Lattice QCD Production on Commodity Clusters at Fermilab

Donald J. Holmgren (djholm@fnal.gov)

Fermi National Accelerator Laboratory

P.O. Box 500

Batavia, IL 60510–0500, USA

CHEP03 – March 2003

A. Singh[1], S. Gottlieb[2], P. Mackenzie[1], J. Simone[1]

[1] Fermilab

[2] Indiana University

http://lqcd.fnal.gov/chep03.pdf

Outline

- The SciDAC Lattice Gauge Computing Program
- Fermilab Lattice Clusters
- Single Node Performance
- MILC Cluster Performance
- Future Work

SciDAC Initiative

- U.S. Department of Energy is funding Lattice QCD via the SciDAC program (Scientific Discovery though Advanced Computing) - http://www.lqcd.org/
  - 3 years of support (FY2002 - FY2004)
  - Collaboration includes most U.S. lattice theorists
  - Funding primarily for software development, but also prototype clusters
- Strategy
  - Two computer hardware approaches: special purpose machines (QCDOC at Columbia/Brookhaven) and commodity clusters at Fermilab and Jefferson Lab
  - Create software infrastructure to allow legacy and new LQCD codes to run on both types of hardware
  - Anticipate best performance/price on QCDOC until 2004-2005, clusters afterwards
- Software
  - QMP - communications library, basically a small subset of MPI with less overhead, will run over QCDOC mesh, Myrinet GM, MPI
  - QLA - single node linear algebra, lattice aware, optimized for some architectures
  - QDP - lattice-wide computations
  - QIO - parallel file I/O
  - Optimized inverters
- Hardware
  - Investigations of gigabit ethernet mesh (Z. Fodor approach) and custom network

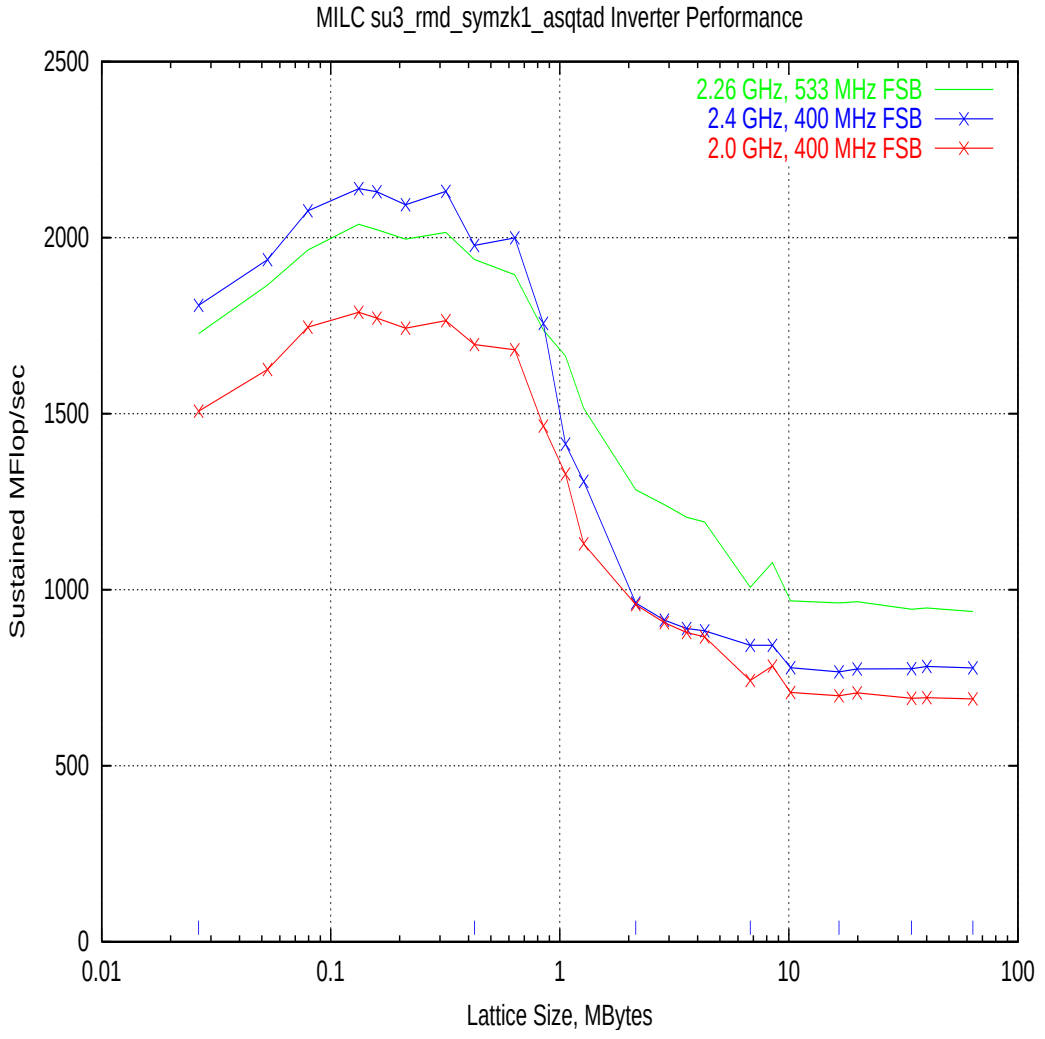
Fermilab Lattice Clusters

	Pentium III	Xeon #1	Xeon #2	Itanium2
Speed	700 MHz	2.0 GHz	2.4 GHz	900 MHz
# Nodes	80	48	128	8
# Processors	160	96	256	10
Memory	256 MB SDRAM	1 GB DDR200	1 GB DDR200	1 GB DDR266
Chipset	440GX	E7500	E7500	zx1
Myrinet	LANai-9 Copper	LANai-9 Fiber	LANai-9 Fiber	LANai-7 LAN
Other Network		GigE Mesh (16)		Dolphin SCI
Vendor	SGI (VA Linux)	SteelCloud	CSI	HP
Funding	DOE	SciDAC	SciDAC	SciDAC
Date in Service	Jan 2001	July 2002	Jan 2003	May 2003

- 256 processors hits the rsh barrier
  - run out of priviledged ports
  - switch to ssh or mpd
- Fiber Myrinet is much easier to wire
  - but, starting to see laser failures (0.5 per month)

