11^{th} *International Conference on MOdeling, Optimization and SIMlation - MOSIM16 - August 22-24-2016- Montreal - Canada "Toward circular Economy"*

STAFFING OPTIMIZATION WITH CHANCE CONSTRAINTS FOR EMERGENCY CALL CENTERS

Thuy Anh Ta, Pierre L'Ecuyer, Fabian Bastin

Département d'Informatique et de Recherche Opérationnelle
Université de Montréal; also CIRRELT and GERAD, Canada
tathuyan@iro.umontreal.ca, lecuyer@iro.umontreal.ca, bastin@iro.umontreal.ca

ABSTRACT: *We consider a staffing problem with probabilistic constraints in an emergency call center. The aim is to minimize the total cost of agents while satisfying chance constraints defined over the service level and the average waiting time, in a given set of time periods. We provide a mathematical formulation of the problem in terms of probabilities and expectations. We define a sample average approximation (SAA) version of this problem whose solution converges to that of the exact problem when the sample size increases. We also propose a quick and simple simulation-based (heuristic) algorithm to compute a good (nearly optimal) staffing solution for the SAA problem. We illustrate and validate our algorithm with a simulation model based on real data from the 911 emergency call center of Montreal, Canada.*

KEYWORDS: *Emergency call center, staffing, chance constraints, simulation, service level, average waiting time.*

1 INTRODUCTION

Call centers are broadly defined as centralized systems used for receiving or transmitting customers requests by telephone. Some call centers play an important role in real life, such as the telephone services of financial institutions or the 911 emergency services in North America. The call center industry has been developing strongly and rapidly in recent years, in terms of both work-force and economic scope. For instance, in 2014, in the United States, agents providing customer service ranked *6th* in the list of the largest occupations, with approximately 2.5 million agents (Bureau of Labor Statistics 2015b). The annual salary cost of agents was estimated at US $91.5 billion in 2014 (Bureau of Labor Statistics 2015a).

In call centers in general, calls of different types are handled by agents of different skills. Each call type requires a specific skill and each agent group has a selected number of skills. The agent groups are distinguished by the set of call types they can serve (also known as the *skill set*). Calls arrive randomly according to some stochastic processes. An arriving call can be served immediately or must be placed in a waiting queue. Waiting calls may abandon after a random patience time. The *staffing* and *scheduling* problems deal with minimizing the cost under a set of constraints on the quality-of-service (QoS). More precisely, based on distributional forecasts of arrival call volumes and a stochastic model of the entire call center, the task is to decide how many agents of each skill group to have at each time period of the day. In a *staffing* problem, one must decide how many agents are needed without considering constraints on agent work schedules and availability. In a *scheduling* problem, a set of admissible work schedules is specified, and one must determine the number of agents of each skill group having each work schedule.

In this paper, we focus on *emergency call centers*, a specific type in which the response times must be very short, much shorter than for other typical service systems, because they typically involve a situation where the safety of people or property is at risk and requires immediate assistance. Lewis, Herbert, Summons & Chivers (2007) identify this as an important factor in defining the staffing levels at emergency call centers. High effectiveness of these call centers require rapid response to calls and a high standard of agent capability. The service level must be very high and average waiting times very low. Lafond (2012) gives an example in which 90% of the 911 calls must be answered within 10 seconds during the busy hour (the hour with the largest call volume during the day) and 95% of calls must be answered within 20 seconds overall. The 911 call center in Montreal requires that 95% of all arriving calls are answered within 2 seconds (plus a connecting time of about 4 seconds, for a total of 6 seconds). This requirement of high service levels and low average waiting times in emergency call centers implies that the occupancy of agents must be low. That is, these call centers are overstaffed in comparison with other typical business call centers. We will

take advantage of this property in our development of a quick staffing algorithm adapted to emergency call centers.

