Parametric Stochastic Programming Models for Call-Center Workforce Scheduling

Noah Gans
OPIM Department
The Wharton School
U. Pennsylvania

Haipeng Shen
Department of Statistics and OR
UNC Chapel Hill

Yong-Pin Zhou
ISOM Department
The Foster School
U. Washington

Nikolay Korolev Alan McCord Herbert Ristock
Genesys Telecommunications Laboratories, Inc.

April 2012

Abstract

We develop and test an integrated forecasting and stochastic programming approach to workforce management in call centers. We first demonstrate that parametric forecasts can be used to drive stochastic programs whose results are stable with relatively small numbers of scenarios. We then extend our approach to include forecast updates and two-stage stochastic programs with recourse and provide a general modeling framework for which recent, related models are special cases. In our formulations, the inclusion of multiple arrival-rate scenarios allows the call centers to meet long-run average QoS targets, while the use of recourse actions help them to lower long-run average costs. Experiments with two large sets of call-center data highlight the complementary nature of these elements.

1 Introduction

Inbound telephone call centers handle service requests that originate from customers calling in, and they use a hierarchical staffing and scheduling system (Gans et al. 2003, Akşin et al. 2007).). The process begins with forecasts of the arrival rate of calls over a planning horizon, which may range from a day to several weeks. The forecasts then drive queueing models that determine how staffing levels affect system congestion over short, 15-minute to 1-hour, time intervals within the horizon. The queueing formulae determine staffing levels over the short time intervals, and, in turn, constraints to be met as the call center develops staff schedules. A rostering process then matches employees with required schedules. In this way, the forecasted arrival process of calls to the center drives employee schedules.

Traditionally, call centers assume that arrival-rate forecasts are correct. They use point forecasts of arrival-rates to determine staffing levels and, in turn, deterministic staffing-level requirements drive scheduling decisions. But arrival-rate forecasts are often not perfect, and when realized arrival rates do not match the forecast, system performance naturally deviates from managers' expectations. Higher-than-expected arrival rates lead to understaffing, which drives up waiting times and abandonment rates, while unexpectedly low arrival rates result in overstaffing and the "overservice" of customers.

Work within the statistics and operations management literatures has begun to address the problem of how call centers – and other high volume service businesses – can better manage the capacity-demand mismatch that results from arrival-rate uncertainty. Earlier papers have explored the impact of arrival-rate uncertainty (Grassman 1988, Chen and Henderson 2001, Jongbloed and Koole 2001, Ross 2001), and more recent work has explicitly modeled arrival-rate uncertainty and its effects (Robbins et al. 2006, Steckley et al. 2009). Statistical models in Whitt (1999), Avramidis et al. (2004), Brown et al. (2005), Weinberg et al. (2007), Shen and Huang (2008), Aldor-Noiman et al. (2009), Ibrahim and L'Ecuyer (2011), Taylor (2012), and others have sought to better characterize the distribution of arrival rates, by time of day, as they evolve.

Operations management papers account for uncertainty when making staffing and scheduling decisions. Maman (2009) extends many-server heavy-traffic limits to explicitly account for an arrival-rate distribution. Papers by Harrison and Zeevi (2005), Bassamboo et al. (2005), Bassamboo et al. (2006), Bassamboo and Zeevi (2009), Bertsimas and Doan (2010), and Gurvich et al. (2010) have used stochastic programming (Birge and Louveaux 1997) to account for arrival-rate uncertainty when making short-run staffing and call-routing decisions. More recent papers, such as Robbins et al. (2010), Robbins and Harrison (2010), and Liao et al. (2012) extend the stochastic programming framework to employee scheduling, and Mehrotra et al. (2010) uses mid-day recourse actions to adjust pre-scheduled staffing levels in reaction to realized deviations from arrival-rate forecasts.

While each of these streams of research has made important progress in addressing elements of the problems caused by arrival-rate uncertainty, none addresses the whole problem. Statistical papers dedicated to forecasting have used traditional measures of fit for realized arrival counts to assess forecast quality. They have not, however, considered the downstream cost and quality of service (QoS) implications of arrival-rate forecast errors. While operations management papers have looked carefully at the cost and QoS implications of stochastic scheduling methods, they have not used sophisticated statistical forecasting methods to better capture the nature of arrival-rate uncertainty. In turn, their measures of cost and QoS improvements may not accurately reflect the gains that can be made when better forecasting and scheduling methods are used in concert.

In this paper we integrate these statistical and operations-management approaches, marrying the use of more sophisticated forecasting methods with stochastic programming formulations of call-center staffing and scheduling problems. Our work is data driven, and we use two large sets of call-center data to evaluate the elements of our approach. One set of empirical results shows that parametric forecasting methods can be used to more efficiently solve stochastic scheduling problems that have traditionally been solved using sampling-based scenarios. Another set of empirical tests highlights the complementary roles that the use of *a priori* scenario-based stochastic programming and the use of recourse actions can play in addressing arrival-rate uncertainty. More specifically, we make the following contributions.

In §2 we develop low-dimensional, parametric arrival-rate forecasts and use Gaussian quadrature to transform their continuous distributions into discrete scenarios (Miller and Rice 1983). We apply this scenario generation scheme to demonstrate that only a small number of scenarios is needed to capture the bulk of arrival-rate uncertainty in simple, one-stage stochastic workforce-scheduling problems.

Because scenarios are based on arrival rates, while updates are based on the realization of arrival counts, rather than rates, the development of forecasts suitable for two-stage stochastic programs with recourse is not trivial. While it is not clear to us how one would develop an effective sampling approach to the generation of forecasts suitable for recourse programs, in §3 we are able to use our parametric approach to this end.

- We develop a Bayesian procedure that uses realized arrival counts in the early stage of the planning horizon to update the forecast distribution for arrival rates during the later stage.

- We then extend this updating approach in a manner suitable for two-stage stochastic programs with recourse, using a multi-stage, tree-based approach to generate arrival-rate scenarios for the later periods of a two-stage forecast.

- We use these forecasts to evaluate the relative effectiveness of a family of six workforce management programs, which vary in their use of scenarios – one versus many – as well as in the sophistication of their updating schemes: no update, simple *ex post* updates, and more sophisticated *a priori* stochastic programs with recourse.

Section 4 then tests these six schemes using two sets of call center data. In both sets of tests we find that the use of multiple scenarios helps to stabilize system performance and leads to average abandonment rates that better match *a priori* targets, while recourse actions help to lower costs. We also find that the value of the more sophisticated *a priori* two-stage recourse programs varies across the two examples: in one the two schemes provide nearly identical cost savings, and in the other the more sophisticated *a priori* two-stage recourse program provides some additional savings.

More broadly, our work provides a general framework for formulating and solving workforce management and scheduling decisions, within which previous, related work represents special cases. Our results show that effective solutions to workforce scheduling problems will take the form of stochastic programs with recourse. This approach explicitly accounts for arrival-rate uncertainty, as well as the ability to change these decisions in response to updates in arrival-rate forecasts.

2 Parametric Forecasts for Stochastic Programming

In this section, we develop efficient methods for formulating and solving simple, single-stage stochastic programs for workforce scheduling in call centers. While the basic formulation of this stochastic program is nearly identical to that in Robbins and Harrison (2010), our approach to scenario generation uses Gaussian quadrature to discretize continuous forecast distributions and differs from the sampling schemes that are common to that paper, Bertsimas and Doan (2010), and others. This section's empirical tests show that, in fact, our Gaussian quadrature approach can greatly reduce the numbers of scenarios needed to stably solve these stochastic programs.

We also show that, for measures of QoS that are convex in the number of agents staffed during a given interval, we can further reduce the nonlinear constraints that model system performance across multiple scenarios into a single set of piecewise linear constraints. The transformation does not rely on the method by which scenarios have been generated – through discretization or via sampling – and allows us to efficiently perform tests that compare the two methods on problems with large numbers of scenarios.

The ability to work with few scenarios, as well as to collapse large numbers of scenarios, when warranted, becomes particularly important when solving the two-stage stochastic programs with recourse that we analyze in Section 3.

2.1 Parametric Forecast

Our historical data comprise a $D \times I$ matrix of arrival counts, $\mathbf{N} = (N_{di})$, where $d \in \mathcal{D} = \{1, \ldots, D\}$ indexes days and $i \in \mathcal{I} = \{1, \ldots, I\}$ indexes the 30-minute intervals within each day. We refer to the dth row of $\mathbf{N}$, denoted as $\mathbf{N}_d = (N_{d1}, \ldots, N_{dI})$, as the *intraday call volume profile* of the dth day.

We model N_{di} as a Poisson random variable with an uncertain arrival-rate Λ_{di}. Denote $\mathbf{\Lambda}_d = (\Lambda_{d1}, \ldots, \Lambda_{dI})$ as the *dth intraday arrival-rate profile*. We are interested in forecasting $\mathbf{\Lambda}_{D+h}$, the intraday arrival rate profile for a future day $D + h$, where h is a positive integer.

Because the underlying rate profiles are uncertain and unobservable, our forecasting model uses the count profiles $\{\mathbf{N}_1, \ldots, \mathbf{N}_D\}$ to form an I-dimensional time series. The dimensionality of the vector time series is typically high; for example, there are $I = 26$ half-hour periods in a 13-hour working day. A good forecasting model must reduce the dimensionality.

We develop a forecasting model that combines dimension reduction, a key idea of the data-driven approach of Shen and Huang (2008), with parametric modeling, which allows us to efficiently discretize the forecast distribution. In terms of forecast accuracy, our model performance is comparable to that of Weinberg et al. (2007) and Shen and Huang (2008), both of which have been shown to work well for arrival-rate forecasting.

Specifically, the following square-root transformation stabilizes the variance of the count data and approximately normalizes the observations. Together, these effects improve forecast accuracy and make the transformed counts amenable for standard statistical modeling. Its proof can be found in Brown et al. (2010).

Proposition 1 (Brown et al. 2010) *Suppose a random variable N has a Poisson distribution with rate Λ. As $\Lambda \to \infty$, $y \equiv \sqrt{N + 1/4}$ has a Gaussian distribution with mean $\sqrt{\Lambda}$ and variance $1/4$.*

Thus, instead of directly modeling the call volumes N_{di}, we build our forecasting model using the square root of the call volumes. Such a square-root transformation has been used in the call center forecasting literature (Brown et al. 2005, Weinberg et al. 2007, Shen and Huang 2008).

We then consider the following forecasting model for the square-root-transformed counts $y_{di} \equiv \sqrt{N_{di} + 1/4}$, $d \in \mathcal{D}$, $i \in \mathcal{I}$:

$$\begin{aligned} y_{di} &= \sqrt{\Lambda_{di}} + \epsilon_{di}, \quad \epsilon_{di} \sim \mathrm{N}(0, \sigma^2), \\ \Theta_{di} &\equiv \sqrt{\Lambda_{di}} = \omega_d \vartheta_{l_d,i}, \\ \omega_d - \alpha_{l_d} &= \beta(\omega_{d-1} - \alpha_{l_{d-1}}) + \eta_d, \quad \eta_d \sim \mathrm{N}(0, \phi^2), \\ \vartheta_{l_d,i} &\geq 0, \quad \sum_{i=1}^{I} \vartheta_{l_d,i} = 1, \end{aligned} \tag{1}$$

where l_d is day-of-the-week of day d, ω_d is the daily total arrival rate (on the square-root scale), α_{l_d} is the adjustment for the day of the week, and $\vartheta_{l_d,i}$ is the intraday rate proportion for the ith time interval that also depends on the corresponding day of the week.

