Noname manuscript No.
(will be inserted by the editor)

Profit-oriented shift scheduling of inbound contact centers with skills-based routing, impatient customers, and retrials

Stefan Helber and Kirsten Henken

Leibniz Universität Hannover, Institut für Produktionswirtschaft, Königsworther Platz 1, 30167 Hannover, Germany
e-mail: stefan.helber@prod.uni-hannover.de

October 30, 2007, revised March 31, 2008

Abstract This paper presents a profit-oriented shift scheduling approach for inbound contact centers. The focus is on systems in which multiple agent classes with different qualifications serve multiple customer classes with different needs. We assume that customers are impatient, abandon if they have to wait, and that they may retry. A discrete-time modeling approach is used to capture the dynamics of the system due to time-dependent arrival rates. Staffing levels and shift schedules are simultaneously optimized over a set of different approximate realizations of the underlying stochastic processes to consider the randomness of the system. The numerical results indicate that the presented approach works best for medium-sized and large contact centers with skills-based routing of customers for which stochastic queueing models are rarely applicable.

Key words: Call center, contact center, workforce scheduling, shift scheduling, stochastic programming

1 Introduction

Contact centers are the multi-channel successors of phone-based call centers. Customers can use phone, fax, e-mail etc. to reach the agents working in an inbound contact center in order to receive some kind of service. Contact centers have become the prevalent instrument of customer service in many industries. This paper treats the problem to determine shift schedules for the different agent classes working in an inbound contact center over the course of a day.

The first step of the traditional approach to solve this problem is to divide the day into separate intervals, often with a length of 30 minutes.

In the so-called *stationary independent period-by-period* (SIPP) approach or variants thereof (Green et al., 2001, 2007), these intervals are treated in isolation under the assumption that the system is never overloaded. Stationary queueing models are then used to determine staffing needs for each time interval given the projected workload and a required service level. After the staffing needs for each agent class and time interval have been determined, one seeks the number of agents working on the different shifts to meet these requirements at minimum cost. The last step, often called rostering, is to assign personnel to the determined shifts.

Apart from all problems related to forecasting call arrivals (Aksin et al., 2007, Sect. 2.1), this approach is accompanied by several problems: Firstly, it ignores that staffing levels and hence also service levels during different intervals are interrelated because agents work according to shifts which span multiple intervals. This cannot be considered if requirements planning and shift scheduling are separated into two subsequent planning steps. Secondly, if customers are impatient, hang up and retry or unanswered e-mails are carried over into subsequent intervals, the periods are also not independent as assumed in the SIPP approach (Jiménez and Koole, 2004; Stolletz, 2007). Thirdly, in order to use the SIPP approach, a stochastic queueing model (or a time-consuming simulation) is required. Even the most tractable Markovian models suffer from an explosion of the state space if multiple customer and agent classes as well as retrials are considered. Only rather small call centers with skills-based routing (SBR) can be analyzed, often under the restrictive assumptions of Markovian queueing models (e.g. Stolletz, 2003; Stolletz and Helber, 2004). Fourthly, even within a 30-minute period, the call arrival rate can change significantly such that the system may hardly reach a stationary or steady state based on the average call arrival rates for the period.

In this paper, we assume that for the agents a set of possible shifts is given and that for each class of agents the number of agents assigned to each shift is sought. We allow for arrival rates to change continuously over the day, whereas the number of agents on service can only be changed at distinct moments in time, due to the predefined shift types. Difference equations with time periods on the order of magnitude of a minute are used to model system dynamics. In order to determine shift schedules, we solve a linear mixed-integer optimization model using a standard solver. We found that unlike in many other approaches for contact center shift scheduling, computation times *decrease* as the system size increases, making large systems particularly easy to solve. To incorporate randomness into the model, we perform the optimization of the shift schedule over a set of different scenarios simultaneously. This leads to a kind of simulation optimization approach. It yields plans which are to some extent robust with respect to the randomness of the problem. The numerical results indicate that the method performs best for medium-sized and large call centers with SBR. Our approach has four important features:

Firstly, we model systems in which the unfinished workload of one interval of the day is carried over to the subsequent intervals. Thus we treat both the case of waiting customers that hang up to retry later and the case of incoming e-mails that are served when the call volume decreases. In such a setting, the different periods of the day cannot be treated in isolation. For this reason, we determine staffing requirements and shift schedules simultaneously. Secondly, we focus on systems where multiple customer classes are served by multiple agent classes such that a given customer class can possibly be served by multiple agent classes and vice versa. The routing of customers to agents is based on priorities. Both the customer and the agent classes differ with respect to their particular operational and economical parameters. Thirdly, our stochastic integer programming approach is completely numerical. Like a discrete-event simulation, it requires neither a theoretical analysis of a probabilistic queueing model nor any particular assumptions about distributions of random variables. In this paper, however, we only study the case of inhomogeneous Poisson arrivals and service times that are independent and either deterministic or exponentially distributed. Finally, the objective is to find shift schedules that maximize the profit from the operation, possibly subject to approximate service level constraints, taking into account cost and revenue of the served contacts.

We are not aware of a paper that combines all of these four features. General reviews of the vast technical literature on call or contact centers are given by Gans et al. (2003); Aksin et al. (2007) and, with a particular emphasis on queueing models, by Koole and Mandelbaum (2002). The literature on staffing and shift scheduling for contact centers is quite limited once the practically important aspects of either abandonment and retrials or multiple customer and agent classes are considered. Stationary queueing models or discrete-event simulations are usually used to evaluate any given staffing level or shift schedule. In order to optimize staffing levels or shift schedules, usually either integer programming, local search or some metaheuristic is applied. Apart from the above-mentioned problems of stationarity, such queueing models often rely on the rather questionable assumption of exponentially distributed processing times.

If one wants to model the waiting of multiple customer classes, the state space of a queueing model grows exponentially. A possible remedy is to use stationary blocking models (Franx et al., 2006) of multi-skill call centers. However, this excludes modeling the carry-over of backlog such as unanswered e-mails or call retrials. A different strategy is to combine discrete-event simulation with a cutting plane approach for integer programming (Atlason et al., 2004, 2008). Here the idea is to find cost-minimizing staffing levels that meet a given service level within an integer program, for example with respect to the fraction of calls that are answered within a time limit. In an iterative approach a discrete-event simulation is used to determine whether a tentative schedule meets this service level requirement. Otherwise, simulation is used to calculate a subgradient of the service level function at that point. This subgradient leads to a cutting plane that is added to the integer program to exclude the current tentative solution from the solution space. The problem is then solved again until all service level requirements are met. A practical problem of this approach is that simulation times increase as call volume and system size increase, and hence only a very small example with few periods is presented in Atlason et al. (2004). In Cezik and L'Ecuyer (2007), this approach is extended to the multi-skill setting. However, due to the computational effort to simulate larger and more complex systems, Cezik and L'Ecuyer (2007) study only the single period staffing problem, as opposed to the small multi-period shift scheduling problem considered in Atlason et al. (2004).

Avramidis et al. (2007) develop a two-stage search method to solve the single-period staffing problem for the multi-skill case. In the first stage, a so-called "loss-delay" approximation based on a specific overflow routing is developed to determine analytic estimates of service levels per customer class. This approximation is used within a neighborhood search to direct the search to cost-efficient solutions that respect the service level constraints at least roughly. The second stage uses a more accurate but also more time-consuming simulation to correct a remaining infeasibility from the first stage and/or to reduce the cost of the solution. Avramidis et al. (2007) compare their approach to the one presented in Cezik and L'Ecuyer (2007) and conclude that "... none of the two methods always dominates the other...". The effort to simulate the call center several times is substantial and increases with the size of the call center so that computation times can take (several) hours for a single period.

Shift scheduling of a homogeneous call center with an overall service level constraint is studied by Koole and Van der Sluis (2003) via local search. Ingolfsson et al. (2003) also treat the homogenous case and combine integer programming with the randomization (or uniformization) method to analyze the transient behavior of the system. This approach is limited to customer arrivals according to an inhomogeneous Poisson process and to exponentially distributed processing times. Harrison and Zeevi (2005) and Bassamboo et al. (2006) treat the problem to determine both a single staffing level and a dynamic allocation of servers to activities for a time period during which *average* arrival rates are uncertain and dynamic.

