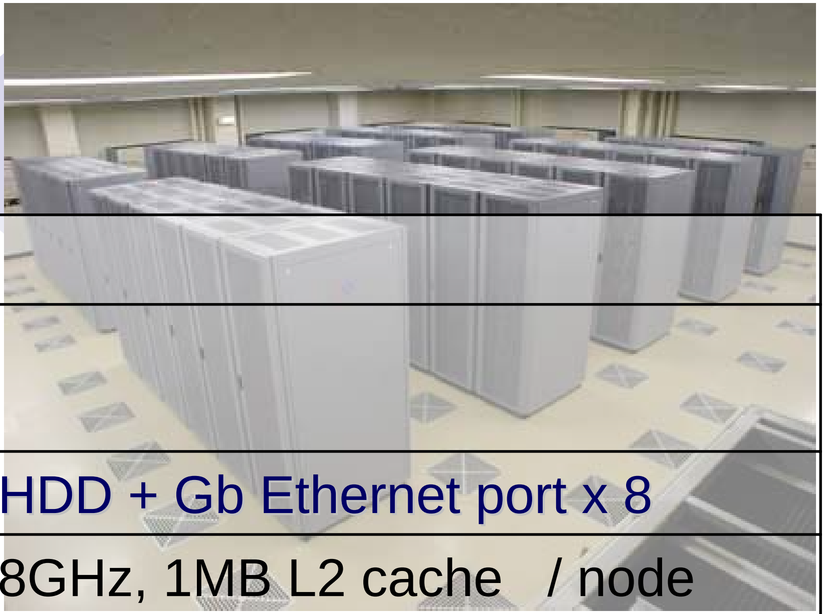
  - NF=2+1 Lattice QCD project

    (Talk by Kuramashi)

1.Introduction (Cont'd)

- In this talk
  - **PACS-CS** system
    - Machine specification
    - Network, CPU's, …
  - Algorithm for Nf=2+1 lattice QCD project
    - Lüscher's domain decomposition HMC (Nf=2)
      + UV-filtered PHMC (Nf=1) algorithm

