An Optimal Callback Policy for General Arrival Processes: A Pathwise Analysis

Baris Ata • Booth School of Business, The University of Chicago
Xiaoshan Peng • Kelley School of Business, Indiana University

Abstract

This paper studies the callback option as an instrument to effectively mitigate congestion due to temporary surges in arrivals to a call center. The call arrival process can be an arbitrary point process, allowing uncertainty and temporary surges in the arrival rate, provided that the system is stable. When a customer arrives, the call center manager examines the system state and decides whether to offer the incoming customer the callback option or not. When the callback option is offered, the customer decides whether to accept the offer. The customer is routed to the offline queue (to be called back later) only if he is offered the callback option and accepts it. Otherwise, he is routed to the online queue. For each customer in the online queue, the call center manager incurs a waiting cost of $h$ per time unit. Similarly, whenever she routes a customer to the offline queue (for a callback later), she incurs a one-time penalty of $p$. Initially, we allow complete foresight policies that look into the entire future. We show that a simple lookahead policy that looks into the future arrivals and service completion times for the next $p/h$ time units and uses the current number of customers in the system who previously rejected a callback offer (but does not look into the accept/reject decisions of future customers) is pathwise optimal. Building on the insights gleaned from the optimal lookahead policies, we also propose a non-anticipating (and implementable) policy by interpreting the lookahead policy in the fluid model. In particular, we show that this policy reduces to the so called line policy, if the arrival rate process follows a Cox-Ingersoll-Ross process. Lastly, we conduct a simulation study which shows that the proposed policies perform well.

1 Introduction

One important aspect of modeling arrivals to a call center is that the arrival rate changes over time as shown in empirical studies. In particular, the recent literature on call centers have shown that the within-day call arrival process has shown two important properties: Time dependence of call arrival rates and over-dispersion of arrival counts; see Ibrahim et al. (2016) and references therein. The call arrival rates show intraday seasonality. In particular, the intraday seasonality can be described by the daily profile, which is calculated by dividing a day into time periods of equal
length (usually 30 to 60 minutes) and averaging the arrival counts within these time periods across days. A natural model to account for time dependence of the arrival rate is the non-homogeneous Poisson process, in which the arrival rate process is a deterministic but time-varying function; see for example Brown et al. (2005). One implication of the non-homogeneous Poisson model is that the variance and the mean of the arrival counts in a given time period are equal. However, this contradicts the empirical evidence: the variance of the arrivals is usually larger than the mean. This observation is called the over-dispersion.

Recent literature has developed new arrival models to explain the level of the over-dispersion observed in the data. For example, Glynn et al. (2018) show that the over-dispersion can be explained by the behavior of the arrival rate process at the mesoscopic time scale. The mesoscopic time scale lasts from minutes to an hour, longer than the impact of the Poisson noise in the arrival counts but shorter than the time scale of the intraday seasonality. In particular, Glynn et al. (2018) observe that when call arrivals are observed on the mesoscopic time scale, they exhibit much higher variability and have no clear seasonality patterns, i.e. the call arrival rate may experience unpredictable fluctuations, i.e. temporary dips or surges, on this time scale. They model the arrival rate process as a mean-reverting stochastic process to characterize the unpredictable fluctuations. Orenshkin et al. (2016) also propose several different arrival rate models which allow flexible auto-correlation structure in the arrival rate process. The flexibility of these models help explain the level of the over-dispersion and the unpredictable fluctuations observed in multiple datasets.

The callback option is an effective way to manage such temporary and unpredictable bursts of the arrivals. It works as follows: when the system is congested, the call center manager notifies the arriving customer and presents him with the option to hang up to be called back later within a reasonable time window. When the system congestion decreases, outbound calls are initiated for such customers. Consequently, the callback option allows the call center manager to shift some of the calls arriving when the system is undergoing temporary arrival surges to a period when the system is less busy. A survey conducted by Software Advise shows that more than half of the customers are willing to wait for more than 30 minutes offline for the call center to get back to them.

---

1 ContactBabel (2016) shows in a survey that the forecast based callback, in which the call center calls the offline customers at a time that suits the call center, is the second most popular (after first-come-first-served) type of callback offered.

2 Further results of the survey can be found at http://csi.softwareadvice.com/3-ways-to-offer-callback-0614/.
which is of the same order of magnitude of the mesoscopic time scale mentioned above. Thanks to the customers who are willing to accept the callback option, the call center manager can improve the online queue performance significantly using the callback option.

This paper develops an effective and simple callback policy using a canonical queueing model. The call center consists of two queues: An online queue and an offline queue. When a customer arrives, the call center manager examines the system state and decides whether to offer the incoming customer the callback option or not. If the call center offers the callback option to the customer, he decides whether to accept o the offer. The customer is routed to the offline queue (to be called back later) only if he is offered the callback option and accepts it. Otherwise, he is routed to the online queue. The call center incurs a waiting cost of $h$ per unit time for each customer waiting in the online queue, whereas it incurs a one-time penalty of $p$ if the customer is routed to the offline queue. We refer the call center manager’s decision as the callback decision. In what follows, we assume that the time commitment for the callback option is sufficiently large so that we drop it in our theoretical model. Indeed, our simulation study shows that the offline delays are acceptable under the policies we derive. The extensive analysis of the data from three different call centers in Orenshkin et al. (2016) shows that none of the arrival models they tested is universally good. The choice of the arrival model should be made based on the data of the specific call center. Thus, it is critical to propose a framework to study the callback policy for general arrival models.

We first assume that the call center manager can observe all future arrival and service times and take a pathwise analysis approach to analyze this problem. We show that a simple lookahead policy is optimal for general arrival processes. This policy keeps track of only the total number of customers in the system (the sum of the numbers of customers in both the online and offline queues), which is insensitive to the routing and priority schemes for work conserving policies and the number of customers in the online queue who have rejected the callback offer. When a new customer arrives, this policy looks into the future and checks how long it would take for the total number of customers in the system to drop below a threshold, which is the current level minus the current number of customers in the system who previously rejected the callback offer. If the time exceeds the threshold of $p/h$, the call center offers the callback option to the incoming customer; otherwise, the customer is routed to the online queue. This policy has several virtues. First, it only needs to keep track of the total number of customers in the system and the current number of
customers in the system who previously rejected the callback offer. Second, it requires only limited future information (over the next \( p/h \) time units) although it is optimal among policies that can look into the entire future. Third, it is pathwise optimal. Thus, the optimality result carries over to general settings with an essentially arbitrary call arrival process provided the system is stable. In addition, the optimality result does not rely on any particular assumption on the customers’ behavior of accepting or rejecting the callback offers, supporting general models.

Although the lookahead policy is not directly applicable in practice because it needs to look into the future, it provides useful insights and helps motivate an effective control policy. For example, it suggests not waiting until the queue builds up to start offering the callback option, motivating a preemptive intervention as opposed to a reactive one. In particular, it suggests offering the callback option as soon as one anticipates a sufficiently high arrival rate, i.e. offering the callback option when the arrival rate exceeds a threshold.

We also propose an implementable policy by interpreting the lookahead policy in the context of a fluid model, which lends itself to a natural non-anticipating policy. We further show that the lookahead policy derived from the complete foresight analysis is optimal for the fluid model. Since the callback option is most useful in mitigating the temporary bursts in the call volumes, we study a special case using the arrival model proposed in Glynn et al. (2018). In particular, Glynn et al. (2018) assumes that the arrival rate process model follows a Cox-Ingersoll-Ross (CIR) model. We show in this case that the lookahead policy for the fluid model simplifies to a policy that offers the callback option when a linear combination of the online queue length and the current arrival rate exceeds a certain threshold. This policy is referred to as the line policy, which is easily implementable in and effective for the original system.

Lastly, we conduct a simulation study to explore the performances of the lookahead and line policies further. Since our analytical results are based on the pathwise and fluid analysis, we check their robustness by varying the value of the volatility term in the CIR process. We also test their performances when the arrival rate process is modeled as a reflected Ornstein-Uhlenbeck process. As one may expect, the lookahead policy far outperforms the optimal non-anticipating policy in the simulation study because it uses the future information. Interestingly, the line policy has an excellent performance in most cases except for the case when the volatility of the arrival rate process dominates the (mean-reverting) drift term. Moreover, the results are robust to customers’
accept/reject decisions of the callback option unless the fraction of customers willing to accept the callback offer is too low. Lastly, we conduct a case study and calibrate the system using a dataset from a US bank call center. We observe that routing a small fraction of customers to the offline queue results in excellent system performance, i.e., small waiting times in the online queue and manageable callback delays (for the offline queue). We also check how abandonments impact the performance and find that as long as the abandonment rates are low, the lookahead and line policies perform about as well as they would without any abandonments (despite the increases in the system load due to the reduced number of abandonments). However, if the abandonment rate is high, the callback delays may become excessive. This is because the callback option significantly lowers the number of abandonments, thereby increasing the system load significantly. Consequently, we conclude that the callback option under the line policy can be an effective way to mitigate the arrival rate variability provided that the abandonment rate is not too high.

The rest of this paper is organized as follows. Section 2 reviews the related literature. Section 3 introduces the queueing model to study the callback option. Section 4 studies the complete foresight policies and proves the optimality of the $p/h$-lookahead policy. Section 5 studies a fluid model and advances a non-anticipating policy by interpreting the lookahead policy in the fluid model. Section 6 presents the numerical study and Section 7 concludes. Appendix A provides the system equations characterizing the evolution of the queue length processes. Appendices B and C provide the proofs of the lemmas in Section 4 and Section 5, respectively. Appendix D proves the optimality of the line policy. Appendix E provides additional theoretical and numerical analysis that are supplementary to the results of Section 6.

2 Literature Review

This paper is related to three streams of literature. The first stream models the nonstationary arrivals to a call center. The second stream focuses on the callback option specifically, whereas the last stream studies admission control decisions for queueing systems. There is a growing literature on modeling the nonstationary arrivals to a call center\(^3\); see Ibrahim et al. (2016) for a recent

\(^3\)Nonstationary demand models are relevant in other operations management applications; see for example, Besbes and Maglaras (2009) for a revenue management application, Shi et al. (2016) for a study of a discharge policy in an emergency department, Chen et al. (2018) for an approximation scheme for arrival data in a psychiatric ward and an emergency department and Ata et al. (2018) for an empirical analysis of time-based pricing in electricity markets.
review. For example, Green and Kolesar (1991) propose the pointwise stationary approximations (PSA) to model the time-varying demand; also see Jennings et al. (1996), Harrison and Zeevi (2005), Bassamboo et al. (2006), Feldman et al. (2008), Liu and Whitt (2012), Gans et al. (2015), Pender and Massey (2017) and Qin and Pender (2018). Whitt (2018) provides an extensive review of research on the performance analysis of queueing systems with time-varying arrival rates; see also Gans et al. (2003), Green et al. (2007) and Aksin et al. (2007) and the references therein.

The second stream of literature studies call blending, a process managing both the inbound and outbound calls in a call center. Inbound calls originate from outside customers, whereas the outbound calls are initiated by the call center; see for example, Bhulai and Koole (2003), Gans and Zhou (2003), Pang and Perry (2015) and Deslauriers et al. (2007). One critical assumption in this literature is that there is infinite supply of the outbound calls. Whitt (1999) studies a system in which some customers can tolerate substantial delays, similar to the system offering the callback option. Two key papers in the second stream are Armony and Maglaras (2004a, b). Armony and Maglaras (2004a, b) study a call center which offers the callback option to arriving customers who upon arrival choose among staying in the online queue, receiving a callback (i.e. joining an offline queue) and balking. Our model and the style of analysis differ significantly from theirs. First, we consider a model of the arrival rate process that allows bursty arrivals. Second, we study a stable i.e. an underloaded system, whereas they focus on the critically loaded regime, i.e. the heavy traffic regime. In particular, in our setting the callback option is only relevant when the system experiences temporary (and unpredictable) surges in the call volume. Thus, the call center manager exercises the callback option judiciously as opposed to offering it throughout the day. The recent paper of Legros et al. (2016) also studies when to offer the callback option using a Markov decision process model. The authors propose a threshold policy whereby the callback option is triggered whenever the online queue length exceeds the threshold. An arriving customer receives a delay announcement, and the callback offer (if the online queue length exceeds the threshold). One important difference of our work from Legros et al. (2016) and Armony and Maglaras (2004a, b) is that we incorporate the time-varying and uncertain nature of the call arrival rate in our model.

Instead of using the staffing level as a mechanism to manage the non-stationary random arrival rate, which may be too costly or even infeasible on the mesoscopic time scale identified by Glynn et al. (2018), we focus on managing the arrival volume itself. The latter mechanism is related to the
admission control literature; see Stidham (1985) and Stidham (2002) for overviews and references therein. Most of the admission control literature formulate the problem as a Markov decision process. Examples include Lewis et al. (1999), Ata and Shneorson (2006), Lewis et al. (2002), Yoon and Lewis (2004), Örmeci et al. (2001), Zayas-Cabán and Lewis (2018), Hampshire and Massey (2010) and Cudina and Ramanan (2011). There are several papers that study the admission control problem using diffusion approximations. Examples include Ward and Kumar (2008), Ata and Olsen (2009), Rubino and Ata (2009) and Kocaga et al. (2015). In particular, Kostami and Ward (2009) also study a system with an offline waiting option. They solve the optimal capacity allocation that minimizes average cost when the offline queue customers are allowed to abandon.

A closely related paper in the admission control literature is Spencer et al. (2014). The authors study an admission problem for an overloaded $M/M/1$ queue under the assumption that the future information is available. The authors propose the no-job-left-behind policy, which effectively rejects those jobs with “excessive” delay (hence, left behind) by looking into the future. The authors show that the no-job-left-behind policy is asymptotically optimal in heavy traffic. Xu and Chan (2016) consider a similar model to manage the admission into an emergency department using the knowledge of future arrivals. They also enhance their policies using the thresholds on the queue length, which diverts arrivals if either the queue length is large or a high number of patients will arrive in future periods. The authors show that the proposed policy provides delay improvements over standard policies used in practice. Our model differs from Spencer et al. (2014) and Xu and Chan (2016) in several ways. First, instead of studying an overloaded system, we assume that the system is stable, i.e. underloaded. However, it may experience temporary surges in the arrival rate due to the novel model of the arrival rate process we assume. The stability assumption simplifies the analysis, allowing us to conduct an exact analysis as opposed to an asymptotic analysis as done in Spencer et al. (2014). Indeed, we prove that the $p/h$-lookahead policy is pathwise optimal, a strong notion of optimality. Lastly, our model allows the customers to reject the callback offers. This feature of our model can be viewed as a job joining the system despite the call center manager’s desire to not admit it in an admission control problem. To the best of our knowledge, this important feature of our model is not considered in the admission control literature previously, including the aforementioned papers.

Another closely related paper is Yoon and Lewis (2004), which also study the admission control
problem with the time-varying arrival rate. The authors consider a system in which the arrival and service rates are bounded, periodic functions of time. They show that under the infinite horizon discounted and average reward optimality criteria, for each fixed time, optimal admission control strategies are threshold-type policies, i.e., the optimal policy admits customers when the queue length is below a threshold. They propose a pointwise stationary approximation (PSA) to approximate the optimal policies, suggest a heuristic to improve the implementation of the PSA and verify its usefulness via a numerical study. We propose a different heuristic in Section 5, which also falls into the threshold-type category. There are several benefits of our proposed policy. First, it is optimal for the associated fluid system. Second, it has a closed-form characterization, making it easily implementable. As we shown in Section 5.2, the closed-form characterization has a simple form for the CIR arrival rate model. Lastly, the numerical study in Section 6 shows that our proposed policy performs well.

3 The Model

We consider a canonical single-server queueing model with a single class of homogenous customers. Customer arrivals are allowed to follow a general point process provided the system is stable. In particular, the arrival times of customers are given by the (increasing) sequence \( \{\tau_i : i \geq 1\} \), where \( \tau_i \) denotes the \( i \)th customer’s arrival time. Then \( A(t) = \sup\{i \geq 1 : \tau_i \leq t\} \) for \( t \geq 0 \) gives the number of arrivals by time \( t \geq 0 \). We assume that the following holds:

\[
E[A(t)] = E\left[\int_0^t \lambda(s) \, ds\right], \quad t \geq 0,
\]

where \( \{\lambda(t) : t \geq 0\} \) is the arrival rate process, which we allow to be a general stochastic process. A commonly used arrival model is the non-homogenous Poisson process, in which \( \{\lambda(t) : t \geq 0\} \) is a deterministic process. Another example, proposed in Glynn et al. (2018), assumes that \( \{\lambda(t) : t \geq 0\} \) follows a Cox-Ingersoll-Ross (CIR) process.

We assume for analytical convenience that the service times are i.i.d. exponential random variables with rate \( \mu \). We also make the crucial assumption that the service times are associated with the server not with the customers. To be specific, letting \( \{\nu_j : j \geq 1\} \) denote the sequence of

\footnote{Our model can be generalized in various directions. For example, we expect most of our results continue to hold for multi-server systems.}
service times, where $\nu_j$ denotes the time between the $j$-th and $(j - 1)$-st service completions, we assume that they are independent of the index of the customer served. This assumption facilitates a policy-invariant characterization of the evolution of the total number of customers in the system. In turn, this property plays a critical role in proving the optimality of the policy proposed in Section 4. In addition, assume that the service times and the arrival process are independent. Letting $S(t)$ denote the total number of customers served by time $t$ if the server works continuously over $[0, t]$, it is given by $S(t) = \sup \{n \geq 1 : \sum_{i=1}^{n} \nu_i \leq t\}$ for $t \geq 0$. In addition, we assume that there is at most one arrival or one service completion at any given time.

We restrict attention to work-conserving policies. As mentioned above, we assume that the system is stable under them, but it may experience temporary surges in customer arrivals, during which the arrival rate exceeds the service rate. (This is facilitated by a general arrival process as assumed above). To accommodate such surges in arrivals, the system offers the callback option to arriving customers. The call center manager’s problem can be described as follows: When a customer arrives, she observes the system and decides whether to offer the callback option to the incoming customer. On the one hand, if the customer does not receive the callback offer, he joins the online queue and waits for service. On the other hand, if he receives the offer, he decides whether to accept the offer. If he declines it, he joins the online queue and waits for service. However, if he accepts the offer, he hangs up and waits for a callback offline, i.e. in the offline queue.

