IIE Transactions (2015) **47**, 414–430

ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/0740817X.2014.928965

Adaptive threshold policies for multi-channel call centers

BENJAMIN LEGROS[1], OUALID JOUINI[1,*] and GER KOOLE[2]

[1]*Laboratoire Génie Industriel, Ecole Centrale Paris, Grande Voie des Vignes, 92290 Châtenay-Malabry, France*
[2]*Department of Mathematics, VU University Amsterdam, 1081 HV Amsterdam, The Netherlands,*
E-mail: oualid.jouini@ecp.fr

Received September 2013 and accepted May 2014

In the context of multi-channel call centers with inbound calls and emails, this article considers a threshold policy on the reservation of agents for the inbound calls. We study a general non-stationary model where calls arrive according to a non-homogeneous Poisson process. The optimization problem consists in maximizing the throughput of emails under a constraint on the waiting time of inbound calls. An efficient adaptive threshold policy is proposed that is easy to implement in the automatic call distributor. This scheduling policy is evaluated through a comparison with the optimal performance measures found in the case of a constant arrival rate and also with other intuitive adaptive threshold policies in the general non-stationary case.

Keywords: Queueing systems, service operations, multi-channel call centers, threshold policies

1. Introduction

Call centers require a very precise match of demand and supply. A delay in the answering of a call, its waiting time, is usually not allowed to exceed 20 seconds (Koole, 2013). Thus, a very accurate prediction of the demand is required. However, this can rarely be obtained, because of the volatility of call arrival patterns. Therefore, there is often a mismatch between demand and the scheduled supply, consisting of rostered call center employees (usually called agents). Moreover, even if the demand is accurately forecasted, a considerable overcapacity should be scheduled to be able to deal with the random Poisson fluctuations of the demand. Usually queueing models are used to quantify this overcapacity, most often Erlang C.

To prevent idle overcapacity, and to limit the necessity to have extremely accurate forecasts, inbound calls are sometimes mixed with other types of customer contacts that have less-strict delay requirements, such as emails or outbound calls. This is called *(call) blending*. The amount of capacity assigned to the other channels is supposed to adapt to the number of inbound calls, giving at the same time a good service level for the inbound calls and a good occupancy of the call center agents.

Due to the strict waiting time requirements on inbound calls it is best to give them priority over the other channels. To maximize agent productivity it would be optimal to assign an outbound job to every idle agent when there are no inbound calls in the queue. This would lead to 100% productivity. However, this policy leads to long waiting times for inbound calls because an agent is never waiting for an inbound call to arrive. Consequently, the service-level constraint on inbound calls might be violated. For instance, consider a simple Markovian queueing model with inbound jobs arriving at rate λ arriving at a queue with infinite capacity, an infinite amount of emails, s polyvalent identical agents, identical service rate μ for both job types. Using the underlying birth–death process, one may easily deduce that the expected waiting time for inbound jobs is $\frac{\rho}{\lambda(1-\rho)}$, where $\rho = \frac{\lambda}{s\mu}$, and the probability of delay of inbound jobs is 1. Therefore, for a given staffing level, this work-conserving policy does not have enough flexibility to reach some predefined service level for inbound jobs. Somehow we should reserve capacity for inbound jobs to obtain an expected waiting time strictly lower than $\frac{\rho}{\lambda(1-\rho)}$ and a probability of delay strictly lower than 1, which allows us to obtain better call service levels. Thus, a more sophisticated assignment policy, other than the work-conserving one, is required. A service level measures a call waiting time performance (for example, the proportion of calls that are answered within a predefined time threshold, or the expected waiting time).

In Bhulai and Koole (2003) and Gans and Zhou (2003) it is shown that an efficient assignment policy has the following form: outbound jobs should only be scheduled when there are no waiting inbound calls and when the number of idle agents exceeds a certain threshold. Thus, the problem of controlling our blended call center reduces to determining the right threshold level. This threshold, however, depends on all of the system parameters, which are

*Corresponding author

the inbound call arrival rate, the inbound call service rate, the email service rate, and the number of agents. However, these parameters, especially the arrival rate, are often hard to determine. This calls for a policy in which the threshold is adapted to the current situation without explicitly using the parameters of the system. In this article, such adaptive policies are studied, both for systems with a constant (but unknown) arrival rate and for the more realistic situation of a fluctuating arrival rate. The parameter that is used to update the threshold is the service level up to that moment, a number that is always available in call centers; We consider the service level that measures the proportion of calls that are answered within a predefined time threshold. The overall objective is to reach a certain service level by the end of the day, while maximizing the number of emails that are done.

We now discuss the relevant literature. There is a rich literature on planning and scheduling in call centers; see Gans *et al.* (2003) and Akşin *et al.* (2007). However, few papers focus on blending. The general context of multi-channel call centers is described in Koole (2013, Chapter 7).

Deslauriers *et al.* (2007) extend the earlier mentioned papers by having different types of agents. Outbound jobs are served only by multi-channel (blended) agents, whereas inbound calls can be served by either inbound-only or blended agents. Inbound callers may balk or abandon. They evaluate several performance measures of interest, including the rate of outbound jobs and the proportion of inbound calls waiting more than a fixed number of seconds. A collection of Continuous-Time Markov Chain (CTMC) models that capture many real-world characteristics while maintaining parsimony that results in fast computation are presented. They discuss and explore the tradeoffs between model fidelity and efficacy and compare different CTMC models with a realistic simulation model of a Bell Canada call center.

Armony and Ward (2010) present an optimization problem: the objective is to minimize the steady-state expected customer waiting time subject to a fairness constraint on the workload division. They show that in such a problem, which is close to ours, a threshold policy outperforms a common routing policy used in call centers (that routes to the agent that has been idle the longest).

Milner and Olsen (2008) consider a call center with contract and non-contract customers. They explore the common use to give priority to contract customers only in off-peak time periods. They show that this choice is a good one under the classical assumptions (such as stationarity). They also present examples when this is not the case. This result is important since we found an insight arguing that the service level for inbound calls has to be very strictly respected during off-peak periods.

This article is organized as follows. Section 2 presents our model. Sections 3 and 4 contain our results, first for a constant arrival rate in Section 3 and then in Section 4 with a fluctuating arrival rate. We end with a short conclusion.

2. Model

We consider a call center modeled as a multi-server queueing system with two types of jobs, foreground jobs (inbound calls) and background jobs (emails). The arrival process of calls is assumed to be a non-homogeneous Poisson process with rate $\lambda(t)$, for $t \geq 0$. Calls arrive at a dedicated First-Come, First-Served (FCFS) queue with infinite capacity. There is an infinite supply of background jobs, waiting for treatment in a dedicated FCFS queue. There are s identical, parallel servers (agents in call center parlance). Each agent can handle both types of jobs. We assume that the service times of foreground and background jobs are exponentially distributed with rates μ and μ_0, respectively. Neither abandonment nor retrials are modeled.

Foreground jobs are more important than background ones in the sense that the former request a quasi-instantaneous answer (waiting time in the order of seconds or minutes), wheres the latter are more flexible and can be delayed for several (tens of) hours. The objective of the call center manager over a working day is to maximize the email throughput while satisfying a constraint on the call waiting time in the queue.

Since the model is transient, we cannot define the waiting time of an arbitrary customer as a unique random variable. There is a random number of served customers during the working period, say Q. If $Q > 0$, we denote by W_n the random variable for the waiting time of customer n, for $n \in \{1, \ldots, Q\}$. We want the expected proportion of calls that wait less than a predefined threshold τ to be at least equals to α; i.e., $E(Q^{-1} \sum_{n=1}^{Q} \mathbf{1}_{W_n \leq \tau}) \geq \alpha$, for $\tau \geq 0$ and $0 \leq \alpha \leq 1$. Note that we do not consider arriving customers at the end of the working period that cannot be served.

We then aim to find the best routing rules in terms of efficiency for the considered problem and ease of implementation in call center software. We assume that preemption of jobs in service is not allowed. This is a quite natural assumption. An agent usually prefers to finish answering an underway outbound job rather than starting it again later on. This is also preferred from an efficiency perspective. Evidently, when the background jobs are outbound calls, then it is not acceptable to preempt.

For a similar model, but with a constant arrival rate and equal service requirements for the two job types, Bhulai and Koole (2003) prove that the optimal policy is a threshold policy with the priority given to calls (some servers reserved for calls). Their result is mainly based on the fact that it is optimal to handle calls as long as the queue of calls is not empty. For our general modeling, the analysis is more complicated. Even for a constant arrival rate but different service requirements, the optimal policy is hard to

obtain and might not be useful in practice (for software implementation, for example). For simplicity and usefulness of the results in practice, we restrict ourselves to the case of threshold policies. Moreover, Bhulai and Koole (2003) numerically show, for more general cases, that the appealing threshold policies are good approximations of the optimal ones. More concretely, the functioning of the call center under a threshold policy is as follows. Let us denote the threshold by u, $0 \leq u \leq s$. Upon arrival, a call is immediately handled by an available agent, if any. If not, the call waits in the queue. When an agent becomes idle, she handles the call at the head of the queue with calls, if any. If not, the agent may either handle an email or she remains idle. If the number of idle agents (excluding her) is at least $s-u$, then the agent in question handles an email. Otherwise, she remains idle. In other words, there are $s-u$ agents that are reserved for calls, so there are at least u agents working at any time.

In this article, we propose an adaptive threshold policy that adjusts the threshold as a function of the process of the call service level. We divide the working day into N identical intervals, each with length θ. The total working duration in a day is D, $D = N\theta$. At the beginning of each interval i ($i = 1, ..., N$), we define the threshold u_i, $0 \leq u_i \leq s$, under which the job routing policy works during interval i. Let T denote the expected throughput of emails over the whole day; i.e., the ratio between the number of treated emails and D. Let also SL be the proportion, for the whole day, of calls that have waited less than τ, $SL = E(Q^{-1}\sum_{n=1}^{Q}\mathbf{1}_{W_n\leq\tau})$, where Q is the random variable measuring the number of served customers during the whole day. In summary, our optimization problem can be formulated as

$$\begin{cases} \text{Maximize} \quad T \\ \text{subject to} \quad SL \geq \alpha, \end{cases} \tag{1}$$

where the decision variables are u_i with $0 \leq u_i \leq s$, for $i = 1, \ldots, N$. It is clear that the best case for calls is such that $u_i = 0$ for all i, which means that no emails are answered and SL is maximized (case of an $M(t)/M/s$ with only calls). We therefore assume from now on that the parameters $\lambda(t)$ for $t \geq 0$, μ and s are such that $SL \geq \alpha$ for $u_i = 0$ ($i = 1, \ldots, N$).

3. Constant arrival rate

We consider a basic case with a constant arrival rate, $\lambda(t) = \lambda$ for $t \geq 0$ and a constant threshold, $u_i = u$ for $i = 1, \ldots, N$ and $0 \leq u \leq s$. The purpose of the analysis in this section is to understand the behavior of the performance measures as a function of the threshold in order to build an efficient method for the threshold adaptation rule (u_i for $i = 1, \ldots, N$) in the case of a non-constant arrival rate. In Section 3.1 we propose a method to compute the performance measures, then in Section 3.2 we use them to provide a useful insight to construct our adaptive policy.

3.1. *Performance measures*

In Section 3.1.1 we provide closed-form formula of the performance measures in the case of equal service rates and study the form of these measures as a function of the threshold. Then in Section 3.1.2 we propose a numerical method to compute the performance measures in the case of unequal service rates. Since we consider a stationary model we can define a unique random variable for the waiting time of an arbitrary customer W and denote by $P(W < \tau)$ the probability that an arbitrary customer waits less than τ ($\tau > 0$).

3.1.1. *Equal service rates*

We consider the case $\mu = \mu_0$. First, we compute the performance measures of interest for calls and emails for a given constant reservation threshold, denoted by u, $0 \leq u \leq s$. We then develop some structural results that will be used in Section 3.2.

Let us define the stochastic process $\{x(t), t \geq 0\}$, where $x(t) \in \{u, u+1, u+2, \cdots\}$ is the number of jobs in service plus the number of jobs in the queue of calls. Since $\mu = \mu_0$, we need not distinguish between the two job types in service. The process $\{x(t), t \geq 0\}$ is a birth–death process. It is similar to that of an $M/M/s$ queue without the states $\{0, 1, \cdots, u-1\}$. The transition rate from state x to state $x-1$ is $\min\{x, s\}\mu$, for $x > u$, and that from state x to state $x+1$ is λ, for $x \geq u$. We denote by a the ratio $\frac{\lambda}{\mu}$. Also, under the stability condition $\frac{\lambda}{s\mu} < 1$, we denote by p_x the steady-state probability to be in state $x \in \mathbb{N}$. In Theorem 1, we give the expression of the email throughput, $T(s, u, a)$, and that of the probability that the call waiting time is less than τ, $SL = P(W < \tau)$.

