Evaluating the Erlang C and Erlang A Models for Call Center Modeling

Working Paper

Thomas R. Robbins
East Carolina University, Greenville, NC
(robbinst@ecu.edu)

D. J. Medeiros
Terry P. Harrison
Pennsylvania State University, University Park, PA
(djm3@psu.edu)
(tharrison@psu.edu)

We consider two queuing models commonly used to analyze call centers; the Erlang C and Erlang A models. The Erlang C is a very simple model that ignores caller abandonment and is the model most commonly used by practitioners and researchers. The Erlang A model allows for abandonment, but performance measures are more difficult to calculate. Several recent papers have advocated the use of the Erlang A model as a more accurate representation of the call center environment. We compare the theoretical performance predictions of these models to a steady state simulation model of a call center where many of the simplifying assumptions used in standard analytical models are relaxed. Our findings support the assertion that the Erlang A model is more accurate, but we find that in contrast to the Erlang C model, Erlang A tends to be optimistically biased. Our findings indicate that neither model clearly dominates the other in all situations and that care must be taken to select the correct model based on call center conditions and the intended purpose of the model.

1. Introduction

Call centers are an important part of many businesses and have become the primary message of communicating with customers for many companies. A call center is a facility designed to support the delivery of some interactive service via telephone communications; typically an office space with multiple workstations manned by agents who place and receive calls (Gans, Koole et al. 2003). Large scale call centers are technically and managerially sophisticated operations and have been the subject of substantial academic research. The literature focused on call centers is quite large, with thorough and comprehensive reviews provided in (Gans, Koole et al. 2003) and (Aksin, Armony et al. 2007). Empirical analysis of call center data is given in (Brown, Gans et al. 2005).

Call centers are examples of queuing systems; calls arrive, wait in a virtual line, and are then serviced by an agent. Call centers are often modeled using the M/M/N queue, or in industry standard terminology - the Erlang C model. The Erlang C model makes many assumptions which are questionable in the context of a call center environment. Specifically the Erlang C

model assumes that calls arrive at a Poisson process with a known average rate, and that they are serviced by a defined number of statistically identical agents with service times that follow an exponential distribution. Most significantly, Erlang C assumes that all callers wait as long as necessary for service without abandoning, *i.e.* hanging up. The model is used widely by both practitioners and academics.

Recognizing the deficiencies of the Erlang C model, many recent papers have advocated using alternative queuing models and staffing heuristics which account for conditions ignored in the Erlang C model. The most popular alternative is the Erlang A model, an extension of the Erlang C model that allows for caller abandonment. For example, in a widely cited review of the call center literature (Gans, Koole et al. 2003) , the authors state "*For this reason, we recommend the use of Erlang A as the standard to replace the prevalent Erlang C model*." Another widely cited paper examines empirical data collected from a call center (Brown, Gans et al. 2005) and these authors make a similar statement; "*using Erlang-A for capacity-planning purposes could and should improve operational performance. Indeed, the model is already beyond typical current practice (which is Erlang-C dominated), and one aim of this article is to help change this state of affairs*."

The purpose of this study is to evaluate the assertion that the Erlang A model is a superior representation of a call center environment. We conduct this analysis by performing a detailed simulation study. We develop a simulation model to predict steady state expected system performance based on a realistic set of modeling assumptions as identified in the literature. We compare key performance metrics from our simulation study to those predicted by the Erlang C and Erlang A models and seek to characterize the error in the theoretical predictions. In this paper we restrict the analysis to cases where the call center has sufficient capacity to handle all calls without abandonment; sometimes referred to as the *quality-driven* and the *quality and efficiency-driven* (QED) regimes (Gans, Koole et al. 2003).

Our findings confirm that the Erlang A model is indeed a more accurate model in the sense that it makes predictions which, over a wide range of input conditions, result in a lower error. However, we also find that Erlang A does not dominate Erlang C under all conditions; in other words there are situations in which the Erlang C model provides a better estimate, even in cases where the abandonment level is non-negligible. Furthermore, we find that while the Erlang C model tends to provide a pessimistic estimate (*i.e.*, the system performs better than predicted),

the Erlang A model often provides an optimistic estimate. While it is well established that Erlang C-based work force management systems tend to overstaff the call center (Gans, Koole et al. 2003) p. 105, we conclude that the use of the Erlang A model may lead to understaffing.

The remainder of this paper is organized as follows. In Section 2 we review the Erlang C and Erlang A models and highlight the relevant literature. In Section 3 we present a general model of a steady state call center environment and review the simulation model we developed to evaluate it. In Section 4 we evaluate the performance of the Erlang C model, while section 5 evaluates the performance of the Erlang A model. In Section 6 we compare the two models. We conclude in Section 7 with summary observations and identify future research questions.

2. Queuing Models and the Associated Literature

Call centers are often modeled as queuing systems. Queuing models are used to estimate system performance so that the appropriate staffing level can be determined in order to achieve a desired performance metric such as the Average Speed to Answer, or the Abandonment rate. The most common queuing model used for inbound call centers is the Erlang C model (Gans, Koole et al. 2003; Brown, Gans et al. 2005). The Erlang C model (M/M/N queue) is a very simple multi-server queuing system as depicted in Figure 1.

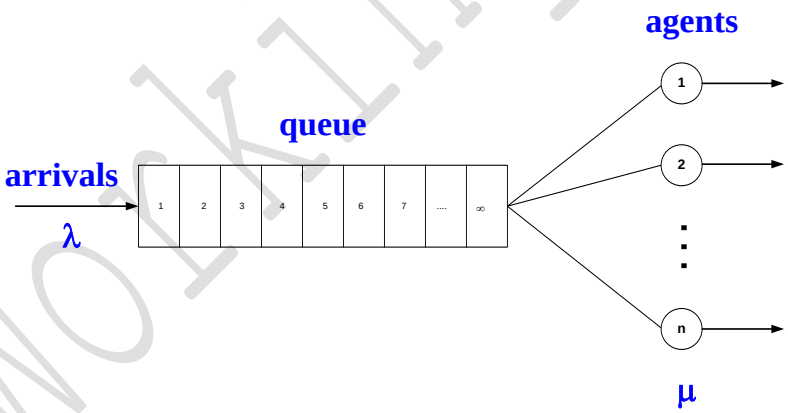

Figure 1 - Erlang C Queuing Model

Calls arrive according to a Poisson process at an average rate of λ. By the nature of the Poisson process, interarrival times are independent and identically distributed exponential random variables with mean λ^{-1}. Calls enter an infinite length queue and are serviced on a First Come – First Served (FCFS) basis. All calls that enter the queue are serviced by a pool of n homogene-

ous (statistically identical) agents at an average rate of $n\mu$. Service times follow an exponential distribution with a mean service time of μ^{-1}.

The steady state behavior of the Erlang C queuing model is easily characterized, see for example (Gans, Koole et al. 2003). The *offered load*, a unit-less quantity often referred to as the number of Erlangs, is defined as

$$R \triangleq \lambda/\mu \tag{1.1}$$

The offered utilization (aka utilization, traffic intensity or occupancy) is defined as

$$\rho \triangleq \lambda/(N\mu) = R/N \tag{1.2}$$

The offered utilization represents the proportion of available agent time spent handling calls under the assumption that all calls are serviced. Given the assumption that all calls are serviced, the offered utilization must be strictly less than one or the system becomes unstable, *i.e.* the queue grows without bound.

This system can be analyzed by solving a set of balance equations and the resulting steady state probability that all *N* agents are busy is

$$P\{Wait > 0\} = 1 - \left(\sum_{m=0}^{N-1} \frac{R^m}{m!}\right) \Bigg/ \left(\sum_{m=0}^{N-1} \frac{R^m}{m!} + \left(\frac{R^m}{N!}\right)\left(\frac{1}{1-R/N}\right)\right) \tag{1.3}$$

Equation (1.3) calculates the proportion of callers that must wait prior to service. Another relevant performance measure for call centers managers is the ***Average Speed to Answer*** (ASA).

$$\begin{aligned} ASA &\triangleq E[Wait] = P\{Wait > 0\} \cdot E[Wait \mid Wait > 0] \\ &= P\{Wait > 0\} \cdot \left(\frac{1}{N}\right) \cdot \left(\frac{1}{\mu_i}\right) \cdot \left(\frac{1}{1-\rho_i}\right) \end{aligned} \tag{1.4}$$

A third important performance metric for call center managers is the ***Telephone Service Factor*** (TSF), also called the "service level." The TSF is the fraction of calls presented which are eventually serviced and for which the delay is below a specified level. For example, a call center may report the TSF as the percent of callers on hold less than 30 seconds. The TSF metric can then be expressed as

$$\begin{aligned} TSF &\triangleq P\{Wait \le T\} = 1 - P\{Wait > 0\} \cdot P\{Wait > T \mid Wait > 0\} \\ &= 1 - C(N, R_i) \cdot e^{-N\mu_i(1-\rho_i)T} \end{aligned} \tag{1.5}$$

A fourth performance metric monitored by call center managers is the ***Abandonment Rate;*** the proportion of all calls that leave the queue (hang up) prior to service. Abandonment rates cannot be estimated directly using the Erlang C model because the model assumes no abandonment occurs.

A substantial amount of research analyzes the behavior of Erlang C model; much of it seeks to establish simple staffing heuristics based on asymptotic frameworks applied to large call centers. (Halfin and Whitt 1981) develop a formal version of the square root staffing principle for M/M/N queues in what has become known as the Quality and Efficiency Driven (QED) regime. (Borst, Mandelbaum et al. 2004) develop a framework for asymptotic optimization of a large call center with no abandonment.

As is the case with any analytical model, the Erlang C model makes many assumptions, several of which are not wholly accurate. In the case of the Erlang C model several assumptions are questionable, but the most problematic is the "no abandonment" assumption, as even low levels of abandonment can dramatically impact system performance (Gans, Koole et al. 2003). Many call center research papers however analyze call center characteristics under the assumption of no abandonment (Jennings, Mandelbaum et al. 1996; Green, Kolesar et al. 2001; Green, Kolesar et al. 2003; Borst, Mandelbaum et al. 2004; Wallace and Whitt 2005; Gans and Zhou 2007).

The Erlang C model assumes also that calls arrive according to a Poisson process. The interarrival time is a random variable drawn from an exponential distribution with a known arrival rate. Several authors assert that the assumption of a known arrival rate is problematic. Both major call center reviews (Gans, Koole *et al.* 2003; Aksin, Armony *et al.* 2007) have sections devoted to arrival rate uncertainty. (Brown, Gans et al. 2005) perform a detailed empirical analysis of call center data. While they find that a time-inhomogeneous Poisson process fits their data, they also find that arrival rate is difficult to predict and suggest that the arrival rate should be modeled as a stochastic process. Many authors argue that call center arrivals follow a doubly stochastic process; a Poisson process where the arrival rate is itself a random variable (Chen and Henderson 2001; Whitt 2006c; Aksin, Armony et al. 2007; Robbins and Harrison 2010). Arrival rate uncertainty may exist for multiple reasons. Arrivals may exhibit randomness greater than that predicted by the Poisson process due to unobserved variables such as the weather or advertising. Call center managers attempt to account for these factors when they develop forecasts, yet forecasts may be subject to significant error. (Robbins 2007) compares four months of week-

day forecasts to actual call volume for 11 call center projects. He finds that the average forecast error exceeds 10% for 8 of 11 projects, and 25% for 4 of 11 projects. The standard deviation of the daily forecast to actual ratio exceeds 10% for all 11 projects. (Steckley, Henderson et al. 2009) compare forecasted and actual volumes for nine weeks of data taken from four call centers. They show that the forecasting errors are large and modeling arrivals as a Poisson process with the forecasted call volume as the arrival rate can introduce significant error. (Robbins, Medeiros et al. 2006) use simulation analysis to evaluate the impact of forecast error on performance measures demonstrating the significant impact forecast error can have on system performance.

Several recent papers consider the issue of setting staffing level requirements in the face of arrival rate uncertainty. (Bassamboo, Harrison et al. 2005) develop a model that attempts to minimize the cost of staffing plus an imputed cost for customer abandonment for a call center with multiple customer and server types when arrival rates are variable and uncertain. (Harrison and Zeevi 2005) use a fluid approximation to solve the sizing problem for call centers with multiple call types, multiple agent types, and uncertain arrivals. (Whitt 2006c) allows for arrival rate uncertainty as well as uncertain staffing, *i.e.* absenteeism, when calculating staffing requirements. (Steckley, Henderson et al. 2004) examine the type of performance measures to use when staffing under arrival rate uncertainty. (Robbins and Harrison 2010) develop a scheduling algorithm using a stochastic programming model that is based on uncertain arrival rate forecasts.

The Erlang C model also assumes that the service time follows an exponential distribution. The memoryless property of the exponential distribution greatly simplifies the calculations required to characterize the system's performance, and makes possible the relatively simple equations (1.3)-(1.5). If the assumption of exponentially distributed talk time is relaxed, the resulting queuing model is the $M/G/N$ queue, which is analytically intractable (Gans, Koole et al. 2003) and approximations are required. However, empirical analysis suggests that the exponential distribution is a relatively poor fit for service times. Most detailed analysis of service time distributions find that the lognormal distribution is a better fit (Mandelbaum A., Sakov A. et al. 2001; Gans, Koole et al. 2003; Brown, Gans et al. 2005).

