CYCLIC RESERVATION SCHEMES FOR EFFICIENT OPERATION OF MULTIPLE-QUEUE SINGLE-SERVER SYSTEMS

Onno J. BOXMA
CWI, P.O. Box 4079, 1009 AB Amsterdam, The Netherlands;
Faculty of Economics, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands
Hanoch LEVY
Department of Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel
Uri YECHIALI
Department of Statistics, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel

Abstract

We study two new cyclic reservation schemes for the efficient operation of systems consisting of a single server and multiple queues. The schemes are the Globally Gated regime and the Cyclic-Reservation Multiple-Access (CRMA). Both procedures possess mechanisms for prioritizing the queues and lend themselves to a closed-form analysis. The combination of these two properties allows for effective and efficient operation of the systems, for which we provide a thorough delay analysis and derive simple rules for optimal operation.

1. Introduction

Queueing systems consisting of N queues served by a single server who incurs switchover periods when moving from one queue to another have been widely studied in the literature and used as a central model for the analysis of a large variety of applications in the areas of telecommunications, computer networks, manufacturing, etc. Very often, such applications are modeled as a polling system in which the server visits the queues in a cyclic or some other pre-determined order. An issue that is crucial for all these applications is the need for a service scheme which will allow designers to prioritize the different queues and thus affect and optimize overall system performance.

In many of these applications, as well as in most polling models, it is common to prioritize the queues by controlling the amount of service given to each queue during the server's visit. A common service policy is the *gated regime*, in which all (and only) the customers present when the server starts visiting the queue are served during that visit. A similar policy is the *exhaustive procedure* in which at each visit the server attends the queue until it becomes completely empty. Another policy is the *limited service* which uses a parameter $k_i \geq 1$ for queue i so that at most k_i customers are served at each visit to that queue. The disadvantage of the gated and the exhaustive policies is that they cannot be used to prioritize the queues. The limited service, in contrast, possesses very strong mechanisms for prioritizing the queues but suffers considerably from the fact that it is not amenable to efficient analysis (the numerical derivation of the mean delay in this system requires a computational effort that is exponential in the number of queues and thus is feasible only for systems consisting of only few queues (see e.g. Leung [18], Blanc [3]), and thus it is very difficult to find an efficient operating strategy for it.

The analysis of polling systems with gated or exhaustive service has been presented in numerous papers in the literature, e.g. Cooper [11], Eisenberg [13], Konheim and Meister [16] – to mention a few, summarized in a book by Takagi [22], and further surveyed recently by Levy and Sidi [17] and Takagi [23]. This paper focuses on the study of schemes which (1) possess mechanisms for prioritizing the queues, and (2) lend themselves to effective analysis. The combination of these two properties allows designers to affect the system performance as well as to predict the outcome of their design. As a result, the operation of these systems can be optimized at the design stage.

Specifically, we focus on two schemes: (1) the Cyclic-Reservation Multiple-Access (CRMA), which has been proposed by Nassehi [20] for the operation of high-speed local area networks, and (2) the Globally-Gated (GG) scheme, which is a new procedure introduced in this paper and resembles the cyclically Gated service policy. The two schemes are represented by a single server who serves customers arriving at N different queues. In both procedures, a cyclic reservation mechanism is used to control the service of these queues.

In the CRMA, a *collector* is sent periodically for a cyclic tour among the queues to collect their current reservations. After a completion of a tour by a collector, its reservations are augmented to a "central" queue and served (once the service of previous reservations has been completed) in the order collected. Typically, a new collector is sent *before* the previous one completes its tour. The CRMA is an access scheme for Gbit/s LANs and MANs which offers a high throughput efficiency and can also be operated on a dual-bus configuration. Another application is in the area of Flexible Manufacturing Systems, where the queues represent work stations producing units or parts that should be completed in a "central" location. The "collectors" move around and bring the jobs to the central location for final processing.

The Globally-Gated (GG) scheme uses a time-stamp mechanism for its cyclic reservation: 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 there serves all the customers who were present at the 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.

The GG scheme may also serve as a model for estimating the performance of more complicated real systems such as a bridge-queue between rings in interconnected networks. One can try to analyze the interference between the rings by assuming that there is a GG service at each individual ring, with the bridge-queue being at the first position in every ring. The GG model is also used in the analysis of Elevator-type polling mechanisms that represent the bi-directional moves of a head addressing a hard disk for writing information on, or reading data from, different tracks (see e.g. Altman et al. [1]).

For both systems, we provide delay analyses including the derivation of the mean value and the Laplace–Stieltjes Transform (LST) of the waiting times incurred in the different queues.

Relatively few studies have dealt with efficient operation of polling systems. We mention Meilijson and Yechiali [19], in which the server is allowed to switch from one queue to another after *each* service completion; Hofri and Ross [15], in which dynamic rules for optimal operation of two-queue systems are studied; Browne and Yechiali [8,9], in which semi-*dynamic* rules for minimizing the cycle time are presented; and Boxma et al. [6,7], in which rules for selecting efficient *static* visit orders are derived. In the present study, we also consider the control problem of the systems and derive simple rules (static and dynamic) for their optimal operation as a function of the parameters and the waiting costs at the different queues.

The structure of the paper is as follows: In section 2, we provide a delay analysis of the cyclic polling system with globally gated service. The section consists of a model description, cycle time analysis, pseudoconservation law derivation and waiting time analysis. In section 3, we analyze the CRMA scheme. The section consists of a description of the system and its prototype queueing model, for which a delay analysis is provided. In section 4, we derive optimal operational rules for both systems and compare them to each other and to other schemes as well. We also discuss some "fairness" issues.

2. Performance analysis of cyclic polling with globally gated service

In this section, we study a cyclic polling system served under the Globally Gated policy, and analyze its performance. This policy resembles in nature the gated service regime, but differs from it in the fact that the gating mechanism is applied at the *same time* for *all* queues. The advantage of this policy lies in its simple structure which allows for derivation of closed-form mean-delay expressions and for effective optimization of the system.

2.1. MODEL DESCRIPTION

We consider a system consisting of N infinite-buffer queues $Q_1, \ldots, Q_N$ and a single server. Customers arrive at Q_j according to a Poisson process rate of λ_j. The service time distribution at Q_j is $B_j(\cdot)$ with first and second moments $\beta_j, \beta_j^{(2)}$, and with LST $\beta_j(\cdot)$. The offered load to this queue is $\rho_j = \lambda_j \beta_j$, and the total system load is $\rho = \sum_{j=1}^{N} \rho_j$. The server moves among the queues in a cyclic order; when leaving Q_j and before moving to the next queue, the server incurs a switchover period whose duration is a random variable S_j with first two moments s_j and $s_j^{(2)}$, and with LST $\sigma_j(\cdot)$. All arrival processes, service times and switchover periods are independent. The total switchover time in a cycle is a random variable $S = \sum_{j=1}^{N} S_j$ whose first two moments are s and $s^{(2)}$, and its LST is given by $\sigma(\omega) = \prod_{j=1}^{N} \sigma_j(\omega)$, Re $\omega \geq 0$.

