Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

O.J. Boxma, H. Levy, J.A. Weststrate

Optimization of polling systems

Department of Operations Research, Statistics, and System Theory Report BS-R8932 December

The Centre for Mathematics and Computer Science is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Research (N.W.O.).

Optimization of Polling Systems

O.J. Boxma
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands;
Faculty of Economics, Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands

H. Levy
Computer Science Department, Tel-Aviv University
Tel-Aviv 69978, Israel

J.A. Weststrate
Faculty of Economics, Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands

This paper deals with the issue of deriving efficient operational rules for polling systems with switchover periods. Specifically, we study the following static optimization problem: Determine the server visit order (polling table) that minimizes the mean total workload. This problem is strongly related with, and in many applications coincides with, that of minimizing the overall mean customer delay in the system. A heuristic approach to the polling table problem is presented, using the exact solution of a related problem, viz.: Determine those server routing probabilities in a random polling system, that lead to the minimal mean total workload. Numerical experiments show that this heuristic approach yields excellent results.

1980 Mathematics Subject Classification: 60K25, 68M20.
Key Words & Phrases: random polling, polling table, mean workload minimization.

1. Introduction

The basic polling system is a system of multiple queues, attended to by a single server in a cyclic order. Polling systems arise naturally in the modelling of many computer, communication and production networks where several users compete for access to a common resource (a central computer, a transmission channel, a carousel in an assembly line). Takagi [23,24] and Levy and Sidi [19] mention a large variety of applications.

Such applications also give rise to several variants of the basic polling system, like:
(i) probabilistic polling: the server visits the queues according a probabilistic routing mechanism. Probabilistic polling may be used to model distributed control systems, in which the decision which station will be served next is achieved in a distributed manner, by cooperation among the stations. Cf. Kleinrock and Levy [15] who specifically mention the example of an exhaustive slotted Aloha system.
(ii) periodic polling: the server visits the queues in a fixed order specified by a polling table in which each queue occurs at least once (cf. Eisenberg [9], Baker and Rubin [1]). Some examples are provided by the token bus protocol in Local Area Networks, and by star polling at a computer with multidrop terminals (polling table [1,2,1,3,...,1,N]).

Probabilistic polling and periodic polling open interesting and useful possibilities for efficient operation and optimization, by allowing various choices of the server routing probabilities respectively the polling table. Optimization in polling systems is a subject which has so far received very little attention in queueing literature. Of the more than 450 references in Takagi's recently updated polling survey [24], almost none is concerned with optimization issues. Most polling studies do not go beyond

the comparison of performance measures under different service disciplines at the queues. One of the few exceptions is the paper of Browne and Yechiali [8]. Using Markov decision processes, they determine a semi-dynamic policy in which the server, at the beginning of a cycle, chooses a visiting order of the queues for this cycle that minimizes the mean duration of the cycle. A somewhat similar problem for the case of unit buffers is considered in [7].

The present study is devoted to optimization in polling systems. The next section contains a global discussion of several polling optimization issues. The goal there is to draw attention to this problem area and to list a number of interesting research themes. The rest of the paper studies the following optimization problem: Determine the polling table that minimizes the mean total workload in a periodic polling model. This problem is equivalent with that of minimizing $\sum \rho_i E\mathrm{W}_i$, a weighted sum of the mean waiting times $E\mathrm{W}_i$ where the weights are the traffic loads ρ_i of the queues. In the practically relevant case that all mean service times are equal, this amounts to minimizing the overall mean customer delay $\mathrm{W} := \sum \lambda_i E\mathrm{W}_i / \sum \lambda_i$, λ_i denoting the arrival rate at the i-th queue. W is perhaps the single most important performance measure in polling systems. The strong relation between total workload and overall mean customer delay adds to the importance of the mean workload as a performance measure and as an objective function for optimization. A heuristic approach to the polling table problem is presented, using the exact solution of a related problem, viz.: Determine those server routing probabilities in a probabilistic polling system that lead to the minimal mean total workload. Section 3 prepares the ground, by reviewing recent polling results which are necessary for tackling these minimization problems. Section 4 considers the case of probabilistic polling; the obtained results are being used in Section 5, which is devoted to periodic polling. Section 6 contains a summary and some plans for the future.

2. Optimization of polling systems

The ultimate goal of performance modelling and analysis is performance improvement and system optimization. Performance analysis can be applied at any stage of development, from the initial design phase to the operational phase. The range of options from which one can choose, and the optimization problems to be tackled, are mainly determined by the stage of development. For example, in designing a local area network there may first exist such channel access options as collision-detection protocols or collision-free token passing protocols. And when a token passing mechanism has been elected, the network configuration may be open for discussion: Should it be a bus, or a ring, or perhaps several interconnected rings? In the latter case, how should stations be assigned to the rings? Which static or dynamic priority rules should be implemented to give certain stations more opportunities to transmit, or longer transmission periods (thus improving some performance measure)?

Similar performance and optimization problems occur naturally in many other settings that give rise to polling models; whether it be in the design of traffic light regulation systems for signalized intersections, or in the development of a robotics system for processing several streams of parts. In the abstract setting of a single server that serves several customer classes, we now briefly discuss optimization criteria and regulation mechanisms.

Optimization criteria

In optimizing a polling system there is generally a trade-off between *efficiency* and *fairness*. From the point of view of minimizing workload in the system it may be efficient to visit heavy traffic queues frequently and for lengthy periods of time; but this may be unfair to the low traffic queues. Performance criteria which are often being studied in polling systems are the mean total workload, the server cycle time, and the individual mean waiting times or a weighted sum of them. Those weight factors may be chosen such as to represent costs, or generally to represent an appropriate balance between efficiency and fairness.

Regulation mechanisms

A natural regulation mechanism in many queueing systems is the customer access mechanism. A few polling studies have allowed finite buffer sizes, but to the best of our knowledge the paper of Browne and Yechiali [7], which considers the routing of the server in a system with unit buffers, is the only one of those in which optimization plays a key role. Here lies an important field of study.

A considerable part of the polling literature is devoted to detailed studies of service policies at the queues. The obtained results allow some comparison between different policies. Unfortunately, most sophisticated deterministic policies do not yield to an exact mathematical analysis. Recently some probabilistic service policies have been introduced, which may be used to approximate the behaviour of deterministic policies and which may be better amenable to mathematical analysis. The latter statement holds in particular for the fractional service policies suggested by Levy [17,18]. In such fractional policies, queue Q_i is assigned a parameter p_i, $0<p_i\leqslant 1$, and - loosely speaking - each of the customers present when the server visits Q_i (and possibly those arriving during their service times, etc.) has a probability p_i of receiving service in this visit period. The choice of the p_i gives rise to interesting optimization problems, which will be discussed in another paper. Another probabilistic policy is the Bernoulli service policy, in which a limit to the number of customers served in a service period is set using the Bernoulli distribution. This policy seems to affect the performance more than the fractional policies, but it is less amenable to mathematical analysis (cf. Servi [22]).

