# Preliminary Results on the Theoretical Potential of Scheduling With Partially Correct Predictive Data

Andrew S. Noetzel

This work was supported by
National Science Foundation Grant GJ-39658

Technical Report No. 36

Department of Computer Sciences
The University of Texas at Austin
Austin, Texas 78712

July, 1974

TABLE OF CONTENTS

## ABSTRACT

In this report, two models of scheduling with partially correct predictive data are presented. The first model is an open queueing network for analysis of waiting time; the second a simple closed network. In both cases, a scheduling discipline that has probabalistic knowledge of a customer's service time is assumed; and the results are parameterized by the probability p that the assumed service time is correct.

An integral expression for the average waiting time is derived for the first model. Numerical evaluations for several service time distributions as a function of p are shown. The results show that order-of-magnitude reductions in average waiting time can be attained as p increases, when the utilization factor is high. The largest part of the decrease occurs for the smaller values of p.

The closed queueing model assumes one FCFS exponential processor, and one processor scheduled by the probabilistically correct estimates. Numerical results, obtained by simulation, show the idleness of the processor can be reduced linearly, by a factor of two, as p increases from zero to one.

## I. Introduction

A predictive scheduling algorithm is one which bases its decisions on available (partial or complete) information about the future characteristics of the job being scheduled. Study of predictive algorithms in queueing theory is limited to a few special cases: the Shortest-Processing-Time-First (SPTF) algorithm for non-preemptive scheduling, and the Shortest-Remaining-Processing-Time-First (SRPTF) discipline for scheduling with preemption (Ref. 5,6). The lack of more extensive analysis may be due to the belief that this study is not practical, since processing times are generally not known in advance, or it may be due to the difficulty of analysis in the general case of scheduling with partially predictable service times.

In this report, some indications of the theoretical potential of predictive algorithms are obtained, in terms of both waiting time and processor utilization. The results are parameterized by the degree of correctness of the predictive data.

The waiting-time result is based on an algorithm that varies from random-priority assignment to shortest-job-first assignment, as the correctness of the predictive data increases from zero to one. Numerical results are shown for two specific processing-time distributions. For the processing-time result, a specific predictive algorithm was compared in simulation against known nonpredictive theoretical models.

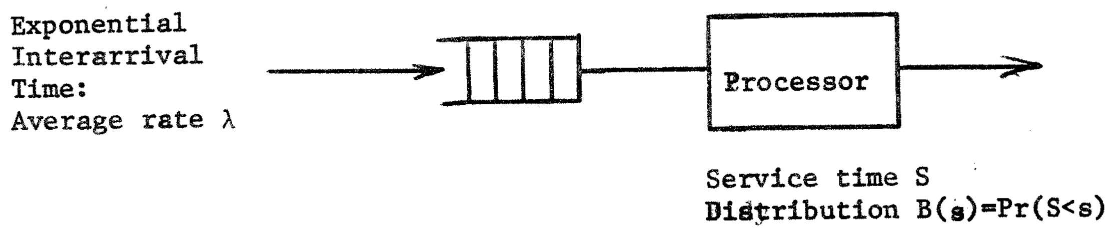
The results demonstrate that there can be considerable advantage in using predictive data, even when it is far from completely correct, to achieve either lower average waiting time, or higher processor utilization. Furthermore, the value of the predictive data increases as the

the variance of the processing-time distribution increases.

Section Two of this report discusses predictive scheduling in an open network, to achieve the benefit of improved average waiting time. Section Three demonstrates the effect of improved utilization in a closed network.

## II. Predictive Scheduling in a Single-Processor Open Network.

A basic model in the study of queueing is the so-called M/G/1 system, consisting of a single processor with a general service time distribution, at which customers (jobs) arrive with exponentially distributed interarrival times.




The utilization $\rho$ of the processor is dependent only on the input rate $\lambda$, and average service rate $E(S)$, where S is the service-time random variable. The relationship $\rho = \lambda E(S)$ holds for every scheduling discipline that does not allow the processor to become idle while there is at least one customer on the queue. In particular, there is no difference between the processor utilization yielded by predictive algorithms (which have advance knowledge of the jobs processing-time) and non-predictive algorithms.