Our forecasting model (1) can be understood as follows. First, on the square-root-transformed scale, the daily total rate (w_d) follows an order-one autoregressive time series model, adjusting for the day of the week (α_{l_d}). Second, each weekday has its own intraday arrival proportion profile, $(\vartheta_{l_d,1}, \ldots, \vartheta_{l_d,I})$. Finally, the transformed arrival rate Θ_{di} is assumed to follow a multiplicative model. By Proposition 1, we know these square-root quantities have approximate Gaussian distributions; hence the errors in the above model are assumed to be Gaussian.

The model easily captures the two-way (intraday and interday) time dependence that is common to call centers and other large-scale service systems. A simpler model is considered by Whitt (1999) for the untransformed arrival rates that assumes all days share a common intraday arrival profile.

Our model can be estimated using nonlinear least squares. Below we provide a set of simple estimates, which are close to the true least-squares estimates (Brown et al. 2005),

$$\hat{\alpha}_{l_d} = \frac{\sum_{d':l_{d'}=l_d} \sum_i \omega_{di}}{\#\{d' : l_{d'} = l_d\}}, \qquad \hat{\vartheta}_{l_d,i} = \frac{\sum_{d':l_{d'}=l_d} \omega_{d'i}}{\sum_{d':l_{d'}=l_d} \sum_i \omega_{d'i}}. \tag{2}$$

The autoregressive coefficient $\hat{\beta}$ can then be estimated using linear regression.

Once the model (1) is estimated, we make use of the time series model for ω_d to obtain a forecast distribution for ω_{D+h}, which is Gaussian. Observing that $\mathbf{\Lambda}_{D+h,i} = \left(\omega_{D+h}\vartheta_{l_{D+h},i}\right)^2$, the forecast distribution for the future arrival rate $\Lambda_{D+h,i}$ follows easily.

2.2 Gaussian Quadrature for Scenario Generation

To simplify notation, we now drop the day subscript in ω_d, $\vartheta_{l_d,i}$ and $\Lambda_{d,i}$ and consider an arbitrary day. The uncertain arrival rate during its ith time period satisfies

$$\Lambda_i = (\omega\vartheta_i)^2 \tag{3}$$

where ω has a (forecast) distribution that is Gaussian with mean ζ and variance ϕ^2.

To account for the uncertainty of Λ_i, recent papers have used stochastic programs with scenarios generated via random sampling from the forecast distribution (Bertsimas and Doan 2010, Gurvich et al. 2010, Robbins et al. 2010, Robbins and Harrison 2010). It is well known, however, that a large number of scenarios may be needed for the sampling approach to be effective (Shapiro and Philpott 2007, §2.2).

In this paper, we exploit the fact that ω has a known Gaussian distribution to generate scenarios through discretization. In particular, we use Gaussian quadrature to derive a discrete approximation, ω^*, for ω, where $\omega^* = \omega_k$ with probability p_k, $k \in \mathcal{K} = \{1, \ldots, K\}$, and ω^* and ω have the identical first $2K-1$ moments. Given these ω_ks and p_ks, the relation (3) naturally leads to the discrete approximation of Λ_i as

$$\Lambda_i^* \;=\; \lambda_{ik} \;\equiv\; \omega_k^2\vartheta_i^2 \qquad \text{with probability } p_k,$$

for $k \in \mathcal{K}$. Details of the discretization procedure can be found in Miller and Rice (1983).

Remark 1 For the degenerate one-scenario case (i.e. $K = 1$), special care is needed to make sure that Λ_i has the correct mean. Because Gaussian quadrature only matches the first moment of ω when $K = 1$, it will not guarantee that the discretized mean matches that for Λ_i, since $\Lambda_i = (\omega\theta_i)^2$. In this case, we set $\omega_1 = \sqrt{\mu^2 + \phi^2}$ instead of the default value μ. □

Observe that (1) and (3) imply that, for all time periods within a given day, arrival-rate uncertainty is essentially only driven by the one-dimensional random scaling factor ω. Thus, we are able to use Gaussian quadrature efficiently for a one-dimensional distribution and avoid the usual problem of curse-of-dimensionality.

2.3 Stochastic Programming Formulation

In the retail banking setting from which we have collected data, there is no explicit customer waiting time or abandonment cost. Rather, these types of call centers often minimize staffing costs, subject to explicit QoS

constraints. In this paper we impose a 3% limit on the expected fraction of incoming calls that abandon before service, a QoS limit that can be regularly attained in larger, well-run call centers.

Let T be the length of the planning horizon, which may range from one day to several weeks. In the context of the math programs we solve in this paper, we fix T to be one day (i.e. $T = I$), but in practice the horizon can easily exceed I. As before, $\mathcal{I} = \{1, \ldots, I\}$ is a set of equally-divided subintervals within a day.

Let $\mathcal{J} = \{1, \ldots, J\}$ be the set of all the feasible work schedules, each of which dictates which intervals within the planning horizon an agent answers calls. For schedule $j \in \mathcal{J}$,

$$a_{ij} = \begin{cases} 1, & \text{if schedule } j \in \mathcal{J} \text{ has an agent answer calls during interval } i \in \mathcal{I}, \text{ and} \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in \mathcal{I}$ and $j \in \mathcal{J}$. We let c_j be the cost of assigning an agent to schedule j. Costs include hourly wages and overtime pay, if the schedule requires it. Depending on the setting, the costs may or may not include a prorated share of benefits payments. The principal decision variables are the numbers of agents to assign to the various schedules: $\{x_j \mid j \in \mathcal{J}\}$.

Recall that λ_{ik} is the forecast arrival rate during interval $i \in \mathcal{I}$ under scenario $k \in \mathcal{K}$ and that p_k is the probability that scenario $k \in \mathcal{K}$ occurs. Observe that, when $K = 1$, then $p_k = 1$, and the stochastic program with one scenario collapses to become a traditional, deterministic workforce-scheduling IP. For *i.i.d.* scenarios based on sampling, each scenario, k, occurs with equal probability $p_k = 1/k$ and is determined by sampling ω and then using the relation (3) to determine the λ_{ik}s.

In any given interval, i, and scenario, k, our stochastic program determines the quality of service experienced by arriving customers using a stationary measure of performance from standard queueing models. In particular, in this paper we track customer abandonment as the measure of QoS and use results from Mandelbaum and Zeltyn (2007) that characterize the stationary behavior of the M/M/n+M (Erlang-A) model. Given Poisson arrivals of constant rate λ, *i.i.d.* exponentially distributed service times with mean $1/\mu$, *i.i.d.* exponentially distributed times until customer abandonment (sometimes called patience) with mean $1/\theta$, and n servers, the paper provides explicit expressions for the calculation of the fraction of arriving customers that abandon before being served, $f(\lambda, \mu, \theta, n)$. In our context, the arrival rate during period i under scenario k is λ_{ik}, and the number of agents on hand during interval i is $n_i = \sum_{j \in \mathcal{J}} a_{ij} x_j$. Together with μ and θ they determine the expected number of customers abandoning during period i under scenario k, $\lambda_{ik} f(\lambda_{ik}, \mu, \theta, n_i)$.

Remark 2 Our use of the above results implicitly makes two common assumptions. The first is that the arrival rate is constant over interval i. While common, this assumption is not necessarily innocuous. Nevertheless, effective measures can be taken to account for time-inhomogeneity within intervals. For a characterization of time-inhomogeneity, see Brown et al. (2005), and for effective responses see Feldman et al. (2008) as well as the review article Green et al. (2007). The second is that, even if the arrival rate were constant, the use of

stationary performance measures assumes that the event rate during interval i under scenario k is large enough that transient effects, due to initial conditions at the start of the interval, are not significant. In medium to large call centers this is typically the case. □

Let α^* be an upper bound on the expected abandonment rate over the planning horizon. Then we wish to solve the following nonlinear stochastic integer program, which minimizes total staffing cost, subject to constraints on expected abandonments.

$$\begin{array}{lcll} \min \sum_{j\in\mathcal{J}} c_j x_j & & & \\ \text{subject to} & & & \\ \sum_{k\in\mathcal{K}} p_k \lambda_{ik} f(\lambda_{ik}, \mu, \theta, \sum_{j\in\mathcal{J}} a_{ij} x_j) & = & \alpha_i \quad & i \in \mathcal{I} \\ \sum_{i\in\mathcal{I}} \alpha_i & \leq & \alpha^* \bar{\lambda} & \\ x_j & \in & \mathbb{Z}^+ \quad & j \in \mathcal{J}, \end{array} \tag{4}$$

where $\bar{\lambda} = \sum_{k\in\mathcal{K}} p_k \sum_{i\in\mathcal{I}} \lambda_{ik}$ is the expected number of arrivals over the planning horizon, and $\mathbb{Z}^+$ is the set of non-negative integers. The first set of nonlinear constraints defines the expected number of abandoning calls for each interval, i. The second constraint defines the upper bound on the expected global abandonment rate.

We emphasize that the upper bound, $\alpha^*\bar{\lambda}$, holds only in expectation, across the entire arrival-rate distribution. If we consider every potential abandonment to have the same (unknown) implicit cost, the Lagrangian relaxation of (4) would minimize expected total cost of staffing and abandonment. Because the cost of abandonment is unknown, however, call centers instead place direct constraints on expected QoS.

Remark 3 While for a specific arrival-rate realization the QoS constraint may be violated, given a correctly forecast arrival-rate distribution and many *i.i.d.* days, the long-run average abandonment rate should be less than or equal to α^*. Of course, in our AR(1) setting, arrival-rate distributions need not be *i.i.d.*, and forecast distributions need not be correct. Nevertheless we conjecture that, in our case, the long-run average abandonment rate will still fall at or below α^* and are working to formally prove it is so (Gans et al. 2012). □

It would be straightforward to define other measures of QoS as well. Common variants of (4) define QoS in terms of delay in queue, place limits on tail probabilities, and require that QoS targets be met over sub-intervals within the planning horizon. For an interesting discussion of the time-horizon over which QoS targets are set, see §7 in Gurvich et al. (2010).

The fact that $f(\lambda_{ik}, \mu, \theta, n_i)$ may be nonlinear in n_i makes the stochastic program (4) potentially difficult to solve. Nevertheless, Armony et al. (2007) show that, given $\mu \geq \theta$, $f(\lambda_{ik}, \mu, \theta, n_i)$ is nonincreasing in n_i, with decreasing differences (discretely convex). This is typically the case. For example, see Zohar et al. (2002) and Brown et al. (2005). In Section 2.4 we will see that the desired relationship also holds in our data.

Given $\mu \geq \theta$, we can use a common transformation to replace the nonlinear constraints with a larger set of linear constraints that provides a lower bound on $\sum_{k\in\mathcal{K}} p_k \lambda_{ik} f(\lambda_{ik}, \mu, \theta, n_i)$. For each $i \in \mathcal{I}$ and $n > 0$ we define slopes, m_{in}, and intercepts, b_{in},

$$\begin{aligned} m_{in} &= \textstyle\sum_{k\in\mathcal{K}} p_k \left[\lambda_{ik}(f(\lambda_{ik}, \mu, \theta, n) - f(\lambda_{ik}, \mu, \theta, n-1))\right] \\ b_{in} &= \textstyle\sum_{k\in\mathcal{K}} p_k \lambda_{ik} f(\lambda_{ik}, \mu, \theta, n) - n \cdot m_{in}, \end{aligned} \tag{5}$$

where $m_{i0} = -\mu$ and $b_{i0} = \sum_{k\in\mathcal{K}} p_k \lambda_{ik}$. Then we replace each of the I constraints that define the α_i's in (4) with a set of $\mathcal{N}_i = \{0, \ldots, N_i\}$ linear constraints,

$$(\textstyle\sum_{j\in\mathcal{J}} a_{ij} x_j) m_{in} + b_{in} \leq \alpha_i, \qquad n \in \mathcal{N}_i,$$

where N_i is large enough that the abandonment rate is essentially zero: $\sum_{k\in\mathcal{K}} p_k f(\lambda_{ik}, \mu, \theta, N_i) \approx 0$.

