AIPS MEMO NO. 48

Benchmarking AIPS on a VAX 8800.

Mark Calabretta, Paul Rayner

CSIRO Division of Radiophysics,
P.O. Box 76, Epping NSW 2121, Australia.

22 September 1986

Abstract

The 15APR86 AIPS DDT benchmarking package was run on a VAX 8800 and several smaller DEC computers, the VAX 11/750, VAX 11/780, VAX 8200, and VAX 8500. (Results of an arithmetic benchmark unrelated to AIPS are also presented). As a dual processor machine, we considered the VAX 8800 in terms of its peak throughput processing power (11 x VAX 11/780), and also its ability to satisfy an individual AIPS user (4 x VAX 11/780). The IO performance of a VAX 8800 running AIPS cannot quite satisfy its enhanced CPU processing capability, but this shortcoming diminishes as more and more AIPS jobs are loaded onto it.

1 INTRODUCTION

The Commonwealth Scientific and Industrial Research Organization (CSIRO) of Australia is currently building an aperture synthesis radiotelescope known as the Australia Telescope (AT). Presently under consideration is the choice of an offline computer to support AIPS (Astronomical Image Processing System), which is to be used for the data reduction.

Digital Equipment Corporation have suggested a VAX 8800 as a solution to our processing requirements, and made available to us their facilities at Marlboro, Massachusetts for benchmarking purposes. During the course of our work we had access to, and benchmarked, a number of smaller VAX computers. The AT will have two VAX 8200's as synchronous and asynchronous control computers, the latter running AIPS, and we were interested to see how it would perform.

Results for a VAX 11/750 and VAX 11/780 were obtained from computers currently operated by CSIRO and the Anglo-Australian Observatory in Sydney.

The AIPS DDT (or PFT) benchmarking procedures have already been discussed at length in AIPS memos, numbers 36, 38 and 44. These will be taken as a prerequisite of this memo.

2 DETAILS OF THE DDT BENCHMARK

The hardware and software configuration adopted for the benchmark is summarized in Table 1. On arrival at Marlboro the

TABLE 1. The hardware and software configuration.

Hardware:

VAX 8800 with 112 Mbytes of memory (224k pages),
or VAX 8650 with 68 Mbytes
or VAX 8500 with 20 Mbytes
or VAX 8200 with 24 Mbytes

Four RA 81 disk drives,
One HSC 50 high speed disk controller,
One TU 80 tape drive.

Operating system parameters for the VAX 8800:

Version	VMS 4.4
Working set size	8192 pages
Working set extent	16000 pages
File high-water marking	(see text)

VAX 8800 was not immediately available, and we used a VAX 8500 and VAX 8650 for installing AIPS. Later, when we were finished with the VAX 8800, we were given a VAX 8200 to benchmark. We always had exclusive use of the hardware that had been assigned to us.

DEC claimed that FORTRAN optimization had been debugged in release 4.4 of the compiler and requested that we use it for the benchmark, which we did without any problems. We also ran no-optimized for comparison with results published in the previous AIPS memos and found that the optimized code ran some 20% faster in CPU time.

Several runs were made to test aspects of the configuration or environment which could affect the benchmark results. We tested for reproducibility of the results, and for differences in running the benchmark from a terminal versus VMS batch. Some time into the benchmark we discovered that file high-water marking was enabled; after disabling it we tested to see what its effect might have been. None of these factors produced any significant differences.

In general we did our tests using the small DDT problem, but also ran the medium problem for comparison. The large problem would have absorbed too much of our limited time. Table 2 characterizes the different size problems.

We ran DDT only in mode "M", for which the tasks always start with a master image supplied on the DDT tape. In mode "T", the alternative, successive tasks use the results of preceeding tasks, thereby measuring the accumulation of error.

In running the 15APR86 version of DDT we were aware that significant changes were to have been introduced in the 15JUL86 AIPS. NRAO kindly supplied us with a pre-release of the 15JUL86 AIPS, and an updated set of DDT procedures and master files. However, although we got the 15JUL86 version to run to completion, we found that the master image files supplied did not correspond to the adverbs of the DDT procedures, thus causing complete disagreement between the computed and master images for a few of the tasks. On the other hand, we noted that there was no significant difference in times between the two versions, and so continued to use 15APR86 which we trusted more.

TABLE 2. The three sizes of DDT problems.

Size	Visibilities	Image size	Clean components
Small	8000	256 x 256	2000
Medium	13000	512 x 512	5000
Large	53000	1024 x 1024	15000

The architecture of the VAX 8800 needs to be understood in order to appreciate the DDT results. To the casual user, the VAX 8800 looks like any other VAX series computer. For example, it has a standard VAX instruction set, so executable modules can be transferred to it from any other VAX. However, it differs from most other VAXes in having two CPUs, each of which access the same memory. In hardware terms both CPUs are equal, and under a later release of VMS they will be treated equally by the operating system, although we understand that VMS won't support a single process accessing more than one processor at a given instance. Under VMS 4.4, however, the "PRIMARY" processor handles all kernel mode operations (principally IO) and, on becoming compute bound, swaps its work load to the "ATTACHED" processor. This swapping is very fast since only the register contents and some other control information needs to be passed between processors via a high speed internal bus. However, since IO can only be performed by the PRIMARY processor, both processors will be under-utilized at times when their IO requirements clash.

Being a dual-processor machine, the VAX 8800 needs at least two processes to keep both of its CPUs occupied. Thus, unlike the benchmarks described in previous AIPS memos, we needed to see how the VAX 8800 would perform when loaded with at least two DDTs. Moreover, as we discovered later, the VAX 8800 is slightly IO bound in running AIPS, thus leaving the CPUs idle unless there are other processes to fill in the gaps. Hence, we studied the performance of the VAX 8800 as it was loaded with one, two, three, and then four simultaneous AIPS jobs.

The set up for running multiple AIPS differed slightly from a normal AIPS configuration. In particular, there were four identical data areas each with its own set of DDT image files and AIPS system files. The four VMS batch files which initiated the DDT jobs were all identical, save for redefining the logical names DA00 and DA01 to point uniquely to one of the four data areas. This was simpler than setting up four DDTs under separate user numbers, and also gave us the convenience of a separate AIPS accounting file for each. Scratch files would have been distributed as usual by each DDT over the four disks available. The multiple streams were initiated nearly simultaneously by using VMS batch procedures, and they generally finished at about the same time.

