Characteristics of a Large Shared Memory Production Workload

Su-Hui Chiang and Mary K. Vernon
suhui@cs.wisc.edu *vernon@cs.wisc.edu*

Computer Sciences Department
University of Wisconsin-Madison

Abstract

This paper characterizes the production workload that highly utilizes the NCSA Origin 2000. The characterization includes the distributions of job interarrival time, requested number of processors, requested memory, requested runtime, actual runtime as a fraction of requested runtime, and the ratio of memory usage to memory request. Conditional distributions are defined as needed for generating a synthetic workload with the same characteristics, including the key correlations observed among the job parameters. Characteristics of the O2K workload that differ from previously reported production workload characteristics are also noted.

1 Introduction

To better understand the performance of job scheduling policies, and to create realistic synthetic workloads for evaluating the performance of alternative policies, it is useful to understand the characteristics of workloads that occur in actual production systems.

Several production parallel workloads have been analyzed previously [FeNi95, Hoto96, Feit96, SGS96, HSO96, Down97, Feit97, WLF$^+$96, JPF$^+$97, SYZ99, SSN99]. Some characteristics are similar across most of these previous workloads. For example, most jobs request power-of-two processors and a large fraction of the jobs are serial. Other characteristics have varied. For example, some workloads have a positive correlation between job runtime and the number of processors [FeNi95, WLF$^+$96], while the correlation is more complex in other workloads [Hoto96].

In this paper, we analyze six one-month production workloads on the Origin 2000 (O2K) at the National Computational Science Alliance (NCSA). This system is interesting for several reasons. First, the system (i.e., 1520 processors, 616 GB of memory) is larger than systems previously studied. Second, the jobs have longer running times (i.e., up to 400 hours or 16 days 16 hours) compared to a maximum runtime of only a few tens of hours in the systems previously studied. Third, the jobs collectively require a higher fraction (i.e., 90-100%) of the available processing time than in previous workloads. Finally, the O2K is a shared memory system; thus individual jobs may request a different fraction of the memory than the processors on the system. Most systems previously studied schedule processors only, with a default memory allocation equal to the memory associated with the processors allocated to the job.

The contributions of this paper are threefold. First, characteristics of the NCSA O2K workload that differ from previous workloads are identified. Second, we provide significant new measures and analysis. Third, we provide conditional distributions and a ”roadmap” for creating a synthetic workload that has the observed distributions as well as the key correlations among the distributions of requested processors, requested runtime, requested memory, actual runtime, and peak memory usage. To our knowledge, the roadmap and requisite conditional distributions have not been provided in previous workload studies.

Characteristics of the O2K workload that differ from previously reported workloads include the following.

- The coefficient of variation (CV) of the job interarrival time is in the range of 1-2 during each period of approximately stationary hourly arrival rate.
- Jobs require an *average* of 50-100 processor hours, depending on the month.
- 10-15% of the jobs run for over 20 hours.
- 15-20% of the jobs each month request a higher fraction of memory than processors.
- Jobs that request more processors tend to request and use less memory per processor.
- Greater than 50% of the jobs have actual runtime less than 20% of their requested runtime.
- There is not an appreciable correlation between the number of processors requested and the job runtime.

The new measures and analyses provided include:

- Memory demand, equal to the product of the amount of memory requested and the actual job runtime, is provided as a measure of system load.
- A comparison of the workload mix from month to month.
- We identify peak, intermediate, and low periods of approximately stable job arrival rate per hour, rather than using the (12-hour) ”day” and ”night” periods in [FeNi95, WLF$^+$96, Down97].

Table 1. NCSA O2K Space Shared Hosts

Host Name	Number of Processors	Memory	
		Total (GB)	Per Processor (MB)
eir	128	64	512
nerthus	128	64	512
hod1	128	64	512
jord1	128	32	256
saga1	128	32	256
huldra	128	32	256
mimir	128	32	256
modi2	64	16	256

Table 2. NCSA O2K Job Class Definitions

Class Name	Resource Request	
Time Component	Job Run Time	
vst (very short)	$\leq$ 5 hrs	
st (short)	$\leq$ 50 hrs	
mt (medium)	$\leq$ 200 hrs	
lt (long)	$\leq$ 400 hrs	
Size Component	# Processors	Memory
sj (small)	$\leq$ 8	$\leq$ 2 GB
mj (medium)	$\leq$ 16	$\leq$ 4 GB
lj (large)	$\leq$ 64	$\leq$ 16 GB*

(* $\leq$ 25 GB in October-December 2000)

- We examine whether the jobs submitted on different days or during different periods of the day are statistically similar or not.
- Distributions of requested total memory, requested runtime, and processor utilization, are provided.
- We provide a more extensive analysis of the correlations among the job characteristics. For example, the correlations between requested number of processors and requested memory, or between the ratio of actual to requested runtime and the requested runtime have, to our knowledge, not been studied in previous workload analyses.
- Characteristics of the fifteen largest jobs submitted each month are provided.

The remainder of the paper is organized as follows. Section 2 provides an overview of the NCSA O2K job classes and resources as well as a brief review of related work. Section 3 provides an overview of the system load. Sections 4 and 5 characterize the O2K workload in terms of requested resources and resource usage, respectively, pointing out the differences compared to previous workloads, and providing a roadmap for creating a synthetic workload. Section 6 provides the characteristics of the fifteen jobs that used the largest processing time each month. Section 7 concludes the paper and identifies topics for future research.

2 Background

2.1 NCSA O2K System and Job Traces

The NCSA O2K processors are partitioned into twelve hosts. Eight hosts, with a total of 960 processors, are used for space-shared scheduling of batch jobs that do not request exclusive use of a (dedicated) host. Table 1 shows the processor and memory resources of each of these eight space-shared hosts. Three other hosts, with a total of 512 processors, have a higher priority for running batch jobs that request a dedicated host, one at a time. Each of the three hosts will run other batch jobs with short requested run-time (i.e., less than five hours) in space-shared mode if there are no jobs waiting to run that have requested a dedicated host. The remaining one host runs interactive jobs only.

This paper provides the characteristics of all of the batch jobs that do not request a dedicated host. These jobs have more widely varying processor and memory requests than the jobs that request a dedicated host.

The jobs analyzed were submitted in the six one-month periods (i.e., January - March and October - December 2000). The LSF job scheduler [LSF], locally tuned for the NCSA workload, was used during the first three months. A priority-backfill scheduler similar to the Maui Scheduler [MS] replaced the LSF scheduling algorithm at the end of June 2000 [ChVe01], although LSF continues to log information about each job.

Each job requests a number of processors, an amount of memory, and a maximum running time. Based on these requests, the job is classified in one of four time classes (i.e., vst, st, mt, or lt) and one of three size classes (i.e., sj, mj, or lj), as defined in Table 2. For example, a job that requests 16 processors and 2 gigabytes of memory is an mj job. Jobs that require more than 64 processors can request a dedicated host. Table 4 shows that each month on the eight space shared hosts, the submitted mj and lj jobs typically require 65-70% of the total processing time available while upwards of 5500 sj jobs also need to be scheduled.

A job is killed if it exceeds its requested run time by one hour or has a number of processes greater than the number requested plus one.[1] Only a small fraction (un-

---

[1] A process is counted if it has used a cpu for at least 6 seconds (i.e., 20%) during a 30-second window. Under LSF scheduling, for the purpose of determining whether a job has exceeded its requested maximum runtime, the job's runtime was computed as its total cpu time, divided by the number of requested processors. For all other

der 2%) of the jobs each month exceed their requested processors. The jobs that are killed are included in the workload characterization.

The LSF log contains the arrival time, actual and requested job runtime, and requested processors and memory of each job. Jobs that have zero runtime (i.e., they failed or were aborted immediately) are excluded in our analysis. A software daemon (JMD) developed at NCSA records the actual resource usage of each job every 30 seconds during its execution. The memory and cpu usage of each job is obtained from the JMD log.

2.2 Related Work

Several workload studies [FeNi95, Feit96, Hoto96, HSO96, SGS96, WLF$^+$96, WMKS96, DoFe99] report the measured distributions of the number of requested processors and actual job runtime, on various production systems (e.g., NASA Ames Intel iPSC/860, Argonne SP/1, Cornell Theory Center (CTC) SP/2, SDSC Intel Paragon, PSC T3D). Several of these studies also report the distribution of job interarrival time [FeNi95, HSO96, WLF$^+$96], and the relationship between the average or distribution of runtime and requested number of processors [FeNi95, Hoto96, HSO96, DoFe99]. The studies in [Feit97, SSN99] focus on the memory usage of jobs on the LANL CM-5 and SDSC CRAY T3E. [Feit97] also reports the distribution of the fraction of requested memory used by a job. [HSO96] reports the distribution of requested memory per node on the CTC SP/2.

Based on job traces from production systems, several previous papers propose mathematical distribution functions that closely approximate the observed distributions. [Feit96] proposes distributions of the job interarrival time, the requested number of processors, and the actual runtime with mean conditioned on the requested number of processors, derived from an analysis of six workloads. Four papers analyze the CTC SP/2 workload and propose distributions of job interarrival time [JPF$^+$97], the requested number of processors [Down97], runtime [DoFe99], the product of requested number of processors and runtime [JPF$^+$97, Down97, DoFe99], and the runtime conditioned on number of requested processors [DoFe99]. [SYZ99] characterizes the arrival patterns observed on the CTC SP/2. The study in [FeWe98] proposes a distribution for the requested runtime as a multiple of actual runtime for a different SP/2 system.

The differences between these previous workloads and the workload analyzed in this paper are noted as results are presented in Section 3- 5.