Bhulai et al. (2008) present a two-step approach to solve the staffing (Step 1) and shift scheduling (Step 2) problem for multi-skill call centers of a realistic size. In Step 1, stationary blocking models (Franx et al., 2006) of multi-skill call centers are used to determine staffing levels for each interval. Given these required staffing levels, shift schedules are created via integer programming in Step 2 to meet these staffing requirements. For multi-skill agent groups, this includes the decision which skill set is actually used in a given period. The approach by Bhulai et al. (2008) is probably the first practically applicable shift scheduling approach for large and heterogenous contact centers that deals with randomness in a systematic way. However, it suffers from the above-mentioned problem to separate the staffing level decision from the shift scheduling decision and is therefore unable to deal with an intertemporal workload carry-over because of retrying customers. In addition, it assumes an exogenously given service level. However, if calls generate revenues, it is economically beneficial to compute the time-dependent profit-maximizing service level endogenously, even though this may not be the current practice of call center management. Furthermore, their method relies on assumptions about the distribution of processing times in the blocking model which our method does not require. On the other hand, our model does not allow to specify the fraction of calls that are answered with a time limit. This makes a direct quantitative comparison of the approaches difficult. However, we can impose approximate limits on the average waiting times and the fraction of served calls.

Avramidis et al. (2007) extend the method proposed in Cezik and L'Ecuyer (2007) to the multi-period shift scheduling problem and compare their results to those obtained by the two-stage approach presented by Bhulai et al. (2008). Given that simulation is used iteratively with linear programming, it is not surprising that computation times are substantial and increase with the size of the call center. As opposed to Bhulai et al. (2008), Avramidis et al. (2007) treat the staffing and the shift scheduling problem simultaneously (like we do in this paper) and report (as can be expected) solutions that are better then those by Bhulai et al. (2008) in their more traditional two-step approach. They do not model call retrials as we do in this paper. Another both interesting and practically important difference is that the numerical effort and accuracy of our method *decreases* as the size of contact center increases.

In Henken (2007), the profit-oriented shift-scheduling problem for a contact center with two customer and three agent classes is solved heuristically, based on deterministic dynamic fluid models. This overestimates the profit from the operation of a stochastic system and yields schedules that are not very robust with respect to the randomness in the system. In addition, the heuristic optimization procedure in Henken (2007) is purely profit-oriented and does not reflect service level requirements.

The completely different approach of a commercial software package based on "artificial intelligence" search techniques is characterized in a non-technical way by Fukunaga et al. (2002). The details of the approach are not revealed and the performance of this approach relative to other approaches is not reported.

The remainder of this paper is organized as follows: In Section 2 the contact center model is explained in detail and we outline the difference equations that describe the dynamics of the number of customers in the system. The basic shift schedule optimization model is presented in Section 3. There we also discuss several options to incorporate different realizations of the original stochastic processes in the deterministic model. Numerical results are discussed in Section 4. The paper concludes with comments on the managerial implications of the results and suggestions for further work in Section 5.

2 Modeling heterogeneous contact centers

2.1 System description

We study contact centers with multiple customer or contact classes $c \in \mathcal{C}$ and multiple agent classes $a \in \mathcal{A}$. Let $\mathcal{C}_a \subseteq C$ denote the set of customer classes c that can be served by agent class a and let $\mathcal{A}_c \subseteq A$ denote the set of agent classes a that can serve customer class c. In Figure 1, an example of such a contact center with two customer classes and three agent classes is depicted. In this system, each customer class is served by a specialized class of agents. A third class of flexible generalists can serve both customer classes. We assume that the waiting space for each customer class is unlimited. This assumption is reasonable for phone calls as the number of phone lines usually exceeds the number of active agents and waiting customers usually hang up after some time. The waiting space for e-mails can also usually be considered to be "practically" infinite. The number of agents of class a serving at time t is denoted by N_{at}. It is the result of the shift schedule.

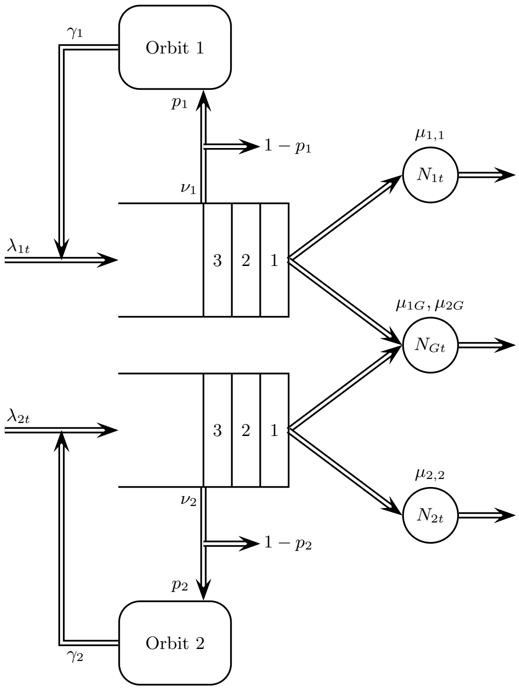

Fig. 1 M-designed contact center with retrials (Henken, 2007)

Customers of class c arrive with a time-dependent rate λ_{ct} at time t. They are served by agents of class a with rate μ_{ca}. Waiting customers abandon with rate ν_c. After abandoning, they join the "orbit" of customers who will retry with probability p_c. Customers in the orbit retry with rate γ_c. Our methodology does not require any particular distribution of interarrival times, processing times, times to abandon and times to retry. This yields a substantial degree of freedom to use the one probability distribution that best matches the empirical data, instead of the one for which a stochastic (possibly Markovian) queueing model can be solved analytically.

In a contact center with multiple customer and agent classes, routing rules are needed. Often routing is based on static priorities. Assume that a customer arrives while agents of different agent classes that could serve this customer idle. In this case the problem of agent class selection arises. Let $pr1_{ac}$ be the agent class selection priority for agent class a and customer class c. A smaller value of $pr1_{ac}$ indicates a higher priority to route arriving customers of class c to idle agents of class a.

Now assume that an agent finishes a service while customers of different classes that can be served by this agent are waiting. In this situation the problem of customer class selection emerges. Let $pr2_{ac}$ be the customer class selection priority. A smaller value of $pr2_{ac}$ indicates a higher priority for idle agents of class a to serve waiting customers of class c. These assumptions allow to model a broad variety of contact center topologies that can be found in practice. While static priority rules are often found in practice due to their simplicity, one can expect to find a better performance in systems with dynamic priority rules that reflect the achieved transient performance of the system. However, this adds a substantial amount of complexity to the problem and is therefore beyond the scope of this paper.

In many real-world contact centers, we observe non-preemptive service disciplines, i.e., a service is not interrupted when a customer with a higher priority arrives. The reason is that customers strongly dislike to have their individual service interrupted because a more important customer arrives. In the simulation model used to evaluate our approach, we therefore model non-preemptive service, while our numerical linear programming method implicitly assumes preemptive service. The reason for the implicit assumption of preemptive service is that in the linear program the originally discrete customers are modeled as a fluid, see below. This difference becomes less relevant as contact centers get larger. For the system in Figure 1 we assume that customers give priority to their respective class of specialists and that generalist agents give priority to class 1 customers.

2.2 Approximating a dynamic stochastic system in continuous time via multiple scenarios of difference equations

The generic model of a contact center presented in Section 2.1 describes a system in which discrete events happen randomly in continuous time. System parameters such as arrival rates λ_{ct} as well as the number of available agents N_{at}, which is to be determined, are time-dependent. Both the number of customers in the system and the number of customers in the orbit form stochastic processes in discrete space (as customers can be counted) and continuous time. Instead of using a stochastic queueing model of the system based on probability theory as in the SIPP approach, we use difference equations to describe the dynamic processes in the system. As these dynamic processes are stochastic, we consider multiple different scenarios $s \in \mathcal{S}$ simultaneously.

To motivate this approach, we now concentrate on the dynamics of these stochastic processes and the relationship between system size and process variability. Consider a contact center with a single class of impatient customers that arrive according to an inhomogeneous Poisson process. The time-dependent arrival rate is depicted in Figure 2. Assume that processing times and waiting time tolerances are exponentially distributed with rates 1 and 2 per minute, respectively.

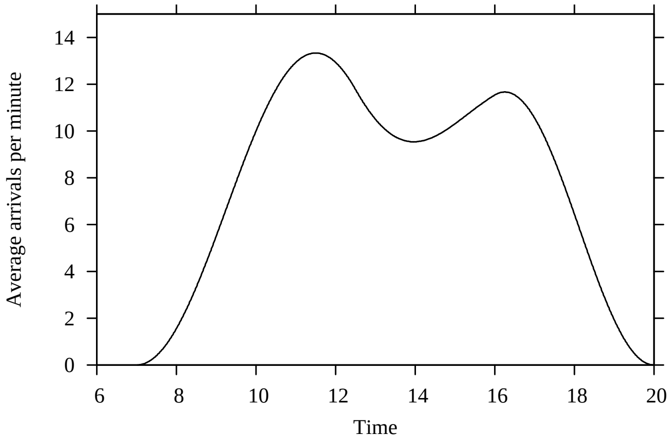

Fig. 2 Time-dependent arrival rate

In a base case we assume that 15 agents are scheduled constantly throughout the day. The graph on the upper left-hand side of Figure 3 shows a simulated sample path realization of the number of customers in the system. It exhibits a substantial degree of variability. Now we scale the system by multiplying both the arrival rate and the number of servers by 10, 100, and 1000. The other three graphs in Figure 3 show the respective sample paths.