We encountered only one significant problem in running multiple DDTs. It arose in connection with restarting task APCLN, (referred to as "APRES" in DDT parlance). Task SUBIM, which creates the input image for restarting APCLN, has OUTSEQ set to 0, but APCLN itself has it specifically set to 1 on restart. The first job stream to execute SUBIM would create an image file with SEQ = 1 as anticipated, but subsequent streams would all set SEQ > 1. When APCLN attempted to restart for these later streams, it would not find the original output image, and proceeded to clean off a further 2000 components. Unfortunately, we did not catch this bug in good time, but it should not have affected the benchmark results

seriously, since APRES is only a minor contributor, and furthermore, timings could be approximated from the one stream which did execute as intended.

Results of the DDT benchmark are given in table form in Appendix 1. Graphs of the same data are discussed below.

3 ARITHMETIC BENCHMARK

In the free time between running different variations of DDT, we ran a set of simple benchmarks completely unrelated to AIPS. One of these measured the time taken to perform basic machine functions such as multiply and divide. The results are presented in Appendix 2 where the first table gives the time in microseconds, and the second gives times relative to the VAX 11/780. The geometric mean of each column is given, and its inverse provides a figure of merit (FOM) for each machine. Note that the FOMs for the VAX 8300 and VAX 8800 should be multiplied by two, since the stated figure represents the contribution from only one processor. Note also that the double precision floating point format is "D".

4 RESULTS AND DISCUSSION

The VAX 8800 passed the DDT certification tests.

We were primarily interested in comparing the performance of the VAX 8800 against the Convex C-1. So far we have no 15APR86 DDT results for the Convex, so it was necessary to make comparisons against the 15APR85 PFT results published in previous AIPS memos. There are significant differences between the DDT and PFT benchmark packages. However, we could compare each machine against a VAX 11/780 (the usual standard), since one was included in each benchmark. A difficulty arose in connection with optimization of the AIPS code, however, since neither the results presented here for the VAX 11/780 nor those of previous AIPS memos are for optimized code. The form chosen to scale the unoptimized results for comparison purposes was

$$\text{Corrected CPU time} = 0.8226 * \text{CPU},$$
$$\text{Corrected real time} = \text{Real} - 0.1774 * \text{CPU}.$$

Thus, the corrected real time for the VAX 11/780 running the small 15APR86 DDT problem would be 5428 seconds (see Appendix 1).

Multiple benchmark streams have yet to be run on a Convex, or any machine other than a VAX 8800, so we need to make some assumptions concerning the timings. This is not necessarily simple given that VAXes, and apparently the Convex, use overlapped IO. A reasonable estimate for machines on which AIPS is CPU bound could be

obtained by assuming that each of two jobs would get the CPU for half the time, thus doubling the real time for each. A further level of refinement is to allow that AIPS might not be completely CPU bound, and reckon on multiple processes being able to exercise the CPU for close to 100% of the time. We thus suggest

Real time for N jobs = Real time for 1 job
+ (N - 1) * CPU time for 1 job.

For the VAX 8800 itself, the real time for N jobs is that pertaining to the particular one of the N jobs which finished last.

However plausible, the above formula is certainly based on conjecture. It might give reasonable values for small N, but is unlikely to do so as N becomes large. For example, in a machine which makes extensive use of IO caching, multiple processes might start to compete for cache memory. Further, non-linear degradations have been noted in VAXes as they are loaded to the point where they start to page fault chronically, and also as they begin to outswap processes. Thus we urge that future AIPS benchmarking address the question of multiple processes.

Graphs 1 to 3, which are based on the above formulae, illustrate the relative performance of the VAX 8800 and Convex C-1 as a function of work load. The processing power is measured against the real time taken by a VAX 11/780 running a single small problem. Specifically, for our results

Equivalent processing power = (N * 5428) / Real time for N jobs.

Note that this formula yields a value slightly greater than unity for the equivalent processing power of a VAX 11/780 running multiple jobs. As expected, the Convex C-1 is rather faster than the VAX 8800, and this shows up most strongly for a single process utilizing only one of the VAX 8800's CPUs. The full CPU performance of the VAX 8800 was not realized in running AIPS since the machine is somewhat IO limited. Indeed, the ratio of CPU to real times was only 66% for a single small DDT, improving to 85% for the medium problem. Note that the difference between real and CPU times has a different interpretation for multiple processes, since it includes a timesharing component together with the excess IO time.

It is interesting to compare the actual performance of the VAX 8800 with its limiting performance if the IO problem could be solved. Graphs 1 to 3 depict this by identifying limiting real time with the measured CPU times. Clearly, when running AIPS the VAX 8800 approaches its performance limit only as it is loaded with many jobs. A user with exclusive use of a VAX 8800 would definitely need to utilize the multi-tasking abilities of AIPS to load the machine efficiently, particularly in having a second process to occupy the other CPU.

Support for symmetric multiprocessors in a future release of VMS will help the VAX 8800 somewhat, in that the ATTACHED processor will not be dependent on the PRIMARY processor for IO. We estimate that this would increase its throughput processing power by about 3/4 of a VAX 11/780 equivalent for two DDT streams running the small problem.

AIPS might also be made to handle its IO more efficiently on machines with large memory. At the moment the double buffered IO works in units of only about 16 blocks on a VAX, much less than desirable, because of the need to support limited memory machines such as the Modcomp. Rather than change the buffer size in all occurrences of calls to the higher level AIPS IO routines such as MINI3 and MDIS3, it occurred to us that it would be possible to implement a form of IO caching in the low level AIPS routine ZQIO. By such means we might possibly squeeze another VAX 11/780 equivalent out of each of the CPUs in the VAX 8800.

Any attempts to make the VAX 8800 run faster come up against a hard limit at approximately eleven VAX 11/780 equivalents, compared to about twenty eight for the Convex C-1. The VAX 8800 could support an array processor, but it is difficult to estimate what improvement in performance is possible, and at what cost. We consider that a VAX 8800 with a powerful array processor would certainly have much improved throughput processing power, probably exceeding that of the Convex. From the point of view of an individual user, however, results given in AIPS memo 44 for a VAX 11/780 and VAX 8600 with FPS 120B array processor hint that, being IO limited, the combination would probably still be slower than a Convex for a single process, although it may start to win with multiple processes.

Finally, it should be noted that processing power is only one aspect of the decision to procure a computer system. Processing power per dollar is probably a more relevant quantity, and naturally the total cost of the system, including peripheral devices and system software, is a major concern. In our case, the computer will be used for applications other than AIPS, for example database management for the AT, so we must consider the impact of changing operating systems. Being remote from the USA, hardware maintenance is also an issue in Australia.