Table 3. Notation

Symbol	Definition
M	Requested Memory
P	Requested Processors
T	Actual Runtime
M×T	Memory Demand
P×T	Processor Demand

3 Overview of the O2K Workload

This section provides an overview of the total load on the O2K as well as the load due to the large jobs submitted during each month studied. The fifteen largest jobs each month are further characterized in Section 6. Table 3 defines some notation that will be used in the remainder of the paper. Note that the processor demand for a job is the product of the number of requested processors and the actual runtime of the job. Similarly, the memory demand is the product of the amount of requested memory and the job runtime.

3.1 Monthly System Load

Table 4 summarizes the overall load on the NCSA O2K during each of the six one-month periods. The processor demand (proc demand) is summed over all jobs or all jobs in one of the twelve job classes, and expressed as a fraction of the total processing time available on the eight space-shared hosts during the month (i.e., the product of the total number of processors and the total time during the month). Note that processor demand can be greater than 100% because in a typical month 50% of the vst jobs (collectively having a total of 4-9% processor demand) run on the three hosts that give priority to dedicated jobs. The memory demand is similarly reported as a fraction of total available memory on the eight space-shared hosts. We use the resources on eight space shared hosts for measuring processor and memory demand because these are the only hosts that are guaranteed to be available for running the batch jobs that are characterized.

The key observations from Table 4 are:

- The total processor demand each month is extremely high, i.e., typically 90-100% of the total available processing time on the eight hosts.
- The overall memory demand is somewhat lower than the processor demand each month, but is still in the range of 70-80% during most months.
- The job mix is similar from month to month. That is, with a few exceptions, the number of jobs in each class and the processor and memory demands of each job class are fairly similar from month to month. One exception is the December 2000 workload, which has

---

purposes, and for the Maui scheduler, job runtime is defined simply as the amount of (wall clock) time the job occupies the processors.

Table 4. Total Monthly Processor and Memory Demand By Job Class

Month	Overall	Job Class											
		vst_sj	st_sj	mt_sj	lt_sj	vst_mj	st_mj	mt_mj	lt_mj	vst_lj	st_lj	mt_lj	lt_lj
Jan 2000													
#jobs	9652	3622	2606	553	71	950	589	163	61	671	252	91	23
proc demand	88%	2%	9%	11%	3%	2%	9%	13%	6%	4%	8%	12%	10%
mem demand	76%	1%	6%	7%	3%	1%	5%	10%	6%	1%	11%	10%	17%
Feb 2000													
#jobs	11290	5296	2269	466	71	1128	698	219	33	686	314	90	20
proc demand	96%	2%	9%	11%	3%	3%	10%	13%	3%	6%	18%	12%	5%
mem demand	78%	1%	7%	7%	3%	2%	5%	11%	5%	1%	10%	12%	15%
Mar 2000													
#jobs	12101	4671	2678	472	57	1808	631	216	70	850	500	123	25
proc demand	94%	2%	11%	9%	3%	4%	11%	15%	4%	4%	14%	14%	3%
mem demand	83%	1%	7%	6%	3%	2%	7%	9%	8%	2%	16%	18%	4%
Oct 2000													
#jobs	9768	3012	2488	580	278	881	627	241	50	957	367	209	78
proc demand	90%	1%	11%	9%	7%	2%	10%	11%	2%	5%	14%	13%	4%
mem demand	84%	1%	6%	7%	9%	1%	6%	6%	2%	2%	6%	18%	20%
Nov 2000													
#jobs	8708	2982	2279	416	60	711	497	187	16	912	513	110	25
proc demand	91%	2%	10%	8%	3%	2%	9%	12%	3%	6%	20%	13%	3%
mem demand	63%	1%	5%	5%	2%	1%	5%	6%	1%	2%	10%	11%	14%
Dec 2000													
#jobs	7798	2581	2190	565	164	801	252	215	59	667	176	113	15
proc demand	102%	2%	11%	10%	9%	2%	5%	18%	4%	6%	11%	13%	12%
mem demand	68%	1%	6%	8%	5%	1%	2%	10%	6%	2%	4%	13%	9%

Table 5. Summary of the Load of Various Large Job Classes

Month	Job Class	#Jobs	(%)	Demand/Demand All		Job Size			
				Processor Demand	Memory Demand	Avg Processors Requested	Avg Memory Requested (GB)	Avg Actual Runtime (hours)	Avg P×T
Oct 2000	P×T > avg P×T = 66h	1793	(18%)	89.3%	79.0%	15.2	2.3	42.5	319
	T > 20hr	1312	(13%)	71.3%	80.9%	6.7	1.9	59.6	348
	P > 16, T>10m	695	(7%)	35.7%	9.7%	42.2	4.7	8.5	329
	M > 4GB, T>10m	653	(7%)	15.8%	51.5%	22.0	10.1	15.0	154
	P > 16, M >4, T>10m	265	(3%)	10.7%	7.0%	47.0	9.8	6.1	258
Nov 2000	P×T > avg P×T = 72h	1495	(17%)	89.4%	83.3%	19.4	2.6	35.5	376
	T > 20hr	1065	(12%)	70.7%	79.6%	8.0	1.6	53.1	418
	P > 16, T>10m	767	(9%)	40.7%	21.7%	42.1	4.9	9.2	333
	M > 4GB, T>10m	502	(6%)	20.9%	53.7%	27.5	9.4	16.8	261
	P > 16, M > 4, T>10m	246	(3%)	14.4%	17.1%	46.8	12.0	8.6	367
Dec 2000	P×T > avg P×T = 93h	1188	(15%)	87.8%	76.0%	14.6	1.9	63.7	537
	T > 20hr	1293	(17%)	82.2%	87.7%	6.5	1.5	69.5	462
	P > 16, T>10m	455	(6%)	36.1%	21.2%	39.2	4.2	16.8	577
	M > 4GB, T>10m	310	(4%)	15.9%	36.1%	23.2	9.5	19.4	372
	P>16, M > 4, T>10m	97	(1%)	11.8%	15.9%	53.0	14.4	16.6	882

fewer jobs but higher processor demand (overall and in several job classes) than in the other months.

- The vast majority (i.e., 95%) of the total processor demand is due to (st,mt,or lt) jobs that request runtimes greater than five hours. On the other hand, a large number of vst jobs must also be scheduled.
- The large (lj) jobs have a relatively high processor and memory demand (typically equal to 35% or more). This is a challenging workload for the job scheduling policy, as it is difficult to find free resources for these large jobs when total system load is above 90%.

3.2 Monthly Load Due To Large Jobs

Previous work has shown that a large fraction of the processor demand is due to a small fraction of the jobs. For example, on an iPSC/860 90% of the jobs had runtime under 15 minutes, but the remaining 10% of the jobs account for 90% of the total processor demand by all jobs [FeNi95]. Hotovy [Hoto96] reported that, fewer than 50% of the jobs on the CTC SP/2 use more than 1 processor, but they account for over 90% of the total processor demand of all jobs.

The characteristics of the NCSA O2K workload are somewhat different. For example, 65-70% of the jobs request more than one processor (Figure 4) and 40% of the jobs have runtime greater than one hour (Figure 12). Table 4 shows that 50-55% of the O2K jobs are in the vst class, and the other 45-50% of the jobs account for approximately 90% of the processor demand each month.

To determine the extent to which various small sets of jobs dominate processing and/or memory demand on the O2K, five classes of large jobs are summarized in Table 5. The class ”P×T > avg P×T”, is the set of jobs that each have processor demand (P×T) greater than the average processor demand over all jobs in the given month. The average processor demand for the month is given in parentheses (e.g., 93 processor hours in December 2000). In addition, T > 20 hours, P>16, M>4 GB, and T>10 minutes are used to define four classes, three of which are subsets of the ’lj’ class. For each job class, the table shows the total number of jobs (and the fraction of all submitted jobs), the processor and memory demand expressed as a fraction of the respective demand of all submitted jobs, and the average of each of four size measures for jobs in the class.

Key observations from Table 5 include the following.

- Average processor demand per job is in the range of 50-100 processor hours, depending on the month.
- The load of each large job class is very similar across different months.
- Processor and memory usage are dominated by the 10-15% of jobs with running time greater than 20 hours or by the 15-18% of jobs with higher than average processor demand.
- No more than 3% of the jobs in the months we analyzed have P > 16, M > 4 GB, and T > 10 minutes.

4 Resource Request Characteristics

This section analyzes job arrival characteristics, and the observed distributions of requested number of processors, memory, and runtime. The analysis includes identifying periods per day of approximately stationary arrival rate, and determining whether resource request distributions vary among different months, days of the week, or periods of the day. Conditional distributions that capture the key correlations among the workload parameters are also provided. Section 5 will provide actual job runtime and memory usage distributions, as well as correlations between the usage distributions and the requested resources.

This section and Section 5 provide a ”roadmap” for creating a synthetic workload with characteristics similar to the observed O2K workload. In other words, the workload characteristics are presented in the order in which synthetic job characteristics could be assigned so as to capture the key correlations among the distributions. Sections 4.6 and 5.3 summarize the roadmap.

4.1 Job Arrival Rate

Figure 1 shows the number of job arrivals each day during October 2000, with the number submitted on each particular day of the week (e.g., Monday) grouped together. The figure shows that the number of arrivals is typically 350-400 per weekday and 150-200 per weekend day. Occasionally there are weekdays that have greater than 500 arrivals. Other months have similar arrivals per day, except that (1) in February 2000, the number of arrivals per weekend day was closer to 350, (2) during the last 2-3 weeks of December the number of arrivals per weekday is 150-200, and (3) on holidays (e.g., Thanksgiving, Christmas, New Years day), the number of arrivals is lower than a typical weekend day.