As the arrival rate and number of servers increase, the scaled process $Q^{(n)}(t)/n$ of the number of customers divided by the scaling factor n apparently becomes less variable and eventually approaches a (deterministic) fluid limit. Mandelbaum et al. (1998) study fluid and diffusion approximations based on this scaling. If the system gets in a sense "less variable" as its size increases, the relative importance of the system dynamics over the randomness increases. Multiple numerical scenarios s, i.e., independent realizations or samples path of this stochastic process, are required to approximately capture the randomness in the system.

It would be natural to use a set of coupled differential equations in a fluid approximation. However, in order to be able to deal with the system dynamics within a discrete-time linear program, we now directly develop *difference* equations to describe how the state of the system evolves over time. Denote by QC^s_{ct} the number of *waiting* customers of class c at the beginning of a discrete period t, for example a minute, in scenario s. Let ar^s_{ct} denote the (exogenous) primary arrivals in period t, RE^s_{ct} the retrials, AB^s_{ct} the number of customers who abandon and E^s_{cat} the number of customers who exit the system after being served by an agent of class a. Then the dynamics for customer class c in scenario s can be modeled via the following difference equation for period t:

$$QC^s_{c,t+1} = QC^s_{ct} + ar^s_{ct} + RE^s_{ct} - AB^s_{ct} - \sum_{a \in \mathcal{A}_c} E^s_{cat} \qquad (1)$$

The number of served customers E^s_{cat} depends on the number N_{at} of active agents of class a at time t, which is identical over all scenarios and depends on the shift schedule. It also depends on the number mu^s_{cat} of customers of class c served per period t by an agent of type a in scenario s. If a single class of customers c is served by a single class a of agents, the following simple function results:

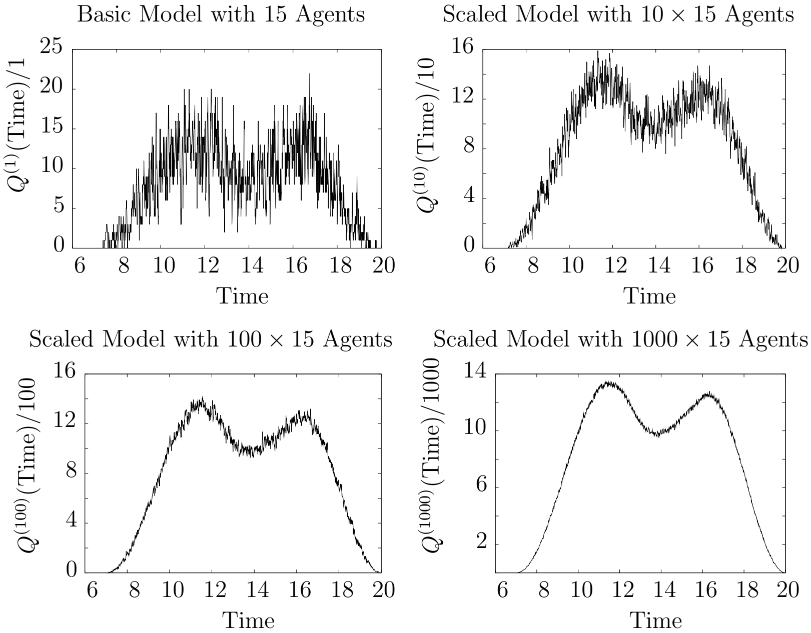

Fig. 3 Scaled number of customers in the system

$$E^s_{cat} = \min\Big(QC^s_{ct} + ar^s_{ct} + RE^s_{ct} - AB^s_{ct},\ mu^s_{cat} N_{at}\Big) \qquad \forall c,t,a,s \qquad (2)$$

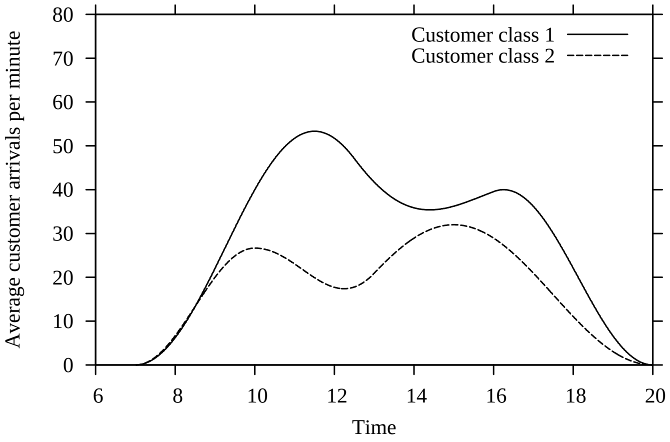

Fig. 4 Average call arrival rates for a contact center

Figure 4 shows for two customer classes fictitious average arrival rate functions and Figure 5 shows sample path realizations for these average arrival rate functions from a simulation run. If a schedule is optimized over several different scenarios $s \in \mathcal{S}$ of call arrivals simultaneously, one can expect to find a solution with some degree of robustness with respect to the uncertainty of call arrivals.

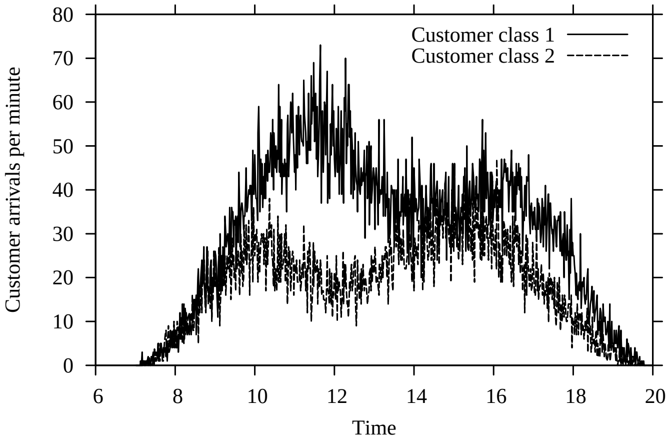

Fig. 5 Sample paths of call arrivals for a contact center

2.3 Shift schedules

We assume that a set of basic shift types $k \in \mathcal{K}$ is given. Table 1 presents an example with 31 shift types. Shift types 1 to 12 are long shifts of 7.5 hours with a half-hour break after 3.5 hours and shift-specific starting times. Shift types 13 to 31 take four hours without a break.

Table 1 Schematic presentation of the basic shift types

Type	Interval i				
k	1 5 7:00 - 9:30	6 10 9:30-12:00	11 15 12:00-2:30	16 20 2:30 - 5:00	21 26 5:00 - 8:00
1	1 1 1 1 1	1 1 1 1	1 1 1 1 1		
2	1 1 1 1	1 1 1 1	1 1 1 1 1	1	
$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$
11			1 1 1 1 1	1 1 1 1	1 1 1 1 1
12			1 1 1 1	1 1 1 1	1 1 1 1 1 1
13	1 1 1 1 1	1 1 1			
14	1 1 1 1	1 1 1 1			
$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$
30				1 1 1	1 1 1 1 1
31				1 1	1 1 1 1 1 1

For each basic shift type k an indicator parameter s_{ki} equals 1 if an agent following this shift type k is on duty during interval i, and 0 otherwise. For example, at t corresponding to 10:45 am, $s_{1,i(t)} = 0$ as the break for shift type $k = 1$ starts at 10:30 am and ends at 11:00 am.

3 The shift schedule optimization model

3.1 Basic optimization model

In this section we present the basic shift schedule optimization model taking multiple scenarios into account. The notation is summarized in Table 2. We use lower case letters for input data and upper case letters for decision variables. In addition to the modeling assumptions and notation presented in Section 2, we assume the following:

- The length of a period in the difference equations is Δt, e.g., a minute. It has to be distinguished from the length of the time intervals of the shifts of, for example, 30 minutes.
- Customers arrive at the system at the beginning of a period. This holds both for retrials and primary arrivals. The number of primary arrivals of customers of class c in period t of scenario s is ar_{ct}^{s} and the number of retrials is RE_{ct}^{s}.