Theorem 1. *For $0 \leq u \leq s$, we have*

$$T(s,u,a) = \mu\left(\sum_{k=0}^{s-u}\frac{a^k u!}{(u+k)!} + \frac{a^{s-u}u!}{s!}\frac{a}{s-a}\right)^{-1} \times \left(u + \sum_{k=1}^{s-u}\frac{a^k u!}{(u+k-1)!} + \frac{a^{s-u+1}u!}{(s-1)!(s-a)}\right) - \lambda, \tag{2}$$

$$P(W < \tau) = 1 - C(s,u,a)e^{-\tau(s\mu-\lambda)}, \tag{3}$$

with

$$C(s,u,a) = \frac{a^{s-u}u!}{s!(1-a/s)}\left(\sum_{k=0}^{s-u}\frac{a^k u!}{(u+k)!} + \frac{a^{s-u}u!}{s!}\frac{a}{s-a}\right)^{-1}. \tag{4}$$

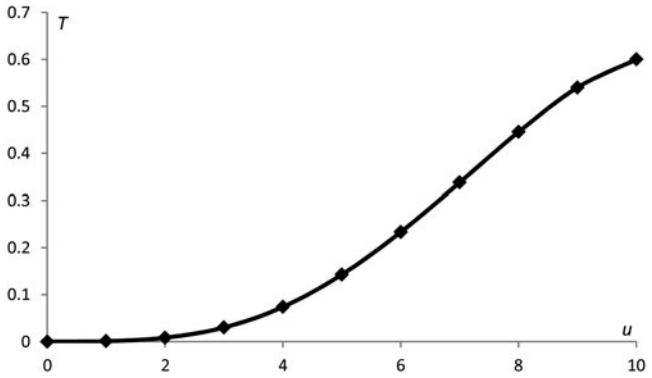

Fig. 1. E-mail throughput ($s = 10$, $\mu_0 = \mu = 0.2$, $\lambda = 1.4$).

Proof. For $0 \le x < u$, we have $p_x = 0$. For $0 \le k \le s-u$, we have $p_{u+k} = \frac{a^k u!}{(u+k)!} p_u$. For $k \ge 0$, we have $p_{s+k} = \frac{a^k}{s^k} p_s$. Since all probabilities sum up to one, we obtain

$$p_u = \left(\sum_{k=0}^{s-u} \frac{a^k u!}{(u+k)!} + \frac{a^{s-u} u!}{s!} \frac{a}{s-a} \right)^{-1}. \tag{5}$$

The email throughput can be seen as the overall throughput (of calls and emails) minus the call throughput. Thus,

$$T(s,u,a) = \sum_{k=0}^{s-u} (u+k)\mu p_{u+k} + s\mu \sum_{k=1}^{\infty} p_{s+k} - \lambda.$$

After some algebra, we deduce that

$$T(s,u,a) = \mu p_u \left(u + \sum_{k=1}^{s-u} \frac{a^k u!}{(u+k-1)!} + \frac{a^{s-u+1} u!}{(s-1)!(s-a)} \right) - \lambda.$$

Note that the lower bound of $T(s,u,a)$ is $T(s,0,a) = 0$, which corresponds to the case when all servers are reserved for calls. As for the upper bound, it is $T(s,s,a) = s\mu - \lambda$, which corresponds to the case of no server reservation for calls (the infinite amount of emails leads to $s\mu$ as a total throughput for the two job types).

The call service level, $P(W > \tau)$, is obtained using the PASTA property. We have $P(W > \tau) = \sum_{n=0}^{\infty} p_{s+n} P(W > \tau | x = n+s)$, where $P(W > \tau | x = s+n)$ is the conditional probability that the waiting time of a new call exceeds τ, given that it finds all servers busy and n calls waiting ahead in the queue, $n \ge 0$. It is easy to see that this conditional waiting time follows an Erlang distribution with $n+1$ stages and a rate of $s\mu$ per stage. Then, $P(W > \tau | x = s+n) = \sum_{k=0}^{n} e^{-s\mu\tau} \frac{(s\mu\tau)^k}{k!}$, which leads to

$$\begin{aligned} P(W > \tau) &= \sum_{n=0}^{\infty} p_s \frac{a^n}{s^n} \sum_{k=0}^{n} e^{-s\mu\tau} \frac{(s\mu\tau)^k}{k!} \\ &= \lim_{n\to\infty} \left(p_s e^{-s\mu\tau} \sum_{k=0}^{n} \sum_{n=k}^{\infty} \frac{(s\mu\tau)^k}{k!} \left(\frac{a}{s}\right)^n \right). \end{aligned}$$

Observing that $\sum_{n=k}^{\infty} \left(\frac{a}{s}\right)^n = \left(\frac{a}{s}\right)^k \frac{1}{1-a/s}$ implies

$$P(W > \tau) = C(s,u,a) e^{-\tau(s\mu - \lambda)}, \tag{6}$$

with

$$\begin{aligned} C(s,u,a) &= \frac{p_s}{1-a/s} = \frac{a^{s-u} u!}{s!(1-a/s)} \\ &\quad \times \left(\sum_{k=0}^{s-u} \frac{a^k u!}{(u+k)!} + \frac{a^{s-u} u!}{s!} \frac{a}{s-a} \right)^{-1}. \end{aligned} \tag{7}$$

Note that the upper bound of $C(s,u,a)$ is $C(s,s,a) = 1$ (no server reservation for calls, then, any arriving call has to wait for service), and its lower bound is $C(s,0,a) = \frac{a^s}{s!(1-a/s)} \left(\sum_{k=0}^{s} \frac{a^k}{k!} + \frac{a^{s+1}}{s!(s-a)} \right)^{-1}$; that is, all servers are reserved for calls, which corresponds for calls to a standard $M/M/s$ queue with no emails. This completed the proof of the theorem. ■

In Proposition 1, we prove monotonicity results of the system performance measures as a function of the threshold.

Proposition 1. *For $a > 0$, the following holds:*

1. *The email throughput T is strictly increasing and neither convex nor concave in u, for $0 \le u \le s$. However, the end of the email throughput, for $0 \le s-2 \le u \le s$, is concave in u.*
2. *The call service level $P(W < \tau)$ is strictly decreasing and concave in u, for $0 \le u \le s$.*

Proof. Let us prove the first statement. From Equation (2), we have

$$T(s,u,a) = \mu \frac{\frac{1}{(u-1)!} + \frac{a}{u!} + \frac{a^2}{(u+1)!} + \cdots + \frac{a^{s-u}}{(s-1)!} + \frac{a^{s-u+1}}{(s-1)!(s-a)}}{\frac{1}{u!} + \frac{a}{(u+1)!} + \frac{a^2}{(u+2)!} + \cdots + \frac{a^{s-u-1}}{(s-1)!} + \frac{a^{s-u}}{(s-1)!(s-a)}} - \lambda,$$

for $0 \le u \le s$. Thus $T(s,u,a) = \mu(a + \frac{1}{g_u}) - \lambda$, with

$$g_u = \frac{1}{u} + \frac{a}{u(u+1)} + \cdots + \frac{a^{s-u-1}}{u(u+1)(u+2)\cdots(s-1)} + \frac{a^{s-u}}{u(u+1)(u+2)\cdots(s-1)(s-a)},$$

for $0 < u \leq s$ (and $T(s, 0, a) = 0$). We may write for $0 < u < s$:

$$\begin{aligned} g_{u+1} - g_u = &\left(\frac{1}{u+1} - \frac{1}{u}\right) \\ &+ \left(\frac{a}{(u+1)(u+2)} - \frac{a}{u(u+1)}\right) + \cdots \\ &+ \left(\frac{a^{s-u-1}}{(u+1)(u+2)\cdots(s-1)(s-a)} \right. \\ &\left. - \frac{a^{s-u-1}}{u(u+1)\cdots(s-1)}\right) \\ &+ \left(-\frac{a^{s-u}}{u(u+1)\cdots(s-1)(s-a)}\right). \end{aligned} \quad (8)$$

Since each term of the summation in the right-hand side of Equation (8) is strictly negative, $g_{u+1} < g_u$ for $0 < u < s$. Then, g_u is strictly decreasing in u for $0 < u \leq s$. We also have $T(s, 1, a) > 0 = T(s, 0, a)$. This implies that $T(s, u, a)$ is strictly increasing in u, for $0 \leq u \leq s$. Figure 1 illustrates that in general the throughput is neither convex nor concave. Let us now prove that the end of the email throughput, for $s - 2 \leq u \leq s$ and $a > 0$, is concave in u. For $s \geq 2$, we have $T(s, s, a) = s\mu - \lambda$, $T(s, s-1, a) = \frac{\mu}{s}(s^2 - s + a) - \lambda$, and $T(s, s-2, a) = \frac{\mu}{s^2-s+a}(s^3 - 3s^2 + 2(a+1)s - 2a + a^2) - \lambda$. This implies $T(s, s-1, a) - T(s, s-2, a) = \frac{\mu(s-1)}{s(s^2-s+a)}(s^2 - a^2)$, and $T(s, s, a) - T(s, s-1, a) = \frac{\mu}{s}(s - a)$, which gives $T(s, s-1, a) - T(s, s-2, a) = (T(s, s, a) - T(s, s-1, a))\frac{(s-1)(s+a)}{s^2-s+a}$. Since for $s \geq 2$ and $a > 0$, $(s-1)(s+a) - (s^2 - s + a) = a(s-2) \geq 0$, we may write $T(s, s-1, a) - T(s, s-2, a) \geq T(s, s, a) - T(s, s-1, a)$. Then the end of the throughput is concave, which finishes the proof of the first statement of the proposition.

In what follows, we prove the second statement of the proposition. Let us define the sequence f_u as $f_u = s!(1 - a/s)C(s, u, a)$, for $0 \leq u \leq s$. Using Equation (3), it suffices then to prove that f_u is strictly increasing and convex in u. We start by proving that f_u is strictly increasing in u. We have

$$f_u = \left(\frac{a}{s!(s-a)} + \sum_{k=0}^{s-u} \frac{a^{k+u-s}}{(u+k)!}\right)^{-1},$$

for $0 \leq u \leq s$. Since for $0 \leq u < s$ we have $\sum_{k=0}^{s-u} \frac{a^{k+u-s}}{(u+k)!} = a^{u-s}(\sum_{k=0}^{s-u} \frac{a^k}{(u+k)!})$ and $\sum_{k=0}^{s-(u+1)} \frac{a^{k+u+1-s}}{(u+1+k)!} = a^{u-s}(\sum_{k=1}^{s-u} \frac{a^k}{(u+k)!})$, we deduce that $f_{u+1}^{-1} - f_u^{-1} = -\frac{a^{u-s}}{u!} < 0$. This implies that $f_u < f_{u+1}$, for $0 \leq u < s$. Then, $P(W < \tau)$ is strictly decreasing in u, for $0 \leq u \leq s$.

We next focus on the proof of convexity of f_u in u (for $s \geq 2$). We do so by proving that $f_u + f_{u+2} - 2f_{u+1} > 0$, for $0 \leq u \leq s - 2$. Since $f_u + f_{u+2} - 2f_{u+1} = f_u f_{u+1} f_{u+2}(f_{u+2}^{-1} f_{u+1}^{-1} + f_{u+1}^{-1} f_u^{-1} - 2f_{u+2}^{-1} f_u^{-1})$, it suffices to prove that $f_{u+2}^{-1} f_{u+1}^{-1} + f_{u+1}^{-1} f_u^{-1} - 2f_{u+2}^{-1} f_u^{-1} > 0$, for $0 \leq u \leq s - 2$. Observing that $f_{u+1}^{-1} = f_u^{-1} - \frac{a^{u-s}}{u!}$ and $f_{u+2}^{-1} = f_u^{-1} - \frac{a^{u-s}}{u!} - \frac{a^{u+1-s}}{(u+1)!}$, we obtain

$$\begin{aligned} &f_{u+2}^{-1} f_{u+1}^{-1} + f_{u+1}^{-1} f_u^{-1} - 2f_{u+2}^{-1} f_u^{-1} \\ &= \left(f_u^{-1} - \frac{a^{u-s}}{u!} - \frac{a^{u+1-s}}{(u+1)!}\right)\left(f_u^{-1} - \frac{a^{u-s}}{u!}\right) \\ &\quad + \left(f_u^{-1} - \frac{a^{u-s}}{u!}\right) f_u^{-1} - 2\left(f_u^{-1} - \frac{a^{u-s}}{u!} - \frac{a^{u+1-s}}{(u+1)!}\right) f_u^{-1} \\ &= \frac{a^{u-s}}{u!}\left(f_u^{-1}\left(-1 + \frac{a}{u+1}\right) + \frac{a^{u-s}}{u!}\left(1 + \frac{a}{u+1}\right)\right), \end{aligned}$$

for $0 \leq u \leq s - 2$.

