ON THE POTENTIAL AND RISK OF CLUSTERS IN NUMERICAL SIMULATION

M. M. Resch
High Performance Computing Center, Stuttgart, Germany
e-mail: resch@hlrs.de

Кластеры являются распространенной вычислительной платформой в численном моделировании. Их первыми и главными преимуществами являются низкая входная цена для малых систем и потенциал расширения этих систем до тысяч процессоров. Кроме того, кластеры обеспечивают стандартную технологию, которая помогает сохранять переносимость программного обеспечения и таким образом сокращают расходы на его разработку. Тем не менее, есть и обратная сторона. Увеличивающееся число компонент также повышает уровень сложности. А с этим уменьшается стабильность систем, в то время как общие расходы на эксплуатацию подобных систем увеличиваются.

Многие ученые концентрируются только на одном аспекте этой проблемы. Но без общего и глобального представления прогресс не может быть достигнут. Одним из препятствий для прогресса является недальновидное мнение о кластерах как о дешевых системах. Обычно рассматриваются только затраты на инвестиции, а дальнейшие расходы игнорируются, поскольку они сразу не являются очевидными для научной среды. Анализ общей стоимости владения системой показывает неправильность такого подхода. В том числе мы установили, что для превращения кластеров в производительные системы должно быть уменьшено количество обслуживающего персонала. Соответственно, мы должны действовать в различных направлениях, чтобы достичь успеха.

Introduction

The costs of processors and networks have changed dramatically over the last years. What actually happened were two important things.

Improvement of standard components: While special purpose components have not been able in recent years to continue their exponential performance increase, standard components have caught up with them and might even top them in the near future. Partially this was due to a lack of investment in special purpose systems in the USA in the late 90s. Partially it was, however, caused by increasing technical difficulties in keeping the exponential growth that we saw in the last decades that was so well described by Moore's law. And last but not least the fact that production facilities for new technologies have become extremely expensive has slowed down the innovation process and has driven small companies out of the market.

Extension of the market: Since the gap between special purpose systems and standard components has narrowed, the whole community was able to benefit from the drop in prices.

This has extended the reach of technical computing and has brought powerful systems to scientific groups worldwide that for long had not been able to compete with large supercomputing centres. This extension of the market has created a positive feedback loop. With every extension the volume grows and a reduction of prices was possible for high volume vendors.

Decreasing prices and increasing performance have made standard based systems attractive to a variety of users. However, at the same time the community is faced with a widening gap between peak and sustained performance. This indicates that there is a lack of quality in the market that is somewhat hidden by an increase in raw but meaningless performance numbers.

In this paper we deal with an analysis of the potential of the hardware and software of clusters themselves. We then look at the level of achievable performance extracted from clusters of PCs. In the third chapter we have a look at a concept that is new to the community but is gaining increased importance with contracting budgets: Total Cost of Ownership (TCO). We present preliminary results that give an indication of the best architecture with respect to TCO. We conclude with some remarks on possible strategies for users in a low cost high volume market.

1. Levels of Performance

Low cost systems are designed for a mass market. Users in this mass market are driven by price considerations rather than by performance considerations. Consequently the sweet spot for system designers is mainly determined by costs. Prices for such systems are dropping. This has a positive side effect for numerical simulation experts because it also reduces their costs. The downside is that increasingly the special support for numerically intensive applications is reduced because vendors want to cut costs by reducing the number of product lines. As a consequence, from the point of view of performance, these systems are characterized by two factors.

1.1. Memory Performance

First, of all these systems do not put any emphasis on memory performance. The classical mass market does not have a quest for this. Most of these applications need large memory and high quality graphics. Hence, these are the issues that are important when developing a new processor. Both the memory used and the memory subsystem are not designed for highest performance.

A good number for understanding the impact of memory bandwidth is the *relative memory speed (rsm)*. We define it here to be the number of bytes per second that can be transferred to or from memory for each floating point operation that the processor can perform. This measure makes sense because we know that for many typical numerical applications it has to be in the range of $rsm > 1.5$ to achieve substantial levels of performance. Consequently traditional supercomputing architectures have a bandwidth of $rsm = 3$ and higher. Low cost cluster solutions on the other hand are currently operating with arange of about $rsm \approx 1.0$. What is even worse, vendors keep the bandwidth of the memory constant while increasing the clock rate of their processor family. This results in even lower values for rsm.