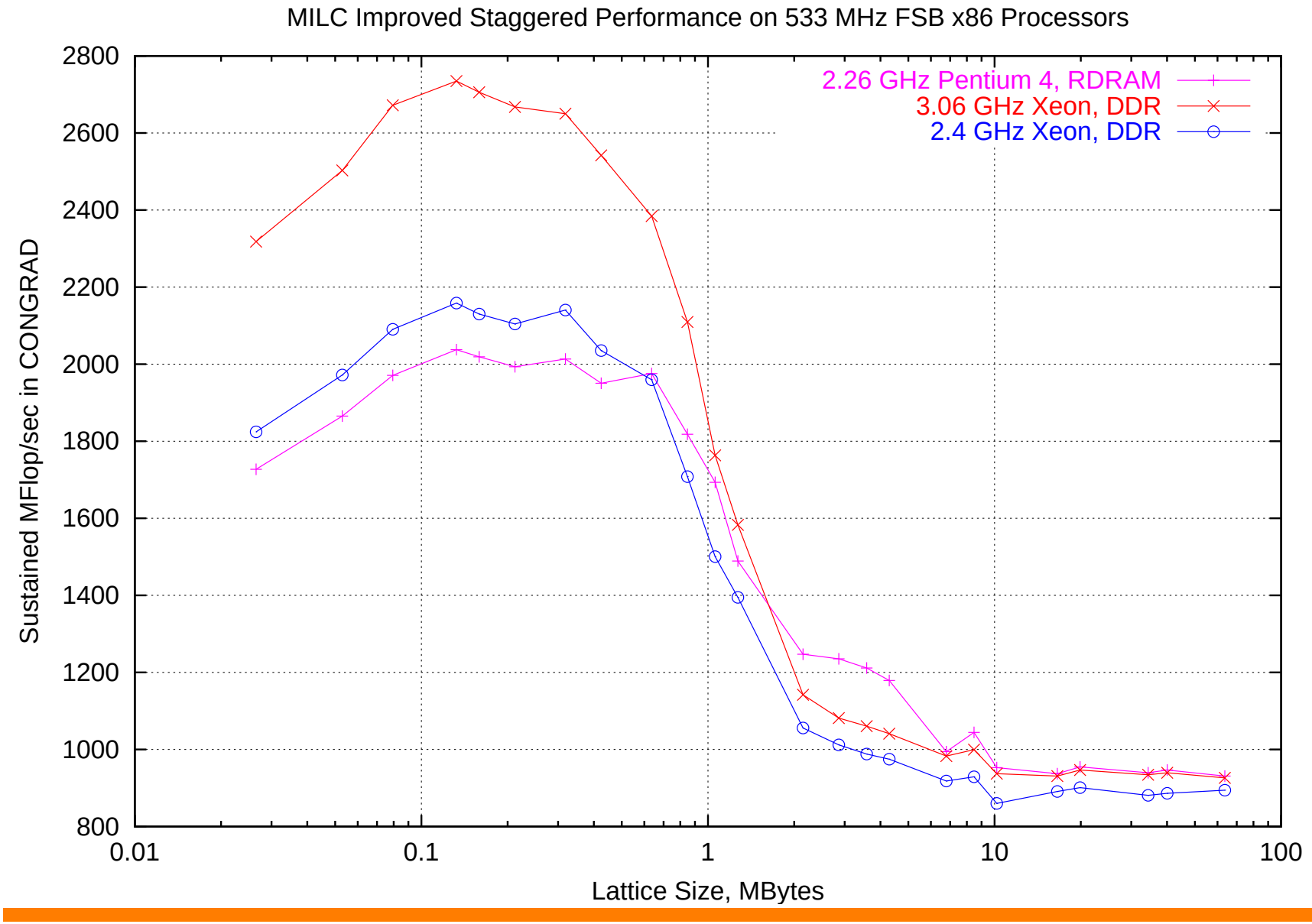
Single Node Performance

- MILC Improved Staggered Performance
  - Each site is 1656 bytes large
  - Blue ticks mark $(2,4,6,8,10,12,14)^4$
  - L2 cache (512K) near 4^4
  - FPU dominates for lattices smaller than 4^4
  - Memory bandwidth dominates for lattices larger than 4^4
  - 533 MHz FSB systems just now available with PCI-X



Single Node Performance

E7501 and i850E Chipsets



Single Node Performance

In-Cache Floating Point Performance

Processor	Matrix-Vector	Matrix-HWVector	Matrix-Matrix
700 MHz Pentium III	305	300	302
1.2 GHz Athlon	679	806	637
1.4 GHz / 400 MHz P4	622	627	661
2.0 GHz / 400 MHz Xeon	905	823	954
2.26 GHz / 533 MHz P4	1046	1053	1111

- Performance in MFlop/sec, single precision, using MILC “C” library
- MILC SU3 matrices and vectors:
  - Matrix = 3x3 complex
  - Vector = 3x1 complex
  - HWVector = 3X2 complex
- See http://lqcd.fnal.gov/qcdstream/

Single Node Performance

Floating Point Performance in Main Memory

Access Pattern	Matrix-Vector (MFlop/Sec)	Matrix-Matrix (MFlop/sec)
In Cache	905	954
Sequential	710 (1553)	815 (2590)
Strided	139 (540)	292 (1326)
Mapped	131 (483)	265 (1202)

- Measured on 2.0 GHz Xeon system (E7500, interleaved DDR) using qcdstreams
- Measured on a 900 MHz Itanium 2 system (interleaved DDR) using John Dupuis' optimized kernels
- Measured during loop over `i` with these patterns:
  - In Cache: `mat_vec(a[0], b[0], c[0])`
  - Sequential: `mat_vec(a[i], b[i], c[i])`
  - Strided: `mat_vec(a[i*s], b[i*s], c[i*s])`, s=stride constant
  - Mapped: `mat_vec(a[map[i]], b[map[i]], c[map[i]])`

Single Node Performance

Memory Bandwidth

Processor	Memory Type	Chipset	Bandwidth MB/sec
Pentium III	100 MHz SDRAM	440GX	330
Athlon	DDR200	760MP	700
Xeon	DDR200	GC-HE	935
	DDR200	E7500	1240
	DDR266	E7501	1506
	PC800 RDRAM	i860	1305
	(SSE assist)	i860	2121
Pentium 4	PC800 RDRAM	i850	1320
	PC1066 RDRAM	i850E	2035
Itanium 2	DDR266	zx1	2460

- Measured with the Streams benchmark
- Shows rate of copying *double* to *double* at 100% CPU utilization
- Values depend on *memory type, interleave, bus width*

  SSE assist = 128 bit loads/stores, cache bypass writes

Single Node Performance

Optimized SSE SU3 Matrix Algebra - In Cache Performance

Operation	"C" Cycles	SSE Cycles	MFlops/GHz
Matrix-Vector Multiply	124	57	1158
Matrix-Matrix Multiply	414	130	1523
Matrix-HWVector Multiply	268	73	1808

- In cache performance
- Results shown are for Pentium 4 (Xeon results are equivalent)
- See http://lqcd.fnal.gov/sse/
- Matrix-HWVector is M. Luescher's code
- Implemented via inline GCC assembler macros - subroutine calls cost 20-30 cycles

Single Node Performance

SSE SU3 Linear Algebra Performance

- Performance degradation out of cache:

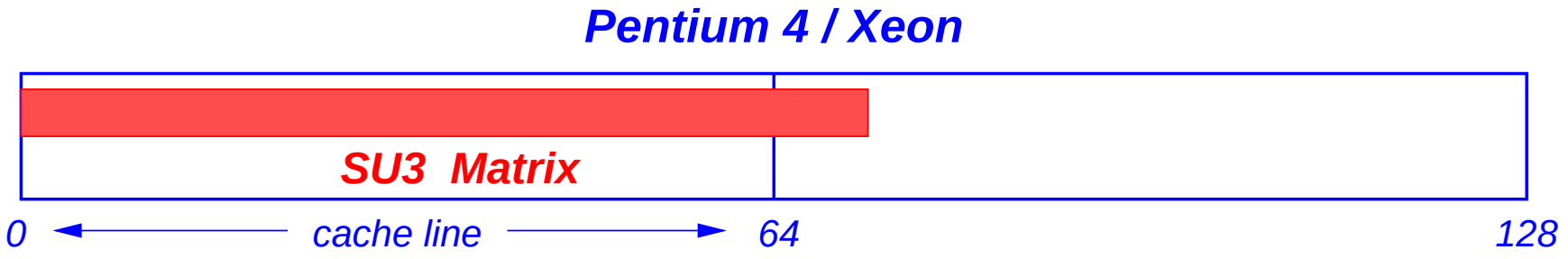
Code	Pattern	Matrix-Vector	Matrix-HWVector	Matrix-Matrix
"C"	In Cache	905	823	954
	Sequential	710	729	815
	Strided	139	212	292
	Mapped	131	196	265
SSE	In Cache	2124	3514	2985
	Sequential	846	1135	1310
	Strided	157	288	328
	Mapped	173	309	371

