1992

S.C. Borst, O.J. Boxma, J.H.A. Harink, G.B. Huitema

Optimization of fixed time polling schemes

Department of Operations Research, Statistics, and System Theory Report BS-R9237 December

CWI is the research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a non-profit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands organization for scientific research (NWO).

Optimization of Fixed Time Polling Schemes

S.C. Borst
O.J. Boxma
CWI, P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

J.H.A. Harink
G.B. Huitema
PTT Research Tele-informatics, P.O. Box 15000, 9700 CD Groningen, The Netherlands

Abstract

This paper is concerned with the problem of deriving efficient operational rules for polling queues according to a fixed time polling scheme. A fixed time polling scheme specifies the visit order of the queues, but also the starting times for each visit. This problem arose from the need to efficiently collect files with call records from telecommunication switches.
Using a simple approximation for the mean waiting times in a polling system with a non-cyclic visit order of the queues, we present an approach to the problem of minimizing a weighted sum of the mean waiting times at the various queues. This approach is tested via numerical experiments that are partly based on actual data for call records.

1991 Mathematics Subject Classification: 60K25, 68M20.
Keywords & Phrases: telecommunication switch, polling system, waiting time minimization.
Note: Part of the second author's research has been supported by the European Grant BRA-QMIPS of CEC DG XIII.

1 Introduction

Until now, telephone subscribers in the Netherlands have received bills stating only the rental charge and the total cost of telephone calls. These two-monthly bills are based on pulse metering within the switches. The pulses are accumulated and periodically dumped on tape. Finally, the tapes are sent by mail to the billing centre, where they are processed.
In the course of 1992 PTT Telecom has gradually introduced itemized billing. Itemization means that the total call charges are broken down on the bill into categories such as local, long-distance and international calls, special services such as the special-tariff information services and car telephone charges. It will enable subscribers to obtain a detailed picture of all the calls made in particular categories.
In order to offer this new service the trajectory from switches to the billing system has been redesigned to manage the call data, see Figure 1.1. Instead of processing tapes, a newly developed mediation system collects files with call records from the switches. A call record

Report BS-R9237
ISSN 0924-0659
CWI
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

roughly is an electronic set of information concerning a call such as time, duration and the called party. Further the mediation will unify the different call record formats, that is to say, make them independent of the switch and store the data until the billing centre requests the data. Finally in the billing centre the call records are tariffed and the bills are made up.
The set-up of the network: switches - mediation - billing centre will fit within the framework of a Telecom Management Network (TMN). In general, a TMN provides an organized architecture interconnecting various types of Operations Systems (OSs) and telecommunications equipment, using an agreed architecture with standardized interfaces, see [3]. Here in terms of TMN, the mediation interconnects the Network Elements (switches) and an Operations System, viz. the billing centre. Besides this OS, the mediation may in the future also interconnect other OSs which provide (management) services based on the information contained in call records. For these OSs one can think of management systems concerned with, e.g., traffic analysis, fraud detection, trend analysis and provision of marketing information.

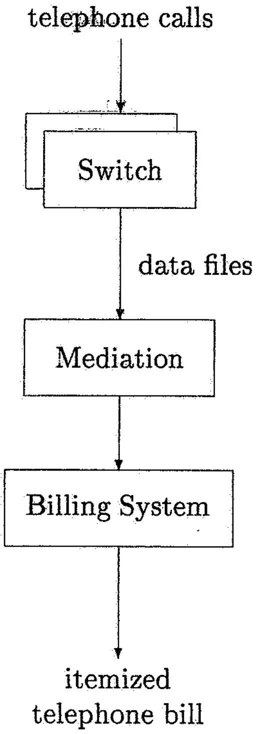

Figure 1.1. The trajectory from switches to the billing system.

The effectiveness of the services provided by the different OSs depends heavily on the performance of the data collection process of the mediation. For instance, a large time delay between making a call and the arrival of the corresponding call record at the billing centre would make near real-time billing impossible. This time delay consists of the waiting time at a switch and the transmission time to the billing centre.
In this paper we will consider the time delay in the billing process. In particular we focus on minimizing the time delay between the generation of files with call records in a switch and the arrival of these files at the mediation. In order to get data files from a specific switch

to the mediation, the mediation sets up a communication link with that switch and requests the data. Since the mediation takes the initiative, we speak of a polling process.
The mediation system that is presently implemented in the network of PTT Telecom only allows for polling based on a so called fixed time polling (ftp) scheme. That means, the mediation initiates a polling session according to a schedule with switches and starting times. The built-in guidelines to determine the ftp schedule are not satisfactory. The first step of this method consists of determining the polling frequencies, more or less proportional to the average number of call records created at the switches. From these frequencies one obtains the starting times by evenly spacing the polling sessions over the day. This set-up clearly does not lead to an overall minimal time delay since the visits to a switch are scheduled without taking notice of visits to other switches.
The problem here is to find an ftp scheme that minimizes the mean waiting time. In this paper we study this optimization problem by modelling the network as a single server multiple queue system. Here the mediation behaves as a server attending the queues, i.e., polling the switches for data files. It appears that one can apply to this case some theoretical results on efficient visit orders in periodic polling systems, see [2]. This approach leads to an ftp scheme which, compared with the ftp scheme currently implemented in the mediation of PTT Telecom, reduces the mean waiting time by approximately 70%, see [6]. The developed method may also have its merits in analyzing and optimizing the performance of time-limited access protocols in local area networks.

This paper is organized as follows. In section 2 the data collection process of the mediation is modelled as a queueing theoretic polling system. In section 3 the problem of constructing an ftp scheme that minimizes the mean total waiting cost per unit of time (under some side-constraint) is formulated as a mathematical program. Unfortunately this program appears to be NP-hard. Hence in section 4 a heuristic method is developed to solve the mathematical program. The accuracy of the heuristics is tested in section 5 by simulation experiments, partly based on real network data. It is shown that the proposed ftp schemes give good results compared with some other natural, but less sophisticated, ftp schemes. In addition it is investigated whether perturbation of the proposed ftp scheme leads to further improvements. Finally, in section 6 a conclusion is presented.

2 Modelling of polling call records from switches

In this section the data collection process of the mediation will be modelled as a queueing theoretic polling system. We first describe the basic polling model.
The basic polling model is a set of n queues $Q_1, \ldots, Q_n$, served by a single server, S, which visits the queues in cyclic order. The interarrival times of customers arriving at Q_i are independent, identically distributed stochastic variables, their distribution being $A_i(\cdot)$, $i = 1, \ldots, n$. The arrival intensity at Q_i is λ_i, $i = 1, \ldots, n$, and the total arrival intensity is $\lambda = \sum_{i=1}^{n} \lambda_i$. Customers arriving at Q_i are called type-i customers. The service times of type-i customers are independent, identically distributed stochastic variables. Their distribution $B_i(\cdot)$ has first moment β_i, $i = 1, \ldots, n$. The offered traffic load, ρ_i, at Q_i is defined as $\rho_i := \lambda_i \beta_i$, $i = 1, \ldots, n$, and the total offered load, ρ, as $\rho := \sum_{i=1}^{n} \rho_i$. When swapping into Q_i, S incurs

a switch-over period of type i; switch-over durations (of type i) are independent stochastic variables $\mathbf{S}_i$ with mean s_i. The interarrival, service, and switch-over processes are independent stochastic processes.
At each queue the server operates according to some service discipline. E.g. the exhaustive service discipline, i.e., S serves customers until the queue is empty, or the gated service discipline, i.e., S serves exactly those customers present at the queue at the beginning of the visit period.
A generalization of the basic cyclic polling system is the polling system with fixed (generally non-cyclic) visit order. This order is described in a table (*polling table*) in which the number of visits given to Q_i, m_i, is at least 1. The size of a table, which is the total number of visits in a cycle, is $m = \sum_{i=1}^{n} m_i$.
Polling models have recently received much attention in the queueing literature, partly because of their applicability in the performance analysis of computer and communication networks. The survey of Takagi [12] contains 455 references, more than half of which appeared after 1985. Most of these papers are concerned with the exact or approximate ***analysis*** of polling systems. Only recently some studies have been devoted to the issue of *optimization* of polling systems, in particular the optimal routing of the server along the queues. The Proceedings of the 13-th International Teletraffic Congress contain some reviews of static [1] and semi-dynamic [13] server routing in polling systems.

We shall now describe how the data collection process of the mediation fits into the depicted general framework. When a telephone call is established, data is collected by the switch and formatted into a call record. Upon completion of the call, the record is transferred to some storage medium. Here the records are grouped into switch files. When the file is full, it is closed and is ready for polling by the mediation. The mediation initiates a polling session with a switch when the prescibed starting time for this switch holds. The starting time is found in the ftp scheme. During a polling session all the switch files which are closed are transmitted to the mediation.
In this situation we clearly can define a polling model for a single server visiting queues in some fixed order. Here the mediation is the server S and the queues $Q_1, \ldots, Q_n$ are the connected switches onto the mediation. The switches have a storage capacity for files which is sufficient for some days. So the queues can in fact be considered to have infinite storage capacity. The customers which are served at the queues are the switch files with call records. The lengths of these files depend on the type of switch involved. For example presently the Dutch public switched telephone network contains five different types of switches. Calls on a switch are assumed to occur according to a Poisson process. Therefore the interarrival time of switch files, that is the time between the completion of two consecutive data files, will be Erlang distributed. For large files, the Erlang distribution converges rather quickly to a deterministic distribution. Therefore we here assume that the arrival process of switch files is almost deterministic. The service time of a customer at a queue is in fact the transmission time of a data file from the corresponding switch to the mediation. This transmission time depends on the length of a switch file and the (effective) transmission rate. We assume that the service times are almost deterministic. Finally, we also assume an almost deterministic switch-over process. In the present case a switch-over time is the time the mediation needs

to establish a communication link with the next switch to be polled. The server operates according to the gated service discipline, as during a polling session all switch files which are closed are transmitted to the mediation.