The notable exception here is the AMD Opteron processor [2] which has an rsm of 1.125. Besides vector systems the Opteron is also the only available architecture in the mass market that scales memory bandwidth with processor speed. So it can keep the rsm constant at an acceptable level. First results are good.

For all other systems what has to be expected in the future is a further increase in processor performance with memory speed increasingly lagging behind. Our key number rsm will drop further and will soon reach values of about 0.5. At that relative speed of the memory one can not expect any system to provide a reasonable level of performance. This can not be made up for by larger caches. Caches grow in size only with the size of main memory. Furthermore, complex cache hierarchies require a complex management which is hardly ever provided in a way that could compensate for the low memory bandwidth.

The problem is aggravated by the trend of vendors to put multiple processors in a node and even multiple cores on a die. The memory bandwidth is kept constant with the effect that rsm reaches unacceptably low levels. Tests reveil that for many dual node systems the second processor is either of no use or even slows down the computation and should be turned of.

The problem of memory bandwidth is well understood and is partially (Opteron) addressed by hardware vendors. With increasing speed of processors, now memory latency starts to become an issue. It is currently in the range of several 100 CPU cycles. There is a danger that with further rising speed the latency will remain constant over time and thus become relatively larger. This issue is hardly discussed yet. It will hurt the programmer in pretty much the same way as the memory bandwidth problem.

1.2. Clustering

Second, low cost solutions in high performance computing rely on the cluster concept. An increase in speed is achieved by increasing the number of processors. This requires parallelizing codes which in turn brings in the communication problem. To estimate the communication costs we again have to deal with bandwidth and latency. Since MPI allows overlapping computation with communication we have the additional software problem.

With respect to latency cluster networks [3 – 5] typically do at least as well as special purpose networks of supercomputers. In fact these networks are so heavily tuned that they usually outperform traditional approaches. For a cluster one can expect to see a latency in the range of 3–5 μs unless Ethernet is used. For traditional supercomputers these latencies are in the range of 5–7 μs. For bandwidth again we can look at one parameter to get an idea of what the costs for such communication are. We look at the number of processor cycles that we loose for the transfer of a message of size 1 MB. This is a much better indicator for the communication speed than bandwidth and latency because it relates the communication costs to the speed of the processor and thus indicates how we reduce performance by communicating. For a typical of the shelf system [3 – 5] the loss of operations as defined above is 8.5 MFLOP. This is about four times higher than the loss of operations for a traditional supercomputer (2.1 MFLOP). None of these is going to change over the next years. So when working with low cost solutions these are the limitations we have to be aware of.

2. Total Cost of Ownership

When talking about low cost solutions we typically consider the purchase price to be the key figure. However, this is a wrong approach. The costs for any solution in computing are at least the following:

— Investment Costs: The actual price for buying a system. Very often this price already includes others of the following;

— Maintenance Costs: This includes costs for keeping spare parts, having vendor staff on site and other running costs for maintaining the system;

— Staff Costs: Employees of the owner who look after the system have to be paid for. These costs very often are ignored because system administration is considered to be a minor activity. For medium and large systems this is not the case;

— Energy Costs: The main costs are for power consumption and for the cooling of larger installations;

— Room Costs: For larger installations there is a cost for housing the system which has to be considered;

— Software Costs: Costs for software that is not freely available.

For small low cost systems (like clusters with about 32 to 64 processors) the costs for room and staff may be negligible if both room and system administration staff is already available (and willing to work overtime). Staff costs should, however, be considered anyway, because these systems require special skills and a considerable amount of human intervention.

When analysing large installations of the last four to five years one can get a rough understanding of the total cost of ownership for different hardware solutions. We indicate here the cost in million Euros for a sustained performance of 1 TFLOP/s. The reader has to keep in mind that sustained performance is heavily depending on the type of application chosen. Figures given here relate to average performance levels. Considering all this we come up with table 1.