5 ACKNOWLEDGEMENTS

We wish to thank Ian Smith of DEC Sydney and Mike Greenfield of DEC Marlboro for their support during the benchmarking. We also thank the Digital Equipment Corporation for their hospitality during our stay in Marlboro.

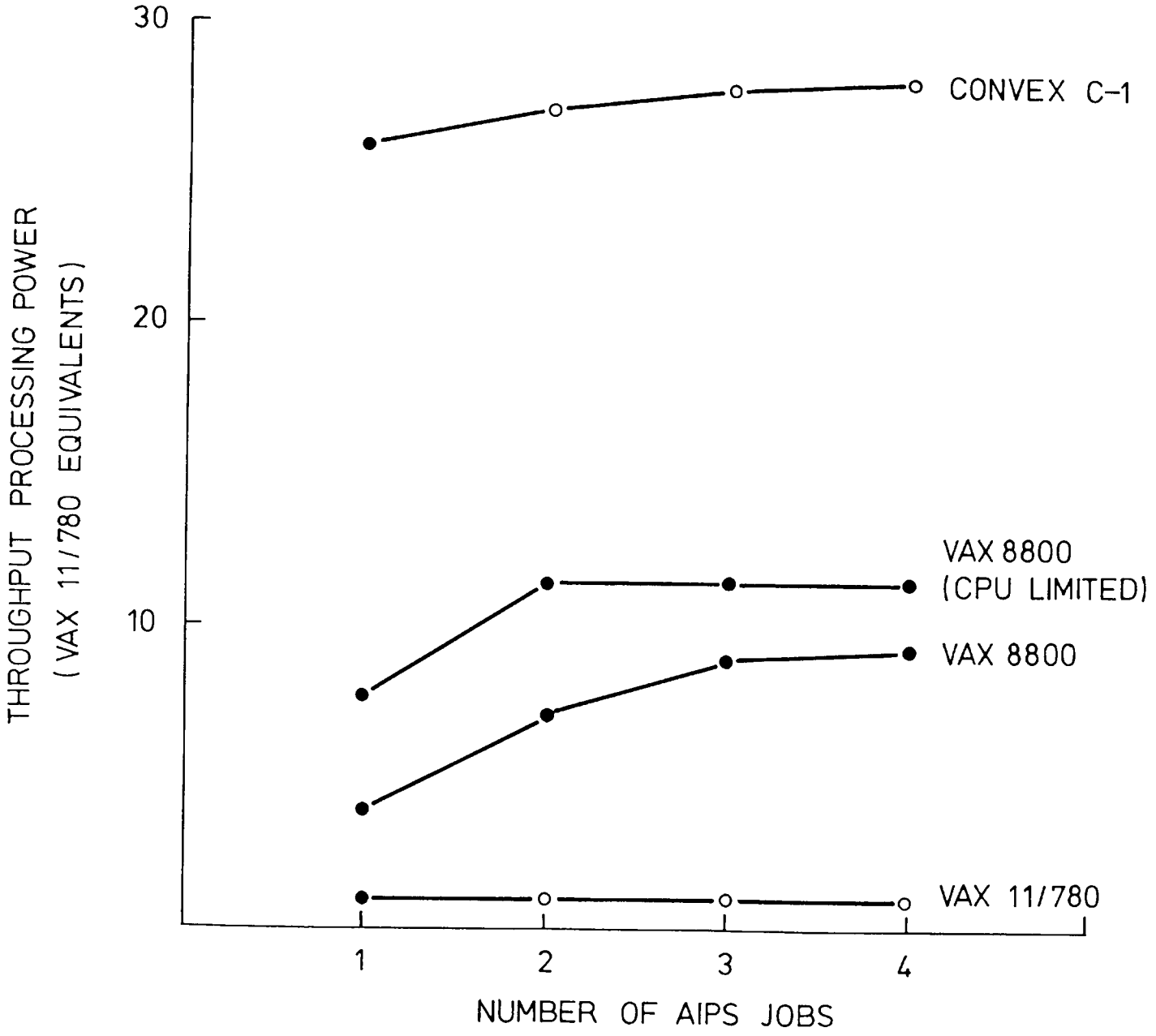

Figure 1. This diagram illustrates the total work capacity, the throughput processing power, of the various machines of interest, without regard to the time taken by an individual process. In this and the other figures filled circles represent actual measurements, whereas open circles are extrapolations according to formulae given in the text. A general slight increase in throughput processing power is expected as the number of jobs increases because IO for one process may overlap with CPU intensive usage by another. The VAX 8800 exhibits quite a significant increase for other reasons, as explained in the text.