Remark 4 Here, N_i refers to the number of linear constraints used to define a lower bound on $f(\lambda_{ik}, \mu, \theta, n)$. In §2.1, however, N_{di} was used to define arrival counts. We will continue to use N in both cases, and its meaning should be clear from the context: stochastic program or arrival-rate forecast. □

The revised linear, integer stochastic program becomes

$$\begin{aligned} &\min \textstyle\sum_{j\in\mathcal{J}} c_j x_j \\ &\text{subject to} \\ &(\textstyle\sum_{j\in\mathcal{J}} a_{ij} x_j) m_{in} + b_{in} \leq \alpha_i \quad i \in \mathcal{I},\ n \in \mathcal{N}_i \\ &\textstyle\sum_{i\in\mathcal{I}} \alpha_i \leq \alpha^* \bar{\lambda} \\ &x_j \in \mathbb{Z}^+ \quad j \in \mathcal{J}. \end{aligned} \tag{6}$$

Essentially, we are replacing the abandonment rate function in (4) by the maximum of the linear functions defined by all the m_{in}s and the b_{in}s, which is piece-wise linear and convex, a standard practice in math programming. In our case, this substitution is exact because the variables only take on integer values.

Remark 5 Robbins and Harrison (2010) use a variant of (6) in which the QoS constraint becomes $\sum_{i\in\mathcal{I}} \alpha_i \leq \alpha^* \sum_{k\in\mathcal{K}} p_k \sum_{i\in\mathcal{I}} \lambda_{ik} + \delta$, and the objective function is augmented to include a penalty for abandonments above the nominal target of α^*: $\min \sum_{j\in\mathcal{J}} c_j x_j + p\delta$. Rather than taking expected values, the paper explicitly represents many scenarios and uses a computationally intensive L-shaped decomposition method (Birge and Louveaux 1997, §5.1) to solve the math program. □

Thus, rather than solving the stochastic program (4), which is nonlinear and may have a large number of scenarios, we use the definitions (5) to develop a deterministic, piecewise linear, "certainty equivalent" program (6). From (5) we see that the computational effort needed for the transformation is linear in the number of scenarios, K.

2.4 Setup for Empirical Tests of Quadrature and Sampling-Based Scenarios

With the machinery developed in §2.1–2.3 at our disposal, we are now in the position to perform large-scale tests of the efficacy of our quadrature-driven scenarios. We run these tests using a dataset from a European retail bank's call center operations.

Our dataset consists of historical arrival counts, abandonment counts, and service-time averages, as well as the rules and parameters the bank's workforce management system uses to schedule agents to meet demand. The arrival and service-time data cover 176 weekdays in 2007. The call center is open 13 hours each weekday, from 8 a.m. to 9 p.m., and it reports arrival counts and average service times for 30-minute intervals. Hence for each day we have 26 intervals of data, and we set the planning horizon to be $T = 26$ intervals, or one day.

Agents work either 7 or 9-hour days, without overtime, and with specific rules for meal breaks: a lunch break lasts half an hour and must occur between 11 a.m. and 2 p.m.; a late lunch break also lasts half an hour and must occur between 4:30 p.m. and 6 p.m. An agent qualifies for either the lunch break or the late lunch break if her/his shift contains a half-hour period within that time window. If her/his shift contains a half-hour period within each time window, then s/he qualifies for both breaks. Enumeration shows that there are $J = 262$ feasible schedules.

The bank did not share payroll information with us, and the fact that it used no overtime in constructing regular schedules motivates us to use a normalized cost of 1 for each half-hour an agent works. Therefore $c_j = \sum_{i \in \mathcal{I}} a_{ij}$.

We apply the approach described in §2.1 to forecast future arrival rates. Each forecast uses the previous 100 days of arrival counts to forecast the next day's rates. Therefore we have 76 days (days 101 to 176) of out-of-sample forecasts that we use to develop scenarios and run stochastic programs.

We use these parametric forecasts as the basis for quadrature and sampling-based scenario-generation schemes. For the former, we follow the procedure detailed in Miller and Rice (1983). For the latter, we sample ω from its normal distribution, once for each scenario, and apply (3).

Other data used in the stochastic program include the following. The service times in our dataset average 121 seconds, so we set the service rate to be $\mu = 1800/121 \approx 14.6$ services per agent per 30-minute interval. To estimate the abandonment rate, we divide the total number of abandoned calls in our dataset by the total waiting time in queue for all served calls. This provides an upper bound on the abandonment rate of $\theta = 3.93$ calls per 30-minute interval, or equivalently, a lower bound on average caller patience of $1800/3.93 \approx 458$ seconds.[1] This implies that, on average, customers are willing to wait (at least) about 3.7 service times before

[1]The Kaplan-Meier estimator for exponentially-distributed patience divides the total number of abandonments by the sum of the delays of all calls, including those that are served and those that abandon the queue before being served. See Zohar et al. (2002). Our dataset includes records of average delay in queue only for served calls, however, and we include the total delay of served calls in the

abandoning the queue. The data clearly satisfy the requirement that $\mu \geq \theta$.

All of the math programs use an expected daily abandonment rate target of $\alpha^* = 3\%$. In most cases the QoS constraint in the optimal solution is tight or nearly tight.

2.5 Empirical Comparison of Quadrature and Sampling-Based Scenario Generation

We compare the results of stochastic programs that use scenarios generated using Gaussian quadrature to those that use scenarios based on sampling of the forecast distribution. Our results suggest that the quadrature-based approach is more efficient and stable, in that very few scenarios are needed to obtain performance that is reliable with respect to our two performance measures: abandonment rate and cost. Our comparison has three parts.

2.5.1 Distributions of Total Cost for One Day

We begin with a detailed look at the two methods' performance for a single out-of-sample forecast day. Using data from days 1 through 100 we generate a normally-distributed forecast for ω for day 101 and then use the forecast to create 909 stochastic programs that we solve. We generate scenarios for 9 of the stochastic programs – with 1, 4, 9, 16, 25, 49, 100, 225, and 400 scenarios – using Gaussian quadrature. For each of 9 stochastic programs, we also create 100 analogous *i.i.d.* sets of scenarios for day 101, each set using an appropriate number of *i.i.d.* samples from the arrival-rate forecast. Thus we generate, run, and evaluate 101 instances of 9 stochastic programs for a total of 909. For each of these 909 instances, we record the stochastic program's objective function value, the total cost of staffing day 101.

Figure 1 summarizes the total costs (the objective function value) of the 909 solutions. Results are grouped, by number of scenarios, along the horizonal axis, from 1 up to 400. For each number of scenarios, the vertical axis reports total cost. Each box of the box-and-whisker plot shows the 25th percentile, 50th, and 75th percentiles of the cost of the 100 sampling-based instances of the problem for that number of scenarios. The whiskers are 1.5 times the interquartile range (75th percentile - 25th percentile) and are used as thresholds for determining outliers. The circles above and below the whiskers are the outliers. The dashed lines running across the whiskers display the 2.5% and 97.5% percentiles of the sampling-based results, and the solid line running across the boxes plots shows the cost of the analogous quadrature-based program.

Several features of Figure 1 are of note. As expected, as the number of scenarios increases, the distribution of results for sampling-based instances becomes less dispersed. Similarly, it is not surprising that, as the number of scenarios increases, the average cost of sampling-based programs and the cost of quadrature-based programs, are (generally) non-increasing. Formulations with fewer scenarios display a well-known, systematic downward bias that results from solving a convex minimization problem with stochastic data (Shapiro 2000, §5.2). Of

denominator of our calculation. If we were to include the waiting time of abandoning calls, it would therefore lower the estimate of θ.

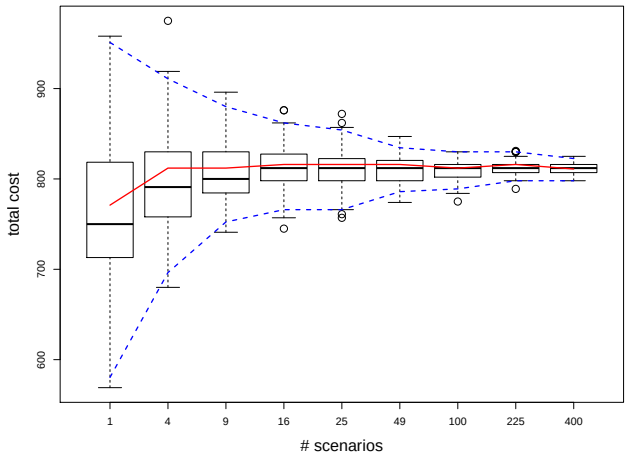

Figure 1: Day 101 Confidence Intervals, by Number of Scenarios

more interest to us is the fact that the cost of quadrature-based solutions remains nearly constant from 4 to 400 scenarios, and for instances with 16 scenarios or more, the cost is nearly identical to the median costs of the sampling-based programs.

2.5.2 First Differences in Total Cost for Each of 76 Days

Our second set of tests compare the results of all 76 days. In these tests, we formulate and solve 1,368 math programs: 9 stochastic programs – with 1, 4, 9, 16, 25, 49, 100, 225, and 400 scenarios – for each of the two scenario-generation scheme and each of 76 days. Here, we only create one sampling-based instance of a stochastic program for each day and number of scenarios. For each of the 18 stochastic programming solutions we generate for a given day, we simulate a common sample path of arrivals to determine a sample abandonment rate and sample cost per handled call.

To develop a consistent measure of performance across all 76 days, we then use first differences. Given the relative stability of results for 400 scenarios, we use the 400-scenario results as benchmarks against which we compare that of formulations with fewer scenarios. For each type scenario-generation scheme, on each day we record the first difference between the performance of schemes with 1 through 225 scenarios to that of the math program with 400 scenarios.

Figure 2 plots the first differences, by number of scenarios, of the total costs. The left panel plots the differences for math programs with quadrature-based scenarios, and the right panel plots the differences for those using sampling-based scenarios. Each of the 76 lines in a panel plots the first differences for one day, by number of scenarios. The black circles show the means of the first differences across all 76 days.

Figure 2's results again suggest that, for quadrature-based formulations with more than one scenario, the means of the first differences are all quite close to zero; there is no apparent bias introduced by using fewer

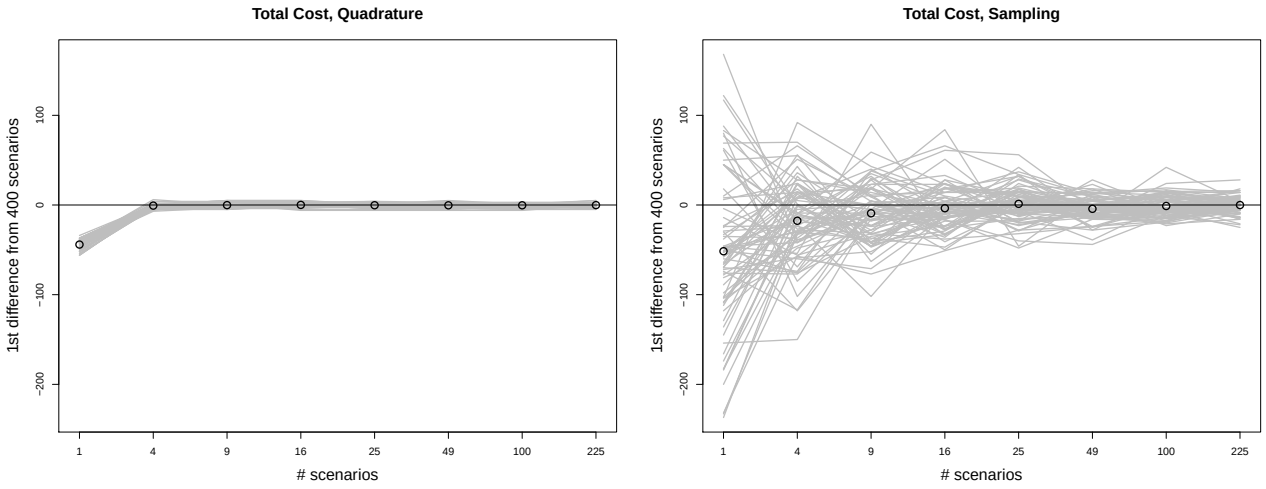

Figure 2: First Differences in Cost, by Number of Scenarios

scenarios, as long as there are more than one. Formulations using one scenario, again, display the expected systematic bias toward understaffing that was noted in Figure 1. Visual inspection also suggests that the results for sampling-based formulations are systematically noisier than those using quadrature-generated scenarios.

