OPTIMAL DYNAMIC CONTROL OF POLLING SYSTEMS

Uri Yechiali

Dept. of Statistics, Tel Aviv University, Tel Aviv, Israel
and
Bezeq, The Israel Telecommunication Corp., Ltd.

ABSTRACT

Optimal dynamic control and scheduling of the server's visits to the various channels in Polling Systems are difficult problems that only recently have been successfully attacked (cf. Browne & Yechiali [1988a], [1989a]). The control problem is 'which of the K channels to visit next' when the server exits a given channel, and how to achieve optimal schedules based on the *dynamic* evolution of the system.

In this paper we exhibit a class of optimal and 'fair' policies which preserve the cyclic nature of polling systems by guaranteeing a visit to every channel in each cycle (Hamiltonian tour), and at the same time be adaptive to the dynamically changing environment. These policies are K-step look-ahead dynamic procedures which turn out to be extremely simple index-type rules in a form amenable to direct implementation.

We first consider systems with Exhaustive, Gated, Binomial-Exhaustive, Binomial-Gated, Bernoulli-Exhaustive and Bernoulli-Gated service disciplines and derive the control rules that *minimize* dynamically the expected duration of each new cycle, with the idea that by reducing the durations of cycles, waiting times are also reduced. We next analyze systems with 'switching times' between channels, and show that, in all cases, the optimal policies follow from a *general scheduling principle*. This principle is further used to control systems with *mixed* types of channels.

We then treat a system where each channel has a buffer of unit size, and derive the dynamic optimal control that minimizes the costs incurred per cycle. Such systems resemble the classical K-machines repairman problem.

Finally, we present and study the newly introduced Globally-Gated cyclic service scheme (Boxma, Levy & Yechiali [1990]), and the (globally-gated) Elevator-type policy (Altman, Khamisy & Yechiali [1990]). For the former we show that the *long-run minimal cost* is achieved by dynamically optimizing each Hamiltonian tour *individually*. The Elevator-type policy is shown to be the '*fairest*', in the sense that the expected waiting times are *equal for all channels*. This is the *first discovered* non-symmetric scheme that achieves such a goal.

1. INTRODUCTION

Models of polling systems have important applications in telecommunication systems, multiple access protocols, local area computer networks, multiplexing schemes in ISDNs, reader-head's movements in a computer's hard disk, flexible manufacturing systems, road traffic control, etc. (See, for example, surveys of Grillo [1990], Takagi [1990], and Levy & Sidi [1990].) As such they have been the subject of extensive research in recent years, most notably in the context of *cyclic queues* (see Cooper & Murray [1969], Cooper [1970], Eisenberg [1972], Watson [1984], Takagi [1986], Boxma & Groenendijk [1987]). The focus of much of this research has been on evaluating performance measures of fixed-template server's routing schemes, usually with the Exhaustive, Gated or Limited service policies. Optimal server routing procedures were only recently studied and dynamic policies derived (Browne and Yechiali [1988a], [1989a]) for systems where either all channels are of the Exhaustive type, or all channels follow the Gated regime. Probabilistic (yet static) service disciplines – the Binomial-Gated (Levy [1989]) and the Binomial-Exhaustive (Groenendijk, see Boxma [1989]) – were proposed to help deal with the control of polling systems by assigning different service proportions to distinct channels. Other 'fractional' service policies are the Bernoulli-Gated and the Bernoulli-Exhaustive introduced in Browne & Yechiali [1989b].

In all of the above systems it is assumed that each channel's buffer size is unlimited. In some applica-

tions buffers are of *unit* size (Browne & Yechiali [1988b]). Such systems resemble the classical K-machines repairman problems and are of considerable interest. Recently, the Globally-Gated regime was proposed and analyzed (Boxma, Levy & Yechiali [1990]), followed by the Elevator-type polling mechanism (Altman, Khamisy & Yechiali [1990]). These two new regimes possess characteristics that allow for tractable and efficient control of the systems.

In this paper we concentrate on deriving optimal *dynamic control* policies for efficient and 'fair' operation of polling systems. We consider 'homogeneous' systems with the *same* service discipline at all channels (although with different parameters) as well as systems with a *mixed* set of channels; systems where buffer sizes are limited, as well as systems where buffer capacities at the various queues are unlimited.

The dynamic control problem in all of the various configurations is 'which of the K queues to serve next' when the server exits a particular channel (for static optimization of polling systems, see Boxma, Levy & Weststrate [1990], and Boxma [1991]). A common measure of effectiveness is the *weighted sum of expected waiting times* of jobs in the system. In order to minimize this measure one may try to formulate a (semi) Markov Decision Process, calculate the multi-dimensional one-step transition probabilities as well as the one-step expected 'cost' (based on derivations in Levy & Yechiali [1975] and in Yechiali [1976]), and write down Bellman's optimal equations. This has been accomplished (see Browne & Yechiali [1988a], [1989a]), but it appears that there is no simple tractable solution to these equations. The idea then is to consider a *related* measure of effectiveness and to optimize systems' performance over this criterion. Such a criterion is the *minimization of cycle times*. Implicitly, if cycle times are reduced, then waiting times may also be reduced. Our objective will therefore be to develop *dynamic* control policies that *minimize the expected duration of each cycle* based on the dynamic evolution of the system. We require that each cycle will be composed of a Hamiltonian tour, in which every channel is visited *exactly once* – thus providing a degree of fairness between the channels – but the order in which channels are visited may change from one cycle to the other depending on the dynamic changes in time. Surprisingly, it turns out that this criterion leads to *very simple* index-type rules in a form amenable to *direct implementation*.

In section 2 we describe the basic (cyclic) polling scheme, and discuss the various service disciplines. We present previous results on system's control when all channels follow the *same* service discipline: all

Gated or all Exhaustive. In section 3 we develop the dynamic rule to minimize cycle times under the Binomial-Gated service mechanism. Section 4 deals with the Bernoulli-Gated service procedure, while switching times are introduced in section 5. Sections 6 and 7 treat the Binomial-Exhaustive and the Bernoulli-Exhaustive service regimes, respectively, while in section 8 a general scheduling principle is presented which allows the control of systems with a *mixed* set of channels. Systems with a unit-buffer at each channel are studied and dynamically optimized in section 9. The Globally-Gated policy is introduced and optimized in sections 10 and 11. We show that for this procedure, the *long run minimal cost* is indeed achieved by optimizing each cycle *individually* and determining the Hamiltonian tour to be traversed according to a simple scheduling rule used in minimizing *weighted sum of completion times* in a K-job single-processor scheduling problem. Section 12 is devoted to a (globally-gated) Elevator-type polling policy. It is revealed that this policy yields the *same mean waiting time in all channels*. No other nonsymmetric scheme is known to exhibit such a 'fairness' phenomenon.

2. POLLING SYSTEMS, SERVICE REGIMES AND DYNAMIC CONTROL

A polling system, or cyclic queue, is composed of K queues, (channels) labelled $i = 1, \ldots, K$. Jobs arrive at channel i in a Poisson stream of intensity λ_i, independently of the other channels. There is a single server in the system which moves from channel to channel in a 'cyclic' fashion, i.e., the server stays at channel i ($i = 1, \ldots, K$) for a length of time determined by the queue discipline and then moves to channel $i + 1$. Upon 'completion' of channel K, the server reverts to channel 1 and so on, hence the name 'cyclic'.

