USQCD Software:
Status and Future Challenges

Richard C. Brower
All Hands Meeting @ FNAL
May15-16 , 2009

Code distribution: http://www.usqcd.org/software.html

Topics (for Round Table)

- Status:
  - Slides from SciDAC-2 review, Jan 8-9, 2009

- Some Future Challenges:
  - Visualization
  - QMT: Threads in Chroma
  - GPGPU code: clover Wilson on Nvidia
  - BG/Q, Cell (Roadrunner/QPACE), BlueWaters,..
  - Multi-grid and multi-lattice API for QDP
  - Discussion of Performance metrics

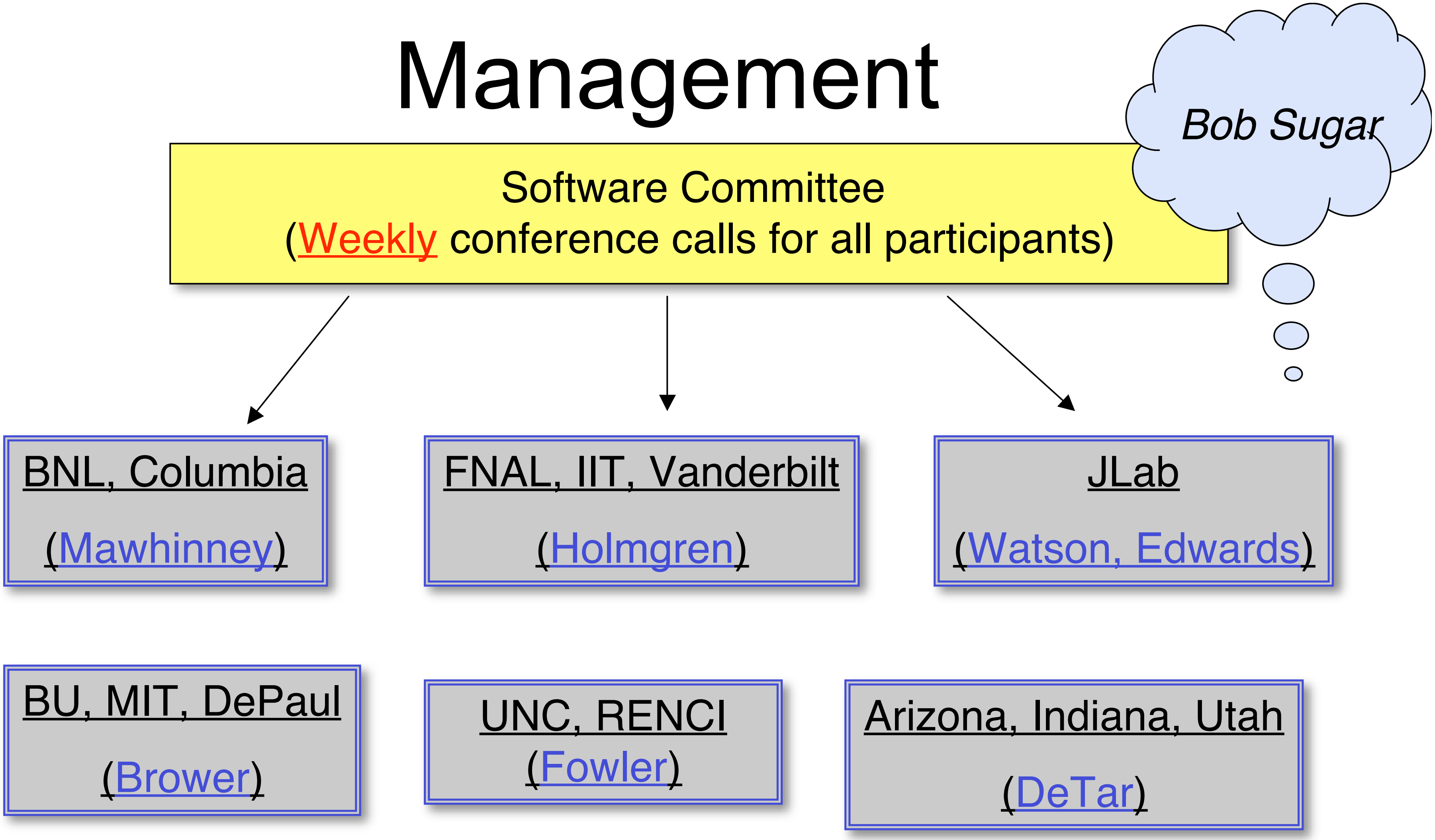
LGT SciDAC Software Committee

Rich Brower (chair)	brower@bu.edu
Carleton DeTar	detar@physics.utah.edu
Robert Edwards	edwards@jlab.org
Rob Fowler	rjf@renci.org
Don Holmgren	djholm@fnal.gov
Bob Mawhinney	rmd@phys.columbia.edu
Pavlos Vranas	vranas2@llnl.gov
Chip Watson	watson@jlab.org

Major Participants in SciDAC Project

Arizona	Doug Toussaint	North Carolina	Rob Fowler*
	Alexei Bazavov		Allan Porterfield
BU	Rich Brower *		Pat Dreher
	Ron Babich	ALCF	James Osborn
	Mike Clark	JLab	Chip Watson*
BNL	Chulwoo Jung		Robert Edwards*
	Oliver Witzel		Jie Chen
	Efstratios Efstathiadis		Balint Joo
Columbia	Bob Mawhinney *	Indiana	Steve Gottlieb
DePaul	Massimo DiPierro		Subhasish Basak
FNAL	Don Holmgren *	Utah	Carleton DeTar *
	Jim Simone		Mehmet Oktay
	Jim Kowalkowski	Vanderbilt	Theodore Bapty
	Amitoj Singh		Abhishek Dubey
LLNL	Pavlos Vranas*		Sandeep Neema
MIT	Andrew Pochinsky	IIT	Xien-He Sun
	Joy Khoriaty		Luciano Piccoli

Management



Bob Sugar

Software Committee
(Weekly conference calls for all participants)

BNL, Columbia
(Mawhinney)

FNAL, IIT, Vanderbilt
(Holmgren)

JLab
(Watson, Edwards)

BU, MIT, DePaul
(Brower)

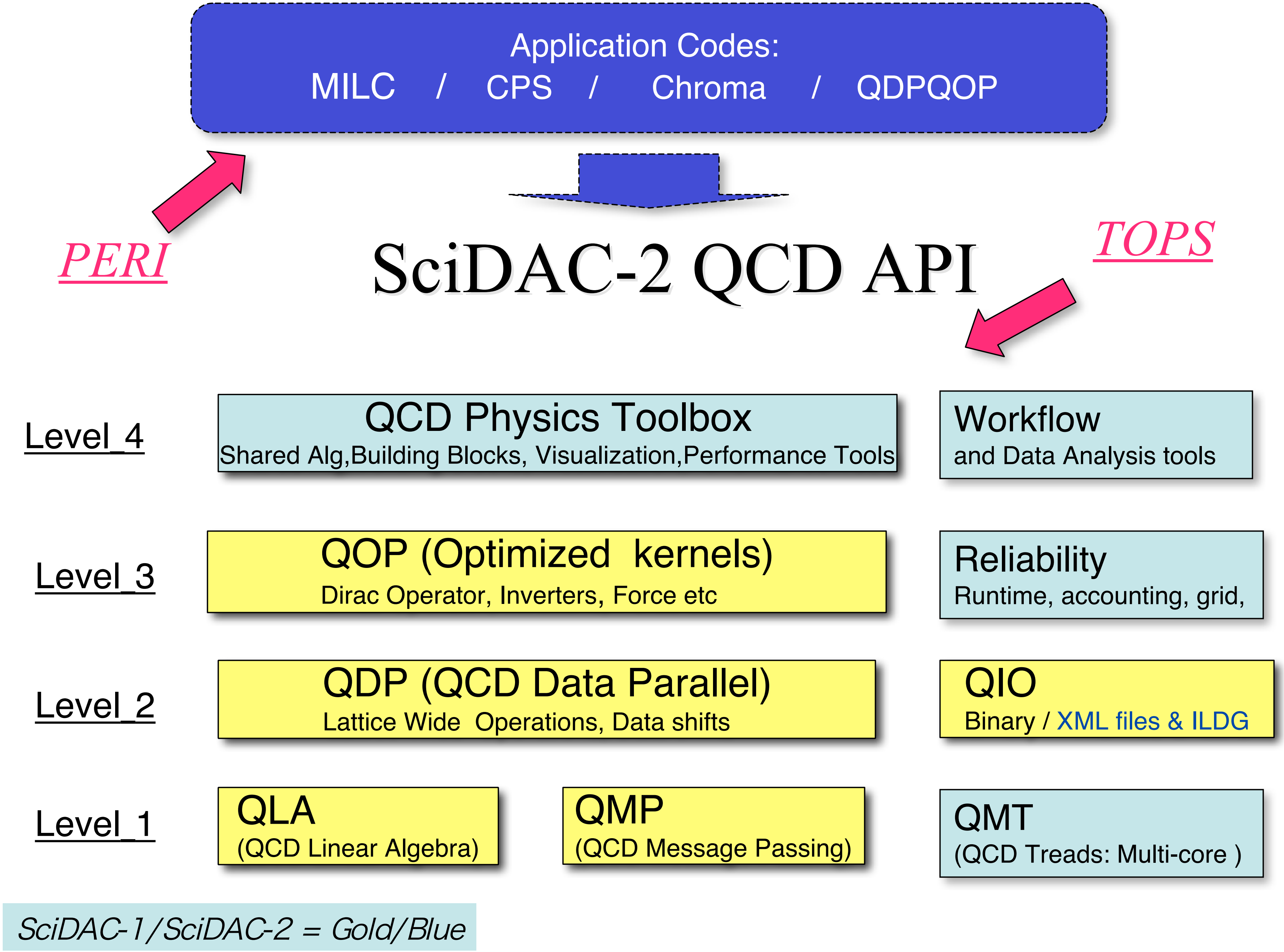
UNC, RENCI
(Fowler)

Arizona, Indiana, Utah
(DeTar)

ANL, LNLL
(Vranas)

Annual workshops to plan next phase:
Oct 27-28, 2006; Feb. 1-2, 2008; Nov. 7-8, 2008
http://super.bu.edu/~brower/scidacFNAL2008/

SciDAC-2 QCD API



Application Codes:
MILC / CPS / Chroma / QDPQOP

PERI

TOPS

Level_4	**QCD Physics Toolbox** Shared Alg,Building Blocks, Visualization,Performance Tools	**Workflow** and Data Analysis tools
Level_3	**QOP (Optimized kernels)** Dirac Operator, Inverters, Force etc	**Reliability** Runtime, accounting, grid,
Level_2	**QDP (QCD Data Parallel)** Lattice Wide Operations, Data shifts	**QIO** Binary / XML files & ILDG
Level_1	**QLA** (QCD Linear Algebra) **QMP** (QCD Message Passing)	**QMT** (QCD Treads: Multi-core)