The distribution of waiting times of jobs in the queue does vary, however, under different scheduling algorithms, and so does the average waiting time. In fact, it is well known that the predictive Shortest-Processing-Time-First (SPTF) algorithm yields the lowest average waiting time of all non-preemptive algorithms. The average wait time $W_{SPTF}$ (including processing time) of jobs scheduled by the SPTF algorithm is given by (Ref. 6):

$$W_{SPTF} = E(S) + \frac{\lambda E(S^2)}{2} \int_0^{\infty} \frac{dB(s)}{\left(1-\lambda \int_0^{s} \tau dB(\tau)\right)^2}$$

When the jobs are scheduled on a First-Come-First-Served basis the average waiting time is:

$$W_{FCFS} = E(S) + \frac{\lambda E(S^2)}{2(1-\rho)}$$

A scheduling algorithm that assigns jobs on the basis of partially correct predictive data is expected to yield an average waiting time within the range bounded by the foregoing values. To obtain an expression for this continuum, a result from the general ease of scheduling according to assigned priorities is used. The waiting time $W(t,s)$ of a job that has priority $t$ and service time $s$ is

$$W(t,s) = s + \frac{\lambda E(S^2)}{2(1-\rho_t)}$$

where $\rho_t$ is the utilization of all jobs whose priority is greater than $t$.

Now, suppose jobs are scheduled according to randomly-assigned priorities. Let the probability distribution for priority $T$ be identical to that of the processing-time random variable $S$. The priority is in fact the predicted processing time: the scheduling rule selects as having highest priority the job with the smallest predicted processing time. With probability $p$, $T = S$, otherwise, the predicted processing time is independent of the actual time.

Let $b(t,s)$ be the joint probability density function for $T$ and $S$. Then,

$$b(t,s) = pb_1(t,s) + (1-p)b_2(t,s)$$

where $b_1(t,s)$ is the joint distribution for the case when the prediction is the same as the processing time.

$$b_1(t,s) = f(t)\,\partial\,(t,s)$$

$\partial\,(t,s)$ is the probability mass function for probability uniformly distributed on the line $t = s$, and zero elsewhere, and $f(t)$ is the marginal density function for both processing time and priority.

$b_2(t,s)$ is the joint distribution for the case when processing time and priority are independent and have the same probability density function $f(x)$. Then, $b_2(t,s) = f(t)f(s)$.

$\rho_t$ is computed as the average input rate times the average of processing time $s$ weighted by the probability that $t<s$.

$$\rho_t = \lambda \int_0^\infty s \int_0^t b(\tau,s)\, d\tau ds$$

$$= \lambda\, p \int_0^\infty s \int_0^t f(\tau)\, \partial\,(\tau,s)\, d\tau ds$$

$$+ \lambda(1-p) \int_0^\infty s\, f(s) \int_0^t f(\tau)\, d\tau ds$$

But

$$\int_0^t f\,(\tau)\,\partial\,(\tau,s)\, d\tau = \begin{cases} f(s) & \text{if } s \leq t \\ 0 & \text{if } s > t \end{cases}$$

Then,

$$\rho_t = \lambda p \int_0^t f(s)ds + \lambda(1-p)\, F(t)\, E(S)$$

The scheduling discipline that dispatches jobs according to their predicted processing time will be called the random priority (RP) algorithm. The expected wait time $W_{RP}$ is

$$W_{RP} = E(S) + \frac{E(S^2)}{2} \int_0^\infty \frac{dB(s)}{\left(1-\lambda p \int_0^t \tau dB(\tau) + \lambda(1-p)B(t)E(s)\right)^2}$$

With p = 1.0, here, the algorithm becomes the SPTF algorithm. With P = 0, the priorities are independent of processing time. Then, because

$$\int_0^\infty \frac{dB(t)}{(1-\rho B(t))^2} = \frac{1}{1-\rho} \quad \text{whenever } B(t) \text{ is a probability}$$

distribution, the expression for the waiting time is the same as that of the FCFS algorithm.