Since $\frac{a^{u-s}}{u!} > 0$, we thus need to prove that

$$f_u^{-1}\left(-1 + \frac{a}{u+1}\right) + \frac{a^{u-s}}{u!}\left(1 + \frac{a}{u+1}\right) > 0,$$

for $0 \leq u \leq s - 2$. Using Equation (5), we may write $f_u^{-1} = \frac{a^{u-s}}{u!} p_u^{-1}$, for $0 \leq u \leq s - 2$. Therefore,

$$\begin{aligned} &f_u^{-1}\left(-1 + \frac{a}{u+1}\right) + \frac{a^{u-s}}{u!}\left(1 + \frac{a}{u+1}\right) \\ &= \frac{a^{u-s}}{u!}\left(p_u^{-1}\left(-1 + \frac{a}{u+1}\right) + 1 + \frac{a}{u+1}\right), \end{aligned}$$

for $0 \leq u \leq s - 2$. It remains then to prove that

$$p_u^{-1}\left(-1 + \frac{a}{u+1}\right) + 1 + \frac{a}{u+1} > 0,$$

for $0 \leq u \leq s - 2$.

For $0 \leq u \leq s - 2$ and $0 < a < s$, the derivative in a of $p_u^{-1}(.)$ is given by

$$\begin{aligned} \frac{\partial p_u^{-1}}{\partial a} &= \sum_{k=0}^{s-u} \frac{k a^{k-1} u!}{(u+k)!} + \frac{u!}{s!} \frac{(s-u+1)a^{s-u}(s-a) + a^{s-u+1}}{(s-a)^2} \\ &= \sum_{k=0}^{s-u} \frac{k a^{k-1} u!}{(u+k)!} + \frac{u!}{s!} \frac{a^{s-u}\left((s-u+1)(s-a) + a\right)}{(s-a)^2}. \end{aligned}$$

Since $0 < a < s$ and $u < s$, we obtain $\frac{\partial p_u^{-1}}{\partial a} > 0$. Thus, p_u^{-1} is strictly increasing in a and, as a consequence, the expression $p_u^{-1}(-1 + \frac{a}{u+1}) + 1 + \frac{a}{u+1}$ is strictly increasing in a, for $0 \leq u \leq s - 2$ and $0 < a < s$. Moreover, we have

$$\begin{aligned} \lim_{a \to 0, a > 0} p_u^{-1} = &\lim_{a \to 0, a > 0} \left(\sum_{k=0}^{s-u} \frac{a^k u!}{(u+k)!} \right. \\ &\left. + \frac{a^{s-u} u!}{s!} \frac{a}{s-a}\right). \end{aligned}$$

Since

$$\lim_{a \to 0, a > 0} \frac{a^{s-u} u!}{s!} \frac{a}{s-a} = 0, \quad \text{and}$$

$$\lim_{a \to 0, a > 0} \sum_{k=0}^{s-u} \frac{a^k u!}{(u+k)!} = 1,$$

we get

$$\lim_{a \to 0,\, a > 0} p_u^{-1} = 1.$$

This implies

$$\lim_{a \to 0,\, a > 0} p_u^{-1} \left(-1 + \frac{a}{u+1}\right) + 1 + \frac{a}{u+1} = 0,$$

$0 \le u \le s-2$. Using now the fact that $p_u^{-1}(-1+\frac{a}{u+1}) + 1 + \frac{a}{u+1}$ is strictly increasing in a, we deduce that $p_u^{-1}(-1+\frac{a}{u+1}) + 1 + \frac{a}{u+1} > 0$, for $a > 0$ and $0 \le u \le s-2$. This completes the proof of the proposition. ■

Note that in the particular remaining case $a = 0$ (i.e., no calls, $\lambda = 0$), the email throughput T is increasing and linear in u. We have $T = u\mu$, for $0 \le u \le s$.

3.1.2. *Unequal service rates*

In this section we focus on the performance evaluation (email throughput and call waiting time distribution) for the case of unequal service rates, $\mu \neq \mu_0$. In contrast with the case of equal service rates, the performance expressions are here too cumbersome to allow the development of useful structural results. The results of this section are, however, still useful for the numerical experiments in Section 3.2 in order to build insights on the threshold policy for the more general case with a non-constant call arrival rate.

As in Bhulai and Koole (2003), our approach consists in using a Markov chain analysis to derive the steady-state probabilities of the system, from which the performance measures are characterized thereafter. To simplify the presentation, we focus on the particular case $u = s$. The analysis for the case $u = 0$ is obvious, and that of the remaining cases, $0 < u < s$, is done similarly to the case $u = s$. It simply adds a finite number of additional equations but does not impact the general form of the steady-state probabilities. Consider the stochastic process $\{(x(t), y(t)), t \ge 0\}$, where $x(t)$ is the number of waiting calls in the queue and $y(t)$ is the number of emails being in service, $x \in \mathbb{N}$, $y \in \{0, 1, \cdots, s\}$. This process is a Markov chain. For $x \ge 0$ and $0 \le y \le s$, the transition rate from (x, y) to $(x+1, y)$ is λ. For $x \ge 1$ and $0 \le y \le s$, the transition rate from (x, y) to $(x-1, y)$ is $(s-y)\mu$. For $x \ge 1$ and $1 \le y \le s$ the transition rate from (x, y) to $(x-1, y-1)$ is $y\mu_0$. For $0 \le y \le s$, the transition rate from $(0, y)$ to $(0, y+1)$ is $(s-y)\mu$. Due to the priority of inbound calls over emails, no transition exists from (x, y) to $(x, y-1)$, for $x > 0$ and $1 \le y \le s$. Under the stability condition $\frac{\lambda}{s\mu} < 1$, we denote by $p_{x,y}$ the steady-state probability that the system is in state (x, y). Thanks to the Markov chain structure, we solve the steady-state equations using standard results from the theory of linear difference equations (see, for example, Queffélec and Zuily (2013)).

For $y = s$ and $x > 0$, we have $p_{x,s}(\lambda + s\mu_0) = \lambda p_{x-1,s}$. Then $p_{x,s} = (\frac{\lambda}{\lambda + s\mu_0})^x p_{0,s}$. For $0 \le y < s$, and $x > 0$ we have

$$(\lambda + (s-y)\mu + y\mu_0)p_{x,y} = \lambda p_{x-1,y} + (s-y)\mu p_{x+1,y} + (y+1)\mu_0 p_{x+1,y+1}. \quad (9)$$

The homogeneous equation associated with Equation (9) is

$$(s-y)\mu z^2 - (\lambda + (s-y)\mu + y\mu_0)z + \lambda = 0, \quad (10)$$

with z as a variable for $z \in \mathbb{C}$. It has two solutions denoted by z_y and z'_y and are given by

$$z_y = \frac{1}{2(s-y)\mu}\left(\lambda + (s-y)\mu + y\mu_0 - \sqrt{(\lambda + (s-y)\mu + y\mu_0)^2 - 4(s-y)\lambda\mu}\right), \quad (11)$$

$$z'_y = \frac{1}{2(s-y)\mu}\left(\lambda + (s-y)\mu + y\mu_0 + \sqrt{(\lambda + (s-y)\mu + y\mu_0)^2 - 4(s-y)\lambda\mu}\right), \quad (12)$$

for $0 \le y < s$. In Proposition 2, we provide the intervals where z_y and z'_y are ranging.

Proposition 2. *For $0 \le y < s$, we have $0 \le z_y < 1$ and $z'_y > 1$.*

Proof. Let us first prove that $z'_y > 1$. We have $\mu_0 > 0$. Since z'_y increases in μ_0, Equation (12) implies

$$z'_y > \frac{1}{2(s-y)\mu}\left(\lambda + (s-y)\mu + \sqrt{(\lambda + (s-y)\mu)^2 - 4(s-y)\lambda\mu}\right). \quad (13)$$

Observing that $(\lambda + (s-y)\mu)^2 - 4(s-y)\lambda\mu = (\lambda - (s-y)\mu)^2$, Inequality (13) becomes

$$z'_y > \frac{1}{2(s-y)\mu}\left(\lambda + (s-y)\mu + |\lambda - (s-y)\mu|\right),$$

where $|t|$ is the absolute value of t, for $t \in \mathbb{R}$. Consider the case $\lambda \le (s-y)\mu$, thus $|\lambda - (s-y)\mu| = -\lambda + (s-y)\mu$, which leads to $z'_y > 1$. Consider now the remaining case; i.e., $\lambda > (s-y)\mu$. Then $|\lambda - (s-y)\mu| = \lambda - (s-y)\mu$, which implies $z'_y > \frac{\lambda}{(s-y)\mu} > 1$. In summary, we have $z'_y > 1$.

Let us now prove that $0 \le z_y < 1$. From Equation (10), we may write $(s-y)\mu z_y z'_y = \lambda$. Since $\lambda \ge 0$ and $z'_y > 1 > 0$, we obtain $z_y \ge 0$.

In what follows, we prove that $0 \le z_y < 1$. For $\lambda = 0$ we have $z_y = 0$, then the result immediately follows. For $\lambda > 0$,

the derivative of z_y in μ_0 is given by

$$\frac{\partial z_y}{\partial \mu_0} = \frac{y\left(\sqrt{(\lambda+(s-y)\mu+y\mu_0)^2-4(s-y)\lambda\mu}-(\lambda+(s-y)\mu+y\mu_0)\right)}{2(s-y)\mu\sqrt{(\lambda+(s-y)\mu+y\mu_0)^2-4(s-y)\lambda\mu}},$$

for $\mu_0 > 0$. It is straightforward to see that the numerator is strictly negative, for $\lambda > 0$, and the denominator is strictly positive. Therefore, z_y is strictly decreasing in μ_0. Equation (11) then implies

$$z_y < \frac{1}{2(s-y)\mu}\left(\lambda+(s-y)\mu - \sqrt{(\lambda+(s-y)\mu)^2-4(s-y)\lambda\mu}\right), \quad (14)$$

for $\mu_0 > 0$. Using the same discussion as that after Inequality (13), we deduce that $z_y < 1$. This completes the proof of the proposition. ■

Because of the last term on the right-hand side of Equation (9) and the fact that the $2(s+1)$ roots $z_0, z_1, \cdots, z_s, z_0', z_1', \cdots, z_s'$ are all distinct, $p_{x,y}$ can be written as a linear combination of z_i^x and $z_i'^x$ for $y \le i \le s$. Since $z_y' > 1$, the convergence of the stationary probabilities forces the linear factors of $z_y'^x$ to be all equal to zero. We therefore obtain, for $0 \le y \le s$ and $x \ge 0$:

$$p_{x,y} = \sum_{i=y}^{s} A_{i,y} z_i^x, \quad (15)$$

with $z_s = \frac{\lambda}{\lambda+s\mu_0}$ and $A_{i,y} \in \mathbb{R}$ for $0 \le y \le s$ and $y \le i \le s$. The stationary probabilities are now written as a function of a finite number of unknown parameters, namely, the $A_{i,y}$ for $0 \le y \le s$ and $y \le i \le s$. In what follows, we compute these $\frac{(s+1)(s+2)}{2}$ parameters. Using Equation (9), we obtain

$$A_{i,y+1} = A_{i,y}\frac{-(s-y)\mu z_i^2+(\lambda+(s-y)\mu+y\mu_0)z_i-\lambda}{(y+1)\mu_0 z_i^2}, \quad (16)$$

for $0 \le y < i \le s$. Recall that z_i is a root of the equation $(s-i)\mu z^2-(\lambda+(s-i)\mu+i\mu_0)z+\lambda = 0$. Thus,

$$A_{i,y}\frac{-(s-i)\mu z_i^2+(\lambda+(s-i)\mu+i\mu_0)z_i-\lambda}{(y+1)\mu_0 z_i^2} = 0.$$

