Computing

DL/SCI/TM99T

technical memorandum Daresbury Laboratory

DL/SCI/TM99T

LENDING COPY

WORKSTATION CLUSTERS FOR SCIENTIFIC APPLICATIONS: AN ASSESSMENT OF VARIOUS NETWORKS

by

A.P. RENDELL, R.J. ALLAN, M.P. BURROW, D.R. EMERSON, M.F. GUEST, G.Y. GUO, N.M. HARRISON, V.R.SAUNDERS, P. SHERWOOD and R.R. WHITTINGTON, SERC Daresbury Laboratory

JANUARY, 1994

94/19

Science and Engineering Research Council
DARESBURY LABORATORY
Daresbury, Warrington WA4 4AD

ISSN 0144-5677

IMPORTANT

The SERC does not accept any responsibility for loss or damage arising from the use of information contained in any of its reports or in any communication about its tests or investigations.

Science and Engineering Research Council,
Daresbury Laboratory.

DRAFT: DiscoTech 93/3

Workstation Clusters for Scientific Applications: An Assessment of Various Networks:

A.P. Rendell, R.J. Allan, M.P. Burrow, D.R. Emerson, M.F. Guest, G.Y. Guo,
N.M. Harrison, V.R. Saunders, P. Sherwood, and R.R. Whittington

Theory and Computational Science Division,
S.E.R.C. Daresbury Laboratory,
Warrington WA4 4AD, UK

Abstract

We report the results of running a variety of scientific application codes on a cluster of HP and a cluster of IBM workstations. The HP cluster comprises 4 HP720 machines linked via Ethernet and Fibre Distributed Data Interface (FDDI), while the IBM cluster comprises 3 IBM 530H workstations linked via Ethernet and IBM's proprietary Serial Optical Channel Converter (SOCC). The latency and bandwidth of each network has been measured. The former is found to be roughly equivalent (2-3msec) for the three different interconnects, although it decreases to about 1msec if the SOCC is used with IBM's optimized version of the Parallel Virtual Machine (PVM) message passing harness. The measured bandwidth of FDDI and SOCC are below 30% of their quoted peaks, which is attributed to limitation in the workstation/network interface.

For application codes with a high compute to communicate ratio, FDDI or SOCC has little to offer over a dedicated ethernet link. For applications requiring high bandwidth communications (e.g. two-electron integral transformation quantum chemistry) FDDI and SOCC are of considerable benefit. Applications requiring many short messages show some benefit from the increased bandwidth, but this is tempered by latency considerations.

1 Introduction

There is considerable current interest in running parallel scientific application codes on workstation "clusters". Such clusters are formed by linking together a number of UNIX based workstations using a local area network (LAN). Communication between processes running on the different workstations is achieved via message passing, and to this end a number of communication harnesses have been developed, e.g. FORTNET[1], PVM[2], TCGMSG[3] etc.

In comparison to more traditional parallel computers the workstation cluster approach has some advantages:

1. The system can be configured according to the specifications of a particular application, e.g. large memory to disk ratio;
2. The entry level price is low and the system can easily be extended as funding becomes available;
3. Input/Output can be performed in a truly parallel fashion if each machine is configured with a local disk;
4. Machines need not be dedicated to the cluster, e.g. a machine may only form part of the cluster overnight when it is not used for other functions.

For many applications, however, the viability of such a system is critically determined by the performance of the LAN. Traditionally this has been Ethernet achieving transfer rates of about 1MB/sec., but recently a variety of new networks have become widely available. The manufacturers quoted bandwidth for several network technologies is given in table 1.

Table 1: Network Speeds

Technology	Speed (Mbits/sec)
Ethernet	10
FDDI	100
IBM SOCC	220
UltraNet	250
HiPPi	800 and 1600
Fibre Channel	266

In this paper we investigate the performance of Ethernet, FDDI and SOCC for a number of parallel scientific application codes. Two clusters are used and the application codes cover a wide spectrum of computational science. In the next section we outline the configuration of the clusters and compare the single node performance, as measured using a number of simple kernel codes, with some other processors. The communication rates achieved for the different networks is discussed in section 3. Section 4 considers some application kernels, while section 5 discusses some full application codes. Conclusions concerning the general applicability of workstation clusters for scientific computing is given in section 6.