The service discipline used by the server is the Globally-Gated (GG) discipline, which works as follows: At the cycle-beginning, namely, when the server reaches Q_1, all customers present at $Q_1, \ldots, Q_N$ are marked. During the coming cycle (i.e. the visit of queues $Q_1, \ldots, Q_N$), the server serves all (and only) the marked customers; customers who meanwhile arrive at the queues will have to wait until being marked at the next cycle-beginning, and will be served at the next cycle. Since at every cycle the server serves all the work that arrived during the previous cycle, the condition for ergodicity is $\rho < 1$. (Comparison with the case of zero switchover times immediately shows that $\rho < 1$ is a necessary condition; in section 2.2 and the appendix, it will become clear that $\rho < 1$ is also sufficient.) Similarly to the regular gated or exhaustive schemes, the server keeps switching from one queue to the next even when there are no customers present in the system.

It should be noted here that "global" versions of some other service policies (e.g. globally exhaustive) can be easily imagined and analyzed.

2.2. CYCLE TIMES

Let $X_1, \ldots, X_N$ denote the queue lengths at the start of an arbitrary cycle. Clearly, wih C the length of a cycle,

$$E[\mathrm{e}^{-\omega C} | X_1, \ldots, X_N] = E[\mathrm{e}^{-\omega S}] \prod_{j=1}^{N} \beta_j^{X_j}(\omega), \quad \text{Re } \omega \geq 0. \tag{2.1}$$

In its turn, the length of a cycle determines the joint queue-length distribution at the beginning of the next cycle:

$$E[z_1^{X_1} \ldots z_N^{X_N} | C = t] = \exp\left(-\sum_{j=1}^{N} \lambda_j (1 - z_j) t \right), \quad |z_1| \leq 1, \ldots, |z_N| \leq 1. \tag{2.2}$$

Unconditioning we find, for $|z_1| \leq 1, \ldots, |z_N| \leq 1$,

$$E[z_1^{X_1} \dots z_N^{X_N}] = \gamma\left(\sum_{j=1}^{N} \lambda_j(1-z_j)\right), \tag{2.3}$$

with

$$\gamma(\omega) = E[\mathrm{e}^{-\omega C}], \quad \mathrm{Re}\ \omega \geq 0. \tag{2.4}$$

From (2.1) and (2.3),

$$\begin{aligned}\gamma(\omega) &= E\,[\mathrm{e}^{-\omega S}]E\left[\beta_1^{X_1}(\omega)\dots\beta_N^{X_N}(\omega)\right]\\ &= E\,[\mathrm{e}^{-\omega S}]\gamma\left(\sum_{j=1}^{N} \lambda_j(1-\beta_j(\omega))\right), \quad \mathrm{Re}\ \omega \geq 0.\end{aligned} \tag{2.5}$$

Let us denote, for Re $\omega \geq 0$, $\delta(\omega) = \sum_{j=1}^{N} \lambda_j(1-\beta_j(\omega))$ and recursively denote

$$\begin{aligned}&\delta^{(0)}(\omega) = \omega,\\ &\delta^{(i)}(\omega) = \delta(\delta^{(i-1)}(\omega)), \qquad i = 1, 2, 3, \dots .\end{aligned} \tag{2.6}$$

Applying (2.5) iteratively we find, for every $M = 1, 2, \dots,$

$$\begin{aligned}\gamma(\omega) &= E[\mathrm{e}^{-\omega S}]E[\mathrm{e}^{-\delta(\omega)S}]\gamma(\delta^{(2)}(\omega))\\ &= \dots\\ &= \prod_{i=0}^{M} E[\mathrm{e}^{-\delta^{(i)}(\omega)S}]\gamma(\delta^{(M+1)}(\omega)).\end{aligned}$$

Now, it can be shown that $\lim_{M\to\infty} \delta^{(M+1)}(\omega) = 0$ and that the infinite product $\prod_{i=0}^{\infty} E[\mathrm{e}^{-\delta^{(i)}(\omega)S}]$ converges if $\rho < 1$ (see theorems 1 and 2 of the appendix). Hence,

$$\gamma(\omega) = E[\mathrm{e}^{-\omega C}] = \prod_{i=0}^{\infty} E[\mathrm{e}^{-\delta^{(i)}(\omega)S}] = \prod_{i=0}^{\infty} \sigma(\delta^{(i)}(\omega)), \quad \mathrm{Re}\ \omega \geq 0. \tag{2.7}$$

Similar to theorem 2.1 of Boxma and Cohen [5], one can now complete the proof of the fact that the Markov chains formed by successive queue length vectors, and by successive cycle times, are both positive recurrent if $\rho < 1$. Differentiating (2.5) once and twice yields

$$EC = ES + \left(\sum_{j=1}^{N} \lambda_j\,\beta_j\right)EC,$$

and

$$EC^2 = s^{(2)} + 2s\rho EC + \rho^2 EC^2 + \sum_{j=1}^{N} \lambda_j \beta_j^{(2)} EC,$$

from which we may derive closed form expressions for the first two moments of the cycle time:

$$EC = \frac{s}{1-\rho}, \tag{2.8}$$

$$EC^2 = \frac{1}{1-\rho^2}\left[s^{(2)} + 2s\rho EC + \sum_{j=1}^{N} \lambda_j \beta_j^{(2)} EC\right]. \tag{2.9}$$

Introducing C_P and C_R, the past and residual time, respectively, of a cycle, we can write for Re ω_P, $\omega_R \geq 0$ the LST of the joint distribution of C_P and C_R (cf. Cohen [10], p. 113):

$$\begin{aligned} E[\mathrm{e}^{-\omega_P C_P - \omega_R C_R}] &= \int_{t=0}^{\infty} \frac{t}{EC}\, \mathrm{d}Pr\{C < t\} \int_{u=0}^{t} \frac{\mathrm{d}u}{t}\, \mathrm{e}^{-\omega_P u}\, \mathrm{e}^{-\omega_R (t-u)} \\ &= \frac{1}{EC} \int_{t=0}^{\infty} \mathrm{e}^{-\omega_R t}\, \mathrm{d}Pr\{C < t\} \int_{u=0}^{t} \mathrm{e}^{-(\omega_P - \omega_R)u}\, \mathrm{d}u \\ &= \frac{1}{EC}\, \frac{1}{\omega_P - \omega_R}\, [\gamma(\omega_R) - \gamma(\omega_P)]. \end{aligned} \tag{2.10}$$

It follows in particular that the LST and the mean value of C_R and C_P are:

$$E[\mathrm{e}^{-\omega C_P}] = E[\mathrm{e}^{-\omega C_R}] = \frac{1-\gamma(\omega)}{\omega EC},$$

$$EC_P = EC_R = \frac{E[C^2]}{2EC} = \frac{1}{1-\rho^2}\left[(1-\rho)\frac{s^{(2)}}{2s} + s\rho + \frac{1}{2}\sum_{j=1}^{N} \lambda_j \beta_j^{(2)}\right], \tag{2.11}$$

in which we used (2.8) and (2.9).