Notice that the gated service discipline may conflict with the fixed time aspect of the present polling scheme. That is, a current polling session may need more time than planned. In the polling model we assume that in case of a conflict the gated service discipline goes before the fixed time aspect of the polling scheme, i.e., the long polling session and the next one are carried out sequentially. In reality, the multitasking aspect of the mediation makes it possible to continue those long sessions without delaying the next polling session. According to the ftp scheme the new scheduled session is simply run parallel to the current one. In this case the mediation polls more data in a shorter time but internally the mediation can not keep up with the processing of the data files. The net effect is as if the polling sessions were carried out sequentially.

In reality data collection systems like the mediation system may have a number of communication ports over which independent polling sessions can be run. In order to avoid complex modelling we here consider a mediation system with only one communication port.

From the Introduction we recall that we are interested in the problem of constructing an ftp scheme that minimizes the time delay between the generation of files and the arrival of these files at the mediation, in particular the mean waiting time of files. To keep down the conflicts between the gated service discipline and the fixed time aspect of the polling scheme, we solve the problem under the side-constraint of most rarely exceeding the available visit times. In fact we shall consider the more general optimization problem of minimizing the mean total waiting cost per time unit $\sum_{i=1}^{n} c_i \lambda_i \mathrm{E}\mathbf{W}_i$, where $\mathbf{W}_i$ denotes the waiting time of an arbitrary type-i customer and c_i is an arbitrary positive parameter reflecting the cost of waiting one unit of time at Q_i. This general case contains the original problem of minimizing the mean waiting time (take $c_i = c$). We shall also assume more general distributions of the interarrival, service, and switch-over times.

3 Constructing an ftp scheme I

As stated in the previous section, we are interested in the problem of constructing an ftp scheme that minimizes the mean total waiting cost per unit of time, under the side-constraint of most rarely exceeding the available visit times. Starting from rather simple approximations, we formulate in the present section the problem under consideration as a mathematical program. In view of its NP-hardness we describe in the next section a heuristic method for solving the mathematical program. The approach bears resemblance to the approach to a similar problem in Kruskal [9] for polling systems with deterministic arrival, service, and switch-over processes, and in Boxma, Levy, & Weststrate [2] for polling systems with a Poisson arrival process and general service and switch-over processes.

We first introduce some additional notation. We represent an ftp scheme by a vector pair (P, T), $P_k \in \{1, \ldots, n\}$, $T_k \geq 0$, $k = 1, \ldots, m$. The vector P contains the polling table associated with the ftp scheme, i.e., the k-th visit is to queue P_k, $k = 1, \ldots, m$. The vector T contains the extended visit times associated with the ftp scheme, i.e., T_k is the time between the start of the k-th visit and the start of the $(k+1)$-th visit, $k = 1, \ldots, m$, where $m + 1$ is

to be understood as 1.
Several further quantities associated with an ftp scheme will also appear to be of interest. Denote by $\mathbf{U}_k$ the k-th available visit time, i.e., the k-th extended visit time minus the switch-over time into queue P_k, by $\mathbf{V}_k$ the k-th required visit time, i.e., the time the server needs to do the work during the k-th visit, and by SC_k the k-th subcycle time, i.e., the time between the start of the k-th visit and the start of the previous visit to queue P_k, $k = 1, \dots, m$. Denote by C the cycle time, i.e., the time the server needs to pass through the ftp scheme once. By the nature of the ftp scheme, T_k, SC_k, and C are deterministic, but $\mathbf{U}_k$ and $\mathbf{V}_k$ are not, except when respectively the switch-over process and the arrival and service processes are deterministic, the interarrival time in addition being a divisor of the cycle time.
Of course T_k, $\mathbf{U}_k$, $\mathbf{V}_k$, SC_k, and C are closely related. Firstly, the k-th available visit time $\mathbf{U}_k$ is what remains of the k-th extended visit time T_k, after the switch-over time into queue P_k,

$$T_k = \mathbf{U}_k + \mathbf{S}_{P_k}, \qquad k = 1, \dots, m. \tag{3.1}$$

Secondly, by the nature of the gated service discipline, the k-th required visit time $\mathbf{V}_k$ equals the amount of work that arrives at queue P_k during the k-th subcycle time SC_k, $k = 1, \dots, m$. The k-th subcycle time is composed of the extended visit times between the start of the k-th visit and the start of the previous visit to queue P_k,

$$SC_k = \sum_{l=1}^{m} h_{kl} T_l, \qquad k = 1, \dots, m. \tag{3.2}$$

Here the matrix $H = (h_{kl})$ is defined by

$$h_{kl} = \begin{cases} 1 & \text{if } P_{l+1}, \dots, P_{k-1} \neq P_k \\ 0 & \text{otherwise} \end{cases} \qquad k, l = 1, \dots, m,$$

i.e., h_{kl} indicates whether the l-th extended visit time belongs to the k-th subcycle time: $h_{kl} = 0$ iff the k-th subcycle time begins after the start of the l-th visit.
The cycle time is composed of all the extended visit times,

$$C = \sum_{l=1}^{m} T_l. \tag{3.3}$$

The cycle time may as well be viewed as consisting of all the subcycle times corresponding to any Q_i,

$$C = \sum_{\{k: P_k = i\}} SC_k, \qquad i = 1, \dots, n. \tag{3.4}$$

Notice that substituting (3.2) into (3.4) indeed yields (3.3), as $\sum_{\{k: P_k = i\}} h_{kl} = 1$, $i = 1, \dots, n$, $l = 1, \dots, m$.

To be able to formulate the problem as a mathematical program, we now express the mean total waiting cost per unit of time, as well as the side-constraint of most rarely exceeding the available visit times, in terms of the ftp scheme.
To start with the latter, we may represent the side-constraint by

$$\Pr\{\mathbf{U}_k < \mathbf{V}_k\} \leq \alpha_{P_k}, \qquad k = 1, \ldots, m, \tag{3.5}$$

with α_i, $i = 1, \ldots, n$, prespecified bounds for the probabilities of exceeding the available visit times. As (3.5) does not really fit into the framework of a mathematical program, we replace (3.5) by a constraint of the type

$$\mathrm{E}\mathbf{U}_k - Y_k \geq \mathrm{E}\mathbf{V}_k + Z_k, \qquad k = 1, \ldots, m.$$

Here Y_k and Z_k are measures for the variability of respectively $\mathbf{U}_k$ and $\mathbf{V}_k$ such that $\Pr\{\mathbf{U}_k < \mathrm{E}\mathbf{U}_k - Y_k\} \leq \gamma_{P_k}\alpha_{P_k}$ and $\Pr\{\mathbf{V}_k > \mathrm{E}\mathbf{V}_k + Z_k\} \leq (1-\gamma_{P_k})\alpha_{P_k}$, with $0 \leq \gamma_i \leq 1$, $i = 1, \ldots, n$. Thus we slightly strengthen (3.5), as $\Pr\{\mathbf{U}_k \geq \mathbf{V}_k\} \geq \Pr\{\mathbf{U}_k \geq \mathrm{E}\mathbf{U}_k - Y_k, \mathbf{V}_k \leq \mathrm{E}\mathbf{V}_k + Z_k\} = \Pr\{\mathbf{U}_k \geq \mathrm{E}\mathbf{U}_k - Y_k\}\Pr\{\mathbf{V}_k \leq \mathrm{E}\mathbf{V}_k + Z_k\} = 1 - \alpha_{P_k} + \gamma_{P_k}(1-\gamma_{P_k})\alpha_{P_k}^2$. From (3.1) we have $\mathrm{E}\mathbf{U}_k = T_k - s_{P_k}$, $k = 1, \ldots, m$. As $\mathbf{V}_k$ equals the amount of work that arrives at Q_{P_k} during SC_k, we have $\mathrm{E}\mathbf{V}_k = \rho_{P_k} SC_k$, $k = 1, \ldots, m$. We further take $Y_k = \delta_{P_k} s_{P_k}$, $k = 1, \ldots, m$, and $Z_k = \epsilon_{P_k}\rho_{P_k} SC_k$, $k = 1, \ldots, m$. Here δ_i and ϵ_i are measures for the variability of respectively the switch-over times into Q_i and the interarrival and service times at Q_i such that

$$\Pr\{\mathbf{S}_i > (1+\delta_i)s_i\} \leq \gamma_i\alpha_i,$$

$$\sum_{h=0}^{\infty}\left(A_i^{h*}(\hat{SC}_i) - A_i^{(h+1)*}(\hat{SC}_i)\right)\left(1 - B_i^{(h+1)*}(\rho_i(1+\epsilon_i)\hat{SC}_i)\right) \leq (1-\gamma_i)\alpha_i,$$