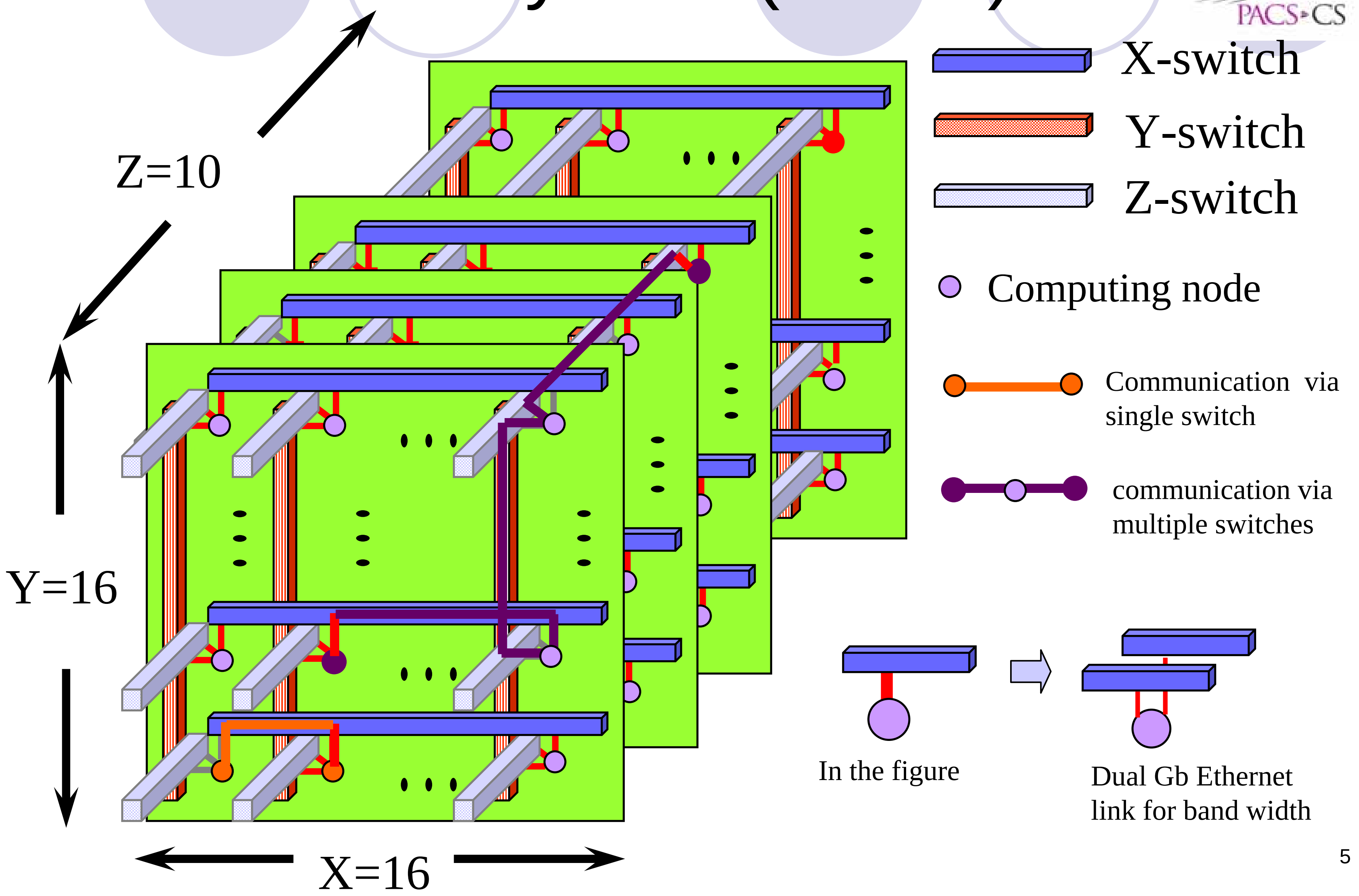
2.PACS-CS system



#nodes	2560 (16x16x10)
peak performance	14.3Tflops
node	single CPU + memory + HDD + Gb Ethernet port x 8
CPU	Intel LV Xeon EM64T, 2.8GHz, 1MB L2 cache / node
memory	2GB/node (5.12TB/system)
network	3dimensional hyper-crossbar uses dual Gb Ethernet/link
network performance	250MB/s/direction 750MB/s/node (3dim. simultaneous send/receive)
local HDD	160GBx2 (RAID-1) (410TBx2/system)
# racks	59 racks
footprint	100㎡
power	545 kW

PACS-CS

2.PACS-CS system (cont'd)

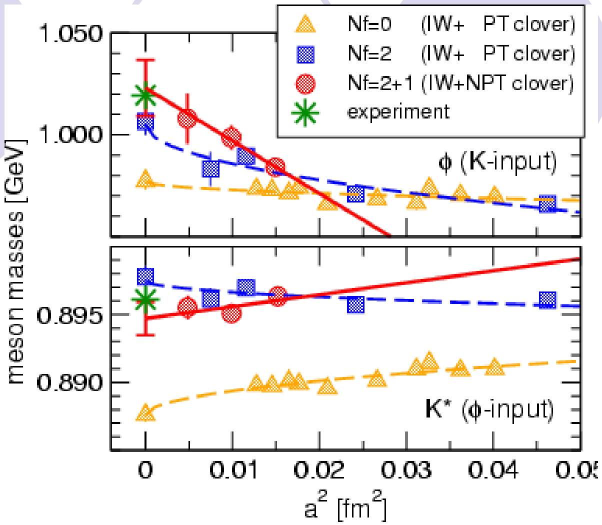


2.PACS-CS system (cont'd)

- Software
  - OS
    - Linux (64 bit mode, EM64T)
    - SCore (cluster middleware developed by PC Cluster Consortium http://www.pccluster.org/index.html.en)
    - 3D HXB driver based on SCore PMv2 driver
  - Programming
    - MPI for communication (shipped with Score) PMv2
    - Library for 3D HXB network
    - Intel Fortran, C, C++
  - Job execution
    - System partition (256nodes, 512nodes, 1024nodes, ...)
    - Batch queue using PBS
    - Job scripts for file I/O

3. Nf=2+1 Project

- Using PACS-CS We further promote Nf=2+1 lattice QCD study of JLQCD/CP-PACS collab.



JLQCD CP-PACS joint collaboration Nf=2+1 meson spectrum (Lattice 2006)

  - Deep Chiral limit

$$M_{\pi} < 300\ \text{MeV} \qquad m_{ud} \approx 10-20\ \text{MeV}$$

  - Large Physical Volume (Baryon)

$$L > 3-4\ \text{fm}$$

PACS-CS Network system is not so strong. => Efficient algorithm is required.

3. Nf=2+1 Project (cont'd)

- Dynamical Lattice QCD simulation with small quark masses.
  - Action
    - Iwasaki-RG gauge + O(a)-improved Wilson quarks
  - Hybrid Monte Carlo (HMC) Algorithm
    - Recent Improvement
      - Preconditioning Technique / IR-UV separation

        [de Forcrand, Takaishi, NPB(Proc.Suppl.)53,Lat96]
      - Multi-time scale Molecular Dynamics (MD) Integrator

        [Sexton-Weingarten, NPB 380(92)]

    - Hasenbusch's preconditioner, Lüscher's SAP preconditioner, etc.