Several previous papers have provided the average number of jobs that arrive during each hour on weekdays

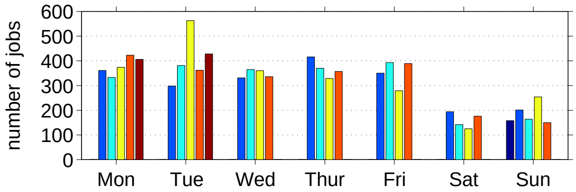
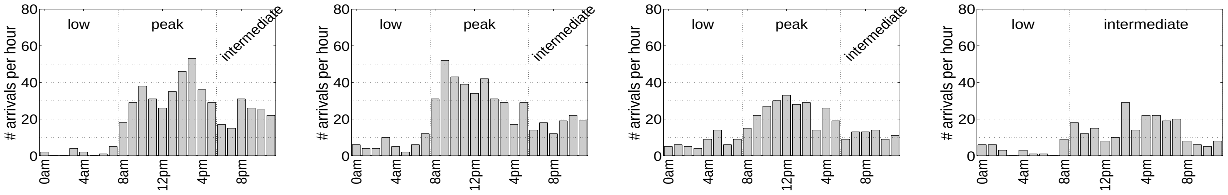
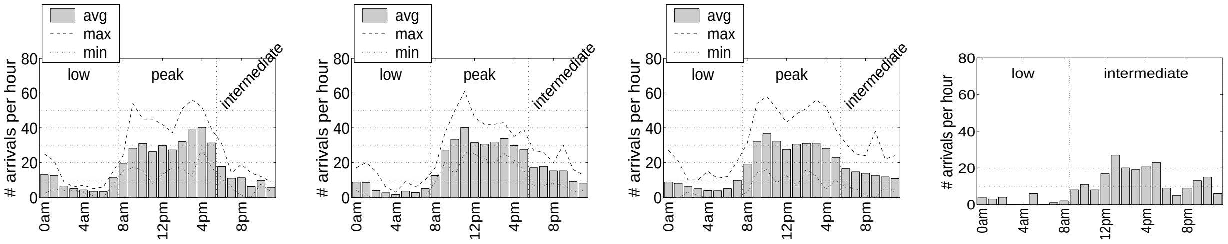


Figure 1. Number of Jobs Submitted Per Day
(October 2000)



(a) Wed, 3/00 Week 1 (b) Thur, 2/00 Week 2 (c) Thur, 10/00 Week 3 (d) Sat, 3/00 week 2



(e) All Fridays (2/00) (f) M-F, 3/00 Week 3 (g) All Weekdays (2/00) (h) Sun, 11/00 week2

Figure 2. Number of Job Arrivals Each Hour

and weekends on various parallel systems [FeNi95, WLF$^+$96, HSO96] including a network of workstations [Gib97a]. The pattern of job arrivals is very similar on each of these systems other than the Paragon in [WLF$^+$96]. That is, during weekdays there is a peak arrival period between 8am-6pm, a very low arrival period before 8am, and an intermediate arrival period after 6pm, with less variation in the number of job arrivals per hour during weekends. The previous papers have noted the higher arrival rate from 8am-6pm, but not the intermediate arrival rate after 6pm. They also do not comment on whether the job arrival pattern is similar or different across different days of the week, and do not examine whether other job characteristics (such as runtime) differ depending on period of the day or day of the week (e.g., Monday vs Friday).

Plots of the number of jobs that arrive each hour for each day in the six months of O2K workload did not reveal any distinct characteristics for any particular day during the work week or during the weekend. Figure 2 shows that the arrival pattern varies somewhat from one weekday to another (Figure 2 (a-c)) or one weekend day to another (Figure 2(d),(h)), but the average number of arrivals each hour is approximately the same whether computed over all Fridays (or other day of the week) in a given month (e.g., Figure 2(e)), or over Monday through Friday in a given week (e.g., Figure 2(f)), or over all weekdays in any given month (e.g., Figure 2(g)).

The figure shows that when the average number of arrivals per hour is computed over a large sample, such as all weekdays in a month, three periods of approximately stationary average arrival rate are identified. The three periods are labeled peak, intermediate, and low in the figures. Allowing for statistical fluctuations due to finite sample sizes, as well as for fluctuations due to occasional system downtimes, these three periods appear to be present during each weekday in the six months studied. Analogously, two periods (intermediate and low) of approximately stationary arrival rate were identified on weekend days, as shown in the figure.

The arrive rate per hour on weekdays is typically around 30 during peak, 15-20 during intermediate, and 5-10 during low periods.

4.2 Job Interarrival Times

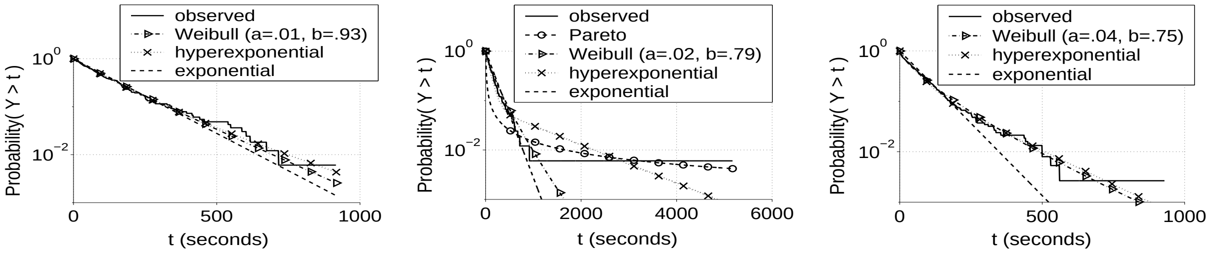
For the purpose of determining interarrival time distributions, we consider weekday peak periods from 9am-5pm because the 8-9am and 5-6pm periods have average arrival rate that is slightly lower than the other peak hours. For analyzing all other job characteristics, the weekday peak period is defined to be 8am-6pm.

Excluding periods when the O2K is down, during each period of approximately stationary hourly arrival rate, the coefficient of variation (CV) of the interarrival time is in the range of 1-2. Higher CVs (i.e., CV = 2.5-6) have been reported in other systems [FeNi95, WLF$^+$96, JPF$^+$97, SYZ99] for the distribution of interarrival times during 12-hour day and night periods (rather than during periods of stationary arrival rate).

We have investigated several models for the observed O2K interarrival time distribution in each period, including the exponential with the same mean interarrival time, a two-stage hyperexponential with the same mean and CV, the Weibull, and the gamma distribution.[2] The fit of

---

[2] In each period, the parameters of the two-stage hyperexponential distribution are computed using the standard algorithm such that the

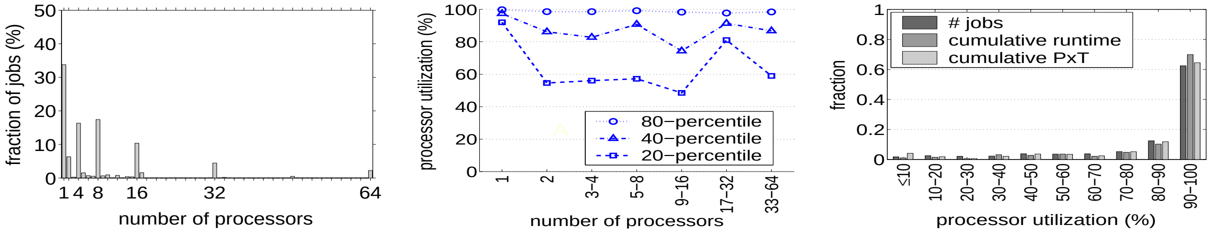


(a) Representative Distribution
1/11/00 9am-5pm Excluding Downtime
(mean = 139 seconds, cv = 1.12)

(b) Distribution Including Downtime
1/11/00 9am-5pm
(mean = 170 seconds, cv = 2.48)

(c) High Load Distribution
1/27/00 9am-5pm (no downtime)
(mean = 76 seconds, cv = 1.33)

Figure 3. Distributions of Job Interarrival Time (Y)



(a) Representative Distribution of
Requested Processors
(mean = 8.3, cv = 1.42)

(b) Processor Utilization
vs. Requested Processors
(actual runtime > 10 min)

(c) Distribution of
Processor Utilization
(actual runtime > 10 min)

Figure 4. Processor Requests and Utilization

these distributions is illustrated for two different periods in Figure 3(a) and (c). The fit of the gamma distribution (not shown) is similar to the fit of the Weibull distribution. The complement of the cumulative distribution is shown on a log scale, to more clearly show the fit of the tail of the distribution.

As shown in the figure, the two-stage hyperexponential distribution fits the observed interarrival time distribution better than the exponential distribution, even during periods when the CV is quite close to 1. In fact, in all periods (including all peak, intermediate, and low periods), the two-stage hyperexponential with the measured mean and CV had the best fit for the observed distribution.[3] Figure 3(b) shows that if periods when the system is down[4] (e.g., the period during 10:12-11:38am on January 11), are not eliminated from the measured interarrival time distribution, one might erroneously conclude that the interarrival times are better approximated by a heavy-tailed distribution such as the Pareto distribution.

4.3 Number of Requested Processors

Figure 4(a) plots a typical distribution of the number of requested processors for jobs submitted during weekday peak periods in a given month. The distribution is similar for other months and for most intermediate and low periods on weekdays and weekends.[5] Occasionally during one of the intermediate or low arrival rate periods, the number of jobs that request 32 or 64 processors is slightly higher than typical.