SciDAC-1/SciDAC-2 = Gold/Blue

SciDAC-2 Accomplishments

- Pre-existing code compliance
  - Integrate SciDAC modules into MILC (Carleton DeTar)
  - Integrate SciDAC modules into CPS (Chulwoo Jung)
- Porting API to new Platforms
  - High performance on BG/L & BG/P
  - High performance on Cray XT4 (Balint Joo)
  - Level 3 code generator (QA0), MDWF (John Negele)
- Algorithms/Chroma
  - Tool Box -- shared building blocks (Robert Edwards)
  - Eigenvalue deflation code: EigCG
  - 4-d Wilson Multi-grid: TOPS/QCD coll. (Rob Falgout)
  - International Workshop on Numerical Analysis and Lattice QCD [(http://homepages.uni-regensburg.de/~blj05290/qcdna08/index.shtml)](http://homepages.uni-regensburg.de/~blj05290/qcdna08/index.shtml)

New SciDAC-2 Projects

- Workflow (Jim Kowalkowski)
  - Prototype of workflow app at FNAL and JLab (Don Holmgren)
  - http://lqcd.fnal.gov/workflow/WorkflowProject.html

- Reliability
  - Prototype for monitoring and mitigation
  - data production and design of actuators

- Performance (Rob Fowler)
  - PERI analysis of Chroma and QDP++
  - Threading strategies on quad AMD
  - Development of toolkit for QCD visualization (Massimo DiPierro)
  - Conventions for storing time-slice data into VTK files
  - Data analysis tools

QCD Visualization

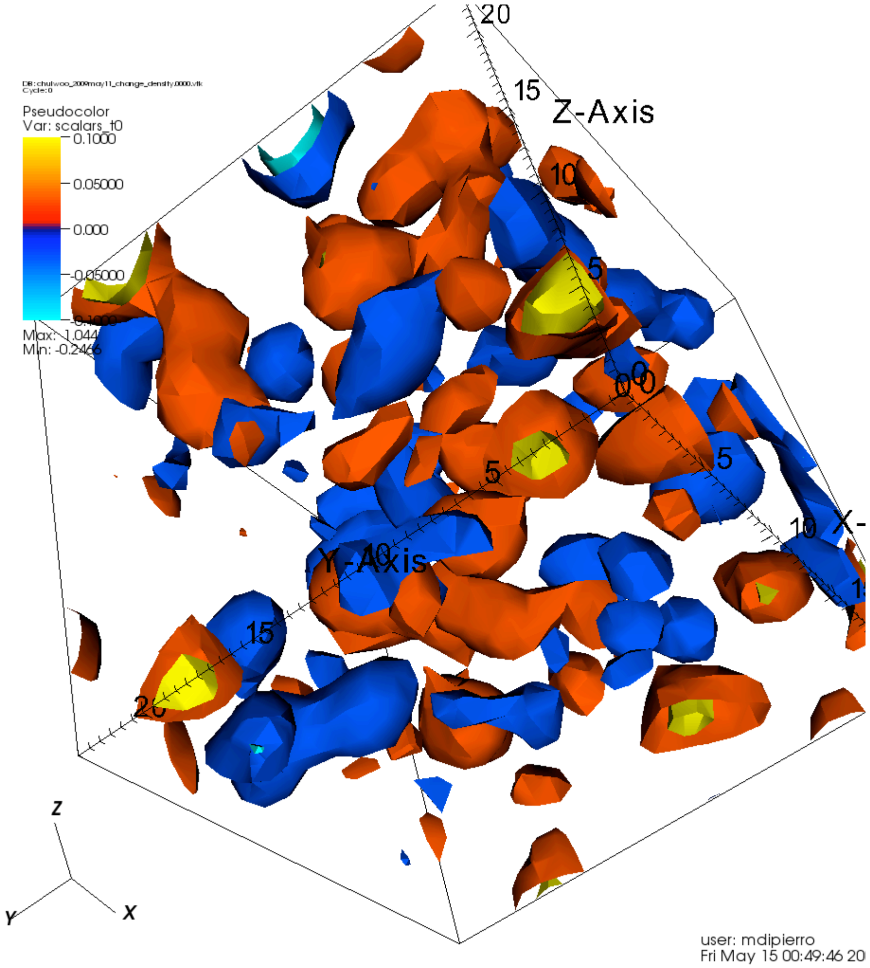
http://web2py.appspot.com/qcd

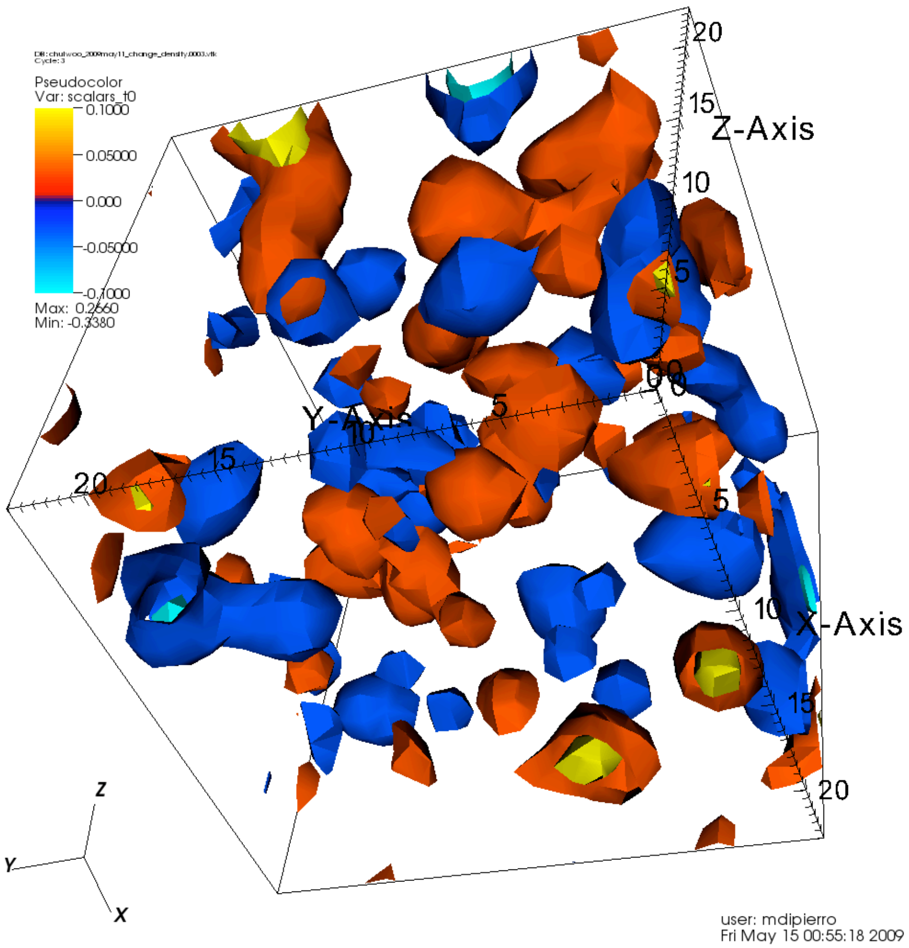
Visualization Runs

- Completed:
  - ~ 500 64×24^3 DWF RHMC (Chulwoo)
  - ~ 500 64×24^3 Hasenbusch Fermions with 2nd order Omelyan integrator (Mike Clark)
- In progress... Asqtad Fermions and different measses.

http://www.screencast.com/users/mdipierro/folders/Jing/media/3de0b1eb-11b0-463d-af28-9cee600a0dee

Topological Charge





Multi & Many-core Architectures

- New Paradigm: Multi-core not Hertz
  - Chips and architectures are rapidly evolving
  - Experimentation needed to design extensions to API

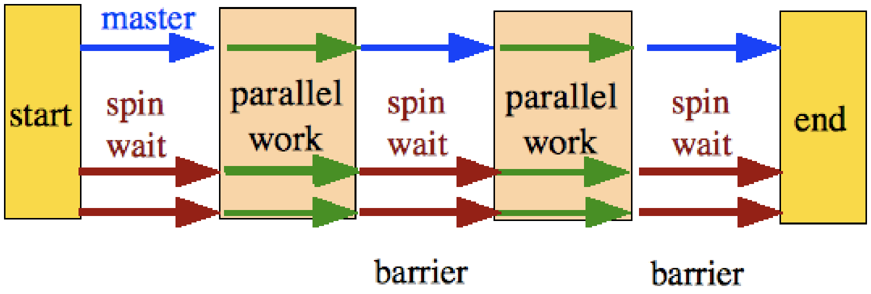
- Multi-core: O(10) (Balint Joo)
  - Evaluation of strategies (JLab, FNAL, PERC et al)
  - QMT: Collaboration with EPCC (Edinburgh, UKQCD)

- Many-core targets on horizon: O(100)
  - Cell: Roadrunner & QPACE (Krieg/Pochinsky) (John Negele)
  - BG/Q successor to QCDOC (RBC)
  - GPGPU: 240 core Nvidia case study (Rich Brower)
  - Power 7+ GPU(?): NSF BlueWaters
  - Intel Larabee chips

Threading in Chroma running on XT4

- Data Parallel Threading (OpenMP like)
- Jie Chen (JLab) developed QMT (QCD Multi Thread)
- Threading integrated into important QDP++ loops
  - SU(3)xSU(3), norm2(DiracFermion), innerProduct(DiracFermion)
  - Much of the work done by Xu Guo at EPCC, B. Joo did the reductions and some correctness checking. Many thanks to Xu and EPCC
- Threading integrated into important Chroma loops
  - clover, stout smearing : where we broke out of QDP++
- Threaded Chroma is running in production on Cray XT4s
  - see about a 36% improvement over PureMPI jobs with same core sizes.

QMT Highlights

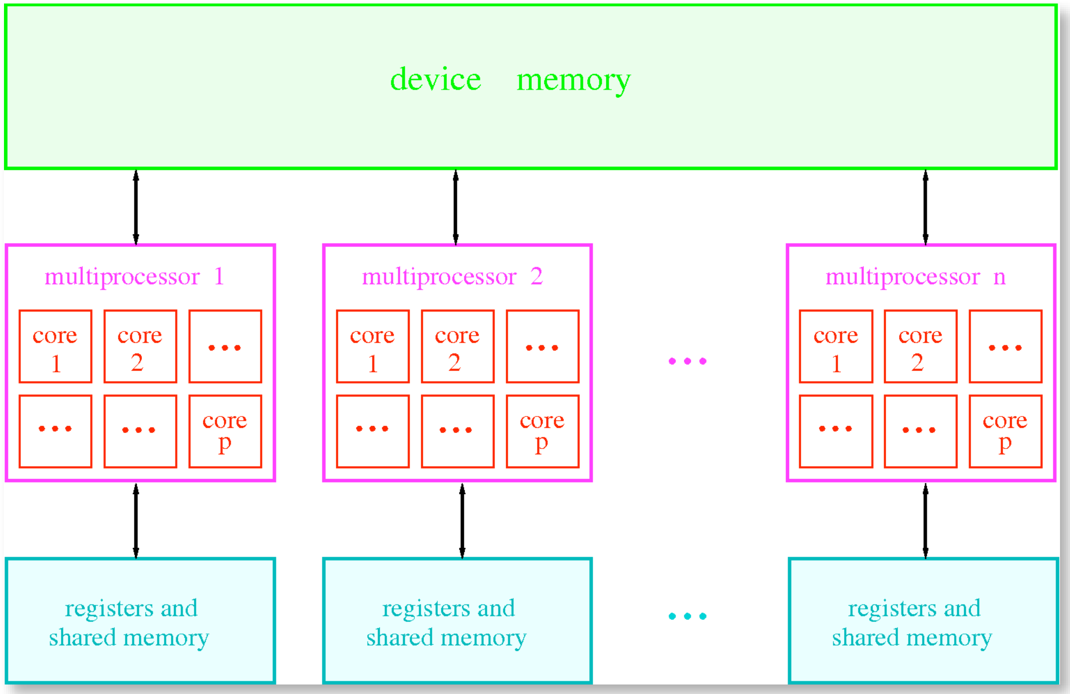


- Threads spawned at startup, joined at end
  - Worker threads spin waiting for work (never idled)
- Master thread shares in parallel work
- Parallel region ended with barrier; called automatically
- Opteron/Intel barrier uses cache coherency for speed
- Like OpenMP #omp_parallel over functions but
  - ThreadArgs and function need to be written for every case.