3. Nf=2+1 Project (cont'd)

- In this work, we employ
  - Lüscher's domain decomposition preconditioned HMC algorithm for Nf=2 part (LDDHMC) [Lüscher , JHEP 0305 '03,CPC 165 '05]
  - UV-filter preconditioned Polynomial HMC algorithm for Nf=1 part (UVPHMC)
    - Lüscher's algorithm = semi local HMC algorithm.

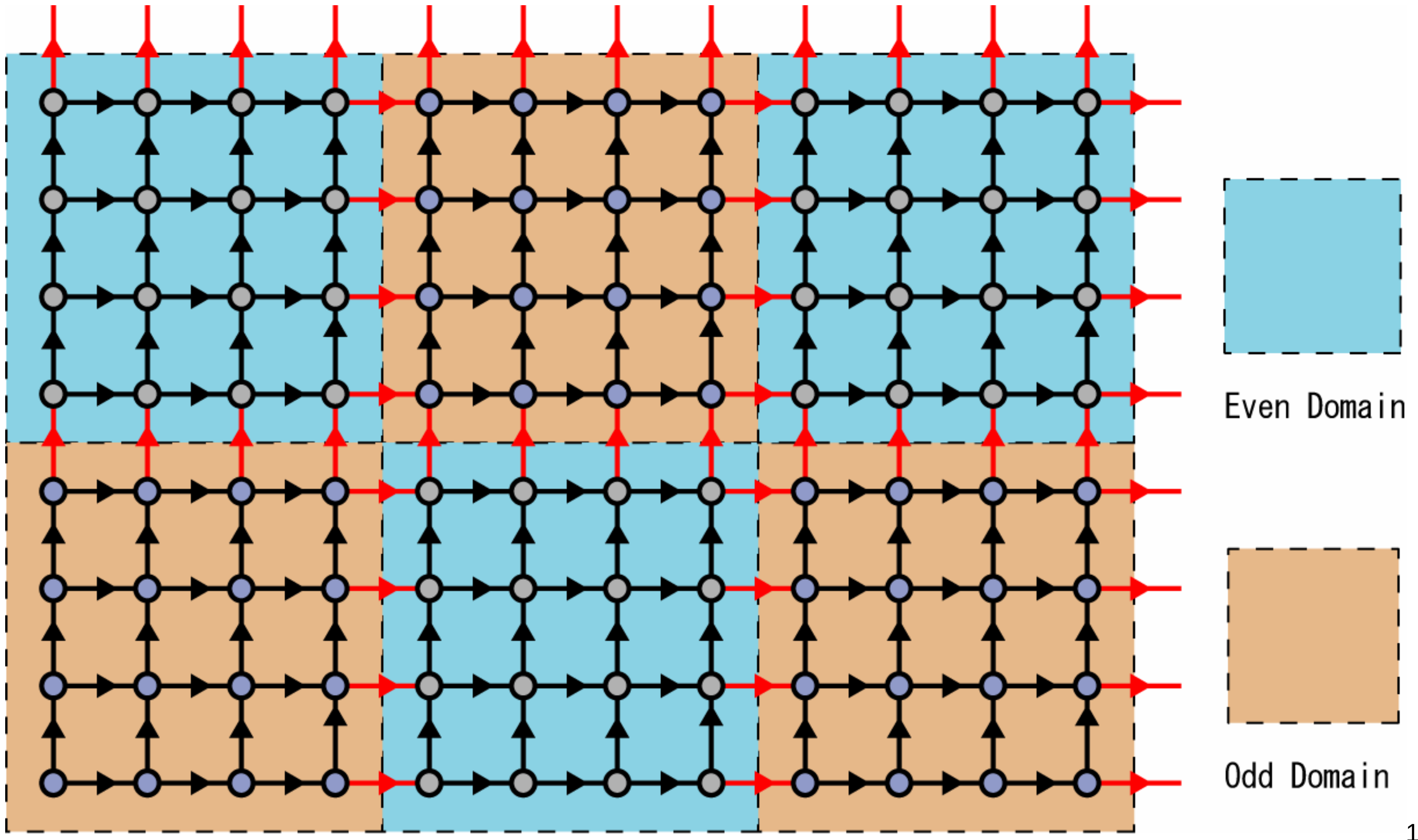
      = dead links/alive links => Communication cost is reduced.

      Well Fit with PACS-CS system feature
    - Autocorrelation tends to become longer.

