ANALYSIS AND CONTROL OF POLLING SYSTEMS

Uri Yechiali

Department of Statistics & Operations Research, School of Mathematical Sciences, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel
Email: uriy@math.tau.ac.il

Abstract. We present methods for analyzing continuous-time multi-channel queueing systems with Gated, Exhaustive, or Globally-Gated service regimes, and with Cyclic, Hamiltonian or Elevator-type polling mechanisms. We discuss issues of dynamically controlling the server's order of visits to the channels, and derive easily implementable index-type rules that optimize system's performance. Future directions of research are indicated.

Keywords: Multi-channel queueing systems, polling, gated, exhaustive, globally-gated, conservation laws, Hamiltonian tours, Elevator polling, dynamic control.

1 Introduction

Queueing systems consisting of N queues (channels) served by a single server which incurs switch-over periods when moving from one channel to another have been widely studied in the literature and used as a central model for the analysis of a wide variety of applications in the areas of computer networks, telecommunication systems, multiple access protocols, multiplexing schemes in ISDNs, reader-head's movements in a computer's hard disk, flexible manufacturing systems, road traffic control, repair problems and the like. Very often such applications (e.g. Token Ring networks in which N stations attempt to transmit their messages by sharing a single transmission line) are modeled as a polling system where the server visits the channels in a cyclic routine or according to an arbitrary polling table.

In many of these applications, as well as in most polling models, it is customary to control the amount of service given to each queue during the server's visit. Common service policies are the Exhaustive, Gated and Limited regimes. Under the Exhaustive regime, at each visit the server attends the queue until it becomes completely empty, and only then is the server allowed to move on. Under the Gated regime, all (and only) customers (packets, jobs) present when the server starts visiting (polls) the queue are served during the visit, while customers arriving when the queue is attended will be served during the next visit. Under the K_i-Limited service discipline only a limited number of jobs (at most K_i) are served at each server's visit to queue i. There is extensive literature on

the theory and applications of these models. Among the first works are Cooper & Murray [1969] and Cooper [1970] who studied the cyclic Exhaustive and Gated regimes with no switchover times. Eisenberg [1972] generalized the results of Cooper & Murray by allowing changeover times and by considering a *general* polling table, i.e., by allowing a general configuration of the server's (periodic) sequence of visits to the channels. Many other authors have investigated various aspects of polling systems, and for a more detailed description the reader is referred to a book [1986] and an update [1990] by Takagi, and to a survey by Levy & Sidi [1990].

Recently, Globally-Gated regimes were proposed by Boxma, Levy & Yechiali [1992], who provided a thorough analysis of the *cyclic* Globally-Gated (GG) scheme. . Under the Globally-Gated regime the server uses the instant of cycle beginning as a reference point of time, and serves in each queue only those jobs that were present there at the cycle-beginning.

A special, yet important, polling mechanism is the so-called Elevator (or scan)-type (cf. Shoham & Yechiali [1992], Altman, Khamisy & Yechiali [1992]): instead of moving cyclically through the channels, the server first visits the queues in one direction, i.e. in the order $1, 2, \ldots, N$ ('up' cycle) and then reverses its orientation and serves the channels in the opposite direction ('down' cycle). Then it changes direction again, and keeps moving in this manner back and forth. This type of service regime 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. Among its advantages is that it saves the return walking time from channel N to channel 1.

All the above models studied *open* systems with external arrivals, where jobs exit the system after service completion. Altman & Yechiali [1992] studied a *closed* system in which the number of jobs is fixed. They analyzed the Gated, Exhaustive, Mixed and Globally-Gated regimes and derived measures for system's performance.

One of the main tools used in the analysis of polling systems is the derivation of a set of multi-dimensional Probability Generating Functions (PGS_i's) of the number of jobs present in the various channels at a polling instant to queue i ($i = 1, 2, \ldots, N$). The common method is to derive PGF_{i+1} in terms of PGF_i and from the set of N (implicit) dependent equations in the unknown PGF_i's one can obtain expressions which allow for *numerical* calculation of the mean queue size or mean waiting time at each queue. The Globally-Gated regime stands out among the various disciplines as it yields a *closed-form* analysis and leads to *explicit* expressions for performance measures, such as mean and second moment of waiting time at each queue, as well as the Laplace-Stieltjes Transform (LST) of the cycle duration.

Most of the work on polling systems has been concentrated on obtaining equilibrium mean-value or approximate results for the various service disciplines. Browne & Yechiali [1989a], [1989b] were the first to obtain *dynamic* control policies for systems under the Exhaustive, Gated or Mixed service regimes. At the beginning of each cycle the server decides on a *new* Hamiltonian tour and

visits the channels accordingly. Browne & Yechiali showed that if the objective is to minimize (or maximize) cycle-duration, then an index-type rule applies. Such a rule makes it extremely easy for practical implementations. For the Globally-Gated regime Boxma, Levy & Yechiali [1992] showed that minimizing weighted waiting costs for each cycle *individually*, minimizes the long-run average weighted waiting costs of all customers in the system. A surprising result holds for the Globally-Gated Elevator-type mechanism (Altman, Khamisy & Yechiali [1992]): mean waiting times in *all* channels are the *same*.

In this tutorial we present and discuss (i) *analytical* techniques used in studying polling systems, and (ii) methods derived and applied for *dynamic control* of such systems.

In sections 3 and 4 we present the basic tools for analyzing polling systems with Gated or with Exhaustive service regimes, respectively. Section 5 discusses conservtion laws and optimal visit frequencies. In section 6 we address the issue of dynamic control of polling systems having service regimes with linear growth of work. Secion 7 studies the Globally-Gated regime, and in section 8 the Elevator-type polling mechanism is analyzed. Future directions of research are indicated in section 9.

2 Models and Notation

A polling system is composed of N channels (queues), labeled $1, 2, \ldots, N$, where 'customers' (messages, jobs) arrive at channel i according to some arrival process, usually taken as an independent Poisson process with rate λ_i. There is a single server in the system which moves from channel to channel following a prescribed order ('polling table'), most-commonly cyclic, i.e., visiting the queues in the order $1, 2, , \ldots, N-1, N, 1, 2, \ldots$. The server stays at a channel for a length of time determined by the service discipline and then moves on to the next channel.

Each job in channel i ($i = 1, 2, \ldots, N$) carries an independent random service requirement B_i, having distribution function $G_i(\cdot)$, Laplace-Stieltjes Transform $\widetilde{B}_i(\cdot)$, mean b_i, and second moment $b_i^{(2)}$. 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 it 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 m_i customers and serves only them before moving on to the next channel. As such, the total service time in channel i is distributed as the sum of m_i ordinary *service requirements*. Under *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). According to the recently introduces Globally-Gated service regime, at the *start* of the cycle *all* channels are 'gated off' *simultaneously*, and only customers gated at that instant will be served during the coming cycle.