We refer the call center manager’s decision of offering the callback option as the callback policy in the foregoing development. The callback policy is represented with the sequence $I = \{i_k : k \geq 1\}$ of the indices of the customers receiving the callback option. The sequence $A = \{j_k : k \geq 1\}$ denotes the indices of customers who are willing to accept the callback offer (if offered one), whereas its complement $R = \{1, 2, \ldots\}\setminus A$ denotes the sequence of customers who will reject it (if offered) and stay in the online queue. The set $A$ (and thus its complement $R$) is not known ex ante, i.e. an index may appear in the set $A$ even if the customer does not receive the callback option for a given policy. The call center manager only observes the customers’ accept/reject decisions when a callback offer is made. In addition, let the sequence $I_1$ denote the indices of the customers who joined the online queue, i.e. those who either did not accept the callback offer or did not receive it. Similarly, let $I_2$ denote the indices of customers who joined the offline queue. Note that $I_2 = I \cap A$, i.e. only the customers in set $A$ may join the offline queue, which happens if they are offered the callback
option. This state of affairs can equivalently be described as the call center manager routing the arriving customers to one of two queues: the online queue versus the offline queue; see Figure 1. Given a callback policy \( I \), let \( A_1(t) \) and \( A_2(t) \) denote the cumulative numbers of customers routed to the online and offline queues up to time \( t \), respectively. Note that

\[
A_1(t) = \sup \{ k : i_k \in I, \, \tau_{i_k} \leq t \}, \quad t \geq 0,
\]

where we set \( \sup \emptyset = 0 \) for notational convenience. In addition, we have that \( A_2(t) = A(t) - A_1(t) \) for \( t \geq 0 \). For each customer waiting in the online queue, the call center manager incurs a waiting cost of \( h \) per time unit. On the other hand, there is a one-time penalty of \( p \) for sending a customer to the offline queue.

In addition to the callback policy, the call center manager also decides on how to prioritize the customers waiting in the online and offline queues. For simplicity, we allow preemptive-resume scheduling. Given the cost structure, it is optimal to give strict (preemptive) priority to the online queue. Thus, we focus attention on the work-conserving service policy that gives strict priority to the online customers. Also we assume that both the online and offline queues are served in a FCFS fashion. Let \( Q_1(\cdot) \) and \( Q_2(\cdot) \) denote the online and offline queue length processes, respectively. In addition, let \( Q(t) \) denote the total number of customers in the system at time \( t \). The evolutions of \( Q(\cdot) \), \( Q_1(\cdot) \) and \( Q_2(\cdot) \) are fully characterized by the primitive processes \( \{ A(t) : t \geq 0 \} \) and \( \{ S(t) : t \geq 0 \} \), the set \( A \) indicating the indices of customers who are willing to accept the callback option, the callback policy \( I \) and the service policy; see Appendix A for the system equations characterizing their evolutions. The long-run average cost in a callback policy \( I \), denoted by \( C^I \), is given as follows:

\[
C^I = \lim_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ pA_2(t) + h \int_0^t Q_1(s) \, ds \right].
\]

The call center manager strives to choose a policy \( I \) that minimizes \( C^I \). Clearly, customers in the offline queue will be served eventually, because the system is stable. However, a natural question
that arises is what the magnitude of delay they experience is. Section 5.2 specializes our analysis to the case where the arrival process is a Poisson process with stochastic intensity. In particular, we will model the intensity (or the arrival rate) process as a mean-reverting diffusion, e.g. CIR process; see Glynn et al. (2018). In this setting, the delays experienced by the customers in the offline queue can be estimated using the mean-reversion speed of the diffusion. Our numerical analysis in Section 6.3 shows that mean offline delays are less than 30 minutes in most cases.

In what follows, we first consider the complete foresight policies, where the call center manager knows the sequences of arrival times \( \{\tau_i : i \geq 1\} \) and service times \( \{\nu_j : j \geq 1\} \) in Section 4. Building on the insights gleaned from the complete foresight analysis, we also study a fluid model and characterize its optimal policy in Section 5. Using that optimal policy, we then propose a non-anticipating policy for the original model. Moreover, we show in Section 5.2 that the non-anticipating policy reduces to a simple policy, called the line policy, under the CIR model of the arrival rate process proposed by Glynn et al. (2018).

4 The Optimal Policy with Complete Foresight

This section considers the system with complete foresight. To be specific, the call center manager knows the sequences of arrival times \( \{\tau_i : i \geq 1\} \) and service times \( \{\nu_j : j \geq 1\} \). Without loss of generality, it suffices to analyze a busy period of the system, i.e. the period when the system is not empty. Because the system is stable, the number of arrivals in a busy period is finite almost surely. Fix a busy period and assume there are \( n \) arrivals in that busy period, whose arrival times are given by \( \tau_1, \tau_2, \ldots, \tau_n \). In this section, we will reuse the notation \( \mathcal{A}, \mathcal{R}, \mathcal{I}, \mathcal{I}_1 \) and \( \mathcal{I}_2 \) to denote the same quantities as in Section 3, but restricting attention to the customers arriving in the current busy period, i.e. customers \( \{1, 2, \ldots, n\} \). Thus, \( \mathcal{A} \subseteq \{1, 2, \ldots, n\} \) is the set of customers who are willing to accept the callback offer if they are offered one, whereas its complement \( \mathcal{R} = \{1, 2, \ldots, n\} \setminus \mathcal{A} \) is the set of customers who will reject it and stay in the online queue. The set of customers receiving the callback option is denoted by the set \( \mathcal{I} \subseteq \{1, 2, \ldots, n\} \), referred as the callback policy. Recall that the customers in set \( \mathcal{I}_2 = \mathcal{I} \cap \mathcal{A} \) are removed from the online queue and join the offline queue. In what follows, we consider the total queue length process \( Q \) and ask what happens if we remove a subset \( \mathcal{I}_2 \) of customers upon their arrival and route them to the offline queue. Namely,
we are interested in characterizing the resulting online queue length process \( Q_1 \). To this end, we let \( Q_1^T = \Phi(Q, I_2) \) denote the online queue length process resulting from routing the customers in the set \( I_2 \) to the offline queue. The operator \( \Phi(\cdot, \cdot) \) is defined suitably to represent this operation. Thus, the optimal callback policy seeks the set \( I \subseteq \{1, 2, \ldots, n\} \) of customers to offer the callback option so as to minimize the total cost during this busy period, i.e.

\[
\min_{I \subseteq \{1, 2, \ldots, n\}} p|I_2| + hH(\Phi(Q, I_2)),
\]

where \( |I_2| \) denotes the cardinality of the set \( I_2 \) and \( H(\tilde{Q}) \) the area under the queue length process \( \tilde{Q} \). Thus, the second term in Equation (1) is the holding cost incurred by the customers in the online queue during the busy period under the callback policy \( I \).

Two types of customers join the online queue: Customers who receive the callback offer and reject it, and those who do not receive the callback offer. To facilitate the analysis to follow, we consider two virtual queues within the online queue: Queue \( r \) and queue \( n \). Customers who receive the callback offer and reject it join queue \( r \), whereas those who do not receive the callback offer join queue \( n \). Letting \( Q_r(t) \) and \( Q_n(t) \) denote the lengths of queues \( r \) and \( n \), respectively, note that

\[
Q_1(t) = Q_r(t) + Q_n(t), \; t \geq 0.
\]

In addition, defining the random time

\[
s_i = \inf \{ t \geq \tau_i : Q(t) = Q(\tau_i) - Q_r(\tau_i) \}, \; i = 1, 2, \ldots, n,
\]

the following definition introduces the proposed policy.

**Definition 1.** The \( w \)-lookahead policy offers the callback option to customer \( i \) if \( s_i \geq \tau_i + w \) for \( i = 1, 2, \ldots, n \).

The \( w \)-lookahead policy only uses the revealed preferences of customers regarding the callback offers. Namely, it does not assume that we know the preferences of those customers who are not offered the callback option. Neither does it require the knowledge of future customers’ preferences. In other words, when a customer arrives, the call center manager does not know if he belongs to set \( A \) or set \( R \). She only finds out if she offers the callback option, because customers in the set \( A \) would accept, whereas those in set \( R \) would reject. Although the call center manager does not know customers’ preference ex ante, she adjusts her callback decisions after observing rejections.
of the callback option. To be specific, Equation (2) suggests that the call center manager is more likely to offer the callback option to incoming customers after observing more rejections.

The following theorem establishes that the \( p/h \)-lookahead policy is optimal.

**Theorem 1.** The \( p/h \)-lookahead policy is optimal under possible rejections of the callback offers.

The rest of this section is dedicated to proving Theorem 1. To this end, Section 4.1 considers the special case when all customers receiving the callback offer take it, i.e. \( A = \{1, 2, \ldots, n\} \). We show that the lookahead policy is pathwise optimal for this special case. Building on this result, Section 4.2 proves the optimality of the lookahead policy for the general case (i.e. a general set \( A \)).

### 4.1 The optimal policy when all customers accept the callback offers

This subsection studies the special case of \( A = \{1, 2, \ldots, n\} \). Under this assumption, the set of customers routed to the offline queue is equivalent to the set of those receiving the callback option, i.e. \( I_2 = I \cap A = I \). Thus, the problem stated in Equation (1) simplifies to the following:

\[
\min_{I \subseteq \{1, 2, \ldots, n\}} p|I| + hH(\Phi(Q, I)).
\]  

(3)

In addition, the sequence of times \( \{s_i : i \geq 1\} \) given in Equation (2) that are used to define the policy becomes

\[
s_i = \inf\{t \geq \tau_i : Q(t) = Q(\tau_i-)\}, \ i \geq 1.
\]  

(4)

Note that in this case, the time \( s_i \) denotes the next time (after customer \( i \)'s arrival) when the number of customers in the system \( Q(t) \) falls back to the level just before customer \( i \)'s arrival. Recall our assumption that service times are associated with the server and not with the customers. Consequently, the evolution of \( Q(t) \) is independent of the callback or service policies provided that the service policy is work conserving. Figure 8 in Appendix B.3 provides the routing decisions of two customers when the system operates under the \( p/h \)-lookahead policy for a given sample path.

The main result of this subsection, stated next, proves that the \( p/h \)-lookahead policy is optimal when all customers accept the callback offers, i.e. \( A = \{1, \ldots, n\} \).

**Proposition 1.** The \( p/h \)-lookahead policy is optimal when all customers accept the callback offers.
a greedy fashion (Definition 2). Second, we observe that this greedy policy is optimal; see Lemma 1. Third, we analyze a virtual system in which customers in the online queue are served under last-come-first-served (LCFS) policy. We show a key property of the greedy policy, which leads to a much simpler characterization of it in this virtual system; see Lemma 3. Fourth, this property and the optimality of the greedy policy is used to deduce the optimality of the $p/h$-lookahead policy. Lastly, we argue that the $p/h$-lookahead policy is optimal in the original system (where customers in the online queue are served under first-come-first-served (FCFS) fashion) as well.

The greedy policy is defined iteratively. First, assume that all customers are routed to the online queue. In this system, denoted by superscript $\emptyset$, the online queue length process $Q_1^{\emptyset}(t)$ coincides with the total number of customers in the system. That is $Q_1^{\emptyset}(t) = Q(t)$ for $t \geq 0$. Next, we define the $K$-greedy policy, which gives the indices of $K$ customers who are routed to the offline queue.

**Definition 2.** ($K$-greedy policy). The $K$-greedy policy chooses $K$ customers to be routed to the offline queue as follows:

$$i_k^* = \text{argmax}_{i \in J_k^{C-1}} \left[ H(Q_k^{k-1}) - H(\Phi(Q_k^{k-1}, \{i\})) \right], \ k = 1, \ldots, K, \tag{5}$$

where $i_k^*$ denotes the index of the $k^{th}$ customer routed to the offline queue, and $J_k = J_{k-1} \cup \{i_k^*\}$, $Q_k^0 = \Phi(Q_k^{k-1}, \{i_k^*\})$ with $J_0 = \emptyset$ and $Q_1^0 = Q_1^\emptyset$.

The $K$-greedy policy first picks customer $i_1^*$ whose removal results in the largest holding cost savings. Removing customer $i_1^*$ results in the new online queue length process $Q_1^1$. The policy repeats this process until $K$ customers are removed from the queue. The following lemma shows that the $K$-greedy policy maximizes the total holding cost savings among all policies that remove $K$ customers; see Lemma 6 of Spencer et al. (2014) and its proof for a proof of this result.

**Lemma 1.** For any $K \leq n$ and $\mathcal{I} \subseteq \{1, 2, \ldots, n\}$ with $|\mathcal{I}| = K$, the following holds:

$$H(Q_1^K) - H(Q_1^K) \geq H(Q_1^0) - H(\Phi(Q_1^0, \mathcal{I})).$$

That is, $H(Q_1^K) \leq H(\Phi(Q_1^0, \mathcal{I}))$.

Thus, the problem simplifies to finding the optimal greedy policy. That is,

$$\min_{0 \leq K \leq n} pK + hH(Q_1^K). \tag{6}$$

\footnote{If there are multiple indices that achieve the maximum, we pick the smallest index.}
Finding the optimal value of $K$ boils down to comparing the marginal cost $p$ of routing a customer to the offline queue and the marginal savings in the holding costs, i.e. $H(Q_1^{K-1}) - H(Q_1^K)$. The marginal savings $H(Q_1^{K-1}) - H(Q_1^K)$ include two parts: The waiting cost customer $i^*_K$ would have incurred if he were not routed to the offline queue and the negative externalities he imposes on others in the online queue. To characterize the potential marginal savings, one needs to compute the (intermediate auxiliary) queue length processes $Q_1^1, \ldots, Q_1^K$. Surprisingly, we next provide an explicit construction of the optimal greedy policy that uses only the initial online queue-length process $Q_1^0(t)$, or equivalently, the total queue length for the system $Q(t)$. This crucially relieves us from computing the (intermediate auxiliary) queue length processes $Q_1^1, \ldots, Q_1^K$.

Given a callback policy, the evolution of the online queue length $Q_1(t)$ is independent of the service discipline used to prioritize customers within the online queue provided it is work conserving. To repeat, this is because the service times are associated with the server not with the customers. Moreover, as observed by Hassin (1985), under the last-come-first-served (LCFS) service discipline, a customer joining the (online) queue does not impose any externalities on others. Rather, the LCFS service discipline ensures that he internalizes all such externalities that would have been incurred under the FCFS service discipline. Therefore, in what follows we study a virtual system that serves the online queue in a LCFS fashion, which will clearly have the same performance as the original system. Then for this system, we identify the optimal callback policy and argue that it is also optimal for the original system which serves the online queue in a FCFS fashion.

As a preliminary to characterizing the optimal callback policy in this virtual system, the next lemma characterizes the waiting times of the customers in the system when there are no callback offers. To this end, recall that the time $s_i$ (for $i \geq 1$), defined in Equation (4), denotes the next time (after customer $i$ joins the system) when the total number of customers in the system $Q(t)$ falls back to the level just before customer $i$’s arrival.

**Lemma 2.** In the system with no callback offers, customer $i$ completes his service at time $s_i$ under the LCFS discipline. Thus, the waiting time of customer $i$, denoted by $w_i$, is given by $w_i = s_i - \tau_i$, for $i = 1, \ldots, n$.

This result has long been recognized in the queueing theory literature; see for example, page 321 of Gross and Harris (1974). Recall that $H(Q)$ denotes the total area under $Q(t)$ during the busy
period. It is easy to show that \( H(Q_1^0) = \sum_{i=1}^{n} w_i \), which can be proved geometrically by exploiting the equivalence of tracing the area either vertically or horizontally in calculating it.

The following lemma is the key to proving our result and shows that under the \( K \)-greedy policy, the waiting times of the customers remaining in the online queue are the same as those in the original system; see Appendix B.1 for its proof. The following notation is needed to state it: Consider the \( k \)th iteration of the \( K \)-greedy policy \((k \leq K)\) and the resulting online queue length process \( Q_k^1 \). Let \( w_k^i \) denote the waiting time of customer \( i \in J_k^C \), i.e. those customers who are still in the online queue after the \( k \)th iteration. Then the holding cost for the online queue \( Q_k^1 \) after the \( k \)th iteration is given as follows:

\[
H(Q_k^1) = \sum_{i \in J_k^C} w_k^i \quad \text{for} \quad k = 1, \ldots, n.
\]

(7)

**Lemma 3.** Given \( K \leq n \), we have that \( w_k^i = w_i \) for \( i \in J_k^C \) and \( k = 0, 1, \ldots, K \).

We conclude from Lemma 3 that choosing \( K^* = |I^*| \), where \( I^* = \{ i = 1, 2, \ldots, n : w_i > p/h \} \), the \( K^* \)-greedy policy is optimal. To see this, note from Lemma 3 and Equation (7) that

\[
H(Q_k^1) = \sum_{i \in J_k^C} w_i \quad \text{for} \quad k = 1, \ldots, n.
\]

Substituting this into Equation (5) of Definition 2 gives

\[
i_k^* = \text{argmax}_{i \in J_k^C} w_i \quad \text{for} \quad k = 1, \ldots, K.
\]

Thus, the \( K \)-greedy policy chooses the customers with the largest \( K \) waiting times \( w_i \) in the original system to route to the offline queue. Given the formulation in Equation (6), i.e. the penalty \( p \) for removing customer \( i \) from the online queue and the holding cost savings of \( hw_i \), the optimal policy should remove customer \( i \) as long as his waiting time \( w_i \) exceeds \( p/h \). This implies that the customers in the set \( I^* \) should be sent to the offline queue.

A few important observations are in order. First, all that is relevant to determine the set \( I^* \) is the set of waiting times \( \{ w_i : i = 1, \ldots, n \} \) which are determined by \( w_i = s_i - \tau_i \) for \( i = 1, \ldots, n \), as in Equations (4) and Lemma 2. Second, the calculation of \( w_i \) depends only on the total queue length process \( \{ Q(t) : t \geq 0 \} \) (see Equation (4)), which is invariant to the routing, priority and service discipline (within each queue) decisions. Third, what matters ultimately is that the waiting times of customers who remain in the system are less than or equal to \( p/h \). These observations prove that not only the order of removal from the online queue does not matter (in particular, the removal decision can be made upon a customer’s arrival) but also the optimal set \( I^* \) of customers routed to the offline queue is precisely the same as the one under the \( p/h \)-lookahead policy. To repeat, these
follow because the order of removal from the online queue does not impact the evolution of the total queue length process, and hence, \( w_i = s_i - \tau_i \) for \( i = 1, \ldots, n \), and that customer \( i \) should be removed if and only if \( w_i > p/h \). The last observation is that the evolution of the resulting online queue length process \( Q_1(\cdot) \) is invariant to the service discipline used to prioritize customers within the online queue. To be specific, if we fix a routing policy \( I \), the evolution of the resulting online queue length process \( Q_1(\cdot) \) (using LCFS) is identical to that of the original system, which serves the customers within the online queue in a FCFS fashion. This is because the service times are associated with the server not with the customers as discussed above. Thus, with a fixed routing policy \( I \), the average cost (see Equation (3)) of the system that uses LCFS is the same as the average cost of the original system. Since the \( p/h \)-lookahead policy minimizes the average cost of the system that uses LCFS among the complete foresight policies, it also minimizes the average cost of the original system. This completes the proof of Proposition 1.

