Management of Computing

Gordon B. Davis
Editor

Pricing Computer Services: Queueing Effects

HAIM MENDELSON

ABSTRACT: This article studies the effects of queueing delays, and users' related costs, on the management and control of computing resources. It offers a methodology for setting price, utilization, and capacity, taking into account the value of users' time, and it examines the implications of alternative control structures, determined by the financial responsibility assigned to the data processing manager.

1. INTRODUCTION

The importance of queueing effects in studying the performance of computer systems is well known; in fact, queueing models dominate the computer performance evaluation literature.[1] It is not as clear, however, what the implications of these effects on the management and control of computing resources are, thus there is a gap between the technical level and the management level. This gap limits the ability of the data processing manager to make sound resource allocation and capacity decisions, and hampers his ability to justify investments for the purpose of "improving service quality" or "reducing turnaround time" to high-level management. Although top management may well sympathize with the desire to improve service quality, the inability to quantify the associated tradeoff is likely to put the data processing manager at a disadvantage in the competition for the limited financial resources of the organization. If, in addition, the data processing department is not covering its costs, or the utilization of existing resources seems low, the data processing manager is even less likely to be successful in presenting his case to top management.

This article studies the effects of queueing delays, and users' related costs, on the management and control of computing resources. The issue is studied by embedding a fairly general queueing model in the standard microeconomic framework used to study price and capacity decisions in the computing context (cf. [2, 13, 23]. We examine the queueing effects under a number of data processing control structures, determined by the financial responsibility assigned to the data processing manager (cf. [5, 20]). Such control structures may take on a variety of forms, ranging from "free access" (i.e., no explicit chargeout; cf. [17]), through a "complete chargeout" structure which calls for full cost recovery, to the "profit-center" structure, where data processing is managed as an independent profit-maximizing division. This study assesses the impact of users' delay cost on the price, capacity, and utilization obtained under different financial responsibility (or control) structures, and compares them to the results obtained under the objective of maximizing the expected net value of data processing services to the organization.

In addition to the analytic framework suggested by our results, which provides guidance to the data processing manager in the determination of price and capacity as well as assistance in presenting his case to higher-level management, the implications of our results help clarify a number of issues which arise in the context of computing resource management; some of these issues are listed below.

1. Computing centers are often required to set prices to cover their costs (e.g., most of the centers studied by Nolan [20] and Drury [8] were responsible for covering 100 percent of their costs), a requirement which stems from standard cost-accounting practices (cf. [12]). We demonstrate that in the presence of delay costs, cost recovery may be quite undesirable and would lead to

[1] A special issue of *Computing Surveys* (Sept. 1978) may serve as a tutorial on this literature.

This work was partially supported by the Managerial Economics Research Center of the University of Rochester, N.Y.

under-utilization of resources. Our results suggest that the organization as a whole would benefit from prices that lead to a computing center deficit; further, we show that (under the standard assumptions made by cost-accountants), the magnitude of this deficit is equal to the expected overall delay cost incurred by the aggregate user population. In particular, our results reduce to the standard cost-accounting procedure when the cost of delay is zero.

2. It is sometimes argued that computing resource pricing need not take queueing effects into account since these effects are "self-regulating": Congestion leads to delays which deter additional users from joining the system. Our results suggest that the "self-regulated" equilibrium leads to an over-congested system, and that the organization as a whole would benefit from reducing congestion by imposing queueing-related charges. We demonstrate, analytically and graphically, the determination of the expected-net-value maximizing price, and show that it is equal to the expected delay cost inflicted (by the added work load) on the rest of the system.

3. It is often suggested that organizing the computing center as a profit center would lead to an improvement in its overall performance. In contrast, users and data processing managers are often uncomfortable with this suggestion (cf. [20 (ch. 9), 26]. It is well known that one of the risks associated with the profit-center organization is that the computing center will resort to monopoly pricing practices. We demonstrate that the problem of monopoly pricing is further aggravated in the presence of queueing delays since a profit center may significantly reduce *both* the available capacity and the *relative utilization of this capacity*. Thus, the aversion to the profit-center organization may well be justified.

4. Computing center managers often face the accusation that low utilization ratios reflect inefficient usage of computing resources. It is easy to demonstrate that full utilization is undesirable; but data processing managers often have difficulty justifying utilization rates which are far below 100 percent. Our analysis may help the data processing manager determine and justify the level of system utilization. Our results demonstrate that, taking users' delay cost into account, seemingly low utilization ratios are often optimal.

The issue of "turnaround time" in the pricing of computer services is briefly discussed by Sharpe [23 (ch. 11)] and Jensen [13]. The general congestion-control problem has been discussed by Yechiali [30]. Dolan [6, 7] and Pick and Whinston [21] considered a dynamic priority-queueing mechanism that focuses on the incentive-compatibility issue: This mechanism induces users to reveal the true value of their priority index. Other relevant (but loosely related) studies are those by Greenberger [10], Marchand [18], Naor [19], and Yechiali [28, 29].

The plan of this article is as follows. In the next section we set up the framework for our analysis. Section 3 studies the price and capacity that maximize the expected nèt value of computer services to the organization as a whole, and considers the implications of a "free-access" policy. The budgetary implications of net-value maximization are studied in Section 4. The profit-center control structure is studied in Section 5, and our concluding remarks are offered in Section 6.

2. FRAMEWORK FOR ANALYSIS

In this section we set up the framework for our analysis. Any successful economic analysis must abstract from the complexities inherent in the operation of the actual system under study while capturing the major tradeoffs of interest. By focusing on the most important aspects, one obtains insights that can then be applied to the actual system. For example, when the economist describes the "demand for computing" by a continuous decreasing function of "quantity" (per unit of time), he automatically sacrifices some of the important features of actual computer systems for the sake of gaining insights that would not have been attainable otherwise. The degree of abstraction required is not significantly different from that in other areas where microeconomic analysis has achieved remarkable success.

We view the operation of the computer facility as follows. Jobs (or transactions) arrive into the system randomly; they spend a random amount of time in the system, and then depart. The time spent in the system consists of actual processing time and waiting delays. This applies to batch-processing systems, where one observes job (and output) queues in the traditional sense, and to time-sharing systems, where queues are *internal* and the response time includes a waiting component in addition to the processing time. To capture the variety of possible cases, we shall apply a general queueing framework without resorting to any specific architecture.

