

SLALOM: Is Your Computer on the List?

If not, we'd like it to be

John Gustafson, Diane Rover,
Stephen Elbert, and Michael Carter
Ames Laboratory, Ames, IA 50011

Overview

Eight months ago, when the SLALOM benchmark was introduced in *Supercomputing Review*, we were able to show the performance of about 20 computers. That list is now approaching 100 entries, and this month we will present not only the actively marketed computers but some well-known computers that are no longer actively marketed. Only Dongarra's LINPACK list has more entries, and no other benchmark based on *complete application measurement* has as many machines... or as wide a variety. The SLALOM list has the Intel Touchstone Delta, the Macintosh LC, the largest CRAY, the IBM workstations, and the MasPar data-parallel computers, all under a single comparison. These highly disparate architectures can all be compared using the concept of *fixed-time* benchmarking: Run the largest problem possible in under one minute, and use the *problem size* as the figure of merit.

Some people have said that SLALOM is a parallel computer benchmark. It's nothing of the kind. In fact, the backsolving of the equations and the writing of the solution to disk are proving to be major challenges for the parallel machines. SLALOM accommodates any architecture, any language, a very wide range of performance, and any native word size... so yes, it runs on parallel computers. There are at least two dozen entirely different high-performance architectures on the list.

Perhaps the most startling news is that a Japanese-made *uniprocessor* now tops the list. The Siemens S600/20, equivalent to a top-of-the-line Fujitsu computer, climbed past the CRAY Y-MP/8. As many people have pointed out, the term "uniprocessor" might be a misnomer for a machine with enough pipelines to deliver eight multiplications and eight additions every 3.2 nanoseconds! In fact, Japanese computers now bracket the list, with a Fujitsu supercomputer at the top and a Toshiba laptop computer at the bottom.

The Intel iPSC/860 version has been well tuned by people at the Intel Supercomputer Division in Beaverton, and has come up to about 5 MFLOPS per processor. The Touchstone Delta system at Caltech was able to reach 4320 patches, or roughly 1.3 GFLOPS. That run used only 256 of its 512 processors. At the top of the list, the parallel computers continue to threaten, but not overtake, the most expensive vector supercomputers.

Most Wanted List

We haven't heard from everyone yet. Our "most wanted" computers in the SLALOM table include those made by the following vendors:

- Active Memory Technologies (AMT)
- BBN Advanced Computers
- Hewlett-Packard (HP)
- MEIKO
- Nippon Electric Corporation (NEC)
- Stardent
- Thinking Machines (TMI)
- TransTech
- WaveTracer

We hope to add these and other computers to our list by the next time we publish in *Supercomputing Review*.

How to Get SLALOM

SLALOM resides on a Unix workstation at Ames Lab, `tantalus.al.iastate.edu`. For those of you without a nameserver, that's IP address 129.186.200.15. If you connect to this computer through the networks via "ftp", just answer "ftp" to the "username:" prompt, and a carriage return to the "password:" prompt, and you're in. Use your usual ftp commands to peruse the directories and files you find there, downloading whatever interests you. Among other things, you'll find

- Up-to-date reports of all computers measured so far
- Programs for displaying the answer graphically
- Concise definitions of the problem to solve, in Fortran, C, and Pascal
- Parallel versions for SIMD and MIMD environments
- Vectorized versions for traditional pipelined supercomputers
- Examples of answer files for checking your results

If your only network access is e-mail, send a note to `netlib@tantalus.al.iastate.edu`, and a case-sensitive version of the netlib software will mail you back instructions. Please don't ask for a tape, a listing, or "just send me everything!" If you don't know exactly what you want, find a friend on the Internet.

Historical Note

Sometimes we hear people say, "The only performance figure that matters is how long it takes to run my application." But what people *say* matters to them and how they *use* higher performance are two different things. It might be more accurate to say, "The only performance figure that matters is the problem size I can solve in the time I'm willing to wait." Consider the following quotations concerning examples of computing tasks, taken from historical treatises on computing [12]:

The determination of the logarithm of any number would take 2 minutes, while the evaluation of a^n (for any value of n) by the exponential [*sic*] theorem, should not require more than $1\frac{1}{2}$ minutes longer—all results being of twenty figures.

