FERMILAB-CONF-04-484-CD

Lattice QCD Clusters at Fermilab

D. Holmgren*, Paul Mackenzie, Amitoj Singh, Jim Simone, FNAL, Batavia, IL 60510, USA

Abstract

As part of the DOE SciDAC "National Infrastructure for Lattice Gauge Computing" project, Fermilab builds and operates production clusters for lattice QCD simulations. This paper will describe these clusters.

The design of lattice QCD clusters requires careful attention to balancing memory bandwidth, floating point throughput, and network performance. We will discuss our investigations of various commodity processors, including Pentium 4E, Xeon, Opteron, and PPC970. We will also discuss our early experiences with the emerging Infiniband and PCI Express architectures. Finally, we will present our predictions and plans for future clusters.

CURRENT PRODUCTION CLUSTERS

The production clusters for lattice QCD simulations at Fermilab all share a common architecture. User access to these facilities always occurs through a *head node*, which resides on the public internet, and well as on the cluster private ethernet and high performance networks (*e.g.* Myrinet). The head node serves `/home` and `/usr/local` to the cluster nodes via NFS. The worker nodes reside only on the private networks. Any data files required by user jobs can be staged to local disk via TCP/IP over the high performance network. OpenPBS and the Maui scheduler are used to control jobs.

We operate two main production clusters. The first, consisting of 128 dual 2.4 GHz Xeon nodes, is based on SuperMicro P4DPE motherboards. These use the Intel E7500 chipset, which provides a 400 MHz front side bus (FSB). Each node has 1 GB of memory and 20 GB of local disk. We use Myrinet 2000 for the high performance fabric. These nodes were purchased in late 2002 for $1750 each, with the Myrinet costing $1400/node. MILC improved staggered (*asqtad*) code sustains about 815 MFlop/node on this cluster, or $3.9/MFlop.

The second production cluster consists of 128 single 2.8 GHz Pentium 4E nodes, based on Intel SE7210TP1-E motherboards. These motherboards use the Intel E7210 chipset, which provides an 800 MHz FSB and 64 bit, 66 MHz PCI-X. Each node has 1 GB of memory, and 40 GB of local disk. The computers were purchased in June 2004 for $900/node. We have reused a Myrinet 2000 fabric from an older, retired cluster; the estimated cost today for this fabric would be $900/node. On MILC *asqtad* code, this cluster sustains approximately 1 GFlop/node, or $0.90/MFlop incremental cost (*i.e.*, not including the network cost).

We also operate a 34-node prototype cluster for evaluating Infiniband as a high performance fabric. The nodes include 32 dual 2.0 GHz Xeon systems based on the E7500 chipset, and 2 single 3.2 GHz Pentium 4E systems based on the Intel 925X chipset. The former nodes use PCI-X Infiniband host channel adapters (HCA) from TopSpin, and the latter PCI-E HCA's from Mellanox. Two 24-port TopSpin Infiniband switches are used to interconnect the nodes. We can vary the number of cables interconnecting the switches to study the effects of network over subscription on the performance of lattice QCD codes. The TopSpin switches were purchased in May 2004 for $4000, with the HCA's costing $490 and $735, respectively, for the PCI-X and PCI-E versions. The Pentium 4E systems are based on the Abit AA8 motherboard; their cost was $960. We note that the Mellanox `X8` PCI-E HCA's work well in the `X16` PCI Express slots on the AA8 motherboards. These slots are marketed as graphics ports, but our work shows that they may be used as general PCI Express connections.

ASPECTS OF PERFORMANCE

Lattice QCD codes require excellent single and double precision floating point performance, high memory bandwidth, and low latency high bandwidth communications. Memory bandwidth typically constrains performance on single nodes and on clusters. Communications between nodes in a cluster generally use MPI or similar message passing API's.

Floating Point Performance

Most floating point operations in lattice codes occur during SU3 matrix-vector multiplies. These are small (3x3 and 3x1), complex matrices and vectors. For operands in cache, the throughput of these multiplies is dictated by processor clock speed and the capabilities of the floating point unit. Table 1 shows the performance of matrix-vector kernels on four Intel processors introduced since the year 2000. The "C" language kernels used are from the MILC[1] code. Use of SIMD units on Intel processors, as suggested by Fodor[2] and implemented for the SSE unit by Lüscher[3], can give significant performance improvements. The table lists the performance of two styles of SSE implementation. The first, site wise, uses a conventional data layout scheme with the real and imaginary pieces of individual matrix and vector elements adjacent in memory. The second, fully vectorized, follows Pochinsky's[4] practice

* djholm@fnal.gov

of placing the real components of the operands belonging to four consecutive lattice sites consecutively in memory, followed by the four imaginary components. Whereas site wise implementations require considerable shuffling of operands in the SSE registers in order to perform complex multiplies, the fully vectorized form requires only loads, stores, multiplies, additions, and subtractions.