It is interesting to note that traditional supercomputers (vector based systems) and low cost cluster solutions show roughly the same price/performance relation for larger installations. Classical 64-bit based architectures score worse in this comparison. Two things have to be considered, however. First, for smaller systems the price/performance relation may improve for low cost solutions. This is the case, when a low cost solution is part of a larger compute environment and does not introduce additional staff or maintenance costs. Costs are also lower if no specified level of reliability and availability is defined. Second, price/performance varies

Table 1

Price/Performance considering average levels of sustained performance and total cost of ownership for large installations

Type of System	TCO in MEuro for 1 TFLOP/s sustained
Vector Systems (SX, X1)	14
64-bit Micro (IA64, Power, ...)	22
Low Cost Cluster (IA32, Opteron)	14

Table 2

Break down of costs (in %) for various types of architectures for large installations

	Vector Systems	64-bit Micro	Low Cost Cluster
Investment	76.7	66.9	52.3
Maintenance	12.7	18.6	14.6
Software	0.5	0.7	0.5
Staff	2.8	5.0	3.9
Power	4.3	5.3	16.4
Cooling	2.5	3.0	9.4
Room	1.1	0.5	3.0

with the specific application run. Numbers given here are based on average performance results. If a system is dedicated to a specific purpose (a single application or a single type of application) an individual calculation of total cost of ownership will be necessary and will most likely show different results than the ones presented here.

A breakdown of the costs shows where the money typically goes to.

For the cluster solutions it is notable that the costs for electricity are surprisingly high. This reflects the increased level of power consumption which not only leads to high power requests but also elevates the level of power needed for cooling.

Conclusion

The interested researcher should understand from these first results on total cost of ownership that the investment costs are not the only important figures. A thorough analyses of all costs should therefore be considered before making a hardware decision. In many cases this may lead to a low cost cluster. In some cases it may turn out that a cluster is too expensive.

Besides financial costs one has to be aware of further problems which are not yet fully understood and difficult to express in terms of money.

Reliability: Standard parts are produced for the mass market. They need not achieve the same level of reliability that a traditional supercomputer is supposed to achieve today. Hence, the user might be faced with a series of failures of parts of the same type. This may or may not become a problem. Assuming that computers are bought to compute it usually is a problem because compute time is lost and jobs at least get stopped. For the user it is mandatory to learn to write fault tolerant applications.

Complexity: A higher number of slower processors results in a higher software development effort. This goes as deep as having to design new algorithms. For the user that means that the whole approach of simulation (models, algorithms, programming) has to be thought through and may have to be changed.

Low cost clusters may not be as low cost as they initially seem. And with additional problems like reliability and complexity they do have their downsides too. However, they have helped to not only broaden the base of high performance technical computing but have also allowed to build ground breaking system. They will be with us for a time and will help to further computational science and engineering.

References

[1] TOP 500 LIST http://www.top500.org.

[2] JOSEPH E., KAUMANN N., WILLARD C.G. The AMD Opteron processor: A new alternative for technical computing, White Paper, IDC, November 2003.

[3] MYRINET, http://www.myri.com/myrinet/overview/index.html.

[4] QUADRICS, http://www.quadrics.com/.

[5] INFINIBAND http://www.infinibandta.org/home

[6] MPI FORUM MPI: A Message-Passing Interface Standard. Document for a Standard Message-Passing Interface, Univ. of Tennessee, 1995.

[7] MPI Forum MPI2: Extensions to the Message-Passing Interface Standard. Document for a Standard Message-Passing Interface, Univ. of Tennessee, 1997.

[8] Brunst H., Winkler M., Nagel W.E., Hoppe H.-C. Performance optimization for large scale computing: The scalable vampir approach // Computational Science — ICCS 2001. Pt II, LNCS 2074. Springer, 2001. P. 751–760.

[9] Girona S., Labarta J., Badia R.M. Validation of Dimemas communication model for MPI collective communications // Recent Advances in Parallel Virtual Machine and Message Passing Interface, LNCS 1908. Springer, 2000.

[10] The Lustre Project, http://www.lustre.org.

[11] Lang U. Distributed and collaborative visualization of simulation results // Comp. Technologies. 2003. Vol. 8, Special Issue. Proc. Russian-German Adv. Res. Workshop on Comp. Sci. High Perf. Comp. Pt 1. P. 82–96.

Received for publication October 12, 2004