A Performance Comparison Through Benchmarking and Modeling of Three Leading Supercomputers: Blue Gene/L, Red Storm, and Purple

Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Michael Lang, Scott Pakin
{hoisie,gjohnson,djk,mlang,pakin}@lanl.gov
Performance and Architecture Lab (PAL)
Los Alamos National Laboratory

Abstract

This work provides a performance analysis of three leading supercomputers that have recently been deployed: Purple, Red Storm and Blue Gene/L. Each of these machines are architecturally diverse, with very different performance characteristics. Each contains over 10,000 processors and has a system peak of over 40 Teraflops. We analyze each system using a range of micro-benchmarks which include communication performance as well as quantifying the impact of the operating system. The achievable application performance is compared across the systems. The application performance is confirmed via the use of detailed application models which use the underlying performance characteristics as measured by the micro-benchmarks. We also compare the machines in a realistic production scenario in which each machine is used so as to maximize its memory usage with the applications executed in a weak-scaling mode. The results also help illustrate that achievable performance is not directly related to the peak performance.

1. Introduction

The Performance and Architecture Lab (PAL) has analyzed the performance of most of the largest supercomputers in use in the last few years, of which IBM Blue Gene/L, Cray Red Storm (similar to the XT3 product) and ASC Purple (IBM Power5) represent some of the best-in-class architectures that have emerged recently. The actual machines that we utilized for gathering the performance data in this paper are all part of the Advanced Simulation and Computing (ASC) program. Both of the IBM machines are located at the Lawrence Livermore National Laboratory, while the Cray machine is at the Sandia National Laboratory.

The machines under analysis are very diverse architecturally. The philosophy behind Blue Gene/L is based on the virtues of extreme parallelism: lower performance processors, but lots of them with high reliability, and division of tasks for various inter-processor communication needs among five available networks. Red Storm and Purple have more of the look and feel of a traditional cluster, utilizing commodity processors, but uniquely designed networks of different topology. There are fewer processors than in Blue Gene/L, but each of them is more powerful. Each of the three machines also has widely different amounts of memory associated with each processor.

The approach to benchmarking in this paper is a canonical one in that we utilize microbenchmarks and applications. The approach to analysis is based on measured data and on accurate, architecture independent, application models developed by PAL in the last few years that allow performance prediction and provide insight. Through modeling we combine the individual machine characteristics and microbenchmark results, which individually reveal very little, into overall machine performance running real applications.

Given that the architectures are so diverse, we not only compare them head-to-head using the same benchmarks and applications, but also normalize the performance in various ways for a meaningful comparison under realistic scenarios in which such applications are utilized in production. We also insert a historical perspective into the analysis, by further comparing normalized performance on these machines to that on ASCI Q, an earlier machine in the ASC program that currently is the largest production "workhorse" at Los Alamos National Laboratory.

Although benchmarking results of each of the supercomputers considered here have appeared elsewhere [1,12,13,14] this is the first direct comparison of these machines at their largest available configuration that we are aware of. Given that we set ourselves to compare three supercomputers in this paper, we take a broad sweep at the issues, rather than discussing in depth any of the important contributing factors to performance.

To this end, the paper is organized as follows. A brief description of the three architectures is presented in Section 2. In Section 3, a set of microbenchmarks are shown, that have a direct impact on application performance. Section 4 includes measurements and modeled data for the applications under consideration, namely Sweep3D and Sage. Section 4 also presents a head-to-head comparison of the 3 architectures. Conclusions of this work are included in Section 5.

SC2006 November 2006, Tampa, Florida,
USA 0-7695-2700-0/06 $20.00 ©2006 IEEE

2. Architecture Descriptions

A brief description of some architectural features that have a direct impact on application performance follows for each of the three machines. Some of the important architectural factors are summarized in Table 1.

2.1 Blue Gene/L

The basic building block in the system configuration is a board consisting of 32 nodes. The node is a dual core embedded version of the PowerPC 440. Each core (we use "core" and "processor" interchangeably throughout this paper) is clocked at 700 MHz and has a 2.8 GF/s peak floating point performance. Each node has 512 MBytes of memory and no local disk**.** The packaging is very dense, allowing for a standard size cabinet to hold 1024 processors. The topology of the communication network is a 3-D torus. Each node is connected to three communication networks (not counting the JTAG and Ethernet networks) as follows:

- six connections to the 3-D torus network (one per torus direction); this is the main network for point-to-point communications,
- a single connection to the tree network—a high performing network for many collectives, and
- a single connection to a global interrupt network—a "wired OR" network useful for synchronizations (i.e., barriers).

The nodes in the machine can be used in one of two modes:

- *COP: Coprocessor mode* – one processor is dedicated to communication and the other to general processing.
- *VNM: Virtual-Node Mode* – 128K compute processors, in which both processors are used for running the application and for system tasks.

VNM has the potential of increasing performance by up to a factor of two over COP (two processors versus one) but will result with a degree of increased contention within the node (for instance on the memory sub-system), and increased contention between nodes on the communication network. In addition, the memory per node has to be shared among the two processors when using VNM. Thus the memory per processor in this case is approximately 256 MB. In practice, in all cases considered VNM provided an increased level of performance, hence all results presented on Blue Gene/L utilized VNM.

The Blue Gene/L configuration that was available to us consisted of 32K nodes arranged in a 3-D torus as 32×32×32 nodes when used as a single system.