Table 1: SU3 matrix-vector multiply performance

Processor	"C"	Site-Wise	Vector
1.5 GHz Xeon	864	1708	5451
2.4 GHz Xeon	1312	2758	8191
2.8 GHz P4	1531	3221	9562
2.8 GHz P4E	1212	2712	7405

Results are given in MFlop/sec.

Memory Performance

The bandwidth of access to main memory by processors depends upon the width and the clock speed of the data bus. Intel and compatible `ia32` architecture processors use 64-bit data buses exclusively. The effective speed of the so-called *front side bus*, or FSB, has increased from 66 MHz in the mid-90's, to 800 MHz today. The corresponding peak memory bandwidths have increased from 528 MB/sec to 6400 MB/sec. According to Intel roadmaps, processors with 1066 MHz FSB and 8530 MB/sec peak bandwidths will be available by November of 2004. The doubling time for the exponential fit to these bandwidths is 1.87 years. The doubling for achievable bandwidth, measured using the STREAMS[5] benchmark, is 1.71 years. With SSE optimizations, the achieved doubling time decreases to 1.49 years.

From memory bandwidth measurements, using tools such as STREAMS, an estimate of the throughput of SU3 matrix-vector multiply kernels can be made in the case in which all operands come from main memory, typical for lattice QCD codes. For single precision calculations, each matrix-vector multiply requires 96 input bytes, 24 output bytes, and 66 floating point operations. The throughput is given by this flop count divided by the memory access speed, weighted appropriately according to read and write rates. Table 2 shows the main memory matrix-vector throughput for six generations of `ia32` processor, along with the conventional and SSE assisted read and write rates. Comparing Table 2 to Table 1 clearly shows that memory bandwidth constrains lattice QCD code performance.

Communications - I/O Buses

Lattice QCD codes rely on low latency, high bandwidth message passing. Since all network traffic must flow through the I/O bus, the performance of these codes depends upon competent bus implementations. For current processors, at least 64-bit, 66 MHz PCI-X is required to sustain the required I/O rates. We note that the PCI-X

Table 2: Memory bw, and SU3 matrix-vector throughput

Processor	FSB	Read	Write	M-V
PPro 200 MHz	66	98	98	54
P-III 733 MHz	133	880	1005	496
P4 1.4 GHz	400	2070	2120	1144
Xeon 2.4 GHz	400	2260	1240	1067
P4 2.8 GHz	800	4100	3990	2243
P4E 2.8 GHz	800	4565	2810	2232

FSB is given in MHz. Read and write rates are in MBytes/sec, measured using SSE-assisted code except for the PPro. The final column gives inferred SU3 matrix-vector throughput in MFlop/sec.

bus provided by the Intel E7210 has proven to deliver poor bandwidth, constrained by a restricted connection between the north and south bridges. This constriction limits aggregate traffic on the PCI-X bus to approximately 200 MB/sec.

Since early 2004, PCI Express (PCI-E) has become commonly available on commodity motherboards. PCI-E is not a bus, but rather consists of one or more bidirectional 2 GBbit/sec/direction data rate serial pairs. For device driver authors, however, PCI-E looks exactly like PCI. In addition to much higher bandwidths, PCI-E also exhibits better latency that PCI. We expect a strong industry push toward PCI-E this year, with an emphasis on graphics interfaces. The simplifications of using pairs of high speed serial links, rather than wide parallel buses, should lead to manufacturing simplifications on motherboards and interface boards, which should in turn lead to cost reductions.

Communications - Fabrics

Although our two production clusters use Myrinet, our next cluster will use Infiniband. We've based this decision on our success with reusing fabrics. Currently the most cost effective fabric, in terms of latency and bandwidth, is Infiniband with PCI-E HCA's. Infiniband provides more bandwidth than is required by our lattice QCD codes; however, in several years processors will be fast to require this bandwidth.