4.2 Proof of Theorem 1

This subsection is dedicated to proving Theorem 1 by leveraging Proposition 1. The proofs of all lemmas are provided in Appendix B.2. In what follows, we consider two auxiliary systems and the lookahead policies for them. The results in Section 4.1 help establish the optimality of the lookahead policy for the first auxiliary system. The second auxiliary system bridges the original system and the first auxiliary system. We show that in both auxiliary systems, the same customers join the online queue under the lookahead policy. Thus, they incur the same cost. This implies the optimality of the lookahead policy in the second auxiliary system. Finally, we show that the lookahead policy in the second auxiliary system and the original system leads to identical sample paths for the queue length processes. Combining these observations, we conclude that the three lookahead policies (described below) will result in identical sample paths for the online and offline queue length processes in the three systems. Hence, they have the same cost, establishing the optimality of the lookahead policy in the original system. Figure 2 summarizes the connection between the auxiliary systems and the roadmap of the proof.

We attach a ‘\^’ (‘\~’) to various quantities of interest in the first (second) auxiliary system. Consider the first auxiliary system with three queues: Queue \( r \), queue \( n \) and queue 2, where \( \hat{Q}_r(t) \), \( \hat{Q}_n(t) \) and \( \hat{Q}_2(t) \) denote the numbers of customers in each queue at time \( t \), respectively. We assume
that the sets $\mathcal{A}$ and $\mathcal{R}$ are known to the call center manager ex ante in the first auxiliary system; and all customers in set $\mathcal{R}$ are routed to queue $r$. In contrast, a customer in set $\mathcal{A}$ joins queue $n$ if he does not receive the callback offer, whereas he joins queue 2 (the offline queue) if he receives the callback offer. (Recall that all customers in set $\mathcal{A}$ accept the callback offer). Note that $\tilde{Q}_1(t) = \tilde{Q}_r(t) + \tilde{Q}_n(t), \ t \geq 0$. (8)

In this system, customers are served under a work-conserving static priority rule (with preemption) where the customers in the online queue has the priority over customers in the offline queue. Within the online queue, i.e. queue 1, queue $r$ has the strict priority over queue $n$. Therefore, customers in set $\mathcal{R}$, who are sent to queue $r$, do not “see” the rest of the system. In other words, the evolution of queue $r$ is independent of the routing policy $I \subseteq \mathcal{A}$, because its arrival process $\{\tau_i : i \in \mathcal{R}\}$ is independent of the routing policy and queue $r$ enjoys the highest priority among all queues. Thus, the call center manager’s decision reduces to determining who in set $\mathcal{A}$ receive the callback option and are routed to the offline queue. Let $\hat{Q}_a(t)$ denote the total number of customers in set $\mathcal{A}$ (combining queue $n$ and queue 2) in the system at time $t$. That is, $\hat{Q}_a(t) = \hat{Q}_n(t) + \hat{Q}_2(t)$ for $t \geq 0$. Because both the original system and the first auxiliary system operate under work-conserving policies, the total numbers of customers are the same in both system, i.e.

$Q(t) = \hat{Q}_r(t) + \hat{Q}_a(t), \ t \geq 0$. (9)

Next we define the first auxiliary w-lookahead policy and show that it is optimal for the first auxiliary system. To facilitate that definition, we define $\tilde{s}_i$ as follows:

$\tilde{s}_i = \inf\{t \geq \tau_i : \hat{Q}_a(t) = \hat{Q}_a(\tau_i -)\}$. (10)

**Definition 3.** The first auxiliary $w$-lookahead policy offers the callback option to customer $i$ if $i \in \mathcal{A}$ and $\tilde{s}_i \geq \tau_i + w.$
Let \( \tilde{I} \) denote the set of customers who are offered the callback option, i.e. \( \tilde{I} = \{ i \in A : \tilde{s}_i \geq \tau_i + w \} \) in the first auxiliary system. Since customers in set \( \tilde{I} \) belong to the set \( A \), it is also the set of customers who are routed to the offline queue. This policy can be viewed as applying the lookahead policy in Equation (4) to the process \( \tilde{Q}_a(t) : t \geq 0 \) instead of the process \( Q(t) : t \geq 0 \).

Combining this with the fact that queue \( r \) essentially does not “see” the rest of the system because it enjoys the highest (preemptive) priority makes it more-or-less obvious that the first auxiliary \( p/h \)-lookahead policy is optimal for the first auxiliary system as proved in the next lemma.

**Lemma 4.** The first auxiliary \( p/h \)-lookahead policy is optimal for the first auxiliary system.

Next lemma provides an additional characterization of the random time \( \tilde{s}_i \) (for \( i \in A \)).

**Lemma 5.** The following holds: \( \tilde{s}_i = \inf \{ t \geq \tau_i : Q(t) = Q(\tau_i-) - \tilde{Q}_r(\tau_i-) \} \) for \( i \in A \).

Combining Lemmas 4 and 5 with Definitions 1 and 3 shows that the \( p/h \)-lookahead policy is optimal for the first auxiliary system as stated in the next corollary.

**Corollary 1.** The \( p/h \)-lookahead policy is optimal for the first auxiliary system.

In addition, we provide some useful properties of the \( w \)-lookahead policy in the first auxiliary system in Lemma 6.

**Lemma 6.** Consider the first auxiliary system under the \( p/h \)-lookahead policy. If customer \( i \in A \) is not offered the callback option, i.e. he is routed to the online queue, then any other customer \( j \in A \) arriving after him but before his departure, i.e. \( \tau_j \in (\tau_i, \tilde{s}_i) \), is not offered the callback option either, i.e. he is also routed to the online queue. Moreover, queue \( r \) is empty when customer \( i \) departs the system, i.e. \( \tilde{Q}_r(\tilde{s}_i) = 0 \).

The call center manager knows who will accept and reject the callback option before offering it in the first auxiliary system. Next, we consider the second auxiliary system which relaxes this assumption. That is, the call center manager does not know the set \( A \) a priori in the second auxiliary system. The second auxiliary system also has three queues: Queue \( r \), queue \( n \) and queue 2 (the offline queue), and the number of customers in them at time \( t \) are denoted by \( \tilde{Q}_r(t) \), \( \tilde{Q}_n(t) \) and \( \tilde{Q}_2(t) \), respectively. The system operates under a work-conserving static priority rule (with preemption) where queue \( r \) has the highest priority, queue \( n \) has the second highest priority and
queue 2 (the offline queue) has the lowest priority. Moreover, customers within queue \( r \) are served in a FCFS fashion, whereas customers within queue \( n \) are served in a LCFS fashion.

The second auxiliary system serves as a bridge between the original system and the first auxiliary system. On the one hand, the key difference between the first and the second auxiliary systems is how the customers in set \( R \) are routed. In the second auxiliary system, they join queue \( r \) if and only if they are offered the callback option. In particular, they join queue \( n \) if they are not offered the callback option whereas they all join queue \( r \) in the first auxiliary system. On the other hand, the major difference between the second auxiliary system and the original system is the service discipline used to prioritize customers within the online queue. In the second auxiliary system, customers in queue \( r \) have (preemptive) priority over those in queue \( n \), whereas all customers in the online queue (which consists of queue \( n \) and queue \( r \)) are served in FCFS fashion in the original system. Note that

\[
\hat{Q}_1(t) = \hat{Q}_r(t) + \hat{Q}_n(t), \ t \geq 0. \tag{11}
\]

In addition, we define the random time \( \hat{s}_i \) as follows:

\[
\hat{s}_i = \inf \{ t \geq \tau_i : Q(t) = Q(\tau_i) - \hat{Q}_r(\tau_i) \}, \ i = 1, 2, \ldots, n. \tag{12}
\]

Next, we define the second auxiliary \( w \)-lookahead policy.

**Definition 4.** The second auxiliary \( w \)-lookahead policy offers the callback option to customer \( i \) if \( \hat{s}_i \geq \tau_i + w \).

Letting \( \hat{I} \) denote the set of customers who are offered the callback option, i.e.

\[
\hat{I} = \{ i = 1, 2, \ldots, n : \hat{s}_i \geq w + \tau_i \}, \tag{13}
\]

the customers who join the offline queue (queue 2) are given by \( \hat{I} \cap A \). All customers in \( \hat{I} \cap R \) join queue \( r \), whereas customer \( i \notin \hat{I} \) joins queue \( n \). In particular, a customer \( i \in R \) may join queue \( n \) (in contrast to the first auxiliary system) if he is not offered the callback option. As a preliminary, we provide useful properties of the second auxiliary \( w \)-lookahead policy in Lemma 7.

**Lemma 7.** Consider the second auxiliary system under the second auxiliary \( p/h \)-lookahead policy. The following hold:

(i) If customer \( i \) is not offered the callback option (and hence joins the online queue), i.e. \( i \notin \hat{I} \),
then any customer \( j \) arriving after him but before his departure, i.e. \( \tau_j \in (\tau_i, \hat{s}_i) \), is not offered the callback option either (and hence joins the online queue), i.e. \( j \notin \hat{I} \).

(ii) If customer \( i \) is not offered the callback option (and hence joins the online queue), i.e. \( i \notin \hat{I} \), then he leaves the system at time \( \hat{s}_i \). Moreover, queue \( r \) is empty at his departure time, i.e. \( \hat{Q}_r(\hat{s}_i) = 0 \) and \( \hat{Q}_n(\hat{s}_i) = \hat{Q}_n(\tau_i-) \).

(iii) If customer \( i \) is offered the callback option, i.e. \( i \in \hat{I} \), then queue \( n \) is empty upon his arrival, i.e. \( \hat{Q}_n(\tau_i-) = \hat{Q}_n(\tau_i) = 0 \).

Property (i) of Lemma 7 shows that if a customer is not offered the callback option, the call center manager does not offer it to any other customer arriving after him until his departure. In addition, property (ii) implies that when this customer leaves the system, queue \( r \) is empty and the number of customers in queue \( n \) (\( \hat{Q}_n \)) falls back to the level just before his arrival. Lastly, property (iii) shows that if the system offers the callback option, queue \( n \) must be empty.

These three observations imply that the system alternates between the callback and no-callback episodes. In a callback episode, the call center manager offers the callback option to every arriving customer. In contrast, she does not offer the callback option to any arriving customer during a no-callback episode. When the system is in a callback episode, queue \( n \) is empty (i.e. \( \hat{Q}_n = 0 \)) and all customers currently in the online queue are those who rejected the callback option; see property (iii) of Lemma 7. Thus, we arrive at the following key observation: If both queues \( n \) and \( r \) are non-empty (i.e. \( \hat{Q}_n > 0 \) and \( \hat{Q}_r > 0 \)), then the system is in the no-callback episode. Moreover, all customers in queue \( r \) currently must have arrived during the callback episode prior to the current no-callback episode. In other words, all customers in queue \( r \) currently have arrived before those in queue \( n \). That is, the strict priority rule is never really enforced to reverse what would have happened under FCFS service discipline (within the online queue). Therefore, although the second auxiliary system gives priority to queue \( r \) over queue \( n \), the evolution of the queue length processes would have been the same if the online queue (the combination of queues \( r \) and \( n \)) was served in a FCFS fashion (as done in the original system). This is because customers in queue \( r \) arrived before customers in queue \( n \) due to the aforementioned structure of the callback and no-callback episodes. This observation is the main intuition behind the equivalence of the second auxiliary system and the original system, in which the online queue is served in a FCFS fashion.
To elaborate further on the evolution of the second auxiliary system (under the second auxiliary lookahead policy), note that once the system is in the no-callback episode, it stays in the no-callback episode until the online queue is empty. That is, once the system is in the no-callback episode, no arriving customer is offered the callback option until the online queue is empty; see properties (i) and (ii) of Lemma 7. If the current busy period of the online queue starts with the no-callback episode, the no-callback episode lasts until the busy period of the online queue ends. If the current busy period of the online queue starts with the callback episode, the callback episode either lasts until the busy period of the online queue ends or switches to a no-callback episode, which lasts until the end of the busy period of the online queue. To summarize, there are three possible combinations of callback and no-callback episodes for a busy period of the online queue: just one no-callback episode, just one callback episode, or a callback episode followed by a no-callback episode.

The following lemma formally establishes the equivalence of the second auxiliary system and the original system.

**Lemma 8.** The evolutions of the queue length processes in the original system and the second auxiliary system are the same. That is, $Q_r(t) = \hat{Q}_r(t)$, $Q_n(t) = \hat{Q}_n(t)$ and $Q_2(t) = \hat{Q}_2(t)$ for $t \geq 0$.

Finally, the following lemma shows that the same set of customers are routed to the offline queue in the two auxiliary systems. Hence, the two auxiliary systems incur the same cost.

**Lemma 9.** The same set of customers are routed to the offline queue in the two auxiliary systems, i.e. $\tilde{I} = \hat{I} \cap A = \tilde{I} \cap A$. Consequently, we have that $\tilde{Q}_1(t) = \hat{Q}_1(t)$ and $\tilde{Q}_2(t) = \hat{Q}_2(t)$ for $t \geq 0$.

To build some intuition towards this result, we note from Lemma 5 and Equation (12) that showing $\hat{Q}_r(\tau_i^-) = \tilde{Q}_r(\tau_i^-)$ for every customer $i$ receiving the callback option (in either system) should suffice. Consider the second auxiliary system and note from property (iii) of Lemma 7 that for any customer $i$ receiving the callback option, we have that $\hat{Q}_n(\tau_i) = 0$. On the one hand, this implies that all customers in set $\mathcal{R}$, who are currently in the (second auxiliary) system, are in queue $r$. On the other hand, all customers in set $\mathcal{R}$, are always in queue $r$ in the first auxiliary system by construction. This suggests that $\hat{Q}_r(\tau_i^-) = \tilde{Q}_r(\tau_i^-)$. Admittedly, this argument is heuristic and incomplete. Nevertheless, Lemma 9 builds on this intuition and is proved by induction.

Lemmas 8-9 show that fixing a lookahead window $w > 0$, the three lookahead policies defined in this subsection results in identical sample paths for the online and offline queue length processes.
Recall from Lemma 4 that the first auxiliary \( p/h \)-lookahead policy minimizes the objective in Equation (1). Thus, so does the \( p/h \)-lookahead policy in the original system, which completes the proof of Theorem 1. We provide an example in Appendix B.3 to illustrate the difference between and the connections of the dynamics of the two auxiliary systems and the original system.

5 A Non-anticipating Policy via the Fluid Model Analysis

The analysis in Section 4 assumes that the exact times of the future arrivals and service completions are known. Although this assumption usually does not hold in practice, the lookahead policy derived from this analysis provides useful insights for constructing an effective non-anticipating control policy. To this end, this section considers a fluid model and derives one such policy by interpreting the lookahead policy derived in Section 4. In what follows, we prove that this policy is optimal for the fluid model considered. Furthermore, we specialize this policy to the customer arrival model proposed in Glynn et al. (2018), i.e. the CIR model for the arrival rate process. In this case, we show that the fluid lookahead policy reduces to a simple non-anticipating policy, referred to as the line policy, which is easily implementable in and effective for the original system.

5.1 The Lookahead Policy for the Fluid Model

This section formulates and solves a fluid model of the call center manager’s problem. The fluid model solution can then be interpreted in the context of the original problem, providing a policy implementable in the original system. Such approaches have been taken in the literature previously. For example, Maglaras (2000) illustrates how one implements a fluid model solution as a feedback control in the original system using the so-called discrete-review policies\(^6\). At each review point, the call center manager solves a fluid-model control problem taking the current state as the initial condition of the fluid model and implements a plan based on that solution until the next review point. We follow an approach that is conceptually similar. Namely, when a customer arrives, the call center manager takes the current system state as the initial condition for the fluid model and solves the resulting fluid-model control problem. She then decides whether to offer the callback option to the incoming customer based on the solution of the fluid-model control problem.

\(^6\) Also see Harrison (1996, 1998) and Ata and Kumar (2005) for a similar approach that uses discrete-review policies for interpreting diffusion models in the context of queueing networks.
To be more specific, the call center manager solves the fluid control problem starting at time $t_0$ in order to decide whether to offer the callback option to the customer arriving at time $t_0$. That solution yields an optimal callback offer rate function, denoted by $\{\bar{\alpha}(t) : t \geq t_0\}$, where $\bar{\alpha}(t)$ denotes the rate of arriving customers who receive the callback option at time $t$. Then the call center manager offers the callback option to the customers arriving at time $t_0$ with probability $\bar{\alpha}(t_0)/\lambda(t_0)$. Similarly, for future arrivals to the system, she re-solves the fluid-model control problem using the updated system state as the initial condition of the fluid model. The decision of whether to offer the callback option is made based on this new solution.

The rest of this subsection formulates an optimal control problem for a fluid model starting at $t_0$ with a given initial state and solves it. To this end, we first advance a fluid model of the call center manager's problem. We then propose a policy, called the fluid lookahead policy, by interpreting the lookahead policy studied in Section 4 for the fluid model. This facilitates a simple, implementable policy for the original system building on the insights gleaned from the complete foresight analysis. Lastly, we show that the fluid lookahead policy indeed is optimal for the optimal control problem associated with the fluid model.

We let $\bar{\lambda}(t)$ denote the (deterministic) arrival rate at time $t \geq t_0$ in the fluid model. The (deterministic) arrival rate process $\{\bar{\lambda}(t), t \geq t_0\}$ can be interpreted as the arrival rate forecast given the available information at time $t_0$. Throughout the paper, we assume that the current arrival rate is observed whereas it can only be estimated from the observed customer arrivals$^7$. Recall that the service rate is $\mu$. Then letting $\bar{Q}(t)$ denote the total fluid of the customers in the system at time $t \geq t_0$, it is characterized as follows: $\bar{Q}(t_0) = Q(t_0)$ and for $t \geq t_0$,

$$\bar{Q}'(t) = \begin{cases} \bar{\lambda}(t) - \mu, & \text{if } \bar{Q}(t) > 0, \\ (\bar{\lambda}(t) - \mu)^+ & \text{if } \bar{Q}(t) = 0. \end{cases} \tag{14}$$

As in Section 4, we study the system during the current busy period. That is, letting $\bar{T}$ denote the end$^8$ of the current busy period, i.e. $\bar{T} = \inf\{t > t_0 : \bar{Q}(t) = 0\}$, we focus attention on the system behavior during $[t_0, \bar{T}]$. Within the busy period $[t_0, \bar{T}]$, we can ignore the second case on

---

$^7$As a robustness check, we consider the performance of the line policy where the current arrival rate is estimated by the average number of arrivals over the past five minutes. As one would expect for large call volumes, this leads to a close approximation of the original findings for the case of observable arrival rate in a simulate study (available from the authors). Indeed, the maximum difference in performance was 0.3%.