Typically, the server takes a (random) *non-negligible* amount of time to switch between channels. This time is called 'walking' or 'switchover' period. The switchover duration from channel i to the next is denoted by D_i, with LST $\widetilde{D}_i(\cdot)$, mean d_i, and second moment $d_i^{(2)}$. In some applications (e.g. star configuration) the time to move from channel i to channel j ($j \neq i$) is composed of a switch over time D_i, out of channel i, *plus* a switch-in period to channel j, R_j. In other applications, even for a cyclic polling procedure, the switch-in time R_j is incurred *only* if there is *at least* one message in queue i (see Altman, Blanc, Khamisy & Yechiali [1992]), thus saving the switching time into an empty channel.

We will discuss here only systems where each channel has an *infinite buffer* capacity, assuming *steady state* conditions, and we focus on *continuous-time* models where channel i is an $M/G_i/1$ queue with Poisson arrival rate λ_i and service requirements B_i. The analysis will concentrate on *three main service regimes*: Gated, Exhaustive and Globally-Gated.

3 Analysis of the Gated Regime

Let X_i^j denote the number of jobs present in channel j ($j = 1, 2, \ldots, N$) when the server arrives at (polls) channel i ($= 1, 2, \ldots, N$). $\mathbf{X}_i = (X_i^1, X_i^2, \ldots, X_i^N)$ is the state of the system at that instant. Let $A_i(T)$ be the number of Poisson arrivals to channel i during a (random) time interval of length T. Then, for the Gated service regime, the evolution of the state of the system is given by

$$X_{i+1}^j = \begin{cases} X_i^j + A_j\left(\sum_{k=1}^{X_i^i} B_{ik} + D_i\right), & j \neq i \\ A_i\left(\sum_{k=1}^{X_i^i} B_{ik} + D_i\right), & j = i \end{cases} \tag{1}$$

where B_{ik} are all distributed as B_i.

One of the basic tools of analysis is to derive the multidimensional Probability Generating Function (PGF_i) of the state of the system at the polling instant to channel i ($i = 1, 2, \ldots, N$). PGF_i is defined as

$$G_i(\mathbf{z}) = G_i(z_1, z_2, \ldots, z_{i-1}, z_i, z_{i+1}, \ldots, z_N) = E\left[\prod_{j=1}^{N} z_j^{X_i^j}\right]. \tag{2}$$

Then, for the *Gated* regime, while using (1),

$$\begin{aligned} G_{i+1}(\mathbf{z}) &= E\left[\prod_{j=1}^{N} z_j^{X_{i+1}^j}\right] \\ &= E_{\mathbf{X}_i}\left[\prod_{\substack{j=1 \\ j\neq i}}^{N} z_j^{X_i^j} E\left[\prod_{j=1}^{N} z_j^{A_j\left(\sum_{k=1}^{X_i^i} B_{ik}\right)} \,\middle|\, \mathbf{X}_i\right]\right] \cdot E\left[\prod_{j=1}^{N} z_j^{A_j(D_i)}\right] \end{aligned} \tag{3}$$

For a Poisson random variable $A_j(T)$, and with $\widetilde{T}(\cdot)$ denoting the Laplace-Stieltjes Transform (LST) of T, we have

$$E[z_j^{A_j(T)}] = E_T[e^{-\lambda_j(1-z_j)T}] = \widetilde{T}\big[\lambda_j(1-z_j)\big]$$

and

$$E\left[\prod_{j=1}^{N} z_j^{A_j(T)}\right] = \widetilde{T}\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right].$$

Therefore

$$G_{i+1}(\mathbf{z}) = E_{\mathbf{X}_i}\left[\prod_{\substack{j=1\\ j\neq i}}^{N} z_j^{X_i^j}\left(\widetilde{B}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right]\right)^{X_i^i}\right]\cdot\widetilde{D}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right].$$

Thus, for $i = 1, 2, \ldots, N-1, N$ (where we take $N+1$ as 1)

$$G_{i+1}(\mathbf{z}) = G_i\left(z_1, z_2, \ldots, z_{i-1}, \widetilde{B}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right], z_{i+1}, \ldots, z_N\right)\cdot\widetilde{D}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right] \tag{4}$$

Equations (4) define a set of N relations between the various PGFs which are used to derive moments of the variables X_i^j, as follows.

Moments The mean number of messages, $f_i(j) = E(X_i^j)$, present in channel j at a polling instant to channel i is obtained by taking derivatives of the PGFs, where

$$f_i(j) = E(X_i^j) = \left.\frac{\partial G_i(\mathbf{z})}{\partial z_j}\right|_{\mathbf{z}=\mathbf{1}} \tag{5}$$

A set N^2 linear equations in $\{f_i(j) : i, j = 1, 2, \ldots, N\}$ determines their values:

$$f_{i+1}(j) = \begin{cases} f_i(j) + \lambda_j b_i f_i(i) + \lambda_j d_i & j \neq i \\ \lambda_i b_i f_i(i) + \lambda_i d_i & j = i \end{cases} \tag{6}$$

Indeed, equations (6) could be obtained *directly* from (1).

Set $\rho_k = \lambda_k b_k$, $\rho = \sum_{k=1}^{N}\rho_k$, $d = \sum_{k=1}^{N} d_k$. Then, the solution of (6) is given by

$$f_i(j) = \begin{cases} \lambda_j\left(\sum_{k=j}^{i-1}\left[\rho_k\left(\frac{d}{1-\rho}\right) + d_k\right]\right) & j \neq i \\ \lambda_i\left(\frac{d}{1-\rho}\right) & j = i \end{cases} \tag{7}$$

The explanation of (7) is the following. It will be shown shortly that the mean cycle time is $E[C] = d/(1-\rho)$. During that time the mean number of arrivals to channel i is $\lambda_i E[C]$. Also, during a cycle the server renders service to channel k for an average length of time $\rho_k E[C]$. Thus, the elapsed time since the last gating instant of channel j ($j \neq i$) until the polling instant of channel i, is

$\sum_{k=j}^{i-1}\left[\rho_k E[C]+d_k\right]$. Within that time-interval the mean number of arrivals to channel j is $f_i(j)$, as given by (7).

The second moments of the X_i^j are also derived from the set of PGFs (4). Let

$$\begin{aligned} f_i(j,k) &= E[X_i^j X_i^k] = \left.\frac{\partial^2 G_i(\mathbf{z})}{\partial z_j\,\partial z_k}\right|_{\mathbf{z}=1} \qquad (i,j,k=1,2,\ldots,N \text{ not all equal}) \\ f_i(i,i) &= E\left[X_i^i(X_i^i-1)\right] = \left.\frac{\partial^2 G_i(\mathbf{z})}{\partial z_i^2}\right|_{\mathbf{z}=1} \end{aligned} \tag{8}$$

Clearly, $\mathrm{Var}[X_i^i] = f_i(i,i)+f_i(i)-\left(f_i(i)\right)^2$.

Taking derivatives, the solution of (8) is given (see, Takagi [1986]) as a set of N^3 linear equations in the N^3 unknowns $\left\{f_i(j,k)\right\}$.