The call center staffing problem has received a great deal of attention in the literature. It is common to divide the day into several periods of equal length during which the staffing is held constant and the arrival rate is assumed approximately constant. The system is often assumed to be in steady-state within each period. In the case of a single call type and agent group, this crude approximation plus additional simplifying assumptions permit one to use Erlang queueing formula to determine the required staffing within each period. The simplest such model is an $M/M/s$ queue, also known as an Erlang C system (Cooper 1981). In this model, the interarrival times and the service times are independent and exponential, and the system is assumed stationary, with s servers. This model also ignores blocking and customer abandonment. It is not very realistic. For good realism and better accuracy, the staffing and scheduling should be done using simulation. Atlason, Epelman & Henderson (2004) proposed a general methodology, based on the cutting plane method of Kelley Jr. (1960), to optimize the staffing in a call center with a single call type and single skill, under service level (SL) constraints. Their method combines simulation with integer programming and cut generation. In the *multiskill* case, the staffing problems are much more difficult, even for a single period in steady-state. The Erlang formulas and their approximations (for the SL) no longer apply, and simulation seems to be the only reliable tool. Ceẓik & L'Ecuyer (2008) extend the method of Atlason et al. (2004) to multiskill call centers. They also point out difficulties encountered with large problems, and develop heuristic methods to deal with them.

In typical staffing and scheduling problem formulations, the constraints are on average performance measures in the long run. However, even if the long-term average satisfies a given constraint (or target), the QoS on any given day is a random variable that may have a large variance, and may take a value smaller than the target for a significant fraction of the days. To cope with this, managers are often interested also in the probability that the observed (realized) QoS of the day meets the constraints. Gurvich, Luedtke & Tezcan (2010) propose using probabilistic constraints on the (random) QoS values over a given (single) time period. The arrival rates are assumed random and time-independent. They consider probabilistic constraints on the abandonment ratios. More precisely, for a risk level δ chosen by the manager, the requirement is that the QoS constraint can be violated on at most a fraction δ of the arrival rate realizations. Excoffier, Gicquel, Jouini & Lisser (2014) and Excoffier, Gicquel & Jouini (2015) also consider probabilistic constraints, but for a multi-period shift-scheduling problem for a single call type and single-skill call center, with uncertainty in the future call arrival rates. Chan, Ta, L'Ecuyer & Bastin (2014) consider a single-period two-stage stochastic staffing problem under chance constraints, for multiskill call centers with arrival rate uncertainty. They suggest a simulation-based cutting plane method as in (Ceẓik & L'Ecuyer 2008) combined with a local search algorithm. This cutting plane algorithm is however complicated and requires large computing times.

Our aim in this paper is to propose a simpler and faster staffing optimization method for call centers with a single agent group and in which most customers do not wait, just like in 911 emergency call centers. The systems we consider are characterized by the following two properties: (i) all agents are identical and can answer all call types, (ii) the occupancy of agents is low, compared to other type of call centers, and the buildup of a queue is very rare. Because of (ii), the QoS measures in successive time periods are almost independent, so changing the number of agents in one period does not significantly affect what goes on in other periods, and therefore the periods can be staffed (almost) independently from each other. An adjustment for the small amount of dependence can be made afterward, using simulation. We define the staffing problem, formulate a sample average approximation (SAA), and then propose a simple and fast simulation-based heuristic algorithm to obtain a good solution for this SAA. The idea of the method is to first find the "right" number of agents period by period, then adjust for interaction and global constraints, via simulation. We test our method with a model based on real data from the 911 emergency call center in Montreal. The numerical results indicate that the approach works well. Software based on this work has been installed at that call center. We cannot release the data for confidentiality reasons but the software itself is available at `http://www-etud.iro.umontreal.ca/~tathuyan/`.

The remainder of the paper is organized as follow. In Section 2, we define our model and staffing problem using chance constraints with respect to the service level and the average waiting time. In Section 3 we define the SAA of the chance-constrained staffing problem. In Section 4, we propose a simulation-based optimization algorithm to solve the SAA problem for the special case where all agents have all skills (a single group of agents). In Section 5, we report the results of numerical experiments based on real data. Section 6 gives a conclusion.