The standard microeconomic model applied by previous authors (cf. Sharpe [23], Cotton [2], Jensen [13]). summarizes users' valuations by a value function from which their demand curve can be derived. Letting q represent the "quantity of computing" per unit of time, conveniently represented by the number of standardized transactions (or jobs) processed per unit of time, the value function $V(q)$ represents the total value (to the organization as a whole) associated with q transactions per unit of time, and is obtained by aggregation over users' subsystems. The marginal value function represents a relationship between the price per transaction and the number of transactions per unit of time, which depicts the demand curve (see [23 (ch. 2)]): If users are charged a price of p per transaction, the number of transactions per unit of time will be given by the solution of

$$V'(q) = p. \tag{1}$$

How tangible is the value function $V(\cdot)$? In a business environment, quantifying the benefits of computer projects is a necessary part of the systems analysis process; the value function is obtained by aggregating these benefits (or values) over projects (see Sharpe [23 (ch. 2)]; Couger [3] and Kleijnen [15, 16] present comprehensive reviews of methods for quantifying the benefits of computer projects in various circumstances). Yet, there may be situations where quantifying the value function is impractical (consider, for example, a university environment). In such cases, an underlying value function still exists, but it is too difficult to measure. Clearly, qualitative results and insights obtained through modeling apply to both cases. Direct application of numerical results is feasible in the former case, while the latter may require an iterative "trial and error" process.[2]

We now extend the standard microeconomic approach to our queueing framework. It is natural to replace q by the arrival rate of transactions (or jobs) to the computer system. We follow the common assumption[3] that the times between consecutive arrivals to the system are independent identically distributed random variables with finite mean $1/\lambda$; λ is the *arrival rate* to the system (in the frequently-assumed special case where the arrival process is Poisson, λ is the Poisson rate), The arrival rate λ is affected by the value of computer services and the cost of using these services. The value of computer services is represented by the value function $V(\lambda)$, obtained (as in the standard deterministic case mentioned above) by aggregating the values of various user subsystems; $V(\lambda)$ is the expected gross value (per unit of time) corresponding to arrival rate λ. We adopt the usual assumption that the value function $V(\lambda)$ is twice differentiable and strictly concave.

Consider now the analog of the demand equation (1). We assume that users (or user departments) decide on the addition of work load (i.e., increasing the arrival rate by adding new application subsystems) by comparing expected added value to expected added cost (this assumes that aversion to the risk associated with uncertain turnaround time does not play an important role in the evaluation of computer projects and hence may be ignored). When the cost of delay is zero and the price per transaction is p, eq. (1) becomes $V'(\lambda) = p$ (i.e., expected marginal value = marginal user cost = price). However, when the cost of delay is positive, the expected user cost consists of actual payment (price) and expected delay cost. We assume that the delay cost is v per unit of time per job. That is, a user is willing to pay v monetary units for obtaining the processing results one time unit earlier. Now, consider the system in stationary state, and let W denote the expected time a job remains in the system from its arrival epoch until its processing is completed. Then, the expected delay cost per job is $v \cdot W$, and the expected user cost per job is the sum of the direct payment and the expected delay cost, $p + v \cdot W$. It follows that the analog of eq. (1) when delay costs are taken into account is given by

$$V'(\lambda) = p + v \cdot W. \tag{2}$$

Computer systems are typically modeled either as Markovian networks of queues or as special cases of the GI/G/s queueing system. To model variations in the scale (or capacity) of the system, consider first the special case of a Markovian network of queues. If the system contains N servers, we can write the service rate of server i as $\mu \cdot a_i$ ($i = 1, 2, 3, \ldots, N$), where μ is a scale parameter and a_i characterizes service station i. The parameter μ represents the service capacity of the system. When μ is doubled (say), the system can handle a flow of twice as many jobs per unit of time while maintaining the same (stationary) distribution of queue lengths. More generally, we assume that the distributions of service times have a common scale parameter μ which represents the capacity of the system. Thus, in the GI/G/s case, service times are distributed as H/μ where H is a random variable and μ is a parameter representing capacity. This specification determines μ up to a multiplicative constant. We fix μ as the expected output rate, measured in jobs (or transactions) per unit of time, when the system is fully utilized. (Thus, if the system is modeled as a Markovian network of queues, μ is the overall exit rate from the system when all servers are busy; in the special single-server case, μ is simply the service rate. For the GI/G/s model, μ is the number of servers s, divided by the expected service time.) Note that in this model, service times are measured in terms of elapsed time and hence are comparable to arrival rates; this is a convenient aggregation which allows us to study the queueing effects without depending on specific system details.[4] We denote the cost associated with capacity level μ by $C(\mu)$; following the treatment of value and demand as rates per unit of time, $C(\mu)$ is a cost rate per unit of time (thus, if the equipment has been purchased, its purchase price would be converted to the equivalent lease rate[5]).

To close the specification of our model, we now relate the arrival rate and system capacity to expected delay. Consider the system with arrival rate λ and capacity μ in its stationary state. Denote by L the expected number of jobs in the system, and recall that W denotes the expected time a job spends in the system. Let $\rho \equiv \lambda/\mu$ denote the relative utilization of the system. A change in the time scale changes both λ and μ so that ρ remains constant. We assume that L is invariant to such changes in the time scale, that is, we can write[6] $L \equiv L(\lambda, \mu) = f(\rho) = f(\lambda/\mu)$. This assumption will always hold when λ is a scale parameter of the interarrival-time distribution and μ is a common scale param-

[2] The "trial and error" process is discussed in the framework of our model in Section 6.

[3] In fact, even this general specification could be relaxed since we need only the conditions that imply Little's law (see, e.g., [11 (Section 11-3)]).

[4] An alternative would be to focus on a system bottleneck that leads to congestion (CPU, channel, etc.).

[5] An alternative is to consider the expected discounted value and cost throughout.

[6] Formally, $L(\lambda, \mu)$ is a function of two variables and $f(\rho)$ is a function of one variable. For example, in the case of the M/M/1 queue, $L(\lambda, \mu) = \lambda/(\mu - \lambda)$ and $f(\rho) = \rho/(1 - \rho)$. Note that $\partial L/\partial\lambda = f'(\rho)/\mu$.

eter of the distributions of service times. We further assume that the system satisfies Little's law,[7]

$$L = \lambda W, \tag{3}$$

that for any given capacity, W is a strictly increasing and differentiable function of λ, and that as the relative utilization ρ tends to unity, the expected number of jobs in the system, $L(\rho)$, tends to infinity.[8]

Little's law, eq. (3), has an interesting interpretation in the context of our discussion. Recall that v is the cost of delay per job per unit of time. If the expected delay cost of a job is "charged" to it upon arrival, then (since the arrival rate is λ) the expected delay cost to be "charged" to jobs joining the system in one time unit is $\lambda \cdot vW$. If delay costs were "charged" continuously over time, the delay cost incurred by the system per unit of time would be v times the number of jobs present, and its expected value would be $v \cdot L$. Thus, eq. (3) intuitively represents the equivalence of these two "charging" conventions. Furthermore, we have demonstrated that the expected delay cost incurred by system users per unit of time is given by $v \cdot L = v \cdot \lambda W$.

