



SGI Bioinformatics Performance Report

Fall 2001

Table of Contents

INTRODUCTION	2
SGI™ SOLUTIONS FOR LIFE SCIENCES	2
AMDAHL'S LAW: EVALUATING PARALLEL PERFORMANCE	3
BLAST	5
BLAST: Single-Processor Performance	6
High-Throughput BLAST (HT-BLAST)	7
HT-BLAST vs. BLAST: Relative Performance	8
CLUSTAL W	9
CLUSTAL W: Single-Processor Performance	10
CLUSTAL W: Parallel Performance	11
CLUSTAL W: High-Throughput Performance	12
FASTA	13
FASTA: Single-Processor Performance	14
FASTA: Parallel Performance	15
D ² _CLUSTER	16
d ² _cluster: Single-Processor Performance	17
d ² _cluster: Parallel Performance	18
HMMER	19
HMMER: Single-Processor Performance	20
HMMER: Parallel Performance	21
WISE2	22
Wise2: Single-Processor Performance	23
Wise2: Parallel Performance	24
SGI Application Support	25
SGI Crunch Projects	25
SGI Global Scientific Support	26
SGI ORIGIN FAMILY SERVERS FOR BIOINFORMATICS AND CHEMISTRY	26
SGI Origin Family Servers vs. Intel® Servers Running Linux®	27
IRIX FEATURES	28
Weightless Priority	28
Checkpoint and Restart	29
ACKNOWLEDGMENTS	30

For detailed technical support information on bioinformatics applications running on SGI systems please visit the SGI Web site www.sgi.com/solutions/sciences/chembio/resources. You will find general information, technical information, porting notes, and known problems and fixes.

For more information on how SGI can help you solve your computational problems, please contact your local SGI sales representative or distributor, visit our Web site at www.sgi.com/solutions/sciences/chembio, or contact us at chembio@sgi.com.

Introduction

This report provides information on the performance of a number of leading bioinformatics applications running on SGI™ Origin™ family servers. The SGI team of bioinformatics professionals has worked with the software providers to incorporate parallelism and high-throughput capability into these applications in order to take full advantage of the scalability of the shared-memory SGI Origin family servers. Bioinformatics applications such as BLAST, FASTA, and CLUSTAL W can be evaluated as to their computational performance from two perspectives:

1. Throughput performance addresses the need to process a large batch of jobs in the shortest amount of time.
2. Turnaround performance measures how quickly one individual job is completed.

Delivering high-throughput and fast-turnaround performance for the full range of bioinformatics applications is one of the main goals of the SGI Bioinformatics Applications Engineering Team. This report highlights how SGI Origin family systems meet both types of computational demands.

High levels of performance and scalability on SGI Origin family servers stem from highly tuned applications that take advantage of its 64-bit MIPS® R12000™ and R14000™ processors and supporting memory and I/O subsystems. Production computing customers also benefit from a number of key features offered by the IRIX® operating system on SGI Origin family servers, especially in comparison to alternative commodity-based hardware systems. In conjunction with the ability to handle large databases, unique features such as weightless priority and checkpoint restart help maximize system utilization and job throughput in a bioinformatics production environment.

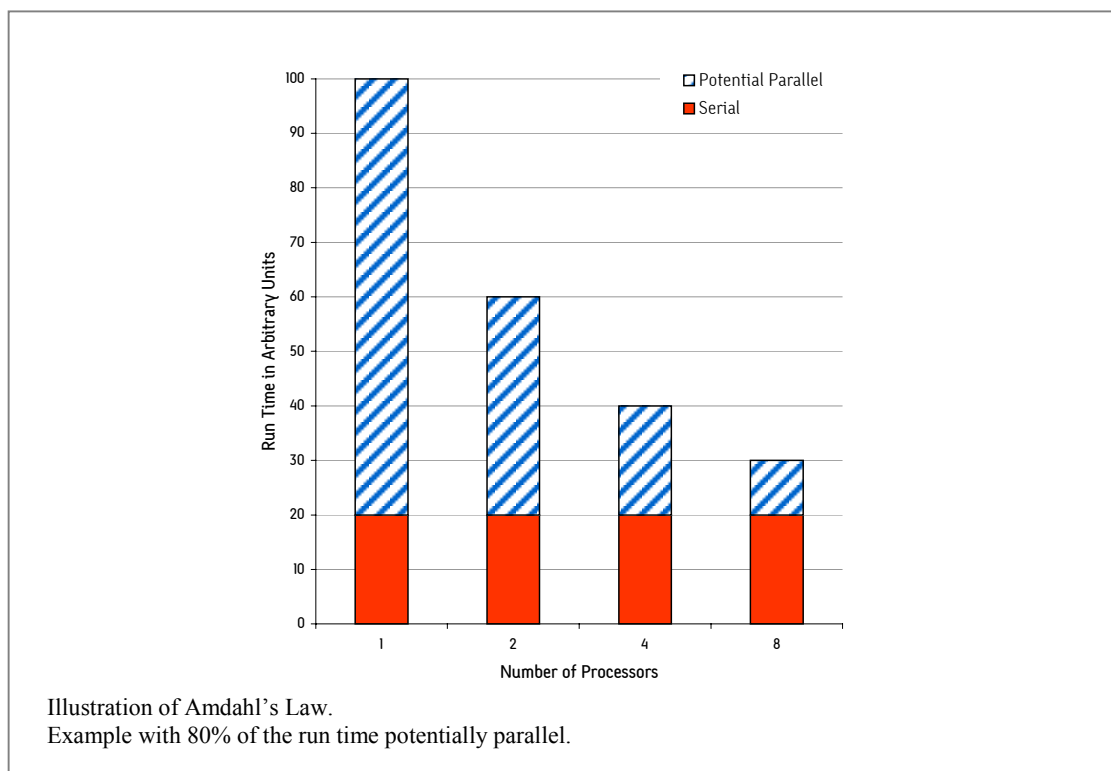
SGI™ Solutions for Life Sciences

SGI has made a long-term commitment to and has been a leader in the life sciences community for more than 15 years. This commitment is expressed in a variety of ways, including providing our customers superior computational platforms and through our support for bioinformatics and computational chemistry applications as well as high-performance and 3D visualization solutions for the life sciences. The SGI commitment is to provide our customers with the tools they need to ensure the most efficient utilization of research computing resources. Our goal is to enable our customers to make the best research decisions and to ultimately reduce their time to achieve research insights.

The SGI worldwide team of scientists and engineers is focused on helping customers solve their hard computational problems. This team works with worldwide commercial and academic software developers to advance and implement algorithms that take full advantage of the high-performance computational and graphical environments available on SGI hardware.

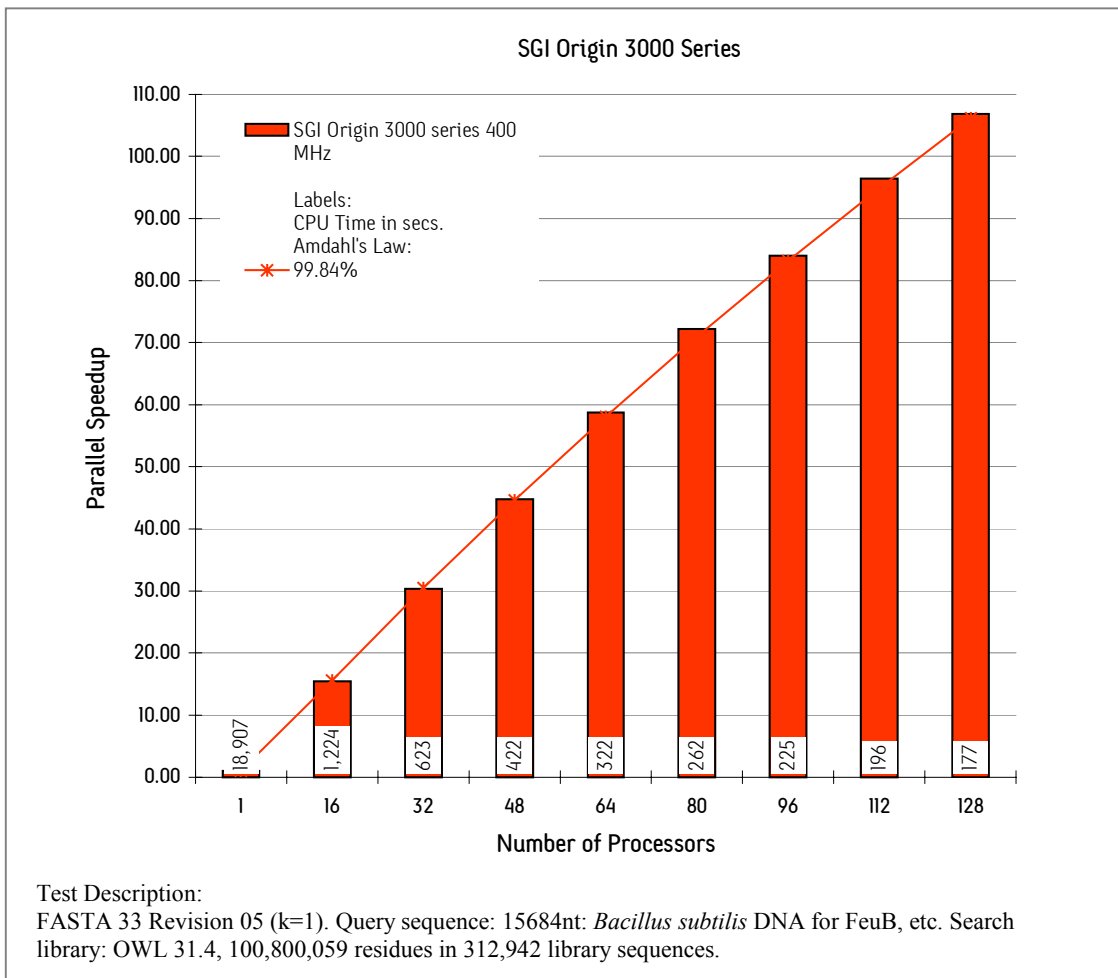
Amdahl's Law: Evaluating Parallel Performance