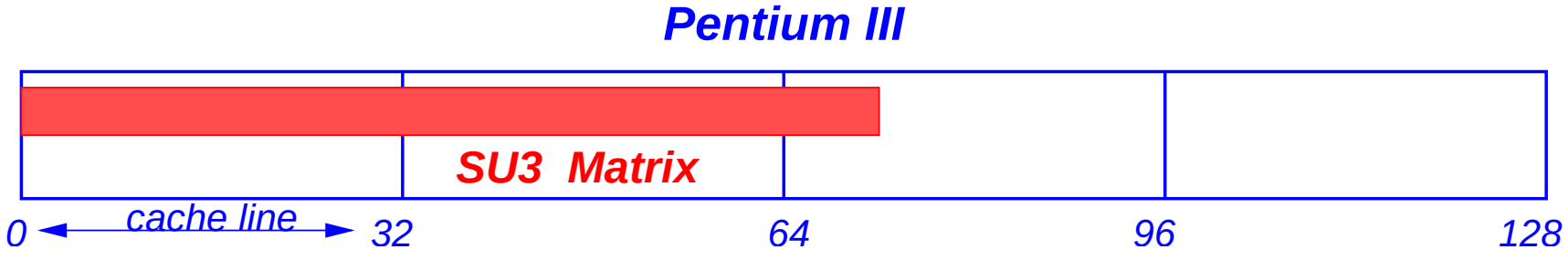
- Results shown are MFlops/sec on 2.0 GHz Xeon (E7500 interleaved DDR)
- Optimization is clearly constrained by memory bandwidth

Single Node Performance

Field Major Optimization

- Standard MILC lattice layout is site major
  - each lattice site has a number of fields (vectors, matrices, scalars)
  - stride between corresponding fields is large (1656 bytes on improved staggered)

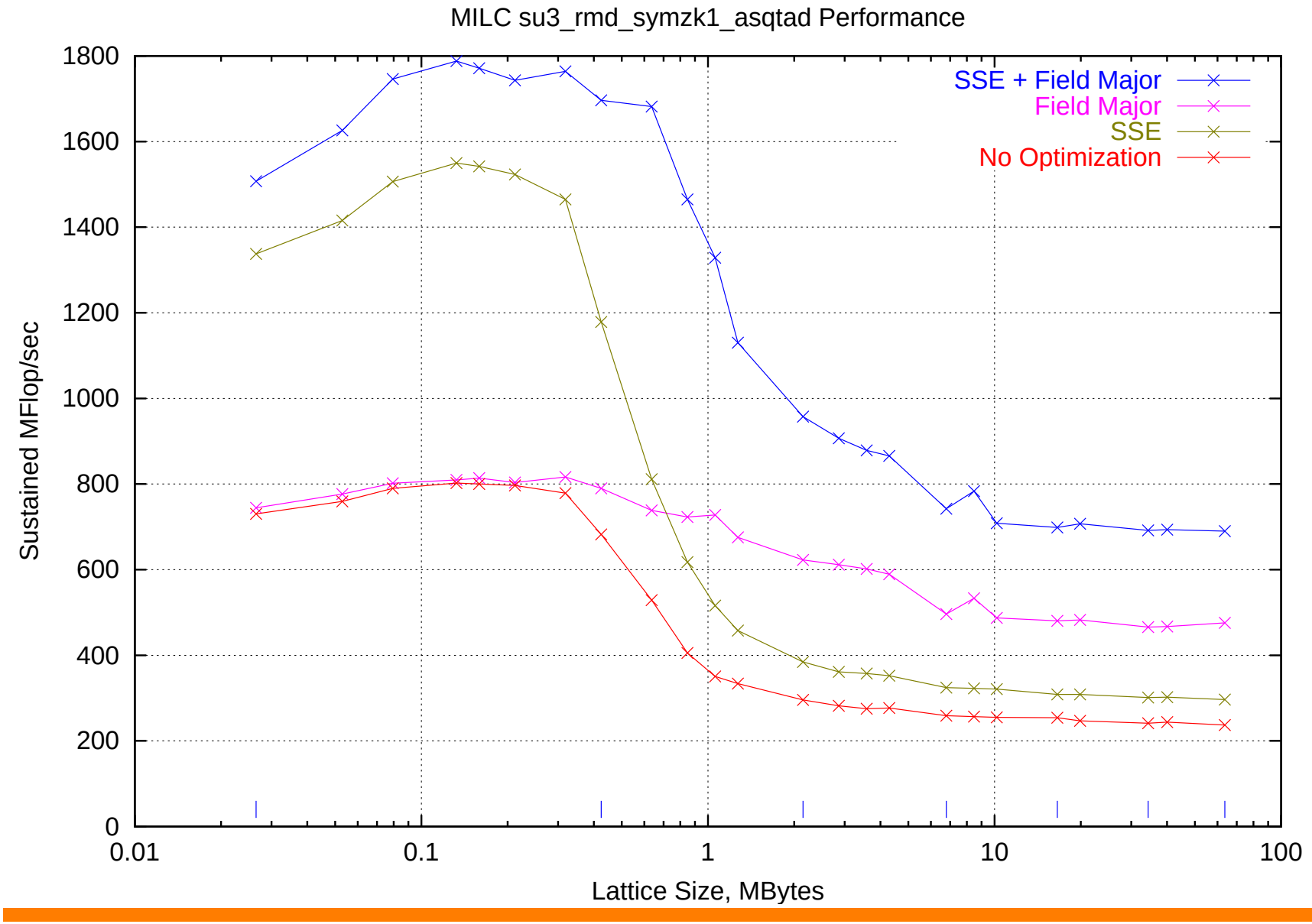




- SU3 matrix size = 72 bytes
  - Load efficiency on Pentium III = 75%
  - Load efficiency on Pentium 4 = 56%
- Field major layout boosts load efficiency
- MILC concept - Steve Gottlieb, MILC Implementation - Dick Foster