As in previous workloads [FeNi95, Hoto96, Feit96, WLF+96, WMKS96, SGS96, Down97], a large fraction (i.e., 30-40%) of the O2K jobs are serial, and most jobs request power-of-two processors. The distribution is similar to the log-uniform distribution proposed in [Down97], except that fewer jobs request 2 or 4 processors, and a small but not insignificant fraction of jobs request numbers of processors that are not a power of

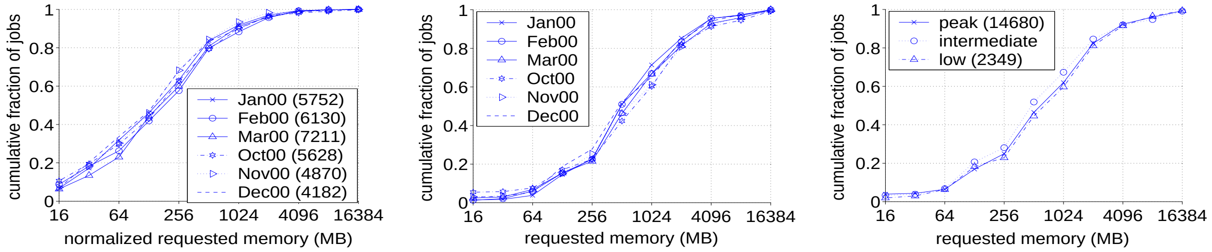
---

products of the probability and the mean service time for each stage are equal [All90]. The maximum likelihood estimates of the Weibull and gamma distribution parameters, with 95% confidence intervals, are computed using Matlab [JKK92, Mat].

[3] The two-stage hyperexponential with CV close to 3 was found to have a good fit for the distribution of all interarrival times, not separately measured over periods of approximately stationary arrival rate, for the CTC SP/2 in [JPF+97]

[4] Downtime is often not recorded in the job logs. We implemented a special daemon process that tracks system downtime on the O2K since the O2K job logs do not contain the down time information.

[5] On a system with restricted processor availability during 6am-8pm on weekdays, the fraction of submitted jobs that request the maximum number of processors was reported to be significantly higher during other hours [FeNi95]. The O2K does not restrict batch job sizes during weekday peak periods.

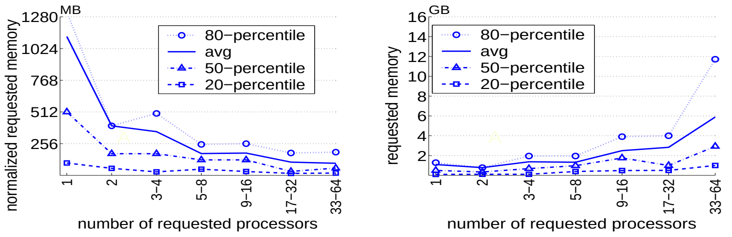


(a) Normalized Requested Memory Weekday Peak Periods Each Month

(b) Requested Memory Weekday Peak Periods Each Month

(c) Requested Memory Each Weekday Arrival Period

Figure 5. Variations in Distribution of Requested Memory (Normalized Requested Memory = Requested Memory/Number of Requested Processors)



(a) Normalized Requested Memory

(b) Requested Memory

Figure 6. Requested Memory vs. Requested Processors

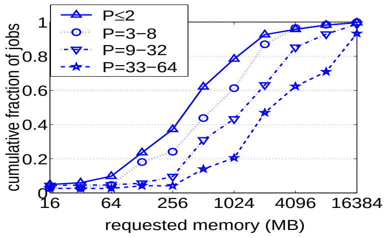


Figure 7. Conditional Distributions of Requested Memory

two. Thus, a harmonic distribution constructed to emphasize small numbers and powers of two, as proposed in [Feit96], is a better characterization of the processor requests on the O2K.

Each O2K job is allocated its number of requested processors until it terminates. Figures 4(b) and (c) show that a high fraction of jobs utilize over 80% requested processors, including jobs that request 32-64 processors, jobs that have dominant actual runtimes, and jobs that have dominant processor demand (P×T). This leads to high total utilization of allocated processors on the O2K, as will be shown in Figure 16(c).

4.4 Requested Memory

The normalized requested memory for a job is defined as the amount of requested memory divided by the number of processors requested. Figures 5(a) - (c) show that the distribution of normalized or total requested memory is very similar during different months and during different periods of the weekday.[6] It is also very similar for different days of the week and during weekend periods (not shown). Note that only a very small fraction (i.e., <1%) of jobs each month request the maximum of 25 GB (or previously 16 GB) of memory.

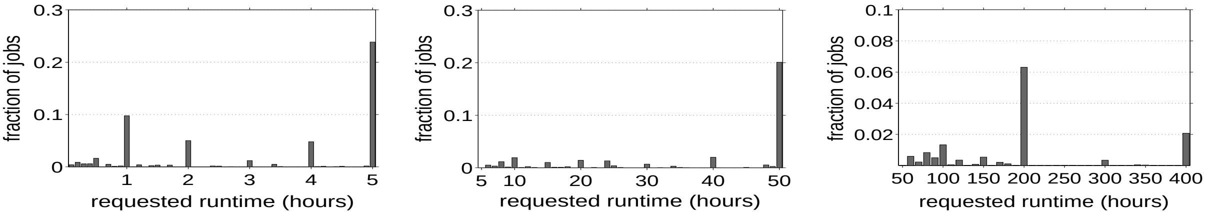
Figure 6 shows that the requested memory (as measured by the mean, or the 20th, 50th or 80th percentile) has a significant correlation with the requested number of processors. Specifically, total requested memory is positively correlated with the requested number of processors, while normalized requested memory is negatively correlated the requested number of processors. To our knowledge, the correlations between these parameters of previous workloads have not been investigated. The curves shown in the figure were observed for both the January - March 2000 and October - December 2000 workloads.

Based on the curves in Figure 6(b), Figure 7 provides the measured conditional distributions of requested memory for four different ranges of the requested number of processors, which can be used to create memory requests that have the observed correlation with job parallelism.

Recall from Table 1 that the average memory available per processor on the O2K is either 256 MB or 512 MB, depending on the host. As shown in Figures 5(a) and 6(a), 35-40% of all jobs have normalized requested memory greater than 256 MB; furthermore 15-20% of all jobs and 50% of the sequential jobs have normalized requested memory greater than 0.5 GB. In contrast, in an SP/2 system where more than 80% of the nodes had 128 MB of memory, 85% of the jobs requested this smallest possible normalized memory [Hoto96].

---

[6] Jobs submitted during weekday intermediate periods request 256 MB to 1 GB of memory slightly more often, and 2-4 GB slightly less often, but the weekday peak distribution is still a fairly good approximation for the intermediate period.

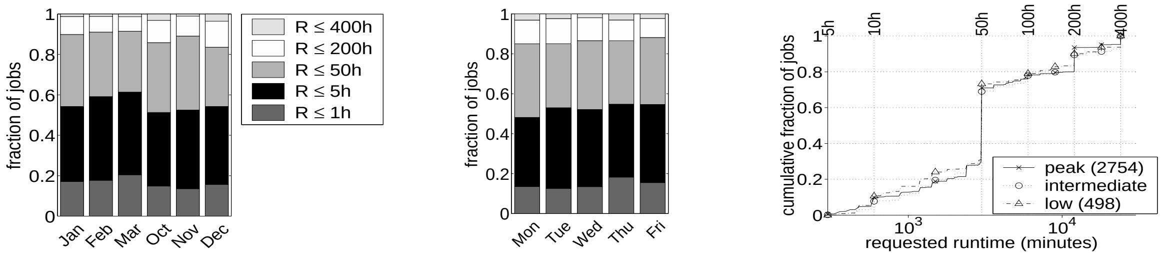


(a) 0-5 hours (vst)
bucket = 6 minutes

(b) 5-50 hours (st)
bucket = 1 hour

(c) 50-400 hours (mt, lt)
bucket = 10 hours

Figure 8. Distribution of Requested Runtime During Weekday Peak Periods



(a) Peak Periods Each Month

(b) Peak Periods Each Day

(c) Each Weekday Arrival Rate Period
(st, mt, lt Jobs: 5-400 hours)

Figure 9. Variation in Distribution of Requested Runtime (R)

4.5 Requested Runtime

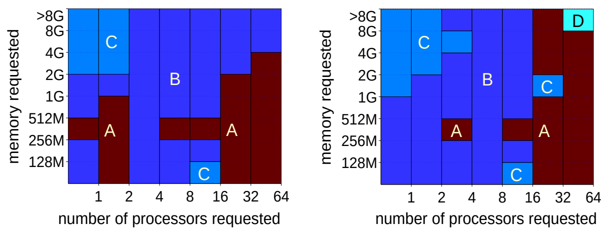
To our knowledge, previous papers have not reported the distribution of requested job runtime, although this parameter is used by many job scheduling policies.

Figure 8 shows the distribution of requested runtime for jobs submitted during weekday peak periods, and Figure 9 compares the distributions of requested runtime for jobs submitted during different periods of approximately stationary arrival rate. General observations about these distributions include the following.

- A large fraction of jobs request the default runtime for the job class (i.e., 5, 50, 200, or 400 hours). These requested runtimes have no special meaning for the current scheduler on the O2K, except that jobs that request up to 5 hours of runtime are eligible to run on the three hosts that have higher priority for jobs that request a dedicated host.
- Nearly all other jobs request some 'round' number such as 1, 2, 10, 20, or 300 hours, with approximately *uniform frequency* for each such requested runtime between a pair of default values.
- The distribution of requested runtime is similar for jobs submitted in different months, although a somewhat larger fraction of jobs request 50 - 400 hours during October through December 2000.
- The distribution of requested runtime is similar for different days of the week. Over each three month period, but not during each month within the period, there are slightly fewer vst jobs submitted on Mondays.
- Allowing for statistical variations in finite sample sizes, the distribution of requested runtime is very similar for jobs submitted during different days of the week, and different weekday arrival rate periods. The distributions during weekend intermediate and low periods, not shown in the figure, are also similar.