Amdahl's Law defines the increase in performance that can be gained by parallelizing an application. It states that performance improvements due to parallelization are limited by the fraction of the code that is not running in parallel. For example, if 80% of the run time of a hypothetical application can be run in parallel (blue-striped or gray bars), then as more processors are added, the nonparallel portion of the code (red or solid bars) dominates the performance. When this application is run with eight processors, the nonparallel or serial run time dominates the overall run time and very little, if any, speed increase can be achieved by using additional processors. Furthermore, in the limit case of applying an infinite number of processors to solve this problem, the speedup to be gained will never be more than five.



Amdahl's Law is of greatest importance in understanding and gauging parallel performance; thus, included on each of the following parallel-speedup graphs is the theoretical performance curve predicted by Amdahl's Law. Large deviations from this curve could indicate performance anomalies.

Here, the example from a FASTA run shows how the measured speedup (bars) closely matches the theoretical speedup (curve, corresponding to 99.84% parallelism), indicating that no extraneous hardware or system software events negatively affect the parallel performance of FASTA on the SGI™ Origin™ 3000 series system.



BLAST

BLAST (Basic Local Alignment Search Tool) is a collection of searching programs for biological sequence databases. The program uses the BLAST algorithm (see references below) to compare protein or DNA sequence queries to protein or DNA sequence databases.

References

Altschul, S. F., and W. Gish (1996). Local alignment statistics. Ed. R. Doolittle. *Methods Enzymol.* 266: 460–80.

Karlin, S., and S. F. Altschul (1993). Applications and statistics for multiple high scoring segments in molecular sequences. *Proc. Natl. Acad. Sci.* 90: 5873–7.

BLAST: Single-Processor Performance

The following benchmarks compare the relative single-processor performance of the BLASTN and TBLASTX programs on SGI Origin family servers.

Table 1 shows the relative performance of the MIPS R14000 500 MHz processor in an SGI Origin 3000 series system compared to the MIPS R12000 400 MHz processor in an SGI Origin 3000 series system. For BLASTN and TBLASTX, one can expect about a 25% performance improvement with the 500 MHz SGI Origin 3000 series servers, compared to the 400 MHz SGI Origin 3000 series servers.

Table 1 BLAST
SGI Origin 3000 Series Relative Performance

Executable	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
BLASTN	1,484	1,168	1.27
TBLASTX	3,332	2,730	1.22
Average			1.25

Test description:

Both benchmarks use a nonredundant nucleotide database (1,961,177,913 total letters in 614,801 sequences). The BLASTN benchmark uses a 100 EST sequence query (26,999 total letters in 100 sequences), while the TBLASTX benchmark uses a 10 EST sequence query (3,296 total letters in 10 sequences).

High-Throughput BLAST (HT-BLAST)

HT-BLAST is a wrapper around the original BLAST program that is optimized for use in a high-throughput production environment, where multiple query sequences are being searched against multiple databases. SGI developed HT-BLAST, which now supports NCBI-BLAST, PSI-BLAST/PHI-BLAST, and WU-BLAST in high-throughput mode on SGI™ IRIX OS-based systems.

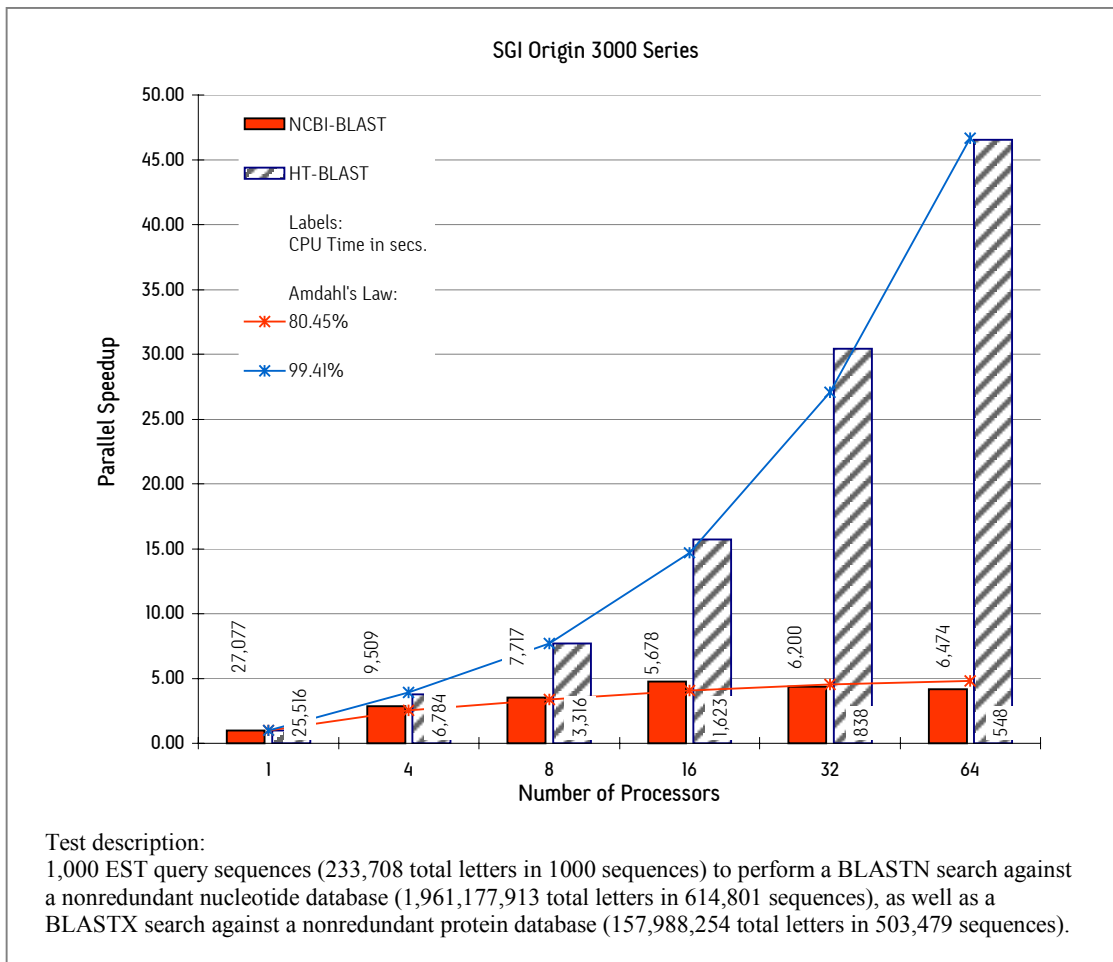
For more information about HT-BLAST, please see the white paper entitled *High-Throughput BLAST*:
www.sgi.com/solutions/sciences/chembio/resources/papers/HTBlast/HT_Whitepaper.html.

HT-BLAST executables are freely available from:

www.sgi.com/solutions/sciences/chembio/resources/blast/ht-blast.html.