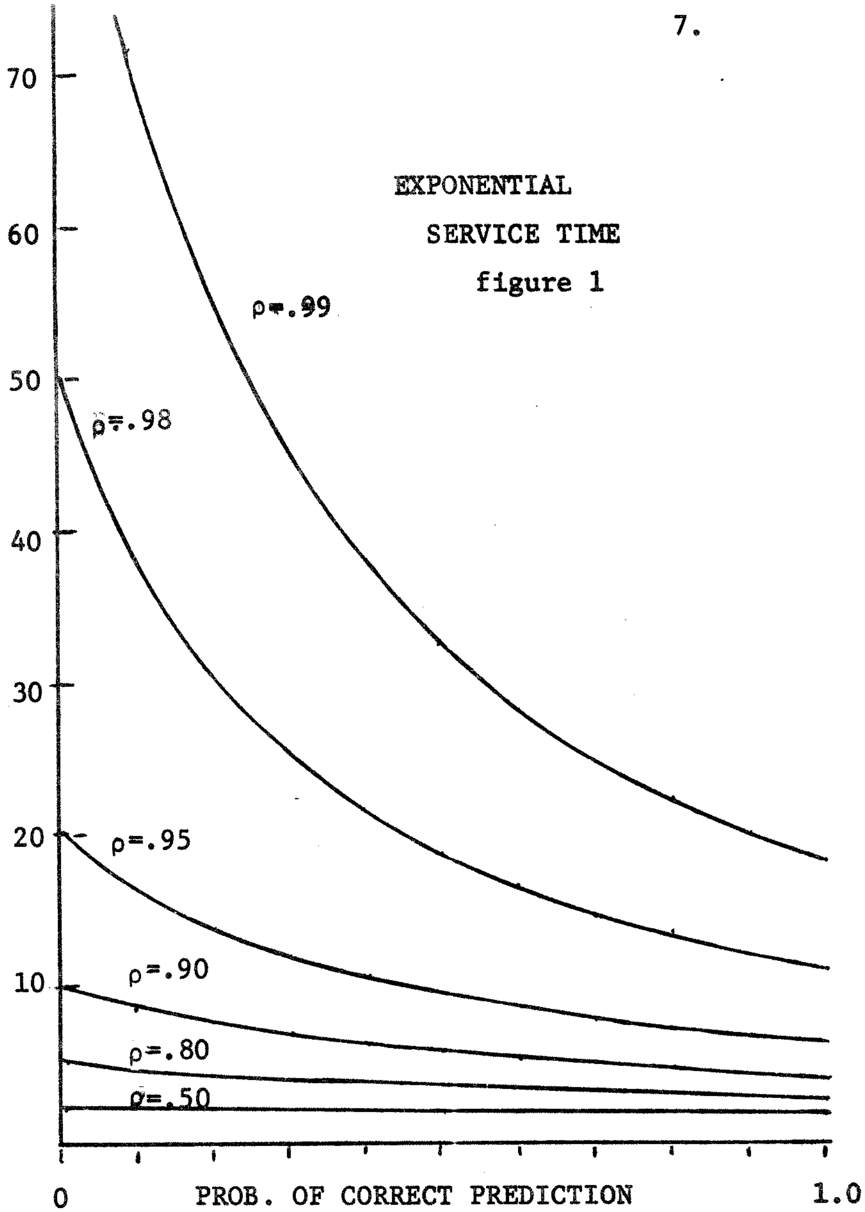
The variation of waiting time as a function of p is demonstrated in figures 1 and 2, for exponential and hyperexponential service time, respectively. Here, it is seen that for moderate values of $\rho$ ($\rho \approx .8$), the average waiting time is reduced by a factor of two, as the reliability of the predictive data increases. Order-of-magnitude decreases in waiting time are achieved when the utilization is high ($\rho \simeq .99$) Furthermore, the greatest part of the decrease in waiting time occurs when the probability of correctness is low. The conclusion is that the predictive data need not be highly accurate in order to be helpful in achieving small average wait times when the utilization factor is high.