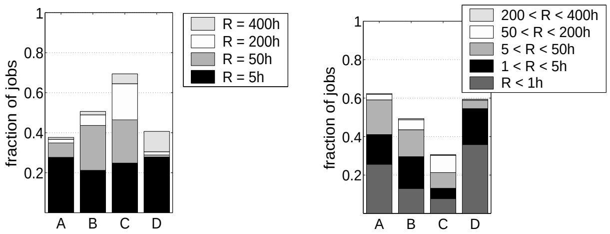
Analysis of the distribution of requested runtime for each range of requested processors paired with each range of requested memory, reveals four distributions (named A through D) that are conditioned on the requested processors and requested memory as shown in Figure 10. The distributions are provided in Figure 11. Although the ranges of processor and memory requests over which each conditional distribution applies are complex, the similarities between Figures 10(a) and (b) are also significant. That is, the recurrence of the distributions in the two different three-month periods suggests that the identified distributions are meaningful.

The conditional distributions can be used to generate a synthetic requested runtime after determining the processor and memory requests. Distribution A has a high fraction of jobs that request less than or equal to 5 hours of runtime, and smaller fractions that request 10, 20, or 50 hours. Distribution B has significant fractions of jobs that request 5 hours, 50 hours, and non-default values



(a) Jan-March 2000 (b) Oct-December 2000

Figure 10. Conditional Distributions (A-D) of Requested Runtime



(a) Default Values of R (b) Ranges Between Defaults

Figure 11. The Distribution Functions

smaller than 50 hours; a smaller fraction request 200 hours. Distribution C has significant fractions of requests for 5, 50, and 200 hours, and a smaller but noticeable fraction that request 400 hours. Distribution D has the highest fractions of jobs that request 1-5 hours and 400 hours. Since most jobs request less than 2 GB of memory, and since distribution A has a significantly lower average requested runtime than distributions B and C, jobs that request greater than 16 processors have a significantly lower average requested runtime than jobs that request fewer processors. For example, the average requested runtime is approximately ten hours larger for serial jobs than that for the 64-processor jobs.

4.6 Summary of Requested Resources

The procedure for creating a synthetic workload that approximates the resource requests of the O2K weekday peak workloads can be summarized as follows:

- Job interarrival times have a two-stage hyperexponential distribution, with mean approximately equal to two minutes and CV in the range of 1-2.
- Requested number of processors has a specialized harmonic distribution that emphasizes powers of two and small numbers, as shown in Figure 4(a).
- The distributions of requested memory, conditioned on the requested number of processors, are given in Figure 7.
- The requested runtime distributions, conditioned on both the requested number of processors and requested memory, are given in Figures 10 - 11. These specialized distributions have significant probability mass at 5, 50, 200, and/or 400 hours, and relatively uniform probability for round numbers between these values, as illustrated in Figure 8.

To create a synthetic workload for intermediate and low arrival rate periods, only the mean interarrival time needs to be modified. Arrival rates for the non-peak arrival periods are provided in Figure 2.

5 Execution Characteristics

Sections 5.1 and 5.2 provide the distributions of actual job runtime and memory usage, respectively. Relationships among these quantities and the job resource requests for processors, memory, and runtime are also provided. Overall average utilization of allocated memory per month is compared against overall average utilization of allocated processors per month. Distribution of processor utilization was analyzed in Section 4.3. Section 5.3 summarizes the execution characteristics.

5.1 Actual Job Runtime

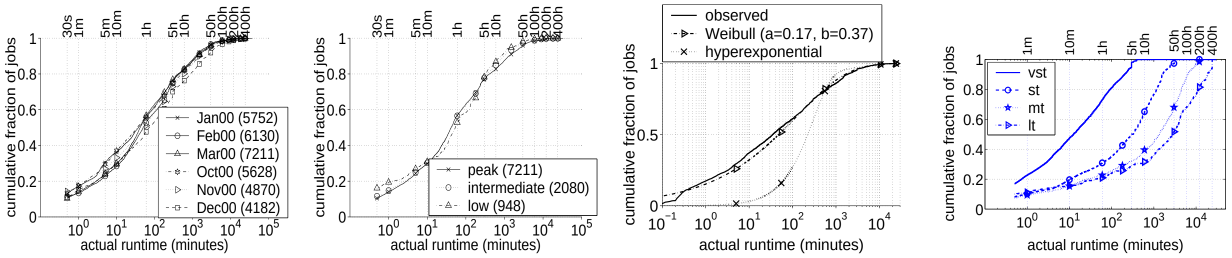
Many previous papers report distributions of actual runtime over a collection of jobs that arrived over a period of several months, without establishing whether the workload is (approximately) stationary over those months. In addition, previous workload studies report on the correlation between actual runtime and number of requested processors [FeNi95, WLF$^+$96, Hoto96] as well as the distribution of the ratio of actual to requested runtime [FeWe98], but not on the correlation between requested runtime and the ratio of actual runtime to requested runtime.

5.1.1 Distribution of Actual Runtime

Figures 12(a) and (b) show that the distribution of actual runtime on the O2K is similar for jobs submitted during weekday peak periods in different months and for jobs submitted during non-peak arrival rate periods, although the coefficient of variation (CV) is slightly lower for jobs submitted during low arrival rate periods. The distribution is also similar for jobs submitted during different days of the week (not shown).

The CV of the overall runtime distribution for the O2K workloads is approximately equal to 3; a runtime CV in the range of 2-5 has been reported for several previous parallel workloads [FeNi95, WLF$^+$96, Gib97b].

Plots of the complement of the runtime distribution on a log-log scale (not shown) reveal that the observed distributions of runtime are not heavy tailed, in contrast to the process runtimes observed on Unix systems [HoDo97] and a hypothetical model of parallel job



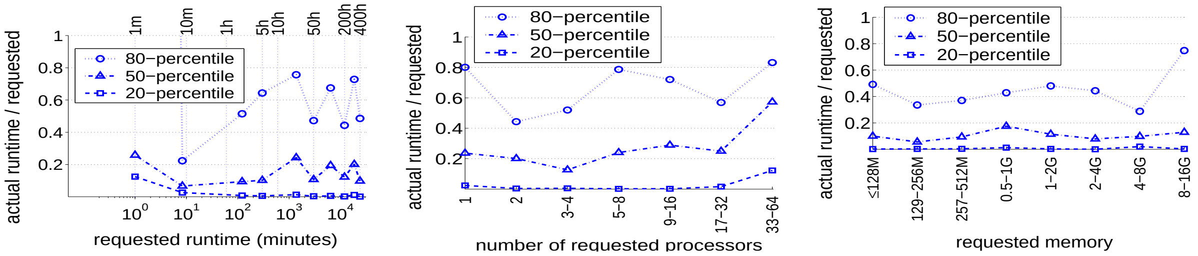
(a) Peak Periods Each Month

(b) Each Weekday Arrival Rate Period (March 2000)

(c) Example Curve Fit Weekdays Peak, Oct 2000 (mean = 566 minutes, cv = 3)

(d) Conditioned on Requested Runtime Classes

Figure 12. Distribution of Actual Runtime



(a) Ranges of Requested Runtime (R)

(b) Ranges of Requested Processors (R Between Defaults)

(c) Ranges of Requested Memory (R = 50, 200, or 400 hrs)

Figure 13. Actual Runtime/Requested Runtime For Various Resource Request Classes
(October-December 2000)

runtimes in [Harc00]. The two-stage hyperexponential distribution has been used to model the observed runtime distribution in a previous parallel workload [Feit96]. As shown in Figure 12(c), the Weibull distribution provides a significantly better fit for actual runtimes on the O2K.[7] The gamma distribution (not shown) does not fit the tail of the observed distribution. Since the observed distribution is approximately piecewise linear in the logarithm of the observed actual runtime, a piecewise log-uniform distribution as proposed in [DoFe99] fits the observed distribution nearly as well as the Weibull distribution.

Similar results are obtained for modeling the distribution of total processing time (i.e., total number of cpu hours rather than total runtime) per job. That is, the Weibull and piecewise log-uniform distributions closely approximate the observed distribution, whereas the gamma, two-stage hyperexponential, and heavy-tailed Pareto distributions do not.

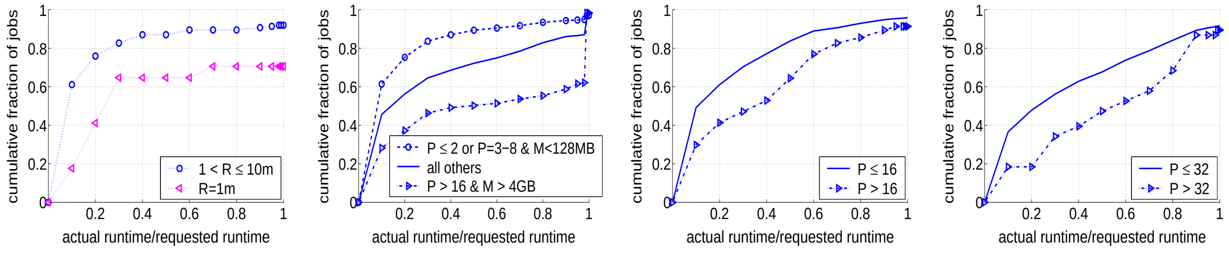
Plots of the average and percentiles of the actual runtime for each range of requested processors (omitted to conserve space) reveal that there is no appreciable correlation between the actual runtime and requested number of processors, in contrast to previous workloads on an iPSC/860 [FeNi95], a Paragon [WLF+96], and an SP/2 [Hoto96].