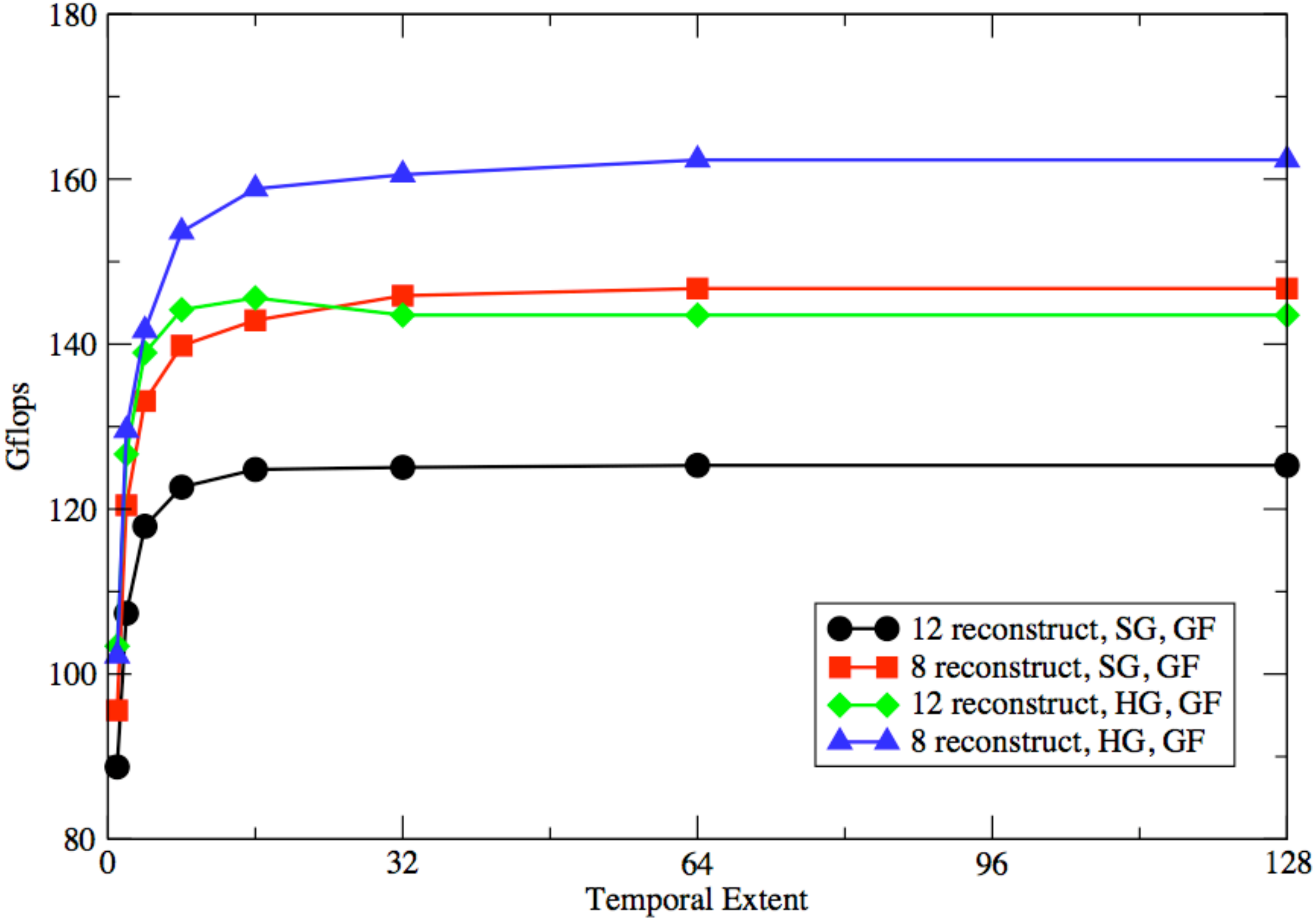
```
#define QUITE_LARGE 10000


typedef struct {
  float *float_array_param;
} ThreadArgs;


void threadedKernel( size_t lo, size_t hi, int id,
                     const void* args)
{
    const ThreadArgs* a = (const ThreadArgs *)args;
    float *fa = a->float_array_param;
    int i;
    for( i=lo; i < hi ; ++i) {   /* DO WORK FOR THREAD */ }
}


int main( int argc, char *argv[] )
{   float my_array[ QUITE_LARGE ];
    ThreadArgs a = { my_array };
    qmt_init();
    qmt_call( threadedKernel, QUITE_LARGE, &a );
    qmt_finalize();
}
```

SIMD threads on 240 core GPGPU

- *Coded in CUDA: Nvidia's SIMD extension for C*
- *Single GPU holds entire lattice*
- *One thread per site*

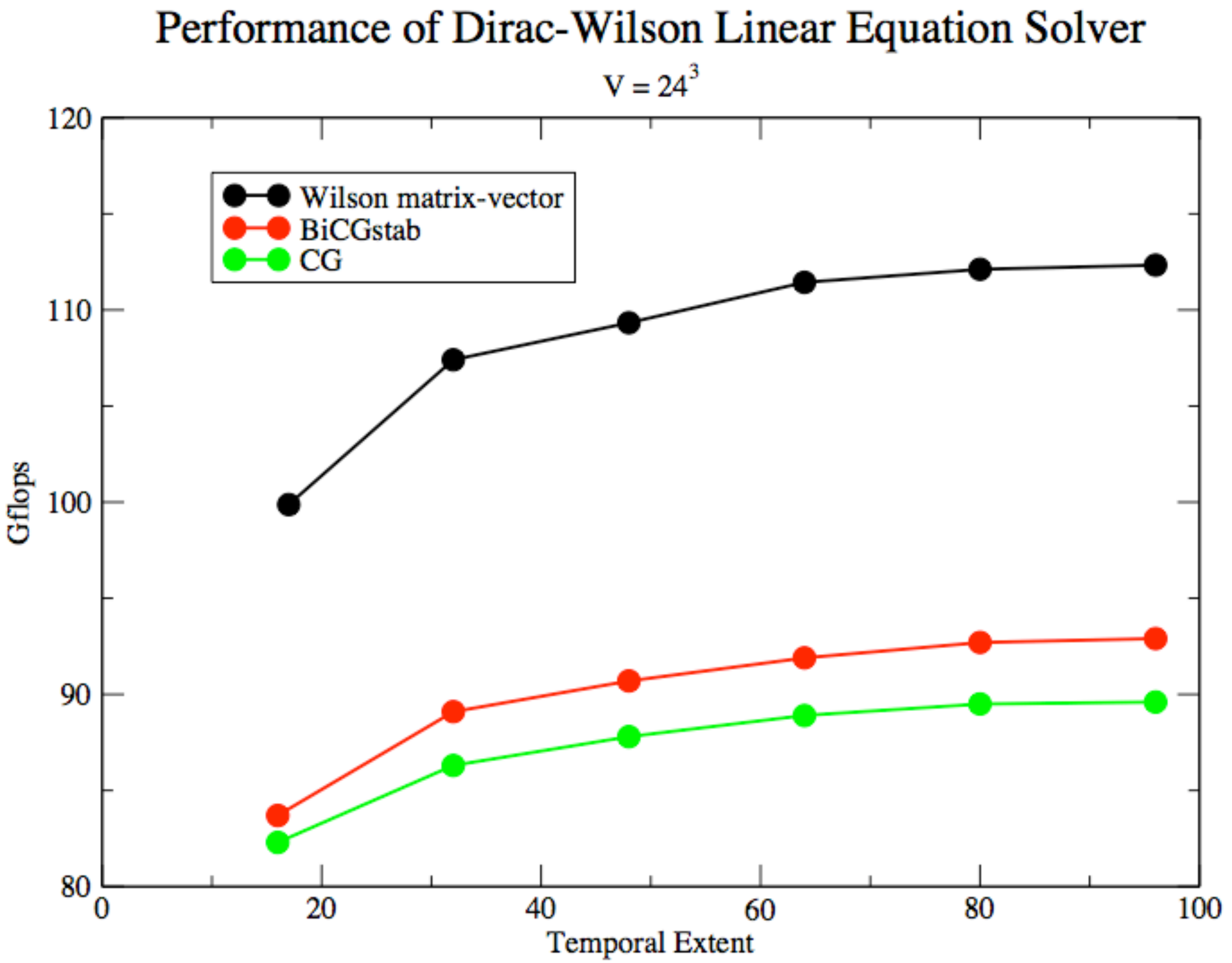


Soon a common language all GPGPU venders, Nvidia (Tesla), AMD/ATI and Intel (Larabee): OpenCL (Computing Language) http://www.khronos.org/registry/cl/



WILSON MATRIX-VECTOR PERFORMANCE

HALF PRECISION ($V=32^3xT$)


Performance of Dirac-Wilson Linear Equation Solver
$V = 24^3$
Wilson matrix-vector
BiCGstab
CG
Gflops
120
110
100
90
80
Temporal Extent
0
20
40
60
80
100

GPU Hardware

GTX 280
Flops: single 1 Tflop, double 80 Gflops
Memory 1GB, Bandwidth 141 GBs^{-1}
230 Watts, \$290

Tesla 1060
Flops: single 1 Tflop, double 80 Gflops
Memory 4GB, Bandwidth 102 GBs^{-1}
230 Watts, \$1200