Subtracting this quantity from the right-hand side of Equation (16) leads to

$$A_{i,y+1} = A_{i,y}\frac{(i-y)(\mu(1-z_i)-\mu_0)}{(y+1)\mu_0 z_i}, \quad (17)$$

for $0 \le y < i \le s$. Therefore,

$$\begin{aligned} A_{i,y} &= A_{i,i}\prod_{k=y}^{i-1}\frac{(k+1)\mu_0 z_i}{(i-k)(\mu(1-z_i)-\mu_0)} \\ &= A_{i,i}\left(\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right)^{i-y}\frac{i!}{y!(i-y)!} \\ &= A_{i,i}\left(\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right)^{i-y}\binom{i}{y}, \end{aligned} \quad (18)$$

for $0 \le y < i < s$. Thus, it remains to compute the $s+1$ parameters $A_{y,y}$, for $0 \le y \le s$. Using Equation (15), we obtain $p_{0,s} = A_{s,s}$ and $p_{0,s-1} = A_{s-1,s-1} + A_{s,s-1}$. From Equation (18), we may write $A_{s,s-1} = A_{s,s}\frac{s\mu_0 z_s}{\mu(1-z_s)-\mu_0}$. Using now the boundary equation $\lambda p_{0,s} = \mu p_{0,s-1}$ and $z_s = \frac{\lambda}{\lambda+s\mu_0}$ implies the following relation between $A_{s-1,s-1}$ and $A_{s,s}$:

$$A_{s-1,s-1} = A_{s,s}\frac{\lambda}{\mu}\frac{\lambda+s\mu_0}{\lambda+s(\mu_0-\mu)}. \quad (19)$$

The other boundary equations are

$$\begin{aligned} p_{0,y}(\lambda+(s-y)\mu) =& (s-y)\mu p_{1,y} + (y+1)\mu_0 p_{1,y+1} \\ &+ (s-y+1)\mu p_{0,y-1}, \end{aligned} \quad (20)$$

for $0 < y < s$. Using Equation (15), we have $p_{0,y} = \sum_{i=y}^{s} A_{i,y}$, $p_{1,y} = \sum_{i=y}^{s} A_{i,y} z_i$, and $p_{1,y+1} = \sum_{i=y+1}^{s} A_{i,y+1} z_i$. Then, using Equation (17) we obtain $p_{1,y+1} = \sum_{i=y+1}^{s} A_{i,y}\frac{(i-y)(\mu(1-z_i)-\mu_0)}{(y+1)\mu_0}$. Thus, $p_{0,y}(\lambda+(s-y)\mu) - (s-y)\mu p_{1,y} - (y+1)\mu_0 p_{1,y+1} = \sum_{i=y}^{s} A_{i,y}(\lambda+(s-i)\mu(1-z_i)+(i-y)\mu_0)$, for $0 < y < s$. Moreover, using Equation (18) implies

$$\begin{aligned} &p_{0,y}(\lambda+(s-y)\mu) - (s-y)\mu p_{1,y} - (y+1)\mu_0 p_{1,y+1} \quad (21) \\ &= \sum_{i=y}^{s} A_{i,i}\left(\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right)^{i-y}\binom{i}{y}(\lambda+(s-i)\mu(1-z_i) \\ &\quad +(i-y)\mu_0), \end{aligned}$$

for $0 < y < s$. Finally we deduce from Equation (20) that

$$\begin{aligned} A_{y-1,y-1} =& \sum_{i=y}^{s} A_{i,i}\left(\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right)^{i-y}\binom{i}{y} \\ &\times\left(\frac{\lambda+(s-i)\mu(1-z_i)+(i-y)\mu_0}{(s-y+1)\mu}\right. \\ &\left.-\frac{y}{i-y+1}\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right), \end{aligned} \quad (22)$$

for $0 < y < s$. Note that this expression is also true for $y = s$; replacing y by s in Equation (22) leads to Equation (19). Since all probabilities sum up to one, we have

$$\sum_{y=0}^{s}\sum_{i=y}^{s}\frac{A_{i,i}}{1-z_i}\left(\frac{\mu_0 z_i}{\mu(1-z_i)-\mu_0}\right)^{i-y}\binom{i}{y} = 1. \quad (23)$$

Equations (19), (22), and (23) form a system of $s+1$ independent linear equations that can be easily numerically

solved and leads to the coefficients $A_{i,i}$, for $0 \le i \le s$. This finishes the characterization of all steady-state probabilities, $p_{x,y}$, for $x \ge 0$ and $0 \le y \le s$.

The email throughput $T(\lambda, \mu, \mu_0, s)$ may be written as

$$T(\lambda, \mu, \mu_0, s) = \mu_0 \sum_{y=1}^{s} \sum_{x=0}^{\infty} y p_{x,y} = \mu_0 \sum_{y=1}^{s} \sum_{i=y}^{s} \frac{y A_{i,y}}{1 - z_i}. \quad (24)$$

From the stability condition on inbound calls, we may write

$$\lambda = \sum_{y=0}^{s} \sum_{x=0}^{\infty} (s - y) \mu p_{x,y}$$

or, equivalently,

$$\lambda = \mu \left(s \sum_{y=0}^{s} \sum_{x=0}^{\infty} p_{x,y} - \sum_{y=0}^{s} \sum_{x=0}^{\infty} y p_{x,y} \right).$$

Since $\sum_{y=0}^{s} \sum_{x=0}^{\infty} p_{x,y} = 1$ and $\sum_{y=0}^{s} \sum_{x=0}^{\infty} y p_{x,y} = \frac{T(\lambda,\mu,\mu_0,s)}{\mu_0}$ we obtain $T(\lambda, \mu, \mu_0, s) = \mu_0(s - \frac{\lambda}{\mu})$. Note that for $\mu_0 = \mu$, the result here coincides with that of the previous section; i.e., $T(\lambda, \mu, \mu, s) = s\mu - \lambda$.

As for the call waiting performance, it is given by

$$P(W > \tau) = \sum_{y=0}^{s} \sum_{x=0}^{\infty} p_{x,y} P(W > \tau | (x, y)),$$

where $P(W > \tau|(x, y))$ is the conditional probability that the waiting time of a new call exceeds τ, given that it finds y emails in service, $s - y$ calls in service, and x calls waiting ahead of it in the queue, for $0 \le y \le s$ and $x \ge 0$. The computation of $P(W > \tau|(x, y))$, for $0 \le y \le s$ and $x \ge 0$, is as follows. For $x = 0$ and $0 \le y \le s$, the new call has to wait for a service completion of one of the y emails, or one of the $s - y$ calls, so $P(W > \tau|(0, y)) = e^{-\tau(y\mu_0+(s-y)\mu)}$. For $x = 1$ and $0 < y \le s$, the probability that the next service completion is that of an email is $\frac{y\mu_0}{y\mu_0+(s-y)\mu}$. Thus, the waiting time of the new call follows a hypoexponential distribution consisting of the summation of two exponential random variables with rates $y\mu_0 + (s - y)\mu$ and $(y - 1)\mu_0 + (s - y + 1)\mu$ with probability $\frac{y\mu_0}{y\mu_0+(s-y)\mu}$, and it follows an Erlang distribution with two phases and $y\mu_0 + (s - y)\mu$ as a rate per stage with probability $1 - \frac{y\mu_0}{y\mu_0+(s-y)\mu}$. This leads to

$$\begin{aligned} &P(W > \tau|(1, y)) \\ &= \frac{y\mu_0}{y\mu_0 + (s - y)\mu} \\ &\times \frac{((y - 1)\mu_0 + (s - y)\mu)e^{-\tau(y\mu_0+(s-y)\mu)} - (y\mu_0 + (s - y)\mu)e^{-\tau((y-1)\mu_0+(s-y)\mu)}}{\mu - \mu_0} \\ &+ \frac{(s - y)\mu}{y\mu_0 + (s - y)\mu} e^{-\tau(y\mu_0+(s-y+1)\mu)}(1 + \tau(y\mu_0 + (s - y)\mu)), \end{aligned}$$

for $0 \le y \le s$. One can continue in the same way to derive all of the conditional waiting time probabilities for $x > 1$, which finishes the characterization of the performance measures (email throughput and call waiting time distribution) in the case of unequal service rates.

3.2. *Construction of the adaptative threshold policy*

In this section, we use the performance evaluation results to find an insight on how we should adapt the threshold as a function of the intensity of the call arrivals. The objective is to maximize the throughput of emails while reaching the constraint on the call waiting times for the whole day. We find that during the periods with low demand, the need to have a good service level is more important than during the periods with high demand. On the basis of this observation, we build a method for adapting the threshold. We then evaluate this method by comparing it with the optimal threshold policy.

3.2.1. *Numerical observations*

For a given time interval long enough to reach the stationary regime, one can use the results of Section 3.1 to obtain the optimal threshold, denoted by u^*, for Problem (1). Consider now a working day with two time intervals, each with a different call arrival rate and on each of which the stationary regime is reached. Following common practice and most call center models in the literature, it is appropriate to assume that a system with constant parameters achieves a steady-state quickly within short—half hour or hour—intervals (Green *et al.*, 2001; Gans *et al.*, 2003).

We want to find the optimal couple of thresholds that answers our optimization problem, where the call service level constraint is for the whole day. We denote the proportion of the length of the first (second) time interval by I_1 (I_2) and the corresponding mean arrival rate by λ_1 (λ_2). We have $I_1 + I_2 = 1$. Without loss of generality, we consider cases where $\lambda_1 \le \lambda_2$. In Table 1, we consider various scenarios for arrival rates, service rates, and relative time durations between the two intervals. Using the results of Section 3.1, we give the optimal threshold of each interval in isolation; i.e., the highest threshold that verifies the service-level constraint. They are denoted by u_1^* and u_2^* for I_1 and I_2, respectively. The symbol "—" in Table 1 is used for the cases where the call service level can not be met, even with a threshold equal to zero. We also evaluate the couple of thresholds which answers Problem (1) on the set of the two intervals. This couple is found by an exhaustive test of all the possible values for the couple (u_1, u_2). We denote by $(u_1, u_2)^*$ this optimal couple. Note that for this couple, Problem (1) does not have to be answered on each interval but rather on the set of the two intervals. Finally, we give the performance measures for each interval and for the set of the two intervals for the couple $(u_1, u_2)^*$. In summary, our optimization problem can be formulated as finding the

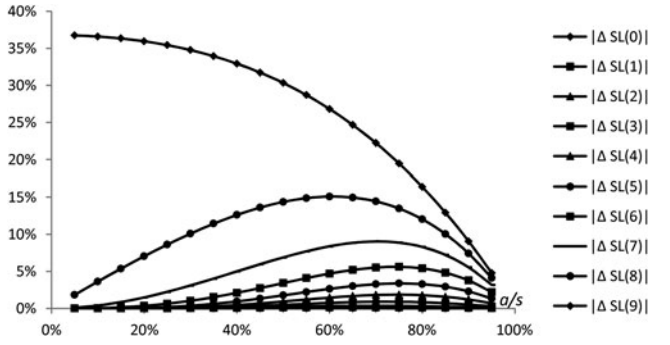

Fig. 2. Sensitivity of the service level ($s = 10$, $\tau = 30$ seconds, $\mu = \mu_0 = 0.2$).

best couple (u_1, u_2) that answers the following problem:

$$\begin{cases} \text{Maximize} \;\; I_1 T_{\lambda_1}(u_1) + I_2 T_{\lambda_2}(u_2), \\ \text{subject to} \;\; I_1 \frac{\lambda_1}{\lambda_1+\lambda_2} SL_{\lambda_1}(u_1) + I_2 \frac{\lambda_2}{\lambda_1+\lambda_2} SL_{\lambda_2}(u_2) \geq \alpha. \end{cases} \tag{25}$$

An important observation from Table 1 is that the choices for the threshold are done for values of u close to s. The reason for this is related to the concavity of the call service level (see Proposition 1). We observe three possible situations corresponding to the three parts of Table 1. In the first situation u_1^* (respectively u_2^*) is always higher or equal to u_1 (respectively lower or equal to u_2) for the optimal couple $(u_1, u_2)^*$. In the second one we have $(u_1, u_2)^* = (u_1^*, u_2^*)$. In the last situation u_1^* (respectively u_2^*) is always lower or equal to u_1 (respectively higher or equal to u_2) for the optimal couple $(u_1, u_2)^*$.

Although the first situation does not seem to be the most intuitive one, it corresponds to most cases. In order to respect the overall call service level, we observe that we should strictly respect the service level during the interval with a small arrival rate (I_1), and more flexibility is accepted when the arrival rate is high (I_2). In what follows, we explain why this insight holds in most practical cases. We first justify that $|\Delta SL(u)|$ ($\Delta SL(u) = SL(u+1) - SL(u)$ for $0 \leq u < s$) is decreasing in the workload in most practical cases. Second, using this assumption we prove that there is less waste for the call service level, when increasing the threshold during higher workload periods. Finally, we derive the required conditions under which the insight does hold.