Prior to making the decision to switch to Infiniband, we performed a number of synthetic and application benchmark tests on our prototype cluster. We compared the performance of our older Myrinet 2000 network with Infiniband connected via both PCI-X and PCI-E. Using the Pallas MPI benchmark suite[6], the aggregate bidirectional bandwidths observed were 300, 620, and 1120 MB/sec, respectively, for the Myrinet, PCI-X Infiniband, and PCI-E Infiniband networks. Short message one-way latencies were 11, 7.6, and 4.3 μsec, respectively. Fig.1 shows the performance of Myrinet 2000 and Infiniband networks on the Pallas MPI Sendrecv benchmark, which measures aggregate bidirectional bandwidth. Infiniband PCI-X adapters were used on E7501 chipset dual Xeon motherboards; the Myrinet data were from E7500 chipset dual Xeon systems. Two versions of MPI were used for the In-

finiband tests, MVAPICH from OSU[7], and MPIPRO[8]. Myricom's mpich-gm was used for the Myrinet tests. We note that current Myrinet hardware offers substantially greater bandwidth performance, with aggregate bidirectional bandwidths approaching 500 MB/sec.

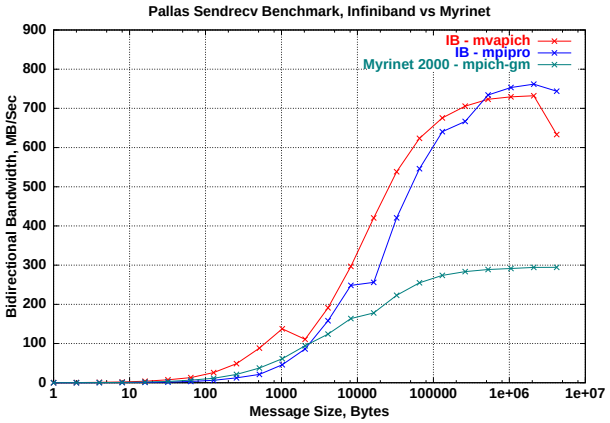

Figure 1: Measured Myrinet and Infiniband performance for MPI message passing.

Processor Observations

We maintain an active program of measuring the performance on lattice QCD codes and relevant synthetic benchmarks of the various processor and network options available on the market[9]. We have the following observations related to processors:

- The new Intel `ia32` chips based on 90nm design rules (Pentium 4E, Xeon "Nacona") have lower floating point performance than chips of the earlier generation with the same clock speeds. This difference is attributed to longer instruction latencies. However, these new processors exhibit better performance when lattices extend into main memory; this effect seems to be due to better automatic hardware prefetching heuristics. Further, we have been able to improve performance on lattice QCD codes by carefully adding software prefetch hints. Although these hints were helpful for Pentium III processors, they have not been effective in all prior Pentium 4 models.
- As many other projects have noted, we found that dual Opteron systems exhibited very good SMP scaling, in many case, nearly 100%. This results from the fact that each Opteron has an integrated memory controller and these systems have a separate memory bus attached to each controller. Processors in SMP systems are linked via hypertransport channels. A given processor can address both local memory, and memory attached to the other processor. However, the latter access suffers from latency and bandwidth penalties. In order to gain the best performance from these machines for lattice QCD code, it is critical that NUMA-aware kernels such as the Linux 2.6.x series are used. These allow processes to be locked to processors, and they also provide system calls giving control over whether local or non-local memory is allocated. The *libnuma* library [10] provides useful shell level and API tools to invoke these system calls.
- The IBM PPC970 processor, also known by the Apple name G5, has superb double precision floating point performance. However, even though these processors have 1066 MHz memory buses, they have less effective memory bandwidth for numerical codes than comparable Intel processors. The data bus on the PPC970 is unique - it is split into a 32 bit read only portion, and a 32 bit write only portion. Simultaneous reads and writes can occur, so for balanced reads and writes, such as when copying blocks of memory, the PPC970 has excellent performance. Numerical codes, on the other hand, tend to do more reading than writing. An SU3 matrix-vector multiply, for example, requires four times as many reads as writes. For such asymmetric access patterns, the memory bus is effectively narrower than the 64 bit Intel counterpart.
- For lattice QCD codes running on only a single processor, by far the most cost effective platform is a low cost desktop based on the Pentium 4E processor. These systems have the fastest memory bus (800 MHz), the principal bottleneck for these codes.

PRICE/PERFORMANCE TRENDS

Fig. 2 shows the measured and estimated price/performance values for six clusters built since late 1998. The oldest cluster shown used Pentium II processors with 100 MHz memory buses, and the newest will be based upon Pentium 4E processors with 800 MHz FSB. From the fit to these data, the doubling time for price/performance is 1.25 years.

Given the historical performance trends, along with vendor roadmaps, we can attempt predictions of future lattice QCD cluster price/performance. These predictions are based upon the following assumptions:

- Intel `ia32` processors will be available at 4.0 GHz and 1066 MHz FSB in 2005.
- Intel `ia32` processors will be available either singly at 5.0 GHz, or in dual core equivalence (*eg.* dual core 4.0 GHz processors) in 2006.
- Greater than 1066 MHz equivalent memory bus speed will be available by 2006 through fully buffered DIMM technology or other advancements.
- The cost per node of high performance networks, such as Infiniband, will drop as these networks increase in sales volume and the network interfaces are embedded on motherboards.