Each job in channel i carries an independent random service requirement distributed as V_i having distribution function $G_i(\cdot)$, and Laplace-Stieltjes Transform (LST) $\tilde{V}_i(\cdot)$, $i = 1, \ldots, K$. The queue discipline determines how many jobs are to be served in each channel. The disciplines most often studied are the *Exhaustive, Gated* and *Limited service* regimes. To illustrate these regimes, assume the server arrives to channel i to find m_i jobs (customers) waiting. Under the *Exhaustive* regime, the server must service channel i until it is empty before he is allowed to move on. This amount of time is distributed as the sum of m_i *ordinary busy periods* in an $M/G_i/1$ queue. Under the *Gated* regime, the server gates off those customers

already present upon arrival to channel i, and serves only them before moving on to channel $i+1$. As such, the total service time in channel i is distributed as the sum of m_i *ordinary service requirements*. Under the *Limited service* regimes, the server must serve either 1 job, at most k_i jobs, or deplete the queue at channel i by 1 (i.e., stay one busy period of $M/G_i/1$ type).

In the Binomial-Gated, Binomial-Exhaustive, Bernoulli-Gated or Bernoulli-Exhaustive regimes, channel i is characterized by a parameter p_i ($0 \leq p_i \leq 1$), which determines the 'fraction' of service given to this channel. Specifically, let $N_i(m_i)$ be a Binomial random variable with parameters m_i and p_i. Then, according to the Binomial-Gated (BG) policy the server resides in channel i until he serves $N_i(m_i)$ *customers*, while according to the Binomial-Exhaustive (BE) policy he stays there for $N_i(m_i)$ *busy periods*. That is, under the BE policy, when the server exits channel i he leaves behind him $m_i - N_i(m_i)$ waiting jobs, whereas under the BG policy he leaves behind him $m_i - N_i(m_i) + A_i$ customers, where A_i is the number of *new* arrivals to channel i during the visit time of the server. Clearly, the Gated and Exhaustive regimes are special cases of the BG and BE policies, respectively, when $p_i = 1$ for all i.

The Bernoulli-Gated and Bernoulli-Exhaustive disciplines differ from their Binomial counter-parts in that the decision whether to serve customers in channel i or not, is probabilistically made *before* the server 'switches into' the channel. With probability p_i he enters the queue, and with probability $1 - p_i$ he skips it. When the decision is to enter and render service, then according to the Bernoulli-Gated (BRG) regimes, service is completed only to those m_i customers present at the moment of decision, whereas according to the Bernoulli-Exhaustive (BRE) scheme, the server resides at queue i for m_i busy periods. (Note that in an 'ordinary' Bernoulli service mechanism the server 'flips a coin' after *each* service completion, whereas in *our* Bernoulli Gated and Exhaustive procedures he flips a coin only *once*, before switching into the channel).

Typically, the server takes a *non-negligible* amount of time to switch between channels. These switching times are assumed random and complicate the analysis. One important aspect of the distinction between the Binomial regimes and the Bernoulli schemes becomes evident when non-negligible switching times are involved. In the Bernoulli schemes those 'overhead' costs are saved if the decision is *not* to enter the channel at the current cycle.

Dynamic optimal control of cyclic polling systems where service is either of the Gated-type everywhere, or of

the Exhaustive-type at all channels was only recently achieved (Browne & Yechiali [1988a], [1989a]). Suppose that at the beginning of a cycle the state of the system is $(n_1, n_2, \ldots, n_K)$, where n_i is the number of jobs waiting in channel i ($1 \leq i \leq K$). Assume further that switching times between channels are negligible. The objective is to choose a path, or Hamiltonian tour, through the queues so as to *minimize* the expected time of traversing this path. It was shown for *both* service disciplines – the fully Gated and the fully Exhaustive – that this measure of effectiveness is *minimized* if the channels are ordered by *increasing* values of the *index* n_i/λ_i. This is a *surprising* result, as the index n_i/λ_i *does not include the service times* at the various channels. It is surprising as well that the *same* index-rule holds for *both* service regimes (although, obviously, the *duration* of a Gated-type cycle that starts with $(n_1, n_2, \ldots, n_K)$ differs from its Exhaustive counter-part starting with the same system-state).

The dynamics of the control are such that at the *end* of each cycle a *new* system-state is observed, say $(n'_1, n'_2, \ldots, n'_K)$, and the server follows a *new* path governed by a new order: increasing values of n'_i/λ_i, etc. This is an extremely simple rule which can be directly implemented. Moreover, suppose that, for one reason or another, there are systems where the objective is to *maximize* the duration of each cycle. Then, the index-rule that determines the order of visits to the channels is simply *reversed*: the server completes a Hamiltonian tour determined by a *decreasing* order of n_i/λ_i. Extended versions of the above results for a system with customer's feedback are presented in Browne [1990].

3. MINIMIZING CYCLE TIME UNDER THE BINOMIAL-GATED POLICY

In this section we extend the previous results on fully Gated and fully Exhaustive systems to the Binomial-Gated (BG) service policy. We calculate the expected duration of a cycle and determine the optimal control that minimizes each new cycle's duration.

Suppose, as before, that at time 0 the state of the system is $(n_1, n_2, \ldots, n_K)$, where n_i is the number of customers present in queue i. Suppose also that the server visits the channels following the order (policy) $\pi_0 = (1, 2, \ldots, K)$, and the service discipline is Binomial-Gated. Suppose (for the time being) that switching times are negligible. Let X_j be the server's sojourn time in channel j if he finds there m_j customers upon entering the queue. Then, the LST of

X_j is given by (see Browne & Yechiali [1989b]):

$$\widetilde{X}_j(s) = [p_j \widetilde{V}_j(s) + (1 - p_j)]^{n_j} \widetilde{S}_{j-1}(\lambda_j p_j(1 - \widetilde{V}_j(s))) , \tag{1}$$

where $S_{j-1} = \sum_{i=1}^{j-1} X_i$ denotes the exit time of the server from channel $j - 1$.

From Eq. (1) it readily follows that

$$E(X_j) = n_j p_j E(V_j) + b_j p_j E(S_{j-1}) \tag{2}$$

where $b_j \equiv \lambda_j E(V_j)$ is the average amount of work flowing to channel j per unit time. By adding $Z_{j-1} \equiv E(S_{j-1})$ to both sides of Eq. (2) we obtain a system of difference equations

$$Z_j - (1 + p_j b_j) Z_{j-1} = n_j p_j E(V_j) , \quad (Z_0 = 0) \tag{3}$$

whose solution is

$$Z_j = \sum_{i=1}^{j} p_i n_i E(V_i)[\prod_{r=i+1}^{j} (1 + p_r b_r)], \quad (j = 1, 2, \ldots, K). \tag{4}$$