HT-BLAST vs. BLAST: Relative Performance

The benefits of using HT-BLAST versus the original BLAST become very clear when searching a large number of query sequences against multiple databases. While the scaling of the original BLASTN and BLASTX programs may be limited to four to eight CPUs on any system, HT-BLAST can scale much higher. Furthermore, the relative performance of HT-BLAST vs. BLAST actually increases with the number of processors applied. For example, the results below show that HT-BLAST with 16 CPUs is more than three times faster than the original NCBI-BLAST with the same number of CPUs. As the number of processors applied increases to 32 CPUs for both HT-BLAST and NCBI-BLAST, HT-BLAST is more than seven times faster than NCBI-BLAST and almost 12 times faster with 64 CPUs of a 400 MHz SGI Origin 3000 series server.



CLUSTAL W

CLUSTAL W is a widely used, multiple-alignment program for biological sequences. The program uses the CLUSTAL W algorithms (see reference below) to progressively align multiple protein or DNA sequences.

SGI applications engineers have developed a parallel version of CLUSTAL W that performs the pairwise alignments, the guide tree formation, and the final multiple alignment using multiple processors. For large numbers of sequences (greater than 500), pairwise alignment computations are the most time-consuming steps in CLUSTAL W. Because each pairwise alignment can be calculated independently, the algorithm can be parallelized very efficiently. However, to get an even better scalability for 16 or more processors, additional steps were taken to parallelize the guide tree calculation and the progressive multiple-alignment stages of the algorithm.

SGI applications engineers have also developed a throughput version of CLUSTAL W, HT-CLUSTAL W, which is optimized to perform well in high-throughput production environments where a significant number of multiple sequence alignments need to be calculated on a regular basis. An example of such an environment would be a CLUSTAL W Web server that gets multiple requests for alignments of various sizes. Another example would be a stream of multiple alignments coming from database searches. Each database search generates various numbers of high-scoring hits, which then need to be aligned with a query used in the search in order to get better insight into biological functions.

For more information about Parallel CLUSTAL W and HT-CLUSTAL W, please see the white paper entitled *Performance Optimization of Clustal W: Parallel Clustal W, HT Clustal, and MULTICLUSTAL*: www.sgi.com/solutions/sciences/chembio/resources/papers/Clustal_DevNews/Clustal_DevNews.pdf. Parallel CLUSTAL W and HT-CLUSTAL W executables are available via www.sgi.com/solutions/sciences/chembio/resources/CLUSTAL W/.

References

Thompson, J. D., D. G. Higgins, and T. J. Gibson (1994). CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* 22: 4673–80.

CLUSTAL W: Single-Processor Performance

In Table 2, the relative performances of the SGI Origin 3000 series with different MIPS processors are shown.

Researchers running CLUSTAL W on an SGI Origin 3000 series system equipped with MIPS R14000 500 MHz processors can expect about a 25% performance improvement compared to the MIPS R12000 400 MHz processors.

Table 2 CLUSTAL W
SGI Origin 3000 Series Relative Performance

Test Case	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
Test100	92	73	1.26
Test200	349	280	1.25
Test400	1,398	1,125	1.24
Test600	3,970	3,245	1.22
Average			1.24

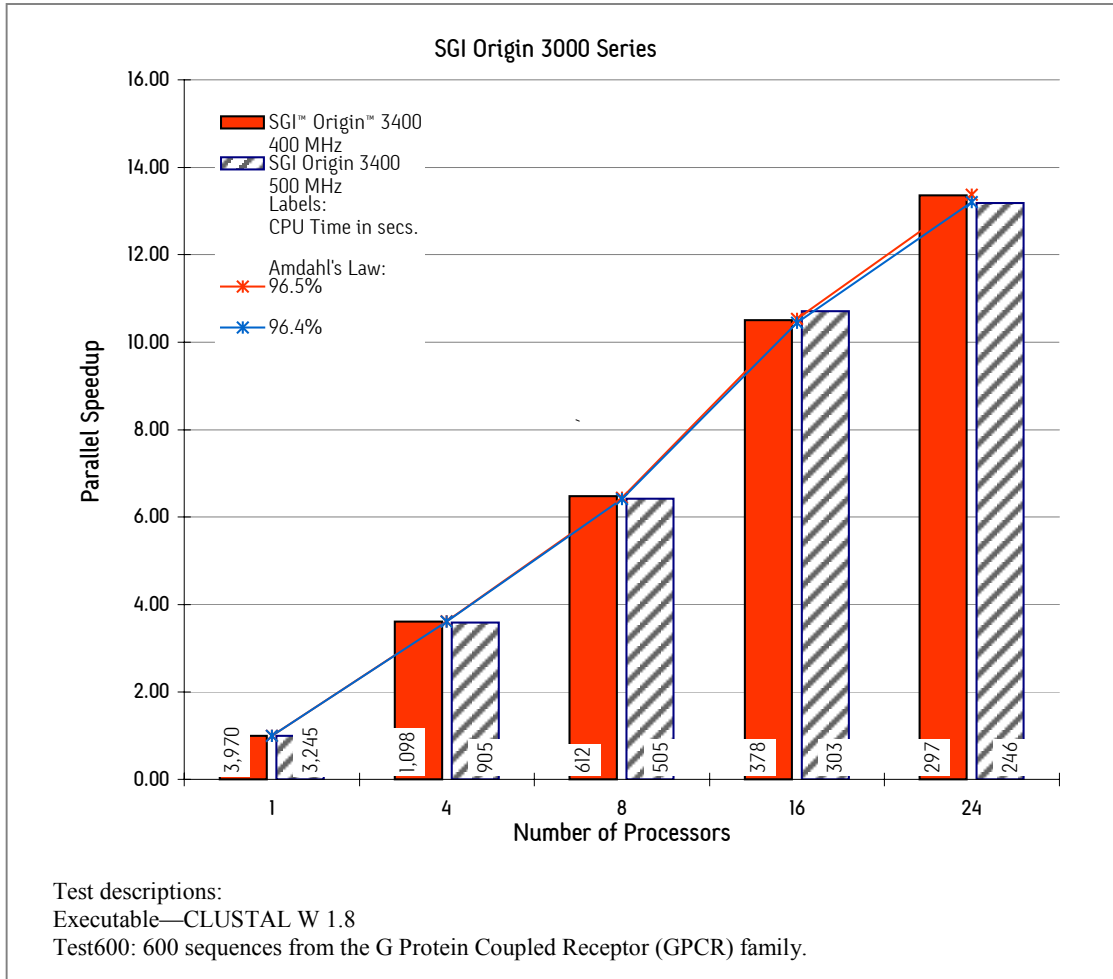
Test descriptions:

Executable—CLUSTAL W 1.8

Test100, Test200, Test400, Test600: 100, 200, 400, 600 sequences respectively from the G Protein Coupled Receptor (GPCR) family.

CLUSTAL W: Parallel Performance

The turnaround time for this particular job can be improved by a factor of more than 13 when 24 processors of an SGI Origin 3000 series system with MIPS R12000 400 MHz or MIPS R14000 500 MHz processors are used. The parallel speedups of both systems are similar, but the time to solution of the 500 MHz processor is at least 20% faster than the 400 MHz processor for each number of processors used.