—*On a Proposed Analytical Machine*
P. Ludgate, 1878

The work of counting or tabulating on the machines can be so arranged that, within a few hours after the last card is punched, the first set of tables, including condensed grouping of all the leading statistical facts, would be complete.

—*An Electric Tabulating System*
H. Hollerith, 1889

Since an expert [human] computer takes about eight hours to solve a full set of eight equations in eight unknowns, k is about $1/64$. To solve twenty equations in twenty unknowns should thus require 125 hours... The solution of general systems of linear equations with a number of unknowns greater than ten is not often attempted.

—*Computing Machine for the Solution of large Systems of Linear Algebraic Equations*
J. Atanasoff, 1940

Another problem that has been put on the machine is that of computing the position of the moon for any time, past or future... Time required: 7 minutes.

—*Electrons and Computation*
W. J. Eckert, 1948

...13 equations, solved as a two-computer problem, require about 8 hours of computing time. The time required for systems of higher order varies approximately as the cube of the order. This puts a practical limitation on the size of systems to be solved... It is believed that this will limit the process used, even if used iteratively, to about 20 or 30 unknowns.

—*A Bell Telephone Laboratories Computing Machine*
F. Alt, 1948

Tracking a guided missile on a test range... is done on the International Business Machines (IBM) Card-Programmed Electronic Calculator in about 8 hours, and the tests can proceed.

—*The IBM Card-Programmed Electronic Calculator*
J. W. Sheldon and L. Tatum, 1952

Computer speeds have increased by many orders of magnitude over the last century, but human patience is unchanging. The computing jobs cited in publications typically take from minutes to hours, whether the computer uses gears, vacuum tubes, or VLSI. Pick any fixed-size benchmark, and it will soon be obsoleted by hardware advances that make the benchmark ridiculously small. People tend to forget the numerator in the ratio that defines the “speed” of computing. Give a scientist a faster supercomputer, and it will be put to use solving a problem not previously attempted... not reducing the execution time of last year’s problem.

A Scalable Benchmark for Scalable Computers

A given make of parallel processor can offer a performance range of over 8000 to 1, so the scaling issue exists even if applied to a computer of current vintage.

It’s not easy to use conventional benchmark techniques on every possible size of a large parallel ensemble like an nCUBE or an Intel computer. Usually you’ll see footnotes like, “We were unable to run the problem on small numbers of processors because of insufficient memory.” Or the performance graph is given as a collage of incomplete curves, each for a particular problem size. The use of the fixed-time method simplifies the issue by changing the question.

As Figure 1 shows, even computers that scale by 1024 to 1 can be compared using SLALOM. *None* of the machines in our database have had insufficient memory to run for one minute.

