Preprint
UCRL-JC-140092

The Characteristics of Workload on ASCI Blue-Pacific at Lawrence Livermore National Laboratory

A.B. Yoo, M.A. Jette

This article was submitted to International Association of Science and Technology for Development, International Conference on Applied Informatics, Insbruck, Austria, February 19-22, 2001

August 14, 2000

The Characteristics of Workload on ASCI Blue-Pacific at Lawrence Livermore National Laboratory*

Andy B. Yoo and Morris A. Jette

Lawrence Livermore National Laboratory
Livermore, CA 94551
e-mail: {yoo2 | jette1}@llnl.gov

1 Motivation

Symmetric multiprocessor (SMP) clusters have become the prevalent computing platforms for large-scale scientific computation in recent years mainly due to their good scalability. In fact, many parallel machines being used at supercomputing centers and national laboratories are of this type [2, 3, 4, 19]. It is critical and often very difficult on such large-scale parallel computers to efficiently manage a stream of jobs, whose requirement for resources and computing time greatly varies. Understanding the characteristics of workload imposed on a target environment plays a crucial role in managing system resources and developing an efficient resource management scheme.

A parallel workload is analyzed typically by studying the traces from actual production parallel machines. The study of the workload traces not only provides the system designers with insight on how to design good processor allocation and job scheduling policies for efficient resource management [8, 1, 16, 6] but also helps system administrators monitor and fine-tune the resource management strategies and algorithms [12]. Furthermore, the workload traces are valuable resource for those who conduct performance studies through either simulation or analytical modeling. The workload traces can be directly fed to a trace-driven simulator in a more realistic and specific simulation experiments [18]. Alternatively, one can obtain certain parameters that characterize the workload by analyzing the traces, and then use them to construct a workload model [10, 20, 7, 11, 14] or to drive a simulation in which a large number of runs are required.

Considering these benefits, we collected and analyzed the job traces from ASCI Blue-Pacific, a 336-node IBM SP2 machine at Lawrence Livermore National Laboratory (LLNL) [3]. The job traces used span a period of about six months, from October 1999 till the first week of May 2000. The IBM SP2 machine at the LLNL uses *gang scheduling LoadLeveler* (GangLL) [13] to manage parallel jobs. User jobs are submitted to the GangLL via a locally developed resource manager called *Distributed Production Control*

*This work was performed under the auspices of the U.S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

System (DPCS) [5]. The DPCS prioritizes jobs based upon a fair-share resource allocation hierarchy and uses a back-fill algorithm [17] to optimize scheduling. The DPCS records all of its activities as well as accounting information for user jobs in a log, from which we collected the job traces. The log can provide quite extensive information on jobs submitted to the DPCS, but we concentrate only on the information pertaining to the service and resource demands of the jobs.

In this paper, we report the results of our workload study in three categories. Job submission and execution characteristics, resource requirement analysis, and system utilization analysis. Submission and execution characteristics of a job include parameters pertaining to queueing activities in the system such as job submission rate, job wait time, and job service time. In resource requirement analysis, the demands for computing nodes and main memory from each job are analyzed. As a part of the resource requirement analysis, we have conducted correlation analysis in an attempt to determine whether there are any correlations between various resource demands and job execution time. Finally, we analyzed how the system is used by groups of jobs exhibiting different resource demands and submission and execution characteristics.

The contributions of this study are two-fold. First, the workload on the ASCI Blue-Pacific represents a typical workload in a very large-scale scientific computing environment and hence is unique in terms of the amount of and the variation in computation and resource demands. To the best of our knowledge, this is the first attempt to analyze the workload on such a very large-scale scientific computing platform. Second, the workload characteristics reported in this paper are very comprehensive. This paper provides comprehensive information on various resource demands of jobs. Notably, the memory usage of jobs on a large IBM SP2 has never been reported in the literature. We are confident that this work will be a valuable resource for those who conduct performance studies or develop resource management schemes for very large-scale parallel computers.

2 Job Submission and Execution Characteristics

Fig. 1 shows the inter-arrival distributions for weekday and weekend jobs. As Fig. 1 clearly indicates, most jobs have very short inter-arrival time. The mean job inter-arrival time is 14.31 minutes. Furthermore, the mean job inter-arrival time measurements for weekday and weekend jobs are 8.46 minutes and 44.84 minutes, respectively. About 85% of weekday jobs have the inter-arrival time of 10 minutes or less. For weekend jobs, the distribution curve fattens out because there were not many jobs submitted during the weekends. Still, the inter-arrival times of majority of jobs (about 65%) are less than 10 minutes. There are small percentage of jobs that exhibit inter-arrival time of more than 60 minutes. This is partly due to scheduled downtime for maintenance. Jobs submitted after an extended down time will exhibit an unusually long inter-arrival time and this accounts for the small rises at the end of the distribution curves. The shape and large coefficient of variation of these distribution curves suggest that the job inter-arrival