with $\hat{SC}_i$ a rough estimate for the subcycle times corresponding to Q_i, preferably as pessimistic as possible from the viewpoint of determining ϵ_i. Thus we slightly strengthen (3.5) further as

$$\Pr\{\mathbf{U}_k < \mathrm{E}\mathbf{U}_k - Y_k\} = \Pr\{\mathbf{S}_{P_k} > (1+\delta_{P_k})s_{P_k}\},$$

$$\Pr\{\mathbf{V}_k > \mathrm{E}\mathbf{V}_k + Z_k\} = \Pr\{\mathbf{V}_k > \rho_{P_k}(1+\epsilon_{P_k})SC_k\},$$

$$\Pr\{\mathbf{V}_k > t\} \leq \sum_{h=0}^{\infty}\left(A_{P_k}^{h*}(SC_k) - A_{P_k}^{(h+1)*}(SC_k)\right)\left(1 - B_{P_k}^{(h+1)*}(t)\right), \qquad t \geq 0,$$

particularly for $t = \rho_{P_k}(1+\epsilon_{P_k})SC_k$. The latter inequality holds, since $A_{P_k}^{h*}(SC_k) - A_{P_k}^{(h+1)*}(SC_k)$ is the probability that $(h+1)$ type-P_k customers arrive during SC_k, under the pessimistic assumption that the first customer arrives at the beginning of SC_k, and $1 - B_{P_k}^{(h+1)*}(t)$ is the probability that the service of $(h+1)$ type-P_k customers is not finished within time t, $t \geq 0$. Instead of $Z_k = \epsilon_{P_k}\rho_{P_k}SC_k$ one might take e.g. $Z_k = \epsilon_{P_k}\rho_{P_k}SC_k + (1+\zeta_{P_k})\beta_{P_k}$. Here ζ_i is a measure for the variability of the service times at Q_i such that $1 - B_i((1+\zeta_i)\beta_i) \leq (1-\gamma_i)\alpha_i$, $i = 1, \ldots, n$. The factor $(1+\zeta_i)\beta_i$ would avoid that $\epsilon_i \to \infty$ in the hypothetic situation that $\hat{SC}_i \downarrow 0$. If indeed one takes $Z_k = \epsilon_{P_k}\rho_{P_k}SC_k + (1+\zeta_{P_k})\beta_{P_k}$ instead of $Z_k = \epsilon_{P_k}\rho_{P_k}SC_k$, then everywhere $(1+\delta_i)s_i$ is to be replaced by $(1+\delta_i)s_i + (1+\zeta_i)\beta_i$.

When the distributions of the interarrival, service, and switch-over times are specified, there are no serious complications in expressing δ_i and ϵ_i in terms of α_i. Nevertheless, in real life one may rather determine δ_i and ϵ_i empirically than make a questionable assumption about the distributions of the interarrival, service, and switch-over times, needed in expressing δ_i and ϵ_i in terms of α_i.

Concluding, we represent the side-constraint of most rarely exceeding the available visit times by

$$T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k}, \qquad k=1,\ldots,m. \tag{3.6}$$

Notice that summing (3.6) with respect to $k = 1,\ldots,m$, using (3.3) and (3.4), yields $C \geq \left(\rho + \sum_{i=1}^{n} \epsilon_i \rho_i\right) C + \sum_{i=1}^{n} m_i(1+\delta_i)s_i$. Apparently $\rho + \sum_{i=1}^{n} \epsilon_i\rho_i < 1$ is a necesssary condition for the extended visit times to be all non-negative. In the proof of Lemma 4.1 it will also appear to be a sufficient condition. In the sequel the condition $\rho + \sum_{i=1}^{n} \epsilon_i\rho_i < 1$ is always assumed to hold.

We now express the mean total waiting cost per unit of time in terms of the ftp scheme. Unfortunately a polling system with an ftp scheme is not likely to be amenable to an exact analysis, except when the arrival, service, and switch-over processes are deterministic. We therefore feel justified in resorting to approximations for the mean waiting times. The approximations should be simple enough to lend themselves to optimization purposes. In fact we feel even further justified in resorting to approximations for the mean waiting times, by the knowledge that in hardly any polling system the exact expressions for the mean waiting times are simple enough to lend themselves to optimization purposes. Fortunately we do not really need to bother about restricting ourselves to simple expressions for the mean waiting times. As we are primarily concerned with minimizing a weighted sum of the mean waiting times rather than evaluating the mean waiting times themselves, the approximations do not need to be very accurate as long as they rightly capture the *behavior* of the mean waiting times. E.g., approximations that systematically deviate from the mean waiting times by some constant additive or multiplicative factor would be perfectly suitable for minimizing a weighted sum of the mean waiting times.

We now approximate the mean waiting time of an arbitrary type-i customer. We condition on the event that we are dealing with an arbitrary type-i customer that arrives during the k-th subcycle time SC_k with $P_k = i$, which occurs with probability SC_k/C. Further we act as if the available visit times are never exceeded. The waiting time of an arbitrary type-i customer that arrives during the k-th subcycle time SC_k with $P_k = i$, is then composed of (i) the time from its arrival to the start of the next visit to Q_i, i.e., the residual lifetime $\mathbf{RSC}_k$ of SC_k at the arrival epoch of the customer and (ii) the time from the start of the next visit to Q_i to the start of its service, i.e., the time the server needs to do the work $\mathbf{PV}_k$ that arrived at Q_i during the past lifetime $\mathbf{PSC}_k$ of SC_k at the arrival epoch of the customer. As SC_k is deterministic, $\mathbf{ERSC}_k = \frac{1}{2}SC_k$, $\mathbf{EPSC}_k = \frac{1}{2}SC_k$. Further we use the approximation $\mathbf{EPV}_k \approx \rho_i \mathbf{EPSC}_k$, which in fact is exact for a Poisson arrival process.

Concluding, we approximate the mean waiting time of an arbitrary type-i customer by

$$\mathbf{EW}_i \approx \frac{1+\rho_i}{2C} \sum_{\{k:P_k=i\}} SC_k^2, \qquad i=1,\ldots,n, \tag{3.7}$$

which yields for the mean total waiting cost per unit of time

$$\sum_{i=1}^{n} c_i\lambda_i \mathbf{EW}_i \approx \frac{1}{2C}\sum_{i=1}^{n} c_i\lambda_i(1+\rho_i) \sum_{\{k:P_k=i\}} SC_k^2 = \frac{1}{2C}\sum_{k=1}^{m} c_{P_k}\lambda_{P_k}(1+\rho_{P_k})SC_k^2. \tag{3.8}$$

Having expressed the mean total waiting cost per unit of time, as well as the side-constraint of most rarely exceeding the available visit times, in terms of the ftp scheme, we are now able to formulate the problem as a mathematical program.

Problem (I).

$$\text{minimize} \quad \frac{1}{2C}\sum_{i=1}^{n} c_i\lambda_i(1+\rho_i)\sum_{\{k:P_k=i\}} SC_k^2 = \frac{1}{2C}\sum_{k=1}^{m} c_{P_k}\lambda_{P_k}(1+\rho_{P_k})SC_k^2 \tag{3.9}$$

$$\text{subject to} \quad T_k \geq \rho_{P_k}(1+\epsilon_{P_k})SC_k + (1+\delta_{P_k})s_{P_k}, \qquad k=1,\dots,m; \tag{3.10}$$

$$SC_k = \sum_{l=1}^{m} h_{kl}T_l, \qquad k=1,\dots,m; \tag{3.11}$$

$$C = \sum_{l=1}^{m} T_l; \tag{3.12}$$

$$m_i = |\{k : P_k = i\}| \geq 1, \qquad i=1,\dots,n; \tag{3.13}$$

$$h_{kl} = \begin{cases} 1 & \text{if } P_{l+1},\dots,P_{k-1} \neq P_k \\ 0 & \text{otherwise} \end{cases} \qquad k,l=1,\dots,m; \tag{3.14}$$

$$P_k \in \{1,\dots,n\}, \qquad k=1,\dots,m; \tag{3.15}$$

$$T_k \geq 0, \qquad k=1,\dots,m. \tag{3.16}$$

For a specific parameter choice problem (I) amounts to the problem of partitioning $m-2$ numbers into 2 sets, such that the sums of the numbers in both sets are as equal as possible, which is known to be NP-hard; cf. Lemma 3.1.

Lemma 3.1
Problem (I) is NP-hard.

Proof
See appendix A.

□

Lemma 3.1 suggests that there is little hope of solving problem (I) exactly in a reasonable amount of time. In the next section we therefore describe a method for solving problem (I) approximately.

4 Constructing an ftp scheme II

In the previous section we formulated the problem under consideration as a mathematical program. In view of its NP-hardness we describe in the present section a heuristic method for solving the mathematical program. The idea is to divide problem (I) into three subproblems, which are somewhat easier to handle, viz., successively:

1. Determination of the visit numbers $m_1,\dots,m_n$.
2. Determination of the visit order.
3. Determination of the extended visit times $T_1,\dots,T_m$.