- Individual waiting customers have a waiting time tolerance of $\frac{1}{\nu_c}$ in continuous time, i.e., on average an individual waiting customer hangs up with rate ν_c. The expected number of abandonment events during a time interval of length Δt is hence $\nu_c \Delta t$ per waiting customer of class c. However, for each occasion a customer has to wait, he can hang up only once. In the context of the *discrete* time model, we assume that only those customers already waiting at the beginning of period t can hang up. Those who hang up do so immediately. The fraction of the waiting customers that hang up is $\min(1, nu_c)$ with $nu_c = \nu_c \Delta t$. The minimum function is required in the discrete time model to avoid that more customers abandon than are waiting at the beginning of the period. The fraction p_c of those customers who abandon join the orbit.
- Only those customers already in the orbit at the beginning of period t can retry. Those who retry do so immediately. The fraction of the customers in the orbit that retry is $\min(1, ga_c)$ where $ga_c = \gamma_c \Delta t$ depends on the rate (in continuous time) γ_c at which a single customer in the orbit calls again. Again, the minimum function is required as at most all customers in the orbit can retry.
- Those customers who already waited at the beginning of a period plus those who arrived or retried minus those who abandoned are available to be served in the period.
- Customers that are served leave the system at the end of a period. The number E^s_{cat} of customers of class c that is served by agents of class a in period t of scenario s is the minimum of those that are available to be served and that can potentially be served. The number that can potentially be served depends on the capacity this agent class devotes to customer classes with higher priority.
- Agents work according to shifts $k \in \mathcal{K}$.
- The total number of agents of class a on duty at time t is N_{at}. It depends on the integer number X_{ak} of agents of class a working shift k.
- Only a maximum number $n_a^{\max}$ of agents of class a can be scheduled for the day.
- Each processed customer of class c leads to a deterministic revenue rv_c.
- A customer in the system causes a line cost l_c per period.
- The wage of an agent of class a working according to schedule k is w_{ak}.
- Primary arrivals ar^s_{ct} and the potential number of processed customers per agent mu^s_{ct} are scenario-specific realizations of random variables.
- The objective is to find a shift schedule X_{ak} which maximizes the average profit over the different scenarios.

Table 2 Notation

Sets and indices	
$a \in \mathcal{A}$	set of agent classes, each agent belongs to one class
$\mathcal{A}_c \subset \mathcal{A}$	set of agent classes that can serve customer class c
$c \in \mathcal{C}$	set of customer classes, each customer belongs to one class
$\mathcal{C}_a \subset \mathcal{C}$	set of customer classes that can be served by agent class a
$i \in \mathcal{I}$	time intervals (e.g., half-hours)
$k \in \mathcal{K}$	shift types
$s \in \mathcal{S}$	scenarios (sample paths)
$t \in \mathcal{T}$	time periods (e.g., minutes)
Input data	
ar_{ct}^s	primary arrivals of customer class c in period t of scenario s
Δt	length of a period
$\overline{fs}_c^{\min}$	minimum fraction of served contacts of class c
ga_c	fraction of customers of class c in the orbit that retry during a period
γ_c	retrial rate of customers of class c in the orbit in continuous time
l_c	line cost of class c per time unit
mu_{cat}^s	number of customers of class c that can be served per agent of class a serving this class in period t and scenario s
$n_a^{\max}$	maximum number of available agents of type a
ν_c	abandonment rate of waiting customers of class c in continuous time
nu_c	abandonments $nu_c = \nu_c \Delta t$ per period and waiting customer
p_c	fraction of abandoning customers that are willing to retry
s_{ki}	indicator, equals 1 if an agent working shift type k is active in interval i, 0 otherwise
rv_c	revenue per served contact of class c
w_{ak}	wage of an agent of class a working a shift of type k
$\overline{w}_c^{\max}$	maximum waiting time of contacts of class c
Decision variables	
AB_{ct}^s	real-valued number of abandoning customers of class c in period t of scenario s
E_{cat}^s	real-valued number of customers of class c served by agents of class a in period t of scenario s
N_{at}	integer number of agents of class a on duty in period t
QC_{ct}^s	real-valued number of customers of class c waiting in the system at the beginning of period t in scenario s
QO_{ct}^s	real-valued number of customers of class c in the orbit at the beginning of period t in scenario s
RE_{ct}^s	real-valued number of retrials of customers of class c in period t of scenario s
X_{ak}	integer number of agents of class a working shift type k

This leads to the following optimization problem **P**:

$$\begin{aligned}\text{Max } & \frac{1}{|S|}\sum_{s\in\mathcal{S}}\sum_{c\in\mathcal{C}}\sum_{t\in\mathcal{T}}\left(\left(\sum_{a\in\mathcal{A}_c}(rv_c-\frac{l_c}{mu^s_{cat}})E^s_{cat}\right)-l_cQC^s_{ct}\right)\\ & -\sum_{a\in\mathcal{A}}\sum_{k\in\mathcal{K}}w_{ak}X_{ak}\end{aligned} \tag{3}$$

subject to

$$QC^s_{c,t+1} = QC^s_{c,t}+ar^s_{ct}+RE^s_{ct}-AB^s_{ct}-\sum_{a\in\mathcal{A}_c}E^s_{cat}, \qquad \forall c,t,s \tag{4}$$

$$QO^s_{c,t+1} = QO^s_{c,t}-RE^s_{ct}+p_cAB^s_{ct}, \qquad \forall c,t,s \tag{5}$$

$$AB^s_{ct} = \min(1,nu_c)\,QC^s_{ct}, \qquad \forall c,t,s \tag{6}$$

$$RE^s_{ct} = \min(1,ga_c)\,QO^s_{ct}, \qquad \forall c,t,s \tag{7}$$

$$E^s_{cat} \le QC^s_{ct}+ar^s_{ct}+RE^s_{ct}-AB^s_{ct}-\sum_{\substack{\tilde{a}\in\mathcal{A}_c\neg\{a\}\\ pr1_{c\tilde{a}}<pr1_{ca}}}E^s_{c\tilde{a}t}, \qquad \forall c,t,a,s \tag{8}$$

$$E^s_{cat} \le mu^s_{cat}\big(N_{at}-\sum_{\substack{\tilde{c}\in\mathcal{C}_a\neg\{c\}\\ pr2_{a\tilde{c}}<pr2_{ac}}}\frac{E^s_{\tilde{c}at}}{mu^s_{\tilde{c}at}})\big) \qquad \forall c,t,a,s \tag{9}$$

$$N_{at} = \sum_{k\in\mathcal{K}}s_{kt}X_{ak}, \qquad \forall a,t \tag{10}$$

$$\sum_{k\in\mathcal{K}}X_{ak} \le n^{\max}_a, \qquad \forall a \tag{11}$$

$$X_{ak} \in\{0,1,2,3,\ldots\}, \qquad \forall a,k \tag{12}$$

$$N_{at} \in\{0,1,2,3,\ldots\}, \qquad \forall a,t \tag{13}$$

$$QC^s_{c,t},\ QO^s_{c,t},\ AB^s_{ct},\ RE^s_{ct} \ge 0, \qquad \forall c,t,s \tag{14}$$

$$E^s_{cat} \ge = 0, \qquad \forall c,t,a,s \tag{15}$$

In the objective function (3) the average profit over the scenarios is calculated by subtracting from the revenue of the processed customers the line cost of customers in service or waiting and the wages for the agents. The balance equations (4) for the number QC^s_{ct} of customers waiting in the contact center reflect retrials RE^s_{ct}, abandonment AB^s_{ct} and the fact that multiple agent classes a serve customer class c. In the balance equations (5) for the number QO^s_{ct} of customers in the orbit, we take into account that only a fraction p_c of the abandoning customers of class c joins the virtual queue in the orbit to retry later. The retrying process in Equations (7) is formally "self-service" in the orbit. The number E^s_{cat} of customers of class c that are served by agents of class a in period t is doubly limited. The first limit in Inequalities (8) is the number of customers of the respective class that are available to be served by a particular agent class. The second limit in Inequalities (9) is the maximum capacity for this combination of customer class c and agent class a. It depends on the number N_{at} of agents of class a available during period t. However, due to the customer class selection priority of the agents, we need to subtract the capacity of this agent class that is already devoted to customer classes with a higher priority. Given the profit-maximization objective of our optimization problem, it is usually not efficient to let an agent idle while customers for that agent are available. For this reason, usually one of the inequalities (8) or (9) will almost always be tight. However, there can be cases where contacts such as e-mails do not generate revenues *and* are perfectly patient *and* and many agents that can deal with these contacts are available that none of the inequalities is tight for a given combination of customer class c, agent class a, period t and scenario s. In Equations (10) the total number of active agents at time t is computed. The upper limit of available agents of each class is represented in Inequalities (11).

This basic model aims at maximizing the profit from the served calls. The profit-maximizing service level with respect to waiting times etc. is hence determined endogenously. If the per-call revenue of a customer class does not exceed the cost per call, no specialized agents for this class will be scheduled and, possibly, no customers will be served. In many real-world applications there is no direct revenue associated with a served call, e.g., for support calls. Therefore the model needs to be extended to enforce some pre-specified level of service for these customer classes.