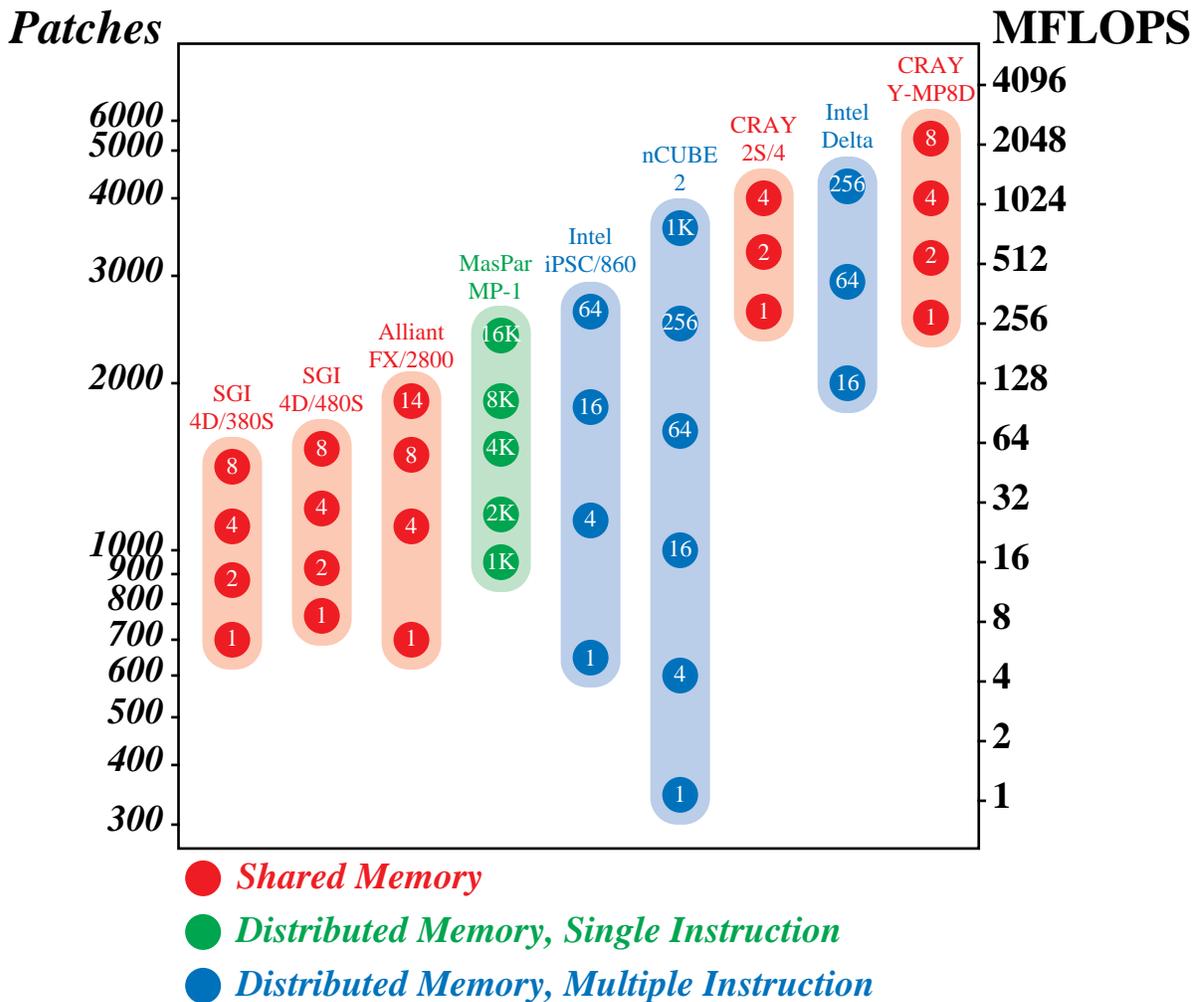


Figure 1. SLALOM Performance for Parallel Product Families

The fixed-time benchmark concept is *not* the same as generic rate comparisons, such as “transactions per second,” “logical inferences per second,” or “spin updates per second.” In fixed-time performance comparison, a complete computing job is scaled to fit a given amount of time, whereas rate comparisons use the asymptotic speed of a supposedly generic task. As with MFLOPS or MIPS metrics, generic rate comparisons are usually vague in defining the unit of work in the numerator. Floating-point operations, instructions, transactions, logical inferences, and spin updates come in many different sizes and varieties. True fixed-time benchmarking considers the entire application. A complete application usually contains many different work components with different scaling properties.

The Report

There are now 82 computer configurations in the “Actively Marketed” list.

The SLALOM Benchmark Report

S calable
L anguage-independent
A mes
L aboratory
O ne-minute
M easurement

The following ranks computers that are actively marketed. All runs are very close to 60 seconds.