4. Algorithm

$$Z = \int DU \det[D[U]]^{N_f} e^{-S_G[U]}$$

- Nf=2 part (LDDHMC)
  - Two domains



4.Algorithm (cont'd)

- Preconditioning D : O(a)-improved Wilson-Dirac op.

$$D = 1 + (1+T)^{-1} M, \quad T : \text{clover term}, \, M : \text{hopping matrix}$$

$$D = \begin{pmatrix} D_{ee} & D_{eo} \\ D_{oe} & D_{oo} \end{pmatrix} = \begin{pmatrix} D_{ee} & 0 \\ 0 & D_{oo} \end{pmatrix} \begin{pmatrix} 1 & D_{ee}^{-1} D_{eo} \\ D_{oo}^{-1} D_{oe} & 1 \end{pmatrix}$$

D_{ee} : Even => Even domain

D_{eo} : Odd => Even domain

D_{oo} : Odd => Odd domain

D_{oe} : Even => Odd domain

$$\det[D] = \det \begin{pmatrix} D_{ee} & 0 \\ 0 & D_{oo} \end{pmatrix} \det \begin{pmatrix} 1 & D_{ee}^{-1} D_{eo} \\ D_{oo}^{-1} D_{oe} & 1 \end{pmatrix}$$

$$= \det[D_{ee}] \det[D_{oo}] \det[1 - D_{ee}^{-1} D_{eo} D_{oo}^{-1} D_{oe}]$$

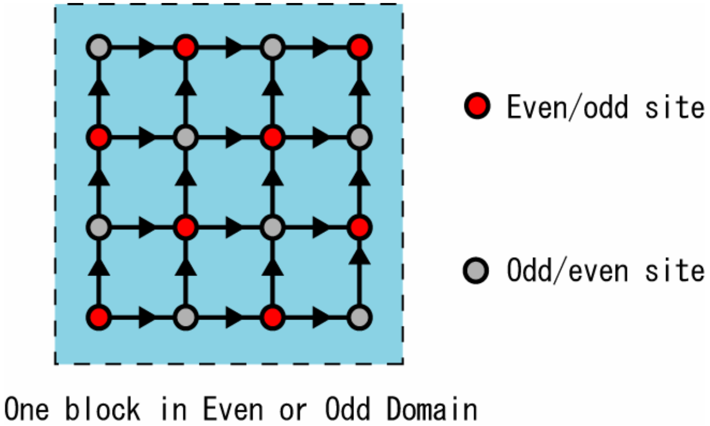
$$= \det[D_{ee}] \det[D_{oo}] \det[\hat{D}_{ee}]$$

$D_{ee}, D_{oo}, \hat{D}_{ee}$ have smaller condition number

4.Algorithm (cont'd)

- $D_{ee}, D_{oo}, \hat{D}_{ee}$ can be further preconditioned.
  - If the block lattice size is an even number, we can apply even-odd site preconditioning on D_{ee}, D_{oo} .

$$\det[D_{ee}] \Rightarrow \det[D'_{ee}]$$



  - $\hat{D}_{ee}$ is also preconditioned by using its spin structure.

$$\det[\hat{D}_{ee}] \Rightarrow \det[\hat{D}'_{ee}]$$

  - Preconditioned determinant

$$\det[D] = \det[D'_{ee}]\det[D'_{oo}]\det[\hat{D}'_{ee}]$$

- Preconditioned HMC partition function

$$Z = \int DUDP \,|\det[D[U]]|^2 \, e^{-Tr[P^2]/2 - S_G[U] + 2Tr[Log[1+T]]}$$

$$= \int DUDP \,|\det[D'_{ee}]|^2 |\det[D'_{oo}]|^2 |\det[\hat{D}'_{ee}]|^2 \, e^{-Tr[P^2]/2 - S_G[U] + 2Tr[Log[1+T]]}$$

$$= \int DUDPD\phi_e D\phi_o D\chi_e e^{-Tr[P^2]/2 - S_G[U] + 2Tr[Log[1+T]] - |D'^{-1}_{ee}\phi_e|^2 - |D'^{-1}_{oo}\phi_o|^2 - |\hat{D}'^{-1}_{ee}\chi_e|^2}$$

