Library Express Request

NUI

Requested: 07/15/2002

38590

Requestor: BENT, JOHN

Email: johnbent@cs.wisc.edu

Classification: GRAD

Address:

Recipient:

Search Until: 10/15/2002

Fill only if free

Citation - Request type: journal

Conference paper:

Matt W. Mutka, Miron Livny: Profiling Workstations' Available Capacity for Remote Execution. pp.529-544

In ISBN 0444703470,

"Performance '87:Computer performance modelling, measurement and evaluation, 12th IFIP WG 7.3 International Symposium"

Brussels, Belgium, 7-9 December 1987

ISSN/ISBN

OCLC # 17677695

Call number

DNO

Updated

Notified

Organized by

Université Libre de Bruxelles
and
Philips Research Laboratory, Brussels

PERFORMANCE '87

Proceedings of the 12th IFIP WG 7.3 International Symposium on Computer Performance Modelling, Measurement and Evaluation
Brussels, Belgium, 7–9 December 1987

edited by

P.-J. COURTOIS
Philips Research Laboratory
Avenue Van Becelaere, 2
Brussels, Belgium

and

G. LATOUCHE
Université Libre de Bruxelles
Campus Plaine CP 212
Boulevard du Triomphe
Brussels, Belgium

1988

NORTH-HOLLAND
AMSTERDAM · NEW YORK · OXFORD · TOKYO

[16] M. Carey, M. Livny, and H. Lu, "Dynamic task allocation in a distributed database system," Proc. 5th International Conference on Distributed Computing Systems, Denver, May 1985.

[17] S. Zhou, "An Experimental Assessment of Resource Queue Length as Load Indices," Proc. Winter USENIX Conference, Washington, D.C., pp. 73-82, January 21-24, 1987.

[18] S. Zhou, "Predicting Job Resource Demands: a Case Study in Berkeley UNIX," in preparation.

[19] W. Joy, "An Introduction to the C Shell," Computer Science Division, University of California, Berkeley, November 1980.

[20] W. Joy, E. Cooper, R. Fabry, S. Leffler, K. McKusick, and D. Mosher, "4.2BSD System Manual," Computer Systems Research Group, University of California, Berkeley, July 1983.

[21] K. McKusick, M. Karels, and S. Leffler, "Performance Improvements and Functional Enhancements in 4.3 BSD," Proc. Summer USENIX Conference, June 1985, Portland, OR, pp. 519-531.

[22] B. Bershad, "Load Balancing with Maitre d'," Tech Report, UCB/CSD 85/276, Computer Science Division, University of California, Berkeley, December 1985.

PERFORMANCE '87
P.-J. Courtois and G. Latouche (editors)

Profiling Workstations' Available Capacity For Remote Execution

Matt W. Mutka and Miron Livny

Department of Computer Sciences
University of Wisconsin
Madison, WI 53706

ABSTRACT

Powerful workstations have become widely available as sources of computing cycles. These stations are brought together into networks for the distribution of mail and the sharing of servers. The networks allow for the sharing of computing capacity among the stations. In order for capacity sharing to be effective, there must be algorithms that allocate the available capacity and long periods when owners do not use their stations. To understand the profile of station availability, we analyzed the usage patterns of a group of workstations. The workstations were available approximately 70% of the time observed. Large capacities were steadily available on an hour to hour, day to day, and month to month basis. These capacities were available not only during the evening hours and on weekends, but during the busiest times of normal working hours. A stochastic model was developed based on an analysis of the relative frequency distribution and the correlation of available and non-available interval lengths. A 3-stage hyperexponential cumulative distribution has been fitted to the observed cumulative relative frequency of available periods. The fitted distribution closely matches the observed relative frequency distribution. This stochastic model is important as an artificial workload generator for the performance evaluation of remote capacity sharing strategies of a network of workstations. The model assists in the design of resource management algorithms that take advantage of knowing the characteristics of the usage patterns.

1. Introduction

Many users now have workstations on their desks to serve their computing needs. These stations are powerful tools that provide users available computing capacity when needed. The stations can execute computationally intensive programs that formerly were run on mainframes. When supplied with a workstation, the user owns the resource. Owners can control access to these stations and configure them with software to suit their individual needs. To improve the quality of provided service, stations are brought together into networks to allow sharing of servers, and the distribution of mail. The network of stations provide flexibility and availability of computing service. This is an improvement to the previous times when during peak working times of the day many users competed for computing capacity of mainframe computers. The total computing capacities of these workstation networks can be huge. As an example, in addition to other computers, the University Of Wisconsin Computer Sciences department has over 100 DEC MicroVAXII® workstations. Each workstation has a single chip processor which has roughly the capacity of a VAX 11/780 [1]. This capacity is a little less than 2 MIPS [2]. This means the department has, from workstations alone, about 200 MIPS at its disposal.

† This research was supported in part by the National Science Foundation under grant MCS81-05904 and by the Wisconsin Alumni Research Foundation.

® VAX 11/780 and MicroVAXII are trademarks of Digital Equipment Corporation.

An additional improvement in the quality of service can be obtained if the network of stations allows the sharing of computing capacity. When a user needs more capacity than what his/her workstation can supply, available capacity can be allocated from other stations. Networks where computing capacity is shared among stations can allow a single user to expand the capacity of his/her station to that of the entire network. The capacity of the user's station is expanded by remotely executing background jobs. We have observed in our department a large number of background jobs that could exploit capacity sharing if the opportunity existed. These jobs ran for long periods of time with little interaction from users. As an example, we have observed a user who maintained a queue of 20-30 background jobs awaiting execution for several months. Each job executed about 1 hour on a MicroVAXII workstation. Another user maintained a queue of more than a dozen jobs where each job consumed weeks of cpu time. One of the jobs was observed to consume 1 month of cpu time on a MicroVAXII. We call networks that allow the remote execution of background jobs *LOcal COmputing capacity eXpanded (LOCOX)* networks. LOCOX networks can contain not only workstations, but other processors (which we call *processor bank nodes*) that have no specific owners but serve exclusively as sources of extra computing cycles. Figure 1 illustrates this environment.