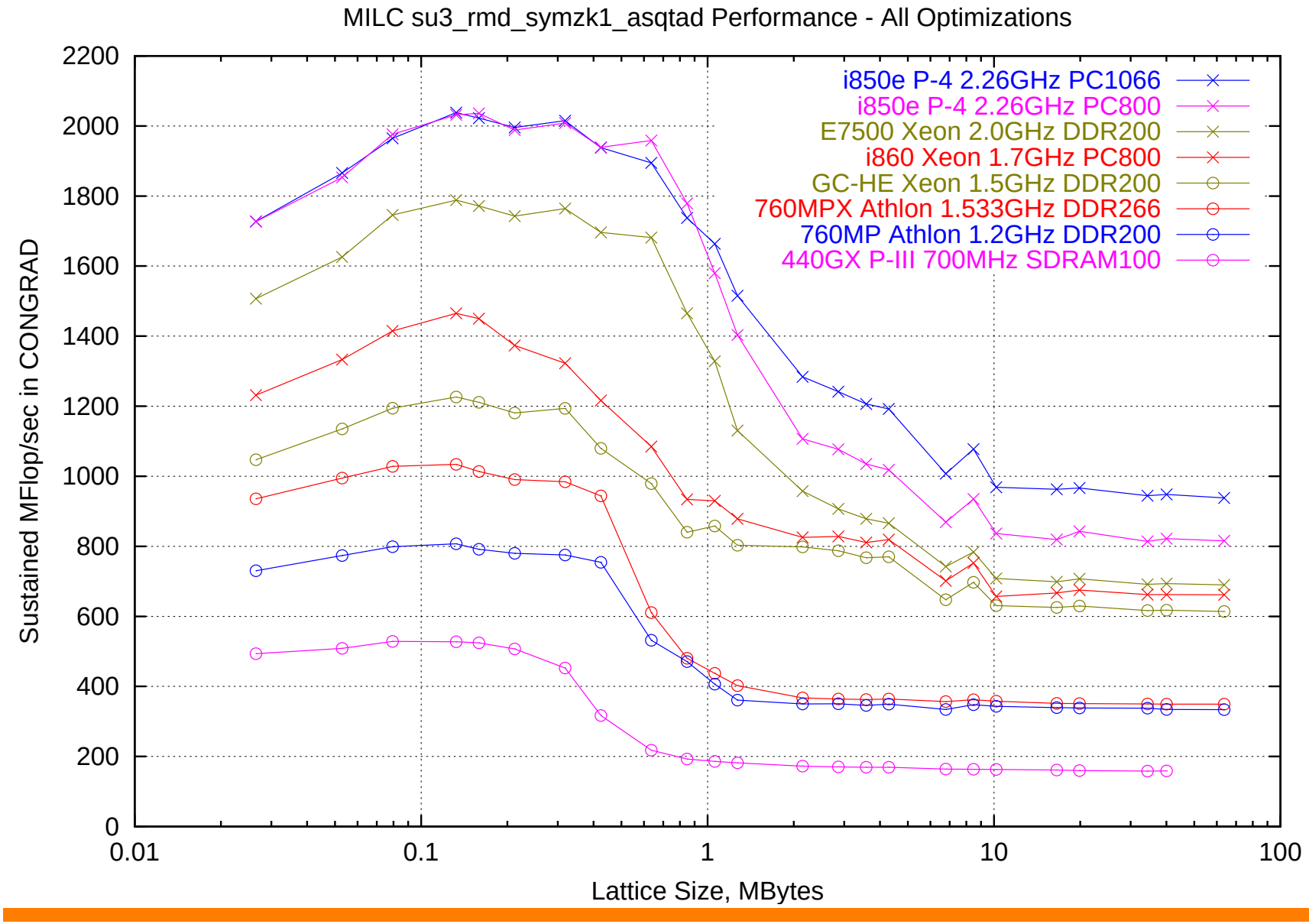
Single Node Performance

All Optimizations - 2.0 GHz Xeon, E7500



Single Node Performance

Survey of MILC Performance with All Optimizations



SMP Performance

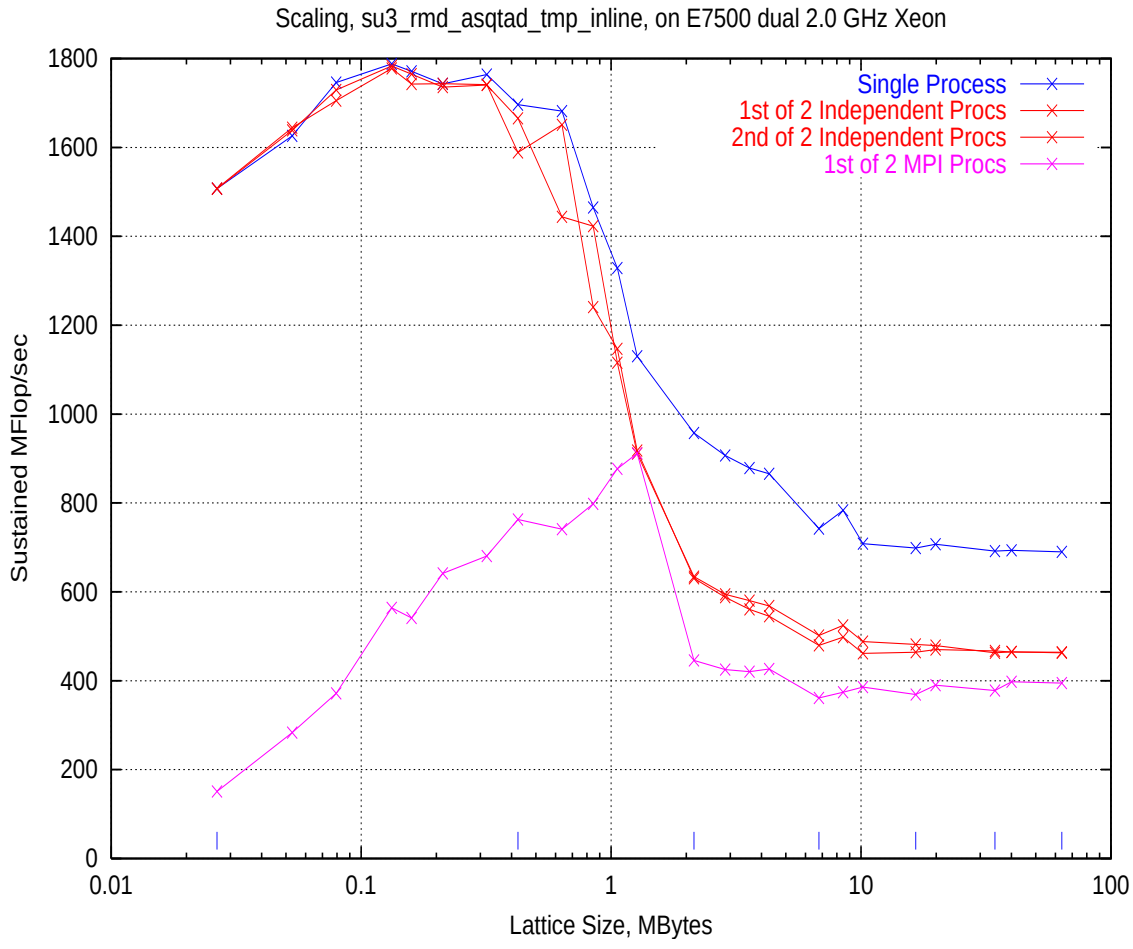
Why Consider SMP?

- If communication bandwidth sufficient, minimize cost of network interfaces
- Fast, wide PCI buses, and PCI-X
  - Single Pentium 4 mainboards have only narrow, slow PCI
- Incremental cost of second processor is low
  - If codes scale well, better performance/price
- Server class features
  - Hardware management - IPMI, BIOS redirect
  - Integrated video, network
- Minimize number of machines to manage