2.5.3 First Differences in Cost per Handled Call and Abandonment Rate Across 76 Days

Observe that our use of the objective function value as a measure of solution stability is crude: two optimal solutions, $x \neq y$, can have $\sum_{j \in \mathcal{J}} c_j x_j = \sum_{j \in \mathcal{J}} c_j y_j$. Nevertheless, Figures 1 and 2 both show that, even with this rough measure of stability, sampling-based solutions appear to be relatively unstable for smaller numbers of scenarios.

We would similarly like to rule out the possibility that the apparent stability of quadrature-based solutions is overstated by the use of objective function values. While one alternative would be to use an L^2 (or some other) norm, the possibility that an IP such as (6) has multiple optimal solutions motivates us to look instead at differences in the abandonment rates across solutions.

For our third test, we therefore consider both costs and abandonment rates for all 76 out-of-sample days. For each day we use each stochastic program's optimal solution to determine each half hour's staffing level and then use discrete event simulation to calculate a sample realization of the abandonment rate.

More specifically, the numbers of agents on hand each half hour is determined by the optimal solution to the IP (6). For each of the 76 out-of-sample days we simulate one sample path of a 13-hour day. The simulated arrival process is driven by our dataset's 30-minute arrival counts. Because we do not have the arrival time of each call in the dataset, we turn each 30-minute count into the realization of a Poisson arrival process by distributing the counted number of arrivals as *i.i.d.* samples, each of which has an arrival time that is uniformly distributed over the half-hour. (See §2.3 in Ross (1996).) For each arrival, we sample a virtual service time and

virtual patience, each of which is exponentially distributed with mean $1/\mu$ and $1/\theta$, respectively.

Within each simulation run, the number of agents on hand may decrease from one half hour interval to the next. In such cases, we remove agents with shortest remaining service time first. (Idle agents have zero remaining service times.) At the end of the simulated day, we also check to make sure that the number of unserved calls left in queue is not large enough to bias abandonment-rate statistics, which are calculated as fractions of arriving calls. Across 1,368 test simulations, the average number of calls left in the end of the queue was 1.21, out of a total of 8,435.79 average daily arrivals. The maximum across all simulations was 15, on a day with 8,631 arrivals.

For each simulation run, we track two statistics. First, we calculate the fraction of arrivals that abandon before being served: the number of abandonments, divided by the number of arriving calls, including those in queue at the end of the day. Second, we calculate the average cost per handled call: the staffing cost, as recorded in the objective function value, divided by the number of served calls. Here, the number of served calls equals the number of arrivals, less the number of abandonments (and again includes calls left in queue at the end of the day). We use cost per served call, rather than the average cost over all calls, so that staffing plans with high abandonment rates do not appear to have artificially low cost per call.

number of	abandonment rate		cost per handled call	
scenarios	quadrature	sampling	quadrature	sampling
1	0.0097	0.0488	0.0017	0.0083
4	0.0016	0.0215	0.0003	0.0045
9	0.0016	0.0133	0.0003	0.0034
16	0.0010	0.0086	0.0002	0.0030
25	0.0016	0.0065	0.0002	0.0022
49	0.0014	0.0051	0.0002	0.0014
100	0.0014	0.0054	0.0002	0.0012
225	0.0016	0.0037	0.0002	0.0010

Table 1: Standard Deviations of 1st Differences in Abandonment Rate and Cost per Handled Call

Table 1 reports the standard deviations of the first differences for both statistics. In most cases, the standard deviation of the quadrature approach is one order of magnitude less. As before, the results suggest that stable quadrature-based solutions are obtained by using as few as four scenarios, while many more sampling-based scenarios are needed.

3 Efficacy of Stochastic Programs with and without Recourse

In the previous section, we demonstrated the effectiveness of using Gaussian quadrature to generate scenarios for simple, one-stage stochastic programs. In this section, we build on the machinery developed in §2 to compare the effectiveness of this one-stage approach with schemes that, part-way through the planning horizon, use arrival-count data to revise the arrival-rate forecast and adjust schedules accordingly.

For example, additional agent capacity may be requested for the 2nd part of the horizon. Often this additional capacity comes in the form of extended hours for existing agents, the addition of short-term call-in agents, or the use of outsourcing capacity. Conversely, capacity may be reduced during the 2nd part of the horizon, typically by encouraging agents to take unpaid time off. The use of forecast updates, combined with these so-called "recourse" actions, adds complexity to both the forecasting and scheduling approaches described in §2.

We consider two forms of updates. In §3.1, we describe a less sophisticated scheme that uses the one-stage stochastic programming approach of §2 before the start of the planning horizon. Part-way through the horizon, this method then utilizes realized arrival counts to update the arrival-rate forecast and solve a related one-stage stochastic program with recourse actions for the remaining time intervals. In §3.2 we detail a more complex scheduling method. This scheme follows the same overall approach as the simpler one, but it uses a more sophisticated, two-stage recourse program in the initial planning phase. In setting initial staffing levels, this two-stage program explicitly accounts for the later use of recourse actions, which the simpler approach of §3.1 does not.

In §4 we test the efficacy of the various schemes on two sets of call-center data. The first test uses the European call-center data we use in §2. The second uses analogous arrival-count data from the call-center network of a North American retail bank.

3.1 Simple Forecast Updates and Recourse Actions

The simpler update scheme begins by forecasting and scheduling using the approach developed in §2, without considering the fact that recourse actions can be taken part-way through the planning horizon. Then after some update time, $i^* \in \{1, \ldots, I-1\}$, it uses the new arrival counts for the early-stage intervals $i \in \mathcal{I}_e \equiv \{1, \ldots, i^*\}$ to develop a revised forecast for late-stage intervals $i \in \mathcal{I}_l \equiv \{i^*+1, \ldots, I\}$, and it runs a new one-stage stochastic program. This second math program begins with the optimal solution from the original scheduling program, (6), and it includes decision variables that represent recourse actions that add or remove agent capacity from the initial solution. Additional constraints limit recourse actions according to work rules and the numbers of available agents. The solution of the second math program determines how employee schedules will change in response to the updated arrival-rate forecast.

Note that this simpler scheme is a generalization of that proposed by Mehrotra et al. (2010), in which the initial math program and the one-stage recourse program are simple IPs that use only one scenario. The formulation below generalizes that scheme to use multiple scenarios.

3.1.1 Forecast Updates and Scenario Generation

As in §2, let $\Lambda_i = (\omega\vartheta_i)^2$ be the uncertain arrival rate of the ith interval within a particular day. We assume that, using historical data, we have already obtained a forecast distribution for ω as $\omega \sim \mathrm{N}(\zeta, \phi^2)$.

We define i^* to be the index of an interval during the day after which we update the arrival-rate forecast and adjust agent schedules. We update the forecast, having observed the numbers of calls arriving during $\mathcal{I}_e$, the intervals prior to the update, and denote these counts as $N_e = \{N_1, \ldots, N_{i^*}\}$. According to the forecasting model (1), we then have

$$y_i \equiv \sqrt{N_i + 1/4} = \sqrt{\Lambda_i} + \epsilon_i = \omega\vartheta_i + \epsilon_i, \quad i \in \mathcal{I}_e, \tag{7}$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$ and $\omega \sim \mathrm{N}(\zeta, \phi^2)$. Note that both σ^2 and ϑ_i can be assumed to be known, given the model (1) has been estimated using historical data. Denote the vector of transformed observations collectively as $\mathbf{y}_e = (y_1, \ldots, y_{i^*})^\mathsf{T}$.

The problem of interest is to update the distribution of Λ_i for $i \in \mathcal{I}_l$ based on the new information, $\mathbf{y}_e$. The following proposition describes how the updated distribution for ω, which drives Λ_i, is calculated.

Proposition 2 *Let $\boldsymbol{\vartheta}_e = (\vartheta_1, \ldots, \vartheta_{i^*})^\mathsf{T}$, and let I_{i^*} be the identity matrix of dimension i^*. Then, having observed $\mathbf{y}_e$, the posterior distribution of ω is normal with a mean of*

$$\zeta^* = \gamma(\boldsymbol{\vartheta}_e^\mathsf{T}\boldsymbol{\vartheta}_e + \gamma)^{-1}\zeta \; + \; (\boldsymbol{\vartheta}_e^\mathsf{T}\boldsymbol{\vartheta}_e + \gamma)^{-1}\boldsymbol{\vartheta}_e^\mathsf{T}\mathbf{y}_e, \tag{8}$$

where $\gamma = \sigma^2/\phi^2$, and a variance of

$$\phi^{*2} \; = \; \phi^2 - \phi^4\boldsymbol{\vartheta}_e^\mathsf{T}(\phi^2\boldsymbol{\vartheta}_e\boldsymbol{\vartheta}_e^\mathsf{T} + \sigma^2 I_{i^*})^{-1}\boldsymbol{\vartheta}_e. \tag{9}$$

From (8), we see that the posterior mean of ω, ζ^*, is a weighted average of the original point forecast, ζ, and the new observations, $\mathbf{y}_e$. From (9), we see that the posterior variance, ϕ^{*2}, is always smaller than the original variance ϕ^2. Thus, the updating process shifts the forecast mean to account for new arrival counts, and these additional data also reduce uncertainty of the forecast. The updated distribution of $\Lambda_i, i \in \mathcal{I}_l$, then naturally follows in light of (3).

Remark 6 While the approach taken here is parametric, our updated mean, ζ^* is equivalent to that obtained using the penalized-least-squares (PLS) update that was developed as a part of the purely data-driven approach proposed by Shen and Huang (2008). □

Once we have obtained the updated forecast distribution for the arrival rates of the late stage, we again use the Gaussian quadrature approach of §2.2 to generate a new set of scenarios. The number of updated scenarios may (in theory) differ from that previously used, and we differentiate updated scenario data by labeling the revised set of scenarios $k \in \mathcal{K}' = \{1, \ldots, K'\}$, with probabilities $\{p'_k \mid k \in \mathcal{K}'\}$ and scenario-dependent rates $\{\lambda'_{ik} \mid i \in \mathcal{I}_l, k \in \mathcal{K}'\}$.

3.1.2 Recourse Program for Simple Updates

With the initial set of schedules and the revised forecasts in hand, we solve a schedule-update problem. Its solution determines a set of recourse actions to be used to adjust staffing levels over the second half of the planning horizon.