Computing capacity is available to share because stations are not used by their owners continuously. A number of studies have addressed the problem of allocating remote capacity in a distributed environment. These include papers on the V-Kernel [3], Process Server [4], and the NEST research project [5]. For all of these system, the authors recognized that workstations often are available for remote cycles. The authors of the paper on the V-Kernel stated that many workstations are available even during the busiest times of the day. However, these papers did not profile the availability of the workstations. The focus of these studies was the design of facilities to provide remote execution of jobs on workstations. To evaluate approaches of managing capacity sharing, we need to understand the characteristics of workstation usage. This means we must properly profile the *workload* of workstation activity. A major component of any study is the workload used. No system evaluation study can avoid confronting the problem of modeling a workload [6]. If inappropriate workloads models are chosen when studying scheduling policies, then inappropriate results can occur. Performance indices to be evaluated in a study are critically dependent on the workload processed by the system [7]. This paper explores the patterns of activity which owners have with their workstations, and characterizes the extent which capacity is available for sharing. A model of the workstation utilization is developed as a stochastic process. Our work enables others the opportunity to use realistic workloads when evaluating capacity sharing policies. Also, when usage patterns are understood, algorithms that take advantage of the patterns can be designed. This work enables one to estimate the expected capacity available from a workstation network, and therefore to predict the turnaround time for background jobs that execute remotely.

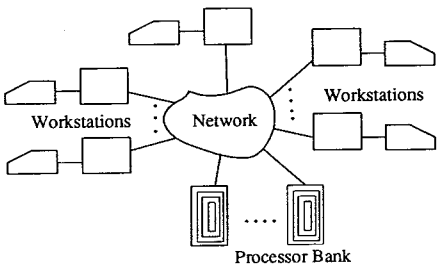

Figure 1.
LOCOX Network

To obtain the workload usage profile, we monitored 11 stations at our department for a period of 5 months. In order to model the workstations as a stochastic process, workstation usage patterns were analyzed to provide insight on how workstations are used, and how much capacity is available. The distribution of workstations' available and non-available intervals has been profiled. The correlation between available and non-available periods has been characterized. We looked at how the availability of a station changes from hour to hour, day to day, and month to month.

Many studies have used exponential distributions for the interarrival times and the service demand of jobs [8-12]. A number of researchers have observed that the cpu requirements of jobs on multiuser computers are not exponential. However, these studies profiled processes and not users. One example is of Leland and Ott [13] who present a study from the observation of 9.5 million Unix® processes and showed that probability distribution of cpu time used is far from exponential. The distribution of cpu capacity used by processes was approximately $1-rx^{-c}$, where $1.05<c<1.25$. Another example is presented by Zhou [14] who traced Unix processes on a VAX 11/780 computer. The trace included the arrival patterns and cpu demands of the jobs. The arrival and cpu demands were observed to have a large coefficient of variation.

Our work profiles the workload of a network of workstations and not of a multiuser computer. It gives the characteristics of the availability of workstation for remote cycles. It is not a model of individual job's response times or cpu utilization. We conservatively profile how workstations can be used by other users without compromising the owner's control of the workstation. The workstation owner maintains control and only allows cycles to be used by others if the user load of the station is lower than a very small threshold. We present the behavior of the user as a stochastic process with hyperexponential state time distributions.

In section 2 we describe the technique used for acquiring data on workstation usage. An analysis of this data is presented in section 3. Section 4 includes a description of a family of stochastic models based on the analysis of section 3. In section 5 we describe how the workload model can assist the design of resource allocation algorithms. Conclusions and a description of continuing work are laid out in section 6.

2. Technique Of Acquiring Data

We have monitored the usage patterns of 11 DEC MicroVAXII workstations running under Berkeley Unix 4.2BSD over a period of five months from the first of September to the end of January. The stations observed were owned by a variety of users. They were 6 workstations owned by faculty, 3 by systems programmers, and 2 by graduate students. Two additional stations used by systems programmers that were only available for monitoring from September through November have their traces included in the results reported.

We have obtained the profile of *available* and *non-available* periods of each of the workstations. An unavailable period, *NA*, occurs when a workstation is being used, or was recently used by its owner so that the average user cpu usage is above a threshold (one-fourth of one percent) or was above the threshold within the previous 5 minutes. The average cpu usage follows the method the Unix operating system uses for the calculation of user load in a similar way as the *ps(1)* [15] command (process status) computes the cpu utilization of user processes. This load is a decaying average that includes only the user processes, and not the system processes. The value of the threshold is chosen so that activities resulting from programs such as time of day clocks or graphical Representations of system load do not generate user loads that arise above the threshold. An available period, *AV*, occurs whenever the workstation is not in the *NA* state.

The workstation usage patterns were obtained by having a monitoring program [16] executing on each workstation. The monitor on each station executes as a system job and does not affect the user load. When the workstation is in the *NA* state, the monitor on each workstation looks at the user's load every minute. If the user's load is below the low threshold for at least 5 minutes, the workstation's state becomes *AV*. During this time the workstation's monitor will have its

® Unix is a trademark of AT&T Bell Laboratories.

"screen saver" enabled. The monitor looks at the user's load every 30 seconds when the workstation is in the *AV* state. Any user activity, even a single stroke at the keyboard or mouse, will cause the "screen saver" to be disabled and all user windows on the workstation's screen to be redrawn. This activity brings the user load above the threshold, and causes the state to become *AV*. If no further activity occurs, approximately seven minutes pass before the station's state changes to *AV*. This is because it takes the user load average 2-3 minutes to drop below the threshold, and an imposed waiting time of 5 minutes. The waiting period is imposed so that users who stop working only temporarily are not disturbed by the "screen saver" reappearing as soon as they are ready to type another command. The waiting time is adjustable, but it has been observed that it is a good value to choose without causing an annoyance to users [17]. This conservatively decides whether a station should be a target for remote cycles. Stations are idle much more than what appears in the *AV* state. The user load with the imposed waiting time is used as a means of detecting availability because the station should not be considered a source of remote cycles if an owner is merely doing some work, thinking for a minute, and then doing some more work. Otherwise a station would be a source of remote cycles as soon as the owner stopped momentarily. The workstation's owner would suffer from the effect of swapping in and out of his/her processes, and the starting and stopping activities of the remote processes.

The monitor keeps records of the workstation's last 100 state changes. Every ten hours, one of the workstations gathers the records from all other workstations. This station maintains a log for the entire LOCOX network. When records are gathered from a workstation, the user load is not affected. This is because the monitor on the workstation which sends the records executes as a system job. Since *NA* states last at least 7 minutes, at most 9 state changes can occur within an hour. Therefore, we will not lose records due to our sampling rate. Some records were lost because a few stations had their monitors disabled for a short while, and then enabled later. This happened rarely and has little significance on our traces. During intervals when station monitors were disabled, the time was marked as an *NA* interval.