Result (4) may be explained intuitively as follows: $p_i n_i E(V_i)$ is the expected sojourn time of the server in queue i due to the original n_i jobs present at time 0. During that period of time one expects $\lambda_{i+1} p_i n_i E(V_i)$ new arrivals to channel $i+1$, but only a fraction p_{i+1} of them will be served, requiring $p_{i+1} b_{i+1} p_i n_i E(V_i)$ time. Thus, the total expected delay in channels i and $i + 1$ caused by the original n_i customers in queue i will be $p_i n_i E(V_i)(1 + p_{i+1} b_{i+1})$. Proceeding in this manner it follows that the total expected delay caused to the cycle by the n_i initial customers in channel i is $p_i n_i E(V_i)[\prod_{r=i+1}^{K} (1 + p_r b_r)]$. Therefore, the expected total cycle time, following policy π_0, is the sum of the expected delays caused by all initial customers present at the start of the cycle

$$Z_K \equiv C(\pi_0) = \sum_{i=1}^{K} p_i n_i E(V_i)[\prod_{r=i+1}^{K} (1 + p_r b_r)] . \tag{5}$$

Define $a_i \equiv p_i n_i E(V_i)$, and $\alpha_i \equiv p_i b_i$. a_i is the initial expected processing time requirement at channel i, called its *core*, while α_i is the expected *growth* in service requirement at channel i for every unit time delay in performing service to channel i. Thus,

$$C(\pi_0) \equiv \sum_{i=1}^{K} a_i[\prod_{r=i+1}^{K} (1 + \alpha_r)] \tag{6}$$

Similarly, if the server polling sequence is determined by the policy $\pi = (\pi(1), \pi(2), \ldots, \pi(K))$, then the mean cycle length is

$$C(\pi) = \sum_{i=1}^{K} a_{\pi(i)}[\prod_{r=i+1}^{K} (1 + \alpha_{\pi(r)})] . \tag{7}$$

Applying an interchange argument it can be shown (Browne & Yechiali [1990]) that Eq. (7) is minimized if the channels are visited following a sequence determined by ordering the channels via *increasing* values of a_i/α_i. We therefore conclude

Theorem 1. *Suppose that at time 0 the state of the system is* $(n_1, n_2, \ldots, n_K)$. *Then, for the Binomial-Gated policy, the cycle time is minimized if the server visits the channels in an order determined by increasing values of* n_i/λ_i.

Proof: $a_i/\alpha_i = p_i n_i E(V_i)/(p_i b_i) = n_i/\lambda_i$. Q.E.D.

Remark It is interesting to note that the optimal policy is *independent* of the p_i's and $E(V_i)$'s, and it is the *same* as the optimal policy for the *regular* Gated regime (see section 2 above). A further insight into these surprising results will be gained in section 8, where we derive a *general optimization principle* for a wide class of scheduling problems.

4. CYCLE TIME UNDER THE BERNOULLI-GATED SCHEME

Consider now the Bernoulli-Gated service discipline. If m_j customers are present at channel j when the server reaches the station, then his sojourn time there is

$$X_j = \begin{cases} \sum_{k=1}^{m_j} V_{jk}, & \text{with probability } p_j \\ 0, & \text{otherwise} \end{cases}$$

where V_{jk} are all distributed as V_j and are independent. Therefore, conditioning on the number of (Poisson) arrivals during the time S_{j-1}, one can show (Browne & Yechiali [1989b]) that the LST of X_j is given by:

$$\widetilde{X}_j(s) = p_j \left[\widetilde{V}_j(s) \right]^{n_j} \widetilde{S}_{j-1} \left(\lambda_j(1 - \widetilde{V}_j(s)) \right) + (1 - p_j), \tag{8}$$

from which it readily follows that

$$E(X_j) = p_j n_j E(V_j) + p_j b_j E(S_{j-1}) . \tag{9}$$

Writing, as in Eq. (3), $Z_j - (1 + p_j b_j) Z_{j-1} = p_j n_j E(V_j)$, we get that Eqs. (4) and (5) hold in this case as well, with the *same core* $a_i = p_i n_i E(V_i)$ and *growth rate* α_i. That is, the *same* order of visits – by increasing values of n_i / λ_i – minimizes the cycle time under the Bernoulli-Gated regime.

5. SWITCHING TIMES

The above analyses need be only slightly modified to account for switching times. Assume that a direct switch from station i to station j takes time $\theta_i + T_j$, where θ_i is the time to *switch out* of queue i and T_j is the time to *switch into* channel j (T_i and θ_i are independent of each other and of X_j, T_j and θ_j for all $j \neq i$). Let Y_j denote the total server occupation time with channel j during one cycle, so that now the exit time from channel j is $S_j = \sum_{i=1}^{j} Y_i$ with mean $Z_j \equiv E(S_j)$. Assuming that the customers are gated only *after* the server switches into a channel, then, for the Binomial-Gated,

$$Y_j = T_j + \sum_{k=1}^{N_j(m_j)} V_{jk} + \theta_j ,$$

where $m_j = n_j + A_j(S_{j-1} + T_j)$, and $A_j(T)$ denotes the number of Poisson arrivals to channel j during a time interval of length T.

It follows that

$$\tilde{Y}_j(s) = \tilde{\theta}_j(s) [p_j \tilde{V}_j(s) + (1 - p_j)]^{n_j} \cdot$$
$$\cdot \tilde{S}_{j-1}(\lambda_j p_j (1 - \tilde{V}_j(s))) \tilde{T}_j(s + \lambda_j p_j (1 - \tilde{V}_j(s))) , \quad (10)$$

and

$$E(Y_j) = p_j n_j E(V_j) + p_j b_j E(S_{j-1}) +$$
$$+ (1 + p_j b_j) E(T_j) + E(\theta_j) . \quad (11)$$

Upon identifying $p_i n_i E(V_i) + (1 + p_i b_i) E(T_i) + E(\theta_i)$ as the 'core', a_i, and $p_i b_i$ as the 'growth rate', α_i, we can write, for the Hamiltonian tour $\pi_0 = \{1, 2, , \ldots, K\}$,

$$Z_K = \sum_{i=1}^{K} [p_i n_i E(V_i) + (1 + p_i b_i) E(T_i) + E(\theta_i)] \cdot$$
$$\cdot \left[\prod_{r=i+1}^{K} (1 + p_r b_r) \right] . \quad (12)$$

From our previous principles we obtain

Theorem 2. *The order of visits that minimizes cycle time in a Binomial-Gated policy with switching times is determined by an increasing order of*

$$\frac{p_i n_i E(V_i) + (1 + p_i b_i) E(T_i) + E(\theta_i)}{p_i b_i} \quad (13)$$

Now, for the Bernoulli-Gated with switching times and routing policy π_0, suppose that the coin is flipped *after* leaving channel $j-1$, and *before* entering station j. Then,

$$Y_j = \begin{cases} T_j + \left[\sum_{k=1}^{n_j + A_j(S_{j-1} + T_j)} V_{jk} \right] + \theta_j , \\ \qquad\qquad\qquad\qquad \text{with probability } p_j \\ 0 , \qquad\qquad\qquad\qquad \text{otherwise} , \end{cases}$$

Assuming, as before, that the customers are gated only *after* the server switches into a channel, then (see Browne & Yechiali [1989b]),

$$\tilde{Y}_j(s) = p_j [\tilde{\theta}_j(s)][\tilde{V}_j(s)]^{n_j} \tilde{T}_j(s + \lambda_j(1 - \tilde{V}_j(s))) \cdot$$
$$\cdot \tilde{S}_{j-1}(\lambda_j(1 - \tilde{V}_j(s)))] + (1 - p_j) \quad (14)$$

$$E(Y_j) = p_j [E(\theta_j) + n_j E(V_j) +$$
$$+ (1 + b_j) E(T_j) + b_j E(S_{j-1})] \quad (15)$$