$^8$Note that $\bar{T}$ depends on the initial number $\bar{Q}(t_0)$ of customers in the system.
the right-hand side of Equation (14). Integrating both sides of Equation (14) gives the following:

\[
\bar{Q}(t) = \bar{Q}(t_0) + \int_{t_0}^{t} \bar{\lambda}(s) \, ds - \mu(t - t_0), \quad t \in [t_0, \bar{T}].
\]  

(15)

Another important quantity of interest is the dynamics of the online queue, which is governed by the call center manager’s callback policy. To facilitate the analysis to follow, let \( \bar{q}(t) \) denote the total fluid in the online queue and \( \bar{\alpha}(t) \) denote the rate of arriving customers who are offered the callback option at time \( t \) for \( t \in [t_0, \bar{T}] \). The call center manager’s callback policy is then denoted by \( \{\bar{\alpha}(t) : t \in [t_0, \bar{T}]\} \). We allow customers to decline the callback option and let \( f_t(\bar{\alpha}) \in [0, \bar{\alpha}] \) denote the actual rate of customers accepting the offer if the callback offer rate at time \( t \) is \( \bar{\alpha} \).

We assume that the function \( f_t(\cdot) \) is continuous and strictly increasing, so its inverse function, denoted by \( f_t^{-1}(\cdot) \), exists. In what follows, we consider a relaxed formulation and allow the call center manager to choose the servicer rate\(^9\) \( \bar{\mu}(t) \) for serving the online queue at time \( t \in [t_0, \bar{T}] \).

This is done merely for analytical convenience. As the reader will see below, the optimal policy sets \( \bar{\mu}(t) = \mu \) whenever \( \bar{q}(t) > 0 \), i.e. the server gives strict priority to the online queue as assumed in the preceding sections. Therefore, given the call center manager’s callback and service policies, denoted by \( \bar{\alpha}(\cdot) \) and \( \bar{\mu}(\cdot) \), respectively, the dynamics of \( \bar{q}(t) \) is characterized as follows:

\[
\bar{q}'(t) = \bar{\lambda}(t) - f_t(\bar{\alpha}(t)) - \bar{\mu}(t), \quad t \in [t_0, \bar{T}],
\]  

(16)

with \( \bar{q}(t_0) = q_0 = Q_1(t_0) \). Since the online queue length \( \bar{q}(t) \) cannot be negative, we impose the following state space constraint for the online queue length:

\[
\bar{q}(t) \geq 0, \quad t \in [t_0, \bar{T}].
\]  

(17)

Let \( \pi = \{(\bar{\alpha}(t), \bar{\mu}(t)) \in [t_0, \bar{T}]\} \) denote an admissible policy, consisting of both the callback and the service policies, such that \( \bar{\alpha}(t) \in [0, \bar{\lambda}(t)] \) and \( \bar{\mu}(t) \in [0, \mu] \) for \( t \in [t_0, \bar{T}] \) and that both \( \bar{\alpha}(\cdot) \) and \( \bar{\mu}(\cdot) \) are piecewise continuous. Let \( \Pi \) denote the set of all such admissible policies. Thus, the

\( \text{The rest of the service effort } \mu - \bar{\mu}(t) \text{ is devoted to serving the offline queue whenever it is nonzero.} \)
optimal control problem \((P)\) associated with the fluid model is defined as follows\(^{10}\):

\[
\min_{\pi \in \Pi} \int_{t_0}^{T} (h\bar{q}(t) + pf_t(\bar{\alpha}(t))) \, dt
\]

\[
(P) \quad \text{subject to (16) – (17), } \bar{q}(t_0) = \bar{q}_0 \text{ and } \bar{q}(T) = 0.
\]

The optimal control problem \((P)\) seeks an admissible policy that minimizes the sum of the holding cost of the online queue and the callback penalty incurred during the current busy period \([t_0, T]\) of the system. In particular, we are interested in \(\bar{\alpha}(t_0)\) for the given initial state.

Next, we introduce the fluid lookahead policy, denoted by \(\{\alpha(t), t \in [t_0, T]\}\), by interpreting the callback policy of Section 4 in the context of the fluid model. Recall that the \(p/h\)-lookahead policy studied in Section 4 offers the callback option to customer \(i\), who arrives at time \(t_0\), if \(s_i - t_0 > p/h\) where the time \(s_i\) is given in Equation (2). In the fluid model, we replace \(s_i\) with \(\bar{t}_r\), where

\[
\bar{t}_r = \inf \left\{ t > t_0 : \bar{Q}(t) \leq \bar{Q}(t_0) - \bar{q}(t_0) \right\}.
\]

The definition of \(\bar{t}_r\) differs from that of \(s_i\) in two ways: First, we replace the term \(Q_r(t_0)\) with \(\bar{q}(t_0)\). Second, we replace the equality \(Q(t) = Q(t_0) - Q_1(t_0)\) with \(\bar{Q}(t) \leq \bar{Q}(t_0) - q(t_0)\)\(^{11}\).

Substituting (15) into (18) and rearranging the terms gives:

\[
\bar{t}_r = \inf \left\{ t > t_0 : \int_{t_0}^{t} \bar{\lambda}(s) \, ds - \mu(t - t_0) + \bar{q}(t_0) \leq 0 \right\}.
\]

The fluid lookahead policy suggests that a customer arriving at time \(t_0\) should be offered the callback option if (and only) if \(\bar{t}_r - t_0 > p/h\). To be specific, if \(\bar{t}_r - t_0 > p/h\), then\(^{12}\)

\[
\alpha(t_0) = \begin{cases} 
\bar{\lambda}(t_0), & \text{if } \bar{q}(t_0) > 0, \\
\min(f_t^{-1}(\bar{\lambda}(t_0) - \mu), \bar{\lambda}(t_0)), & \text{if } \bar{q}(t_0) = 0.
\end{cases}
\]

\(^{10}\)The terminal condition \(\bar{q}(T) = 0\) holds automatically for all non-idling policies. We use this formulation because it fits into the standard framework in Seierstad and Sydsæter (2002).

\(^{11}\)The heuristic motivation for the former change is that if there are any customers during a callback episode, those are the ones that rejected the callback offer. The latter change is made to address the case where the online queue may be empty and the server is primarily serving the offline queue initially. The change ensures \(\bar{t}_r = t_0\) in that case.

\(^{12}\)Note that it is never economical to route the arriving customers to the offline queue if they can be served instantly and the online queue is empty. If the call center manager chooses to route these customers to the offline queue, the system incurs the penalty, but receives no saving in the holding cost. Thus, in the proposed policy, if the current arrival rate \(\bar{\lambda}(t_0)\) is high enough, the system only routes the excess arriving customers \(\bar{\lambda}(t_0) - \mu\) to the offline queue by offering the callback option to customers at rate \(f_t^{-1}(\bar{\lambda}(t_0) - \mu)\) when the online queue is empty. In other words, the call center manager offers the callback option to customers at the rate \(f_t^{-1}(\bar{\lambda}(t_0) - \mu)\) if the arriving rate \(\bar{\lambda}(t_0)\) is high enough.
Otherwise, if $\tilde{t}^* - t_0 \leq p/h$, all arriving customers at time $t_0$ are routed to the online queue, i.e. $\alpha(t_0) = 0$. We next extend the definition of the fluid lookahead policy to all $t \in [t_0, \tilde{T}]$. To facilitate the definition, we define an auxiliary function $\Gamma(\cdot, \cdot)$ as follows:

$$\Gamma(t_1, t_2) = \int_{t_1}^{t_2} (\mu - \lambda(u)) \, du, \quad t_0 \leq t_1 \leq t_2 \leq \tilde{T}. \quad (20)$$

**Definition 5.** Under the fluid lookahead policy, the callback offer rate $\alpha(t)$ for $t \in [t_0, \tilde{T}]$ is given as follows: Fix a time $t \in [t_0, \tilde{T} - p/h]$. If there exists $s \in (0, p/h]$ such that $\bar{q}(t) - \Gamma(t, t + s) \leq 0$, then the system is in the no-callback episode, i.e. $\alpha(t) = 0$. Otherwise, if $\bar{q}(t) - \Gamma(t, t + s) > 0$ for $s \in (0, p/h]$, then the system is in the callback episode, i.e.

$$\alpha(t) = \begin{cases} 
\lambda(t), & \text{if } \bar{q}(t) > 0, \\
\min(f_t^{-1}(\lambda(t) - \mu), \bar{\lambda}(t)), & \text{if } \bar{q}(t) = 0.
\end{cases} \quad (21)$$

In addition, the system does not offer the callback option during $[\tilde{T} - p/h, \tilde{T}]$\textsuperscript{13}, i.e. $\alpha(t) = 0$ for $t \in [\tilde{T} - p/h, \tilde{T}]$.

In addition, we assume that the associated service policy, denoted by $\{\mu(t) : t \geq t_0\}$ gives static priority to the online queue, i.e. for $t \in [t_0, \tilde{T}]$,

$$\mu(t) = \begin{cases} 
\mu, & \text{if } \bar{q}(t) > 0, \\
\min(\mu, \bar{\lambda}(t)), & \text{if } \bar{q}(t) = 0.
\end{cases} \quad (22)$$

The fluid lookahead policy provides an easily implementable rule to determine the callback offer decision for each incoming customer. We end this section by showing that the fluid lookahead policy indeed is an optimal solution to the optimal control problem (P) associated with the fluid model under the following technical assumptions: (i) The arrival rate process $\bar{\lambda}(t)$ (for $t \in [t_0, \tilde{T}]$) is a continuous function; (ii) The arrival rate $\bar{\lambda}(t) > 0$ and $\bar{\lambda}(t) - \mu$ is not identically equal to zero or $f_t(\bar{\lambda}(t))$ on any interval $(a, b) \subseteq [t_0, \tilde{T}]$\textsuperscript{14}.

**Proposition 2.** The fluid lookahead policy $\{(\alpha(t), \mu(t)) : t \in [t_0, \tilde{T}]\}$ solves the optimal control problem (P).

\textsuperscript{13}By the definition of $\tilde{T}$, the entire system is empty at $\tilde{T}$. Thus, for $t \in [\tilde{T} - p/h, \tilde{T}]$, $\bar{q}(t) - \Gamma(t, \tilde{T}) \leq 0$ holds automatically. In other words, the system does not offer any callback option to customers arriving during $[\tilde{T} - p/h, \tilde{T}]$.

\textsuperscript{14}This assumption simplifies significantly the characterization of the dynamics of the online queue provided in this section. However, we believe that the optimality result of the fluid lookahead policy still holds, though the characterization and the proof will be more involved.
Appendix C provides the proof of Proposition 2, which we briefly outline next. First, we provide a complete characterization of the system dynamics under the fluid lookahead policy. This generalizes the structure of the callback versus no-callback episodes mentioned in Section 4. To establish the optimality of the fluid lookahead policy, we next drive a set of sufficient conditions that the Hamiltonian of the optimal control problem (P) needs to satisfy. The sufficient conditions follow from the framework provided in Chapter 5 of Seierstad and Sydsaeter (1987). We then explicitly construct the associated adjoint function of the optimal control problem using the aforementioned characterization of the system dynamics. Lastly, we verify that the Hamiltonian of the optimal control problem (P) and the constructed adjoint function satisfy the sufficient conditions under the fluid lookahead policy.

5.2 The Line Policy: The Fluid Lookahead Policy for the CIR process

This subsection specializes the fluid lookahead policy to the specific arrival model proposed in Glynn et al. (2018). In this case, the fluid lookahead policy has a simple form such that the system offers the callback option if a linear combination of the current online queue length and the current arrival rate exceeds a threshold. This policy is referred to as the line policy.

Glynn et al. (2018) models the arrival rate process as a Cox-Ingersoll-Ross (CIR) process, and provides empirical support for using it as a model of arrivals to a call center. Following Glynn et al. (2018)\textsuperscript{15}, we assume that the arrival rate process \{\lambda(t) : t \geq 0\} follows a CIR process. In particular, it satisfies the following stochastic differential equation:

\[ d\lambda(t) = a(b - \lambda(t)) \, dt + \sigma \sqrt{\lambda(t)} \, dW(t), \quad t \geq 0, \]

where \(a, b\) and \(\sigma\) are positive constants and \(\{W(t) : t \geq 0\}\) is the standard Brownian motion. The arrival process \(A(\cdot)\) is then a Poisson process with the stochastic intensity process \{\lambda(t) : t \geq 0\} given in Equation (23); see page 23 of Bremaud (1981) or page 476 of Jeanblanc et al. (2009) for the formal definition. Note that \(b\) is the long-run average arrival rate, i.e. \(\lim_{t \to \infty} E[A(t)]/t = \lim_{t \to \infty} E[\lambda(t)] = b\).

\textsuperscript{15}For simplicity, our model of the arrival rate process \{\lambda(t) : t \geq 0\} ignores the time-of-day effect. Although it is straightforward to incorporate the time-of-day effect into the CIR model for our (numerical) analysis, we do not see that as central to the paper, because we are mainly concerned with managing the variations in the arrival rate on the mesoscopic time scale. Moreover, as noted above, our results on the lookahead policies hold for general arrival processes, subsuming those with the time-of-day effect.
Note that the fluid approximation of the arrival rate process is characterized by the differential equation \( \bar{\lambda}'(t) = a(b - \bar{\lambda}(t)) \) for \( t \geq t_0 \), where \( \bar{\lambda}(t_0) = \bar{\lambda}_0 = \lambda(t_0) \). Solving it gives the following:

\[
\bar{\lambda}(s + t_0) = b(1 - e^{-as}) + \bar{\lambda}_0 e^{-as}, \quad s \geq 0.
\]

(24)

We assume that the fluid system is stable, i.e. \( b < \mu \). To state the main result of this subsection, we next define the line policy formally.

**Definition 6.** *(The line policy)* The call center manager offers the callback option to a customer arriving at time \( t_0 \) if \( Q_1(t_0) + A\lambda(t_0) \geq B \), where \( A, B \) are positive constants viewed as tuning parameters.

The line policy suggests that if the online queue length is high enough, i.e. \( Q_1(t) \geq B \), then the system always offers the callback option to the incoming customer. In this case, the system uses the callback option as an intervention tool to reduce the congestion in the online queue. Moreover, if the current arrival rate \( \lambda(t) \) is high, i.e. \( \lambda(t) \geq B/A \), the system offers the callback option to all incoming customers, even when the current online queue is empty, i.e. \( Q_1(t) = 0 \). In this case, the system anticipates large arrivals in the near future and uses the callback option as a preventive tool to avoid congestion in the online queue. The following proposition shows that the fluid lookahead policy simplifies to the line policy; see Appendix D for its proof.

**Proposition 3.** If the arrival rate process follows a CIR process, the fluid lookahead policy reduces to the line policy with \( A = (1 - e^{-ap/h})/a \) and \( B = (\mu - b)p/h + b(1 - e^{-ap/h})/a \).

Thus, the line policy is the optimal in the context of the fluid model.

**Corollary 2.** If the arrival rate process follows a CIR model, the line policy with \( A = (1 - e^{-ap/h})/a \) and \( B = (\mu - b)p/h + b(1 - e^{-ap/h})/a \) is optimal for the optimal control problem \( (P) \).

Note that the analysis in the fluid system ignore the variability of the Poisson process given the arrival rate process. Thus, the parameters \( A \) and \( B \) provided in Proposition 3 are not accurate for the original system. Nevertheless, we take the form of the line policy and treat the parameters \( A \) and \( B \) as the tuning parameters. Next section tests the performance of the line policy numerically and shows that it achieves excellent performance among non-anticipating policies.
6 Numerical Study

This sections uses Monte Carlo simulation\textsuperscript{16} to study the performance of the lookahead and line policies proposed in Sections 4-5. It consists of three studies. The first study compares the performance of the lookahead and line policies. The second one studies the value of the callback option to investigate when it is most valuable. To do so, it compares the call center’s performance with and without the callback option in various settings. The third study uses a data set from a US bank call center and explores the performance of the lookahead and line policies in practice. Moreover, it helps quantify the impact of the callback option on the customers’ abandonment behavior and the delays experienced in the offline queue.

6.1 The performances of the lookahead and line policies

This subsection considers three cases: The first case assumes that all customers accept the callback offer if they receive one. It also assumes that the arrival rate process follows a CIR model and investigates the impact of the volatility term of the arrival rate process on the system performance. The second case considers a reflected Ornstein-Uhlenbeck process as the arrival rate process and explores the system performance. The last case allows customers to reject the callback offer if they receive one and explores the impact of the rejection rate on the system performance.

We compare the performances of three policies: The lookahead policy under the complete foresight assumption (LH), the line policy (Line), and the MDP policy, i.e. the optimal non-anticipating policy derived from the associated Markov Decision Process (MDP); see Appendix E.1 for details of the computation of the MDP policy. In doing so, we optimize the tuning parameters $A$ and $B$ of the line policy. The MDP policy provides a lower bound for the line policy, because they both are non-anticipating. Moreover, the LH policy provides a lower bound for both, because it assumes the knowledge of all future arrival and service times.

As mentioned above, the first case models the arrival rate by a CIR process motivated by Glynn et al. (2018), see Equation (23). Recall from the Introduction that the variability of the arrival rate process captures the uncertainty at the mesoscopic scale. Thus, we assume $a = 1/30$ to be moderately small because its inverse captures the speed of adjustment of the mean reversion effect.

\textsuperscript{16}We use the exact simulation method proposed in Giesecke et al. (2011) for a Poisson process with stochastic intensity to simulate the arrival process.
We assume that $b = 0.9$, which equals the long term average arrival rate. In addition, we set the service rate $\mu = 1$. We compare the performances of the three policies for different volatility levels. To be specific, we consider $\sigma \in \{0.01, 0.02, 0.05, 0.1, 0.2, 0.5\}$. The coefficient of variations of the stationary distribution of the arrival rate process for these $\sigma$ values are $.04, .08, .21, .41, .82, 2.04$, respectively. Note that the parameters of the CIR process estimated from the US Bank data are $\sigma = 0.2$ and $a = 0.00873$ and $b = 1$, corresponding to the coefficient of variation 0.14. Thus, we chose this range of parameters to capture the cases of low, medium and high variability. Moreover, we choose the range of $p/h \in \{5, 10, 20, 50\}$ to study a broad spectrum of call center performance; see Appendix E.2 for details. Figure 3 depicts the trade-off between the average waiting time in the online queue and the fraction of customer routed to the offline queue. It shows that the line policy has excellent performance overall. Its performance is virtually identical to that of the MDP policy in all cases except for the case of $\sigma = 0.5$, i.e. the high volatility case. In that case, the Brownian term, i.e. the volatility term, dominates the behavior of the CIR process, whereas the line policy is derived focusing on the mean-reversion term of the CIR process.\footnote{When $\sigma = 0.5$, the stationary distribution of the arrival rate process, i.e. a gamma distribution with the shape parameter 0.24 and rate parameter 0.27, is not centered around the mean. Instead, its probability density function is skewed and has high value around zero. This sheds some light onto why the line policy does not perform well in this case.}

Table 7 compares the costs of the three policies. The top table shows the performance difference between the LH and MDP policies. LH policy outperforms the MDP policy significantly, because it can use future information regarding the arrival and service times. The bottom table compares the line policy and the MDP policy. It shows that the line policy does extremely well unless $\sigma$ is very high (relative to $\alpha$), i.e. the volatility term of the CIR process dominates.