3.2 Enforcing service level constraints

If one wants to serve the customers even though there are no direct revenues related to each call, an economically rational approach is to minimize cost subject to some exogenously defined service constraint. In our modeling approach, the service quality can be expressed in terms of the average waiting time or the fraction of customers that are served. Both quantities can be limited (from above or below, respectively) for either each single period or the complete planning horizon of a day. Based on discussions with call center managers it is our impression that the fraction of calls that are served is of utmost importance. We define an aggregate measure $\overline{FS}_c$ of the fraction of the primary contacts (not counting retrials) that are eventually served at the end of the day:

$$\overline{FS}_c^s = \frac{\sum_{t\in\mathcal{T}} \sum_{a\in\mathcal{A}_c} E_{cat}^s}{\sum_{t\in\mathcal{T}} ar_{ct}^s}, \qquad \forall c, s \qquad (16)$$

A minimum aggregated fraction of served callers $\overline{fs}_c^{\min}$ in each scenario can be enforced via the following simple constraints:

$$\overline{FS}_c^s \geq \overline{fs}_c^{\min}, \qquad \forall c, s \qquad (17)$$

The instantaneous waiting time for customers arriving during period t can be roughly approximated as the number of waiting customers, divided by the rate at which customers are either served or abandon at that moment in time. (It is an approximation as these rates can change within this instantaneous waiting time.) If m denotes the length Δt of a period t (in seconds), a measure W_{ct}^s of the instantaneous waiting time (in seconds) can be computed as follows:

$$W_{ct}^s = \frac{QC_{ct}^s}{\left(AB_{ct}^s + \sum_{a\in\mathcal{A}_c} E_{cat}^s\right) m^{-1}}, \qquad \forall c, t, s \qquad (18)$$

To compute an aggregate measure $\overline{W}_c^s$ of the waiting time, a weighting factor should reflect the different numbers of customers facing specific instantaneous waiting times during periods of low or high traffic, respectively. We decided *not* to use the relative number of arriving calls as the weighting factor, because it is in practice very difficult to distinguish primary arrivals from retrials. However, it is very easy to measure calls that are served or abandoned. For this reason in our model the instantaneous waiting time is weighted by the relative number of leaving customers, and the following measure of the aggregated waiting time $\overline{W}_c^s$ results:

$$\begin{aligned} \overline{W}_c^s &= \sum_{t\in\mathcal{T}} \frac{\left(AB_{ct}^s + \sum_{a\in\mathcal{A}_c} E_{cat}^s\right)}{\sum_{t\in\mathcal{T}} \left(AB_{ct}^s + \sum_{a\in\mathcal{A}_c} E_{cat}^s\right)} W_{ct}^s \\ &= \frac{\sum_{t\in\mathcal{T}} QC_{ct}^s}{\sum_{t\in\mathcal{T}} \left(AB_{ct}^s + \sum_{a\in\mathcal{A}_c} E_{cat}^s\right) m^{-1}}, \qquad \forall c, s \qquad (19) \end{aligned}$$

A maximum $\overline{W}_c^s \leq \overline{w}_c^{\max}$ of the aggregated waiting time in each scenario s and customer class c can be enforced as follows:

$$\sum_{t\in\mathcal{T}} QC_{ct}^s \leq \overline{w}_c^{\max} \sum_{t\in\mathcal{T}} \left(AB_{ct}^s + \sum_{a\in\mathcal{A}_c} E_{cat}^s \right) m^{-1}, \qquad \forall c, s \qquad (20)$$

In this form the constraint has the *linear* form that can be solved using mixed-integer linear programming.

3.3 Optimizing over the single mean process or multiple scenarios

The model presented in Section 3.1 optimizes a shift schedule over a set $\mathcal{S}$ of scenarios that differ with respect to two parameters, the number of primary arrivals ar^s_{ct} and the processing rate mu^s_{cat} of an agent of class a serving customers of class c. Three different approaches to deal with randomness of these parameters are summarized in Table 3.

Table 3 Different approaches to deal with randomness

Approach	1	2	3
Number of scenarios $\lvert\mathcal{S}\rvert$	1	20 (10)	20 (10)
Arrivals ar^s_{ct}	$\lambda_{ct}\Delta t$	$\sim$POI($\lambda_{ct}\Delta t$)	$\sim$POI($\lambda_{ct}\Delta t$)
Processed customers mu^s_{cat} per period and agent	$\mu_{cat}\Delta t$	$\mu_{cat}\Delta t$	$\sim \frac{\text{POI}(\mu_{cat}\Delta t N^{**}_{at})}{N^{**}_{at}}$
Resulting staffing level	N^{*}_{at}	N^{**}_{at}	N^{***}_{at}

In Approach 1, only the (single) mean process is modeled as both the arrivals and the number of processed customers per agent are set to their respective average values. This leads to the single scenario of a deterministic fluid model. In Approaches 2 and 3, we try to optimize the shift schedule over 20 different scenarios simultaneously. If this does not lead to a solution with an optimality gap of the MIP solver of at most 1% within 1000 seconds, we only consider the first 10 scenarios (out of the 20) and try again. Approach 2 models arrivals as realizations of random variables that are Poisson-distributed with parameter $\lambda_{ct}\Delta t$, whereas processing is deterministic as in Approach 1. In Approach 3, we additionally model Poisson-distributed numbers of customers that can potentially be processed. Here we have to adjust the realization mu^s_{cat} of the number of processed customers *per agent* because multiple agents of class a may work in parallel on customer class c. This leads to a superposition of Poisson departure processes. However, the exact number of these agents is unknown, to be determined within the optimization. We approximate it by the total number of available agents N^{**}_{at} from the solution of Approach 2, compute a realization of a Poisson-distributed random variable with parameter $\mu_{cat}\Delta t N^{**}_{at}$ and normalize this again through a division by N^{**}_{at}. (If Approach 2 did not lead to a solution, we used N^{*}_{at} from Approach 1 instead.) This allows us to have (within the framework of a MIP) numbers of potentially processed customers that are approximately Poisson with a mean that is proportional to the number of agents on duty. Falling back on results from Approach 2 (or 1) can be interpreted as a kind of iterative approach which we found to be reasonable as total staffing levels from the different approaches didn't differ too much.

We did not consider random abandonment or retrials for two reasons: On the one hand, we observed that very often just considering random arrivals already led to quite robust shift schedules so that the additional benefit of treating random processing was quite limited. On the other hand, the "flow" of abandoning and retrying customers is small relative to the flow of primary arrivals and served customers if the contact center offers a good service, so treating random abandonment and retrials should not be expected to improve the performance of the method.

We used a period length of one minute for any period t in our model. There is a tradeoff between numerical accuracy and computation times: If one uses, for example, a second instead of a minute as the period length, the originally discrete-event process is modeled more accurately. On the other hand, the number of real-valued decision variables increases by a factor of 60, which effects the solution times. In a pretest we came to the conclusion that given the computing power that is available to us today, a period length of a minute is not unreasonable.

After some numerical experimenting, we always added a fixed value of 0.0001 to each sampled parameter ar_{ct}^{s} in Approaches 2 and 3 and of 0.00001 to each sampled parameter mu_{cat}^{s} in Approach 3. This helped to avoid numerical difficulties with the solution of the LP relaxation of Problem **P** in Approaches 2 and 3. An alternative way to overcome these feasibility problems is to experiment with the "eprhs" feasibility tolerance parameter of the CPlex software that specifies to which extent a problem's basic variable may violate its bounds.

4 Numerical results

To evaluate the performance of the shift scheduling approach, we performed a systematic numerical study. We studied the M-designed system with two customer classes and three agent classes shown in Figure 1. The shift types were those introduced in Table 1. We assumed sinusoidal average arrival rate functions like those depicted in Figure 4 that were generated by the equation

$$\lambda(t) = \begin{cases} \frac{1}{2}m_1 \cdot \left(1 - \cos\left(2\pi \frac{t-t_0}{t_2-t_0}\right)\right) & \text{for } t_0 \leq t < t_1 \\ \frac{1}{2}m_1 \cdot \left(1 - \cos\left(2\pi \frac{t-t_0}{t_2-t_0}\right)\right) & \\ \quad +\frac{1}{2}m_2 \cdot \left(1 - \cos\left(2\pi \frac{t-t_1}{t_3-t_1}\right)\right) & \text{for } t_1 \leq t < t_2 \\ \frac{1}{2}m_2 \cdot \left(1 - \cos\left(2\pi \frac{t-t_1}{t_3-t_1}\right)\right) & \text{for } t_2 \leq t < t_3 \end{cases} \tag{21}$$

with the parameters in Table 4. All other parameters are also presented in this table with the exception of the hourly wage, which was assumed to be 15 for the specialists and 18 for the generalists. Given the identical productivity, generalists are therefore 20% more expensive than specialists. We assumed that there is no limit on the number of agents for each class, i.e, $n_a^{\max} = \infty$. The length Δt of a period t was set to 60 seconds.