Figure 2 reveals that, as the workload increases, the sensitivity of the service level for a given threshold ($\Delta SL(u) = SL(u+1) - SL(u)$ for $0 \leq u < s$) first increases and then decreases. In Lemma 1, we prove for $\mu = \mu_0$ that the last part of the curves $|\Delta SL(u)|$ decreases in the workload. Table 2 provides some numerical illustrations for the value of a/s above which the curve $|\Delta SL(u)|$ decreases in a. We observe for this example that the values of a/s are lower than 80%. In practice, the agent utilization in call centers is usually higher than 80% (see Koole (2013)). If a situation with a low workload happens, the threshold would increase and reach its maximal values ($u = s - 1$ or $u = s$). Since the last part of the curves $|\Delta SL(u)|$ as function of the workload decreases, the practical situations are likely to be those where the sensitivity of $SL(u)$ decreases in the workload.

Lemma 1. *The following holds for $\mu = \mu_0$, $0 \leq a \leq s$ and $s \geq 2$.*

1. *$|\Delta SL(s-1)|$ is decreasing in a if $s - \frac{1}{\tau\mu} \leq 0$, otherwise $|\Delta SL(s-1)|$ is first increasing then decreasing in a.*
2. *There exists a value of a, $0 < a < s$, above which $|\Delta SL(u)|$ is decreasing in a, for $0 \leq u < s - 1$.*

Proof. Using Theorem 1, we have $|\Delta SL(u)| = |SL(u+1) - SL(u)| = e^{-\tau s\mu(1-a/s)}|C(s, u+1, a) - C(s, u, a)|$, for $0 \leq u \leq s-1$. Let us now prove the first statement of the Lemma. Replacing u by $s-1$ in Equation (3) leads to

$$\begin{aligned} C(s, s-1, a) &= \frac{a^{s-(s-1)}(s-1)!}{s!(1-a/s)} \left(\sum_{k=0}^{s-(s-1)} \frac{a^k (s-1)!}{((s-1)+k)!} \right. \\ &\quad \left. + \frac{a^{s-(s-1)}(s-1)!}{s!} \frac{a}{s-a} \right)^{-1} \\ &= \frac{a/s}{(1-a/s)} \left(1 + a/s + (a/s)^2 \frac{1}{1-a/s} \right)^{-1} \\ &= a/s, \end{aligned}$$

for $0 \leq a \leq s$. We also have $C(s, s, a) = 1$. Thus, $|\Delta SL(s-1)| = e^{-\tau s\mu(1-a/s)}(1 - a/s)$, for $0 \leq a \leq s$. We have

$$\frac{\partial |\Delta SL(s-1)|}{\partial a} = \frac{1}{s} e^{-\tau s\mu(1-a/s)}(-1 + (s-a)\tau\mu),$$

for $0 \leq a \leq s$. The expression $-1 + (s-a)\tau\mu$ decreases in a. The equation $-1 + (s-a)\tau\mu = 0$ in the variable a is equivalent to $a = s - \frac{1}{\tau\mu}$. If $s - \frac{1}{\tau\mu} \leq 0$ then $\frac{\partial |\Delta SL(s-1)|}{\partial a} \leq 0$ for $0 \leq a \leq s$ and $|\Delta SL(s-1)|$ decreases in a. Otherwise, if $s - \frac{1}{\tau\mu} > 0$, $|\Delta SL(s-1)|$ is first increasing from 0 to $s - \frac{1}{\tau\mu}$ and then decreasing from $s - \frac{1}{\tau\mu}$ to s as a function of a ($0 \leq a \leq s$). Note that $s - \frac{1}{\tau\mu} < s$. We then deduce that the last part of the curve of $|\Delta SL(s-1)|$ is always decreasing as a function of a.

We next prove the second statement. We can write $C(s, u, a)$ as $C(s, u, a) = \frac{a^{s-u}u!}{s!(1-a/s)} p_u$. If $0 \leq u < s$, we have

$$\lim a \to 0, a > 0 \; \frac{a^{s-u}u!}{s!(1-a/s)} = 0.$$

From the proof of Proposition 1, we know that

$$\lim a \to 0, a > 0 \; p_u = 1.$$

We then obtain

$$\lim a \to 0, a > 0 \; C(s, u, a) = 0,$$

Table 1. Optimal couples of thresholds ($s = 10$, $\tau = 30$ seconds, $\alpha = 80\%$)

λ_1	λ_2	μ	μ_0	$I_1(\%)$	$I_2(\%)$	u_1^*	u_2^*	$(u_1, u_2)^*$	$P(W_1 < \tau)(\%)$	$P(W_2 < \tau)(\%)$	$P(W < \tau)(\%)$	T_1	T_2	T
1	1	0.2	0.2	50	50	8	8	(8, 8)	84.04	84.04	84.04	0.758	0.758	0.758
1	1.3	0.2	0.2	50	50	8	6	(8, 7)	84.04	77.99	80.62	0.758	0.401	0.580
0.5	1.5	0.2	0.2	50	50	9	—	(8, 4)	96.81	74.79	80.30	1.169	0.055	0.611
1	1.3	0.2	0.2	67	33	8	6	(8, 7)	84.04	77.99	81.66	0.758	0.401	0.639
1	1.3	0.2	0.2	80	20	8	6	(8, 8)	84.04	69.15	80.39	0.758	0.552	0.711
0.5	1.5	0.2	0.2	90	10	9	—	(9, 7)	88.19	63.94	82.13	1.350	0.277	1.243
1	1.5	0.2	0.2	50	50	8	—	(7, 5)	90.92	72.93	80.13	0.604	0.111	0.357
1	1.5	0.2	1	50	50	10	—	(10, 7)	89.34	74.94	80.70	5.191	0.961	3.076
1	1.5	0.2	1	80	20	10	—	(10, 10)	89.34	67.56	83.40	5.191	2.908	4.734
1.3	1.4	0.2	1	50	50	9	8	(9, 9)	83.51	77.09	80.18	2.863	2.440	2.652
1.3	1.4	0.2	1	80	20	9	8	(9, 10)	83.51	68.19	80.26	2.863	3.621	3.014
1.3	1.4	1	0.2	80	20	9	9	(9, 10)	88.63	60.45	82.10	1.616	1.794	1.742
1.3	1.4	1	0.2	50	50	9	9	(9, 9)	88.63	87.77	88.18	1.616	1.598	1.601
0.5	1	0.2	0.2	50	50	9	8	(9, 8)	88.19	84.04	85.42	1.350	0.758	1.054
0.2	1	0.2	0.2	50	50	9	8	(10, 7)	59.34	90.92	85.66	1.800	0.604	1.202
0.1	1	0.2	0.2	50	50	9	8	(10, 8)	61.33	84.04	81.97	1.900	0.758	1.468
0.01	1	0.2	0.2	50	50	9	8	(10, 8)	63.03	84.04	83.83	1.990	0.758	1.513

for $0 \leq u < s$. We can also write $C(s, u, a)$ as

$$C(s, u, a) = \left(a/s + s!(1 - a/s) \sum_{k=0}^{s-u} \frac{a^{k+u-s}}{(u+k)!} \right)^{-1}.$$

We have

$$\lim a \to s, a < s \; a/s + s!(1 - a/s) \sum_{k=0}^{s-u} \frac{a^{k+u-s}}{(u+k)!} = 1,$$

for $0 \leq u \leq s$. Thus,

$$\lim a \to s, a < s \; C(s, u, a) = 1,$$

for $0 \leq u \leq s$. Since we have $e^{-\tau s \mu} \leq e^{-\tau s \mu (1-a/s)} \leq 1$ for $0 \leq a \leq s$, we obtain

$$\lim a \to s, a < s \; |\Delta SL(u)| = \lim a \to 0, a > 0 \; |\Delta SL(u)| = 0,$$

for $0 \leq u < s - 1$.

Since $|\Delta SL(u)|$ is not zero, the curve of $|\Delta SL(u)|$ has at least one extremum in the variable a for $0 < a < s$. This proves that there exists a value of a ($0 < a < s$) after which $|\Delta SL(u)|$ decreases in a for $0 \leq u < s - 1$ and completes the proof of the lemma. ■

Assuming now that the sensitivity of the call service level is decreasing in the workload, we prove in Proposition 3 that there is less waste for the call service level, when increasing the threshold during higher workload periods. For the call service level constraint in Problem (25), Corollary 1 completes Proposition 3 by providing the necessary conditions under which it is better to increase the threshold during high workload periods.

Proposition 3. *If $\lambda_1 < \lambda_2$ and $\Delta SL(u)$ is decreasing in the workload, then $|\Delta SL_{\lambda_1}(u_1^*)| > |\Delta SL_{\lambda_2}(u_2^*)|$.*

Proof. We distinguish two cases; $u_1^* = u_2^*$ or $u_1^* > u_2^*$. The other case $u_1^* < u_2^*$ does not exist because $\lambda_1 < \lambda_2$. If $u_1^* = u_2^*$ then increasing u is less sensitive in SL_{λ_2} than in SL_{λ_1} since the sensitivity of SL is decreasing in the workload; i.e., $|\Delta SL_{\lambda_1}(u_1^*)| > |\Delta SL_{\lambda_2}(u_2^*)|$. Consider now the case $u_1^* > u_2^*$. Since SL is decreasing and concave in u (see Proposition 1), we deduce that SL_{λ_1} is more sensitive to the increasing of u starting from u_1^* than starting from u_2^*; i.e., $|\Delta SL_{\lambda_1}(u_1^*)| > |\Delta SL_{\lambda_1}(u_2^*)|$. Starting from u_2^*, SL_{λ_1} is more sensitive to the increasing of u than SL_{λ_2}; i.e., $|\Delta SL_{\lambda_1}(u_2^*)| > |\Delta SL_{\lambda_2}(u_2^*)|$. As a consequence SL_{λ_2} is less sensitive to the increasing of u starting from u_2^* than SL_{λ_1} would be starting from u_1^*; i.e., $|\Delta SL_{\lambda_1}(u_1^*)| > |\Delta SL_{\lambda_2}(u_2^*)|$. ■

Corollary 1. *If $\frac{I_2 |\Delta SL_{\lambda_2}(u_2^*)|}{I_1 |\Delta SL_{\lambda_1}(u_1^*)|} \lambda_2 < \lambda_1 < \lambda_2$ and $\Delta SL(u)$ is decreasing in the workload, there is less waste for the call service level on the two intervals, when increasing the threshold during the higher workload period.*

Proof. Using Proposition 3 we know that $|\Delta SL_{\lambda_1}(u_1^*)| > |\Delta SL_{\lambda_2}(u_2^*)|$, or $\frac{|\Delta SL_{\lambda_2}(u_2^*)|}{|\Delta SL_{\lambda_1}(u_1^*)|} < 1$. We can then find values of λ_1, λ_2, I_1, and I_2 that satisfy the inequality. $\frac{I_2 |\Delta SL_{\lambda_2}(u_2^*)|}{I_1 |\Delta SL_{\lambda_1}(u_1^*)|} \lambda_2 < \lambda_1 < \lambda_2$, then with the first inequality we have $I_2 |\Delta SL_{\lambda_2}(u_2^*)| \lambda_2 < I_1 |\Delta SL_{\lambda_1}(u_1^*)| \lambda_1$, and, finally:

$$I_1 \frac{\lambda_1}{\lambda_1 + \lambda_2} |\Delta SL_{\lambda_1}(u_1^*)| > I_2 \frac{\lambda_2}{\lambda_1 + \lambda_2} |\Delta SL_{\lambda_2}(u_2^*)|.$$

This finishes the proof of the corollary. ■

Table 2. Value of a/s above which $|\Delta SL(u)|$ decreases in a ($s = 10$, $\tau = 30$ seconds, $\mu = \mu_0 = 0.2$)

	u									
	0	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*
a/s	0.56	0.62	0.67	0.72	0.73	0.74	0.74	0.67	0.53	0

Table 3. Interval of validity of Corollary 1 ($\lambda_2 = 1.3$, $s = 10$, $\mu_0 = \mu = 0.2$, $\tau = 30$s, $\alpha = 80\%$)

	λ_1							
	0.01	*0.1*	*0.25*	*0.5*	*0.75*	*1*	*1.25*	*1.29*
$\frac{\lvert\Delta SL_{\lambda_2}(u_2^*)\rvert}{\lvert\Delta SL_{\lambda_1}(u_1^*)\rvert}\lambda_2$	0.1834	0.1836	0.1850	0.1905	0.5603	0.4697	0.7810	0.76662
$\frac{\lvert\Delta SL_{\lambda_2}(u_2^*)\rvert}{\lvert\Delta SL_{\lambda_1}(u_1^*)\rvert}\lambda_2 < \lambda_1$	False	False	True	True	True	True	True	True

In practice, the changes in the threshold are likely to be made by the Automatic Call Distributer (ACD) at predefined and equal intervals of time; i.e., $I_1 = I_2$. The condition for the result in Corollary 1 to hold then becomes

$$\frac{|\Delta SL_{\lambda_2}(u_2^*)|}{|\Delta SL_{\lambda_1}(u_1^*)|} < \frac{\lambda_1}{\lambda_2} < 1.$$

Note that this condition does not happen only for the extreme situations with very high differences between the mean arrival rate values ($\lambda_1 << \lambda_2$). An illustration is given in Table 3.

