SCOPE - The Specific Cluster Operation and Performance Evaluation Benchmark Suite

Panagiotis Melas and Ed J. Zaluska

Electronics and Computer Science
University of Southampton, U.K.

Abstract. Recent developments in commodity hardware and software have enabled workstation clusters to provide a cost-effective HPC environment which has become increasingly attractive to many users. However, in practice workstation clusters often fail to exploit their potential advantages. This paper proposes a tailored benchmark suite for clusters called Specific Cluster Operation and Performance Evaluation (SCOPE) and shows how this may be used in a methodology for a comprehensive examination of workstation cluster performance.

1 Introduction

The requirements for High Performance Computing (HPC) have increased dramatically over the years. Recent examples of inexpensive workstation clusters, such as the Beowulf project, have demonstrated cost-effective delivery of high-performance computing (HPC) for many scientific and commercial applications. This establishment of clusters is primarily based on the same hardware and software components used by the current commodity computer "industry" together with parallel techniques and experience derived from from Massively Parallel Processor (MPP) systems. As these trends are expected to continue in the foreseeable future, workstation cluster performance and availability is expected to increase accordingly.

In order for clusters to be established as a parallel platform with MPP-like performance, issues such as internode communication, programming models, resource management and performance evaluation all need to be addressed [4]. Prediction and performance evaluation of clusters is necessary to assess the usefulness of current systems and provide valuable information to design better systems in the future.

This paper proposes a performance evaluation benchmark suite known as the Specific Cluster Operation and Performance Evaluation (SCOPE) benchmark suite. This benchmark suite is designed to evaluate the potential characteristics of workstation clusters as well as providing developers with a comprehensive understanding of the performance behaviour of clusters.

A. Bode et al. (Eds.): Euro-Par 2000, LNCS 1900, pp. 183–188, 2000.

2 Performance Evaluation of HPC Systems and Clusters

Clusters have emerged as a parallel platform with many similarities with MPPs but at the same time strong quantitative and qualitative differences from other parallel platforms. MPPs still have several potential advantages over clusters of workstations. The size and the quality of available resources per node is in favour of MPPs, e.g. communication and I/O, subsystems, memory hierarchy. In MPP systems software is highly optimised to exploit the underlying hardware fully while clusters use general-purpose software with little optimisation.

Despite the use of Commodity Off The Shelf (COTS) components the classification of clusters of workstations is somewhat loose and virtually every single cluster is built with its own individual architecture and configuration reflecting the nature of each specific application. Consequently there is a need to examine closer the performance behaviour of the interconnect network for each cluster. Existing HPC benchmark suites for message-passing systems, such as PARKBENCH and NAS benchmarks, already run on clusters of workstations but only because clusters support the identical programming model as the MPP systems these benchmarks were written for [1, 2]. Although the above condition is theoretically sufficient for an MPP benchmark to run on a workstation cluster ("how much"), it does not necessary follow that any useful information or understanding about specific performance characteristics of clusters of workstations will be provided. This means that the conceptual issues underlying the performance measurement of workstation clusters are frequently confused and misunderstood.

3 The Structure of the SCOPE Benchmark

The concept of this tailored benchmark suite is to measure the key information to define cluster performance, Following EuroBench and PARKBENCH [3, 5] methodology and re-using existing benchmark software where possible. SCOPE tests are classified into single-node-level performance, interconnection-level performance and computational model level performance (Table 1).

Single node tests are intended to measure the performance of a single node-workstation of a cluster, (these are also known as *basic architectural benchmarks*) [3, 5]. Several well-established benchmarks such as LINPACK, or SPEC95 are used here as they provide good measures of single-node hardware performance.

In order to emphasise the importance of internode communication in clusters, the SCOPE low-level communication tests include additional network-level tests to measure the "raw" performance of the interconnection network. Optimisation for speed is a primary objective of low-level tests using techniques such as cache warm-up and page alignment. Performance comparisons through these levels provide valuable information within the multilayered structure of typical cluster subsystems. Latency and bandwidth performance can be expressed as a function of the message size and Hockney's parameters r_∞ and $n_{1/2}$ are directly applicable.