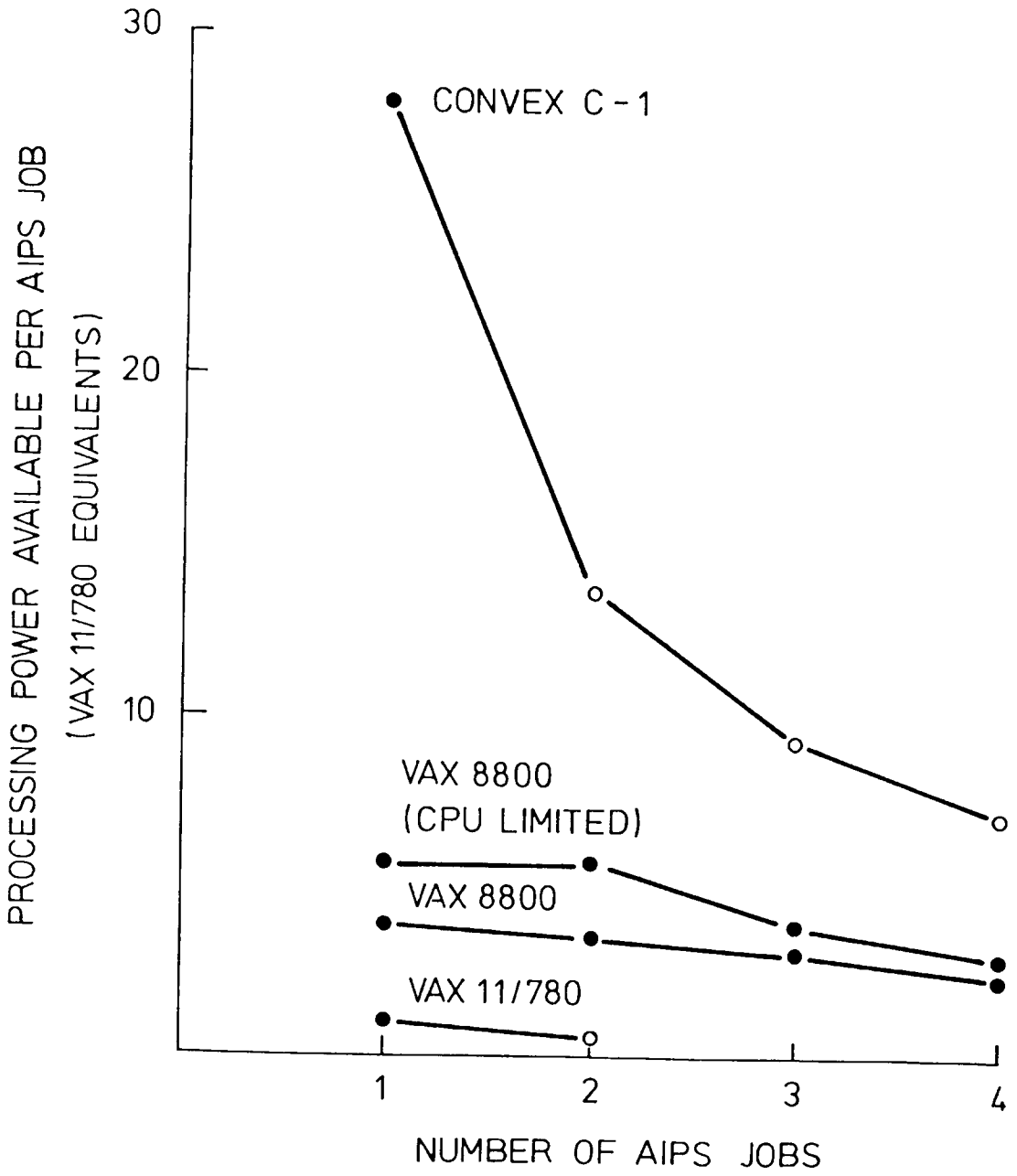

Figure 2. This figure is derived from, and is complimentary to, Figure 1. It shows how the processing power available for a particular process degrades as the machine is loaded with more work. Machines other than the VAX 8800 would exhibit an approximately inverse dependence on the number of processes.

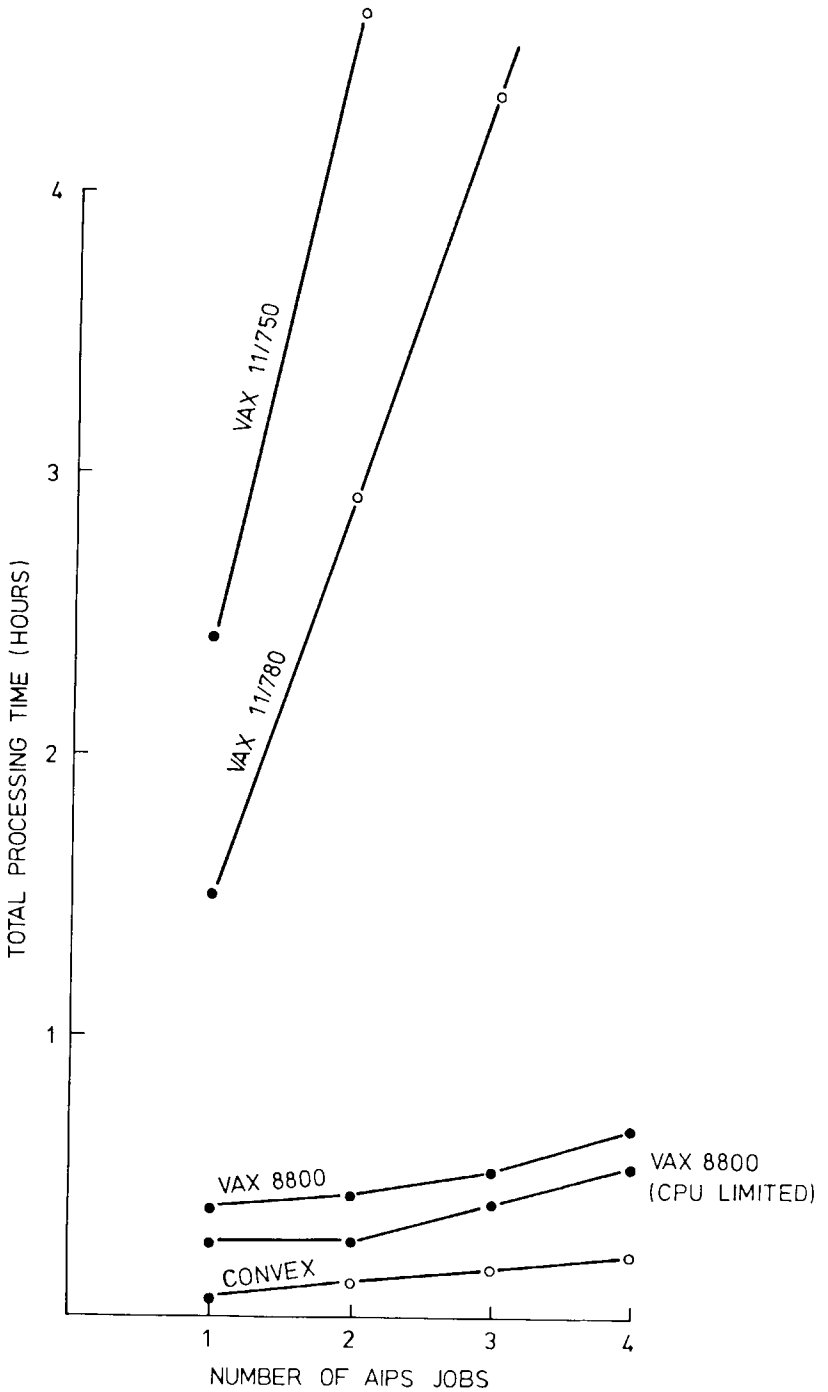

Figure 3. Probably the question of most concern to the astronomers who will actually be using the machines is "How long will it take to process my data?". This graph dramatically illustrates the users "agony" in doing significant image processing on smaller machines, particularly in competition with other users.

6 APPENDIX 1.

Notes.