For the results of figure one, the service time distribution is $B(t)=1-e^{-t}$, which has mean and variance one. The hyperexponential distribution used for figure two is

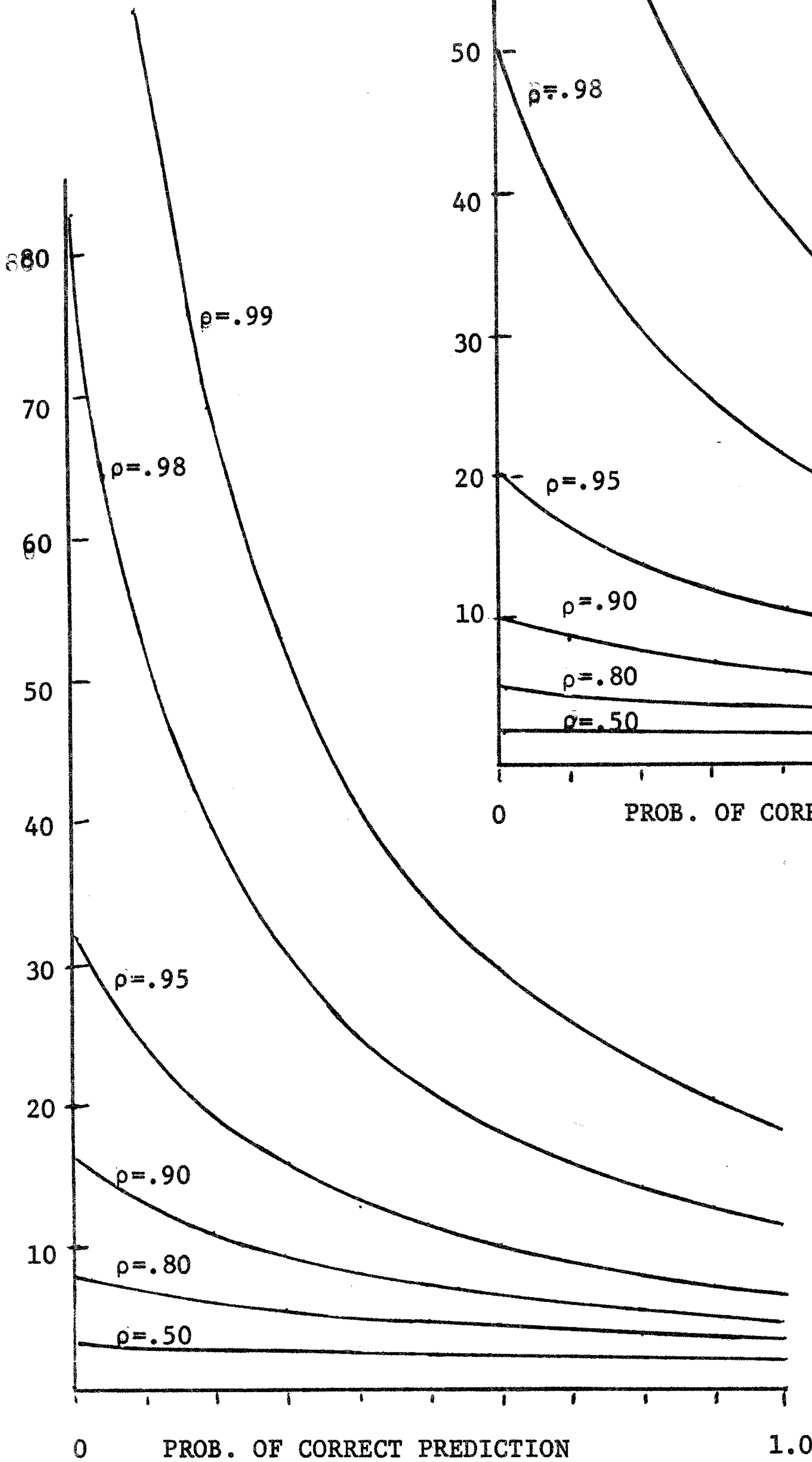
$$B(t) = 1-\frac{1}{2}(e^{-.55t} + e^{-5.5t}),$$

which has mean 1.0 and variance of 2.173. Because the decrease in waiting time with increased correctness of predictive data is greater for the hyperexponential case, these instances of processing-time distribution support the intuition that the value of predictive data increases with the variance of the processing time.