Cycle Time The mean cycle time is obtained from the balance equation $E[C] = \rho E[C]+d$. Hence,

$$E[C] = \frac{d}{1-\rho}\ .$$

The mean sojourn time of the server at channel i is $f_i(i)b_i = \rho_i E[C]$, and the number of jobs served in a cycle is clearly, $\sum_{i=1}^{N} f_i(i) = \left(\sum_{i=1}^{N}\lambda_i\right)E[C]$.

The PGF of L_i and Waiting Times

Consider the probability generating function, $Q_i(z) = E(z^{L_i})$, of the number of customers, L_i, left behind by an arbitrary departing customer from channel i in a polling system with arbitrary service regime. As the distributions of the number of customers in the system at epochs of arrival and epochs of departure are identical, then by the well known PASTA phenomenon (Poisson Arrivals See Time Averages), $Q_i(z)$ also stands for the generating function of the number of customers at channel i in a steady state condition at an arbitrary point of time.

Let T_i be the total number of customers served in channel i during a visit of the server to that channel, and let $L_i(n)$ $(n=1,2,\ldots,T_i)$, be the sequence of random variables denoting the number of customers that the n-th departing customer from channel i (counting from the moment that the channel was last polled) leaves behind it. Then the PGF of L_i is given by (see, Takagi [1986], p. 78)

$$Q_i(z) = \frac{E\left(\sum_{n=1}^{T_i} z^{L_i(n)}\right)}{E(T_i)}\ . \tag{9}$$

As $L_i(n) = X_i^i - n + A_i\left(\sum_{k=1}^{n} B_{ik}\right)$, the evaluation of the expression for $Q_i(z)$ becomes

$$Q_i(z)=\frac{1}{E(T_i)}E\left(\sum_{n=1}^{T_i} z^{X_i^i-n+A_i\left(\sum_{k=1}^{n}B_{ik}\right)}\right)=\frac{1}{E(T_i)}E\left(z^{X_i^i}\sum_{n=1}^{T_i} z^{-n+A_i\left(\sum_{k=1}^{n}B_{ik}\right)}\right)$$

$$= \frac{1}{E(T_i)} E\left(z^{X_i^i} \sum_{n=1}^{T_i} z^{-n} e^{-\lambda_i (\sum_{k=1}^{n} B_{ik})(1-z)} \right) = \frac{1}{E(T_i)} E\left(z^{X_i^i} \sum_{n=1}^{T_i} \left[\frac{\widetilde{B}_i(\lambda_i(1-z))}{z} \right]^n \right)$$

$$= \frac{1}{E(T_i)} E\left(z^{X_i^i} \cdot \frac{\widetilde{B}_i(\lambda_i(1-z))}{z} \cdot \frac{1 - \left[\frac{\widetilde{B}_i(\lambda_i(1-z))}{z} \right]^{T_i}}{1 - \frac{\widetilde{B}_i(\lambda_i(1-z))}{z}} \right)$$

$$= \frac{\widetilde{B}_i(\lambda_i(1-z))}{E(T_i)[z - \widetilde{B}(\lambda_i(1-z))]} E\left[z^{X_i^i - T_i} \left(z^{T_i} - \left[\widetilde{B}_i(\lambda_i(1-z)) \right]^{T_i} \right) \right] . \qquad (10)$$

Let W_{q_i} denote the *queueing* time of an arbitrary message at queue i, and let $W_i = W_{q_i} + B_i$ denote the sojourn (residence) time of a message in the system. As the messages left behind by a departing message from channel i have *all* arrived during its residence time W_i, we have

$$Q_i(z) = \sum_{k=0}^{\infty} P\left(\begin{matrix} \text{number of messages} \\ \text{at channel } i = k \end{matrix} \right) z^k = \sum_{k=0}^{\infty} z^k \int_0^{\infty} e^{-\lambda_i w} \frac{(\lambda_i w)^k}{k!} dP(W_i \le w)$$

$$= \widetilde{W}_i[\lambda_i(1-z)] = \widetilde{W}_{q_i}[\lambda_i(1-z)] \widetilde{B}_i[\lambda_i(1-z)]$$

Hence,

$$\widetilde{W}_{q_i}(s) = \frac{Q_i(1 - s/\lambda_i)}{\widetilde{B}_i(s)} \qquad (11)$$

For the Gated regime, $X_i^i = T_i$, and therefore

$$Q_i(z) = \frac{\widetilde{B}_i(\lambda_i(1-z))}{E(T_i)[z - \widetilde{B}_i(\lambda_i(1-z))]} \left(E[z^{X_i^i}] - E\left[(\widetilde{B}_i(\lambda_i(1-z)))^{X_i^i} \right] \right) \qquad (12)$$

(see also Takagi [1986], p. 109).

As $E(T_i) = E(X_i^i) = \lambda_i E[C]$, using (11) and (12) leads to

$$E(W_{q_i}) = \frac{E\left((X_i^i)^2\right) - E(X_i^i)}{2\lambda_i E(X_i^i)} (1 + \rho_i) = \frac{(1+\rho_i) f_i(i,i)}{2\lambda_i^2 E[C]} \qquad (13)$$

By Little's law, $E[L_i] = \lambda_i [E(W_{q_i}) + b_i]$.

4 Exhaustive Regime

To derive the PGF of the state of the system at a polling instant to channel $i+1$ we use the law of motion

$$X_{i+1}^j = \begin{cases} X_i^j + A_j \left(\sum_{k=1}^{X_i^i} \Theta_{ik} + D_i \right) , & j \neq i \\ A_i(D_i) , & j = i \end{cases} \qquad (14)$$

where Θ_i denotes the length of a *regular* busy period in an M/G$_i$/1 queue, and Θ_{ik} are all distributed as Θ_i. Then,

$$G_{i+1}(\mathbf{z}) = E\left[\prod_{j=1}^{N} z_j^{X_{i+1}^j}\right]$$

$$= E_{\mathbf{X}_i}\left[\prod_{\substack{j=1\\ j\neq i}}^{N} z_j^{X_i^j} \cdot E\left[\prod_{\substack{j=1\\ j\neq i}}^{N} z_j^{A_j\left(\sum_{k=1}^{X_i^i}\theta_{ik}\right)}\middle|\mathbf{X}_i\right]\right] \cdot E\left[\prod_{j=1}^{N} z_j^{A_j(D_i)}\right]$$

$$= E_{\mathbf{X}_i}\left[\prod_{\substack{j=1\\ j\neq i}}^{N} z_j^{X_i^j}\left(\widetilde{\theta}_i\left[\sum_{\substack{j=1\\ j\neq i}}^{N}\lambda_j(1-z_j)\right]\right)^{X_i^i}\right] \widetilde{D}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right]$$

Hence,

$$G_{i+1}(\mathbf{z}) = G_i\left(z_1, z_2, \ldots, z_{i-1}, \widetilde{\theta}_i\left[\sum_{\substack{j=1\\ j\neq i}}^{N}\lambda_j(1-z_j)\right], z_{i+1}, \ldots, z_N\right) \cdot \widetilde{D}_i\left[\sum_{j=1}^{N}\lambda_j(1-z_j)\right] \tag{15}$$

To get the N^2 values of $f_i(j)$ one can differentiate (15) or use directly (14). The result is

$$f_{i+1}(j) = \begin{cases} f_i(j) + \lambda_j E(\Theta_i) f_i(i) + \lambda_j d_i & j \neq i \\ \lambda_j d_i & j = i \end{cases} \tag{16}$$

where $E(\Theta_i) = b_i/(1-\rho_i)$ is the mean duration of a regular busy period at channel i.

