Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

FLUID APPROXIMATIONS FOR A PRIORITY CALL CENTER WITH TIME-VARYING ARRIVALS

Ahmad D. Ridley

Department of Mathematics
University of Maryland, College Park
College Park, MD 20742, U.S.A.

Michael C. Fu

Robert H. Smith School of Business
University of Maryland, College Park
College Park, MD 20742, U.S.A.

William A. Massey

Department of Operations Research and Financial Engineering
Princeton University
Princeton, NJ 08544, U.S.A.

ABSTRACT

We model a call center as a preemptive-resume priority queue with time-varying arrival rates and two priority classes of customers. The low priority customers have a dynamic priority where they become high priority if their waiting-time exceeds a given service-level time. The performance of the call center is measured by the mean number in system for the two customer classes. A fluid approximation is proposed to estimate the mean number in system for each class. The quality of the approximation is tested by comparing it with a stochastic simulation model of the system. Finally, using the fluid approximations, we discuss how to compute the mean number in system for each class and estimate the overall staffing level, or number of agents.

1 INTRODUCTION

Telecommunication call centers have become the primary channel of customer interaction, sales, and service for many businesses. Traditionally, customers contacted the call center by talking to customer service representatives over the telephone. Now, customers can also contact an agent over the Internet (e.g., a Web chat session), by e-mail, or by fax. The growth of call centers has been substantial over the last two decades. According to some industry estimates, 4.5 million people worked in North American call centers in 1995, and over 10 million will have worked in call centers by 2004. In 1997, there were approximately $60,000$ to $90,000$ call centers in the United States, representing explosive growth above estimates from the late 1970's (Kim 1997). Currently, more than 50 percent of all business transactions are done over the telephone. Call centers are used by banks and financial institutions, computer product help desks, government offices, and other organizations to provide their customers with efficient and convenient service. Their managers and planners have a much more difficult job today than in the past. With far more products and services being sold and supported, call center managers struggle to deliver different service levels to different types of callers with different needs and issues. Today's call center technologies provide greater flexibility in routing and queueing calls. As a result, call center designers and managers can prioritize certain types of incoming calls and allow customers to access call agents with different skill sets. However, this makes planning and managing call centers more complex. Therefore, call centers have become an important part of today's business transactions.

A traditional call center has several main components typical of a telephone, or circuit-switched, network. These are namely, an automatic call distributor (ACD), an interactive voice response unit (IVR), desktop computers, and telephones. The ACD receives incoming customer calls, and distributes these calls to the call agents (Fischer et al. 1998). There are a finite number of trunks (i.e., telephone lines) connecting a set of customer service representatives (CSRs), or call agents, to the ACD. As customer calls arrive, the ACD receives them and routes them to an idle customer service representative (CSR), who provides the necessary service. If no CSR is available, the calls are routed to the IVR where customer transactions are handled automatically. Afterwards, if the customer chooses to speak with a call agent, they are either immediately routed to an available agent, or placed in a queue (i.e., on hold) if no agents are available. The call agents responds to calls routed to them using their telephone and desktop computer. For example, if the agent is answering a telephone call, he or she can access the customer information database through the desktop computer. The heart of a traditional call center is this dynamic routing of a new or pending call by the ACD to the most appropriate and available CSR. This call routing or assignment process must take into consideration such factors as the call priority, call arrival time, and CSR skills and availability. Thus, this call assignment process can become quite complex. It requires a dynamic, real-time management of all CSR skill levels and availability, the call/caller identity and status, and customer information databases. Consequently, there are several call center components that complete call routing and handling tasks, the heart of a call center.

2 PROBLEM SETTING

The basic structure of the call center can be described as a finite capacity, multi-server system. Customer calls arrive at the call center at varying rates on a finite number of trunks. These calls are terminated at the ACD/PBX switch and are routed to a group of call agents. In a multimedia call center, these calls can be voice, email, fax, or (eventually) video. Queueing models will be used to analyze the performance of the call center. Current analytical models applied in practice are based on classical Erlang queueing theory. Through his research on the telephone network in the early 1900's, Erlang showed that the arrival process of calls over the network to any destination could be modelled as a Poisson process (Hall 1991). Although these models are primarily used for producing daily call forecasts and agent work schedules, they do attempt to explain the randomness that exists in call centers. This randomness is caused by the variability of call arrival patterns and call durations.

