Scientific Application-Based Performance Comparison of SGI Altix 4700, IBM POWER5+, and SGI ICE 8200 Supercomputers

Subhash Saini, Dale Talcott, Dennis Jespersen, Jahed Djomehri, Haoqiang Jin, and Rupak Biswas
NASA Advanced Supercomputing Division
NASA Ames Research Center
Moffett Field, California 94035-1000, USA
{Subhash.Saini, Dale.R.Talcott, Dennis.Jespersen, Jahed.Djomehri, Haoqiang.Jin, Rupak.Biswas}@nasa.gov

***Abstract*—The suitability of next-generation high-performance computing systems for petascale simulations will depend on various performance factors attributable to processor, memory, local and global network, and input/output characteristics. In this paper, we evaluate performance of new dual-core SGI Altix 4700, quad-core SGI Altix ICE 8200, and dual-core IBM POWER5+ systems. To measure performance, we used micro-benchmarks from High Performance Computing Challenge (HPCC), NAS Parallel Benchmarks (NPB), and four real-world applications—three from computational fluid dynamics (CFD) and one from climate modeling. We used the micro-benchmarks to develop a controlled understanding of individual system components, then analyzed and interpreted performance of the NPBs and applications. We also explored the hybrid programming model (MPI+OpenMP) using multi-zone NPBs and the CFD application OVERFLOW-2. Achievable application performance is compared across the systems. For the ICE platform, we also investigated the effect of memory bandwidth on performance by testing 1, 2, 4, and 8 cores per node.**

I. INTRODUCTION

Developing petascale scientific and engineering simulations for difficult large-scale problems is a challenging task for the supercomputing community. The suitability of next-generation high-performance computing technology for these simulations will depend on a balance among several performance factors attributable to processor, memory, local and global network, and input/output (I/O) characteristics. As new technologies are developed for these subsystems, achieving a balanced system becomes difficult. In light of this, we present an evaluation of the SGI Altix 4700 Density, SGI Altix ICE 8200, and IBM POWER5+ computing systems. We use the High Performance Computing Challenge (HPCC) micro-benchmarks to develop a controlled understanding of individual subsystems, and then use this information to analyze and interpret the performance of NAS Parallel Benchmarks (NPB) and four real-world applications.

In the past, Dunigan et al. studied performance of the SGI Altix 3700 [1]. Biswas et al. studied application-based performance characterization of the Columbia supercluster comprised of SGI Altix 3700 and SGI Altix 3700 Bx2 systems [2]. Saini et al. compared performance of the 3700 Bx2 with the SGI Altix 4700 Bandwidth system [3-5]. Both the 3700 and the 3700 Bx2 are based on a single-core Intel Itanium processor, whereas the 4700 is based on the dual-core Itanium processor. Hoisie et al. conducted performance comparison through benchmarking and modeling of three supercomputers: IBM Blue Gene/L, Cray Red Storm, and IBM Purple [6]. Purple is an Advanced Simulation and Computing (ASC) system based on the single-core IBM POWER5 architecture and is located at Lawrence Livermore National Laboratory (LLNL) [7]. The paper concentrated on system noise, interconnect congestion, and performance modeling using two applications, namely SAGE and Sweep3D. Oliker et al. studied scientific application performance on candidate petascale platforms: POWER5, AMD Opteron, IBM BlueGene/L, and Cray X1E [8]. To the best of our knowledge, this present paper is the first to compare performance of the dual-socket, dual-core Intel Itanium Montvale-based Altix 4700 Density system, the dual-core POWER5+, and the dual-socket, quad-core Intel Xeon-based ICE 8200 system using HPCC, NPB, and four full-scale, production quality MPI and hybrid (MPI+OpenMP) applications [9-10].

The present study uses low-level HPCC benchmarks that measure processor, memory, and network performance of the systems at the subsystem level to gain insights into performance of the NPBs and four production applications on the selected architectures. We explore the issues involved with hybrid applications and the effects of memory bandwidth limits of multi-core systems. While I/O is often important for some applications, none of the benchmarks or applications considered here has significant I/O needs, thus the I/O characteristics of the systems will receive only cursory examination.

The remainder of this paper is organized as follows: Section II details the architectures of the Altix 4700, ICE 8200, and POWER5+ computing systems; Section III describes the suite of HPCC benchmarks, the NPBs, the hybrid multi-zone NPBs and application OVERFLOW-2, and the four real-world applications; Section IV presents and analyzes results from running these benchmarks and applications; and Section V contains a summary and conclusions of the study and future work.

II. HIGH-END COMPUTING PLATFORMS

This section briefly describes the SGI Altix 4700, IBM POWER5+, and SGI ICE 8200 systems.

A. SGI Altix 4700 Density

The Altix 4700 Density system (hereinafter called "Altix") is composed of Individual Rack Units (IRU) [11-12]. Each IRU holds eight processor blades, with each blade containing two dual-core Itanium2 sockets. This particular 4700 system consists of eight racks with four IRUs in each rack. Each IRU also contains four routers to connect to the NUMAlink4 network. Altogether, the Altix system contains 512 dual-core Intel Itanium2 p9000 series sockets. The Altix system's 1.6 GHz Itanium2 processors have 32 KB of L1 cache, 1 MB of L2 instruction cache, 256 KB of L2 data cache, and 9 MB of on-chip L3 cache for each core. The Front Side Bus (FSB), which transports data between memory and the two cores, runs at 667 MHz. The processors are interconnected via the NUMAlink4 network with a fat-tree topology and a peak bidirectional bandwidth of 6.4 GB/s. The peak performance of the Altix system is 6.8 Tflop/s.

B. IBM POWER5+ Cluster

The POWER5+ chip is a reengineered version of the POWER5, using 90-nanometer (nm) processor technology. The technology shrink enabled IBM to place two processor cores on a chip instead of one. The IBM POWER5+ system (hereafter called "POWER5+") used for our tests contains forty 16-way SMP nodes [13]. These nodes are interconnected via a two-link network adapter to the IBM High-Performance Switch (HPS) [14]. The POWER5+ processor core includes private L1 instruction and data caches. Each dual chip module (DCM) contains a POWER5+ chip (dual-core with on-chip L2) and an L3 cache chip. Both L2 and L3 are shared between the two cores. Eight DCMs comprise an IBM POWER5+ node. All memory within a single node is coherent. Multiple nodes, connected with an HPS, make up a cluster.

The L1 instruction cache has a 64 KB capacity and is two-way set associative, while the L1 data cache has a 32 KB capacity and is four-way set associative. The POWER5+ chip has 1.92 MB of L2 cache divided equally over three modules, which are 10-way set associative with a cache line of 128 bytes. The 36 MB off-chip L3 cache is 12-way set associative with a cache line of 256 bytes. The L3 caches are also partitioned in three parts, each serving as a "spill cache" for their L2 counterpart; data that have to be flushed out of the L2 cache are transferred to the corresponding L3 cache part. The L2 cache modules are connected to the cores by the Core Interface Unit, a 2(cores) x 3(L2 modules) crossbar with a peak bandwidth of 40 bytes/cycle, per port. This enables the transfer of 32 bytes to either the L1 instruction or data cache of each core, and the storing of 8 bytes to memory at the same time.

The POWER5+ cluster uses the proprietary HPS network to connect nodes [12]. A switchboard is a basic component of the network providing 16 ports connected to the HPS adapters in nodes and 16 links to other switchboards. Internally, each switchboard has eight switch chips connected to form a multistage omega (Ω) network. The Ω-network uses $n \log_2 n$ connections and there are $\log_2 n$ switching chips.

C. SGI Altix ICE 8200 Cluster

The SGI ICE 8200 system (hereafter called "ICE") uses quad-core Intel Xeon processors [15]. These processors are based on Intel's 65-nm process technology. The processor chip holds two dies, each containing two processor cores. Key features include 32 KB L1 instruction cache and 32 KB L1 data cache per core and 4 MB shared L2 cache per die (8 MB total L2 cache per chip). The 1,333 MHz FSB is a quad-pumped bus running off a 333 MHz system clock, which results in a 10.7 GB/s data rate. The processor has streaming single instruction multiple data (SIMD) Extensions 2 (SSE2) and Streaming SIMD Extensions 3 (SSE3). The ICE system uses a high-speed 4xDDR (Double Data Rate) InfiniBand (IB) interconnect [16]. Each IRU includes two switch blades, eliminating external switches altogether. The fabric connects the service nodes, leader nodes, and the compute nodes. There are two IB fabrics on the ICE: one for MPI (ib0), and the other for I/O (ib1). The ICE system's IB network uses Open Fabrics Enterprise Distribution software. Tests were run with both the vendor MPI library (MPT) and the open source MPI for IB on Mellanox IB-Verbs API layer (MVAPICH) library.

System characteristics of the three supercomputer architectures are summarized in Table I.