The solution of (16) is

$$f_i(j) = \begin{cases} \lambda_j\left(\sum_{k=j+1}^{i-1}\rho_k\left(\frac{d}{1-\rho}\right) + \sum_{k=j}^{i-1} d_k\right) & j \neq i \\ \lambda_i(1-\rho_i)\left(\frac{d}{1-\rho}\right) & j = i \end{cases} \tag{17}$$

The interpretation of (17) is the following. The mean cycle time is *again* $E[C] = d/(1-\rho)$, which is derived from the *same* balance equation as for the Gated regime. The fraction of time that the server stays at channel i is ρ_i, hence, during the time interval since the server leaves (an empty) channel i until it arrives there again, the mean number of accumulated messages at i is $\lambda_i(1-\rho_i)E[C]$. For channel $j \neq i$, the total switchover times from the moment the server last exited the channel until it enters channel i is $\sum_{k=j}^{i-1} d_k$, and the mean time spent in each of the channels $k = j+1, j+2, , \ldots, i-1$, is $\rho_k E[C]$. Thus, the expected number of jobs accumulated at channel j when the server polls channel i is given by $\lambda_j\left(\sum_{k=j}^{i-1} d_k + \sum_{k=j+1}^{i-1}\rho_k\left(\frac{d}{1-\rho}\right)\right)$. The PGF of the number of messages at channel i can be obtained by using result (10). For the Exhaustive case, the number of customers served during a visit to channel i is $T_i = X_i^i + A_i\left(\sum_{k=1}^{X_i^i}\Theta_{ik}\right)$, so that $E(T_i) = f_i(i) + \lambda_i f_i(i) E(\Theta_i) = f_i(i)/(1-\rho_i)$, and by using (17), $E(T_i) = \lambda_i E[C]$.

The PGF of the number of messages at channel i at an arbitrary point of time is given by Takagi [1986], p. 79:

$$Q_i(z) = \frac{1}{\lambda_i E[C]} \cdot \frac{\widetilde{B}_i[\lambda_i(1-z)]}{z - \widetilde{B}_i[\lambda_i(1-z)]} [E[Z^{X_i^i}] - 1] \tag{18}$$

The mean number of messages at channel i and the mean queueing times are derived from (18),

$$E[L_i] = \rho_i + \frac{\lambda_i^2 b_i^{(2)}}{2(1-\rho_i)} + \frac{f_i(i,i)}{2\lambda_i(1-\rho_i)E[C]}$$

$$E[W_{q_i}] = \frac{\lambda_i b_i^{(2)}}{2(1-\rho_i)} + \frac{f_i(i,i)}{2\lambda_i^2(1-\rho_i)E[C]}$$

Again, the values of $f_i(i,i)$ have to be calculated numerically by solving a set of N^3 linear equations in the unknowns $f_i(j,k)$ derived (see (8)) by differentiating the PGFs in (15).

Remarks on Computational Methods

Several numerical procedures have been proposed for computing the mean waiting times in polling systems with Gated or with Exhaustive service regimes. The procedure mentioned above of determining the mean delay in various channels by solving a set of N^3 linear equations is called the Buffer Occupancy method. It is of high computational complexity, but can also be applied to solve models with switch-in times or with limited-service regimes. A more efficient procedure is known as the Station Time method (see Ferguson & Aminetzah [1985]). This is an iterative procedure which has been applied to a number of polling systems, but cannot be directly used for closed networks or for open systems with customers' routing. Sarkar & Zangwill [1989] have developed an algorithm for cyclic (Exhaustive or Gated) systems were the mean waiting times are obtained by solving a set of only N linear equations (thus requiring $O(N^3)$ computational steps). Recently, Konhein, Levy & Srinivasan [1993a] introduced a Descedant Set (DS) approach which is based on counting the number of descedants generated in the system by each customer. The method can be applied to variations of Exhaustive or Gated polling systems which are based on *fixed order* of visits, and can also be used to derive second and higher delay moments. It is claimed that the DS is superior to other methods due to its low computational complexity, even though it is based on the buffer occupancy variables. In a further effort to develop efficient computational methods, the same authors [1993b] introduced the Individual Station (IS) technique which, like the DS procedure, allows for the determining of mean waiting time at one or more selected nodes without having to obtain mean waiting times at all channels simultaneously. The IS is superior to the DS for systems with high utilization factor, while the DS would be preferred for systems with very large N.

5 Conservation Laws and Visit Frequencies

In an arbitrary single-server system (with single or multiple queues) when no work is generated or lost within the system, the amount of work present does not depend on the order of service – and hence equals the amount of work in the 'corresponding' system with a single queue and FCFS service discipline. This 'principle' of work conservation yields useful expression which we now discuss. Suppose that no switching times are incurred in our polling system, and assume cyclic or any order of the server's visits. Then it is well known (see, Kleinrock [1975]) that the expected amount of work in the system is constant, i.e.,

$$\sum_{i=1}^{N} b_i E[L_i] = \sum_{i=1}^{N} \rho_i E(W_i) = \rho \frac{\sum_{i=1}^{N} \lambda_i b_i^{(2)}}{2(1-\rho)} \equiv \overline{W} \ . \tag{19}$$

When switching times are incurred, Boxma & Groenendijk [1987] and Boxma [1989] have derived the so called 'pseudo-conservation laws' and showed that for an arbitrary polling system with *mixed* channels

$$\sum_{i=1}^{N} \rho_i E(W_i) = \overline{W} + \rho \frac{d^{(2)}}{2d} + \frac{d}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{N} \rho_i^2 \right] + \sum_{i=1}^{N} EM_i^{(1)} \tag{20}$$

where $EM_i^{(1)}$ is the expected *unfinished* work at the ith queue at an (arbitrary) instant of departure of the server from that queue. Result (20) holds for any service regime, and $EM_i^{(1)}$ depends *only* on the service discipline in channel i. For the Exhaustive service regime $EM_i^{(1)} = 0$ for every i, so that

$$\sum_{i=1}^{N} \rho_i E(W_i) = \overline{W} + \rho \frac{d^{(2)}}{2d} + \frac{d}{2(1-\rho)} \left[\rho^2 - \sum_{i=1}^{N} \rho_i \right] \ . \tag{21}$$

For the Gated regime, we use (7) and write

$$EM_i^{(1)} = \left[(f_i(i) b_i) \lambda_i \right] b_i = \rho_i^2 \left(\frac{d}{1-\rho} \right) \ .$$

Hence, for the Gated,

$$\sum_{i=1}^{N} \rho_i E(W_i) = \overline{W} + \rho \frac{d^{(2)}}{2d} + \frac{d}{2(1-\rho)} \left[\rho^2 + \sum_{i=1}^{N} \rho_i^2 \right] \ . \tag{22}$$