2.3. A PSEUDOCONSERVATION LAW FOR THE MEAN WAITING TIMES

Let W_j be the waiting time of an arbitrary customer at Q_j. In Boxma [4], a pseudoconservation law was derived for a large variety of polling systems. This law provides an expression for the weighted sum of the mean waiting times, $\sum_{j=1}^{N} \rho_j EW_j$. It is easy to see that the analysis provided there carries over to the globally gated system and thus we can use (3.21) of Boxma [4]:

$$\sum_{j=1}^{N} \rho_j EW_j = \rho \frac{\sum_{j=1}^{N} \lambda_j \beta_j^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)}\left[\rho^2 - \sum_{j=1}^{N} \rho_j^2\right] + \sum_{j=1}^{N} EM_j^{(1)}, \tag{2.12}$$

where $EM_j^{(1)}$ ($j = 1, \ldots, N$) is the expected amount of work left in Q_j when the server leaves this queue.

Let V_j be the visit time of the server at Q_j. Noting that the mean cycle time is $EC = s/(1-\rho)$, we obtain the mean visit time as $EV_j = \rho_j s/(1-\rho)$, from which it follows that

$$EM_j^{(1)} = \rho_j [EV_1 + s_1 + EV_2 + s_2 + \ldots + EV_j] = \rho_j \sum_{i=1}^{j-1} \left(\rho_i \frac{s}{1-\rho} + s_i\right) + \rho_j^2 \frac{s}{1-\rho}, \quad j = 1, \ldots, N. \tag{2.13}$$

Substitution of (2.13) into (2.12) yields

$$\sum_{j=1}^{N} \rho_j EW_j = \rho \frac{\sum_{j=1}^{N} \lambda_j \beta_j^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{1-\rho}\rho^2 + \sum_{j=2}^{N} \rho_j \sum_{i=1}^{j-1} s_i, \tag{2.14}$$

which is the pseudoconservation law for cyclic polling with globally gated service discipline.

Comparison of (2.12) and (2.13) with the corresponding expressions for ordinary gated service (cf. (3.21) and (3.23) of Boxma [4]) shows that

$$\sum_{j=1}^{N} \rho_j E[W_j|_{\text{globally gated}}] - \sum_{j=1}^{N} \rho_j E[W_j|_{\text{gated}}] = \sum_{j=1}^{N} \rho_j \sum_{i=1}^{j-1} \left(\rho_i \frac{s}{1-\rho} + s_i\right) = \frac{s}{2(1-\rho)}\left[\rho^2 - \sum_{j=1}^{N} \rho_j^2\right] + \sum_{j=2}^{N} \rho_j \sum_{i=1}^{j-1} s_i, \tag{2.15}$$

which is also equal to the difference between the expected total amount of work in the globally gated regime and the regular gated system.

2.4. WAITING TIMES

In section 2.3, we derived an exact expression for a weighted sum of the mean waiting times at all queues, cf. (2.14). We now determine the individual waiting time distributions (their LST) and their expected values. Consider an arbitrary customer K at Q_k. His waiting time is composed of (i) a residual cycle time C_R, (ii) the service times of all customers who arrive at $Q_1, \dots, Q_{k-1}$ during the cycle in which K arrives, (iii) the switchover times of the server between Q_1 and $Q_2, \dots, Q_{k-1}$ and Q_k, and (iv) the service times of all customers who arrive at Q_k during the past part C_P of the cycle in which K arrives. Let us first determine EW_k:

$$EW_k = EC_R + \sum_{j=1}^{k-1} \lambda_j \beta_j (EC_P + EC_R) + \sum_{j=1}^{k-1} s_j + \rho_k EC_P. \tag{2.16}$$

It should be noted that the mean length of the cycle in which K arrives equals $EC_P + EC_R$ instead of EC; the cycle is an atypical cycle since it contains the arrival of K. From (2.16) and (2.11),

$$\begin{aligned} EW_k &= \left(1 + 2\sum_{j=1}^{k-1} \rho_j + \rho_k\right) EC_R + \sum_{j=1}^{k-1} s_j \\ &= \left(1 + 2\sum_{j=1}^{k-1} \rho_j + \rho_k\right) \frac{1}{\rho(1+\rho)} \left[\rho \frac{s^{(2)}}{2s} + \frac{s\rho^2}{1-\rho} + \rho \frac{\sum_{j=1}^N \lambda_j \beta_j^{(2)}}{2(1-\rho)}\right] + \sum_{j=1}^{k-1} s_j, \\ & \qquad k = 1, \dots, N. \end{aligned} \tag{2.17}$$

It readily follows that $EW_1 < EW_2 < \dots < EW_N$. In particular,

$$EW_{k+1} - EW_k = (\rho_{k+1} + \rho_k) EC_R + s_k.$$

One can easily check that the pseudoconservation law (2.14) is satisfied, observing that

$$\sum_{k=1}^{N} \rho_k \left(1 + 2\sum_{j=1}^{k-1} \rho_j + \rho_k\right) = \rho + \left[\rho^2 - \sum_{j=1}^{N} \rho_j^2\right] + \sum_{k=1}^{N} \rho_k^2 = \rho(1+\rho).$$

We now turn to the *distribution* of W_k, using the four-term composition observed in the beginning of this section. Noting that, cf. (2.2), the generating function of the number of arrivals at Q_j in an interval of length t equals $\mathrm{e}^{-\lambda_j(1-z)t}$, we can write:

$$\begin{aligned} E\left[\mathrm{e}^{-\omega W_k}\right] &= E\left[\mathrm{e}^{-\omega(S_1 + \dots + S_{k-1})}\right] \\ &\times \int_{t_P=0}^{\infty} \int_{t_R=0}^{\infty} \mathrm{d}Pr[C_P < t_P, C_R < t_R] \, \mathrm{e}^{-\omega t_R} \, \mathrm{e}^{-\sum_{j=1}^{k-1} \lambda_j (1-\beta_j(\omega))(t_P + t_R)} \, \mathrm{e}^{-\lambda_k (1-\beta_k(\omega)) t_P}, \\ & \qquad \mathrm{Re}\, \omega \geq 0. \end{aligned} \tag{2.18}$$