2 Configuration and Single Node Performance

For the following benchmarks two clusters of workstations have been used. These are as follows:

1. **HP720 cluster:** this cluster comprises four HP 9000/720 workstations each with 64MByte of memory and an internal 400MByte hard disk. The machines were running HP UX8.07 and were linked via twisted-pair Ethernet and FDDI. Both networks form part of the Daresbury LAN, although currently there is significantly less traffic on the FDDI network.
2. **IBM530H cluster:** this cluster comprises three IBM RS6000 530H workstations each with 48MByte of memory and an internal 1GByte hard disk. The machines were running AIX 3.02 and were linked via Ethernet and SOCC. The Ethernet connection was part of the LAN while the SOCC was a dedicated point to point connection between each IBM.

To assess the relative performance of the different workstations we have run two single node benchmark codes, and compared the results with those obtained on a variety of other processors. The results of these benchmarks (table 2) show that these two processors have roughly equivalent performance.

Table 2: Single Node Performance Comparison

	MMO[a]	DIAG[b]
HP 9000/720	8.5	31.9
IBM RS6000/530H	6.9	26.4
iPSC/860	21.2	62.3
CONVEX C-3840	5.2	36.0
CRAY Y-MP/464	1.5	16.0

[a] Time in seconds for benchmark based on matrix multiplication.
[b] Time in seconds for benchmark based on matrix diagonalization.

3 Network Communications

Table 1 lists the network speeds for a variety of interconnect technologies. These rates are however theoretical, in that they ignore the interface which the vendor must provide to link their backplane bus to the network, and neglect software aspects associated with providing a robust communication protocol. Furthermore, in addition to the bandwidth the latency or time taken to send a zero length message from one processor to another, is critical in determining the suitability of a particular network for a given application.

In the following table we list the peak bandwidth and latency obtained on the different networks using a variety of communication harnesses. The rates are given by the message length in bytes divided by half the elapsed time required to send data to and receive it back from a remote machine. Peak bandwidths are the asymptotic rates obtained as the amount of data transferred increases, while the latency is given as half the elapsed time obtained when the message length is extrapolated to zero bytes.

Table 3: Peak Bandwidths and Minimum Latency Obtained Using a Variety of Harnesses

Harness	Network	Machine	Latency (mSec)	Bandwidth (Mbytes/sec)
TCGMSG[a]	Ethernet	HP720	2.3	0.9
PVM[b]	Ethernet	HP720	1.2	0.6
TCGMSG	FDDI	HP720	2.4	3.5
PVM	FDDI	HP720	1.4	0.9
TCGMSG	Ethernet	IBM530H	3.3	0.9
TCGMSG	SOCC	IBM530H	3.3	5.7
PVMe[c]	SOCC	IBM530H	0.9	5.3

[a] TCGMSG Theoretical Chemistry Group Message Passing System developed at Argonne National Laboratory and Pacific Northwest Laboratory. The harness is based on UNIX TCP/IP sockets.
[b] PVM Parallel Virtual Machine developed at Oak Ridge National Laboratory. Version 2.3 was used, which communicates between daemons on each machine using UNIX UDP sockets.
[c] PVMe is a modification of PVM 2.3 performed by IBM ECSEC. It aims to optimize PVM for the SOCC.

On the LAN both the observed bandwidth and latency are critically dependent on other activities within the laboratory. In an attempt to compensate for this the numbers given represent the best results obtained after running the benchmark several times on different days.

The most striking aspect of these results is that the performance of FDDI and SOCC is substantially below the quoted peak performance given in table 1. Thus whilst Ethernet is capable of reaching over 90% of its theoretical peak, FDDI on the HP and SOCC on the IBM reach only 28 and 18% of theirs respectively. In the case of FDDI on the HP this disappointing performance was attributed mainly to hardware restrictions on their EISA FDDI interface board. As a consequence the FDDI interface on the new HP735 has been radically redesigned and is now reported to achieve transfer rates of approximately 7-8MBytes/sec (measured using PVM).