Collective communication routines are usually implemented on top of single peer-to-peer calls, therefore their performance is based on the efficiency of the

Table 1. The structure of the SCOPE suite

Test Level		Test Name	Comments
SINGLE NODE		LINPACK, SPEC95, etc	Existing tests
LOW-LEVEL	Network level	Latency/Bandwidth	Pingpong-like
	Message-passing	Latency/Bandwidth	Pingpong-like
	Collective	Synchronise	Barrier test
		Broadcast	Data movement
		Reduce	Global comput.
		All-to-all	Data movement
KERNEL-LEVEL	Operation	Shift operation	Send-Recv-like
		Gather operation	Vectorised op.
		Scatter operation	Vectorised op.
		Broadcast operation	Vectorised op.
	Algorithmic	Matrix multiplication	Row/Column
		Relaxation algorithm	Gauss-Seidel
		Sorting algorithm	Sort (PSRS)

algorithm implemented (e.g. binomial tree), peer-to-peer call performance, group size (p) and the underlying network architecture. Collective tests can be divided into three sub-classes: synchronisation (i.e. barrier call), data movement (i.e. broadcast and all-to-all) and global computation (i.e. reduce operation call).

Traditionally kernel-level tests use algorithms or simplified fractions of real applications. The SCOPE kernel-level benchmarks also utilise algorithmic and operation tests. Kernel-level operation tests provide information on the delivered performance at the kernel-level of fundamental message passing operations such as broadcast, scatter and gather.

This section of the benchmark suite makes use of a small set of kernel-level algorithmic tests which are included in a wide range of real parallel application algorithms. Kernel-level algorithmic tests will measure the overall performance of a cluster at a higher programming level. The kernel-level algorithms included at present in SCOPE are matrix-matrix multiplication algorithms, a sort algorithm (Parallel Sort with Regular Sampling) and a 2D-relaxation algorithm (mixed Gauss-Jacobi/Gauss-Seidel). A particular attribute of these tests is the degree in which they can be analysed and their provision of elementary level performance details which can be used to analyse more complicated algorithms later.

In addition, other kernel-level benchmark tests such as the NAS benchmarks can also be used as part of the SCOPE kernel-level tests to provide application-specific performance evaluation. In a similar way the SCOPE benchmarks (excluding the low-level network tests) can also be used to test MPP systems.

4 Case Study Analysis and Results

This section demonstrates and briefly analyses the SCOPE benchmark results obtained with our experimental SCOPE implementation on a 32-node beowulf cluster at Daresbury and the 8-node Problem Solving Environment beowulf cluster at Southampton. The architecture of the 32-node cluster is based on Pentium III 450 MHz CPU boards and is fully developed and optimised. On the other hand, the 8-node cluster is based on AMD Athlon 600MHz CPU boards and is still under development. Both clusters use a dedicated 100 Mbit/sec network interface for node interconnection, using the MPI/LAM 6.3.1 communication library under Linux 2.2.12.