TABLE I. SYSTEM CHARACTERISTICS OF ALTIX 4700, ICE 8200, AND POWER5+

Model	SGI Altix 4700	SGI ICE 8200	IBM POWER5+
Total number of cores	1,024	4,096	640
No. of cores per socket	2	4	2
Processor used	Dual-core Intel Itanium2 (Montavle)	Quad-core Intel Xeon (Clovertown)	Dual-core IBM POWER5+
Core clock frequency (GHz)	1.67	2.66	1.9
Floating point/clock/core	4	4	4
Peak perf./core (Gflops)	6.67	10.64	7.6
Technology (nm)	130	65	90
L1 cache size (KB)	32	32 (I) & 32 (D)	64 (I) & 32 (D)
L2 cache size (KB)	256 (I + D)	8 MB shared by 2 cores	1.92 MB (I+D) shared
L3 cache size (MB)	9 (on-chip)	NA	36 (off-chip)
Local memory per node (GB)	8	8	32
Cores per node	4	8	16
Local memory/core (GB)	2	1	2
Total memory (GB)	2,048	4,096	1,280
Frequency of FSB (MHz)	667	1,333	533
Transfer rate of FSB (GB/s)	6.4	10.7	8.5
Interconnect	NUMAlink4	InfiniBand	HPS (Federation)
Network topology	Fat tree	Hypercube	Multi-Stage
Operating system	Linux SLES 10	Linux SLES 10	AIX 5.3
Fortran compiler	Intel 10.0.026	Intel 10.1.008	xlf 10.1
C Compiler	Intel 10.0.026	Intel 10.1.008	xlc 8.0
MPI	mpt-1.16.0.0	mpt-1.18.b30 & mvapich-0.9.9	POE 4.3
Page sizes	16 KB	4 KB	4 KB, 64 KB, 16 MB
File system	CXFS	Lustre	GPFS

III. Benchmarks and Applications Used

Our evaluation approach recognizes that application performance is the ultimate measure of system capability; however, understanding an application's interaction with a computing system requires a detailed understanding of individual component performance of the system. Keeping this in mind, we use low-level HPCC benchmarks that measure processor, memory, and network performance of the architectures at the subsystem level. We then use the insights gained from the HPCC benchmarks to guide and interpret performance analysis of the NPBs and four full-scale applications. In addition, we also explore the hybrid-programming model (MPI+OpenMP), especially for the quad-core ICE nodes and for the symmetric multi-processing (SMP) POWER5+ nodes.

A. HPC Challenge Benchmarks

The HPC Challenge Benchmarks [10] are multifaceted and intended to test various attributes that can contribute significantly to understanding the performance of high-end computing systems. These benchmarks stress not only the processors, but also the memory subsystem and system interconnects. They provide a good understanding of an application's performance on the computing platforms, and are good indicators of how supercomputing systems will perform across a wide spectrum of real-world applications. Four HPCC benchmarks, namely HPL, PTRANS, STREAM, and FFT capture important performance characteristics that affect most real-world applications.

B. NAS Parallel Benchmarks

In this section, we present a brief description of the MPI and multi-zone hybrid (MPI+OpenMP) versions of the NAS parallel benchmarks.

1) NPB MPI Version

The NPB suite is comprised of well-known codes for testing the capabilities of parallel computers and parallelization tools [7]. The benchmarks were derived from CFD codes and are widely recognized as a standard indicator of parallel computer performance. The original NPB suite contains eight benchmarks comprising five kernels (CG, FT, EP, MG, and IS) and three compact applications (BT, LU, and SP). The MPI version of the NPB suite is a source implementation of the "pencil-and-paper" specifications using the MPI message passing interface. We used the NPB3.3 distribution in our study.

2) Multi-Zone Hybrid MPI+OpenMP NPB

Recently, the NPBs were expanded to include the new multi-zone version, called NPB-MZ [17]. The original NPBs exploit fine-grain parallelism in a single zone, while the multi-zone benchmarks exploit multiple levels of parallelism for efficiency, and to balance the computational load. NPB-MZ contains three application benchmarks: BT-MZ, SP-MZ, and LU-MZ, which mimic the overset grid (or zone) system found in the OVERFLOW code. BT-MZ (uneven-sized zones) and SP-MZ (even-sized zones) test both coarse-grain and fine-grain parallelism and load balance. LU-MZ is similar to SP-MZ but has a fixed number of zones (4x4=16). For our experiments, we used the hybrid MPI+OpenMP implementation of the NPB-MZ from the NPB3.3 distribution.

C. Science and Engineering Applications

In this section, we describe the four production applications used in our study: one structured CFD application (OVERFLOW-2), one Cartesian grid application (CART3D), one unstructured tetrahedral CFD application (USM3D), and one application from climate modeling (ECCO). All four applications are production codes.

1) OVERFLOW-2

OVERFLOW-2 is a general purpose Navier-Stokes solver for CFD problems [18]. The MPI version, a Fortran90 application, has 130,000 lines of code. The code uses an overset grid methodology to perform high-fidelity viscous simulations around realistic aerospace configurations. The main computational logic of the sequential code consists of a time loop and a nested grid loop. The code uses finite differences in space with implicit time stepping. It uses overset-structured grids to accommodate arbitrarily complex moving geometries. The dataset used is a wing-body-nacelle-pylon geometry (DLRF6), with 23 zones and 36 million grid points. The input dataset is 1.6 GB in size, and the solution file is 2 GB.

The hybrid decomposition for OVERFLOW involves OpenMP parallelism underneath the MPI parallelism. All MPI ranks have the same value of OMP_NUM_THREADS and this value can be one or higher. The OpenMP shared-memory parallelism is at a fairly fine-grained level.

2) CART3D

CART3D is a high-fidelity, inviscid CFD application that solves the Euler equations of fluid dynamics [19]. CART3D includes a solver called Flowcart, which uses a second-order, cell-centered, finite-volume upwind spatial discretization scheme, in conjunction with a multi-grid accelerated Runge-Kutta method for steady-state cases. In this study, we used the geometry of the Space Shuttle Launch Vehicle (SSLV) for the simulations. The SSLV uses 24 million cells for computation. The input dataset is 1.8 GB and the application requires 16 GB of memory to run. We used the MPI version of this code.

3) USM3D

USM3D is a 3D unstructured tetrahedral, cell-centered, finite-volume Euler and Navier-Stokes flow solver [20]. Spatial discretization is accomplished using an analytical reconstruction process for computing solution gradients within tetrahedral cells. The solution is advanced in time to a steady-state condition by an implicit Euler time-stepping scheme. A single-block, tetrahedral, unstructured grid is partitioned into a user-specified number of contiguous partitions, each containing nearly the same number of grid cells. Grid partitioning is accomplished by the graph partitioning software Metis. Communication among partitions is accomplished by suitably embedded MPI calls into the solver. The test case used a mesh with 10 million tetrahedra, requiring about 16 GB of memory and 10 GB of disk space.

4) ECCO

Estimating the Circulation and Climate of the Ocean (ECCO) is a global ocean simulation model for solving the fluid equations of motion using the hydrostatic approximation [19]. ECCO heavily stresses processor performance, I/O, and scalability of an interconnect. ECCO performs a large number of short message global operations using the MPI_Allreduce function. The ECCO test case uses 50 million grid points and requires 32 GB of system memory and 20 GB of disk to run. It writes 8 GB of data using Fortran I/O. The test case is 1/4° global ocean simulation with a simulated elapsed time of two days.

IV. Results

In this sub-section, we present performance results of selected HPCC benchmarks, NPBs, and application codes. We use the HPCC results to analyze and understand the results for the NPBs and applications.

A. HPC Challenge Benchmarks

In Figure 1, we plot performance of the compute-intensive, embarrassingly parallel DGEMM (matrix-matrix multiplication) for the three systems [8, 20, 21]. Here, performance on ICE is the best, followed by the POWER5+ and Altix systems, and is proportional to the theoretical one-core peak performance of 10.64, 7.6, and 6.67 Gflop/s respectively. Achieved performance is 83%, 94%, and 93% of the peak on the ICE, POWER5+, and Altix, respectively. For the POWER5+ and Altix, performance is almost constant. However, for the ICE system, performance is highest for four cores and remains almost constant from 8 to 512 cores. For four cores, only half of the node (one core from each die) is used, effectively doubling memory bandwidth available for each process.

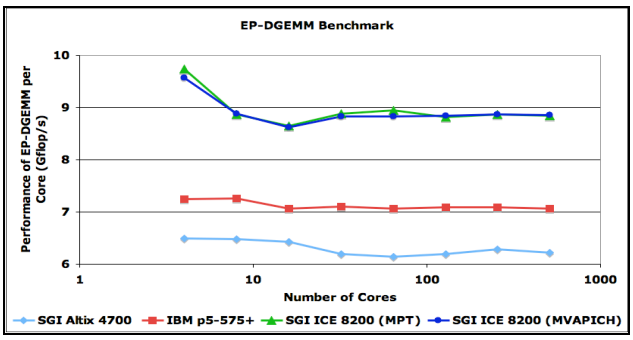


Figure 1. Performance of EP-DGEMM on Altix, POWER5+, and ICE.