Another basic regulation mechanism in polling systems is the server routing between queues. Cyclic routing is more and more becoming a naive strategy, dating from the days in which not enough computing power was available to implement something more sophisticated. A fixed routing scheme may still be attractive, but in such a scheme it should be possible to visit some stations more frequently than others. Nowadays many designers try to build a good *polling table,* but there are no clear-cut rules on how to form the table. The main goal of the present paper is to provide and test such rules.

It obviously makes sense to combine consideration of service policies and server routing strategies. For example, instead of including a queue several times in the polling table and serving one customer at each visit, it may be better to visit it only once or twice and provide exhaustive service. In the present study we do not touch upon this issue, but the results that we obtain can be used for a further investigation in this direction.

One may go a step beyond fixed, static, routing schemes. In dynamic routing the server visit order is changing dynamically, being determined by the system state during its operation. For example, it may be natural to observe the contents of the queues and to serve next the most heavily loaded queue. The advantage of dynamic server routing is that it is very sensitive to the actual system state and can thus be used to improve its performance. The disadvantages are that it requires information gathering during operation and that it is generally very hard to analyze. For systems without switchover times in which the queue to be served next is chosen after each service completion on the basis of complete information about the buffer contents, and with the goal of minimizing the weighted sum $\sum c_i E\mathrm{W}_i$, a simple $c\mu$ rule holds (see [19] for some references). But when switchover times are positive, results are very scarce. Hofri and Ross [11] show for a two-queue model with switchover times that the optimal switchover rule is of a threshold type, i.e., there exist thresholds that determine when the server switches from one queue to the other. As mentioned in Section 1, Browne and Yechiali [7,8] study semi-dynamic server routing, in which the server visits all the queues exactly once in a cycle, but chooses a new cycle order at the end of each cycle.

Above we have indicated some global optimization issues in polling systems. By now a vast body of knowledge concerning polling systems is available. We believe it is time to develop optimization techniques to improve their performance, borrowing methods and insight from such fields as non-linear programming, Markov decision theory and control theory.

3. Workloads and waiting times in polling systems - a brief review

Model description

A single server, S, serves N infinite-capacity queues (stations) $Q_1, \ldots, Q_N$, switching from queue to

queue. Customers arrive at all queues according to independent Poisson processes. The arrival intensity at Q_i is λ_i, $i=1,...,N$. Customers arriving at Q_i are called class-i customers. The service times of class-i customers are independent, identically distributed stochastic variables. Their distribution $B_i(.)$ has first moment β_i and second moment $\beta_i^{(2)}$, $i=1,...,N$. The offered traffic load, ρ_i, at Q_i is defined as $\rho_i := \lambda_i\beta_i$, $i=1,...,N$, and the total offered load, ρ, as $\rho := \sum_{i=1}^{N} \rho_i$. The switchover times of S between the various queues are independent stochastic variables. We specify them further when the need arises.

The scheduling discipline is the procedure for deciding which customer(s) should be in service at any time. In the polling models under consideration, the scheduling discipline can be decomposed into three components: (i) the server routing between queues; (ii) the switchover times between queues; (iii) the service policy at each queue. With regard to those service policies we restrict ourselves here mainly to exhaustive service (S empties each queue that he visits) and gated service (S serves exactly those customers in the queue who were present upon his arrival at that queue).
A case can be made for not including switchover times into the scheduling discipline. We have chosen to include them, because of the crucial influence that their presence has on the concepts to be discussed in the next paragraph.

Workloads - work conservation and work decomposition
One of the most fundamental properties that single-server multi-class service systems may possess is the property of *work conservation.* The scheduling discipline (server behaviour) is work conserving if (i) S serves at constant rate, (ii) he serves if and only if at least one customer is present, and (iii) his behaviour does not affect the amount of service given to a customer, or the arrival time of any customer. In this case a sample path consideration shows that the amount of work in the system is the same, whatever server behaviour with the above-mentioned properties occurs.
In a polling system with switchover times of the server between classes, the principle of work conservation is *violated* in the sense that the service process is interrupted although work is still present. Recently it has been shown that, under certain conditions, a simple extension of the work conservation principle holds, viz. a *work decomposition principle.* Before discussing this extension we introduce the following assumptions, which hold in the rest of the paper.

ASSUMPTIONS 3.1
1. All involved stochastic processes possess an equilibrium distribution.
2. All arrival, service and switchover processes are independent stochastic processes.
3. Apart from the switchovers, the server behaviour is work conserving.

For the case that S visits the classes in a fixed cyclic order, it has been proven in [3] under Assumptions 3.1 that the steady-state amount of work, $\hat{\mathbb{V}}$, in the system with switchover times is distributed as the sum of two independent quantities, viz. (i) the steady-state amount of work, $\mathbb{V}$, in the corresponding system without switchover times and (ii) the steady-state amount of work, $\mathbb{Y}$, in the system at an arbitrary switchover epoch:

$$\hat{\mathbb{V}} \stackrel{D}{=} \mathbb{V} + \mathbb{Y}. \tag{3.1}$$

Here $\stackrel{D}{=}$ denotes equality in distribution, and 'the corresponding system without switchover times' indicates a single-server multi-class system with exactly the same arrival and service demand process as the system under consideration, but without switchover times (hence work conserving).

The work decomposition result was subsequently proven for more general server visit orders and a more general service interruption mechanism [2]. The work decomposition formula (3.1) is in particular valid for two special polling schemes that generalize cyclic polling and that play a central role in the remainder of this study: periodic polling [4] and probabilistic (Markovian) polling [6].

In (3.1) $\mathbf{V}$ is completely independent of the scheduling discipline, but $\mathbf{Y}$, and hence also $\hat{\mathbf{V}}$, *does* depend on it. Naturally, $\mathbf{Y}$ and $\mathbf{V}$ should decrease

(i) with decreasing switchover times;
(ii) with increasingly 'efficient' visit order;
(iii) with increasing exhaustiveness of the service at the queues.

We return to the first two properties later on in this section. The third property has been formalized in [20]. In that paper a general framework is presented for the comparison of different service policies in polling systems. A sample path comparison is made which allows the evaluation of the policies based on the total amount of work $V(t)$ in the system at any time t. This comparison concerns policies operating with the same realizations of the arrival, service and switchover processes and of the polling order. These processes and the polling order are allowed to be quite general. The only restrictions on the server behaviour outside switchover periods are that it should be work conserving and that the server does not wait idling in an empty queue. The sample path comparison leads to the following results:

(i) The workload at any time t under the exhaustive service policy is less than or equal to the workload at t under any other arbitrary policy:

$$V_{exhaustive}(t) \leqslant V_{policy}(t).$$

(ii) With a similar notation,

$$V_{k-limited}(t) \leqslant V_{m-limited}(t) \text{ for } k \geqslant m;$$

(under the c-limited policy, S serves at most c customers before leaving the queue).

Similar comparisons are made for *stochastic* policies like the Bernoulli and binomial-gated policies: an ordering is proven w.r.t. the parameter of the related (Bernoulli, binomial) probability distribution.