Generally, our goal is to develop better estimates of a call center performance than the standard Erlang approximations. Specifically, we develop fluid models and a simulation model to approximate the mean number in the system. Our call center is a help desk with two customer classes and a preemptive-resume priority queue discipline. Here, we assume that there are enough telephone lines to prevent any call blocking. Also, we assume that the service level for the high priority class is high enough that no calls abandon the system. Note that in a general call center environment, these assumptions are not always valid.

In our model, upon entering the system, a high priority call can preempt a low priority call from service, when all the agents, or servers, are busy. Once prempted, the low priority call re-enters the beginning of the low priority queue. and waits for service. In general, once the call returns to a server, it resumes its service from its previous preemption point, i.e. it does not lose its previous service time. If the low priority call does not complete service within a gven amount of time, it will "abandon" its queue, and enter the end of the higher priority queue. In this regard, the low priority calls are guarenteed service within a reasonable time, and thus have a dynamic priority. Our goal is to show that the fluid approximations of the call center performance are close to the actual performance, as measured by a discrete-event simulation model. Figure 1 shows the diagram of our model.

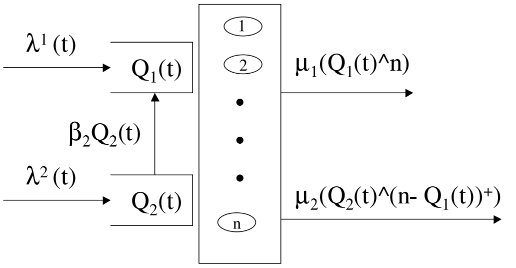

Figure 1: The Two-Customer Class $M_t/M/n$ Queue with Abandonment

Our model extends the model of Mandelbaum et al. (2001), which incorporates multiple-customer classes and dynamic priorities. We determine the fluid approximations for the mean number of high and low priority customers in the system. The low priority customers will have a dynamic priority, varying over time. Thus, at some point, these customers could be upgraded to the high priority class.

3 FLUID APPROXIMATIONS

Service systems models, such as call center models, belong to the class of stochastic service network models. These network models form a special family of non-stationary Markov processes where parameters such as inter-arrival and service rates are time-dependent. More importantly, these models have functional strong laws of large numbers and functional central limit theorem results for the mean number of customers in the system (Mandelbaum et al. 1998). The results are developed using an asymptotic limiting process, where the number of servers are scaled up in response to a scaling up of the arrival rates. The individual service and abandonment rates are not scaled. Note that the resulting limit theorems are diffusion, and not really heavy-traffic, limit results.

These limit theorems lead to a tractable set of fluid approximations represented as a system of ordinary differential equations (ODEs). By numerically solving these differential equations using the fourth-order Runge-Kutta method, we compute values for performance measures such as the mean number of customers in the system. More importantly, we can use this method to approximate the performance of our call center model, which is otherwise analytically intractable using Erlang analytical approximations (Mandelbaum et al. 1998). Therefore, we apply an alternative, possibly more robust, method to the performance analysis of service systems, such as call centers.

3.1 Two Customer Classes

Our fluid approximations for the mean number in the system will be derived for the two customer class, preemptive-resume priority, $M_t/M/n$ queue. Since the high priority customers can preempt the lower priority ones, these customers will essentially receive service as if no other type of customer is present in the system. Thus, the high priority customer class results will be almost the same as the results for the single customer class. The only difference is the dynamic priority process for the low priority customers, where these customers can abandon their queue and enter the high priority queue as a high priority customer. This process adds an extra term to the differential equations describing the process for the high priority customers.

Now, the M_t/M_t/n number in system, or queue length, process $Q \equiv \{Q(t) \mid t \geq 0\}$, as defined in Mandelbaum et al. (2001) for the single customer class case, must be defined for two customer classes.

3.2 Asymptotic Queue Length Results

The results and theorems presented in this section are adapted from those stated by Mandelbaum et al. (2001) and Mandelbaum et al. (1998). However, customers are now grouped into two classes: high priority and low priority. High priority customers are labelled as class-1 customers while low priority customers are labelled as class-2 customers. Thus, all of the random variables of the stochastic processes, discussed in Mandelbaum et al., are now random vectors. For example, the random variable $Q(t)$ is now defined as:

$$\mathbf{Q}(t) = \{Q_1(t), Q_2(t)\},$$

for all positive real numbers t. Here, $Q_1(t)$ and $Q_2(t)$ are the corresponding quantities for class-1 and class-2 customers, respectively.