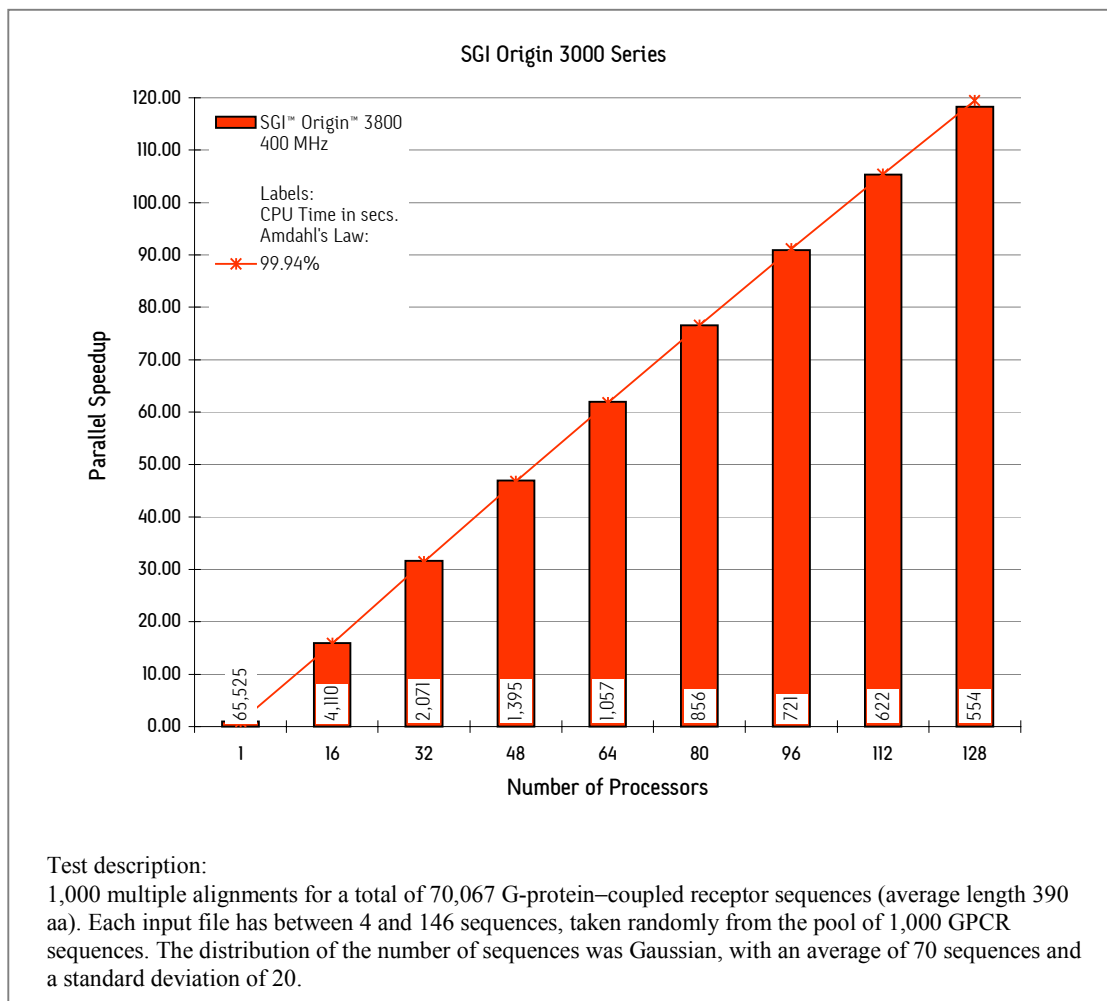


CLUSTAL W: High-Throughput Performance

The need to calculate a large number of multiple alignments of various sizes is often seen in high-throughput research environments. In order to produce a representative scenario, a profile of diverse multiple alignments was created and used as input for HT-CLUSTAL W.

HT-CLUSTAL W was used to calculate 1,000 multiple alignments for a total of 70,067 G-protein-coupled receptor sequences (average length 390 aa). Each input file contained between 4 and 146 sequences taken randomly from the pool of 1,000 GPCR sequences. The distribution of the number of sequences was Gaussian with the average of 70 sequences and standard deviation of 20. The total time it takes to finish all multiple alignments can be reduced from almost eighteen and a half hours, when running on one 400 MHz MIPS R12000 CPU, to under nine and a half minutes, when using 128 of the same processors.

This experiment shows superior scalability and load balance that allows for the efficient distribution of several CLUSTAL W tasks across multiple CPUs of SGI Origin family servers. The speedup is nearly linear, even at 128 processors, with no evidence of degradation.



FASTA

FASTA is a collection of sequence-comparison programs for biological sequence databases. These programs are widely used in bioinformatics and can be used to search sequence databases, evaluate similarity scores, and identify periodic structures based on local sequence similarity. One of the programs, called FASTA itself, uses the FASTA algorithm (see references below) to compare a protein sequence query to a protein sequence library or a DNA sequence query to a DNA sequence library.

Another program, FASTX, uses the FASTX algorithm to compare a DNA sequence query to a protein sequence library, translating the DNA sequence in three frames and following frame shifts in the alignment.

The following performance graphs demonstrate the excellent performance of FASTA and the parallel scalability of FASTX on the SGI Origin family server.

References

Pearson, W. R. Pearson and D. J. Lipman (1988). Improved tools for biological sequence comparison. *Proc. National Acad. Sci. USA* 85: 2444—8.

Pearson, W. R. (1996). Effective protein sequence comparison. *Methods Enzymol.* 266: 227—58.

FASTA: Single-Processor Performance

In Table 3, the relative performance of several computers is shown using two different executables from FASTA (Version 33, Revision 05), with both cases using a k-value of 1. The SGI Origin 3000 series system with MIPS R14000 500 MHz processors runs these cases close to 1.25 times faster than the SGI Origin 3000 series equipped with MIPS R12000 400 MHz processors. This is essentially the ratio of the frequencies.

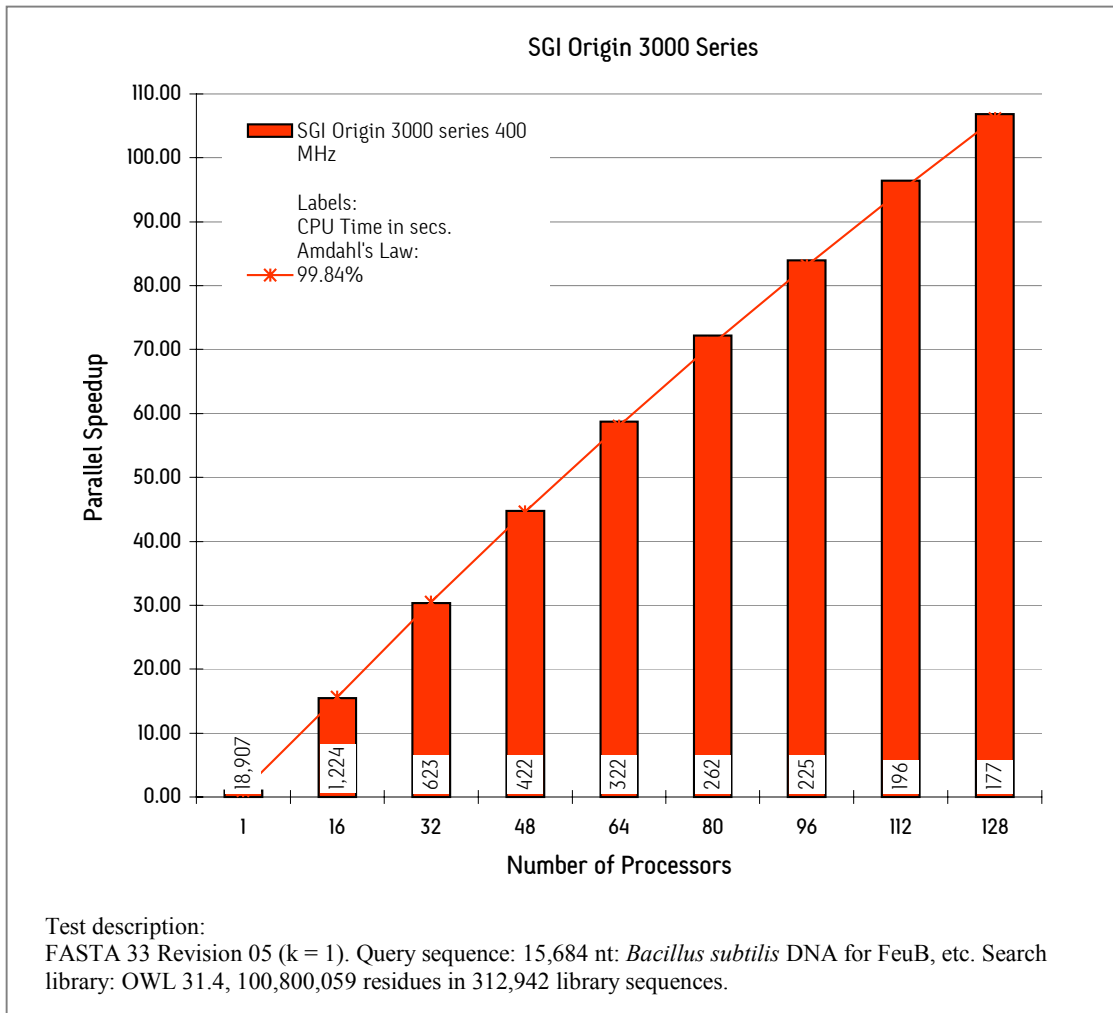
Table 3 FASTA
SGI Origin 3000 Series Relative Performance

Executable	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
Fasta	341	273	1.25
Fastx	465	373	1.25
Average			1.25

Tests descriptions:
 Database for both cases: OWL 31.4, 100,800,059 residues in 312,942 sequences.
 Fastx: fastx33t05 with the following query sequence: 500 nt: *Mycoplasma capricolum* gene for rpmH and dnaA, partial.
 Fasta: fasta33t05 with the query sequence: il12rec.aa: 662 aa: IL12 receptor component (*Homo sapiens*).