Thus,

$$Z_j - (1 + p_j b_j) Z_{j-1} =$$
$$= p_j [n_j E(V_j) + (1 + b_j) E(T_j) + E(\theta_j)] , \quad (16)$$

which results in arranging the channels in increasing order of

$$\frac{n_j E(V_j) + (1 + b_j) E(T_j) + E(\theta_j)}{b_j} \quad (17)$$

It is interesting to note that the policy dictated by Eq. (17) is *identical* to the optimal policy derived for the *pure* Gated regime. Note also that the (small) difference between result (13) and policy (17) is due to the fact that in the derivation of Eq. (13) the server switches *with probability* 1 to channel j and only *then* the value of the random variable $N_j(m_j)$ is realized, whereas in the derivation of Eq. (17) the coin is flipped *before* the server switches into the channel. Thus, while the growth rate $p_j b_j$ is *identical* for the Binomial-Gated and the Bernoulli-Gated regimes, the cores are *different*. For the former the core is $a_i = E(T_i) + p_i [n_i E(V_i) + b_i E(T_i)] + E(\theta_i)$, whereas for the latter the core is $p_i [E(T_i) + n_i E(V_i) + b_i E(T_i) + E(\theta_i)]$.

6. THE BINOMIAL-EXHAUSTIVE POLICY

Consider now the Binomial-Exhaustive regime where the server, if he finds m_i customers in queue i, stays there until the queue length is depleted by $N_i(m_i)$ customers (i.e., for $N_i(m_i)$ *busy periods*), where $N_i(m_i)$ is Binomially distributed with parameters m_i and p_i. This is the Binomial-generalization of the Exhaustive class of disciplines.

Suppose first that there are no switching times. Then, using the same notation as for the Binomial-Gated case, we derive

$$\tilde{X}_j(s \mid m_j) = \sum_{m=0}^{m_j} [\tilde{B}_j(s)]^m \binom{m_j}{m} p_j^m (1 - p_j)^{m_j - m}$$

$$= [p_j \tilde{B}_j(s) + (1 - p_j)]^{m_j} \equiv R_j^{m_j}(s)$$

where B_j is the length of a regular busy period in an $M/G_j/1$ queue, and $\tilde{B}_j(s)$ is its LST with mean $E(B_j) = E(V_j)/(1 - b_j)$. Under policy π_0 the number of customers present in channel j when the server enters the channel is $m_j = n_j + A_j(S_{j-1})$. Hence,

$$\tilde{X}_j(s \mid S_{j-1}) = R_j^{n_j}(s) \cdot \exp\{-\lambda_j p_j (1 - \tilde{B}_j(s)) S_{j-1}\}$$

from which we derive

$$\tilde{X}_j(s) = [p_j \tilde{B}_j(s) + (1 - p_j)]^{n_j} \tilde{S}_{j-1}(\lambda_j p_j (1 - \tilde{B}_j(s))) \tag{18}$$

$$E(X_j) = \frac{n_j p_j E(V_j)}{1 - b_j} + \frac{p_j b_j E(S_{j-1})}{1 - b_j}. \tag{19}$$

We can now identify $p_j n_j E(V_j)/(1 - b_j)$ as the 'core' of channel j, and $p_j b_j/(1 - b_j)$ as its 'growth rate'. Correspondingly, it is immediate that the expected cycle length has the evaluation

$$Z_K = \sum_{i=1}^{K} \left(\frac{n_i p_i E(V_i)}{1 - b_i} \right) \left[\prod_{r=i+1}^{K} \left(1 + \frac{p_r b_r}{1 - b_r} \right) \right], \tag{20}$$

and that the *optimal policy* is to, once again, order the channels in an increasing order of n_i/λ_i, which is *identical* to the optimal policy for the Binomial-Gated and again independent of p_i and $E(V_i)$.

When *switching times* are incurred, utilizing previous notation, we can readily modify the above by observing that Y_j, the server's occupation time with channel j, can be written as

$$Y_j = T_j + \sum_{k=0}^{N_j(m_j)} B_{jk} + \theta_j$$

where $m_j = n_j + A_j(S_{j-1} + T_j)$, and B_{jk} are distributed like B_j. Hence,

$$\tilde{Y}_j(s \mid T_j, S_{j-1}) = \tilde{\theta}_j(s) R_j^{n_j}(s) \exp\{-sT_j\} \cdot$$
$$\cdot \exp\{-\lambda_j(S_{j-1} + T_j)p_j(1 - \tilde{B}_j(s))\}$$

so that

$$\tilde{Y}_j(s) = \tilde{\theta}_j(s)[p_j \tilde{B}_j(s) + (1 - p_j)]^{n_j} \cdot$$
$$\cdot \tilde{S}_{j-1}(\lambda_j p_j(1 - \tilde{B}_j(s))) \cdot \tag{21}$$
$$\cdot \tilde{T}_j(s + \lambda_j p_j(1 - \tilde{B}_j(s)))$$

and

$$E(Y_j) = p_j n_j E(V_j)/(1 - b_j) +$$
$$+ [p_j b_j/(1 - b_j)]E(S_{j-1}) + \tag{22}$$
$$+ [1 + p_j b_j/(1 - b_j)]E(T_j) + E(\theta_j).$$

Similar to the previous derivations, this leads to a mean cycle time

$$Z_K = \sum_{i=1}^{K} \{[p_i n_i E(V_i) + (1 - b_i + p_i b_i)E(T_i) +$$
$$+ (1 - b_i)E(\theta_i)]/(1 - b_i)\} \left[\prod_{r=i+1}^{K} \left(\frac{1 + p_r b_r}{1 - b_r} \right) \right]. \tag{23}$$

We conclude

Theorem 3. *The optimal sequence of visits by the server is determined by arranging the queues in an increasing order of*

$$\frac{p_i n_i E(V_i) + (1 - b_i + p_i b_i)E(T_i) + (1 - b_i)E(\theta_i)}{p_i b_i}$$

7. THE BERNOULLI-EXHAUSTIVE SCHEME

Under this schere, if the server enters channel j and finds m_j customers, he resides there for m_j *busy periods*. As before, the decision whether to enter or not is governed by a Bernoulli trial with probability of success p_j. As $m_j = n_j + A(S_{j-1})$, then, without switching times, we have

$$X_j = \begin{cases} \sum_{k=1}^{m_j} B_{jk}, & \text{with probability } p_j \\ 0, & \text{otherwise}, \end{cases}$$

so that

$$\tilde{X}_j(s) = p_j(\tilde{B}_j(s))^{n_j}\tilde{S}_{j-1}(\lambda_j(1 - \tilde{B}_j(s))) + (1 - p_j) \ . \tag{24}$$

This leads to

$$E(X_j) = p_j[n_jE(B_j) + \lambda_jE(B_j)E(S_{j-1})] =$$
$$= \frac{p_j}{1 - b_j}(n_jE(V_j) + b_jE(S_{j-1})) , \tag{25}$$

and

$$Z_j - \left(1 + \frac{p_jb_j}{1 - b_j}\right) = \frac{p_jn_jE(V_j)}{1 - b_j} \ .$$

Identifying

$$a_j = \frac{p_jn_jE(V_j)}{1 - bj} \quad \text{and} \quad \alpha_j = \frac{p_jb_j}{1 - bj} ,$$

the optimal order of visits is determined by increasing values of $a_i/\alpha_i = n_i/\lambda_i$, exactly as in the case for the Binomial-Exhaustive regime without switching times.