Little's law, eq. (3), allows us to represent the *marginal* delay cost in a particularly useful form. Fix the capacity of the system at μ, and differentiate eq. (3) with respect to λ; this yields

$$v \cdot (\partial L/\partial\lambda) = v \cdot W + v \cdot \lambda \cdot (\partial W/\partial\lambda). \tag{4}$$

While our analysis requires only the mathematical validity of eq. (4), it is useful to interpret it as a decomposition of the marginal delay cost into a "self-regulating" term and an "externality" term. First, the left-hand side of eq. (4) represents marginal delay cost (with respect to λ): the expected overall delay cost per unit of time was shown to be $v \cdot L$, and its derivative is $v \cdot (\partial L/\partial\lambda)$. Now, an (infinitesimal) increase in λ increases the expected overall delay cost in two ways:

1. The added jobs themselves incur a delay cost which, to a first-order approximation, has expected value $v \cdot W$ per job; we call this term of eq. (4) the "self-regulating" term since it represents a cost inflicted on the user that generated the added work load.

2. The added work load increases the expected delay cost inflicted on all current users by $v \cdot \lambda \cdot (\partial W/\partial\lambda)$ per added job (to the first order).[9] We call this term the "externality" term: When a user adds work load to the system, he inflicts a cost on all other users by increasing their overall expected delay. Unlike the "self-regulating" term, this cost component is not perceived by the generating user (since it is inflicted on others), and hence it represents an externality.

The decision problem of the data processing manager facing these demand and cost functions can be separated into two related subproblems: First, for a given system configuration with fixed capacity μ, he has to decide how to allocate the limited resources among competing users; this problem is known as the short-run (or fixed-capacity) problem. Second, he has to decide on the optimal scale (or capacity) of the computing facility; this problem is known as the long-run (or variable-capacity) problem.

In the fixed-capacity (short-run) problem, μ is given and the manager sets a price p which determines the arrival rate λ through eq. (2) (note that W in eq. (2) depends on λ). The expected gross benefit per unit of time (to the organization as a whole) is given by $V(\lambda)$; the expected delay cost per unit of time is equal to $v \cdot L$ ($= v \cdot \lambda W$, by eq. (3)). In the long-run (variable-capacity) problem, μ becomes variable and has to be determined by balancing the expected benefits of increasing capacity against the capacity cost. Clearly, the approach taken by the data processing manager in solving these problems depends on the objective function being maximized. The first objective considered in the following section is maximizing the expected net value of computing services to the organization as a whole.

3. NET-VALUE MAXIMIZATION

In this section we derive the price and capacity that maximize the expected net value of data processing services to the organization as a whole (cf. [23 (ch. 1)], for a general discussion of this objective). We first consider the short-run problem, where processing capacity μ is fixed. Since μ is fixed, the cost of capacity may be ignored; the price p uniquely determines the arrival rate λ so the net-value maximization problem may be written as

$$\max_{\lambda}\{V(\lambda) - v \cdot L\}, \tag{5}$$

where $L = L(\lambda, \mu)$ and μ is fixed.

The first term in eq. (5) is value per unit of time; the second term, $v \cdot L$, is the expected aggregate delay cost per unit of time (= expected number of jobs in system × delay cost per job per unit of time). Note that unlike the usual deterministic formulation, the capacity constraint $\lambda \leq \mu$ need not appear explicitly in eq. (5). since its violation would imply an infinite delay cost. The first-order condition for maximizing eq. (5) is

$$V'(\lambda) = v \cdot (\partial L/\partial\lambda). \tag{6}$$

The arrival rate λ_0 obtained from eq. (6) equates marginal value to marginal delay cost. To find the price that induces this arrival rate, we substitute the price-arrival rate relationship of eq. (2) into eq. (6) to obtain

$$p = v \cdot (\partial L/\partial\lambda) - v \cdot W. \tag{7}$$

Using eq. (4) to decompose the marginal delay cost $v \cdot (\partial L/\partial\lambda)$, we obtain

$$p = v \cdot \lambda \cdot (\partial W/\partial\lambda), \tag{8}$$

[7] Little's law is a central result in queueing theory that holds under very general conditions. The reader is referred to [11 (Section 11-3)] for a detailed discussion. For our purposes, it is sufficient to point out that both the GI/G/s queueing system and Markovian networks of queues satisfy Little's law.

[8] This assumption is not needed for the analysis, but due to its common applicability it will be followed in the graphic representations and intuitive interpretations.

[9] To see this, note that increasing λ to $\lambda + \Delta\lambda$ increases W to $W + \partial W/\partial\lambda \cdot \Delta\lambda + o(\Delta\lambda)$; thus, the expected added delay cost inflicted on current users is $v \cdot \lambda \cdot \partial W/\partial\lambda \cdot \Delta\lambda + o(\Delta\lambda)$ per unit of time.

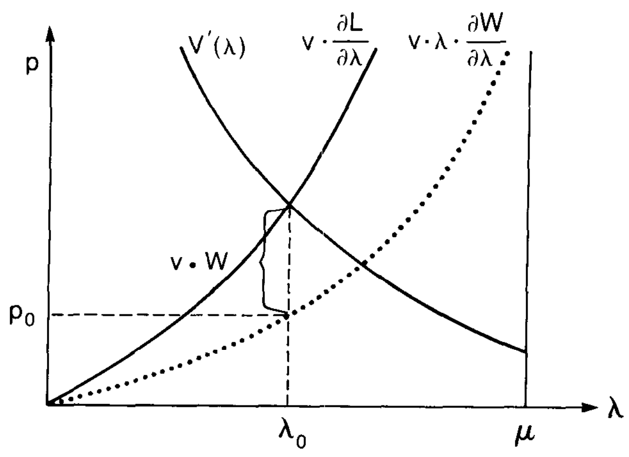

FIGURE 1. The Short-Run Pricing Problem in the Presence of Delay Costs.

where the right-hand side of eq. (8) is evaluated at $\lambda = \lambda_0$. We have thus shown that the net-value maximizing price is equal to the "externality" term of the marginal delay cost decomposition, eq. (4), (representing the expected marginal delay cost inflicted on the rest of the system), evaluated at $\lambda = \lambda_0$.