FASTA: Parallel Performance

Good parallel performance on a multiprocessor system allows the user to achieve fast turnaround time of a single job, enabling runs to be made in a much shorter amount of time than on a single-processor system. The benchmark in this graph shows the excellent parallel scalability that can be achieved for a single FASTX job on a 128-CPU SGI Origin 3000 series system. The results show how the solution time of a single-processor FASTX job can be reduced from almost five and a half hours to just three minutes by using 128 MIPS R12000 processors running at 400 MHz. For the scientist, this decreased turnaround time offers better interactivity for quickly investigating specific problems and making faster research decisions.



d²_cluster

The d²_cluster program, developed at the University of Houston, clusters a set of gene sequences using the d² measurement of similarity (see references below). The program complexity scales as N² where N is the number of sequences. The program is used by the South African National Bioinformatics Institute (SANBI) and was used to produce the Sequence Tag Alignment and Consensus Knowledgebase (STACK) database.

All of the benchmarks on the following two pages use the same standard d²_cluster (v1.22) benchmark composed of 10,000 EST sequences (3,673,416 total letters in 10,000 sequences), available at <ftp://ftp.sanbi.ac.za/SANBI/benchmarks>. A word length of 6 bases was used, any sequences less than 50 bases were ignored, a floating window of 150 bases was used to compare sequences, and a d² score of 0.04 was used for the clustering.

References

Carpenter, J. E., A. Christoffels, Y. Weinbach, and W. A. Hide. Assessment of the parallelization approach of *d2_cluster* for high performance sequence clustering. Submitted for publication.

Torney, D. C., C. Burks, D. Davison, and K. M. Sirotkin (1990). In *Computer and DNA*. G. Bell and T. Marr, Eds.; Addison-Wesley: New York, pp. 109–25.

d²_cluster: Single-Processor Performance

Table 4 shows the relative single-processor performance of the MIPS R14000 500 MHz and the MIPS R12000 400 MHz processors in SGI Origin 3000 series servers. The performance of the 500 MHz SGI Origin 3000 series system is about 25% faster than that of the 400 MHz SGI Origin 3000 series system.

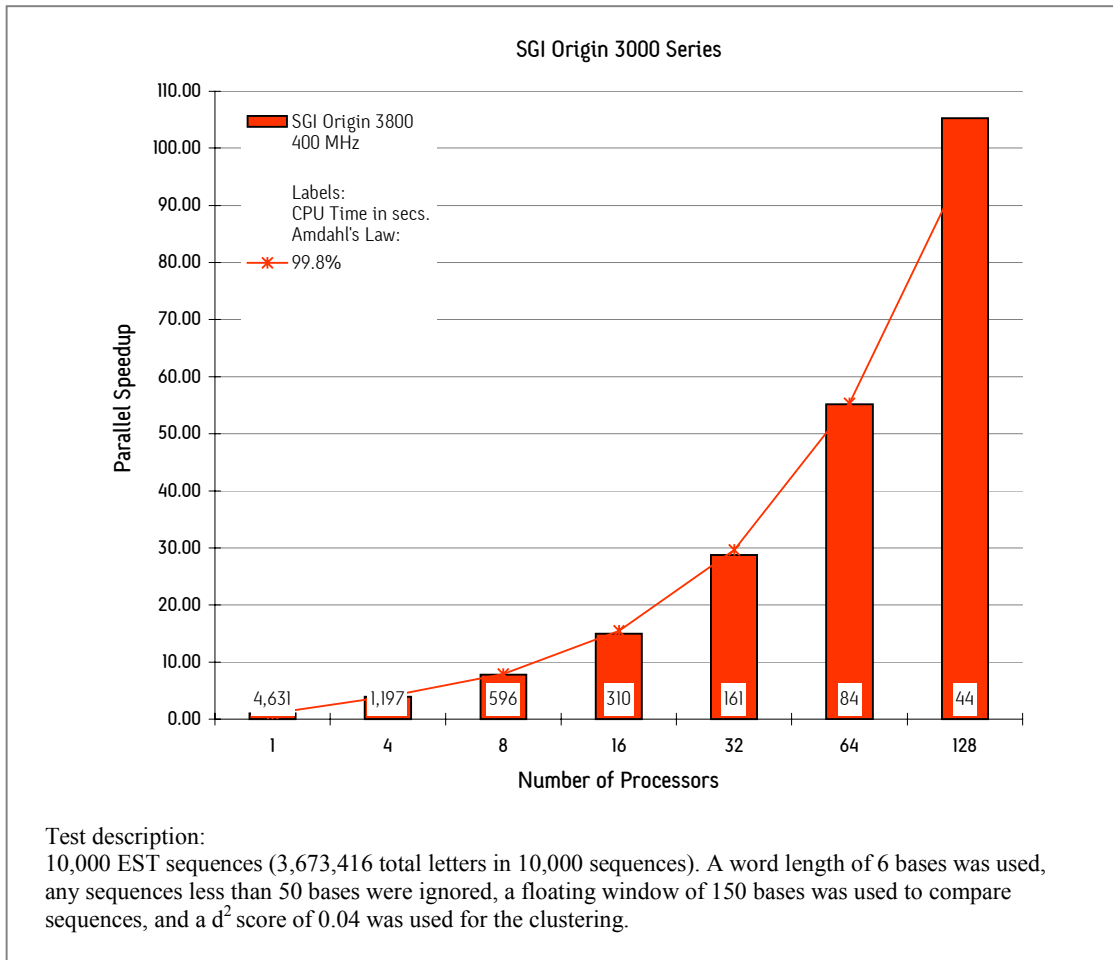
Table 4 d²_cluster
SGI Origin 3000 Series Relative Performance

Test Case	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
10,000 ESTs	4,631	3,736	1.24

Test description:
10,000 EST sequences (3,673,416 total letters in 10,000 sequences). A word length of 6 bases was used, any sequences less than 50 bases were ignored, a floating window of 150 bases was used to compare sequences, and a d² score of 0.04 was used for the clustering.

d²_cluster: Parallel Performance

The parallel performance of d²_cluster using the same benchmark is demonstrated below. The clustering time was reduced from around 77 minutes on one processor to less than 45 seconds on 128 MIPS R12000 400 MHz processors in an SGI Origin 3000 series system. The SGI Bioinformatics Applications Team helped to parallelize this application for SGI Origin family servers.



HMMER

HMMER, developed by Dr. Sean Eddy at Washington University, is a set of tools for biological sequence analysis. Algorithms in the HMMER package use profile Hidden Markov Models (profile HMM) to model primary structure consensus for protein or DNA families. Because they contain information from many sequences, HMMs can be more sensitive for biological database searches than BLAST or FASTA.

References

Eddy, S. R. (1996). Hidden Markov Models. *Curr. Opin. Struct. Biol.* 6: 361–5.

HMMER: Single-Processor Performance

For the two types of jobs shown in this report, HMMER 2.1.1 running on the SGI Origin 3000 series with MIPS R14000 500 MHz microprocessors is more than 20% faster than when run on the MIPS R12000 400 MHz–based machines. This is just under what is to be expected from the frequency ratio of the two microprocessors.