If we take into account switching times, we write

$$Y_j = \begin{cases} T_j + \sum_{k=1}^{n_j + A_j(S_{j-1} + T_j)} B_{jk} + \theta_j , \\ \qquad\qquad\qquad\qquad \text{with probability } p_j \\ 0 , \qquad\qquad\qquad\qquad \text{otherwise} \end{cases}$$

so that

$$\tilde{Y}_j(s) = p_j\left[\tilde{\theta}_j(s)[\tilde{B}_j(s)]^{n_j}\tilde{T}_j(s + \lambda_j(1 - \tilde{B}_j(s)))\cdot \right.$$
$$\left. \cdot\tilde{S}_{j-1}(\lambda_j(1 - \tilde{B}_j(s)))\right] + (1 - p_j) \tag{26}$$

and

$$E(Y_j) = p_j\left[E(\theta_j) + n_j\frac{E(V_j)}{1 - b_j} + \right.$$
$$\left. + (1 + \frac{b_j}{1 - b_j})E(T_j) + \frac{b_j}{1 - b_j}E(S_{j-1})\right] \ . \tag{27}$$

Setting

$$a_j = p_j\left[\frac{n_jE(V_j) + E(T_j) + (1 - b_j)E(\theta_j)}{1 - b_j}\right] ,$$

and

$$\alpha_j = \frac{p_jb_j}{1 - b_j} ,$$

the optimal sequence is determined by the index

$$\frac{a_j}{\alpha_j} = \frac{n_jE(V_j) + E(T_j) + (1 - b_j)E(\theta_j)}{b_j}$$

which is identical to the case with (fully) Exhaustive regime.

8. UNIFICATION VIA A GENERAL PRINCIPLE AND APPLICATION TO MIXED SETS OF CHANNELS

Consider K tasks that must be sequentially performed in a non-preemptive manner by a single processor. All tasks are available at time 0 (as is the processor). Task i carries a random initial processing requirement of expected size a_i, called its core, but if processing is delayed until t, the expected requirement has grown to $a_i + \alpha_it$ (i.e., α_i is the expected growth per unit time delay in performing task i) $i = 1,\ldots,K$. Browne and Yechiali [1990] showed that the dynamics of this process is such that if the tasks are performed following the policy $\pi = (\pi(1), \pi(2), \ldots, \pi(K))$, then the total time to process all K tasks has expectation

$$C(\pi) = \sum_{i=1}^{K} a_{\pi(i)} \prod_{r=i+1}^{K} (1 + \alpha_{\pi(r)}) ,$$

which is minimized when following the permutation based on increasing values of the critical quantity a_i/α_i, the ratio of each task's core to its growth rate.

Our representation of the cycle times for the above four service disciplines in terms of cores (a_i) and growth rates (α_i) allows us to use this principle and immediately solve for cases with Mixed channels, where the service discipline is not common for all channels, but rather, some channels require a pure Exhaustive regime, others – a pure Gated mode, and others – one form or other of 'fractional-type' discipline. In addition, some channels may require switch-in or switch-out times or both. The above general scheduling principle leads directly to

Theorem 4. The mean cycle time is minimized if the channels are arranged by increasing values of a_i/α_i, where, if a channel is Binomial-Exhaustive, then

$$a_i = [p_in_iE(V_i) + (1 - b_i + p_ib_i)E(T_i) +$$
$$+ (1 - b_i)E(\theta_i)]/(1 - b_i)$$
$$\alpha_i = p_ib_i/(1 - b_i)$$

whereas if it is Binomial-Gated,

$$a_i = p_in_iE(V_i) + (1 + p_ib_i)E(T_i) + E(\theta_i)$$
$$\alpha_i = p_ib_i \ .$$

If a channel is Bernoulli-Gated, then

$$a_i = p_i[n_iE(V_i) + (1 + b_i)E(T_i) + E(\theta_i)]$$
$$\alpha_i = p_ib_i ,$$

whereas, if it is Bernoulli-Exhaustive,

$$a_i = p_i[n_iE(V_i) + E(T_i) + (1 - b_i)E(\theta_i)]/(1 - b_i)$$
$$\alpha_i = p_ib_i/(1 - b_i) \ .$$

9. SYSTEMS WITH A UNIT-BUFFER AT EACH CHANNEL

Suppose that each channel can store at most one request at a time and all arrivals to a channel that find the 'buffer' full (occupied) are lost to the system for ever. An occupied channel reopens only upon the completion of the occupier's service request. Equivalently, we may consider a system that takes an exponentially distributed amount of time to generate a new service request at each queue after the service to the previous job has been completed. As such, it is similar to the K machines repairman problem whose optimal dynamic service schedule is being sought. If interarrival and service times at all channels are exponential and identical among channels (a so called 'symmetric' system), then the problem can be formulated as a simple continuous-time Markov chain and the system-state stationary probabilities are easily calculated. However, only recently have Takine, Takahashi & Hasegawa [1988] provided a probabilistic analysis of an *asymmetric* system with non-exponential service times. Here our goal is to derive the dynamic rule to control the server's route if he is allowed to choose his path, or schedule, across the channels at the beginning of each new cycle so as to optimize a measure of system performance. We assume the server has full system state information, i.e., knows which channels are occupied at every decision epoch. Admittedly this assumption might nullify the application to local area networks (see Takagi [1986]) unless we assume further that token passing delays (switching times) are insignificant relative to packet transmission (service) times. Browne and Yechiali [1988b] derived optimal dynamic policies for the semi-symmetric case (identical arrival rates with arbitrary and distinct service times) which can be used to prioritize the channels and could thus be used in conjunction with general polling tables, as discussed in Baker and Rubin [1987].

In Browne and Yechiali's model the following cost structure is imposed on the system: a holding cost at rate $\$h_i$ per unit time a type i job is held in queue, and a penalty cost consisting of a payment of $\$g_i$ per type i job lost to the system, $i = 1, \ldots, K$. The penalty cost could denote the entrance fee to a secondary transmission network that accepts the overflows of the primary system.

To ease exposition and illustrate some basic ideas, we will first analyze the system with zero switching times ($\theta_i = T_i = 0 \ \forall i$).

Let $c_i(a, t)$ denote the *total* cost incurred in channel i in the (time) interval $(a, t]$ *without* channel i having

been served in the said interval.