Waiting times - conservation laws and pseudoconservation laws

Consider a single-server multi-class system for which Assumptions 3.1 hold. First we restrict ourselves to the case of zero switchover times. Introduce $\mathbf{W}_n$, the waiting time (excluding service time) of a class-n customer. Under the assumption that the scheduling discipline is non-preemptive, and that only information about the current state and the past of the queueing process is used in making scheduling decisions, it can be shown that (cf. Kleinrock [13,14]):

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{n=1}^{N} \lambda_n \beta_n^{(2)}}{2(1-\rho)}. \tag{3.2}$$

Kleinrock called (3.2) a *conservation law* to indicate that a change in the scheduling discipline (under the above restrictions) does not lead to a change in $\sum \rho_n E\mathbf{W}_n$.

Under the same conditions as above, one obtains [2] in the case of non-zero switchover times, using (3.1):

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{n=1}^{N} \lambda_n \beta_n^{(2)}}{2(1-\rho)} + E\mathbf{Y}. \tag{3.3}$$

This has been coined a *pseudoconservation law:* a change in the visit order or service policy at a queue generally *does* lead to a change in $E\mathbf{Y}$, and hence in the lefthand side of (3.3). We shall specify $E\mathbf{Y}$ for the following server visit orders: (I) cyclic polling, (II) periodic polling, and (III) Markovian

polling. In the sequel, the groups of queues that are being served under the exhaustive (gated) service policy are denoted by e (g).

I. CYCLIC POLLING

Formula (3.3) now reduces to the following pseudoconservation law [3]:

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{n=1}^{N} \lambda_n \beta_n^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)}[\rho^2 - \sum_{n=1}^{N} \rho_n^2] + \frac{s}{1-\rho} \sum_{n \in g} \rho_n^2. \tag{3.4}$$

Here s and $s^{(2)}$ denote the mean and second moment of the sum of the switchover times in one cycle. It should be noted that $E\mathbf{Y}$ appears to be roughly linearly dependent on the mean total switchover time s; $E\mathbf{Y}$ and $E\hat{\mathbf{V}}$ appear to increase roughly linearly with increasing switchover times. It is also noteworthy that the order of the queues in the cycle does not influence the mean workload of the system or the weighted sum of mean waiting times in (3.4) as long as this order does not affect s.

II. PERIODIC POLLING

First some additional notation. The order in which S visits the queues is specified in a polling table $T = \{T(m),\ m = 1,...,M\}$. The i-th entry $T(i)$ is the index of the i-th queue polled in the cycle. This queue is referred to as the i-th 'pseudostation'. For example, $T = \{1,2,1,3\}$ denotes a cycle in which Q_1, Q_2, Q_1, Q_3 are consecutively visited. The first and third pseudostation both refer to Q_1. s_m and $s_m^{(2)}$ indicate the mean and second moment of the switchover time between the m-th and $m+1$-st pseudostations; s denotes the mean of the total switchover time in one cycle. The mean of the visit time of pseudostation m is denoted by $E\mathbf{VI}_m$. Generally there is no simple expression available for these mean visit times, but they can be obtained by solving a simple set of linear equations, cf. [1,4]. Finally we introduce the $M \times M$ (0,1) matrix $Z = (z_{ij})$, where $z_{ij} = 0$ unless none of the table entries $T(i+1), \ldots, T(j)$ equals $T(i)$, in which case $z_{ij} = 1$. The following pseudoconservation law has been proven in [4] ($\tilde{g}$ denotes the group of gated pseudostations):

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \sum_{m=1}^{M} \frac{s_m^{(2)}}{2s} + \tag{3.5}$$

$$\sum_{k=1}^{M} \rho_{T(k)} \sum_{m \neq k} \frac{s_m}{s} z_{km} \sum_{j=k}^{m-1} (s_j + E\mathbf{VI}_{j+1}) + \sum_{j \in \tilde{g}} \rho_{T(j)} E\mathbf{VI}_j \sum_{m=1}^{M} \frac{s_m}{s} z_{jm} + \sum_{m \in \tilde{g}} \rho_{T(m)} \frac{s_m}{s} E\mathbf{VI}_m.$$

The sum $\sum_{j=k}^{m-1}$, with $1 \leqslant k,m \leqslant M$, should be interpreted as a cyclic sum.

III. MARKOVIAN POLLING

Again, we first need some notation. S is assumed to move between the N queues according to an irreducible, positive recurrent discrete-time parameter Markov chain $\{\mathbf{d}_n,\ n = 0,1,...\}$ with stationary transition probabilities $p_{ij} = Pr\{\mathbf{d}_{n+1} = j \mid \mathbf{d}_n = i\}$, $i,j = 1,...,N$, $n = 0,1,...$. The limiting and stationary distribution of this Markov chain is denoted by $q_i = \lim_{n \to \infty} Pr\{\mathbf{d}_n = i\}$, $i = 1,...,N$. The switchover times of S between Q_i and Q_j are i.i.d. stochastic variables with mean s_{ij} and second moment $s_{ij}^{(2)}$.
An important quantity in this model is $\mathbf{T}_{ki}$, the time between a departure of S from Q_i and the last previous departure from Q_k, $k,i = 1,...,N$. Generally, determination of all $E\mathbf{T}_{ki}$ requires the solution of N sets of N linear equations. In [6] the following pseudoconservation law has been proven:

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \frac{\sigma}{1-\rho} \sum_{k \in g} \frac{\rho_k^2}{q_k} + \frac{\rho}{2\sigma} \sum_{i=1}^{N} q_i \sum_{j=1}^{N} p_{ij} s_{ij}^{(2)} + \frac{1}{\sigma} \sum_{i=1}^{N} q_i \sum_{j=1}^{N} p_{ij} s_{ij} \sum_{k \neq i} \rho_k E\mathbf{T}_{ki}, \tag{3.6}$$

with

$$\sigma := \sum_{i=1}^{N} q_i \sum_{j=1}^{N} p_{ij} s_{ij}.$$

Kleinrock & Levy [15] restrict themselves to the case that $p_{ij} = p_j$ (this is referred to as *random polling*) and $s_{ij} = s_i$, $s_{ij}^{(2)} = s_i^{(2)}$ for all $i,j \in \{1,...,N\}$. In this case $q_k = p_k$, $k = 1,...,N$, $E\mathbf{T}_{ki} = (\sigma/(1-\rho))[(\rho_i/q_i) - (\rho_k/q_k) + (1/q_k)]$, $k,i = 1,...,N$, and (3.6) reduces to:

$$\sum_{n=1}^{N} \rho_n E\mathbf{W}_n = \rho \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1-\rho)} - \frac{\sigma}{1-\rho} \sum_{k \in e} \frac{\rho_k^2}{p_k} + \frac{\sigma}{1-\rho} \sum_{k=1}^{N} \frac{\rho_k}{p_k} - \sum_{i=1}^{N} \rho_i s_i + \frac{\rho}{2\sigma} \sum_{i=1}^{N} p_i s_i^{(2)}, \tag{3.7}$$

with

$$\sigma = \sum_{i=1}^{N} p_i s_i.$$

In Sections 4 and 5 we shall use some of the above conservation laws to attempt minimization of the mean workload in a polling system with either Markovian polling or periodic polling, when the server transition probabilities, respectively the table, can be freely chosen.