It follows that for the *same* set of parameters, whenever switchover times are incurred the mean amount of work in the system under the Exhaustive regime is *smaller* then that under the Gated discipline. Furthermore, expressions (21) and (22) enabled Boxma, Levy & Weststrate (see, Boxma [1991]) to develop 'good' visit frequencies of the server to the various channels so as to construct a polling table that will *reduce* the value of the expected amount of work in the system,

as expressed in (20). For the Exhaustive and for the Gated regimes the visit frequencies v_i^{exh}, and v_i^{gated} are given by

$$v_i^{\text{exh}} = \frac{\sqrt{\rho_i(1-\rho_i)/d_i}}{\sum_{j=1}^{N}\sqrt{\rho_j(1-\rho_j)/d_j}}$$

$$v_i^{\text{gated}} = \frac{\sqrt{\rho_i(1+\rho_i)/d_i}}{\sum_{j=1}^{N}\sqrt{\rho_j(1+\rho_j)/d_j}}$$

For example, in a 3-channel case for which the calculated visit frequencies are 0.52, 0.32 and 0.16, the approximate visit frequencies are 1/2, 1/3 and 1/6, respectively, such that a (periodic) polling table of size 6 is constructed with the order of visits [1,2,1,3,1,2].

Another approach in the attempt to control and otimize the visit frequencies of the server to the various channels is the Cyclic Bernoulli Polling (CBP) introduced by Altman & Yechiali [1993]. The server moves *cyclically* among the N channels where *change-over* times between stations are composed of two parts: *walking* times required to 'move' from one channel to another and *switch-in* times that are incurred *only* when the server actually enters a station to render service. Upon arrival to channel i the server switches in with probability p_i, or moves on to the next channel (with probability $1-p_i$) without serving any customer. Altman & Yechiali analyzed the Gated and Exhaustive regimes and defined a mathematical program to find the *optimal* values of the switch-in probabilites $\{p_i\}_{i=1}^{N}$ so as to *minimize* the expected amount of unfinished work in the system. Any CBP scheme for which the optimal p_i's are not equal to 1 yields a *smaller* amount of expected unfinished work in the system than that in the standard cyclic procedure with equivalent parameters. They showed that even in the case of a *single queue*, it is *not always* true that $p_1 = 1$ is the best strategy, and derived conditions under which it is optimal to have $p_1 < 1$.

6 Dynamic Control of Server's Visits: Hamiltonian Tours

A basic question that arises when planning efficient polling systems concerns the order of visits performed by the server. For *static* order one can think of a 'good' polling table that optimizes some measure of effectiveness. Steps in this direction were taken, as mentioned in section 5, by various authors. However, a more reaching goal is to control the system *dynamically*, so that the server will modify its order of visits in response to the stochastic evolution of the system. In other words, the general control problem facing the server when it exits a specific channel, is "*which of the channels to visit next*?". In trying to solve this problem Browne & Yechiali [1989a], [1989b] developed and formulated semi-Markov Decision Processes (SMDP) for the Gated and for the Exhaustive regimes. They derived a set of optimality equations where the objective is to minimize mean weighted waiting costs. However, these equations are non-tractable,

so that one should look for alternative methods. An appealing approach is to look for semi-dynamic control schemes. The idea is to dispatch the server to perform Hamiltonian tours, each tour different from its previous one, depending on the state of the system at the *beginning* of the tour, so as to optimize some measure of effectiveness.

Specifically, suppose that at the beginning of a cycle the state of the system is $(n_1, n_2, \ldots, n_N)$, where n_i is the number of jobs waiting in channel i $(1 \leq i \leq N)$. Assume for the moment that switching times between channels are negligible. The objective is to choose a path (Hamiltonian tour) through the queues so as to *minimize* the expected time of traversing this path. It was shown by Browne & Yechiali [1989a], [1989b] that for *both* service disciplines – the fully Gated and the fully Exhaustive – 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 $(n_1, n_2, \ldots, n_N)$ 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 Hamiltonian cycle a *new* system-state is observed, say $(n'_1, n'_2, \ldots, n'_N)$, 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.

To understand the above surprising result Browne & Yechiali [1990] studies a *general* scheduling problem with a *linear growth of work*, as follows.

Consider a single-processor system with N jobs waiting to be performed sequentially. Let a_i be the *initial* (expected) processing time requirement of job i $(i = 1, 2, \ldots, N)$, called the 'core'. If job i is delayed and is started at time t, then its processing requirement *grows linearly* with the delay to

$$Y_i(t) = a_i + \alpha_i t$$

where α_i is the *growth rate* of work requirement by job i. Consider the processing order $\pi_0 = (1, 2, \ldots, N)$, and let Y_i denote the *actual* processing length of job i under π_0. Let $S_k = \sum_{i=1}^{k} Y_i$ be the completion time of job k under π_0 $(S_0 = 0)$. Then $Y_j = a_j + \alpha_j S_{j-1}$. By adding S_{j-1} to both sides we obtain a set of difference equations

$$S_j - (1 + \alpha_j) S_{j-1} = a_j \qquad (j = 1, 2, \ldots, N) \tag{23}$$

The solution of (23) is

$$S_j = \sum_{i=1}^{j} a_i \prod_{r=i+1}^{j} (1 + \alpha_r) \qquad (j = 1, 2, \ldots, N) \tag{24}$$

so that the *makespan* is $S_N = S_N(\pi_0) = \sum_{i=1}^{N} a_i \prod_{r=i+1}^{N}(1+\alpha_r)$.

The objective is to find a visit order π that *minimizes the makespan* $S_N(\pi)$ over all $n!$ possible permutations π.

Consider now the processing sequence $\pi_1 = (1, 2, \ldots, j-1, j+1, j, j+2, \ldots, N)$, where the order of jobs j and $j+1$ is interchanged. The corresponding makespan is $S_N(\pi_1)$. Then, it is easy to show that $S_N(\pi_0) < S_N(\pi_1)$ iff $a_j/\alpha_j < a_{j+1}/\alpha_{j+1}$. That is, the makespan is *minimized* (maximized) if we process the jobs in an increasing (decreasing) order of the ratio index a_i/α_i, i.e., 'core' divided by 'growth rate'.

Consider again the Gated regime. If $(n_1, n_2, \ldots, n_N)$ is the state of the system at the *start* of the Hamiltonian tour, then $a_i = n_i b_i$. The growth rate (i.e., the amount of work flowing to channel i per unit of time) is ρ_i. Hence,

$$\frac{a_i}{\alpha_i} = \frac{n_i b_i}{\lambda_i b_i} = \frac{n_i}{\lambda_i} \ .$$

For the Exhaustive regime, $a_i = n_i E(\Theta_i) = n_i\left(\frac{b_i}{1-\rho_i}\right)$, whereas $\alpha_i = \frac{\rho_i}{1-\rho_i}$ (the duration of time that the server has to stay in channel i grows linearly at a rate of $\frac{b_i}{1-\rho_i}$ for each new arrival. As the rate of arrivals is λ_i, we have $\alpha_i = \frac{\rho_i}{1-\rho_i}$). Thus, for the Exhaustive case

$$\frac{a_i}{\alpha_i} = \frac{n_i\left(\frac{b_i}{1-\rho_i}\right)}{\left(\frac{\rho_i}{1-\rho_i}\right)} = \frac{n_i b_i}{\rho_i} = \frac{n_i}{\lambda_i}$$

which is the *same* index as for the Gated regime.