The second case investigates how the line policy’s performance depends on the way the volatility...
Relative difference of the cost under the LH and MDP policies

<table>
<thead>
<tr>
<th>p/h</th>
<th>σ</th>
<th>.01</th>
<th>.02</th>
<th>.05</th>
<th>.1</th>
<th>.2</th>
<th>.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>29.20%</td>
<td>29.24%</td>
<td>28.60%</td>
<td>25.35%</td>
<td>21.92%</td>
<td>11.58%</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>27.41%</td>
<td>27.63%</td>
<td>26.67%</td>
<td>23.22%</td>
<td>18.19%</td>
<td>10.47%</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>25.21%</td>
<td>25.22%</td>
<td>24.86%</td>
<td>21.10%</td>
<td>14.95%</td>
<td>9.80%</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>21.83%</td>
<td>22.43%</td>
<td>21.61%</td>
<td>19.00%</td>
<td>13.32%</td>
<td>8.49%</td>
<td></td>
</tr>
</tbody>
</table>

Relative difference of the cost under the MDP and line policies

<table>
<thead>
<tr>
<th>p/h</th>
<th>σ</th>
<th>.01</th>
<th>.02</th>
<th>.05</th>
<th>.1</th>
<th>.2</th>
<th>.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>–</td>
<td>–</td>
<td>0.68%</td>
<td>0.99%</td>
<td>0.94%</td>
<td>1.07%</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>0.56%</td>
<td>–</td>
<td>3.00%</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>0.55%</td>
<td>0.70%</td>
<td>2.68%</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.74%</td>
<td>0.50%</td>
<td>0.70%</td>
<td>0.96%</td>
<td>1.22%</td>
<td>5.94%</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: The relative difference of the costs under the LH, MDP and line policies. We use ‘–’ to denote the case when the cost difference is within .5%

of the arrival rate process is modeled. In particular, it models the arrival rate process by a reflected Ornstein-Uhlenbeck (OU) process. Similar to the CIR process, the reflected OU process also has a state-dependent drift term, but its infinitesimal variance is constant (unlike the CIR process). We use the same parameters as those in the CIR process and let the volatility σ take the values of 0.01, 0.02, 0.05, 0.1, 0.2 and 0.5. Table 2 provides the optimality gap between the line policy and the

<table>
<thead>
<tr>
<th>p/h</th>
<th>σ</th>
<th>.01</th>
<th>.02</th>
<th>.05</th>
<th>.1</th>
<th>.2</th>
<th>.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>–</td>
<td>–</td>
<td>0.74%</td>
<td>0.83%</td>
<td>0.94%</td>
<td>4.23%</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>0.99%</td>
<td>1.28%</td>
<td>5.85%</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>–</td>
<td>–</td>
<td>0.66%</td>
<td>0.84%</td>
<td>1.61%</td>
<td>6.49%</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>–</td>
<td>0.64%</td>
<td>0.62%</td>
<td>0.98%</td>
<td>3.81%</td>
<td>15.83%</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: The relative difference of the costs under the MDP and line policies. We use ‘–’ to denote the case when the cost difference is within .5%

MDP policy, showing that the line policy performs well in most cases except for σ = 0.5, when the arrival rate process is mainly driven by the diffusion term. Note that the reflected O-U process is more variable than the CIR process with the same value of σ. This explains the larger optimality gaps in Table 2 comparing to those in Table 7.

The third case compares the three policies when the customers are allowed to reject the callback offers. We fix p/h = 10 and σ = 0.1 and vary the rejection rate, i.e. the fraction of customers that may reject the callback options if they are offered one, from 10% to 90%. In particular, we assume that a customer rejects the callback offer with a certain probability if he is offered one in the simulation. (Note that our model in Section 4 is more general, allowing for an arbitrary set R of customers who reject the callback offer.) Also, we adopt the CIR process as the arrival rate process.

\footnote{The coefficients of variation of the stationary distribution of the arrival rate process are 0.06, 0.12, 0.29, 0.58, 1.15 and 2.88, respectively.}
Figure 4 shows the fraction of customers receiving the callback option, the fraction of customers routed to the online queue, average waiting times of the online queue for various rejection rates under the three policies. In particular, we use the same values of the tuning parameters of the line policy as the ones used in the first case whereas the optimal MDP policy are re-computed for each rejection rate. When the rejection rate is not too high (i.e. less than 60%), the fraction of customers routed to the offline queue and the average online waiting time under all policies are nearly the same as the ones of the no rejection case (the first study). When the rejection rate high (i.e. higher than 80%), the online waiting time increases significantly. The main reason is that with high rejection rate, the call center manager has to offer the callback option to many customers and has less control on who ends up in the offline queue. Table 7 in Appendix E.2 compares the costs under the three policies. It shows that the line policy performs well compared to the MDP policy.

6.2 The value of the callback option

This subsection studies the value of the callback option in various settings to investigate when it is most valuable. To this end, we define the relative value of the callback option under a policy \( \pi \), denoted by \( \eta^\pi \) as follows:

\[
\eta^\pi = 1 - \frac{\text{Average cost under the callback policy } \pi}{\text{Average cost without the callback option}}.
\]

In particular, we are interested in how the variability of the arrival rate process affects the relative value of the callback option. Appendix E.3 considers an \( M/D/1 \) queue and an \( M/G/1 \) queue in heavy traffic and a deterministic system with periodic arrival rate. The analysis of these simple systems suggests that the callback option is more valuable when the arrival rate process is more volatile. Since the analysis of the relative value of the callback option is not tractable for more complex systems,
complex systems, we verify this conjecture numerically. To this end, we use the same setting as the one in the first study in Section 6.1 and calculate the relative value of the callback option of the three policies studied. Figure 5 shows that the relative values of the callback option under each of the three policies increase as the volatility term $\sigma$ increases.

Figure 5: The relative values of the callback option under three policies for $p/h = 5, 10, 20$ and 50.

6.3 The performance of the callback policies for a US bank call center

This subsection uses the individual call level data from a US bank call center\textsuperscript{19} to study the performance of the call center under the lookahead and the line policies; see Appendix E.4 for the detailed description of the data set. In addition, we study the impact of the abandonments on their performance. We assume that the arrival process follows a Poisson process with its intensity following a CIR process\textsuperscript{20}. We use the Bayesian approach to estimate the parameters that characterizing the arrival process via Markov Chain Monte Carlo (MCMC) method, i.e. the constants $a$, $b$ and $\sigma$ in Equation (23); see Appendix E.4 for a detailed description of the estimation procedure and the estimates of the parameters.

First, we analyze the performance of the proposed policies in a baseline model which assumes that the customers waiting in the online queue do not abandon. In addition, the customers always take the callback option and join the offline queue if they are offered one. We summarize the key performance metrics, e.g. the fraction of customers routed to the offline queue, average waiting times of the online and offline queues, in Table 3. As done in Section 6.1, the range of $p/h \in \{0.5, 1, 1.5, 2, 3, 4, 5\}$ minutes is chosen to explore a broad spectrum of call center performances. Note that the service rate was normalized to $\mu = 1$ in Section 6.1. Consequently, $p/h$ is the

\textsuperscript{19}This data set is publicly available at Service Enterprise Engineering (SEE) Center, Technion.

\textsuperscript{20}The analysis in Section 4, i.e. the optimality of the lookahead policy in a complete foresight system, only applies to the single-server queue. However, the call center has multiple agents. In the Monte Carlo simulation, we assume the system as a multi-server queue. Since we focus on using the callback option to mitigate the impact of the temporary arrival surges, we ignore the time-of-day effect shown in the data. Please check Appendix E.4 for detailed description of how we calibrate the primitives for the discrete-event simulation.
expected number of service completions in \( p/h \) time units. In the US bank dataset, the mean service time is 233 seconds and the calibrated number of server is 30, which corresponds to about one service completion per 7.8 seconds. Thus, the range of \( p/h \) corresponds to 3.8 - 38 service completions, which is comparable to the range of 5 - 50 considered in Section 6.1. The average waiting time is 79.43 seconds in the base case without the callback option. Table 3 shows that the

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Lookahead Policy (LH)} & \text{\% of offline cust.} & \text{Online wait. (sec)} & \text{Offline wait. (min)} & \text{Ave. Cost} \\
\hline
0.5 & 20.09\% & 8.18 & 5.94 & $6.33 \\
1.0 & 12.65\% & 11.46 & 9.19 & $10.79 \\
1.5 & 9.31\% & 13.94 & 11.75 & $13.95 \\
2.0 & 7.20\% & 15.95 & 13.78 & $16.21 \\
3.0 & 5.08\% & 19.17 & 19.35 & $19.94 \\
4.0 & 3.88\% & 21.61 & 25.40 & $22.59 \\
5.0 & 3.05\% & 23.49 & 30.74 & $24.38 \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
\text{MDP} & \text{\% of offline cust.} & \text{Online wait. (sec)} & \text{Offline wait. (min)} & \text{Ave. Cost} & \text{Change from LH} \\
\hline
0.5 & 26.01\% & 11.17 & 4.59 & $10.30 & 62.06\% \\
1.0 & 13.61\% & 16.23 & 7.97 & $15.46 & 43.24\% \\
1.5 & 9.81\% & 18.93 & 10.52 & $18.88 & 35.29\% \\
2.0 & 6.68\% & 22.26 & 14.50 & $21.51 & 32.72\% \\
3.0 & 4.71\% & 25.32 & 21.18 & $25.18 & 26.31\% \\
4.0 & 3.75\% & 27.38 & 29.32 & $27.84 & 23.27\% \\
5.0 & 2.64\% & 30.10 & 33.31 & $29.66 & 21.66\% \\
\hline
\end{array}
\]

\[
\begin{array}{|c|c|c|c|c|c|}
\hline
\text{Line Policy (LP)} & \text{\% of offline cust.} & \text{Online wait. (sec)} & \text{Offline wait. (min)} & \text{Ave. Cost} & \text{Change from MDP} \\
\hline
0.5 & 25.82\% & 11.25 & 4.56 & $10.31 & 0.05\% \\
1.0 & 13.56\% & 16.28 & 7.98 & $15.47 & 0.06\% \\
1.5 & 9.77\% & 19.00 & 10.55 & $18.91 & 0.16\% \\
2.0 & 6.67\% & 22.33 & 14.54 & $21.57 & 0.25\% \\
3.0 & 4.22\% & 26.31 & 21.27 & $25.33 & 0.58\% \\
4.0 & 2.85\% & 29.66 & 29.37 & $28.08 & 0.87\% \\
5.0 & 2.44\% & 31.02 & 33.38 & $29.97 & 1.04\% \\
\hline
\end{array}
\]

Table 3: The fraction of customers sent to the offline queue, average waiting times of the online and offline queues under LH, LP and MDP for various \( p/h \) values (assuming \( h = 1 \)).

average waiting time of the online queue reduces significantly by routing a small fraction of the customers to the offline queue under all three policies. In other words, the callback option helps the call center manager smooth the temporary arrival surges effectively. In addition, the waiting times of the offline queue under all three policies are reasonable by the industry standards; see the survey conducted by Software Advise mentioned in Introduction. This result is driven by the mean-reverting nature of the arrival process that causes the arrival surges at the mesoscopic scale. After the temporary surges end, which last from minutes to half-hour, the call center is able to use its excess capacity to serve the offline customers.

Next, we study the impact of the abandonments on the performance of these policies. We consider three abandonment scenarios: High, medium and low abandonment scenarios. We assume
that the abandonment time distribution and the service capacity are the same across three scenarios. The only difference of these three scenarios is the arrival processes. We use the data from December 2002, January 2003, and February 2003 of the system to estimate the parameters $a$, $b$, and $\sigma$ for the three scenarios. The system is operated under the lookahead policy or the line policy calculated from the system with no abandonment.

Figure 6 summarizes the simulated performance metrics of the system under the lookahead and line policies in the low, medium and high abandonment scenarios. Comparing the performance metrics of the no-abandonment and low-abandonment scenarios, we conclude that the performance metrics of the no-abandonment system are close to those of a system with low abandonments for both the lookahead and line policies. Therefore, the lookahead and line policies continue to do well when the abandonment rate is low. In the medium abandonment scenario, the fraction of customers routed to the offline increases significantly because the callback option keeps many customers in the system who would abandon in the system with no callback option, though the average online waiting time does not change much from the no abandonment case. This leads to a significant reduction in the fraction of abandoning customers under both policies. However, it also leads to a significant increase in the waiting time of the offline queue. In the high abandonment scenario, the average waiting time of the online queue under the high abandonment scenario also does not change significantly compared to the no abandonment case, but the fraction of customers routed to the offline queue increases significantly. Thus, the system is operated in the overloaded regime.

21 The arrival volumes of the brokerage customers decreased significantly from December 2002 to February 2003 while the service capacity did not change significantly. Thus, we use the data of these three months to study systems with different abandonment scenarios. To study the low abandonment scenario, we simulate the abandonments on top of the simulation of the baseline model (no abandonment study). To study the medium and high abandonment scenarios, we conduct the same calibration using the data from January 2003 and December 2002, respectively.
In this scenario, although the callback option ensures excellent performance of the online queue, some customers in the offline queue will never be served. Therefore, unless the call center manager uses busy signals to divert some arriving customers or increases the capacity to accommodate the increase in the system load due to the reduction in abandonments, the callback option may not be effective when the abandonment rate is high.

7 Concluding Remarks

This paper studies a call center in which its manager can offer the callback option to customers when the system is congested. We propose an easily implementable, yet effective callback policies under general arrival models. The analysis of the policies also provides some useful managerial insights. For example, it suggests offering the callback option as soon as the call center anticipates a sufficiently high arrival rate, rather than waiting until the queue has built up.

One limitation of our analysis is that it ignores the service quality requirements for the offline customers. In practice, many call centers may provide exact or vague (e.g. an estimated time window) time commitment to customers when they offer the callback options. Thus, one interesting extension of this analysis to study how such a service quality commitment for the offline customers change the structure of the optimal callback policy. It is also interesting to study how the information provided in the announcement of callback option affects customers’ accept/reject decisions and the system performance. Another possible extension is to allow for heterogeneity in customers’ delay sensitivity. The structure of an optimal callback policy remains unknown when customers’ delay sensitivity is heterogeneous.

References


40
A The Characterization of the System Dynamics

Recall that we assume the system is stable. We also assume that the system is empty initially. Denoting the cumulative amount of time the server is busy over \([0, t]\) by \(T(t)\), the number of customers in the system at time \(t\), denoted by \(Q(t)\), is given as follows:

\[
Q(t) = A(t) - S(T(t)) \geq 0, \quad t \geq 0.
\]

We restrict attention to work-conserving policies. That is, \(T(t)\) increases if and only if \(Q(t) > 0\). Clearly, we have that

\[
T(\cdot) \text{ is nondecreasing with } T(0) = 0,
\]

\[
0 \leq T(t) - T(s) \leq t - s, \quad 0 \leq s \leq t.
\] (25)

It follows from Equation (25) that \(T\) is Lipschitz continuous. Thus, it is absolutely continuous and differentiable almost everywhere with respect to the Lebesgue measure on \([0, \infty)\). A time \(t > 0\) is called a regular point if \(T\) is differentiable at time \(t\). Recall that we restrict attention to work-conserving policies. Thus, the following hold at regular times: For \(t \geq 0\),

\[
\dot{T}(t) = 1 \text{ whenever } Q(t) > 0,
\]

where \(\dot{T}(\cdot)\) denotes the derivative of \(T(\cdot)\).

Given a routing policy \(I\), recall that \(A_1(t)\) and \(A_2(t)\) denote the cumulative numbers of customers routed to the online and offline queues up to time \(t\), respectively. To formally describe the evolution of the online and offline queue lengths, let \(S_1(t)\) and \(S_2(t)\) denote the total number of online and offline customers served by time \(t\), respectively. Thus, the online and offline queue lengths, denoted by \(Q_1(t)\) and \(Q_2(t)\), respectively, are given as follows: For \(k = 1, 2\),

\[
Q_k(t) = A_k(t) - S_k(t), \quad t \geq 0,
\]

\[
Q_k(t) \geq 0, \quad t \geq 0.
\]

Because we restrict attention to the work-conserving policy that gives strictly priority to the online
queue, the following holds at all regular times: For \( t \geq 0 \),

\[
S_1(t) = \int_0^t \mathbb{1}_{\{Q_1(t) > 0\}} \, dS(T(t)),
\]

\[
S_2(t) = S(T(t)) - S_1(t),
\]

(26)

(27)

where \( \mathbb{1}_{\{\cdot\}} \) is the indicator function. In words, Equation (26) says that as long as there are customers in the online queue, the server works on that queue, i.e. it gives strict priority to the online queue. Similarly, Equation (27) implies whenever the online queue is empty, the server works on the offline queue (provided it is not empty). Note that the evolution of the total number of customers \( Q(t) \) in the system depends only on the arrival and service processes. It is independent of the routing and service policies as long as the latter is work-conserving.

B Proofs of the Lemmas in Section 4

This section consists of the proofs of the lemmas in Section 4.

B.1 Proofs of the lemmas in Section 4.1

Proof of Lemma 3. We prove the statement by induction. This is true for \( k = 0 \) by the definition of \( Q_1^0 \), i.e. \( Q_1^0 = Q_1^0 = Q \).

As the inductive assumption, suppose that the statement is true for \( k - 1 \) and \( i \in \mathcal{I}_{k-1}^C \) (for \( k = 1, \ldots, n \)), i.e. \( w_i^{k-1} = w_i \). We then show that it is true for \( k \) and \( i \in \mathcal{I}_k^C \), i.e. \( w_i^k = w_i \). Recall that the greedy rule defined in Definition 2 picks \( i_k^* \) in the \( k \)th iteration. Thus, \( \mathcal{I}_k^C = \mathcal{I}_{k-1}^C \setminus \{i_k^*\} \). The proof proceeds by considering the following two cases: \( i > i_k^* \) and \( i < i_k^* \).

Case 1: \( i > i_k^* \). That is, customer \( i \) arrives after customer \( i_k^* \). Since the customers are served in the LCFS fashion, the waiting time of customer \( i \) is independent of whether customer \( i_k^* \) is in the online or offline queue. Thus, the waiting time of customer \( i \) is unchanged after removing customer \( i_k^* \), i.e.

\[
w_i^k = w_i^{k-1} = w_i \quad \text{for} \quad i \in \mathcal{I}_k^C \cap \{i_k^* + 1, \ldots, n\},
\]

where the second equality follows from the inductive assumption.