Now, the limit theorem for the functional strong law of large numbers can be restated for our model. The initial conditions for the queue length process satisfy the following asymptotic assumption:

$$\lim_{\eta\to\infty} \frac{1}{\eta}\mathbf{Q}^{\eta}(0) = \mathbf{Q}^{(0)}(0) \text{ a.s.}, \tag{1}$$

where $\mathbf{Q}^{(0)}(0) = \left\{Q_1^{(0)}(0), Q_2^{(0)}(0)\right\}$ is constant. Thus, the functional strong law of large numbers theorem for our model is:

Theorem 3.1

$$\lim_{\eta\to\infty} \frac{1}{\eta}\mathbf{Q}^{\eta} = \mathbf{Q}^{(0)}, \text{ } a.s., \tag{2}$$

where the convergence is uniform on compact sets of t. Moreover, $\mathbf{Q}^{(0)} = \{\mathbf{Q}^{(0)}(t) \mid t \geq 0\} = \left\{Q_1^{(0)}(t), Q_2^{(0)}(t) \mid t \geq 0\right\}$ *is uniquely determined by* $\mathbf{Q}^{(0)}(0)$ *and the differential equation:*

$$\begin{aligned} \frac{d}{dt}Q_1^{(0)}(t) &= \lambda_1(t) - \mu(Q_1^{(0)}(t) \wedge n) \\ &\quad -\beta[Q_2^{(0)}(t) - (n - Q_1^{(0)}(t))^{+}]^{+}; \qquad (3) \\ \frac{d}{dt}Q_2^{(0)}(t) &= \lambda_2(t) - \mu[Q_2^{(0)}(t) \wedge (n - Q_1^{(0)}(t))^{+}]^{+} \\ &\quad -\beta[Q_2^{(0)}(t) - (n - Q_1^{(0)}(t))^{+}]^{+}, \qquad (4) \end{aligned}$$

where:

$$[Q_2^{(0)}(t) - (n - Q_1^{(0)}(t))^{+}]^{+}$$

is the number of customers in the low priority queue.

This theorem states rigorously that $\mathbf{Q}^{\eta} \approx \eta\mathbf{Q}^{(0)}$ for large η, where $\mathbf{Q}^{(0)}$ is called the *fluid approximation* for $\mathbf{Q}^{\eta}$. The proof of the theorem is given in Mandelbaum et al. (1998).

4 SIMULATION

4.1 Simulation Model

The final method used to compute the mean number in the system for high and low priority customers is a discrete-event simulation model. Our model approximates a $M_t/M/n$ queue where the arrival process is a non-stationary Poisson process, as described earlier. The inter-arrival times are independent and identically distributed (iid) random variables. They have an exponential distribution with a time-varying arrival rate, $\lambda(t)$. The model has two queues: one for high priority customers and one for low priority customers. The service times of the servers are iid random variables that are independent of the inter-arrival times. They have an exponential distribution with service rate μ_i. Since the customer priority does not depend on the service time value, the service time is generated once the customer reaches a server. Thus, each server can have a different associated service time distribution. The details of the algorithms used for the simulation model are outlined below.

4.2 Simulation Components

The C-program used to implement the simulation model consists of several components. The simulation starts in the empty-and-idle state, where no customers are present and all of the servers are idle. The basic inputs of the simulation are the arrival, service and abandonment rates for each customer class, the number of servers, the stopping time, and the target service levels for each customer class. One run of the simulation is repeated until a given stopping criteria is reached. Here, one run of the simulation is stopped after a finite horizon time is reached (20 hours). However, it can also be stopped after a certain number of customer completions. Independent replications of the simulation are performed until a certain precision of the performance measures is attained. In our case, replications of the simulation are performed until the standard error of the mean number in system reaches a precision of 0.0001. Finally, the random numbers, which model the stochastic nature, are generated using a pseudo-random number generator.