Multiple jobs on the Blue Gene/L can be executed in different partitions. The arrangement of the nodes within a partition affects the achievable application performance. The optimal mapping of processes to nodes is application dependent, and a good mapping for one application is not necessarily good for another. Applications perform best when their logical topology matches the physical processor layout. For some of the applications, such as Sweep3D there is an optimal mapping that results in higher performance than the default mapping. Specifically, the performance improvement on the largest configuration when using optimal mapping is 4%. The largest observed performance improvement across machine sizes in Sweep3D was 15%. However, for other applications including SAGE, no single mapping will be optimal since in the general case with adaptive-mesh refinement (AMR) turned on, the communication requirements may change dynamically from each application cycle to the next.

A detailed description of the Blue Gene/L architecture can be found in [1] and [2].

2.2 Red Storm

Red Storm comprises 10,368 nodes arranged in a 3-D mesh (27x16x24 nodes) with torus links in only the *z* dimension. Each node contains a single 2 GHz Opteron processor giving a peak performance of 4 GF/s, 1 GB of physical memory, and a SeaStar router — integrated directly onto the HyperTransport network — which implements the global 3-D mesh network. Physically, Red Storm occupies four aisles on the machine-room floor; each aisle comprises 27 rows of cabinets (excluding I/O and network cabinets); each cabinet contains three cages; each cage contains eight slots; and each slot is filled with four nodes.

The communication pattern in applications is mapped to MPI ranks by processor, then slot, then cage, then cabinet rather than in a more common *z*, then *y*, then *x* pattern. Although for meshes/tori this could have a direct bearing on application performance, this mapping did not have a significant effect on the Red Storm due to its abundance of bandwidth between nodes. A single node on Red Storm cannot saturate the links, as it will be shown in Section 3.5. Specifically, we observed a performance improvement on Sweep3D of 1% or less when using an optimal mapping.

A detailed description of the Red Storm architecture can be found in [3].

2.3 Purple

The machine configuration consists of 1,536 Squadron IH 8-way Power5 nodes for a total of 12,288 CPUs. The processors are clocked at 1.9 GHz, with a peak floating point speed of 7.6 GF/s. There is 32 GB of memory on each node. The Purple processors can operate in single threaded or multi-threaded (SMT) modes, for the latter there is a maximum of 16 threads per node. However, in multi-threaded mode, it is suggested that MPI threads use

	Blue Gene/L	Red Storm	Purple	ASCI Q
Year introduced	2005	2005	2005	2002
Processor Core	Power PC-440	Opteron	Power5	Alpha EV68
Clock Speed	700 MHz	2.0 GHz	1.9 GHz	1.25 GHz
Peak Core Perf	2.8 GF/s	4 GF/s	7.6 GF/s	2.5 GF/s
Memory/Node	512 MB	1 GB	32 GB	8 GB
Peak Link Uni-BW	175 MB/s (6 links)	3.8 GB/s (6 links)	2 GB/s (2 links)	320 MB/s (2 links)
Node OS	Light-weight kernel	Light-weight kernel	Full AIX	Full Tru64
Cores/node	2	1	8	4
Node count	65,536	10,368	1,536	2,048
Network Topology	32×32×64 Torus	27×16×24 Mesh+	Fat Tree (4-ary)	Fat Tree (4-ary)
Total Memory	32 TB	10 TB	49 TB	16 TB
System Peak	360 TF/s	41.5 TF/s	93 TF/s	20 TF/s

Table 1. Architectural Summary.

no more than 8 of the available threads, the rest being dedicated to system tasks. We show the impact of these two modes of operation on system noise in Section 3.1. The Federation interconnect is a fat-tree topology with 3 levels of switches. The base Federation switch element is a 8-way switch with 4 ports down and 4 ports up. Eight of these switches are packaged together to form a 16 ports up and 16 ports down switch. The full machine requires 480 32-way switches. Each node has 2 network ports.

A detailed description of the Purple architecture can be found in [4].

3. Microbenchmarks

Results and analysis from microbenchmarks with direct implications on application performance are presented here. A more comprehensive set of benchmarking results can be found in [5, 6].

3.1 System Noise

We term “system noise” as the interference from the operating system or hardware that delays the execution of an application. The level of noise is determined by the compounded effect of various system tasks at process and at kernel level that are not synchronized [8].

The overhead of the system software was measured across all nodes on each machine using PAL’s computational noise benchmark, PSNAP [11]. This test consists of the repeated measurement of a single computation which has a known expected run-time (of typically 1ms). This computation is executed millions of times and the actual time taken to complete each task is recorded. From this the average overhead (time above the expected run-time) and also its distribution for each node and across the system are analyzed.

Figure 1 shows the average additional time taken, as a percentage, to execute the known computational task. This is plotted for the first 1530 nodes in the Blue Gene/L, Red Storm and Purple systems. The curves look very similar for the rest of the nodes in the machines. The percentage overhead represents the slow down applications would incur for a job of size 1 node due to intrusion of the operating system during execution. In a parallel job, the noise level will depend on the synchronization level of OS across the system.

It can be seen that there is negligible noise in the Blue Gene/L system, and that is true when using either COP mode or VNM. This is to be expected since Blue Gene/L uses a microkernel based operating system which performs only a minimal number of OS functions on the compute nodes. The slowdown in the coprocessor case is below 0.17%. In VNM there can be contention for resources due to both processors being active. However, even in this case the maximum slowdown is less than 0.2% Very similarly, and for the same reasons, the noise on Red Storm is also extremely low. The average slowdown a process would see on each node from non-application interference is negligible, just under 0.008%.

On Purple, however, each node runs a full instance of the AIX operating system, including a large complement of daemons, approximately 130 of them, the noise is significant. Purple's processors support symmetric multithreading (SMT), which converts each physical processor into two virtual processors. With SMT enabled, the application runs on the eight physical processors but the OS sees sixteen virtual processors and schedules daemons on the "idle" virtual processors. We measured noise on Purple both with and without SMT enabled. The noise level without SMT is lower than with SMT enabled due to the additional interruptions needed for scheduling the two threads per processor. The average slowdown without SMT is 0.6%. With SMT the average slowdown is 3.1%. The average noise with SMT is indicated by the continuous horizontal line in Figure 1. The impact of the noise on application performance is discussed in Section 4.1.