3.2.2. *Our adaptive threshold policy*

We propose for Problem (1) an Adaptive Threshold Policy (ATP) that adjusts the threshold as a function of the call workload. This policy is based on the the first-, and second-order monotonicity properties of the performance measures as a function of the threshold u and on the observation drawn in Section 3.2.1. As mentioned in Section 2, the threshold is re-evaluated at the beginning of each interval i ($i = 1, \ldots, N$). The threshold associated with interval i is denoted by u_i. The global service level for the whole day (all N intervals) is denoted by SL, and the global one from interval 1 to interval i is denoted by SL_i, for $i = 1, \ldots, N$.

If SL_i is higher (lower) than α at the beginning of an interval i ($i = 2, \ldots, N$) then the policy increases (decreases) the threshold. To update the threshold, we use a real parameter denoted by c_i ($i = 1, \ldots, N$). The threshold u_i is defined as the closest integer to c_i, for $i = 1, \ldots, N$. Note that the parameter c_i is chosen to be real in order to smooth the change in the threshold u_i. We start with $u_1 = c_1 = s$. For $i \geq 2$, if we need to increase the threshold (in the case of $SL_i > \alpha$), then we consider $c_i = c_{i-1} + 1 - c_{i-1}/s$. If we need to decrease the threshold (in the case where $SL_i < \alpha$), then $c_i = c_{i-1} - c_{i-1}/s$. In the remaining case ($SL_i = \alpha$), we consider $c_i = c_{i-1}$.

In what follows, we discuss the efficiency of how ATP updates the threshold. The main two characteristics of ATP are as follows:

1. An increasing (decreasing) of the threshold in case the measured call service level is better (worse) than the target service level.
2. A decreasing speed in the increasing (decreasing) of the threshold when this threshold increases (decreases).

From Proposition 1, we know that the throughput increases and the call service level decreases in u. Thus, the threshold should be increased when the measured service level is better than the target service level and vice versa. This justifies the first characteristic of ATP.

The second characteristic of ATP is justified by the convexity of the performance measures and the correlation between them. Consider a situation with sufficiently high call service levels; for example, during light workload periods. The threshold u should then reach high values close to the number of agents. For high values of the threshold, we know from Proposition 1 that the call service level is decreasing and concave, and the email throughput is increasing and concave in u. An illustration is given in Figs. 3(a) and 3(b). Therefore, increasing u would go with only a little improvement in the email throughput and at the same time a high loss in the call service level. This situation is well managed by ATP. As u increases, ATP decreases the speed of increasing u, which reduces the non-efficient situations with high values of the threshold. Moreover, ATP behaves as required by the insight derived in Section 3.2.1. From the insight, we know that for the optimization problem (1) we should strictly respect the call service level constraint during light workload periods. ATP conservatively increases a high threshold, which is the way to give importance to the respect the call service level constraint.

Consider now a situation with poor call service levels; for example, during high workload periods. The threshold u should then reach small values. For small values of the threshold, we know from Proposition 1 that the call service level is decreasing and concave in u; i.e., almost in sensitive to the decreasing of u. This does not hold for the email throughput. An illustration is given in Figs. 3(c) and 3(d). Therefore, decreasing u would go with only a little improvement in the call service level and at the same time a high loss in the email throughput. This situation is again well managed by ATP. As u decreases, ATP decreases the speed of decreasing u, which reduces the non-efficient situations with small values of the threshold. Again, ATP behaves as required by the insight derived in Section 3.2.1. From the insight, we know that for the optimization problem (1) it is tolerated to violate the call service level constraint during high workload periods. ATP conservatively decreases a low threshold, which is the way to give less importance to the respect of the call service level constraint.

3.2.3. *Evaluation of the ATP*

In this section, we evaluate the quality of the ATP policy by comparing it with the optimal one. First, we provide the optimal threshold policy. Because of the discrete nature of the threshold, one may see that the threshold should

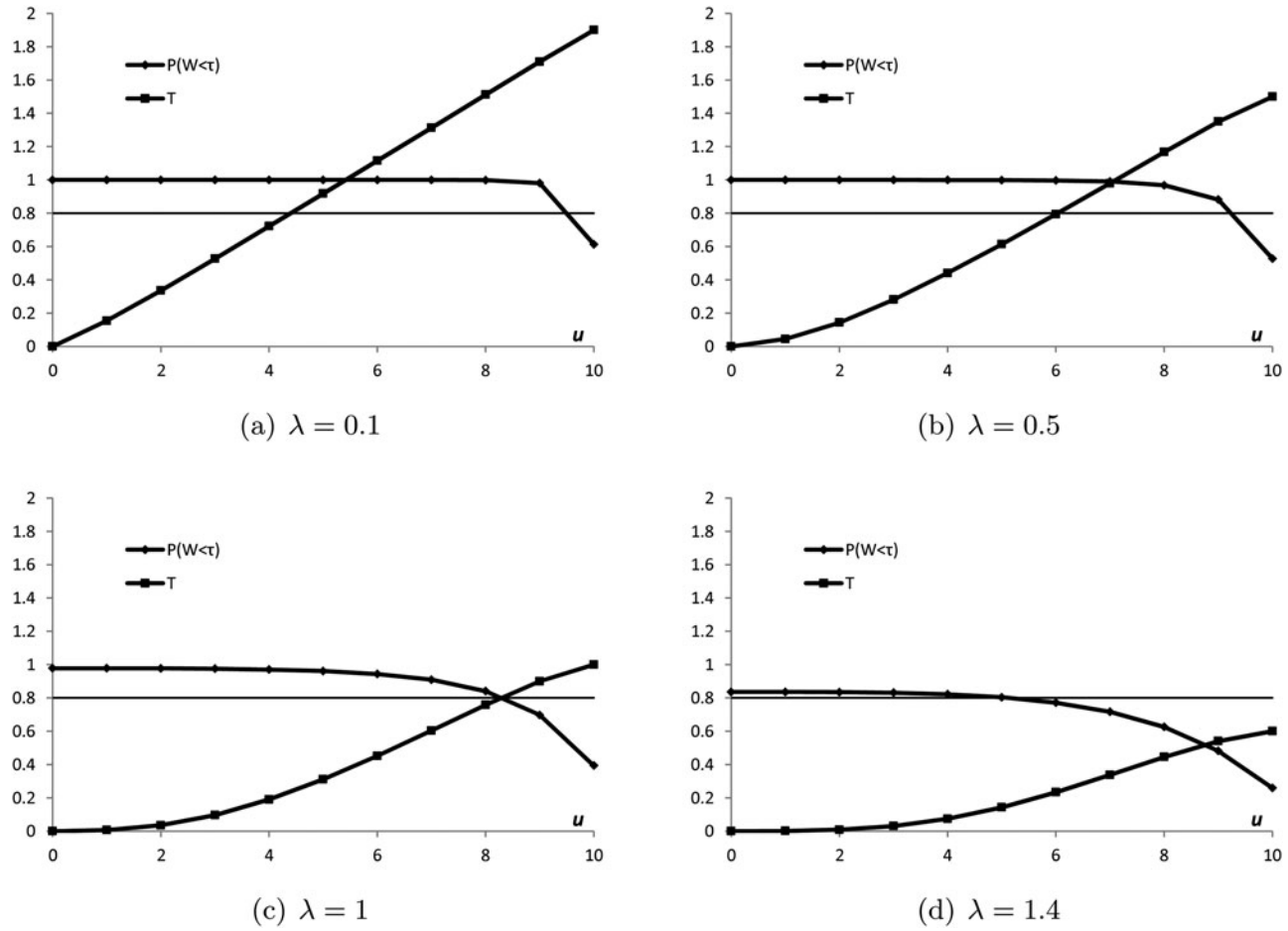

Fig. 3. Performance measures ($s = 10$, $\mu = \mu_0 = 0.2$, $\tau = 0.5$, $\alpha = 80\%$).

vary between two or more values. The reason for this is that we need to exactly satisfy the constraint on calls in Problem (1) in order to maximize the email throughput. From Bhulai and Koole (2003), we know that to exactly satisfy the constraint on calls, randomization is optimal for threshold policies. For both cases $\mu_0 = \mu$ and $\mu_0 \neq \mu$, Theorem 2 provides a weak condition that leads to the optimal randomization policy between two threshold values. A randomized threshold policy, between two thresholds u_1 and u_2 and with a randomization parameter $p \in [0, 1]$, works as follows. At each event (an inbound call arrival or a service completion), the value of the threshold value changes from u_1 to u_2 with probability p, stays in u_1 with probability $1 - p$, changes from u_2 to u_1 with probability $1 - p$, stays in u_2 with probability p.

Theorem 2. *Consider* $0 \leq u_1, u_2 \leq s$ *such that* $SL(u_1) \leq \alpha \leq SL(u_2)$. *If there exists* $\gamma \in \mathbb{R}$ *for which randomizing between* u_1 *and* u_2 *maximizes* $T(u) + \gamma SL(u)$ *and leads to a call service level exactly equal to* α, *then randomizing between* u_1 *and* u_2 *is optimal.*

Proof. Let $p \in [0, 1]$ be the parameter of randomization between u_1 and u_2. Assume that we can find a couple $(u_3, u_4) \neq (u_1, u_2)$ and a parameter of randomization $q \in [0, 1]$ such that the constraint on calls is also saturated and $SL(u_3) \leq \alpha \leq SL(u_4)$. We have $pT(u_1) + (1 - p)T(u_2) + \gamma pSL(u_1) + \gamma(1 - p)SL(u_2) \geq qT(u_3) + (1 - q)T(u_4) + \gamma qSL(u_3) + \gamma(1 - q)SL(u_4)$. Since $\gamma pSL(u_1) + \gamma(1 - p)SL(u_2) = \gamma qSL(u_3) + \gamma(1 - q)SL(u_4) = \gamma\alpha$, we deduce that $pT(u_1) + (1 - p)T(u_2) \geq qT(u_3) + (1 - q)T(u_4)$. Then the couple (u_1, u_2) is optimal, which completes the proof. ■

The randomization between two thresholds allows for the constraint on calls to be met exactly. For our system with constant parameters, we believe that the randomization is between two successive thresholds. Since the throughput is neither convex nor concave it is difficult to rigorously prove this result. However, if we denote by u^* ($0 \leq u^* \leq s$) the highest threshold that verifies $SL(u^*) > \alpha$, we numerically checked that with $\gamma = -\frac{T(u^*+1) - T(u^*)}{SL(u^*+1) - SL(u^*)}$ (for $0 \leq u^* < s$), the expression $T(u) + \gamma \times SL(u)$ is strictly increasing from $u = 0$ to $u = u^*$, strictly decreasing from $u = u^* + 1$ to $u = s$ and $T(u^*) + \gamma SL(u^*) = T(u^* + 1) + \gamma SL(u^* + 1)$. Then for all of the considered numerical situations the optimal policy is a randomization between two adjacent values

Table 4. Comparison under the steady-states assumption (θ=15 minutes)

	Optimal c	*Optimal T*	*ATP T*	*Difference (%)*
Scenario 1 ($\lambda = 4, \mu = \mu_0 = 0.2, s = 28$)	25.49	1.39	1.37	1.46
Scenario 2 ($\lambda = 0.02, \mu = \mu_0 = 0.2, s = 1$)	0.13	0.02	0.02	0.00
Scenario 3 ($\lambda = 18, \mu = \mu_0 = 0.2, s = 100$)	93.91	1.65	1.58	4.43
Scenario 4 ($\lambda = 4, \mu = 0.27, \mu_0 = 0.15, s = 28$)	26.63	1.89	1.89	0.00
Scenario 5 ($\lambda = 4, \mu = 0.17, \mu_0 = 1, s = 28$)	23.21	2.00	1.79	11.73

when $0 \leq u^* < s$. When $u^* = s$, the optimal policy is to keep the threshold constant and equal to s.