For each schedule assignment, $j \in \mathcal{J}$, we define a set of feasible recourse actions, $\mathcal{H}_j = \{1, \ldots, H_j\}$. If $a_{ij} = 0$ for some $i \in \mathcal{I}_l$ then we may be able to extend schedule j to have an agent assigned to that schedule to work during interval i. Similarly, if $a_{ij} = 1$ for some $i \in \mathcal{I}_l$ then we may be able to reduce schedule j to have an agent assigned to that schedule idle during interval i. We therefore let

$$r_{ijh} = \begin{cases} +1, & \text{if recourse action } h \text{ extends schedule } j \text{ by having an agent work during interval } i; \\ -1, & \text{if recourse action } h \text{ reduces schedule } j \text{ by having an agent idle during interval } i; \\ 0, & \text{otherwise,} \end{cases}$$

for $i \in \mathcal{I}_l$, $j \in \mathcal{J}$ and $h \in \mathcal{H}_j$. If we define a dummy schedule J to have $a_{iJ} = 0$ for all i and $c_J = 0$, then we can represent the ability to use call-in agents or outsourcing capacity in a similar fashion. As with the a_{ij}s, feasible columns – $(r_{i^*+1,j,h}, \ldots, r_{Tjh})^{\mathsf{T}}$ – are determined by company policy and employee work rules.

The decision variables $\{z_{jh} \mid j \in \mathcal{J}, h \in \mathcal{H}_j\}$ denote the set of recourse actions to be taken after the schedule update. For each z_{jh}, a coefficient d_{jh} defines the unit cost of taking the recourse action. Positive costs are associated with the addition of work hours, either through schedule extensions or the use of call-in capacity. Negative costs (savings) may results from the ability to reduce agents' working hours.

The math program, below, is an analogue of that for the first-stage problem, (6), with piecewise-linear constraints providing lower bounds to expected abandonment rates across the K' scenarios. As in (6), each $\mathcal{N}_i$ is the set of linear constraints used to bound the expected number of abandoning customers in interval $i \in \mathcal{I}_l$, with $\{(m_{in}, b_{in}) \mid n \in \mathcal{N}_i\}$ defining the slopes and intercepts. Here the x_j's are numbers that were previously determined via (6).

Given fixed initial schedule, x, the following math program then finds a set of recourse actions, z, that

minimize recourse costs, subjected to the revised QoS constraint, α_l:

$$\begin{array}{lll}
\min \sum_{j\in\mathcal{J}} \sum_{h\in\mathcal{H}_j} d_{jh} z_{jh} & & \\
\text{subject to} & & \\
(\sum_{j\in\mathcal{J}} a_{ij} x_j + \sum_{j\in\mathcal{J}} \sum_{h\in\mathcal{H}_j} r_{ijh} z_{jh}) m_{in} + b_{in} & \leq & \alpha_i \qquad i \in \mathcal{I}_l;\ n \in \mathcal{N}_i \\
\sum_{i\in\mathcal{I}_l} \alpha_i & \leq & \alpha_l \bar{\lambda}' \\
\sum_{h\in\mathcal{H}_j} z_{jh} & \leq & x_j \qquad j \in \mathcal{J} \\
z_{jh} & \in & \mathbb{Z}^+ \qquad j \in \mathcal{J},\ h \in \mathcal{H}_j.
\end{array} \tag{10}$$

Here, the first two constraints of (10) define the piecewise-linear lower bounds on expected abandonment rates and enforce a revised QoS limit, $\alpha_l \bar{\lambda}'$, only over $\mathcal{I}_l$.

Specifically, we let $\bar{\lambda}' = \sum_{i\in\mathcal{I}_l} \sum_{k\in\mathcal{K}'} p'_k \lambda'_{ik}$ be the revised expected aggregate arrival rate over the second part of the planning horizon and let

$$\alpha_l(N_e) = \left[\sum_{k\in\mathcal{K}} p'_k \sum_{i\in\mathcal{I}_l} \lambda'_{ik} f(\lambda'_{ik}, \mu, \theta, \sum_{j\in\mathcal{J}} a_{ij} x_j) \right] / \bar{\lambda}' \tag{11}$$

denote the revised expectation of late-stage abandonment that would have occurred had the original staffing plan, x, been maintained. We write $\alpha(N_e)$ to emphasize the fact that QoS update, α_l, is driven by the early-interval arrival counts, N_e.

Our definition of α_l ensures that, on a sample-path basis, expected abandonment over the late part of the planning horizon remains the same with and without recourse actions. In turn, over the entire planning horizon, our simple recourse scheme will achieve the same expected QoS as the one-stage stochastic schedule, developed in §2, at a (weakly) lower cost. Thus, the recourse scheme offers a Pareto improvement of scheduling without recourse.

Remark 7 Our definition of α_l has the virtue of being straightforward to calculate and analyze. Nevertheless, there are other definitions of α_l that we can consider. For example, we can use the optimal solution to (6) to define the expected late-period abandonment rate given only the initial forecast (1): $\alpha_l = (\sum_{i\in\mathcal{I}_l} \alpha_i)/(\sum_{i\in\mathcal{I}_l} \lambda_i)$. More generally, one may look for mapping of $\alpha_l(N_e)$ that satisfies other objectives, such as best stabilizing late-interval abandonment or minimizing of expected late-interval costs. Space limitations prevent us from explicitly considering the form of other $\alpha_l(\,\cdot\,)$ functions, though we are working on their characterization (Gans et al. 2012). □

To summarize, we operationalize the simple scheme in two stages. Before the start of the planning horizon, we forecast call volumes as in §2 and solve (6) to determine an initial set of schedules, $\{x_j \mid j \in \mathcal{J}\}$, which we use to staff the call center over the early part of the planning horizon, $\mathcal{I}_e$. During this initial period, we collect

arrival count data and use the results of Proposition 2 to update the arrival-rate forecast during later part of the horizon. We feed the initial schedule, $\{x_j \mid j \in \mathcal{J}\}$, along with the updated forecast, into the recourse program (10), whose solution determines optimal schedule adjustments and, in turn, the number of agents on hand in each period $i \in \mathcal{I}_l$.

3.2 Forecasting and Scheduling Using Two-Stage Recourse Programs

A more complex approach explicitly takes the ability to use recourse actions into account when solving the initial forecast and staffing plan. To support this approach both the initial forecast (1) and scheduling program (6) become more elaborate.

The solution to a two-stage recourse program determines optimal values for both initial scheduling decisions and later recourse actions. But in practice only the values of the initial scheduling decisions, $\{x_j \mid j \in \mathcal{J}\}$, are used. As in the simpler updating scheme, they are used to schedule staff during the early part of the planning horizon, $\mathcal{I}_e$, and after i^* we use the forecast update and recourse machinery developed in §3.1 to update the forecast, based on actual data, and to determine the specific set of recourse actions to be implemented.

Thus, the practical difference between the simpler and the more complex updating schemes is that the former develops its initial scheduling decision, $\{x_j \mid j \in \mathcal{J}\}$, without regard to recourse actions, while the latter develops the initial schedule explicitly accounting for the cost effectiveness of possible recourse actions. For example, if adding agent capacity after the update is more expensive than initially overstaffing and then sending agents home, then the two-stage recourse program may set initial staffing levels to be higher than the simpler early-stage program, which does not account for the relative costs of later capacity increases and decreases.

3.2.1 Scenario Generation for Recourse

In the more complex approach, we structure the initial forecast as follows. As before, we divide the planning horizon into two stages, an early stage, $\mathcal{I}_e$, and a late stage, $\mathcal{I}_l$, and we generate scenarios on a stage-specific basis. We construct early-stage scenarios $k \in \mathcal{K}$ that represent arrival-rate patterns over only the first part of the horizon. With each early-stage scenario $k \in \mathcal{K}$ we then associate a distinct set of late-stage scenarios, $\mathcal{L}_k = \{1, \ldots, L_k\}$. Each set of second-stage scenarios, $\mathcal{L}_k$, occurs conditional on scenario k being realized during the early part of the horizon.

As in §3.1.1, we assume the original forecast distribution for ω is $\mathrm{N}(\zeta, \phi^2)$, and we use the same quadrature scheme to generate early-stage scenarios: $\lambda_{ik} \equiv \omega_k^2 \vartheta_i^2$ with probability p_k for $k \in \mathcal{K}$. Because we must construct late-stage scenarios before the start of the planning horizon, before any arrival counts are observed,

the results of Proposition 2 do not apply directly, however.

More specifically, Proposition 2 uses the arrival-count pattern during the early stage of the horizon, $\mathbf{y}_e$, to update the forecast distribution of ω. However, the two-stage stochastic program with recourse is run before the start of the planning horizon, when we have not yet observed $\mathbf{y}_e$. In fact, in generating late-stage scenarios we must consider all possible realized arrival patterns in the early stage; i.e., all possible $\mathbf{y}_e$.

Nevertheless, we can use the proposition's results as the basis for a more elaborate update mechanism, one in which we generate potential sample paths, $\mathbf{y}_e$. For each generated sample path, we calculate a conditional update, and having calculated all such potential updates, we then aggregate them as a mixture.

Consider the kth early-stage arrival-rate scenario ω_k. The model (7) suggests that

$$y_i = \omega_k \vartheta_i + \epsilon_i, \quad i \in \mathcal{I}_e,$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$. Applying the Gaussian quadrature approach of §2.2 to the distribution of the error term, ϵ_i, we obtain a discrete approximation with a set, $\mathcal{B} = \{1, \ldots, B\}$, of discrete atoms, $\{\epsilon_b \mid b \in \mathcal{B}\}$, each with probability $\{q_b \mid b \in \mathcal{B}\}$.

Using this discrete approximation for each of the i^* early-stage error terms, we see that, for any given ω_k, $\mathbf{y}_e$ has B^{i^*} possible sample paths,

$$\mathbf{y}^k_{b_1,\ldots,b_{i^*}} \equiv \boldsymbol{\vartheta}_e \omega_k + (\epsilon_{b_1}, \ldots, \epsilon_{b_{i^*}})^\mathsf{T} \quad \text{with probability} \quad p_{b_1,\ldots,b_{i^*}} \equiv \prod_{i=1}^{i^*} q_{b_i}.$$

The superscript, k, emphasizes the fact that these B^{i^*} sample paths are constructed given early-stage scenario k.

Now consider one "realized" count scenario $\mathbf{y}^k_{b_1,\ldots,b_{i^*}}$. As in Proposition 2, the updated distribution for ω is Gaussian with mean defined as in (8),

$$\zeta^k_{b_1,\ldots,b_{i^*}} = \gamma(\boldsymbol{\vartheta}_e^\mathsf{T}\boldsymbol{\vartheta}_e + \gamma)^{-1}\zeta \;+\; (\boldsymbol{\vartheta}_e^\mathsf{T}\boldsymbol{\vartheta}_e + \gamma)^{-1}\boldsymbol{\vartheta}_e^\mathsf{T}\mathbf{y}^k_{b_1,\ldots,b_{i^*}},$$

and variance ϕ^{*2} defined as in (9). We denote the updated ω_k as $\omega^k_{b_1,\ldots,b_{i^*}}$.

Hence, the updated distribution for ω_k over all possible sample paths of $\mathbf{y}_e$ is a mixture of Gaussian distributions, $\omega^k_{b_1,\ldots,b_{i^*}}$, where the mixture probabilities are $p_{b_1,\ldots,b_{i^*}}$. Note that, because each $\omega^k_{b_1,\ldots,b_{i^*}}$ has variance of ϕ^{*2}, the mixture does as well. For each early-stage scenario, k, we then discretize this mixture distribution into L_k scenarios, $\{\omega_{kl} \mid l \in \mathcal{L}_k\}$ with probabilities $\{p_{kl} \mid l \in \mathcal{L}_k\}$.

