

Scalability of a tera-scale linux-based clusters for parallel *ab initio* molecular dynamics applications

Hongsuk Yi, Jungwoo Hong, Hyoungwoo Park & Sangsan Lee
Supercomputing Center, Korea Institute of Science and Technology Information, Korea

Abstract

KISTI supercomputing center has initiated a TeraCluster project to build a linux-based cluster with tera-flops performance. The main goal of the project is to provide a resources composed of PC clusters that meet the level of computing power required by the grand challenge applications in Korea. In the beginning of 2002, we have built a prototype of TeraCluster with 128 computing nodes, Phase-I cluster, from the performance evaluations of KISTI grand challenge applications. In particular, we have investigated the performance and scalability of *ab initio* molecular dynamics and quantum chromodynamics (QCD) applications on the Phase-I and compare with the Cray T3E, other PC cluster as well as IBM-POWER4 system. As a results, the QCD application shows an excellent parallel scalability for all the tested machines. While the parallel performance of the *ab initio* molecular dynamics applications does not scale at all due to the collective communication among computing processors. From the perfor-

Table 1: The hardware and software components of four different PC-based platforms and the basic results of NPB and PMB benchmark suite.

System	DS10	UP2000	Cray T3E	Phase-I
CPU architecture	EV6	EV6	EV5	Pentium-4
Network	Fast Ethernet	Myrinet	FDDI	Myrinet
CPU clock (MHz)	466	667	450	1700
Number of nodes	64	64	128	128
R_{peak} (Gflop/s)	59.6	85.3	108	435.2
R_{max} (Gflop/s)	21.0	43.6	82.0	208.3
Latency (μs)	140.0	13.0	1.3	12.5
Bandwidth (MB/s)	8.9	139.7	157.0	65.1

mance evaluation on the Cray T3E and other clusters as well as the IBM-POWER4 system, the Phase-I cluster is found to be a good platform for the *ab initio* molecular dynamics applications.

1 Introduction

KISTI supercomputing center has built two different kind of PC clusters in order to design a tera-flops cluster[1] in the beginning of 2000. Each 64 computing nodes clusters are composed of the DS10 board with 466 MHz Ev6 and UP2000 board with 667 MHz Ev6 Alpha processors. The inter-node communication of the DS10 and UP2000 systems are connected with Fast Ethernet and Myrinet, respectively. Each computing nodes have 250 MB memory and 20 GB local IDE disk, while the server node has 128 MB memory and 30 GB disk. The systems operate on Linux Redhat 6.0 with a kernel level of 2.2.0. The MPICH is used as a MPI communication library, and portable batch system is selected as a queuing system. The performance evaluations on these clusters have been a success story with the collaboration of a few industrial partners in Korea[2]. From the performance evaluations of the major KISTI applications[3], we can design the first phase of the TeraCluster with 128 computing nodes. Here we denote the first phase of TeraCluster as the Phase-I cluster which is based on PCs, each with a Pentium-4 1.7 GHz processor

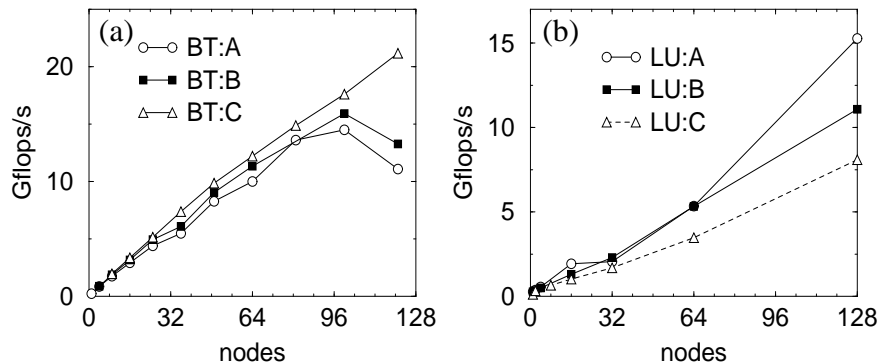


Figure 1: The sustained NPB performance of (a) BT and (b) LU benchmarks for three different problem size: The size of 64^3 , 102^3 , and 162^3 is used for class A, class B, and class C, respectively.

and 1GB RDRAM memory. This system with its Myrinet communication represents a good balance between CPU power and network bandwidth. The sustained performance of R_{max} reaches 208.3 Gflops and details are summarized in Table I.