Machine, environment	Processors	Patches	MFLOPS	Measurer	Date
Seimens S600/20, 312 MHz, Fortran 77+LAPACK	1	5610	3065.	A.Rohnfelder(v), KF Karlsruhe	4/22/91
Cray Y-MP8D, 167 MHz, Fortran+LAPACK (Strassen)	8	5120	2130.	J. Brooks (v), Cray Research	9/21/90
Intel Delta (i860) 40 MHz, Fortran+coded Daxpy	256	4320	1260.	E. Kushner (v), Intel	5/30/91
Cray-2S/4, 244 MHz, Fortran+LAPACK (Strassen)	4	4204	1160.	M. Ess (v), Cray Computer	5/27/91
Cray Y-MP8D, 167 MHz, Fortran+LAPACK (Strassen)	4	4096	1190.	J. Brooks (v), Cray Research	9/21/90
nCUBE 2, 20 MHz, Fortran+assembler	1024	3736	821.	J. Gustafson, Ames Lab	2/8/91
Cray-2S/4, 244 MHz, Fortran+LAPACK (Strassen)	2	3280	560.	M. Ess (v), Cray Computer	5/27/91
Cray Y/MP-8D, 167 MHz, Fortran+LAPACK (Strassen)	2	3200	557.	J. Brooks (v) Cray Research	9/21/90
Intel Delta (i860) 40 MHz, Fortran+coded Ddot	64	3120	487.	E. Kushner (v), Intel	5/30/91
Siemens S400/10, 125 MHz, Fortran+various opts.	1	2738	285.	F. Schmitz, KFK	2/21/91
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	64	2640	299.	E. Kushner (v), Intel	5/24/91
Fujitsu VP400-EX, 71 MHz, Fortran+various opts	1	2598	283.	F. Schmitz, KFK	3/12/91
Cray-2S/4, 244 MHz, Fortran+LAPACK (Strassen)	1	2588	279.	M. Ess (v), Cray Computer	5/27/91
Cray Y/MP-8D, 167 MHz, Fortran+LAPACK (Strassen)	1	2560	283.	J. Brooks (v), Cray Research	9/21/90
nCUBE 2, 20 MHz, Fortran+assembler	256	2506	253.	J. Gustafson, Ames Lab	2/8/91
MasPar MP-1, 12.5 MHz, parallel C+assembler	16384	2431	232.	W. Baugh (v), MasPar	5/28/91
Intel Delta (i860) 40 MHz, Fortran+coded Ddot	16	1986	129.	E. Kushner (v), Intel	5/30/91
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	32	1920	118.	E. Kushner (v), Intel	1/25/91
MasPar MP-1, 12.5 MHz, parallel C+assembler	8192	1919	109.	W. Baugh (v), MasPar	5/31/91
IBM 3090/200J VF, 69 MHz, VS Fortran 2.4+ESSL	1	1834	105.	J. Shearer (v), IBM	5/31/91
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	16	1830	102.	E. Kushner (v), Intel	5/24/91
Alliant FX/2800, Fortran+KAI Libraries	14	1736	89.3	J. Perry (v), Alliant	1/24/90
nCUBE 2, 20 MHz, Fortran+assembler	64	1623	71.6	J. Gustafson, Ames Lab	4/8/91
IBM RS/6000 550, 42 MHz, Fortran+ESSL	1	1610	63.5	J. Shearer (v), IBM	5/31/91
MasPar MP-1, 12.5 MHz, plural C+assembler	4096	1535	63.5	M. Carter, Ames Lab	4/8/91
Hitachi EX60+IVF, 61 MHz, IBM VS Fortran+ESSL	1	1522	61.2	J. Coyle, ISU	5/21/91
Alliant FX/2800, Fortran+KAI Libraries	8	1502	58.9	J. Perry (v)	1/24/90
Silicon Graphics 4D/480S, 40 MHz, Fortran	8	1500	59.0	O. Schreiber (v), SGI	4/2/91
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	8	1392	46.8	E. Kushner (v), Intel	1/25/91
Silicon Graphics 4D/380S, 33 MHz, Fortran	8	1352	46.5	O. Schreiber (v), SGI	4/2/91
IBM RS/6000 530, 25 MHz, Fortran+ESSL	1	1347	43.4	J. Shearer (v), IBM	5/31/91
IBM RS/6000 540, 30 MHz, Fortran+ESSL	1	1337	42.3	J. Shearer (v), IBM	5/15/91
FPS M511EA, 33 MHz, Fortran+LAPACK	1	1197	30.2	B. Whitney (v), FPS	1/24/91
MasPar MP-1, 12.5 MHz, parallel C+assembler	2048	1183	29.9	M. Carter, Ames Lab	4/8/91
Silicon Graphics 4D/480S, 40 MHz, Fortran	4	1164	28.7	O. Schreiber (v), SGI	4/2/91
Alliant FX/2800, Fortran+KAI Libraries	4	1139	26.9	J. Chmura (v), Alliant	12/7/90
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	4	1138	25.8	E. Kushner (v), Intel	5/24/91
Silicon Graphics 4D/380S, 33 MHz, Fortran	4	1128	26.1	O. Schreiber (v), SGI	4/2/91
IBM RS/6000 520, 20 MHz, Fortran+ESSL	1	1091	23.8	J. Shearer (v), IBM	1/9/91
nCUBE 2, 20 MHz, Fortran+assembler	16	1017	18.7	J. Gustafson, Ames Lab	4/8/91
MasPar MP-1, 12.5 MHz, parallel C+assembler	1024	959	16.2	M. Carter, Ames Lab	4/8/91