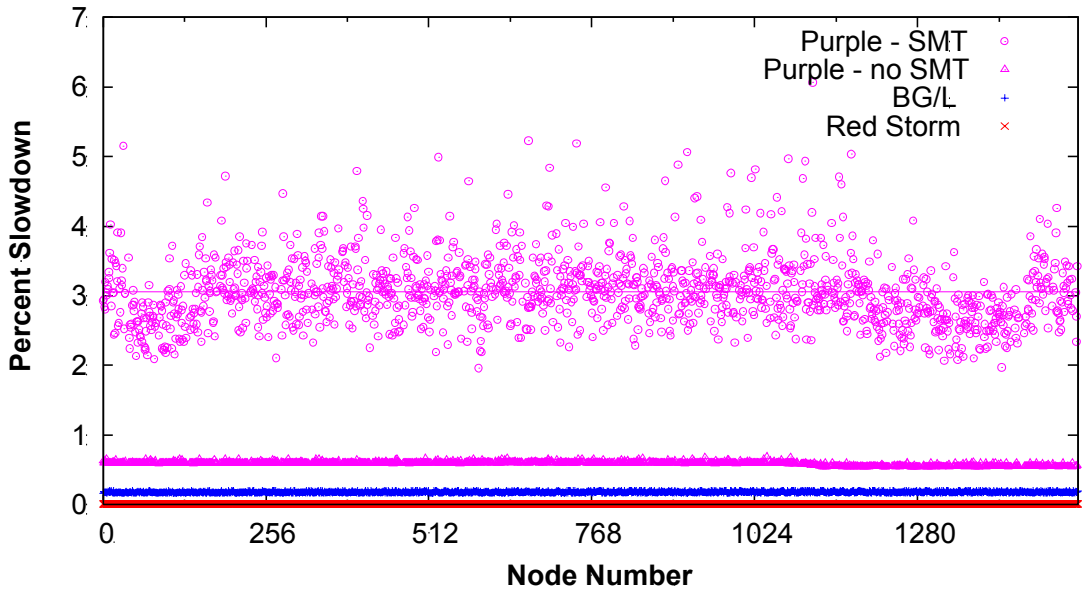

Figure 1. Slowdown caused by System Noise.

3.2 Near-Neighbor Communication Performance

The communication performance between a pair of processors residing on adjacent nodes in the networks of the three machines is analyzed by a standard unidirectional ping-pong type communication. The bandwidth achieved for a uni-directional communication is shown in the semilog plot in Figure 2(a) for both small and large messages. On Blue Gene/L, the saturation point on the measured bandwidth curve is 154 MB/s, close to the peak bandwidth of the link of 175 MB/s. The zero-byte message latency is 2.8µs, see Table 2.

On Red Storm, the maximum bandwidth measured between any pair of nodes was 1660 MB/s, see Figure 2(a), with most of the pairs communicating at approximately 1150 MB/s. The maximum link bandwidth on Red Storm is much higher (3800 MB/s), but the achievable bandwidth is limited by the injection rate of the processor. The minimum latency was measured at 7µs, see Table 2. The pairing of the nodes on Red Storm for these experiments was chosen such that each node pair are physically adjacent.

On Purple, the bandwidth curve shows a maximum close to 3000 MB/s. Given that the peak link speed is 2000 MB/s, this value is explained by the fact that 2 links are available and the messages are striped. The measured zero-byte latency was 4.4µs, see Table 2.

It is also interesting to note the message size that achieves half the peak bandwidth of the network. This is often referred to by the symbol $n_{1/2}$. This metric has an impact on application performance. If the message size is less than $n_{1/2}$, it will be latency bound, if larger than $n_{1/2}$ the bandwidth will dominate.

The value of $n_{1/2}$ for unidirectional communications on Blue Gene/L is approximately 1.4 KB from the data in Table 2. This is quite low resulting in the peak bandwidth of the network being reached for relatively small messages when compared with other current networks. However, the peak bandwidth is also low compared with other networks. On Red Storm and Purple $n_{1/2}$ is 16 KB, and 39KB, respectively.

A bidirectional ping is the same as the uni-directional ping except that the communicating processors in the pair exchange messages between each other at the same time. This can result in contention on the network links and in increased processing required to prepare and simultaneously send/receive data. This type of communication is typical of many applications in which boundary information are exchanged, including SAGE.

The bandwidth curves for bi-directional bandwidth are shown in Figure 2(b), and the latencies are summarized in Table 2. For example, on Blue Gene/L, the latency for a zero-byte message in this case is 3.7µs, an increase of 0.9µs over the unidirectional case. The peak bandwidth, as seen in Figure 2(b) dropped from 154 MB/s to 151 MB/s. The implication of the small drop in bandwidth is that the network interface is able to handle incoming and outgoing messages simultaneously without having to serialize those operations. There is a much more marked drop in the bi-directional bandwidth of Red Storm and Purple, but that is offset by the fact that the peak bandwidths on these machines are roughly one order of magnitude higher than Blue Gene/L.