NORMALIZED
WAIT TIME



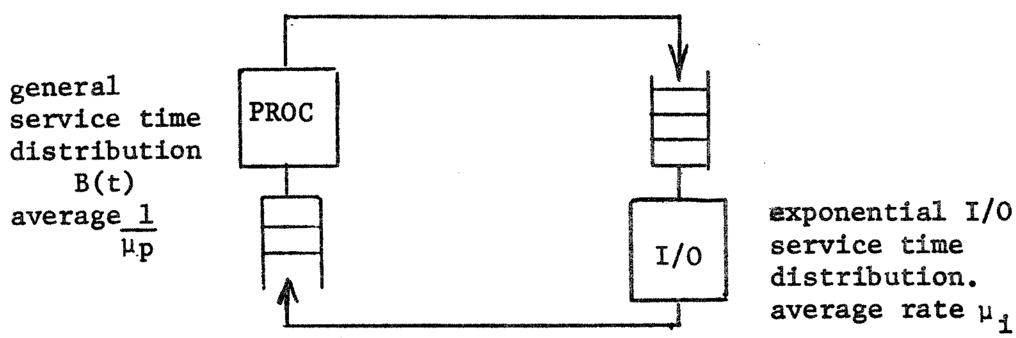

EXPONENTIAL
SERVICE TIME
figure 1




HYPEREXPONENTIAL
SERVICE TIME
figure 2

## III. Predictive Scheduling in Closed Queueing Networks

Queueing Networks that can be studied analytically are those for which the scheduling discipline at each processor yields an output stream that is a Markov process. Examples of such disciplines are Last-Come-First-Served-Preempt/Resume, Processor Sharing and (for exponential service times) FCFS. (Ref. 4) But for these cases, each individual processor is loaded in the same way as it would be if it were the only element of an open queueing network; the scheduling discipline does not influence utilization. Analytical results for classes of more interesting scheduling displines, in particular, networks with mixed predictive and non-predictive stations, have not been obtained. As an indication of the expected effectiveness of such dispçiplines, the simple queueing network shown below will serve for comparison of analytical and numerical (simulation) results.




Here, the station marked I/O will always have an exponential service time, with parameter $\mu_i$, and will always be scheduled by the FCFS discipline. If then, the processor station also has an exponential service time, (parameter $\mu_p$) every discipline that does not use advance knowledge of the service time will result in the processor utilization ħp. (Ref. 5)

$$\eta_p = \frac{\rho-\rho^{N+1}}{1-\rho^N},$$

where N is the total number of jobs in the network, and $\rho = \frac{\mu_i}{\mu_p}$.
In the particular instance $\mu_i = \mu_p$, $\eta_p = \frac{N}{N+1}$ by L'Hopital's rule.

The utilization of the processor may be improved if the processor scheduling discipline has advance knowledge of the processing time of each job. But in this case, analysis of the network is difficult, even for exponentially-distributed processing times, because the rate at which jobs leave the processor to join the I/O queue is not a Markov process.

If the non-preemptive Shortest-Job First discipline is in effect, for example, and two jobs are on queue for the processor when a job completes, then the job with the minimum of the two processing times is selected for the processor. If the processing-times of jobs have distribution B(t), with density function b(t), then the processing rate in effect after the selection of the minimum-processing-time job (of two jobs) is computed as follows: Let $bm(t|s)$ be the processing-time density of the smaller job, given that the other job has processing time S. Then

$$bm(t,s)=\begin{cases}\frac{b(t)}{B(s)} & \text{for } t<s\\ 0 & \text{for } t>s\end{cases}$$

Removing the conditioning on S, bm(t) --the density function for the minimum processing time--is obtained.

$$\begin{aligned} bm(t) &= \int_0^\infty bm(t|s)\ dB(s) \\ &= \int_t^\infty \frac{b(t)}{B(s)}\ dB(s) \\ &= -b(t)\ \ln\big(B(t)\big) \end{aligned}$$

If the processing-time distribution B(t) is exponential, then the distribution in effect for the minimum of two jobs is not.