Finally, the Erlang C model assumes that agents are *homogeneous*. More precisely, it is assumed that the service times follow the same statistical distribution independent of the specific agent handling the call. Empirical evidence supports the notion that some agents are more effi-

cient than others and the distribution of call time is dependent on the agent to whom the call is routed. In particular more experienced agents typically handle calls faster than newly trained agents (Armony and Ward 2008). (Robbins 2007) demonstrated a statistically significant learning curve effect in an IT help desk environment. Many of the assumptions of the Erlang C model are questionable in the context of a real call center. The fit of the Erlang C model in a call center environment is analyzed in (Robbins, Medeiros et al. 2010).

2.1. **The Erlang A Model**

Given the prevalence of caller abandonment in modern call centers, the *no abandonment* assumption of the Erlang C model may be problematic. Unfortunately, models that allow for abandonment are significantly more complex and difficult to characterize. The simplest abandonment model is the $M/M/N+M$, or Erlang A model. The model was originally presented by Palm in a 1946 paper written in Swedish. It was presented in English in (Palm 1957) . The Erlang A model is presented in detail in (Gans, Koole et al. 2003) and (Mandelbaum and Zeltyn 2004).

Erlang A extends the Erlang C model by allowing abandonment. In the Erlang A model each caller posses an exponentially distributed *patience time* with mean θ^{-1}. If the offered waiting time, the time a caller with infinite patience would be required to wait, exceeds the customer's patience time, the caller will abandon the queue and hang up (Mandelbaum and Zeltyn 2004). While the exponentially distributed patience time makes the calculations tractable, they are by no means straightforward. In particular, calculation of the performance metrics requires an evaluation of the incomplete Gamma function

$$\gamma(x,y) \triangleq \int_0^y t^{x-1}e^{-t}dt, \quad x>0, \quad y\geq 0$$

Details on how to calculate performance metrics for the Erlang A model are provided in (Mandelbaum and Zeltyn 2009). Following their notation, we define the basic building blocks J as

$$J = \frac{e^{\lambda/\theta}}{\theta}\cdot\left(\frac{\theta}{\lambda}\right)^{\frac{n\mu}{\theta}}\cdot\gamma\left(\frac{n\mu}{\theta},\frac{\lambda}{\theta}\right)$$

and ε as

$$\varepsilon = \frac{\sum_{j=0}^{n-1} \frac{1}{j!}\left(\frac{\lambda}{\mu}\right)^j}{\frac{1}{(n-1)!}\left(\frac{\lambda}{\mu}\right)^{n-1}}$$

We can then calculate the probability of waiting as

$$P\{Wait > 0\} = \frac{\lambda J}{\varepsilon + \lambda J} \cdot (1-\theta)$$

(Garnett, Mandelbaum et al. 2002) outline a method for an exact calculation of the Erlang A performance metrics, and also provide approximations based on an asymptotic analysis of the queue. These same authors provide a downloadable software tool, 4CallCenters, to perform these calculations (Garnett and Mandelbaum 2002). (Whitt 2006a) develops deterministic fluid models to provide simple first-order performance descriptions for multiserver queues with abandonment under heavy loads.

The inclusion of abandonment has a profound effect on the performance of the queuing system. The impact of abandonment is discussed in detail in (Garnett, Mandelbaum et al. 2002). First, the issue of system stability is no longer a concern. In an Erlang C system the traffic intensity defined in equation (1.2) must be strictly less than one for a steady state to exist; an intensity of one or more leads to an infinite queue size. No such limit exists when using Erlang A. Furthermore, even very low levels of caller abandonment can dramatically alter system performance. Comparisons of Erlang C and Erlang A models are developed in (Mandelbaum and Zeltyn 2004) and (Garnett, Mandelbaum et al. 2002). (Whitt 2005) examines the fit of the Erlang A model. (Whitt 2006b) examines the sensitivity of the Erlang A model to changes in the model parameters. Several papers examine staffing and scheduling issues in call centers were abandonment is allowed (Bassamboo, Harrison et al. 2005; Avramidis, Gendreau et al. 2007; Robbins and Harrison 2010).

In order to develop a tractable model, the Erlang A model assumes an exponentially distributed patience. (Brown, Gans et al. 2005) examine abandonment and a customer's willingness to wait in detail. A customer's patience is in general an unobservable metric; since only customers whose patience expires abandon, the data is right censored. The exponential distribution of patience implies that the hazard rate for abandonment is constant over time. (Brown, Gans et al. 2005) and (Gans, Koole et al. 2003)show hazard rate graphs estimated from empirical data for two different call types. Both graphs reveal hazard functions that are not constant; in contrast

they show a sharp peak near the origin indicating a substantial portion of customers are unwilling to wait at all. Callers who abandon immediately are said to *balk*. The graphs also show another peak at 60 seconds after an announcement indicating a customer's position in the queue. The hazard function shows a general decline over the range of values plotted (0 to 400 secs.) Several other studies of patience curves have concluded that patience can be best modeled as a Weibull distribution (Gans, Koole et al. 2003). The Weibull distribution supports a constant, increasing, or decreasing hazard rate.

While many papers have noted the deficiencies of the Erlang C model, and advocated the use of the Erlang A model, a systematic analysis of the error associated with each model is lacking. Our paper seeks to close this gap in the literature.

3. Call Center Simulation

3.1. The Modified Model

In this section we present a revised model of a call center, relaxing several key assumptions discussed previously. In our model calls arrive at a call center according to a Poisson process. Calls are forecasted to arrive at an average rate of $\hat{\lambda}$. The realized arrival rate is λ, where λ is a normally distributed random variable with mean $\hat{\lambda}$ and standard deviation σ_λ. The time required to process a call by an average agent is a lognormally distributed random variable with mean μ^{-1} and standard deviation σ_μ. Arriving calls are routed to the agent who has been idle for the longest time if one is available. If all agents are busy the call is placed in a FCFS queue. When placed in queue a proportion of callers will balk; *i.e.* immediately hang up. Callers who join the queue have a patience time that follows a Weibull distribution. If wait time exceeds their patience time the caller will abandon. Calls are serviced by agents who have variable relative productivity r_i. An agent with a relative productivity level of 1 serves calls at the average rate. An agent with a relative productivity level of 1.5 serves calls at 1.5 times the average rate. Agent productivity is assumed to be a normally distributed random variable with a mean of 1 and a standard deviation of σ_r.

3.2. **Experimental Design**

In order to evaluate the performance of the Erlang C and Erlang A models against the simulation model, we conduct a series of designed experiments. Based on the assumptions for our call center discussed previously, we define the following set of nine experimental factors. We also define a range of values for these parameters that give us a reasonable representation of a variety of call center environments.

	Factor	**Low**	**High**
1	Number of Agents	10	100
2	Offered Utilization ($\hat{\rho}$)	65%	95%
3	Talk Time (mins)	2	20
4	Patience β	60	600
5	Forecast Error CV	0	.2
6	Patience α	.75	1.25
7	Talk time CV	.75	1.25
8	Probability of Balking	0	.25
9	Agent Productivity Standard Deviation	0	.15

Table 1-Experimental Factors

The forecasted arrival rate in the simulation is a quantity derived from other experimental factors by

$$\hat{\lambda} = \hat{\rho} N \mu \tag{1.6}$$

Given the relatively large number of experimental factors, a well designed experimental approach is required to efficiently evaluate the experimental region. A standard approach to designing computer simulation experiments is to employ either a full or fractional factorial design (Law 2007). However, the factorial model only evaluates corner points of the experimental region and implicitly assumes that responses are linear in the design space. Given the anticipated non-linear relationship of errors, we chose instead to implement a Space-Filling Design based on Latin Hypercube Sampling as discussed in (Santner, Williams et al. 2003). Given a set of *d* experimental factors and a desired sample of *n* points, the experimental region is divided into n^d cells. A sample of *n* cells is selected in such a way that the centers of these cells are uniformly spread when projected onto each of the *d* axes of the design space. While the LHS design is not perfectly orthogonal like a factorial design, the design does provide for a low correlation be-

tween input factors greatly reducing the risk of multicollinearity. We chose our design point as the center of each selected cell. This experimental design allows us to select an arbitrary number of points for any experiment.

3.3. **Simulation Model**

Our call center model is evaluated using a straightforward discrete event simulation model. The purpose of the model is to predict the long term, steady state behavior of the queuing system. The model generates random numbers using a combined multiple recursive generator (CMRG) based on the Mrg32k3a generator described in (L'Ecuyer 1999). Common random numbers are used across design points to reduce output variance. To reduce any start up bias we use a warm up period of 5,000 calls, after which all statistics are reset. The model is then run until 25,000 calls have been serviced and summary statistics are collected. For each design point we repeat this process for 500 replications and report the average value across replications. Our primary analysis is based on an experiment with 1,000 design points.

The specific process for each replication is as follows. The input factors are chosen based on the experimental design. The average arrival rate is calculated based on the specified talk time, number of agents, and offered utilization rate according to equation (1.6). A random number is drawn and the realized arrival rate is set based on the probability distribution of the forecast error. That arrival rate is then used to generate Poisson arrivals for the replication. Agent productivities are generated using a normal distribution with mean one and standard deviation σ_p. Each new call generated includes an exponentially distributed interarrival time, a lognormally distributed average talk time, a Weibull distributed time before abandonment, and a Bernoulli distributed balking indicator. When the call arrives it is assigned to the longest idle agent, or placed in the queue if all agents are busy. If sent to the queue the simulation model checks the balking indicator. If the call has been identified as a balker it is immediately abandoned, if not an abandonment event is scheduled based on the realized time to abandon. Once the call has been assigned to an agent, the realized talk time is calculated as the product of the average talk time and the agent's productivity. The agent is committed for the realized talk time. When the call completes the agent processes the next call from the queue, or, if no calls are queued becomes idle. If a call is processed prior to its time to abandon, the abandonment event is cancelled. If not, the call is abandoned and removed from the queue. Over the course of the simulation we collect sta-

tistics on the proportion of customers forced to wait, the average speed to answer, the abandonment rate, and the TSF defined as the proportion of callers waiting less than 30 seconds. Extensive testing of our simulation model verifies that all metrics are calculated consistently with the Erlang C and Erlang A predictions when the simulation is configured to support those model's assumptions.

After all replications of the design point have been executed the results are compared to the theoretical predictions of the Erlang C and Erlang A models. In each case we calculate the error as the difference between the theoretical value and the simulated value. For the Erlang C model we use the standard analytical calculations using the same values of arrival rate, talk time, and the number of agents used in the simulation. When testing against the Erlang A model the comparison is a bit more complicated. The first challenge is we wish to eliminate any approximation errors in our comparison, so rather than use an approximate calculation for the Erlang A model we rerun the simulation configured to be consistent with the Erlang A model assumptions, *i.e.* no balking, homogeneous agents, exponential talk time and exponential patience. The simulation is run using common random numbers from the original simulation. We feel that this approach allows us to focus on the error associated with the Erlang A assumptions, rather than the numerical issues associated with estimating Erlang A performance measures. The second challenge is how to set the patience parameter for the Erlang A calculation. Recall that this parameter is not directly observable since data is heavily censored. Since we are attempting to fit the Erlang A model to observed data, we approximate the Erlang A parameter θ with observed values as in (Gans, Koole et al. 2003) and (Brown, Gans et al. 2005) by

$$\theta = \frac{P\{Abandon\}}{E[Wait]} \tag{1.7}$$

4. Erlang C Experimental Analysis

4.1. Summary Observations

We conducted an experiment with 1,000 design points. Based on our analysis we can make the following summary observations:

- The Erlang C model is, on average, subject to a reasonably large error over this range of parameter values.
- Measurement errors are strongly correlated across performance measures.

- The Erlang C model is on average pessimistically biased (the real system performs better than predicted) but may become optimistically biased when utilization is high and arrival rates are uncertain.
- Measurement error is high when the real system exhibits high levels of abandonment. The error is strongly positively correlated with the realized abandonment rate.
- The Erlang C model is most accurate when the number of agents is large and utilization is low.
- Errors decrease as caller patience increases.

We now review our experimental results in more detail.

4.2. **Correlation and Magnitude of Errors**

The magnitude of errors generated by using the Erlang C model across our test space is high on average, and very high in some cases. Predicted and simulated values , and error magnitudes are summarized in Table 2 for the five primary performance measures

	Erlang C Prediction			Simulation			Error (Prediction - Simulation)			
	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	% Positive
Prob Wait	0.00%	17.85%	77.53%	0.01%	9.89%	50.10%	-7.99%	7.96%	49.39%	71.80%
ASA	0.000	36.683	1117.935	0.000	3.282	31.499	-2.798	33.401	1098.963	83.60%
TSF	26.53%	86.50%	100.00%	66.21%	94.30%	100.00%	-51.61%	-7.80%	3.61%	28.60%
Abandonment Rate	0.00%	0.00%	0.00%	0.00%	2.40%	14.29%	-14.29%	-2.40%	0.00%	0.00%
Utilization	65.02%	79.99%	94.99%	63.16%	77.07%	90.86%	0.03%	2.93%	13.96%	100.00%

Table 2 - Erlang C Analysis Metrics

With the exception of ASA, all the metrics are percentages and therefore bound to values between zero and one. As an unbounded measure, ASA has a long right tail. Because Erlang C assumes no abandonment it forecasts a very long average wait time with utilization is high. In our worst-case scenario ASA is more than 1100 seconds. But, since we assume that real callers have finite patience, the actual maximum ASA is much smaller at about 31.5 seconds, and the error rate is very high. The errors across the key metrics are highly correlated with each other, and highly correlated with the realized abandonment rate. Table 3 shows a correlation matrix of the errors generated from the Erlang C model and the abandonment rate calculated from the simulation.