The solution is demonstrated graphically in Figure 1. The optimal arrival rate, λ_0, is obtained by intersecting the marginal value curve depicting $V'(\lambda)$ with the marginal delay cost curve, depicting $v \cdot (\partial L/\partial\lambda)$ (see eq. (6)). However, the price leading to $\lambda = \lambda_0$ is *not* read off the point of intersection of these two curves since each job already incurs an expected delay cost of $v \cdot W$. Thus, to avoid double charging (once by waiting and one more time through the monetary payment), the payment to the computing center has to be reduced by $v \cdot W$ (see eq. (7)). This gives rise to the price $p_0 = v \cdot \lambda \cdot (\partial W/\partial\lambda)$ which is read off the curve depicting the "externality" term (the dotted curve in Figure 1) at $\lambda = \lambda_0$.

It is instructive to compare the net-value maximizing result to the result obtained (for the same system configuration) under a "free-access" policy. Such a policy, which is based on the notion that computing is a "priceless" good, has been implemented in the computing centers of several academic institutions, notably Dartmouth College (see [17]).[10] One reason that such a system may be attractive is its "self-regulating" nature: When the system becomes overloaded, service quality deteriorates. Consequently, users are discouraged and they reduce their work load. We now examine the relationship between the "self-regulating" result and the net-value maximizing result.

Clearly, access under the "free-access" policy is not completely free since users still incur waiting delay costs which restrict their utilization of the computing facility. The "free access" arrival rate λ_F is obtained by setting $p = 0$ in eq. (2); thus, λ_F is the solution of

$$V'(\lambda_F) = v \cdot W. \tag{9}$$

How is λ_F related to the net-value maximizing arrival rate, λ_0? The "self-regulating" cost term included in eq. (9) is strictly lower than the overall marginal delay cost (see eq. (4)); since $V'(\lambda)$ is strictly decreasing, this implies $\lambda_F > \lambda_0$, as might be expected.[11] Thus, we have shown that the "free-access" policy leads to an over-congested system,[12] and "self-regulation" has to be augmented by explicit pricing to the benefit of the organization as a whole.

Next, we address the long-run problem, where capacity μ is variable. The problem now is to find the optimal scale of the computing facility where the cost of capacity, $C(\mu)$, is taken into account. This problem is combined with that of finding the price p, or equivalently the arrival rate λ, that will optimally utilize this capacity. The variable-capacity problem can be formulated as

$$\max_{\lambda,\mu}\{V(\lambda) - v \cdot L(\lambda, \mu) - C(\mu)\}.$$

The first-order conditions are

$$V'(\lambda) = v \cdot (\partial L/\partial\lambda) \tag{10a}$$

and

$$C'(\mu) = -v \cdot (\partial L/\partial\mu). \tag{10b}$$

Recalling that $L(\lambda, \mu) = f(\lambda/\mu)$, we have $\partial L/\partial\lambda = (1/\mu)f'(\lambda/\mu)$ and $\partial L/\partial\mu = -(\lambda/\mu^2)f'(\lambda/\mu)$; factoring out $f'(\lambda/\mu)$, we obtain

$$C'(\mu) = (\lambda/\mu) \cdot V'(\lambda). \tag{11}$$

It is instructive to compare this solution to that of the usual deterministic case, where queueing delays are absent. In the deterministic case, capacity is expanded up to the point where marginal value equals marginal capacity cost, and the system operates at full utilization. In the stochastic case, the queueing delays imply the optimality of excess capacity, and the relative utilization $\rho = \lambda/\mu$ is less than unity. Since $\rho < 1$, the marginal capacity cost is not equated to marginal value, but rather to marginal value × relative utilization.

To gain further insight to the factors affecting the determination of capacity and capacity utilization, we consider a simple example. Assume that the cost of capacity is linear:

$$C(\mu) = A + b \cdot \mu, \tag{12}$$

[10] As might be expected, the "free-access" policy as described by Luehrmann and Nevison [17] has been completely modified since its inception.

[11] For a similar result on the difference between "self-optimization" and "overall optimization" in a different context, see Yechiali [28, 29].

[12] While Luehrmann and Nevison [17] advocate the use of the "free-access" policy, they use data which may (indirectly) demonstrate this point. They compare [17 (p. 958)] their system with a comparable system at the Massachusetts Institute of Technology with the following results:

	Dartmouth	M.I.T.
CPU time (hours/month)	482	375
Terminal hours (hours/day)	1000	about 300

Assuming (as they do) that the systems indeed possess similar characteristics, an increase of about 30 percent in actual processing (for which CPU time is a proxy; this number is also consistent with the estimate of the increase in demand at Dartmouth, see [17 (p. 957)]) has increased the average delay by a factor of about 2.5! (In the absence of externalities, the number of terminal hours at Dartmouth would have been about (482/375) × 300).

where the constant A represents fixed overhead costs and the marginal capacity cost is b. Let the demand function belong to the class of isoelastic demand functions:

$$V'(\lambda) = k/\lambda^{\alpha}. \tag{13}$$

This corresponds to a value function of the form $V(\lambda) = k \cdot \lambda^{1-\alpha}/(1-\alpha)$. We require $0 < \alpha \leq 1$ to guarantee that value is positive; the unit-elasticity case $\alpha = 1$ corresponds to the logarithmic value function, $V(\lambda) = k \cdot \ln \lambda$. Now, eq. (10a) and eq. (11) read

$$k = v \cdot (\lambda^{\alpha}/\mu) \cdot f'(\lambda/\mu) \tag{14a}$$

and

$$k = b \cdot \mu \cdot \lambda^{\alpha-1}. \tag{14b}$$

These equations may be solved easily to yield the optimal values of λ and μ. In the unit-elasticity case ($\alpha = 1$), we can substitute $\rho = \lambda/\mu$ and obtain the separable equations

$$k = v \cdot \rho f'(\rho) \tag{15a}$$

and

$$k = b \cdot \mu. \tag{15b}$$

Equation (15b) determines optimal capacity at $\mu = k/b$; here, capacity is invariant to the specific queueing characteristics, and depends only on the demand characteristics and capacity cost. Furthermore, the demand for capacity is the same as the underlying demand function. Equation (15a) determines the optimal utilization ratio, ρ, which depends on the delay and demand characteristics, and is independent of capacity costs. This separation, which prevails in the unit-elasticity case, allows the data processing manager to set capacity without regard to the queueing characteristics and to take them into account when setting price (which determines the utilization of the system).