Table 4 Problem parameters

Customer class	1 (Sales)	2 (Support)
Call arrivals		
t_0	7 am	7 am
t_1	12:30 pm	10 am
t_2	4 pm	1 pm
t_3	8 pm	8 pm
m_1 (S/M/L)	(200/800/3200)	(100/400/1600)
m_2 (S/M/L)	(150/600/2400)	(120/480/1920)
Processing rates μ_{ca}		
Type-1 specialists	$12h^{-1}$	-
Type-2 specialists	-	$12h^{-1}$
Generalists	$12h^{-1}$	$12h^{-1}$
Abandonment rates ν_c	$240h^{-1}$	$12h^{-1}$
Retrial rates γ_c	$0.5h^{-1}$	$4h^{-1}$
Retrial probability p_c	0.5	0.5
Per call revenue rv_c	10.0	1.3 (Cases 1-3) or 0.0 (Cases 4-6)
Hourly line cost l_c	6.0	0.0

Of particular interest is the distinction between small, medium-sized and large contact centers. For this reason, we scaled the arrival rate function (21) of the small contact center (S) by a factor of 4 to generate the arrival rate function for the medium-sized contact center (M), and by a factor of $4^2 = 16$ for the large center (L). This resulted in systems with peak numbers of active agents between 20 and 30 for the small system, 80 and 90 for the medium-sized system, and 370-390 in the large system. With respect to the two customer classes we chose the parameters such that the first customer class is always highly profitable, but customers are very impatient. This models a sales channel. The other class models customers that generate very little profit directly or no profit at all, but are much more patient, as in a support channel. For each size class (S, M, or L) we studied six cases 1 - 6 with specific parameters reported in Table 5. In cases 1 to 3 we assumed that the per-call revenue of a support call slightly exceeds the direct per-call cost of this call, if served by the least costly agent class (the specialists). In cases 4 to 6 we always assumed that support calls do not generate any direct revenue and therefore enforced a minimum $\overline{FS}_2^{\min}$ of the aggregated fraction of served calls or a maximum $\overline{W}_2^{\max}$ on the aggregated waiting time. For each system size and case all three approaches (see Table 3) were applied. We used CPlex 10.0 on a 3 GHz Pentium 4 PC with 4 GB

Table 5 Revenue for support customers and service level limits

Case (S, M, L)	1	2	3	4	5	6
rv_2	1.3	1.3	1.3	0	0	0
$\overline{FS}_2^{\min}$ [%]	0.0	0.0	99.9	0.0	90.0	99.9
$\overline{W}_2^{\max}$ [sec.]	∞	60.0	∞	60.0	∞	∞

RAM to solve the models. The branch&bound process was aborted if an integer solution was known to be at most 1% away from the optimum of the MIP model or the computation time limit of 1000 seconds per approach and number of scenarios (20 or 10 in Approaches 2 and 3) was exceeded. Based on a suggestion by one referee, we alternatively tried to speed up the branch&bound process by solving an LP relaxation of our problem **P**, rounding up fractional X_{ak} variables and solving the problem again for the remaining continuous variables. This should result in a first feasible solution to be used in a "warm start" of the CPlex solver and speed up the solution process by cutting off a part of the solution space. However, in particular for the small call center the solution of the LP relaxation very often resulted in very small fractional X_{ak} values so that rounding up led to very weak bounds. In general, this rounding approach did not work better than our "standard" approach.

The shift schedules resulting from the optimization model were then evaluated via a discrete event simulation model coded in C++ which is based on a simulation model used in Feldman (2004); Feldman et al. (2008). For each system, 1000 replications were made to compute 95%-confidence intervals of the profit with a relative half-width always below 0.25%.

With respect to the algorithmic performance of the approach the numerical results show the following: Modeling the contact center via a fluid model (Approach 1) always led to a MIP model that could be solved. In Tables 6 to 8 we present for each combination of problem case and solution approach the (average) profit as computed by the MIP solver as well as the simulated profit for the computed shift schedule. The next line presents the relative deviation of the MIP objective function value from the simulation results. The simulated values of the fraction of served calls $\overline{FS}_2$ and the aggregated waiting time $\overline{W}_2$ of the second customer class (support channel) are reported below. If we compare the results for Approach 1 in Tables 6 to 8, we see that for the small contact center, the fluid approach substantially overestimates the profit that is associated with the proposed shift schedule, while for the large contact center the deviation is only in the area of 4-5%.

Approach 2 (using multiple sample paths of call arrivals) apparently led to a much better estimate of the profit, even for the small contact center. In addition, the resulting shift schedule, when evaluated via simulation, turned out to be better than those from Approach 1. It is also interesting to observe, that in this approach the service level limits for customer class 2 were met much better than via Approach 1. In the schedules computed via Approach 1, usually relatively few flexible generalist agents are scheduled, as these

Table 6 Results for the small contact center

Case	1S	2S	3S	4S	5S	6S
Approach 1						
Profit (MIP)	11941	11914	11837	10925	10925	10680
Profit (SIM)	9882	9979	10009	8938	8938	8892
Rel. Dev. [%]	20.84	19.39	18.26	22.23	22.23	20.10
$\overline{FS}_2$ [%]	83.40	90.32	93.04	87.00	87.00	92.50
$\overline{W}_2$ [sec.]	84.45	52.27	38.49	68.11	68.11	41.84
Approach 2						
Profit (MIP)	10719	10819	9151	9581	9674	-
Profit (SIM)	10280	10293	9838	9192	9230	-
Rel. Dev. [%]	4.27	5.11	-6.98	4.23	4.81	-
$\overline{FS}_2$ [%]	92.42	96.34	99.89	95.85	96.60	-
$\overline{W}_2$ [sec.]	41.85	21.01	0.65	23.66	19.67	-
Approach 3						
Profit (MIP)	9784	-	-	8650	8561	-
Profit (SIM)	10268	-	-	9178	9241	-
Rel. Dev. [%]	-4.71	-	-	-5.75	-7.36	-
$\overline{FS}_2$ [%]	91.11	-	-	96.63	96.60	-
$\overline{W}_2$ [sec.]	48.62	-	-	19.56	19.58	-
Dev Best						
Appr1 [%]	-3.88	-3.05	0.00	-2.76	-3.28	0.00
Appr2 [%]	0.00	0.00	-1.71	0.00	-0.12	-
Appr3 [%]	-0.12	-	-	-0.15	0.00	-
AvScnOpt	9945	9824	-	8824	8705	-
AvUB	9995	9874	-	8869	8749	-
RelDev Appr3 [%]	2.11	-	-	2.46	2.15	-

are assumed to be 20% more expensive than the specialists. The schedules resulting from Approach 2, however, are usually more robust and hence yield a higher average profit in the simulation than those from Approach 1 as they substitute specialists by generalists.

The additional effort to consider sample path realizations of processing rates in Approach 3, however, had a rather limited additional benefit. For the small contact center, three out of the six cases could not be solved within the given time limit, whereas all cases for the medium-sized and large center were solvable. The lower part of Tables 6 to 8 (Dev Best) shows

Table 7 Results for the medium-sized contact center

Case	1M	2M	3M	4M	5M	6M
Approach 1						
Profit (MIP)	48128	48128	47777	43763	43673	43101
Profit (SIM)	43636	43636	44067	39548	39395	39546
Rel. Dev. [%]	10.29	10.29	8.42	10.66	10.86	8.99
$\overline{FS}_2$ [%]	93.76	93.76	96.55	92.26	92.89	96.55
$\overline{W}_2$ [sec.]	34.98	34.98	19.85	42.86	39.60	19.85
Approach 2						
Profit (MIP)	46263	46082	42929	-	41875	38159
Profit (SIM)	42952	44154	44235	-	39551	39564
Rel. Dev. [%]	7.71	4.37	-2.95	-	5.88	-3.55
$\overline{FS}_2$ [%]	88.24	96.38	99.93	-	96.15	99.92
$\overline{W}_2$ [sec.]	62.81	20.82	0.43	-	21.98	0.49
Approach 3						
Profit (MIP)	44637	44548	38837	39040	40217	33293
Profit (SIM)	43471	44338	41317	41063	39850	35910
Rel. Dev. [%]	2.68	0.47	-6.00	-4.93	0.92	-7.29
$\overline{FS}_2$ [%]	92.25	97.14	100.00	95.14	97.57	100.00
$\overline{W}_2$ [sec.]	42.92	16.46	0.01	27.68	14.12	0.00
Dev Best						
Appr1 [%]	0.00	-1.58	-0.38	-3.69	-1.14	-0.04
Appr2 [%]	-1.57	-0.41	0.00	-	-0.75	0.00
Appr3 [%]	-0.38	0.00	-6.60	0.00	0.00	-9.24
AvScnOpt	45099	44794	39047	40611	40640	34495
AvUB	45324	45018	39242	40814	40843	34667
RelDev Appr3 [%]	1.52	1.04	1.03	4.35	1.53	3.96

the deviation from the best schedule over all three approaches. In general, Approach 2 performed best. However, for the large contact center even the pure fluid Approach 1 was almost as good.