3. Analysis Of Data

We want to represent the usage patterns of the workstations as a stochastic process so it can be used to model workstation availability. Such a model can be used in performance evaluation studies of LOCOX networks, or to support design decisions of resource management algorithms that take advantage of knowing the properties of the usage pattern. With the workload modeled as a stochastic process, we do not need to use traces to drive simulation models.

In order to define a stochastic process we have to know the distributions of *AV* and *NA* state lengths, and how state lengths are correlated. The data gathered from each workstation was analyzed to determine the relative frequency distributions of the *AV* and *NA* state lengths. Individuals stations were analyzed and the characteristics of their distributions are reported. We show how the length of *AV* intervals was correlated to the length of subsequent *NA* intervals, and vice versa. We report how the availability of remote capacity varied from hour to hour, day to day, and month to month.

3.1. Distribution Of Usage Patterns

A graph of the cumulative relative frequency of the *AV* states for all of the stations during the entire time monitored is shown in figure 2 (the solid line). For each time *t* on the horizontal axis, the corresponding *percentage* on the vertical axis is the percentage of *AV* intervals that were less than *t+1* minutes. The figure shows that there were many short *AV* intervals of less than a few minutes and many very long intervals of an hour or longer. The solid line curve in figure 3 shows the cumulative relative frequency of *NA* state lengths. As in figure 2, for each time *t* on the horizontal axis of figure 3, the corresponding *percentage* on the vertical axis was the percentage of *NA* intervals less than *t+1* minutes.

When we look at figures 2 and 3, we notice that there were many short intervals for both the *AV* and *NA* graphs. This leads us to believe that a significant component in the relative frequency was from short intervals. However, there were more long intervals than what one would expect to see in an exponential distribution. The graphs show that the percentage of intervals larger than

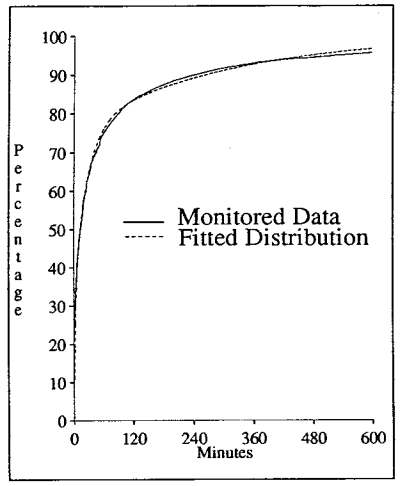

Figure 2.
Distribution Of *AV*
State Durations (all stations)

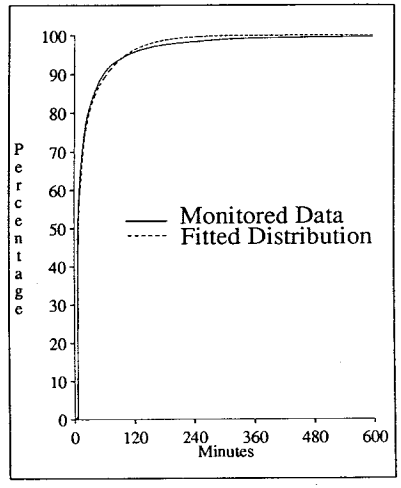

Figure 3.
Distribution Of *NA*
State Durations (all stations)

one hour is greater for *AV* intervals than for *NA* intervals. In figure 2 we see that there was a large number of *AV* intervals beyond 5 hours long (300 minutes). This leads us to the belief that the *AV* periods were dominated by three types of periods: short, medium, and long. Short intervals occurred when users did some work, and then stopped to think for a while before resuming the use of their workstations. Medium intervals were the result of users leaving their desk for short intervals, or stopping to do other work during the day. Since users left their offices in the evening and weekends, scheduled long meetings, and taught or attended classes, long available intervals occurred. For the *NA* periods we have identified two types of periods: short and long. The short component is the result of frequent short activities. The user typed a few simple commands and then stopped to do something else. The user might have had some jobs that executed for short intervals even when he/she was not at the station. These short jobs contributed to the user load which briefly made the station unavailable. The long components are the result of prolonged activity by the user. Long intervals occurred if the user had long running jobs to execute, which continued to execute after the user left the station. With this intuition, we seek to match a distribution to each of the relative frequencies observed. Figures 2 and 3 appear to have exponential components because they increase similarly to an exponential distribution and have long tails. A *mixture distribution* of exponentials seems to be a good candidate to fit the observed data [18]. This distribution is sometimes referred to as a *k-stage hyperexponential distributions*, when the distribution has *k* components [19]. The *k*-stage hyperexponential distribution function of a random variable T is defined as

$$F(T)=\sum_{i=1}^{i=k}\alpha_i F_i(t), \text{ where } F_i(t)=1-e^{-\lambda_i t}, \text{ and } \sum_{i=1}^{i=k}\alpha_i=1. \qquad (1)$$

We look for a k-stage distribution that fits our monitored data well and has a small number of components. Each component *i* of a *k*-stage distribution introduces two parameters that must be adjusted: λ_i and α_i. On one hand, the more components introduced the better the fit is, but on the other hand it is more complex to assign values to a large number of parameters. It is

important to capture the characteristics of a relative frequency distribution with as few components as possible. For the *AV* relative frequency distribution, a good match was achieved by using a 3-stage hyperexponential distribution. The stages represent the small, medium, and large *AV* intervals. The components were assigned the expected values of 3, 25, and 300 minutes. Weights were assigned by using a least-squares fit [20] for these components to obtain the following 3-stage distribution

$$F(T_A)=0.32(1-e^{\frac{-t}{3}})+0.44(1-e^{\frac{-t}{25}})+0.24(1-e^{\frac{-t}{300}}). \qquad (2)$$

The small component contributes approximately 1/3 of the distribution. The larger components account for approximately 2/3 of the distribution of which a little less than 2/3 is the medium component, and the remainder is the largest component.

Figure 4 shows the match of the cumulative distribution to the monitored traces for *AV* intervals smaller than 60 minutes. The curve derived analytically for figure 4 was generated from equation 2. The distribution of intervals that were less than 60 minutes is an important portion of the distribution to match. This is the region one must study to determine whether it is worthwhile to use workstations as a source for remote execution. We see that the match is excellent between the two curves. Figure 2 shows the match for *AV* intervals that were up to 600 minutes in length. The overall difference between the fitted distribution and the relative frequency distribution is very small.