	Simulated Abandonment Rate	*Prob Wait Error*	*ASA Error*	*TSF Error*	*Utilization Error*
Simulated Abandonment Rate	1.000				
Prob Wait Error	.867	1.000			
ASA Error	.766	.722	1.000		
TSF Error	-.880	-.987	-.759	1.000	
Utilization Error	.970	.861	.745	-.873	1.000

Table 3 – Correlation Matrix for Erlang C Model

Correlations between measurement errors are strong. The errors all move, on average, in an optimistic or pessimistic direction together. ProbWait and ASA are positively correlated; it is desirable for both these measure to be low. ProbWait is negatively correlated with TSF; a measure we want to be high. Measurement error is also highly correlated with abandonment rate. Given the high correlation between measures we will utilize ProbWait as a proxy for the overall error of the Erlang C model. Additional data on the Average Speed to Answer metric is provided in the on-line supplement to this paper.

In Figure 2 we show a scatter plot of the ProbWait predicted by the Erlang C model and the corresponding value from the simulation.

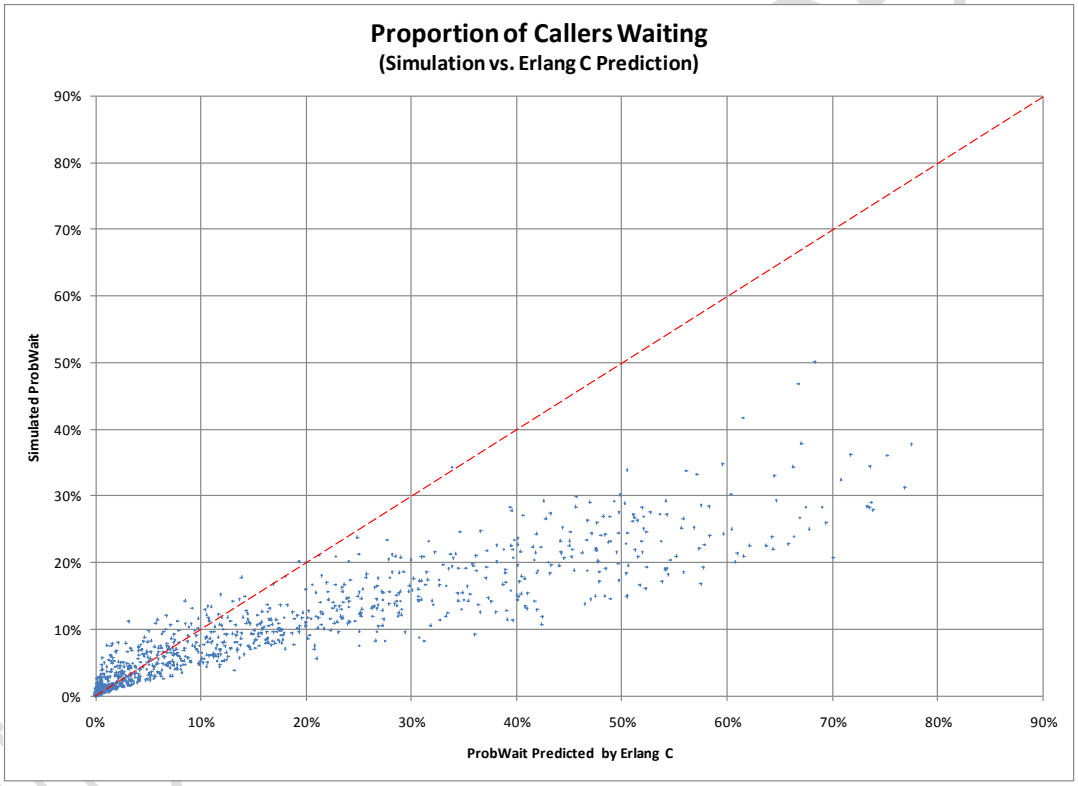

Figure 2 - ProbWait Predicted by Erlang C vs. Simulated

The dashed line in this figure represents points where the predicted value equals the simulated value. Points to the right of the line indicate scenarios where the simulated system performed better than predicted; a situation we refer to as a pessimistic prediction. The graph shows that for relatively high values of ProbWait this system performs substantially better than predicted. Average error rates are reasonably high under the Erlang C model, with errors being pessimistically skewed. Figure 3 shows a histogram of the ProbWait error.

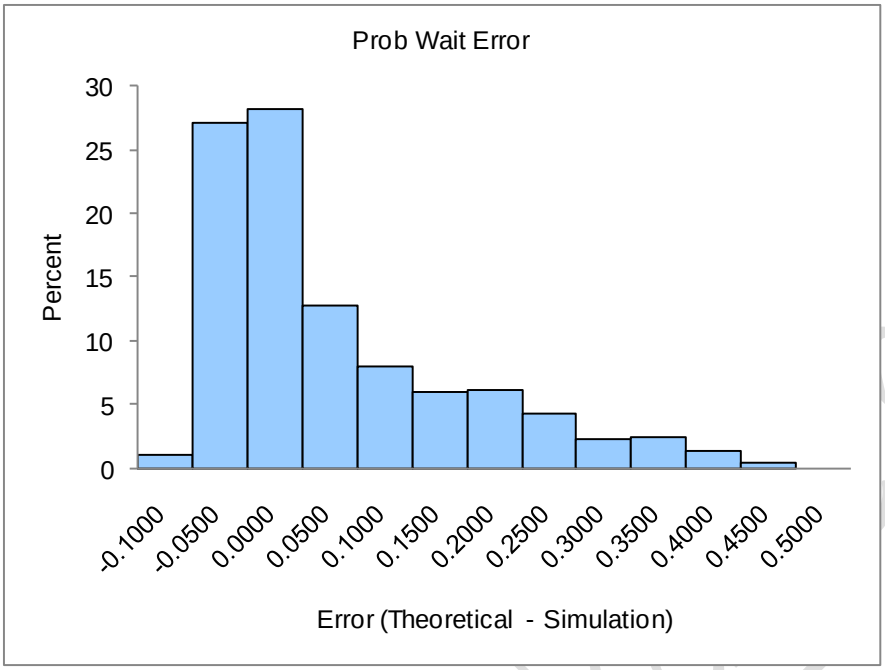

Figure 3 – Histogram of Erlang C Prob Wait Errors

The average error is 7.96%, and the data has a strong positive skew; 72% of the errors being positive. The ProbWait error has a sample skewness of 1.30. The largest error is 49.4%, the smallest is -8.0%. A positive error implies that the model predicted a higher proportion of calls would have to wait than actually waited. In other words the system performed better than predicted. In this situation the prediction was conservative, or pessimistic estimate, assuming the system would behave worse than it did. A negative error implies an optimistic bias, assuming the system would behave better than it did. Our data shows that by ignoring abandonment our system tends to make pessimistic estimates; the system behaves better than the model predicts. It is somewhat paradoxical that abandonment improves overall performance, but since some callers chose to exit the queue and get out of the way, the time spent waiting by callers that do not abandon is reduced and fewer callers must wait at all.

4.3. **Drivers of Erlang C Error**

Having established that error rates are high under the Erlang C model, we now turn our attention to characterizing the drivers of that error. As discussed in the previous section, Erlang C errors are highly correlated with the realized abandonment rate. The notion that abandonment is a major driver of errors in the Erlang C model is further illustrated in Figure 4.

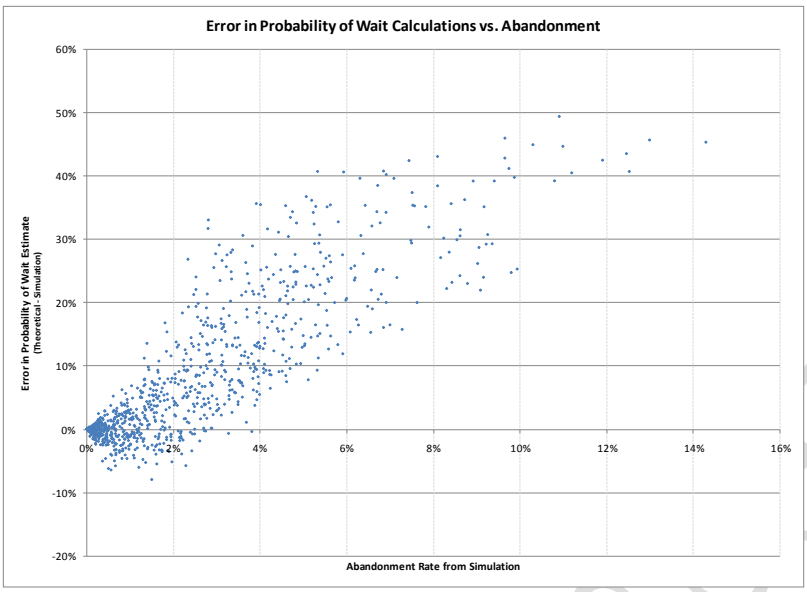

Figure 4 – Scatter Plot of Erlang C Errors and Abandonment Rate

This graph shows the error in the ProbWait measure on the vertical axis and the abandonment rate from the simulation analysis on the horizontal axis. The graph clearly shows that as abandonment increases, the error in the ProbWait measure increases as well. The graph also reveals the optimistic errors, *i.e.* errors in which the system performed worse than predicted, only occur with relatively low abandonment rates. The average abandonment rate for optimistic predictions was .74%. The graph also reveals that significant error can be associated with even low to moderate abandonment rates. For example, for all test points with abandonment rates of less than 5%, the average error for ProbWait is 4.8%. For test points in which abandonment ranged between 2% and 5% the average ProbWait error is 12.2%.

To assess how each of the nine experimental factors impacts the error, we perform a regression analysis with ProbWait error as the dependent variable. For the independent variable we use the nine experimental factors normalized to a [-1,1] scale. This normalization allows us to better assess the relative impact of each factor. The LHS sampling method provides an experimental design where the correlation between experimental factors is low, greatly reducing risks of multicollinearity. The results of the regression analysis are shown in Table 4.

Regression Analysis

R^2	0.746		
Adjusted R^2	0.744	n	1000
R	0.864	k	9
Std. Error	0.058	Dep. Var.	**Prob Wait Error**

ANOVA table

Source	*SS*	*df*	*MS*	*F*	*p-value*
Regression	9.6689	9	1.0743	323.87	8.38E-288
Residual	3.2839	990	0.0033		
Total	12.9529	999			

Regression output

					confidence interval	
variables	*coefficients*	*std. error*	*t (df=990)*	*p-value*	*95% lower*	*95% upper*
Intercept	0.0797	0.0018	43.745	1.52E-233	0.0761	0.0832
Num Agents	-0.0721	0.0032	-22.778	1.11E-92	-0.0783	-0.0658
Offered Utilization	0.1500	0.0032	47.365	1.09E-256	0.1438	0.1562
Talk Time	0.0184	0.0032	5.829	7.53E-09	0.0122	0.0246
Patience	-0.0134	0.0032	-4.233	2.52E-05	-0.0196	-0.0072
AR CV	-0.0260	0.0032	-8.206	7.05E-16	-0.0322	-0.0198
Talk Time CV	-0.0035	0.0032	-1.096	.2734	-0.0097	0.0027
Patience Shape	-0.0027	0.0032	-0.858	.3912	-0.0089	0.0035
Probability of Balking	0.0228	0.0032	7.172	1.44E-12	0.0165	0.0290
Agent Heterogeneity	0.0050	0.0032	1.585	.1133	-0.0012	0.0112

Table 4 - Regression Analysis of ProbWait Errors – Erlang C

The model is statistically significant with a relatively high R^2 value of .746. Given the normalization of the experimental factors, the magnitude of the regression coefficients provides a direct assessment of the impact that each factor has on the measurement error. The factor that most strongly influences the error is the offered utilization, the magnitude of its coefficient being more than twice the value of the next measure and more than five times the magnitude of all other factors. The size of the call center, measured as the number of agents, has a major impact on errors. Factors related to willingness to wait, *i.e.* Patience, Patience Shape, and Probability of Balking, all have low to moderate impacts, but with the exception of Patience Shape are statistically significant. Talk time is also a statistically significant factor with a moderate impact, though the variability of talk time is not statistically significant. Agent heterogeneity has a low impact that is not statistically significant.

The regression analysis indicates that the most important drivers of Erlang C errors are the size and utilization of the call center. This is further illustrated in Figure 5. This graph shows the results of a separate experiment where the number of agents and utilization factors are varied in a controlled fashion. All other experimental factors are held at their mid-point.