1) Units for IO are VAX specific, real and CPU times are given in seconds.
2) Figures given for CNVRT, COMB, UVSRT, and UVDIF are for the first of a number of runs (4, 8, 2, and 2 respectively). However, the ratio of CPU to REAL time is an average over all runs for these tasks.
3) Totals do not include the results for AIPS.
4) APRES problem, see text. Assumed values for the VAX 8800 quadruple small DDT were IO = 1300, Real = 60, and CPU = 14.0
5) The 15JUL86 DDT is not directly comparable with 15APR86, especially in so far as the latter uses task CNVRT (although apparently it doesn't need to do anything), whereas the former now includes task UVDIF. Comparing the remaining tasks for the small problem on the VAX 8800, totals for each are

Version	IO	Real	CPU
15APR86	20570	1394	932.2
15JUL86	21903	1364	967.7

Differences between the two are probably not significant.

DDT benchmark results using 15APR86 AIPS.

Radiophysics VAX 11/751
Unloaded machine,
1200 baud terminal,
Unoptimized, single job,
SMALL problem, mode 'M'.
7/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	17564	27129	703.0	na
CNVRT	138	11	7.5	0.68
COMB	646	49	31.9	0.65
SUBIM	514	26	10.5	0.41
UVSRT	990	60	36.4	0.60
APCLN	1663	1799	1725.7	0.96
APRES	1071	223	190.8	0.86
ASCAL	1362	2876	2789.3	0.97
MXMAP	1493	357	313.5	0.88
MXCLN	4289	3067	2907.8	0.95
UVMAP	1854	304	261.9	0.86
VM	8233	1652	1475.1	0.89
total	22253	10424	9750.4	0.94

Radiophysics VAX 11/751
LOADED machine,
9600 baud terminal
Unoptimized, single job,
SMALL problem, mode 'M'.
24/Jun/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	28807	37380	1332.8	na
CNVRT	137	19	7.8	0.46
COMB	645	94	32.6	0.46
SUBIM	554	32	10.9	0.34
UVSRT	1367	77	38.8	0.37
APCLN	1752	3301	1724.0	0.52
APRES	1134	523	196.6	0.38
ASCAL	1428	12240	2881.4	0.24
MXMAP	1528	1279	324.6	0.25
MXCLN	4551	11013	3069.4	0.28
UVMAP	1957	497	270.2	0.54
VM	8475	2105	1539.8	0.73
total	24721	31616	10167.9	0.32

AAO Epping VAX 11/780
Unloaded machine,
1200 baud terminal
Unoptimized, single job,
SMALL problem, mode 'M'.
3/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	15390	26424	393.4	na
CNVRT	109	8	5.2	0.70
COMB	526	30	19.9	0.65
SUBIM	425	19	6.2	0.33
UVSRT	788	41	21.3	0.49
APCLN	1378	1140	1109.7	0.97
APRES	858	130	111.4	0.86
ASCAL	1236	1872	1834.6	0.98
MXMAP	1195	221	197.7	0.89
MXCLN	4045	1929	1862.9	0.97
UVMAP	1523	188	164.1	0.87
VM	7461	948	858.8	0.91
total	19544	6526	6191.8	0.95

AAO Epping VAX 11/780
Unloaded machine,
1200 baud terminal
Unoptimized, single job,
MEDIUM problem, mode 'M'.
4-5/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	18252	29711	511.1	na
CNVRT	236	20	17.3	0.86
COMB	946	68	57.3	0.84
SUBIM	760	33	10.5	0.32
UVSRT	1233	66	35.6	0.50
APCLN	7123	9917	9790.3	0.99
APRES	2339	506	481.7	0.95
ASCAL	1799	8217	7462.6	0.91
MXMAP	2932	711	676.0	0.95
MXCLN	10795	12089	11931.0	0.99
UVMAP	3440	552	519.2	0.94
VM	28679	4080	3702.4	0.91
total	60282	36259	34683.9	0.96

The following two runs test the reproducibility of results using a VAX 8500.

	DEC Marlboro VAX 8500 Slightly loaded machine, 9600 baud terminal 15APR86 DDT Unoptimized, single job, SMALL problem, mode 'M'. 11/Jul/86 (first)				DEC Marlboro VAX 8500 Unloaded machine, 9600 baud terminal 15APR86 DDT Unoptimized, single job, SMALL problem, mode 'M'. 11/Jul/86 (second)			
TASK	IOCNT	REAL	CPU	RATIO	IOCNT	REAL	CPU	RATIO
AIPS	16234	3976	128.5	na	15793	221141	125.0	na
CNVRT	125	4	1.6	0.42	124	4	1.6	0.41
COMB	595	34	7.1	0.27	614	21	7.0	0.32
SUBIM	475	17	2.1	0.12	477	13	2.0	0.16
UVSRT	894	65	7.3	0.15	894	39	7.6	0.20
APCLN	1460	489	356.5	0.73	1494	406	356.3	0.88
APRES	925	59	40.4	0.68	947	60	40.0	0.67
ASCAL	1303	711	672.9	0.95	1329	752	713.2	0.95
MXMAP	1341	98	65.5	0.67	1341	98	65.0	0.66
MXCLN	4164	693	622.4	0.90	4172	686	612.7	0.89
UVMAP	1628	132	58.5	0.44	1674	92	55.0	0.60
VM	7676	431	315.1	0.73	7684	435	315.0	0.72
total	20576	2733	2149.4	0.79	20750	2606	2175.4	0.83

The following sequence tests the effect of changes in the environment under which the benchmark is run.