4.2.1 Arrival Process

One of the main components of the stochastic simulation is the arrival process. We choose to approximate the true arrival rate function as a piecewise linear function over a set of disjoint 30-minute time subintervals, $(t_a, t_{a+1}]$ which partition the overall finite-time horizon interval $[0, T]$, where $a = 1, 2, \ldots, m-1$ and m represents the number of 30-minute subintervals. Thus, the arrival time of the k-th customer, A_k, is used to advance the overall simulation time, S, where $S \leq T$, into the next time subinterval. We compute the arrival time, A_k, by generating a random inter-arrival time, X_k, between customer $k-1$ and k, and adding X_k to the current simulation time S. Since we have a Poisson arrival process, X_K has an exponential distribution with arrival rate $\lambda(t)$. Therefore, it can be generated using the inverse transform method.

Since our model supports two types of customers, $\lambda(t)$ is the overall arrival rate and is define as:

$$\lambda(t) = \lambda_1 + \lambda_2 \tag{5}$$

where the arrival rates for the high priority customers, $\lambda_1(t)$, and the low priority customers, $\lambda_2(t)$, also vary with time. Now, we randomly determine the call type of each customer upon their arrival. Here, based on Poisson thinning, a customer will have call type i with probability $\lambda_i(t)/\lambda(t)$.

Now, an arriving customer who finds at least one server idle enters service immediately at some server, n_i, $i = 1, 2, \ldots, n$, where n is the total number of servers. Server n_i is chosen from all the other idle server using an ordered search algorithm. In other words, if servers 1 and 2 are both idle, then server 1 is chosen to provide service. In practice, calls are switched to agents in this manner, although more efficient methods exist depending on the type of call centers. Once the customer enters service, we generate an exponentially distributed service time,Y_k, with mean service time $1/\mu$ (independent of time), for the k-th customer using the inverse transform method.

If all the servers are busy, the customer either enters the appropriate queue, or preempts a lower priority customer already in service. Now, if a customer enters the queue, then an abandonment time is computed depending on the priority class. If the arriving customer preempts a lower priority customer in service, then the preempted customer is placed at the head of the appropriate queue. The arriving customer is then sent to the vacant server. Finally, the departure time for the arriving customer, as well as the arrival time of the next customer, is generated.

4.2.2 Abandonment Process

There is the abandonment process from each of the two queues. If a high priority customer abandons its queue (queue 1), then it leaves the system completely. However, if a low priority customer abandons queue 2, then it is “upgraded" to queue 1. Note that the upgraded customer is placed at the end of queue 1, and its priority changes from low to high. Thus, this customer priority becomes *dynamic*. A customer that abandons queue 2 for queue 1 receives service as a high priority customer, but its performance is measured as if it is still a low priority customer.

4.2.3 Departure Process

The other main component is the departure process. Here, a customer leaves the system after completing service at some server, n_i. Thus, the server n_i is now available, and the next customer to enter service at server n_l is chosen based on its priority. If there are any customers in the high priority queue (queue 1), then the customer at the head of the queue will enter into service. If, however, there are no customers in queue 1, then the customer at the head of the low priority queue, queue 2, will enter into service. Of course, if there are no customers in either queue, then server n_i remains available, or idle, until a new customer enters the system.

4.3 Model Verification

We compare our simulation model results for the mean number in the system with known analytical and computational results. This allows us to verify the basic operation of our simulation, independent of our fluid approximations model. Since analytical results for queues with a non-stationary Poisson arrival process do not exist in closed analytical form, we assume a stationary Poisson arrival process in all models. First, we compare our model with the $M/M/n$, two-customer class, non-preemptive service priority queue for the mean number in system for each customer class. Second, we compare our model with the $M/M/1$, two-customer class, preemptive-resume service priority queue for the same performance measure. Third, we compare our model with a call center simulation model created by Rodney Wallace. Rodney is performing Ph.D research on modelling call centers with skill-based routing of calls to agents (Wallace 2003). Rodney's model can be adjusted to match our model. Thus, our model results can be verified by comparison with his computational results for the mean number in system. Finally, we reduce the number of customer classes from 2 to 1 by setting one of the inter-arrival rates close to 0. Then, we compare our model results to those from both the $M/M/1$ and $M/M/n$ queues for a the singel customer case. If our model is accurate, our single-customer class results will match those from the two queues for the mean number in the system values. Therefore, our stationary simulation model results can be verified with known analytical queue results.