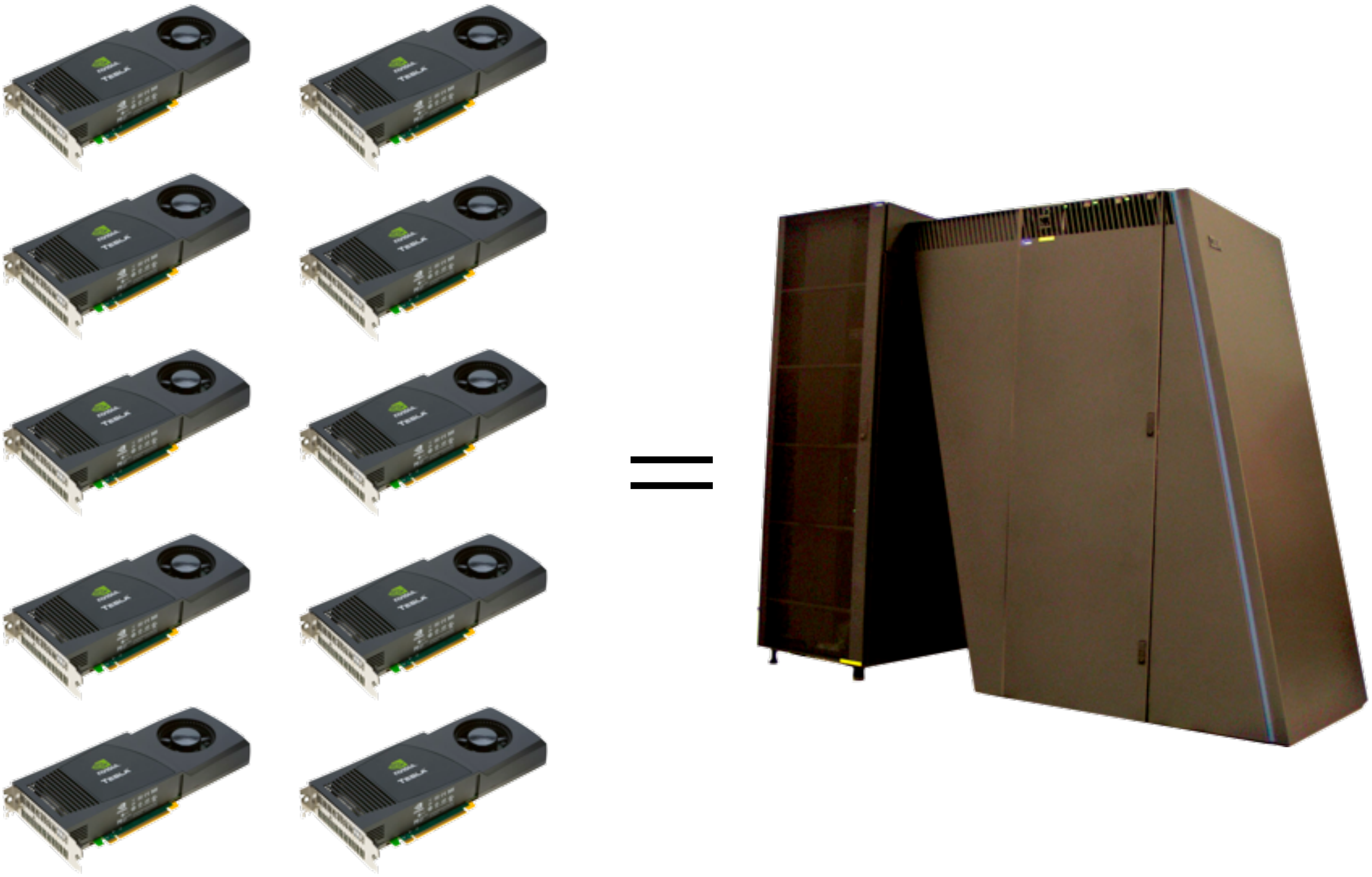
Tesla 1070
Flops: single 4 Tflops, double 320 Gflops
Memory 16GB, Bandwidth 408 GBs^{-1}
900 Watts, \$8000

Nvidia Tesla Quad S1070 1U System $8K

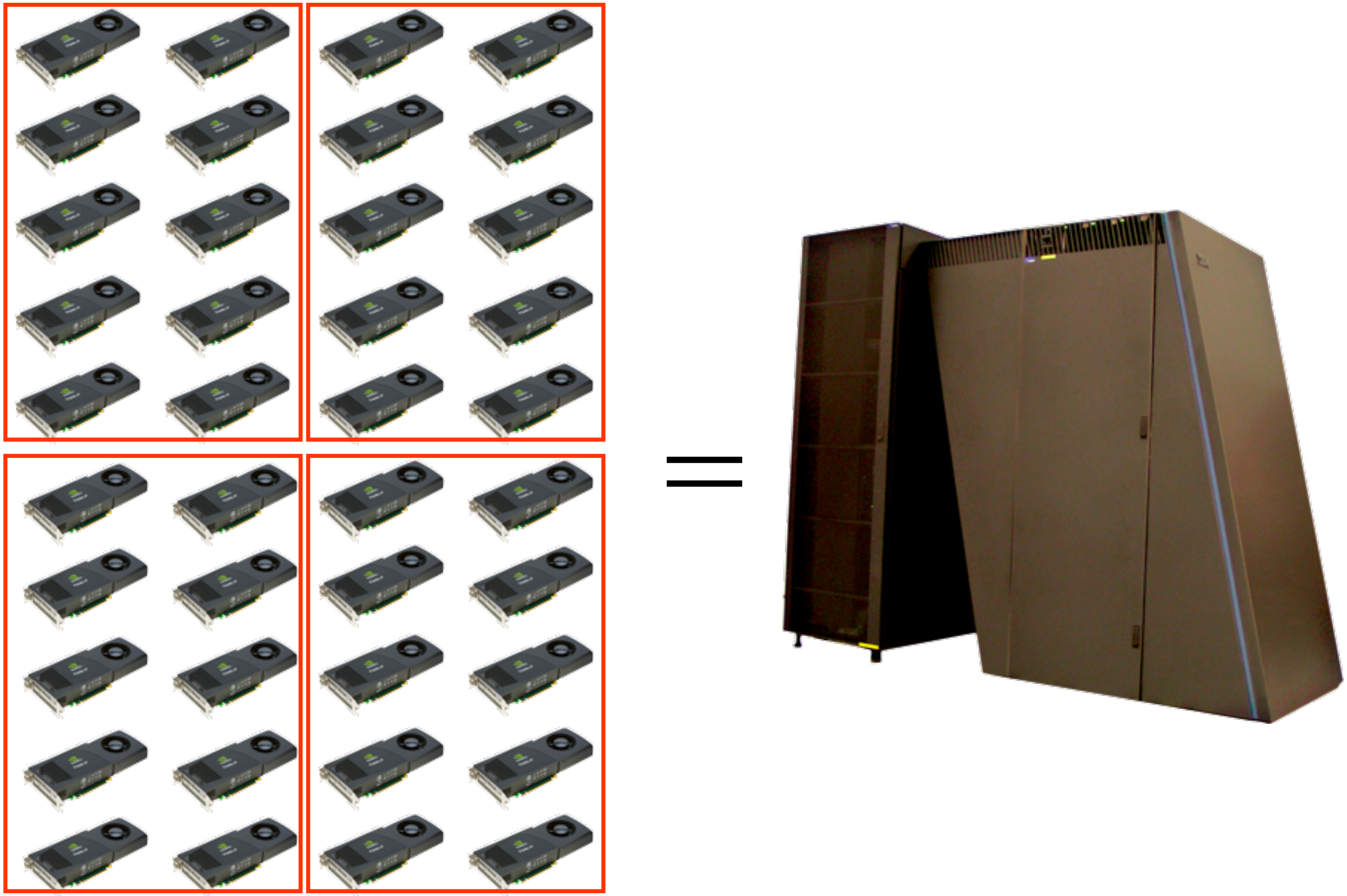
Processors	4 x Tesla T10P
Number of cores	960
Core clock	1.5 Hz
Performance	4 Teraflops
memory BW	16.0 GB
bandwidth	408 GB/sec
Memory I/0	2048 bit,800MHz
Form factor	1U (EIA 19” rack)
System I/O	2 PCIe x 16 Gen2
Typical power	700 W

- *SOFTWARE*
  - *Very fine grain threaded QCD code runs very well on 240 core single node*
  - *Classic algorithmic tricks plus SIMD coding style for software*
- *ANALYSIS CLUSTER:*
  - *8 Quad Tesla system with estimated 4 Teraflops sustained for about $100K hardware!*

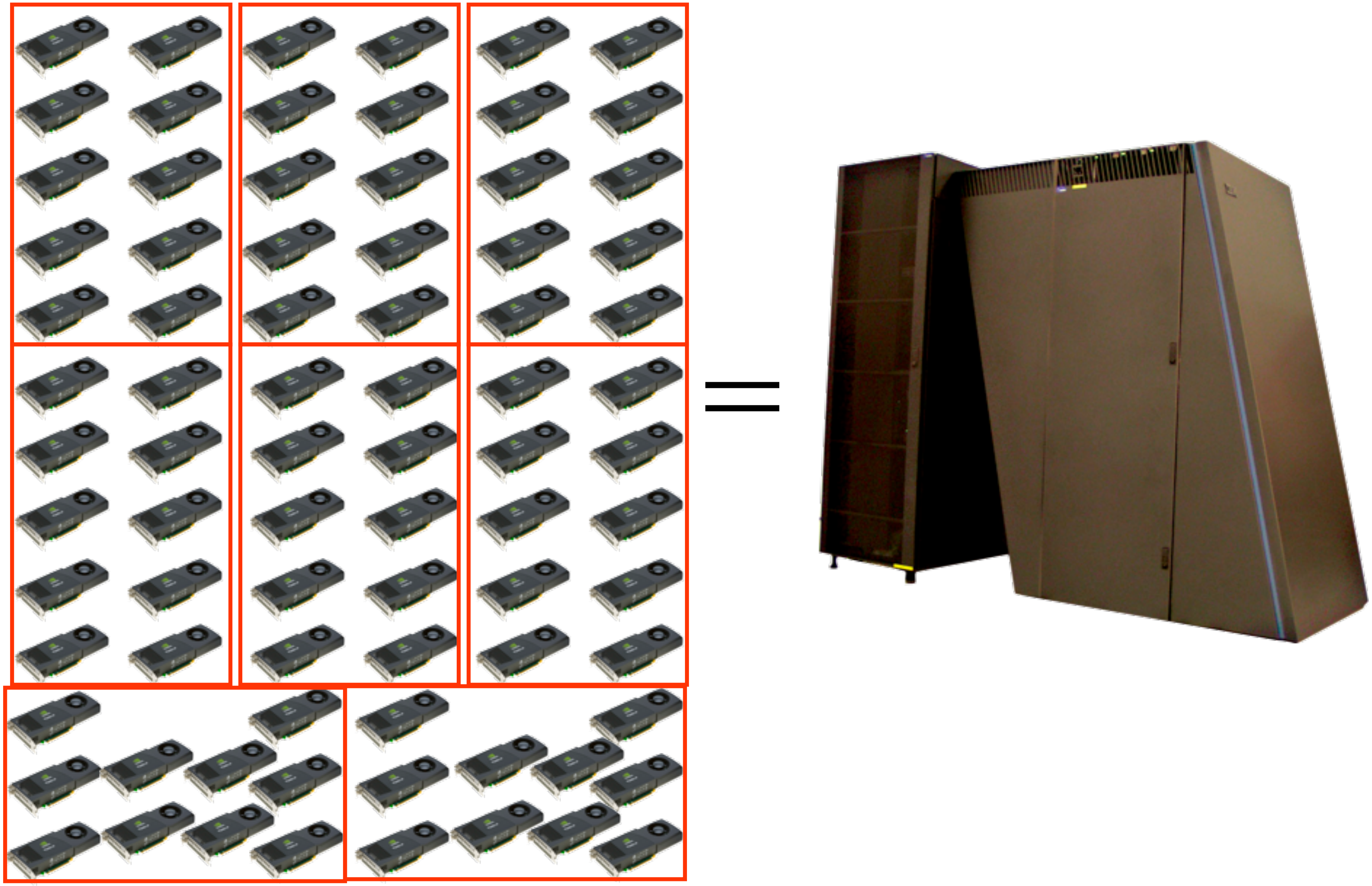
How Fast is Fast?