REMARK 3.1
The principle of work conservation implies that, in the case of zero switchover times, *any* server visit order leads to the same mean workload. But for positive switchover times and given service policies at the queues, the mean total workload will be relatively small when the server visit order is such that mean visits are relatively long w.r.t. switchover times. In the next two sections we shall investigate this in detail for random polling and periodic polling.

REMARK 3.2
From the equations determining the mean visit times $E\mathbf{VI}_m$ in (3.5), respectively the mean interdeparture times $E\mathbf{T}_{ki}$ in (3.6) (cf. [4] respectively [6]), it can be seen that these quantities depend on λ_j and β_j only through their product ρ_j. Hence only the first term in the righthand sides of (3.5) and (3.6) depends on individual arrival rates and service time moments; and this first term does not depend on the choice of the polling table respectively the server transition probabilities. The implication is that for the optimal choice of the table or the transition probabilities, only traffic loads matter and not individual arrival rates and service time moments.

4. OPTIMIZATION OF RANDOM POLLING SYSTEMS

Consider the Markovian polling system described at the end of Section 3. In the present section we are interested in the following problem. Suppose that for given arrival, service and switchover processes and service disciplines at the queues, the system designer still has the freedom to choose the server transition probabilities p_{ij}, $i,j = 1,...,N$. He wants to choose them such that the mean steady-

state amount of work in the system is minimized. For simplicity a restriction is made to the case of random polling: $p_{ij} \equiv p_j$, with all switchover times i.i.d. with first moment σ and second moment $\sigma^{(2)}$.

Intuitively one expects that queues with heavy traffic should be visited more frequently than low traffic queues. But how much more frequently? In purely cyclic polling it is well known that the ratios of mean visit times of the queues are equal to the ratios of the offered traffic loads. Should visit frequencies in random polling obey the same rule in order to minimize mean workload? Does the choice of service discipline matter?

In the relatively simple case under consideration, it will turn out that these questions can be easily answered using the pseudoconservation law (3.7). Furthermore, the expressions for optimal visit frequencies appear to be extremely appealing, being simple, robust and elegant.

It follows from (3.7) and the fact that, cf. [6],

$$E\hat{V} = \sum_{n=1}^{N} \rho_n E W_n + \sum_{n=1}^{N} \frac{\beta_n^{(2)}}{2\beta_n} \rho_n, \tag{4.1}$$

that minimization of $E\hat{V}$ w.r.t. $p_1, \ldots, p_N$, under the conditions $p_1 + \cdots + p_N = 1$, $p_1 \geqslant 0, \cdots, p_N \geqslant 0$, amounts to the following problem.

$$Min \left[\frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1-\rho)} - \frac{\sigma}{1-\rho} \sum_{k \in e} \frac{\rho_k^2}{p_k} + \frac{\sigma}{1-\rho} \sum_{k=1}^{N} \frac{\rho_k}{p_k} - \rho\sigma + \frac{\rho}{2\sigma}\sigma^{(2)} \right] \tag{4.2}$$

s.t.

$$p_1 + \cdots + p_N = 1, \quad p_1 \geqslant 0, \quad \cdots, \quad p_N \geqslant 0.$$

This is a classical non-linear optimization problem with linear constraints. Introducing the Lagrange multiplier L, and omitting all terms in (4.2) that do not involve the probabilities p_i, we want to minimize the unconstrained Lagrangian function

$$F := -\frac{\sigma}{1-\rho} \sum_{k \in e} \frac{\rho_k^2}{p_k} + \frac{\sigma}{1-\rho} \sum_{k=1}^{N} \frac{\rho_k}{p_k} + L\left(\sum_{k=1}^{N} p_k - 1\right) \tag{4.3}$$

in the non-negative 2^N-tant. The Kuhn-Tucker points of this expression are obtained by putting $\frac{\partial F}{\partial p_k} = 0$, $k = 1, \ldots, N$ and $\frac{\partial F}{\partial L} = 0$, yielding

(i) if Q_k has exhaustive service:

$$-\frac{\sigma}{1-\rho} \frac{\rho_k - \rho_k^2}{p_k^2} + L = 0;$$

(i) if Q_k has gated service:

$$-\frac{\sigma}{1-\rho} \frac{\rho_k}{p_k^2} + L = 0.$$

The convexity of F in $(p_1, \ldots, p_N)$ readily implies that the admissible stationary point yields the minimum of $E\hat{V}$:

If Q_k has exhaustive service:

$$p_k = \frac{\sqrt{\rho_k(1-\rho_k)}}{\sum_{j \in e} \sqrt{\rho_j(1-\rho_j)} + \sum_{j \in g} \sqrt{\rho_j}}; \tag{4.4}$$

and if Q_k has gated service:

$$p_k = \frac{\sqrt{\rho_k}}{\sum\limits_{j\in e} \sqrt{\rho_j(1-\rho_j)} + \sum\limits_{j\in g} \sqrt{\rho_j}}. \tag{4.5}$$

As announced, these optimal server routing probabilities are remarkably simple. Allocation is according to a square root rule, and only the offered traffic loads play a role. In the exhaustive case the influence of this load is quite small. It should be noted that the visit frequency for a queue Q_k with exhaustive service is *decreasing* in ρ_k for $\rho_k > 0.5$. Still, it is easily seen that among any two queues with exhaustive service the one with higher load has a higher visit frequency.

REMARK 4.1
When S meets n customers in Q_k, his mean visit period equals $n\beta_k$ in the case of gated service, and $n\beta_k/(1-\rho_k)$ in the case of exhaustive service. This gives some feeling as to why a queue with exhaustive service should receive fewer visits than a queue with gated service and the same traffic load, and why relatively few visits should be made to a queue with exhaustive service in heavy traffic.

REMARK 4.2
We have also performed the above minimization for the cases of binomial-gated and binomial-exhaustive service. In binomial-gated service, when S finds m customers present at Q_k he serves n out of those m with probability $\binom{m}{n}\alpha_k^n(1-\alpha_k)^{m-n}$ $(0<\alpha_k\leqslant 1)$ and then leaves the queue; in the same situation under binomial-exhaustive service, S selects n out of those m customers with probability $\binom{m}{n}\gamma_k^n(1-\gamma_k)^{m-n}$ $(0<\gamma_k\leqslant 1)$ and serves those customers, and the ones arriving during their service, etc. Denoting the binomial-exhaustive (binomial-gated) queues by *be (bg)*, and now including e in *be* $(\gamma_k=1)$ and g in *bg* $(\alpha_k=1)$, we find if Q_k has binomial-exhaustive service:

$$p_k = \frac{\sqrt{\rho_k(1-\rho_k)/\gamma_k}}{\sum\limits_{j\in be} \sqrt{\rho_j(1-\rho_j)/\gamma_j} + \sum\limits_{j\in bg} \sqrt{\rho_j/\alpha_j}}, \tag{4.6}$$

and if Q_k has binomial-gated service:

$$p_k = \frac{\sqrt{\rho_k/\alpha_k}}{\sum\limits_{j\in be} \sqrt{\rho_j(1-\rho_j)/\gamma_j} + \sum\limits_{j\in bg} \sqrt{\rho_j/\alpha_j}}. \tag{4.7}$$

REMARK 4.3
It follows from (4.1) that, in the practically relevant case that all mean service times are equal, minimizing $E\hat{V}$ amounts to minimizing $\sum\lambda_i EW_i \;/\; \sum\lambda_i$, the overall mean waiting time.