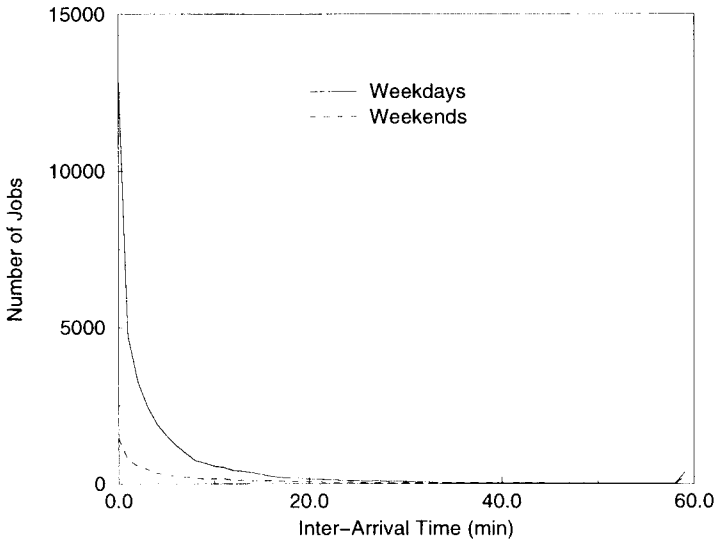

Figure 1: Inter-arrival time distributions for weekday and weekend jobs.

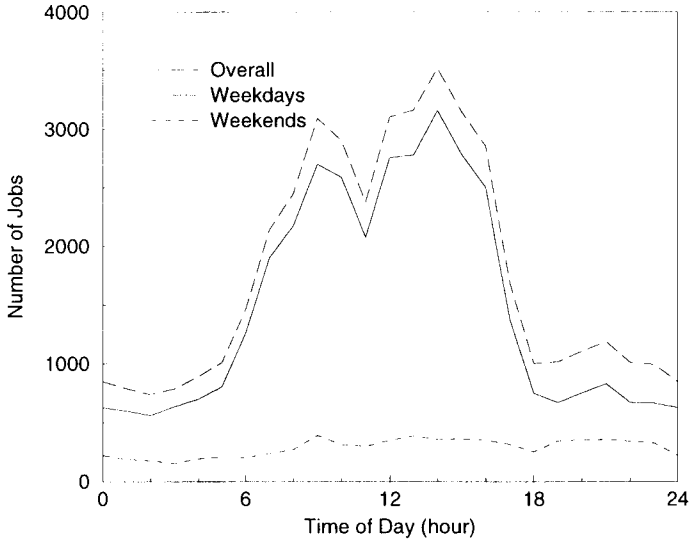

Figure 2: Job submission as a function of time of the day.

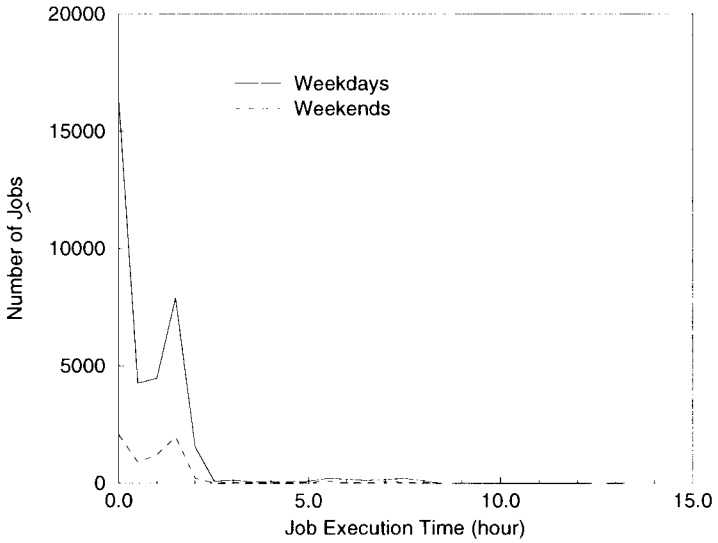

Figure 3: Job execution time distributions for weekday and weekend jobs.

time distribution probably can be fitted adequately with Hyper Erlang Distribution of Common Order which was modeled in [14].

Fig. 2 shows the number of submitted jobs for different hours of the day. Here, 85% of jobs are submitted during the weekdays. The job submission rate varies widely through the day with a sharp increase at 6 AM, a modest dip at noon, and steep decrease after 4 PM. Among the observed jobs, 58% were submitted during the normal business hours (8 AM to 5 PM). A small depression around 11 AM is probably due to the lunch break. Jobs submitted during weekends constitute 15% of total jobs, and the number of those weekend jobs is relatively uniformly distributed over the time of the day.

Fig. 3 displays the execution time distributions for weekday and weekend jobs. An important finding here is that most jobs are short-running. Fig. 3 reveals that 55% of total jobs ran for less than an hour. Overall, 91% of total weekday jobs ran for less than two hours. The scheduling policy employed at LLNL limits the jobs initiated between 8 AM and 5 PM on weekdays to two hours. This time limit increases to eight hours at night and twelve hours on weekends. Rather than execute different jobs at different times, many users maintain a two hour time limit on most of their jobs so they can be initiated at any time during the week.