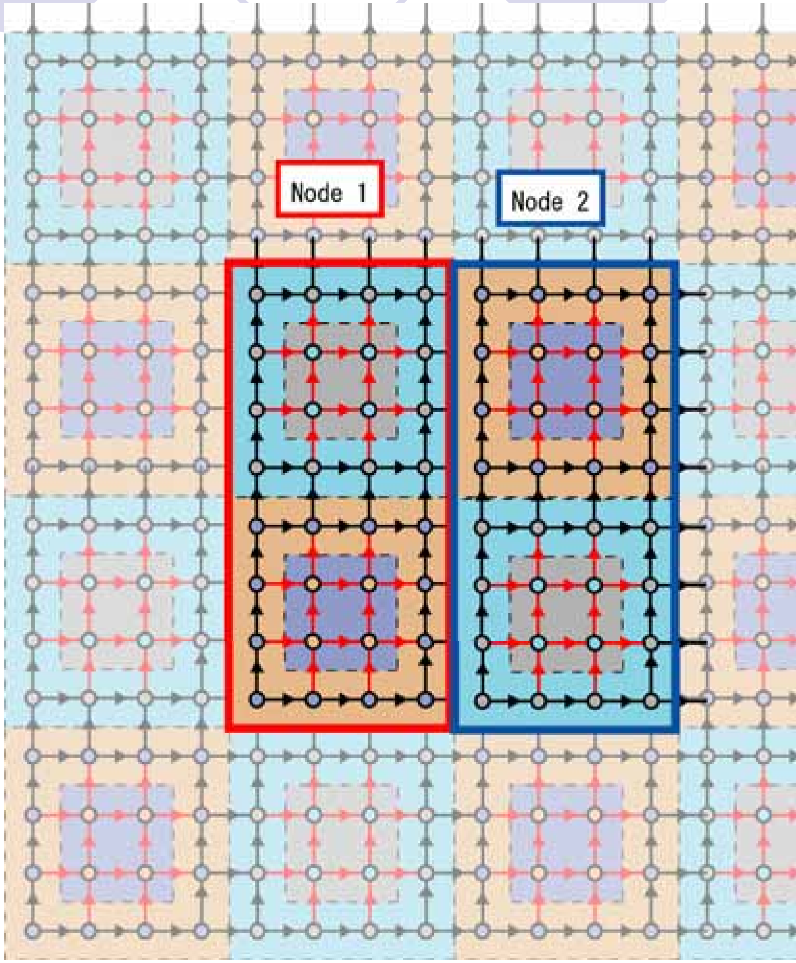
ϕ_e : Even-domain, even-site pseudo-fermion

ϕ_o : Odd-domain, even-site pseudo-fermion

χ_e : Even-domain, edge-site, spin-projected pseudo-fermion

4.Algorithm (cont'd)

- Parallelization
  - A single node contains both even and odd blocks.
  - With this node partitioning, operation of D_{eo} (D_{oe}) requires communication.
  - No communication for D_{ee} (D_{oo}).



$$|\hat{D}'_{ee}{}^{-1}\chi_e|^2$$

Global problem, but Well preconditioned.

$$|D'_{ee}{}^{-1}\phi_e|^2, \quad |D'_{oo}{}^{-1}\phi_o|^2$$

Semi Local problem

UV-IR Separation

4.Algorithm (cont'd)

- HMC

$$Z = \int DUDPD\phi_e D\phi_o D\chi_e e^{-Tr[P^2]/2 - S_G[U] + 2Tr[Log[1+T]] - |D'^{-1}_{ee}\phi_e|^2 - |D'^{-1}_{oo}\phi_o|^2 - |\hat{D}'^{-1}_{ee}\chi_e|^2}$$

(A) $S_G - 2Tr[Log[1+T]]$: Local action (UV physics)

(B) $|D'^{-1}_{ee}\phi_e|^2, |D'^{-1}_{oo}\phi_o|^2$: Semi Local action (UV physics)

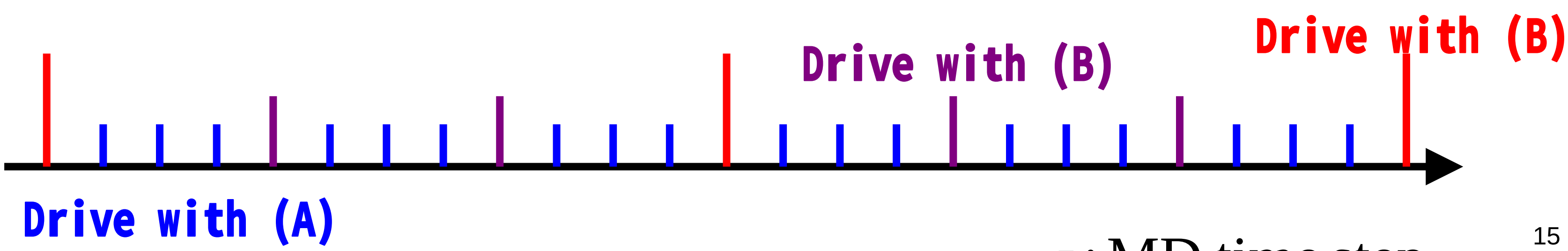
(C) $|\hat{D}'^{-1}_{ee}\chi_e|^2$: Non Local action (IR physics)

- HMC algorithm with this action
  - MD integrator = Sexton-Weingarten multi time scale MD.