Let $Q_i(a)$ denote the state of channel i at time a, i.e.,

$$Q_i(a) = \begin{cases} 1 & \text{if buffer } i \text{ is occupied at } a \\ 0 & \text{if buffer } i \text{ is vacant at } a \, . \end{cases}$$

As

$$E\big(c_i(a, t) \mid Q_i(a) = 1\big) = [h_i + \lambda_i g_i](t - a) \quad (28)$$

and

$$
\begin{aligned}
E\big(c_i(a, t) \mid Q_i(a) = 0\big) &= \\
&= \int_a^t \lambda_i e^{-\lambda_i(x-a)}[h_i + \lambda_i g_i](t - x)dx \\
&= [h_i + \lambda_i g_i]\left((t - a) - \frac{1 - e^{-\lambda_i(t-a)}}{\lambda_i}\right) , \quad (29)
\end{aligned}
$$

we have

$$
\begin{aligned}
E\big(c_i(a, t) \mid Q_i(a)\big) &= \\
&= [h_i + \lambda_i g_i]\left((t - a) - \frac{1 - e^{-\lambda_i(t-a)}}{\lambda_i}\big(1 - Q_i(a)\big)\right) .
\end{aligned}
$$

Consider a special instant where the system starts at time $a = 0$ with all buffers full, i.e., $Q(0) = (1, 1, \ldots, 1) \equiv \underline{1}$. (We restrict our attention to the class of Non-Idling as well as Nonpreemptive policies.) By the assumptions above, the cycle time, C, is obviously $C \equiv \sum_{i=1}^{K} V_i$, which is *invariant* with respect to policy.

Consider now policy $\pi_0 = (1, 2, \ldots, K)$. Under π_0 channel i will be occupied until time $\sum_{j=1}^{i} V_j$, whereupon its buffer frees up at its service completion, so that $Q_i(\sum_{j=1}^{i} V_j) = 0$. We can therefore use equations (28) and (29) directly (with $\lambda_i = \lambda \ \forall i$) to evaluate the expected total cost incurred by channel i under π_0 as

$$
\begin{aligned}
E\big(c_i(0, C) \mid \underline{Q}(0) &= \underline{1}, \pi_0\big) = \\
&= E\Bigg\{ (h_i + \lambda g_i)\sum_{j=1}^{i} V_j + (h_i + \lambda g_i) \cdot \\
&\quad \cdot \left[\left(C - \sum_{j=1}^{i} V_j\right) - \frac{1 - e^{-\lambda(C - \sum_{j=1}^{i} V_j)}}{\lambda}\right] \Bigg\} \\
&= (h_i + \lambda g_i)\left[E(C) - \frac{1}{\lambda}\right] + \\
&\quad + \left(\frac{h_i + \lambda g_i}{\lambda}\right)\prod_{j=i+1}^{K} \tilde{V}_j(\lambda) .
\end{aligned}
$$

Therefore, the total expected cost incurred by the system following tour π_0 is

$$E\left[\sum_{i=1}^{K} c_i(0,C) \mid \underline{Q}(0) = \underline{1}, \pi_0\right] =$$

$$= \left(E(C) - \frac{1}{\lambda}\right)\sum_{i=1}^{K}(h_i + \lambda g_i)+ \qquad (30)$$

$$+ \sum_{i=1}^{K}\left(\frac{h_i + \lambda g_i}{\lambda}\right)\prod_{j=i+1}^{K} \tilde{V}_j(\lambda) \ .$$

As obviously only the second term in equation (30) is affected by policy, it is that term we need to minimize. By applying an interchange argument, Browne and Yechiali [1988b] showed:

Theorem 5. *The tour of minimal expected cost is prescribed by the policy π^* which orders the channels in decreasing values of the index*

$$\lambda\frac{h_i + \lambda g_i}{1 - \tilde{V}_i(\lambda)} \qquad (31)$$

Remarks

If $\lambda \to 0$ then the index (31) reduces directly to the classical '$c\mu$ rule' which orders channels by decreasing values of the ratio: [cost rate/expected service time]. This follows since

$$\lim_{\lambda\searrow 0} \lambda\frac{h_i + \lambda g_i}{1 - \tilde{V}_i(\lambda)} = \frac{h_i}{E(V_i)} \ .$$

However, for $\lambda > 0$, while $h_i + \lambda g_i$ can be considered the effective cost rate per unit time for a *serviced* arrival (with no other costs being incurred), $1 - \tilde{V}_i(\lambda)$ is the *probability* of at least one arrival *during* a type i service. As such, π^* is sensitive to *all* the moments of V_i (assuming $G_i(\cdot)$ is determined by all its moments) while the $c\mu$ rule depends only on $E(V_i)$.

Switching Times

Suppose, as before, that the server needs an independent random time T_i to switch in, or set-up, channel i prior to service, and an independent random time θ_i to switch out of channel i post service there. We further assume that the channel (or buffer) frees up at termination of service. Then, applying similar type of analysis, the following holds:

Theorem 6. *When switching times are included, the Hamiltonian tour of minimal expected cost is achieved by ordering the channels in decreasing values of the index*

$$\lambda\frac{(h_i + \lambda g_i)\tilde{\theta}_i(\lambda)}{1 - \tilde{T}_i(\lambda)\tilde{V}_i(\lambda)\tilde{\theta}_i(\lambda)} \ . \qquad (32)$$

10. GLOBALLY GATED REGIME

Boxma, Levy and Yechiali [1990] introduced a (cyclic) Globally Gated (GG) service scheme which uses a time-stamp mechanism for its operation: the server moves cyclically among the queues, and uses the instant of cycle-beginning as a reference point of time; when it reaches a queue it serves there all (and only) customers who were present at that queue at the cycle-beginning. This strategy can be implemented by marking all customers with a time-stamp denoting their arrival time. In its nature the GG policy resembles the regular Gated policy. However, the GG policy leads to a simpler mathematical model which in turn allows for derivation of closed-form expressions for the mean delay in the various queues. As a result, the operation of the polling system by the GG policy is easy to control and optimize. As in earlier sections the system consists of K infinite-buffer channels, the offered load to queue i is $b_i = \lambda_i E(V_i)$ and the total system load-rate is $\rho \equiv \sum_{i=1}^{K} b_i$. We assume that, when leaving queue i and before starting service at the next queue, the server incurs a switchover, or 'walking time' period, whose duration is a random variable θ_i. The total 'walking time' in a cycle is $\theta \equiv \sum_{i=1}^{K} \theta_i$. (Clearly, other 'Global' versions, such as Globally Exhaustive, can be easily imagined and analyzed.)

Boxma, Levy and Yechiali derived the LST of the cycle time C and calculated its mean and second moment:

$$E(C) = E(\theta)/(1 - \rho) \ , \qquad (33)$$

which follows from the relation $E(C) = E(\theta) + (\sum_{j=1}^{K} b_j)E(C)$,

$$E(C^2) = \frac{1}{1 - \rho^2}\left[E(\theta^2) + 2E(\theta)\rho E(C)+ \right.$$
$$\left. + \sum_{j=1}^{K} \lambda_j E(V_j^2)E(C)\right] \ . \qquad (34)$$

They further showed that the mean waiting time for an arbitrary customer at channel k ($k = 1, 2, \ldots, K$)

is given by

$$E(W_k) = \left[1 + 2\left(\sum_{j=1}^{k-1} b_j\right) + b_k\right] E(C_R) + \sum_{j=1}^{k-1} E(\theta_j),$$
(35)

where $E(C_R) = \frac{E(C^2)}{2E(C)}$ is the mean residual time of a cycle.

It readily follows from (33), (34) and (35) that $E(W_1) < E(W_2) < \cdots < E(W_K)$. In particular,

$$E(W_{k+1}) - E(W_k) = (b_{k+1} + b_k)E(C_R) + E(\theta_k).$$

That is, if the server always performs a cycle by traversing the channels in the *same* order, it is advantageous to belong to a queue with a small index. In other words, the closer a channel is positioned to the starting point of the cycle – the better.