Machine, environment	Processors	Patches	MFLOPS	Measurer	Date
Silicon Graphics 4D/480S, 40 MHz, Fortran	2	908	14.4	O. Schreiber (v), SGI	4/2/91
IBM RS/6000 320, 20 MHz, Fortran+block Solver	1	895	13.7	S. Elbert, Ames Lab	1/30/91
Silicon Graphics 4D/380S, 33 MHz, Fortran	2	884	13.4	O. Schreiber (v), SGI	4/2/91
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	2	845	11.4	E. Kushner (v), Intel	2/5/91
SKYbolt, 40 MHz i860/i960, C+assembler Ddot	1	831	11.1	C. Boozer (v), SKY Computers	1/9/91
SKYstation, 40 MHz, C+assembler Ddot	1	793	9.77	C. Boozer (v), SKY Computers	1/28/91
Convex C220, Fortran+various opts.	1	760	8.24	P. Hinker, LANL	2/14/91
Silicon Graphics 4D/480S, 40 MHz, Fortran	1	758	8.66	O. Schreiber (v), SGI	4/2/91
Silicon Graphics 4D/35, 37 MHz, Fortran	1	739	8.07	O. Schreiber (v), SGI	4/2/91
Silicon Graphics 4D/380S, 33 MHz, Fortran	1	700	6.96	O. Schreiber (v), SGI	4/2/91
Alliant FX/2800, Fortran	1	693	6.76	J. Chmura (v), Alliant	12/7/90
Intel iPSC/860, 40 MHz, Fortran+coded Ddot	1	647	5.46	E. Kushner (v), Intel	1/25/91
FPS-500 (33 MHz MIPS+vec. unit), Fortran	1	619	4.97	P. Hinker, LANL	11/12/90
nCUBE 2, 20 MHz, Fortran+assembler	4	617	4.63	J. Gustafson, Ames Lab	2/8/91
SUN 4/490, 25 MHz, C	1	542	3.25	I. Novack, JPL	5/15/91
DECStation 5000, 25 MHz, Fortran	1	534	3.25	S. Elbert, Ames Lab	1/30/91
Silicon Graphics 4D/25, 20 MHz, Fortran+block Solver	1	507	2.83	S. Elbert, Ames Lab	1/30/91
SPARCStation 2GX, C	1	505	2.69	C. Boozer, SKY Computers	2/6/91
Solbourne 5E/930, 40 MHz, C	1	461	2.25	I. Novack, JPL	5/15/91
SUN 4/370, 25 MHz, C	1	451	1.97	J. Gustafson, Ames Lab	4/9/91
Solbourne 5/620, 25 MHz, C	1	442	2.02	I. Novack, JPL	5/15/91
DECStation 5000, 25 MHz, Pascal	1	432	1.82	D. Rover, Ames Lab	1/31/91
DECStation 3100, 16.7 MHz, Fortran+block Solver	1	418	1.70	S. Elbert, Ames Lab	1/30/91
Si. Graphics 4D/20, 12.5 MHz, Fortran+block Solver	1	401	1.52	S. Elbert, Ames Lab	1/30/91
SUN 4/370, 25 MHz, Fortran	1	397	1.41	J. Gustafson, Ames Lab	4/9/91
DECStation 2100, 12.5 MHz, Fortran+block Solver	1	377	1.29	S. Elbert, Ames Lab	1/30/91
SUN 4/060 SPARC I, 25 MHz, C	1	358	1.06	I. Novack, JPL	5/15/91
nCUBE 2, 20 MHz, Fortran+assembler	1	354	1.13	J. Gustafson, Ames Lab	8/13/90
Motorola MVME181 (20 MHz 88000), Fortran	1	289	0.676	R. Blech, NASA	10/17/90
Sequent Symmetry, 33 MHz, C	1	253	0.479	M. Carter, Ames Lab	1/3/91
Mac Ilfx, (40 MHz 68030+68882), Think C	1	235	0.357	J. Gustafson, Ames Lab	5/10/91
Amiga 3000 (25 MHz 68030+68882) SAS C 5.10a	1	230	0.336	R. Bless, U of Karlsruhe	4/13/91
Mac Ilci, (25 MHz 68030+68882) Think C	1	190	0.211	J. Gustafson, Ames Lab	5/10/91
VAXStation 3520, C	1	181	0.197	M. Carter, Ames Lab	1/24/91
Mac Ilsi, (20 MHz 68030+68882), Think C	1	175	0.170	J. Gustafson, Ames Lab	5/16/91
Mac SE/30, (16 MHz 68030+68882)	1	163	0.143	J. Gustafson, Ames Lab	5/10/91
Cogent XTM (T800 Transputer, 20 MHz) Fortran	1	149	0.133	C. Vollum (v), Cogent	6/11/90
Mac Ilsi, (20 MHz 68030 only) Think C	1	73	0.0219	J. Gustafson, Ames Lab	5/10/91
Mac LC, (16 MHz 68020 only) Think C	1	34	0.0042	J. Gustafson, Ames Lab	5/15/91
Amiga 2000 (7 MHz 68000), SAS C 5.10a	1	32	0.00363	R. Bless, U of Karlsruhe	4/24/91
Toshiba 1000, 6 MHz 8088, Turbo C	1	12	0.000646	P. Hinker, LANL	11/14/90