Less complexity is introduced when matching the *NA* intervals because its relative frequency distribution has fewer long intervals. A good match for the *NA* intervals, T_{NA}, is obtained if we use a 2-stage hyperexponential distribution. The two components have the expected values of 7 and 55 minutes. Since each *NA* interval lasted at least 7 minutes, the distribution is modified so that the probability that an interval is less than 7 minutes is zero. The distribution of *NA* intervals is defined as

$$F(T_{NA})=\begin{cases} 0.68(1-e^{\frac{-t}{7}})+0.32(1-e^{\frac{-t}{55}}), & \text{if } t\geq 7 \\ 0, & \text{if } 0\leq t<7. \end{cases} \qquad (3)$$

Figure 5 shows the match between the cumulative distributions of T_{NA} and the monitored relative frequency for *NA* intervals less than 60 minutes. The curve derived analytically for figure 5 is generated from equation 3. The match in figure 5 is very good. Figure 3 shows the match for *NA* intervals that were up to 600 minutes in length.

Beyond the first five minutes, the greatest amount the fitted curve in figure 2 deviates from the curve of the monitored data at any point is approximately 2.5% from below and 1.0% from above the monitored data. For figure 3, the greatest the fitted curve deviates beyond the seven minute interval from the observed data is 1.0% from below and 1.5% from above. By using the Kolmogorov-Smirnov test (KS test) for curve matching [21], we calculate the likeliness that our observed data could be generated from a random sequence of our fitted distribution. If random sequences were generated from the distribution of equation 2, it would have approximately an 45% chance of deviating from below as much as our monitored data, and an 85% chance from above. Random sequences generated from a distribution like equation 3 would have approximately an 85% chance of deviating from below, and an 80% chance from above as much as our observed data. These ranges mean that random sequences with distributions of equations 2 and 3 are likely to deviate as much as our monitored data. This gives added confidence in using equations 2 and 3 as matches for our observed data. We believe that equations 2 and 3 can serve as means of artificially describing *AV* and *NA* interval characteristics for studies involving remote allocation strategies of workstations.

Each individual workstation had its own usage patterns. The overall results depend on how individual workstation usage patterns differ. Most of the workstations can be characterized by their own mixture distributions as defined by equation 1. They can be characterized by three exponential components for the *AV* intervals, and 2 exponential components for the *NA* intervals.

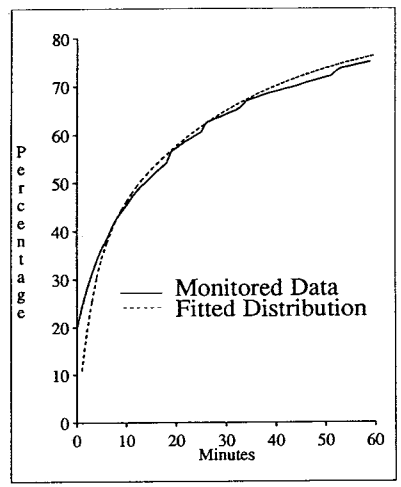

Figure 4.
Distribution Of *AV*
State Durations (all stations)

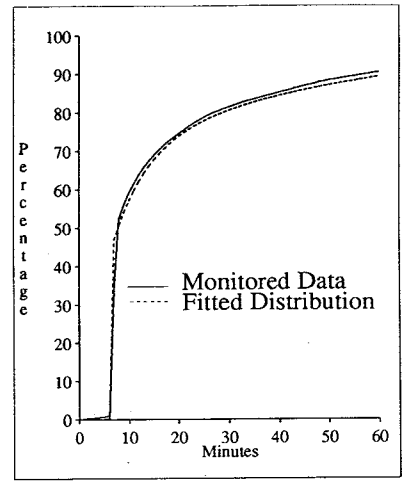

Figure 5.
Distribution Of *NA*
State Durations (all stations)

A good fit was obtained for each of these stations taken individually. Although the magnitude and contribution of each component varied, each station's *AV* distribution had a short component with expected values that ranged from 3-5 minutes, a medium component that ranged from 25-60 minutes, and a large component that ranged from 150-600 minutes. For the *NA* components, the stations had short components ranging from 7-11 minutes, and long components from 45-95 minutes. Two stations did not follow these characteristics. They were stations 5 and 12. They had an unusual large number of short *AV* and *NA* intervals. If our analysis did not include stations 5 and 12, the overall relative frequency distributions would have fewer short intervals. Nevertheless, we would still need a three-component hyperexponential distribution to match the AV intervals and a two-component distribution to match the *NA* intervals. By doing the analysis without these stations, we get extremely good matches of the fitted distributions to the relative frequencies by using the same exponential components with only a small difference in the weighting of each component.

3.2. Correlation Of Available And Non-Available States

When we build an artificial workload generator, information beyond the distribution of the states is needed. Given that distributions that closely match the observed distributions of the two types of intervals can be generated, we need to know how the length of *AV* and *NA* intervals correlate. What can the length of the current interval tell us about the length of the next interval?

Pairs of *NA* and *AV* periods were analyzed to determine whether such a correlation exists. We looked at the traces and labeled *AV* intervals as short, medium, or long samples. All samples that were less than 9 minutes were labeled short samples. (Ninety-five percent of intervals of an exponential distribution with mean of 3 minutes are less than 9 minutes.) Intervals greater than 9 minutes and less than 75 minutes (which is the 95 percentile of an exponential of the medium distribution with mean 35 minutes) were called medium samples. The remaining *AV* samples were labeled large samples. We similarly classified *NA* samples less than 21 minutes (the 95 percentile of an exponential distribution with mean of 7 minutes) to be short samples, and the remaining

samples long.

With this labeling method, we show a conditional probability graph in figure 6. It shows that 41% of all *AV* samples were short, 36% were medium, and 23% were long. Equation 2 weighted the components as 32% short, 44% medium, and 24% long. Our labeling gave a greater percentage of short intervals than what appears in equation 2. This was expected since some intervals from the medium and long components are less than 9 minutes in length and therefore counted as short. This is demonstrated in figure 4 by the curve generated by the distribution function of equation 2. About 41% of its intervals shown in figure 4 are less than 9 minutes. Of the *NA* samples, figure 6 shows that 74% were short, and 26% were long.

We show how *NA* periods followed *AV* periods in figure 6. The conditional probability distribution is very close to the unconditional probability distribution. Short, medium, and long *AV* periods followed *NA* intervals in approximately the same proportion that they occurred. Therefore we conclude that there is no correlation between the length of the *AV* and *NA* periods. This observation was verified by computing the correlation coefficient [21] of *NA* and *AV* periods. It was a very small positive value.