11. OPTIMAL ORDER OF VISITS UNDER THE GLOBALLY GATED REGIME

I. Static Optimization

deals with finding the *repeated* order of visits that minimizes the mean waiting cost of an arbitrary customer in the system. Denoting by c_k the cost of a type-k job being delayed one unit of time, the (static) objective is to find a repeated visit order that minimizes the expression $\sum_{k=1}^{K} \lambda_k c_k E(W_k)$. From (35),

$$\sum_{k=1}^{K} \lambda_k c_k E(W_k) =$$

$$= E(C_R) \sum_{k=1}^{K} \lambda_k c_k \left(1 + b_k + 2\sum_{j=1}^{k-1} b_j\right) +$$

$$+ \sum_{k=1}^{K} \lambda_k c_k \sum_{j=1}^{k-1} E(\theta_j),$$

in which the only factor that depends on the order of the queues is

$$\sum_{k=1}^{K} \lambda_k c_k \sum_{j=1}^{k-1} \{2E(C_R)b_j + E(\theta_j)\}.$$

Using an interchange argument one can readily show that, in this case too, an *index rule* for optimal ordering of the queues holds. Namely, the *minimal* value for $\sum_{k=1}^{K} \lambda_k c_k E(W_k)$ is obtained by arranging the queues in an *increasing* order of the index

$$u_j = \frac{2E(C_R)b_j + E(\theta_j)}{\lambda_j c_j} = \frac{b_j E(C^2)/E(C) + E(\theta_j)}{\lambda_j c_j}.$$

Several special cases of this result and variations of the problem analyzed above are of interest:

(i) Special costs

Consider the special case in which $c_j = E(V_j)$, namely the waiting cost is proportional to the mean service time. In this case the optimization objective becomes $\sum_{k=1}^{K} b_k E(W_k)$, which is the term of the pseudoconservation law (see Boxma and Groenendijk [1987], Boxma [1989],[1991]). Optimization now is equivalent to the minimization of the mean amount of work in the system. Here we have

$$u_j = 2E(C_R) + \frac{E(\theta_j)}{b_j},$$

and thus the queues should be arranged in an *increasing* order of $E(\theta_j)/b_j$.

(ii) Negligible Switching Times

Consider the case where $E(\theta_j) \ll 2E(C_R)b_j$ for all queues. Then, the queues should be arranged in an *increasing* order of $E(V_j)/c_j$, which is, again, a $c\mu$-type rule.

(iii) Fixed topology

In some applications the cyclic order of the queues is predetermined and is not left to free choice. In this case the optimal design of the system is achieved by selecting the gating point (namely, choosing which queue will be the first on the cycle). For these cases there is no simple index rule but the optimization problem can be easily solved by comparing the expressions achieved for the K possible cases.

II. Dynamic Optimization

is achieved as described in previous sections. At the *beginning* of each cycle the current queue lengths, $n_1, \ldots, n_K$ are evaluated and the visit order of the *next* cycle is determined. By the very nature of the Globally Gated scheme, the visit order taken in one cycle *does not affect* the future stochastic behavior of the system. Moreover, the cycle-time duration $C(n_1, \ldots, n_K)$ is the *same* for any Hamiltonian tour of the queues. Thus, if we consider the costs incurred during a cycle by the customers present at its initiation *together* with the costs incurred by the new arrivals between two cycle-beginnings, the *long run minimal cost* can be achieved by optimizing each cycle *individually*.

The mean total waiting cost incurred during the coming cycle is:

$$\sum_{k=1}^{K} c_k \left[n_k \sum_{j=1}^{k-1} [n_j E(V_j) + E(\theta_j)] + E(V_k) \sum_{i=1}^{n_k - 1} i \right] +$$

$$+ \sum_{k=1}^{K} c_k \lambda_k E\left[C(n_1, \ldots, n_K)^2\right]/2$$

where the first term is the contribution to total cost of the customers present at the cycle-beginning, and the second is due to the customers arriving *during* the cycle starting with $n_1, n_2, \ldots, n_K$ (see Yechiali [1976]). The only term that depends on the order of visits is $\sum_{k=1}^{K} c_k \sum_{j=1}^{k-1} [n_j E(V_j) + E(\theta_j)]$. It readily follows that the optimal order for the next cycle is determined by *increasing* values of the indices

$$u_j = \frac{n_j E(V_j) + E(\theta_j)}{c_j n_j}.$$

which is, once more, a $c\mu$-type rule.

12. ELEVATOR-TYPE POLLING POLICY

An Elevator-type polling policy is the following: instead of moving cyclically through the stations, the server first serves stations in one direction, i.e. in the order of $1, 2, \ldots, K$ ('up' direction) and then reverses its orientation and serves the channels in the opposite direction ('down'), i.e. going through stations $K, K - 1, \ldots, 2, 1$. It then again changes direction, and keeps moving in this manner back and forth. This type of service discipline is encountered in many applications, e.g. it models a common scheme of addressing a hard disk for writing (or reading) information on (or from) different tracks. It is important to note that the Elevator scheme 'saves' the return walking time from station K to station 1 (when comparing it to cyclic polling systems). This may be a significant factor in decreasing the cycle duration and expected waiting times, since in many systems this return walking time is considerably large. In the case of hard disks this return walking time represents the movement of the head all the way back from the Kth track to the first. We assume that it takes the *same* (random) time to 'walk' from channel j to channel $j + 1$ as it takes to move 'backwards' from $j + 1$ to j.

All the service disciplines that have been considered in the literature with relation to cyclic movement (e.g. the Gated, Exhaustive, Limited, Globally Gated) can be implemented also with the Elevator approach. We shall present here a globally gated version of the Elevator service. We shall call the period during which the server moves up an 'up cycle' and the period during which the server moves down a 'down cycle'. A 'cycle' will be either an up cycle or a down cycle.

Altman, Khamisy and Yechiali [1990] introduced the following globally-gated version of the Elevator scheme. Consider a moment when the server is ready to start service at station 1 and the system state is $(n_1, n_2, \ldots, n_K)$. Then a 'global' gate is 'closed' and the server starts its *up cycle*, moving from 1 to K, serving in channel i only those n_i jobs that were present at the *beginning* of this cycle. As soon as the *last* job of the n_K jobs at channel K is completed, a *new* 'global' gate is closed, the system state is $(n_1', n_2', \ldots, n_K')$ and the server starts its *down cycle* serving at i only the n_i' marked customers. Then, a 'global' gate is closed again, and the server starts its up cycle, etc.

Think now of a different interpretation of the (globally gated) Elevator scheme. Assume that when completing the up movement there is an extra walking time of *zero* duration from station K to station 1. When the server now arrives at station 1, a gate is closed, the server *jumps* back to station K in *zero* time and then goes down to state 1, serving all customers marked when the gate was closed. After serving station 1 the gate is again closed and the server goes up again, serving all customers present at the moment the gate was closed, then the server jumps again, etc.