In Figure 2, we plot performance of the compute-intensive global high-performance LINPACK (G-HPL) benchmark for each of the three systems [20]. Performance of G-HPL on the POWER5+ is highest. The ICE is either second or last, depending on the MPI library. Within a node on ICE (i.e., up to 8 cores), performance of both MPT and MVAPICH is almost equal. However, beyond 8 cores, performance using MVAPICH is much better and this gap between MPT and MVAPICH keeps increasing as the number of cores increases. This is due to better remote data access using MVAPICH (see Figure 6).

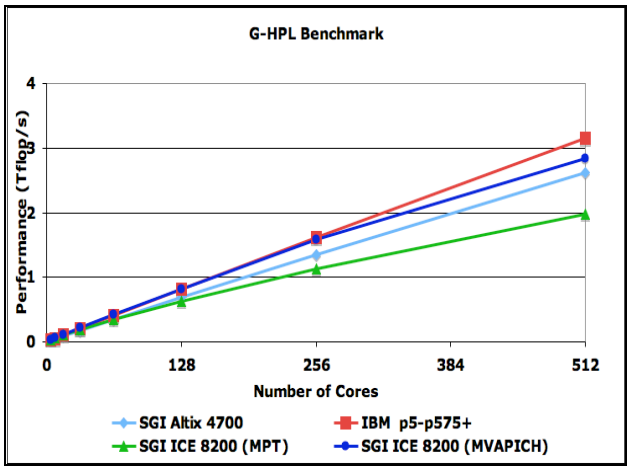


Figure 2. Performance of G-HPL on Altix, POWER5+, and ICE systems.

In Figure 3, we plot memory bandwidth using the EP-STREAM benchmark for each system [23]. The average measured memory bandwidth is 1.52 GB/s for the Altix, 4.2 GB/s for the POWER5+, and 0.677 GB/s for the ICE. Measured bandwidths are close to the theoretical value for the POWER5+, and much less for the other systems. The FSB frequencies are 667 MHz, 533 MHz, and 1,333 MHz for the Altix, POWER5+, and ICE systems respectively. The Altix and POWER5+ systems can load two 64-bit words (16 bytes) per FSB clock; the ICE can load 8 bytes for each FSB per FSB clock. Therefore, total peak theoretical bandwidth per local memory is 10.7 GB/s (667 MHz x 16 bytes), 8.5 GB/s (533 MHz x 16 bytes), and 21.3 GB/s (1,333 MHz x 8 bytes x 2) for the Altix, POWER5+, and ICE systems respectively. For the Altix and POWER5+, this bandwidth is available to a single core if other cores are idle. However, for the ICE, each core is limited to the bandwidth of one FSB (10.7 GB/s), which is half that of the whole memory sub-system. Averaged over the number of cores per FSB, the theoretical peak read bandwidths are 2.67 GB/s, 4.26 GB/s, and 2.67 GB/s for the Altix, POWER5+, and ICE respectively. Write bandwidths are half these values.

Compare the ICE bandwidths at 4 and 8 cores. With half the cores idle, the other cores see twice the bandwidth. In fact, for this case, performance of *SingleSTREAM_Triad* and *StarSTREAM_Triad* are almost the same. In the *SingleSTREAM* benchmark, only a single core is performing computations. In the *StarSTREAM* benchmark, each core in the program is performing computations. Because there is little memory contention in the four-core case, *StarSTREAM* is nearly as fast as *SingleSTREAM*. For the POWER5+, memory bandwidth for four and eight cores is almost double that of the bandwidth for 16 to 512 cores. The reason for this is similar to the ICE system's situation—idle cores leave memory bandwidth available to the active cores. Performance goes down when using 16 cores and above, since both cores per processor chip are used. Because the L2 cache, L3 cache, and memory bus are shared, performance is halved.

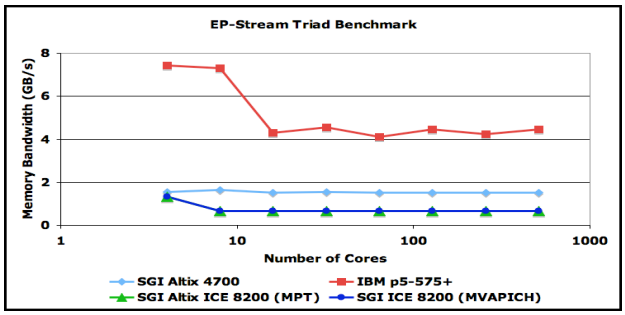


Figure 3. Performance of EP-STREAM on Altix, POWER5+, and ICE.

A figure of merit we can derive from the previous benchmarks is GB/Gflops. This indicates how many bytes of memory bandwidth are available for each floating point operation, as measured by EP-STREAM and EP-DGEMM. For the Altix, POWER5+, and ICE, GB/Gflops is 0.23, 0.55, and 0.063 respectively.

In Figure 4, we plot the random-ordered ring latency for 4 to 512 processors for the three systems. On the POWER5+, latency is 2.5 µs from 4 to 16 cores, and then gradually increases and becomes constant with a value of 14 µs beyond 128 cores. The initial low latency reflects message passing that stays within a node. Above 16 processes, overhead is due to going through extra stages of the HPC switch.

On the ICE system, latency is about 0.86 and 0.98 µs for 4 and 8 cores respectively, and then drastically increases for 16 and 32 cores, after which the increase is more gradual. The reasons are similar to those on the POWER5+: within-node communication is quick, while off-node is slower. Within a node (8 cores), latency using MPT is lower than that of MVAPICH. However, as the number of cores increases beyond 8, the latency of MPT increases slightly more than that of MVAPICH.

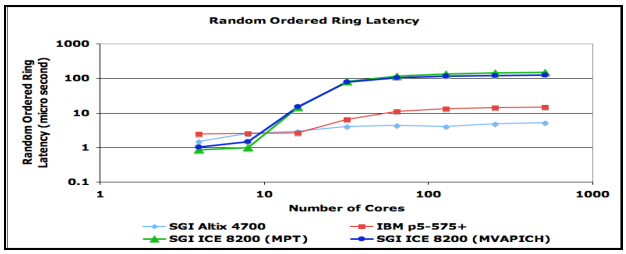


Figure 4. Performance of random-ordered ring latency for Altix, POWER5+, and ICE systems.

In Figure 5, we show the random-ordered ring bandwidth for the three systems. Again, the POWER5+ and ICE systems show rapid drop-offs in performance once communication is off-node. Performance stabilizes at the two-node number, with small decreases as process counts increase. The NUMAlink4 interconnect in the Altix shows excellent scaling across the range of processes tested, and is the clear winner from 32 processes up to the highest count tested. On ICE, within a node, the bandwidth for MPT is higher than that of MVAPICH. However, beyond 8 cores, the bandwidth of MVAPICH is marginally higher than that of MPT.

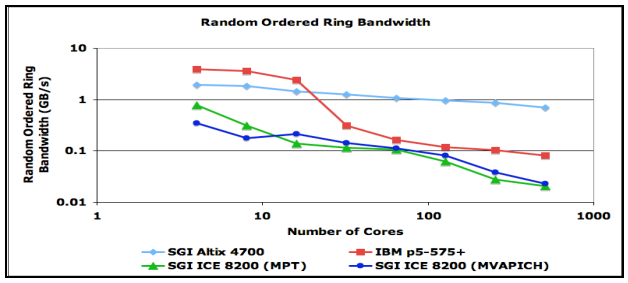


Figure 5. Performance of random-ordered ring on Altix, POWER5+, and ICE

In Figure 6, we plot performance of the Random Access benchmark as Giga UPdates per second (GUPS) for 4 to 512 processors for all three systems [22]. GUPS measures the rate at which a system can update individual elements of a table spread across global system memory. GUPS profiles the memory architecture of a system and is a measure of performance similar to Gflop/s. In Figure 6, we see the benchmark scales very well for the Altix and POWER5+. On the ICE system, the MPT version of the benchmark performs well only within a node (4 and 8 cores). Beyond a node, performance degrades drastically and then becomes constant from 32 to 512 cores. However, using MVAPICH, performance improves slowly up to 64 cores and then increases almost linearly from 64 to 512 cores. MVAPICH is tuned for IB and performs well here.

