PC Clusters for Lattice QCD

Don Holmgren
djholm@fnal.gov
Fermilab

Consortial International Workshop on Computational Physics 2004, NCHC, Taiwan

December 2, 2004
Outline

• Introduction to lattice QCD
• The US program
  – SciDAC software and hardware
  – Post-SciDAC plans
• Cluster requirements
• Predictions
**Introduction to Lattice QCD**

- **Lattice QCD** is the numerical simulation of QCD
  - The QCD action, which expresses the strong interaction between quarks mediated by gluons:
    \[
    S_{\text{Dirac}} = \overline{\psi} (i \mathcal{D} + m) \psi
    \]
    where the *Dirac operator* ("dslash") is given by
    \[
    \mathcal{D} \psi = \sum_{\mu} \gamma_{\mu} (\partial_{\mu} + igA_{\mu} (x)) \psi (x)
    \]
  - **Lattice QCD** uses discretized space and time
  - A very simple discretized form of the *Dirac operator* is
    \[
    \mathcal{D} \psi (x) = \frac{1}{2a} \sum_{\mu} \gamma_{\mu} [U_{\mu} (x) \psi (x + a\hat{\mu}) - U_{\mu}^{\dagger} (x - a\hat{\mu}) \psi (x - a\hat{\mu})]
    \]
    where \( a \) is the lattice spacing
  - Other forms (e.g., higher order in \( a \)) lead to alternate discrete quark *actions*
• A quark, $\psi(x)$, depends upon $\psi(x + a\mu)$ and the local gluon fields $U_\mu$
  – $\psi(x)$ is complex 3x1 vector, and the $U_\mu$ are complex 3x3 matrices. Interactions are computed via matrix algebra
  – On a supercomputer, the space-time lattice is distributed across all of the nodes
• Sparse matrix techniques, such as conjugate gradient, are used to invert the Dirac operator

• MILC (MIMD Lattice Computation) is one of many lattice QCD codes: http://media4.physics.indiana.edu/~sg/milc.html
  – MILC runs on many types of platforms
  – most common form now uses MPI to run on clusters
  – used for performance measurements in this talk

• Lattice codes on parallel machines require:
  – strong floating point
  – high memory bandwidth
  – excellent network latency and bandwidth

• Physics programs require terascale facilities
  – major programs of study need 1 to 10 TFlop-years
The US Program in Lattice QCD

  - Original proposal submitted by most of the US lattice theorists (66, including machine builders)
  - Hardware sites:
    - Brookhaven National Laboratory (QCDoc)
      - Hardware funded separately
    - Jefferson Lab, Fermilab (clusters)
      - Prototype clusters: Myrinet, gigE mesh, Infiniband
- Goals:
  - Implement software infrastructure which allows implementation of physics codes which will run on QCDoc and clusters
  - Prototype clusters, optimizing price/performance
  - Establish common user environments which are independent of hardware
- ~ $2M/year, 75% for software (9 people)
“QCD on a Chip”  (http://phys.columbia.edu/~cqft/)
• Designed by Columbia University with IBM
  – in many ways, a precursor to Blue Gene/L
• Based on special Power PC core:
  – ~ 500 MHz G4 with 1 GFlops double precision
  – 4 MB of embedded DRAM
  – 12 bidirectional, 1 Gbit/sec serial links
  – 2.6 GB/sec external memory interface
  – fast ethernet
• Architecture:
  – PPC cores connected in 6-D torus
  – up to 20K processors
  – up to 50% of peak (~ 5 TFlops from 10K cpus)
QCDOC
Daughter Card with Two Processors
SciDAC Prototype Clusters

- **Jefferson Lab** ([http://lqcd.jlab.org](http://lqcd.jlab.org))
  - 128 node single 2.0 GHz Xeon, Myrinet (9/2002)
  - 256 node single 2.66 GHz Xeon, 3-D gigE mesh (9/2003)
    - Pr. Fodor's gigE mesh machines have had a huge impact on recent designs
  - 384 NODE SINGLE 2.8 GHz Xeon, 4-D gigE mesh (now)
    - $1700/node including networking