Although the *AV* and *NA* intervals were uncorrelated, a correlation between pairs of intervals of the same type was identified. An *AV* pair is two *AV* periods that are separated by a single *NA* period. Likewise, an *NA* pair is two *NA* periods that are separated by a single *AV* period. A correlation was expected because of the way individuals use their workstations. Users tend to have a cluster of short idle periods, or a cluster of several long idle periods. Some users work on their workstations infrequently, so they have mostly long *AV* intervals separated by long or short *NA* intervals. Figure 7 shows the conditional probability graph of *AV* pairs. Short *AV* intervals were more likely (64%) to be followed by short *AV* intervals. Medium *AV* intervals were more likely (52%) to be followed by medium *AV* intervals. Similarly, long *AV* intervals were more likely (48%) to be followed by long *AV* intervals. Furthermore, if a long interval was not followed by a long interval, then it was more likely to be followed by a medium interval instead of a short interval. Short, medium, and long *AV* intervals were nearly twice as likely to follow the corresponding short, medium, or long *AV* interval than any other kind of *AV* interval. A correlation also existed for *NA* pairs as shown in figure 8. However, it was much less significant. Short *NA* intervals followed short *NA* intervals only slightly more than they followed long *NA* intervals.

3.3. Monthly And Daily Variation Of Availability Of Workstations

The availability of remote cycles of individual stations varied from month to month, but the total system availability of remote cycles remained steady for the entire 5 months. Table 1 shows the percentage of time each station was available for remote cycles from month to month. The row labeled as *system* gives the system availability of cycles. Notice how the system availability

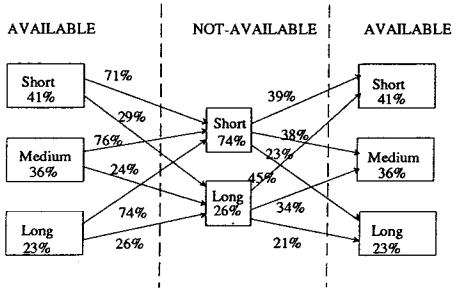

Figure 6.
Conditional Probability of *AV*, *NA* State Changes

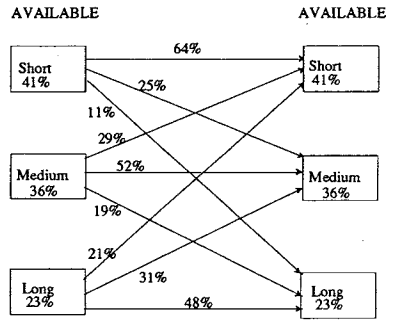

Figure 7.
Conditional Probability of *AV* to *AV* State Changes

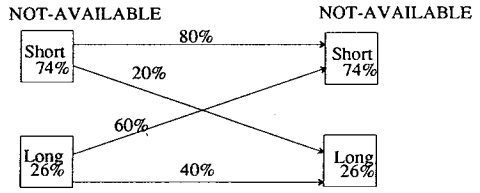

Figure 8.
Conditional Probability of *NA* to *NA* State Changes

remained steady within 3% deviation from the average. This means that there was a large steady amount of available cycles. The column and row labeled "COV" represent the coefficient of variation, which is the standard deviation of the availability divided by the average. The coefficient of variation was computed for each station and for each month. There was a small variation in most cases. Overall, the stations' availability was stable. Figure 9 graphically shows how some individual stations varied their availability, while the system availability remained steady. We emphasize that the actual idle time of the workstations was much larger than the available time reported. Our available time value was conservative. A portion of the larger idle time is due to the fact that any user activity causes a workstation to be unavailable for at least seven minutes. If the seven minute interval for each busy period did not occur, the system availability would increase approximately 4%. Additional idle time occurred during the *NA* intervals between user activities. Therefore, if we were less conservative, there would have been greater observed availability.

The availability of remote cycles varied during the course of a day. It varied during the work week and the weekend. It is assumed that there would be a lot of available capacities during the evenings, and on the weekends when most people were not working. One might wonder if there was large available capacity during normal working times. Figure 10 shows how the availability of remote cycles varied during the week from Monday through Friday between 8am and 5pm (8-17 hour). It shows how some individual stations' availability changed during the day. Early in the morning the system availability was large, and then dropped to about 50% between 2-4 in the afternoon (14-16 hour). Even at the busiest time of the day there was a large amount of capacity to use. Figure 11 shows the system availability of capacity on the weekends between midnight and 11pm (0-23 hour). It confirms the intuition that there was a larger amount of capacity available at those times. The availability on weekends was between 70-80%. The

Machine Name	Specific Months Monitored, Percent of Time In Available State						
	September	October	November	December	January	Average	COV
Station 1	89	80	81	88	84	84	0.1
Station 2	86	89	91	94	90	90	0.0
Station 3	73	28	26	63	67	51	0.4
Station 4	87	84	85	86	81	85	0.1
Station 5	21	0	45	45	32	29	0.6
Station 6	78	81	74	63	69	73	0.1
Station 7	67	27	87	57	91	66	0.3
Station 8	85	72	70	70	47	69	0.2
Station 9	79	88	83	85	80	83	0.0
Station 10	82	89	82	80	78	82	0.1
Station 11	81	84	86	81	86	84	0.0
Station 12	3	96	17	-	-	39	-
Station 13	80	86	75	-	-	80	-
System	70	70	69	74	73	71	
COV	0.4	0.4	0.4	0.2	0.3	0.3	

Table 1: Availability Of Stations From Month To Month.

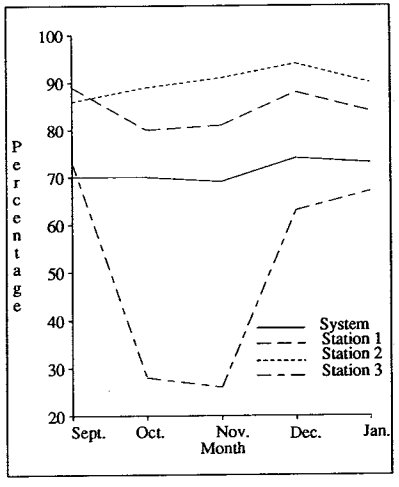

Figure 9.
Availability Of Remote Cycles (Month To Month)

busiest time for the workstations on weekends is shown to be between 2pm and 5pm (14-17 hour).