2 MODEL AND PROBLEM FORMULATION

2.1 CALL CENTER MODEL

We consider a call center model in which incoming calls arrive at random according to an arbitrary arrival process. There is a single group of agents that can serve all calls. In 911 emergency call centers, and in the simulation models and programs that we use in our experiments, there are different call types with a different arrival process for each call type and different service time distributions. However, the distinction of different call types in the simulation model has no impact on the problem formulation and methodology presented in this paper. So to simplify the notation and reduce the "distraction" from the main topic of the paper, we will assume in our problem formulation and algorithms that all calls types are aggregated in a single call type. The arrival processes can be arbitrary, but are usually non-stationary Poisson with random arrival rates which are dependent across periods; see, e.g., Avramidis, Deslauriers & L'Ecuyer (2004), Ibrahim, Ye, L'Ecuyer & Shen (2016), and Oreshkin, Régnard & L'Ecuyer (2016). Arriving calls that find all servers occupied line up in an infinite buffer queue, and are served in a FCFS order, unless they abandon before. The day is divided into P periods of equal length, labeled from 1 to P. The *staffing* vector $y = (y_1, ..., y_P)^{\mathrm{T}}$ represents the number of agents in the center, in each period.

2.2 PERFORMANCE MEASURES

In call center systems, performance measures allow to assess the quality of service and efficiency in a call center. They can be defined per period or globally over the day. Constraints are imposed on these measures to ensure that the center meets its goals and objectives. These performance measures can be computed (or estimated) based on the observed data. These measures can be defined in many different ways; there is no single convention of formula. In many optimization problems studied so far, a common approach is to impose the constraints on the expected (average) performance measures over an infinite time horizon. In the present work, we consider instead probabilistic constraints on the performance measures over each period and over the day, which are random variables.

The *service level*, a widely used measure in industry, is defined as *the fraction of calls answered within a given time* τ, where τ is a parameter called *acceptable waiting time*. For a given time interval and a given staffing y, let $A(\tau, y)$ be the number of calls served after a waiting time less than or equal to τ during the given time interval, let N be the total number of calls arriving during this time interval, and $L = L(\tau, y)$ be the number of calls who abandoned after a waiting time no larger than τ during the same time interval. Since the arrival and service times are random, the SL in a given time period is a random variable

$$S(\tau, y) = \frac{A(\tau, y)}{N - L(\tau, y)}. \tag{1}$$

This definition of SL in (1) is used in our problem formulation with chance constraints. For a given staffing y, no reliable formula or quick algorithm is available to estimate the distribution of SL; it can be estimated accurately only with a long (stochastic) simulation.

A different definition of SL was used in most previous articles; e.g., (Atlason et al. 2004, Avramidis, Chan & L'Ecuyer 2009, Avramidis, Chan, Gendreau, L'Ecuyer & Pisacane 2010), etc.:

$$\overline{S}(\tau, y) = \frac{\mathbb{E}[A(\tau, y)]}{\mathbb{E}[N - L(\tau, y)]}. \tag{2}$$

The SL in this definition (2) represents the fraction of calls answered within τ over an infinite number of independent and identically distributed (i.i.d.) copies of the given time interval. It can be computed by Erlang formulas in very simplified models (Cooper 1981), and by simulation otherwise.

Another important performance measure is the *average waiting time*, simply defined by the total wait of all calls during the given time period, divided by the number of calls in that period. Similar to the SL, when computed over a given time period it is a random variable

$$W(y) = \frac{T}{N} \tag{3}$$

where T is the sum of waiting times of calls (served or abandoned) that arrived during the given time interval. An alternative definition represents the average waiting time in the long run, over independent replications of the given time interval:

$$\overline{W}(y) = \frac{\mathbb{E}[T]}{\mathbb{E}[N]}. \tag{4}$$

Other performance measures are proposed in Jouini, Koole & Roubos (2013).

2.3 PROBLEM FORMULATION

We now define our staffing problem for a single-skill call center, with chance constraints. The goal is to minimize the operating cost of the center under a set of chance constraints on the QoS. The day is divided into periods (e.g., 30 minutes or one hour). The objective function is the sum of the costs of all the agents, where the cost of an agent is a deterministic function of its set of skills. All our development could

handle easily additional constraints on other performance measures such as the abandonment ratio, the occupancy ratio, etc.