Ad 1. Determination of the visit numbers.
To simplify the determination of the visit numbers, we forget about the visit order for now. So we ignore of which extended visit times the k-th subcycle time is composed, but of course we do not ignore that the subcycle times corresponding to any Q_i together make up the cycle time, cf. (3.4). Translated to problem (I), we replace the constraints (3.11), (3.14), and (3.15) by the constraint (3.4). It is easily verified that in the resulting problem the optimal subcycle times corresponding to Q_i are all equal, $i = 1, \ldots, n$, i.e., $SC_k = C/m_{P_k}$, $k = 1, \ldots, m$. Observe that $\sum_{h=1}^{H} x_h^2$, under the constraint $\sum_{h=1}^{H} x_h = X$, is minimal for $x_h = X/H$, $h = 1, \ldots, H$. Notice that $SC_k = C/m_{P_k}$, $k = 1, \ldots, m$, suggests spacing the visits to the various queues as evenly as possible, as intuitively is indeed expected to be optimal. As seen from (3.10) and (3.12), all the optimal extended visit times at Q_i are then all equal too, $i = 1, \ldots, n$, i.e., $T_k = \rho_{P_k}(1 + \epsilon_{P_k})C/m_{P_k} + (1 + \delta_{P_k})s_{P_k}$, $k = 1, \ldots, m$. Denote by D_i the common value of all these optimal extended visit times at Q_i, $i = 1, \ldots, n$. As we forget about the visit order for now, the ultimate extended visit times at Q_i will probably deviate from D_i. Remember that because of the constraint (3.10) the extended visit times can not be determined before the visit order is determined. Nevertheless, D_i will probably be a good indication for the ultimate extended visit times at Q_i, which will be useful in determining a good visit order later on. To simplify the determination of the visit numbers even further, we relax the integrality constraint (3.13) for now too. Concluding, we formulate the problem of determining the visit numbers as follows.

Problem (II).

$$\text{minimize} \quad \sum_{i=1}^{n} \frac{c_i \lambda_i (1 + \rho_i) C}{2 m_i} \tag{4.1}$$

$$\text{subject to} \quad D_i = \frac{\rho_i (1 + \epsilon_i) C}{m_i} + (1 + \delta_i) s_i, \quad i = 1, \ldots, n; \tag{4.2}$$

$$C = \sum_{i=1}^{n} m_i D_i; \tag{4.3}$$

$$m_i > 0, \qquad i = 1, \ldots, n. \tag{4.4}$$

Notice that the objective function as well as the constraints are homogeneous with regard to $(m_1, \ldots, m_n, C)$, as is to be expected, since concatenating an ftp scheme several times does not make any difference. So we know beforehand that the optimal solution contains a positive scaling factor with regard to $(m_1, \ldots, m_n, C)$. Using the Lagrangean multiplier technique, we find that the optimal solution is

$$D_i^* = \frac{\rho_i (1 + \epsilon_i) C^*}{m_i^*} + (1 + \delta_i) s_i, \qquad i = 1, \ldots, n; \tag{4.5}$$

$$C^* = \frac{R \sum_{i=1}^{n} \sqrt{c_i \lambda_i (1 + \rho_i)(1 + \delta_i) s_i}}{1 - \rho - \sum_{i=1}^{n} \epsilon_i \rho_i}; \tag{4.6}$$

$$m_i^* \quad = \quad R\sqrt{\frac{c_i\lambda_i(1+\rho_i)}{(1+\delta_i)s_i}}, \qquad i=1,\ldots,n. \tag{4.7}$$

Here R is the positive scaling factor mentioned above, due to which some freedom remains in determining the total number of visits m. One may e.g. choose R such that $| \frac{m_i^* - [m_i^*]}{m_i^*} | \leq \eta_i$, $i = 1,\ldots,n$, with $[x]$ denoting the nearest integer to x and η_i prespecified bounds for the relative deviation of the true, integer visit numbers from the desirable, generally non-integer visit numbers. Alternatively, one may choose R such that the cycle time has some desirable value, which matches e.g. the daily pattern in the actual telecommunication application. For (4.5), (4.6), and (4.7), the objective function (4.1) takes the value

$$\frac{\left(\sum\limits_{i=1}^{n}\sqrt{c_i\lambda_i(1+\rho_i)(1+\delta_i)s_i}\right)^2}{2\left(1-\rho-\sum\limits_{i=1}^{n}\epsilon_i\rho_i\right)}. \tag{4.8}$$

As seen from the argumentation preceding the formulation of problem (II), formula (4.8) provides a lower bound for the value of (3.9) for the optimal solution of problem (I). As far as the determination of the visit numbers is concerned, problem (II) may thus be conceived as relaxation of problem (I).

Remark 4.1 Apart from the slack coefficient δ_i, formula (4.7) agrees with results in Kruskal [9] for polling systems with deterministic arrival, service, and switch-over processes, and in Boxma, Levy, & Weststrate [2] for polling systems with a Poisson arrival process and general service and switch-over processes. In the latter paper it has already been argued that the optimal visit numbers should be quite robust with respect to the distributions of the interarrival, service, and switch-over times.

□

In the special case that we confine ourselves beforehand to strictly cyclic polling, i.e., $m_i = 1$, $i = 1,\ldots,n$, problem (II) reduces to

$$\text{minimize} \quad \sum_{i=1}^{n}\frac{c_i\lambda_i(1+\rho_i)C}{2} \tag{4.9}$$

$$\text{subject to} \quad D_i = \rho_i(1+\epsilon_i)C + (1+\delta_i)s_i, \qquad i=1,\ldots,n; \tag{4.10}$$

$$C = \sum_{i=1}^{n} D_i. \tag{4.11}$$

The only feasible and hence optimal solution is

$$D_i^* \quad = \quad \rho_i(1+\epsilon_i)C^* + (1+\delta_i)s_i, \qquad i=1,\ldots,n; \tag{4.12}$$

$$C^* \quad = \quad \frac{\sum\limits_{i=1}^{n}(1+\delta_i)s_i}{1-\rho-\sum\limits_{i=1}^{n}\epsilon_i\rho_i}. \tag{4.13}$$

For (4.12) and (4.13), the objective function (4.9) takes the value

$$\frac{\left(\sum_{i=1}^{n} c_i\lambda_i(1+\rho_i)\right)\left(\sum_{i=1}^{n}(1+\delta_i)s_i\right)}{2\left(1-\rho-\sum_{i=1}^{n}\epsilon_i\rho_i\right)}. \tag{4.14}$$

Because of the Hölder inequality (4.14) can not be smaller than (4.8), as is to be expected, when we confine ourselves to strictly cyclic polling. In fact the difference between (4.14) and (4.8) gives a rough estimate of the increase in the mean total waiting cost per unit of time, when we confine ourselves to strictly cyclic polling.

Ad 2. Determination of the visit order.
To facilitate the determination of the visit order, we assume that the extended visit times at Q_i are all equal to D_i^*, the indication for the extended visit times at Q_i obtained in (4.5). Translated to problem (I), we replace the constraint (3.10) by the constraint $T_k = D_{P_k}^*$. As seen from the proof of Lemma 3.1 however, the determination of the optimal visit order for fixed m_i and fixed $T_k = D_{P_k}^*$ is still NP-hard. Nevertheless, we rather solve problem (I) for fixed m_i and fixed $T_k = D_{P_k}^*$ approximately than an even further garbled version of problem (I) exactly.
In appendix B we describe the Golden Ratio procedure, which is an approved method for spacing the visits to the various queues as evenly as possible, cf. [7], [8], and [10]. To be specific, define $X_k = \sum_{l=1}^{m} h_{kl}$, i.e., X_k is the number of visits between the start of the k-th visit and the start of the previous visit to queue P_k, $k = 1,\ldots,m$. The Golden Ratio procedure aims at making the numbers X_k with $P_k = i$ as equal as possible. In fact the numbers X_k with $P_k = i$ are guaranteed to take at most three different values. However, these three different values are *not* guaranteed to be all nearly equal. Moreover, the Golden Ratio procedure aims at making the *numbers* of visits X_k with $P_k = i$ as equal as possible, instead of the *periods* between visits SC_k with $P_k = i$, as we should, cf. the argumentation preceding the formulation of problem (II). In other words, the Golden Ratio procedure aims at solving problem (I) for $T_k = 1$, $k = 1,\ldots,m$, instead of $T_k = D_{P_k}^*$, $k = 1,\ldots,m$. Lastly, the Golden Ratio procedure does not take into account the coefficients $c_i\lambda_i(1+\rho_i)$ in the objective function (3.9) to weigh the improvement in the spacing of the visits to one queue against the deterioration in the spacing of the visits to another queue. In appendix C we describe a procedure based on extremal splittings, which to some extent meets these objections.
Whichever of these procedures is used, it is always worthwhile to make sure that the visit order is optimal with respect to some neighborhood. One may e.g. attempt to improve the visit order by interchanging pairs of consecutive visits.