5.1.2 Actual vs. Requested Runtime

Figure 12(d) and Figures 13(a)-(c) show that there is a large discrepancy between actual and requested runtime for most jobs. For example, approximately 10% of the st, mt, and lt jobs (which request over 5, 50, or 200 hours of runtime, respectively) actually complete in under one minute. Another 10-20% of such jobs complete in under one hour. Furthermore, more than 50% of all jobs have actual runtime that is less than 20% of their requested runtime, and 50% of the jobs that request default runtimes have actual runtime less than 10-12% of their requested runtime. In contrast, Feitelson and Weil [FeWe98] reported that the ratio of the actual to requested runtime on an SP/2 ranges uniformly from 0-99%. [FeWe98] also simulated the SP/2 job trace with accurate requested runtimes and showed that the average of the job turnaround time divided by

---

[7] The hyperexponential distribution shown in the figure matches only the first two moments of the observed distribution. Using the algorithm in [JPF+97] to fit a hyperErlang distribution to the observed O2K runtime distribution, results in a two-stage hyperexponential distribution that matches the first three non-central moments of the observed data, but this distribution has a slightly worse fit to the full distribution than the fit of the hyperexponential distribution shown in the figure. Note that if a linear scale is used for the x axis in the figure the hyperexponential distribution will appear to be more accurate.



(a) R $\leq$ 10min (b) R = 5 hrs (c) R = 50, 200, or 400 hrs (d) R Between Defaults

Figure 14. Conditional Distributions of Actual Runtime/Requested Runtime
(R: Requested Runtime)

actual runtime (i.e., the mean slowdown) improved for a non-preemptive scheduler similar to the current O2K scheduler. Similarly, mean turnaround time and mean slowdown for jobs on the O2K might be improved if requested runtimes could be specified more accurately.

A key question is how to generate actual runtimes in a synthetic workload if requested runtimes are generated as discussed in Section 4.5. The percentiles in Figure 13(a) were computed for the default requested runtimes (i.e., 5,50,200, and 400 hours), each range between the defaults, and the following ranges of requested runtime (R): R=1 minute, 1 min. $<$ R $\leq$ 10 min., 10 min $<$ R $<$ 5 hrs.[8]. This figure shows that the ratio of actual to requested runtime is statistically somewhat higher for jobs that request greater than 5 hours of runtime but not one of the default values than for jobs that request one of the default values. Figure 13(b) shows that the distribution for the former category of runtime requests is similar for each number of requested processors up to 32, but has a much higher fiftieth percentile if the requested number of processors is greater than 32. A similar graph (not shown) shows that the distribution for jobs that request 5 hours of runtime is similar for each number of requested processors greater than 16. For each category of requested runtime and requested number of processors that has a similar distribution of the ratio of actual to requested runtime, we plotted the distribution as a function of requested memory, as illustrated in Figure 13(c) for requested runtime equal to 5 hours and requested processors greater than 16. In Figure 13(c) the distribution is different for requested memory greater than 4 GB; in all other cases, the distribution is not significantly sensitive to the requested memory.

Based on these considerations, Figure 14 provides distributions of the ratio of actual to requested runtime conditioned on the requested runtime, requested processors, and requested memory. Recall that actual runtime can exceed requested runtime by one hour before the job is killed. The results in Figure 14(a) show that a significant fraction of the jobs that request one minute of runtime exceed their request. The results in Figure 14(b) show that a significant fraction of the jobs that request 5 hours of runtime are tuned to run for exactly 5 hours.

A synthetic workload generator can use the distributions provided in Figure 14 to generate the actual runtime as a fraction of requested runtime after requested runtime and number of processors have been generated.

5.2 Memory Usage

For each job the JMD logs record the job's peak memory usage during each 30 second interval of runtime. A small percentage (i.e., 0-4%) of the jobs that have at least one minute of actual runtime during each month do not have JMD log information about actual cpu and memory usage; these jobs are not included in the analysis of memory usage characteristics in this section.

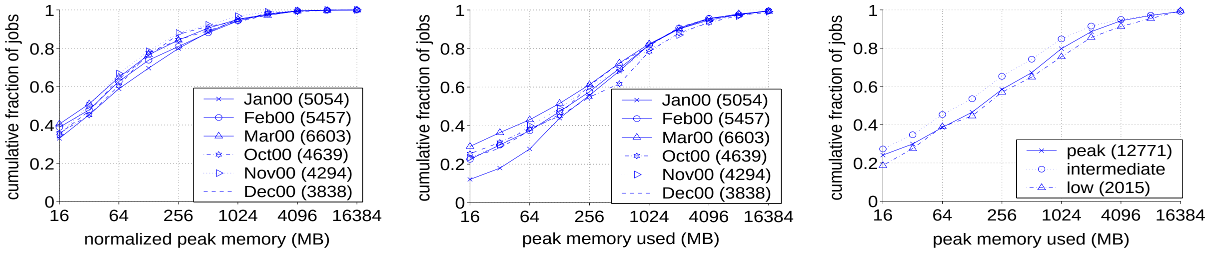
The peak memory used by the job is the maximum of these 30-second peak values. The average memory used by the job is computed as the average of the 30-second peak values. The *normalized* peak memory used is defined to be the ratio of the peak memory used divided by the number of processors requested. The peak (or average) *memory efficiency* is defined as the ratio of peak (or average) memory used to memory requested.

Figure 15 shows that there is some variability in the distributions of peak memory usage over different months and over different arrival rate periods. In particular, jobs submitted during intermediate arrival rate periods tend to have somewhat lower peak memory usage. In the remainder of this section we focus on further characterizing representative peak memory usage for jobs submitted during weekday peak arrival rate periods in October through December 2000. Parameters of the representative distributions for these periods could be adjusted to reflect the observed variations for different months or for the intermediate arrival rate period.

Similar to previous workloads [Feit97, SSN99], Figure 15(a) shows that a large fraction (i.e., approximately

---

[8] The value of each percentile for each range of runtime is plotted against the average requested runtime in the range.

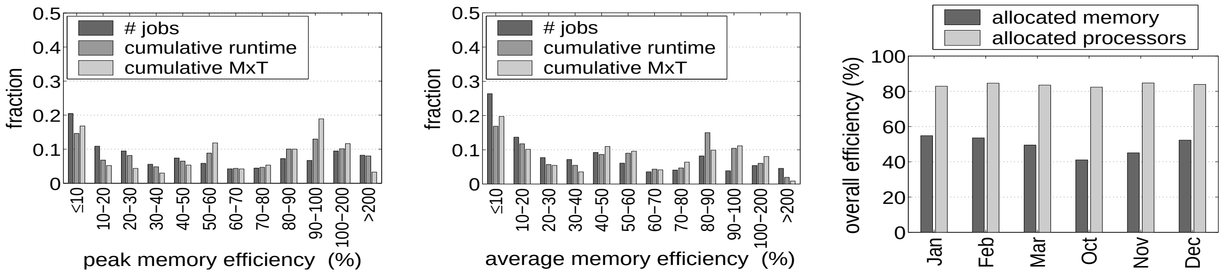


(a) Normalized Peak Memory
Weekday Peak Periods Each Month

(b) Peak Memory
Weekday Peak Periods Each Month

(c) Peak Memory
Each Weekday Arrival Period
(October-December 2000)

Figure 15. Variations in Distribution of Peak Memory Used
(Normalized Peak Memory = Peak Memory Used/Number of Requested Processors)



(a) Peak Memory Efficiency

(b) Average Memory Efficiency

(c) Overall Average Efficiency

Figure 16. Memory Efficiency

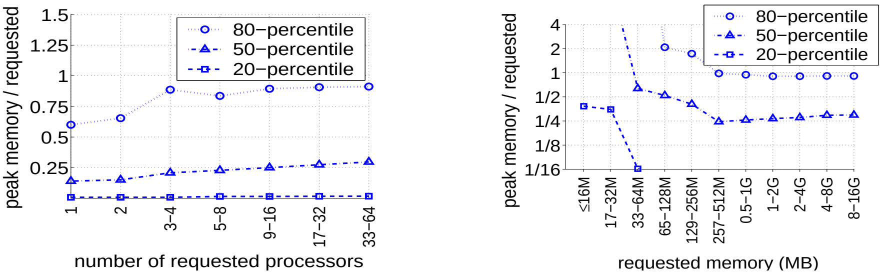
50%) of the jobs on the O2K have a very small (i.e., under 32 MB) peak memory usage per processor. On the other hand, another significant fraction (about 5%) of the jobs on the O2K have normalized peak memory usage greater than 1 GB per processor.

Figure 16 shows that there is a large discrepancy between requested memory and memory utilized per job. In particular, Figures 16(a) and (b) show that the respective ratio of average or peak memory usage to the requested memory is distributed fairly uniformly over each 10% range from 10-100%, and 15-20% of the jobs have peak memory usage higher than their requested memory. A similar result was reported for a CM-5 workload in [Feit97]. Figure 16(c) shows that the average memory efficiency varies between 40-55% from month to month, which is significantly lower than the time-average utilization of the requested processors. As with the large errors in requested runtime on the O2K, jobs might be scheduled more efficiently if memory requests were more accurate.