Given the staffing vector y, let $S_p(\tau_p, y)$ be the fraction of calls answered within τ_p seconds during period p (the SL); $S_0(\tau_0, y)$ the fraction of calls answered within τ_0 seconds during the day; $W_p(y)$ the average waiting time during period p; and $W_0(y)$ the average waiting time (AWT) of all calls during the day. These are random variables whose distributions depend on the entire staffing. The constraints are of the form: *the individual probabilities that the SL and AWT constraints are satisfied are no smaller than some given thresholds.*

More specifically, the SL and AWT constraints have the form:

$$\begin{aligned} \mathbb{P}[S_p(\tau_p, y) \geq s_p] \geq r_p \quad \forall p, \\ \mathbb{P}[W_p(y) \leq w_p] \geq v_p \quad \forall p, \end{aligned}$$

where the s_p are SL targets, the w_p are AWT targets, and r_p, v_p are given constants in $(0, 1)$. We denote $g_p^1(y) = \mathbb{P}[S_p(\tau_p, y) \geq s_p] - r_p$ and $g_p^2(y) = \mathbb{P}[W_p(y) \leq w_p] - v_p, \quad \forall p$. The chance-constrained staffing problem is then:

$$\min c^{\mathrm{T}} y = \sum_{p=1}^{P} c_p y_p$$

subject to:

$$\begin{aligned} g_p^j(y) \geq 0 \text{ for } j = 1, 2, \quad \forall p, \\ y \geq 0 \text{ and integer}, \end{aligned} \tag{P}$$

where $c = (c_1, \ldots, c_P)^{\mathrm{T}}$, and c_p is the cost of an agent in period p.

3 SAMPLE AVERAGE APPROXIMATION PROBLEM

There are no formulas to compute exactly the probability functions g_p^j in (P), but they can be estimated by simulation. Suppose we simulate n independent days, i.e., n simulation runs. Let ω represent the source of randomness, i.e., the sequence of all independent $U(0, 1)$ random numbers that drive the successive simulation runs, regardless of their number n. We assume that ω is fixed while y can vary; this is the idea of *common random numbers* (Asmussen & Glynn 2007, L'Ecuyer 2007). Let $\hat{S}_p^i(\tau_p, y)$ and $\hat{W}_p^i(y)$ be the SL and AWT in period p for the i-th simulated day, given staffing vector y, for ω fixed. Problem (P) is approximated by the following SAA:

$$\min c^{\mathrm{T}} y$$

subject to:

$$\begin{aligned} \hat{g}_p^1(y) &:= \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}[\hat{S}_p^i(\tau_p, y) \geq s_p] - r_p \geq 0, \quad \forall p, \\ \hat{g}_p^2(y) &:= \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}[\hat{W}_p^i(y) \leq w_p] - r_p \geq 0, \quad \forall p, \\ y &\geq 0 \text{ and integer}, \end{aligned} \tag{S_n}$$

where $\mathbb{I}$ is the 0-1 indicator function.

Convergence of the optimal value and the set of optimal solutions of the SAA to those of the exact problem with probability 1 as $n \to \infty$ are established in the master thesis of Ta (2013), to which we refer the reader for further details.

4 SIMULATION METHOD

This section presents a simple simulation-based optimization algorithm for our problem. The general idea is to replace Problem (P) by the SAA version (S_n), and try to solve it. For that, we increase or decrease the number of agents at each period separately, while keeping the (SAA) chance constraints satisfied. To simplify the description, we suppose that all cost coefficients are $d_p = 1$, for $1 \leq p \leq P$, but generalizing to different cost coefficients is straightforward.

4.1 SIMULATION-BASED ALGORITHM

Our simulation-based optimization algorithm, named chance constraints simulation-based (CCS) algorithm, consist of five stages as described below

Stage 1: `Initialize` We can choose an arbitrary initial staffing level. Two specific strategies are considered below. The simplest way is to start with a staffing equal to 0 for all periods. We can also choose an initial staffing level by using the Erlang C formula since, in some cases, we may expect that Erlang C gives a staffing level which is close to a good solution. Note that Erlang C gives a staffing level satisfying the constraints on the expected SL in the long run.

Stage 2: `Increase` With the initial staffing, there may exist some chance constraints in (S_n) that are satisfied while others are not. One natural approach would be to increase the staff number in some periods in which the constraints on the SL or AWT are violated, until these constraints are satisfied. Therefore we consider the periods in which the constraints are not satisfied, and increase the number of agents in these periods until the constraints in these periods are satisfied.

Stage 3: `Decrease` After stage 2, all constraints in periods are satisfied. However, we can sometimes decrease the number of agents in several periods such

that the constraints in these periods are still satisfied. In stage 3, we decrease the number of agents as much as possible, under the condition that the constraints in the individual periods are still satisfied.