We can now reintroduce the switchover and switch-in times. For illustration, assume a star-configuration of the system. Recall that D_i is the switchover time out of i and R_j denotes the switch-in duration into j. Then, for the Gated regime, assuming gating occurs after switch-in is completed,

$$\begin{aligned} a_i &= n_i b_i + (1+\rho_i) r_i + d_i \\ \alpha_i &= \rho_i \ , \end{aligned}$$

so that $a_i \alpha_i = \left[n_i b_i + (1+\rho_i) r_i + d_i\right]/\rho_i$. For the Exhaustive

$$\begin{aligned} a_i &= \frac{r_i}{1-\rho_i} + \frac{n_i b_i}{1-\rho_i} + d_i \\ \alpha_i &= \rho_i/(1-\rho_i) \ , \end{aligned}$$

so that $a_i/\alpha_i = \left[r_i + n_i b_i + d_i(1-\rho_i)\right]/\rho_i$.

It should be emphasized that the scheduling principle a_i/α_i can be applied to *any* system with a mixed set of service regimes among the channels: Gated, Exhaustive, Binomial or Bernoulli Gated, Binomial or Bernoulli Exhaustive, etc. (see, Yechiali [1991]). All that one has to do is to calculate (once) α_i for every channel, and then, at the beginning of each new Hamiltonian tour, to calculate the current 'core' a_i at each channel. Then, performing a visit tour that follows an *increasing* (decreasing) order of a_i/α_i will *minimize* (maximize) cycle duration.

Browne & Yechiali [1991] further employed the above ideas to achieve dynamic scheduling in systems with only a unit buffer at each channel.

7 The Globally-Gated Regime

A drawback both of the Gated and the Exhaustive regimes is that they are not 'fair' with regard to the FCFS principle. To help resolve this dichotomy, Boxma, Levy & Yechiali [1992] 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 mathematical model which 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 N infinite-buffer channels, the rate of offered load to queue i is $\rho_i = \lambda_i b_i$ and the total system load-rate is $\rho \equiv \sum_{i=1}^{N} \rho_i$. When leaving queue i and before starting service at the next queue, the server incurs a random switchover period D_i. The total 'walking' time in a cycle is $D \equiv \sum_{i=1}^{N} D_i$. (Clearly, other 'Global' versions, such as Globally Exhaustive, can be easily imagined and analyzed.)

Cycle Time

Assume, without loss of generality, that a cycle starts from channel 1. Let $(X_1^1, X_1^2, \ldots, X_1^j, \ldots, X_1^N) = (X_1, X_2, \ldots, X_j, \ldots, X_N)$ be the state of the system at the beginning of the cycle. Then, the cycle duration is

$$C = D + \sum_{j=1}^{N} \sum_{k=1}^{X_j} B_{jk} \ .$$

The LST of C is derived as follows

$$E\big(e^{-wC} \mid (X_1, X_2, \ldots, X_N)\big) = \widetilde{D}(w) \prod_{j=1}^{N} \big(B_i(w)\big)^{X_j} \ . \tag{25}$$

On the other hand, the length of a cycle determines the joint queue-length distribution at the beginning of the next cycle. Hence

$$E\Big[\prod_{j=1}^{N} z_j^{X_j}\Big] = E_C\left[E\Big[\prod_{j=1}^{N} z_j^{X_j} \mid C\Big]\right] = E_C\left[\exp\Big[-\sum_{j=1}^{N} \lambda_j (1 - z_j) C\Big]\right]$$
$$= \widetilde{C}\Big[\sum_{j=1}^{N} \lambda_j (1 - z_j)\Big] \ . \tag{26}$$

Combining (25) and (26)

$$\widetilde{C}(w) = \widetilde{D}(w)\widetilde{C}\left[\sum_{j=1}^{N} \lambda_j \left(1 - \widetilde{B}_j(w)\right)\right] . \tag{27}$$

The mean cycle time is derived from (27)

$$E[C] = d + \left(\sum_{j=1}^{N} \lambda_j b_j\right) E[C] .$$

That is, $E[C] = d/(1-\rho)$, as for the Gated and the Exhaustive regimes. The second moment of C is derived from (27)

$$E[C^2] = \left[d^{(2)} + \left(2d\rho + \sum_{j=1}^{N} \lambda_j b_j^{(2)}\right) E[C]\right] \Big/ (1-\rho^2) . \tag{28}$$

Let C_P and C_R denote, respectively, the past and residual duration of a cycle. It is well known that

$$\widetilde{C}_P(w) = \widetilde{C}_R(w) = \frac{1 - \widetilde{C}(w)}{wE[C]}$$

and $E[C_P] = E[C_R] = \frac{E[C^2]}{2E[C]}$.

Pseudo-Conservation law

To derive a pseudo-conservation law we use (20) and the observation that for the cyclic GG regime, $E(X_j) = \rho_j E[C]$ and

$$EM_j^{(1)} = \rho_j \left[\sum_{i=1}^{j} \left[E(X_i)b_i + \sum_{i=1}^{j-1} d_i\right]\right] = \rho_j \sum_{i=1}^{j-1} \left(\rho_i \frac{d}{1-\rho} + d_i\right) + \rho_j^2 \frac{d}{1-\rho} . \tag{29}$$

Substituting (29) in (20) yields

$$\sum_{j=1}^{N} \rho_j E(W_j) = \overline{W} + \rho \frac{d^{(2)}}{2d} + \frac{d}{1-\rho}\rho^2 + \sum_{j=2}^{N} \rho_j \sum_{i=1}^{j-1} d_i . \tag{30}$$

Waiting Times

Consider an arbitrary job K at channel k. The cycle age at the job's arrival instant is C_P. The job's waiting time is composed of (i) the residual cycle time C_R, (ii) the service times of all customers who arrive at channels 1 to $k-1$ during the cycle in which K arrives, (iii) the switchover times of the server through channels 1 to k, and (iv) the service times of all customers that arrive at channel k during the past part of the cycle, C_P. Then

$$E(W_k) = E[C_R] + \sum_{j=1}^{k-1} \rho_j \left(E[C_P] + E[C_R]\right) + \sum_{j=1}^{k-1} d_j + \rho_k E[C_P]$$

$$= \left(1 + 2\sum_{j=1}^{k-1} \rho_j + \rho_k\right) E[C_R] + \sum_{j=1}^{k-1} d_j \ . \tag{31}$$

It readily follows that

$$E(W_{k+1}) - E(W_k) = (\rho_{k+1} + \rho_k) E[C_R] + d_k$$

so that, for the cyclic GG regime, we *always* have

$$E(W_1) < E(W_2) < \ldots < E(W_N) \ . \tag{32}$$

Boxma, Weststrate & Yechiali [1993] extended the cyclic GG model to the case where the server suffers periods of breakdown, and applied the results to real-world repairman problems where both preventive and corrective maintenance actions are considered.