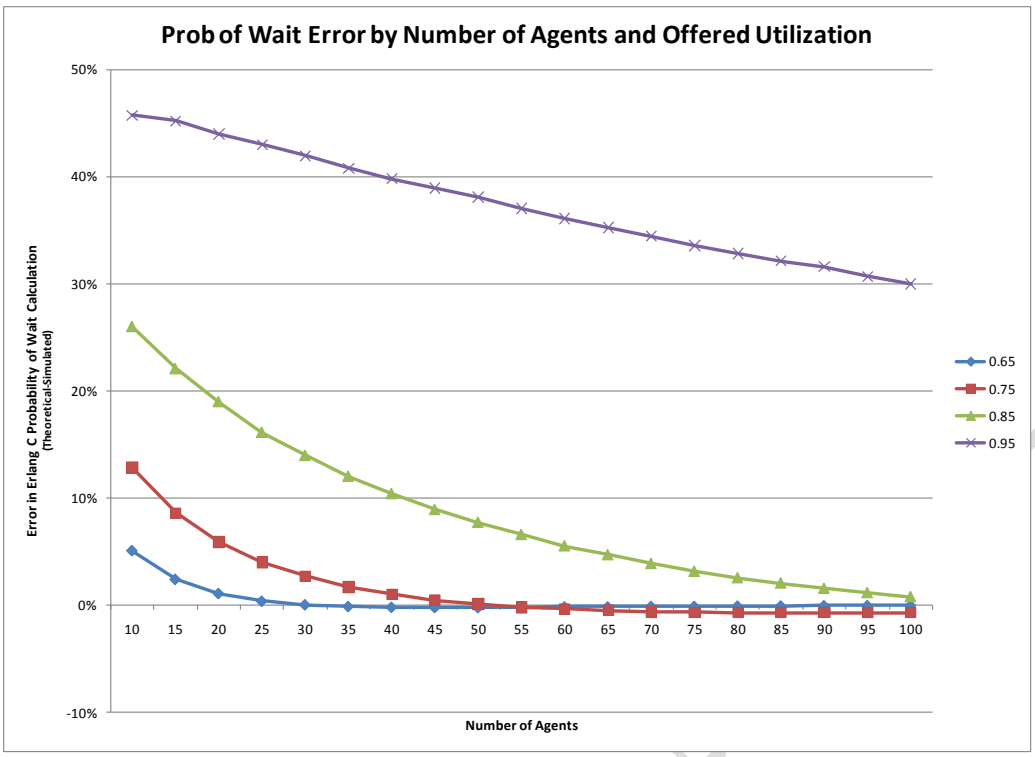

Figure 5 - Erlang C ProbWait Errors by Call Center Size and Utilization

This graph demonstrates that the Erlang C model tends to provide relatively poor predictions for small call centers. This error tends to decrease as the size of the call center increases. However, the graph also illustrates that for busy centers the error remains high. For a very busy call center, running at 95% offered utilization, the error rate remains at 30%, even with a pool of 100 agents. The errors tend to track with abandonment; abandonment rates increase with utilization and decrease with the agent pool.

The conclusion that abandonment behavior drives the Erlang C error is further illustrated in Figure 6. In this experiment we systematically vary two *Willingness to Wait* parameters. Specifically, we vary the balking probability and the β factor of the patience distribution.

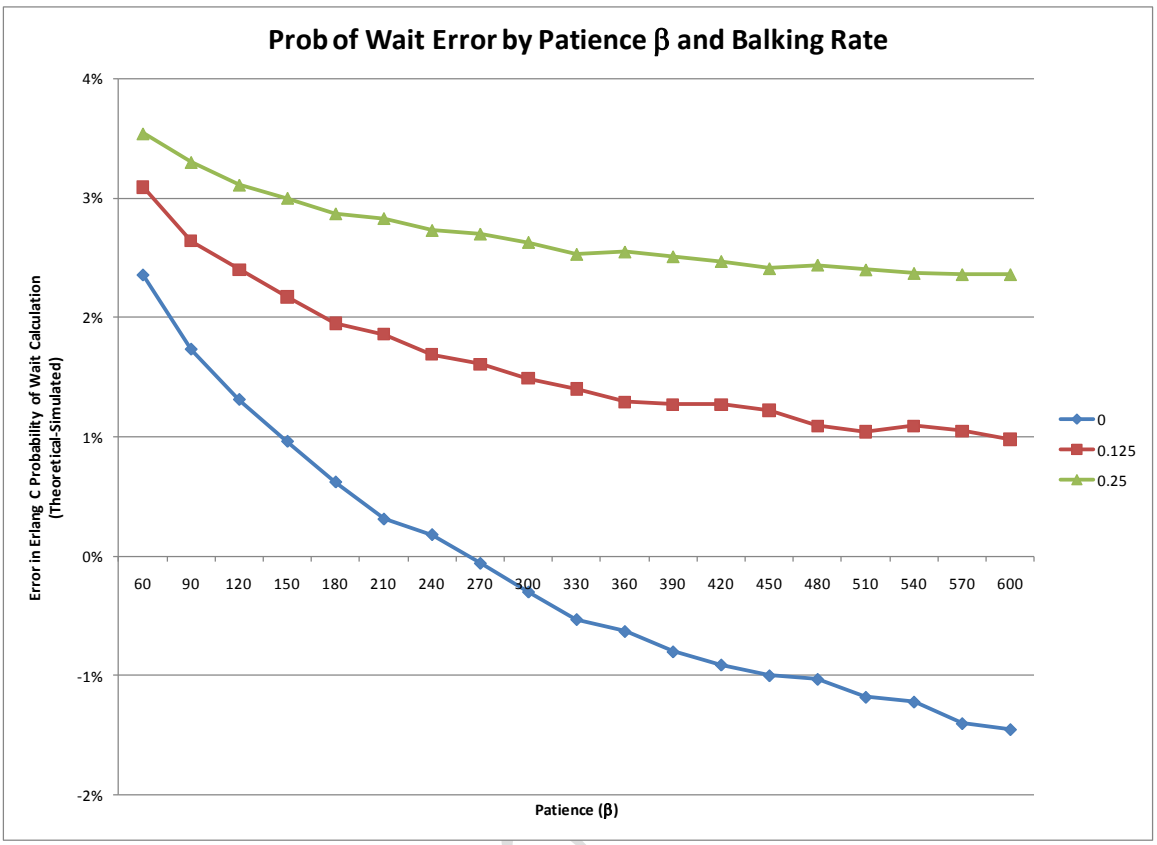

Figure 6 - Erlang C ProbWait Errors by Willingness to Wait

This analysis verifies that the more likely callers are to balk, the higher the error rate. The analysis also shows that when callers are more patient, the error rates decrease. The more likely callers are to abandon, either immediately or soon after being queued, the higher the abandonment rate and the less accurate the Erlang C measures become. The scale of this graph also reinforces the notion that this effect is much smaller than the utilization effect. The vertical axis of Figure 5 spans a range of 60%, while the vertical axis of Figure 6 spans a range of only 6%.

An additional factor of interest is the uncertainty associated with the arrival rate. While its overall effect is not large, about 1.8%, it has effects that are dissimilar to other experimental factors as illustrated in Figure 7. This graph shows the results of an experiment that varies the coefficient of variation of the arrival rate error and the number of agents while holding all other factors at their mid-points.

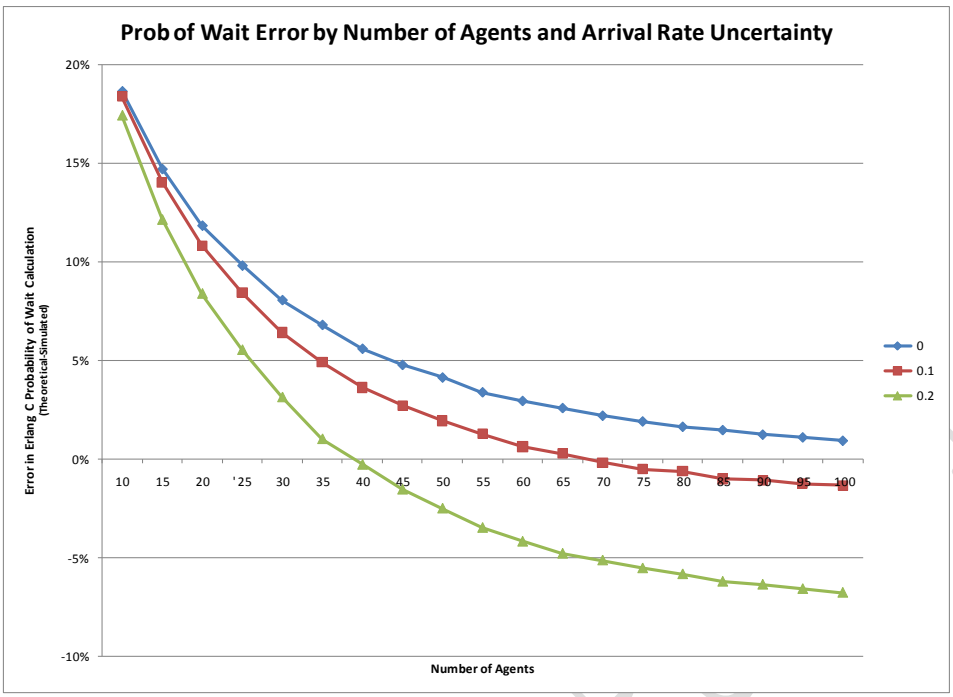

Figure 7 – Erlang C ProbWait Errors by Call Center Size and Forecast Error

This experiment shows that for small call centers arrival rate uncertainty has a small effect, but that effect becomes more pronounced for larger call centers. It is also worth noting that arrival rate uncertainty has an optimistic effect, and for high levels of uncertainty the model exhibits a optimistic bias. Arrival rate uncertainty is a major factor leading to a optimistic estimate from the Erlang C model; of the 21.9% of test points with a optimistic bias, the average arrival rate uncertainty was 15.5%. In short, we conclude that when we assume arrival rates are known with certainty when they are in fact subject to some uncertainty, systems tend to perform on average worse than predicted by the Erlang C model. This is true even though the mean error is zero and the distribution of the error is symmetric.

The Erlang C model is commonly applied to predict queuing system behavior in call center applications. Our analysis shows that when we test the Erlang C model over a range of reasonable conditions, predicted performance measures are subject to large errors. The Erlang C model works reasonably well for large call centers with low to moderate utilization rates, but factors that tend to generate caller abandonment; *i.e.* high utilization, small agent pools, and impatient callers, cause the model error to become quite large. While the model tends to provide a pessimistic estimate, arrival rate uncertainty will either reduce that bias or lead to an optimistic bias. It is clear that great care must be taken before using the Erlang C model to make any calculations that require a high level of precision in a real call center environment.

5. Erlang A Experimental Analysis

5.1. Summary Observations

We utilize the same 1,000 design points for the analysis of the Erlang A model. In summary we find the following

- Erlang A errors are, on average relatively small. The average error for the ProbWait measure from our sample was 1.28%.
- Errors across measures are moderately correlated.
- Errors tend to be optimistic, *i.e.* the system performs worse than predicted.
- Arrival Rate uncertainty has the largest impact on Erlang A prediction error.
- When arrival rates are uncertain, the Erlang A model becomes less accurate as call center size increases.

5.2. Correlation and Magnitude of Errors

The magnitude of errors generated by using the Erlang A model across our sample is on average relatively low. Predicted, simulated, and error magnitudes are summarized in Table 5.

	Erlang A Prediction			Simulation			Error (Prediction - Simulation)			
	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max	% Positive
Prob Wait	0.00%	8.61%	50.16%	0.01%	9.89%	50.10%	-9.09%	-1.28%	3.33%	22.90%
ASA	0.000	2.263	27.290	0.000	3.282	31.499	-15.046	-1.019	0.830	3.70%
TSF	67.13%	95.53%	100.00%	66.21%	94.30%	100.00%	-1.84%	1.23%	13.22%	93.30%
Abandonment Rate	0.00%	2.07%	14.90%	0.00%	2.40%	14.29%	-2.56%	-0.33%	1.34%	29.30%
Utilization	64.23%	78.22%	91.85%	63.16%	77.07%	90.86%	-0.34%	1.15%	4.81%	97.90%

Table 5 - Erlang A Analysis Metrics

Since Erlang A assumes abandonment occurs, it does not predict the excessive wait times forecast by the Erlang C model and the error in the ASA calculation is quite small. In general errors had a small optimistic bias, with the simulated system tending to perform slightly worse than predicted by the model. Errors exhibit a moderately strong correlation as illustrated in Table 6.

	Simulated Abandonment Rate	Prob Wait-Error	ASA-Error	TSF-Error	Abandonment Rate-Error	Utilization-Error
Simulated Abandonment Rate	1.000					
Prob Wait-Error	-.005	1.000				
ASA-Error	-.622	.468	1.000			
TSF-Error	.360	-.790	-.776	1.000		
Abandonment Rate-Error	-.033	.783	.432	-.823	1.000	
Utilization- Error	.250	-.568	-.481	.727	-.802	1.000

Table 6 – Correlation matrix for the Erlang A Model

Correlations between most measurement errors are statistically significant at the .01 level, with the magnitudes of the correlations moderate to high. The errors in the ASA and TSF measures correlate strongly with the realized abandonment rate, though the ProbWait error does not. We will again use ProbWait as a surrogate for the overall error of the model. Figure 8 shows a scatter plot of predicted and simulated ProbWait values.