In Table 4, we propose five representative scenarios with constant arrival rates and compare the optimal throughput with the one found with the ATP. Although the ATP method is not optimal, the difference with the optimum is quite small. This shows the advantage of ATP in the case of constant arrival rates. Recall that our main purpose in this article is the analysis of the case with a fluctuating arrival rate. In the next section, we consider the case of a fluctuating arrival rate and evaluate the performance of ATP through a comparison with other intuitive methods.

4. Non-constant arrival rates

In Section 4.1 we compare ATP with methods that use constant step sizes. Then in Section 4.2 we analyze the impact of the parameters on the choice of the method. In Section 4.3 we propose some other intuitive adaptive methods.

We consider cases where the length of the working day equals 8 hours ($D = 8$ h) and a frequent possibility of re-evaluating the real threshold c, at the beginning of each time interval with length $\theta = 1$, 5, or 15 minutes. We use simulation to obtain the performance measures. For each scenario, we run n replications. We then introduce a measure of the bias after the n simulations, denoted by $\overline{r}_n$ and calculated as $\overline{r}_n = \frac{\sum_{k=1}^{n} \text{Max}(\alpha - \overline{SL}_k, 0)}{n}$, where $\overline{SL}_k$ is the service level of simulation k ($1 \leq k \leq n$). Since the value of $\overline{r}_n$ should be as small as possible, we introduce a coefficient A that would be the aversion of the call center manager to the risk and introduce an utility indicator denoted by U_n and given by $\overline{T}_n - A \times \overline{r}_n$, where $\overline{T}_n$ is the expected throughput after n simulations. The confidence intervals are a safe way to evaluate the required number of equivalent simulations, n. The confidence interval for a proportion p and a risk of 5% is $(p - 1.96\sqrt{\frac{p(1-p)}{n}}, p + 1.96\sqrt{\frac{p(1-p)}{n}})$ in which n is the number of terms used to calculate the proportion p. If we want a precision of one decimal we need $2 \times 1.96\sqrt{\frac{0.8(1-0.8)}{n}} < 0.001$ then $n > 2\,458\,624$. In order to have safe results we run each simulation 3000 000 times.

4.1. *Comparison with constant step methods*

We propose different scenarios to compare ATP with constant step size methods. We denote by h the step size ($0 < h \leq 1$). When we need to increase (respectively decrease) the real threshold c_i after i intervals ($1 \leq i < N$) under the case $SL_i > \alpha$ (respectively $SL_i < \alpha$) we add h to c_i (respectively we add $-h$ to c_i). In each scenario we use an aversion of risk equal to 100 and initialize the system with $c_0 = u_0 = s$. In some scenarios the number of agents varies over the day. When the number of agents decreases, we could be in a situation in which $c > s$; i.e., the number of busy agents becomes higher than the new value for s. To avoid such a situation, we force in the simulation the change of c to the new smaller value of s. Any undertaken task by a removed agent is lost. In all scenarios the constraint on calls is such that the proportion of calls that wait less than 30 seconds is at least 80%, $\tau = 30$s and $\alpha = 80\%$. We consider the following scenarios:

- Scenario 1: $\lambda = 4$, $\mu = \mu_0 = 0.2$, $s = 28$, and $N = 480$ ($\theta = 1$ min);
- Scenario 2: $\lambda = 4$, $\mu = \mu_0 = 0.2$, $s = 28$, and $N = 32$ ($\theta = 15$ min);
- Scenario 3: $\lambda = 4, \mu = 0.27, \mu_0 = 0.15, s = 28$, and $N = 480$;
- Scenario 4: $\lambda = 4$, $\mu = 0.17$, $\mu_0 = 1$, $s = 28$, and $N = 480$;
- Scenario 5: λ linearly decreasing from 5 to 3, $\mu = \mu_0 = 0.2$, $s = 34$ if $\lambda > 4.5$, $s = 28$ if $4.5 > \lambda > 3.5$, $s = 23$ in the remaining cases, and $N = 480$;
- Scenario 6: λ linearly increasing from 3 to 5, $\mu = \mu_0 = 0.2$, $s = 34$ if $\lambda > 4.5$, $s = 28$ if $4.5 > \lambda > 3.5$, $s = 23$ in the remaining cases, and $N = 480$;
- Scenario 7: During the first quarter of the period λ is linearly increasing from one to five, during the second quarter λ is linearly decreasing from 5 to 3, during the third quarter λ is linearly increasing from 3 to 5, and during the last quarter λ is linearly decreasing from 5 to

Table 5. Comparison between ATP and constant step methods

	h	$\overline{T}$	$\overline{SL}\,\%$	$\overline{r}$	U		h	$\overline{T}$	$\overline{SL}\,\%$	$\overline{r}$	U
Sc 1	0.1	1.17	80.6	0.0046	0.71	Sc 2	0.1	1.53	72.15	0.0782	−6.3
	0.2	1.12	80.5	0.0036	0.77		0.2	1.38	78.7	0.0201	−0.63
	0.5	1.04	80.1	0.0032	0.72		0.5	1.23	81.4	0.0063	0.60
	1	0.98	80.0	0.0035	0.63		1	1.19	80.7	0.0062	0.57
	ATP	1.09	80.7	0.0027	0.82		ATP	1.12	85.6	0.0008	1.04
Sc 3	0.1	1.85	80.3	0.0023	1.62	Sc 4	0.1	3.20	78.9	0.0314	0.06
	0.2	1.80	80.3	0.0017	1.63		0.2	3.07	79.5	0.0277	0.30
	0.5	1.68	80.3	0.0013	1.55		0.5	3.05	79.2	0.0278	0.27
	1	1.57	80.2	0.0014	1.43		1	3.14	79.2	0.0281	0.33
	ATP	1.72	81.0	0.0003	1.68		ATP	2.95	78.9	0.0264	0.31
Sc 5	0.1	1.19	79.9	0.0067	0.52	Sc 6	0.1	1.05	83.2	0.0014	0.91
	0.2	1.13	80.1	0.0037	0.76		0.2	1.04	81.7	0.0021	0.83
	0.5	1.08	80.0	0.0033	0.75		0.5	1.04	80.8	0.0025	0.79
	1	1.01	79.9	0.0033	0.68		1	1.04	80.2	0.0032	0.72
	ATP	1.12	80.4	0.0018	0.93		ATP	1.09	82.1	0.0007	1.02
Sc 7	0.1	1.38	81.6	0.0010	1.28	Sc 8	0.1	3.04	78.8	0.0246	0.59
	0.2	1.37	81.2	0.0015	1.21		0.2	2.84	79.5	0.0201	0.83
	0.5	1.27	80.4	0.0017	1.10		0.5	2.72	79.3	0.0178	0.94
	1	1.24	80.3	0.0011	1.14		1	2.76	78.9	0.0188	0.88
	ATP	1.38	81.2	0.0005	1.33		ATP	2.83	79.7	0.0172	1.11
Sc 9	0.1	1.41	81.6	0.0002	1.39	Sc 10	0.1	1.31	81.43	0.0028	1.03
	0.2	1.41	81.5	0.0004	1.37		0.2	1.25	81.30	0.0021	1.04
	0.5	1.38	81.5	0.0002	1.36		0.5	1.22	81.20	0.0019	1.03
	1	1.36	81.6	0.0002	1.34		1	1.19	80.80	0.0016	1.03
	ATP	1.37	82.4	0.0000	1.37		ATP	1.21	82.53	0.0010	1.11
Sc 11	0.1	0.61	80.5	0.0047	0.13	Sc 12	0.1	0.59	80.5	0.0047	0.12
	0.2	0.56	80.5	0.0037	0.19		0.2	0.52	80.4	0.0038	0.14
	0.5	0.52	80.2	0.0029	0.23		0.5	0.50	80.4	0.0031	0.19
	1	0.48	80.0	0.0038	0.10		1	0.47	80.0	0.0039	0.08
	ATP	0.54	80.8	0.0026	0.28		ATP	0.53	80.8	0.0026	0.27

1, $\mu = \mu_0 = 0.2$, $s = 34$ if $\lambda > 4.5$, $s = 28$ if $4.5 > \lambda > 3.5$, $s = 23$ in the remaining cases, and $N = 480$;

- Scenario 8: The period T is divided into 10 sub-periods and the value of λ alternates between the values 5 and 0.5; i.e., it is 5 in the first sub-period, 0.5 in the second one, again 5 in the third one, and so on, $\mu = \mu_0 = 0.2$, $s = 28$ and $N = 480$;
- Scenarios 9, 10, 11, and 12: For further practical evidence, we relax the assumption of the exponential distribution for call and email service times. We instead consider a log-normal distribution with expected service rates of $\mu = \mu_0 = 0.2$. Let us denote by cv the coefficient of variation of a given distribution. It is defined as the ratio between its standard deviation and its expected value. We vary the standard deviation in scenarios 9, 10, 11, and 12 in order to reach $cv = 0$, 0.5, 1.5, and 2, respectively. In all scenarios, we choose $\lambda = 4$, $s = 28$, and $N = 480$ ($\theta = 1$ minute).

The results are shown in Table 5. We consider values of h = 0.1, 0.2, 0.5, and 1. We observe that ATP performs better or at least similar to the constant step methods with an aversion of risk equal to 100.

In Figs. 4(a), 4(b), and 4(c), we present the evolution of the threshold, the proportion of customers that wait less than 30 seconds, and the email throughput as a function of time in one simulation of scenario 2. This is an illustration that could help to understand why ATP is efficient. With a small value of h ($h = 0.2$), the initialization has an important impact on the evolution of the threshold. At the beginning with $u_0 = c_0 = s = 28$, there is a need to decrease the threshold. A small value of h does not allow us to do this decrease quickly enough. Then there is a need to keep on decreasing the threshold in order to have a chance to reach the service level on calls over the whole day. On the other hand, a high value of h ($h = 1$) goes with a fluctuation of the threshold, with sometimes poor call service levels and other times poor email throughput. Note that the higher is h, the faster the service level converges its target. In what follows we go further in analyzing the impact of the main parameters on the choice of h.

4.2. *Impact of the parameters*

In this section, we analyze the impact of the parameters for the choice of a constant value for h.

Impact of the number of intervals, N: The comparison between scenarios 1 and 2 in Table 5 indicates that there is a link between h and N. In scenario 2 with only 32 intervals, a small value of h does not allow one to reach the call service level constraint. In scenario 1, a large number of intervals and a high value for h lead to an important fluctuation from $u = 0$ to s. An advantage of ATP is its ability to adapt to the number of intervals. A high number of intervals can lead to a high probability of reaching extreme and inefficient states ($u = 0$ or s). Thanks to slowing

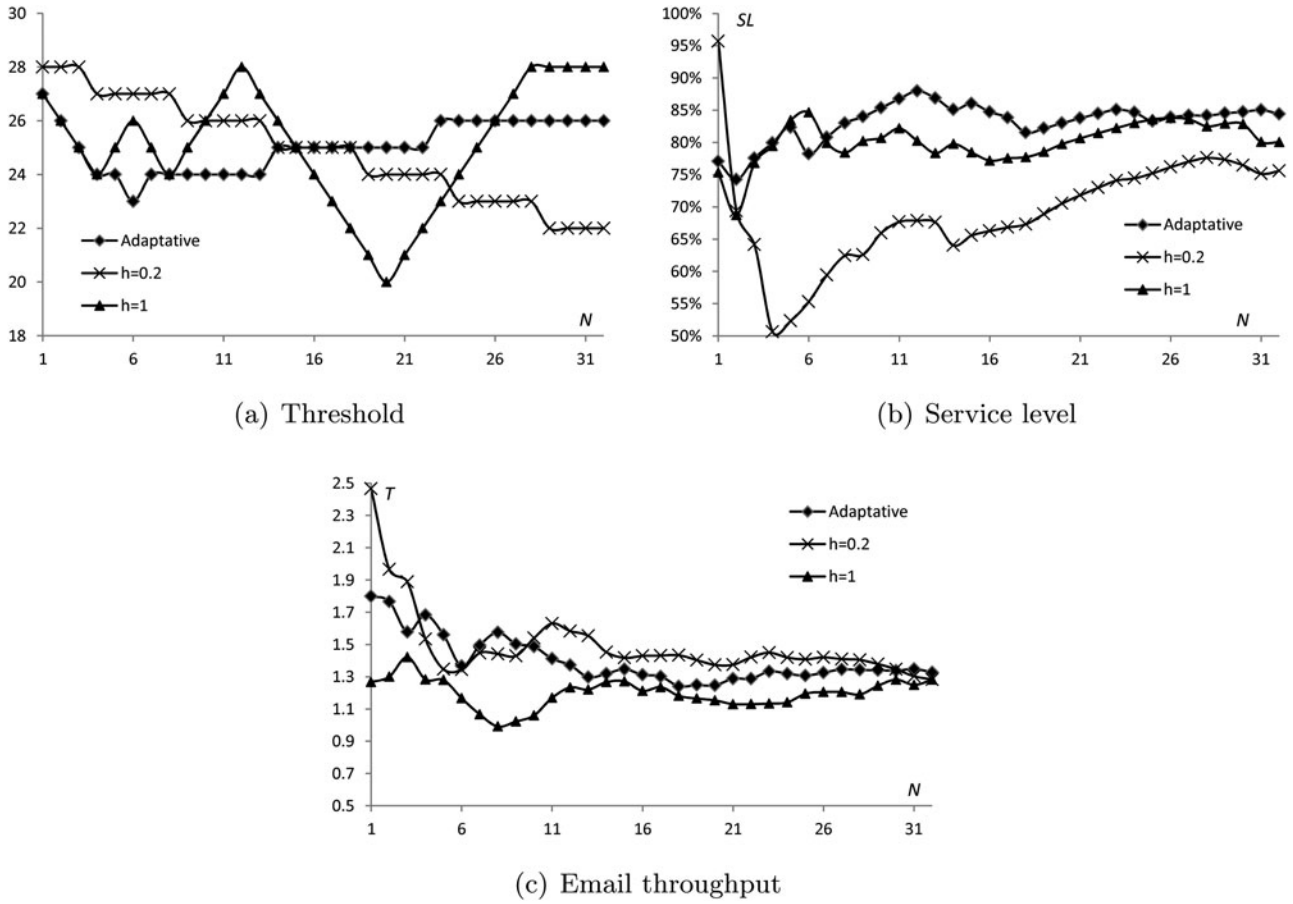

Fig. 4. Evolution of the threshold, the service level, and the throughput (scenario 2).