	DEC Marlboro VAX 8800 Unloaded machine, 9600 baud terminal 15APR86 DDT Unoptimized, single job, SMALL problem, mode 'M'. 13/Jul/86				DEC Marlboro VAX 8800 Unloaded machine, 9600 baud terminal 15APR86 DDT Optimized, single job, SMALL problem, mode 'M'. 13/Jul/86				DEC Marlboro VAX 8800 Unloaded machine, VMS batch 15APR86 DDT Optimized, single job, SMALL problem, mode 'M'. 13/Jul/86				DEC Marlboro VAX 8800 Unloaded machine, VMS batch, no FHWM 15APR86 DDT Optimized, single job, SMALL problem, mode 'M'. 14/Jul/86			
TASK	IOCNT	REAL	CPU	RATIO	IOCNT	REAL	CPU	RATIO	IOCNT	REAL	CPU	RATIO	IOCNT	REAL	CPU	RATIO
AIPS	16467	2474	79.4	na	16335	2237	76.4	na	16225	2157	66.8	na	15843	2242	65.6	na
CNVRT	126	4	1.0	0.28	124	3	1.0	0.26	131	4	1.0	0.27	125	4	0.9	0.27
COMB	604	19	4.1	0.20	607	19	3.6	0.18	623	20	3.7	0.18	734	20	4.1	0.19
SUBIM	477	14	1.3	0.10	467	13	1.2	0.09	480	15	1.3	0.08	541	19	1.4	0.07
UVSRT	880	36	4.3	0.12	888	35	3.4	0.10	880	35	3.4	0.10	1053	32	3.4	0.11
APCLN	1476	231	198.0	0.86	1487	205	171.5	0.84	1522	204	171.7	0.84	1828	205	171.9	0.84
APRES	938	41	20.4	0.50	957	36	13.3	0.37	974	35	13.4	0.38	1215	35	13.6	0.39
ASCAL	1315	392	354.0	0.90	1332	360	321.6	0.89	1367	340	301.6	0.89	1471	348	308.4	0.89
MXMAP	1335	70	34.8	0.50	1349	63	26.8	0.42	1378	63	26.3	0.42	1683	65	26.7	0.41
MXCLN	4173	405	326.2	0.81	4186	350	265.8	0.76	4350	352	266.7	0.76	4511	390	315.3	0.81
UVMAP	1654	69	29.1	0.42	1653	62	20.9	0.34	1667	62	21.2	0.34	1990	59	22.2	0.38
VM	7660	297	161.2	0.54	7644	251	104.1	0.41	7778	258	104.7	0.41	8598	250	106.2	0.42
total	20638	1578	1134.4	0.72	20694	1397	933.2	0.67	21150	1388	915.0	0.66	23749	1427	974.1	0.68

The following is the only run of the 15JUL86 pre-release supplied by NRAO.

DEC Marlboro VAX 8800
Unloaded machine,
9600 baud terminal
15JUL86 DDT (pre-release)
Optimized, single job,
SMALL problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	22191	3505	110.8	na
UVDIF	549	36	27.4	0.76
COMB	498	13	1.8	0.14
SUBIM	588	17	1.5	0.09
UVSRT	1065	32	4.4	0.14
APCLN	1429	226	195.8	0.87
APRES	848	31	13.5	0.43
ASCAL	1440	348	318.4	0.91
MXMAP	1643	57	24.0	0.42
MXCLN	4129	345	277.7	0.81
UVMAP	1592	47	18.2	0.39
VM	8671	248	112.4	0.45
total	22452	1400	995.1	0.71

The following four sets investigate the behaviour of the VAX 8800 as it is loaded with progressively more jobs.

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, single job,
SMALL problem, mode 'M'.
13/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	16225	2157	66.8	na
CNVRT	131	4	1.0	0.27
COMB	623	20	3.7	0.18
SUBIM	480	15	1.3	0.08
UVSRT	880	35	3.4	0.10
APCLN	1522	204	171.7	0.84
APRES	974	35	13.4	0.38
ASCAL	1367	340	301.6	0.89
MXMAP	1378	63	26.3	0.42
MXCLN	4350	352	266.7	0.76
UVMAP	1667	62	21.2	0.34
VM	7778	258	104.7	0.41
total	21150	1388	915.0	0.66

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, double job,
SMALL problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	15571	2521	66.3	na
CNVRT	127	4	0.9	0.31
COMB	756	26	4.0	0.16
SUBIM	547	22	1.3	0.06
UVSRT	1058	40	3.5	0.09
APCLN	1858	250	204.9	0.82
APRES	****	***	*****	****
ASCAL	1505	363	319.7	0.88
MXMAP	1700	77	26.9	0.35
MXCLN	4509	360	266.6	0.74
UVMAP	2028	79	22.5	0.28
VM	8612	273	114.6	0.42
total	23923	1532	978.7	0.64

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, double job,
SMALL problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	15507	2264	65.9	na
CNVRT	128	4	0.9	0.29
COMB	737	23	3.8	0.17
SUBIM	542	22	1.3	0.06
UVSRT	1056	40	3.5	0.09
APCLN	1839	218	177.2	0.81
APRES	1223	38	13.8	0.36
ASCAL	1484	338	305.4	0.90
MXMAP	1688	62	26.5	0.43
MXCLN	4523	357	268.4	0.75
UVMAP	1987	74	21.8	0.29
VM	8636	261	105.8	0.41
total	23843	1437	928.4	0.65

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, triple job,
SMALL problem, mode 'M'.
14-15/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	16577	3000	70.6	na
CNVRT	127	4	1.0	0.28
COMB	735	24	3.9	0.17
SUBIM	539	20	1.2	0.06
UVSRT	1047	58	3.8	0.07
APCLN	1831	293	177.2	0.60
APRES	****	***	*****	****
ASCAL	1470	451	325.8	0.72
MXMAP	1676	86	27.4	0.32
MXCLN	4481	421	272.3	0.65
UVMAP	1971	86	22.1	0.26
VM	8582	348	106.9	0.31
total	23675	1838	955.4	0.52

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, triple job,
SMALL problem, mode 'M'.
14-15/Jul/86

IOCNT	REAL	CPU	RATIO
15976	2783	68.3	na
125	4	0.9	0.23
747	24	4.0	0.16
539	16	1.4	0.09
1051	49	3.6	0.09
1830	289	176.6	0.61
1216	47	13.8	0.29
1472	443	313.2	0.71
1676	76	27.4	0.36
4494	398	268.0	0.67
1973	77	22.1	0.29
8619	320	106.3	0.33
23742	1743	937.3	0.54

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, triple job,
SMALL problem, mode 'M'.
14-15/Jul/86

IOCNT	REAL	CPU	RATIO
16712	3009	70.2	na
125	4	1.0	0.23
746	26	4.0	0.16
539	20	1.3	0.06
1057	52	3.5	0.07
1831	306	177.0	0.58
****	***	*****	****
1467	434	311.5	0.72
1689	83	27.2	0.33
4484	428	271.7	0.63
1977	83	22.6	0.27
8614	339	108.1	0.32
23745	1822	941.7	0.52

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
SMALL problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	16900	3569	72.1	na
CNVRT	127	5	1.0	0.22
COMB	761	30	4.3	0.13
SUBIM	548	25	1.3	0.05
UVSRT	1070	69	3.4	0.06
APCLN	1867	395	176.8	0.45
APRES	****	***	*****	****
ASCAL	1498	587	308.4	0.53
MXMAP	1689	104	29.0	0.28
MXCLN	4506	505	271.1	0.54
UVMAP	2009	102	22.8	0.22
VM	8617	400	107.1	0.27
Total	23992	2282	939.2	0.41