Figures 4 and 5 present the job wait time distributions and the average job wait time as a function of time of the day, respectively. Fig. 4 shows that about 60% of total jobs wait on the queue for less than one hour. Many jobs do have to wait in the queue for many hours before they get scheduled. Further investigation has revealed that these jobs with long wait time usually request a large number of nodes. We have also observed that the jobs submitted during the weekday afternoon have longer wait time. This can be explained in conjunction with job submission behavior of users as mentioned earlier. As shown in

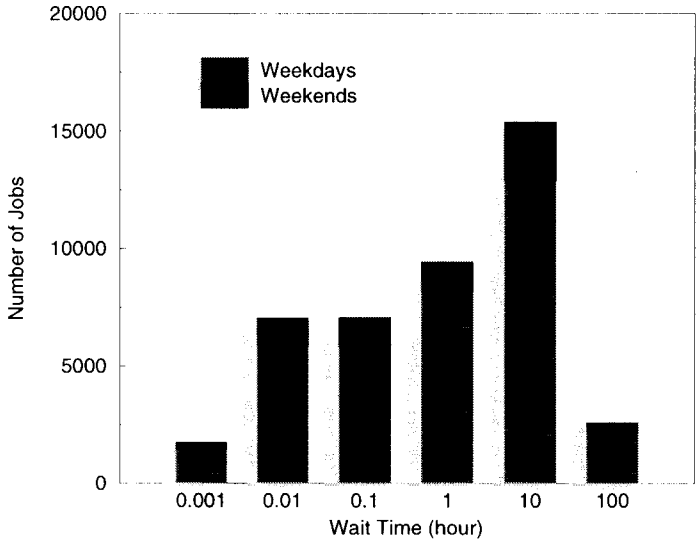

Figure 4: Job wait time distributions for weekday and weekend jobs.

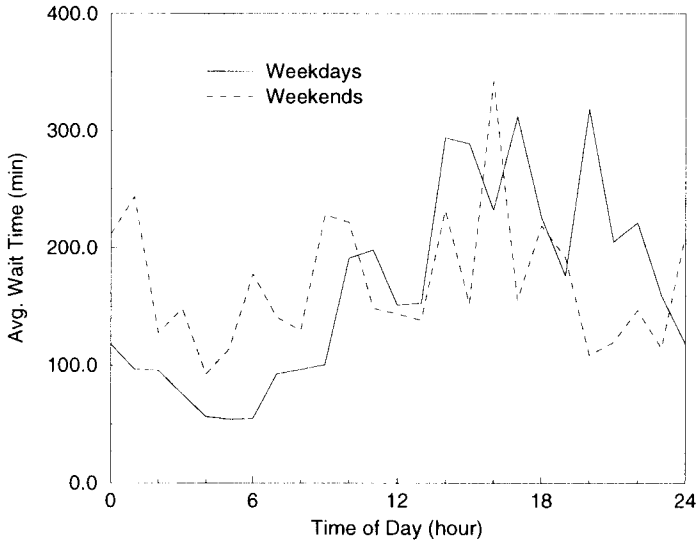

Figure 5: Average job wait time for different hours of the day.

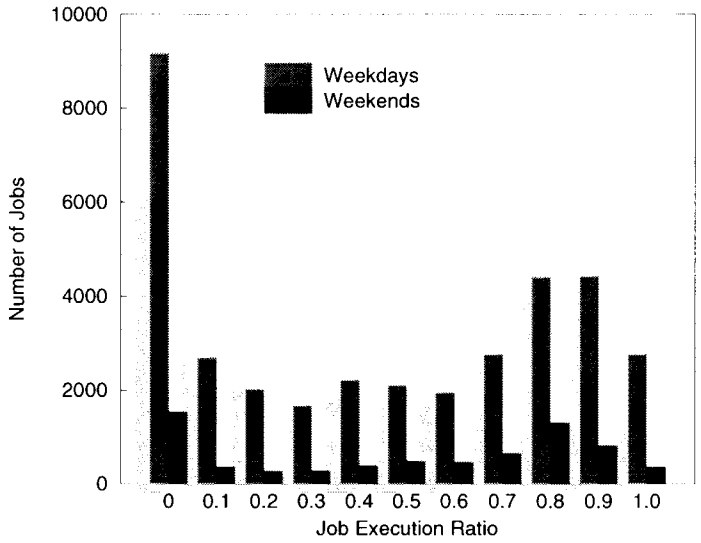

Figure 6: Job execution ratio distributions for weekday and weekend jobs.

Fig. 2, most weekday jobs are submitted from 9 AM to 4 PM, and therefore they have to wait longer in the queue. There is a peak at 8 PM, because jobs submitted at this time of the day may have to wait behind the larger jobs submitted earlier.

Fig. 6 depicts the distributions of job execution ratio. The execution ratio of a job is defined as the ratio of actual execution time of the job to maximum execution time that user has requested. It is used as an indicator of how closely users estimate the execution time of their jobs. Many users who submit jobs during weekdays grossly overestimate the execution time as shown in Fig. 6, where 33% of weekday jobs have the execution ratio of less than 0.1. The reason for this is two-fold. First, many jobs submitted during weekdays are still in development stage and hence error-prone. These jobs tend to terminate prematurely due to the errors. The weekend jobs tend to exhibit a higher execution ratio because they tend to be more mature and longer-running. The second reason is that users are most concerned about running their jobs successfully and hence usually neglect to provide a close time limit estimate for their jobs.