Here, we use a simple approach, dividing the approximate support of the mixture distribution – from roughly 3 standard deviations below the mean of the lowest distribution in the mixture to 3 standard deviations above the mean of the highest distribution in the mixture – into L_k equal-length intervals, setting p_{kl} to be the conditional

probability that, given early-stage scenario $k \in \mathcal{K}$, ω is in the lth interval, and letting ω_{kl} be the corresponding conditional expectation of ω.

Remark 8 An alternative would be to use the Gaussian quadrature approach, described in §2.2, to discretize the mixtures defined above. Because these second-stage scenarios are not implemented and are used only indirectly, to account for the effect of later recourse actions on earlier scheduling decisions, we have opted for computational efficiency at the expense of precision. □

Finally, we repeat the above procedure for each early-stage scenario, $k \in \mathcal{K}$, in order to generate the corresponding late-stage scenarios, $l \in \mathcal{L}_k$. For each k, we transform the resulting ω_{kl}s according to (3) to yield the late-stage arrival rates, $\{\lambda_{ikl} = (\omega_{kl}\vartheta_i)^2 \mid i \in \mathcal{I}_l,\ k \in \mathcal{K},\ l \in \mathcal{L}_k\}$.

3.2.2 Two-Stage Recourse Program Formulation

As in the simpler scheme, the two-stage recourse program includes decision variables, $\{x_j \mid j \in \mathcal{J}\}$, that represent initial scheduling decisions, implemented before the start of the planning horizon. It also determines possible late-horizon recourse decisions, which can vary by scenario. Decision variables $\{z_{jhk} \mid j \in \mathcal{J}, h \in \mathcal{H}_j, k \in \mathcal{K}\}$ represent the full set of these recourse decisions. If early-stage scenario $\hat{k}$ is realized, then recourse decisions $\{z_{jh\hat{k}} \mid j \in \mathcal{J}, h \in \mathcal{H}_j\}$ are taken.

As in (6) and (10), we formulate the piecewise-linear version of the two-stage recourse program. We let $\{\alpha_i \mid i \in \mathcal{I}_e\}$ be the expected number of abandoning calls in each interval of the early part of the planning horizon. Similarly, for each early-stage scenario, $k \in \mathcal{K}$, we let $\{\alpha_{ik} \mid i \in \mathcal{I}_l\}$ be the analogous quantities in the second part of the planning horizon. As before, α^* is an upper bound on the expected abandonment rate over the entire horizon.

Again, each $\mathcal{N}_i$ is set of linear constraints that bounds the expected number of abandoning customers in intervals $i \in \mathcal{I}_e$, with $\{(m_{in}, b_{in}) \mid n \in \mathcal{N}_i\}$ defining slopes and intercepts. We let $\mathcal{N}_{ik} = \{0, \ldots, N_{ik}\}$ define the analogous sets of constraints for each interval $i \in \mathcal{I}_l$ in the later part of the planning horizon and under each early-stage scenario, $k \in \mathcal{K}$, with slopes and intercepts $\{(m_{ikn}, b_{ikn}) \mid n \in \mathcal{N}_{ik}\}$.

Then the solution to the stochastic integer program, below, determines an optimal set of scheduling and recourse decisions. Its objective is to minimize the expected cost of initial scheduling and recourse decisions, subject to an upper bound on the expected number of abandonments over the planning horizon, $\alpha^*\bar{\lambda}$. The first set of constraints provides lower bounds on the expected numbers of abandonments during intervals $i \in \mathcal{I}_e$, and the second set provides a similar bound for each scenario $k \in \mathcal{K}$ during $i \in \mathcal{I}_l$. The third constraint enforces an upper bound on the expected number of abandonments over the planning horizon, and the fourth

set of constraints ensures that at most one recourse action may be taken for each employee.

$$
\begin{array}{l}
\min \sum_{j\in\mathcal{J}} c_j x_j + \sum_{k\in\mathcal{K}} p_k \sum_{j\in\mathcal{J}} \sum_{h\in\mathcal{H}_j} d_{jh} z_{jhk} \\
\text{subject to} \\
\begin{array}{lcll}
(\sum_{j\in\mathcal{J}} a_{ij} x_j) m_{in} + b_{in} & \leq & \alpha_i & i \in \mathcal{I}_e, n \in \mathcal{N}_i \\
(\sum_{j\in\mathcal{J}} a_{ij} x_j + \sum_{j\in\mathcal{J}} \sum_{h\in\mathcal{H}_j} r_{ijh} z_{jhk}) m_{ikn} + b_{ikn} & \leq & \alpha_{ik} & i \in \mathcal{I}_l;\ k \in \mathcal{K};\ n \in \mathcal{N}_{ik} \\
\sum_{i\in\mathcal{I}_e} \alpha_i + \sum_{k\in\mathcal{K}} p_k \sum_{i\in\mathcal{I}_l} \alpha_{ik} & \leq & \alpha^* \bar{\lambda} & \\
\sum_{h\in\mathcal{H}_j} z_{jhk} & \leq & x_j, & j \in \mathcal{J},\ k \in \mathcal{K} \\
x_j & \in & \mathbb{Z}^+ & j \in \mathcal{J} \\
z_{jhk} & \in & \mathbb{Z}^+ & j \in \mathcal{J},\ h \in \mathcal{H}_j,\ k \in \mathcal{K}
\end{array}
\end{array}
\tag{12}
$$

We operationalize the more sophisticated scheme as in §3.1. First, we use the x determined by the optimal solution of recourse program (12) to assign agent schedules in the early part of the planning horizon. Then after interval i^* we update the arrival-rate forecast and use (10) to determine the actual recourse actions to be taken. For the current case, we define the late-stage revision of the QoS target as

$$
\alpha_l(N_e) \;=\; \sum_{k\in\mathcal{K}} p_k^{'} \sum_{i\in\mathcal{I}_l} \alpha_{ik} \;/\; \bar{\lambda}', \tag{13}
$$

which uses revised scenarios probabilities, $\{p_k^{'} \,|\, k \in \mathcal{K}\}$, along with the 2nd-stage abandonment-rate targets from (12), $\{\alpha_{ik} \,|\, i \in \mathcal{I}, k \in \mathcal{K}\}$, to ensure that the use of (12) followed by an update that uses (10) matches the expected abandonment rate attained by (6) on a daily basis.

Observe that, in contrast to the simple scheme, the use of (12) followed by (10) is not guaranteed to provide weakly lower costs than (6). While the simple scheme's early-stage schedules exactly match those of (6), the cost of the early stage schedules determined by (12) might be higher or lower than those of (6). Thus, while we expect the use of the more sophisticated 2-stage recourse scheme to lower long-run average costs, over many days, it need not provide the simpler update scheme's guarantee of (weakly) lower costs each day.

4 Numerical Tests of Six Scheduling Schemes

We now have the machinery necessary to precisely define each of six scheduling schemes we will evaluate. (See Table 2.) SP1 and SPm solve the stochastic program (6) to find a schedule x and perform no updating. UP1 and UPm solve (6) at the start of the planning horizon, update the initial arrival-rate forecast after interval i^*, and then solve (10) to determine recourse actions to take in $\mathcal{I}_l$. RP1 and RPm solve (12) at the start of the planning horizon, update the initial arrival-rate forecast after interval i^*, and then solve (10) to determine recourse actions to take in $\mathcal{I}_l$. Schemes SP1, UP1, and RP1 use $K = 1$ scenarios, while SPm, UPm, and RPm use $m > 1$ scenarios.

Label	Description
SP1	one-stage stochastic program with 1 scenario and no updating
SPm	one-stage stochastic program with $m > 1$ scenarios and no updating
UP1	one-stage stochastic program with 1 scenario and mid-horizon updating
UPm	one-stage stochastic program with $m > 1$ scenarios and mid-horizon updating
RP1	two-stage recourse program with 1 scenario and mid-horizon updating
RPm	two-stage recourse program with $m > 1$ scenarios and mid-horizon updating

Table 2: Six Scheduling Schemes to be Tested

Note that three of these schemes correspond to approaches found elsewhere: 1) SP1 is a traditional IP, driven by a point forecast; 2) SPm is the quadrature-based version of Robbins and Harrison (2010) evaluated in §2; and 3) RP1 is the simple forecast-update approach evaluated in Mehrotra et al. (2010).

For the scheduling schemes that use multiple scenarios, we construct sets of $m = 100$ scenarios so that the information content across schemes is relatively constant. In particular, for RP100, we use $K = 10$ early-stage scenarios, and for each early-stage scenario we use $L_k = 10$ late-stage scenarios. Thus, for the second part of the planning horizon, the number of scenarios is 100.

We calculate each of the $L_k = 10$ late-stage scenarios using a mixture of B^{i^*} posterior, normal distributions, as described in §3.2.1. We let $B = 2$ so that calculation uses a standard binomial tree (Cox et al. 1979).

To ensure that the information content for SP100 is analogous to that of RP100 we use the latter's forecast data in SP100 as well. To do so, groups of 10 scenarios in SP100 have the same early-stage rate profiles as that associated with one of the 10 early-stage scenarios in RP100. Then within a given set of 10 with the same early-stage profile, each has a different late-period profiles just as in RP100.

In both the UP100 and RP100 schemes, we determine actual recourse actions using the sampling-based update program, (10). In both cases the recourse program has $K' = 100$ distinct scenarios.

In the following sections we test these schemes on two sets of data. The first set of tests uses the European retail bank data described in §2.4. A second set of tests uses arrival-count data from a network of call centers operated by a North American retail bank.

4.1 Empirical Results for European Retail Bank

In addition to the data described in §2.4, our empirical tests for the European bank use the definition of the recourse actions and costs that are used in (10) and (12). Because the European bank's labor practices are highly restricted (compared with those in the US, for example), it does not currently use recourse actions, however. Therefore, we use a set of recourse actions and costs that are analogous to those used by Mehrotra et al. (2010) .

Specifically, we consider three sets of recourse actions. For any worker assigned to first-stage schedule, $j \in \mathcal{J}$, we consider two generic actions: 1) the ability to extend the worker's shift, beyond the time it would normally end; and 2) the ability to send the worker home early, before his or her schedule would normally end.[2] In the former case, we assume a traditional overtime cost of 1.5 per agent per half hour, a 50% premium over the base rate of 1 per interval, and in the latter we define the cost to be -0.75. A third set of recourse actions is the ability to (outsourced or) call in workers who are not scheduled to work on a given day. We assume that the cost of this action is 2 per agent per half hour; these agents receive double-time pay. Given our problem parameters – 26 half-hour intervals per day and 262 feasible initial schedules – the number of feasible recourse actions totals 4,973.

We choose a fixed update interval, revising the initial forecast and determining recourse actions at 11 a.m. – that is, after $i^* = 6$ intervals of arrival-count data are observed. In contrast, Mehrotra et al. (2010) do not set a fixed interval, i^*, for an update. Rather, they perform a sequential procedure that looks for the first period, i^*, for which they can reject the null hypothesis that the arrival-rate pattern comes from the initial forecast distribution.

Remark 9 While this more sophisticated sequential procedure is of interest, it would add significant complexity to the large number of forecasts we generate. Of perhaps greater practical interest would be a search for an optimal static update interval. Such a test would be straightforward, though time consuming, and we do not pursue it here. □

As in §2.5 we use 176 days of weekday arrival data from the European bank. For each of the last 76 days, we use the previous 100 days to construct out-of-sample forecasts for the following day, and we run the math programs required to implement the six scheduling schemes of interest. For each day and each scheme, we use the optimal solutions to the initial scheduling and, in the case of the UP and RP schemes, update programs to determine staffing counts for each half-hour of the day. Then for each day and each scheme, we run a single sample path of a discrete event simulation to generate the number of realized abandonments. From the objective function values, arrival counts, and abandonment counts we calculate the realized abandonment rate and cost per handled call.