5 COMPARISON RESULTS OF TWO METHODS

5.1 Overview

Our goal is to compare two different estimates of the mean number in system for both customer classes. Thus, we will compare our results from the fluid model to the simulation model for the $M_t/M/n$, two-class, preemptive-resume, dynamic priority queue.

5.2 Call Center Data

We begin our computation of numerical results by defining the queueing model parameters. The parameter values are taken from a real-world, helpdesk call center, in which calls represent requests for information technology (IT) support (e.g., network support, password resets, application support, etc.). The helpdesk is simulated over a 12-hour day in our fluid model and our simulation model. Thus, each independent replications of our simulates the performance of the helpdesk over the course of a day. All the rates used in the methods are per minute rates. Note that in the Laplace transform inversion method, the arrival rate, λ, is constant over time. However, in both the fluid and simulation methods, a piecewise constant function is used for the time-varying arrival rate function, λ_t. The duration of each value of λ_t is 30 minutes. Thus, λ_t varies every 30 minutes during the 12 hours, or 360 minutes of our time horizon. Figure 2 contains a graph of our inter-arrival rate parameter values.

We use a value of $n = 20$ where n is the number of agents, or servers. Since we are using asymptotic limits for the fluid approximations, we must scale the inter-arrival rates and the number of agents towards infinity in order to compute accurate estimates, as described in 3. The scale factor we use is 25. Our service rates for the high and low priority calls, μ_1 and μ_2 are approximately $\mu_1 = 0.1151$ customers per minute and $\mu_2 = 0.1151$ customers per minute respectively. For the single server, stationary queues, $n = 1$, $\mu_1 = 5$ customers per minute and $\mu_2 = 5$ customers per minute. Note that no scale factor is used for the stationary queues.

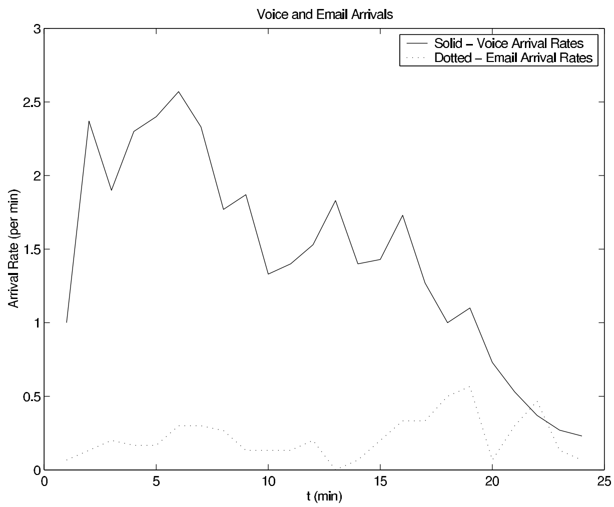

Figure 2: Graph of Time-Varying Inter-Arrival Rates for Both Customer Classes

Finally, the initial conditions of the call center model are that no high or low priority calls are present at the beginning of the day. Thus, in our fluid approximations method, $Q_1^\eta(0) = Q_2^\eta(0) = 0$.

5.2.1 Service Levels and Abandonment Rates

Additionally, the abandonment rate for the high priority customers, or voice calls, is measured in practice as a percentage of voice calls that enter the system. Usually, about 5 percent of customers abandon from the system. The abandonment rate for low priority customers is based on the target service level for the class. Although industry service levels vary greatly, we use a a target service level between 8 and 24 hours for a low priority call, such as an email. Thus, if a low priority customer has been in queue for about 8 hours, then that customer abandons the low priority queue. It then either enters the end of the high priority queue, or the end of a separate queue for upgraded low priority customers only. In both our fluid approximations and simulation methods, we model this low priority abandonment rate in two ways. First, we assume the abandonment rate, β, is deterministic with a value of $1/8$ customers per hour, or $1/480$ customers per minute. Note that here each low priority customer has the same abandonment time. Second, we assume that the abandonment rate, β, is exponentially distributed with a mean of $1/8$, or $1/480$ customers per minute. Thus, here each low priority customer has a different abandonment time.

6 NUMERICAL RESULTS

The numerical results compare the mean number in the system at time points t_i, spaced 30 minutes apart, for each customer class from the fluid approximations and the discrete-event simulation. Below, we show the graphical comparison of the results from our two models: the fluid approximations and the simulation. Figures 3 and 4 show the comparison of the mean number in the system for the high and low priority customers. Note that for most of the t_i's, the estimates are very close, especially for the high priority calls.