3 Resource Requirements

We present the results of node requirement analysis in Fig. 7. Two things are clearly noticeable in the graph. First, the great majority of jobs require small number of nodes for their execution. Among the jobs observed, 57% require 16 nodes or less, and 25% are medium-sized jobs requiring with 32 to 64 nodes. However, only a small fraction of the observed jobs (4%) require more than 64 nodes. The second finding is that many jobs have sizes that are a power of two (56%). This may be attributed to the nature

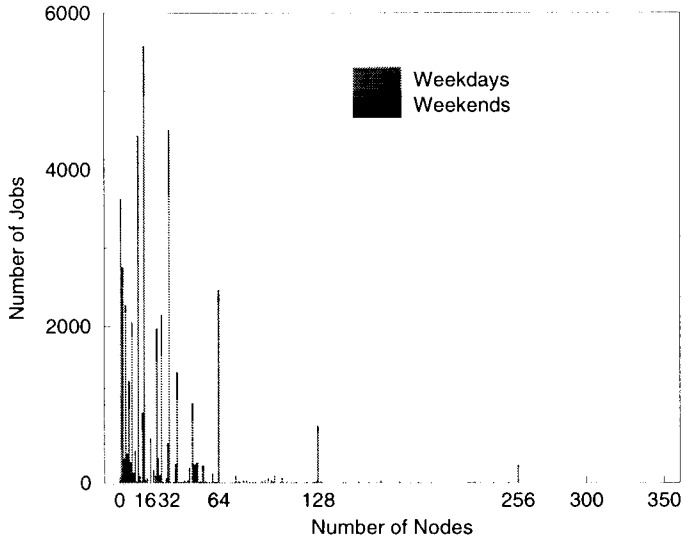

Figure 7: Node usage distributions for weekday and weekend jobs.

Power-of-two sizes	Weekday jobs	Weekend jobs	Total
1	3100 (7.16%)	529 (1.22%)	3629 (8.38%)
2	2533 (5.85%)	224 (0.52%)	2757 (6.37%)
4	2184 (5.04%)	89 (0.21%)	2273 (5.25%)
8	1870 (4.32%)	187 (0.43%)	2057 (4.85%)
16	4936 (11.40%)	643 (1.49%)	5579 (12.89%)
32	3676 (8.49%)	827 (1.91%)	4503 (10.40%)
64	2284 (5.27%)	179 (0.41%)	2463 (5.68%)
128	664 (1.53%)	68 (0.16%)	732 (1.69%)
256	196 (0.45%)	31 (0.07%)	227 (0.52%)

Table 1: Distributions of the power-of-two jobs.

of the jobs running on Blue-Pacific, most of which are scientific applications where the problem space is usually divided into two or four subspaces recursively. Table 1 gives the break-down information on jobs with power-of-two sizes. It is interesting to see that our findings concur with what has been reported in [20], where the traces from an Intel Paragon at the San Diego Supercomputer Center (SDSC) were analyzed.

Fig. 8 presents the average number of nodes requested by jobs for different time of the day. During the normal business hours (8 AM to 5 PM), the graph for weekday jobs exhibits a steady node requirement behavior, conforming with what is shown in Fig. 7, where the size of most jobs ranges from 1 to 64. This is also true for the weekend jobs. It is interesting to see there are peaks in both graphs (8 PM for weekday and 2 PM for weekend jobs). The peak in the weekday curve is due to a relatively small number of users who want to run large jobs overnight as Fig. 2 indicates. This increases the average job size considerably. The peak in the weekend curve suggests that users tend to submit large jobs in weekend afternoon.

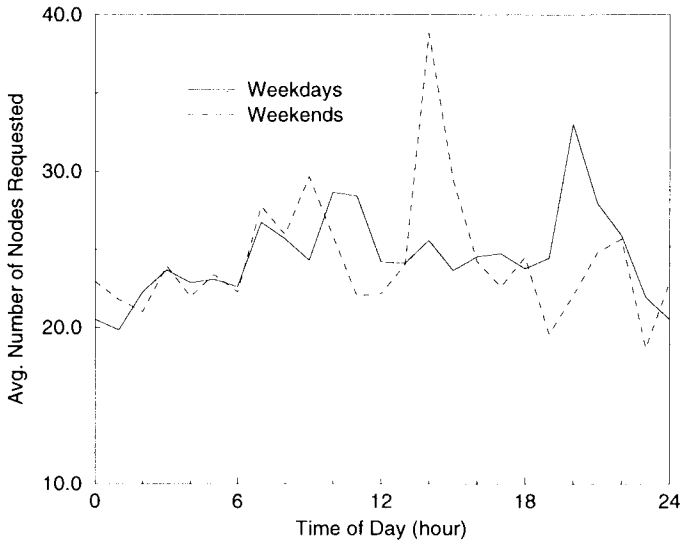

Figure 8: Average number of nodes used by jobs as a function of time of the day.

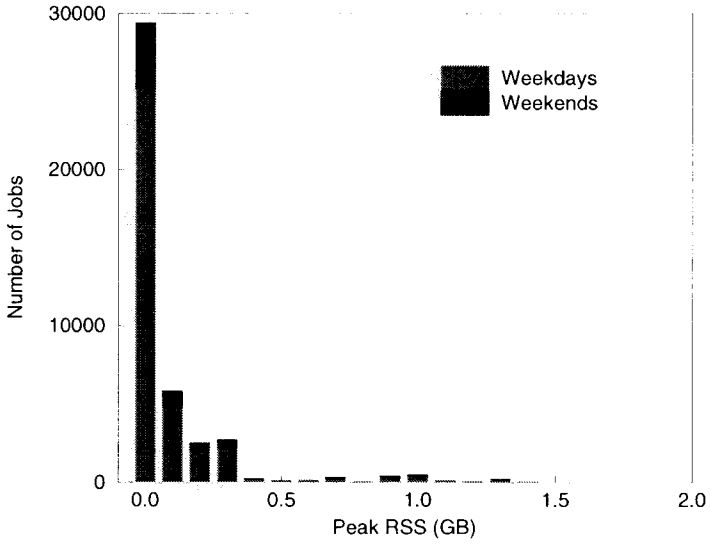

Figure 9: Peak RSS distributions for weekday and weekend jobs.