A remote capacity scheduler is likely to want to know on an hourly basis how individual stations are used. It might wish to know the likelihood that a job placed at a station will be preempted in the next hour. Table 2 gives a profile of the hourly utilization of individual workstations. A station's hourly utilization is the percentage of the hour the station was in the *NA* state. It shows that on the average the hourly utilization of a workstation was less than 10% (*NA* for less than 6 minutes) for 53% of the time. For 21% of the time, the hourly utilization was greater than 90%. The only other significant frequency is the 10-20% hourly utilization. This is due primarily to single 7 minute *NA* intervals occurring during an hour period. Table 2 shows that if each hour

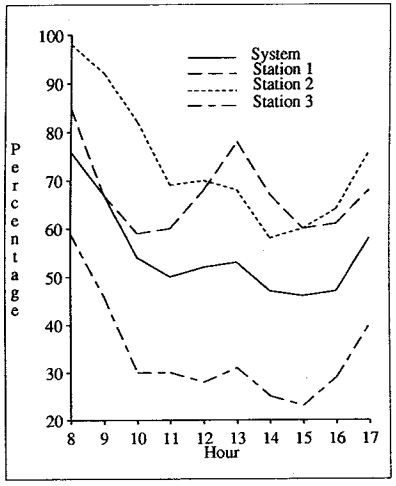

Figure 10.
Availability of Remote Cycles During Weekdays (Mon-Fri, 8am-5pm)

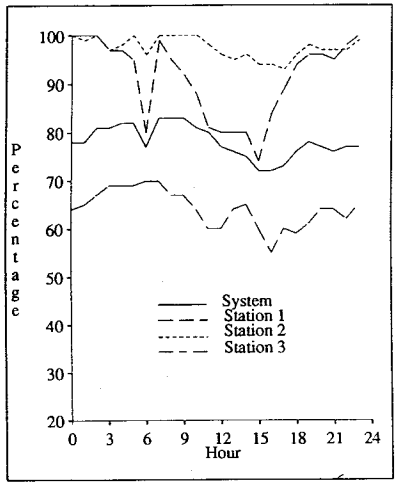

Figure 11.
Availability of Remote Cycles During Weekends (Sat-Sun)

was observed individually, a station was either available for almost the entire hour (*NA* for more than 50 minutes), or was busy for the whole hour.

The row labeled "COV" in table 2 is the coefficient of variation. Across all the stations, the variation of the hourly utilization was small with the exception of the hours that were 10-20% and 90-100% utilized. Although most stations were often busy for the entire hour if they were busy at all, the larger variation in the 90-100% hourly utilization occurred because some infrequently used stations such as stations 1 and 2 were rarely kept busy for an entire hour at a time. The large variation in the 10-20% category occurred because some stations, such as station 3, automatically ran short programs periodically even when the owner was not at the console. These programs kept the station unavailable for at least 7 minutes of a large number of hours. Because most stations do not have these programs, this utilization category has a greater variation.

In addition to the utilization of individual stations, the utilization of the entire system is of interest to a remote capacity scheduler. It would be beneficial to know the relative frequency distribution of the system utilization, SU_l, of all intervals of length l. The system utilization during an interval is the average number of stations in the *NA* state during the interval divided by the total number of stations. The SU_l would help a scheduler estimate the fraction of the system capacity that is available for the next l minutes. Knowledge of the SU_l would help a scheduler know how likely all stations would be in the *NA* state simultaneously. Our analysis of the traces of the stations shows that it was highly unlikely that all stations were in the *NA* state at the same time. We observed that during the five months the system was monitored, the longest period in which no station was available was 10 minutes. This means that from a practical point of view, there was always one or more stations available. The longest period that one would have to wait for 2 stations to become available was 25 minutes. The longest period that one would have to wait for 3 stations to become available was 2 hours.

Table 3 shows the relative frequency distribution of the system utilization for intervals lengths of 60 minutes, 30 minutes, 10 minutes, and 1 minute. Notice that the system utilization was less than 40% for almost 80% of all hour intervals. This means that the probability that at

Machine Name	Percentage Of Hours With This Utilization									
	0-10	10-20	20-30	30-40	40-50	50-60	60-70	70-80	80-90	90-100
station 1	72	8	3	3	2	2	2	2	3	4
station 2	72	8	3	3	2	2	2	2	2	3
station 3	8	44	2	2	2	2	2	1	2	36
station 4	73	7	3	2	2	2	2	2	2	7
station 5	10	4	8	5	4	3	3	2	4	58
station 6	46	17	7	4	2	2	4	2	3	11
station 7	59	3	2	1	1	1	1	1	1	31
station 8	41	13	6	5	3	3	4	3	3	18
station 9	69	5	3	2	2	2	1	1	2	14
station 10	59	14	5	3	2	2	2	1	2	9
station 11	67	5	1	1	1	1	2	1	2	19
station 12	55	2	1	1	1	1	1	1	1	37
stations 13	58	3	2	1	1	1	1	1	1	31
Average	53	10	4	3	2	2	2	1	2	21
COV	0.4	1.1	0.3	0.5	0.3	0.4	0.5	0.4	0.1	0.8

Table 2: Hourly Utilization Of Individual Stations.

least 6 stations were available is almost 80%. There was never an hour where the average number of *NA* stations was greater than 9 stations.

Figure 12 shows how, on an hourly basis, the system utilization and individual station utilization compare. The solid curve in the figure comes from data in Table 3 and the dashed line comes from data in Table 2. Individual stations were likely to be either *AV* or *NA* for entire hours, while the system was likely to have a total of 2-4 stations in the *NA* state. Because individual stations are likely to be either *AV* or *NA* for the entire hour, the prediction of whether a station is available for an entire hour can be approximated by a Bernoulli distribution. We can view the station as having a probability p that it is in the *NA* state, and $1\text{-}p$ that it is in the *AV* state. If we assume that the behavior of each station is independent, then the probability that k stations are in the *NA* state, n_k, can be approximated by the binomial distribution

$$n_k = \left[\frac{11!}{k!(11-k)!} \right] p^k (1-p)^{11-k}.$$

The dashed line in figure 13 is the Bernoulli density function for $p = 0.3$ and the solid line is the corresponding binomial density function. $P = 0.3$ is shown because the hourly utilization of the individual stations was more than 50% for approximately 30% of the time. Notice the similarity of the shapes of the curves in figures 12 and 13. This indicates that the stations can be viewed as independent and the system utilization can be approximated by a binomial distribution.