Case 2: \( i < i_k^* \). That is, customer \( i \) arrives before customer \( i_k^* \). We discuss two sub-cases in
this case. The first sub-case is when customer $i$ completes service before customer $i_k^*$ arrives, i.e.

$$\tau_i^* > \tau_i + w_i^{k-1} = \tau_i + w_i = s_i,$$

where the first equality follows from the inductive assumption $w_i^{k-1} = w_i$ and the second equality follows from Lemma 2. Thus, removing customer $i_k^*$ from queue $Q_1^{k-1}$ does not affect the waiting time of customer $i$ because customer $i$ has left the queue by time $\tau_i^*$. To be specific, we have that $w_i^k = w_i^{k-1} = w_i$.

The second sub-case is when customer $i$ is still in the system when customer $i_k^*$ arrives, i.e. $\tau_i < \tau_i^* < s_i$. We show next that such a customer does not exist, i.e. $i \not\in I_k^C$. We prove it by contradiction. Suppose there exists an index $i \in I_k^C$ such that $\tau_i < \tau_i^* < s_i = \tau_i + w_i^{k-1}$.

Now consider the online queue $Q_1^{k-1}$ after $k-1$ deletions, and note that customer $i$ will be in the queue at least during the period $[\tau_i, s_i]$. In particular, we have that $Q_1^{k-1}(t) \geq 1$ for $t \in [\tau_i, s_i)$. Define

$$t_1 = \inf\{t > \tau_i : Q_1^{k-1}(t) = 0\}$$

as the first time when $Q_1^{k-1}$ hits zero after customer $i$ arrives at time $\tau_i$. Note that $t_1 > s_i > \tau_i^*$.

Thus, the following holds:

$$t_1 = \inf\{t > \tau_i^* : Q_1^{k-1}(t) = 0\},$$

because customer $i_k^*$ arrives after customer $i$. Essentially, we will argue that the greedy policy would pick customer $i$ to remove from the online queue in step $k$ (instead of customer $i_k^*$) which would be a contradiction. To see this, we will show next that the reduction in the area under the online queue length process due to removing customer $i$ from $I_k^{C-1}$ is precisely $t_1 - \tau_i$. That is,

$$H(Q_1^{k-1}) - H(\Phi(Q_1^{k-1}, \{i\})) = t_1 - \tau_i.$$

Figure 7 illustrates the change in the online queue length dynamics due to the removal of customer $i$. Similarly, the reduction in the area under the online queue length process due to removing customer $i_k^*$ from $I_k^{C-1}$ is $t_1 - \tau_i^*$, i.e.

$$H(Q_1^{k-1}) - H(\Phi(Q_1^{k-1}, \{i_k^*\})) = t_1 - \tau_i^*.$$
Figure 7: The solid line shows the queue length process $Q_{k-1}(t)$. The dash line shows the resulting queue length process after deleting a customer arriving at time $\tau_i$. The time $t_1$ is the first time when the queue hits zero. The shadowed area $(t_1 - \tau_i)$ is the savings from removing this customer.

Note that both customers $i$ and $i^*_k$ are in the set $I_{k-1}^C$. In addition, we have that

$$H(Q_{k-1}) - H(\Phi(Q_{k-1}, \{i\})) = t_1 - \tau_i > t_1 - \tau^*_k = H(Q_{k-1}) - H(\Phi(Q_{k-1}, \{i^*_k\})),$$

where the inequality follows from the assumption that $\tau_i < \tau^*_k$. This contradicts the definition of $i^*_k$ given in Equation (5). In other words, we should have removed customer $i$ instead of customer $i^*_k$ in the $k$th step. This completes the proof. 

\[\Box\]

B.2 Proofs of the lemmas in Section 4.2

Proof of Lemma 4. Note that for $\mathcal{I} \subseteq \mathcal{A}$,

$$\Phi(Q_{i}^0, \mathcal{I}) = \Phi(\bar{Q}_r + \bar{Q}_a, \mathcal{I}) = \bar{Q}_r + \Phi(\bar{Q}_a, \mathcal{I}),$$

where the first equity follows from $Q_{i}^0(t) = Q(t) = \bar{Q}_r(t) + \bar{Q}_a(t)$ for $t \geq 0$ and the second one follows because queue $r$ has strict priority and no customer $i \in \mathcal{R}$ belongs to $\mathcal{I} \subseteq \mathcal{A}$. Substituting this into Equation (1), we obtain for $\mathcal{I} \subseteq \mathcal{A}$ that

$$p|\mathcal{I} \cap \mathcal{A}| + H(\Phi(Q_{i}^0, \mathcal{I} \cap \mathcal{A})) = p|\mathcal{I}| + H(\bar{Q}_r + \Phi(\bar{Q}_a, \mathcal{I})) = H(\bar{Q}_r) + p|\mathcal{I}| + H(\Phi(\bar{Q}_a, \mathcal{I})),$$

where the last equality follows from the additivity of the integral used to calculate it. (Recall that $H(Q)$ is the total area under the queue process $Q$, i.e. the integral of the queue process $Q$). It follows from Proposition 1 (when it is applied with $\bar{Q}_a$ in place of $Q$) that the resulting set $\mathcal{I}$ of the first auxiliary $p/h$-lookahead policy minimizes $p|\mathcal{I}| + H(\Phi(\bar{Q}_a, \mathcal{I}))$. Moreover, because $H(\bar{Q}_r)$ is fixed,
i.e. it does not depend on $I$, the set $\tilde{I} \subseteq A$ prescribed by the first auxiliary \( p/h \)-lookahead policy minimizes the objective given in Equation (1). Hence, it is optimal for the first auxiliary system. \( \square \)

**Proof of Lemma 5.** We prove this lemma in two steps. We first show that $Q(\tilde{s}_i) = Q(\tau_i-) - \tilde{Q}_r(\tau_i-)$. We then show that $Q(t) > Q(\tau_i-) - \tilde{Q}_r(\tau_i-)$, $t \in [\tau_i, \tilde{s}_i]$. 

By definition of $\tilde{s}_i$ in Equation (10), the time $\tilde{s}_i$ is the time when a customer in queue $\tilde{Q}_a$ (either $\tilde{Q}_n$ or $\tilde{Q}_2$) enters the service. Since the customers in queue $\tilde{Q}_r$ enjoys the static priority (with preemption), we must have that $\tilde{Q}_r(\tilde{s}_i) = 0$. Substituting this equation into Equations (9) and (10), we have the following: For $i \in A$,

$$Q(\tilde{s}_i) = \tilde{Q}_a(\tilde{s}_i) = \tilde{Q}_a(\tau_i-) = Q(\tau_i-) - \tilde{Q}_r(\tau_i-).$$

The first equality follows from Equation (9) and that $\tilde{Q}_r(\tilde{s}_i) = 0$. The second equality follows from Equation (10) and the fact that $\tilde{Q}_r(\cdot)$ is right-continuous while the last equality follows from Equation (9).

In addition, the following holds for $t \in [\tau_i, \tilde{s}_i)$, which completes the proof:

$$Q(t) \geq \tilde{Q}_a(t) > \tilde{Q}_a(\tau_i-) = Q(\tau_i-) - \tilde{Q}_r(\tau_i-).$$

The first inequality follows from Equation (9) and the fact that $\tilde{Q}_r(t) \geq 0$. The second inequality follows from the definition of $\tilde{s}_i$ given in Equation (10), whereas the equality follows from Equation (9). \( \square \)

**Proof of Lemma 6.** Let $j$ be such that $\tau_j \in (\tau_i, \tilde{s}_i)$. It follows from Equation (10) that $\tilde{Q}_a(\tilde{s}_i) = \tilde{Q}_a(\tau_i-)$. In addition, since $\tau_j \in (\tau_i, \tilde{s}_i)$, it also follows from Equation (10) that $\tilde{Q}_a(\tau_j-) > \tilde{Q}_a(\tau_i-)$. Thus, we have that $\tilde{Q}_a(\tilde{s}_i) = \tilde{Q}_a(\tau_i-) < \tilde{Q}_a(\tau_j-)$. Once again, by the definition of $\tilde{s}_j$ in Equation (10), we conclude that $\tilde{s}_j < \tilde{s}_i$. Therefore, the following holds:

$$\tilde{s}_j < \tilde{s}_i < \tau_i + w < \tau_j + w,$$

where the second inequality follows from the fact that $i \not\in \tilde{I}$ and the last inequality follows from $\tau_j > \tau_i$ by assumption. Therefore, $j \not\in \tilde{I}$; see Definition 3.

We next show that $\tilde{Q}_r(\tilde{s}_i) = 0$. Note that at time $\tilde{s}_i$ the service of a customer in either queue $n$
or queue 2 (the offline queue) is completed. Queue \( r \) must be empty at time \( \hat{s}_i \) for that to happen because it has strict preemptive priority over queue \( n \) and the offline queue. Thus, \( \hat{Q}_r(\hat{s}_i) = 0 \). □

**Proof of Lemma 7.** (i) Suppose customers \( i_0, i_1, \ldots, i_k \) arrive in \([\tau_i, \hat{s}_i]\). We proceed with a proof by induction. Note by assumption that \( i = i_0 \notin \hat{I} \), which constitutes the induction basis. As the induction hypothesis, we assume that (i) holds for \( i_0, \ldots, j \) and we show that it also holds for customer \( j+1 \), i.e. \( j + 1 \notin \hat{I} \). Note that there are no arrivals to queue \( r \) during \((\tau_i, \tau_j+1)\) because all arriving customers join queue \( n \) by the induction hypothesis. Then the only potential changes to \( \hat{Q}_r \) during \((\tau_i, \tau_j+1)\) are due to service completions (of customers in queue \( r \)). We consider two possible cases: \( \hat{Q}_r(\tau_j+1-\hat{s}_i) > 0 \) and \( \hat{Q}_r(\tau_j+1-\hat{s}_i) = 0 \).

If \( \hat{Q}_r(\tau_j+1-\hat{s}_i) > 0 \), then all service effort during \((\tau_i, \tau_j+1)\) is dedicated to serving queue \( r \). Thus, all jobs who depart the system during \((\tau_i, \tau_j+1)\) belong to queue \( r \). Consequently, during \((\tau_i, \tau_j+1)\) the total number of customers in the system decreases by the same amount \( \hat{Q}_r \) decreases. That is,

\[
Q(\tau_i) - Q(\tau_j+1-) = \hat{Q}_r(\tau_i) - \hat{Q}_r(\tau_j+1-).
\]

Rearranging the terms then gives the following:

\[
Q(\tau_j+1-) - \hat{Q}_r(\tau_j+1-) = Q(\tau_i) - \hat{Q}_r(\tau_i) = Q(\tau_i-) - \hat{Q}_r(\tau_i-) + 1 > Q(\tau_i-) - \hat{Q}_r(\tau_i-).
\]

The last equality follows from the fact that customer \( i \) enters the system but does not join queue \( r \) at time \( \tau_i \). Thus, it follows from this equality and the definition of \( \hat{s}_i \) (see Equation (12)) that

\[
Q(\hat{s}_i) = Q(\tau_i-) - \hat{Q}_r(\tau_i-) < Q(\tau_j+1-) - \hat{Q}_r(\tau_j+1-).
\]

It follows from the definition of \( s_{j+1} \) (see Equation (12)) that \( \hat{s}_{j+1} \) is the first time when the queue length \( Q(t) \) falls back to the level of \( Q(\tau_j+1-) - \hat{Q}_r(\tau_j+1-) \), i.e. \( Q(t) \geq Q(\tau_j+1-) - \hat{Q}_r(\tau_j+1-) \) for \( t \in [\tau_{j+1}, s_{j+1}] \). Thus, we conclude that \( \hat{s}_{j+1} < \hat{s}_i \). Therefore, the following holds:

\[
\hat{s}_{j+1} < \hat{s}_i < \tau_i + w < \tau_{j+1} + w,
\]

where the second inequality follows from the fact that \( i \notin \hat{I} \) and the definition of \( \hat{I} \) (see Equation (13)). Thus, we conclude from the definition of \( \hat{I} \) and Equation (28) that customer \( j + 1 \) is not offered the callback option, i.e. \( j + 1 \notin \hat{I} \).
If \( \hat{Q}_r(\tau_{j+1}^-) = 0 \), then
\[
Q(\tau_{j+1}^-) - \hat{Q}_r(\tau_{j+1}^-) = Q(\tau_{j+1}^-) > Q(\tau_i^-) - \hat{Q}_r(\tau_i^-) = \hat{Q}(\hat{s}_i),
\]
where the inequality follows from the definition of \( \hat{s}_i \) in Equation (12) and that \( \tau_{j+1} < \hat{s}_i \). By the definition of \( \hat{s}_{j+1} \), we conclude that \( \hat{s}_{j+1} < \hat{s}_i \). Therefore, Equation (28) holds as well. Thus, we conclude that customer \( j + 1 \) is not offered the callback option, i.e. \( j + 1 \not\in \hat{I} \), in this case as well, concluding the proof of (i).

(ii) Suppose there are \( k_i \geq 0 \) arrivals during \([\tau_i, \hat{s}_i]\). It follows from part (i) of this lemma that all of them join queue \( I \). It also follows from the definition of \( \hat{s}_i \) (see Equation (12)) that
\[
Q(\hat{s}_i) = Q(\tau_i^-) - \hat{Q}_r(\tau_i^-).
\]
In other words, the total number of customers in the system decreases by \( \hat{Q}_r(\tau_i^-) \). Therefore, there are \( k_i + \hat{Q}_r(\tau_i^-) \) service completions during \([\tau_i, \hat{s}_i]\). Moreover, by definition of \( \hat{s}_i \), the number of service completions during \([\tau_i, \hat{s}_i]\) is \( k_i + \hat{Q}_r(\tau_i^-) - 1 \). Since queue \( r \) has the highest priority, all customers in queue \( r \) will be served first so that \( \hat{Q}_r(\hat{s}_i) = 0 \).

Next, we argue that customer \( i \) remains in queue \( I \) during \([\tau_i, \hat{s}_i]\). Suppose not, i.e. he receives service and departs the system before \( \hat{s}_i \). Because queue \( r \) has the highest priority and queue \( I \) is served under LCFS discipline, there must be \( k_i + \hat{Q}_r(\tau_i^-) \) or more service completions during \([\tau_i, \hat{s}_i]\), which contradicts the fact that there are \( k_i + \hat{Q}_r(\tau_i^-) - 1 \) service completions during \([\tau_i, \hat{s}_i]\) as established immediately above. In particular, because customer \( i \) remains in queue \( I \), the online queue is always nonempty in \((\tau_i, \hat{s}_i)\). Therefore, the \( k + \hat{Q}_r(\tau_i^-) \) customers who leave the queue during \((\tau_i, \hat{s}_i)\) are from the online queue, with \( k \) of them from queue \( I \) and \( \hat{Q}_r(\tau_i^-) \) of them from queue \( r \). Thus, the queue length of queue \( I \) remains the same, i.e. \( \hat{Q}_n(\hat{s}_i) = \hat{Q}_n(\tau_i^-) \). Since queue \( I \) is served under the LCFS discipline, the last customer who leaves the queue is the first one among the \( k_i \) arrivals, i.e. customer \( i \). In other words, customer \( i \) leaves the queue at time \( \hat{s}_i \).

(iii) We proceed with a proof by contradiction. Suppose \( \hat{Q}_n(\tau_i^-) > 0 \). In particular, there exist \( i' \not\in \hat{I} \) such that \( \tau_i \in (\tau_i', \hat{s}_{i'}) \). Then it follows from part (i) of this lemma that \( i \not\in \hat{I} \), which contradicts the assumption that \( i \in \hat{I} \). Thus, \( \hat{Q}_n(\tau_i^-) = 0 \). Moreover, because \( i \in \hat{I} \), customer \( i \) does not join queue \( I \). Instead, he joins queue \( r \) if \( i \in \mathcal{R} \) and he joins the offline queue if \( i \in \mathcal{A} \). So we conclude that \( \hat{Q}_n(\tau_i) = \hat{Q}_n(\tau_i^-) = 0 \). \( \Box \)
Proof of Lemma 8. As done throughout the paper, we consider a busy period and suppose that there are \( n \) customer arrivals during it. Assuming at most one event can happen at any fixed time, there are \( 2^n \) events in the busy period. That is, the state of the system changes at \( 2^n \) points in time. It will be convenient, however, to denote the event times as \( 0 < t_1 < t_2 < \cdots < t_{2n} \).

(Notice that the set of arrival times \( \{\tau_1, \ldots, \tau_n\} \) is a subset of the event times \( \{t_1, \ldots, t_{2n}\} \); and the set \( \{t_1, \ldots, t_{2n}\} \backslash \{\tau_1, \ldots, \tau_n\} \) corresponds to the service completion times. We also let \( t_0 = 0 \) for notational convenience.

Note that it suffices to show that

\[
Q_r(t) = \hat{Q}_r(t), \quad Q_n(t) = \hat{Q}_n(t) \quad \text{and} \quad Q_2(t) = \hat{Q}_2(t), \quad t \geq 0,
\]

(29)
at \( t = t_i \) for \( i = 0, 1, \ldots, 2n \), because the system state changes only at those times. We proceed by induction. As the induction basis, we note that Equation (29) holds for \( i = 0 \), because both the original system and the auxiliary system are empty at time zero.

As the inductive hypothesis, we assume that Equation (29) holds for \( l = 0, 1, \ldots, i \) and show that it holds for \( l = i + 1 \) as well. Because there is no event in between \((t_i, t_{i+1})\), the following holds:

\[
Q_k(t_{i+1}^-) = \hat{Q}_k(t_{i+1}^-) \quad \text{for} \quad k = r, n, 2.
\]

In which follows, we consider two cases depending on whether the event at time \( t_{i+1} \) is an arrival or a departure.

Case 1: Suppose a customer, say customer \( j \) arrives at time \( t_{i+1} \). In particular, \( \tau_j = t_{i+1} \). Note that because \( Q_r(t_{i+1}^-) = \hat{Q}_r(t_{i+1}^-) \), it follows from Equations (2), (10) and Lemma 5 that \( s_j = \hat{s}_j \). Consequently, in both system, customer \( j \) is routed to the same queue. To be specific, if \( s_j = \hat{s}_j \geq \tau_j + w \), then both systems offers the callback option to the customer \( j \). If he accepts it and joins the offline queue, then the offline queue increases by one in both systems. That is,

\[
Q_2(t_{i+1}) = \hat{Q}_2(t_{i+1}) = Q_2(t_{i+1}^-) + 1 = \hat{Q}_2(t_{i+1}^-) + 1.
\]

Other quantities remain unchanged. If customer \( j \) rejects the offer, then both \( Q_r \) and \( \hat{Q}_r \) increase by one while other quantities remain unchanged. On the other hand, if \( s_j = \hat{s}_j < \tau_j + w \), then
neither system offers the callback option; and both \( Q_n \) and \( \hat{Q}_n \) increase by one, whereas all other quantities remain unchanged. Thus, we conclude that Equation (29) holds at time \( t_{i+1} \) in Case 1.