- **Fermilab** ([http://lqcd.fnal.gov](http://lqcd.fnal.gov))
  - 48 dual 2.0 GHz Xeon, Myrinet (8/2002)
  - 128 dual 2.4 GHz Xeon, Myrinet (1/2003)
  - 32 dual 2.0 GHz Xeon, Infiniband (7/2004)
  - 128 single 2.8 GHz P4, Myrinet (7/2004)
    - $900/node, reused Myrinet from 2000
  - 260 single 3.2 GHz P4, Infiniband (2/2005)
    - ~ $900/node + Infiniband ($880/node)
Jefferson Lab gigE Cluster
Fermilab Myrinet Cluster
SciDAC Software Stack

- **QMP** – communications (~ MPI subset)
- **QLA/QLA++** – optimized linear algebra
  - Intel/AMD, IBM PPC implementations
  - Site and vector operations
- **QDP/QDP++** - data parallel operations
  - High level of expression
  - Lattice wide operations
- **QIO/QIO++**
  - API for loading and storing data
  - Direct support for International Lattice Data Grid standards
    - “QCDml” (formalized metadata)
    - Binary interchangeable file format
    - Presentations at Lattice'04:
      [http://lqcd.fnal.gov/lattice04/ildg.html](http://lqcd.fnal.gov/lattice04/ildg.html)
Hardware Plans

- **QCDOC**
  - US 5 Tflop machine in production March 2005
  - 5 Tflop machines earlier for UKQCD, Japan
- **Fermilab**
  - Expansion to 520-node Infiniband cluster by Summer 2005
- **2006-2008 (DOE HEP program)**
  - ~ $2M/year for hardware and operations
  - New ~1024 node cluster each year
    - Replace 1/3\textsuperscript{rd} of hardware every year
  - Looking for similar funding from DOE Nuclear program for hardware at Jefferson Lab
Allocations

• Machine time at the two (three) sites is (will be) allocated to the community via formal proposal process each six months
  – JLAB
    • Mostly domain wall fermion action (SZIN or CHROMA codes)
    • Emphasis on large problems spanning entire cluster
      – gigE mesh machines are ideal for this
  – Fermilab – weak decays, heavy quark physics
    • Mostly MILC improved staggered action
    • Emphasis on many simultaneous small jobs of various sizes
      – demands flexibility of switched networks rather than meshes
    • Emphasis on parallel file I/O and mass storage interfaces
    • Most jobs are physics analysis using gauge configurations generated on bigger machines, though some coarse configuration generation is in progress
  – QCDOC
    • Primary use will be generation of gauge configurations
Collaboration Goals

- Common user environment
- Single sign-on to access all three facilities
- Facile transfers of data between sites and national mass storage facilities
- Computational grid?
  - is this really needed for ~ 3 very heterogeneous machines?
  - data access and movement will greatly benefit from grid developments, however
    - access data by physics parameters, not by filenames
Designing Clusters
Boundary Conditions

- Simplifications to make the talk fit the time:
  - I will only discuss in detail Intel processors
    - AMD, G5 will be mentioned
    - I'm happy to discuss other processors at the break
  - For other processor and network results, see http://lqcd.fnal.gov/benchmarks/
    - Performance results will be from MILC “asqtad” codes
  - The trends discussed are not dependent upon the specific choices of hardware or action
Generic Single Node Performance

- MILC is a standard MPI-based lattice QCD code
- "Improved Staggered" is a popular "action" (discretization of the Dirac operator)
- Cache size = 512 KB
- Floating point capabilities of the CPU limits in-cache performance
- Memory bus limits performance out-of-cache
Floating Point Performance (In cache)

- Most flops are SU3 matrix times vector (complex)
  - SIMD instructions (MMX, SSE, 3DNow) give a significant boost
    - Site-wise SSE (M. Lüscher)
    - Fully vectorized SSE (A. Pochinsky)
      - requires a memory layout with 4 consecutive reals, then 4 consecutive imaginaries

![Matrix-Vector Performance Chart]