$$\frac{dU_\mu}{d\tau} = iP_\mu U_\mu , \quad \frac{dP_\mu}{d\tau} = F_\mu$$

F_μ : MD force from action

- MD Force strength (A) > (B) > (C)



τ : MD time step

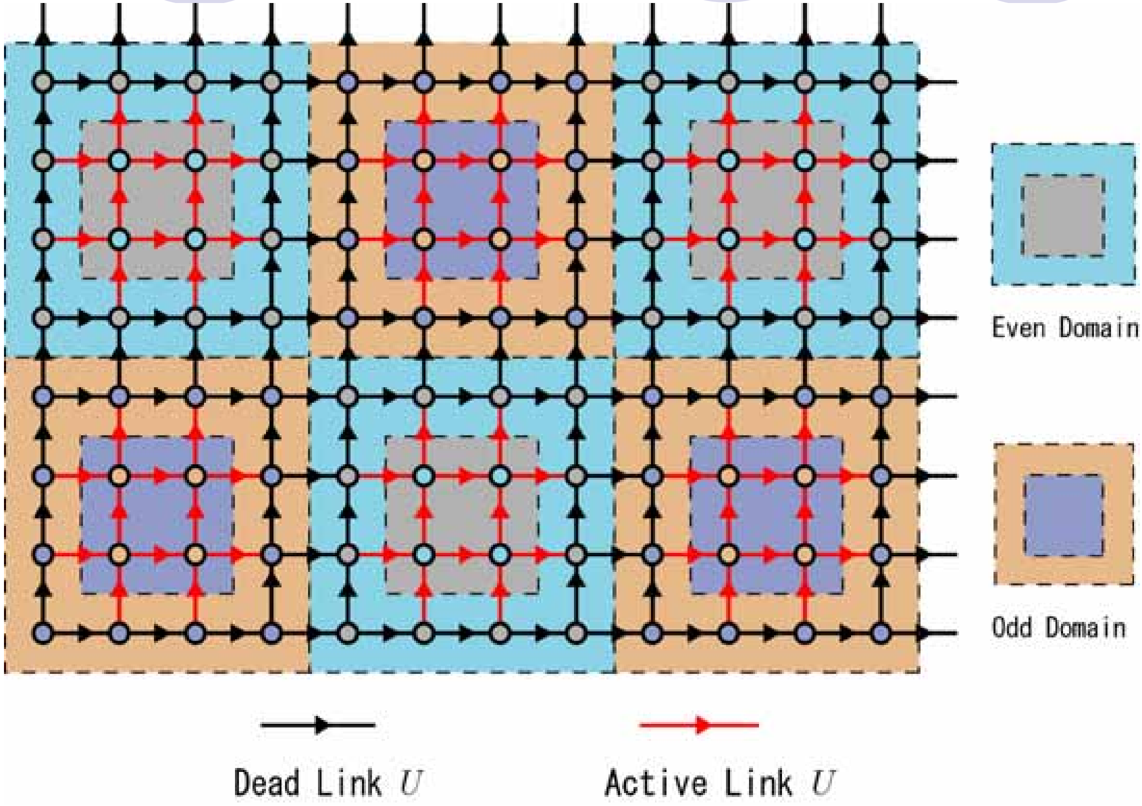
- Dead/Alive link method
  - Using the multi times scale MD integrator, gauge links are frequently updated.
  - During the MD evolution full link and momentum update requires communication every after link update:

$$U_{\mu}(n)_{[\tau+\Delta\tau]} = \exp[i\Delta\tau\, P_{\mu}(n)_{[\tau]}]U_{\mu}(n)_{[\tau]}$$

  - This communication cost can be reduced by restricting links to be updated.

4.Algorithm (cont'd)

- Dead/Alive link method



- Do not integrate;
  - Links on the single slice surface of each hyper cube blocks
  - Block connecting links

- Dead/Alive link method
  - In order to work with this method, MD Force for even/odd domain active links should not contains odd/even domain active links.
  - Iwasaki-RG and O(a)-improved (Clover) quarks. OK
    - The ratio active/dead link is important in autocorrelation.

      Larger block size is preferred, but it requires more powerful single nodes. (Cost vs autocorrelation tradeoff?)
  - To evolve dead links, random shift of lattice origin is carried out after each HMC trajectory.

UV-filtered PHMC algorithm for Nf=1 part

- UV-filter preconditioner <= Proposed for Multiboson algorithm

  [de Forcrand, NPB(Proc.Suppl.)73(Lat98);
  Alexandrou, de Forcrand, D'Elia & Panagopoulos, PRD61(00)]

- We apply UV-filter preconditioner to Polynomial HMC algorithm
  - We start with globally even-odd site preconditioned Wilson-Dirac operator

$$D_{ee} = 1 - (1+T)^{-1}{}_{ee} M_{eo} (1+T)^{-1}{}_{oo} M_{oe} = 1 - \hat{M}_{ee}$$

  - The UV-filter preconditioner $P[U]$

$$P[U] = \exp[s\hat{M}_{ee}]$$

s: tunable parameter.