NOTES:

A "(v)" after the name of the person who made the measurement indicates a vendor. Vendors frequently have access to compilers, libraries, and other tools that make the performance higher than that achievable by a customer.

Intel entries for 8 and 32 nodes used a one-dimensional scattered decomposition; other Intel and nCUBE entries used a two-dimensional scattered decomposition that currently works only for even-dimensional hypercubes.

The IBM RS/6000 workstations were not all measured using the same algorithm. Be careful not to compare machines submitted on different dates even when all other information is identical. A recent improvement to the SetUp routines by J. Shearer allowed the 25 MHz model 530 to surpass the older algorithm on a 30 MHz model 540.

If MFLOPS seem inconsistent with preceding/following entries, it is because either the number of seconds is significantly less than 60 or a different version of the algorithm was used. Operation counts are reduced as more efficient methods are found. Rankings are by patch count, not MFLOPS.

Performance within a product line

The “speedup” column is the ratio of the MFLOPS rate to that of the smallest member of the product line for which we have SLALOM measurements. Since MFLOPS are a poor method of assessing performance, the speedup column should be viewed only as a rough guide to the scalability of a product line via parallel processing. This form of speedup can be greater than the number of processors because faster computers spend a greater fraction of the time on the Solver, raising the MFLOPS rate per processor. This “changing profile” effect, noted in past SLALOM reports, tends to compensate for the increasing communication and load imbalance that result from using more processors.