<table>
<thead>
<tr>
<th>Processor</th>
<th>&quot;C&quot; Code MFlop/sec</th>
<th>Site SSE MFlop/sec</th>
<th>Vector SSE MFlop/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xeon, 1.5 GHz</td>
<td>4000</td>
<td>2000</td>
<td>8000</td>
</tr>
<tr>
<td>Xeon, 2.4 GHz</td>
<td>6000</td>
<td>4000</td>
<td>10000</td>
</tr>
<tr>
<td>P4, 2.8 GHz</td>
<td>7000</td>
<td>5000</td>
<td>12000</td>
</tr>
<tr>
<td>P4E, 2.8 GHz</td>
<td>7000</td>
<td>5000</td>
<td>12000</td>
</tr>
</tbody>
</table>
Memory Performance

- Memory bandwidth limits – depends on:
  - Width of data bus – always 64 bits for x86 processors
  - (Effective) clock speed of memory bus (FSB)

- FSB history:
  - pre-1997: Pentium/Pentium Pro, EDO, 66 Mhz, 528 MB/sec
  - 1998: Pentium II, SDRAM, 100 Mhz, 800 MB/sec
  - 1999: Pentium III, SDRAM, 133 Mhz, 1064 MB/sec
  - 2000: Pentium 4, RDRAM, 400 MHz, 3200 MB/sec
  - 2003: Pentium 4, DDR400, 800 Mhz, 6400 MB/sec
  - 2004: Pentium 4, DDR533, 1066 MHz, 8530 MB/sec
  - Doubling time for peak bandwidth: 1.87 years
  - Doubling time for achieved bandwidth: 1.71 years
  - 1.49 years if SSE included
Memory Bandwidth Trend

- PPro 200 MHz
- P III 733 MHz
- P4 1.4 GHz
- Xeon 1.5 GHz
- Xeon 2.4 GHz
- P4 2.8 GHz
- P4E 2.8 GHz

Copy
SSE Copy
**Memory Bandwidth Performance**

**Limits on Matrix-Vector Algebra**

- From memory bandwidth benchmarks, we can estimate sustained matrix-vector performance in main memory.
- We use:
  - 66 Flops per matrix-vector multiply
  - 96 input bytes
  - 24 output bytes
  - \( \text{MFlop/sec} = \frac{66}{(96/\text{read-rate} + 24/\text{write-rate})} \)
- read-rate and write-rate in MBytes/sec
- Memory bandwidth severely constrains performance for lattices larger than cache

<table>
<thead>
<tr>
<th>Processor</th>
<th>FSB</th>
<th>Copy</th>
<th>SSE Read</th>
<th>SSE Write</th>
<th>M-V MFlop/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>PPro 200 MHz</td>
<td>66 MHz</td>
<td>98</td>
<td>-</td>
<td>-</td>
<td>54</td>
</tr>
<tr>
<td>P III 733 MHz</td>
<td>133 MHz</td>
<td>405</td>
<td>880</td>
<td>1005</td>
<td>496</td>
</tr>
<tr>
<td>P4 1.4 GHz</td>
<td>400 MHz</td>
<td>1240</td>
<td>2070</td>
<td>2120</td>
<td>1,144</td>
</tr>
<tr>
<td>Xeon 2.4 GHz</td>
<td>400 MHz</td>
<td>1190</td>
<td>2260</td>
<td>1240</td>
<td>1,067</td>
</tr>
<tr>
<td>P4 2.8 GHz</td>
<td>800 MHz</td>
<td>2405</td>
<td>4100</td>
<td>3990</td>
<td>2,243</td>
</tr>
<tr>
<td>P4E 2.8 GHz</td>
<td>800 MHz</td>
<td>2500</td>
<td>4565</td>
<td>2810</td>
<td>2,232</td>
</tr>
</tbody>
</table>
Memory Bandwidth Performance
Limits on Matrix-Vector Algebra

![Graph showing memory bus limits on matrix multiplication performance for different processors.](image)
**Performance vs Architecture**

- Memory buses:
  - Xeon: 400 MHz
  - P4: 800 MHz
  - P4E: 800 MHz

- P4 vs Xeon shows effects of faster FSB

- P4 vs P4E shows effects of change in CPU architecture
  - P4E has better heuristics for hardware memory prefetch, but longer instruction latencies
Balanced Design Requirements
Communications for Dslash