5. Optimization of periodic polling systems

Consider the polling system with a polling table, as described in Section 3. In the present section we are interested in the following problem. Suppose that for given arrival, service and switchover processes and service disciplines at the queues, the system designer still has the freedom to choose the polling table. He wants to choose it such that the mean steady-state amount of work in the system is minimized. For simplicity a restriction is made to the case that all switchover times are i.i.d. with first moment σ and second moment $\sigma^{(2)}$.

As in the case of the previous section, the mean workload $E\hat{V}$ is linearly related to $\sum \rho_n EW_n$ according to (4.1). Hence, for a polling table with only exhaustive and gated service, minimization of $E\hat{V}$ over all possible polling tables amounts to minimization of the expression in the righthand side of the pseudoconservation law (3.5) over all such tables. If (an upper bound on) the size of the table, M, is given, then this requires the solution of an integer programming problem (S. Browne [private communication]). Below we will be concerned with the case in which there is no restriction on the table size. The number of possible tables is now unlimited, and it is a priori not clear whether a given 'good' table cannot be improved upon by taking a much larger table with a very similar structure (for example: replace a 60-entry table with 59 $\times Q_1$ followed by once Q_2, by a 6001 entry table composed of 99 subsequent such 60-entry patterns followed by 60 $\times Q_1$ and once Q_2).

In this section we present an approximate approach to the problem of choosing an optimal polling table. The approach consists of three steps:

Step 1. Determine 'good' ratios of occurrence of all queues in the table.
Step 2. Based on these ratios, determine a 'good' table size M and the numbers of occurrence of each queue in the table.
Step 3. Given this M and these numbers of occurrence, determine a 'good' ordering of the queues.

Below we discuss each of these steps in some detail. Subsequently we present numerical results to illustrate the accuracy of the procedure. In [5] more numerical experiments will be reported upon, along with an analysis of some generalizations.

Step 1. Determination of occurrence ratios

Consider a random polling system with the same arrival, service and switchover distributions and the same service policies as in the periodic polling system. In the previous section it has been shown that the square root probability assignments (4.4) (for exhaustive service) and (4.5) (for gated service) minimize the mean workload in the random polling system. We propose to choose the occurrence frequencies $f_1, \ldots, f_N$ of the various queues in the polling table according to exactly the same square root assignments. Obviously there is no guarantee that this yields an optimal ratio. On the other hand, since the two systems possess the same properties apart from the fact that in the first one the queues are chosen in random order and in the second one in periodic order, it seems natural that approximately the same visit frequencies should optimize both.

Remark 5.1

Recently we have found a quite different argument that leads to exactly the same square root assignment in polling tables for exhaustive service, and to a slightly different square root assignment for gated service ($\sqrt{\rho_i(1+\rho_i)}$ instead of $\sqrt{\rho_i}$, which in most cases yields only minor differences). Details will be presented in [5].

Step 2. Determination of the table size

Let $f_1,\ldots,f_N$ (with $\sum f_i = 1$) be the occurrence frequencies obtained in step 1. We want to choose a table size M such that $Mf_1, \ldots, Mf_N$ either are integers or are within a predetermined small positive distance ϵ from an integer (such that the sum of these integers equals M). The resulting integers $n_1, \ldots, n_N$ will be the numbers of occurrence of the N stations. ϵ determines how accurately we wish

to approximate the occurrence frequencies. In a different context, this procedure has been proposed by Panwar et al. [21]. As step 2 seems to be the least crucial one in the present heuristic, we have in most numerical experiments restricted ourselves to examples where all frequencies are such rational numbers that all Mf_i are integers for a reasonably small M. Below we shall report on such examples. Our limited experience with the procedure of Panwar et al. [21] suggests that it is not necessary to take ϵ very small; we have several examples where $\epsilon=0.25$ (leading to a small table) yields a better result than a much smaller ϵ (that leads to a larger table). We might add that for practical purposes the table can be quite large; a table of several hundred entries should not pose any difficulty in most systems.

Step 3. Determination of the order within the table

In the previous steps we have determined the table size M and the numbers of occurrence $n_1, \ldots, n_N$ of the queues in this table, with $n_i \approx Mf_i$. We would like to find a table order in which, for each i, the numbers of visits to other queues between consecutive visits to Q_i are (nearly) equal. The following example demonstrates that exact equality cannot always be reached. Let $M=6$, $n_1=1$, $n_2=2$ and $n_3=3$. There is no order in which Q_2 is visited each third time and in which Q_3 is visited after each visit to any other queue.

This example was taken from Hofri & Rosberg [10]. They consider a conflict-free distributed protocol for access of N transmission stations to a common channel. They use a weighted Time Division Multiplexing (TDM) protocol; the weight factors refer to the frequencies with which time slots are assigned to the stations. TDM systems are very similar to polling systems. Two main differences, which make TDM better amenable to an exact analysis, are: In TDM each station is visited by the server for a fixed time slot, regardless of whether there are messages present; and TDM does not require switchover times between stations. Hofri & Rosberg investigate two weighted TDM policies for assigning the slots to the stations, for given weight factors $f_1, \ldots, f_N$. One is a 'random' control policy in which each slot is with probability f_i assigned to the i-th station (note the similarity with random polling, where a visit period is assigned instead of a time slot). The other one is a deterministic policy, which appears to be much better than the random policy: the 'Golden Ratio policy'. We describe this policy in detail, because we propose to use it also for determining a 'good' polling order.

Let $\phi^{-1} := \frac{1}{2}(\sqrt{5}-1)=0.618034\ldots$. ($\phi^{-1}$ is also known as the Golden Ratio; it is related to the Fibonacci numbers $F_1, F_2, \ldots$ via $F_k = [\phi^k-(1-\phi)^k]/\sqrt{5}$.) Put the M numbers $\phi^{-1} mod 1, 2\phi^{-1} mod 1, \ldots, M\phi^{-1} mod 1$ in increasing order (this corresponds to placing them on a circle of unit circumference). Let the j-th smallest number correspond to the j-th position in the table. Assign $\phi^{-1} mod 1, 2\phi^{-1} mod 1, \ldots, n_1\phi^{-1} mod 1$ to Q_1, $(n_1+1)\phi^{-1} mod 1, \ldots, (n_1+n_2)\phi^{-1} mod 1$ to Q_2, etc. The table is thus determined.