4.Algorithm (cont'd)

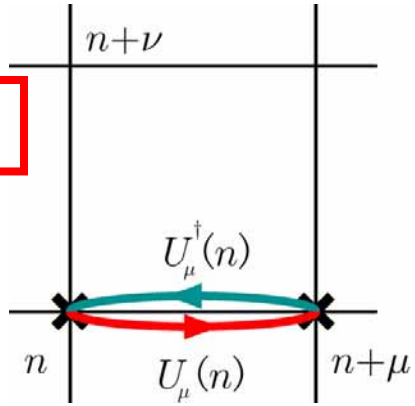
- Using the preconditioner

$$\det[D] = \det[D_{ee}] = \det[D_{ee}\exp[s\hat{M}_{ee}]\exp[-s\hat{M}_{ee}]]$$

$$= \det[D_{ee}\exp[s\hat{M}_{ee}]] \times e^{-sTr[\hat{M}_{ee}]}$$

$$= \det[Q] \times e^{-S_{UV}}$$

UV-IR Separation



where Suv is still local action.

$$S_{UV} = s\mathrm{Tr}[\hat{M}_{ee}]$$

$$= s\kappa^2 \sum_{n,\mu} \mathrm{tr}_{\mathrm{color,dirac}}[(1+T)^{-1}(n)(1-\gamma_\mu)U_\mu(n)(1+T)^{-1}(n+\mu)(1+\gamma_\mu)U_\mu^+(n)]$$

- Q is preconditioned operator. When s=1,

$$Q[U] = P[U]D[U] = \exp[\hat{M}_{ee}](1-\hat{M}_{ee}) = 1 - \frac{(\hat{M}_{ee})^2}{2} - \frac{(\hat{M}_{ee})^3}{3} - \ldots$$

$Q = 1 + O(K\text{^}4)$

- Introducing polynomial approximation for *1/Q*, Nf=1 part partition function becomes

$$Q^{-1} \approx \sum_{j=0}^{N_{poly}} c_j (\hat{M}_{ee})^j = \left[\sum_{j=0}^{N_{poly}/2} d_j^* (\hat{M}_{ee})^j \right] \left[\sum_{j=0}^{N_{poly}/2} d_j (\hat{M}_{ee})^j \right] = T^* T$$

$$\begin{aligned} \det[D] &= \det[Q] e^{-S_{UV}} \\ &= \frac{\det[Q(T^* T)]}{\det[T^* T]} e^{-S_{UV}} = \frac{\det[W]}{|\det[T]|^2} e^{-S_{UV}} \\ &= \int D\varphi_e \det[W] \exp[-S_{UV} - |T\varphi_e|^2] \end{aligned}$$

$$\varphi_e : \text{ even-site pseudo-fermion field}$$

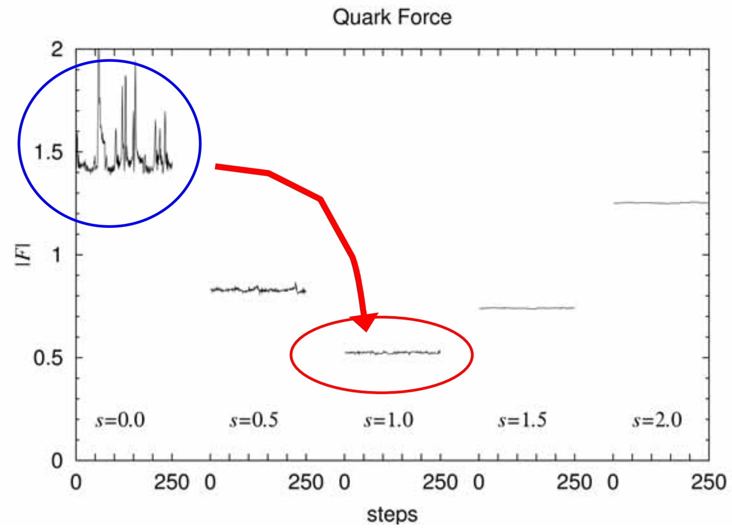
- *W* : correction factor. $W \approx 1$ Noisy Metropolis test

- Nf=1 UVPHMC does not fit with the domain decomposition structure of Nf=2 part.