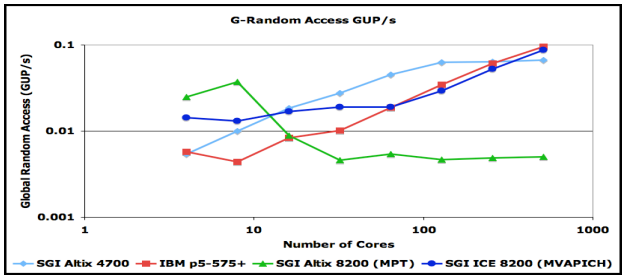


Figure 6. Performance of RandomAccess benchmark on Altix, POWER5+, and ICE systems.

Figure 7 shows performance of the parallel matrix transpose (PTRANS) benchmark [20, 21]. PTRANS exchanges messages simultaneously between pairs of processors. This benchmark is a useful test for measuring total communication capacity of the system interconnects. It should be noted that performance of PTRANS strongly depends on configuration of the process grid. Performance is best when the number of communicating pairs is minimized. For example, a matrix of 3x3 processes has 3 communicating pairs, namely 2-4, 3-7, and 6-8. However, a 1x9 process grid has 36 communicating pairs (1-2, 1-3, 1-4, 1-5, 1-6, 1-7, 1-8, 1-9, 2-3, …, 8-9). For each system, we tried all possible configurations of the process grid. The results presented are for the configuration that yields the best performance. Up to 32 cores, performance is highest on the POWER5+ and lowest on the ICE system. However, from 64 cores onwards, the Altix has the highest performance followed by POWER5+. Among the three systems, performance on the ICE system is lowest. On ICE within a node, performance of MPT and MVAPICH is almost the same. However, beyond 8 cores, performance of MVAPICH is higher than that of MPT due to lower message latency. This benchmark uses "all-to-all" communication and therefore stresses the global network. Overall, scalability of the Altix system's network is best in the

entire range of processors from 4 to 512. The POWER5+ performs well when communication does not involve the interconnect (up to 16 processes). Performance plateaus initially when the HPS is involved, then improves again, but not as well as NUMAlink4. Scalability of the POWER5+'s network is limited by additional stages of the Omega network.

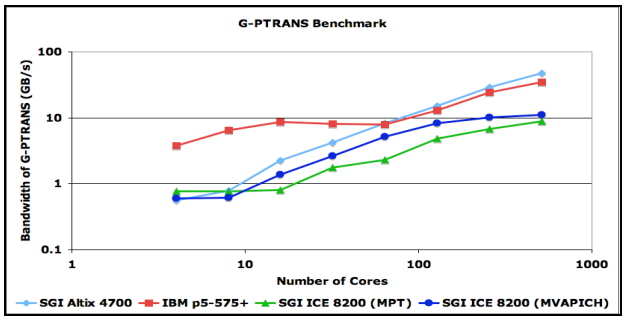


Figure 7. Performance of PTRANS benchmark on Altix, POWER5+, and ICE systems.

In Figure 8, we plot performance of the G-FFTE benchmark on the Altix, POWER5+, and ICE systems for 4 to 512 cores. The G-FFTE benchmark measures floating-point execution rate of a double precision complex 1D Discrete Fourier Transform [24]. In G-FFTE, since cyclic distribution is used, all-to-all communication takes place only once. The benchmark stresses inter-processor communication of large messages. Both G-FFTE and PTRANS are strongly influenced by the memory bandwidth (EP STREAM copy) and the inter-process bandwidth (random-ordered ring). Like PTRANS, G-FFTE also performs a parallel 2D transpose of a matrix involving all-to-all communication stressing the global network. For this reason, qualitatively, performance of PTRANS and G-FFTE benchmarks is quite similar. On ICE within a node, performance of G-FFTE using MPT is better than when using MVAPICH because the former has lower latency and higher bandwidth. However, beyond 8 cores performance of FFT using MVAPICH is better than MPT because the MVAPICH library is tuned for IB and provides lower latency and higher bandwidth than MPT.

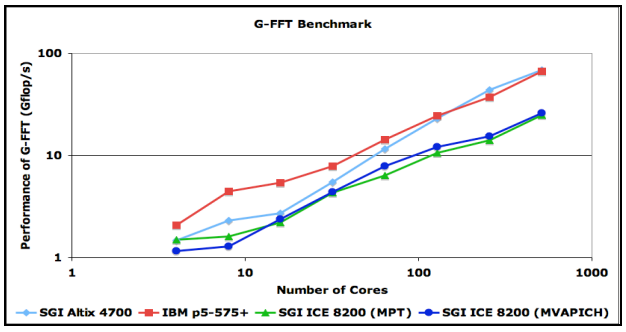


Figure 8. Performance of G-FFT benchmark on Altix, POWER5+, and ICE.

Up to 128 cores, the POWER5+ system has the best performance and scalability. Up to 32 cores, performance and scalability are almost identical on the Altix and ICE systems. However, beyond 32 cores, both performance and scalability on the Altix is greater than that of the ICE system due the higher latency and lower bandwidth of IB compared to NUMAlink.

B. *NAS Parallel Benchmarks*

In this sub-section, we present results for six (MG, CG, FT, BT, LU, and SP) of the MPI NPBs [7].

Figure 9 displays performance of the NPB Class C MG benchmark for the Altix, POWER5+, and ICE systems for 16 to 512 processors. Up to 256 processors, performance ranking is: POWER5+, Altix, then ICE. The MG benchmark is a memory-bound benchmark and has highly structured short- and long-distance communications. Its performance correlates with the STREAM memory bandwidth of these systems, namely 4.2 GB/s, 1.5 GB/s, and 0.677 GB/s respectively up to 256 cores. On the Altix system, performance of the MG benchmark increases at 512 cores because there is now enough combined L3 cache to hold all the data. Additionally, the NUMAlink4 interconnect out-performs POWER5+'s HPS.

Figure 10 displays performance of the NPB Class C CG benchmark for the Altix, POWER5+, and ICE systems, for 16 to 512 processors. Here, performance is almost the same from 16 to 64 processors for the Altix and POWER5+ systems, and much higher than the ICE system. For 128 and 256 processors, the Altix system's performance is better than that of the POWER5+, and performance of both is better than that of ICE. The reason for this is the CG benchmark is memory-bound due to indirect addressing used in its sparse matrix solver, and is network latency-bound due to a large number of small messages. Therefore, CG performs well on the Altix and POWER5+ systems, and performs poorly on the ICE system.

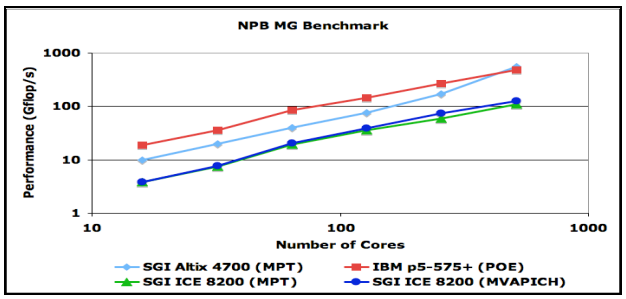


Figure 9. NPB Class C MG benchmark on Altix, POWER5+, and ICE.

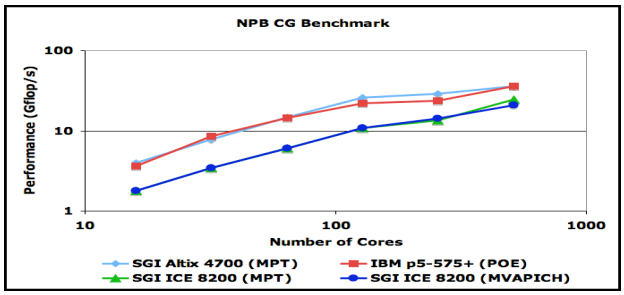


Figure 10. NPB Class C CG benchmark on Altix, POWER5+, and ICE.

Figure 11 captures performance of the NPB Class C FT benchmark for the Altix, POWER5+, and ICE systems for 16 to 512 processors. The POWER5+ outperforms the Altix, which, in turn, outperforms the ICE system for the entire range of processors. The performance gap between the POWER5+ and Altix systems and the ICE system gradually widens. The reason for this is the FT benchmark is both compute-bound as well as memory-bound and depends largely on the bisection bandwidth

due to all-to-all communication to transpose the matrix, and therefore, correlates with memory bandwidth and bisection bandwidth. On the ICE, MVAPICH significantly outperforms MPT.

Figure 12 shows the performance of the NPB Class C BT benchmark for the Altix, POWER5+, and ICE systems for a range of processors from 16 to 484. We do not have results for 512 cores since this benchmark requires square grids. In the entire range of cores performance on the POWER5+ is higher than on the Altix, which in turn, is higher than the ICE system. BT is mainly compute-bound and as such, performance correlates with the floating-point performance and with the memory bandwidth.

Figure 13 captures performance of the NPB Class C LU benchmark for the Altix, POWER5+, and ICE systems for 16 to 512 processors. Once again, the Altix and POWER5+ do much better than the ICE, with the Altix leading at high processor counts. LU's 2-D pipelined communication pattern generates many small messages. As predicted by the GUPS micro-benchmark, MPT on the ICE does poorly here.