Peak memory usage is an important job characteristic in that it defines the amount of memory that must be allocated to the job (in a shared memory system such at the O2K) in order to guarantee that the job doesn’t experience any memory interference during execution. To further characterize peak memory usage for the purpose of creating synthetic workloads, we analyze the correlations between peak memory efficiency and other job characteristics. To that end, Figure 17 plots the percentiles of the peak memory efficiency per job, for ranges of requested number of processors, requested memory, and actual runtime.

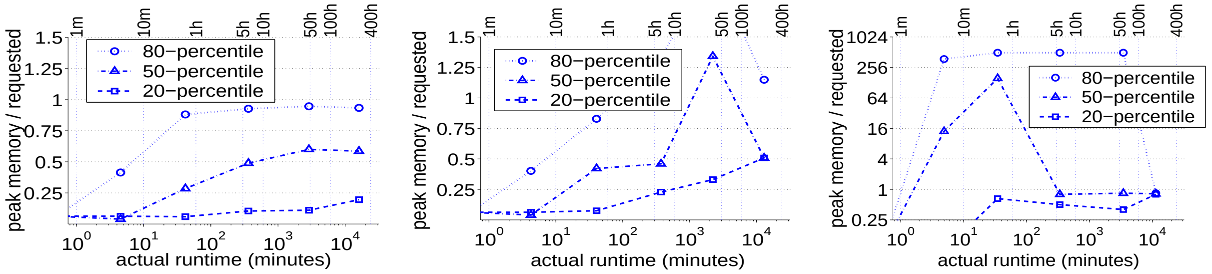
Figure 17(a) shows that the peak memory efficiency per job is fairly insensitive to the requested number of processors. An implication of this result is that peak memory usage is positively correlated with the number of requested processors, since requested memory and requested processors are positively correlated (see Figure 6(b)). Similarly, there is a negative correlation between the normalized peak memory usage and the number of requested processors ((see Figure 6(a)). In contrast, in the CM-5 [Feit97] workload, the jobs with a larger number of processors not only use a larger amount of memory, but also a larger amount of per-processor memory.

Figure 17(b) shows that the distribution of peak memory efficiency is significantly different for jobs that request fewer than 128 MB of memory than for jobs that request more than 128 MB of memory. From this figure, noting that very few jobs request 33-64 MB of memory and that the 80-percentile value for these memory requests is unreliable based on the small number of jobs, we parti-



(a) Ranges of Requested Processors

(b) Ranges of Requested Memory

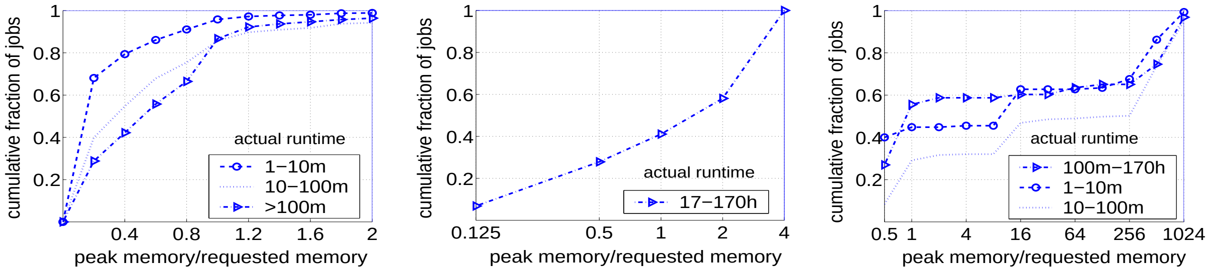


(c) Ranges of Actual Runtime (Requested Memory > 128 MB)

(d) Ranges of Actual Runtime (Requested Memory: 33-128 MB)

(e) Ranges of Actual Runtime (Requested Memory ≤ 32 MB)

Figure 17. Peak Memory/Requested Memory For Various Resource Request Classes



(a) > 128 MB and 33-128 MB for 0-17h, >170h

(b) 33-128 MB

(c) ≤ 32 MB

Figure 18. Conditional Distributions of Peak Memory/Requested Memory
(Weekday Peak Periods, October-December 2000)

tion the jobs into three classes according to their memory request (i.e., ≤ 32 MB, 33-128 MB, and > 128 MB) and provide the percentiles of peak memory efficiency as a function of actual job runtime for each class in Figures 17(c) - (e).[9]

Note that all jobs that have runtime under one minute use a very small fraction of their memory request; in fact, nearly all such jobs use less than 64 MB of memory.

The distribution of peak memory used as a fraction of requested memory is very similar for runtime greater than 10 minutes and memory request greater than 32 MB, except for jobs with memory request of 33-128 MB and actual runtime in the range of 1,000-10,000 minutes (i.e., approximately 17-170 hours). Thus, Figure 18 provides the requisite distributions for generating peak memory as a fraction of memory requested after requested memory and actual runtime have been generated as described earlier in this paper.

5.3 Summary of Execution Characteristics

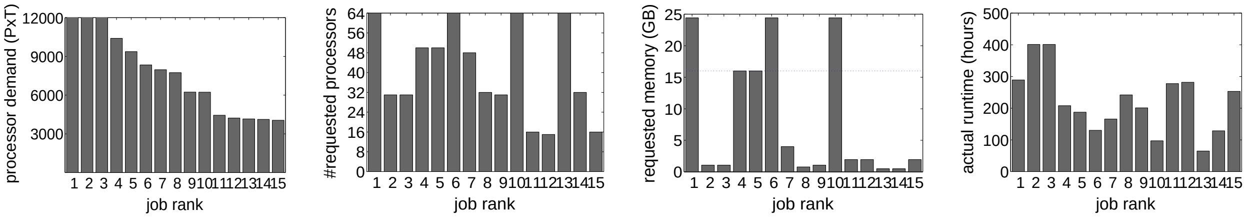
To generate actual runtime and peak memory usage to complete a synthetic workload that is representative of the O2K weekday peak arrival rate workloads, these characteristics are obtained from the distributions in Figure 14 and 18, respectively. Processor utilization can also be generated from the distribution in Figure 4(b). To create a synthetic workload for intermediate arrival

---

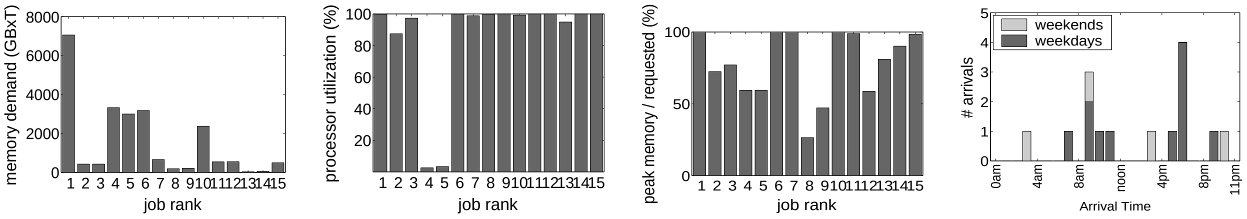
[9] The ranges of actual runtime over which the percentiles in Figure 17(c)-(e) are computed are: 0-1 minute, 1-10 minutes, 10-100 minutes, 1.7-17 hours, 17-170 hours, and above 170 hours.

Table 6. **Largest Jobs Each Month**

Month	#Jobs	Demand/Demand All		Job Size			
		Processor Demand	Memory Demand	Avg Processors Requested	Avg Memory Requested (GB)	Avg Actual Runtime (hours)	Avg Processor Demand (P×T hours)
Jan 2000	15	17.1%	15.7%	27.7	6.1	311.0	7173
Feb 2000	15	16.5%	17.2%	36.1	7.7	254.5	7031
Mar 2000	15	15.6%	11.0%	36.1	6.6	209.5	6944
Oct 2000	15	9.1%	3.4%	41.7	3.8	102.9	3888
Nov 2000	15	11.1%	9.2%	39.8	5.1	168.4	4657
Dec 2000	15	16.6%	13.3%	40.5	8.0	222.0	8044



(a) P×T hours (b) #Requested Processors (c) Requested Memory (d) Actual Run Time



(e) M×T hours (f) Processor Utilization (g) Peak Memory Efficiency (h) Job Arrival Times

Figure 19. Characteristics of the Fifteen Largest Jobs for A Recent Month (December 2000)

rate periods, the peak memory usage might be adjusted slightly as shown in Figure 15(c). Low arrival rate periods have approximately the same distributions of actual runtime and peak memory usage as weekday peak arrival rate periods.

Note that, depending on the purpose of the synthetic workload, any of the characteristics that are not needed can be ignored.

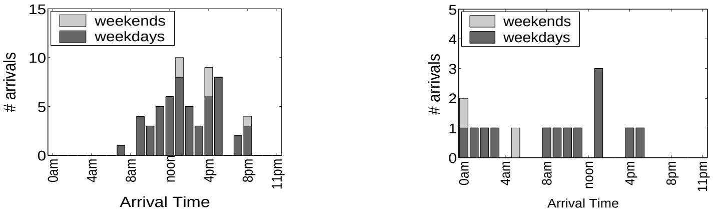
6 The Largest Jobs Each Month

This section provides characteristics of the fifteen jobs that have the largest processor demand (i.e., the product of the number of requested processors and the actual runtime of the job) each month.

Table 6 summarizes the total processor and memory demand, and the average job size of these largest jobs for each month. Figure 19 provides more detailed characteristics of the largest jobs in December 2000, a recent month in which the average processor demand per largest job is high.

For most months, the 15 largest jobs account for 10-15% of the total processor and memory demand for the month. Also for most months, these largest jobs have an average of over 200 hours of actual runtime and and average demand of 7000-8000 processor hours, while the respective averages over all jobs in the monthly workload is at least an order of magnitude smaller.