The simplest manner to change (increase and decrease) the number of agents in Stage 2 and 3 is that we might change the number of agents by at most one unit in a single period at each iteration. After each change of the number of agents, we perform a simulation. This approach can be time-consuming as it may require many simulations. In order to save computational time, we use bisection method to change (increase and decrease) the number of agents in several periods at the same time, i.e., in each iteration, we use a bisection method to increase or decrease the number of agents, in all selected periods (where increase is required or decrease appears to be acceptable) simultaneously.

Stage 4: `Increase-Last` We consider the constraints for the aggregated QoS over the whole day. These constraints may be unsatisfied. We will increase the staffing levels until these constraints are satisfied. We may have plenty of choices to choose the periods in which we increase the number of agents. Here are several examples.

- At each iteration, we choose the period with the smallest SL, and add one agent in this period. After adding the new agent and running simulation, we check if the constraints over the whole day are satisfied. This stage ends as soon as these constraints are satisfied.
- We consider the differences between the estimations of the probabilities that the constraints on the SL are satisfied and the target of the probabilities in all periods. The number of agents in the period with the lowest difference would be increased. For more detail, we consider constraints on the SL in all periods $p = 1, ..., P$. After setting any new staffing level and running simulations, we can compute $\hat{g}_p^1(y)$, $\forall p$. In stage 4, the number of agents in the period in which $\hat{g}_p^1$ is lowest will be increased. In the numerical experiment, we use this method to increase the number of agents.

Stage 5: `Correction` Changing the staffing in one period can alter the performance (such as SL, etc.) in other periods as well. Atlason et al. (2004) present an example showing that the staffing level in one period can have a considerable effect on the SL in another period. The SL depends on the staffing level in the previous period because a low staffing level in an earlier period results in a queue build-up, which increases waiting in the next period. The staffing level in a later period affects the SL in an earlier period due to a fact that arrival calls in the earlier period may still be waiting at the beginning of the next period and thus are served earlier if there are more servers in that period. In some call centers, e.g., the 911 emergency call center, this effect is very small because there is rarely a queue in the system (the agents are not very busy), but in general, this effect could be very important. Since our algorithms are based on changing the number of staffing in periods, the manner and the order of periods of changing the number of agents may have noticeable affect on the results.

Therefore, to improve the quality of solutions, after the four previous stages, we add the present `Correction` stage, in which we consider all the periods one by one. For each period, we try to decrease the number of agents in this period as much as possible, under the constraint that the staffing level is still feasible for the sample problem.

4.2 ANALYSIS OF THE ALGORITHM

There is no proof that our CCS algorithm always converges to an optimal solution, and this is why we call it a heuristic, but at least we have a proof that it terminates in finite time with a feasible solution to the SAA.

Proposition 1 *Suppose that the sample problem (S_n) is feasible. Then the CCS algorithm terminates at a feasible solution in a finite number of iterations.*

Proof. Suppose that $y^* = (y_1^*, ..., y_P^*)$ is a feasible solution of the sample problem (S_n), that $y_0 = (y_{01}, ..., y_{0P})$ is an initial staffing level, and that our algorithm does not stop after a finite number of iterations.

Assume also that the stage `Increase` does not stop after a finite number of iterations, i.e., the algorithm CCS cannot find a staffing level which satisfies the constraints in all periods. However, for each $1 \leq p \leq P$, after a finite number of increases of the staffing in the period p, the number of agents in this period will be equal or greater to y_p^*. Therefore, the algorithm CCS can always find solutions which satisfy all the constraints for any period after a finite number of iterations.

In the stage `Decrease`, we decrease the staffing in all periods such that they still satisfy the constraints in all periods. Since the number of agents in each period is non-negative, this stage terminates after a finite number of iterations.

Suppose now that the stage `Increase-Last` does not stop after a finite number of iterations, i.e., we cannot increase the staffing to satisfy the constraints in the whole day. However, after a finite number of increases, we will obtain a staffing level $y = (y_1, ..., y_P)$ such that $y_p \geq y_p^*$ for all $1 \leq p \leq P$. Thus, this

staffing level satisfies the constraints over the whole day. Therefore, the stage `Increase-Last` stops after a finite number of iterations.