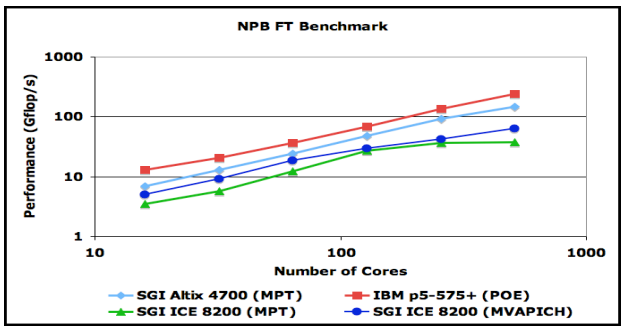


Figure 11. NPB Class C FT benchmark on Altix, POWER5+, and ICE.

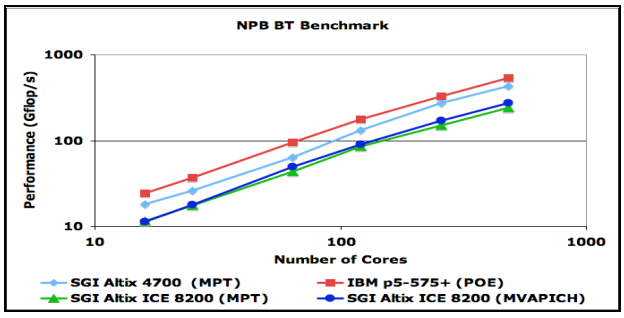


Figure 12. NPB Class C BT benchmark on Altix, POWER5+, and ICE.

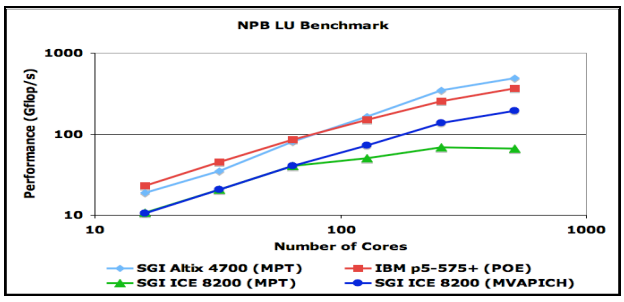


Figure 13. NPB Class C LU benchmark on Altix, POWER5+, and ICE.

Figure 14 displays performance of the NPB Class C SP benchmark for the Altix, POWER5+, and ICE systems for 16 to 480 processors. This benchmark has both nearest-neighbor and long-range communication. Once again, superior memory bandwidth of the POWER5+ system places it first, and the memory-starved ICE system last.

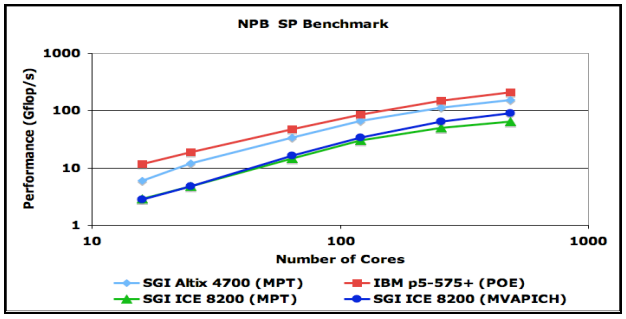


Figure 14. NPB Class C SP benchmark on Altix, POWER5+, and ICE.

C. *Scientific and Engineering Applications*

In the following, we present results for four real-world applications on the Altix, POWER5+, and ICE systems. Results for ICE use the MPT library. MVAPICH results are similar, except for CART3D, where there was not enough memory to run with MVAPICH for the test dataset.

1) OVERFLOW-2 (MPI)

In this sub-section, we present and analyze results of the simulation using the CFD application OVERFLOW-2 on the three systems [16].

Figure 15 shows wall-clock time for 8 to 512 processors for OVERFLOW-2. Performance of OVERFLOW-2 on the Altix and POWER5+ systems is better than on the ICE system across the whole range of processors. OVERFLOW-2 is memory-bound and performance is better on the Altix and POWER5+ systems as compared to ICE because memory bandwidth of the Altix and POWER5+ is better than the ICE system (1.5 GB/s and 4.2 GB/s versus 0.67 GB/s). Further, memory bandwidth of the ICE system (0.67 GB/s) is almost half that of the Altix (1.5 GB/s). Although the POWER5+ system's memory bandwidth is about three times that of the Altix, the Altix outperforms the POWER5+. This turns out to be because the Intel compiler does a better job of optimizing certain heavily used routines than the POWER5+ compiler does.

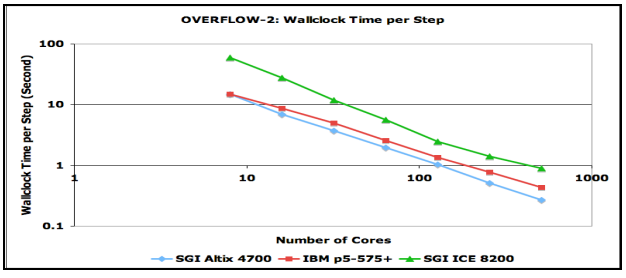


Figure 15. Wall-clock time (compute time + communication time) for OVERFLOW-2 for Altix, POWER5+, and ICE systems.

Figure 16 shows the same cases, but looking only at compute time per step. Qualitatively, Figures 15 and 16 are the

same except the times in Figure 15 are higher than those in Figure 16 (the times in Figure 15 include both compute and communication time). The ICE system, in spite of having the highest floating-point operations per clock (10.64 Gflop/s vs. 6.4 Gflop/s and 7.6 Gflop/s), performs the worst. The ratio of GB/Gflop is the lowest for ICE—memory bandwidth is inadequate to feed the floating-point units.

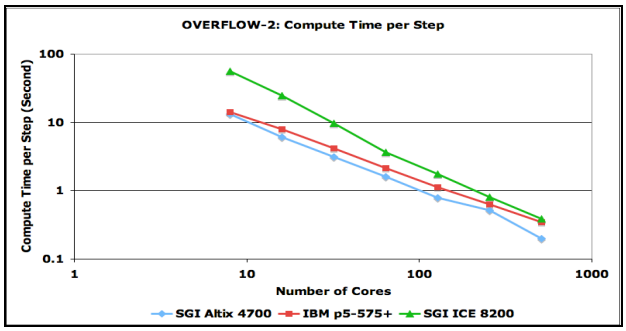


Figure 16. Compute time of OVERFLOW-2 for Altix, POWER5+, and ICE systems.

Figure 17 shows the communication time per step. Communication time is lower on the Altix and POWER5+ systems than on ICE. The slightly lower time for the POWER5+ at 8 processes can be explained by the extra memory bandwidth available from using only half the cores in a node. For 128 processors and up, communication time on the Altix and POWER5+ systems becomes almost the same—for large numbers of processors, there is less data to be sent and these data are being communicated in parallel. The ICE performs less well, as predicted by the HPCC latency and bandwidth results (Figures 4 and 5).

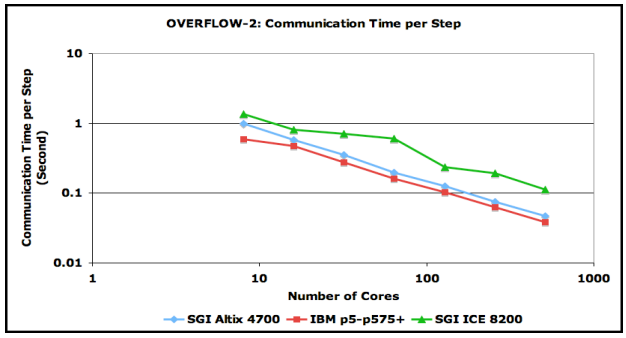


Figure 17. Communication time for OVERFLOW-2 for Altix, POWER5+, and ICE systems.

2) *CART3D*

In this sub-section, we present and analyze results of the simulation using the CFD application CART3D on each of the systems [17].

Figure 18 shows wallclock (execution) time per step for 16 to 512 processors for the MPI version of CART3D. Performance of CART3D is best on the POWER5+ system and worst on the ICE system. The Altix falls in between the two but closer to the POWER5+. Because CART3D is both memory-intensive and compute-intensive, it benefits from a faster processor clock and better memory bandwidth. Thus, this application performs best on the POWER5+ with its high (highest of the three systems) GB/Gflop ratio. We could not run CART3D on ICE at 256 and 512 cores due to lack of memory on the node which contains the MPI rank 0 process.