Machine, environment	Processors	Patches	MFLOPS	Measurer	Date	“Speedup”
Cray Y-MP8D, 167 MHz	8	5120	2130.	J. Brooks (v)	9/21/90	7.53
Cray Y-MP8D, 167 MHz	4	4096	1190.	J. Brooks (v)	9/21/90	4.20
Cray Y-MP8D, 167 MHz	2	3200	557.	J. Brooks (v)	9/21/90	1.97
Cray Y-MP8D, 167 MHz	1	2560	283.	J. Brooks (v)	9/21/90	1.00
Intel Delta (i860) 40	256	4320	1260.	E. Kushner (v)	5/30/91	9.77
Intel Delta (i860) 40	64	3120	487.	E. Kushner (v)	5/30/91	3.78
Intel Delta (i860) 40	16	1986	129.	E. Kushner (v)	5/30/91	1.00
Cray-2S/4, 244 MHz	4	4204	1160.	M. Ess (v)	5/27/91	4.16
Cray-2S/4, 244 MHz	2	3280	560.	M. Ess (v)	5/27/91	2.00
Cray-2S/4, 244 MHz	1	2588	279.	M. Ess (v)	5/27/91	1.00
nCUBE 2, 20 MHz	1024	3736	821.	J. Gustafson	2/8/91	727.
nCUBE 2, 20 MHz	256	2506	253.	J. Gustafson	2/8/91	224.
nCUBE 2, 20 MHz	64	1623	71.6	J. Gustafson	4/8/91	63.4
nCUBE 2, 20 MHz	16	1017	18.7	J. Gustafson	4/8/91	16.5
nCUBE 2, 20 MHz	4	617	4.63	J. Gustafson	2/8/91	4.10
nCUBE 2, 20 MHz	1	354	1.13	J. Gustafson	8/13/90	1.00
Intel iPSC/860, 40 MHz	64	2640	299.	E. Kushner (v)	5/24/91	54.8
Intel iPSC/860, 40 MHz	16	1830	102.	E. Kushner (v)	5/24/91	18.7
Intel iPSC/860, 40 MHz	4	1138	25.8	E. Kushner (v)	5/24/91	4.73
Intel iPSC/860, 40 MHz	1	647	5.46	E. Kushner (v)	1/25/91	1.00
MasPar MP-1, 12.5 MHz	16384	2431	232.	B. Baugh (v)	5/28/91	14.3
MasPar MP-1, 12.5 MHz	8192	1855	109.	M. Carter	4/7/91	6.73
MasPar MP-1, 12.5 MHz	4096	1535	63.5	M. Carter	4/8/91	3.92
MasPar MP-1, 12.5 MHz	2048	1183	29.9	M. Carter	4/8/91	1.85
MasPar MP-1, 12.5 MHz	1024	959	16.2	M. Carter	4/8/91	1.00
Alliant FX/2800	14	1736	89.3	J. Perry (v)	1/24/90	13.2
Alliant FX/2800	8	1502	58.9	J. Perry (v)	1/24/90	8.71
Alliant FX/2800	4	1139	26.9	J. Chmura (v)	12/7/90	3.98
Alliant FX/2800	1	693	6.76	J. Chmura (v)	12/7/90	1.00
Silicon Graphics 4D/480S	8	1500	59.0	O.Schreiber (v)	4/2/91	6.81
Silicon Graphics 4D/480S	4	1164	28.7	O.Schreiber (v)	4/2/91	3.31
Silicon Graphics 4D/480S	2	908	14.4	O.Schreiber (v)	4/2/91	1.66
Silicon Graphics 4D/480S	1	758	8.66	O.Schreiber (v)	4/2/91	1.00
Silicon Graphics 4D/380S	8	1352	46.5	O.Schreiber (v)	4/2/91	6.68
Silicon Graphics 4D/380S	4	1128	26.1	O.Schreiber (v)	4/2/91	3.75
Silicon Graphics 4D/380S	2	884	13.4	O.Schreiber (v)	4/2/91	1.93
Silicon Graphics 4D/380S	1	700	6.96	O.Schreiber (v)	4/2/91	1.00

Computers No Longer Marketed

From time to time, we will publish lists of SLALOM performance for computers that are no longer actively marketed. We feel that current and historical computers should not be mixed in the same list, so we intend to move entries from the main list to this one when we learn that a particular model has been superseded or is no longer available from the original vendor.