Similarly, the stage `Correction` also terminates after a finite number of iterations. ■

In conclusion, the algorithm CCS terminates after a finite number of iterations. Obviously, it returns staffing levels which satisfy all the constraints (S_n), so they deliver upper bounds for the cost of the SAA problem. Moreover, in our algorithm, in the stage `Decrease`, we try to decrease the number of agents as much as possible, and the stage `Increase-Last` stops as soon as we find a staffing level which satisfies the constraints over the whole day. After that, in the stage `Correction`, we try to reduce the number of agents in all periods as much as possible, provided that we still obtain feasible solutions. Therefore, we can expect that our algorithm returns good solutions to the SAA. Our empirical experiments support that.

5 NUMERICAL EXPERIMENTS

5.1 DATA AND EXPERIMENTAL SETTING

In this section, we test the performance of the CCS algorithm with a call center model built to be representative of real data sets obtained from a 24-hour emergency call center (911). We first describe our experimental setting.

The emergency call center is operated 24 hours a day for 7 days a week and has one skill group. We assume that the callers do not abandon. The service time is modeled using the *Johnson* S_U distribution. Each day is divided into P time periods of equal length. Let $\mathbb{X} = (X_1, \ldots, X_P)$ be the vector of arrival counts in those P periods, and assume that the arrivals come from a Poisson process with a random rate Λ_p, constant over period p. Suppose moreover $\Lambda = (\Lambda_1, \ldots, \Lambda_P)$ and $\Lambda_p = B_p \lambda_p$ where B_p is a non-negative random variable with $\mathbb{E}[B_p] = 1$ for each p. B_p is called the *busyness factor* for period p and we denote $\mathbb{B} = (B_1, \ldots, B_P)$. Oreshkin et al. (2016) studied and compared different arrival process models in the context of 911 call centers, and found that the use of a normal copula for $\mathbb{B}$ is appropriate. Each B_p is assumed to have a $\Gamma(\alpha_p, \alpha_p)$ distribution with cumulative distribution function G_p, i.e. $B_p = G_p^{-1}(\Phi(Z_p))$, where Φ is the standard normal distribution function and $Z = (Z_1, \ldots, Z_P) \sim \text{Normal}(0, R^Z)$, a multivariate normal vector with mean zero and covariance matrix R^Z.

We use real data sets collected in a week (from Monday to Sunday) and denote each model by the name of the day. A day is divided in 48 half-hour periods, and we test our algorithm using different parameter sets as follows.

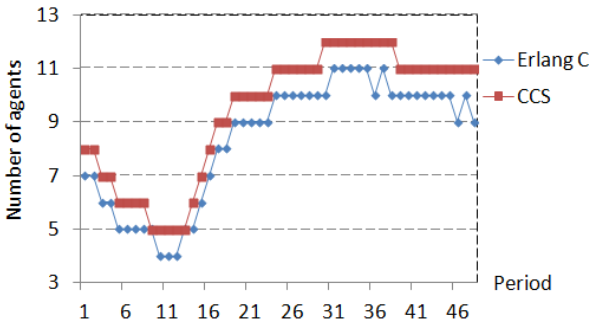
1. Case 1: the SL is very high and the average waiting times are very low, i.e., the agents have a very low occupancy. The parameters are defined as follow: $\tau_0 = \tau_p = 2$ (seconds), $s_0 = s_p = 0.95$, $r_0 = 0.95$ and $r_p = 0.85$, $w_0 = w_p = 2$ (seconds), $v_0 = 0.95$ and $v_p = 0.85$ for $1 \leq p \leq P$.
2. Case 2: the QoS constraints are less demanding and the occupancy is higher, compared to Case 1. More precisely, we choose $\tau_0 = \tau_p = 120$ (seconds), $s_0 = s_p = 0.8$, $r_0 = 0.95$ and $r_p = 0.85$, $w_0 = w_p = 120$ (seconds), $v_0 = 0.95$ and $v_p = 0.85$ for $1 \leq p \leq P$.
3. Case 3: we consider higher occupancy call centers: $\tau_0 = \tau_p = w_0 = w_p = 300$ (seconds), while other parameters are similar as Case 2.

Only the first case really corresponds to the type of situation targeted by this paper. We nevertheless try the other cases for comparison.