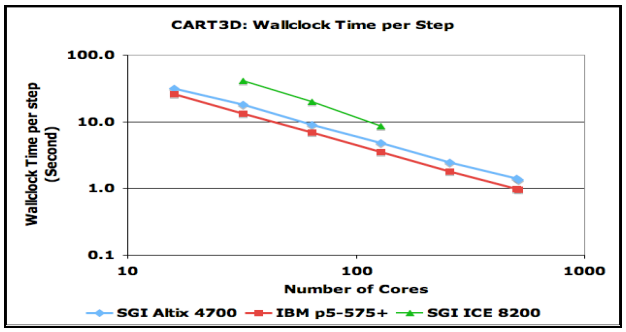


Figure 18. Wallclock time per step of CART3D for Altix, POWER5+, and ICE systems.

3) *USM3D*

In this subsection, we present results of the USM3D application on the three systems [18].

To test the effect of the memory subsystem, we plot the cycle wallclock time per step for a range of processors in Figure 19. Performance of USM3D is better on the POWER5+ system than on the Alix and ICE systems. This is because USM3D is an unstructured mesh-based application and memory-bound from indirect addressing which does not make good use of the L2 or L3 caches—it depends exclusively on the memory bandwidth, which is highest for the POWER5+ (4.2 GB/s) and lowest for the ICE system (0.67 GB/s). Beyond 256 processors, USM3D scaling is poor for this dataset, and performance becomes limited by communications.

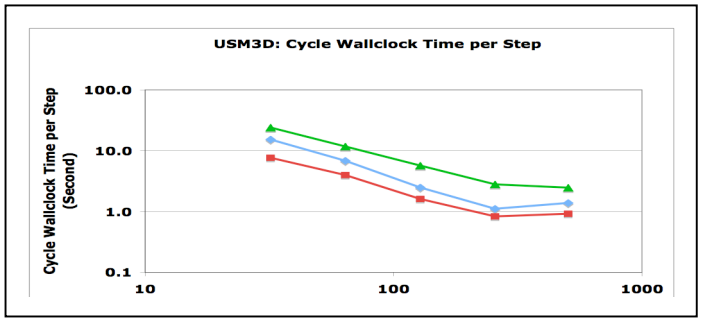


Figure 19. Wallclock time per step for USM3D for Altix, POWER5+, and ICE.

4) *ECCO*

In this sub-section, we present and analyze results of the simulation using the climate modeling application ECCO on each of the systems [19].

In Figure 20, we show wall-clock and I/O time for ECCO. This code is memory-bound for small processor counts while its performance for large processor counts depends on network latency. Since the POWER5+ system has the highest memory bandwidth (4.2 GB/s), ECCO performs much better on this system than on the Altix or ICE. ECCO performs worst on the ICE system, as it has the lowest memory bandwidth (0.67

GB/s). Performance of the Altix with a memory bandwidth of 1.5 GB/s falls in between the POWER5+ and ICE systems. Figure 20 also includes wall-clock time for writing 8 GB of data for all three systems. Writing time is about 85 seconds for the Altix and ICE systems, and about 28 seconds for the POWER5+ system.

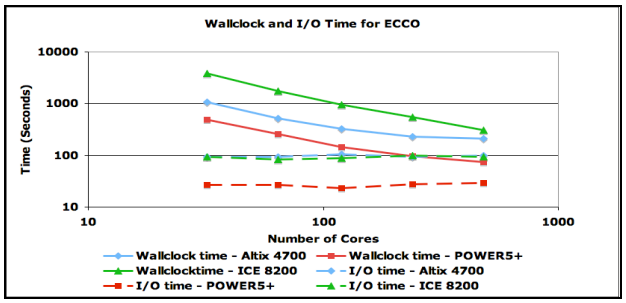


Figure 20. Wall-clock and I/O time for ECCO for Altix, POWER5+, and ICE.

Figure 21 shows the I/O write checkpoint bandwidth for ECCO. The average write bandwidth is about 84 MB/s on the Altix and 88 MB/s on the ICE system, and is about 5% of the peak theoretical value of 2 GB/s. On the POWER5+, it is about 300 MB/s and theoretical peak is 4 GB/s. The I/O in ECCO is performed by a package that provides a capability for writing single-record direct-access Fortran binary files. The reason for a low effective write I/O rate is that the application opens and closes dozens of files and I/O time includes the time for opening and closing the data and metadata files.

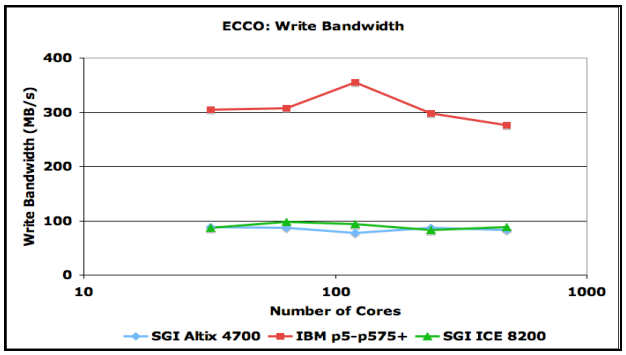


Figure 21. Write bandwidth for ECCO for Altix, POWER5+, and ICE.

D. Hybrid Benchmark and Application

In this sub-section we present the results for two hybrid (MPI+OpenMP) multi-zone compact applications, namely BT-MZ and SP-MZ, and for a hybrid application, OVERFLOW-2.

1) Hybrid Multi-zone Compact Applications

To examine performance response of the hybrid MPI+OpenMP programming model, we tested multi-zone versions of the NPBs on the three parallel systems. For a given number of cores, we ran the benchmarks in different process-thread combinations. We use the notation "N_m x N_o" to indicate the number of MPI processes (equal to the number of zone groups) used for the first-level parallelization and the number of OpenMP threads for the second-level parallelization within each zone group. The number of MPI processes is limited by the number of zones for a given problem size, while the number of OpenMP threads is limited by the number of cores available on an SMP node. The total Gflop/sec results reported by the benchmarks for the Class C problem from the best N_m x N_o combination for a given core count are included in Table II for the BT-MZ benchmark, and in Table III for the SP-MZ benchmark. Since a limited number of zones (=16) are defined for the LU-MZ benchmark and, thus, only a limited number of MPI processes can be used, we did not include results for this benchmark here. On the ICE system, we also ran the benchmarks in a *scaled* configuration, that is, only four of the eight cores in each node were used.

The best performance with multi-zone benchmarks is usually achieved by maximizing the number of zone groups, as long as the workload can be balanced. For Class C, the number of zones is 256 for both BT-MZ and SP-MZ. Due to the uneven zone sizes in BT-MZ, the optimal number of zone groups is 64, thus 64 MPI processes. Beyond that, multi-level parallelism from OpenMP threads is needed for additional performance gain. On the other hand, the equal-sized zones in SP-MZ allow efficient use of the zonal parallelism up to 256 MPI processes. In general, this is what we have observed in Tables 2 and 3. However, on the ICE system, the eight-way results show a preference of two OpenMP threads over one. For example, for the 32-core case, the 16 x 2 combination produces better results than 32 x 1. This is correlated with the very low latency within a node observed on the ICE system, as compared to other systems, showing the benefit of using OpenMP threads.

Overall, the Itanium2-based Altix system shows better performance for both BT-MZ and SP-MZ when the number of OpenMP threads does not exceed two. On 256 and 512 cores, BT-MZ requires 4 and 8 OpenMP threads respectively, and we observe good scaling on the POWER5+ and ICE systems, both having flat-memory SMP nodes. There is quick performance degradation on the Altix, which has a NUMA architecture. It points to the importance of low-latency, flat-memory SMP nodes for fine-grained parallelization like OpenMP. Lastly, we observe substantial performance improvement from the scaled configuration (4 cores per node) on the ICE system in comparison to the full configuration (8 cores per node): 10-20% for BT-MZ and 30-50% for SP-MZ. This can be explained by the limited memory bandwidth available for cores on the ICE system. The program actually runs faster on a given amount of hardware by leaving half the cores idle. The other two systems show much less impact.

2) Hybrid (MPI+OpenMP) OVERFLOW-2

We tested the hybrid MPI+OpenMP version of OVERFLOW-2 on the three systems. In Figure 22, we plot wall-clock time per step for hybrid OVERFLOW-2 on Altix, POWER5+, and ICE. ICE numbers are for either MPT or MVAPICH, whichever was better.

Each line of these figures in Figure 22 shows performance of the hybrid code for a fixed number of cores as the number (N_o) of OpenMP threads is varied. On the Altix, the best performance occurs for either one or two OpenMP threads. Beyond four OpenMP threads, performance degrades quickly for a given core count. On the POWER5+ it is beneficial to use OpenMP. The best results are obtained when the number of OpenMP threads is either two or four, and performance is

relatively constant throughout the available range of threads for a given core count. On ICE, performance generally degrades slightly as the number of OpenMP threads increases beyond one. Overall, the best results were obtained on the Altix system when the number of OpenMP threads (N_o) was either one or two; the worst results were obtained on the ICE system. The hybrid version shows the benefit of using OpenMP threads within an SMP node on the POWER5+, and outperforms a pure MPI version (i.e., when N_o is equal to one).