Performance Per Watt



Performance Per $



DATA: for high resolution QCD

- Lattice scales:
  - a(lattice) << $1/M_{proton}$ << $1/m_\pi$ << L (box)
  - 0.06 fermi << 0.2 fermi << 1.4 fermi << 6.0 fermi

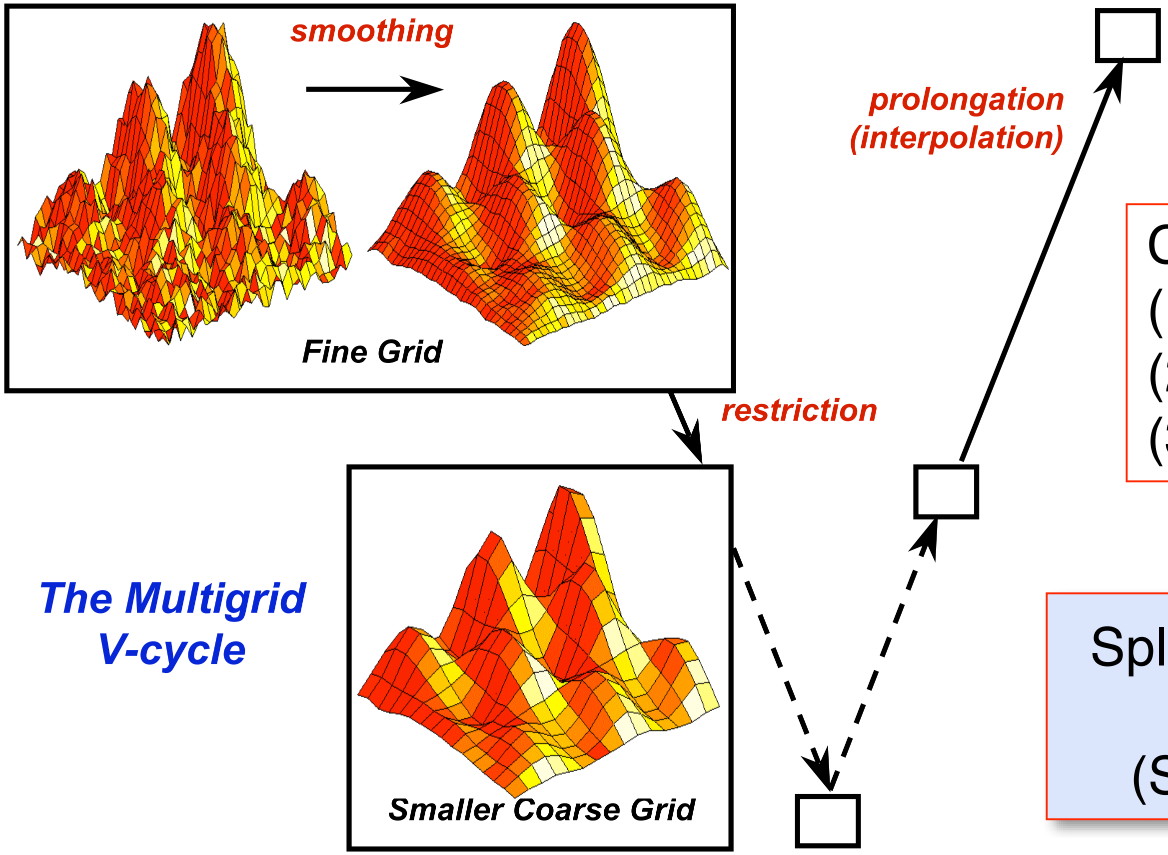
$$3.3 \quad x \quad 7 \quad x \quad 4.25 \simeq 100$$

- Opportunity for Multi-scale methods
  - Wilson MG and Schwarz “deflation” works!
  - Domain Wall is beginning to be understood?
  - Staggered soon by Carleton/Mehmet Oktay

ALGORITHM: curing ill-conditioning

Slow convergence of Dirac solver is due small eigenvalues for vectors in near-null space, $\mathcal{S}$.

$D: \mathcal{S} \simeq 0$

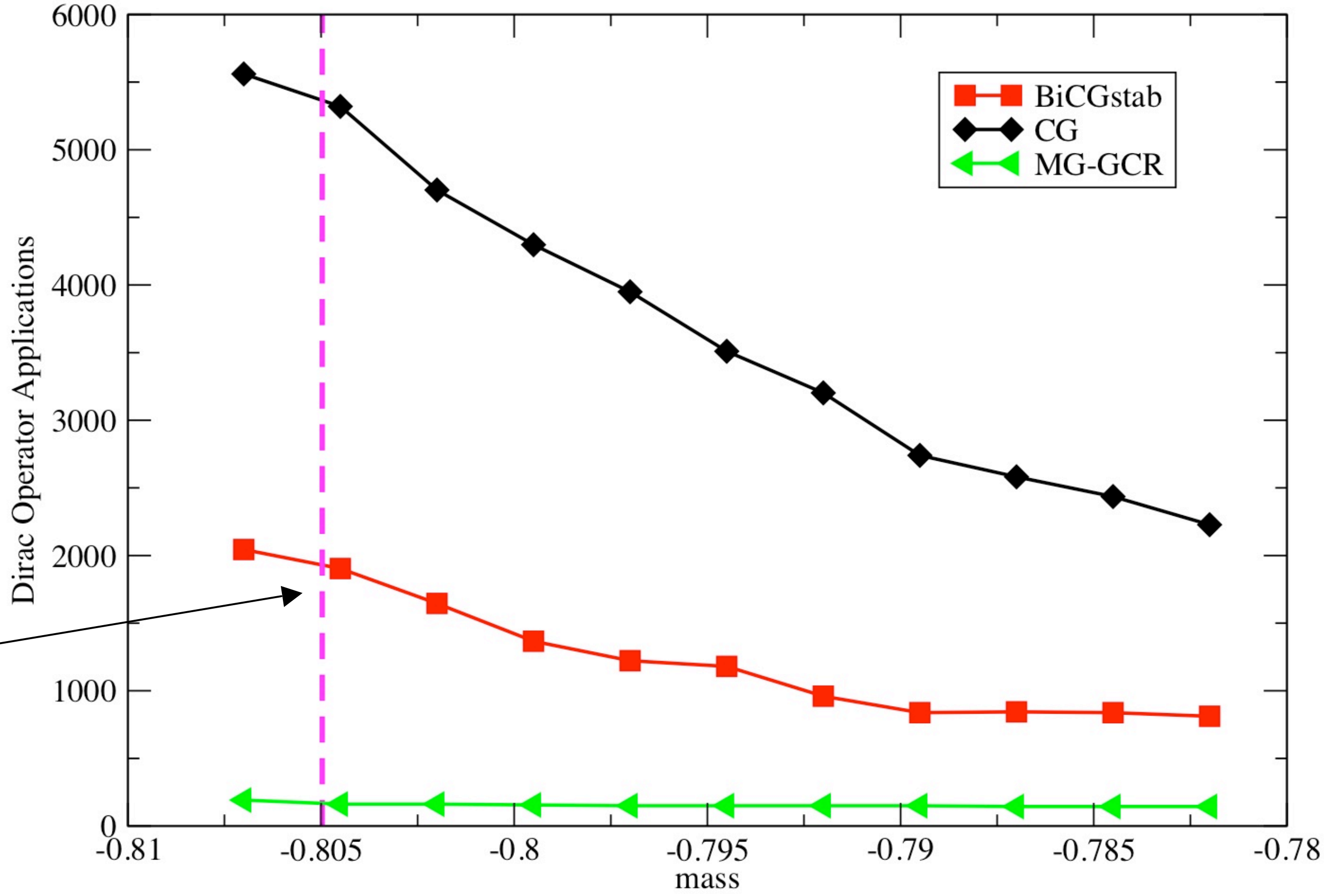


The Multigrid V-cycle

Common feature of
(1) Deflation (EigCG)
(2) Schwarz (Luescher)
(3) Multi-grid algorithms

Split space into near null $\mathcal{S}$
&
(Schur) complement $\mathcal{S}_\perp$.

Multigrid QCD *TOPS* project

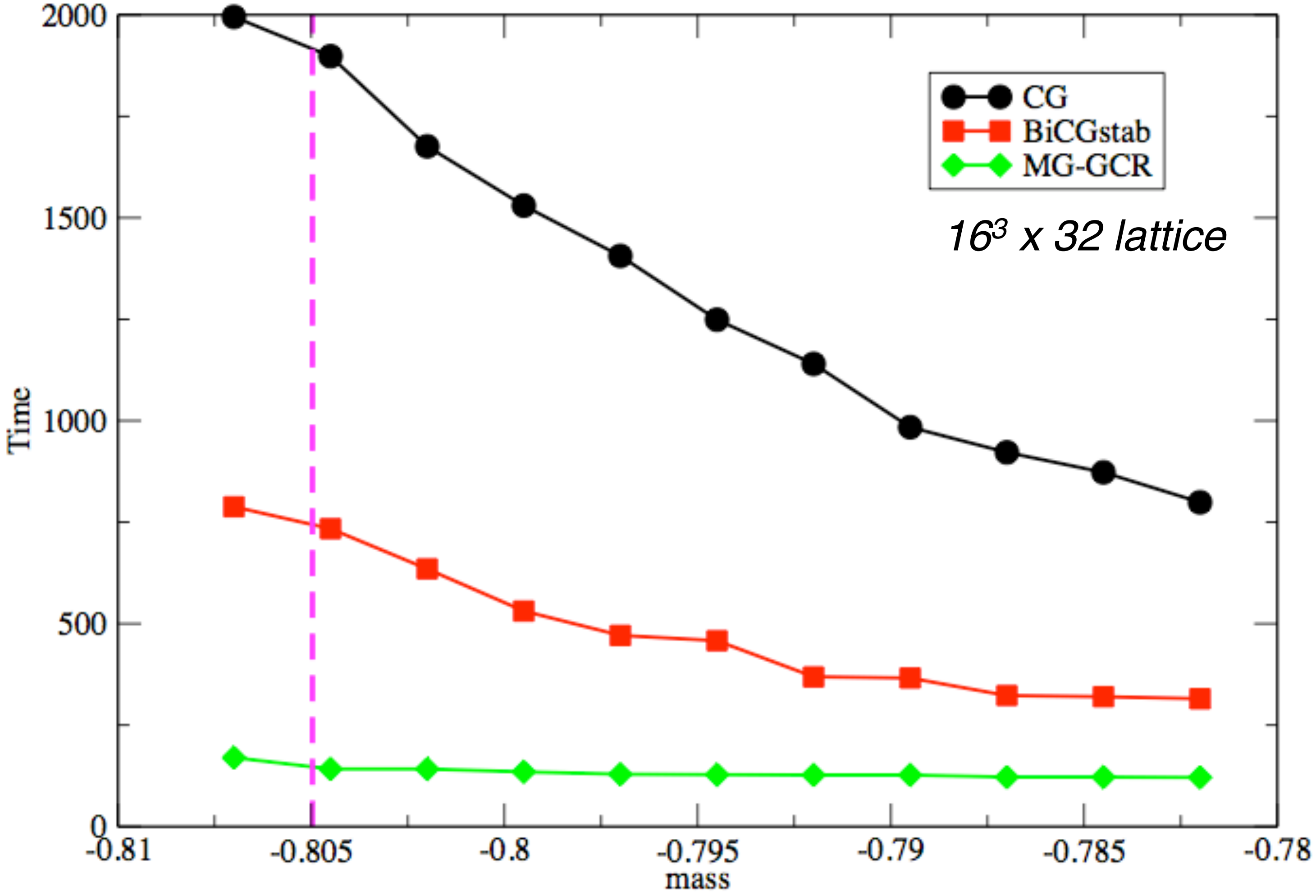


2000 iterations at limit of "zero mass gap"