Static Optimization

Let c_k be the cost rate of a waiting job at queue k. Then, the mean weighted waiting cost of an arbitrary job in the system is

$$\sum_{k=1}^{N} \left(\lambda_k \Big/ \sum_{j=1}^{N} \lambda_j\right) c_k E(W_k) \ . \tag{33}$$

By substituting (31) into (33) and using an interchange argument it follows that the cycle which *minimized* (33) is determined by an *increasing* order of the index

$$u_j = \frac{2E[C_R]\rho_j + d_j}{\lambda_j c_j}$$

If d_j is negligible, the above index reduces to the index b_j/c_j, which is the well known "$c\mu$" rule.

Dynamic Control

An important characteristic of the GG regime is that the order of visits selected for one cycle *does not affect* the future stochastic behaviour of the system. Moreover, *any* Hamiltonian tour that starts from state $(n_1, n_2, \ldots, n_M)$ yields the *same* cycle duration $C(n_1, n_2, \ldots, n_N)$. Thus, if we consider the costs incurred *during the cycle* by the customers *present* at its initiation and add to it the costs incurred along that cycle by the *new* arrivals, then the *long-run miminal cost* can be achieved by determining a new optimal Hamiltonian tour for each cycle *independently*.

The mean total weighted cost incurred during a cycle starting with $(n_1, n_2, \ldots, n_N)$ is

$$\sum_{k=1}^{N} c_k \left[n_k \sum_{j=1}^{k-1} (n_j b_j + d_j) + b_k \sum_{i=1}^{n_k - 1} i \right] \tag{34}$$

$$+ \sum_{k=1}^{N} c_k \lambda_k E\left[C(n_1, n_2, \ldots, n_N)^2\right]/2$$

where the first term is the contribution to total cost incurred by the customers present at the cycle beginning, and the second term is due to the customers arriving during that cycle (see, Yechiali [1976]). The only term in (34) that depends on the order of visits is $\sum_{k=}^{N} c_k n_k \sum_{j=1}^{k-1} (n_j b_j + d_j)$. It follows (by an interchange argument) that the optimal order of visits that minimizes expected total costs of the coming cycle is determined by an *increasing* order of the (Gittins) index

$$\frac{n_j b_j + d_j}{n_j c_j} .$$

Again, for negligible d_j this index reduces to the "$c\mu$" rule (i.e., b_j/c_j).

8 Elevator-Type Polling

In an Elevator-type (scan) polling mechanism the server alternates between 'up' and 'down' cycles. In an 'up' cycle it visits the channels in the order $1, 2, \ldots, N-1, N$, and in a 'down' cycle the order of visits is reversed to $N, N-1, \ldots, 2, 1$. This type of polling procedure is encountered in many applications, e.g., it models a common scheme of addressing a hard disk for writing (reading) information on (from) different tracks. It is important to note that the Elevator-type polling saves the return walking time from channel N to channel 1. A comprehensive analysis of Elevator-type polling with four different service regimes can be found in Shoham & Yechiali [1992]. Here we present the Globally-Gated (GG) regime as discussed in Altman, Khamisy & Yechiali [1992].

According to the Elevator-type polling with GG service regime all channels are gated off at the beginning of the 'up' cycle, where the system-state is $(n_1^{\text{up}}, n_2^{\text{up}}, \ldots, n_N^{\text{up}})$, and the server resides in channel i for n_i^{up} regular service durations. At the end of the up cycle all channels are gated again, the system-state is $(n_1^{\text{down}}, n_2^{\text{down}}, \ldots, n_N^{\text{down}})$, and the server starts its down cycle, serving n_i^{down} customers in channel i. We assume that the down walking time from channel $i+1$ to channel i has the same distribution as the up walking time D_i from channel i to channel $i+1$. A key observation is that arbitrary up and down cycles have the *same* distribution, which differs from its *cyclic* GG counter-part only in that it is smaller by the 'saved' walking time D_N. Hence, the results derived for the cycle time distribution (27) and for mean waiting times (31) in a *cyclic* GG regime are directly applicable to the Elevator case, with $D_N = 0$.

Waiting Times

Consider an arbitrary job at channel k. Since all cycles are distributed *alike*, the job arrives during an up or a down cycle with equal probabilites, 0.5. Hence, its mean waiting time is given by

$$E(W_k) = 0.5E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves down}\end{matrix}\right) + 0.5E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves up}\end{matrix}\right) . \tag{35}$$

The expression for $E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves down}\end{matrix}\right)$ is given by (31), with $d_N = 0$, whereas, by reversing the order of visits, we have

$$E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves up}\end{matrix}\right) = \left(1 + 2\sum_{j=k+1}^{N} \rho_j + \rho_k\right) E[C_R] + \sum_{j=k}^{N-1} d_j . \tag{36}$$

Combining (35) with (31) and (36) yields the *surprising result*

$$E(W_k) = (1+\rho)E[C_R] + 0.5d . \tag{37}$$

That is, expected waiting times are *equal* in *all channels*. This is the only-known non-symmetric polling system that exhibits such a "fairness" phenomenon. An explanation of result (37) is the following. An arbitrary arrival has to wait, on the average, $E[C_R]$ units of time until the cycle (up or down) in which it arrives terminates. Then, it waits until the server moves back to channel k, which requires, on the average (taking into account both directions), $\frac{1}{2}\left[\left(E[C_R] + E[C_p]\right)\rho + d\right]$ units of time.

Optimal Arrangement of Channels The interesting result that $E(W_k)$ is the same for all channels, independent of their location, leads to considering channels' arrangement such that the *variation* in waiting times will be small.

Let $a_i = 2E[C_R]\rho_i + d_i$ $(i = 1, 2, \ldots, N)$. Then

$$E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves down}\end{matrix}\right) = E[C_R](1+\rho_k) + \sum_{i=1}^{k-1} a_i$$

$$E\left(W_k \middle| \begin{matrix}\text{server}\\ \text{moves up}\end{matrix}\right) = E[C_R](1+\rho_k) + \sum_{i=k+1}^{N} a_i + d_k$$

Let $\Delta_k = E(W_k \mid \text{down}) - E(W_k \mid \text{up}) = \sum_{i=1}^{k-1} a_i - \sum_{i=k+1}^{N} a_i - d_k$. Now, $\Delta_1 = -\sum_{i=2}^{N} a_i - d_1 < 0$, $\Delta_N = \sum_{i=1}^{N-1} a_i > 0$ (recall that $d_N = 0$), and Δ_k is a monotone increasing function of k.