TABLE II. PERFORMANCE RESULTS OF NPB BT-MZ CLASS C BENCHMARK ON ALTIX, POWER5+, AND ICE SYSTEMS

Machine	**SGI Altix 4700**		**IBM POWER5+**		**SGI Altix ICE 8200**			
					8 cores per node		***4 cores per node***	
# Cores	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s
8	8 x 1	17	8 x 1	16	8 x 1	13	8 x 1	15
16	16 x 1	34	16 x 1	30	8 x 2	27	16 x 1	30
32	32 x 1	67	32 x 1	59	16 x 2	52	16 x 2	56
64	64 x 1	132	32 x 2	115	32 x 2	94	32 x 2	107
128	64 x 2	237	64 x 2	220	64 x 2	176	64 x 2	206
256	64 x 4	405	64 x 4	407	64 x 4	339	64 x 4	371
512	64 x 8	419	64 x 8	667	64 x 8	556	128 x 4	620

TABLE III. PERFORMANCE RESULTS OF NPB SP-MZ CLASS C BENCHMARK ON ALTIX, POWER5+, AND ICE SYSTEMS

Machine	**SGI Altix 4700**		**IBM POWER5+**		**SGI Altix ICE 8200**			
					8 cores per node		***4 cores per node***	
# Cores	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s	N_m x N_o	Gflop/s
8	8 x 1	12	8 x 1	10	4 x 2	8	8 x 1	10
16	16 x 1	25	16 x 1	20	8 x 2	16	16 x 1	20
32	32 x 1	50	32 x 1	41	16 x 2	26	32 x 1	40
64	32 x 2	107	32 x 2	84	32 x 2	58	64 x 1	79
128	128 x 1	241	128 x 1	161	64 x 2	113	128 x 1	169
256	256 x 1	490	256 x 1	321	128 x 2	224	256 x 1	390
512	256 x 2	723	256 x 2	622	256 x 2	560	256 x 2	608

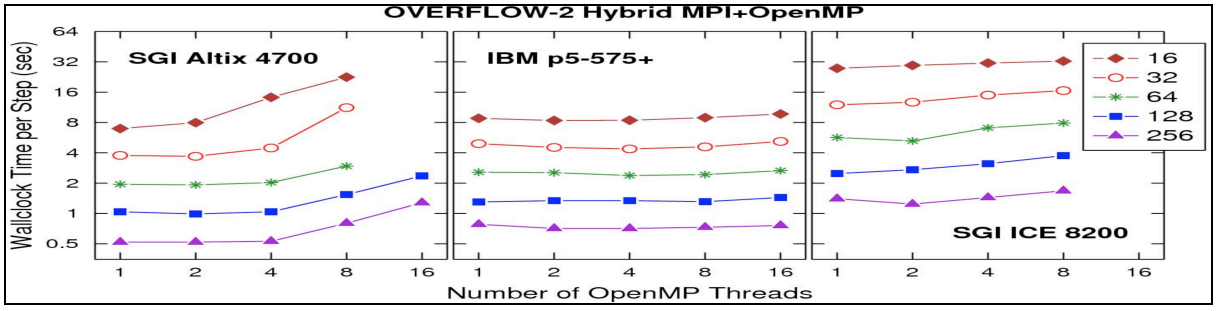


Figure 22. Wall-clock time per step as a function of number of OpenMP threads for Altix, POWER5+, and ICE systems.

The performance of hybrid OVERFLOW is strongly affected by two competing factors. The first factor is the number of MPI processes. As this number increases, the total number of grid points increases (due to grid splitting for load balancing producing extra points at splitting boundaries), leading to increased computational work for the flow solver. (see Table IV.) In addition, as the number of MPI processes increases, the total communication volume increases. The second factor is OpenMP overhead. This has both fixed and per-thread components. As the number of threads increases, the per-thread overhead grows relative to actual work performed.

So, if for a given number of cores the performance of hybrid OVERFLOW is best with some value of OpenMP threads that is greater than 1, this is a sign that the overhead due to OpenMP is more than compensated for by the reduction in computation time due to a smaller number of grid points,

TABLE IV. TOTAL NUMBER OF GRID POINTS AS A FUNCTION OF NUMBER OF DOMAIN GROUPS

Number of groups	Total no. of grid points (in millions)
16	37
32	38
64	41
128	43
256	47

and by a reduction in communication time due to a smaller total communication requirement. Conversely, if for a given number of cores the best performance is without OpenMP, this is a sign that the overhead and possible inefficiency due to

OpenMP outweigh the extra computation and communication costs of not using more processes.

E. Multi-Core Effects on the SGI ICE 8200

In this section, we present the results of three applications (CART3D, ECCO, and USM3D) on a subset of the cores in the ICE system to measure impact of limited memory bandwidth.

We ran all three applications on 1, 2, 4, and 8 cores per node on the ICE system. To review, each node contains two Xeon Intel Quad-Core 64-bit processors (8 cores in all) on a single board, as an SMP unit. The core frequency is 2.66 GHz and supports 4 floating-point operations per clock period with a peak performance of 10.6 Gflop/s/core or 42.6 GFlop/s per node. Each node contains 8 GB of memory. The memory subsystem has a 1,333 MHz FSB, and dual channels with 533 MHz Fully Buffered DIMMS. Both processors share access to the memory controllers in the memory controller hub (MCH or North Bridge).

In Figure 23, we plot the wall-clock time per step for CART3D using 1, 2, 4, and 8 cores per node. (The per-node memory of 8 GB was not enough for 8 processes per node for the 256 and 512 core cases.) Up to 128 cores, performance is highest for one core and then successively worse for 2, 4 and 8 cores. The reason for this is that when all eight cores of a node are used, two processes share each L2 cache and four processes share each FSB. When only four cores of a node are used, then the L2 caches are private, but each FSB is still shared by two cores. When just two cores of a node are used, then each core has its own set of memory resources, but the two cores share the interconnect with the rest of the system. However, when only one core is used, then it has a full 4 MB of L2 cache, a full FSB bus, and the full InfiniBand host channel adapter (HCA) by itself. In summary, there is significant performance degradation due to sharing of memory resources. The performance difference is highest at 32 cores, and doubling the number of cores reduces the performance by half, which is due to reduction in the memory bandwidth by half. The performance difference decreases as the total number of cores increases because, for a large number of cores, communication becomes more important.

Wall-clock time per step for ECCO is plotted in Figure 24. Qualitatively, the performance is almost the same as that of CART3D and for the same reasons.

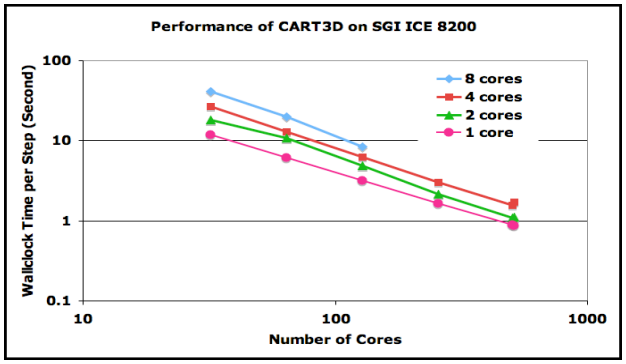


Figure 23. Performance of CART3D on various cores of the ICE system.

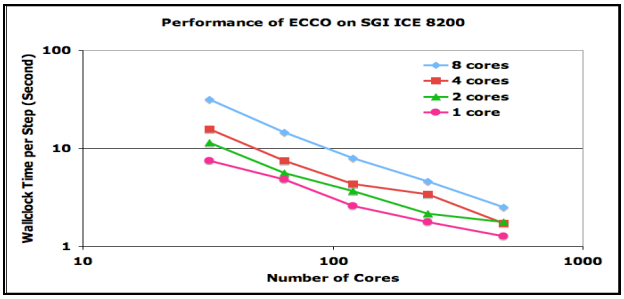


Figure 24. Wall-clock time per step of ECCO on various cores of the ICE.

In Figure 25 we plot the cycle wallclock time per step for USM3D. Qualitatively the performance is almost the same as that of CART3D and for the same reasons.

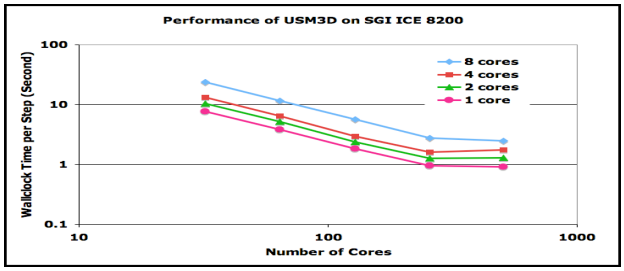


Figure 25. Wall-clock time per step of USM3D on various cores of the ICE.