Consider for the moment, the *cyclic* Globally Gated discipline. As was indicated before, for each cycle along any sample path, the cycle duration is *unchanged* if we alter the order of the stations being served and/or the order of the walking times. This follows from the fact that the number of customers served in each station is determined at the instant the gate is closed, and therefore is not influenced by any change of order of visits to the stations. In particular, the cycle duration remains unchanged if every second cycle the stations are served in the order $K, K-1, \ldots, 2, 1$ with the order of walking times inverted. It can be seen that if the walking time from station K to station 1 is zero, then this model coincides with the Elevator discipline. Hence the distribution of the cycle duration for the Elevator scheme is equal to the distribution of cycle duration for the case of cyclic Globally Gated service discipline with *zero* walking time from station K to station 1. As a result, equations (33) and (34) *hold in this case too*, with the trivial modification that $\theta_K \equiv 0$.

Waiting Times

Consider now an arbitrary customer arriving at station k. It has probability 0.5 to arrive during an up cycle and probability 0.5 to arrive during a down cycle. Thus

$$E[W_k] = 0.5\big(E[W_k \mid \text{up}] + E[W_k \mid \text{down}]\big) . \quad (36)$$

To compute the expected waiting time of a customer arriving during an up cycle, we may assume that all previous cycles are up cycles, since the number of customers in each station at the instant that the gate is closed is not influenced by the type of previous cycles. Hence, from (35),

$$E[W_k \mid \text{up}] = \left(1 + 2\sum_{i=1}^{k-1} b_i + b_k\right)E(C_R) + \sum_{i=1}^{k-1} E(\theta_i) . \tag{37}$$

Similarly we have

$$E[W_k \mid \text{down}] = \left(1 + 2\sum_{i=k+1}^{K} b_i + b_k\right)E(C_R) + \sum_{i=k}^{K-1} E(\theta_i) . \tag{38}$$

Combining equations (36),(37) and (38) we obtain, for $k = 1, 2, \ldots, K$,

$$E[W_k \mid \text{Elevator}] = (1 + \rho)E(C_R) + E(\theta)/2 . \tag{39}$$

Result (39) reveals an *interesting phenomenon*: in the (globally gated) Elevator regime *expected waiting times in all channels are the same*. This is the *only* known non-symmetric polling system that possesses such a property. In such, the Elevator discipline is the 'fairest' of all other service procedures (see Boxma [1991] for a discussion of 'fairness'), and *any order* of the channels yields expression (39).

13. CONCLUSION

We have presented and derived optimal *dynamic* control policies for various polling systems with a single server and Poisson arrivals. Boxma [1991] addressed another aspect of the problem of optimization in polling system, viz. "Determination of that polling table in a (static) periodic polling model that minimizes a certain weighted sum of the mean waiting times". Using both avenues in designing, operating and control of polling systems will lead to more efficient, better-managed and 'fairer' systems.

References

[1] Altman, E., A. Khamisy and U. Yechiali [1990], "Threshold Service Policies in Polling Systems", Technical Report, Dept. of Statistics, Tel Aviv University.

[2] Baker, J.E. and I. Rubin [1987], "Polling with a General-Service Order Table", *IEEE Transactions in Communications* **35**, 283-288.

[3] Boxma, O.J. and W.P. Groenendijk [1987], "Pseudo-Conservation Laws in Cyclic-Service Systems", *J. of Applied Probability* **24**, 949-964.

[4] Boxma, O.J. [1989], "Workloads and Waiting Times in Single-Server Systems with Multiple Customer Classes", *Queueing Systems* **5**, 185-214.

[5] Boxma, O.J, H. Levy and J.A. Weststrate [1990], "Optimization of Polling Systems", in *Performance '90*, P.J.B. King, I. Mitrani and R.J. Pooley (Eds.), North Holland, 349-361.

[6] Boxma, O.J., H. Levy and U. Yechiali [1990], "Cyclic Reservation Schemes for Efficient Operation of Multiple-Queue Single-Server Systems", Technical Report 188/90, Dept. of Computer Science, Tel Aviv University.

[7] Boxma, O.J. [1991], "Analysis and Optimization of Polling Systems", in *Proceedings of 13th International Teletraffic Congress*, Copenhagen, June 1991.

[8] Browne, S. and U. Yechiali [1988a], "Dynamic routing in Polling Systems", in *Teletraffic Science for New Cost-Effective Systems, Networks and Services*, M. Bonatti (Ed.), [Proceedings of ITC-12, Torino, June 1-8, 1988], 1455-1466, Elsevier Sciences Publishers, 1989.

[9] Browne, S. and U. Yechiali [1988b], "Dynamic Scheduling in Single Server Multi-Class Service Systems with Unit Buffers", *Naval Research Logistics* (forthcoming, 1991).

[10] Browne, S. and U. Yechiali [1989a], "Dynamic Priority Rules for Cyclic-Type Queues", *Advances in Applied Probability* **21**, 432-450.

[11] Browne, S. and U. Yechiali [1989b], "Dynamic Server Routing in Binomial-Gated, Binomial-Exhaustive, Bernoulli-Gated, Bernoulli-Exhaustive and Mixed Polling Systems", Technical Report, Dept. of Statistics, Tel Aviv University.

[12] Browne, S. and U. Yechiali [1990], "Scheduling Deteriorating Jobs on a Single Processor", *Operations Research* **38**, 495-498.

[13] Browne, S. [1990], "Optimal Polling in Multiclass Systems with Feedback and Customer Routing", Technical Report, Graduate School fo Business, Columbia University.

[14] Cooper, R.B. and G. Murray [1969], "Queues Served in Cyclic Order", *The Bell System Technical Journal* **48**, 675-689.

[15] Cooper, R.B. [1970], "Queues Served in Cyclic Order: Waiting Times", *The Bell System Technical Journal* **49**, 399-413.

[16] Eisenberg, M. [1972], "Queues with Periodic Service and Changeover Times", *Operations Research* **2**, 70-76.

[17] Grillo, D. [1990], "Polling Mechanism Models in Communication Systems – Some Application Examples", in *Stochastic Analysis of Computer and Communication Systems*, H. Takagi (Ed.), North-Holland, 659-698.

[18] Levy, Y. and U. Yechiali [1975], "Utilization of Idle Time in an M/G/1 Queueing System", *Management Science* **22**, 202-211.

[19] Levy, H. [1989], "Analysis of Cyclic-Polling Systems with Binomial-Gated Service", in *Performance of Distributed and Parallel Systems*, H. Hasegawa, H. Takagi and Y. Takahashi (Eds.), North-Holland, 127-139.

[20] Levy, H. and M. Sidi [1990], "Polling Systems: Applications, Modelling and Optimization", *IEEE Transactions on Communications* (forthcoming).

[21] Takagi, H. [1986], *Analysis of Polling Systems*, MIT Press.

[22] Takagi, H. [1990], "Queueing Analysis of Polling Models: An Update", in *Stochastic Analysis of Computer and Communications Systems*, H. Takagi (Ed.) North-Holland, 267-318.

[23] Takine, T., Takahashi and T. Hasegawa [1988], "Exact Analysis of Asymmetric Polling Systems with Single Buffers", *IEEE Transactions on Communications* **36**, 1119-1127.

[24] Watson, K.S. [1984], "Performance Evaluation of Cyclic Service Strategies – a Survey", in *Performance '84*, E. Gelenbe (Ed.), North Holland, 521-533.

[25] Yechiali, U. [1976], "A New Derivation of the Khintchine-Pollaczek Formula", in *Operational Research '75*, K.B. Haley (Ed.), North Holland, 261-264.