Table 2. SCOPE benchmark suite results for Beowulf clusters

SCOPE Test		Daresbury 32-node cluster			PSE 8-node cluster		
Network latency, BW		63 μs 10.87 MB/s			47.5 μs 10.95 MB/s		
MPI latency, BW		74 μs 10.86 MB/s			60.3 μs 10.88 MB/s		
Collective tests							
	Size	2-node	4-node	8-node	2-node	4-node	8-node
Synch	-	150 μs	221 μs	430 μs	118 μs	176 μs	357 μs
Broadcast	1 KB	301 μs	435 μs	535 μs	243 μs	324 μs	495 μs
Reduce	1 KB	284 μs	302 μs	866 μs	245 μs	258 μs	746 μs
All-to-all	1 KB	293 μs	580 μs	1102 μs	260 μs	357 μs	810 μs
Kernel-level tests (600X600 matrix)							
	Size	2-node	4-node	8-node	2-node	4-node	8-node
Bcast op.	600x600	0.277 s	0.552 s	1.650 s	0.337 s	0.908 s	1.856 s
Scatter op.	600x600	0.150 s	0.229 s	0.257 s	0.212 s	0.239 s	0.245 s
Gather op.	600x600	0.163 s	0.265 s	0.303 s	0.192 s	0.243 s	0.266 s
Shift op.	600x600	0.548 s	0.572 s	0.573 s	0.617 s	0.615 s	0.619 s
	Size	2-node	4-node	8-node	2-node	4-node	8-node
Matrix	1080x1080	50.5 s	25.5 s	14.0 s	123 s	57.6 s	27.4 s
Relaxation	1022x1022	126 s	66.0 s	36.0 s	148 s	76.3 s	41.9 s
Sorting	8388480	78,8 s	47.9 s	24.9 s	46.4 s	29.6 s	21.8 s

The first section of Table 2 gives results for the SCOPE low-level tests, clearly the cluster with the fastest nodes give better results for peer-to-peer and collective tests. TCP/IP level latency is 47.5 μs for the PSE cluster and 63 μs for the Daresbury cluster while the effective bandwidth is around 10.9 MB/s for both clusters. The middle section of Table 2 presents the results for kernel-level operation tests for array sizes of 600x600 on 2, 4 and 8 nodes. The main difference between the low-level collective tests and the kernel-level operation tests

is the workload and the level at which performance is measured, e.g. the low-level tests exploit the use of cache, while kernel-level operations measure buffer initialisation as well. The picture now is reversed, the Daresbury cluster giving better results over the PSE cluster. Both clusters show good scalability for the scatter/gather and shift operation tests.

The last part of Table 2 presents results for the kernel-level algorithmic tests. The matrix multiplication test is based on the Matrix Row/Column Striped algorithm for 1080x1080 matrix size. The PSRS algorithm test sorts floating point vectors of size 8 million cell array. The multi-grid relaxation test presented is a mixture of Gauss-Jacobi and Gauss-Seidel iteration methods on a 1022x1022 array over 1000 iterations.

Results from these tests indicate a good (almost linear) scalability for the first two tests with a communication overhead. The implementation of the sort algorithm requires a complicated communication structure with many initialisation phases and has poor scalability. The performance difference between these clusters measured by the kernel-level tests demonstrates clearly the limited development of the PSE cluster which at the time of the measurements was under construction.

5 Conclusions

Workstation clusters using COTS have the potential to provide, at low cost, an alternative parallel platform suitable for many HPC applications.

A tailored benchmark suite for clusters called Specific Cluster Optimisation and Performance Evaluation (SCOPE) has been produced. The SCOPE benchmark suite provides a benchmarking methodology for the comprehensive examination of workstation cluster performance characteristics. An initial implementation of the SCOPE benchmark suite was used to measure performance on two clusters and the results of these tests have demonstrated the potential to identify and classify cluster performance.

Acknowledgements

We thank Daresbury Laboratory for providing access to their 32-node cluster

References

[1] F. Cappello, O. Richard, and D. Etiemble. Performance of the NAS benchmarks on a cluster of SMP PCs using a parallelization of the MPI programs with OpenMP. *Lecture Notes in Computer Science*, 1662:339–348, 1999.

[2] John L. Gustafson and Rajat Todi. Conventional benchmarks as a sample of the performance spectrum. *The Journal of Supercomputing*, 13(3):321–342, May 1999.

[3] R. Hockney. *The Science of Computer Benchmarking*. SIAM, 1996.

[4] Dhabaleswar K. Panda and Lionel M. Ni. Special Issue on Workstation Clusters and Network-Based Computing: Guest Editors' introduction. *Journal of Parallel and Distributed Computing*, 40(1):1–3, January 1997.

[5] Adrianus Jan van der Steen. *Benchmarking of High Performance Computers for Scientific and Technical Computation*. PhD thesis, ACCU, Utrecht, Netherlands, March 1997.