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
SMALL problem, mode 'M'.
14/Jul/86

IOCNT	REAL	CPU	RATIO
16753	3716	70.5	na
126	4	0.9	0.20
749	34	3.9	0.12
543	23	1.4	0.06
1063	81	3.7	0.06
1874	355	174.5	0.49
****	***	*****	****
1490	580	308.3	0.53
1684	90	26.7	0.30
4499	531	270.5	0.51
2001	107	21.9	0.20
8640	413	110.1	0.27
23969	2278	935.9	0.41

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
SMALL problem, mode 'M'.
14/Jul/86

IOCNT	REAL	CPU	RATIO
15870	3702	68.1	na
125	3	1.0	0.24
734	33	4.0	0.14
544	24	1.3	0.05
1055	68	3.6	0.05
1852	470	260.3	0.55
****	***	*****	****
1492	575	323.2	0.56
1673	87	27.2	0.31
4504	527	271.7	0.52
1974	106	22.4	0.21
8602	419	106.2	0.25
23855	2372	1034.9	0.44

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
SMALL problem, mode 'M'.
14/Jul/86

IOCNT	REAL	CPU	RATIO
15944	3315	69.8	na
127	4	0.9	0.26
741	29	4.5	0.14
551	16	1.2	0.08
1073	64	3.7	0.07
1855	349	175.3	0.50
****	***	*****	****
1504	528	313.4	0.59
1689	87	26.9	0.31
4510	426	271.4	0.64
1999	99	21.9	0.22
8638	369	106.1	0.29
23987	2031	939.3	0.46

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, single job,
MEDIUM problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	18792	7311	94.5	na
CNVRT	254	7	3.0	0.46
COMB	1166	32	10.6	0.39
SUBIM	879	34	2.1	0.06
UVSRT	1537	50	5.4	0.10
APCLN	7635	1628	1534.3	0.94
APRES	2709	98	57.6	0.59
ASCAL	2049	1381	1326.4	0.96
MXMAP	3425	145	83.0	0.57
MXCLN	11231	1992	1843.8	0.93
UVMAP	3915	122	64.8	0.53
VM	29848	854	435.7	0.51
total	64648	6343	5366.7	0.85

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
MEDIUM problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	21317	15415	96.2	na
CNVRT	255	9	3.1	0.34
COMB	1161	46	10.9	0.23
SUBIM	877	43	2.1	0.05
UVSRT	1527	103	5.4	0.06
APCLN	7634	3237	1572.2	0.49
APRES	****	***	*****	****
ASCAL	2058	2575	1344.3	0.52
MXMAP	3435	212	84.9	0.40
MXCLN	11229	3285	1958.6	0.60
UVMAP	3919	229	63.9	0.28
VM	29840	1306	432.4	0.33
total	64635	11180	5533.4	0.49

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
MEDIUM problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	20760	15686	92.1	na
CNVRT	253	13	3.2	0.25
COMB	1159	54	10.8	0.21
SUBIM	876	40	2.2	0.05
UVSRT	1499	101	5.5	0.06
APCLN	7629	3217	1563.9	0.49
APRES	****	***	*****	****
ASCAL	2047	2783	1512.3	0.54
MXMAP	3429	221	85.2	0.39
MXCLN	11245	3131	1881.3	0.60
UVMAP	3904	224	64.6	0.29
VM	29848	1313	435.2	0.33
total	64589	11232	5619.8	0.50

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
MEDIUM problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	21065	15340	94.5	na
CNVRT	252	10	2.9	0.28
COMB	1154	51	10.7	0.22
SUBIM	872	39	2.1	0.05
UVSRT	1491	104	5.4	0.07
APCLN	7614	3220	1565.7	0.49
APRES	****	***	*****	****
ASCAL	2033	2579	1355.1	0.53
MXMAP	3413	198	84.0	0.42
MXCLN	11219	3160	1881.3	0.60
UVMAP	3898	236	66.1	0.28
VM	29816	1332	430.6	0.32
total	64462	11064	5459.5	0.49

DEC Marlboro VAX 8800
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, quadruple job,
MEDIUM problem, mode 'M'.
14/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	20340	12754	90.4	na
CNVRT	256	8	3.0	0.38
COMB	1160	49	11.6	0.28
SUBIM	879	39	2.1	0.05
UVSRT	1499	102	5.4	0.07
APCLN	7628	3116	1563.4	0.50
APRES	2700	135	55.6	0.51
ASCAL	2045	2736	1343.7	0.49
MXMAP	3420	218	83.9	0.38
MXCLN	11224	3208	1877.7	0.59
UVMAP	3917	229	65.3	0.29
VM	29857	1441	439.0	0.30
total	64585	11281	5450.7	0.48

DEC Marlboro VAX 8200
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, single job,
SMALL problem, mode 'M'.
15/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	17351	7966	405.7	na
CNVRT	125	8	5.6	0.71
COMB	733	39	23.2	0.59
SUBIM	539	23	6.8	0.30
UVSRT	1124	51	20.8	0.42
APCLN	1829	1727	1634.9	0.95
APRES	1212	103	81.2	0.79
ASCAL	1469	1697	1633.1	0.96
MXMAP	1671	198	159.4	0.80
MXCLN	4481	1906	1777.3	0.93
UVMAP	1988	169	129.0	0.76
VM	8586	731	605.1	0.83
total	23757	6652	6076.4	0.91

DEC Marlboro VAX 8200
Unloaded machine,
VMS batch
15APR86 DDT
Optimized, single job,
MEDIUM problem, mode 'M'.
15/Jul/86

TASK	IOCNT	REAL	CPU	RATIO
AIPS	25977	38516	698.3	na
CNVRT	252	24	20.9	0.88
COMB	1152	77	62.6	0.81
SUBIM	870	36	10.1	0.28
UVSRT	1490	73	32.0	0.44
APCLN	7603	13130	12920.9	0.98
APRES	2692	368	335.0	0.91
ASCAL	2026	6710	6605.9	0.98
MXMAP	3404	552	504.7	0.91
MXCLN	11201	12631	12395.7	0.98
UVMAP	3887	449	388.5	0.87
VM	29801	2747	2473.0	0.90
total	64378	36797	35749.3	0.97

Comparison of machines in terms of the VAX 11/780
for the 15APR86 DDT benchmark package.