SMP Performance

SMP Performance on Dual Processor 2.0 GHz Xeon - E7500 Interleaved DDR

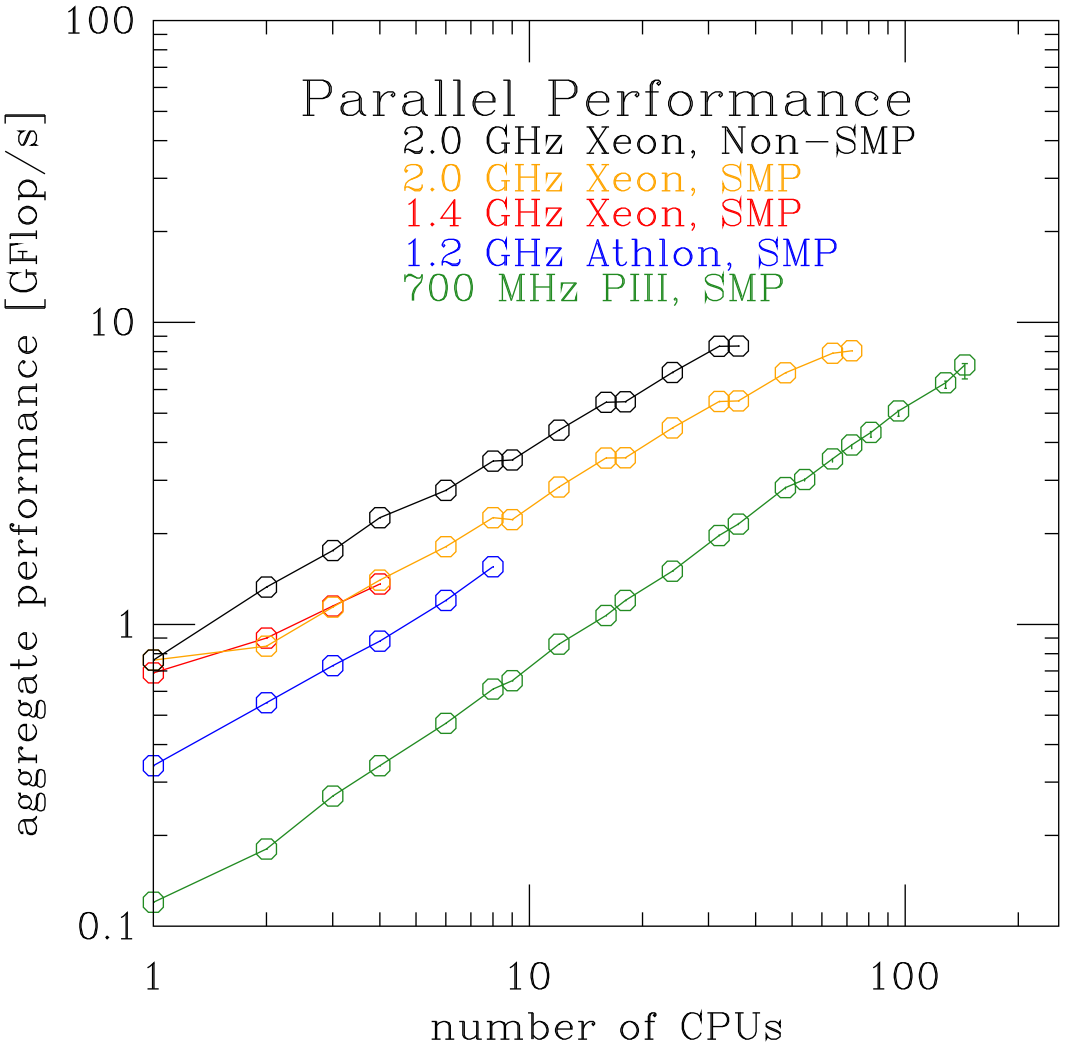
- Scaling on dual cpu machine:
  - 65% on independent processes
  - 55% on cooperative processes
- Not surprising since single processes are memory bandwith bound



Cluster Performance

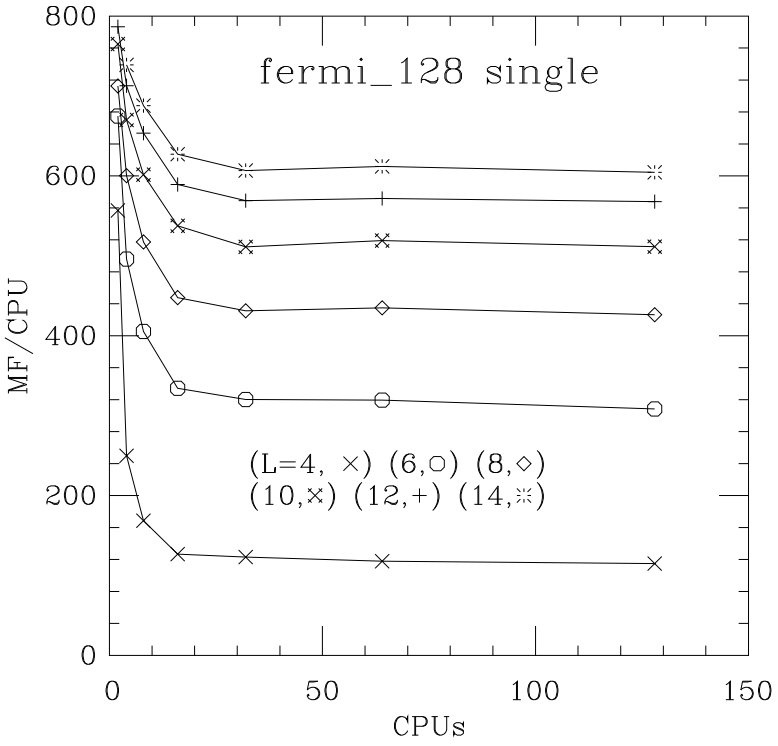
Scaling - Constant Volume per Run

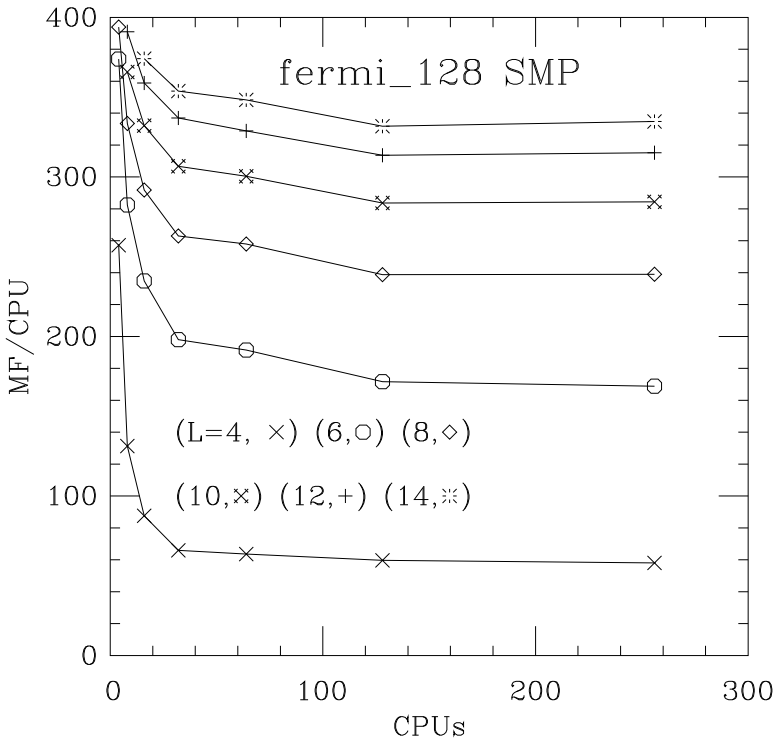
- Fixed total volume (12^3 X 24) is divided among nodes
- Graph shows aggregate performance
- Communications over Myrinet using *mpich-gm*



Cluster Performance

Scaling - Constant Volume per Processor





- Data from 2.4 GHz Xeon, E7500 chipset
- MILC runs with fixed lattice volume per node (L^4, L = 4,8,10,12,14)
- Number of communications directions increases with node count: For Non-SMP, 2 nodes = 1 direction, 4 nodes = 2 directions, 8 nodes = 3 directions, 16 nodes = 4 directions