	Unidirectional Latency [µs]	$n_{1/2}$ [KBytes]	Bidirectional latency [µs]
Blue Gene/L	2.8	1.4	3.7
Red Storm	6.9	16	9.2
Purple	4.4	39	6.3

Table 2. Performance Characteristics: uni- and bi-directional latencies and $n_{1/2}$

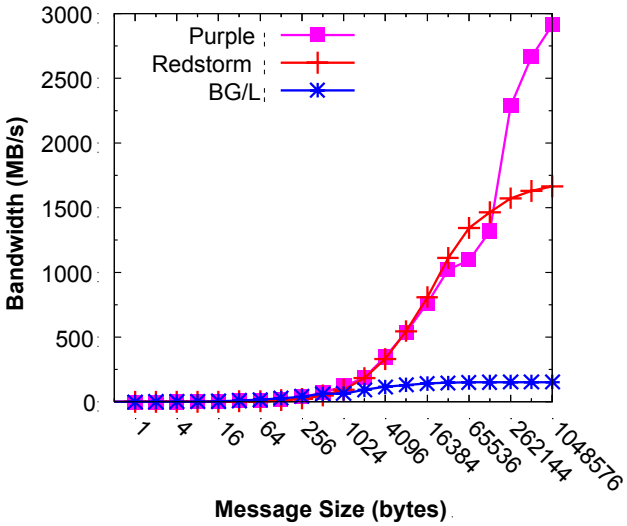

Figure 2(a). Unidirectional Bandwidth.

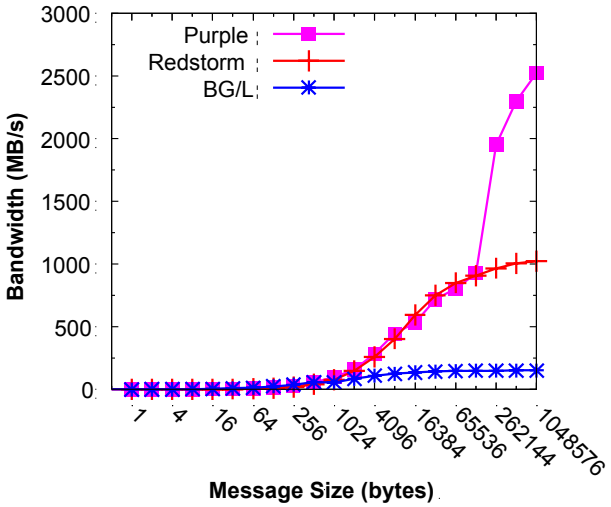

Figure 2(b). Bidirectional Bandwidth.

3.3 Communication Performance from Processor 0 to all other Processors

In this test the task residing on the processor with MPI rank zero sends a zero-byte message to a processor in each of the other nodes sequentially and the message time is recorded. On Blue Gene/L this was performed in COP mode.

The resulting message time is plotted in Figure 3 for only the first 1024 nodes in the configuration. A clear structure in the data can be seen for Blue Gene/L. This results from the sequence of hops a message takes to traverse the torus network to the other processors. The first 1024 nodes reside on the first 32×32 plane of Blue Gene/L and the general trend is for the time to increase with processor ID and then to decrease after rank 512—this is the point at which the use of the torus links in the y dimension start to become beneficial in reducing the hop distance. A similar effect can be seen every 32 nodes (a row). The worst case latency for a zero-byte communication is slightly below 8µs.

On Red Storm the minimum latency is 7.11µs and the maximum is 9.71µs. If the MPI mapping were done in the z-, y-, and x-axis ordering, we should see a regular saw-tooth waveform that gradually increases in height that is typical of a three-dimensional mesh (with a torus link in the first dimension). However, the MPI task ranking does not map cleanly to the physical 3-D processor arrangement. This may be an important consideration for latency-sensitive applications as the latency is not as low as it could be for near-neighbor communications. In addition it is an important consideration for bandwidth sensitive applications, as logical near-neighbor communications may well undergo significant communication contention in the network as a result of the MPI task mapping.

To prove this important point of optimal mapping, a simple reordering of MPI ranks was achieved by specifying the optimal node ordering on job launching. The re-ordering was done on a 336 processor job and the logical arrangement of MPI ranks was chosen so to match the physical arrangement of processors in the 3-D mesh unlike the default ordering. In this case the arrangement was a logical 14x24x1 processor mesh. The zero-byte latency between processor 0 and each other node in this arrangement was measured and is shown in Figure 4. Here we see that the re-arrangement now clearly shows the physical arrangement of processors in the mesh – the high frequency saw-tooth waveform is repeated every 24 processors, and there is one period of the lower frequency triangular waveform resulting from the torus links in the physical Z processor dimension. By re-ordering the MPI-ranking in this way to match the 3-D physical mesh, it may be possible for some applications to achieve a higher level of performance. In practice, this beneficial effect could be achieved with an optimized job launcher, with an MPI library that is cognizant of the architecture, or with custom MPI communicators.

This underscores the importance of optimal mapping for tori, such as the Blue gene/L and Red Storm, in achieving high performance. This performance consideration was anticipated in Section 2.

On the Purple machine, due to system noise, the curve is considerably less clear. However, we can see the latency increasing as a step function corresponding with traversing the switch levels in the network.

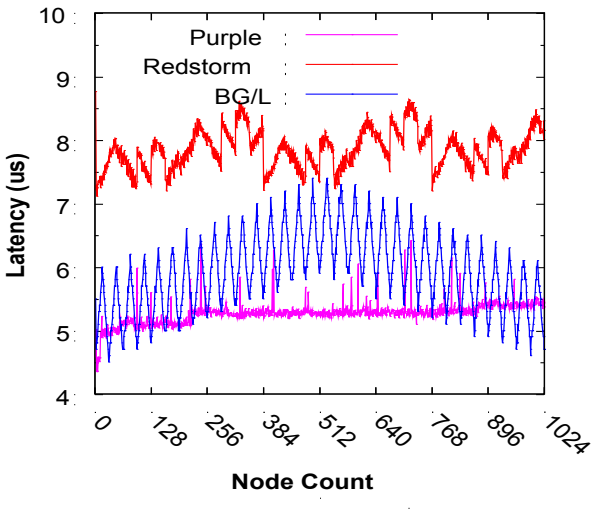

Figure 3: Latency from Node 0 to each other Node.