Using (2.10), it follows that

$$E[e^{-\omega W_k}] = \prod_{j=1}^{k-1} \sigma_j(\omega) E\left[\exp\left(-\sum_{j=1}^{k} \lambda_j(1-\beta_j(\omega))C_P - \left\{\sum_{j=1}^{k-1} \lambda_j(1-\beta_j(\omega)) + \omega\right\} C_R\right)\right]$$

$$= \prod_{j=1}^{k-1} \sigma_j(\omega) \cdot \frac{1}{EC} \cdot \frac{\gamma(\sum_{j=1}^{k} \lambda_j(1-\beta_j(\omega))) - \gamma(\sum_{j=1}^{k-1} \lambda_j(1-\beta_j(\omega)) + \omega)}{\omega - \lambda_k(1-\beta_k(\omega))},$$

$$\mathrm{Re}\,\omega \geq 0. \qquad (2.19)$$

We can rewrite (2.19) into

$$E[e^{-\omega W_k}] = E[e^{-\omega W_k}|_{M/G/1}] \cdot \prod_{j=1}^{k-1} \sigma_j(\omega) \frac{1-\rho}{s} \frac{1}{\omega(1-\rho_k)}$$

$$\times \left[\gamma\left(\sum_{j=1}^{k} \lambda_j(1-\beta_j(\omega))\right) - \gamma\left(\sum_{j=1}^{k-1} \lambda_j((1-\beta_j(\omega)) + \omega\right)\right], \qquad (2.20)$$

where $E[e^{-\omega W_k}|_{M/G/1}]$ denotes the LST of the waiting time distribution in Q_k if that queue were an arbitrary $M/G/1$ queue in isolation. Equation (2.20), therefore, demonstrates that the delay in the kth queue at the GG system decomposes into two parts, one of them being the delay incurred in the corresponding (isolated) $M/G/1$ system. Note that similar decompositions have been observed for other systems like an $M/G/1$ with vacation periods and an $M/G/1$ with set-up times (see, for example, Doshi [12] for a survey). Note also that the globally gated system with only one queue coincides with the regular gated system with one queue, for which a similar decomposition property is known to hold (see Fuhrmann and Cooper [14], pp. 1119–1120 for further references).

2.5. COMPARISON TO GATED SERVICE

As stated above, the globally gated service policy can be used as a substitute for the regular gated service. An interesting question, therefore, is how the performance of the two policies compares.

One comparison of the system has been provided in eq. (2.15), in which it was shown that the mean amount of work in the globally gated system is higher than that in the regular gated system. The remaining question is, therefore, how the individual waiting times compare.

To discuss this question, let us consider a fully symmetric system in which the parameters of all queues are identical (namely, $\lambda_i = \lambda_j$, $\beta_i = \beta_j$, $\beta_i^{(2)} = \beta_j^{(2)}$, $s_i = s_j$ and $s_i^{(2)} = s_j^{(2)}$, for all i and j). Under these conditions, when the system is operated under the regular gated policy, the mean waiting times in all queues are *identical* (and can be derived easily from the pseudoconservation law):

$$E\left[W_i \middle| \begin{smallmatrix}\text{symmetric}\\ \text{gated}\end{smallmatrix}\right] = \frac{N\lambda_i\,\beta_i^{(2)}}{2(1-\rho)} + \frac{s^{(2)}}{2s} + \frac{s\rho}{2(1-\rho)} + \frac{s\rho_i}{2(1-\rho)}. \tag{2.21}$$

In contrast, the mean delay in the symmetric globally gated system is *not* identical for all queues since this policy discriminates among the queues and gives higher priority to lower index queues. Of interest, therefore, is to compare the mean delay of Q_1 and Q_N in the globally gated system to the mean delay (same for all queues) in the regular (symmetric) gated system. The mean delay in Q_1 is given by (cf. (2.17)):

$$E\left[W_1 \middle| \begin{smallmatrix}\text{symmetric}\\ \text{globally gated}\end{smallmatrix}\right] = \frac{1+\rho_i}{1+\rho}\left[\frac{s^{(2)}}{2s} + \frac{s\rho}{1-\rho} + \frac{N\lambda_i\,\beta_i^{(2)}}{2(1-\rho)}\right]. \tag{2.22}$$

Subtracting (2.22) from (2.21), replacing each occurrence of $s^{(2)}$ by $s^{(2)} - s^2 + s^2$, and noting that $\rho = N\rho_i$, we obtain:

$$E\left[W_i \middle| \begin{smallmatrix}\text{symmetric}\\ \text{gated}\end{smallmatrix}\right] - E\left[W_1 \middle| \begin{smallmatrix}\text{symmetric}\\ \text{globally gated}\end{smallmatrix}\right] = \left[\frac{N\lambda_i\,\beta_i^{(2)}}{2(1-\rho)} + \frac{s^{(2)}-s^2}{2s}\right]\left[1 - \frac{1+\rho_i}{1+\rho}\right] + \frac{s}{2}$$

$$+\frac{(N+1)s\rho_i}{2(1-\rho)} - \frac{1+\rho_i}{1+\rho}\left[\frac{s}{2} + \frac{s\rho}{1-\rho}\right]. \tag{2.23}$$

Algebraic manipulation of (2.23) yields that the last three terms sum up to zero, and thus we have:

$$E\left[W_i \middle| \begin{smallmatrix}\text{symmetric}\\ \text{gated}\end{smallmatrix}\right] - E\left[W_1 \middle| \begin{smallmatrix}\text{symmetric}\\ \text{globally gated}\end{smallmatrix}\right] = \left[\frac{N\lambda_i\,\beta_i^{(2)}}{2(1-\rho)} + \frac{s^{(2)}-s^2}{2s}\right]\frac{\rho-\rho_i}{1+\rho}, \tag{2.24}$$

which implies that the mean delay in the symmetric regular gated system is significantly higher than that in Q_1 of the symmetric globally gated system.

The comparison between Q_N in the globally gated system and an arbitrary queue in the symmetric regular gated system is obvious: the mean delay of the former is significantly higher than that of the latter; this is directly implied by the dominance result below eq. (2.17) and by the fact that the pseudoconservation law of the globally gated system has a larger value than that of the regular gated system.

Another comparison may be made when we assume that $S = 0$ (implying that $s = 0$ and $s^{(2)}/s = 0$). In such a case, eq. (2.17) reduces to

$$E[W_k \mid \text{globally gated}] = \frac{(1 + 2\sum_{j=1}^{k-1}\rho_j + \rho_k)}{1+\rho}\;\frac{\sum_{j=1}^{N}\lambda_j\,\beta_j^{(2)}}{2(1-\rho)}$$

$$= \left[1 + \frac{\sum_{j=1}^{k-1}\rho_j - \sum_{j=k+1}^{N}\rho_j}{1+\rho}\right]\sum_{k=1}^{N}\rho_k\,E\left[W_k \middle| \begin{smallmatrix}\text{regular}\\ \text{gated}\end{smallmatrix}\right]/\rho. \tag{2.25}$$

It follows that $E[W_k|\mathrm{GG}] < \sum_{k=1}^{N}(\rho_k/\rho)E[W_k|\mathrm{G}]$ if and only if $\sum_{j=1}^{k-1}\rho_j < \sum_{j=k+1}^{N}\rho_j$. In particular, any Q_k for which $\sum_{j=1}^{k-1}\rho_j < \sum_{j=k+1}^{N}\rho_j$ has a smaller mean waiting time than the mean waiting time in a symmetric ordinary gated system. If all queues have the same traffic characteristics, this condition reduces to $k-1 < N-k$, i.e. $k < (N+1)/2$. In other words, stations positioned at the "first half circle" prefer the GG regime, while the others prefer the regular gated scheme.