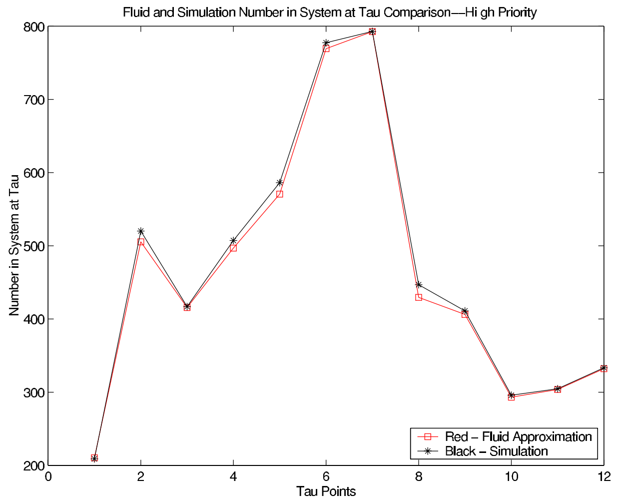

Figure 3: Fluid and Simulation Comparison of Number in System at τ_i - High Priority

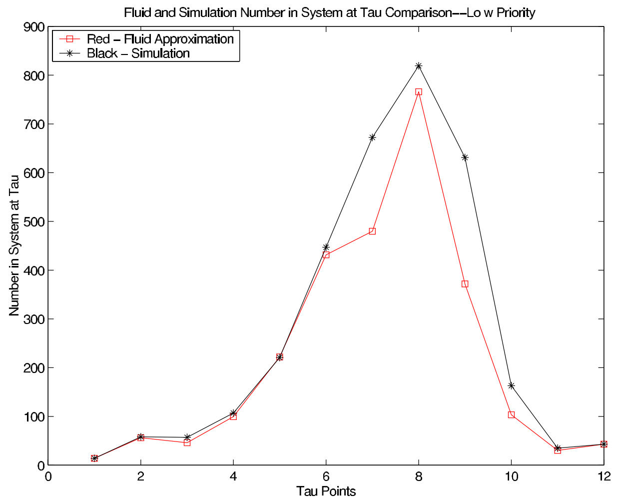

Figure 4: Fluid and Simulation Comparison of Number in System at τ_i - Low Priority

7 CONCLUSIONS AND FUTURE WORK

7.1 Conclusions

We obtain fairly accurate fluid approximations to the simulation results for the mean number in system for the high and low priority customer classes. Note that in the fluid approximations, the number of differential equations is in-

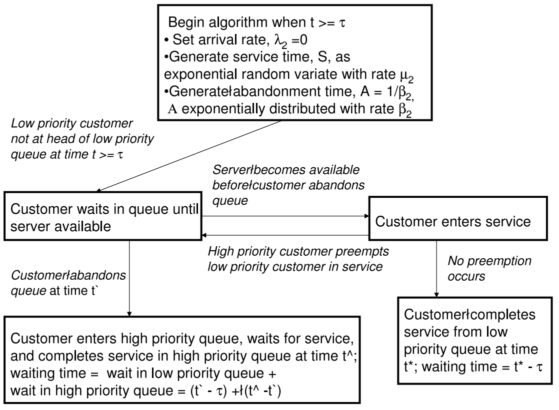

Figure 5: Outline of Low Priority Fluid Approximation Algorithm

dependent of the number of servers in the call center model. Thus, our approximations does not grow in complexity with the call center model. However, it is more likely that the simulation will increase in complexity with the call center model. Therefore, our fluid apprximation is a much more scalable solution than the simulation.

7.2 Future Work

We will use our fluid approximations and simulation to determine the mean virtual waiting time of the high and low priority customers by extending the waiting-time results in Mandelbaum et al. (1998) to the two-customer class case. The waiting time computation for the low priority customers is more complex than for the high priority customers. If these customers are preempted and abandon to the high priority queue, their waiting time will be a combination of their time in the low priority queue, their partial-service time at a server before each preemption, and their waiting-time in the high priority queue. Figure 5 outlines a suggested algorithm for computing the low priority waiting time.