**Case 2:** In this case, there is a departure at time \( t_{i+1} \). We consider four further sub-cases:

(a) The online queue is empty in both systems at time \( t_{i+1} \). That is, \( Q_1(t_{i+1}^{-}) = \hat{Q}_1(t_{i+1}^{-}) \).

Equivalently, we have that

\[
Q_r(t_{i+1}^{-}) = Q_n(t_{i+1}^{-}) = 0 \quad \text{and} \quad \hat{Q}_r(t_{i+1}^{-}) = \hat{Q}_n(t_{i+1}^{-}) = 0.
\]

Therefore, the offline queue decreases by one in both systems, i.e. \( Q_2 \) and \( \hat{Q}_2 \) decreases by one, whereas the other quantities remain the same.

(b) Queue \( n \) is empty whereas queue \( r \) is not empty in both systems. That is

\[
Q_r(t_{i+1}^{-}) = \hat{Q}_r(t_{i+1}^{-}) > 0 \quad \text{and} \quad Q_n(t_{i+1}^{-}) = \hat{Q}_n(t_{i+1}^{-}) = 0.
\]

In both systems, the online queue has higher priority than the offline queue. Because queue \( n \) is empty in both systems, the server works on a customer in queue \( r \) in both systems. Thus,

\[
Q_r(t_{i+1}) = \hat{Q}_r(t_{i+1}) = Q_r(t_{i+1}^{-}) - 1 = \hat{Q}_r(t_{i+1}^{-}) - 1,
\]

whereas all other queues remain the same, proving the induction hypothesis.

(c) Queue \( r \) is empty whereas queue \( n \) is not empty in both systems. That is

\[
Q_r(t_{i+1}^{-}) = \hat{Q}_r(t_{i+1}^{-}) = 0 \quad \text{and} \quad Q_n(t_{i+1}^{-}) = \hat{Q}_n(t_{i+1}^{-}) > 0.
\]

By an argument similar to the one in Case 2(b), we conclude that both \( Q_n \) and \( \hat{Q}_n \) decrease by one and all other queue lengths remain the same.

(d) Both queue \( n \) and queue \( r \) are nonempty in both systems. That is,

\[
Q_r(t_{i+1}^{-}) = \hat{Q}_r(t_{i+1}^{-}) > 0 \quad \text{and} \quad Q_n(t_{i+1}^{-}) = \hat{Q}_n(t_{i+1}^{-}) > 0.
\]

The second auxiliary system picks a customer in queue \( r \) to enter service because queue \( r \) has the highest priority in that system. Next, we argue that the original system also picks a customer in queue \( r \) to enter service. To this end, we first note that the same set of customers are routed to queue \( r \) in both systems so far by the induction hypothesis. Because we also
have \( Q_r(t_{i+1}^-) = \hat{Q}_r(t_{i+1}^-) \), the same number of customers have left queue \( r \) in both systems by time \( t_{i+1}^- \). Moreover, since queue \( r \) in both systems is served with the FCFS service discipline, the set of customers who have entered service from queue \( r \) in both systems are the same. Therefore, queue \( r \) in both systems consist of the same customers at time \( t_{i+1}^- \).

Let index \( j \) correspond to the customer with the smallest index present in queue \( n \) of the second auxiliary system at time \( t_{i+1}^- \). He arrived at time \( \tau_j \) and will leave the system at time \( \hat{s}_j \) in the second auxiliary system by part (iii) of Lemma 7. Since customer \( j \) has not left the system yet, we have that \( t_{i+1} \in [\tau_j, \hat{s}_j) \). Thus, it follows from part (i) of Lemma 7 that all customers arriving during \( [\tau_j, \hat{s}_j) \) are routed to queue \( n \) in the second auxiliary system. The same (until time \( t_{i+1}^- \)) is true for the original system by the induction hypothesis. Therefore, all customers in queue \( r \) at time \( t_{i+1}^- \) (in either system) must have arrived before \( \tau_j \). Moreover, all customers in queue \( n \) at time \( t_{i+1}^- \) (in either system) must have arrived at or after time \( \tau_j \). Then, because the online queue (combination of queue \( r \) and queue \( n \)) is served in a FCFS basis in the original system, customers in queue \( r \) have priority over customers in queue \( n \). Therefore, a customer in queue \( r \) is picked to enter service at time \( t_{i+1} \) in the original system as well. Thus, Equation (29) holds for \( t_{i+1} \).

This completes the proof. \( \square \)

**Proof of Lemma 9.** We first prove that \( \tilde{Q}_i(t) = \hat{Q}_i(t) \) for \( i = 1, 2 \) and \( t \geq 0 \). Along the way we also establish that \( \tilde{I} = \tilde{I} \cap A = \hat{I} \cap A \). We start by showing that

\[
\tilde{Q}_1(t) \leq \hat{Q}_1(t), \ t \geq 0.
\]  

(30)

Note that all customers \( i \in \mathcal{R} \) join queue \( r \) in the first auxiliary system. Moreover, all customers who join queue \( r \) in the second auxiliary system belong to the set \( \mathcal{R} \). Thus, if a customer joins queue \( r \) in the second auxiliary system, he joins queue \( r \) in the first auxiliary system as well. Recall that queue \( r \) has the highest priority among all queues in both auxiliary systems and that the service times are associated with the server and not with the customers. Thus, we conclude that

\[
\tilde{Q}_r(t) \geq \hat{Q}_r(t) \geq 0.
\]  

(31)

50
In particular, the following holds:

\[ Q(t) - \hat{Q}_r(t) \leq Q(t) - \hat{Q}_r(t), \quad t \geq 0. \]

Using this, we conclude from Lemma 5 and Equation (12) that for \( i \in A \), \( \tilde{s}_i \leq \hat{s}_i \). Consequently, if customer \( i \in A \) is routed to the online queue in the first auxiliary system, i.e. \( \tilde{s}_i < \tau_i + w \), then he is routed to the online queue in the second auxiliary system as well, because \( \hat{s}_i < \tilde{s}_i < \tau_i + w \). In addition, also note that all customers in the set \( R \) are routed to the online queue in both systems.

To summarize, all customers who are routed to the online queue in the first auxiliary system are also routed to the online queue in the second auxiliary system. Because the online queue has the strict preemptive priority over the offline queue in both systems, we conclude that \( \tilde{Q}_1(t) \leq \hat{Q}_1(t) \) for \( t \geq 0 \).

Next, we turn to proving \( \tilde{Q}_1(t) = \hat{Q}_1(t) \) for \( t \geq 0 \). Because \( \tilde{Q}_1(t) \leq \hat{Q}_1(t) \) for \( t \geq 0 \) as proved immediately above, it suffices to show \( \tilde{Q}_1(t) = \hat{Q}_1(t) \) for all busy periods of the online queue in the second auxiliary system. Furthermore, since the online queue has the strict preemptive priority in both systems (and that the service times are associated with the server and not with the customers), it suffices to show that the same set of customers are routed to the online queue. In what follows, we will show this for the first busy period of the online queue in the second auxiliary system. The same argument can be repeated inductively for the other busy periods of the online queue to complete the proof.

To this end, consider the second auxiliary system and let \([t_1, t_2]\) denote the first busy period of its online queue. Also let \( i_1, \ldots, i_k \) denote the customers arriving in \([t_1, t_2]\). We consider the following two cases. First, assume that all of them receive the callback option, i.e. \( i_l \in \hat{I} \) for \( l = 1, \ldots, k \). In this case, customer \( i_l \) joins either queue \( r \) (if \( i_l \in R \)) or the offline queue (if \( i_l \in A \)). In particular, no customer joins queue \( n \) and we conclude that \( \hat{Q}_n(t) = 0 \) for \( t \in [t_1, t_2] \). In this case, we also conclude the following for \( t \in [t_1, t_2] \):

\[ \tilde{Q}_1(t) \leq \hat{Q}_1(t) = \hat{Q}_r(t) \leq \tilde{Q}_r(t) \leq \hat{Q}_1(t), \quad (32) \]

where the first inequality follows from (30). The equality follows from Equation (11) and that \( \hat{Q}_n(t) = 0 \) for \( t \in [t_1, t_2] \). The second inequality follows from (31), whereas the last one follows
from (8) and that $\tilde{Q}_n(t) \geq 0$ for $t \in [t_1, t_2]$. Therefore, in this case, it follows Equation (32) that $\hat{Q}_1(t) = \tilde{Q}_1(t)$ for $t \in [t_1, t_2]$.

In the second case, there exists a customer who does not receive the callback option. Let $i^* \leq i_k$ be the first such customer. Then by definition, we have that

$$\hat{Q}_n(t) = 0 \quad \text{for} \quad t \in [t_1, \tau_{i^*}) \quad \text{and} \quad \hat{Q}_n(\tau_{i^*}) = 1.$$  \hspace{1cm} (33)

Thus, Equation (32) holds for $t \in [t_1, \tau_{i^*})$, which implies that $\hat{Q}_1(t) = \tilde{Q}_1(t)$ for $t \in [t_1, \tau_{i^*})$. The following observation facilitates the analysis to follow: Equation (32) also implies that $\hat{Q}_r(t) = \tilde{Q}_r(t)$ for $t \in [t_1, \tau_{i^*})$. Therefore, the following holds:

$$\hat{Q}_n(t) = \hat{Q}_1(t) - \hat{Q}_r(t) = \tilde{Q}_1(t) - \tilde{Q}_r(t) = \bar{Q}_n(t) = 0, \quad t \in [t_1, \tau_{i^*}).$$  \hspace{1cm} (34)

The first and third equalities follow from Equations (8) and (11). The last equality follows from Equation (33).

We complete the proof by showing that $\hat{Q}_1(t) = \tilde{Q}_1(t)$ for $t \in [\tau_{i^*}, t_2]$. To this end, we first show that $\hat{s}_{i^*} = t_2$. That is, the busy period of the online queue ends when customer $i^*$ departs the second auxiliary system. To see this, note from part (ii) of Lemma 7 and Equation (34) that $\hat{Q}_r(\hat{s}_{i^*}) = 0$ and $\hat{Q}_n(\hat{s}_{i^*}) = \hat{Q}_n(\tau_{i^*}) = 0$. Thus, it follows from (11) that $\hat{Q}_1(\hat{s}_{i^*}) = 0$. It also follows from Part (ii) of Lemma 7 that $\hat{s}_{i^*}$ is the time when customer $i^*$ leaves the system, which implies that $\hat{Q}_1(t) > 0$ for $t \in [\tau_{i^*}, \hat{s}_{i^*})$. Thus, we conclude that $\hat{s}_{i^*} = t_2$. Second, we note from part (i) of Lemma 7 that $i^*, \ldots, i_k \not\in \hat{I}$, i.e. all customers arriving after $i^*$ (customers $i^* + 1, \ldots, i_k$) join the online queue (in particular, they join queue $n$) in the second auxiliary system. Therefore, to show that $\hat{Q}_1(t) = \tilde{Q}_1(t)$ for $t \in [\tau_{i^*}, t_2]$, it suffices to show that all customers in $\{i^*, \ldots, i_k\}$ join the online queue in the first auxiliary system as well. To show this, we discuss two cases.

**Case 1:** Customers $i^*, \ldots, i_k$ belong to the set $\mathcal{R}$, i.e. $\{i^*, \ldots, i_k\} \subseteq \mathcal{R}$. Recall that all customers in the set $\mathcal{R}$ join queue $r$ in the first auxiliary system. Therefore, the desired result holds trivially in this case.

**Case 2:** There exists a customer $j \in \{i^*, \ldots, i_k\} \cap \mathcal{A}$. Let $j^*$ be the smallest such index. Note
that \( \hat{Q}_1(\tau_j -) = \tilde{Q}_1(\tau_j -) \) because customers \( i^*, i^* + 1, \ldots, j^* - 1 \) belong to set \( R \) and they all join the online queue in both systems. Thus, no one joins the offline queue during \([\tau_{i^*}, \tau_{j^*})\). Also note that \( \hat{Q}_1(t) > 0 \) for \( t \in [\tau_{i^*}, \tau_{j^*}) \) because customer \( i^* \) leaves the system at \( \hat{s}_{i^*} = t_2 \) (in the second auxiliary system). Consequently, no customer in the offline queue enters service during \([\tau_{i^*}, \tau_{j^*})\) in the second auxiliary system. Therefore, the offline queue length does not change during \([\tau_{i^*}, \tau_{j^*})\), i.e. \( \tilde{Q}_2(t) = \tilde{Q}_2(\tau_{i^*} -) \) for \( t \in [\tau_{i^*}, \tau_{j^*}) \). Therefore, the following equality, which will be used below, holds:

\[
\begin{align*}
Q(\tau_{j^*} -) - \hat{Q}_1(\tau_{j^*} -) = \tilde{Q}_2(\tau_{i^*} -) = Q(\tau_{i^*} -) - \hat{Q}_1(\tau_{i^*} -).
\end{align*}
\] (35)

It remains to show that customers \( j^*, \ldots, i_k \) all join the online queue \( \tilde{Q}_1 \) in the first auxiliary system. Recall that \( j^* \) is the first customer in \( \{i^*, \ldots, i_k\} \cap A \). Therefore, we have that \( \{i^*, \ldots, j^* - 1\} \subseteq R \) and that customers \( i^*, \ldots, j^* - 1 \) all join queue \( r \) in the first auxiliary system. Therefore, it follows from Equation (34) that \( \tilde{Q}_n(\tau_{j^*} -) = 0 \). In addition, the following holds:

\[
\begin{align*}
Q(\hat{s}_{i^*}) &= Q(\tau_{i^*} -) - \hat{Q}_r(\tau_{i^*} -) \\
&= Q(\tau_{i^*} -) - \hat{Q}_1(\tau_{i^*} -) \\
&= Q(\tau_{j^*} -) - \hat{Q}_1(\tau_{j^*} -) \\
&= Q(\tau_{j^*} -) - \tilde{Q}_r(\tau_{j^*} -) \\
&= Q(\tau_{j^*} -) - \tilde{Q}_r(\tau_{j^*} -).
\end{align*}
\] (36)

The first equality follows from definition of \( \hat{s}_{i^*} \) in Equation (12). The second equality follows from Equation (33). The third equality follows from Equation (35). The fourth equality follows because \( \hat{Q}_1(\tau_{j^*} -) = \tilde{Q}_1(\tau_{j^*} -) \). The last equality follows because \( \tilde{Q}_n(\tau_{j^*} -) = 0 \). In addition, it follows from Equation (12) that

\[
Q(t) > Q(\hat{s}_{i^*}) = Q(\tau_{i^*} -) - \hat{Q}_r(\tau_{i^*} -) \text{ for } t \in [\tau_{i^*}, \hat{s}_{i^*}).
\]

Combining this with Equation (36) and that \( \tau_{j^*} \geq \tau_{i^*} \), we conclude that

\[
Q(t) > Q(\hat{s}_{i^*}) = Q(\tau_{j^*} -) - \hat{Q}_r(\tau_{j^*} -) \text{ for } t \in [\tau_{j^*}, \hat{s}_{i^*}) \subseteq [\tau_{i^*}, \hat{s}_{i^*}).
\]
It follows from this and Lemma 5 that \( \tilde{s}_{j^*} = \hat{s}_{i^*} \), which implies that

\[
\tilde{s}_{j^*} = \hat{s}_{i^*} < \tau_{i^*} + w \leq \tau_{j^*} + w.
\]

Therefore, we have that \( j^* \notin \tilde{I} \) and \( j^* \in \mathcal{A} \), i.e. customer \( j^* \) does not receive the callback offer and joins the online queue (in particular, queue \( n \)) in the first auxiliary system. Then, it follows from Lemma 6 that all customers in \( \mathcal{A} \) arriving during \([\tau_j, \tilde{s}_j]\) also join the online queue in the first auxiliary system. In addition, customers in \( \mathcal{R} \) arriving during \([\tau_j, \tilde{s}_j]\) join the online queue in the first auxiliary system by definition. Thus, because \( \hat{s}_{i^*} = \tilde{s}_{j^*} = t_2 \), we conclude that all customers arriving during \([\tau_j, t_2]\) join the online queue in the first auxiliary system, completing the proof. \( \square \)

**B.3 An example illustrating the dynamics of the three systems**

We first provide an example illustrating the\( p/h\)-lookahead policy when no customer rejects the callback option. Figure 8 shows the routing decisions of two customers when the system operates under the \( p/h\)-lookahead policy for a given sample path.

![Figure 8: The routing decisions of two customers when the system operates under the \( p/h\)-lookahead policy for a given sample path. Left panel: Customer 1 is routed to the offline queue because \( s_1 \geq \tau_1 + p/h \). Right panel: Customer 2 stays in the online queue because \( s_2 < \tau_2 + p/h \).](image)

Next, we suppose that the set \( \mathcal{A} = \{1, 2, 4\} \). The dynamics of queue \( r \), queue \( n \), the online queue and the offline queue in all three systems are summarized in Figure 9. Customers 3 and 5 are routed to queue \( r \) automatically in the first auxiliary system because they are in set \( \mathcal{R} \). Customers 1 and 4 are offered the callback option and join the offline queue. In both the second auxiliary system and the original system, the system offers the callback option to customers 1, 3 and 5. However, only customers 1 and 4 take the offer and join the offline queue. Comparing the two auxiliary systems, we observe that although the dynamics of queue \( r \) and queue \( n \) are different, the same set of customers (customer 1 and 4) join the offline queue. Comparing the second auxiliary system and the original system, we observe that the dynamics of all queues are identical in both systems,
Figure 9: The dynamics of the three systems (the first and second auxiliary systems as well as the original one) of the example shown in Figure 8 with the set $\mathcal{A} = \{1, 2, 4\}$. The left (middle) panel shows the first (second) auxiliary system whereas the right panel shows the original system. The indices in the figures denote the indices of the customers who just arrive or just complete service. Those who received the callback offer are highlighted using the bold font. The callback (C) and no-callback (NC) episodes of the second auxiliary and the original systems are shown in the middle and right panels.

though they have different service priority rule within the online queue. In both systems, customers 3 and 4 arrive during the callback episode. Both systems switch to the no-callback episode when customer 5 arrives. Thus, customer 3 is served before customer 5 because customers in $\hat{Q}_r(\cdot)$ has strict priority over customers in $\hat{Q}_n(\cdot)$ in the second auxiliary system. However, in the original system, customer 3 is served before customer 5 because the online queue is served under the FCFS fashion. This illustrates the earlier the observation that all customers in queue $r$ currently must have arrived during the callback episode prior to the current no-callback episode.