The expected benefit of predictive information in scheduling in this network can be obtained by comparing the processor utilization $\eta_p$, above, for the case of any scheduling algorithm with no predictive data (and exponenential processing time) with simulation of the above network in which the processor is scheduled by the preemptive Shortest-Predicted-Remaining-Processing-Time (SPRPT) algorithm.

The SPRPT algorithm used in the simulation runs is parameterized by p: the probability that the estimated processing time of a job, which determines the scheduling priority of the job, is the same as the actual processing time of the job. When a job is run on the processor for a period, the estimated processing time is reduced to become the estimate of the remaining processing time for the job. If the estimate time reaches zero before the actual processing period completes, the estimate was known to be in error, and is then reset to (and, for determining scheduling priorities, remains set to) the average time for the exponentially-distributed jobs.

The results of the simulation runs are given in figure three. The cumulative probabilities of n or fewer jobs on the processor queue (for $0 \leq n \leq N-1$, where $N = 5$) are plotted against the probability p that the predictive data is correct.

When p = 0, scheduling follows randomly assigned priorities. The probability of m jobs on the processor queue is $P(m) = \frac{N}{N+1}$ for all m.

As p increases, P(0) and 1-P(N) decreases until they reach values of about half their original values. For n in the middle of the range

(0,N), P(n) increases.

The variation of the P(n) values with p is shown to be linear in the figure. The exact theoretical relationship may not be linear, but the nonlinearity is not significant enough to be observable.

The same processor-scheduling algorithm was used in a simulation run in which the processing-time distribution was the hyperexponential function

$$B(t) = 1 - \frac{1}{2}\left(e^{-.55t} + e^{-5.5t}\right).$$

The results are shown in figure 4.

The effect of the predictive data in reducing idleness is greater, because P(0) has a greater value under the randomly-assigned priorities discipline (p = 0). The reduction of P(0) appears non-linear, with the greater part of the decrease at larger values of p.

The algorithm did not attempt to dynamically recompute the expectation of the remaining processing time on the basis of processing time already elapsed, when the original estimate was found to be incorrect. Such a comparison would involve considerable extra overhead, since it requires evaluation, at every scheduling opportunity, of the expectation of a normalized segment of the processing-time distribution. If such evaluation were performed, it would reduce the values of P(0) for the mid-range of p, but not for p=0, (randomly-assigned priorities) or p=1 (no incorrect predictions).