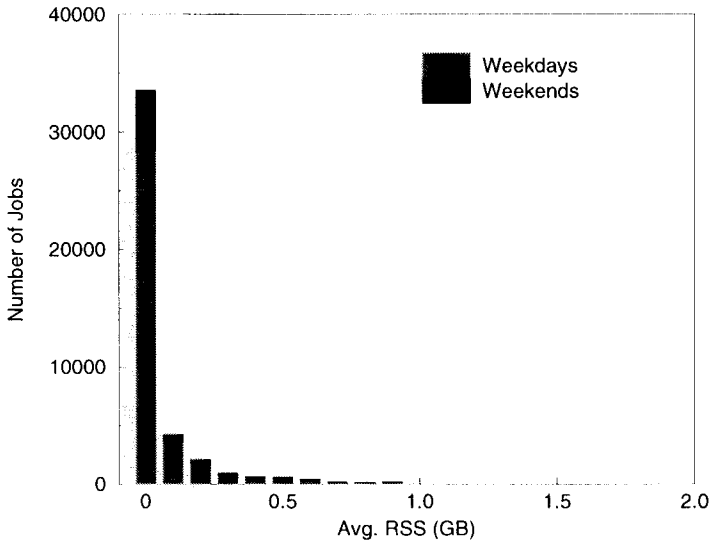

Figure 10: Average RSS distributions for weekday and weekend jobs.

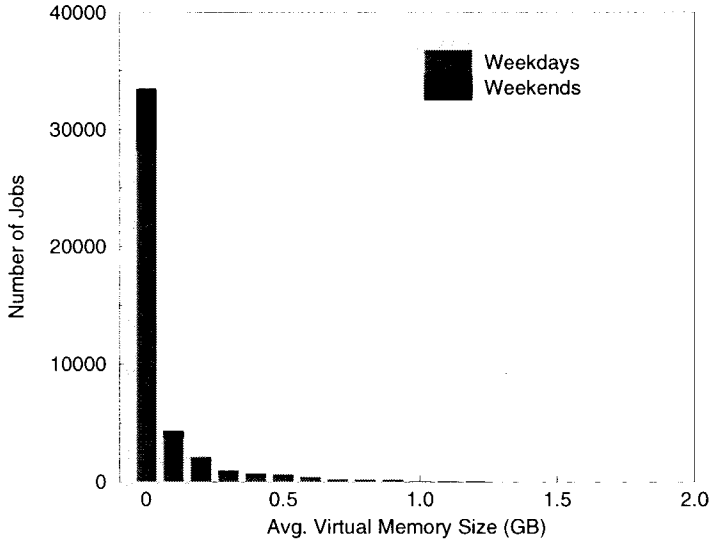

Figure 11: Average virtual memory size distributions for weekday and weekend jobs.

Memory usage	Weekdays jobs (GB)	Weekend jobs (GB)
Peak RSS	0.14	0.17
Mean Avg. RSS	0.08	0.09
Mean Avg. virtual memory size	0.08	0.09

Table 2:

Resources	Correlation coefficients
Node count	0.006897
Peak RSS	0.034837
Avg. RSS	0.049698
Avg. virtual memory size	0.049352

Table 3: Correlation coefficients for wall clock time of and various resource requirements of jobs.

Figures 9 to 11 show the memory usage characteristics of jobs. They present peak Resident Set Size (RSS), average RSS, and average virtual memory size distributions, respectively. All values report memory use on a per node basis. Average memory values are computed by dividing the memory integral of the job by its execution time then dividing by its node count. To the best of our knowledge, this is the first study that reports the memory usage characteristics of jobs on a large-scale IBM SP2 machine or any other ASCI platforms. These figures clearly indicate that most jobs demand a very small amount of memory. Of those jobs we observed, 94% use peak RSS of less than 0.5 GB, and 78% average less than 0.1 GB of RSS and virtual memory. Both figures 10 and 11 look very similar because the address space of most jobs can be fit into the physical memory of 1.5 GB. Our findings concur with a previous report, where the memory usage of jobs on an CM-5 machine at the Los Alamos National Laboratory (LANL) has been analyzed [9]. Currently, LLNL employs the gang scheduling of jobs for better responsiveness. One major concern is very poor paging performance of IBM AIX operating system. With such poor paging performance, a memory thrashing can easily results in a significant decrease in system performance. Fortunately, the small memory usage of jobs implies that we may time-share several jobs without being hit by high paging overhead. Table 2 reports the average memory size of weekday and weekend jobs for the three types of memory usage measures.

We also have conducted a correlation analysis to see if there exist close ties between execution time and resource requirements of jobs. We have considered wall clock time and CPU time consumed by jobs in this analysis. Our motivation was to find correlation between resource demand and execution time and to use the information in job scheduling. We expected to see certain degree of correlations, especially between the node requirement and the execution time, but surprisingly found none. Tables 3 and 4 summarize the results from the analysis.

4 System Utilization

In this section, we report various system utilization characteristics of jobs. We represent the system utilization in a time-space measurement unit, called Node-Week, which is defined as product of the

Resources	Correlation coefficients
Node count	-0.013919
Peak RSS	-0.042889
Avg. RSS	0.047282
Avg. virtual memory size	0.046641

Table 4: Correlation coefficients for CPU time of and various resource requirements of jobs.