3. Performance analysis of CRMA

3.1. SYSTEM DESCRIPTION

Cyclic-Reservation Multiple-Access (CRMA) is a scheme which has been proposed by Nassehi [20] for controlling the access to high-speed local area and metropolitan area networks and is based on slotted unidirectional folded bus architecture. The system consists of N stations and is a *headend* which is responsible for controlling the access to the bus. The bus begins at the headend, traverses stations 1 through N, folds back, returns via stations N through 1 and ends at the headend. The bus segment preceding the fold is called *outbound* and the segment following the fold is called *inbound*. The architecture is built to provide efficient communication among the N stations. Data packets are transmitted by a sending station on the outbound and received by the appropriate receiving station on the inbound.

The transmission in CRMA is slotted, and is based on a reservation scheme. Periodically, the headend sends a *collector* (a *reserve command* in Nassehi's nomenclature). When a collector passes via a station on the outbound, the station marks on a collector the number of slots it requires for sending the packets it currently has in its buffer (packets for which earlier reservation was not made). Thus, after passing station N, a collector completes collecting the reservations and now makes its way back on the inbound. When reaching the headend, it joins a queue in which all the collectors are served in a first-come-first-served (FCFS) order. When its turn to be served arrives, the headend generates a stream of slots (a "train") whose length is equal to the total number of slots reserved on this collector. This train is now used by the stations to transmit their data: when the train passes through a station (on the outbound), the station transmits the data units for which reservations were made on this train. The data is then received by the destination on the inbound.

3.2. A QUEUEING MODEL FOR THE PERFORMANCE ANALYSIS OF CRMA

We present a prototype queueing model which seems to reflect most of the essential characteristics of the CRMA. A single server serves N queues $Q_1, \ldots, Q_N$. Each queue is of the $M/G/1$ type with the same traffic characteristics as described in section 2.1. However, in distinction with the globally gated scheme, the server controls the system by periodically sending out "collectors" at random intervals F_i which are i.i.d. variables with distribution $F(\cdot)$, first moment f, second moment $f^{(2)}$ and LST $\varphi(\cdot)$. The collectors move from the service facility via $Q_1, Q_2, \ldots, Q_N$ and return from queue N directly to the service facility. The total travel time of each collector is the constant

$$L = \sum_{k=0}^{N} s_k ,$$

with s_i the time to move from Q_i to Q_{i+1} and s_0, s_N the times to reach Q_1 from the service facility, or to return from Q_N, respectively. A collector collects all customers that are present at each queue that it passes, and returns with a batch of customers to be served at the service facility. The batches join a queue in their order of arrival. When taken into service, the customers of a batch are served in the order in which they were collected; customers of each specific queue are served in the order of their arrival at that queue. It should be noted that collectors cannot overtake one another, since their travel times are all deterministic and the same.

3.3. WAITING TIMES

Our aim is to derive the waiting time distribution of an arbitrary customer K who arrives at, say, Q_k. Generally, the waiting time W_k of customer K is composed of five terms:

$$W_k = M_k + R_k + D_k + B_k + E_k ,$$

where

M_k = time between the instant of arrival of K and the arrival of the next collector at Q_k;

R_k = time for this collector to return to the service facility;

D_k = waiting time for the batch containing K to be taken into service;

B_k = time required to serve the customers of $Q_1, \ldots, Q_{k-1}$ that are in the same batch as K;

E_k = time required to serve the customers of Q_k that are in the same batch as K but have arrived before him.

In order to derive the distribution function of W_k, we consider all five terms in turn:

M_k: $M_k = M$ is the residual (future) lifetime of the renewal process having underlying distribution function $F(\cdot)$, common for all queues. Hence,

$$Pr\{M < t\} = \int_0^t \frac{1 - F(u)}{f} \mathrm{d}u, \tag{3.1}$$

and

$$EM = \frac{f^{(2)}}{2f}. \tag{3.2}$$

R_k: Clearly, the time for a collector to travel from Q_k to the service facility depends on the location of queue k, and is given by:

$$R_k = \sum_{j=k}^{N} s_j. \tag{3.3}$$

D_k: To determine the distribution of D_k, we may represent the headend service facility by a *GI/G/*1 queue in which a customer represents a batch of messages brought in by a collector in the CRMA system; in this *GI/G/*1 queue, the interarrival distribution is $F(\cdot)$ and the service time of an arriving customer *depends* on the length of the interval between his arrival and the previous one. Also note that the service time of a customer may be zero. Finally, note that D_k corresponds to the batch containing the tagged customer K. This implies that the arrival interval of D_k at the *GI/G/*1 queue is not a typical interval, and neither is the batch size. At this point, without any further assumptions, we write down the service time LST of an *arbitrary* customer (batch) in this *GI/G/*1 queue *arriving t time units after the previous customer* (collector); it equals

$$\prod_{i=1}^{N} \sum_{n_i=0}^{\infty} \mathrm{e}^{-\lambda_i t} \frac{(\lambda_i t)^{n_i}}{n_i!} \beta_i^{n_i}(\omega) = \exp\left(-\sum_{i=1}^{N} \lambda_i (1 - \beta_i(\omega))t\right), \quad \mathrm{Re}\,\omega \geq 0. \tag{3.4}$$

B_k: Similar to the calculation in (3.4), we can write for $\mathrm{Re}\,\omega \geq 0$:

$$E[\mathrm{e}^{-\omega B_k} | \text{collector's interval} = y] = \exp\left(-\sum_{j=1}^{k-1} \lambda_j (1 - \beta_j(\omega))y\right). \tag{3.5}$$

Again, it should be noted that the collector's interval containing K is not a typical collector's interval; it has density $(y/f)\mathrm{d}F(y)$.

E_k: Noting that customers of Q_k that are in the same batch as K but have arrived before him must have arrived in the past part of the interval between collectors arrivals at Q_k, we can write:

$$E[\mathrm{e}^{-\omega E_k}] = \int_{t=0}^{\infty} \frac{1-F(t)}{f} \sum_{i=0}^{\infty} \mathrm{e}^{-\lambda_k t} \frac{(\lambda_k t)^i}{i!} \beta_k^i(\omega)\,\mathrm{d}t$$

$$= \int_{t=0}^{\infty} \mathrm{e}^{-\lambda_k(1-\beta_k(\omega))t} \frac{1-F(t)}{f}\,\mathrm{d}t$$

$$= \frac{1}{f\lambda_k(1-\beta_k(\omega))}\left[1-\varphi\Big(\lambda_k\big(1-\beta_k(\omega)\big)\Big)\right], \quad \operatorname{Re}\omega \geq 0. \qquad (3.6)$$

Note that M and E_k are *dependent*, relating to the residual and past part of a collector's interval.