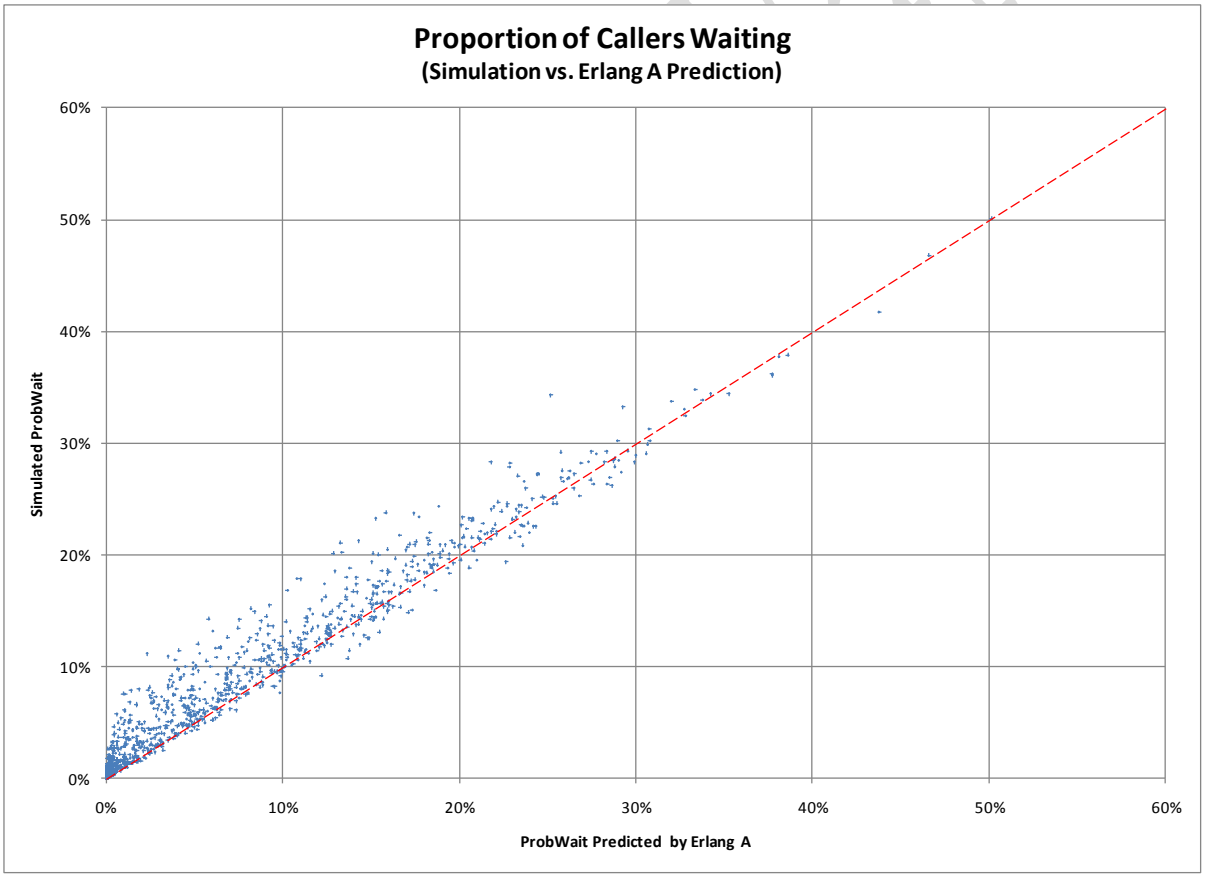

Figure 8 - ProbWait Predicted by Erlang A vs. Simulated

The magnitude of the error when using the Erlang A model is relatively low, but negatively biased; in most cases a larger proportion of calls must wait than was predicted by the model. There

is no clear change in the error as ProbWait increases. Figure 9 shows a histogram of the ProbWait errors.

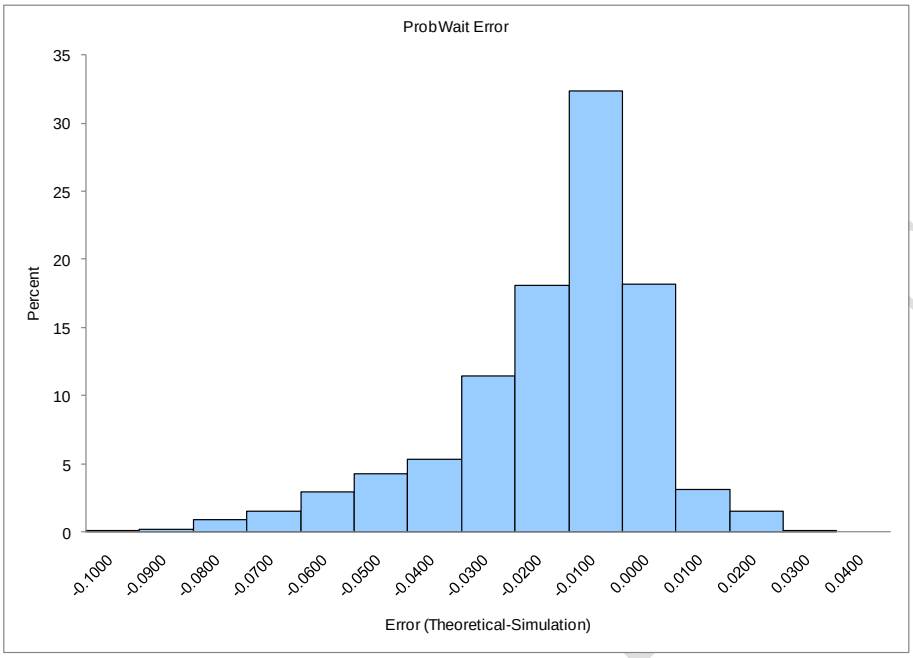

Figure 9 – Histogram of Erlang A ProbWait Errors

The average error in our sample was -1.28%. Errors are skewed and tend to be optimistic, *i.e.* the system performs worse than predicted 77.1% of the time. The ProbWait error has a sample skewness of -1.16. The sample standard deviation of the error is 1.87%

5.3. **Drivers of Erlang A Errors**

To assess how each of the nine experimental factors impacts the error, we perform a regression analysis. The dependent variable is the ProbWait error. For the independent variable we use the nine experimental factors normalized to a [-1,1] scale. This normalization allows us to better assess the relative impact of each factor. The results of the regression analysis are shown in Table 7.

Regression Analysis

R^2	0.624		
Adjusted R^2	0.620	n	1000
R	0.790	k	9
Std. Error	0.012	Dep. Var.	**Prob Wait-Error**

ANOVA table

Source	*SS*	*df*	*MS*	*F*	*p-value*
Regression	0.2179	9	0.0242	182.23	4.12E-203
Residual	0.1316	990	0.0001		
Total	0.3495	999			

Regression output

					confidence interval	
variables	*coefficients*	*std. error*	*t (df=990)*	*p-value*	*95% lower*	*95% upper*
Intercept	-0.0128	0.0004	-35.159	2.13E-176	-0.0135	-0.0121
Num Agents	-0.0068	0.0006	-10.780	1.07E-25	-0.0081	-0.0056
Offered Utilization	-0.0021	0.0006	-3.270	.0011	-0.0033	-0.0008
Talk Time	-0.0001	0.0006	-0.236	.8138	-0.0014	0.0011
Patience	0.0009	0.0006	1.351	.1769	-0.0004	0.0021
AR CV	-0.0231	0.0006	-36.437	4.62E-185	-0.0243	-0.0218
Talk Time CV	0.0007	0.0006	1.067	.2861	-0.0006	0.0019
Patience Shape	-0.0011	0.0006	-1.784	.0747	-0.0024	0.0001
Probability of Balking	0.0061	0.0006	9.603	6.12E-21	0.0049	0.0073
Agent Heterogeneity	0.0060	0.0006	9.519	1.29E-20	0.0048	0.0073

Table 7 - Regression Analysis of ProbWait Errors – Erlang A

The model is statistically significant with a relatively high R^2 value of .624. Given the normalization of the experimental factors, the magnitude of the regression coefficients provides a direct assessment of the impact that a factor has on the measurement error. The factor that most strongly influences the error is the uncertainty in the arrival rate, with an impact more than 3 times all other factors. Arrival rate uncertainty negatively biases the prediction, so the more uncertain the arrival rate, the more optimistic the prediction is likely to be.

This effect is shown more clearly in a separate experiment the results of which are illustrated in Figure 10. In this experiment we vary the number of agents and the degree of arrival rate uncertainty, holding all other factors at their midpoints.

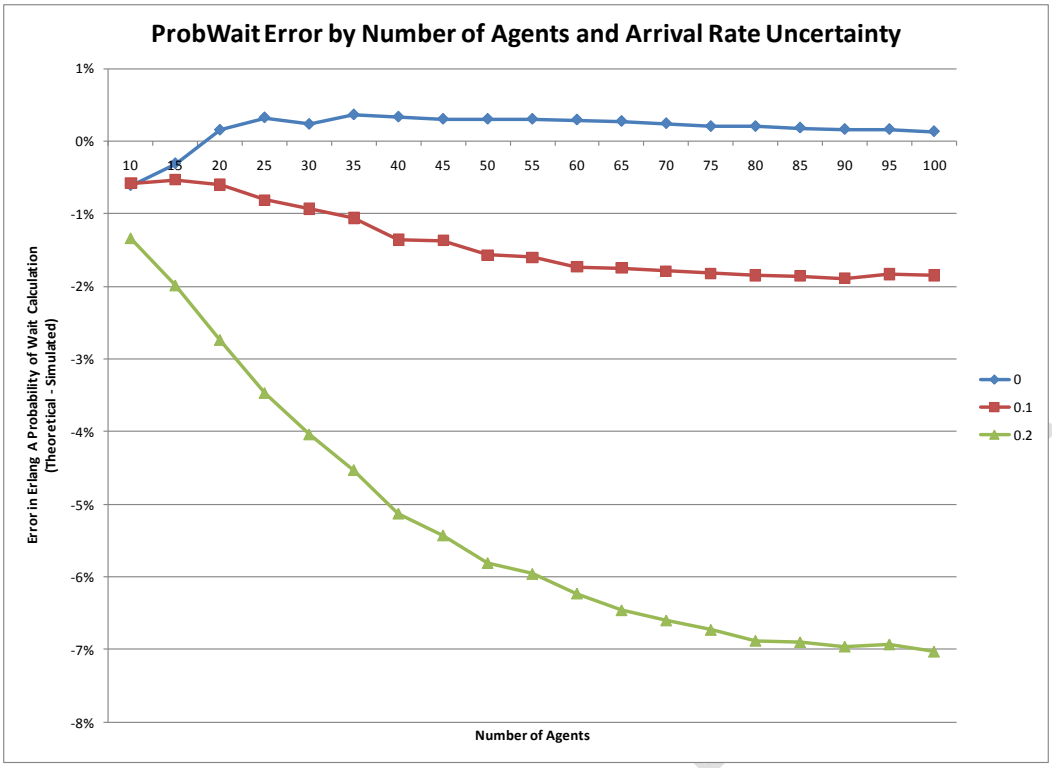

Figure 10-Erlang A ProbWait Errors by Call Center Size and Forecast Error

This graph does show the clear and dominant impact of arrival rate uncertainty. With accurate forecasts the model generates very accurate estimates of performance measures with a slightly pessimistic bias, *i.e.* actual waits smaller than predicted. When arrival rates become uncertain the predictions become optimistic. With high levels of uncertainty the error becomes more sensitive to call center size with the degree of bias increasing with call center size.

Erlang A is a model that many authors advocate is a superior choice for modeling the real world call centers. Our analysis shows that when we test the Erlang A model over a range of reasonable conditions, predicted performance measures are subject to low to moderate errors. However these errors tend to be optimistic with the system performing worse than predicted which could potentially lead to understaffing the call center. We find that the error associated with Erlang A predictions is most strongly impacted by uncertainty in arrival rates.

6. Comparing the Erlang C and Erlang A Models

6.1. Overview

In this section we compare the relative performance of the Erlang C and Erlang A models. We compare prediction errors between the two models for each of the 1,000 points in our experimental design.

Figure 11 shows a scatter plot of error in the ProbWait calculation for each observed point. The dashed lines indicate the areas where the magnitude of the errors in the Erlang C and Erlang A models are the same. Note the different scales; Erlang C error occurs over a range of -8% to 50%, while Erlang A error occurs over a range of -9.1% to 3.3%. The average error of the Erlang C model is 7.96%, while the average error from the Erlang A model is -1.28%. The overall assertion that the Erlang A model is more accurate is in general supported by the data; the average error is smaller and the range of errors is much smaller. However, as the figure shows the model is not universally more accurate.

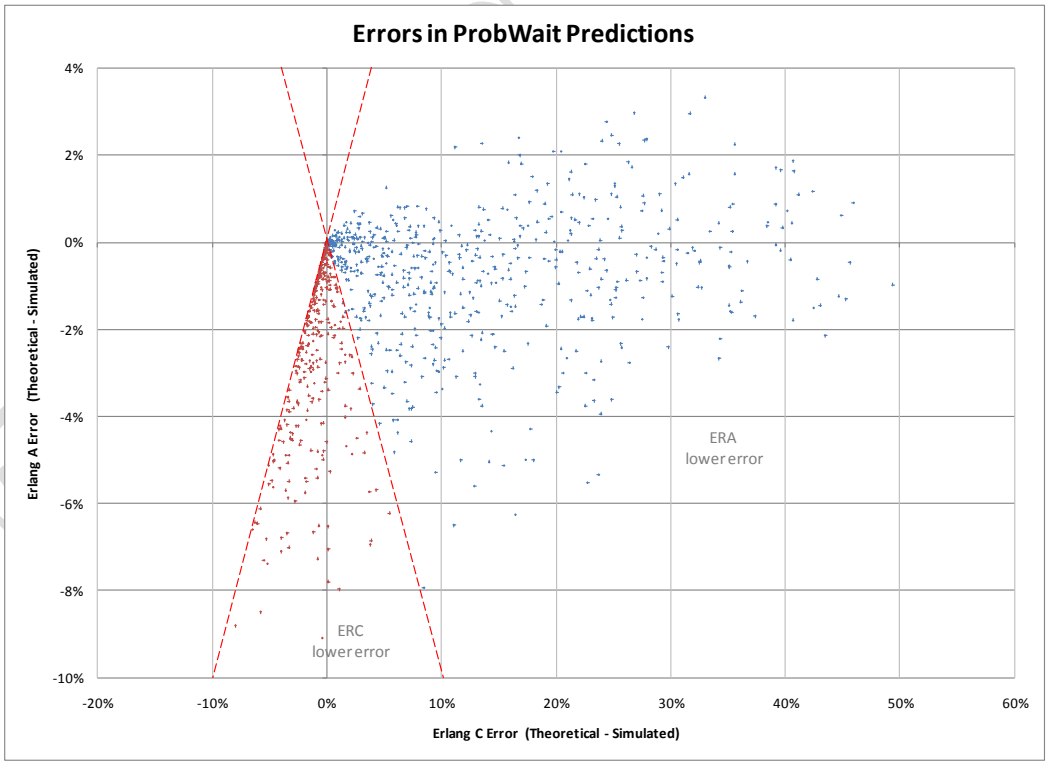

Figure 11 – Comparing ProbWait Errors for Erlang C and A

For points to the right of the two dashed lines the Erlang A model was more accurate. For the points in the triangular region between the two dashed lines the Erlang C model was more accurate. Of the 1,000 points tested, the absolute value of the error from the Erlang A model is less than the absolute value of the Erlang C model error 63.5% of the time, while the Erlang C model was more accurate 36.5% of the time.