Hofri & Rosberg [10], and in particular also Itai & Rosberg [12], discuss a number of properties of the thus obtained assignment. Theorem 5.1 of Itai & Rosberg [12] states that the circle of unit circumference is divided into intervals of at most three different lengths (two if M is a Fibonacci number). As a corollary, they conclude that for each station i, too, there are at most three different interval lengths between successive placements (two if n_i is a Fibonacci number). Consequently, distances between consecutive occurrences of station i in the polling table are also quite evenly spaced. This provides the motivation for using the Golden Ratio (GR) policy in our periodic polling problem.

For the same reason, GR has been applied to several other problems where more or less equidistant spacings of several kinds of items have to be accomplished. See Knuth [16] for an extensive discussion of its properties, and its application to open address hashing (how to distribute keys uniformly over a hashing table); see Itai & Rosberg [12], Hofri & Rosberg [10] and Panwar et al. [21] for performance studies of multi-access protocols which use the Golden Ratio policy, and for a discussion of properties of this policy.

TABLE I

A 2-queue case : comparison of the 'optimal' polling table and
the Golden Ratio polling table.
The service strategy at each queue is exhaustive.
$\rho_1 = 0.878$, $\rho_2 = 0.05$.

	n_2	1		2		3		4	
n_1		table	$\hat{EV}$	table	$\hat{EV}$	table	$\hat{EV}$	table	$\hat{EV}$
1	opt	12	**15.036**	122	16.061	1222	17.087	12222	18.112
	gr	21	15.036	212	16.061	2212	17.087	22212	18.112
2	opt	211	15.405	2121	**15.036**	12122	15.569	122122	16.061
	gr	112	15.405	1212	15.036	21212	15.570	212122	16.131
3	opt	2111	15.816	21211	15.221	121212	**15.036**	2212121	15.403
	gr	1211	15.816	21211	15.221	212121	15.036	2122121	15.403
4	opt	21111	16.245	211211	15.405	1121212	15.162	21212121	**15.036**
	gr	21111	16.245	211121	15.436	2121121	15.162	21211212	15.505
5	opt	211111	16.683	2112111	15.614	11211212	15.284	112121212	15.132
	gr	111121	16.683	1121121	15.614	11211212	15.284	112121212	15.132
6	opt	2111111	17.125	21112111	15.816	112112112	15.405	1121211212	15.221
	gr	1121111	17.125	11211112	15.839	112121112	15.430	1212121112	15.252

TABLE II

A 3-queue case : comparison of the 'optimal' polling table and
the Golden Ratio polling table.
The service strategy at each queue is gated.
$\rho_1 = 0.54$, $\rho_2 = 0.24$, $\rho_3 = 0.06$.

	n_2	1		2		3		4	
n_1		table	$\hat{EV}$	table	$\hat{EV}$	table	$\hat{EV}$	table	$\hat{EV}$
1	opt	123	16.433	2123	18.579	21223	21.464	221223	24.425
	gr	213	16.433	2312	18.859	32212	21.464	222132	24.943
2	opt	1213	15.034	12123	15.099	212123	16.317	2122123	17.686
	gr	1312	15.034	31212	15.099	212132	16.524	2132122	18.071
3	opt	11213	15.497	121213	**14.589**	1212123	14.943	21212123	15.842
	gr	31211	15.497	212131	14.589	2132121	14.943	21221213	15.953
4	opt	112113	16.321	1211213	14.590	12121213	14.709	212121213	15.087
	gr	211131	16.448	2131121	14.903	21211213	14.807	212131212	15.543
5	opt	1121113	17.399	12112113	15.009	121211213	14.748	1212121213	14.981
	gr	1131121	17.399	11211213	15.009	112131212	14.748	1312121212	14.981
6	opt	11121113	18.528	121112113	15.476	1211211213	14.889	12121121213	15.036
	gr	11211113	18.607	112131112	16.656	1312121112	15.124	12121211312	15.279

Numerical results

We present 4 tables and one figure to give an indication of the accuracy of the procedure outlined in steps 1,2 and 3 above. In the limited space of this paper, it is impossible to cover the wide range of possible arrival rates, service and switching characteristics and service disciplines. We have restricted ourselves to constant switchover times, equal to one, and to negative exponential service time distributions with unit mean at all queues (Remark 3.2 implies that it is no restriction to choose all mean service times equal).

Discussion of Tables I and II

Tables I and II are respectively concerned with an extensive analysis of one 2-queue and one 3-queue example. In the 2-queue example of Table I, both queues receive exhaustive service. Q_1 has heavy traffic, whereas Q_2 has low traffic. Step 1 recommends an occurrence ratio of 3:2. In the table we have investigated all 24 ratios $n_1:n_2$ ranging from 1:1 to 6:4. For each of these ratios we have selected n_1 and n_2 to be the numbers of visits paid to the queues, getting $M=n_1+n_2$, and we have found the best ordering consisting of n_1,n_2 visits by calculating $E\hat{V}$ for all those orderings. Under the heading 'opt' the lowest such $E\hat{V}$ and the corresponding table have been displayed. Under the heading 'gr', we display the result of applying GR to each (n_1,n_2) combination. gr coincides with opt in 17 out of the 24 cases; in the other 7 cases its largest workload difference is less than 4%. This supports the usefulness of the GR policy (step 3). Step 1 (taking the ratio 3:2) did not lead to the best ratio; 1:1 (or 2:2, or 3:3, etc.) yields a mean workload which is 1.3% lower. On the other hand, taking a ratio equal to $\rho_1:\rho_2$ would have led to a GR result that is approximately 47% worse (for $n_1=17$, $n_2=1$; slightly larger errors are found for $n_1=18$, $n_2=1$ and $n_1=35$, $n_2=2$).

In Table I it appears to be slightly better to alternate visits to Q_1 and Q_2, than to visit Q_1 twice in a row. In the case of exhaustive service it may in most applications be unnatural to have a positive switchover time between consecutive visits to the same queue. Still, if such positive switchover times exist, examples can be given in which repeated visits to the same queue yield a lower $E\hat{V}$ than alternate visit patterns.

In the 3-queue case of Table II, where all queues receive gated service, step 1 suggests a ratio 3:2:1. Fixing $n_3=1$, we again let (n_1,n_2) range from 1:1 to 6:4, searching exhaustively in this range among all possible orderings. gr coincides with opt in 11 out of the 24 cases; in the other 13 cases it is again off by less than 4%. The predicted ratio 3:2:1 here indeed yields the lowest mean workload.