5.2 NUMERICAL RESULTS

We test our algorithm with each experimental setting presented above, with $n = 1000$ in (S_n). In all the tests, the staffing levels given by the Erlang C formula were always less than those prescribed by our CCS algorithm, and the difference was always small, as illustrated in Figure 1. This suggests that the Erlang C staffing can provide a very good initial solution for our CCS algorithm.

Figure 1 – *Staffing given by Erlang C and the CCS for* `Monday` *- Case 1.*



After having obtained staffing solutions for all cases, we analyze the quality of these solutions by performing an out-of-sample (OOS) validation experiment over 10000 simulated days. The results show that staffing levels given by the CCS satisfy most constraints in all cases, in which all the constraints associated with AWT are satisfied. Furthermore, for

violated constraints, the probability values are very close to the targets (within a neighborhood of 2%). We list in Table 1 which constraints were violated. The empty cells correspond to models where all the constraints are satisfied. We use a reduced notation to identify the violated constraints, e.g., $\mathbb{P}^S_{24}$ means that the constraint $\mathbb{P}[S_{24}(\tau_{24}, y) \geq s_{24}]$ was violated in the SAA. For Case 2, the number of violated constraints is smaller, compared to Case 1, and in Case 3, all constraints are satisfied. These results suggest that the CCS algorithm works well in all the considered cases. It may be noted that with the staffing given by the Erlang C, most of the constraints are not satisfied, and the differences with the targets are often beyond 20%.

Table 1 – *Violated constraints*

Models	Case 1	Case 2	Case 3
`Monday`	$\mathbb{P}^S_{24} = 0.844$ $\mathbb{P}^S_{19} = 0.840$	$\mathbb{P}^S_{43} = 0.848$	
`Tuesday`	$\mathbb{P}^S_{17} = 0.845$ $\mathbb{P}^S_{35} = 0.848$		
`Wednesday`	$\mathbb{P}^S_{44} = 0.833$	$\mathbb{P}^S_{48} = 0.832$	
`Thursday`	$\mathbb{P}^S_{26} = 0.833$		
`Friday`		$\mathbb{P}^S_{1} = 0.843$	
`Saturday`		$\mathbb{P}^S_{8} = 0.847$	
`Sunday`	$\mathbb{P}^S_{35} = 0.842$		

To improve the quality of these solutions after the OOS tests, we can then try to increase the number of agents in periods where the constraints are not satisfied. For example, for the model `Monday` in Case 1, the SL constraint in period 24 is not satisfied. We increase the number of agents by one unit in that period, and perform an OOS evaluation with this new staffing level. The estimated probability that the SL constraint in period 24 is satisfied then increases and becomes larger than the target 0.85, i.e., the chance constraint in SL for this period is satisfied in the SAA.

In addition, we can also try to improve the solutions by decreasing the numbers of agents in periods where the corresponding constraints are satisfied. More precisely, in each model, we choose the period in which the estimated probability that the constraint on the SL is the largest, and try to remove one agent. When doing that, for all the models, we found in the OOS evaluations that the new staffing levels were infeasible. That is, we were unable to remove agents from the staffing levels obtained by the CCS while keeping the chance constraints satisfied. We also observed that when we changed the number of agents in a period, the probability values of other periods were not affected in Case 1. This confirms our remark regarding the properties of emergency call centers. This is however not the case in Cases 2 and 3 when the occupancy is higher.

Finally, we briefly discuss the computing time for optimizing the staffing using the CCS algorithm. We used a computer with an Intel(R)-Core(TM) i5-3.20GHz, and running Window 10. The computer has multi-processors but we only use one, as the code is not parallelized. For all the tests, the CCS algorithm (with the sample size of 1000) requires less than 1 minute to return a solution. The cutting plane method of Atlason, Epelman & Henderson (2008) is expected to be more expensive, as it requires several simulation runs to generate cuts, and repeatedly solve a linear programming problem.

6 CONCLUSION

In this paper, we have considered a staffing problem under chance constraints for emergency call centers. We formulated a SAA version of the problem, which permits us to deal with the problem using simulation. We proposed a simple simulation-based algorithm that can be used to quickly approximate the optimal solution. We have assessed the performance of our method by using a model based on real data from a 911 emergency call center in Montreal, under different occupancy levels. The results indicate that our approach performs well in all the cases we have examined.