α SA/αAMG: Adaptive Smooth Aggregations Algebraic MultiGrid

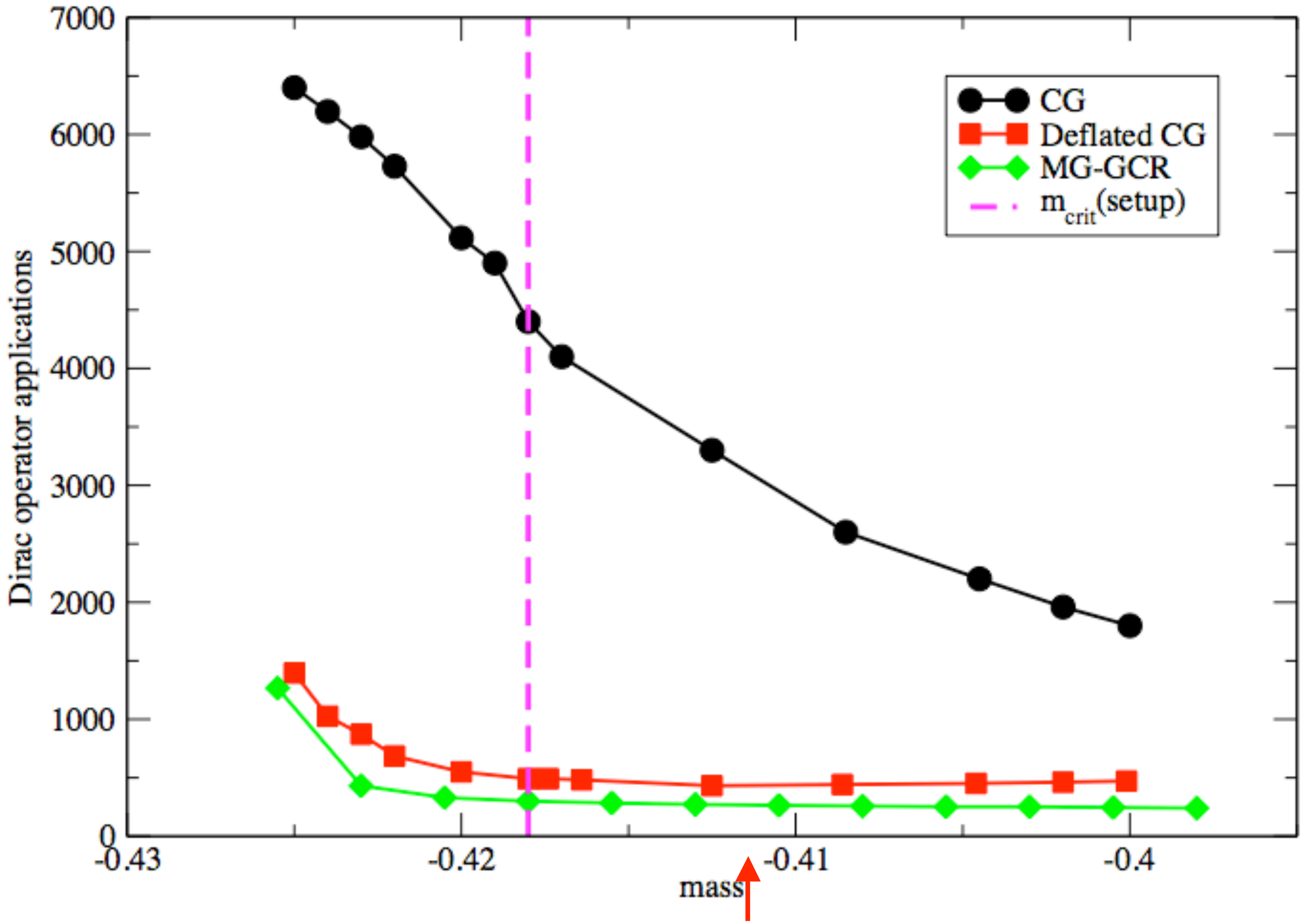
see Oct 10-10 workshop (http://super.bu.edu/~brower/MGqcd/)

Relative Execution times



Brannick, Brower, Clark, McCormick,Manteuffel,Osborn and Rebbi,
"The removal of critical slowing down" Lattice 2008 proceedings

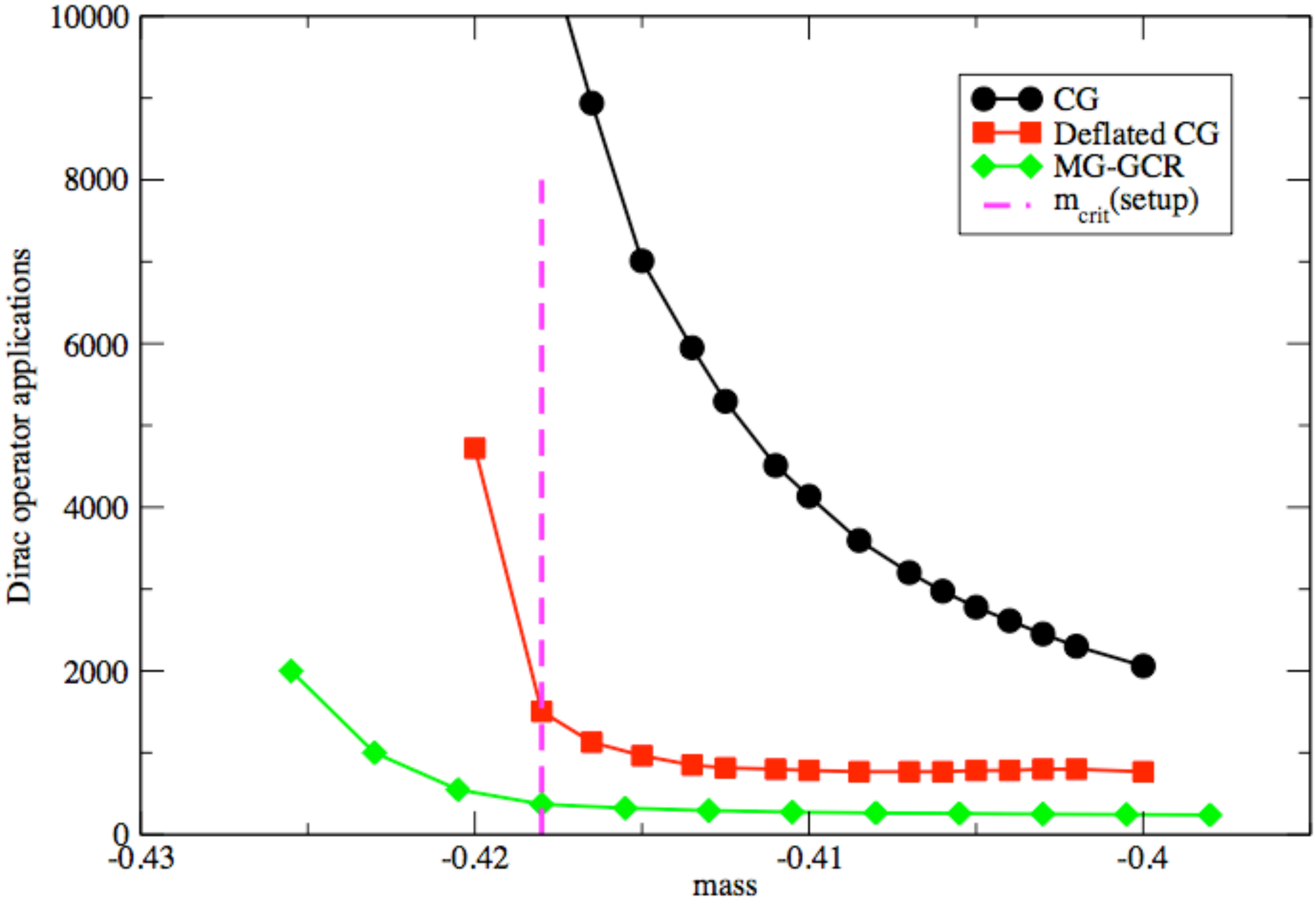
MG vs EigCG (240 ev)



$m_{sea} = -0.4125.$

16^3 x 64 asymmetric lattice

MG vs EigCG (240 ev)



24^3 x 64 asymmetric lattice

Multi-lattice extension to QDP

- Uses for multiple lattices within QDP:
  - “chopping” lattices in time direction
  - mixing 4d & 5d codes
  - multigrid algorithms
- Proposed features
- keep default lattice for backward compatibility
  - create new lattices
  - define custom site layout functions for lattices
  - create QDP fields on the new lattices

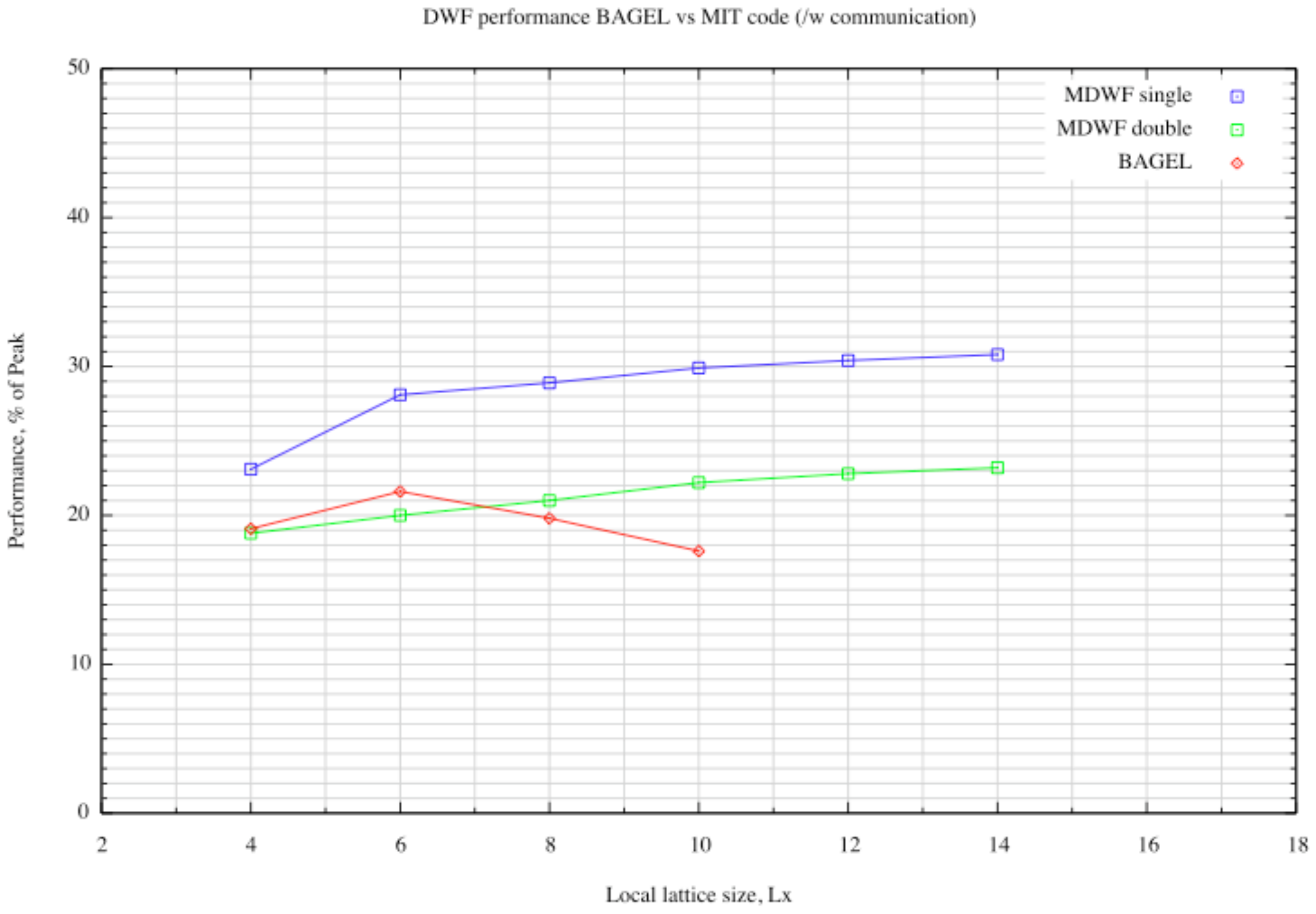
(James Osborn & Andrew)