We will then use those approximations to predict an actual staffing level for our call center model. Our criteria for changing the staffing level, or number of servers, in our model uses a comparison of the mean virtual waiting-time for each customer class to the corresponding target service level, or mean waiting-time. The simple staffing algorithm is the following:

1. Choose an initial staffing level, or value for the number of servers, and target service level for the high and low priority customers. These values are determined from our actual call center data.
2. Compute the mean virtual waiting-time using the fluid approximations for each customer class.
3. If the percentage of mean virtual waiting-times is greater than the target service level for either class, then increment the number of servers by 1.

4. Repeat the second step until the target service level is satisfied for both classes of customers.

We can substitute this predicted staffing level into our simulation. Finally, we will verify the accuracy of our staffing prediction by comparing the mean virtual waiting time for each class from the simulation with the target service level.

ACKNOWLEDGMENTS

The first two authors also thank Professor Jian-Qiang Hu of Boston University for many useful discussions.

REFERENCES

Fischer, M., D. Garbin, and Gharakhanian. 1998. Performance Modelling of Distributed Automatic Call Distribution Systems. *Telecommunications Systems-Modeling, Analysis, Design and Management* 9 (2): 133–152.

Hall, R. 1991. *Queueing Methods for Services and Manufacturing*. Prentice Hall International Series in Industrial and Systems Engineering. Englewood Cliffs, New Jersey: Prentice Hall.

Kim, G. 1997. Call Centers: Present and Future. *X-Change*: 26–29.

Mandelbaum, A., W. Massey, and M. Reiman. 1998. Strong Approximations for Markovian Service Networks. *Queueing Systems* 30: 149–201.

Mandelbaum, A., W. Massey, M. Reiman, R. Rider, and A. Stolyar. 2001. Queue Lengths and Waiting Times for Multiserver Queues with Abandonments and Retrials. *Proceedings of the Fifth INFORMS Telecommunications Conference*: 20–41.

Wallace, R. B. 2003. *Performance Modelling of Call Center with Skill-Based Routing*. Ph.D. Thesis, George Washington University.

AUTHOR BIOGRAPHIES

AHMAD D. RIDLEY completed his Ph.D in the Applied Mathematics and Scientific Computation program of the Department of Mathematics at the University of Maryland, College Park in September of 2003. His research on the performance analysis of call centers combines the areas of numerical analysis, probability and statistics, queueing theory, and stochastic simulation. He currently works as an engineer at Mitretek Systems, a non-profit scientific research and systems engineering organization. Mitretek conducts basic and applied research, and applies systems engineering techniques to create innovative technological solutions in the public interest. His e-mail address is `<aridley@mitretek.org>`.

MICHAEL C. FU is a Professor in the Decisions and Information Technologies at the Robert H. Smith School of Business at the University of Maryland, College Park. He has a joint appointment with the Institute for Systems Research and an affiliate appointment with the Department of Electrical and Computer Engineering, both in the Clark School of Engineering. His research interests include simulation modeling and analysis, production/inventory control, applied probability and queueing theory, with application to manufacturing and finance. In 1995 he received the Business School's Allen J. Krowe Award for Teaching Excellence. He is co-author of the book, *Conditional Monte Carlo: Gradient Estimation and Optimization Applications*, which was awarded the INFORMS College on Simulation Outstanding Publication Award in 1998. He is currently the Simulation Area Editor of Operations Research, and serves, or has recently served, on the editorial boards of Management Science, INFORMS Journal on Computing, IIE Transactions, and Production and Operations Management. His e-mail address is `<mfu@rhsmith.umd.edu>`.

WILLIAM A. MASSEY is an Edwin S. Wiley Professor in the Department of Operations Research and Financial Engineering at Princeton University. His research interests include performance and pricing models for telecommunication systems, queueing systems with time-varying rates, as well as asymptotic analysis and stochastic bounds for stochastic networks. He has over 50 publications in these fields. Before joining the Princeton faculty in 2001, he worked as a member of technical staff in the Mathematical Sciences Research Center at Bell Labs, Lucent Technologies for over 20 years. He is a member of INFORMS and the American Mathematical Society (AMS), and was a co-founding member of the Conference for African-American Researchers in the Mathematical Sciences (CAARMS). His e-mail address is `<wmassey@princeton.edu>`.