Given the randomness of interarrival and processing times, we can expect that a shift schedule which maximizes the average profit over a set of different and stochastically independent scenarios is (ex post) suboptimal for each single scenario if this scenario is treated in isolation. This is a typical problem of stochastic programming with integer recourse. If we treat a scenario in isolation and determine the scenario-specific optimal shift schedule,

Table 8 Results for the large contact center

Case	1L	2L	3L	4L	5L	6L
Approach 1						
Profit (MIP)	192774	192764	191276	176109	175857	172574
Profit (SIM)	184302	184113	184492	167052	166614	166009
Rel. Dev. [%]	4.60	4.70	3.68	5.42	5.55	3.95
$\overline{FS}_2$ [%]	96.81	96.82	98.64	95.69	95.96	98.64
$\overline{W}_2$ [sec.]	18.51	18.41	8.00	24.69	23.23	8.00
Approach 2						
Profit (MIP)	190105	189619	183207	173043	172597	164057
Profit (SIM)	183504	183840	184976	166268	166258	166284
Rel. Dev. [%]	3.60	3.14	-0.96	4.07	3.81	-1.34
$\overline{FS}_2$ [%]	92.34	96.12	99.92	95.92	96.10	99.93
$\overline{W}_2$ [sec.]	42.56	22.30	0.46	23.46	22.38	0.44
Approach 3						
Profit (MIP)	187938	186877	177342	170891	170170	158853
Profit (SIM)	183775	184152	181740	166224	166401	163314
Rel. Dev. [%]	2.27	1.48	-2.42	2.81	2.26	-2.73
$\overline{FS}_2$ [%]	89.90	96.61	99.99	96.46	96.87	99.99
$\overline{W}_2$ [sec.]	54.96	19.62	0.05	20.51	18.15	0.06
Dev Best						
Appr1 [%]	0.00	-0.02	-0.26	0.00	0.00	-0.17
Appr2 [%]	-0.43	-0.17	0.00	-0.47	-0.21	0.00
Appr3 [%]	-0.29	0.00	-1.75	-0.50	-0.13	-1.79
AvScnOpt	187906	187870	178393	171050	170760	159964
AvUB	188845	188809	179285	171905	171614	160764
RelDev Appr3 [%]	0.48	1.02	1.08	0.59	0.84	1.19

we also compute a scenario-specific upper bound on the objective function value. In order to assess the average quality of our solutions resulting from Approach 3 (where both interarrival and processing times are realizations of random variables), we therefore determined for each case an *average upper bound* of the MIP by averaging over the objective function values of the (isolated or ex-post) solutions to 100 independent scenarios. In the bottom part of Tables 6 to 8 we report these average objective function values (AvScnOpt). We terminated the optimization when the optimality gap was at most 0.5%. Multiplying AvScnOpt by 1.005 therefore gives an average

Table 9 Average computation times of the MIP solver (seconds)

Approach \ System size	Small (S)	Medium (M)	Large (L)
1	50	21	17
2	1436	694	397
3	1457	1605	913

upper bound (AvgUB). The last line (RelDev Appr3) reports the relative deviation of the MIP solution of Approach 3 from this average upper bound. Note that these deviations are all relatively small and decrease as the size of the system increases. We therefore conjecture that it is easier to determine high quality schedules for a large system than for a small system.

If we compare the profit values of the simulated schedules for increasing workloads and system sizes, we observe the superlinear increase of the profit which demonstrates nicely the economies of scale in contact centers.

The average computation times of the MIP solver in Table 9 decrease as the system size increases. The smallest computation times are observed for the purely deterministic Approach 1. If both arrivals and processing are modeled as realizations of random variables in Approach 3, the highest computational effort results. (The computation times reported in Table 9 exceed the computation time limit of 1000 seconds in those cases where we first tried 20 scenarios without finding a feasible schedule and then reduced the number of scenarios to 10. In these cases we included the first 1000 seconds for the first attempt in the reported computation time.) If a contact center is very large, many of the X_{ak} variables in the LP-relaxation of Problem **P** tend to have a relative large value, i.e., $X_{ak} >> 1$. In this case, the LP-relaxation provides a tight bound and the branch&bound process of the MIP solver is fast. For a small call center the opposite is the case. We think that this explains why our method finds shift schedules for larger contact centers more quickly than for small centers.

In order to study the system behavior, we now consider in more detail case 1S from Table 6. In this small call center the class 2 customers generate a per-call revenue that exceeds the direct cost of a call, if it is answered by a specialist and no exogenous service level limit is imposed. Figures 6 and 7 show the different structure of the solutions, if we explicitly model random customer arrivals in Approach 2 instead of the mean arrival process in Approach 1. While in the solution to Approach 1, almost exclusively specialists are scheduled, the solution to Approach 2 uses many generalists and no specialists for class 2 at all. This is a much more robust solution.

In Figure 8 we present a simulation result of Case 1S for the schedule resulting from Approach 2. The figure shows that the waiting times of the class 2 customers are much higher than those of class 1. This is due to the lower profitability of class 2 calls and the higher impatience of class 1 customers. The function of the average waiting times exhibits a characteristic ramp profile which increases during the morning hours (as the number of agents increases at distinct moments in time) and decreases in the after-

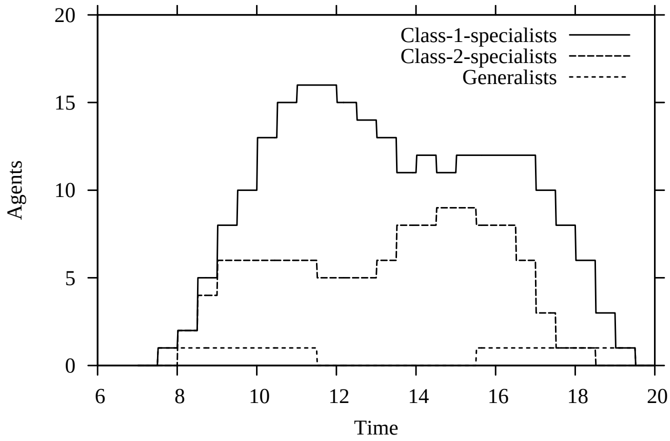

Fig. 6 Staffing level for Case 1S (Approach 1)

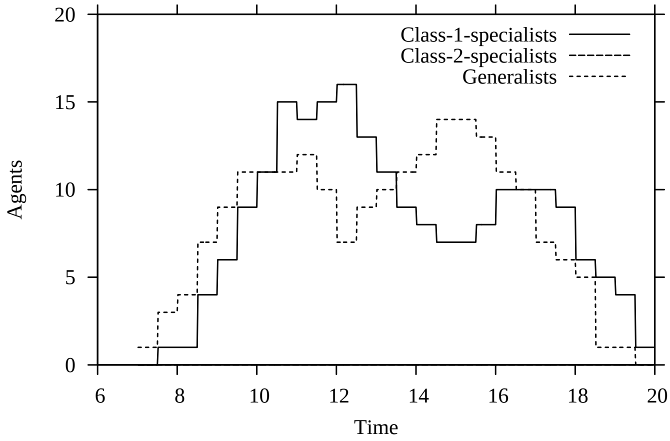

Fig. 7 Staffing level for Case 1S (Approach 2)

noon. The worst and most variable service is offered in the early morning and late afternoon when small numbers of agents are faced with strongly changing arrival rates. In the middle of the day, when both the number of customers and of agents in the system reaches peak levels, the system offers the lowest waiting times due to its economies of scale.