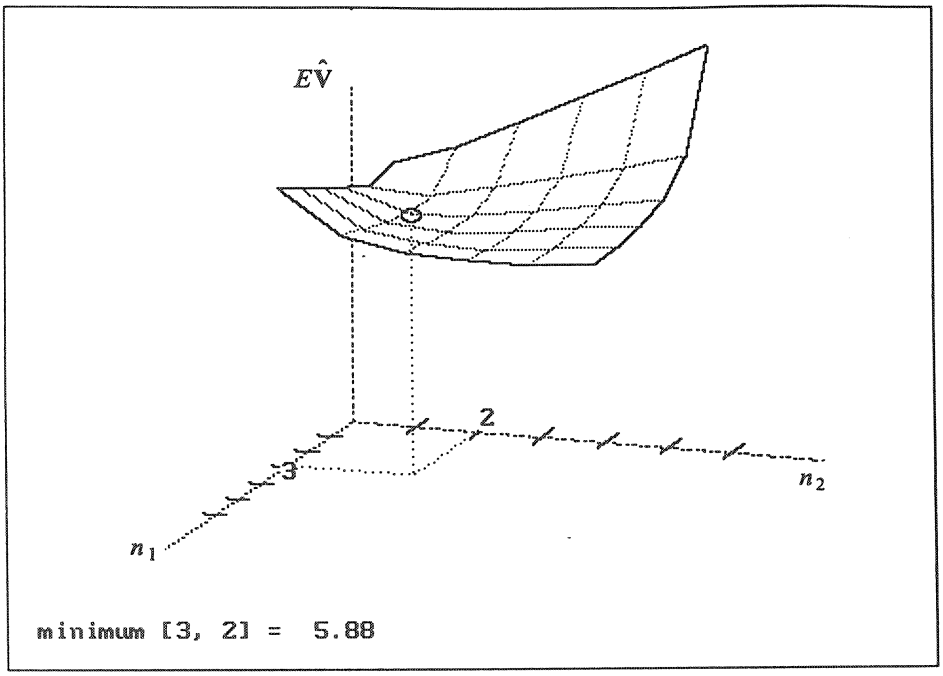

Figure 1 $E\hat{V}$ for a 3-queue case with exhaustive service at Q_1 and gated service at Q_2 and Q_3. $\rho_1=0.66$, $\rho_2=0.10$, $\rho_3=0.025$; $n_3=1$.
Note: this is case 6 of Table IV.

TABLE III

Some 2-queue cases : comparison of the 'optimal' polling table,
the Golden Ratio polling table and random polling.

case	Q_1		Q_2		optimal		GR-approximation			GR-neighb-approx.			Random	
	dis	ρ_1	dis	ρ_2	$\hat{EV}$	table	$\hat{EV}$	%	table	$\hat{EV}$	%	table	$\hat{EV}$	%
1	e	0.66	e	0.06	3.574	12	3.597	-0.6	112	3.574	0.0	1212	4.018	-12.42
2	e	0.12	e	0.05	0.389	12	0.395	-1.5	21211	0.389	0.0	1212	0.475	-22.11
3	g	0.80	g	0.05	12.555	211111	12.600	-0.4	21111	12.555	0.0	111121	13.575	-8.12
4	g	0.62	g	0.35	81.200	1121212	81.200	0.0	2121121	81.200	0.0	2121121	95.237	-17.29
5	g	0.563	g	0.25	10.484	11211212	10.487	-0.0	21211	10.487	-0.0	21211	12.301	-17.30
6	e	0.724	g	0.05	4.541	12	4.573	-0.7	112	4.541	0.0	21	5.028	-10.72

TABLE IV

Some 3-queue cases : comparison of the 'optimal' polling table,
the Golden Ratio polling table and random polling.

case	Q_1		Q_2		Q_3		optimal		GR-approximation			GR-neighb-approx.			Random	
	dis	ρ_1	dis	ρ_2	dis	ρ_3	$\hat{EV}$	table	$\hat{EV}$	%	table	$\hat{EV}$	%	table	$\hat{EV}$	%
1	e	0.617	e	0.158	e	0.015	6.087	2121213	6.224	-2.3	21211213	6.087	0.0	2132121	7.870	-29.29
2	e	0.325	e	0.11	e	0.025	1.697	121213	1.697	0.0	212131	1.697	0.0	212131	2.249	-32.52
3	e	0.385	e	0.385	e	0.015	6.480	212121213	6.943	-7.1	212131212	6.493	-0.2	2132121	8.837	-36.10
4	g	0.625	g	0.225	g	0.025	18.647	1211211213	18.781	-0.7	112131212	18.781	-0.7	112131212	22.775	-21.27
5	g	0.80	g	0.05	g	0.05	22.857	1112113	23.067	-0.92	211131	22.857	0.0	1131121	26.550	-16.16
6	e	0.66	g	0.10	g	0.025	5.875	121213	5.875	0.0	212131	5.875	0.0	212131	7.439	-26.62

Discussion of Figure 1

These tables, and many more examples we have investigated, suggest that there are usually many points around the optimum (with a different occurrence ratio and/or a different table ordering) that yield almost as small a mean workload; but when the occurrence ratio is too far removed from the optimal one, then $E\hat{V}$ shoots up sharply. This shape of $E\hat{V}$ is illustrated in Figure 1. As in Table II, this example concerns a 3-queue model with $n_3=1$. For each $(n_1,n_2,1)$ combination, $E\hat{V}$ has been calculated by determining the lowest value among all (n_1+n_2+1)-size tables with occurrence ratio $n_1{:}n_2{:}1$.

The above conclusions suggest a refinement of the GR policy: apply GR not only to the case $(n_1, \cdots, n_N)$, but also to all its 2^N neighbours, and then take the best result. Note that this would have produced the optimum in Table I; a simpler, much cheaper and also very effective alternative would be to check only the $2N$ *immediate* neighbours, obtained by changing just one n_i.

Discussion of Tables III and IV

The extensive neighbourhood search has been used in Tables III and IV. In Table III six 2-queue cases are considered, with the service combinations exhaustive-exhaustive, exhaustive-gated and gated-gated ('dis' denotes the service discipline). The 'optimal' $E\hat{V}$ and corresponding table (checking all tables of size n_1+n_2 with $n_1{:}n_2$ ranging from 1:1 to 6:6) were printed, along with the GR approximation for the (n_1,n_2) combination suggested by step 1 and the result obtained by applying GR to the eight neighbours of that (n_1,n_2) combination ("GR-neighb-approx."). GR coincides with the 'optimal' result in only 1 out of 6 cases, but it is never more than 1.5% off; the neighbourhood search scores 5 out of 6. Very similar results are obtained in Table IV, which considers six 3-queue cases. Here the difference between the 'optimal' and GR results once equals 7.1%; step 1 does suggest the right ratio, but GR does not produce the most sensible order. In fact this is also the only case in Tables III and IV, where a search of the $2N$ immediate neighbours does not produce the same result as the extensive neighbourhood search - supporting our claim that only checking the immediate neighbours is a very effective alternative to extensive neighbourhood search.

For comparison purposes we also record in Tables III and IV the optimal $E\hat{V}$ for the equivalent *random polling* system. While the random polling system has turned out to be an excellent tool for predicting the best operational rule for periodic polling, its actual $E\hat{V}$ is considerably higher.