  - define subsets on new lattices
  - define shift mappings between lattices and functions to apply them
  - include reduction operations as special case of shift
- existing math functions API doesn’t need changing
  - only allow operations among fields on same lattice
- also add ability for user defined field types
  - user specifies size of data per site
  - QDP handles layout/shifting
  - user can create math functions with inlined site loops

A. Pochinsky’s: Moebius DW Fermion Inverter

MDWF Performance: 4 - d Communications

3. Comparison of Bagel and MDWF (non-optimized) comm. in 4 dim

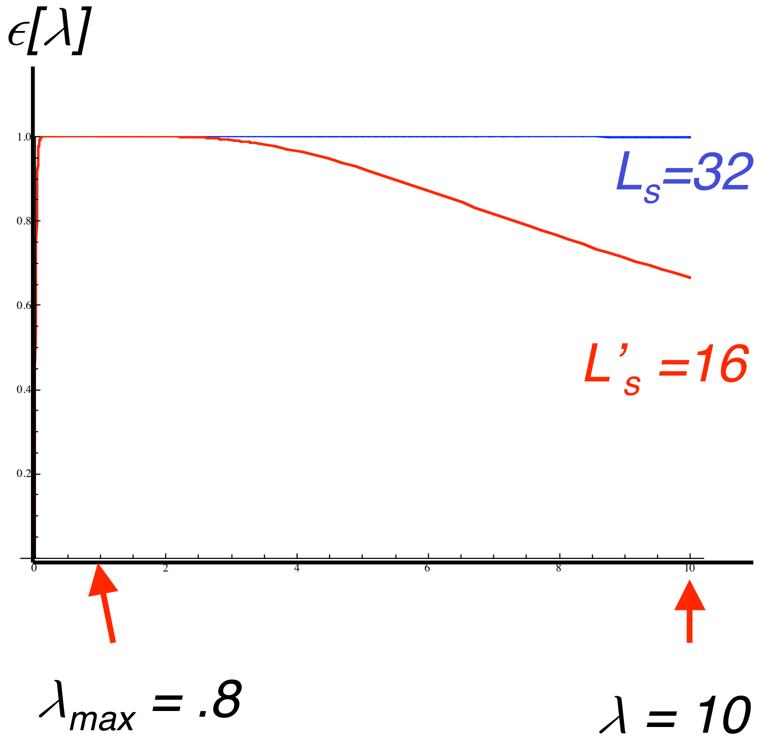


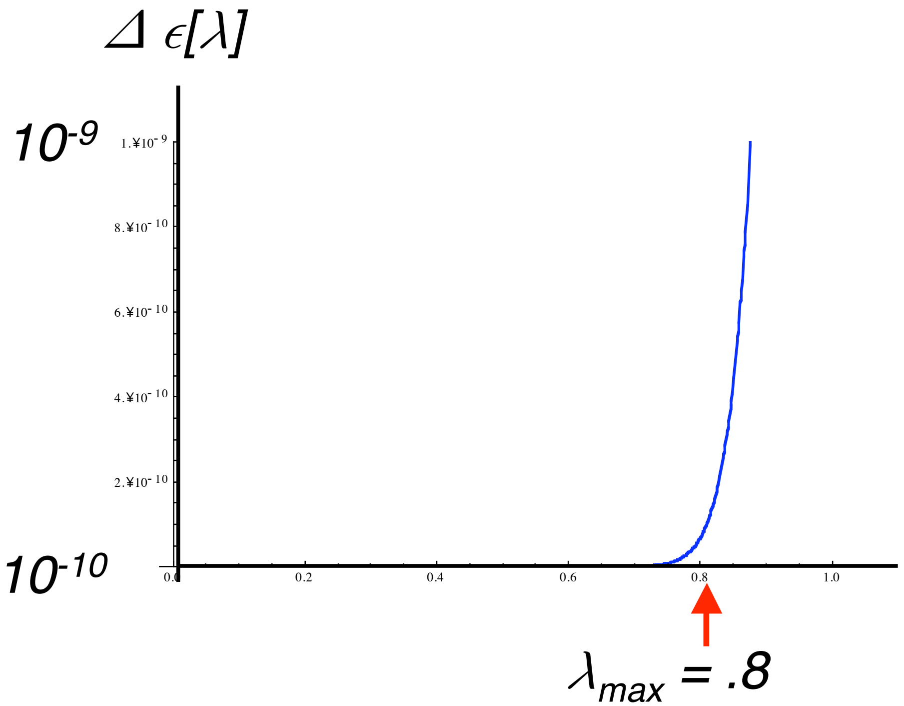
Insensitivity to Moebius scale α at fixed αL_s

Max Error at largest e.v. of $H = \gamma_5 D_{wil}[M_5]/(2 + D_{wil}[M_5])$

$$|(\epsilon_{L_s}[H] - \epsilon_{L'_s,\alpha}[H])\phi_\lambda| \leq |\epsilon_{L_s}[\lambda_{max}] - \epsilon_{L'_s,\alpha}[\lambda_{max}]|$$

Example $\alpha = L_s/L'_s = 2$





New Challenges for SciDAC-x† (?)

- Many-core: “flops” are almost free!
  - SU(3) manifold is S_3 x S_5 , so read 8 reals and re-compute 18 floats (see Bunk and Sommer, 1982) 16 bit mixed precisions works beautifully. (Mike Clark’s code)
  - Reduce MPI traffic for Multi-GPU using domain decomposition (see Luescher’s multi-level Schwartz algorithms)
- Algorithmic & Data complexity:
  - Modify API for multiple grids and intra-grid data transfers
  - Include general Gauge and Fermion Reps
  - Rapid prototyping and shared components
  - Share and reuse of eigenvectors and preconditioners
- SciDAC “Requirement”!
  - Develop collaboration with other SAP and Center! (PETSc ?)
  - Utilize Out Reach Center (Software Mirror ?)
  - Publish Software Methodology

† Michael R. Strayer

EXTRAS

SciDAC-2 Tutorials

- A. Pochinski and J. Osborn, "Data Parallel Software for Lattice QCD", SciDAC tutorial, MIT (2007).
- B.Joo,Lecture course at Institute of Nuclear Theory Summer School on Lattice QCD 2007, http://www.int.washington.edu/talks/WorkShops/int_07_2b)
- C. DeTar, "MILC with SciDAC C", HackLatt tutorial presentation, Edinburgh EPCC (2008).
- Tutorials at the HackLatt'XX series of workshops, held annually in Edinburgh, UK:
  - B. Joo, 2005, 2006, 2008
  - David Richards. 2007
- B. Joo, Repeat of 2006 tutorial: Trinity College, Dublin, Dec 2006

SciDAC-2 Presentations

- A. Bazavov (MILC Collaboration) "Upcoming Large-Scale Simulations of Highly Improved Staggered Quarks in Lattice QCD", University of Cambridge, November 4, 2008; University of Glasgow (video conferenced to Edinburgh), November 7, 2008; University of Wales, Swansea, November 10, 2008.

- S. Gottlieb, "Gauge Force Speedup", Pathways to Blue Waters Workshop, NCSA, October 15--17, 2008.

- R. Brower, "Scalng, SciDAC API & Mutligrid vs multi-core -- search for a new paradigm", Pathways to Blue Waters Workshop, NCSA, October 15--17, 2008

- A. Bazavov (MILC Collaboration) "Upcoming Large-Scale Simulations of Highly Improved Staggered Quarks in Lattice QCD", University of Cambridge, November 4, 2008; University of Glasgow (video conferenced to Edinburgh), November 7, 2008; University of Wales, Swansea, November 10, 2008

- B. Joo, NCCS Oak Ridge Booth, SC 2008, Austin, TX and
  Poster at Falls Creek Workshop, Tennessee, 2008 and
  ORNL Users Meeting, Oak Ridge, April 2008

Publications, Reports, Proceedings

- S. Bazavov [MILC Collaboration], "HISQ action in dynamical simulations", PoS (LAT2008) (2008).
- Robert J Fowler, Todd Gamblin, Allan K Porterfield, Patrick Dreher, Song Huang, and Balint Joo. Performance engineering challenges: the view from RENCI. *J. Phys: Conf. Ser*, page 5pp, October 2008.
- Y. Zhang, R. Fowler, K. Huck, A. Malony, A. Porterfield, D. Reed, S. Shende, V. Taylor, and X. Wu. US QCD computational performance studies with PERI. *J. Phys: Conf. Ser*, 78(012083):5pp, August 2007.
- Jie Chen and William Watson III, "Software Barrier Performance on Dual Quad Core Opterons", Proceedings of NAS'08, International Conference on Networking, Architecture and Storage, 2008, 303-309. IEEE Digital Object Identifier 10.1109/NAS.2008.27
- Jie Chen, William Watson III, and Weizhen Mao, "Multi-Threading Performanc on Commodity Multi-Core Processors", proceedings of 9th International Conference on High Performance Computing in Asia Pacific Region (HPC-Asia 2007

- Yunlian Jiang, Xipeng Shen, Jie Chen, and Rahul Tripathi, “ Analysis and Approximation of Optimal Co-Scheduling on Chip Multiprocessors”, In the Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques (PACT), Oct.,2008.

- R. Dutton, W. Mao, J. Chen, and W. Watson, III, Parallel Job Scheduling with Overhead: A Benchmark Study, Proceedings of the IEEE International Conference on Networks, Architecture, and Storage (NAS), 326-333, 2008

- M. DiPierro, ``QCD Visualization tookit'' presented at the XXV International Symposium Lattice Field Theory in Regensburg, Germany (2007); at the 4th High End Visualization Workshop in Obergurgl, Tyrol, Austria.

- S.~Bazavov [MILC Collaboration], "HISQ action in dynamicalsimulations", PoS LAT2008 (to be published, 2008)
- S.~Basak [MILC Collaboration], "Electromagnetic splittingsof hadrons from improved staggered quarks in full QCD.", PoS LAT2008 127 (to be published 2008).