Machine, environment	Processors	Patches	MFLOPS	Measurer	Date
Seimens S600/20, 312 MHz, Fortran 77+LAPACK	1	5610	3065.A.Rohnfelder(v), KF Karlsruhe		4/22/91
Myrias SPS2 (17 MHz 68020), Fortran	64	399	1.56	J. Roche (v) Myrias	6/21/90
nCUBE 1, 6 MHz,CFG Fortran+assembler	4	204 59.78	0.281	J. Gustafson, Ames Lab	4/30/90
Mac Ilcx, 16 MHz 68030 +68882, Think C, V4.00 (68030+68881 enabled)	1	162 59.32	0.142	J. Gustafson Ames Lab	5/10/91
nCUBE 1, 6 MHz, CFG Fortran 1.7 + Assembler	2	153 59.44	0.141	J. Gustafson Ames Lab	4/30/90
VAX 11/780, VMS 5.3-1 Fortran (fort/f77/nodebug)	1	140 59.30	0.115	I. Novack JPL	5/15/91
Mac Plus, 16MHz, MC68030 +68882, Symantic Pascal v3	1	124 59.37	0.0863	J. McInerney Novellus	1/29/91
nCUBE 1, 6 MHz, CFG Fortran 1.7 + Assembler	1	114 59.84	0.0703	J. Gustafson Ames Lab	4/30/90
IBM PC-AT, 8 MHz 80286 + 80287 CFG Fortran 1.7	1	67 58.64	0.0216	J. Gustafson Ames Lab	4/30/90
Zenith PC-AT, 6MHz 80286 + 80287 MS QuickPascal v1	1	55 59.15	0.0140	D. Rover Ames Lab	12/6/90
Mac Ilcx, 16 MHz 68030 only, Think C, V4.00 (no coprocessor)	1	44 57.40	0.00730	J. Gustafson Ames Lab	5/10/91
Mac Plus, 16MHz, MC68030 Symantic Pascal v3	1	32 57.57	0.00451	J. McInerney Novellus	1/29/91
Mac Plus, 8 MHz, MC68000 Symantic Pascal v3	1	12 56.05	0.000622	J. McInerney Novellus	1/29/91

Acknowledgments

We thank everyone who has participated in this effort. In particular, analysts at Alliant, Cogent, Cray, IBM, Intel, MasPar and Myrias have contributed suggestions, ideas, and versions of the SLALOM program. Much of the work was performed at the Scalable Computing Laboratory at Ames Laboratory/Center for Physical and Computational Mathematics.

*This work is supported by the Applied Mathematical Sciences Program of the Ames Laboratory-U.S. Department of Energy under contract number W-7405-ENG-82.

REFERENCES

1. V. Faber, O. Lubeck, and A. White, "Superlinear Speedup of an Efficient Sequential Algorithm is Not Possible," *Parallel Computing*, Volume 3, 1986, pages 259–260.
2. J. L. Gustafson, "Reevaluating Amdahl's Law," *Communications of the ACM*, Volume 31, Number 5, May 1988.
3. D. P. Helmbold. and C. E. McDowell, "Modeling Speedup(n) greater than n," 1989 *International Conference on Parallel Processing Proceedings*, 1989, Volume III, pages 219–225.
4. D. Parkinson, "Parallel Efficiency can be Greater than Unity," *Parallel Computing*, Volume 3, 1986, pages 261–262.
5. B. Randell, editor, *The Origins of Digital Computers: Selected Papers*, Second Edition, Springer-Verlag, 1975, pages 84, 138, 227, 229, 283, and 306.