Finally we have tested three 10-queue cases, in each of which $\rho_2=\cdots=\rho_{10}$ whereas ρ_1 is such that Step 1 suggests an 11-entry table with Q_1 occurring twice and the other queues once. In each case, this table indeed leads to the minimal value but the ordering produced by GR gives a slightly larger value. The results are:

case 1. $\rho_1=0.255$, $\rho_2=\cdots=\rho_{10}=0.05$, all queues exhaustive: 'opt' = 11.445, 'gr' = 11.596.

case 2. $\rho_1=0.24$, $\rho_2=\cdots=\rho_{10}=0.06$, all queues gated: 'opt' = 20.811, 'gr' = 20.935.

case 3. $\rho_1=0.4$, $\rho_2=\cdots=\rho_{10}=0.06$, Q_1 exhaustive, all other queues gated: 'opt' = 78.328, 'gr' = 79.375.

Conclusions from the numerical experiments

The square root assignment of visit frequencies (step 1) performs excellently. Ratios in the direct neighbourhood of the obtained ratio usually yield results of comparable quality.

The procedure for determining the table size (step 2) has not been extensively tested; it seems to be the least crucial part of the approach.

The Golden Ratio policy for determining the exact visit order of the queues in the table generally works very well. It is extremely easy to apply, and the mean workload that it produces hardly ever exceeds the mean workload for the best order for given table entries $(n_1, \ldots, n_N)$ by more than a few percent.

The combined procedure, with the refinement of applying GR also to the neighbours of the table entry vector $(n_1, \ldots, n_N)$ found via steps 1 and 2, has been tested for a large number of queueing models with high, medium and low traffic, and with exhaustive and/or gated service disciplines. In each case, the mean workload was also calculated for ALL tables of 'reasonable' size and table entry

vector. In almost all cases the two approaches led to the same result; the largest relative difference was 0.7%.

6. Summary and plans for the future

This paper has been devoted to a discussion of optimization problems in polling systems. It has been pointed out that there is a wide range of challenging and important polling optimization problems, only very few of which have been tackled so far. In the present study the issue of deriving efficient server visit rules for polling systems with switchover times has been considered. Specifically, the following optimization problem has been studied: Determine the polling table that minimizes the mean total workload in periodic polling. For the case that the switchover times between all queues have the same distribution, a simple visit rule has been proposed. This rule is based on the exact solution for the related problem of minimizing the mean total workload in a random polling system. Numerical tests show that the rule performs extremely well.

Presently we are generalizing the random polling model to allow Markovian server routing (transition probabilities p_{ij}) and non-identically distributed switchover times. The Lagrangian approach no longer yields explicit expressions for the optimal routing probabilities; a numerical analysis appears to be needed. As in the random polling case, we intend to investigate the use of the optimal Markovian server routing frequencies in polling table design.

The robustness of the rule (which depends on the arrival rates and mean service times only through their product, the traffic load, and which appears to be not very sensitive to the visit frequencies in the neighbourhood of the optimal frequency) gives us some hope that similar rules work reasonably well for a much larger class of systems; e.g., systems with more general arrival processes. This issue is left for future investigation.

Acknowledgement

The authors are indebted to Professor S. Browne (Columbia University) for interesting discussions.

References

[1] Baker, J.E., Rubin, I. (1987). Polling with a general-service order table. *IEEE Trans. Commun.*, Vol. COM-35, 283-288.

[2] Boxma, O.J. (1989). Workloads and waiting times in single-server systems with multiple customer classes. *Report Centre for Mathematics and Computer Science, Amsterdam; to appear in Queueing Systems.*

[3] Boxma, O. J., Groenendijk, W.P. (1987). Pseudo-conservation laws in cyclic-service systems. *J. Appl. Prob.* **24**, 949-964.

[4] Boxma, O.J., Groenendijk, W.P., Weststrate, J.A. (1988). A pseudoconservation law for service systems with a polling table. *Report Centre for Mathematics and Computer Science, Amsterdam; to appear in IEEE Trans. Commun.*

[5] Boxma, O.J., Levy, H., Weststrate, J.A. (1990). Paper in preparation.

[6] Boxma, O.J., Weststrate, J.A. (1989). Waiting times in polling systems with Markovian server routing. *In: G. Stiege and J.S. Lie (eds.), Messung, Modellierung und Bewertung von Rechensystemen und Netzen (Springer, Berlin) pp. 89-104.*

[7] Browne, S., Yechiali, U. (1988). Dynamic scheduling in single server multi-class service systems with unit buffers. *Report Graduate School of Business, Columbia University (NY).*

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

[9] Eisenberg, M. (1972). Queues with periodic service and changeover times. *Oper. Res.* **20**, 440-451.

[10] HOFRI, M., ROSBERG, Z. (1987). Packet delay under the Golden Ratio weighted TDM policy in a multiple-access channel. *IEEE Trans. Inform. Theory,* Vol. IT-33, 341-349.

[11] HOFRI, M., ROSS, K.W. (1987). On the optimal control of two queues with server set-up times and its analysis. *SIAM J. on Computing* **16**, 399-419.

[12] ITAI, A., ROSBERG, Z. (1984). A Golden Ratio control policy for a multiple-access channel. *IEEE Trans. Autom. Control,* Vol. AC-29, 712-718.

[13] KLEINROCK, L. (1964). *Communication Nets - Stochastic Message Flow and Delay.* Dover, New York.

[14] KLEINROCK, L. (1965). A conservation law for a wide class of queueing disciplines. *Naval Res. Logist. Quart.* **12**, 181-192.

[15] KLEINROCK, L., LEVY, H. (1988). The analysis of random polling systems. *Oper. Res.* **36**, 716-732.

[16] KNUTH, D.E. (1973). *The Art of Computer Programming, Vol. 3.* Addison-Wesley, Reading (MA).

[17] LEVY, H. (1988). Optimization of polling systems: The fractional exhaustive service method. *Report Department of Computer Science, Tel-Aviv University.*

[18] LEVY, H. (1989). Analysis of cyclic-polling systems with binomial-gated service. *In: T. Hasegawa, H. Takagi and Y. Takahashi (eds.), Performance of Distributed and Parallel Systems (North-Holland, Amsterdam) pp. 127-139.*

[19] LEVY, H., SIDI, M. (1989). Polling systems: applications, modeling and optimization. *Report Department of Computer Science, Tel-Aviv University.*

[20] LEVY, H., SIDI, M., BOXMA, O.J. (1988). Dominance relations in polling systems. *Report Department of Computer Science, Tel-Aviv University; to appear in Queueing Systems.*

[21] PANWAR, S.S., PHILIPS, T.K., CHEN, M.-S. (1988). Golden Ratio scheduling for low delay flow control in computer networks. *Report RC 13642, IBM Thomas J. Watson Research Center, Yorktown Heights (NY).*

[22] SERVI, L.D. (1986). Average delay approximation of M/G/1 cyclic service queues with Bernoulli schedules. *IEEE J. Sel. Areas Commun.,* Vol. SAC-4, 813-822.

[23] TAKAGI, H. (1986). *Analysis of Polling Systems.* The MIT Press, Cambridge (MA).

[24] TAKAGI, H. (1990). Queueing analysis of polling models. To appear in: *H. Takagi (ed.), Stochastic Analysis of Computer and Communication Systems (North-Holland Publ. Cy., Amsterdam).*