Machine	SMALL problem Equivalent VAX 11/780's. Real	CPU	MEDIUM problem Equivalent VAX 11/780's. Real	CPU
VAX 11/750	0.62 *	0.64 *	-	-
VAX 8200	0.82	0.84	0.82	0.80
VAX 11/780	1.00 *	1.00 *	1.00 *	1.00 *
VAX 8500	2.44 *	2.85 *	-	-
VAX 8800 (1)	3.91	2 x 5.57	4.75	2 x 5.32
VAX 8800 (2)	7.09	2 x 5.34	-	-
VAX 8800 (3)	8.86	2 x 5.39	-	-
VAX 8800 (4)	9.15	2 x 5.29	10.68	2 x 5.17

Summary of results obtained by NRAO using the 15APR85 PFT benchmark.

VAX 11/780	1.00 *	1.00 *
VAX 8600	4.00 *	3.92 *
VAX 11/780+FPS120B	9.71 *	na
VAX 8600 +FPS120B	13.35 *	na
Alliant FX/1	9.35	10.72
Alliant FX/1(revB)	11.21	11.57
Alliant FX/6	21.77	32.82
Convex C-1	25.85	27.86
Cray X-MP	> 7.59	220.84

* Times which have been modified according to the formula given in the text to make allowance for the speed up expected from FORTRAN optimization are flagged with an asterisk.

7 APPENDIX 2

Simple Arithmetic Benchmark

Copyright Paul Rayner CSIRO Radiophysics

CPU Time in microseconds

OPERATION	V730	V750	UV-2	V780	8300	8500	8650	8800
null loop	9.90	5.00	3.50	2.50	2.20	0.80	0.70	0.40
integer add	7.80	3.50	2.20	1.70	1.60	0.50	0.35	0.19
integer subtract	8.50	3.40	2.20	1.60	1.60	0.60	0.37	0.20
integer multiply	15.20	7.10	6.40	2.40	3.70	0.90	0.49	0.46
integer divide	8.50	10.60	9.90	10.60	4.30	2.90	1.14	2.40
floating add	10.40	3.60	4.40	2.20	2.80	1.10	0.42	0.37
floating subtract	12.70	3.60	4.50	2.30	2.70	1.00	0.42	0.37
floating multiply	14.10	4.40	4.90	2.40	2.90	0.80	0.48	0.53
floating divide	14.30	8.90	5.60	5.40	3.60	1.90	1.19	1.59
integer to float	7.80	3.20	3.00	1.60	2.20	0.60	0.43	0.33
float to integer	11.10	2.90	3.80	3.20	5.50	0.80	0.60	0.60
sine	713.30	152.30	148.10	94.60	131.20	42.90	20.40	24.32
logarithm	216.20	75.80	80.20	41.40	71.10	22.80	11.47	12.45
square root	172.30	57.60	41.10	35.00	33.90	18.10	10.64	9.76
double add	16.80	4.90	6.00	4.60	3.90	1.50	0.60	0.55
double subtract	16.90	4.90	5.90	4.30	4.20	1.20	0.54	0.59
double multiply	25.90	6.30	8.00	6.40	5.40	1.30	1.08	0.75
double divide	26.00	15.00	8.90	10.70	7.40	3.60	3.92	3.16
integer to double	9.90	4.80	3.30	2.80	3.50	0.70	0.67	0.44
double to integer	13.50	2.70	4.30	5.00	5.60	1.40	1.01	0.61
double sine	1081.00	348.10	313.10	240.40	289.30	91.10	47.15	55.05
double logarithm	396.80	133.50	123.20	79.70	92.80	32.00	21.74	23.30
double square root	168.30	73.20	52.10	49.40	52.90	21.30	15.62	13.75
function call	80.30	35.50	21.10	21.50	20.20	4.40	4.15	3.04

Normalised CPU Times

OPERATION	V730	V750	UV-2	V780	8300	8500	8650	8800
null loop	3.96	2.00	1.40	1.00	0.88	0.32	0.28	0.16
integer add	4.59	2.06	1.29	1.00	0.94	0.29	0.21	0.11
integer subtract	5.31	2.13	1.38	1.00	1.00	0.38	0.23	0.13
integer multiply	6.33	2.96	2.67	1.00	1.54	0.37	0.20	0.19
integer divide	0.80	1.00	0.93	1.00	0.41	0.27	0.11	0.23
floating add	4.73	1.64	2.00	1.00	1.27	0.50	0.19	0.17
floating subtract	5.52	1.57	1.96	1.00	1.17	0.43	0.18	0.16
floating multiply	5.88	1.83	2.04	1.00	1.21	0.33	0.20	0.22
floatng divide	2.65	1.65	1.04	1.00	0.67	0.35	0.22	0.29
integer to float	4.88	2.00	1.88	1.00	1.38	0.38	0.27	0.21
float to integer	3.47	0.91	1.19	1.00	1.72	0.25	0.19	0.19
sine	7.54	1.61	1.57	1.00	1.39	0.45	0.22	0.26
logarithm	5.22	1.83	1.94	1.00	1.72	0.55	0.28	0.30
square root	4.92	1.65	1.17	1.00	0.97	0.52	0.30	0.28
double add	3.65	1.07	1.30	1.00	0.85	0.33	0.13	0.12
double subtract	3.93	1.14	1.37	1.00	0.98	0.28	0.13	0.14
double multiply	4.05	0.98	1.25	1.00	0.84	0.20	0.17	0.12
double divide	2.43	1.40	0.83	1.00	0.69	0.34	0.37	0.30
integer to double	3.54	1.71	1.18	1.00	1.25	0.25	0.24	0.16
double to integer	2.70	0.54	0.86	1.00	1.12	0.28	0.20	0.12
double sine	4.50	1.45	1.30	1.00	1.20	0.38	0.20	0.23
double logarithm	4.98	1.68	1.55	1.00	1.16	0.40	0.27	0.29
double square root	3.41	1.48	1.05	1.00	1.07	0.43	0.32	0.28
function call	3.73	1.65	0.98	1.00	0.94	0.20	0.19	0.14

Geometric Mean CPU Times and Figure of Merit

3.98	1.50	1.36	1.00	1.05	0.34	0.21	0.19
0.25	0.67	0.74	1.00	0.95	2.92	4.72	5.30

NOTE: The 8300 and 8800 times are uni-processor only.