Ad 3. Determination of the extended visit times.
At first sight it does not seem to make sense to protract a visit any longer than needed to satisfy the side-constraint of most rarely exceeding the available visit times. Remember that the server will most rarely be busy during the extra time. Still for extreme parameter choices it may make sense to protract a visit. Take e.g. $n = 101$, $\beta_i = 0$, $\delta_i = 0$, $i = 1,\ldots,n$. The side-constraint (3.5) then reduces to $T_k \geq s_{P_k}$, $k = 1,\ldots,m$. Take $c_1\lambda_1 = 10000$, $c_i\lambda_i = 1$, $i = 2,\ldots,100$, $c_{101}\lambda_{101} = 100$, $s_1 = 1$, $s_i = 1$, $i = 2,\ldots,100$, $s_{101} = 100$. From (4.7) we then obtain $m_1 = 100R$, $m_i = R$, $i = 2,\ldots,100$, $m_{101} = R$, which for $R = 1$ yields the polling

table $P_{2i-1} = 1$, $i = 1, \ldots, 100$, $P_{2i-2} = i$, $i = 2, \ldots, 100$, $P_{200} = 101$. It is easily verified that (3.9) is larger for $T_k = s_{P_k}$, $k = 1, \ldots, 200$, than for $T_{2i-2} = 2 > s_{P_{2i-2}}$ (protracted visit), $T_k = s_{P_k}$, $k \neq 2i-2$ for any $i = 2, \ldots, 100$. Nevertheless, for realistic parameter choices it will seldom really pay off to protract a visit. To facilitate the determination of the extended visit times, we therefore assume that the side-constraint of most rarely exceeding the available visit times is satisfied without any slack. Translated to problem (I), we assume that the constraint (3.10) is satisfied without any slack. Thus determining the extended visit times amounts to solving a set of linear equations; cf. Lemma 4.1.

Lemma 4.1
The set of linear equations

$$T_k = \rho_{P_k}(1+\epsilon_{P_k})\sum_{l=1}^{m} h_{kl}T_l + (1+\delta_{P_k})s_{P_k}, \qquad k = 1, \ldots, m, \tag{4.15}$$

has a unique solution; this solution is non-negative.

Proof
See appendix D.

□

Remark 4.2 Notice that summing (4.15) with respect to $\{k : P_k = i\}$ yields for the mean total available visit time at Q_i during a cycle $\sum_{\{k:P_k=i\}} T_k - m_i s_i = \rho_i(1+\epsilon_i)C + m_i\delta_i s_i$, as $\sum_{\{k:P_k=i\}} h_{kl} = 1$, $i = 1, \ldots, n$, $l = 1, \ldots, m$. The mean total available visit time may be viewed as consisting of (i) $\rho_i C$, the time needed to satisfy the stability condition and (ii) $\epsilon_i\rho_i C + m_i\delta_i s_i$, the extra time above $\rho_i C$ needed to satisfy the side-constraint of most rarely exceeding the available visit times.

□

We finally summarize the method for constructing an ftp scheme as follows.
1. Determination of the visit numbers.
Calculate the desirable visit frequencies

$$f_i^* = \frac{\sqrt{\dfrac{c_i\lambda_i(1+\rho_i)}{(1+\delta_i)s_i}}}{\sum\limits_{j=1}^{n}\sqrt{\dfrac{c_j\lambda_j(1+\rho_j)}{(1+\delta_j)s_j}}}, \qquad i = 1, \ldots, n. \tag{4.16}$$

Determine the total number of visits m^*.
One may choose m^* e.g. such that

$$m^* = \sum_{i=1}^{n}[m^* f_i^*], \qquad [m^* f_i^*] \geq 1, \qquad \frac{\left| f_i^* - \dfrac{[m^* f_i^*]}{m^*} \right|}{f_i^*} \leq \eta_i, \qquad i = 1, \ldots, n, \tag{4.17}$$

with $[x]$ denoting the nearest integer to x and η_i prespecified bounds for the relative deviation of the true visit frequencies from the desirable visit frequencies.
Alternatively, one may choose m^* such that

$$m^* = \sum_{i=1}^{n} [m^* f_i^*], \qquad [m^* f_i^*] \geq 1, \qquad C^* = \frac{m^* \sum\limits_{i=1}^{n} \sqrt{c_i \lambda_i (1+\rho_i)(1+\delta_i) s_i}}{\left(1 - \rho - \sum\limits_{i=1}^{n} \epsilon_i \rho_i\right) \sum\limits_{i=1}^{n} \sqrt{\frac{c_i \lambda_i (1+\rho_i)}{(1+\delta_i) s_i}}} \approx C_0,$$

with C_0 some desirable value for the cycle time.
Take $m_i^* = [m^* f_i^*]$, $i = 1, \ldots, n$.
2. Determination of the visit order.
Construct a polling table P^*, using e.g. one of the methods described in the appendices.
Calculate

$$D_i^* = \rho_i (1+\epsilon_i) \sqrt{\frac{(1+\delta_i) s_i}{c_i \lambda_i (1+\rho_i)}} \frac{\sum\limits_{j=1}^{n} \sqrt{c_j \lambda_j (1+\rho_j)(1+\delta_j) s_j}}{1 - \rho - \sum\limits_{j=1}^{n} \epsilon_j \rho_j} + (1+\delta_i) s_i, \qquad i = 1, \ldots, n,$$

as indication for the extended visit times at Q_i.
3. Determination of the extended visit times.
Solve the set of linear equations

$$T_k^* = \rho_{P_k^*} (1 + \epsilon_{P_k^*}) \sum_{l=1}^{m^*} h_{kl}^* T_l^* + (1 + \delta_{P_k^*}) s_{P_k^*}, \qquad k = 1, \ldots, m^*. \tag{4.18}$$

5 Numerical results

5.1 Introduction.

In the previous section a method for constructing an ftp scheme has been developed. In this section the method will be tested by conducting simulation experiments.

Throughout the complete section it is assumed that c_i equals 1, i.e., the mean waiting time is minimized. Moreover, the service times and the switch-over times are assumed to be constant, but possibly varying from queue to queue. The interarrival times are assumed to be almost constant, but also possibly varying from queue to queue. For queue i, the interarrival time equals $1/\lambda_i + \mathbf{WN}_i$, where $\mathbf{WN}_i$ is normally distributed with mean 0 and standard deviation $1/(10\lambda_i)$. These assumptions reflect reality well, see section 1. We replaced the side-constraint (3.6) by $T_k \geq \rho_{P_k}(1+\epsilon_{P_k}) SC_k + (1+\delta_{P_k}) s_{P_k} + \beta_{P_k}$ for the reason mentioned above formula (3.6). Consequently, in comparison with section 3 and 4, s_i is everywhere replaced by $s_i + \beta_i$.

For each simulation experiment we need a network configuration and an ftp scheme. Due to the complexity of the problem the proposed ftp scheme can not be compared with the real optimum. Instead the proposed ftp scheme is compared with some other natural, but less sophisticated, ftp schemes. In addition it is investigated whether perturbation of the proposed ftp scheme leads to further improvements.

In section 5.2 we shall describe two realistic network configurations and the ftp schemes that

can be chosen for these two network configurations. Of course one of these ftp schemes will be the original ftp scheme constructed by the method described in section 4.
In section 5.3 we shall add some theoretical network configurations. The small sizes of these network configurations allow us to test also ftp schemes in the neighborhood of the original ftp scheme. Thus the efficiency of the original ftp scheme can be examined.

5.2 Realistic network configurations.

The first realistic network configuration (NC 1) resembles a part of the real situation in the telecommunications district of Rotterdam. The second realistic network configuration (NC 2) resembles a part of the real situation in the telecommunications district of The Hague. Both realistic network configurations contain the following data:
- the total number of switches (SwitchNr)

and for each switch:
- the number of subscribers (Subsc),
- the mean number of calls per subscriber per day (Calls),
- the effective transmission rate of the data-communication link between the switch and the mediation (TrRt),
- the size of a call record (CRSz),
- the size of a file (FlSz), and
- the switch-over time to the switch (SwOT).

Based on the data mentioned above, λ_i, β_i, s_i and ρ_i can be calculated. The two realistic network configurations are shown in appendix E and appendix F. For both realistic network configurations, each of the following three ftp schemes has been tested.
Ftp scheme 1 (FTP 1) is the original ftp scheme constructed by the method described in section 4. Firstly, values are calculated for all λ_i, β_i, s_i and ρ_i. Secondly, the visit frequencies are computed according to formula (4.16), where δ_i is set to 0 for all i, because the switch-over times are deterministic. Thirdly, the total number of visits m and the visit numbers m_i are calculated according to formula (4.17), where η_i is set to 0.2 for $i = 1, \ldots, n$. Subsequently, the polling table is constructed using the Golden Ratio procedure. Lastly, the set of linear equations (4.18) is solved for ϵ_i equal to 0.01 for $i = 1, \ldots, n$.
Ftp scheme 2 (FTP 2) is the same as FTP 1, except that m_i is set to 1, implying that m equals the total number of queues n. This ftp scheme is treated in the previous section as a special case.
Ftp scheme 3 (FTP 3) is the same as FTP 1, except that $m_i = M\rho_i(1+\epsilon_i)$ and all available visit times have the same length. In comparison with section 3, the k-th available visit time U_k is a constant U for all k. The extended visit times may vary due to the switch-dependent switch-over times: $D_i = U + s_i$. The constant U is determined from the following equations: $D_i \geq \rho_i(1+\epsilon_i)C/m_i + s_i + \beta_i$, $D_i = U + s_i$ and $C = \sum_{j=1}^{n} m_j D_j$. Some straightforward calculations show that $U \geq \frac{\sum_{j=1}^{n} \rho_j(1+\epsilon_j)s_j + \beta_i}{1 - \rho - \sum_{j=1}^{n} \epsilon_j \rho_j}$. In this case U is chosen equal to the maximum over all i of the right-hand side of this expression.
The mean waiting time (in seconds) for the two realistic network configurations and the ftp

schemes mentioned above is shown in Table 5.1.