Modified for improved staggered from Steve Gottlieb's staggered model:
physics.indiana.edu/~sg/pcnets/

Assume:
- $L^4$ lattice
- communications in 4 directions

Then:
- $L$ implies message size to communicate a hyperplane
- Sustained MFlop/sec together with message size implies achieved communications bandwidth

Required network bandwidth increases as $L$ decreases, and as sustained MFlop/sec increases
Communications – I/O Buses

- Low latency and high bandwidths are required
- Performance depends on I/O bus:
  - at least 64-bit, 66 MHz PCI-X is required for LQCD
  - PCI Express (PCI-E) is now available
    - not a bus, rather, one or more bidirectional 2 Gbit/sec/direction (data rate) serial pairs
    - for driver writers, PCI-E looks like PCI
    - server boards now offer X8 (16 Gbps/direction) slots
    - we've used desktop boards with X16 slots intended for graphics – but, Infiniband HCAs work fine in these slots
    - latency is also better than PCI-X
    - strong industry push this year, particularly for graphics (thanks, DOOM 3!!)
    - should be cheaper, easier to manufacture than PCI-X
I/O Bus Performance

Blue lines show peak rate by bus type, assuming balanced bidirectional traffic:
- PCI: 132 MB/sec
- PCI-64: 528 MB/sec
- PCI-X: 1064 MB/sec
- 4X PCI-E: 2000 MB/sec

Achieved rates will be no more than perhaps 75% of these burst rates

PCI-E provides headroom for many years
Communications - Fabrics

- Existing Lattice QCD clusters use either:
  - Myrinet
  - Gigabit ethernet (switched or multi-D toroidal mesh)
  - Quadrics also a possibility, but historically more expensive
  - SCI works as well, but has not been adopted

- Emerging (finally) is **Infiniband**
  - like PCI-E, multiple bidirectional serial pairs
  - all host channel adapters offer two independent X4 ports
  - rich protocol stacks, now available in open source
  - target HCA price of $100 in 2005, less on motherboard

- Performance (measured at Fermilab with Pallas MPI suite):
  - **Myrinet 2000** (several years old) on PCI-X (E7500 chipset)
    - Bidirectional Bandwidth: 300 MB/sec  Latency: 11 usec
  - **Infiniband on PCI-X** (E7500 chipset)
    - Bidirectional Bandwidth: 620 MB/sec  Latency: 7.6 usec
  - **Infiniband on PCI-E** (925X chipset)
    - Bidirectional Bandwidth: 1120 MB/sec  Latency: 4.3 usec
### Balanced Design Requirements

**Dslash and the Network**

Blue curve: measured Myrinet (LANai-9) performance on Fermilab dual Xeon cluster

This gives a very optimistic upper bound on performance – actual performance will be affected by:

- actual message sizes are smaller than modeled
- competition for memory bus
- competition for I/O bus
- competition for interface
- processor overheads for performing the communication

Curvature of network performance graph limits the practical cluster size
Network performance:

- Myrinet 2000 on E7500 motherboards
  - Note: much improved bandwidth, latency on latest Myricom hardware
- Infiniband PCI-X on E7501 motherboards
- Important message size region for lattice QCD is $O(1K)$ to $O(10K)$
Infiniband on PCI-X and PCI-E

Unidirectional bandwidth (MB/sec) vs message size (bytes) measured with MPI version of Netpipe
- PCI-X on E7500
  - "TopSpin MPI" from OSU
  - "Mellanox MPI" from NCSA
- PCI-E on 925X
  - NCSA MPI
  - 8X HCA used in 16X "graphics" PCI-E slot
Infiniband Protocols

Netpipe results, PCI-E HCA's using these protocols:
- "rdma_write" = low level (VAPI)
- "MPI" = OSU MPI over VAPI
- "IPoIB" = TCP/IP over Infiniband
TCP/IP over Infiniband

Options for codes which stream data over sockets:

- "IPoIB" - full TCP/IP stack in Linux kernel
- "SDP" - new protocol, AF_SDP, instead of AF_INET
  - Bypasses kernel TCP/IP stack
  - Socket-based code has no other changes
  - Data here were taken with the same binary, using LD_PRELOAD for libsdp
Processor Observations

Using MILC “Improved Staggered” code, we found:
- The new 90nm Intel chips (Pentium 4E, Xeon “Nacona”) have lower floating point performance at the same clock speeds because of longer instruction latencies
  - but – better performance in main memory (better hardware prefetching?)
- Dual Opterons scale at nearly 100%, unlike Xeons
  - but – must use NUMA kernels + libnuma, and alter code to lock processes to processors and allocate only local memory
  - single P4E systems are still more cost effective
- PPC970/G5 have superb double precision floating point performance
  - but – memory bandwidth suffers because of split data bus.
    32 bits read only, 32 bits write only – numeric codes read more than they write
  - power consumption very high for the 2003 CPUs (dual G5 system drew 270 Watts, vs 190 Watts for dual Xeon
  - we hear that power consumption is better on 90nm chips
MILC Improved Staggered Code ("Asqtad")

Processes used:
- Pentium Pro, 66 MHz FSB
- Pentium II, 100 MHz FSB
- Pentium III, 100/133 FSB
- P4, 400/533/800 FSB
- Xeon, 400 MHz FSB
- P4E, 800 MHz FSB

Performance range:
- 48 to 1600 MFlop/sec
- measured at $12^4$

Doubling times:
- Performance: 1.88 years
- Price/Perf.: 1.19 years !!
Clustering based on:
- Pentium II, 100 MHz FSB
- Pentium III, 100 MHz FSB
- Xeon, 400 MHz FSB
- P4E (estimate), 800 FSB

Performance range:
- 50 to 1200 MFlop/sec/node
- measured at $14^4$ local lattice per node

Doubling Times:
- Performance: 1.22 years
- Price/Perf: 1.25 years
Predictions

Latest (June 2004) Fermi purchase:
- 2.8 GHz P4E
- PCI-X
- 800 MHz FSB
- Myrinet (reusing existing fabric)
- $900/node
- 1.2 GFlop/node, based on 1.65 GF single node performance
  (measured: 1.0 – 1.1 GFlop/node, depending on 2-D or 3-D communications)
Predictions

Late 2004:
- 3.4 GHz P4E
- 800 MHz FSB
- PCI-Express
- Infiniband
- $900 + $1000 (system + network per node)
- 1.4 GFlop/node, based on faster CPU and better network
Predictions

Late 2005:
- 4.0 GHz P4E
- 1066 MHz FSB
- PCI-Express
- Infiniband
- $900 + $900 (system + network per node)
- 1.9 GFlop/node, based on faster CPU and higher memory bandwidth
Predictions

Late 2006:
- 5.0 GHz P4 (or dual core equivalent)
- $\gg 1066$ MHz FSB ("fully buffered DIMM technology")
- PCI-Express
- Infiniband
- $900 + 500\$ (system + network per node)
- 3.0 GFlop/node, based on faster CPU, higher memory bandwidth, cheaper network
Future Hardware Designs

• Past trends in lattice QCD supercomputers:
  – very constrained funding -> cost-obsessed designers
  – custom machines (QCDOC, QCDSP, ACPMAPS)
    • low purchase price
    • high manpower cost, especially talented physics manpower
  – commodity machines
    • custom designs and integrations
    • high manpower cost

• New trends?
  – better commercial machines
    • IBM BG/L
      – 1 TFlops demonstrated on improved staggered codes for about 2X cost of QCDOC
    • Raytheon “Toro” Infiniband mesh clusters
      – terascale commodity Intel clusters for ~ 1.5X cost of homemade lattice QCD clusters
      – end-to-end solutions (disk I/O, scheduling, administration)
      – high availability
For More Information

- Fermilab lattice QCD portal: http://lqcd.fnal.gov/
- Fermilab benchmarks: http://lqcd.fnal.gov/benchmarks/
- US lattice QCD portal: http://www.lqcd.org/
- Visualization courtesy: Pr. D. Leinweber, CSSM, University of Adelaide