From a queueing theoretic point of view, the dependence between interarrival times and service times in a $GI/G/1$ queue is interesting; assuming negative exponential interarrival times (i.e. intervals between departures of collectors from the service facility) may give rise to explicit expressions for the distribution of D_k (this is the subject of a separate study in preparation by one of the authors), and the dependence of B_k and E_k on M can be easily handled.

From a practical (CRMA) point of view, it is more natural to assume that the collector's intervals are *fixed*, equal to f. We restrict ourselves in the remainder of this section to that case, and remark that D_k has the same distribution for all queues $Q_1, Q_2, \ldots, Q_N$, with generic random variable D. Conditioning on the arrival time of K in a collector interval, we obtain

$$E[\mathrm{e}^{-\omega W_k}] = \int_{t=0}^{f} \frac{\mathrm{d}t}{f} \cdot \mathrm{e}^{-\omega(f-t)} \mathrm{e}^{-\lambda_k(1-\beta_k(\omega))t}$$

$$\times\, \mathrm{e}^{-\omega(S_k+\ldots+S_N)} E[\mathrm{e}^{-\omega D}] \mathrm{e}^{-\sum_{j=1}^{k-1}\lambda_j(1-\beta_j(\omega))f}, \qquad (3.7)$$

where the integrand in eq. (3.7) is the product of five terms, each related to M, E_k, R_k, D and B_k, respectively, as expressed in eqs. (3.1), (3.6), (3.3) and (3.5). Thus,

$$E[\mathrm{e}^{-\omega W_k}] = \mathrm{e}^{-\omega f} \frac{1}{f[\lambda_k\big(1-\beta_k(\omega)\big)-\omega]} [1-\mathrm{e}^{-f[\lambda_k(1-\beta_k(\omega))-\omega]}]$$

$$\times\, \mathrm{e}^{-\omega(S_k+\ldots+S_N)} E[\mathrm{e}^{-\omega D}] \mathrm{e}^{-\sum_{j=1}^{k-1}\lambda_j(1-\beta_j(\omega))f}, \quad \operatorname{Re}\omega \geq 0. \qquad (3.8)$$

Here, $E[\mathrm{e}^{-\omega D}]$ has to be determined from the analysis of the $D/G/1$ queue. Observe that for this case (of deterministic collector intervals), the above-mentioned dependence between interarrival times and service times of the $GI/G/1$ (now $D/G/1$) queue does not occur.

Regarding the delay analysis and the $D/G/1$ queue, we are not aware of any exact explicit formulas for $E[e^{-\omega D}]$ and ED. Servi [21] presents expressions for these quantities in the discrete-time case; these expressions can be evaluated by solving for the zeros of a certain equation. In the continuous-time case, when the service time distribution has an LST which is a rational function, then formula (II.5.192) of Cohen [10, p. 324] yields an expression for $E[e^{-\omega D}]$. For most practical purposes, taking a distribution consisting of two exponential phases will give an accurate approximation of ED (cf. Tijms [24, pp. 301–302]).

Taking means in eq. (3.8), or adding the means of the five terms that compose W_k, it follows that

$$\begin{aligned} EW_k &= EM_k + ER_k + ED + EB_k + EE_k \\ &= \tfrac{1}{2} f + \sum_{j=k}^{N} s_j + ED + f \sum_{j=1}^{k-1} \rho_j + \tfrac{1}{2} f \rho_k \,. \end{aligned} \tag{3.9}$$

In section 4, we will investigate the problem of optimally ordering the queues and discuss some "fairness" issues.

4. Optimal visit orders and fairness

A common denominator for the two systems presented in this paper is that they provide simple expressions for the mean values of the delays incurred in the different queues. Moreover, these values differ from each other and depend on the relative location of the queues in the system. These properties can be used for prioritizing the queues and optimizing the system performance. The objective of this section is to derive rules for the optimal operation of these systems and to discuss some issues of "fairness". To this end, let us assume that Q_k is associated with a parameter c_k representing the cost of a customer being delayed a time unit in that queue. The mean waiting cost of a customer at Q_k is obviously $c_k EW_k$, and we are interested in minimizing the waiting cost of an arbitrary customer in the system: $\sum_{k=1}^{N} (\lambda_k/\lambda) c_k EW_k$, where $\lambda = \sum_{i=1}^{N} \lambda_i$. This is equivalent to the minimization of $\sum_{k=1}^{N} \lambda_k c_k EW_k$. Such a minimization will determine the (static) order in which the server visits the various queues. However, a policy in which the order of visits may *change* from one cycle to the next – following the dynamic evolution of the system – is also of interest and we discuss it in the sequel.

4.1. THE GLOBALLY GATED SYSTEM

4.1.1. Fairness

The GG scheme is a much "fairer" procedure than the regular gated service discipline in the sense of serving customers according to their order of arrival. Note that in regular gated, it is quite likely that customers arriving at Q_j are served prior to customers who arrived earlier at Q_i $(i \neq j)$. This happens because when the gate of Q_i is closed, the gates of the following queues still remain open to accept new arrivals.

4.1.2. Static optimization

From (2.17),

$$\sum_{k=1}^{N} \lambda_k c_k EW_k = EC_R \sum_{k=1}^{N} \lambda_k c_k \left(1 + \rho_k + 2 \sum_{j=1}^{k-1} \rho_j \right) + \sum_{k=1}^{N} \lambda_k c_k \sum_{j=1}^{k-1} s_j, \tag{4.1}$$

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

$$\sum_{k=1}^{N} \lambda_k c_k \sum_{j=1}^{k-1} \{2EC_R \rho_j + s_j\}.$$

Using a standard interchange argument (interchanging the order between Q_j and Q_{j+1}), one can easily show that an *index rule* for optimal ordering of the queues holds. Namely, the minimal value for $\sum_{k=1}^{N} \lambda_k c_k EW_k$ is obtained by arranging the queues in an *increasing* order of

$$u_j = \frac{2EC_R \rho_j + s_j}{\lambda_j c_j} = \frac{\rho_j E[C^2]/EC + s_j}{\lambda_j c_j}. \tag{4.2}$$

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 $c_j = \beta_j$, namely the waiting cost is proportional to the mean service time. In this case, the optimization objective becomes $\sum_{k=1}^{N} \rho_k EW_k$, which is the term for the pseudoconservation law (see also (3.14)). Optimization now is equivalent to the minimization of the mean amount of work in the system. Here, we have

$$u_j = 2EC_R + \frac{s_j}{\rho_j},$$

and thus the queues should be arranged in an *increasing* order of s_j/ρ_j.