Cluster Performance

PCI Performance of Common PIII/P4/Xeon Motherboards

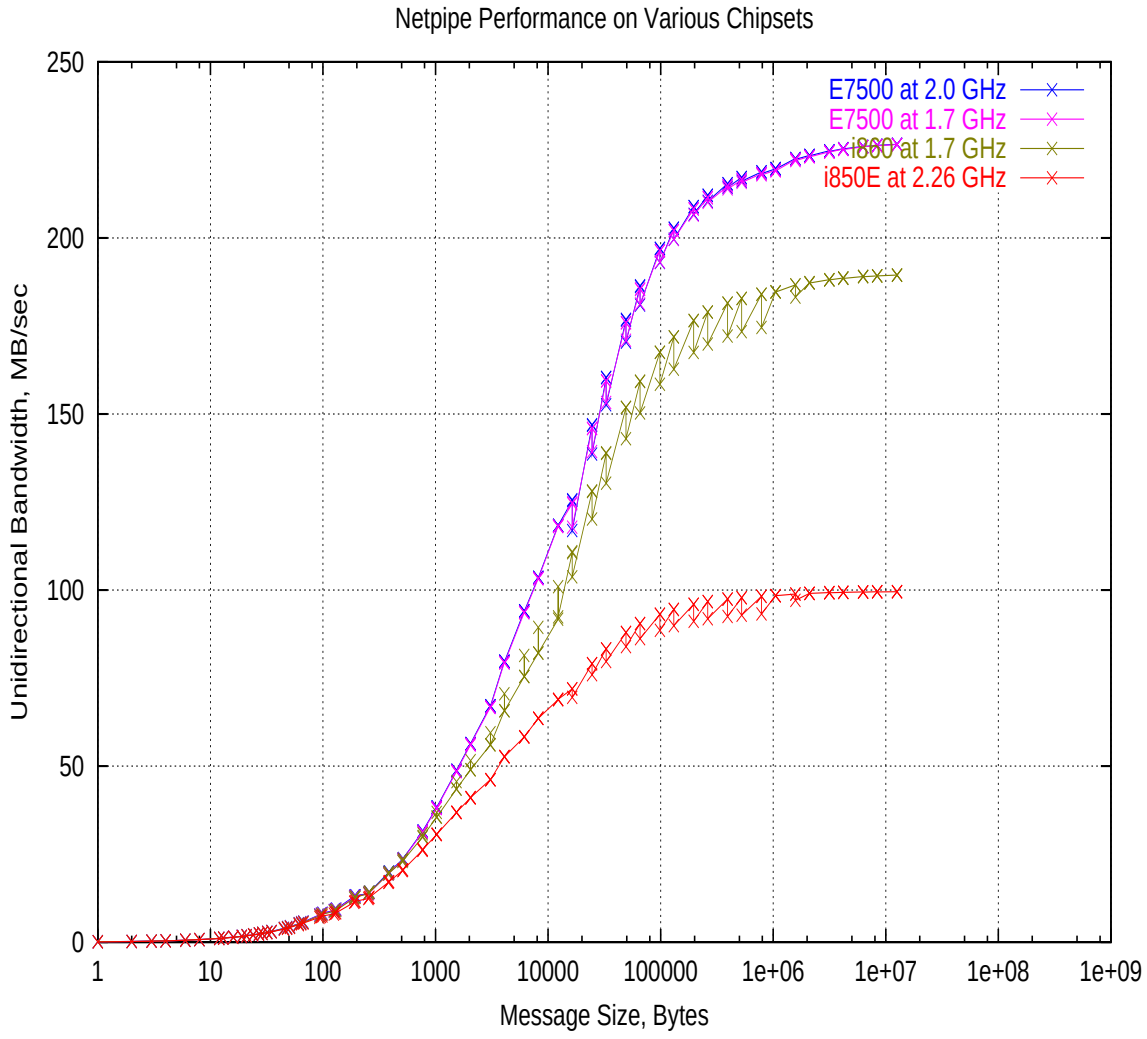
Processor	Chipset	Bus Read (MByte/sec)	Bus Write (MByte/sec)
2.26 GHz Pentium 4	i850E	100	128
700 MHz Pentium III	440GX	125	127
1.7 GHz Xeon	i860	219	294
2.0 GHz Xeon	E7500	423	476
1.7 GHz Xeon	E7500	422	477

- Shown are burst transfer rates between motherboard and NIC as measured by Myrinet GM driver
- Pentium III and Pentium 4 motherboards in table have 32-bit, 33 MHz PCI buses (theoretical maximum transfer rate 133 MB/sec)
- Pentium III motherboards with 64/66 PCI buses are available (theoretical maximum transfer rate 533 MB/sec)
- i860 motherboards with two 64/66 PCI slots must be tweaked to achieve values shown
- E7500 values in red were measured with CPU slowed to 1.7 GHz from normal 2.0 GHz
- Myrinet “wire rate” is 250 MBytes/sec - performance will be constrained on motherboards with PCI transfer rates below 250 MBytes/sec

Cluster Performance

Myrinet Performance - Netpipe

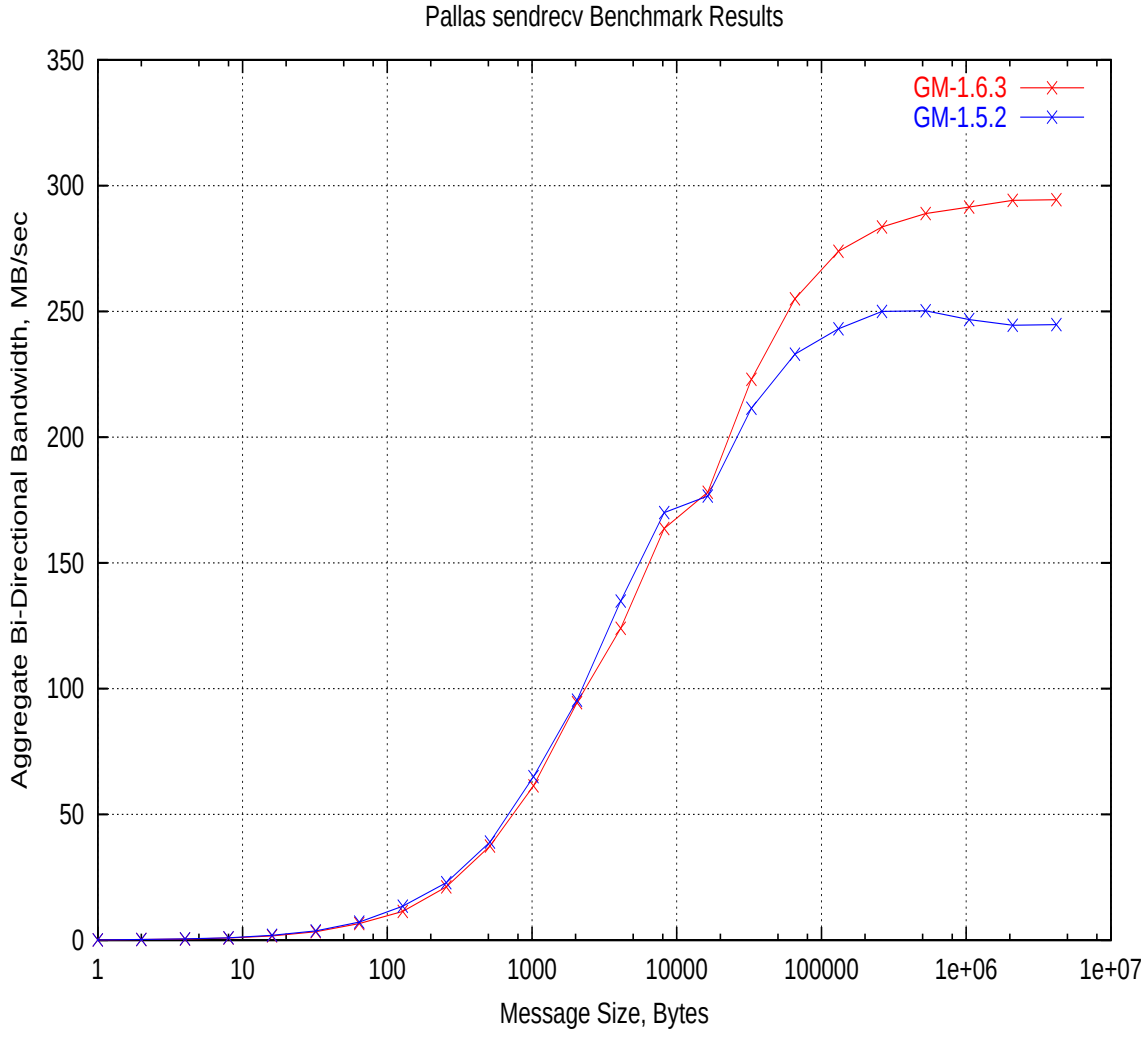
- MILC uses MPI for message passing
- The Netpipe benchmark shows “ping-pong” message exchange performance using MPI over GM
- Slower approach to maximum transfer rates, compared to “bare” GM
- E7500 2.0 GHz and E7500 1.7 GHz curves are on top of each other - no degradation in performance caused by slower CPU