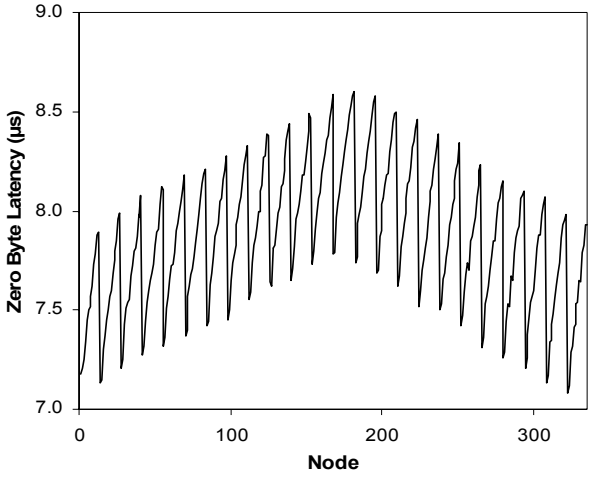

Figure 4: Latency from Node 0 to each other Node after Reordering MPI Ranks on Red Storm

3.4 LogP Communication Analysis

Ping-pong latency measurements may not accurately reflect the communication overheads observed by an application because they fail to consider communication-computation overlap. To estimate the per-message compute time taken away from an application we ran a test in which unidirectional communication is overlapped with "computation" (a simple spin loop) and that computation time is subtracted off from the reported messages latencies, leaving only the communication overhead. Although we used blocking sends and receives our hypothesis was that some overlap is possible because of communication-offload capabilities in the network interface. For instance on Red Storm the SeaStar network chip provides opportunity to offload communications. On all three machines we performed the overhead measurements with the two processes located on adjacent nodes. Blue Gene/L was utilized in virtual-node mode for these experiments.

Figure 5 presents the results of these measurements. The *x* axis is the amount of computation performed and the *y* axis is the length of communication time that could not be overlapped with computation. The difference between the maximal value and the steady-state value in each curve is the communication time which can be overlapped with computation.

On Blue Gene/L, this difference is approximately 0.9 µs, which is reasonable given that blocking communication leaves little opportunity for communication-computation overlap. The numbers are very close to that on Purple as well.

Red Storm shows the highest potential of overlap between communication and computation. From Figure 5 we see that approximately 75% of the latency can be overlapped with communication. This is due to the fact that the SeaStar communication processor handles many of the communication tasks.

Of course, the actual overlap fraction in each case will depend on the workload characteristics.

3.5 Congestion

Communication patterns in applications commonly involve multiple simultaneous messages originating from nodes. "Congestion" in the network is the competition for resources when multiple messages use the same link at the same time. The number of processors per NIC and the availability of paths for messages in the network are the main sources of congestion. Furthermore, the availability of paths in the network is related to both topology and routing mechanism.

Figure 6 shows the bandwidth degradation on the three machines in this communication regime. The fraction of bandwidth (the y-axis) is on a per message basis. The x-axis, labeled "contention level" is the number of simultaneous messages minus 1. For example, contention level zero means that there is a single message using a link at a given time. In the congestion benchmark the N processors that exchange messages are paired (0 & N/2), (1 & N/2 + 1), etc.

On Purple, although there are 2 network ports on each node, the bandwidth per message is lower even when only 2 messages originate from the node. This is due to the static routing of the Federation network which sometimes assigns messages to sub-optimal paths.

The shape of the curve for Blue Gene/L, with step-like decreases, is due to messages alternating between taking one direction and the 180 degree opposite direction. The width of the step (approximately 4) is due to the use of VNM. The full details are in [5].

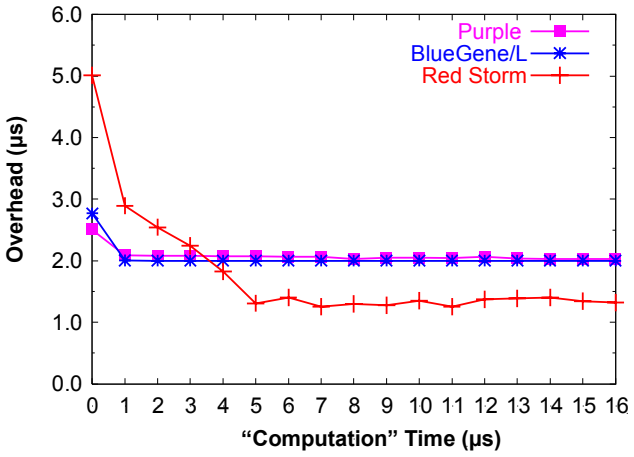

Figure 5. Communication-Computation Overlap.

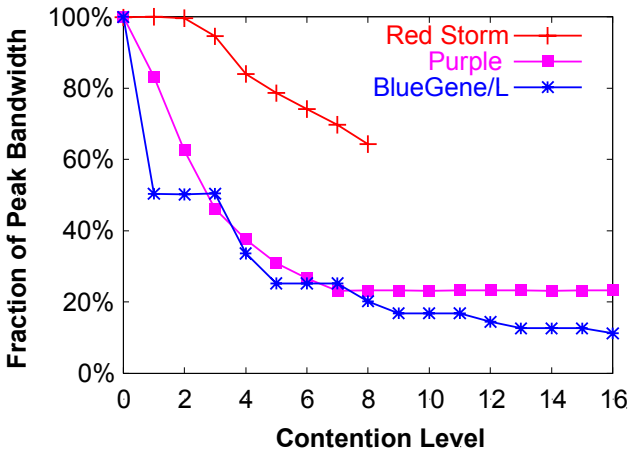

Figure 6. Congestion in the Network.