Utilization, ρ	SU_{60}	SU_{30}	SU_{10}	SU_1
0 ≤ρ< 10	6.0	8.4	13.0	18.0
10 ≤ρ< 20	28.1	25.2	24.4	24.6
20 ≤ρ< 30	28.2	29.1	24.7	22.7
30 ≤ρ< 40	16.2	15.3	15.4	13.4
40 ≤ρ< 50	8.7	8.8	8.9	7.8
50 ≤ρ< 60	6.5	6.7	6.4	5.7
60 ≤ρ< 70	4.4	4.4	4.3	4.0
70 ≤ρ< 80	1.6	1.7	2.1	2.3
80 ≤ρ< 90	0.3	0.4	0.8	1.1
90 ≤ρ≤100	0.0	0.1	0.1	0.4

Table 3: Relative Frequency Distribution Of System Utilization, *SU*

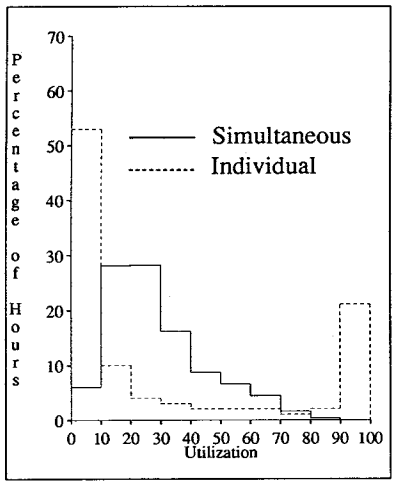

Figure 12.
System And Individual Station Utilization

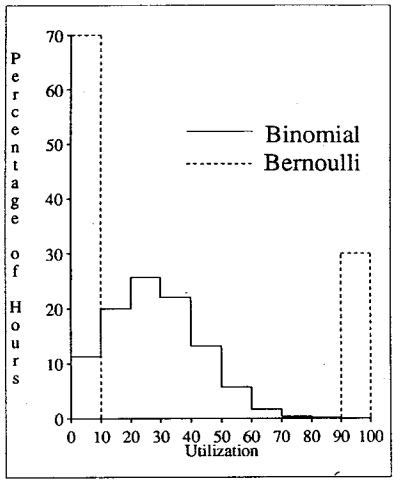

Figure 13.
Bernoulli and Binomial Probability Density Functions

Large availability of system capacity means that if a long running sequential job can be distributed, several stations are likely to be available to serve it. Suppose a user has a job that would execute on a station for 5 hours. Because it is very likely that more than 6 stations are available simultaneously for an hour interval, a version of the long running job distributed into 6 or more processes will likely be able to acquire enough available processing cycles to complete within an hour.

4. Development Of Stochastic Models

From our results we can define a family of models that describe the behavior of the users at different levels of accuracy. A first approximation is a model where state times are independent random variables. In this model, the state lengths are distributed exponentially. The expected value of the distribution is the mean of the observed *AV* and *NA* state lengths.

A more detailed model will include the observations that 3-stage and 2-stage hyperexponential distributions closely fit the relative frequency distributions of the *AV* and *NA* state lengths. In this model the distribution of the state lengths of the first approximation are replaced by the 3-stage and 2-stage hyperexponential distributions.

By taking into account the correlation of the state lengths, further detail is put into the model. This leads to our stochastic model of workstation usage. The length of the next *AV* period depends on the length of the current *AV* period. Likewise, the length of the next *NA* period depends on the length of the current *NA* period. In this model, the type of next interval, whether it is small, medium, or large, is based on the conditional probabilities presented in figures 7 and 8. When the next interval's type is known (small, medium, or large), the next sample's length is specified as a random sample from an exponential distribution that is a component of equations 2 or 3 that represent the sample's type. For example, long *AV* samples are chosen from an exponential distribution that has an expected value of 300 minutes.

Further accuracy can be introduced into the model by giving each workstation its own distributions and correlations of state lengths. A workstation's availability depends on the time of day,

and the day of the week. A model which includes this information is more complex, and more precise. We plan to study the models to understand which level of complexity is necessary in order to capture the important characteristics of workstation availability.

5. Applying The Workload Model To The Design Of Resource Management Algorithms

Design strategies can take advantage of our observations of workstation behavior. Our model describes when, how much, and for how long remote capacity is available. This information is important because there must be long periods of time where owners are not using their stations in order for capacity sharing to be feasible. This is because a remotely placed job using shared capacity is preempted whenever the station's owner resumes with local usage of his/her station. The remotely placed job can remain at the same location until the workstation becomes available again, or can be moved elsewhere. Our observations indicate that workstations monitored for a particular hour tend to remain either available or non-available for the entire hour. This information is ideal for designing schedulers. A scheduler can remotely place a job at a location and know there is a high probability that the job will stay there for most of an hour without being preempted. Another important observation was that a workstation with a long *AV* interval was likely to have its next *AV* interval to be long. This information provides an important heuristic for a scheduler to use when deciding which available stations to allocate as sources of remote capacity. Those workstations with recent long available intervals should be considered before workstations with recent short available intervals when targeting stations as sources of remote cycles.

The results of this paper are useful for the development and evaluation of algorithms that schedule and predict completion times of long running background jobs that execute in a LOCOX network. Algorithms can be developed that schedule jobs based upon the priority for completion that a user gives to the jobs, and predict for the user when the jobs complete. In order to predict completion, an algorithm needs to know the expected availability patterns of remote capacity in the system. This paper gives insight of this information. The scheduling algorithm can internally reserve expected future capacity of the system to jobs based both upon the priority that the user assigns to jobs, and the global priority the user has in relation to other users. From this information, an algorithm could give good estimates of when submitted background jobs are expected to complete. A user could use this information interactively with the scheduler to decide how to give priority to jobs.

Other applications of the workload characteristics and results presented in this paper are possible. These examples only serve to give insight on the benefits of knowing the characteristics of workstation workloads.

6. Conclusions

Networks of powerful workstations have become prevalent in modern computing environments. These networks of workstations provide the possibility of expanding the capacity available to a user beyond his/her local workstation to that of the entire network. With the advent of the LOCOX networks, large computing capacity can be available for general computing usage. However, special attention is particularly important due to the fact that workstations are private resources under the control of their owners and only become sources of remote cycles when owners of workstations make them available. In our LOCOX network, the system availability was approximately 70% for the time observed. Long available intervals were common. Although availability varied from station to station, the variation of each station was typically small. The system availability was stable from month to month. Not only during evenings and weekends was the availability large, but also during the busiest times of weekdays. When stations were observed for hour intervals, they typically were either available for most of the hour, or unavailable for the entire hour.