	NC 1	NC 2
FTP 1	14921.8	8055.13
FTP 2	15602.0	8067.80
FTP 3	22173.1	15331.2

Table 5.1. Mean waiting time for NC 1 and NC 2.

For these two realistic network configurations it is clear that the original ftp scheme (FTP 1) is the most efficient one with FTP 2 as a close second best. Moreover, it is also obvious that FTP 3 is the worst: the mean waiting time increases with 49% respectively 90%.

5.3 Theoretical network configurations.
In order to test the original ftp scheme more thoroughly, six theoretical network configurations are added (NC 3 through NC 8). The first three theoretical network configurations consist of two queues. By means of the first, second and third network configuration, the effect of varying switch-over times, service times respectively arrival rates can be examined. The last three theoretical network configurations consist of four queues. By means of these network configurations, the effect of pairwise variations can be examined. The six theoretical network configurations are described below in terms of λ_i, β_i and s_i.

NC 3: $\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = \beta_2 = 0.5$; $s_1 = 0.05$; $s_2 = 0.45$.
NC 4: $\lambda_1 = \lambda_2 = 0.75$; $\beta_1 = 0.1; \beta_2 = 0.9$; $s_1 = s_2 = 0.25$.
NC 5: $\lambda_1 = 0.15$; $\lambda_2 = 1.35$; $\beta_1 = \beta_2 = 0.5$; $s_1 = s_2 = 0.25$.
NC 6: $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 0.375$; $\beta_1 = \beta_2 = 0.1$; $\beta_3 = \beta_4 = 0.9$; $s_1 = s_3 = 0.05$; $s_2 = s_4 = 0.45$.
NC 7: $\lambda_1 = \lambda_2 = 0.075$; $\lambda_3 = \lambda_4 = 0.675$; $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0.5$; $s_1 = s_3 = 0.05$; $s_2 = s_4 = 0.45$.
NC 8: $\lambda_1 = \lambda_2 = 0.075$; $\lambda_3 = \lambda_4 = 0.675$; $\beta_1 = \beta_3 = 0.1$; $\beta_2 = \beta_4 = 0.9$; $s_1 = s_2 = s_3 = s_4 = 0.25$.

Note that the average service time equals 0.5 seconds, the average switch-over time equals half of the average service time and ρ equals 0.75 for every theoretical network configuration. For the theoretical network configurations, each of the following ftp schemes has been tested. Ftp scheme 1 (FTP 1) is the original ftp scheme constructed by the method described in section 5.2, where δ_i is set to 0, η_i is set to 0.05 and ϵ_i is set to 0.01 for all i. Ftp scheme 2 (FTP 2) is the same as FTP 1, except that m_i is set to 1, implying that m equals the total number of queues n. Ftp scheme 3 (FTP 3) is the same as FTP 1, except that $m_i = M\rho_i(1+\epsilon_i)$ and all available visit times have the same length U. The constant U is determined as in section 5.2. Ftp scheme 4 (FTP 4) is the same as FTP 1, except that the polling table is constructed using not the Golden Ratio procedure but the procedure based on extremal splittings. In addition neighboring ftp schemes of FTP 1 are tested; in these ftp schemes the number of visits to one queue is incremented or decremented by 1. Obviously these changes are incorporated in the method before the polling table is constructed using the Golden Ratio procedure. By introducing these ftp schemes, a part of the neighborhood of FTP 1 can be examined. For NC 3 through NC 5 and NC 6 through NC 8 we examine respectively 4 and 8 ftp schemes in the neighborhood. These ftp schemes will be denoted as follows: FTP-ji where i denotes the queue whose number of visits is changed and j is either 'p' or 'm', where 'p' denotes 'plus' ($m_i := m_i + 1$) and 'm' denotes 'minus' ($m_i := m_i - 1$). In order to get an idea how accurate the numerical results are, the desired visit frequencies (f_i, according to formula (4.16)) and the realized visit frequencies ($r_i = m_i/m$) for FTP 1 are given in Table 5.2 for each theoretical network configuration.

	NC 3	NC 4	NC 5
f_1	0.56789	0.59224	0.21076
f_2	0.43210	0.40776	0.78924

	NC 3	NC 4	NC 5
r_1	0.56250	0.59375	0.21875
r_2	0.43750	0.40625	0.78125

	NC 6	NC 7	NC 8
f_1	0.42524	0.12888	0.14824
f_2	0.22207	0.09807	0.08415
f_3	0.19179	0.43901	0.45777
f_4	0.16091	0.33404	0.30984

	NC 6	NC 7	NC 8
r_1	0.43750	0.12500	0.15151
r_2	0.21875	0.09375	0.08080
r_3	0.18750	0.43750	0.45454
r_4	0.15625	0.34375	0.31313

Table 5.2. The desired and realized visit frequencies.

In Table 5.3 the mean waiting time (in seconds) for the network configurations NC 3 through NC 8 and the ftp schemes mentioned above is given. The 95% confidence interval for the mean waiting time approximately equals the value listed in the table +/- 0.5%.

	NC 3	NC 4	NC 5
FTP 1	4.16043	4.30180	3.65206
FTP 2	**4.01402**	**4.03009**	4.75128
FTP 3	4.23011	19.1517	4.73148
FTP 4	4.07187	4.29875	**3.58310**
FTP p1	4.18632	4.34283	3.62953
FTP m1	4.04351	4.25356	3.67783
FTP p2	4.04045	4.41053	3.64899
FTP m2	4.21479	4.38801	3.65655

	NC 6	NC 7	NC 8
FTP 1	7.03269	6.22546	6.43141
FTP 2	7.13407	7.85229	7.87610
FTP 3	41.2442	9.16545	32.8741
FTP 4	7.21295	**6.08788**	**6.29167**
FTP p1	7.08372	6.20071	6.44589
FTP m1	**6.98311**	6.28151	6.44138
FTP p2	7.01397	6.32260	6.44990
FTP m2	7.13087	6.49027	6.53595
FTP p3	7.10979	6.29503	6.45601
FTP m3	7.04688	6.19059	6.45285
FTP p4	7.15380	6.19944	6.39432
FTP m4	7.46083	6.28023	6.48928

Table 5.3. Mean waiting time for NC 3, NC 4, NC 5, NC 6, NC 7 and NC 8.

The number printed in boldface indicates the optimum. In all cases FTP 1, FTP 4, and the neighboring schemes of FTP 1 give very similar results; FTP 2 is on the average slightly worse and FTP 3 is generally bad. The low variability of the arrival, service, and switch-over processes appears to result in a relative insensitivity to the right choice of the visit numbers in the neighborhood of the desired visit numbers. Remember that a wrong choice for the visit numbers may still be compensated for in the determination of the visit times. In most cases FTP 4 performs slightly better than FTP 1. Apparently in most cases the procedure based on extremal splittings indeed yields a slightly better polling table than the Golden Ratio procedure.

The observation that in NC 3 and NC 4 FTP 2 performs slightly better than FTP 1 and FTP 4, may be explained as follows. As observed in section 4, the visits to the various queues should be spaced as evenly as possible. For FTP 2 the very nature of cyclic polling allows the visits to the various queues to be perfectly evenly spaced. For FTP 1 and FTP 4 the desired visit frequencies in NC 3 and NC 4, cf. Table 5.2, do not even allow the visits to be reasonably evenly spaced. In the derivation of the desired visit frequencies the visits to the queues were however assumed to be perfectly evenly spaced. The fact that nevertheless FTP 2 performs only slightly better than FTP 1 and FTP 4, actually supports the approach used. As Table 5.3 confirms, FTP 2 is likely to outperform FTP 1 and FTP 4 only when the number of queues is small and the difference in the desired visit numbers not too large.

6 Conclusion

In this paper we have examined the problem of deriving efficient operational rules for polling queues according to an ftp scheme. An approach has been presented to the problem of minimizing the mean total waiting cost per unit of time by constructing an efficient ftp scheme, under the side-constraint of most rarely exceeding the available visit times. By reformulating the side-constraint and by using a simple approximation of the mean waiting times in a polling system, the problem has been formulated as an NP-hard mathematical program. A heuristic method for solving this mathematical program has been presented. Thus we have developed a method for constructing an efficient ftp scheme.

The method has been tested by simulation experiments. It is shown that the proposed ftp schemes, FTP 1 and FTP 4, give good results compared with some other natural, but less sophisticated, ftp schemes, FTP 2 and FTP 3. In addition, by perturbation of FTP 1 neighboring ftp schemes have been constructed, partly to further test FTP 1, partly as a first step in improving the method itself.

References

[1] Boxma, O.J. (1991). Analysis and optimization of polling systems. In: *Queueing, Performance and Control of ATM*, eds. J.W. Cohen and C.D. Pack (North-Holland, Amsterdam), 173-183.