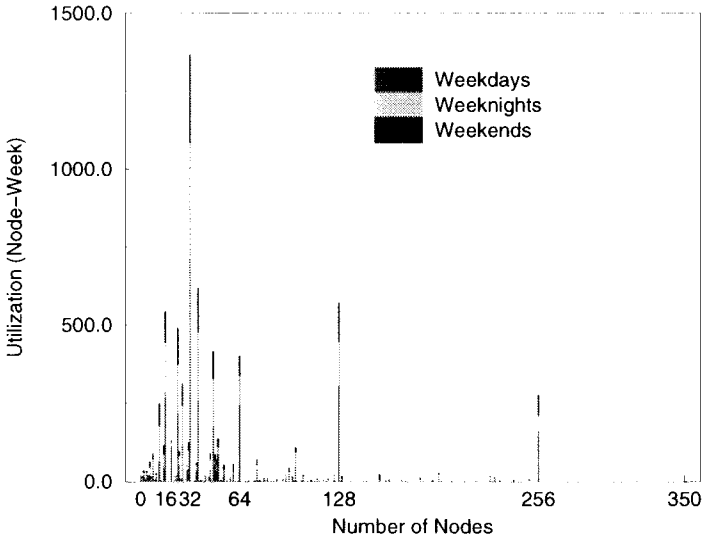

Figure 12: System utilization by jobs of different jobs.

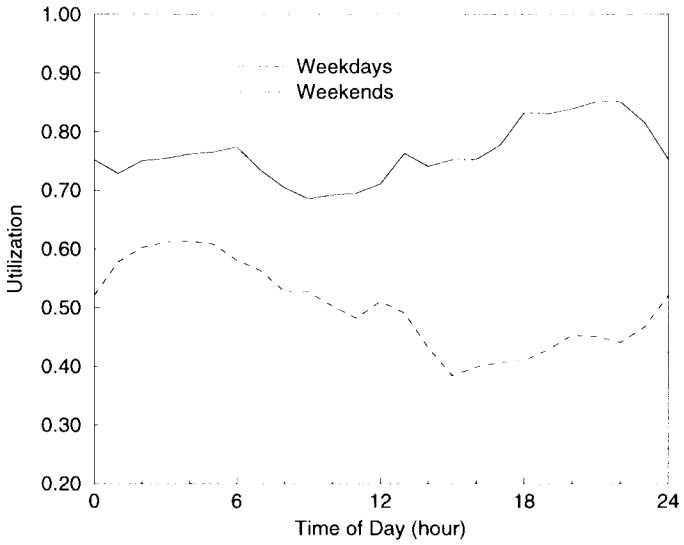

Figure 13: System utilization as a function of time of the day.

number of nodes and the time in weeks. First, Fig. 12 presents the system utilization by jobs of different size. What is noticeable in Fig. 12 is its discordance with Fig. 7. Comparing two graphs, we can see that the system utilization is not proportional to the node requirement of jobs. Although 57% of jobs require 16 or less nodes, this group of jobs constitutes only 17% of total system utilization. It is the larger jobs (with 32 or more nodes), constituting only 29% of the entire population, that utilize most system resources (67% of total system utilization).

Fig. 13 depicts the system utilization as a function of time of the day. System utilization reported is as a percentage of total system capacity and no allowance is made for down-time, which accounts for another 2.5 percent of system capacity. The system utilization ranges from 69% to 85% and 39% to 61% during the weekdays and weekends, respectively. The overall system utilization is 69%. The system utilization is at its lowest (about 70%) at 8 AM during weekdays. This is because there are not many users at this time of the day and the DPCS stops initiating larger or long-running jobs and switches to the daytime scheduling mode. As more users arrive to the system, however, system utilization increases. The system utilization of weekday jobs continues to increase even after 5 PM and reaches its peak (85 %) at 9 PM. This high system utilization during evening can be attributed to those jobs submitted at the end of the day and those jobs submitted earlier but deferred until evening due to their excessive resource requirement. As soon as the smaller jobs submitted at the end of the day complete, the system utilization starts to decrease (after 10 PM) to 75% and remains almost unchanged till 6 AM, when long-running jobs in the queue start to be deferred until the next evening. The system utilization during the weekends is considerably lower than that during the weekdays, mainly due to the dearth of work. The plateau in early hours of the weekend curve is probably due to the long-running jobs submitted on Friday afternoon.

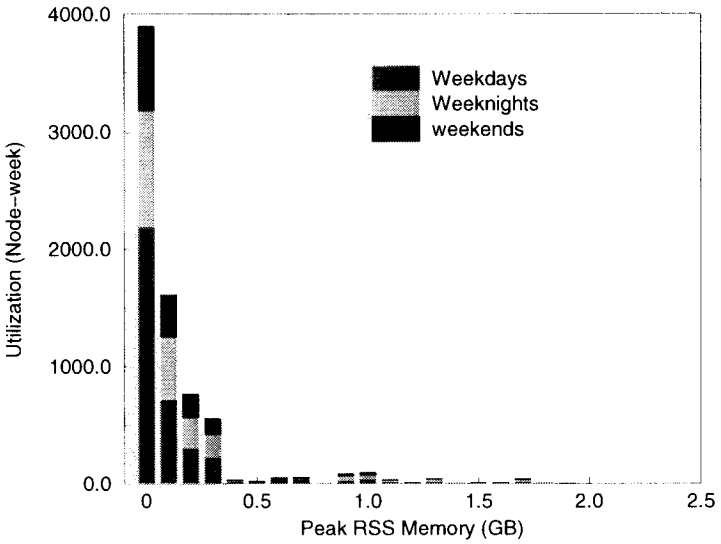

Figure 14: System utilization by jobs of different peak RSS.