2 Performance results

2.1 NPB Benchmark results

We present implementations and results of the NAS Parallel Benchmarks (NPBs) 2.3 suite[4]. The suite is based on Fortran 77 and MPI, and currently consists of eight benchmark programs. Here we describe implementations of BT and LU as well as CG with three different problem size. It is worthwhile to note that BT (block tridiagonal) contains systems of equations resulting from an approximately factored implicit finite difference discretization of the Navier-Stokes equations. Especially, BT solves three sets of uncoupled tridiagonal block matrix. The code requires a square number of processors so performance results are obtained for 4, 9, 16, 25, 36, \dots , etc.

Fig. 1(a) shows the performance of BT for three class of problem size on the Phase-I. The class C of the large problem size gives very good performance and scalability relative to other two

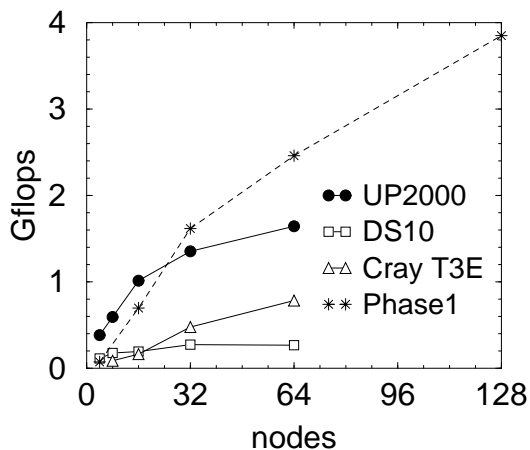


Figure 2: The total performance of the CG class B with different cluster configurations and on the Cray T3E.

problem sizes, obtaining as much as 21.1 Gflops at the 121 processors. The class A of the small size is poorest, with between 96 and 216 Mflops per processor. The performance of the class A and B does not scale well when going from 81 to 121 processors. This result implies that the inter-node communication performance is relatively poor compared to their floating point performance.

Fig. 1(b) shows the performance of LU factorization for three different problem sizes. LU factorization requires that the number of processors must be a power of two so performance results are obtained for 2, 4, 8, 16, \dots , etc. The result is very similar to that shown in Fig. 1(a), except that it is easier to see the scalability. In particular, the class A shows best performance and scalability, followed by the class B and at last the class C. This result indicates that LU benchmark is very sensitive to the small message passing performance of an MPI implementation.

Fig. 2 shows machine performances for conjugate gradient (CG) with the class B on the UP2000, DS10, and Phase-I as well as Cray T3E. CG solves an unstructured sparse linear system by the conjugate gradient method and uses the inverse power method to find an estimate of the largest eigenvalue of a symmet-

ric positive definite sparse matrix. This code requires that the number of processors must be a power of two. It is clear that the Phase-I is the clear leader, obtaining as much as of 3.8 Gflops at the 128 computing nodes. The UP2000 with Alpha 21264-667 MHz processor is faster than the Phase-I with Pentium-4 1.7GHz processor in computing the CG sampling problem within the small cluster size up to 16 processors. However, this advantage is lost on the larger cluster sizes. Indeed, the 32 processor Phase-I outperforms the 32 processor UP2000 cluster. In addition, the DS10 system does not scale at all when going from 16 to 64 processor due to the relatively poor latency and bandwidth performance as summarized in Table I.

2.2 Lattice quantum chromodynamics applications

Lattice quantum chromodynamics (QCD) is a method of studying the strong nuclear force. One of major obstacles in solving the lattice QCD on parallel computers is that calculations of the quark interactions require very intensive computation for a highly non-local matrix determinant. Indeed, the performance of QCD code is strongly depend on the communication performance between inter-node rather than the performance of multiplication of 3×3 complex matrices. Since the communication pattern of the code is point-to-point communication between the nearest neighbor processors, this code is almost ideally suited to parallel computation.

In Fig. 3(a), we illustrate the execution times of the quenched QCD application on the UP2000 and DS10 clusters as well as the Phase-I for the problem size of $L = 12^3 \times 10 \times np$, where np is the number of processors. In this approach, the ratio of the calculational task per processor is fixed, by increasing the problem size along with the number of processors. The execution times of the UP2000 and Phase-I clusters are 2 and 3 times faster than that of the DS10 cluster for the single and 64 processors, respectively. In the DS10 system, the execution time unavoidably increases with the number of processors, indicating that the available message passing system on the DS10, MPI over TCP/IP on a Fast Ethernet network, is not as scalable. However, the linear increase