V. Summary and Conclusions

Our experiments show that for a large number of processors—beyond 128—the performance of MVAPICH is about 10% better than SGI MPT on the ICE system and in some cases, 300% better than the MPT library.

On the ICE system, multi-core performance is very application-dependent. In most cases, leaving cores idle improves performance. For some cases, particularly at higher process counts, using fewer cores per node is not beneficial due to the increase in communication overhead relative to the computation. If the "cost" of idle resources is taken into account, at lower processor counts, in most cases, using all cores yields better performance. At higher processor counts, using 4 cores per node yields a better return.

Memory-bound applications such as ECCO and USM3D do better on the POWER5+ system, particularly for small numbers of processors. OVERFLOW, although memory-bound, does not perform better on the POWER5+ due to compiler issues. ECCO and USM3D are latency-bound at higher processor counts and do not scale on all systems. For large numbers of processors—especially 256 and 512 processors—the performance range narrows across the systems due to increased importance of network communication (latency, bandwidth).

Our experiments show that performance of tested hybrid codes is sometimes the same, but usually inferior, to pure MPI. This was a surprise since we had expected OpenMP to perform well within an ICE node. In fact, on the ICE system, performance of the hybrid model was lower than that of MPI.

To obtain good performance with hybrid OVERFLOW, the following two conditions must hold: It is necessary to have a very low-overhead implementation of OpenMP due to the fine granularity of the OpenMP parallelism; and it is also necessary to have good control over process and data placement, due to the inefficiency that holds if the data required by a processor are not local to that processor. We believe the POWER5+ implementation of OpenMP has low overhead, while the Intel implementation may suffer in this respect. The process placement tools available on the Altix and POWER5+ seem sufficient, while process placement on ICE is not as refined.

Among the three systems studied, the ICE system's MPI latency is smallest within a node. However, latency increases rapidly when communication involves two to four nodes, and then the increase in latency is slower and more gradual. Additionally, interconnect bandwidth is smallest for the ICE system. As a result, the ICE cluster has the smallest bisection bandwidth, and codes based on FFT, which involve all-to-all communication, will not perform or scale well. Within a node of ICE, performance of MPT is better than that of MVAPICH. However, beyond 8 cores, the performance of MVAPICH is better than that of MPT. This is reflected in performance of all HPCC benchmarks (GUPS) and several NPBs (LU), where performance is 3 times that of MPT.

For consistently good performance on a wide range of processors, a balance between processor performance, memory subsystem, and interconnects (both latency and bandwidth) is needed. Overall, for our applications, we found that the performance of POWER5+ is more balanced with respect to these attributes. Its performance is better than Altix and ICE. We also found that ICE is not balanced, as its memory sub-system cannot adequately feed data to the floating-point units. Also the interconnect performance of ICE is very poor. The performance on benchmarks of the SGI supplied MPT library on ICE is very poor relative to MVAPICH, especially on large number of cores. However, with tested production applications, the difference is not significant. Performance of the Altix is between POWER5+ and ICE, except at higher processor counts, where the superior performance of the NUMAlink network allows the Altix to outdo the POWER5+.

ACKNOWLEDGMENT

We gratefully acknowledge Holly Amundson for her assistance in carefully reading and formatting the manuscript.

REFERENCES

[1] Dunigan, T.H., Jr. Vetter, J.S., Worley, P. H. Performance evaluation of the SGI Altix 3700: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1488619

[2] Rupak Biswas, M. Jahed Djomehri, Robert Hood, Haoqiang Jin, Cetin Kiris, and Subhash Saini, An Application-Based Performance Characterization of the Columbia Supercluster, Proceedings of the 2005 ACM/IEEE conference on Supercomputing, Seattle, Washington, Nov. 12-18, 2005.

[3] Subhash Saini, Dennis C. Jespersen, Dale Talcott, Jahed Djomehri and Timothy Sandstrom, Early Performance Evaluation of SGI Altix 4700 Bandwidth Version, 7th International Workshop Performance Modeling, Evaluation, and Optimization of Ubiquitous Computing and Networked Systems (PMEO-UCNS'2008) in Proceedings of 22nd IEEE IPDPS April 14-18, 2008, Miami, Florida USA.

[4] Subhash Saini, Dennis C. Jespersen, Dale Talcott, Jahed Djomehri, and Timothy Sandstrom, Application Based Early Performance Evaluation of SGI Altix 4700 Systems, ACM International Confernce on Computing Frontiers, May 5-7, 2008, Ischia, Italy.

[5] Subhash Saini, Dale Talcott, Timothy Sandstrom, Dennis C. Jespersen, Jahed Djomehri and Rupak Biswas, Performance Comparison of SGI Altix 4700 with IBM POWER5+ and POWER5 Clusters, The International Supercomputing Conference (ISC), Dresden, Germany, June 17-20, 2008.

[6] Adolfy Hoisie, Greg Johnson, Darren J. Kerbyson, Michael Lang, Scott Pakin, A performance comparison through benchmarking and modeling of three leading supercomputers: Blue Gene/L, Red Storm, and Purple, ACM/IEEE Proceedings of Conference on High Performance Networking and Computing, SC 2006, Article 74, Tampa, Florida, USA.

[7] ASC Purple System based on single-core IBM POWER5: http://www.llnl.gov/asc/platforms/purple/configuration.html

[8] L. Oliker, A. Canning, J. Carter, C. Iancu, M. Lijewski, S. Kamil, J. Shalf, H. Shan, E. Strohmaier, S. Ethier, T. Goodale, Scientific Application Performance on Candidate PetaScale Platforms, Proceedings of International Parallel & Distributed Processing Symposium (IPDPS), Long Beach, California 2007.

[9] D. Bailey, J. Barton, T. Lasinksi, and H. Simon, The NAS Parallel Benchmarks, NAS Technical Report RNR-91-002, NASA Ames Research Center, 1991; NAS Parallel Benchmarks.

[10] Subhash Saini, Robert Ciotti, Brian T. N. Gunney, Thomas E. Spelce, Alice Koniges, Don Dossa, Panagiotis Adamidis, Rolf Rabenseifner, Sunil R. Tiyyagura, Matthias Mueller: Performance Evaluation of Supercomputers using HPCC and IMB Benchmarks., Journal of Computational System Sciences, 2007, Special issue on Performance Analysis and Evaluation of Parallel, Cluster, and Grid Computing Systems; HPCC, HPC Challenge Benchmarks: http://icl.cs.utk.edu/hpcc/

[11] SGI Altix 3700 Bx2 Servers and Supercomputers: http://www.sgi.com/pdfs/3709.pdf

[12] SGI Altix 4700: http://www.sgi.com/products/servers/altix/4000/

[13] IBM POWER Architecture: http://www-03.ibm.com/chips/power/index.html; http://www.research.ibm.com/journal/rd/494/sinharoy.html

[14] IBM pSeries High Performance Switch: www.ibm.com/servers/eserver/pseries/hardware/whitepapers/pseries_hps_perf.pdf

[15] Quad-Core Intel Xeon Processor 5300 Series, Features: http://www3.intel.com/cd/channel/reseller/asmo-na/eng/products/server/processors/q5300/feature/index.htm

[16] InfiniBand Trade Association: http://www.infinibandta.org/home,

[17] H. Jin and R.F. Van der Wijngaart, Performance Characteristics of the Multi-Zone NAS Parallel Benchmarks, Journal of Parallel and Distributed Computing, Special Issue, ed. B. Monien, Vol. 66, No. 5, p674, 2006.

[18] OVERFLOW-2: http://aaac.larc.nasa.gov/~buning/

[19] Dimitri J. Mavriplis, Michael J. Aftosmis, Marsha Berger, High Resolution Aerospace Applications using the NASA Columbia Supercomputer, Proceedings of the 2005 ACM/IEEE conference on Supercomputing, Seattle, Washington, Nov. 12-18, 2005.

[20] USM3D: http://aaac.larc.nasa.gov/tsab/usm3d/usm3d_52_man.html

[21] ECCO: Estimating the Circulation and Climate of the Ocean: http://www.ecco-group.org/

[22] TOP500 Supercomputing Sites: http://www.top500.org/

[23] Parallel Kernels and Benchmarks (PARKBENCH): http://www.netlib.org/parkbench/

[24] GUPS (Giga Updates Per Second): http://icl.cs.utk.edu/projectfiles/hpcc/RandomAccess

[25] STREAM: Sustainable Memory Bandwidth in High Performance Computing: http://www.cs.virginia.edu/stream/

[26] Daisuke Takahashi, Yasumasa Kanada: High-Performance Radix-2, 3 and 5 Parallel 1-D Complex FFT Algorithms for Distributed-Memory Parallel Computers. Journal of Supercomputing, 15(2): 207-228, Feb. 2000.