As shown in Figure 19(b) and (d), the top three jobs in December 2000 had run times of 300-400 hours on 32-64 processors. Although two of the largest jobs in December 2000 had very low processor utilization, Figure 19(f) shows that many of the largest jobs achieve nearly 100% utilization of their requested processors. For the other five months analyzed, (1) 2-7 jobs per month have approximately 100% processor utilization, and (2) 2-5 jobs per month have under 50% utilization,



(a) January-March and November 2000

(b) October 2000

Figure 20. Job Arrival Times For the Fifteen Largest Jobs in the Other Months

of which a total of three jobs have utilization under 10%.

As shown in Figure 19(c) the largest jobs request either the maximum memory (16-25 GB), or under 4 GB of memory. Peak memory usage divided by memory requested (Figure 19(g)) tends to be somewhat higher (i.e., 60-100% for most of the largest jobs) than for the workload as a whole. On the other hand, a few of the largest jobs have peak memory efficiency under 20%.

Figures 19(h) and 20 show that most of the largest jobs of each month arrive during weekday peak arrival rate hours. However, during October 2000 a significant fraction of the top jobs were submitted during weekday low arrival rate hours (i.e., midnight to 4am).

7 Conclusions

This paper has provided a characterization of the large production parallel workload submitted to the NCSA O2K over two three month periods. This characterization is more complete than previous parallel workload characterizations in that new characteristics are provided (e.g., distributions of requested runtime, processor and memory utilizations, distribution of requested memory over more flexible range of possible requests), correlations among the characteristics are more fully explored, and conditional distributions are provided for generating synthetic workloads that include the observed correlations in the O2K workload. Another key difference in this analysis as compared with prior work is that job characteristics are provided for jobs that are submitted during periods of approximately stationary job arrival rate. From these characteristics we determined that the jobs submitted in different months or in different arrival rate periods are statistically very similar. The roadmaps for generating similar synthetic workloads are summarized in Sections 4.6 and 5.3.

Interesting characteristics of the O2K workload include: (a) the fifteen largest jobs in a typical month have average running time of over 200 hours and use an average of 4000-8000 processor hours, (b) most requested runtimes are default values (i.e., 5,50,200, or 400 hours), (c) whether or not a default runtime is requested, over half the jobs have actual runtime less than 20% of the requested value, and (d) overall utilization of allocated processors is approximately 80% whereas overall utilization of allocated memory is closer to 50%.

Some of the O2K workload characteristics that differ from previous workloads (e.g., the CV of job interarrival time equal to 1-2 instead of 2.5-6) are most likely due to measuring the O2K characteristics during periods of stationarity. Other differences (e.g., longer runtimes and larger memory requests) are most likely due to general trends in production parallel systems. Still other differences (e.g., lack of correlation between requested runtime and requested number of processors, or the large number of jobs with very inaccurate requested runtime) may either be due to the trend toward more widespread use of parallel/distributed computing, or may instead be reflective of the O2K usage and environment. Characterization of further modern production parallel/distributed computing workloads are needed to distinguish the trends from the environment-specific results.

Topics for future research include (1) analysis of the workload characteristics of the jobs submitted during the occasional high-load days (i.e., the days that have over 500 arrivals rather than the more typical 350 arrivals), to determine whether such jobs have different characteristics than the jobs submitted during typical weekday peak periods, (2) a more detailed analysis of whether statistical fluctuations in finite sample sizes accounts for the small fluctuations observed among daily and monthly workloads on the O2K, (3) analysis of the characteristics of the jobs that request dedicated time on the NCSA O2K hosts, (4) investigation of the feasibility of more accurate requested runtimes, and the impact of more accurate requested runtimes on the performance of the O2K scheduling policy, and (5) analysis of (future) production workloads on clusters of workstations and for more widely distributed computing resources.

Acknowledgments

This work was partially supported by the NSF PACI Program through a partnership grant from the NCSA. The authors thank the anonymous reviewers for comments that improved the quality of the paper.

References

[All90] A. O. Allen. *Probability, Statistics, and Queueing Theory with Computer Science Applications*. Academic Press, 2nd Ed., 1990.

[ChVe01] S.-H. Chiang and M. Vernon. "Production Job Scheduling for Parallel Shared Memory Systems". *Proc. Int'l. Parallel and Distributed Processing Symp. (IPDPS 2001)*, San Francisco, April 2001.

[DoFe99] A. B. Downey and D. G. Feitelson. "The Elusive Goal of Workload Characterization". *Performance Eval. Rev.*, 26(4):14–29, March 1999.

[Down97] A. B. Downey. "Predicting queue times on Space-Sharing Parallel Computers". *Proc. 3rd Workshop on Job Scheduling Strategies for Parallel Processing*, Geneva, April 1997. Lecture Notes in Comp. Sci. Vol. 1291, Springer-Verlag.

[Feit96] D. G. Feitelson. "Packing Schemes for Gang Scheduling". *Proc. 2nd Workshop on Job Scheduling Strategies for Parallel Processing*, pp. 89–110, Honolulu, April 1996. Lecture Notes in Comp. Sci. Vol. 1162, Springer-Verlag.

[Feit97] D. G. Feitelson. "Memory Usage in the LANL CM-5 Workload". *Proc. 3rd Workshop on Job Scheduling Strategies for Parallel Processing*, pp. 78–94, Geneva, April 1997. Lecture Notes in Comp. Sci. Vol. 1291, Springer-Verlag.

[FeNi95] D. G. Feitelson and B. Nitzberg. "Job characteristics of a production parallel scientific workload on the NASA Ames iPSC/860". *Proc. 1st Workshop on Job Scheduling Strategies for Parallel Processing*, pp. 337–360, Santa Barbara, April 1995. Lecture Notes in Comp. Sci. Vol. 949, Springer-Verlag.

[FeWe98] D. G. Feitelson and Ahuva Mu'alem Weil. "Utilization and Predictability in Scheduling the IBM SP2 with Backfilling". *Proc. 12th Int'l. Parallel Processing Symp.*, pp. 542–546, Orlando, March 1998.

[Gib97a] R. Gibbons. "A Historical Application Profiler for Use by Parallel Schedulers". Master's thesis, Univ. of Toronto, Ontario, 1997.

[Gib97b] R. Gibbons. "A Historical Application Profiler for Use by Parallel Schedulers". *Proc. 3rd Workshop on Job Scheduling Strategies for Parallel Processing*, Geneva, April 1997. Lecture Notes in Comp. Sci. Vol. 1291, Springer-Verlag.

[Harc00] M. Harchol-Balter. "Task Assignment with Unknown Duration". *Proc. Int'l. Conf. on Distributed Computing Systems*, Taipei, Taiwan, April 2000.

[HoDo97] M. Harchol-Balter and A. B. Downey. "Exploiting Process Lifetime Distributions for Dynamic Load Balancing". *ACM Trans. on Computer Systems*, 15(3):253–285, August 1997.

[Hoto96] S. Hotovy. "Workload Evolution on the Cornell Theory Center IBM SP2". *Proc. 2nd Workshop on Job Scheduling Strategies for Parallel Processing*, pp. 27–40, Honolulu, April 1996. Lecture Notes in Comp. Sci. Vol. 1162, Springer-Verlag.

[HSO96] S. Hotovy, D. Schneider, and T. O'Donnell. "Analysis of the Early Workload on the Cornell Theory Center IBM SP2". Technical Report TR234, Cornell Theory Center, January 1996.

[JKK92] N. L. Johnson, S. Kotz, and A. W. Kemp. *Univariate Discrete Distributions*. Wiley, 2nd Ed., 1992.

[JPF$^+$97] J. Jann, P. Pattnaik, H. Franke, F. Wang, J. Skovira, and J. Riodan. "Modeling of Workload in MPPs". *Proc. 3rd Workshop on Job Scheduling Strategies for Parallel Processing*, Geneva, April 1997. Lecture Notes in Comp. Sci. Vol. 1291, Springer-Verlag.

[LSF] *Platform Computing Corporation*. North York, Canada. http://www.platform.com/.

[Mat] *MathWorks*. http://www.mathworks.com/.

[MS] *Maui Scheduler*. http://www.mhpcc.edu/maui/.

[SGS96] J. Subhlok, T. Gross, and T. Suzuoka. "Impact of Job Mix on Optimizations for Space Sharing Schedulers". *Proc. 1996 ACM/IEEE Supercomputing Conf.*, Pittsburgh, November 1996.

[SSN99] S. K. Setia, M. S. Squillante, and V. K. Naik. "The Impact of Job Memory Requirements on Gang-Scheduling Performance". *Performance Eval. Rev.*, 26(4):30–39, March 1999.

[SYZ99] M. S. Squillante, D. D. Yao, and L. Zhang. "The Impact of Job Arrival Patterns on Parallel Scheduling". *Performance Eval. Rev.*, 26(4):52–59, March 1999.

[WLF$^+$96] K. Windisch, V. Lo, D. Feitelson, B. Nitzberg, and R. Moore. "A Comparison of Workload Traces from Two Production Parallel Machines". *Proc. 6th Symp. on the Frontiers of Massively Parallel Computation*, pp. 319–326, October 1996.

[WMKS96] M. Wan, R. Moore, G. Kremenek, and K. Steube. "A Batch Scheduler for the Intel Paragon MPP System with a Non-contiguous Node Allocation Algorithm". *Proc. 2nd Workshop on Job Scheduling Strategies for Parallel Processing*, pp. 48–64, Honolulu, April 1996. Lecture Notes in Comp. Sci. Vol. 1162, Springer-Verlag.