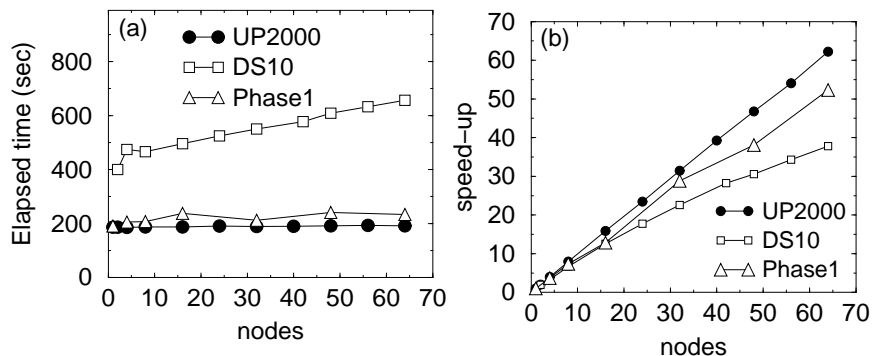


Figure 3: (a) The sustained performance and (b) the corresponding speed-up of the quenched QCD for three different machines.

observed on the UP2000 and the Phase-I system is probably very close to ideal for our parallel implementation.

To elucidate the scalability of the QCD application we calculate the speed-up $\mathcal{S} = T_1/T_P$, where T_1 and T_P is the execution time for single and number of P processors, respectively. Fig. 3(b) shows the obtained \mathcal{S} on the different three PC clusters. It can be seen that a nearly perfect scaling is achieved in this application with the UP2000 cluster up to 64 processors. In addition, UP2000 cluster outperforms the Phase-I system as well as the DS10 cluster. The DS10 cluster also has good scaling up to 16 processor. However, the scalability of the DS10 cluster is rapidly suppressed above the 32 processors due to the relatively poor capacity of the bandwidth and the latency. This result implies that the capacity of bandwidth is crucial for the large scale computation system as a scalable scientific platform.

2.3 *ab initio* molecular dynamics applications

The *ab initio* total energy pseudopotential calculations have been presented many interesting physical properties of new materials. We parallelize the *ab initio* plane wave pseudopotential code by spreading the wavefunctions in both reciprocal space and real space using three-dimensional (3D) fast Fourier transformation (FFT). In order to increase the efficiency of the parallel per-

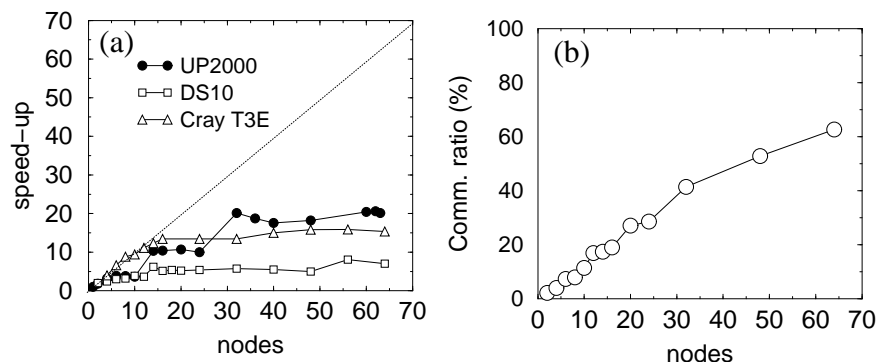


Figure 4: (a) Speedup of the *ab initio* molecular dynamics for the UP2000 and DS10 as well as Cray T3E system, and (b) communication overhead for the *ab initio* application on the Cray T3E system.

formance and scalability, we divide the 3D FFT mesh into the number of computing nodes with the distinct regions of equal volume and assign a single such region to each working processor. In particular, we divide the real space into a set of two-dimensional slabs and assign each layer to the working processors through the cyclic distribution[5]. This method of distribution allows us to minimize the amount of data communication involved in the 3D FFT, while the cyclic distribution leads to a load imbalance for computations.

In this study, we calculate a new solid phase of C_{36} fullerene. The super-cell approximation is used to simulate the periodic boundary condition and the cell size is $11 \times 11 \times 11 \text{ \AA}^3$ for C_{36} . Since C_{36} is molecule we use a single k -point in the calculation. In Fig. 4(a), we illustrate speed-up of the C_{36} fullerene on the UP2000, DS10, and Cray T3E system. As is shown in Fig. 4(a), *ab initio* application achieves poor speed-up on all the tested systems. Among these systems, the Cray T3E seems to be retained the linear scalability up to 16 nodes because the performance of the collective communication is dominated by the system latency. However, the final performance on the Cray T3E system is unsatisfactory. Above 32 processors, the performance improvement of the UP2000 and DS10 clusters as well as Cray T3E are saturated due to the frequent use of FFT subroutine.