In this paper, we have presented the profile of the workload of a LOCOX network and a stochastic model that matches it very closely. Since performance evaluation studies are critically dependent on the workload processed by a system, the workload description is especially important. The model of workstation activity presented is based on the observed distributions of *AV* and *NA* intervals, and their correlation. An exponential distribution does not adequately represent the relative frequency distributions of *AV* and *NA* state lengths. Observations presented in this paper show that 2-stage and 3-stage hyperexponential distributions fit the relative frequency distributions extremely well. We found that the length of *AV* intervals was not correlated with the length of *NA* intervals, but the length of pairs of intervals of the same type were correlated with each other. Additional observations show that each workstation has its own distributions of *AV* and *NA* patterns.

The complexity of a stochastic model depends on the level of details included in the model. It is important to understand which level of complexity is necessary to capture the important characteristics of a workload. To help understand this, we plan to study scheduling algorithms described in [22] with the different stochastic workloads presented in this paper. By contrasting performance results of algorithms when using different workload models, we can learn which level of detail is necessary. In addition, we plan to develop new algorithms for remote capacity scheduling and performance prediction in a LOCOX network that take advantage of the knowledge of the workload characterization.

7. Acknowledgements

The authors would like to thank Mike Litzkow for placing the monitoring program on each of the workstations so that we could gather the data.

References

[1] C. G. Bell, "Technology 86: Minis and Mainframes Expert Opinion", *IEEE Spectrum*, **23**(1), pp. 36-37, (January, 1986).

[2] P. Ein-Dor, "Grosch's Law Re-Revisited: CPU Power And The Cost Of Computation," *Communications Of The ACM* **28**(2) pp.142-151 (February, 1985).

[3] M. M. Theimer, K. A. Lantz, and D. R. Cheriton, "Preemptable Remote Execution Facilities for the V-System," *Proceedings of the 10th Symp. on Operating Systems Principles*, pp. 2-12, (December, 1985).

[4] R. Hagmann, "Processor Server: Sharing Processing Power in a Workstation Environment," *Proceedings of the 6th IEEE Distributed Computing Conference*, Cambridge, MA, (May, 1986), pp. 260-267.

[5] R. Agrawal and A. K. Ezzat, "Processor Sharing In Nest: A Network Of Computer Workstations," *Proceedings of 1st International Conference on Computer Workstations*, (November, 1985).

[6] D. Ferrari, "Workload Characterization An Selection In Computer Performance Measurement," *Computer* **15**(4) pp. 18-24 (July-August, 1972).

[7] D. Ferrari, *Computer Systems Performance Evaluation*, Prentice-Hall, Englewood Cliffs, N.J. (1978). Chapter 5.

[8] J. A. Stankovic, "Simulations of Three Adaptive Decentralized Controlled, Job Scheduling Algorithms," *Computer Networks*, **8**(3), (June, 1984).

[9] J. A. Stankovic, "Stability and Distributed Scheduling Algorithms", *IEEE Transactions on Software Engineering*, **SE-11**(10), (October, 1985).

[10] J. A. Stankovic, "An Application Of Bayesian Decision Theory to Decentralized Control of Job Scheduling," *IEEE Transactions on Computers*, **C-34**(2), (February, 1985).

[11] Y-T Wang and R. J. T. Morris, "Load Sharing in Distributed Systems," *IEEE Transactions on Computers*, **C-34**(3), (March, 1985).

[12] A. D. Eager, E. Lazowska, and J. Zahorjan, "Adaptive Load Sharing in Homogeneous Distributed Systems," *IEEE Transactions on Software Engineering*, **SE-12**(5), (May, 1986).

[13] W. E. Leland and T. J. Ott, "Load-balancing Heuristics and Process Behavior," *Proc. of the 1986 ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*

(May, 1986).

[14] S. Zhou, "A Trace-Driven Simulation Study of Dynamic Load Balancing," University Of California Technical Report UCB/CSD 87/305, (March, 1987).

[15] *Unix 4.2BSD Manual Page for PS (Process Status).*

[16] M. Litzkow, "Remote Unix," *Proceedings of the 1987 Summer Usenix Conference* Phoenix, Arizona (June, 1987).

[17] M. Litzkow, Private Correspondence, Computer Sciences Department, University Of Wisconsin, Madison, Wisconsin, (April, 1987).

[18] K. S. Trivedi, *Probability And Statistics With Reliability, Queueing, And Computer Science Applications*, Prentice-Hall, Englewood Cliffs, N.J., (1982). pp. 129-130.

[19] K. Kobayashi, *Modeling and Analysis: An Introduction to System Performance Evaluation Methodology*, Addison-Wesley Publishing Company, 1981.

[20] M. L. James, G. M. Smith, and J. C. Wolford, *Applied Numerical Methods for Digital Computation*, Harper & Row, Publishers (1977), pp. 285-287.

[21] D. E. Knuth, *The Art Of Computer Programming*, Vol 2: Seminumerical Methods, Addison-Wesley Publishing Company, (1981).

[22] M. W. Mutka and M. Livny, "Scheduling Remote Processing Capacity In A Workstation-Processor Bank Network," *Proceedings of the 7th IEEE Distributed Computing Conference*, Berlin, West Germany, (September, 1987).

PERFORMANCE '87
P.J. Courtois and G. Latouche (editors)

ROUTING AND CAPACITY ALLOCATION IN QUEUEING AND LOSS NETWORKS

F.P.Kelly*

Statistical Laboratory
University of Cambridge
16 Mill Lane
Cambridge CB2 1SB
England

How should demands be routed in a network so as to improve the performance of the network? We consider the question, paying particular attention to the common features of queueing and loss networks. We outline how product-form solutions and related approximation procedures can be used to provide important structural insights into optimization issues. In particular, they lead to implicit shadow prices associated with each route and with each component of the network, where the equations defining these prices have a local character.

There are a number of implications concerning the extent to which control and planning can be decentralized. At one level the results suggest adaptive routing schemes, capable of responding automatically to local overloads or failures. At another level the shadow prices can be used as a basis for pricing policy or the apportionment of revenue between different sections of a network operation.

In a queueing network an arriving demand makes use of different resources sequentially in time, and congestion causes delay. In a loss network an arriving demand requests simultaneous use of different resources, and congestion causes blocking. Despite these differences there is a remarkable similarity between the models used to analyse queueing and loss networks. In both areas product-form solutions arise as exact equilibrium distributions for

* This work was supported in part by the Nuffield Foundation.