6.2. **Optimistic vs. Pessimistic Estimates**

One of the key observations of our analysis has been the pessimistic nature of the Erlang C estimate, and the somewhat optimistic nature of the Erlang A estimate. To further investigate this issue we calculate the prediction percentile for each design point, *i.e.* the proportion of observations where the realized proportion of callers waiting was less than then that predicted by the model. The scatter graph presented in Figure 12 shows the results. This scatter plot shows one point for each of the 1,000 design points. The horizontal axis represents the percentile value in the Erlang C prediction; that is the proportion of the 1000 simulations where the ProbWait measure was less than the theoretical prediction. The vertical axis represents the percentile value of the Erlang A prediction. The diagonal line indicates which model was more conservative; points on the right side of the line indicate that the Erlang C model had a higher percentile value, points to the left the Erlang A.

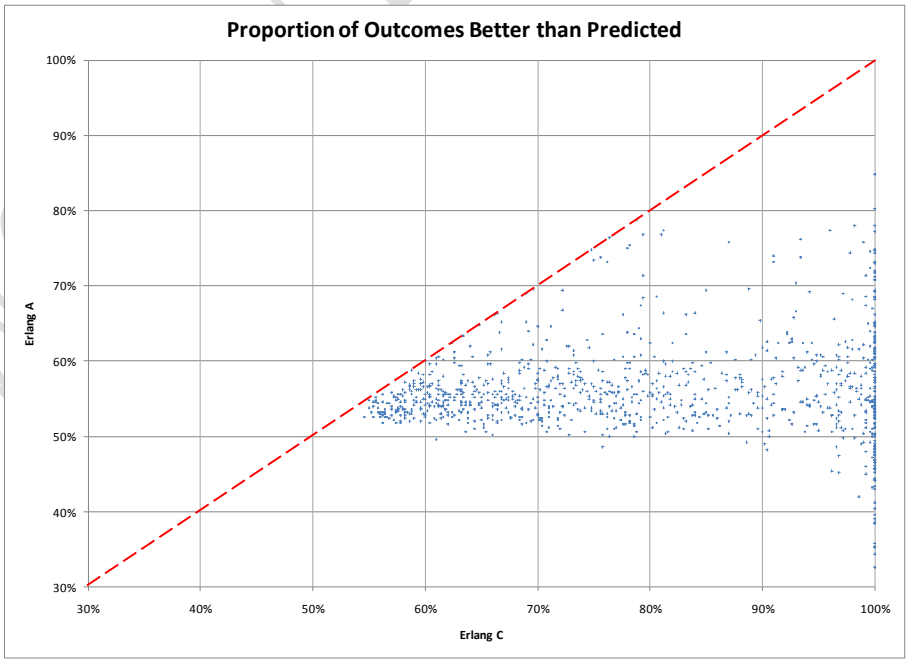

Figure 12 - Comparing ProbWait Percentiles for Erlang C and A

This graph reinforces the notion that the Erlang C model is more conservative. Erlang C has a percentile score higher than or equal to the Erlang A score 100% of the time. The Erlang C model is quite conservative, with a percentile value of 100 in 12.9% of the tested scenarios. The percentile score exceeds 95 in 25.9% of the tested scenarios. In some cases the Erlang A has relatively low percentile scores, as low as 32.6, but overall the Erlang A has a somewhat surprisingly high percentile value, greater than 50% in 93.5% of the test points. This implies that even for points with an optimistic bias, performance will be better than predicted in many cases but far worse in some cases. Due to arrival rate uncertainty, performance measures such as ProbWait are more variable than predicted by a model which assumes known arrival rates. Furthermore the distribution of ProbWait tends to have a relatively high positive skew.

6.3. **Drivers of Relative Performance**

We have seen that the Erlang A model is not universally more accurate than the Erlang C model. An interesting question is under what conditions is the Erlang C model more accurate. To better understand this we segregated the design points into two groups, as illustrated in Figure 11; those where the absolute error of the Erlang C model was smaller, and those where the absolute error of the Erlang A model was smaller.

	Erlang C			Erlang A			Overall	
	Min	Avg	Max	Min	Avg	Max	Average	p value
Num Agents	15	65.0	100	10	49.8	100	55	2.72E-19
Utilization Target	65.02%	74.74%	90.73%	65.11%	82.75%	94.99%	80%	1.47E-48
Talk Time	2.05	10.46	19.92	2.01	11.28	19.99	11	.0172
Patience	61.4	342.5	597.6	60.3	323.5	599.7	330	.0673
AR CV	0.035	0.136	0.200	0.000	0.081	0.200	0.1	5.02E-52
Talk Time CV	0.753	1.008	1.250	0.750	0.996	1.249	1	.2313
Patience Shape	0.753	1.002	1.249	0.750	0.999	1.250	1	.7370
Probability of Balking	0.01%	11.02%	24.84%	0.06%	13.28%	24.99%	12.5%	2.36E-06
Agent Heterogeneity	0.000	0.071	0.150	0.000	0.077	0.150	0.075	.0339
Abandonment	0.01%	0.92%	3.99%	0.00%	3.17%	14.29%	2.40%	3.72E-50

Table 8 - Conditions of Model Accuracy

Table 8 summarizes this data. For each group it presents the minimum, maximum, and average values of each design parameter in each group. It also presents the *p* value generated from a simple hypothesis test that the mean values of each group is the same. Factors such as caller patience, the shape of the patience distribution, and the variability of talk time have little impact on which model is more accurate. Agent Heterogeneity has a moderate impact, with higher levels

of heterogeneity present in cases where Erlang A is more accurate. Similarly, moderately longer talk times are present in the Erlang A group.

The most significant factors are the number of agents in the call center, their utilization, the uncertainty of arrival rates, and the probably of balking. Erlang C tends to be more accurate in call centers with large pools of agents that are moderately utilized, with arrival rates that are less certain. As we have seen, arrival rate uncertainty tends to reduce the error in Erlang C predictions, while increasing the error in Erlang A predictions. Higher levels of balking tend to make the Erlang A model the preferred model. Looking at the realized abandonment rate, it is not surprising that Erlang A is more accurate when abandonment levels are high.

7. Summary and Conclusions

Erlang C is a model commonly applied to the analysis of call centers, and often used as the basis for determining staffing level requirements. Erlang C is a relatively simple model that makes many assumptions that are clearly suspect in the context of a call center. Many authors now advocate the use of the more realistic and more complex and more difficult to calculate Erlang A model. We have conducted a comprehensive simulation analysis that shows that the Erlang C model is in fact subject to significant prediction error and that the Erlang A model is on average much more accurate under steady state conditions.

However, our analysis also finds that while the Erlang C is conservative, making pessimistic predictions most of the time, the Erlang A model often makes overly optimistic predictions. The optimistic bias of the Erlang A model is driven in large part by arrival rate uncertainty, a condition that somewhat paradoxically reduces the error of the Erlang C model. The results of our study suggest that care must be taken when using the Erlang A model to make staffing decisions; particularly in cases where arrival rates are subject to significant uncertainty and service level requirements are strict.

The analysis in this study is restricted to cases where the call center possesses the capacity to handle all calls presented; a requirement for the Erlang C model to have a defined output. In some call center environments staffing costs are significantly higher than the implied cost of customer delay and the number of agents is limited so that essentially all customers must wait before receiving service and agent utilization is close to 100%. This environment is sometimes referred to as the *efficiency-driven regime* (Gans, Koole et al. 2003). In this environment the offered uti-

lization is in excess of 100%, so the Erlang C model becomes unstable and cannot be used to predict performance. The Erlang A model on the other hand allows for abandonment and can be used to predict performance in this regime. A separate study is warranted to determine the accuracy of the Erlang A model in this environment. Our analysis is also focused on long term average performance under steady-state conditions. We do not examine the impact of shifting arrival rates that occurs in real scenarios. In practice many call centers divide the day into a series of short intervals and assume that steady state is achieved in each period. A further study could examine the impact of this assumption on model accuracy.

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**(6): 665-668.

Armony, M. and A. R. Ward (2008). Fair Dynamic Routing in Large-Scale Heterogeneous-Server Systems, Stern School of Business, NYU.

Avramidis, A. N., M. Gendreau, P. L'Ecuyer and O. Pisacane (2007). Simulation-Based Optimization of Agent Scheduling in Multiskill Call Centers. 2007 Industrial Simulation Conference.

Bassamboo, A., J. M. Harrison and A. Zeevi (2005). "Design and Control of a Large Call Center: Asymptotic Analysis of an LP-based Method." Operations Research **54**(3): 419-435.

Borst, S., A. Mandelbaum and M. I. Reiman (2004). "Dimensioning Large Call Centers." Operations Research **52**(1): 17-35.

Brown, L., N. Gans, A. Mandelbaum, A. Sakov, S. Haipeng, S. Zeltyn and L. Zhao (2005). "Statistical Analysis of a Telephone Call Center: A Queueing-Science Perspective." Journal of the American Statistical Association **100**(469): 36-50.

Chen, B. P. K. and S. G. Henderson (2001). "Two Issues in Setting Call Centre Staffing Levels." Annals of Operations Research(108): 175-192.

Gans, N., G. Koole and A. Mandelbaum (2003). "Telephone call centers: Tutorial, review, and research prospects." Manufacturing & Service Operations Management **5**(2): 79-141.

Gans, N. and Y.-P. Zhou (2007). "Call-Routing Schemes for Call-Center Outsourcing." Manufacturing & Service Operations Management **9**(1): 33-51.

Garnett, O. and A. Mandelbaum (2002). 4CallCenters Software (http://iew3.technion.ac.il/serveng/4CallCenters/Downloads.htm).

Garnett, O., A. Mandelbaum and M. I. Reiman (2002). "Designing a Call Center with impatient customers." Manufacturing & Service Operations Management **4**(3): 208-227.

Green, L. V., P. Kolesar and J. Soares (2003). "An Improved Heuristic for Staffing Telephone Call Centers with Limited Operating Hours." Production and Operations Management **12**(1): 46-61.

Green, L. V., P. J. Kolesar and J. Soares (2001). "Improving the SIPP Approach for Staffing Service Systems That Have Cyclic Demands." Operations Research **49**(4): 549-564.

Halfin, S. and W. Whitt (1981). "Heavy-Traffic Limits for Queues with Many Exponential Servers." Operations Research **29**(3): 567-588.

Harrison, J. M. and A. Zeevi (2005). "A Method for Staffing Large Call Centers Based on Stochastic Fluid Models." Manufacturing & Service Operations Management **7**(1): 20-36.

Jennings, O. B., A. Mandelbaum, W. A. Massey and W. Whitt (1996). "Server Staffing to Meet Time-Varying Demand." Management Science **42**(10): 1383-1394.

L'Ecuyer, P. (1999). "Good Parameters and Implementations for Combined Multiple Recursive Random Number Generators." Operations Research **47**(1): 159-164.

Law, A. M. (2007). Simulation modeling and analysis. Boston, McGraw-Hill.

Mandelbaum, A. and S. Zeltyn (2004). Service Engineering in Action: The Palm/Erlang-A Queue, with Applications to Call Centers Draft, December 2004.

Mandelbaum, A. and S. Zeltyn (2009). The M/M/n+G Queue: Summary of Performance Measures, Technical Note, Technion, Israel Institute of Technology.

Mandelbaum A., Sakov A. and Z. S. (2001). Empirical Analysis of a Call Center, Technion - Israel Institute of Technology**:** 73.

Palm, C. (1957). "Research on telephone traffic carried by full availability groups." Tele **1**: 107.

Robbins, T. R. (2007). Managing Service Capacity Under Uncertainty - Unpublished PhD Dissertation (http://personal.ecu.edu/robbinst/), Pennsylvania State University**:** 240.

Robbins, T. R. and T. P. Harrison (2010). "Call Center Scheduling with Uncertain Arrivals and Global Service Level Agreements." European Journal of Operational Research **Forthcoming**: 29.

Robbins, T. R., D. J. Medeiros and P. Dum (2006). Evaluating Arrival Rate Uncertainty in Call Centers. Submitted to 2006 Winter Simulation Conference, Monterey, CA.

Robbins, T. R., D. J. Medeiros and T. P. Harrison (2010). Does the Erlang C model fit in real call centers? 2010 Winter Simulation Conference, Austin, TX.

Santner, T. J., B. J. Williams and W. Notz (2003). The design and analysis of computer experiments. New York, Springer.

Steckley, S. G., S. G. Henderson and V. Mehrotra (2009). "Forecast Errors in Service Systems." Probability in the Engineering and Informational Sciences(23): 305-332.

Steckley, S. G., W. B. Henderson and V. Mehrotra (2004). Service System Planning in the Presence of a Random Arrival Rate, Cornell University.