On Red Storm and Blue Gene/L, the processors were chosen such that they all lie on a single line of nodes in the largest dimension of the mesh, so as to maximize contention. The message size for all these experiments was 1MB.

4. Application Performance

We employ two applications in the comparison of the three architectures: SAGE and Sweep3D. The codes are representative of the workload of the ASC program at Los Alamos and elsewhere. Hydro and deterministic transport account for a sizable portion of many realistic simulations on current ASC systems.

4.1 SAGE

SAGE (SAIC's Adaptive Grid Eulerian hydrocode) is a multidimensional (1D, 2D, and 3D), multi-material, Eulerian hydrodynamics code with adaptive mesh refinement. The code uses second order accurate numerical techniques. SAGE represents a large class of production ASC applications at Los Alamos that routinely run on thousands of processors for months at a time. SAGE is a large-scale parallel code written in Fortran 90, using MPI for inter-processor communications.

In Figure 7 the runtime of SAGE is presented using an input deck containing 13,500 cells per processor in weak-scaling mode. The discrete data points are the measurements, while the curves are the modeled data from the accurate performance model of SAGE developed by PAL [9]. Input to the models consists, among others, of the results from the microbenchmarks described in Section 3.

It is apparent from Figure 7 that the single-processor performance of Blue Gene/L is significantly lower than that of Red Storm and Purple. The Power 5 processor has the best performance on this code, being twice as fast as the Opteron in Red Storm. Scaling of SAGE on Red Storm is better than on Purple. The key issues here are the communication performance and system noise (see section 3.1). With regard to communication performance, Purple has 8 processors that share the 2 NICs on the node, and the contention is high given the communication pattern in SAGE. The microbenchmark utilized in quantifying congestion resembles a key pattern of communication in SAGE. This was analyzed in detail in Section 3.5. Performance at scale is similar between Purple and Red Storm, the single-processor advantage of Purple is compensated by the lower communication time on Red Storm.

4.2 Sweep3D

SWEEP3D is a "compact application", performing a time-independent, Cartesian-grid, single-group, "discrete ordinates" deterministic particle transport computation. Sweep3D is written in Fortran77 with MPI. Estimates are that deterministic particle transport utilize anywhere between 50-80% of the cycles on ASC's extreme-scale parallel machines such as the ones under consideration. A detailed description of the structure of the code and its performance characteristics can be found in [10].

Figure 8 shows Sweep3D measured and modeled performance for the three machines. The Sweep3D performance model is described in [10]. For the purpose of the analysis in this section, Sweep3D ran in weak-scaling mode, utilizing 5x5x400 (in X, Y, Z) grid points per processor. The blocking in one spatial direction and angles was chosen to be 10 and 3, respectively.

As for the case of SAGE, the model matches the data with high accuracy. We see from Figure 8 that Purple and Red Storm have very similar performance. The single processor performance is governed by the clock speed here, because the small memory footprint in Sweep3D allows the code to run mostly in cache.

One essential performance characteristics of Sweep3D is its strong dependence on "pipeline" effects. In a nutshell, the code uses a 2D processor decomposition, and

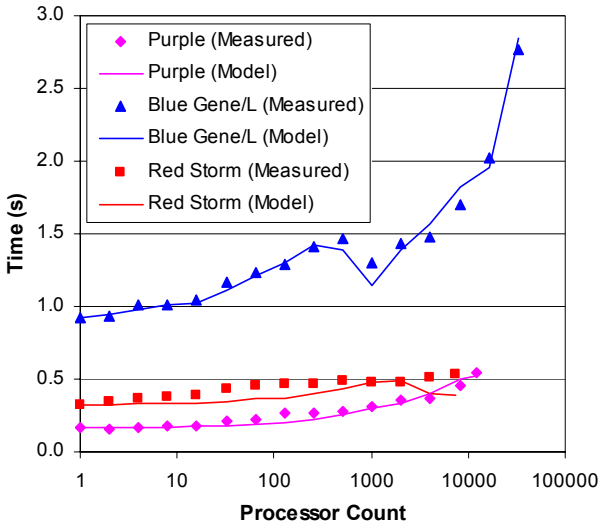

Figure 7. SAGE Performance for the same Input Deck across Machines.

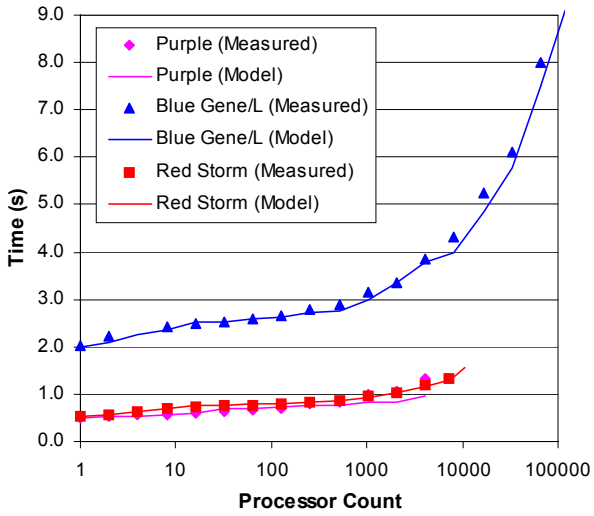

Figure 8. Sweep3D Performance for the same Input Deck across Machines.

wavefronts (sweeps) originating in all corners of the processor array, propagate through this "processor pipeline". Blue Gene/L has a larger processor count and pipeline effects start to dominate. As a result, the run-time increases significantly when using blocks of a fixed size.