Figure 3 summarizes the results of all 76 days. The left panel plots confidence intervals for the 76 realized abandonment rates, and the right panel intervals for the 76 costs per handled call. The intervals' point estimates are calculated as weighted averages of the 76 days' results, with the number of calls handled on a given day

[2]Note that we require schedule adjustments of be made over contiguous sets of intervals. For example, an agent who is originally scheduled to work until 5:00 p.m., and who is asked to work from 6:30 p.m. to 7:00 p.m., must work from 5:00 p.m. to 6:30 p.m. as well.

acting as weights. Similarly, the 1/2-widths of the confidence intervals are calculated using standard deviations whose points are weighted by numbers of calls, along with t-statistics associated with 95% intervals and 75 degrees of freedom.

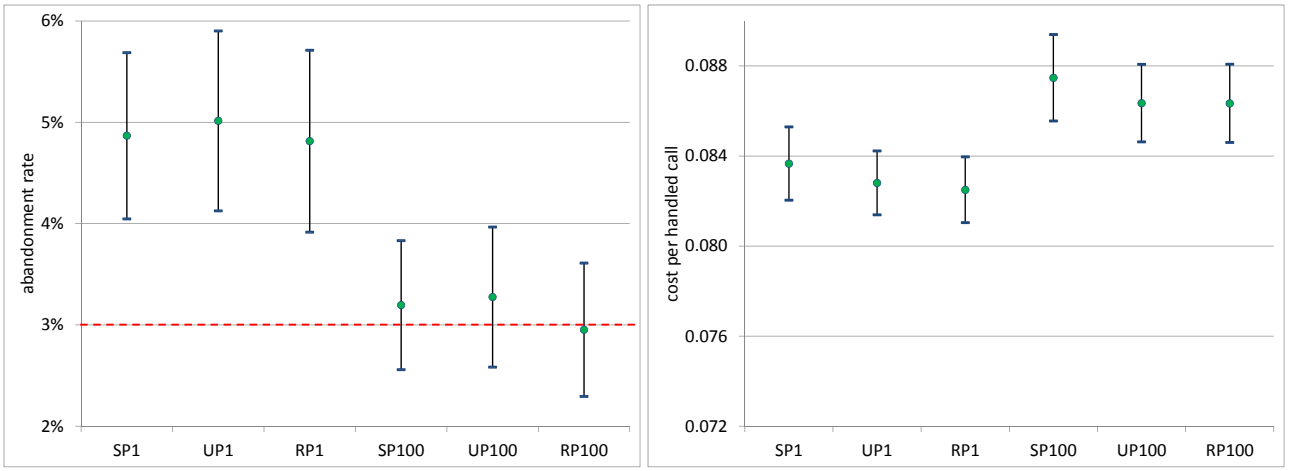

Figure 3: Six Schemes' Performance for European Bank over 76 Days

In the left panel of Figure 3, we see that the three schemes that use 100 scenarios have average abandonment rates that are close to 3%, while those that use only one scenario – point forecasts for arrival rates – have significantly higher average abandonment rates. Conversely, the three schemes that use 100 scenarios have costs that are significantly higher than those that use point forecasts. This result echoes that shown in Figures 1 and 2. Since the abandonment rate is convex and increasing in the arrival rate, schemes that staff to hit an average abandonment rate of α^* across all arrival rates will require more staff and obtain lower average abandonment rates than those that staff to hit α^* for the average of the arrival rates.

The results also suggest that, for a given number of scenarios, average cost and abandonment rates are quite similar across the three schemes. In particular, average cost per handled call is only 1.3% lower for UP100 and RP100 than for SP100, and the wide confidence intervals make the difference look statistically insignificant.

Results from paired t-tests, displayed in Table 3, show that there are systematic differences, however. For example, the cost per handled call of UP100 and RP100 are both significantly lower than that of SP100, with vanishingly small p-values for one-sided t-tests. While average abandonment is not significantly different for SP100 and UP100 it is significantly lower for RP100.

The results have at least three important implications for the European bank's call center. First, schemes that are based on point estimates of arrival rates do not appear to provide adequate staffing to meet long-run QoS targets, while those that are based on distributional forecasts appear to meet QoS goals. Second, although the cost advantage of using recourse actions is statistically significant, the magnitude of the advantage is not large, and for the European bank there appears to be no cost advantage in using the more complex RP100 scheduling

paired	abandonment rate		cost per handled call	
difference	t	p-value	t	p-value
SP100 - UP100	-1.04	0.151	6.96	0.000
SP100 - RP100	2.63	0.005	7.00	0.000
UP100 - RP100	6.14	0.000	0.01	0.497

Table 3: One-Tailed Paired t-Tests Comparing SP100, UP100, and UP100 at European Bank

scheme, instead of the simpler UP100 scheme.

4.2 Empirical Results for a North American Retail Bank

The previous section's results suggest that, for the European bank's call center, the explicit representation of forecast error, rather than the availability of recourse actions, may be of the most practical value. To begin assessing the robustness of this finding, we construct a second set of tests that use arrival-count data from another operation, a network call centers operated by a North American retail bank. This North American bank's call centers operate at a larger scale, with call volumes that are more than six and one half times that of the European call center studied in §4.1.

We control these new tests so that their results are comparable to those from the European bank. In the new tests we continue to use the scheduling, rate, and cost assumptions and parameters used in the original experiments. The only differences between the two sets of numerical tests are the half-hour arrival counts found from day to day.

We have 210 days of arrival-count data for the North American bank and employ the same sampling and scheduling scheme as before, using the previous 100 days' arrival-count data to forecast arrival-rate profiles for days 101 through 210. Thus, in this example we have 110 out-of-sample points that we test. As before, we determine staffing numbers and costs according to the six scheduling schemes and then run a discrete event simulation to generate the number of realized abandonments. Again we use the objective function values, arrival counts, and abandonment counts to calculate realized abandonment rates and costs per handled call.

Figure 4 summarizes the results of the 110 out-of-sample days. The left panel plots confidence intervals for the realized abandonment rates, and the right panel intervals for cost per handled call. As before, the intervals' point estimates and 1/2-widths are determined using weighted calculations, with weights that are numbers of calls handled on each day. The t-statistics used are those associated with 95% intervals and 109 degrees of freedom. To make the plots visually comparable to those in Figure 3, we use the same vertical scales.

The results differ somewhat from those of the European bank. As before, the schemes that use distributional forecasts do a good job of reaching a long-run abandonment target of 3%, while those that use point forecasts

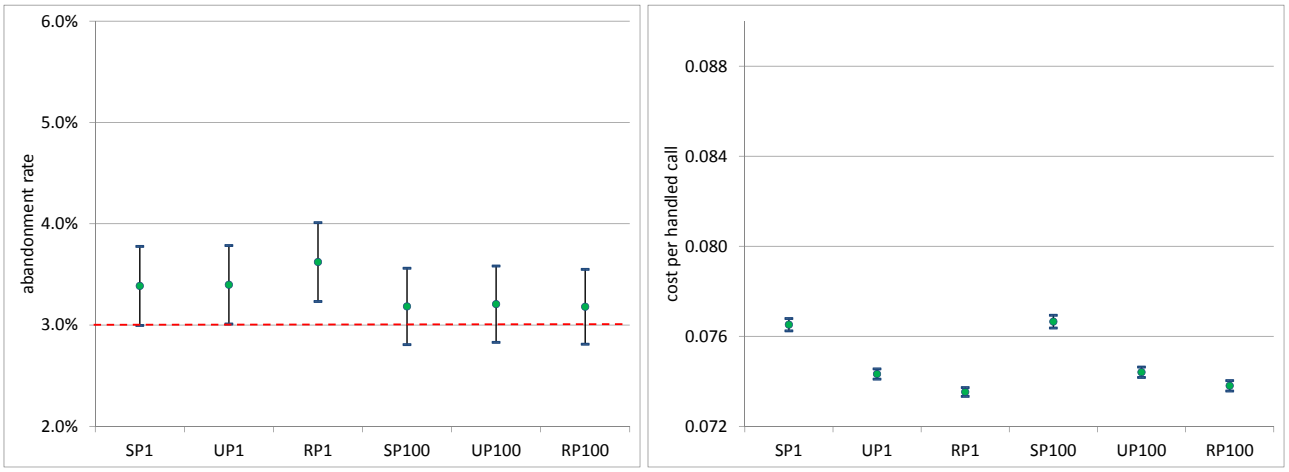

Figure 4: Six Schemes' Performance for North American Bank over 110 Days

appear to have abandonment rates that are biased upward. The differences are not as large as those found in the European bank's results, however. In contrast, the use of recourse actions provides more significant cost savings to the North American bank, with UP100's average cost per handled call falling 2.9% below SP100's, and RP100's costs falling 3.7% below SP100's. Table 4 confirms that, as the plots suggest, there are strong statistical differences among the average costs of the 100-scenario schemes.

paired	abandonment rate		cost per handled call	
difference	t	p-value	t	p-value
SP100 - UP100	-1.04	0.151	46.6	0.000
SP100 - RP100	0.07	0.473	66.7	0.000
UP100 - RP100	0.86	0.194	16.6	0.000

Table 4: One-Tailed Paired t-Tests Comparing SP100, UP100, and UP100 at North American Bank

Our numerical results also suggest that the arrival-rate forecasts are noisier for the European bank than they are for the North American Bank. For example, differences in the widths of the confidence intervals shown in Figures 3 and 4 are revealing. The widths of the North American bank's confidence intervals for abandonments are only 43% to 60% of those for the European banks, and those for average cost per handled call are only 13% to 15% of the European bank's.[3]

Table 5 confirms that, in fact, the CV of the forecast distribution for the daily arrival rate is about four times larger for the European bank than it is for its North American counterpart. Each row reports the distribution of the CV, across all out-of-sample forecasts. There are 76 such forecasts for the European bank and 110 for the North American bank.

[3]Confidence intervals constructed using the North American bank's first 76 days of out-of-sample forecasts are similar. The 1/2-width for abandonment rates are 51% to 71% of those of the European Bank, and those for average cost per handled call are 13% to 16% of the European bank's.

	Minimum	1st Quartile	Median	Mean	3rd Quartile	Maximum
European Bank	0.1141	0.1235	0.1313	0.1338	0.1407	0.1663
North American Bank	0.0284	0.0321	0.0333	0.0333	0.0346	0.0391

Table 5: Distribution of Daily Forecast-Distribution CVs over Out-of-Sample Tests

As we noted in §4.1, the fact that expected number abandoning is increasing convex in the arrival rate implies that schemes that staff to the average arrival rate should systematically understaff. Furthermore, the more disperse the arrival-rate forecast, the stronger the bias. Table 6 shows that, in our setting this intuition holds true. Noisy arrival-rate forecasts also translate into relatively larger cost differences between analogous 1- and 100-scenario schemes: SP1 vs. SP100; UP1 vs. UP100; and RP1 vs. RP100. For example, the percent cost increase between 1 and 100-scenarios is 4.2%-4.6% for the European Bank, an order of magnitude larger than the 0.1%-0.4% increase for the North American bank.