The measured latency of FDDI or SOCC under TCGMSG appear to be comparable with Ethernet. On the SOCC, however, there is considerable benefit in using PVMe, where the latency is roughly a quarter that obtained with Ethernet. Thus for applications where communications take place through a series of short messages there may be little advantage in using FDDI or SOCC (when not using PVMe) over Ethernet.

4 Matrix Kernels

There a number of parallel matrix kernels which are common to several application areas. We have taken two such kernels, ported them to the HP and IBM workstation clusters and tested them using the different networks. The different kernels are as follows and have markedly different communication requirements:

1. **Linear Equation Solver.** In this kernel we solve a linear equation iteratively. Parallelism is achieved by effectively distributing the large matrix vector product across the processors. Communication between processors is minimal and the application would be expected to scale reasonably well. The dimension of the benchmark problem is 1500;
2. **Matrix Diagonalization.** We have benchmarked the EISCUBE routine for diagonalization of real symmetric matrices. This routine is effectively a distributed memory parallel version of the corresponding EISPACK routine. The matrix and resulting eigenvalues are distributed by row around the processors. The bulk of the communication involves messages whose length in bytes is equal to the dimension of the matrix times the length of a double-precision number. The dimension of the benchmark problem is 1024.

Table 4: Speed-up obtained for matrix kernels using the TCGMSG harness

Number of Processors	HP720		IBM530H	
	Ethernet	FDDI	Ethernet	SOCC
Linear Equation Solver				
2	1.89	1.97	1.74	1.91
3	2.71	2.88	2.40	2.56
Matrix Diagonalization				
2	1.71	1.74	1.37	1.41
3	3.65	3.74	2.31	2.35
4	3.90	4.00		

In table 4 we report the speed-up (elapsed time on N processors/elapsed time on 1 processor) obtained for the different kernels. It is interesting to note that for the matrix diagonalization on the HP it is possible to obtain a speed-up greater than 3 when using only 3 processors. This is due to cache effects, which result from the nature of the parallel algorithm; the stride used in certain vector operations decreases as the number of processors increases and hence the number of cache misses can dramatically decease as the number of processors increases. In spite of this, it is still meaningful to compare the speed-ups obtained on Ethernet with those obtained using FDDI or SOCC. These differences are relatively small, indicating that the communication patterns in these algorithms will not exploit the additional capacity of the faster networks.

For the distributed diagonalization we have investigated the scaling of the algorithm with problem size. Using TCGMSG and FDDI on the HPs we have progressively increased the dimension of the matrix to be diagonalized. In figure 1 we plot the elapsed times verses the matrix dimension for 1-4 HP720 workstations. The results show that for small matrices (dimension less than 150) using multiple HPs degrades performance. Only when the matrix dimension exceeds 300 is it advantageous to use all 4 workstations.

5 Scientific Applications

In our initial assessment of the potential of workstation clusters for scientific applications we have ported a variety of application codes to the HP and/or IBM cluster. The codes and a brief description of the benchmark is given below.

Table 5: Summary of Application Speed-up Obtained Using the TCGMSG Harness

Application	3×HP720		3×IBM530H	
	Ethernet	FDDI	Ethernet	SOCC
Direct SCF	2.71	2.71	2.95	2.98
CRYSTAL	2.41	2.57	2.76	2.86
Integral Trans.	1.71	2.17		
Amber (MINIM)	1.89	2.13	1.87	2.05(2.55)[a]
Amber (GIBBS)	2.13	2.18	2.09	2.25(2.50)[a]
PARION	2.32	2.56		
M-KKR-CPA	2.79	2.87	2.89	2.91
Database	2.98	2.98	2.91	2.95
CFD	1.38	1.82	0.88	1.83

[a] Numbers in parenthesis obtained using PVMe

1. **Direct Self-Consistent-Field (SCF)**; in this benchmark a direct SCF calculation is performed on the morphine molecule. The code is parallel in the calculation of the two-electron integrals and requires communications to send a density matrix to each processor and then a global summation of the Fock matrices produced on each node. In comparison to computation, the communication requirements are small and generally involve long messages.