[2] Boxma, O.J., Levy, H., Weststrate, J.A. (1991). Efficient visit frequencies for polling tables: minimization of waiting cost. *Queueing Systems* **9**, 133-162.

[3] CCITT SG-IV, Principles for a Telecommunication Management Network, Recommendation M.30: AP IX-31-E, Blue Book version.

[4] Garey, M.R., Johnson, D.S. (1979). *Computers and Intractability: a Guide to the Theory of NP-Completeness* (Freeman, San Francisco).

[5] Hajek, B. (1985). Extremal splittings of point processes. *Math. Oper. Res.* **10**, 543-556.

[6] Harink, J.H.A., Cramer, P., Huitema, G.B. (1992). Optimization of polling call records from switches. PTT Research Report TI-RA-92-435.

[7] 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.

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

[9] Kruskal, J.B. (1969). Work-scheduling algorithms: a non-probabilistic queueing study (with possible applications to No. 1 ESS). *Bell System Techn. J.* **48**, 2963-2974.

[10] Panwar, S.S., Philips, T.K., Chen, M.-S. (1988). Golden Ratio scheduling for low delay flow control in computer networks. IBM Report RC 13642, Yorktown Heights (NY).

[11] Seneta, E. (1981). *Non-negative Matrices and Markov Chains* (Springer, New York, 2nd ed.).

[12] Takagi, H. (1990). Queueing analysis of polling models: an update. In: *Stochastic Analysis of Computer and Communication Systems*, ed. H. Takagi (North-Holland, Amsterdam), 267-318.

[13] Yechiali, U. (1991). Optimal dynamic control of polling systems. In: *Queueing, Performance and Control of ATM*, eds. J.W. Cohen and C.D. Pack (North-Holland, Amsterdam), 205-217.

APPENDICES

A PROOF OF LEMMA 3.1

Lemma 3.1
Problem (I) is NP-hard.

Proof
To prove that problem (I) is NP-hard, we need to show that the decision variant of problem (I) is NP-complete. A decision problem is said to be NP-complete if (i) it belongs to the class NP and (ii) every problem in the class NP is (polynomially) reducible to it, cf. Garey & Johnson [4]. For brevity let us refer to the decision variant of problem (I) as the decision problem TABLE. TABLE reads as follows: given parameters λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and an arbitrary number r, does problem (I) have a solution which is feasible and for which the value of the objective function is not larger than r?
Obviously TABLE belongs to the class NP. As the notion of reducibility is transitive, it in fact remains to be shown that *some* known NP-hard problem is (polynomially) reducible to TABLE. Here the problem PARTITION turns out to be an appropriate choice as known NP-hard problem. PARTITION reads as follows: given a set $A = \{a_1, \ldots, a_p\}$ of p integers, does A include a subset B, such that $\sum\limits_{a_i \in B} a_i = \sum\limits_{a_i \in A \setminus B} a_i = \frac{1}{2} \sum\limits_{i=1}^{p} a_i$?
We now prove that PARTITION is (polynomially) reducible to TABLE. Given an instance

$a_1, \ldots, a_p$ for PARTITION, construct an instance λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r for TABLE in the following manner.

$$\begin{aligned}
&n = p + 1; \\
&\beta_i = 0, \delta_i = 0, \epsilon_i = 0, \qquad i = 1, \ldots, p+1; \\
&s_i = a_i, c_i\lambda_i = a_i, \qquad i = 1, \ldots, p; \\
&s_{p+1} = 1, c_{p+1}\lambda_{p+1} = 4; \\
&r = \frac{1}{2}\left(\sum_{i=1}^{p} a_i + 2\right)^2 .
\end{aligned} \tag{A.1}$$

We now need to prove that $a_1, \ldots, a_p$ constitute a 'yes' instance for PARTITION iff λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r as defined in (A.1), constitute a 'yes' instance for TABLE.
We first show that $a_1, \ldots, a_p$ constitute a 'yes' instance for PARTITION iff there exists a feasible polling scheme (P, T), such that $m_i = 1$, $i = 1, \ldots, p$, $m_{p+1} = 2$, $SC_k = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$ for both k with $P_k = p + 1$.
$\{\Rightarrow\}$ The set $A = \{a_1, \ldots, a_p\}$ includes a subset B, such that $\sum_{a_i \in B} a_i = \sum_{a_i \in A\setminus B} a_i = \frac{1}{2}\sum_{i=1}^{p} a_i$.
Let us say $B = \{a_{i_1}, \ldots, a_{i_q}\}$, $A\setminus B = \{a_{i_{q+1}}, \ldots, a_{i_p}\}$. Take $P_1 = p + 1$, $P_{k+1} = i_k$ for $k = 1, \ldots, q$, $P_{q+2} = p + 1$, $P_{k+2} = i_k$ for $k = q + 1, \ldots, p$, $T_k = s_{P_k}$, $k = 1, \ldots, p + 2$. Then $SC_1 = \sum_{k=q+2}^{p+2} T_k = \sum_{a_i \in B} a_i + 1 = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$, $SC_{q+2} = \sum_{k=1}^{q+1} T_k = \sum_{a_i \in A\setminus B} a_i + 1 = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$.
$\{\Leftarrow\}$ Let us say $P_{k_1} = p + 1$, $P_{k_2} = p + 1$, $k_1 < k_2$. So $SC_{k_1} = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$, $SC_{k_2} = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$. Now, using (3.4), $C = SC_{k_1} + SC_{k_2} = \sum_{i=1}^{p} a_i + 2 = \sum_{k=1}^{p+2} s_{P_k}$, while, using (3.3), $C = \sum_{k=1}^{p+2} T_k$. Hence $T_k \geq s_{P_k}$, $k = 1, \ldots, p + 2$, implies $T_k = s_{P_k}$, $k = 1, \ldots, p + 2$. Take $B = \{a_{P_k} : k_1 < k < k_2\}$. Then $\sum_{a_i \in B} a_i = \sum_{k=k_1+1}^{k_2-1} T_k = SC_{k_2} - T_{k_1} = \frac{1}{2}\sum_{i=1}^{p} a_i$, $\sum_{a_i \in A\setminus B} a_i = \sum_{k=k_2+1}^{p+2} T_k + \sum_{k=1}^{k_1-1} T_k = SC_{k_1} - T_{k_2} = \frac{1}{2}\sum_{i=1}^{p} a_i$.
We now show that there exists a feasible polling scheme (P, T), such that $m_i = 1$, $i = 1, \ldots, p$, $m_{p+1} = 2$, $SC_k = \frac{1}{2}\sum_{i=1}^{p} a_i + 1$ for both k with $P_k = p + 1$, iff λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i = 1, \ldots, n$, and r as defined in (A.1), constitute a 'yes' instance for problem TABLE.
$\{\Rightarrow\}$ As before $C = \sum_{i=1}^{p} a_i + 2$. So for (P, T) the value of the objective function is

$$\frac{1}{2C}\sum_{i=1}^{n} c_i\lambda_i(1 + \rho_i) \sum_{\{k : P_k = i\}} SC_k^2 =$$

$$\frac{1}{2\left(\sum\limits_{i=1}^{p} a_i+2\right)}\left\{\sum_{i=1}^{p} a_i\left(\sum_{i=1}^{p} a_i+2\right)^2+8\left(\frac{1}{2}\sum_{i=1}^{p} a_i+1\right)^2\right\}=\frac{1}{2}\left(\sum_{i=1}^{p} a_i+2\right)^2=r.$$

{$\Leftarrow$} Problem (I) with λ_i, β_i, s_i, c_i, δ_i, ϵ_i, $i=1,\ldots,n$, and r as defined in (A.1) has a solution which is feasible and for which the value of the objective function is not larger than r. Using the Lagrangean multiplier technique and the argumentation preceding the formulation of problem (II), it is easily verified that the value of the objective function is not larger than $\frac{1}{2}\left(\sum\limits_{i=1}^{p} a_i+2\right)^2$ only for (P,T) with $m_i=R$, $i=1,\ldots,p$, $m_{p+1}=2R$, $SC_k=C/m_{P_k}$, $C=\sum\limits_{i=1}^{p} m_i a_i+m_{p+1}$, which for $R=1$ yields the result that we have in view.

□

B The Golden Ratio procedure

Calculate the numbers $g(k)=k\phi^{-1} \bmod 1$ with $\phi^{-1}=\frac{1}{2}(\sqrt{5}-1)\approx 0.618034$, $k=1,\ldots,m$.
Let the numbers $g(k)$ with $\sum\limits_{j=1}^{i-1} m_j+1\le k\le\sum\limits_{j=1}^{i} m_j$ correspond to the visits to Q_i, $i=1,\ldots,n$.
Put the numbers $g(k)$, $k=1,\ldots,m$, in increasing order.
Let the l-th smallest number correspond to the l-th position in P, $l=1,\ldots,m$.
Formally, $P_{\pi(k)}:=i$ for k with $\sum\limits_{j=1}^{i-1} m_j+1\le k\le\sum\limits_{j=1}^{i} m_j$, π representing the permutation such that $g(k)\le g(l)\Longleftrightarrow\pi(k)\le\pi(l)$, $k,l=1,\ldots,m$.