Note that by eq. (15a), the optimal utilization ratio ρ depends on the relative marginal values of "time" versus "computing": An increase in the relative value of users' time, v/k, induces lower utilization. The behavior of the optimal utilization ratio ρ as a function of v/k clearly depends on the queueing characteristics of the system; to study this behavior for a range of queueing characteristics, we examined it for the family of $M/E_r/1$ models, where the service time distribution is Erlang with r degrees of freedom. The Erlang family of distributions is often used to model computer service times, and includes the exponential case as a special case ($r = 1$) and the deterministic case as a limiting case ($r \rightarrow \infty$). The expected number of jobs in the system in this model is given by the Pollaczek–Khintchine formula (cf. [11 (p. 251)]), which reduces to

$$L = f(\rho) = \rho + (1 + (1/r))(\rho^2/2(1-\rho))$$

for the $M/E_r/1$ model. Now, the solution of eq. (15a) which yields the optimal utilization ρ as a function of v/k for different values of r is straightforward. As it turns out, the graph of ρ as a function of v/k is not very sensitive to r; in Figure 2 we demonstrate the results for the exponential case (solid line), the deterministic case (dotted line), and the Erlang case with $r = 4$ (clearly, all $M/E_r/1$ models are bounded by the exponential and deterministic curves). As one might expect, optimal utilization is a decreasing function of v/k. Very low values of v/k are typical of production-oriented batch systems; in these cases optimal utilization is relatively high but usually significantly less than 100 percent (e.g., 75 percent for $v/k = 1/12$). This follows from the very fast decline of the optimal utilization, ρ, as a function of v/k, in the neighborhood of $\rho = 1$. When v/k becomes higher (which is the typical situation in the case of time-sharing and on-line systems), ρ goes down less sharply. When v is equal to k (which is a reasonable order of magnitude for on-line systems), optimal utilization goes down to about 38 percent. Our results then indicate that even seemingly low utilization ratios are consistent with an overall optimum, especially when time is costly. Significant excess capacity is not necessarily an indication of inefficiency: It is a necessity if turnaround time is to be kept low.

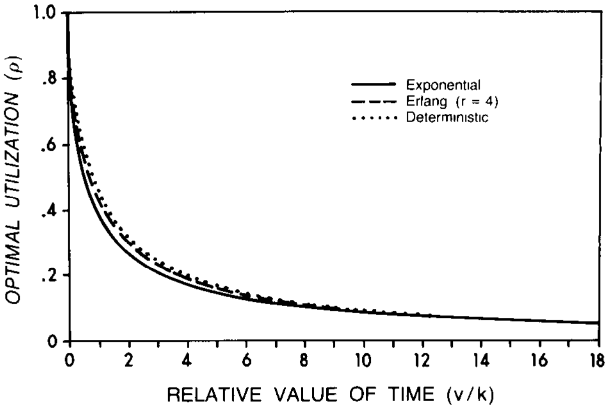

FIGURE 2. Net-Value Maximizing Utilization, ρ, as a Function of the Relative Value of Users' Time, v/k, for Three Models: (1) M/M/1 (solid line); (2) $M/E_4/1$ (dashed line); (3) M/D/1 (dotted line).

4. COST RECOVERY

One of the widely accepted methods for controlling and evaluating the computing center is through its budget. When the long-run average cost is constant and no externalities are present, net-value maximization implies a balanced budget. To see this, let $C(\mu) = b \cdot \mu$ and ignore the queueing delays (i.e., let $v = 0$). Then, it is easy to show that at the optimum, the system is fully utilized and revenue equals cost, that is, the pricing problem is solved by requiring the computing center to balance its budget. This result is the basis for the standard cost-allocation procedure (cf. [12]), and is widely recommended (cf. Sobczak [24]) and used (cf. [8, 20]) by computing centers.

What happens to this result when queueing effects are taken into account? Here, our net-value maximizing condition, eq. (11), implies $\mu \cdot b = \lambda \cdot V'(\lambda)$ and, substituting $V'(\lambda)$ from eq. (2), we obtain $\mu \cdot b = \lambda \cdot [p + v \cdot W]$. Using Little's law, eq. (3), we thus obtain

$$\mu \cdot b = \lambda \cdot p + v \cdot L. \tag{16}$$

The left-hand side of eq. (16) is the expected cost incurred by the computing center per unit of time. On the right-hand side, $(\lambda \cdot p)$ is expected revenue per unit of time, equal to the product of the expected number of jobs arriving per unit of time, λ, by the price per job p; $v \cdot L$ is the expected total delay cost incurred in the aggregate per unit of time. Thus, eq. (16) may be written as

expected cost

= expected revenue + expected delay cost.

The budget of the computing center will *not* be balanced due to the "tax" represented by the delay cost, a "tax" which is paid by users but not collected by the computing center. Requiring the computing center to balance its budget will not lead to an overall optimum, but will rather create incentives leading *away* from the optimal solution.

This result has important policy implications for the management and control of data processing. It suggests that, even under the standard assumptions made by cost accountants (cf. [12]),[13] the widely used standard cost-accounting procedures are misleading when applied to computing centers, due to the importance of queueing effects. Our results[14] suggest that cost-recovery is an undesirable maxim, which leads to underutilization of available resources. Evaluating the data processing manager on the basis of cost recovery is inappropriate; a data processing manager who maximizes the expected net value of computer services to the organization as a whole will report (under the standard conditions) a deficit equal to the magnitude of the delay "tax" imposed on the aggregate user population.

5. PROFIT-CENTER PRICING

Computer centers are often organized as sovereign *profit centers*. Under this control structure, the computing center sets the prices of computing services, basing them on the profitability of providing these services. Then, the computing center operates as an expected-profit maximizer, facing a downward-sloping demand curve. It is well known that such a "monopoly pricing" scheme reduces output and increases prices when compared to the net-value maximizing scheme. In this section we analyze the behavior of a computing center which maximizes its own expected profits in the presence of queueing delays. Clearly, our results also apply to the case of an independent expected-profit-maximizing firm which sells computer services in the marketplace and faces a downward-sloping demand curve.

Given that the cost of delay is incurred by users and does not appear as a cost item in the profit-and-loss statement of the profit center, will the data processing manager have an incentive to take it into account? The answer is positive since users' decisions to utilize the system depend on their costs, which include both price and delay cost. When users face a price p and an expected delay cost of $v \cdot W$ per job, the corresponding arrival rate is given by eq. (2); increasing W reduces the price which can be charged at each level of λ and, consequently, the profits of the computing center.