2. **CRYSTAL**; in this benchmark a solid state SCF calculation[7, 8] is performed on a slab of corundum (Al_2O_3). There are 10 atoms in a unit cell with 100 electrons described by 86 atomic orbitals. The calculation initially involves calculation of the two-electron integrals which are written to local disk on each workstation. These integrals are then read from disk and used to generate multiple Fock matrices. Communication requirements are similar to the direct SCF module discussed previously.

3. **Integral Transformation**; this benchmark simulates the four index integral transformation performed before many post Hartree-Fock electron correlation methods. The initial list of ordered integrals is distributed across the different workstations and two of the four indices are transformed in parallel on each machine. Prior to performing the second half transformation it is necessary to resort the integrals between the different workstations. This step involves considerable interprocessor communications, which would be expected to benefit from a high bandwidth network. This benchmark effectively corresponds to an integral transformation involving 60 basis functions.

4. **AMBER (Assisted Model Building with Energy Refinement)**; the molecular mechanics energy minimization (MINIM), and the evaluation of free energy differences through molecular dynamics (GIBBS) have been benchmarked on the clusters. The calculation is parallel in the evaluation of pair forces. Interprocessor communications involves a global sum whose dimension is a linear function of the number of atoms. The MINIM benchmark calculation is on the enzyme thermolysin (6391 atoms, 1 029 628 non-bonded interactions), whilst the GIBBS benchmark simulates the mutation of cytosine to iminocytosine in a bath of 263 water molecules (804 atoms, 100,778 non-bonded interactions).

5. **M-KKR-CPA (Many atom-Korringa Kohn Rostoker-Coherent Potential Approximation)**; in this code the total electronic energy of substitutionally disordered alloys is calculated. The code requires the integration of the scattering path operator for a variety of energies. Parallelism is achieved by allocating different energies to different processors, using a master/slave strategy to ensure load balancing. Communications are limited and correspond to short messages detailing a particular task. In this benchmark a calculation is performed on face centered cubic Ni.

6. **Database Searching**; in this application we consider a parallel method for searching 3-D chemical databases. The database is resident in memory on each machine. Partial segments of the data base are searched on each node. The application is characterized by having a moderate number of very short messages.

7. **Molecular Dynamics Simulation (PARION)**; in this benchmark a melt of sodium chloride containing 8000 ions is simulated. Parallelism is achieved using a replicated data approach, and requires a global summation during each simulation time step. The code includes a full Ewald summation to handle the long range electrostatic forces.

8. **Computational Fluid Dynamics (FLOW)**; a CFD benchmark code[4] which simulates flow over a cavity has been ported to the clusters. The 2D data space of the CFD application is decomposed into block-shaped domains; one allocated to each processor. Communications between processors primarily involve the exchange of so called halo data, although a small global operation is required to determine the time step in successive iterations. Messages sent in a 64 × 64 size problem are on average 3Kbytes long.

In table 5 we give the speed-up obtained by the various applications on 3 HP 720s and the 3 IBM 530Hs using the different networks. The applications can be grouped together according to their compute to communicate ratio:

In the first category we have the direct SCF, CRYSTAL, M-KKR-CPA and Database searching. These applications are all highly parallel and not surprisingly they show little difference between the speed-ups obtained using Ethernet, FDDI and SOCC. With the exception of CRYSTAL, these applications run very well on the Intel iPSC/860 hypercube. For CRYSTAL the restrictive input/output of the hypercube severely limits the speed-up observed. On the cluster the local disk on each machine removes this bottleneck and the application scales very well.

In the second category we have Integral Transformation, Amber, PARION and FLOW. One would expect these applications to show some benefit from the faster networks, however, since Ethernet, FDDI and SOCC (without PVMe) have roughly equivalent latencies the degree to which they benefit depends critically upon their communication patterns. An application such as the integral transformation involves very long messages which can exploit the enhanced bandwidth of FDDI and hence the dramatically improved speed-up compared with Ethernet. On the other hand, the Amber benchmarks involve relatively small message lengths, thus we see only slight improvement in going from Ethernet to FDDI or SOCC. On the IBM however, AMBER does show some benefit from using PVMe where the latencies have been reduced.