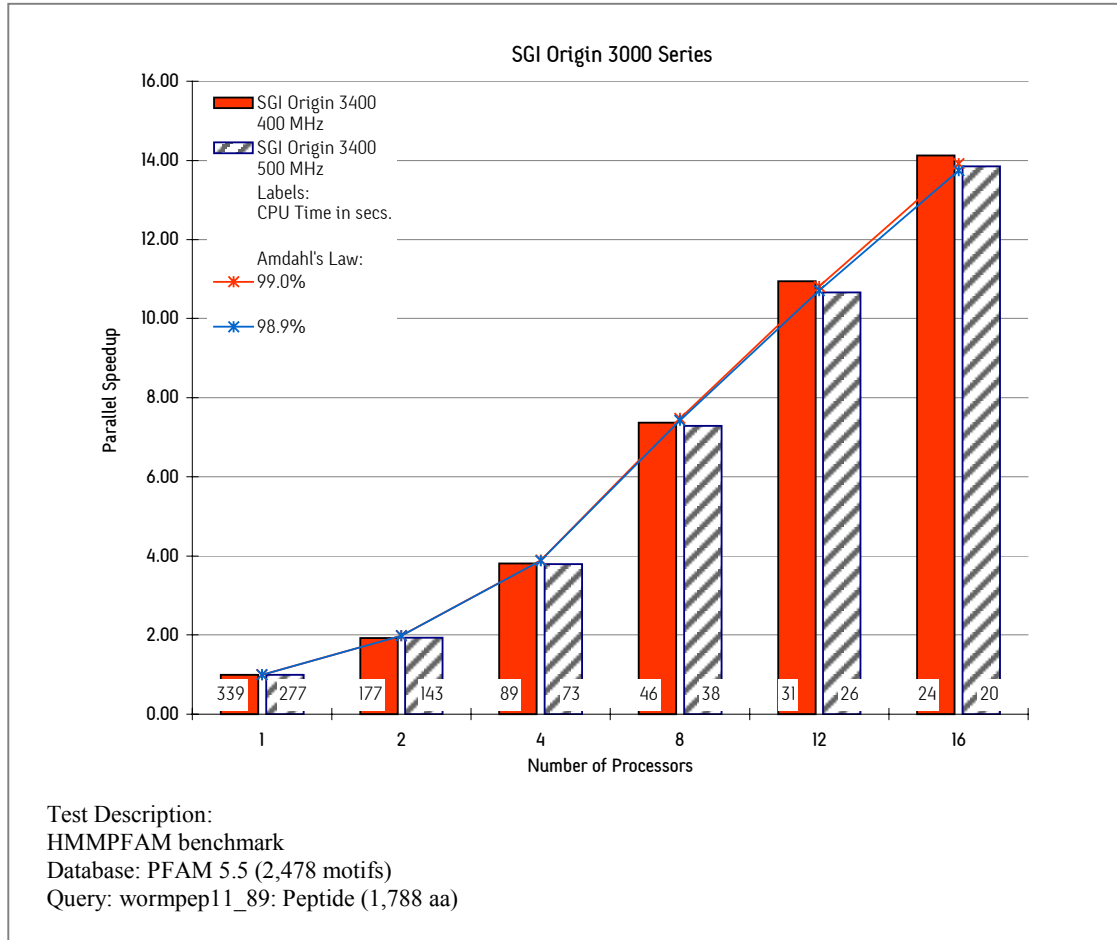
Table 5 HMMER
SGI Origin 3000 Series Relative Performance

Test Case	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
Hmmpfam	339	277	1.22
Hmmsearch	2,232	1,797	1.24
Average			1.23
Test description: hmmpfam: DB: PFAM 5.5 (2,478 motifs) Query: Peptide (1,788 aa) hmmsearch: DB: OWL 31.4, 100,800,059 residues in 312,942 sequences Query: RNA Recognition Motif (72 aa)			

HMMER: Parallel Performance

As in the previous examples, the parallel scaling of SGI Origin 3400 is essentially the same when the server is equipped with 500 or 400 MHz microprocessors.

The following graph shows that in the HMMPFAM case, the Amdahl's Law curve that best describes the scaling of the job goes just a notch down, from 98.8% with 400 MHz parts to 98.7% with the 500 MHz processors. At 16 processors, the run time on 500 MHz is still more than 15% shorter than on the 400 MHz server.



Wise2

Wise2 is a package that can compare a genomic DNA sequence to a protein sequence or profile HMM to predict a gene structure. The benchmarks below use the GENEWISE algorithm in Wise2 to compare a single DNA sequence with the Pfam database of HMM models.

References

Dr. Ewan Birney
The Sanger Centre
<http://www.sanger.ac.uk/Software/Wise2>

Wise2: Single-Processor Performance

The following benchmark compares the relative uniprocessor performance of the genewisedb program on a MIPS R14000 500 MHz processor in an SGI Origin 3000 series system to a MIPS R12000 400 MHz processor in an SGI Origin 3000 series system. The benchmark shows that using the 500 MHz SGI Origin 3000 series system enables genewisedb to run about 15% faster than on the 400 MHz SGI Origin 3000 series system.

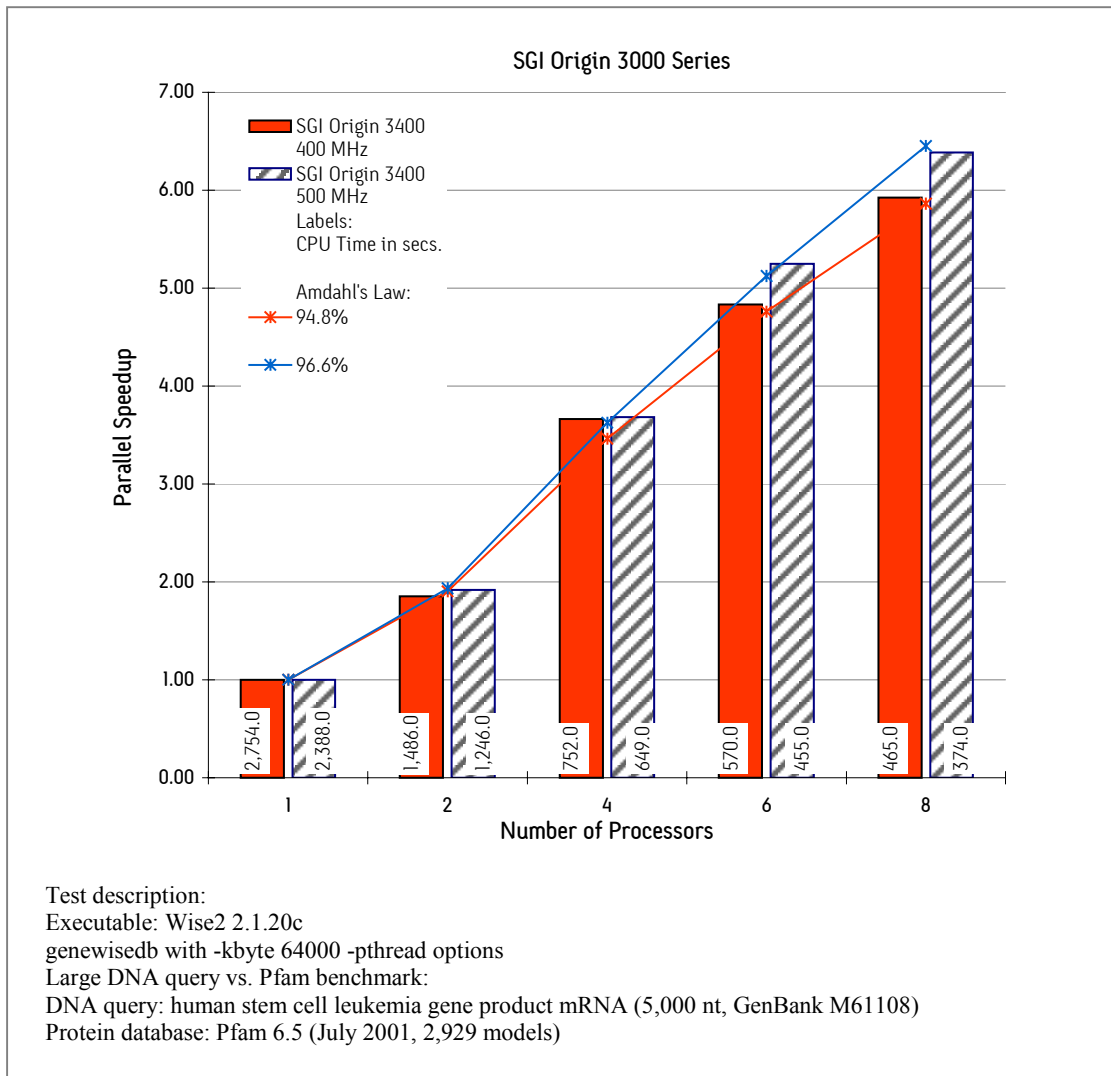
Table 6 Wise2
SGI Origin 3000 Series Relative Performance

Test Case	MIPS R14000 400 MHz Time in Seconds	MIPS R14000 500 MHz Time in Seconds	Speedup 500 MHz over 400 MHz
5000nt_vs_pfam	2,754	2,388	1.15

Test description:
Executable: Wise2 2.1.20c
genewisedb with -kbyte 64000 -pthread options
Large DNA query vs. Pfam benchmark:
DNA query: human stem cell leukemia gene product mRNA (5,000 nt,
GenBank M61108)
Protein database: Pfam 6.5 (July 2001, 2,929 models)

Wise2: Parallel Performance

The following multiprocessor benchmark shows the scaling that can be achieved with the parallelized genewisedb program on both a 500 MHz SGI Origin 3000 series server and a 400 MHz SGI Origin 3000 series server. Using a relatively large nucleotide query (5,000 nt), one can expect genewisedb to scale to about eight processors, reducing the execution time from nearly 40 minutes with a single MIPS R14000 500 MHz processor to a little more than 6 minutes with eight processors.