Let us examine the expected revenues and costs of the computing center when the arrival rate is λ and capacity is μ. The expected cost of capacity per unit of time is, obviously, $C(\mu)$. The expected revenue per unit of time is equal to the product of the expected number of jobs λ by the price per job p. Now, using eq. (2), $p = V'(\lambda) - v \cdot W$, hence the expected revenue per unit of time is equal to $\lambda \cdot p = \lambda \cdot [V'(\lambda) - v \cdot W]$. As noted above, although the delay cost is incurred by users, it affects the revenues of the computing center since it increases the effective cost perceived by users, and hence is quite relevant to the pricing decision. It follows that the expected net-profit rate of the computing center per unit of time is given by

$$\begin{aligned}\lambda \cdot [V'(\lambda) \; - v \cdot W - C(\mu)\\ = \lambda \cdot V'(\lambda) - v \cdot L(\lambda, \mu) - C(\mu).\end{aligned} \tag{17}$$

Clearly, the short-run (or fixed-capacity) problem is similar to the one studied in Section 3; the only difference is that now the value function $V(\lambda)$ is replaced by the "revenue" function,[15] $\lambda \cdot V'(\lambda)$, and marginal value is replaced by "marginal revenue," which is given by

$$\mathrm{MR}(\lambda) = (d/d\lambda)[\lambda V'(\lambda)] = V'(\lambda) + \lambda V''(\lambda) < V'(\lambda)$$

(since $V'' < 0$). The solution is obtained as in Figure 1, with $\mathrm{MR}(\lambda)$ replacing $V'(\lambda)$. Since $\mathrm{MR}(\lambda) < V'(\lambda)$, the profit-center organization brings about a higher price and lower utilization than the net-value maximizing solution. These results are consistent with the usual implications of monopoly pricing schemes.

Consider next the long-run problem. The first-order conditions for maximizing eq. (17) are

$$V'(\lambda) + \lambda V''(\lambda) = (v/\mu) \cdot f'(\lambda/\mu) \tag{18a}$$

and

$$C'(\mu) = v \cdot (\lambda/\mu^2) \cdot f'(\lambda/\mu), \tag{18b}$$

which imply

$$C'(\mu) = (\lambda/\mu) \cdot [V'(\lambda) + \lambda V''(\lambda)]. \tag{19}$$

In the absence of queueing delays, profit-center pricing leads to lower capacity, a lower usage rate, and the same relative utilization (unity) when compared to net-

[13] These assumptions involve linearity of the cost functions and stationarity of demand and cost. Existing criticisms of these procedures focus on these two assumptions.

[14] Our results hold whenever the capacity cost function $C(\mu)$ is linear or concave (thus allowing for fixed overhead or for economies of scale).

[15] More precisely, $\lambda \cdot V'(\lambda)$ is the expected revenue in the absence of delay costs.

value maximization. In the stochastic case, the results depend on the parameters of the problem, but if both $\mu \cdot C'(\mu)$ and $\lambda \cdot \mathrm{MR}(\lambda)$ are increasing functions, the profit center will select a lower capacity and induce a lower arrival rate λ and lower utilization ratio $\rho = \lambda/\mu$. The reduction in the relative utilization ratio means that *the service quality* (measured by the queueing delays) *provided by the profit center will be better than is optimal* for the organization as a whole (in the net-value maximizing sense). The reason for this behavior is that by lowering the load on the system, the average delay goes down and consequently users are willing to pay a higher price and increase the profits of the computing center (recall that by eq. (2), the delay cost reduces the price the computing center can charge).

To demonstrate these effects, we study the examples of the linear cost function given by eq. (12) and constant-elasticity demand function given by eq. (13) for $0 < \alpha < 1$, (When $\alpha = 1$, no optimal solution exists since any proportionate reduction of λ and μ will reduce the capacity cost without altering the other components of eq. (17)). Now, eq. (18a) and eq. (19) read

$$k(1 - \alpha) = v \cdot (\lambda^{\alpha}/\mu) \cdot f'(\lambda/\mu) \tag{20a}$$

and

$$k(1 - \alpha) = b \cdot \mu \cdot \lambda^{\alpha-1}. \tag{20b}$$

Equations (20), which correspond to the price-setting computing center, may be obtained from eq. (14) by replacing k with $k(1 - \alpha)$: The effect of computing center self-profit maximization is equivalent to "discounting" users' (net) valuations of computing services by a factor of $(1 - \alpha)$. The more inelastic the demand curve, the greater is α and the more effective is this "discounting," Intuitively, this follows since the computing center's "monopoly power" increases as demand becomes more inelastic.

As a numerical illustration, consider the case where $\alpha = 1/2$. The net-value maximizing utilization solves

$$k^2 = v \cdot b \cdot f'(\rho) \tag{21a}$$

and the optimal capacity is given by

$$\mu = k^2\rho/b^2. \tag{21b}$$

If the computing facility operates as a profit center, we have

$$k^2 = 4v \cdot b \cdot f'(\rho) \tag{22a}$$

$$\mu = k^2\rho/(4b^2). \tag{22b}$$

Clearly, eq. (22a) and eq. (22b) imply lower utilization, lower capacity, and lower arrival rate than eq. (21), in accordance with the more general results. As one might expect, the optimal utilization ρ increases with k (i.e., as the value of the services goes up) and decreases as the value of users' time goes up, both under net-value maximization and under computing center profit maximization. Unexpectedly, utilization goes down as the capacity cost, b, increases.[16]

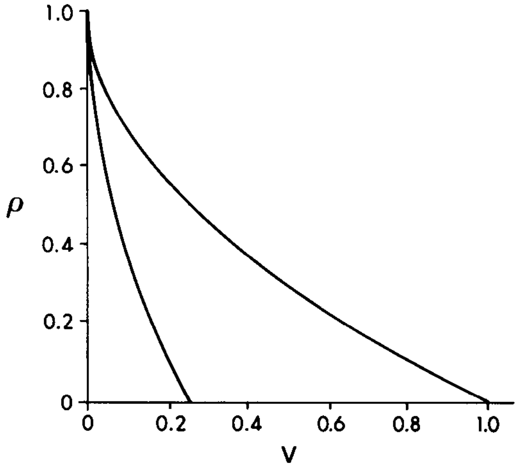

FIGURE 3. Utilization Ratio, ρ, as a Function of the Value of User Time, v, for the M/M/1 Queueing System with $\alpha = 1/2$, $k = 1$, $b = 1$. The upper graph corresponds to the net-value maximizing solution; the lower graph corresponds to the profit-center solution. The latter case leads to significantly lower utilization ratios.

The behavior of the utilization ratio ρ as a function of the value of users' time v is shown in Figure 3 for the M/M/1 queueing system (the behavior for the $M/E_r/1$ system is similar). The figure compares the net-value maximizing solution, $\rho = 1 - \sqrt{v \cdot b/k^2}$, to the profit-center maximizing solution, $\rho = 1 - 2\sqrt{v \cdot b/k^2}$, for a fixed set of parameters ($k = 1$, $b = 1$). As v increases from zero to unity, ρ goes down from full utilization to zero utilization. Figure 3 clearly demonstrates the effect of "overstating" the value of users' time under the profit-center structure. This effect is quite significant, and the resulting reduction in the arrival rate is even more significant since $\lambda = \rho \cdot \mu$, hence the reduction in ρ further amplifies the reduced capacity.