- Multiplying $\hat{M}_{ee}$ always requires communication.
- However the UV-filter preconditioning reduces the magnitude of the MD force.
  - A Test result with Nf=2 UVPHMC algorithm [PACS-CS collab:Lattice06]

$$16^3 \times 48, \beta = 5.2, c_{sw} = 2.02, \kappa = 0.1350, N_f = 2$$

Factor 3 reduction => factor 2 speed up using Sexton-Weingarten method



- We also expect that the computational cost from Nf=1 part is not so high. (Nf=1 ← strange quark : rather heavy mass)

Nf=2+1 Algorithm (Summary)

- Nf=2 part: Lüscher’s domain decomposition preconditioned HMC algorithm
  - Full lattice D x = b solver,

    Lüscher’s [domain decomposition + Neumann] SAP single prec. -preconditioned GCR(k) solver
- Nf=1 part: UV-filtered PHMC algorithm

5. Numerical algorithm test

- TEST1 (1 strange quark + 4 light quark masses)(SR11000)
- No UV-filter for Nf=1 part

$$16^3 \times 32, \beta = 1.9, c_{sw} = 1.715, \kappa_s = 0.1364 \qquad a \approx 0.1\, fm$$

$$\kappa_{ud} = 0.13700, \quad 0.13741, \quad 0.13759, \quad 0.13770$$

- At the lightest quark mass (K=0.13770)
  - MD time scale
  - Block size 8^4

$$d\tau_0 = \tau / (N_0 N_1 N_2) \quad : \quad \text{Guage} + \text{TrLog(Clover)}$$

$$d\tau_1 = \tau / (N_1 N_2) \quad : \quad \text{Nf} = 2 \text{ UV part} + \text{Nf} = 1$$

$$d\tau_2 = \tau / (N_2) \quad : \quad \text{Nf} = 2 \text{ IR part}$$

900 traj. result

Kuramashi,Lat06

$$\tau = 0.5/\sqrt{2}, [N_0, N_1, N_2] = [4,5,14] \qquad \Rightarrow P_{acc}(\text{HMC}) = 86(2)\%$$

$$N_{poly} = 140 \qquad \Rightarrow P_{acc}(\text{GMP}) = 93(1)\%$$

$$m_{ud}^{AWI} = 15.2(12)\ \text{MeV}, \quad M_{PS(\pi)} = 313(16)\ \text{MeV} \quad (\text{volume eff.})$$

300MeV simulation is possible

5. Numerical algorithm test (cont'd)

- TEST2 with UV-filtering for Nf=1 part (SR11000)

$$20^3 \times 40, \beta = 1.9, c_{sw} = 1.715, \kappa_s = 0.1358, \kappa_{ud} = 0.13770$$

$$N_{poly} = 110, s = 1$$

preliminary

- MD time scale
- Block size 10^4

$$d\tau_0 = \tau/(N_0 N_1 N_2 N_3) \quad : \text{Guage} + \text{TrLog(Clover)}$$

$$d\tau_1 = \tau/(N_1 N_2 N_3) \quad : \text{Nf} = 2 \text{ UV part}$$

$$d\tau_2 = \tau/(N_2 N_3) \quad : \text{Nf} = 1 \text{ IR (polynomial) part}$$

$$d\tau_3 = \tau/(N_3) \quad : \text{Nf} = 2 \text{ IR part} + \text{Nf} = 1 \text{ UV part}$$

450 traj. result

$$\tau = 1, [N_0, N_1, N_2, N_3] = [4,4,2,8] \quad \Rightarrow P_{acc}(\text{HMC}) = 74.2(3)\%$$

$$N_{poly} = 110 \quad \Rightarrow P_{acc}(\text{GMP}) = 93.3(1)\%$$

- Even-odd site preconditioned HMC + PHMC algorithm without Sexton-Weingarten requires [CP-PACS JLQCD]

$$\tau = 1, d\tau = 1/128 \quad \Rightarrow P_{acc}(\text{HMC}) \approx 87\%$$

6. Summary

- We have started Nf=2+1 simulations on PACS-CS computer using the described algorithm.
  (32^3x64 lattices, m_{ud} = 60-15MeV)
  - We are now optimizing the computational kernel (Mult) for Xeon CPU using Intel SSE/SSE2/SSE3 instruction.

  - Study on 8x8x8x64 result (not for domain decomp'd version)
    - C with inline assembler 1.87Gflops (33%)
    - C with Intel intrinsic function 1.91Gflops (34%)
    - Fortran 1.45Gflops (26%)
  - Preliminary study : time spent in Nf=2 UV part 25%
    time spent in Nf=2 IR part 25%
  - Solver improvement using single precision preconditioner

$$D_{ee}x_e = b_e \qquad \text{(in even/odd domain)}$$

GCR(k) solver with [SSOR + Neumann single prec.-preconditioner].
40% faster than BiCGStab(L=2) with SSOR preconditoner (in real time).