SGI Application Support

One of the most important areas of focus in the life sciences has been the implementation of scalable, high-throughput algorithms for bioinformatics, many of which have been described in this report. An example was the release in 1999 of a high-throughput version of the leading sequence-comparison application, BLAST—<http://www.sgi.com/solutions/sciences/chembio/resources/blast>. This technology allows high volumes of genetic sequences to be compared to other genetic sequences. HT-BLAST is used at more than 200 labs worldwide. The improved algorithm, developed by SGI, is able to take full advantage of the hardware scalability of SGI Origin family servers. HT-BLAST achieves speedups of more than 240 times on 256 processors. SGI applications engineers have provided algorithmic enhancements yielding comparable performance improvements on other key bioinformatics software such as parallel and high-throughput implementations of CLUSTAL W—[http://www.sgi.com/solutions/sciences/chembio/resources/CLUSTAL W](http://www.sgi.com/solutions/sciences/chembio/resources/CLUSTAL_W)—and parallelized versions of FASTA—<http://www.sgi.com/solutions/sciences/chembio/resources/fasta>.

The SGI Applications Engineering Team provides significant value to customers in both bioinformatics and chemistry. For example, Dr. Roberto Gomperts of SGI has helped implement the initial parallelization in the Gaussian application—<http://www.gaussian.com>, for which he was named a co-author. In addition, the application engineering team focuses on optimizing the performance of computational chemistry applications such as GAMESS-US, CHARMM, AMBER, Dmol, CASTEP, and CNX. The team also provides guidance on the incorporation of leading visualization algorithms to improve the utility of molecular visualization applications such as MolCad. The SGI commitment to improving the efficiency of applications has resulted in significantly shortening the cycle time for life sciences computing.

SGI Crunch Projects

SGI actively collaborates with the world's leading software developers and research labs on projects to demonstrate the feasibility of solving difficult number-crunching scientific problems. Among Crunch projects that SGI has run is the first whole genome annotation, performed in collaboration with European Molecular Biology Laboratory and the European Bioinformatics Institute, under the guidance of Chris Sanders in 1996. During this project, SGI servers were used to calculate the complete annotation of the yeast genome, as it first became available to the scientific community. A 64-processor SGI server was used to complete this calculation in three days, thus reducing the calculation time from an estimated seven months on a single processor. Details can be found at http://www.sgi.com/solutions/sciences/chembio/cust_success/gene. Additional Crunch projects include:

- 3DCrunch—http://www.expasy.ch/swissmod/SM_3DCrunch.html—completed in 1998 with Glaxo Wellcome and others by demonstrating the prediction of 3D structures for the entire SWISS-PROT database
- A microsecond protein simulation completed by Dr. Yong Duan and Prof. Peter Kollman of UCSF—<http://www.amber.ucsf.edu/members/yduan/983445.html>—the results of which appeared in the journal *Science*
- SpaceCrunch—a very large combinatorial chemistry database that was built by Tripos and was searchable by users live during the event
- A virtual molecular docking project called DockCrunch completed by the UK company Protherics—<http://www.protherics.com/crunch/>
- A virtual high-throughput screening project completed in 1999 with Tripos, Inc. on a 256-processor SGI™ 2800 server—<http://www.sgi.com/solutions/sciences/chembio/seminar/pdf/tripos.pdf>

- A virtual docking project study performed in 2000 by Metaphorics, Inc.—<http://www.daylight.com/meetings/mug2000/Weininger/10x40k/10x40k.html>
- Full genome annotations completed by MSI on 22 completed genomes—<http://www.msi.com/life/consortia/index.html>

These projects help to demonstrate how large-scale computing and the use of advanced, scalable computational techniques can benefit the scientific community.

SGI Global Scientific Support

SGI provides scientific support to customers worldwide. The SGI team in Japan has made a deep commitment and alliance with the Japan Genome Project and its research participants. SGI has a Sciences Solutions Team in Japan to support these customers with extremely high computational needs. One 768-processor SGI Origin 3000 series server, three 128-processor SGI™ 2000 series servers, two 256-processor SGI 2000 series servers, and numerous smaller SGI Origin family servers have been installed throughout the Japanese Genome Project sites. These sites include, but are not limited to, the Human Genome Center at the University of Tokyo, the Institute for Chemical Research at Kyoto University, the Institute for Molecular Sciences, the Computer Center in the Tokyo Institute of Technology, and the National Institute for Basic Biology. The SGI Sciences Solutions Team provides bioinformatics algorithm expertise and scalable computing support for these state-of-the-art centers in Japan.

SGI Origin Family Servers for Bioinformatics and Chemistry

Building on the robust cache-coherent SGI™ NUMA architecture that has made the SGI Origin family of servers the most modular and scalable in the industry, the SGI Origin 3000 series delivers flexibility, resiliency, investment protection, and superior performance. Now taking modularity a step further, you can scale CPU, storage, and I/O components independently within each system. Complete multidimensional flexibility allows organizations to deploy, upgrade, service, expand, and redeploy system components in every possible dimension to meet any business demand.

NUMAflex™ is a revolutionary snap-together server system concept that allows you to configure and reconfigure systems brick by brick to meet the exact demands of your applications. It allows you to upgrade CPUs to keep apace of innovation and to isolate and service I/O interfaces on the fly. You pay only for the computation, data processing, or communications power you need, and you expand and redeploy systems with ease as new technologies emerge.

With their high bandwidth, superior scalability, and efficient resource distribution, the new generation of SGI Origin family servers—SGI™ Origin™ 3200, SGI™ Origin™ 3200C, SGI Origin 3400, and SGI Origin 3800—is a performance leader. Leveraging the next generation of SGI NUMA architecture and the IRIX 6.5 operating system, SGI Origin 3000 series servers work with your existing application software and are compatible with other SGI™ IRIX OS–based servers and workstations.

Among the key features of the SGI Origin 3000 series are the increase in memory bandwidth and the lower memory latencies. The increase in memory bandwidth leads to improved processor performance, as more data is able to move from memory and I/O to the processors. The lower memory latency leads to increased processor, I/O, and memory scalability due to faster communications.

SGI Origin Family Servers vs. Intel® Servers Running Linux®

In recent years, the Linux open-source OS combined with commodity hardware has presented customers with an opportunity to purchase commodity-priced computational environments, typically in the form of a server or cluster based on the 32-bit Intel architecture processor (IA-32) running the Linux operating system. For those bioinformatics applications that are integer-intensive, a Linux server based on IA-32 processors can be a good solution. However, these applications and others often require features beyond those that a Linux IA-32 server can provide. Consequently, the decision of which type of server to choose can be complex. To assist in determining which server environment is appropriate for a bioinformatics compute facility, the following table summarizes some of the important points of comparison between SGI Origin family servers (IRIX OS with MIPS processors) and Intel servers running Linux (Linux IA-32).

Features	IRIX OS with MIPS Processors	Linux IA-32
Applications availability	High	High
Scaling	High (1,024 CPUs)	Low (16 CPUs)
Performance		
Integer-intensive applications	Comparable	Comparable
Floating-point applications	Faster	Slower
Large database capabilities		
Memory addressing	1TB (64-bit addressing)	3.8GB (32-bit addressing)
File size	9 million terabytes (XFS™)	2GB (Ext3fs), 64TB (XFS)
Filesystem size	18 million terabytes (XFS)	4TB (Ext3fs), 2TB (XFS)
Large job capabilities	Weightless priority Checkpoint and restart	Not available
Administration and maintenance	Lower	Higher

Applications Availability, Scaling, and Performance

Most of the commonly used bioinformatics applications are readily available for both the IRIX OS with MIPS processors and the Linux IA-32 platforms. However, most of the software is parallelized only for shared-memory systems, with only a limited number parallelized for distributed memory systems such as clusters of systems running the IRIX OS with MIPS processors or Linux IA-32 systems. The scaling of both the software and the hardware, however, is currently limited to 16 CPUs for a Linux IA-32 system under a single-system image, while a system that runs the IRIX OS with MIPS processors can scale up to 1,024 CPUs at this time.

The general performance of bioinformatics applications depends upon the characteristics of the application used. For integer-intensive bioinformatics algorithms such as BLAST and FASTA, a system running the IRIX OS with MIPS processors is generally comparable in performance to a Linux IA-32 system. However, for floating-point-intensive bioinformatics algorithms such as HMMER or Wise2, a system running the IRIX OS with MIPS processors is generally faster in performance than a Linux IA-32 system. For example, Wise2 on a 500 MHz MIPS processor is more than 3.5 times faster than on a 1 GHz Intel Pentium® III processor.