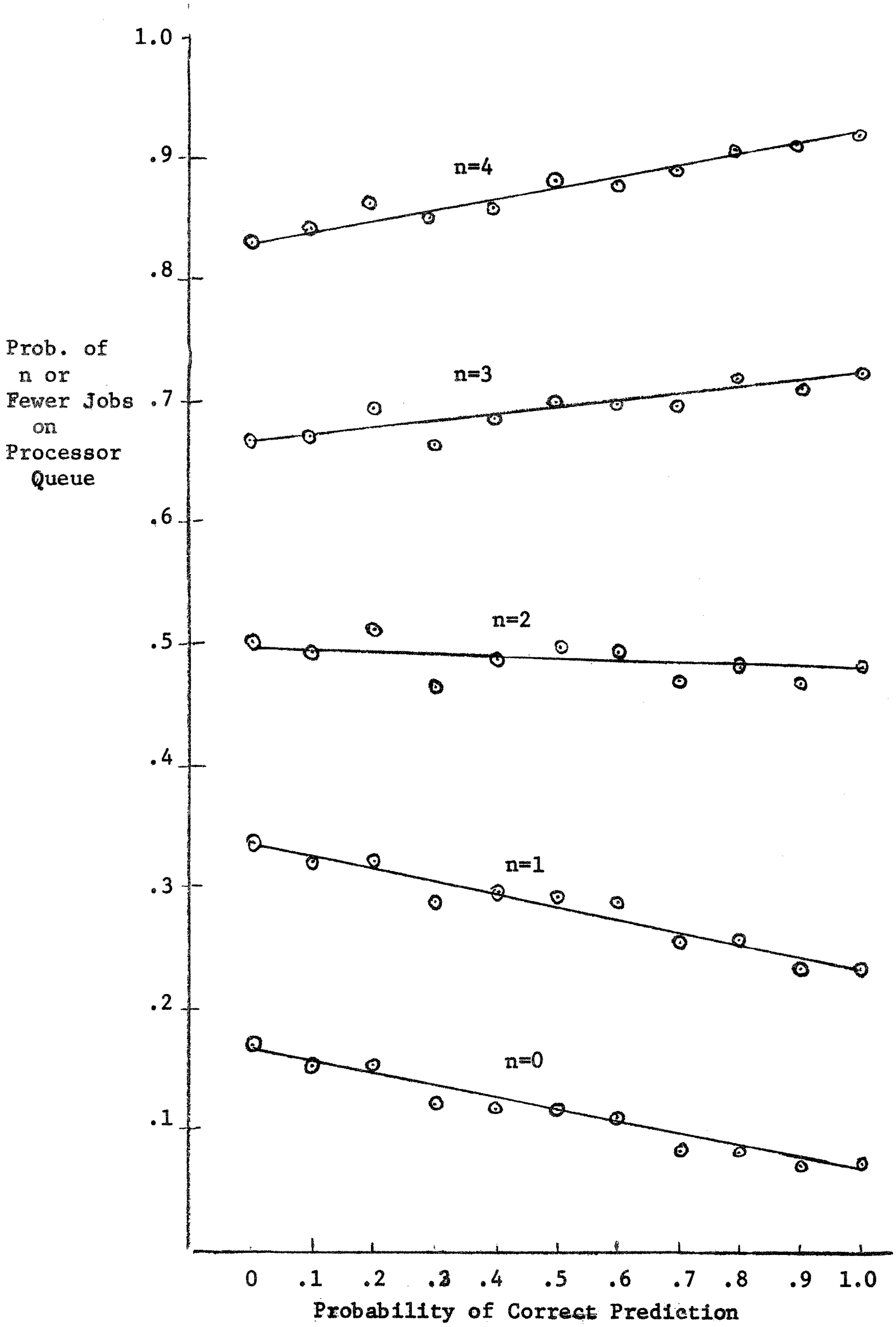



Figure 3:
Distribution of Processor Queue Length vs. Correctness
of Service Period Prediction:
Exponential Service Times