It is often suggested that "the profit-center approach is probably a superior means for motivating the management of the computer facility" since "not only is there motivation to hold costs down, but also to provide quality services" [4 (p. 48)]. In a business environment, however, data sharing, as well as security and integration requirements, inhibit the use of outside facilities and enhance the monopoly power of the corporate computing center. Our results demonstrate that the presence of queueing effects significantly aggravates the problem of monopoly pricing since the profit-center structure creates incentives to significantly reduce *both* the available capacity and its relative utilization. For example, comparing eq. (21) and eq. (22), we note that for any given relative utilization ρ, the capacity under the profit-center structure will be one quarter of the net-value maximizing capacity. Furthermore, the rela-

[16] This follows from the fact that the value function increases only as $\sqrt{\lambda}$ while the capacity cost is linear in μ. Thus, as the marginal capacity cost goes up, λ should decline by proportionately more than μ to maintain the balance.

tive utilization of the (lower) capacity of the profit center will be significantly below the net-value maximizing level of utilization. Thus, our results suggest that while a profit-center structure could provide high quality services by significantly reducing expected turnaround time, this is achieved by reducing quantity to a level which is far below the net-value maximizing level. Thus, the reluctance to organize the computer center as a profit center, manifested, for example, by Nolan's [20] findings, may well be justified.

6. CONCLUDING REMARKS

This article has incorporated queueing effects to the microeconomics of computing center management. Although one might expect these effects to result in some minor modifications to the usual pricing methodology, we have shown that in fact the queueing effects are quite important, both quantitatively and qualitatively. Our results suggest a methodology that could help guide the data processing manager in setting price and capacity, and could be useful in presenting his case to higher-level management. Studying the interrelationships between the financial responsibility assigned to the data processing manager and the effects of users' delay cost on price, capacity, and utilization, we demonstrate that some common maxims can be misleading. In particular, we suggest that maximizing the net value of computer services to the organization as a whole leads (under standard assumptions) to a deficit, and that cost recovery will lead to under-utilization of available resources. We demonstrate that organizing the computing center as a profit center leads to a significant under-investment in capacity, coupled with significant under-utilization of this capacity; in effect, the profit center "overstates" the value of users' time to the detriment of the organization as a whole. We demonstrate the determination of the optimal level of system utilization, and suggest that given the present magnitude of hardware costs, optimal utilization may be well below levels that were considered reasonable in the past.

Our results offer an analytic methodology for setting price and capacity; the implementation of this methodology deserves further discussion. Our methodology is well-suited to business data processing systems, which are characterized by regularly scheduled tasks which undergo explicit cost-benefit analysis as part of the system development process (see [3, 14, 16]). Further, in such systems the value of users' time is directly measurable by the wage rate. The remaining data required for direct application of our methodology consists of three subsets: usage data, queueing-related data, and cost data. Usage data is available in all existing installation accounting systems (cf. [9]); it is a prerequisite for any pricing scheme. Queueing-related data, including utilization and system response to varying utilization, is collected and technically evaluated by installation managers on a routine basis as part of the capacity planning process (cf. [1]). This data is obtained in part by processing installation accounting data, and in part by direct monitoring; Rose [22] provides a practical review of methods in use. Cost data is clearly available. Thus, in the well-organized data processing department, our methodology can be applied directly by processing existing data.

The situation is markedly different in educational/research installations, where the implementation of any pricing scheme is difficult due to the unpredictable nature of demand, the difficulty of estimating delay costs, the heterogeneity of the user population, and the use of "funny money" budgets. Assuming that computer services are priced, the price does not involve "funny money," and the value of time may be estimated, demand information may still be hardly available, implying that the marginal value curve of Figure 1 may be unknown. In fact, demand information may well be rather "soft" even in the business data processing center. In this situation, price may be determined using an iterative "trial and error" process: A price p_1 is set, leading to arrival rate λ_1. As the usage of the system stabilizes, $V'(\lambda_1)$ may be estimated by measuring the expected delay and using eq. (2). Further, the queueing characteristics of the system yield $\partial L/\partial\lambda$. Now, if $V'(\lambda_1)$ is significantly greater than the marginal delay cost, price should be reduced; if it is significantly below the marginal delay cost, price should be increased. This process may be repeated until the desired balance is achieved.

The basic model studied in this paper can be easily expanded to take into account some additional features of computer systems.[17] One such feature is the problem of discontinuous costs, which results from the inability to acquire fractions of computers, channels, etc. The analysis in Jensen [13] (see also Sharpe [23 (pp. 455–459)]) applies here as well.

An important feature that may be taken into account is the cyclical pattern of demand over time. For example, the demand function which prevails overnight is usually different from the daily demand function, while the capacity of the system is fixed. This implies that our model has to be embedded in a standard peak-load pricing model (see, e.g., [25, 27]). The resulting combined model is straightforward, and leads to results which are qualitatively similar to the ones presented in this article.[18]

APPENDIX

Peak-Load Pricing

This Appendix demonstrates the extension of our methodology to the standard peak-load pricing framework developed by Steiner [25] and Williamson [27]; this methodology is designed to study the problem of cyclical changes in demand while capacity is fixed. Consider, for example, a workday that consists of two

[17] A feature which is not easily taken into account is the finite number of potential system users.

[18] The appendix briefly demonstrates this extension.

shifts, 1 and 2 (the generalization to the case of more periods is straightforward). Let $V_i(\lambda_i)$ be the period-i value function, and let v_i be the value of users' time in period (shift) i. Let α_i be the fraction of the workday during which the system is in period i. Since capacity does not vary from period to period, there is one capacity parameter μ; let $C(\mu)$ be the cost of capacity per unit of time.

Consider the variable-capacity net-value maximizing problem. We assume away all interperiod transitory effects. Then, the long-run problem is

$$\max_{\lambda_1,\lambda_2,\mu}\left\{\alpha_1\left[V_1(\lambda_1) - v_1 \cdot f\left(\frac{\lambda_1}{\mu}\right)\right] + \alpha_2\left[V_2(\lambda_2) - v_2 \cdot f\left(\frac{\lambda_2}{\mu}\right)\right] - C(\mu)\right\}$$

which gives rise to the first-order conditions

$$V_1'(\lambda_1) = (v_1/\mu) \cdot f'(\lambda_1/\mu) \tag{A1}$$

$$V_2'(\lambda_2) = (v_2/\mu) \cdot f'(\lambda_2/\mu) \tag{A2}$$

$$\alpha_1\lambda_1 V_1'(\lambda_1)/\mu + \alpha_2\lambda_2 V_2'(\lambda_2)/\mu = C'(\mu). \tag{A3}$$

Equation (A1) and eq. (A2) are analogous to eq. (10a) (noting that $\partial L/\partial\lambda = f'(\lambda/\mu)/\mu$), while eq. (A3) is analogous to eq. (11). Equations (A1)–(A2) demonstrate that once capacity has been determined, the procedure of Figure 1 may be applied in each period separately to determine price (and utilization). Equation (A3) is a straightforward generalization of eq. (11) for determining the capacity of the system.