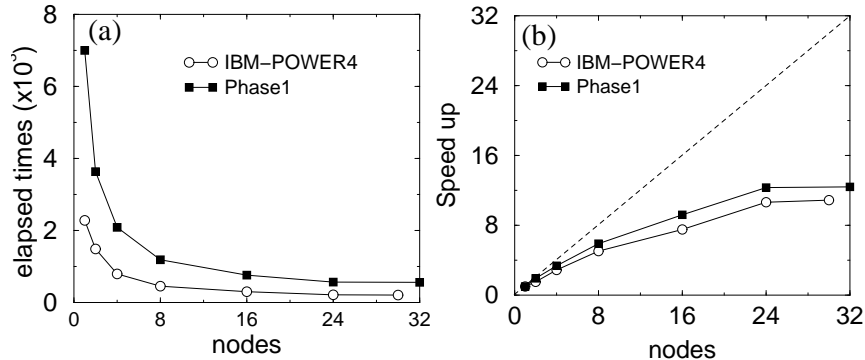


Figure 5: Elapsed times (a) and speedup (b) for the *ab initio* molecular dynamics on the Phase-I cluster and IBM-POWER4 system.

We note that the poor scalability on all tested systems is the intrinsic properties of *ab initio* application and is also attributed to the small problem size of C_{36} fullerene. It is also worthwhile to note that the performance of computation grows more rapidly than the amount of communication as the problem size increases. Fig. 4(b) shows communication overhead of the code on the Cray T3E system. It is clearly seen that the communication overhead becomes dominant as the number of processors increases. This result implies that the parallel *ab initio* application is highly scalable as long as the system latency becomes high quality.

VASP code is another *ab initio* molecular dynamics application[6] based on the density functional theory within the local density approximation. This code also uses the parallelized 3D FFT library. Most of the communication is done by `MPI_Alltoall` routines. The AlSb(001) surface is simulated by repeated slabs containing eight layers of AlSb. The electronic wave functions are expanded in plane waves up to a kinetic energy of 156 eV and the momentum-space integrations are performed using $2 \times 2 \times 1$ Monkhorst-Pack \mathbf{k} -point. All atoms are represented by ultrasoft pseudopotentials as provides with VASP[7]. In the AlSb(001) surface, the problem size is about 3 times larger than that of Fig. 4(a). In Fig. 5(a), we present the performance evaluations of the *ab initio* molecular dynamics using the VASP package on the

Phase-I and compare them with the IBM-POWER4 (1.3GHz) system. As seen in Fig. 5(a), the IBM-POWER4 outperforms the Phase-I cluster clearly. In particular, a single POWER4 with 5.2 Gflops per processor is 3 times faster than the Pentium-4 with 3.4 Gflops per processor. Fig. 5(b) shows the speed-up for the AlSb(001) surface. The VASP application achieves better speed-up on the Phase-I cluster than the IBM-POWER4, implying that the scalability of AlSb(001) surface on the Phase-I cluster is satisfactory.

2.4 Summary

We have investigated the parallel performance and scalability of the *ab initio* scalable scientific applications encountered in quantum physics on the Phase-I as the first phase of the TeraCluster cluster and IBM-POWER4 system. We found that the quenched QCD application is a very numerically intensive code due to a high degree of parallelism through the nearest neighbor communication. The performance and scalability of the *ab initio* molecular dynamics applications does not scale well due to the frequent use of the collective communication library of the 3D FFT. From the performance analysis of the *ab initio* applications, it is clear that the Phase-I cluster offers a reasonable and affordable parallel platform.

Acknowledgments

H.Yi appreciate Dr. S. Han, K. Lee and Prof. S. Kim for useful discussions and their valuable contributions.

Reference

- [1] TeraCluster Project (<http://cluster.or.kr>).
- [2] The work reported here has been performed as a part of the KISTI TeraCluster collaborative project with Compaq Korea, Linux One, Samsung Corp., Samsung Electronics, and Zion

Linux Systems.

[3] We have five grand challenge applications involving the computational fluid dynamics, structural analysis, computational chemistry, lattice quantum chromodynamics, and *ab initio* molecular dynamics applications.

[4] <http://www.nas.nasa.gov/NAS/NPB>.

[5] L. Clarke, I. Stich, and M.C. Payne, Computer Physics Communications **72**, 14 (1992).

[6] G. Kresse, and J. Hafner, Phys. Rev. B **47**, 558 (1993); G. Kresse, and J. Furthmüllerr, Phys. Rev. B **54**, 11 169 (1996).

[7] D. Vanderbilt, Phys. Rev. B **41**, 7892 (1990); G. Kresse, and J. Hafner, J. Phys. Condens. Matter **6**,8245 (1994).