4.3 A Normalized Comparison using Benchmarks and Models of SAGE and Sweep3D

In realistic simulations running on these machines the applications are utilized in weak scaling mode and the memory footprint of the application is maximized. This results in choosing the maximum number of cells per processor that the memory can accommodate. The subgrid sizes per processor for the two applications are listed in Table 3. These were chosen to occupy 75% of the available memory on each machine. The comparison is done based on processing rate per cell, to account for the fact that in this scenario the problem sizes are different.

Using the SAGE and Sweep3D models we analyzed the performance of the three architectures in this scenario. Here we are normalizing the performance to that of another Top 20 machine, namely ASCI Q, running the applications in the same regime. ASCI Q is currently the "workhorse" machine at Los Alamos, and was the largest previous generation supercomputer in the ASC program, hence the relative performance compared to it of the newer architectures inserts a historical perspective in the analysis.

The results are shown in Figure 9 for SAGE. We distinguish 2 regions in the graph. Up to 8,192 processors the comparison is done on an equal processor count basis. After 8,192 PEs, the dotted portions of the 3 curves in the figure, ASCI Q's processor count is fixed, and we predict the performance of SAGE up to the full processor count for each of the machines under consideration. For example, Red Storm is approximately 1.8 times faster than ASCI Q running SAGE on 8,192 processors (largest ASCI Q configurations), but when using all 10,368 processors of Red Storm the performance is roughly 2.5 that of ASCI Q.

At maximum scale, relative to ASCI Q, Purple is more than 4 times faster, Red Storm 2.5 times faster and Blue Gene/L 1.6 times faster. The relatively poor performance on this code on Blue Gene/L is due to the slower processor speed compared to ASCI Q and to the lower bandwidth. The communication time in SAGE is dominated by bandwidth. The shape of curves are in part governed by differences in contention on meshes (for Blue Gene/L and Red Storm) as well as differences in problem size causing different degrees of contention at same scale.

Also, the contention increases on Purple at a greater rate due to a large node size (8-way vs. 4-way on ASCI Q). Note also that the performance is heavily dependent on the sub-grid size per processor especially on the two mesh-based machines.

Figure 10 shows the results for Sweep3D. Best blocking in spatial dimension and in angles was used on all the machines and at all scales for their respective inputs. Again, we show two regions in graph, a continuous curve up to the maximum ASCI Q configuration (8,192 processors), and a dotted line beyond that.

Blue Gene/L at full configuration has the fastest processing rate for this application, due to its very large processor count. The performance of Red Storm and Purple start off similar but at scale Purple's performance is higher mainly due to lower latency on network. Within its communication component, Sweep3D is by far dominated by latency, due to its message pattern consisting of a large number of small, point-to-point communications.

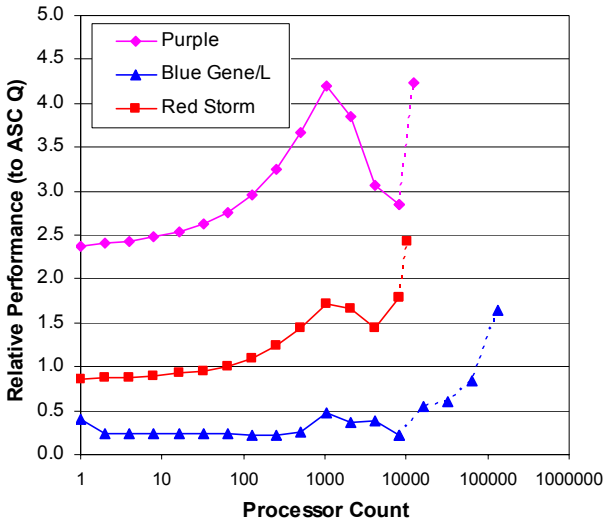

Figure 9. SAGE Relative Performance.

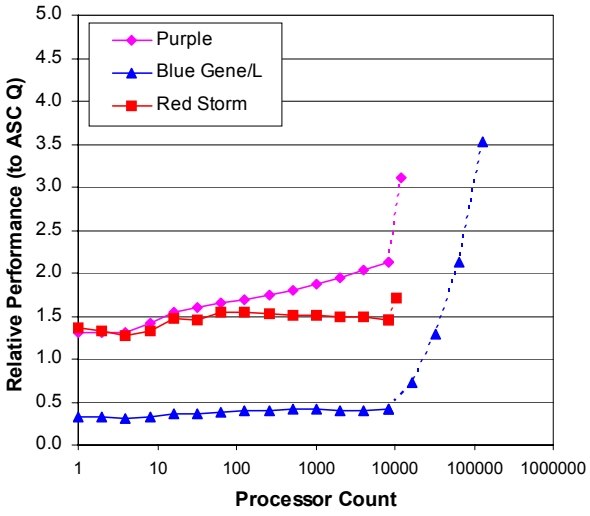

Figure 10. Sweep3D Relative Performance.

	Memory/processor	Relative memory size	Subgrid size (cells) SAGE	Subgrid size (cells) Sweep3D
Blue Gene/L	256KB	1	24K	8x8x75
Red Storm	1GB	4	96K	8x8x300
Purple	4GB	16	384K	8x8x1200
ASCI Q	2GB	8	192K	8x8x600

Table 3. Memory and Input Deck Sizes for SAGE and Sweep3D.

Also interesting to note is the highest relative performance in comparison to peak speed as presented in Table 4. The ratios presented are the performance of the codes for the full configuration to the performance on the full ASCI Q, extracted from Figures 9 and 10.