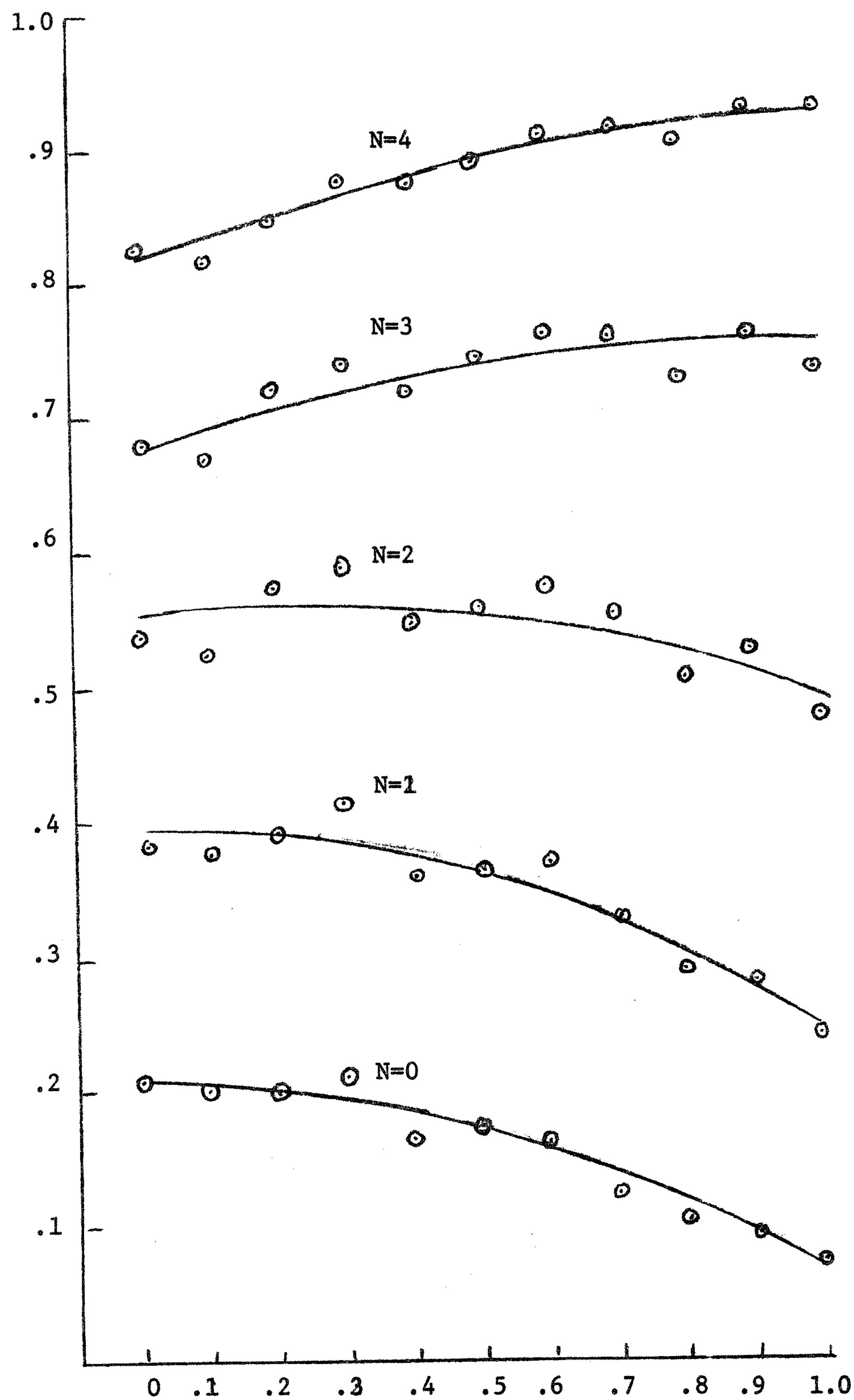



Figure 4:

Distribution of Processor Queue Length vs. Correctness of Service Period Prediction: Hyperexponential Service Time.

## IV. Conclusions

The theoretical analysis of queueing disciplines in which the service period is known, but with less than total accuracy, is difficult. The results expected from such analysis will show the parameters of interest vary from the case of randomly assigned priorities to the case of scheduling with complete foreknowledge (as exemplified by the Shortest-Job-First algorithm) as the variable probability of correctness of the predictive data increases from zero to one.

If the parameter of interest is average waiting time, the estimates of processing time have the greatest part of the reduction of average processing time occurs at the smaller values of correctness of the estimate.

In a simple network in which the processor is scheduled in a preemptive Shortest-Estimated-Remaining-Processing-Time-First, processor idleness can be reduced by a factor of approximately two as the quality of the predictive data is increased. The reduction of processor idleness varies is approximately linearly with the correctness of predictive data.

Even though the analysis of predictive scheduling techniques is incomplete, the results shown here are sufficient to endourage simulation studies and design of more complex schedulers that use predictive data, even where only partially correct predictive data is available.

## REFERENCES


1. Baskett, F.
   Mathematical Models of Computer Systems. Ph.D. Dissertation, The University of Texas at Austin, 1970.

2. Baskett, F., Browne, J. C., and Raike, W. M.
   "The Management of a Multi-Level Non-Paged Memory System," Proc. AFIPS 1970 SJCC, Vol. 36, Montvale, N.J., pp. 459-465.

3. Buzen, J.
   Queueing Networks of Multiprogramming. Ph.D. Dissertation, Division of Engineering and Applied Physics, Harvard University, June 1971.

4. Chandy, K. M.
   The Analysis and Solutions for General Queueing Networks, Proc. Sixth Annual Princeton Conference on Information Sciences and Systems, 1972.

5. Coffman, E. G. and Denning, P. G.
   Operating Systems Theory, McGraw-Hill, 1973.

6. Conway, R. W., Maxwell, W. L. and Miller, L. W.
   Theory of Scheduling, Addison-Wesley, 1967.

7. Estrin, G., Muntz, R. R., and Uzgalis, R. C.
   Modelling, Measurement, and Computer Power, Proc SJCC, 1972, pp. 725-738.