One goal is to arrange the channels such that $\max_{1\le k\le N}\{|\Delta_k|\}$ is as small as possible. Clearly

$$\begin{aligned}\max_{1\le k\le N}\{|\Delta_k|\} &= \max\{|\Delta_1|, |\Delta_N|\}\\ &= \max\left\{\sum_{i=1}^{N} a_i - 2E[C_R]\rho_1, \sum_{i=1}^{N} a_i - 2E[C_R]\rho_N\right\}\end{aligned} \tag{38}$$

It follows from (38) that $\max_{1 \leq k \leq N} \{|\Delta_k|\}$ is *minimized* if channel 1 is the one with the *highest* value of ρ_i and channel N is the one with the *second highest* value of ρ_i (or vice versa).

9 Future Directions of Research

We have presented methods of analysis for single-server, continuous-time, infinite buffers polling systems, and studied several control and optimization problems. Difficult problems are finite-capacity models and limited service regimes, for which only partial solutions are given in the literature (see, bibliography in Takagi [1990]). A few authors have studied polling systems with multiple servers, and recently Browns & Weiss [1992] investigated dynamic priority rules for a system with parallel servers.

All the systems mentioned above are *open*, with external arrivals, where jobs exit the system after service completion. Closed systems should also be investigated, and only recently Altman & Yechiali [1992] analyzed such systems with Gated, Exhaustive or Globally-Gated service regimes.

For other future directions of research we state a recent 'call for papers' on "Discrete-Time Models and Analysis Methods":

> "The past few years have seen an increasing interest in discrete-time models and their solution techniques. One of the driving forces behind this area has been new developments in telecommunications, espacially in high-speed metropolitan area and wide area networks. Tehcnological advances and user demands have shifted the evolution of telecommunication systems towards integrated networks where information is transferred in small, ofted fixed-size, packets, slots or cells (e.g., ATM networks, high-speed LANs and MANs such as DQDB, etc...), operting in a discrete-time environment. The resulting mathematical models of such slotted systems, crucial for the evaluation of design alternatives and their dimensioning, are discrete-time models. The complexity of the stochastic processes involved (e.g., arrival and departure processes) and of the system operation mechanisms (e.g., service mechanism, access protocol, etc...) pose an exciting challenge for the development of efficient and tractable methods for deriving the main performance measures of these systems.
>
> Papers are solicited on discrete-time models and their analysis methods, in particular on, but not restricted to, the following topics:
> — Discrete-time queueing models (polling systems, priority systems, multiserver systems, vacation models, etc...).
> — Exact and approximate solution methods for discrete-time queueing models, with emphasis on the efficiency and the numerical tractability of these methods.
> — Stochastic processes as traffic models for performance studies (taking into account the diversity of time scales, correlations between arrivals, etc...)

— Discrete-time markov chains and their analysis methods".

Naturally, we add to the above topics the interesting and challenging problems of control and optimization of such systems.

Bibliography

1. Altman, E., Blanc, H., Khamisy, A., Yechiali, U.: Gated-type polling systems with walking and switch-in times. Technical Report, Dept. of Statistics & OR, Tel Aviv University 1992.
2. Altman, E., Khamisy, A., Yechiali, U.: On elevator polling with globally-gated regime. Queueing Systems **11** (1992) 85-90.
3. Altman, E., Yechiali, U.: Polling in a closed network. Technical Report SOR-92-14, Dept. of Statistics & OR, New York University 1992.
4. Altman, E., Yechiali, U.: Cyclic Bernoulli polling. ZOR-Methods and Models of Operations Research 38 (1993).
5. Boxma, O.J.: Workloads and waiting times in single-server systems with multiple customer classes. Queueing Systems **5** (1989) 185-214.
6. Boxma, O.J.: Analysis and optimization of polling systems. In: Cohen, J.W., Pack, C.D. (Eds.) Queueing, Performance and Control in ATM. North-Holland, 1991, pp.173-183.
7. Boxma, O.J., Groenendijk, W.P.: Pseudo conservation laws in cyclic service systems. Journal of Applied Probability **24** (1987) 949-964.
8. Boxma, O.J., Levy, H., Yechiali, U.: Cyclic reservation schemes for efficient operation of multiple-queue single-server Systems. Annals of Operations Research **35** (1992) 187-208.
9. Boxma, O.J., Weststrate, J.A., Yechiali, U.: A globally gated polling system with server interruptions, and applications to the repairman problem. Probability in the Engineering and Informational Sciences **7** (1993).
10. Browne, S., Yechiali, U.: Dynamic priority rules for cyclic-type queues, Advances in Applied Probability **21** (1989a) 432-450.
11. Browne, S., Yechiali, U.: Dynamic routing in polling systems. In: M. Bonatti (Ed.) Teletraffic Science for New Cost-Effective Systems, Networks and Services. North-Holland, 1989b, pp.1455-1466.
12. Browne, S., Yechiali, U.: Scheduling deteriorating jobs on a single processor. Operations Research **38** (1990) 495-498.
13. Browne, S., Yechiali, U.: Dynamic scheduling in single-server multiclass service systems with unit buffers. Naval Research Logistics **38** (1991) 383-396.
14. Browne, S., Weiss, G.: Dynamic priority rules when polling with multiple parallel servers. Operations Research Letters **12** (1992) 129-137.
15. Cooper, R.B. Murray, G.: Queues served in cyclic order. Bell System Technical Journal **48** (1969) 675-689.
16. Cooper, R.B.: Queues served in cyclic order: waiting times. Bell System Technical Journal **49** (1970) 399-413.
17. Eisenberg, M.: Queues with periodic service and changeover time. Operations Research **20** (1972) 440-451.
18. Ferguson, M.J., Aminetzah, Y.J.: Exact results for nonsymmetric token ring systems. IEEE Transactions on Communications **33** (1985) 223-231.
19. Kleinrock, L.: Queueing Systems, Vol. 1: Theory. John Wiley, 1975.

20. Konheim, A.G., Levy, H., Srinivasan: Descendant set: an efficient approach for the analysis of polling systems. IEEE Transactions on Communications (to appear 1993a).
21. Konheim, A.G., Levy, H., Srinivasan: The individual station technique for the analysis of polling systems. Technical Report, 1993b.
22. Levy, H., Sidi, M.: Polling systems: applications, modeling and optimization. IEEE Transactions on Communications 8 (1990) 1750-1760.
23. Sarkar, D., Zangwill, W.I.: Expected waiting time for nonsymmetric cyclic queueing systems – exact results and applications. Management Science 35 (1989) 1463-1474.
24. Shoham, R., Yechiali, U.: Elevator-type polling systems. Technical Report, Dept. of Statistics & OR, Tel Aviv University, 1992.
25. Takagi, H.: Analysis of Polling Systems. MIT Press, 1986.
26. Takagi, H.: Queueing analysis of polling models: an update. In: Takagi, H. (ed.) Stochastic Analysis of Computer and Communications Systems. North Holland, 1990, pp.267-318.
27. Yechiali, U.: A new derivation of the Khintchine-Pollaczek formula. In: Haley, K.B. (Ed.) Operational Research '75. North Holland, 1976, pp.261-264.
28. Yechiali, U.: Optimal dynamic control of polling systems. In: Cohen, J.W., Pack, C.D. (Eds.) Queueing, Performance and Control in ATM. North Holland, 1991, pp.205-217.