Wallace, R. B. and W. Whitt (2005). "A Staffing Algorithm for Call Centers with Skill-Based Routing." Manufacturing & Service Operations Management **7**(4): 276-294.

Whitt, W. (2005). "Engineering Solution of a Basic Call-Center Model." Management Science **51**(2): 221-235.

Whitt, W. (2006a). "Fluid Models for Multiserver Queues with Abandonments." Operations Research **54**(1): 37-54.

Whitt, W. (2006b). "Sensitivity of Performance in the Erlang A Model to Changes in the Model Parameters." Operations Research **54**(2): 247-260.

Whitt, W. (2006c). "Staffing a Call Center with Uncertain Arrival Rate and Absenteeism." Production and Operations Management.

Supplementary Material

Average Speed to Answer Analysis

In the body of this paper we focus on the analysis of the probability of waiting as the best representative performance metric. In this supplement we performed a similar analysis on the Average Speed to Answer (ASA) metric. Unlike the other metrics ASA is measured in seconds, not as a percentage. It is therefore unbounded and potentially subject to a larger error.

Erlang C

Erlang C assumes that all callers have an infinite patience and will not abandon the queue. Under these assumptions the model may predict very long wait times when traffic is heavy. However, with finite patience callers will abandon the queue, dramatically impacting the average wait time. In Figure 13 we show a scatter plot of predicted and simulated ASA for each of the one thousand test points. The dashed line represents points where predicted and simulated values are equal.

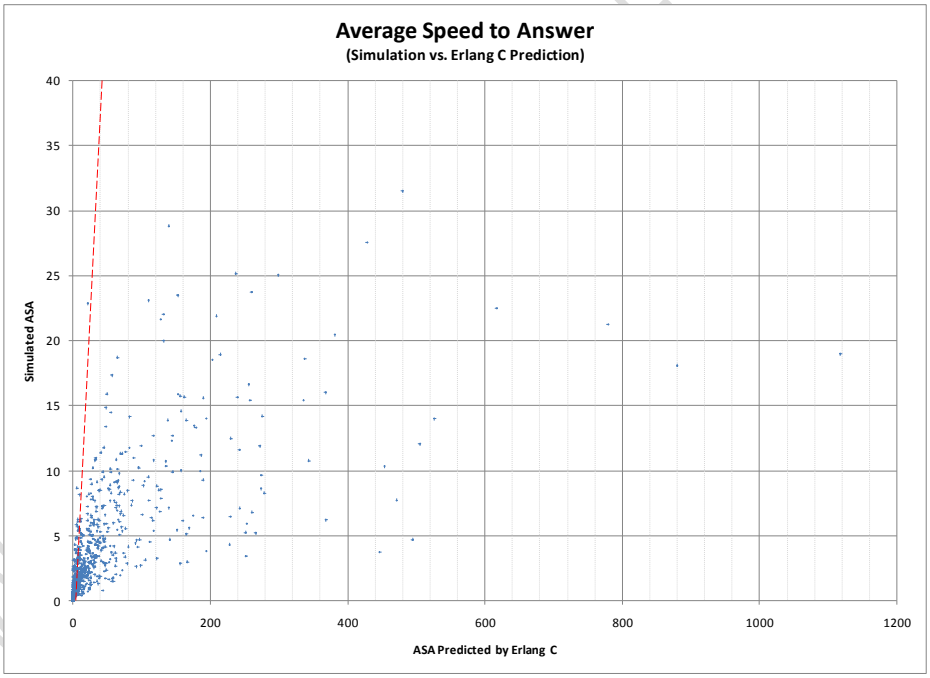

Figure 13 – ASA predicted by Erlang C vs. Simulated

The graph reveals the propensity of the model to predict ASA values that are larger than those realized, often times substantially so. In an extreme case, Erlang C predicts a wait time in excess of 1100 seconds, while the simulation results in a wait time less than 20 seconds. This pessimis-

tic bias is further illustrated in Figure 14 which shows a histogram of the error in the ASA prediction.

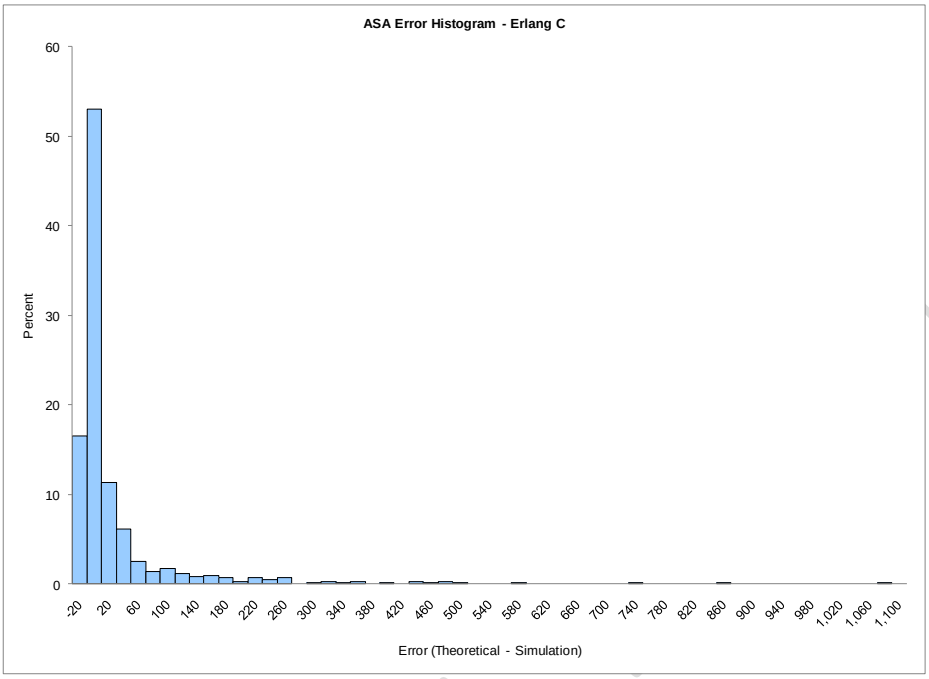

Figure 14 – Histogram of Erlang C ASA Errors

The error has an average of 33 seconds with a strong positive skew; the sample skew statistic is 5.8. In 84% of the test points, the realized wait time was less than the predicted time. The error in the ASA prediction has a strong (.766) positive correlation with the actual abandonment rate. This is further illustrated in Figure 15.

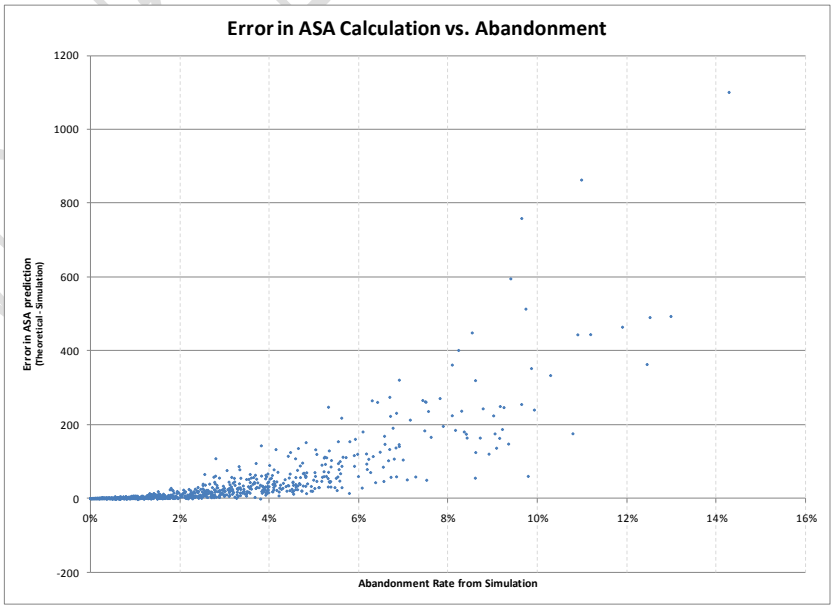

Figure 15 – Scatter Plot of Erlang C Errors and Abandonment Rates

Figure 15 shows that for low abandonment rates the ASA prediction is reasonably accurate, but as abandonment increases the error increases substantially and non-linearly. Clearly in situations with significant abandonment, the Erlang C should be used cautiously when predicting average speed to answer.

To better understand what factors influence error and ASA predictions we can perform a regression analysis using each of our nine experimental factors scaled to [-1,1] as the independent variables. The results of this regression are shown in Table 9.

Regression Analysis

R^2	0.385		
Adjusted R^2	0.380	n	1000
R	0.621	k	9
Std. Error	66.113	Dep. Var.	**ASA Error**

ANOVA table

Source	*SS*	*df*	*MS*	*F*	*p-value*
Regression	2,710,150.1738	9	301,127.7971	68.89	2.41E-98
Residual	4,327,175.3472	990	4,370.8842		
Total	7,037,325.5210	999			

Regression output					*confidence interval*	
variables	*coefficients*	*std. error*	*t (df=990)*	*p-value*	*95% lower*	*95% upper*
Intercept	33.4332	2.0907	15.992	2.24E-51	29.3306	37.5359
Num Agents	-56.0647	3.6311	-15.440	2.37E-48	-63.1902	-48.9391
Utilization Target	63.6746	3.6349	17.518	4.69E-60	56.5417	70.8076
Talk Time	33.9767	3.6250	9.373	4.64E-20	26.8632	41.0903
Patience	-2.9147	3.6263	-0.804	.4217	-10.0308	4.2014
AR CV	-2.1419	3.6347	-0.589	.5558	-9.2745	4.9907
Talk Time CV	-2.9513	3.6407	-0.811	.4178	-10.0957	4.1930
Patience Shape	4.1639	3.6362	1.145	.2524	-2.9716	11.2994
Probability of Balking	7.1053	3.6439	1.950	.0515	-0.0454	14.2559
Agent Heterogeneity	1.0989	3.6284	0.303	.7621	-6.0213	8.2191

Table 9 – Regression Analysis of ASA Errors – Erlang C

In contrast to the ProbWait regression, this model shows only three predictors to be statistically significant at the .05 level; number of agents, utilization target, and talk time. The model performs best in situations with large numbers of agents, low utilization targets, and short talk times, the conditions under which abandonment tends to be low. We further illustrate this in Figure 16. This graph shows the error and ASA prediction as the number of agents increases from 10 to 100, for different levels of offered utilization.

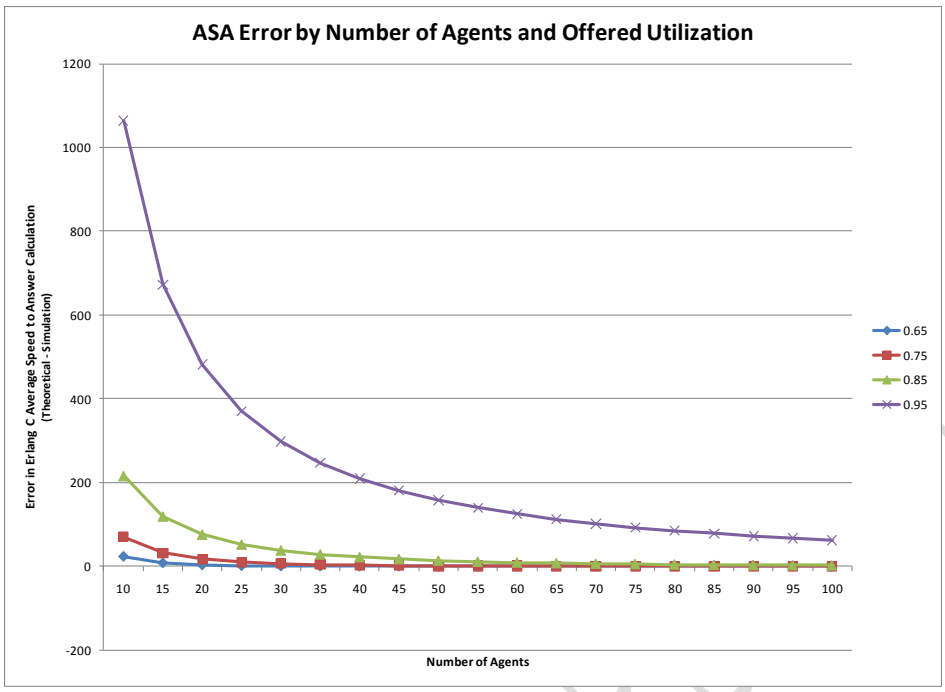

Figure 16 – Erlang C ASA Errors by Call Center Size and Utilization

The model is least accurate for small number of agents, and high utilizations.

Erlang A

In this section we examine the performance of the Erlang A model in predicting average speed to answer.