- K. Barros, R. Babich, R. Brower, M. Clark and C. Rebbi,``Blasting through lattice calculations using CUDA" PoS(LATTICE2008) 045, arXiv:0810.5365

- J. Brannick, R. C. Brower, M. A. Clark, S. F. McCormick, T. A. Manteuffel, J. C. Osborn and C. Rebbi ``The removal of critical slowing down" PoS (LATTICE2008), arXiv:0811.4331

- B. Joo, "Continuing Progress on a Lattice QCD Software Infrastructure", Poster at SciDAC 2008, J. Phys. Conf. Ser. 125:012066, 2008
- B. Joo, "SciDAC-2 Software Infrastructure for Lattice QCD", Poster at SciDAC 2007, J. Phys. Conf. Ser. 78:012034, 2007

- R. Edwards, B. Joo, "The Chroma Software System for Lattice QCD", Nucl. Phys. Proc. Suppl. 140:832, 2005

- P. Coddington, B. Joo, C. M. Maynard, D. Pleiter, T. Yoshie, "Marking Up Lattice QCD Configurations and ensembles", PoS, Lat 2007:84,2007

- W. Watson, B. Joo "ILDG Middleware Working Group Status Report", Nucl. Phys. Proc. Suppl. 140:209-212, 2005

- R. G. Edwards, B. Joo, A. D. Kennedy, K. Orginos, U. Wenger, “Comparison of Chiral Fermion Methods”, PoS LAT2005 (2005), 146
- A. Pochinsky, “The Blue Gene, GCC and lattice QCD: A case study”, J.Phys.Conf.Ser.46:157-160,2006.
- A. Pochinsky, “Domain wall fermion inverter on Pentium 4”, Nucl.Phys.Proc.Suppl.140:859-861,2005.
- A. Pochinsky ”Large scale commodity clusters for lattice QCD, Nucl.Phys.Proc.Suppl. 119:1044-1046,2003.
- A. V. Pochinsky, “Conjugate Gradient for Domain Wall Fermions with 4-d EO Preconditioning”, (2004) http://www.mit.edu/~avp/sse/1.3.3/dwf.pdf
- A. V. Pochinsky, “GigE and Xeon”, (2002) http://www.mit.edu/~avp/lqcd/GigE/report.pdf
- A. V. Pochinsky, “Blue Gene Vector Extensions for GCC”, (2004), http://web.mit.edu/bgl/software/gcc-dh.pdf

- A. Dubey, G. Karsai and S. Abdelwahed, “Compensating for Timing Jitter in Computing Systems with General-Pur Operating Systems”, ISORC (2009).

- A. Dubey, S. Neema, J. Kowalkowski and A. Singh, “Scientific Computing Autonomic Reliability Framework”, eScience (2008).

- A.Dubey, S. Nordstrom, T. Keskinpala, S. Neema,T. Bapty and G. Karsai, “Towards A Model-Based Autonomic Reliability Framework for Computing Clusters” EASE '08 (2008) p 75--85.

- A. Dubey, S. Nordstrom, T. Keskinpala, S. Neema, T.Bapty and G. Karsai, “Towards a verifiable real-time, autonomic, fault mitigation framework for large scale real-time systems”, Innovations in Systems and Software Engineering (2007}) p. 33--52.

- A. Dubey, L, Piccoli, J. B. Kowalkowski, J. N. Simone, Xian-He Sun, G. Karsai and S. Neema, “Using Runtime Verification to Design a Reliable Execution Framework for Scientific Workflows”, EASE '09 (2009).

- S. Nordstrom, A. Dubey, T. Keskinpala, R. Datta, S. Neema and T. Bapty, “ModelPredictive Analysis for Autonomic Workflow Management in Large-scale Scientific Computing Environments”, EASE '07 (2007), pp. 37--42.

- L. Piccoli, X-H. Sun, J. Simone, et. al, “The LQCD Workflow Experience: What We Have Learned” SuperComputing 2007 (2007).

- L. Piccoli, J. Simone, J. Kowalkowski, et. al., “Tracking LQCD Workflows”, Lattice 2008 (2008).

Motivation

- *Algorithms for lighter mass fermions and larger lattice*
  - *The Dirac solver: D $\not\psi$ = b becomes increasingly singular*
  - *“split” vector into near null space D $\mathcal{S} \simeq$ 0 & Complement $\mathcal{S}_\perp$*

- *Basic idea (as always) is Schur decomposition!*
  - *(e = near null, o = complement)*

Schur:

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

Implies

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

with $D_{Schur} \equiv D_{oo} - D_{oe} D_{ee}^{-1} D_{eo}$

3 Approaches to separating out near null space

1. “Deflation”: N_ν exact eigenvector projection

$$D_{smooth} = \widetilde{\psi}^*_{\lambda,x} D_{x,y}\ \psi_{\lambda',y}$$

2. “Inexact deflation” plus Schwarz (Luscher)

$$D_{smooth} = P^\dagger D P$$

3. Multi-grid preconditioning

$$D_{Schur} = [D - DP\frac{1}{P^\dagger D P}P^\dagger D]$$

- 2 & 3 use the same splitting $\mathcal{S}$ and $\mathcal{S}_\perp$

Choosing the Restrictor ($R = P^\dagger$) and Prolongator (P)?

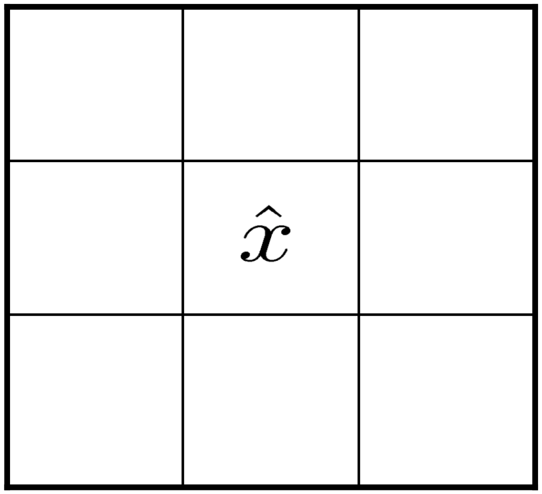
$$D_{yx}\psi_x^{(s)} \simeq 0 \quad s = 1, \cdots, N_\nu$$

- *Relax from random to find near null vectors*
- *Cut up on sublattice (No. of blocks: $N_B = 2\ L^4/4^4$)*

$$P_{\hat{x}s;x} = \begin{cases} \text{basis of } \psi_x^{(s)} & x \in B_{\hat{x}} \\ 0 \quad \text{otherwise} & \end{cases}$$

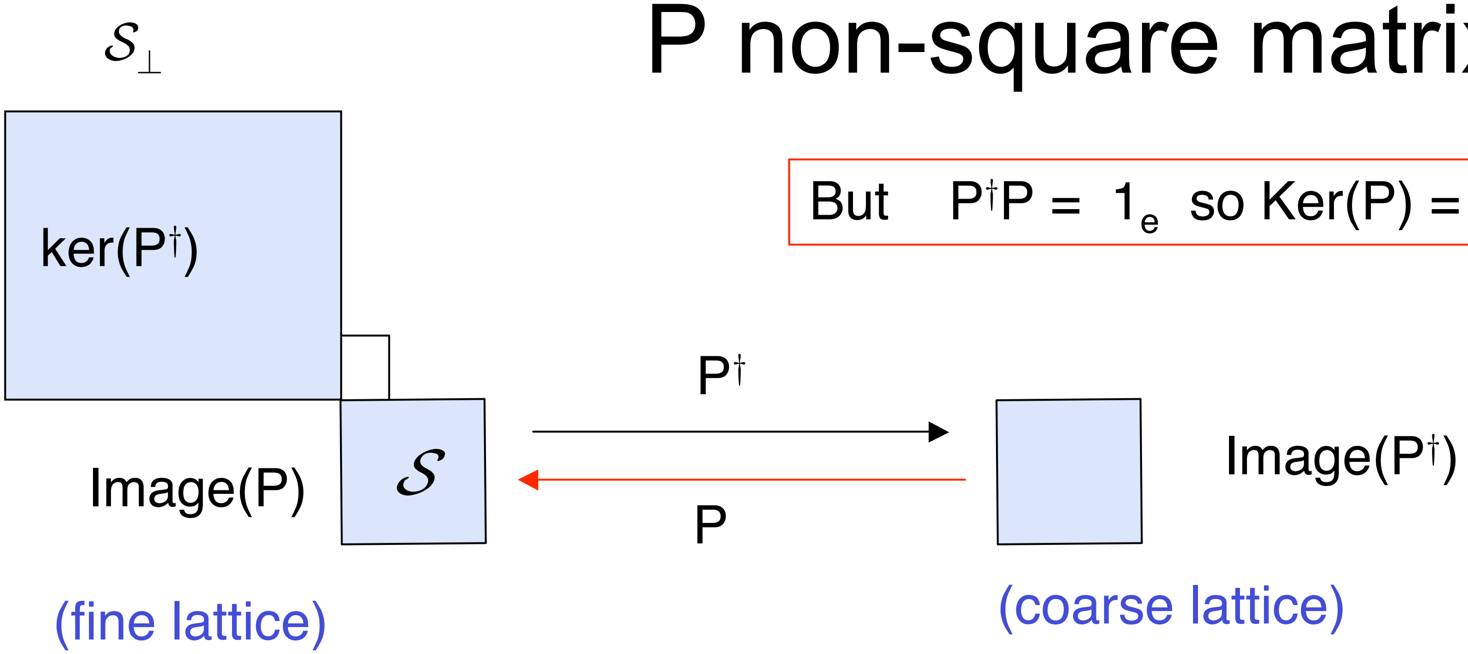
$\mathcal{S}$= Range(P)

dim($\mathcal{S}$)= $N_\nu\ N_B = 2N_\nu\ L^4/4^4$



P non-square matrix

But $P^\dagger P = 1_e$ so $\mathrm{Ker}(P) = 0$



$$P = \left| \begin{array}{ccc} \psi_1 & 0 & 0 \\ \psi_2 & 0 & 0 \\ \psi_3 & 0 & 0 \\ \psi_4 & 0 & 0 \\ 0 & \psi_5 & 0 \\ 0 & \psi_6 & 0 \\ 0 & \psi_7 & 0 \\ 0 & \psi_8 & 0 \end{array} \right|$$

Coarse approximation:

$$D_{ee} \equiv \hat{D} = P^\dagger D P$$

Multigrid Cycle (simplified)

- Smooth: $x' = (1 - A)\, x + b$
  - ➢ $r' = (1 - A)\, r$
- Project: $A_c = P^\dagger A P$ and $r_c = P^\dagger r$
- Solve: $A_c e_c = r_c$
  - ➢ $e = P A^{-1}{}_c P^\dagger r$
- Update: $x' = x + e$
  - ➢ $r' = b - D(x + e)$

    $= [\, 1 - D P (P^\dagger D P)^{-1} P^\dagger]\, r$

oblique projector

Note since $P^\dagger r' = 0$ exact deflation in $\mathcal{S}$

- Real algorithm has lots of tuning!
  - Multigrid is recursive to multi-levels.
  - Near null vectors $\psi_x^{(s)}$ are augmented by recursive using MG itself.
  - pre and pos-smoothing is done by Minimum Residual.
  - Entire cycle is used as preconditioner in CG.
  - γ_5 is preserved $[\gamma_5, P] = 0$

- Current benchmarks for Wilson-Dirac:
  - $V=16^3 \times 32$, β=6.0, $m_{crit} = 0.8049$,
  - Coarse lattice Block = $4^4 \times N_c \times 2$, $N_\nu = 20$.
  - 3 level V(2,2) MG cycle.
  - 1 CG application per 6 Dirac application
  - Note N_ν scales O(1) but deflation N_ν = O(V)