C A procedure based on extremal splittings

Before we give a detailed description, we first sketch the main motivation behind the procedure. Recall that we contemplate to construct a polling table P that approximately minimizes (3.9) for fixed m_i and fixed $T_k=D^*_{P_k}$.
On the one hand, as seen from the argumentation preceding the formulation of problem (II), if the visits to the various queues are perfectly evenly spaced, then the polling table is optimal. In fact substituting $SC_k=\sum\limits_{i=1}^{n} m_i D^*_i/m_{P_k}$ into (3.9) yields a lower bound for the value of (3.9) for the optimal table. On the other hand, if the visits to the various queues are perfectly evenly spaced, then the polling table obviously satisfies the following property: between any two consecutive visits to Q_i there is exactly one visit to every Q_j with $m_i=m_j$, $i\neq j$. For brevity let us refer to this property as property (E). The reverse statement does not hold. Even if the polling table satisfies property (E), then for arbitrary parameter choices the subcycle times may still be arbitrarily far from equal, and the value of (3.9) may still be arbitrarily far from minimal. Nevertheless, if the polling table satisfies property (E), then for reasonable parameter choices the visits are likely to be reasonably evenly spaced, and the polling table is likely to be reasonably good. We therefore use property (E) as the main guideline in constructing a polling table.

Let $M=\{m_i : i\in\{1,\ldots,n\}\}$ be the set of visit numbers that occur. Let $I^{(r)}=\{i\in\{1,\ldots,n\}: m_i=r\}$ be the set of the queues with common visit number r for $r\in M$.

Suppose that one has already constructed a subtable P of size $| P |$ for all the visits to the queues $i \in I^{(r_1)}, \ldots, I^{(r_q)}$ with common visit numbers $r_1, \ldots, r_q \in M$. Initially, $| P | = 0$, $\{r_1, \ldots, r_q\} = \emptyset$.

If $M \backslash \{r_1, \ldots, r_q\} \neq \emptyset$, then select a visit number r from $M \backslash \{r_1, \ldots, r_q\}$. Construct a subtable $Q^{(r)}$ of size $| Q^{(r)} | = r \times | I^{(r)} |$ for all the visits to the queues $i \in I^{(r)}$ by just concatenating r times an arbitrary sequence of the queues $i \in I^{(r)}$. Formally, with $i_1, \ldots, i_{|I^{(r)}|}$ an arbitrary sequence of the queues $i \in I^{(r)}$, $Q^{(r)}_{j+(k-1) \times |I^{(r)}|} := i_j$ for $j = 1, \ldots, | I^{(r)} |$, $k = 1, \ldots, r$. Obviously $Q^{(r)}$ satisfies property (E).

Construct subsequently a subtable $P^{(r)}$ of size $| P^{(r)} | = | P | + | Q^{(r)} |$, by inserting the visits from the subtable $Q^{(r)}$ in the subtable P in the following manner. Put the visits from $Q^{(r)}$ at positions in $P^{(r)}$ as evenly spaced as possible, i.e, put the visit at the k-th position in $Q^{(r)}$ at the $(k + d(k))$-th position in $P^{(r)}$, $k = 1, \ldots, | Q^{(r)} |$. Here $d(k) = \left[(k-1) \times \frac{| P |}{| Q^{(r)} |}\right]$, $k = 1, \ldots, | Q^{(r)} |$, with $[x]$ denoting the nearest integer to x. Put the visits from P at the remaining positions in $P^{(r)}$, i.e., put the visit at the $\chi(l + l_0)$-th position in P at the l-th position in $P^{(r)}$ that is not yet occupied by a visit from $Q^{(r)}$, $l = 1, \ldots, | P |$. Here $\chi(k) = ((k-1) \bmod | P |) + 1$. Choose l_0 from $\{1, \ldots, | P |\}$ such that the objective function (3.9) properly applied to $P^{(r)}$ is minimal. Formally, $P^{(r)}_{k+d(k)} := Q^{(r)}_k$, $k = 1, \ldots, | Q^{(r)} |$, $P^{(r)}_{k+l} := P_{\chi(l+l_0)}$, $k = 1, \ldots, | Q^{(r)} |$, $l = d(k) + 1, \ldots, d(k+1)$. Thus in $P^{(r)}$ the number of visits from P between the k-th and $(k+1)$-th visit from $Q^{(r)}$ equals $d(k+1) - d(k) = \left[k \times \frac{| P |}{| Q^{(r)} |}\right] - \left[(k-1) \times \frac{| P |}{| Q^{(r)} |}\right]$, $k = 1, \ldots, | Q^{(r)} |$. This distancing is closely related to extremal splittings of point processes, cf. Hajek [5]. Notice that the internal visit order from P and $Q^{(r)}$ is maintained. Hence, by induction, $P^{(r)}$ satisfies property (E). Repeat with P replaced by $P^{(r)}$. Finally $| P | = m$, $\{r_1, \ldots, r_q\} = M$.

D Proof of Lemma 4.1

Lemma 4.1

The set of linear equations

$$T_k = \rho_{P_k}(1 + \epsilon_{P_k}) \sum_{l=1}^{m} h_{kl} T_l + (1 + \delta_{P_k}) s_{P_k}, \qquad k = 1, \ldots, m, \tag{D.1}$$

has a unique solution; this solution is non-negative.

Proof

Define the matrix A by $A_{kl} = \rho_{P_k}(1 + \epsilon_{P_k}) h_{kl}$, $k, l = 1, \ldots, m$, and the vector b by $b_k = (1 + \delta_{P_k}) s_{P_k}$, $k = 1, \ldots, m$. Then the set of linear equations (D.1) may be rewritten as $(I - A)T = b$.

As A is a non-negative irreducible matrix, A has a real eigenvalue μ, which is strictly maximal in absolute value, cf. Seneta [11] p. 3-4. For μ holds $\min_{1 \leq l \leq m} \sum_{k=1}^{m} A_{kl} \leq \mu \leq \max_{1 \leq l \leq m} \sum_{k=1}^{m} A_{kl}$, cf. [11] p. 8. $\sum_{k=1}^{m} A_{kl} = \sum_{i=1}^{n} \sum_{\{k:P_k=i\}} \rho_{P_k}(1 + \epsilon_{P_k}) h_{kl} = \rho + \sum_{i=1}^{n} \epsilon_i \rho_i$, as $\sum_{\{k:P_k=i\}} h_{kl} = 1$, $l = 1, \ldots, m$, so $\mu = \rho + \sum_{i=1}^{n} \epsilon_i \rho_i$. As b is a non-negative vector, $\rho + \sum_{i=1}^{n} \epsilon_i \rho_i < 1$ implies that $(I - A)T = b$ has a unique solution $T = (I - A)^{-1} b \geq 0$, cf. [11] p. 30.

□

E Network configuration 1 (NC 1)

SwitchNr	Subsc	Calls	TrRt	CRSz	FlSz	SwOT
1	2816	3	6700	37	2703	300
2	3328	3	6700	68	4412	300
3	3968	3	6700	68	4412	300
4	4096	3	6700	68	4412	300
5	2384	3	6700	68	4412	300
6	6656	3	6700	68	4412	300
7	4736	3	6700	68	4412	300
8	7776	3	6700	37	2703	300
9	5120	3	6700	37	2703	300
10	5376	3	6700	68	4412	300
11	14336	3	6700	68	4412	300
12	7296	3	6700	37	2703	300
13	7296	3	6700	37	2703	300
14	13312	3	6700	37	2703	300
15	8072	3	6700	68	4412	300
16	12288	3	6700	68	4412	300
17	14336	3	6700	68	4412	300
18	9216	3	6700	68	4412	300
19	9728	3	6700	68	4412	300
20	4096	3	6700	68	4412	300
21	20019	3	6700	68	4412	300
22	24267	3	6700	68	4412	300
23	20019	3	6700	68	4412	300

F Network configuration 2 (NC 2).

SwitchNr	Subsc	Calls	TrRt	CRSz	FlSz	SwOT
1	19244	3	45000	34	5882	300
2	24616	3	45000	34	14706	300
3	7680	3	45000	34	5882	300
4	26481	3	6700	37	2703	300
5	18688	3	45000	34	5882	300
6	21760	3	45000	34	5882	300
7	78035	3	45000	34	14706	300
8	15360	3	45000	34	5882	300
9	16000	3	45000	34	5882	300
10	16000	3	45000	34	5882	300
11	14848	3	45000	34	5882	300
12	97824	3	45000	34	14706	300
13	26432	3	45000	34	5882	300
14	17664	3	45000	34	5882	300
15	17664	3	45000	34	5882	300
16	9216	3	45000	34	5882	300
17	10240	3	6700	37	2703	300
18	15616	3	45000	34	5882	300
19	13056	3	45000	34	5882	300
20	9417	3	45000	34	5882	300
21	12544	3	45000	34	5882	300
22	43682	3	45000	34	14706	300
23	17867	3	45000	34	5882	300
24	7680	3	45000	34	5882	300
25	2048	3	45000	34	5882	300
26	14592	3	6700	37	2703	300
27	3072	3	6700	37	2703	300
28	9728	3	45000	34	5882	300
29	7168	3	6700	37	2703	300
30	14000	3	6700	37	2703	300
31	18432	3	45000	34	5882	300
32	18686	3	45000	34	5882	300
33	9984	3	6700	37	2703	300