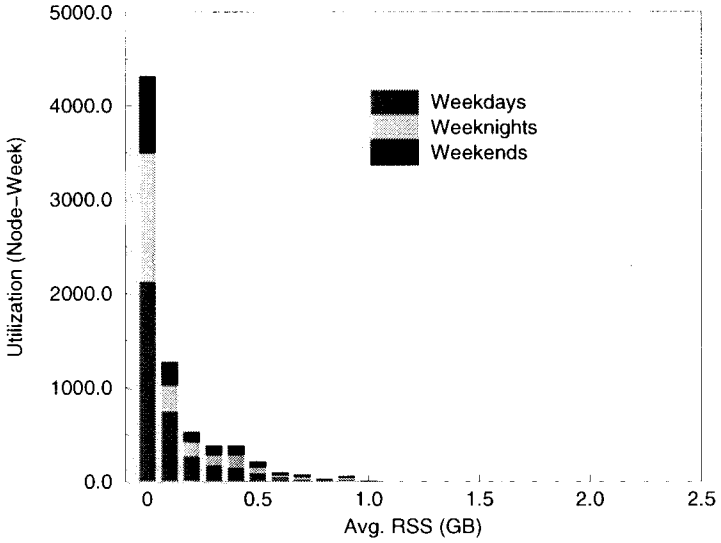

Figure 15: System utilization by jobs of different average RSS.

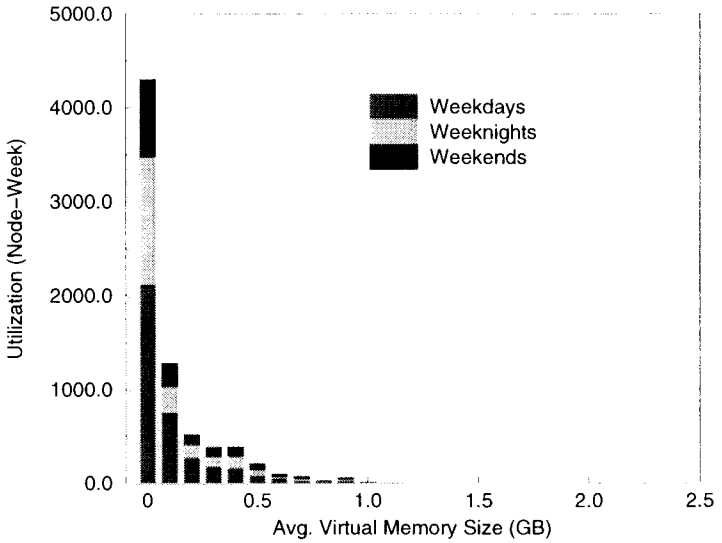

Figure 16: System utilization by jobs of different average virtual memory sizes.

Figures 14 to 16 show the system utilization by groups of jobs with different memory requirement. Unlike Fig. 12, we have observed that the system utilization is pretty consistent with the memory usage distributions reported in figures 9 through 11. Here, the utilization by jobs with small memory usage constitutes a large portion of the total system utilization. Fig. 14 indicates that the utilization by those jobs with peak RSS of less than 0.2 GB constitutes 75% of entire system utilization. Similarly, the utilization by jobs with average RSS and virtual memory size of 0.2 GB or less constitutes 76% of the total system utilization.

Fig. 17 shows the CPU efficiency characteristics of weekday, weeknight, and weekend jobs. The CPU efficiency of a job is defined as the ratio of CPU time used to the total CPU time allocated. Fig. 17 reveals that 47% of the jobs exhibit the CPU efficiency of 90% or higher implying the tasks of most jobs fully utilize all CPU time on all nodes allocated to it. The graph also shows that 14% of jobs have the CPU efficiency of less than 10%.

5 Concluding Remarks

We have analyzed the job traces collected from ASCI Blue-Pacific, a 336-node IBM SP2 machine, and reported the characteristics of the workload imposed on the machine in this paper. This paper provides comprehensive information on the characteristics of a typical workload on large-scale parallel computers in a scientific computing environment. With the wealth of information, this report can help the system managers of large-scale scientific parallel computers design new and better systems and researchers who conduct performance studies on large-scale parallel machines for scientific computation.

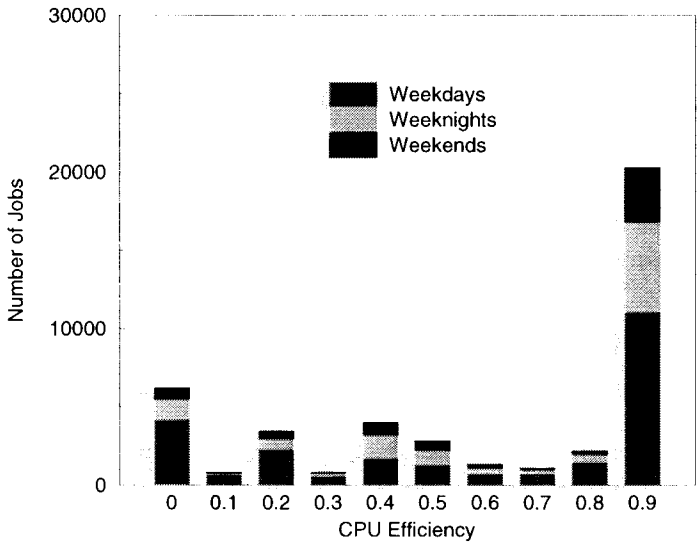

Figure 17: CPU efficiency distributions of jobs.