Cluster Performance

Myrinet Bi-directional Performance - Pallas

- Pallas sendrecv benchmark measures aggregate bi-directional bandwidth
- Significant bi-directional performance improvement in GM-1.6.3



Cluster Performance

Estimating Performance of Slower Systems

- How important is achievable bandwidth?
- To investigate, use Fermilab 2.0 GHz Xeon DDR cluster.
  - To estimate other clusters, substitute one node
  - Frequent barrier sync's cause cluster to run at the speed of the slowest node
- Substitutions for 2.0 GHz E7500 node:

  E7500 at 1.7 GHz

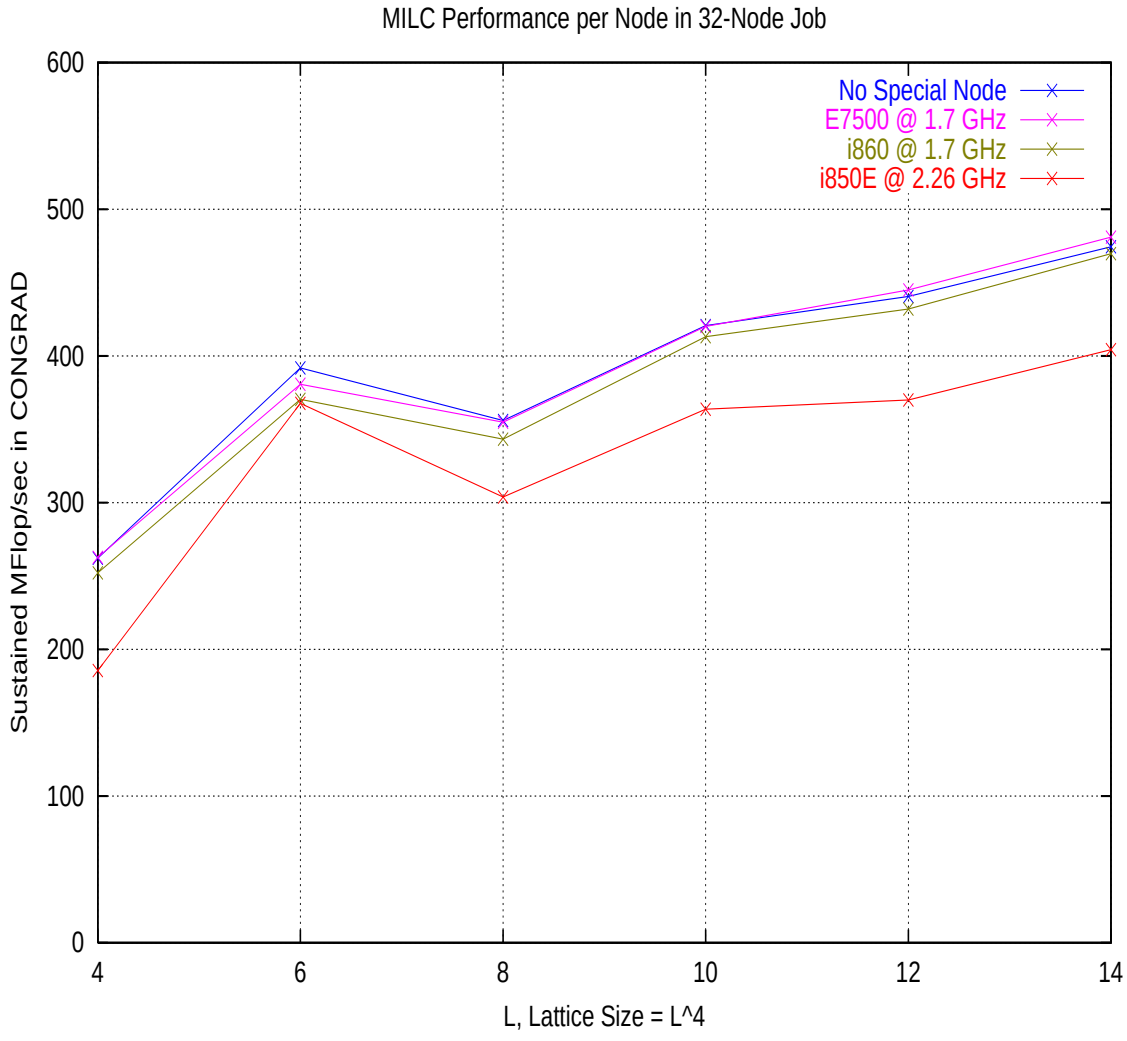
  i860 at 1.7 GHz (RDRAM, poor 64/66 PCI)

  i850E at 2.26 GHz (RDRAM at 533 MHz, poor 32/33 PCI)

Cluster Performance

Estimating Performance of Slower Systems - 32 Nodes

- Performance measured using constant volume per node
- Data shown for 32 nodes, non-SMP
- i860 shows little degradation compared to E7500
- Substantial degradation for i850E
- Each node communicates in 4 directions