It is apparent from Table 4 that the peak speed of a machine is a poor indicator of its performance on realistic workloads, both in terms of absolute and of relative performance. This is most vividly apparent for the Blue Gene/L architecture.

	System Peak (TF/s)	Peak Ratio (to Q)	Ratio (Sweep3D)	Ratio (SAGE)
ASCI Q	20	1	1	1
Red Storm	40	2	1.75	2.45
Purple	93	4.65	3.1	4.3
Blue Gene/L	360	18	3.5	1.6

Table 4. Relative Performance for SAGE and Sweep3D to ASCI Q.

5. Conclusions

We have measured and analyzed the performance of three leading supercomputer architectures, Blue Gene/L, Red Storm and Purple under a realistic application workload.

The methodology employed applications and specifically designed microbenchmarks, that we used to measure performance on all the machines. The analysis was done based on these measurements as well as accurate models of the applications, developed by PAL in the last few years. The models utilize results from the microbenchmarks and the input decks to generate performance predictions and insight into the achievable performance.

Given the architectural differences, we analyzed performance directly, but also employed normalized metrics for a meaningful comparison. We introduced an historical perspective in the analysis by comparing performance not only head-to-head but also against the performance of ASCI Q. ASCI Q uses 5 year-old technology, but which still represents the largest production workhorse at the Los Alamos National Laboratory.

All of the three the machines exhibit a high level of

performance under the workload considered. On a per processor basis, Purple and Red Storm are faster than Blue Gene/L, due to their higher node and network characteristics. However, for one of the applications, Sweep3D, the full Blue Gene/L machine consisting of 128K processors is faster than Purple and Red Storm in overall processing rate. When the communication requirements are heavier on bandwidth, such as is the case for SAGE, the full Blue Gene/L machine is slower than Red Storm or Purple using the same metric. The performance of Red Storm and Purple is very similar under the workload considered. Purple's processor is faster than Red Storm's on SAGE, but Red Storm's better noise properties and network performance compensates for this.

We also show that the sheer peak speed of these machines, as are their peak speed ratios, are a poor indicator of absolute and relative performance under a realistic application workload.

6. Acknowledgements

This work was funded by the ASC Program at Los Alamos. We are thankful to Sue Kelly, Jim Tompkins, Courtenay Vaughn and the other members of the Sandia Red Storm, and to Tom Spelce, Jeff Fier, Dave Fox, Mark Seager, Scott Futral, Terry Jones, Steve Louis, Adam Burtsch and the other members of the Purple and Blue Gene/L teams at Livermore for their help with providing access and for active technical support. Comments from reviewers were extremely helpful in improving the clarity of this paper.

7. References

1. NR Adiga et al. An Overview of the BlueGene/L Supercomputer, NR. In Proceedings. of IEEE/ACM SC'02, Baltimore, Maryland, November 2002
2. http://www.llnl.gov/asci/platforms/bluegenel/resources.html.
3. http://www.sandia.gov/ASC/redstorm.html
4. http://www.llnl.gov/asc/platforms/purple/configuration.html
5. Kevin Barker, Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Michael Lang, Scott Pakin. Current status of Blue Gene/L: Performance results for an ASC L2 milestone, Los Alamos Technical Report, LA-UR-057934, September 2005.
6. Kevin Barker, Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Mike Lang, Scott Pakin. An Initial Performance Analysis of the Red Storm Architecture. Los Alamos Technical Report, LA-UR-05-2029, September 2005.
7. Kei Davis, Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Mike Lang, Scott Pakin, Fabrizio Petrini. A Performance and Scalability Analysis of the BlueGene/L Architecture. In Proceedings of the IEEE/ACM Conference on Supercomputing SC'04, Pittsburgh, PA, November 2004.
8. Fabrizio Petrini, Darren J. Kerbyson, Scott Pakin. The Case of the Missing Supercomputer Performance: Achieving Optimal Performance on the 8,192 Processors of ASCI Q. In Proceedings of the IEEE/ACM Conference on Supercomputing SC'03, Phoenix, AZ, November 2003.
9. Darren J. Kerbyson, Henry J. Alme, Adolfy Hoisie, Fabrizio Petrini, Harvey J. Wasserman, Mike Gittings. Predictive Performance and Scalability Modeling of a Large-Scale Application. In Proceedings of the IEEE/ACM Conference on Supercomputing SC'01, Denver, CO, November 2001.
10. Adolfy Hoisie, Olaf M. Lubek, Harvey J. Wasserman. Performance and Scalability Analysis of Teraflop-Scale Parallel Architectures Using Multidimensional Wavefront Applications. In Int. J. of High Performance Computing Applications, 14(4), Winter 2000, pp. 330-346.
11. P-SNAP v 1.0. Open Source Software for Measuring System Noise. LA-CC-06-025. Available from http://www.c3.lanl.gov/pal/software/.
12. Terry Jones, Adam Moody, Chris Chambreau. Purple Bandwidth Testing. Lawrence Livermore National Laboratory Internal Report. December 2005.
13. R. Brightwell, T. Hudson, K. Pedretti, K.D. Underwood, "Cray's SeaStar Interconnect: Balanced Bandwidth for Scalable Performance", IEEE Micro, 26(3), May/June, 2006, pp. 41-57.
14. F. Gygi, E.W. Draeger, B.R. de Supinski, R.K. Yates, F. Franchetti, S. Kral, J. Lorenz, C.W. Ueberhuber, J. A. Gunnels, J.C. Sexton. Large-Scale First-Principles Molecular Dynamics Simulations on the BlueGene/L Platform using the Qbox Code. In Proceedings of IEEE/ACM Supercomputing SC'05, Seattle WA, November 2005.