Our results show that jobs have very short mean execution time and mean inter-arrival time. In addition, we have found that mean job waiting time is quite long. About 50% of jobs have to wait in the queue for more than an hour before initiation. Another important finding is that most jobs have very small node and memory requirements. The small memory requirement, in particular, favors multiprogramming of parallel jobs. Improved infrastructure to prevent larger memory jobs from sharing nodes will permit us to more fully exploit the gang scheduler and may decrease the job wait times. Finally, the results from this study reveal that most resources are consumed by moderate sized jobs requiring 32 or more nodes, although they constitute relatively small fraction of total jobs observed. This concurs with what has been reported previously in the literature [15].

References

[1] R. H. Arpaci, A. C. Dusseau, A. Vahdat, L. T. Liu, T. E. Anderson, and D. A. Patterson. The Interaction of Parallel and Sequential Workloads on a Network of Workstations. In *Proc. ACM SIGMETRICS 1995 Conf. on Measurement and Modeling of Computer Ssystems*, pages 267–278, May 1995.

[2] ASCI Blue Mountain. `http://www.lanl.gov/asci/bluemtn/bluemtn.html`.

[3] ASCI Blue Pacific. `http://www.llnl.gov/platforms/bluepac`.

[4] ASCI Red. `http://www.sandia.gov/ASCI/Red`.

[5] Distribute Production Control System. `http://www.llnl.gov/liv_comp/DPCS/DPCS_home.html`.

[6] A. B. Downey. A Parallel Workload Model and Its Implications for Processor Allocation. Technical Report CSD-96-922, Computer Science Division, University of California, Berkeley, Nov. 1996.

[7] A. B. Downey and D. G. Feitelson. The Elusive Goal of Workload Characterization. *Performance Evaluation Review*, pages 14–29, Mar. 1999.

[8] D. Feitelson. Packing Scheme for Gang Scheduling. In *Proc. IPPS'96 Workshop on Job Scheduling Strategies for Parallel Processing*, pages 89–110, Apr. 1996.

[9] D. G. Feitelson. Memory Usage in the LANL CM-5 Workload. In *Proc. IPPS'97 Workshop on Job Scheduling Strategies for Parallel Processing*, pages 78–94, 1997.

[10] D. G. Feitelson and B. Nitzberg. Job Characteristics of a Production Parallel Scientific Workload on the NASA Ames iPSC/860. In *IPPS'96 Workshop on Job Scheduling Strategies for Parallel Processing, Vol. 1162 of Lecture Notes in Computer Science*, pages 337–360. Springer-Verlag, Apr. 1996.

[11] G. Haring and G. Kotsis. Workload Modeling for Parallel Processing Ssystems. In *Proc. International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, pages 8–12, 1995.

[12] S. Hotovy. Workload Evaluation on the Cornell Theory Center IBM SP2. In *IPPS'96 Workshop on Job Scheduling Strategies for Parallel Processing, Vol. 1162 of Lecture Notes in Computer Science*. Springer-Verlag, Apr. 1996.

[13] J. E. Moreira et al. A Gang-Scheduling System for ASCI Blue-Pacific. In *Proc. Distributed Computing and Metacomputing (DCM) Workshop, High-Performance Computing and Networking '99*, Apr. 1999.

[14] J. Jann, P. Pattnaik, H. Franke, F. Wang, J. Skovira, and J. Riodan. Modeling of Workload in MPPs. In *IPPS'97 Workshop on Job Scheduling Strategies for Parallel Processing, Vol. 1291 of Lecture Notes in Computer Science*, pages 95–116. Springer-Verlag, Apr. 1997.

[15] V. Lo, J. Mache, and K. Windsch. A Comparative Study of Real Workload Traces and Synthetic Workload Models for Parallel Job Scheduling. In *Proc. IPPS'98 Workshop on Job Scheduling Strategies for Parallel Processing*, pages 1–16, Mar. 1998.

[16] E. W. Parsons and K. C. Sevcik. Multiprocessor Scheduling for High-Variability Service Time Distributions. In *Proc. IPPS'95 Workshop on Job Scheduling Strategies for Parallel Processing*, pages 76–88, Apr. 1995.

[17] J. Skovira, W. Chan, H. Zhou, and D. Lifka. The Easy-LoadLeveler API Project. In *IPPS'96 Workshop on Job Scheduling Strategies for Parallel Processing, Vol. 1162 of Lecture Notes in Computer Science*, pages 41–47. Springer-Verlag, Apr. 1996.

[18] T. Suzuoka, J. Subhlok, and T. Gross. Evaluating Job Scheduling Techniques for Highly Parallel Computers. Technical Report CMU-CS-95-149, School of Computer Science, Carbegie Mellon University, 1995.

[19] Top 500 Supercomputer Sites. `http://www.netlib.org/benchmark/top500.html`.

[20] K. Windisch, V. Lo, D. Feitelson, B. Nitzberg, and R. Moore. A Comparison of Workload Traces from Two Production Parallel Machines. In *Proc. Sixth Symposium on the Frontiers of Massively Parallel Computing*, pages 319–326, Oct. 1996.

University of California
Lawrence Livermore National Laboratory
Technical Information Department
Livermore, CA 94551