Table 6: Estimated Elapsed Times (secs) for CFD Benchmark on Four HP720 Workstations as a Function of Network Latency and Bandwidth

Bandwidth	Latency(msec)					
(MBytes/sec)	0	1	2	3	4	10
0.10	185.78	189.73	193.77	197.80	201.84	226.07
0.25	87.57	91.50	95.53	99.57	103.61	127.83
0.50	54.83	58.75	62.79	66.83	70.87	95.09
0.75	43.92	47.84	51.88	55.92	59.95	84.18
1.00	38.46	42.38	46.42	50.46	54.50	78.72
2.00	30.32	34.20	38.24	42.27	46.31	70.53
3.00	27.68	31.50	35.51	39.55	43.58	67.80
4.00	26.43	30.15	34.15	38.18	42.22	66.44
∞	25.04	26.30	30.12	34.12	38.15	62.37

To further assess the relative importance of latency and bandwidth we have performed a communication trace on the FLOW benchmark. This application has particularly severe communication requirements involving many short messages. Using the communication trace we can reconstruct the benchmark to obtain the theoretical elapsed times which would be obtained from a network with different latency and bandwidth characteristics. The results, given in table 6, show that as the latency tends to zero and the bandwidth goes to infinity the elapsed time would correspond to 25.04 seconds. Thus the maximum possible speed-up is 3.3 on four machines, and is a reflection on how much of the code has been parallelized and whether the parallel code is well load balanced. The observed elapsed time using Ethernet and FDDI is 47.3 and 34.9 seconds respectively. Referring to table 3 these results relate to a latency of between 2 and 3msecs and a bandwidth of about 1MB/sec for ethernet and 3-4MB/sec for FDDI. We see that further enhancing the bandwidth beyond the 3-4MB/sec is likely to decrease the elapsed time by only 4 seconds, whereas halving the latency has the same effect. Thus for this application, network latency is becoming the bottleneck. A full discussion of the FOW CFD benchmark is the subject of a separate report[5, 6].

6 Conclusions

On the HP720 and IBM530H FDDI and SOCC offer a 3-6 times increase in communication bandwidth. Communication latency is measured to be roughly equivalent between the three networks, except a significant enhancement can be achieved by using PVMe on the SOCC.

For many applications a workstation cluster with a dedicated Ethernet (i.e. isolated from the general LAN) offers a powerful parallel compute environment. Futhermore, some parallel applications (e.g. CRYSTAL) are found to run particularly well on such a system as each node can be configured with large amounts of local disk.

The higher bandwidth networks are advantageous for jobs with a large communication requirement, although many of the codes in this category would benefit if the latency of these high speed networks was decreased significantly.

References

[1] R.J. Allan, L. Heck and S. Zurek, *Comp. Phys. Comm.* 59 (1990) 325.

[2] G.A. Geist and V.S. Sunderam, *Concurrency* 4 (1992) 293.

[3] Theoretical Chemistry Group Message Passing System, a toolkit for writing portable parallel programs using a message passing model. Written by R.J. Harrison at Argonne National Laboratory and Battelle Pacific Northwest Laboratory. Source code, documentation and reference material available via anonymous ftp from "/pub/tcgmsg" at "ftp.tcg.anl.gov".

[4] R.J. Blake, D.R. Emerson and R.J. Allan "FLOW: a parallel benchmark code for high seed air flow: version 1" report prepared under contract to NPL under the Esprit III PEPS project.

[5] R.J. Allan and R.J. Blake and D.R. Emerson "Homogeneous workstation clusters for parallel CFD", Daresbury Laboratory technical report: DL/SCI/TM96T.

[6] R.J. Allan and D.R. Emerson, "Homogeneous workstation clusters for parallel CFD", Parallel Computing (Submitted)

[7] C. Pisani, R. Dovesi and C. Roetti "Hartree Fock Ab Initio Treatment of Crystalline Systems" *Lecture Notes In Chemistry* 48, (Heidelberg: Springer-Verlag) (1988); R. Dovesi, C. Pisani, C. Roetti, M. Causà and V.R. Saunders *CRYSTAL88: Program no. 577, Quantum Chemistry Program Exchange* Indiana University, Bloomington, Ind. (1988)

[8] N.M. Harrison and V.R. Saunders, "A Benchmark of CRYSTAL on Workstations: 1", Daresbury Laboratory technical report: DL/SCI/TM90T.