Remark 4.1

Consider the objective function

$$\min \sum_{k=1}^{N} \rho_k \sum_{j=1}^{k-1} s_j,$$

which arises in a scheduling problem of N products on a single machine, with s_i the processing time of the ith product and ρ_i its costs per time unit of delay. The optimal processing order in this problem is also determined by arranging the products in increasing values of s_j/ρ_j.

(ii) *Negligible switching times.* Consider the case where $s_j \ll 2EC_R\, \rho_j$ for all queues. Then, the queues should be arranged in an increasing order of β_j/c_j, which resembles the well-known (and so-called) "$c\mu$" rule.

(iii) *Switch-in and switch-out times.* The model considered above includes switch-out times (namely, the server incurs a switching time S_j when switching out of Q_j). Many systems are characterized by switch-in times instead, or by switch-in times as well as switch-out times. It is easy to see that in these models a similar form to (4.1) is obtained and the structure of the index rule (4.2) remains the same, where s_j in (4.2) has to be replaced by the mean switching time associated with Q_j (switch-in time only, or the sum of switch-in and switch-out times).

(iv) *Fixed topology.* In some applications, the cyclic order of the queues is predetermined and is not left for a 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 expression achieved for the N possible cases.

4.1.3. *Dynamic optimization*

In applications in which the queue lengths can be evaluated at the cycle beginning, the GG policy can be used to *dynamically* control and optimize the system (see Browne and Yechiali [8,9]). At the beginning of each cycle the current queue lengths $X_1, \ldots, X_N$ are evaluated and the visit order for the *next* cycle is determined. Note that 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 $C(X_1, \ldots, X_N)$ is the *same* for any Hamiltonian tour of the queues. Thus, if we consider the costs incurred during a cycle by the customers present in 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:

$$\sum_{k=1}^{N} c_k \left[X_k \sum_{j=1}^{k-1} (X_j\, \beta_j + s_j) + \beta_k \sum_{i=1}^{X_k - 1} i \right] + \sum_{k=1}^{N} c_k \lambda_k E[C(X_1, \ldots, X_N)^2]/2, \quad (4.3)$$

where the first term is the contribution to total cost of the customers present at the cycle beginning, and the second term is due to the customers arriving *during* the cycle starting with $X_1, X_2, \ldots, X_N$ (see Yechiali [26]). The only term that depends on the order of visits is

$$\sum_{k=1}^{N} c_k X_k \sum_{j=1}^{k-1} (X_j \beta_j + s_j).$$

It readily follows that the optimal order for the next cycle is determined by *increasing* values of the indices

$$u_j = \frac{X_j \beta_j + s_j}{c_j X_j}, \tag{4.4}$$

which is, again, a "$c\mu$"-type rule.

4.2. THE CRMA SYSTEM

4.2.1. Fairness

It may be questioned whether, under this mechanism, a heavy load on a given station, say Q_i, causes the waiting times in other queues to be significantly higher than that in Q_i. This is not necessarily the case, as can be seen by calculating the difference in mean waiting times between a pair of queues, say Q_i and Q_k, with $i < k$. Using (3.9), it follows that

$$EW_k - EW_i = -\sum_{j=i}^{k-1} s_j + f \sum_{j=i+1}^{k-1} \rho_j + \tfrac{1}{2} f(\rho_i + \rho_k).$$

It is clear that the above difference *increases* linearly with the total load offered to the intermediate queues, Q_{i+1} to Q_{k-1}, and with the loads offered to queues Q_k and Q_i. On the other hand, it *decreases* linearly with the total intermediate switchover times between the two stations. In particular, for two adjacent queues Q_{k-1} and Q_k, $EW_k > EW_{k-1}$ if and only if $f(\rho_{k-1} + \rho_k) > 2s_{k-1}$, i.e. iff the total load flowing into these two queues during an interarrival time between successive collectors is larger than twice the switchover time between the queues.

In addition, for $\rho < 1$ and small time intervals between collectors, as well as small values of s, each collector will carry a small number of jobs and by virtue of the service discipline, this situation will be close to the "fair" policy of first-come-first-served. Thus, shorter lengths of f and s (i.e. many and quick collectors) increase the degree of "fairness" among customers.

4.2.2. Optimization

Suppose that the positions of the queues $Q_1, \ldots, Q_N$ can be rearranged, and that we wish to order the queues in such a way that

$$\sum_{k=1}^{N} \lambda_k c_k EW_k$$

is minimized. It should be noted beforehand that, for each Q_k, there are two counteracting effects. Being at the beginning of a collector's route has the advantage that a customer of Q_k is served relatively early in a batch, but it has the disadvantage that a relatively long travel time of the collector back to the service facility will be experienced.

Let us assume for the moment that S_k remains the travel time from Q_k to the next queue. This assumption is applicable if the switching time consists of a switch-out time, or in cases where s_k are all equal, $k = 1, 2, \ldots, N-1$ (to, say, $L/(2N)$). Let us also assume again that the collector's intervals are fixed, equal to f.

It follows from (3.9) that minimizing $\sum_{k=1}^{N} \lambda_k c_k EW_k$ with respect to the positions of the queues amounts to

$$\min \sum_{k=1}^{N} \lambda_k c_k \sum_{j=1}^{k-1} [f\rho_j - s_j]. \tag{4.5}$$

According to the $c\mu$ rule, the queues should be placed in *increasing* order of

$$v_j = \frac{f\rho_j - s_j}{\lambda_j c_j}. \tag{4.6}$$

It is interesting to compare this rule with the rule obtained for ordering the queues in a cyclic polling system with a globally gated service regime.

Comparing (4.6) with (4.2), it is seen that the term $s_j/(\lambda_j c_j)$ has a variable effect on the optimal position of Q_j. In the globally gated case, a high value of $s_j/(\lambda_j c_j)$ will move Q_j to a far position in the visit list, while in the CRMA, the position of Q_j is improved with a high value of $s_j/(\lambda_j c_j)$. This becomes even more evident in the case where $s_j = L/(2N)$, where the ratio between f and L plays a crucial role. For large f, the ratio $\rho_j/(\lambda_j c_j) = \beta_j/c_j$ dominates and the same ordering as for globally gated with zero or small switchover times is found; for relatively large L, when $L/(2N) > f\rho_j$, for all j, the station with the largest factor $\lambda_j c_j$ should be put in (the last) position N.

The above observations reveal both the similarity and the difference between the two systems: in GG, the server moves *towards* the customers, whereas in CRMA, the customers are brought *to* the server.

5. Conclusions and future work

In this paper, we have studied the performance characteristics of both the globally gated and the CRMA polling schemes. We provided delay analysis and derived rules for optimal operation of these systems.

The introduction of cyclic reservation schemes opens up a variety of possibilities for new service mechanisms. One such mechanism is the globally exhaustive policy which is similar to the GG procedure. In this policy, given the state vector $(X_1, \ldots, X_N)$ at the beginning of a cycle, the server renders service to queue i for a duration of X_i regular $M/G/1$-type busy periods. The analysis of this system may follow similar lines as the derivations provided in this paper, but leads to considerably more involved formulas.