Using this analogy, our qualitative results carry over to the peak-load pricing model. As an example, we demonstrate the derivation of the budget equation (see Section 4). Letting $C(\mu) = b \cdot \mu$ and substituting $V_i'(\lambda_i) = p_i + v_i \cdot W_i$ for $i = 1, 2$ in eq. (A3), we obtain

$$\mu \cdot b = \alpha_1\lambda_1(p_1 + v_1W_1) + \alpha_2\lambda_2(p_2 + v_2W_2)$$

or

$$\mu \cdot b = \alpha_1\lambda_1 p_1 + \alpha_2\lambda_2 p_2 + \alpha_1 v_1 L_1 + \alpha_2 v_2 L_2 \tag{A4}$$

which is clearly analogous to eq. (22) and states that, as in Section 4,

expected cost

= expected revenue + expected delay cost.

Thus, the discussion of Section 4 follows. In particular, full computing center cost recovery is undesirable.

Acknowledgments. The author acknowledges helpful comments and suggestions by Michael C. Jensen and the referees.

REFERENCES

1. Bronner, L. Overview of the capacity planning process for production data processing, *IBM Syst. J. 19*, 1 (1980), 4–27.
2. Cotton, I. W. Microeconomics and the market for computer services. *Comput. Surv. 7*, 2 (June 1975), 95–111.
3. Couger, J.D. Techniques for estimating system benefits. In *Advanced System Development/Feasibility Techniques*, Couger, Colter and Knapp, Eds., Wiley, N.Y., 1982. pp. 489–499.
4. Cushing, B.E. Pricing internal computer services: The basic issues. *Manage. Account. 57*, 4 (Apr. 1976), 47–50.
5. Dearden, J., and Nolan, R.L. How to control the computer resource. *Harvard Bus. Rev. 51*, 6 (Nov.–Dec. 1973), 68–78.
6. Dolan, R.J. Priority pricing models for congested systems. Ph.D. dissertation, Graduate School of Management, University of Rochester, N.Y., 1976.
7. Dolan, R.J. Incentive mechanisms for priority queueing problems. *Bell J. Econ. 9*, 2 (1978), 421–436.
8. Drury, D.H. A survey of data processing chargeback practices. *Infor. 18*, 4 (Nov. 1980), 342–353.
9. Gladney, H.M., Johnson, D.L., and Stone, R.L. Computer installation accounting, *IBM Syst. J. 14*, 4 (1975), 314–339.
10. Greenberger, M. The priority problem and computer time-sharing. *Manage. Sci. 12*, 7 (1966), 888–906.
11. Heyman, D.P., and Sobel, M.J. *Stochastic Models in Operations Research, Volume I: Stochastic Processes and Operating Characteristics*, McGraw-Hill, N.Y., 1982.
12. Horngren, C.T. *Cost Accounting: A Managerial Emphasis*, 5th ed., Prentice-Hall, Englewood Cliffs, N.J., 1982.
13. Jensen, M.C. Economics of management of university computing resources. Working Paper, Graduate School of Management, University of Rochester, N.Y., Apr. 1977.
14. Kleijnen, J.P.C. *Computers and Profits: Quantifying Financial Benefits of Information*, Addison-Wesley, Reading, Mass., 1980.
15. Kleijnen, J.P.C., and Van Reeken, A.J. Principles of computer charging in a university-type organization. *Commun. ACM 26*, 11 (Nov. 1983), 926–932.
16. Kleijnen, J.P.C. Quantifying the benefits of information systems. *Eur. J. Oper. Res. 15*, 1 (1984), 38–45.
17. Luerhmann, A.W., and Nevison, J.M. Computer use under a free-access policy. *Science 184*, 4140 (May 1974), 957–961.
18. Marchand, M.G., Priority pricing. *Manage. Sci. 20*, 3 (1974), 1131–1140.
19. Naor, P. On the regulation of queue size by levying tolls. *Econometrica 37*, 1 (1969), 15–24.
20. Nolan, R.L. *Management Accounting and Control of Data Processing*, National Association of Accountants, N.Y., (1977).
21. Pick, R.A., and Whinston, A.B. An internal computer charging mechanism for revealing user preferences. Working Paper, Krannert School of Management, Purdue University, West Lafayette, Ind., 1982.
22. Rose, C.A. A measurement procedure for queueing network models of computer systems. *Comput. Surv. 10*, 3 (1978), 263–280.
23. Sharpe, W.F. *The Economics of Computers*. Columbia Univ. Press, N.Y., 1969.
24. Sobczak, J.J. Pricing computer usage. *Datamation 20*, 2 (Feb. 1974), 61–64.
25. Steiner, P.O. Peak loads and efficient pricing. *Q. J. Econ. 71*, 4 (1957), 585–610.
26. Wheelock, A.R. Service or profit center? *Datamation 28*, 5 (May 1982), 167–176.
27. Williamson, O. Peak load pricing and optimal capacity under indivisibility constraints. *Am. Econ. Rev. 56*, 4 (1966), 810–827.
28. Yechiali, U. On optimal balking rules and toll charges in the GI/M/1 queueing process. *Oper. Res. 19*, 2 (1971), 348–370.
29. Yechiali, U. Customers' optimal joining rules for the GI/M/s queue. *Manage. Sci. 18*, 7 (1972), 434–443.
30. Yechiali, U. How long will you wait for what you really want? In *Proceedings of the Conference on Stochastic Control and Optimization*, (Free University, Amsterdam, Apr. 5–6, 1979).

CR Categories and Subject Descriptors: K.6.2 [**Management of Computing and Information Systems**]: Installation Management—*pricing and resource allocation*

General Terms: Economics, Management

Additional Key Words and Phrases: capacity planning, cost recovery, incentives, organizational structure pricing, profit maximization, queueing, utilization.

Received 3/82; revised 6/84; accepted 10/84

Author's Present Address: Haim Mendelson, Graduate School of Management, University of Rochester, Rochester, NY 14627.