ACKNOWLEDGMENTS

This work has been supported by grants from NSERC-Canada and Hydro-Québec, a Canada Research Chair, and an Inria International Chair, to P. L'Ecuyer. We also thank Tien Mai and Wyean Chan for their valuable suggestions.

REFERENCES

References

Asmussen, S. & Glynn, P. W. (2007). *Stochastic Simulation*, Springer-Verlag, New York.

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

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

Avramidis, A. N., Chan, W., Gendreau, M., L'Ecuyer, P. & Pisacane, O. (2010). Optimizing daily agent scheduling in a multiskill call centers, *European Journal of Operational Research* **200**(3): 822–832.

Avramidis, A. N., Chan, W. & L'Ecuyer, P. (2009). Staffing multi-skill call centers via search methods and a performance approximation, *IIE Transactions* **41**(6): 483–497.

Avramidis, A. N., Deslauriers, A. & L'Ecuyer, P. (2004). Modeling daily arrivals to a telephone call center, *Management Science* **50**(7): 896–908.

Bureau of Labor Statistics (2015a). *Occupational employment and wages, May 2014 - Customer Service Representatives*. U.S. Department of Labor. Available online at `//http://www.bls.gov/oes/current/oes434051.htm`, (last accessed September, 2015).

Bureau of Labor Statistics (2015b). *An overview of U.S. occupational employment and wages in 2014*. U.S. Department of Labor. Available online at `http://www.bls.gov/news.release/pdf/ocwage.pdf`, (last accessed September, 2015).

Ceẓik, M. T. & L'Ecuyer, P. (2008). Staffing multiskill call centers via linear programming and simulation, *Management Science* **54**(2): 310–323.

Chan, W., Ta, T. A., L'Ecuyer, P. & Bastin, F. (2014). Chance-constrained staffing with recourse for multi-skill call centers with arrival-rate uncertainty, *Proceedings of the 2014 Winter Simulation Conference*, IEEE Press, pp. 4103–4104.

Cooper, R. B. (1981). *Introduction to Queueing Theory*, second edn, North-Holland, New York, NY.

Excoffier, M., Gicquel, C. & Jouini, O. (2015). Distributionally robust optimization for scheduling problem in call centers with uncertain forecasts,, *Proceedings of the 4th International Conference on Operations Research and Enterprise Systems, ICORES 2015*, pp. 3–20.

Excoffier, M., Gicquel, C., Jouini, O. & Lisser, A. (2014). A joint chance-constrained programming approach for call center workforce scheduling under uncertain call arrival forecasts. manuscript.

Gurvich, I., Luedtke, J. & Tezcan, T. (2010). Staffing call centers with uncertain demand forecasts: A chance-constrained optimization approach, *Management Science* **56**(7): 1093–1115.

Ibrahim, R., Ye, H., L'Ecuyer, P. & Shen, H. (2016). Modeling and forecasting call center arrivals: A literature study and a case study, *International Journal of Forecasting* **32**(3): 865–874.

Jouini, O., Koole, G. & Roubos, A. (2013). Performance indicators for call centers with impatient customers, *IIE Transactions* **45**(3): 341–354.

Kelley Jr., J. E. (1960). The cutting-plane method for solving convex programs, *Journal of the Society for Industrial and Applied Mathematics* **8**(4): 703–712.

Lafond, M. (2012). Using Erlang C to compare PSAPS, *Public Safety Communications Magazine* pp. 30–38.

L'Ecuyer, P. (2007). Variance reduction's greatest hits, *Proceedings of the 2007 European Simulation and Modeling Conference*, EUROSIS, Ghent, Belgium, pp. 5–12.

Lewis, B. G., Herbert, R. D., Summons, P. F. & Chivers, W. J. (2007). Agent-based simulation of a multi-queue emergency services call centre to evaluate resource allocation, *in* L. Oxley & D. Kulasiri (eds), *MODSIM 2007, International Congress on Modelling and Simulation*, Modelling and Simulation Society of Australia and New Zealand, pp. 11–17.

Oreshkin, B., Régnard, N. & L'Ecuyer, P. (2016). Rate-based daily arrival process models with application to call centers, *Operations Research* **64**(2): 510–527.

Ta, A. (2013). *Staffing optimization with chance constrained in call centers*, Master's thesis, Département d'Informatique et de Recherche Opérationnelle, Université de Montréal.