Cluster Performance

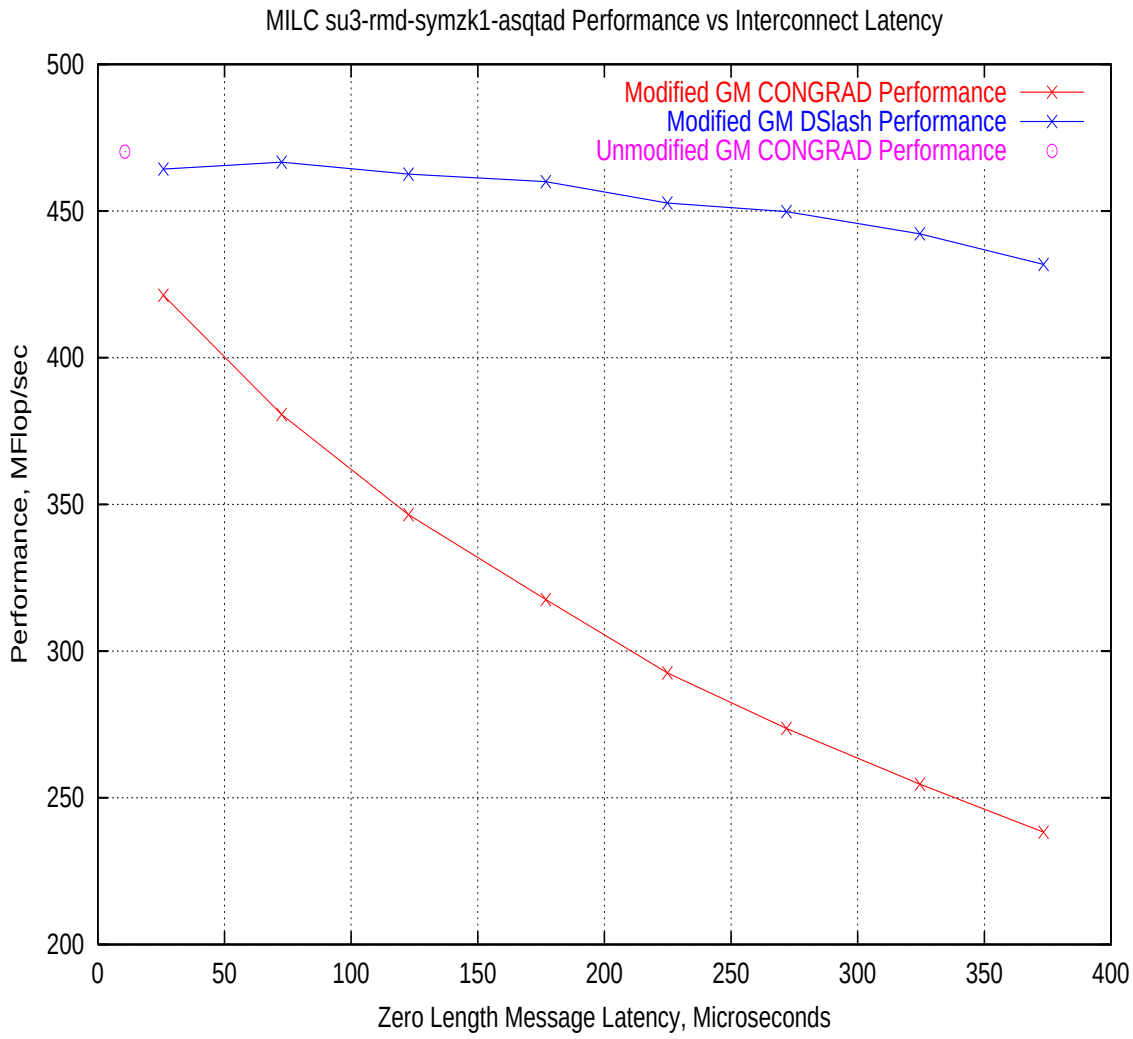
Modifying GM to Vary Latency

- How important is network latency?
- Pr. D.K. Panda at OSU has a QOS version of GM
  - Bandwidth throttling control per connection
  - Throttling works by injecting inter-packet delay in network interface
  - Myrinet uses 4K packets
  - All delay is in network hardware, with no effect on host CPU
- Fermilab modifications
  - Extra delay before first packet to control latency
  - Interface from MPICH-GM

Cluster Performance

MILC Sensitivity to Latency

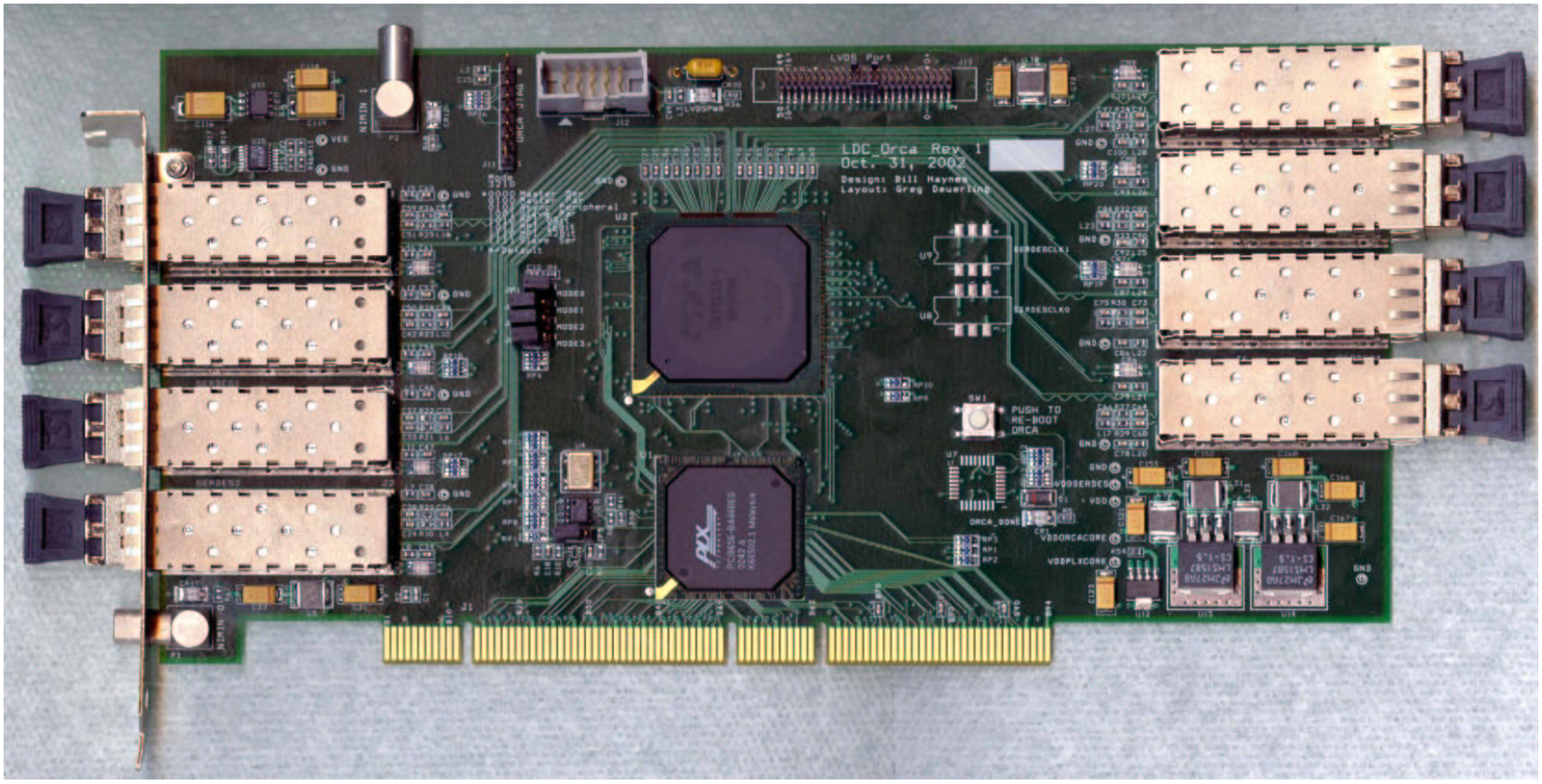
- Data for 16 nodes, 12^4
- Single point is MILC performance with unmodified GM
- CONGRAD performance decreases rapidly with increasing latency
- D-slash (CONGRAD without global sums) is not very sensitive to latency



Prototype Routed Mesh Network

- Motives
  - High performance, switched networks (Myrinet, Quadrics, SCI) have good bandwidth, low latencies, mature software, and high prices
  - Switched gigabit ethernet suffers from lower bandwidth, higher latencies, limited switch sizes, immature software, but low prices
  - Gigabit ethernet meshes are very cheap and have good latencies, but immature software, poor non-nearest-neighbor performance, and rigid configurations
- Weapons
  - FPGA's with multiple high speed serial links are now available
  - Some FPGA's will also have multiple PowerPC CPU's aboard
- Opportunities
  - Fermilab is already building prototype PCI cards with these FPGA's for data acquisition
  - Simple nearest-neighbor mesh appears straightforward - with more complex firmware, routing in the network is possible

FPGA-based NIC



- 8 bidirectional fiber or copper 2 Gbps links (reconfigurable)
- fast/wide PCI interface (PCI-X in next generation)
- long-term goal is to build 4-D or higher mesh with routing
- FPGAs with CPUs could allow sums, reductions to be done by the network

The Next Fermilab Cluster

- Next FNAL cluster purchase will be late Summer 2003
- Depending upon funding, as many as 256 nodes
- Possible architectures:
  - 533 MHz FSB dual Xeon unless something better
  - 800 MHz FSB single P4, only if PCI-X
    - Prescott has new SSE instructions for complex arithmetic
  - Itanium2 if software development is tenable
  - AMD Hammer/Clawhammer/Sledgehammer if AMD delivers
  - PPC970 is a potential wildcard
- Possible networks:
  - GM over Myrinet
  - GigE over Myrinet
  - GigE mesh