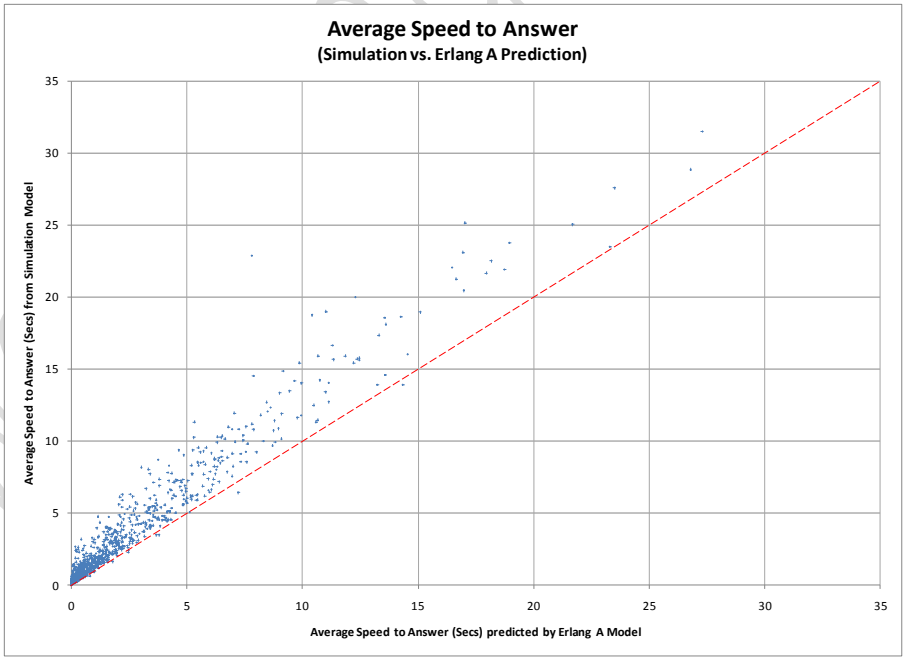

Figure 17 presents a scatter plot of predicted and realized ASA values.

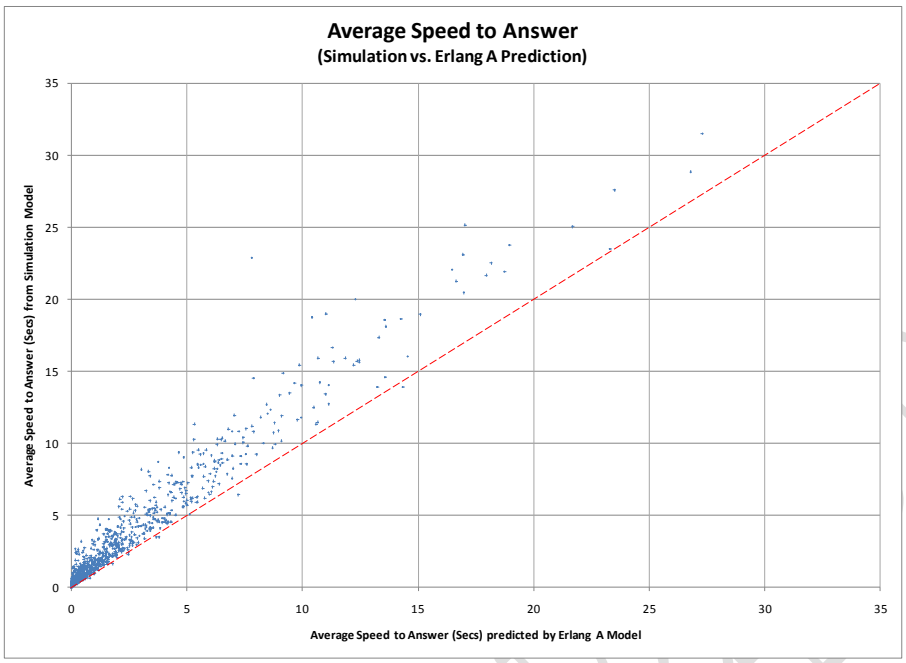

Figure 17 – ASA Predicted by Erlang A vs. Simulated

Erlang A is shown to be reasonably accurate even when the average speed to answer is relatively large. However, this graph also shows that the prediction is optimistically biased; realized ASA tends to be longer than that predicted by the Erlang A model. This is further illustrated in the histogram in Figure 18.

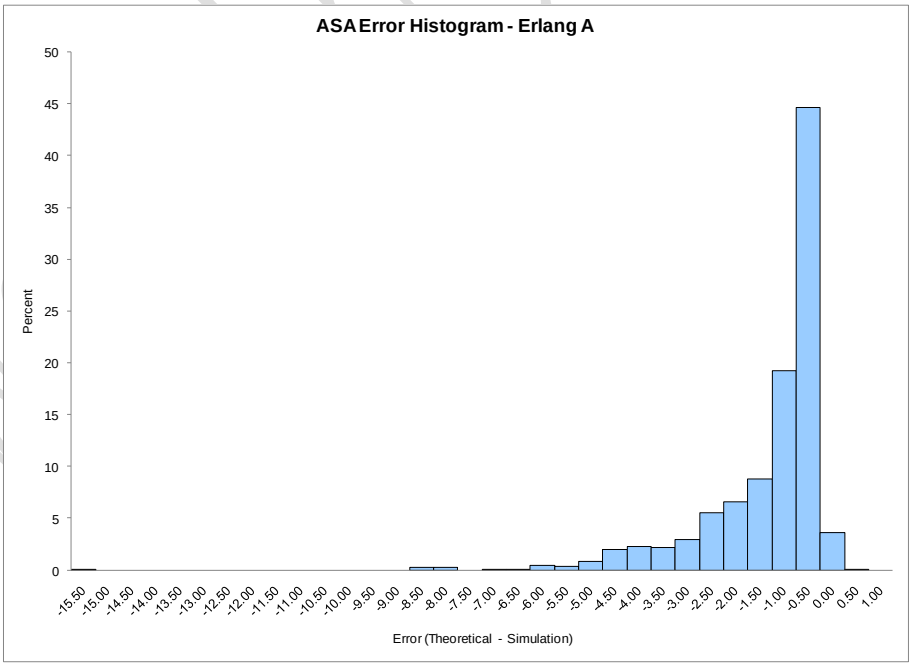

Figure 18 – Histogram of Erlang A ASA Errors

In general the prediction is quite accurate, with an average error of only -1.02 seconds. The data is negatively skewed, with the sample skew statistic of .83. 96.3% of the observations have a negative error, a condition where the actual wait time is longer than predicted.

A regression analysis to evaluate which external factors influence the ASA error in the Erlang A model is presented in Table 10. This regression shows that with the exception of the variability of talk time, all factors are statistically significant at the .05 level. The most influential measures are the utilization target, arrival rate uncertainty, and the number of agents in a call center.

Regression Analysis

R^2	0.591		
Adjusted R^2	0.587	n	1000
R	0.769	k	9
Std. Error	0.856	Dep. Var.	**ASA-Error**

ANOVA table

Source	*SS*	*df*	*MS*	*F*	*p-value*
Regression	1,047.6397	9	116.4044	158.83	2.98E-185
Residual	725.5649	990	0.7329		
Total	1,773.2046	999			

Regression output					*confidence interval*	
variables	*coefficients*	*std. error*	*t (df=990)*	*p-value*	*95% lower*	*95% upper*
Intercept	-1.0199	0.0271	-37.675	2.04E-193	-1.0731	-0.9668
Num Agents	0.6351	0.0470	13.507	2.78E-38	0.5428	0.7274
Utilization Target	-1.0458	0.0471	-22.219	4.62E-89	-1.1382	-0.9535
Talk Time	-0.4633	0.0469	-9.871	5.53E-22	-0.5555	-0.3712
Patience	-0.4612	0.0470	-9.822	8.59E-22	-0.5534	-0.3691
AR CV	-1.0199	0.0471	-21.669	1.58E-85	-1.1122	-0.9275
Talk Time CV	-0.0052	0.0471	-0.111	.9115	-0.0978	0.0873
Patience Shape	-0.1782	0.0471	-3.784	.0002	-0.2706	-0.0858
Probability of Balking	0.3444	0.0472	7.299	5.94E-13	0.2518	0.4370
Agent Heterogeneity	0.2459	0.0470	5.234	2.03E-07	0.1537	0.3381

Table 10 Regression Analysis of ASA Errors – Erlang A

The impact of call center size and arrival rate uncertainty is further illustrated in Figure 19.

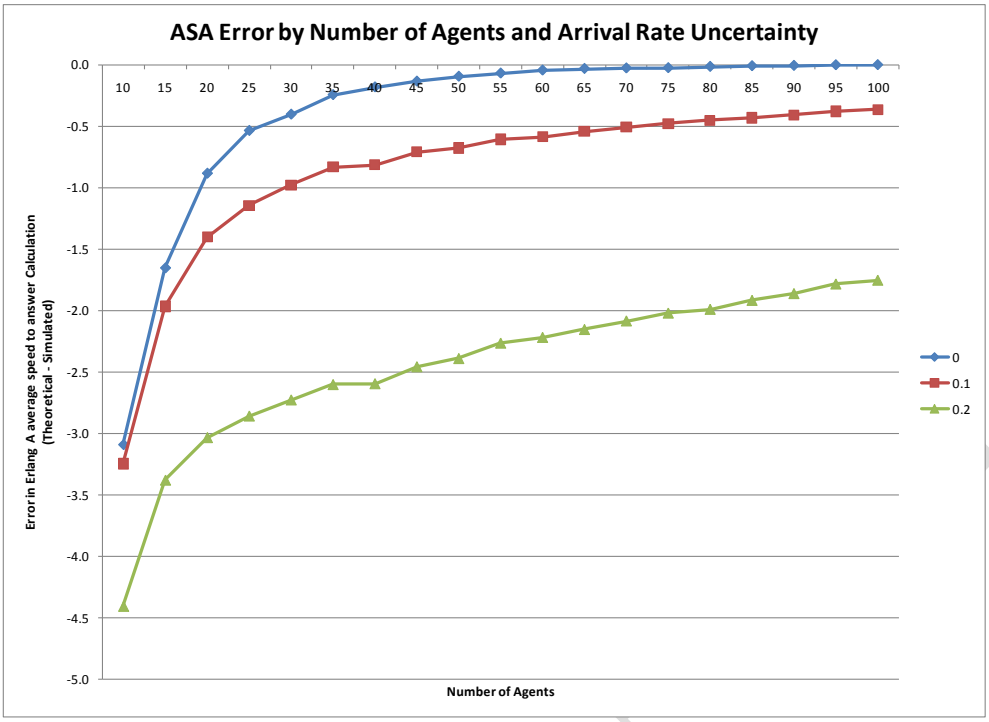

Figure 19 - Erlang A ASA Errors by Call Center Size and Forecast Error

The graph shows that the error decreases substantially as the number of call center agents increases. It also shows that arrival rate uncertainty biases the error in a negative direction.

Comparing the Erlang C and Erlang A Models

The analysis above demonstrates that the Erlang C model is subject to more substantial error in predicting average speed answer then the Erlang A model. This is further illustrated in Figure 20 where we plot the Erlang C and Erlang A error for each of the 1,000 points in our experiment. Each point on the graph represents a design point in our experiment. The horizontal axis represents the difference between the simulated value and the Erlang C prediction, whereas the vertical axis represents the difference between the simulation and the Erlang A prediction.

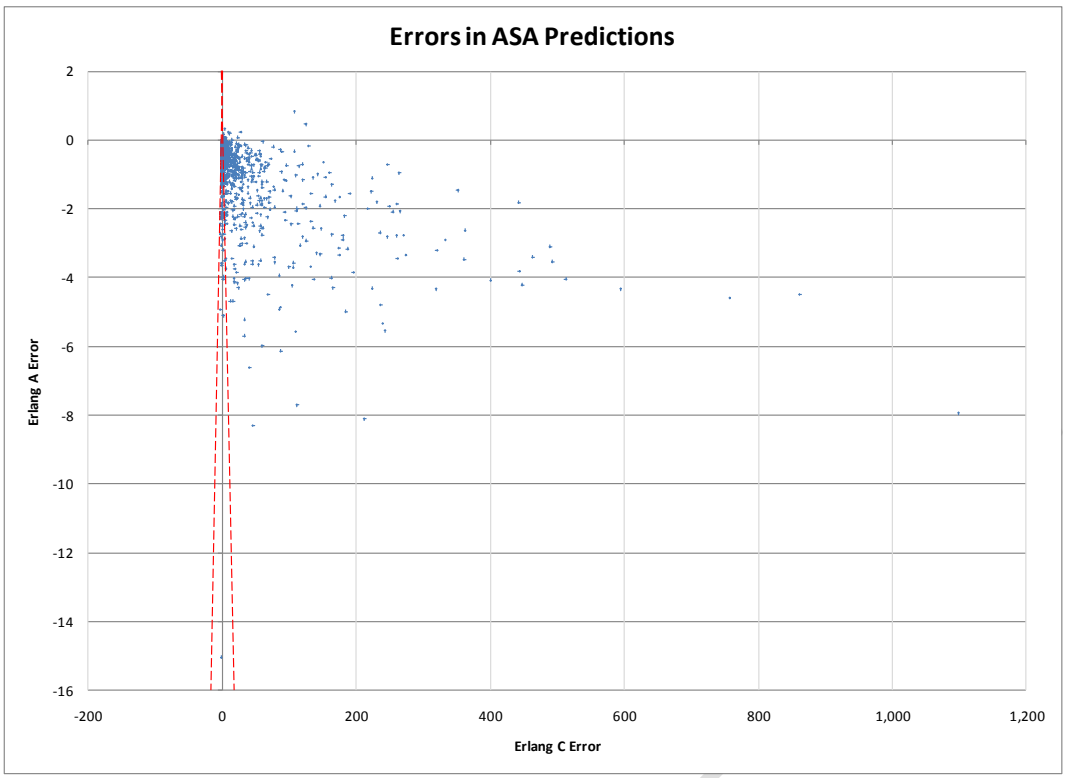

Figure 20 - Comparing ASA Errors for Erlang C and A

Again we see that Erlang C errors occur over a wide range of values; from -2.8 seconds to nearly 1100 seconds. The error in the Erlang A calculation on the other hand occurs over a much narrower range of -15 seconds to .83 seconds. The Erlang A model has a lower absolute error in 78% of the test cases. In those cases where Erlang C error is smaller, the difference tends not to be substantial.