Another variation – within the framework of static polling policies – is the Fair GG, which uses the GG approach but tries to achieve a higher degree of fairness among the queues. Such fairness is obtained by changing the visit order of the servers, e.g. systematically first following the order $Q_1, Q_2, \ldots Q_N$, then the order $Q_2, Q_3, \ldots, Q_N, Q_1$, etc. (as suggested by D. Zukerman).

An interesting policy is one which combines globally gated and regular gated mechanisms in one system. Such strategy can benefit from the advantages of both policies: prioritization of queues achieved by GG, and higher efficiency achieved by regular gated. In such a policy, one sets up on the cycle several gating points; each controls a gate for a subgroup of the queues. The number of customers to be served in the queues belonging to a subgroup is controlled by their "private" gate. The order of service within a subgroup can be determined by methods presented in this paper. Analysis of such hybrid schemes is presented in Altman et al. [2].

Appendix

In this appendix, we prove the two theorems that are being used in section 2.2 to show that $\rho < 1$ is a sufficient condition for ergodicity of the globally gated service discipline. The proofs follow rather similar lines to those in Boxma and Cohen [5]. For simplicity, we restrict ourselves to real $\omega \geq 0$.

THEOREM 1

For $\rho < 1$,

$$\lim_{M \to \infty} \delta^{(M)}(\omega) = 0. \tag{A.1}$$

Proof

By definition, $\delta^{(n)}(\omega) = \delta(\delta^{(n-1)}(\omega)) = \sum_{j=1}^{N} \lambda_j (1 - \beta_j(\delta^{(n-1)}(\omega)))$. Hence,

$$\delta^{(n)}(\omega) \leq \sum_{j=1}^{N} \lambda_j \beta_j \delta^{(n-1)}(\omega) = \rho \delta^{(n-1)}(\omega) \leq \rho^n \delta^{(0)}(\omega) = \rho^n \omega. \tag{A.2}$$

(A.1) now follows immediately. □

THEOREM 2

$$\prod_{i=0}^{\infty} E[\mathrm{e}^{-\delta^{(i)}(\omega)S}] \text{ converges if } \rho < 1. \tag{A.3}$$

Proof

The infinite product is said to *diverge* to zero if

$$\lim_{M \to \infty} \prod_{i=0}^{M} E[\mathrm{e}^{-\delta^{(i)}(\omega)S}] = 0.$$

The theory of infinite products (cf. Titchmarch [25, ch. 1]) shows that the infinite product in (A.3) converges iff

$$\sum_{i=0}^{\infty} \{1 - E[\mathrm{e}^{-\delta^{(i)}(\omega)S}]\} \tag{A.4}$$

converges. Now, for $\rho < 1$, using (A.2),

$$\sum_{i=0}^{\infty} \{1 - E[\mathrm{e}^{-\delta^{(i)}(\omega)S}]\} \le ES\left[\sum_{i=0}^{\infty} \delta^{(i)}(\omega)\right] \le ES\left[\omega \sum_{i=0}^{\infty} \rho^{i}\right] < \infty.$$

So, the series in (A.4), and hence also the infinite product in (A.3), converges if $\rho < 1$. □

References

[1] E. Altman, A. Khamisy and U. Yechiali, Threshold service policies in polling systems, Report, Department of Statistics, Tel Aviv University, Israel (1991).

[2] E. Altman, A. Khamisy and M. Sidi, Polling systems with synchronization constraints, Report, Faculty of Electrical Engineering, Technion, Israel (1990).

[3] J.P.C. Blanc, A numerical approach to cyclic-service queueing models, Queueing Systems 6(1990) 173–188.

[4] O.J. Boxma, Workloads and waiting times in single-server systems with multiple customer classes, Queueing Systems 5(1989)185–214.

[5] O.J. Boxma and J.W. Cohen, The *M/G/*1 queue with permanent customers, IEEE J. Sel. Areas Commun. SAC-9(1991)179–184.

[6] O.J. Boxma, H. Levy and J.A. Weststrate, Optimization of polling systems, *Proc. Performance'90*, ed. P.J.B. King, I. Mitrani and R.J. Pooley (North-Holland, 1990), pp. 349–361.

[7] O.J. Boxma, H. Levy and J.A. Weststrate, Efficient visit orders for polling systems, CWI Report BS-R9017, Amsterdam (1990).

[8] S. Browne and U. Yechiali, Dynamic routing in polling systems, in: *Teletraffic Science*, ed. M. Bonatti, *Proc. ITC-12*, 1988 (Elsevier, 1989), pp. 1455–1466.

[9] S. Browne and U. Yechiali, Dynamic priority rules for cyclic-type queues, Adv. Appl. Prob. 21(1989) 432–450.

[10] J.W. Cohen, *The Single Server Queue* (North-Holland, 1982).
[11] R.B. Cooper, Queues served in cyclic order: Waiting times, Bell. Syst. Tech. J. 49(1970)399–413.
[12] B.T. Doshi, Queueing systems with vacations – a survey, Queueing Systems 1(1986)29–66.
[13] M. Eisenberg, Queues with periodic service and changeover time, Oper. Res. 20(1972)440–451.
[14] S.W. Fuhrmann and R.B. Cooper, Stochastic decomposition in the *M/G/*1 queue with generalized vacations, Oper. Res. 33(1985)1117–1129.
[15] M. Hofri and K.W. Ross, On the optimal control of two queues with server set-up times and its analysis, SIAM J. Comp. 16(1987)399–419.
[16] A.G. Konheim and B. Meister, Waiting lines and times in a system with polling, J. ACM 21(1974)470–490.
[17] H. Levy and M. Sidi, Polling systems: Applications, modeling and optimization, IEEE Trans. Commun. COM-38(1990)1750–1760.
[18] K.K. Leung, Waiting time distribution for token-passing systems with limited-one service via discrete Fourier transforms, *Proc. IEEE Infocom '90* .
[19] I. Meilijson and U. Yechiali, On optimal right-of-way policies at a single-server station when insertion of idle times is permitted, Stochastic Processes Appl. 6(1977)25–32.
[20] M.M. Nassehi, CRMA: An access scheme for high-speed LANs and MANs, IBM Report, Zürich (1989).
[21] L.D. Servi, *D/G/*1 queues with vacations, Oper. Res. 34(1986)619–629.
[22] H. Takagi, *Analysis of Polling Systems* (MIT Press, 1986).
[23] H. Takagi, Queueing analysis of polling models: An update, in: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, 1990), pp. 267–318.
[24] H.C. Tijms, *Stochastic Modelling and Analysis: A Computational Approach* (Wiley, 1986).
[25] E.C. Titchmarsh, *The Theory of Functions*, 2nd ed. (Oxford University Press, 1968).
[26] U. Yechiali, A new derivation of the Khintchine–Pollaczek formula, in: *Operational Research '75*, ed. K.B. Haley (North-Holland, 1976), pp. 261–264.