In Fig. 3, extrapolated price/performance values have been added to Fig. 2. By year, these are the details of the

Table 3: Price/Performance Predictions

Date	Cluster Size	Processor	Performance	Node Cost	Network Cost	Price/Performance
2004	128	2.8 GHz P4E	1.1	$900	$900	$1.64/MFlop
Late 2004	256	3.2 GHz P4E	1.4	$900	$1000	$1.36/MFlop
Late 2005	512	4.0 GHz P4E	1.9	$900	$900	$0.95/MFlop
Late 2006	512	5.0 GHz P4E	3.0	$900	$500	$0.47/MFlop

Performance units are GFlop/node.

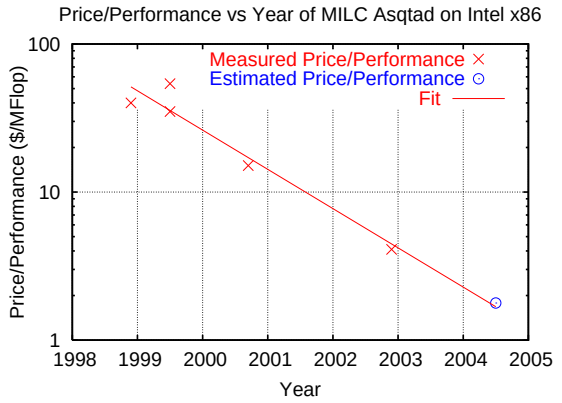

Figure 2: Price/performance history for Intel `ia32` clusters. The measured (1998 - 2003) and estimated (2004) price/performance of clusters running improved staggered action code based on Intel `ia32` clusters at Sandia National Laboratory, Indiana University, and Fermilab.

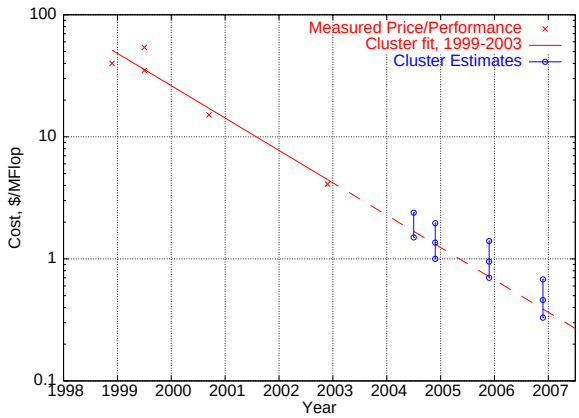

Figure 3: Price/performance predictions for future Intel `ia32` clusters

additional points, also summarized in Table 3: In 2004, the latest Fermilab cluster used 2.8 GHz P4E systems at $900/node. The measured sustained performance of these nodes is approximately 1.1 GFlop/node. A previously purchased Myrinet fabric was used; this fabric has an estimated replacement cost of $900 per node. In late 2004, a cluster based on 3.4 GHz P4E processors with PCI Express and Infiniband would sustain 1.4 GFlop/node, based on the faster processors and the improved communications. In late 2005, a cluster based on 4.0 GHz processors with 1066 MHz FSB would sustain 1.9 GFlop/node, based upon faster processors and higher memory bandwidth. In late 2006, a cluster based on the equivalent of 5.0 GHz processors with memory bandwidth faster than 1066 MHz FSB would sustain 3.0 GFlop/node. In most cases these predictions use technologies predicted a year earlier on vendor roadmaps. For example, 1066 MHz memory buses will appear in 2004, dual core processors in 2005, and fully buffered DIMM technology also in 2005.

REFERENCES

[1] `http://physics.indiana.edu/~sg/milc.html`.

[2] F. Csikor, *et al.*, Comput. Phys. Commun. **134**, 139 (2001), `hep-lat/9912059`.

[3] M. Lüscher, Private communication.

[4] A. Pochinsky, Private communication.

[5] `http://www.cs.virginia.edu/stream/`.

[6] `http://www.pallas.com/e/products/pmb/index.htm`.

[7] `http://nowlab.cis.ohio-state.edu/projects/mpi-iba/`.

[8] `http://www.verarisoft.com/products/cluster/mpipro/`.

[9] `http://lqcd.fnal.gov/benchmarks/`.

[10] `ftp://ftp.suse.com/pub/people/ak/numa/`.