	SP1	SP100	% Diff	UP1	UP100	% Diff	RP1	RP100	% Diff
European Bank	0.084	0.088	4.5%	0.083	0.086	4.2%	0.083	0.086	4.6%
North American Bank	0.077	0.077	0.2%	0.074	0.074	0.1%	0.074	0.074	0.4%

Table 6: Average Cost Per Handled Call at the Two Banks

4.3 Discussion

In both examples, the use of multiple-scenario scheduling schemes was needed to meet long-run average QoS objectives. For both the European and North American banks, scheduling schemes that used 100 scenarios had long run average abandonment rates close to 3%, while those that used only one scenario had confidence intervals that did not cover 3%.

In contrast, the usefulness of the two forms of recourse varied between the two examples. While UP100 and RP100 provided statistically significant cost reductions for the European bank, the overall savings was only 1.3%. In addition, RP100 did not to provide a cost advantage over the simpler UP100. For the North American bank, the benefits appeared to be stronger, however. The average cost savings provided by UP100 was 2.9% and that provided by RP100 was 3.7%. Here RP100's day-by-day costs were statistically different, and lower, UP100's.

We also saw that the large North American centers enjoyed much lower levels of forecast uncertainty, when compared to much smaller the European bank. An interesting question, then, is the extent to which the reduction in forecast uncertainty enjoyed by the North American bank is the result of an increase in scale. For example, an arrival process that is an aggregate of separate, independent arrival processes – such as that obtained by pooling across independent geographic areas – should enjoy a reduced CV of the overall arrival rate. This phenomenon

would represent an as-yet unaccounted for source of economies of scale that warrants further investigation.

It is also worth noting that an initial motivation for the use of fluid models in complex staffing and routing problems, found in Harrison and Zeevi (2005), was that arrival-rate uncertainty may dwarf the lower-level stochastic fluctuations that are modeled by more complex queueing formulae. Our results suggest that this may be the case for European bank but not for its North American counterpart. To the extent that scale is associated with less variable forecasts, the rationale for using fluid models may apply most fully only to smaller operations that we might call "underscale."

5 Conclusion

Our analysis has provided a number of insights into the value of stochastic programming and recourse for call-center workforce management. We used a parametric forecasting scheme to generate stochastic programs that needed only small numbers of scenarios, and our use of a convex measure of QoS allowed us to collapse scenarios in linear time to create efficient, deterministic, piecewise linear, certainty-equivalent versions of these stochastic programs. Together, the forecasting scheme and certainty-equivalent formulation allowed us to simply generate and solve large numbers of two-stage recourse programs.

Numerical tests of our forecasting and scheduling schemes showed how the use of multiple scenarios and of recourse actions provided complementary benefits. Multiple scenarios were needed to achieve long-term QoS goals, and recourse actions improved system costs. A comparison of the European bank's and North American banks's arrival-rate forecasts also suggested that the same pooling effect that is widely recognized in inventory systems may provide an as-yet unaccounted for source of economies of scale in call center operations.

In developing our analysis we have remarked on follow-on work that we believe will further strengthen both the theoretical underpinnings and the practical value of our work. On the one hand, we are working to show that the arrival-rate distributions used in our stochastic programming formulations are consistent with the AR(1) models used to generate those forecasts. On the other hand, we are investigating more complex definitions of the late-interval QoS target α_l that will allow other objectives, such as reducing day-to-day fluctuations in realized abandonment rates.

More broadly, we wish to expand the above analysis to include multiple types of calls. We note that, in order to provide a comprehensive evaluation of the merits of various recourse schemes, we have limited ourselves to an operational model in which only one type of call is served. Nevertheless, call centers commonly handle multiple types of calls, and to accommodate this complexity, the analysis should extend to systems that require skills-based routing and staffing. A number of papers referred to in the introduction – including Harrison and Zeevi (2005), Bassamboo et al. (2005), Bassamboo et al. (2006), Bassamboo and Zeevi (2009), Bertsimas

and Doan (2010), and Gurvich et al. (2010) – use fluid models to account for arrival-rate uncertainty when making short-run staffing and call-routing decisions. We are currently working to incorporate elements of their approaches, extending our analysis to include multiple types of calls.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Numbers CMMI-0645075, CMMI-0800575 and CMMI-0800645.

References

Z. Akşin, M. Armony, and V. Mehrotra. 2007. The Modern Call-Center: A Multi-Disciplinary Perspective on Operations Management Research. *Production and Operations Management*, 16(6):655-688.

S. Aldor-Noiman, P. D. Feigin, and A. Mandelbaum. 2009. Workload Forcasting for a Call Center: Methodology and a Case Study. *Annals of Applied Statistics*, 3(4):1403-1447.

M. Armony, E. Plambeck, and S. Seshadri. 2009. Sensitivity of Optimal Capacity to Customer Impatience in an Unobservable M/M/S Queue (Why You Shouldn't Shout at the DMV). *Manufacturing & Service Operations Management*, 11(1):19-32.

A. N. Avramidis, A. Deslauriers, and P. L'Ecuyer. 2004. Modeling Daily Arrivals to a Telephone Call Center. *Management Science*, 50:896-908.

A. Bassamboo, J. M. Harrison, and A. Zeevi. 2005. Dynamic Routing and Admission Control in High Volume Service Systems: Asymptotic Analysis via Multi-Scale Fluid Limits. *Queueing Systems Theory and Applications,* 51:249–285.

A. Bassamboo, J. M. Harrison and A. Zeevi. 2006. Design and Control of a Large Call Center: Asymptotic Analysis of an LP-Based Method. *Operations Research,* 54:419–435.

A. Bassamboo and A. Zeevi. 2009. On a Data-Driven Method for Staffing Large Call Centers. *Operations Research*, 57(3):714-726.

D. Bertsimas and X. Doan. 2010. Robust and Data-Driven Approaches to Call Centers. *European Journal of Operational Research*, 207(2):1072-1085.

J. R. Birge and F. Louveaux. 1997. *Introduction to Stochastic Programming*. New York: Springer.

L. D. Brown, T. Cai, R. Zhang, L. Zhao, and H. Zhou. 2010. The Root-Unroot Algorithm for Density Estimation as Implemented via Wavelet Block Thresholding. *Probability Theory and Related Fields*, 146:401-433.

L. Brown, N. Gans, A. Mandelbaum, A. Sakov, H. Shen, S. Zeltyn, and L. Zhao. 2005. Statistical Analysis of a Telephone Call Center: a Queueing Science Perspective. *Journal of The American Statistical Association*, 100:36-50.

B. Chen and S. G. Henderson. 2001. Two issues in setting call centre staffing levels. *Annals of Operations Research*, 108:175-192.

J. C. Cox, S. A. Ross, and M. Rubinstein. 1979. Option Pricing: a Simplified Approach. *Journal of Financial Economics*, 7:229-263.

A. Deslauriers, P. L'Ecuyer, J. Pichitlamken, A. Ingolfsson and A. N. Avramidis. 2007. Markov Chain Models of a Telephone Call Center in Blend Mode. *Computers and Operations Research*, 34(6):1616-1645.

Z. Feldman, A. Mandelbaum, W. A. Massey, and W. Whitt. 2007. Staffng of Time-Varying Queues to Achieve Time-Stable Performance. *Management Science*, 54:324-338.

N. Gans, G. Koole, and A. Mandelbaum. 2003. Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing & Service Operations Management*, 5:79-141.

N. Gans, H. Shen, H. Ye, and Y-P. Zhou. 2012. Asymptotic Stability of AR(p)-Driven Workforce Scheduling Models. In preparation.

W. K. Grassman. 1988. Finding the Right Number of Servers in Real-World Queuing Systems. *Interfaces*, 18(2):94-104.

I. Gurvich, J. Luedtke, and T. Tezcan. 2010. Staffing Call Centers with Uncertain Demand Forecasts: A Chance-Constrained Approach. *Management Science*, 56(7):1093-1115.

L. V. Green, P. J. Kolesar, and W. Whitt. 2007. Coping with Time-Varying Demand When Setting Staffng Requirements for a Service System. *Production and Operations Management*, 16:13-39.

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

R. Ibrahim and P. L'Ecuyer, Forecasting Call Center Arrivals: Fixed-Effects, Mixed-Effects, and Bivariate Models. Working Paper, Université de Montréal.

G. Jongbloed and G. Koole. 2001. Managing Uncertainty in Call Centres Using Poisson Mixtures. *Applied Stochastic Models in Business and Industry*, 17:307-318.

S. Liao, G. Koole, C. van Delft, and O. Jouini. 2012. Staffing a Call Center with Uncertain Non-Stationary Arrival Rate and Flexibility. To appear in *OR Spectum*.

S. Maman. 2009. *Uncertainty in the Demand for Service: the Case of Call Centers and Emergency Departments.* Master's Thesis, Technion, Israel Institute of Technology.

A. Mandelbaum and S. Zeltyn. 2007. The M/M/n+G Queue: Summary of Performance Measures. Technical Note, Technion, Israel Institute of Technology.

V. Mehrotra, O. Ozluk, and R. Saltzman. 2010. Intelligent Procedures for Intra-Day Updating of Call Center Agent Schedules. *Production and Operations Management*, 19(3):353-367.

A. C. Miller and T. R. Rice. 1983. Discrete Approximations of Probability Distributions. *Management Science*, 29:352-362.

T. R. Robbins and T. P. Harrison. 2010. A stochastic Programming Model for Scheduling Call Centers with Global Service Level Agreements. *European Journal of Operational Research*, 207:1608-1617.

T. R. Robbins, D. J. Medeiros, and P. Dum. 2006. Evaluating Arrival Rate Uncertainty in Call Centers. In L.F. Perrone, F.P.Wieland, J. Liu, B.G. Lawson, D.M. Nicol, and R.M. Fujimoto (eds.), *Proceedings of the 2006 Winter Simulation Conference*. Piscataway NJ: IEEE, 2180-2187.

T. R. Robbins, D. J. Medeiros, and T. J. Harrison 2010. Cross Training in Call Centers with Uncertain Arrivals and Global Service Level Agreements. *International Journal of Operations and Quantitative Management*, 16(3):307-329.

A. M. Ross. 2001. Queueing Systems with Daily Cycles and Stochastic Demand with Uncertain Parameters. Ph.D. Dissertation, University of California, Berkeley.

S. M. Ross. 1996. *Stochastic Processes*, 2nd ed. New York: Wiley.

A. Shapiro. 2000. Stochastic Programming by Monte Carlo Simulation Methods Stochastic Programming e-Print Series, 2000(3), `http://www.speps.org`.

A. Shapiro and A Philpott. 2007. A Tutorial on Stochastic Programming. Technical Note, Georgia Institute of Technology.

H. Shen and J. Z. Huang. 2008. Interday Forecasting and Intraday Updating of Call Center Arrivals. *Manufacturing & Service Operations Management*, 10:391-410.

S. G. Steckley, S. G. Henderson, and V. Mehrotra. 2009. Forecast Errors in Service Systems. *Probability in the Engineering and Informational Sciences*, 23(2):305-332.

J. W. Taylor. 2012. Density Forecasting of Intraday Call Center Arrivals Using Models Based on Exponential Smoothing. To appear in *Management Science*.

J. Weinberg, L. D. Brown, and J. R. Stroud. 2007. Bayesian Forecasting of an Inhomogeneous Poisson Process with Applications to Call Center Data. *Journal of the American Statistical Association*, 102:1185-1199.

W. Whitt. 1999. Dynamic Staffing in a Telephone Call Center Aiming to Immediately Answer All Calls. 1999. *Operations Research Letters*, 24:205-212.

W. Whitt. 2006. Fluid Models for Many-Server Queues with Abandonments. *Operations Research*, 54:37-54.

E. Zohar, A. Mandelbaum, and N. Shimkin. 2002. Adaptive Behavior of Impatient Customers in Tele-Queues: Theory and Empirical Support. *Management Science*, 48(4):566-583.