down the speed of the threshold reduction when c is small and the slowing down of the threshold increasing when c is high, this is unlikely to happen. ATP also provides a high capacity for reaction to when the threshold is too low or too high, which is important in the case of a small number of intervals.

Impact of the aversion of risk A: The choice of the method for adapting the threshold depends on the risk aversion of the manager. In our simulations we considered that $A = 100$ and showed the efficiency of ATP. This value provides a balance between the performance ($\overline{T}$) and the risk ($\overline{r}$). If we consider the extreme case of an infinite aversion to the risk ($A = \infty$), the choice will be made for the smallest value of $\overline{r}$. We observe that the choice will still be for ATP rather than the other methods even more than with $A = 100$. This is an important advantage of ATP; it is a safe method (i.e., the probability that the service level constraint is eached at the end of the working period is high). Since the threshold c is usually closer to s than to zero, due to of the concavity at the service level we usually have a higher speed in decreasing the threshold than in increasing it, which is safe and explains the small values for $\overline{r}$. On the other hand, if the manager has no risk aversion ($A = 0$) then the choice will be made for the highest average throughput ($\overline{T}$). ATP is then not the best one but it still provides results close to the best ones in Table 5.

Impact of the Email service rate: Consider scenarios 3 and 4. We observe that ATP performs better when the emails are served slower (scenario 3) than when they are served faster (scenario 4) than the calls. When the emails are served faster than the calls, the need to increase c is more important because an email does not occupy an agent for a long period of time, but with our method this increase might be too small. However, we notice that this case has less meaning for our study since the problem of reserving agents is interesting in the case of long service times for background jobs (relatively to calls).

Impact of the service time variability: Consider scenarios 9 to 12. We observe as expected that the increase in the variability in service times deteriorates the system performance. For all policies, the utility decreases in cv (when going from scenario 9 to scenario 12). We also observe that the relative benefit of ATP compared with the constant step methods increases in cv, which means that ATP behaves better against variability than the other policies.

Table 6. Comparison of the methods

	h	$\overline{T}$	$\overline{SL}\,\%$	$\bar{r}$	U		h	$\overline{T}$	$\overline{SL}\,\%$	$\bar{r}$	U
Sc 1	$M1$	1.01	79.9	0.0038	0.62	Sc 2	$M1$	1.25	80.45	0.0080	0.44
	$M2$	0.99	79.9	0.0037	0.62		$M2$	1.13	81.2	0.0068	0.45
	$M3$	1.45	72.6	0.0743	−5.98		$M3$	1.59	68.0	0.0639	−4.79
	$M4a$	1.01	80.9	0.0041	0.59		$M4a$	1.24	80.7	0.0090	0.34
	$M4b$	1.06	80.0	0.0029	0.76		$M4b$	1.20	80.2	0.0099	0.22
	0.1	1.17	80.6	0.0046	0.71		0.1	1.53	72.15	0.0782	−6.3
	0.2	1.12	80.5	0.0036	0.77		0.2	1.38	78.7	0.0201	−0.63
	0.5	1.04	80.1	0.0032	0.72		0.5	1.23	81.4	0.0063	0.60
	1	0.98	80.0	0.0035	0.63		1	1.19	80.7	0.0062	0.57
	ATP	1.09	80.7	0.0027	0.82		ATP	1.12	85.6	0.0008	1.04

4.3. *Comparison with other intuitive methods*

In this section, we compare ATP with other intuitive adaptive methods. We propose the following ones based on the reevaluation of the step h_i after each intervals i ($i = 1, \ldots, N$).

Method 1: The first intuitive idea is to propose a decision based on the distance from the achieved service level and the target after each interval. The intuition is that the need to change the threshold increases with this distance. We initialize with $h_0 = 0$, $c_0 = u_0 = s$, and $SL_0 = 100\%$. For $i \in \{0, \ldots, N-1\}$, we obtain h_i according to

$$h_{i+1} = |SL_i - \alpha|.$$

Method 2: Method 2 is a variation of Method 1. We propose a decision based on the cumulative distance with the service level target, α. The intuition is that the need to change the threshold not only increases with the distance to the target service level but also increases with the time spent above or under this target. More precisely, we initialize by $h_0 = 0$, $c_0 = u_0 = s$, and $SL_{-1} = SL_0 = 100\%$. For $i \in \{0, \ldots, N-1\}$, we obtain h_i according to

$$h_{i+1} = \text{Min}\,\{1, h_i + |SL_i - \alpha|\} \times \mathbf{1}_{(SL_i-\alpha)(SL_{i-1}-\alpha)>0}.$$

Method 3: We propose the same evaluation of h_i as in Method 2 but instead of using the service level SL_i of the last i intervals ($i = 1, \cdots, N$), we use the service level measured only on the last interval i ($i = 1, \cdots, N$). This method is made to correct a too important weight that could be given to the past in the previous method.

Methods 4a and 4b: Methods 4a and 4b are not really intuitive. The idea behind them is the question of when the strongest decisions in the change of the threshold should be taken. If we choose the strongest changes in the threshold at the beginning of the period we could quickly reach the service level constraint (Method 4a). If we choose the strongest changes at the end of the period we could maximize the email throughput at the beginning and do an efficient correction at the end of the working period to reach the service level constraint (Method 4b). More precisely, in Method 4a we propose after i intervals to choose

$$h_i = 1 - \frac{i}{N},$$

and in Method 4b we choose

$$h_i = \frac{i}{N},$$

for $i = 1, \cdots, N$.

We compare the proposed methods in Table 6 with the constant step size methods and ATP under scenarios 1 and 2. We observe that those methods are not as good as ATP and even sometimes not as good as the constant step size methods. Methods 4a and 4b are not efficient for a simple reason; the choices in changing the threshold mainly depend on the demand and not on the closeness to the end of the working day. We observe on other simulations that Method 4a is efficient when the variability in the demand is high at the beginning of the working period and the opposite is true for Method 4b. Although Methods 1 and 2 are the most intuitive, we observe that they are not efficient. The weight of the past is too heavy and entails extreme choices in the threshold (which are often inefficient) to compensate the past values. Method 3 is often more efficient in terms of the email throughput; however, it converges very slowly. We observe on other simulations that Method 3 could be a good proposition only if a working day is long enough (at least 1000 hours). An intermediate solution between Methods 2 and 3 would be to propose a decision in the changes of the threshold based on the average of the service levels measured on all past intervals weighted by coefficients that, are increasing with proximity to the last interval. Many solutions can be proposed in that direction but none of them seems to be efficient for a representative number of scenarios.

5. Conclusions and future research

We considered call centers with inbound calls and an infinite supply of emails. We proposed a scheduling policy, refered to as ATP, where the objective is to do as many emails as possible while satisfying a service-level constraint

on the call waiting time. In the real-life call center context with a fluctuating call arrival rate, the assignment policy for emails adapts itself to the current service level. We showed the efficiency of ATP by comparing its performance with that of other policies. One of the main advantages of ATP is its ability to quickly react when an important change in the arrival process happens and also its ability to avoid inefficient states when the arrival rate remains constant.

Future research on this subject may follow two directions. First, a theoretical modeling for the adaptive blending might be useful to better understand ATP. This is now hindered by the fact that no theory seems to exist on this type of control problems. One of the difficulties in building a Markov chain is the non-exponentiality of the decision interval length defined in the ACD. Another difficulty is the lack of transient results for the performance measures of call center queueing models. Second, the complexity of a real-life call center has been partly avoided in our study. Features such as abandonments, retrials, different types of inbound calls, switching times between different tasks, and a finite number of back office tasks are important, but including them would considerably complicate the analysis.

Acknowledgements

This work was supported by the Agence Nationale de la Recherche under the project ANR-JCJC-SIMI3-2012-OPERA. We also want to express our gratitude to three anonymous reviewers and the Associate Editor for their useful comments that significantly improved this article.

References

Akşin, O., Armony, M. and Mehrotra, V. (2007) The modern call-center: a multi-disciplinary perspective on operations management research. *Production and Operations Management*, **16**, 665–688.

Armony, M. and Ward, A. (2010) Fair dynamic routing in large-scale heterogeneous-server systems. *Operations Research*, **58**(3), 624–637.

Bhulai, S. and Koole, G. (2003) A queueing model for call blending in call centers. *IEEE Transactions on Automatic Control*, **48**, 1434–1438.

Deslauriers, A., L'Ecuyer, P., Pichitlamken, J., Ingolfsson, A. and Avramidis, A. (2007) Markov chain models of a telephone call center with call blending. *Computers & Operations Research*, **34**(6), 1616–1645.

Gans, N., Koole, G. and Mandelbaum, A. (2003) Telephone call centers: tutorial, review, and research prospects. *Manufacturing & Service Operations Management*, **5**, 73–141.

Gans, N. and Zhou, Y.-P. (2003) A call-routing problem with service-level constraints. *Operations Research*, **51**, 255–271.

Green, L., Kolesar, P. and Soares, J. (2001). Improving the SIPP approach for staffing service systems that have cyclic demands. *Operations Research*, **49**(4), 549–564.

Koole, G. (2013). *Call Center Optimization*, MG Books.

Milner, J. and Olsen, T. (2008). Service-level agreements in call centers: perils and prescriptions. *Management Science*, **54**, 238–252.

Queffélec, H. and Zuily, C. (2013). *Analyse pour l'Agrégation*. [Mathematical analysis for the aggregation examination.] Collection: Sciences Sup, Dunod, Paris.

Biographies

Benjamin Legros is a postdoc researcher in Operations Management at Laboratoire Génie Industriel, Ecole Centrale Paris. He received a B.Sc. degree in Industrial Engineering from Ecole Centrale Paris in 2006. He carried out his Ph.D research on the optimization of multi-channel Call Centers at Ecole Centrale Paris and received a Ph.D degree in 2013. His current research interests are in stochastic modeling and operations management of call centers.

Oualid Jouini is an assistant professor in Operations Management at Laboratoire Génie Industriel, Ecole Centrale Paris. He received a B.Sc. degree in Industrial Engineering from Ecole Nationale d'Ingénieurs de Tunis in 2001 and an M.Sc. degree in Industrial Engineering from Ecole Centrale Paris in 2003. He carried out his Ph.D research on Operations Management in Call Centers at Ecole Centrale Paris and received a Ph.D degree in 2006. His current research interests are in stochastic modeling and service operations management. His main application area is call centers and healthcare systems.

Ger Koole is full professor at VU University Amsterdam. He graduated in Leiden on a thesis on the control of queueing systems. Since then he held post-doc positions at CWI Amsterdam and INRIA Sophia Antipolis. His current research is centered around service operations, especially call centers, health care, and, more recently, revenue management. Dr. Koole is founder of a call center planning company, a software company active in the area of online marketing, and of PICA, the VU University/medical center joint knowledge center on health care operations management. He teaches on the theory and applications of stochastic modeling at all levels, from Ph.D. students to professionals in call centers and hospitals.