Our last numerical example addresses the carry-over on undone work from one time interval to the next. We treat the large system, but assume that neither customer class generates any revenue. Class 1 customers contact the center by phone, have an average waiting time tolerance of 15 seconds and call again after (on average) two hours. Customer class 2 sends e-mails that remain in the center until they are served or the center is closed. We demanded that 90% of the original class 1 calls and 99.9% of the e-mails had to be served at the end of the day. Figure 9 presents a graph of the

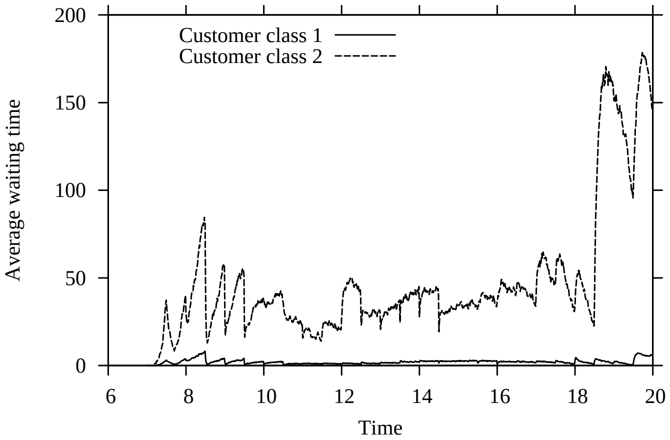

Fig. 8 Simulated average waiting times of the customers

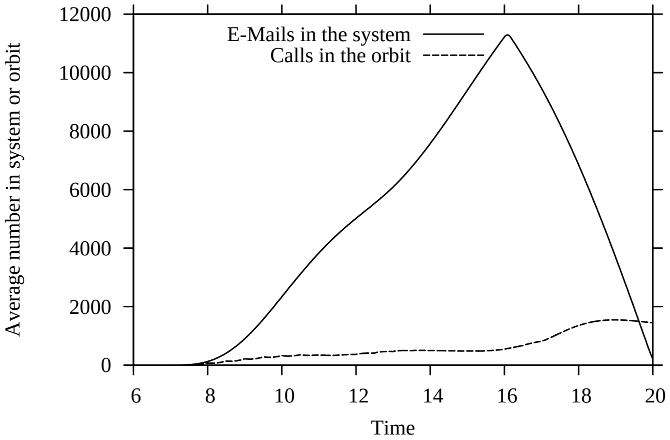

Fig. 9 Simulated average number of calls in the orbit and e-mails in the system

number of class 1 customers (calls) in the orbit and of e-mails in the system. In this case, the different time intervals are clearly interrelated, as opposed to the standard assumptions of the SIPP approach.

5 Managerial implications and suggestions for further research

We presented a model for the shift scheduling problem in dynamic contact centers with skills-based routing, impatient customers and retrials. Unlike the SIPP approach, it does not utilize a stationary queueing model. The intertemporal interdependencies due to retrials or unanswered e-mails can be represented in this approach and profit-maximizing schedules can be approximated. The uncertainty of interarrival and processing times can be incorporated into a simulation optimization approach via an optimization over a set of different scenarios. This leads to schedules for which the average profit is robust to variability in call arrivals and processing. To the best of our knowledge, this is the first profit-oriented shift scheduling approach for contact centers with SBR and retrials.

As the contact center gets larger, the accuracy and efficiency of the approach increases. In general it appears to be sufficient to model random call arrivals to obtain robust and efficient schedules. The additional benefit of modeling random processing times appears to be negligible while the additional numerical effort is substantial. The managerial implications are that at least for large contact centers efficient shift schedules can be found without using stationary queueing models which affects the design of workforce planning systems.

Further research should address the tour scheduling problem over successive days. A problem here is that the precision of the forecasts of contact arrivals typically degrades quickly as the planning horizon is expanded from one-day-ahead to one-week-ahead because of autocorrelation in the time series. It might therefore be necessary to design a hierarchical system that can deal with different degrees of forecasting accuracy and create robust plans. In addition, it might be necessary to model individual agents with their limited temporal availability. It should also be interesting to extend the model to dynamic priority rules that reflect the achieved transient performance of the system. To this end one could introduce an additional set of real-valued decision variables that reflect the fraction of each agent group that is assigned to each customer group. In this paper we concentrated on inhomogeneous Poisson arrivals and deterministic or exponential processing times. We think that the general method can also be used for other distributions of random variables. It is possible to use a simple sampling approach as in Helber et al. (2008) to transform realizations of random (inter-arrival or processing) times following arbitrary continuous distribution into realizations of discrete random numbers of events per period. We already used this approach in the context of flow line analysis.

Acknowledgement: The research reported in this paper was funded by DFG grant HE 3124.

References

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

Atlason, J., M. Epelman, and S. Henderson (2004). Call center staffing with simulation and cutting plane methods. *Annals of Operations Research 127*, 333–358.

Atlason, J., M. A. Epelman, and S. G. Henderson (2008). Optimizing call center staffing using simulation and analytic center cutting-plane methods. *Management Science 54*(2), 295–309.

Avramidis, A., W. Chan, and P. L'Ecuyer (2007). Staffing multiskill call centers via search methods and a performance approximation. *IIE Transactions (to appear).* available from http://www.iro.umontreal.ca/ lecuyer/papers.html.

Avramidis, A., M. Gendreau, P. L'Ecuyer, and O. Pisacane (2007). Optimizing daily agent scheduling in a multiskill call center. Technical report. CIRRELT Report 2007-44, available from http://www.iro.umontreal.ca/ lecuyer/papers.html.

Bassamboo, A., J. M. Harrison, and A. Zeevi (2006). Design and control of a large call center: Asymptotic analysis of an LP-based method. *Operations Research 54*(3), 419–435.

Bhulai, S., G. Koole, and A. Pot (2008). Simple methods for shift scheduling in multiskill call centers. *Manufacturing & Service Operations Management.* Published online in Articles in Advance, January 4, 2008 DOI: 10.1287/msom.1070.0172.

Cezik, M. T. and P. L'Ecuyer (2007). Staffing multiskill call centers via linear programming and simulation. *Management Science, to appear*.

Feldman, Z. (2004). Staffing of time-varying queues to achieve time-stable performance. Master's thesis, Technion - Israel Institute of Technology, Haifa, Israel.

Feldman, Z., A. Mandelbaum, W. A. Massey, and W. Whitt (2008). Staffing of time-varying queues to achieve time-stable performance. *Management Science 54*(2), 324–338.

Franx, G. J., G. Koole, and A. Pot (2006). Approximating multi-skill blocking systems by hyperexponential decomposition. *Performance Evaluation 63*, 799–824.

Fukunaga, A., E. Hamilton, J. Fama, D. Andre, O. Matan, and I. Nourbakhsh (2002). Staff scheduling for inbound call centers and customer contact centers. *Artificial Intelligence Magazine 23*(4), 30–40.

Gans, N., G. Koole, and A. Mandelbaum (2003). Telephone call centers: Tutorial, review and research prospects. *Manufacturing and Service Operations Management 5*, 79–141.

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

Green, L. V., P. J. Kolesar, and W. Whitt (2007). Coping with time-varying demand when setting staffing requirements for a service system. *Production and Operations Management 16*, 13–39.

Harrison, J. M. and A. Zeevi (2005). A method for staffing large call centers based on stochastic fluid models. *Manufacturing and Service Operations Management 7*, 20–36.

Helber, S., K. Schimmelpfeng, R. Stolletz, and S. Lagershausen (2008). Using linear programming to analyze and optimize stochastic flow lines. Technical report, Leibniz Universität Hannover, Institut für Produktionswirtschaft, Königsworther Platz 1, D-30167 Hannover. available online at http://www.wiwi.uni-hannover.de/Forschung/Diskussionspapiere/dp-

389.pdf.

Henken, K. (2007). *Dynamic Contact Centers with Impatient Customers and Retrials.* Ph. D. thesis, Leibniz University Hannover, School of Economics and Management. http://www.henken-midlum.de/kirsten/Wissenschaft.html.

Ingolfsson, A., E. Cabral, and X. Wu (2003). Combining integer programming and the randomization method to schedule employees. Technical Report 02-1, Department of Finance and Management Science, Faculty of Business, University of Alberta.

Jiménez, T. and G. Koole (2004). Scaling and comparison of fluid limits of queues applied to call centers with time-varying parameters. *OR Spectrum 26*, 413–422.

Koole, G. and A. Mandelbaum (2002). Queueing models of call centers: An introduction. *Annals of Operation Research 113*, 41–59.

Koole, G. and E. Van der Sluis (2003). Optimal shift scheduling with a global service level constraint. *IIE Transactions 35*(11).

Mandelbaum, A., W. Massey, and M. I. Reiman (1998). Strong approximations for Markovian service networks. *Queueing Systems 30*, 149–201.

Stolletz, R. (2003). *Performance Analysis and Optimization of Inbound Call Centers.* Number 528 in Lecture notes in Economics and Mathematical Systems. Springer.

Stolletz, R. (2007). Approximation of the non-stationary $M(t)/M(t)/c(t)$-queue using stationary queueing models: The stationary backlog-carryover approach. *European Journal of Operational Research*. doi:10.1016/j.ejor.2007.06.036.

Stolletz, R. and S. Helber (2004). Performance analysis of an inbound call center with skill-based routing. a priority queueing system with two classes of impatient customers and heterogeneous agents. *OR Spectrum 26*, 331–352.