Large Database Capabilities

In addition to scaling and performance, another important consideration is the ability of the system to handle the large database files that are integral to many bioinformatics applications. First, the ability to memory map and address large database files is necessary for good performance of database searching algorithms such as BLAST and FASTA. The sizes of public databases such as GenBank, however, now exceed the 32-bit addressing limit of 3.8GB on a Linux IA-32 system, making the 64-bit addressing capability of a system running the IRIX OS with MIPS processors a necessity for adequate performance of these algorithms. Otherwise, if a database file is larger than 3.8GB on a Linux IA-32 system, then programs such as BLAST will read the file from disk instead

of memory, resulting in degraded performance. Second, the ability to store large database files on disk is a basic requirement for any bioinformatics production system. Systems running the IRIX OS with MIPS processors use the XFS journaled filesystem, which supports individual file sizes up to 9 million terabytes (9 exabytes) and filesystem sizes up to 18 million terabytes (18 exabytes). This should provide adequate capacity for most sites. In contrast, Linux IA-32 systems generally use the Ext3fs filesystem, which supports file sizes only up to 2GB. This should be sufficient for protein databases and small genome databases, but will present a challenge for the latest nucleotide databases. For these large databases, one would first need to split each database into smaller parts and then decide how best to search each part separately and combine the final statistical results. An alternative to Ext3fs is XFS on Linux, which has been open sourced by SGI (<http://oss.sgi.com/projects/xfs>) and currently enables 64TB file sizes. However, Linux IA-32 users would still be limited by the 3.8GB memory limit. In summary, the 32-bit limit imposed by the Linux IA-32 system limits the database sizes that can be memory mapped and stored on disk.

Large Job Capabilities

The ability of a system to handle both large jobs and large numbers of jobs is also an important criterion when selecting a hardware system for bioinformatics. The large 64-bit shared memory and 64-bit filesystem combined with well-balanced CPU, memory, and I/O subsystems make the SGI Origin family servers particularly well-suited to both fast turnaround and high-throughput bioinformatics jobs. There are two additional features that benefit a system in a bioinformatics production environment, both of which are described in more detail in the next section:

Weightless priority—a feature that ensures the highest level of interactive performance while maintaining maximum system utilization with background jobs run with weightless priority

IRIX checkpoint and restart—a feature that ensures the successful completion of long-running calculations

Administration and Maintenance

Finally, the hidden cost of administration and maintenance of a system is often overlooked when selecting a new hardware system. In general, a system running the IRIX OS with MIPS processors with its stable OS, higher mean time between failures, and recognized support from SGI will require less administration and maintenance on the part of the customer. Furthermore, a single shared system such as the SGI Origin family server eliminates the complicated situation in a distributed cluster system of having to maintain multiple database copies on each cluster node. So although the cost is difficult to quantify, given its potentially large value (often exceeding the cost of the hardware), the total cost of ownership (including space and power consumption costs) should be seriously evaluated.

IRIX Features

Weightless Priority

Achieving a balance between interactive and noninteractive usage is an important challenge in managing production computing environments to ensure optimal efficiency in system utilization. In an ideal situation, jobs run interactively will always receive the highest (UNIX®) prioritization and they will be run as if they were the only job on the system. Performance benchmarks, including all the results in this report, are usually run in this highly interactive, one-job-per-machine mode.

In production environments, most systems are severely overloaded with jobs. In an oversubscribed system, interactive performance can be quite poor, even when some of the jobs have been designated to run in the background with lower OS priorities. Research systems supporting staff in multiple sites and multiple countries have a strong need to run jobs interactively on the most heavily utilized systems. The outcome is that machine response can degrade quickly if the situation is not addressed.

A unique feature available under IRIX 6.5.9¹ helps to alleviate this problem. Weightless prioritization of processes and threads allows the important, interactive jobs to run at the highest level of priority, while background weightless jobs are essentially sleeping and in no way affect the turnaround time of the interactive jobs. The system manager can designate the relative importance of different jobs and can ensure that sleeping or weightless jobs are constantly ready to take advantage of available cycles not being utilized by the higher priority, interactive jobs.

The following is an example of this system feature. An eight-processor SGI Origin family server was used for the following experiments. As background activity, eight single-processor FASTA jobs were running all the time, keeping all eight processors busy. The interactive job was modeled by a short parallel CLUSTAL W job running in parallel on eight processors (24 seconds on the stand-alone machine). When both foreground and background jobs are running, the server is essentially oversubscribed by a factor of 2. If no special measures are taken with respect to the FASTA jobs, the execution time for the CLUSTAL W job doubles (48 seconds). However, when the FASTA jobs are submitted with weightless priority, the turnaround time for the CLUSTAL W job is the same as when the job is run on an otherwise idle machine, 24 seconds.

Weightless processes and threads are a very powerful tool to ensure full utilization of computer systems in production environments. As seen in the example above, interactive jobs get full control of the computational resources over weightless processes. As the processes of the interactive jobs free system resources, lower priority jobs immediately make use of these relinquished resources.

Checkpoint and Restart

Another critical need for production computing is ensuring that long-running analytical jobs that take days, weeks, or even months are able to finish. Abnormal termination of these jobs due to electrical, system, or network outages or due to the needs of other users can result in weeks or months of wasted cycles. Applications software often takes into account these possibilities and on a regular basis writes out restart files. For those situations where either the application does not have this recovery mechanism or the checkpoint intervals are very large, the checkpoint and restart feature of IRIX 6.5 can help alleviate this problem.

An excerpt of the IRIX man page follows:

NAME cpr - checkpoint and restart processes;

DESCRIPTION

IRIX Checkpoint and Restart (CPR) offers a set of user-transparent software management tools, allowing system administrators, operators, and users with suitable privileges to suspend a job or a set of jobs in mid-execution and restart them later on. The jobs may be running on a single machine or on an array of networking connected machines. CPR may be used to enhance system availability, provide load and resource control or balancing, and facilitate simulation or modeling.

Checkpoint and restart capabilities allow users to save the state of long-running jobs. This allows for on-demand access to the entire machine for high-priority jobs without losing the state of jobs in progress. Long-running jobs may be protected against failure by periodic checkpointing. Using checkpoint and restart as part of a regularly occurring IRIX `cron` utility enables long-running jobs to suspend, write an image of the job to disk, and then resume execution. If during the ensuing period between `cron` jobs the system fails, the job can be resumed from the checkpoint file. Thus, during a month-long job the only part of a calculation that fails would be from the time between `cron` jobs. Scheduling the checkpoint period to be 24 hours, for example, would ensure that less than one day's calculation is lost.

¹Weightless threads are available under IRIX 6.5.9 and 6.5.10 with a patch; incorporated as part of 6.5.11.

Acknowledgments

BLAST

Dr. Warren Gish
Washington University
<http://blast.wustl.edu/blast/README.html>

Dr. Thomas Madden
National Center for Biotechnology Information
<http://www.ncbi.nlm.nih.gov/BLAST>

CLUSTAL W

Dr. Toby Gibson
European Molecular Biology Laboratory

Dr. Des Higgins
University of County Cork

Dr. Julie Thompson
Institut de Génétique et de Biologie Moléculaire et Cellulaire
<http://www-igbmc.u-strasbg.fr/BioInfo/ClustalW/Top.html>

d2-CLUSTER

Dr. Winston Hide
South African National Bioinformatics Institute, University of Western Cape
<http://www.sanbi.ac.za/>

FASTA

Dr. William R. Pearson
University of Virginia
<http://www.med.virginia.edu/~wrp/pearson.html>

HMMER

Dr. Sean Eddy
Washington University
<http://hmmerr.wustl.edu/>

Wise2

Dr. Ewan Birney
The Sanger Centre
<http://www.sanger.ac.uk/Software/Wise2>

© 2001 Silicon Graphics, Inc. All rights reserved. Specifications subject to change without notice. Silicon Graphics and IRIX are registered trademarks and SGI, NUMAflex, Origin, XFS, and the SGI logo are trademarks of Silicon Graphics, Inc. MIPS is a registered trademark and R12000 and R14000 are trademarks of MIPS Technologies, Inc., used under license by Silicon Graphics, Inc. Intel and Pentium are registered trademarks of Intel Corporation. Linux is a registered trademark of Linus Torvalds. UNIX is a registered trademark of The Open Group in the U.S. and other countries. All other trademarks mentioned herein are the property of their respective owners.