C Proof of Proposition 2

The proof has two major steps: First, we characterize the dynamics of the fluid system under the fluid lookahead policy in Section C.1. Using that characterization, we prove the optimality of the fluid lookahead policy in Section C.2. As mentioned in Section 5.1, Proposition 2 makes the following technical assumptions:
(i) The arrival rate process \( \tilde{\lambda}(t) \) (for \( t \in [t_0, \bar{T}] \)) is a continuous function;

(ii) \( \tilde{\lambda}(t) > 0 \) and \( \tilde{\lambda}(t) - \mu \) is not identically equal to zero or \( f_t(\tilde{\lambda}(t)) \) on any interval \( (a, b) \subseteq [t_0, \bar{T}] \).

Moreover, as stated earlier, the acceptance function \( f_t(\cdot) \) satisfies the following:

- Fixing the time \( t \), the acceptance function \( f_t(\tilde{\alpha}) \) is continuous and strictly increasing in \( \tilde{\alpha} \). In addition, we have that \( f_t(\tilde{\alpha}) \in [0, \bar{\alpha}] \).

- Fixing the \( \tilde{\alpha} \), the acceptance function \( f_t(\tilde{\alpha}) \) is continuous in \( t \).

C.1 Characterization of the system dynamics under the fluid lookahead policy

Without loss of generality, this section restricts attention to the case when \( \bar{T} - t_0 > p/h \). Otherwise, the fluid lookahead policy does not offer the callback option to any customers.

Switching between the callback and no-callback episodes. To facilitate the analysis to follow, we first define the callback and no-callback episodes formally. To be specific, we let \( c_1 = t_0 \) and define \( c_i \) and \( d_i \) (for \( i = 1, \ldots, n \)) recursively as follows:

\[
d_i = \inf\{t \geq c_i : \alpha(t) < \min(f_t^{-1}(\tilde{\lambda}(t) - \mu), \tilde{\lambda}(t))\},
\]

\[
c_{i+1} = \inf\{t \geq d_i : \alpha(t) > 0\},
\]

where \( \inf \emptyset = \bar{T} \) and the recursive definition ends when either the value of \( d_i \) or \( c_i \) (for some \( i \geq 1 \)) equals to \( \bar{T} \). The time \( d_i \) is defined as the ending time of a callback episode while the time \( c_i \) is defined as the ending time of a no-callback episode. We call the period \([c_i, d_i]\) a callback episode and \([d_i, c_{i+1}]\) a no-callback episode. This definition implicitly assumes that the busy period \([t_0, \bar{T}]\) starts with a callback episode, which does not always happen. If the period starts with a no-callback episode, then \( d_1 = c_1 \), i.e. the first callback episode \([c_1, d_1]\) is an empty set. Recall that the system is in the no-callback episode during \([\bar{T} - p/h, \bar{T}]\). Thus, there exists an integer \( n \) such that \( d_n < \bar{T} \).

(We also set \( c_{n+1} = \bar{T} \) for notational convenience only.) In other words, if the system starts with a callback episode at time \( t_0 \), there exists \( n \) callback episodes \([c_i, d_i]\) and \( n \) no-callback episodes \([d_i, c_{i+1}]\) (for \( i = 1, \ldots, n \)) during the current busy period \([t_0, \bar{T}]\). Otherwise, there are \( n-1 \) callback episodes and \( n \) no-callback episodes. Figure 10 illustrates how the callback and no-callback episodes alternate.
Figure 10: The system is in the callback episode during the periods \([c_1, d_1]\). Then the system switches to the no-callback episode at time \(d_1\) and alternates between the callback episodes and no-callback episodes afterwards. The system always ends with a no-callback episode at time \(\bar{T}\).

Next, we characterize the system dynamics within each callback and no-callback episode.

**The system dynamics within a callback episode.** We fix an index \(i\) and discuss the system dynamics during the callback episode \([c_i, d_i]\) (for \(i = 1, \ldots, n\)). It follows from Definition 5 that if the system is in a callback episode at time \(t\), the following holds:

\[
\alpha(t) = \begin{cases} 
\bar{\lambda}(t), & \text{if } \bar{q}(t) > 0, \\
\min(f^{-1}_t(\bar{\lambda}(t) - \mu), \bar{\lambda}(t)), & \text{if } \bar{q}(t) = 0.
\end{cases}
\]

We can further divide the callback episode into several sub-periods based on the online queue length. If the system is empty (over an interval), we say that it is in an empty sub-period of the callback episode. Otherwise, the system in in a busy sub-period of the callback episode. We let \(CE_i\) and \(CB_i\) denote the sets of starting times of the empty and busy sub-periods (in the \(i\)-th callback episode), respectively. To be specific, each sub-period is defined recursively as follows: For \(j \geq 1\),

(i) Let \(s^i_1 = c_i\). If \(s^i_1 = \inf\{t \geq c_i : \bar{q}(t) > 0\}\), then let \(s^i_1 \in CB_i\); otherwise, let \(s^i_1 \in CE_i\).

(ii) If \(s^i_j \in CB_i\), then let \(s^{i,j}_0 = s^i_j\) and define \(s^i_{j+1}\) using the following recursive steps:

(a) \(s^{i,j}_n = \inf\{t \in (s^{i,j}_{n-1}, d_i) : \bar{q}(t) = 0\}\);

(b) If \(s^{i,j}_n \neq \inf\{s \in (s^{i,j}_{n-1}, d_i) : \bar{q}(t) > 0\}\) or \(s^{i,j}_n = d_i\), then \(s^i_{j+1} = s^{i,j}_n\) and \(s^i_{j+1} \in CE_i\); otherwise, let \(n = n + 1\) and repeat step (a).

(iii) If \(s^i_j \in CE_i\), then let \(s^i_{j+1} = \inf\{t \in (s^i_j, d_i) : \bar{q}(t) > 0\}\) and \(s^i_{j+1} \in CB_i\).

(iv) If \(s^i_j = d_i\), then stop and let \(k_i = j - 1\).

The number \(k_i\) is defined as the number of sub-periods within the \(i\)-th callback episode \([c_i, d_i]\). Figure 11 provides an illustrative example of how the sub-periods are defined. Each empty sub-period defines the interval when the online queue is empty. Each busy sub-period defines the block
of consecutive busy periods of the online queue. Step (ii) of the preceding definition defines the end of a busy sub-period by defining $\tilde{s}_{n}^{i,j}$ recursively. The times $\tilde{s}_{n}^{i,j}$ are the times when the online queue hits zero. Note that the online queue may experience multiple busy periods within a busy sub-period of the callback episode. Step (ii) keeps finding the next time when the online queue hits zero until the online queue stays empty on an interval, i.e. when the system switches to an empty sub-period. Thus, the online queue dynamics within a callback episode can be summarized as follows. The callback episode can start with either a busy or an empty sub-period. Then, the system alternates between the busy and empty sub-periods. The callback episode can end with either a busy or an empty sub-period.

![Figure 11: An illustrative example of how the sub-periods are defined.](image)

During an empty sub-period, the system sets the callback offer rate $\alpha(t)$ at the level where exactly $\bar{\lambda}(t) - \mu$ customers are routed to the offline queue. In particular, this implies that $f_t^{-1}(\bar{\lambda}(t) - \mu) \leq \bar{\lambda}(t)$. Otherwise, the online queue length would increase, contradicting the fact that the system in an empty sub-period. During the busy sub-period, the system offers the callback option to all incoming customers by definition. The online queue changes at rate $\bar{\lambda}(t) - f_t(\bar{\lambda}(t)) - \mu$. The following lemma formalizes this characterization.

**Lemma 10.** The following holds: For $t \in (s_j^i, s_{j+1}^i)$ and $j = 1, \ldots, k_i$, if $s_j^i \in CE_i$, then $0 \leq f_t^{-1}(\bar{\lambda}(t) - \mu) \leq \bar{\lambda}(t)$. Therefore, for $t \in (s_j^i, s_{j+1}^i)$ the callback offer rate $\alpha(t)$ is given as follows:

$$\alpha(t) = \begin{cases} \bar{\lambda}(t), & \text{if } s_j^i \in CB_i, \\ f_t^{-1}(\bar{\lambda}(t) - \mu), & s_j^i \in CE_i. \end{cases}$$

**Proof.** Fix $t \in (s_j^i, s_{j+1}^i)$ for $j = 1, \ldots, k_i$ such that $s_j^i \in CE_i$. Since the system is in an empty sub-period, it must that $\bar{q}^i(t) = 0$. We prove the lemma by contradiction. In particular, we assume that $f_t^{-1}(\bar{\lambda}(t) - \mu) > \bar{\lambda}(t)$. Under this assumption, we have that $\alpha(t) = \bar{\lambda}(t)$. Therefore, the
following holds:

\[
\bar{q}'(t) = \bar{\lambda}(t) - f_t(\bar{\lambda}(t)) - \mu(t) > \bar{\lambda}(t) - (\bar{\lambda}(t) - \mu) - \mu = 0,
\]

where the inequality follows from the contradicting assumption. This contradicts the fact that \(\bar{q}'(t) = 0\). Substituting this into Definition 5 completes the characterization of the callback offer rate.

We end the discussion of the system dynamics during a callback episode with the key observation that the callback offer rate \(\alpha(t)\) (for \(t \in [c_i, d_i]\)) is continuous when the system switches from an empty sub-period to a busy sub-period. Note that it is possible that the callback offer rate jumps when the system switches from a busy sub-period to an empty sub-period.

**Lemma 11.** If \(s_j^i \in CE_i\) and \(s_{j+1}^i \in CB_i\) (for \(j = 1, \ldots, k_i - 1\)), then \(\bar{\lambda}(s_{j+1}^i) - \mu = f_t(\bar{\lambda}(s_{j+1}^i))\). Thus, the callback offer rate \(\alpha(t)\) is continuous at time \(s_{j+1}^i\).

**Proof.** It follows from Lemma 10 that \(\alpha(t) = f_t^{-1}(\bar{\lambda}(t) - \mu)\) for \(t \in (s_j^i, s_{j+1}^i)\). This implies that \(f_t^{-1}(\bar{\lambda}(t) - \mu) \leq \bar{\lambda}(t)\) for \(t \in (s_j^i, s_{j+1}^i)\). Equivalently, we have that

\[
\bar{\lambda}(t) - \mu - f_t(\bar{\lambda}(t)) \leq 0 \quad \text{for } t \in (s_j^i, s_{j+1}^i).
\]

Note that the online queue length starts to increase at time \(s_{j+1}^i\) as the system enters the busy sub-period. Therefore, there exists a small \(\epsilon > 0\) such that

\[
\bar{q}'(t) = \bar{\lambda}(t) - \mu - f_t(\bar{\lambda}(t)) \geq 0 \quad \text{for } t \in (s_{j+1}^i, s_{j+1}^i + \epsilon).
\]

Then by the continuity of \(\bar{\lambda}(t)\) and the acceptance function \(f_t\), it must hold that \(\bar{\lambda}(s_{j+1}^i) - \mu = f_t(\bar{\lambda}(s_{j+1}^i))\).

**The system dynamics within a no-callback episode.** We fix an index \(i = 1, \ldots, n\) and discuss the dynamics within the no-callback episode \([d_i, c_{i+1})\). During the no-callback episode, all incoming customers are routed to the online queue. We divide the no-callback episode into several sub-periods, named as the busy sub-period (within the no-callback episode) and the offline sub-period depending on the online queue length. The online queue is non-empty during a busy sub-period, whereas it is empty during the offline sub-period. Moreover, since the arrival rate must be less than or equal to the service rate during the offline sub-periods (otherwise, the online queue increases), the system uses the excess capacity to serve the offline queue customers. The key
observations of the system dynamics of the no-callback episode, which will be proved below, are summarized next:

- A no-callback episode starts with a busy sub-period.
- A no-callback episode only ends when the online queue is empty.
- The length of each busy sub-period is less than or equal to $p/h$. However, the total length of the first block of the busy sub-periods (which may consist of one busy sub-period of length $p/h$ or multiple consecutive busy sub-periods with no offline sub-periods in between them) must be greater than or equal to $p/h$.
- When the no-callback episode ends and the system switches to a callback episode, the derivative of $\bar{q}'(t)$ changes continuously at the switching time.

We first show that if the no-callback episode follows a callback episode, then the no-callback episode must start with a busy period of the online queue.

**Lemma 12.** If $i \geq 2$ or $c_1 < d_1$ for $i = 1$, then $d_i = \inf\{ t \in [d_i, c_{i+1}) : \bar{q}(t) > 0 \}$.

**Proof.** Let $t_1 = \inf\{ t \in [d_i, c_{i+1}) : \bar{q}(t) > 0 \}$. The proof proceeds by contradiction. Suppose that the no-callback episode starts with a period when the online queue is empty, i.e. $d_i < t_1$. Thus, the online queue is empty during $[d_i, t_1)$, i.e. $\bar{q}(t) = 0$ for $t \in [d_i, t_1)$.

Since the system is in the no-callback episode and the online queue remains empty during $[d_i, t_1)$, it must be that $\bar{\lambda}(t) \leq \mu$ for $t \in [d_i, t_1)$. Otherwise, the online queue length may increase. Recall the assumption that $\bar{\lambda}(t)$ does not identically equal to $\mu$ on any interval. Thus, $\bar{\lambda}(t)$ must by strictly less than $\mu$ on a sub-interval of $[d_i, t_1)$. There must exist $0 < t_2 < \min(t_1, d_i + p/2h)$ that $\Gamma(d_i, t_2) > 0$. Let $\epsilon = \Gamma(d_i, t_2)$, i.e. $\epsilon$ denotes the accumulative excess capacity during $[d_i, t_2)$.

Next, we consider the callback episode right before time $d_i$. Since the functions $\bar{q}(t)$ and $\Gamma(t, d_i)$ are both continuous, $\bar{q}(t) - \Gamma(t, d_i)$ is continuous in $t$ with $\bar{q}(d_i) - \Gamma(d_i, d_i) = 0$. Thus, there exists a small $0 < \delta < p/2h$ such that $\bar{q}(d_i - \delta) - \Gamma(d_i - \delta, d_i) \leq \epsilon$. Combining these two facts gives the following:

$$\bar{q}(d_i - \delta) - \Gamma(d_i - \delta, t_2) = \bar{q}(d_i - \delta) - \Gamma(d_i - \delta, d_i) - \Gamma(d_i, t_2) \leq -\epsilon + \epsilon = 0.$$
Therefore, the system should be in a no-callback episode at time $d_i - \delta$, contradicting the fact that the system is in a callback episode at $d_i - \delta$. Thus, it must be that $d_i = t_1$.

The intuition behind this observation is that if an offline sub-period follows immediately after a callback episode, it would cost less if the system ends the callback episode earlier. Consider the last customer arriving during the callback episode. The system offers him the callback option. However, if the system routes him to the online queue, the online queue is empty and the customer will be served instantly as the system has the excess capacity during an offline sub-period.

We use $B_i$ and $O_i$ to denote the sets of the starting times of the busy and the offline sub-periods in the $i$-th no-callback episode. The busy and the offline sub-periods in the no-callback episode $[d_i, c_i+1)$ are defined recursively as follows: For $j \geq 1$,

(i) Let $t_1^i = d_i$ and $t_1^i \in B_i$.

(ii) If $t_j^i \in B_i$, then $t_{j+1}^i = \inf\{t \in (t_j^i, c_i+1] : \bar{q}(t) = 0\}$. If $t_{j+1}^i = \inf\{t \in (t_{j+1}^i, c_i+1] : \bar{q}(t) > 0\}$, then $t_{j+1}^i \in B_i$; otherwise, let $t_{j+1}^i \in O_i$.

(iii) If $t_j^i \in O_i$, then let $t_{j+1}^i = \inf\{t \in (t_j^i, c_i+1] : \bar{q}(t) > 0\}$ and $t_{j+1}^i \in B_i$.

(iv) If $t_j^i = c_i+1$, then stop and let $m_i = j - 1$.

If the period $[t_0, \bar{T}]$ start with an offline sub-period of a no-callback episode, then we have that $t_1^2 = t_2^2$. In this case, the no-callback episode $[d_i, c_i+1]$ has $m_i - 1$ sub-periods. For all other cases, there are $m_i$ sub-periods within the no-callback episode $[d_i, c_i+1)$. Note that it follows directly from the definition above that a busy sub-period of the no-callback episode can be followed by another busy sub-period of the no-callback episode, whereas a busy sub-period of the callback episode can only be followed by either an empty sub-period or a no-callback episode. The dynamics of the online queue can be summarized as follows. It follows from Lemma 12 that the no-callback episode (if it is not the first one) starts with a busy sub-period of the online queue. Then each busy sub-period is either followed by another busy sub-period or switches to an offline period. The system repeats the busy sub-periods or switches between the busy and offline sub-periods until the no-callback episode ends.

Next, we then show that the no-callback episode only ends when the online queue is empty.
Lemma 13. When the no-callback episode ends and the system switches to a callback episode, the online queue is empty, i.e. \( \bar{q}(c_{i+1}) = 0 \).

Proof. Fix an \( i \). We show that \( \bar{q}(c_{i+1}) = 0 \). If the no-callback episode ends with an offline sub-period, then \( \bar{q}(c_{i+1}) = 0 \) by the continuity of the online queue fluid. Thus, it suffices to show that \( \bar{q}(c_{i+1}) = 0 \) if the no-callback episode ends with a busy sub-period. We show this by contradiction. Suppose that \( \bar{q}(c_{i+1}) > 0 \). Let \( \epsilon > 0 \) be a small constant such that \( [c_{i+1} - \epsilon, c_{i+1}] \) are still in the current busy sub-period. Therefore, \( \bar{q}(t) > 0 \) for \( t \in [c_{i+1} - \epsilon, c_{i+1}] \). This implies that for \( s \in (0, \epsilon) \),

\[
\bar{q}(c_{i+1} - \epsilon) - \Gamma(c_{i+1} - \epsilon, s) = \bar{q}(c_{i+1} - \epsilon + s) > 0.
\]

In addition, the following holds: For \( s \in (\epsilon, p/h] \),

\[
\bar{q}(c_{i+1} - \epsilon) - \Gamma(c_{i+1} - \epsilon, c_{i+1} - \epsilon + s) = \bar{q}(c_{i+1}) - \Gamma(c_{i+1}, c_{i+1} - \epsilon + s) > 0,
\]

where the first equality holds because the system is in the no-callback episode and the inequality holds because the system switches to the callback episode at time \( c_{i+1} \). These two inequalities contradict the fact that the system is still in the no-callback episode at time \( c_{i+1} - \epsilon \). Thus \( \bar{q}(c_{i+1}) = 0 \). \( \square \)

The next lemma characterizes the lengths of the sub-periods.

Lemma 14. The following hold:

(i) The length of each busy period does not exceed \( p/h \), i.e. for \( i = 1, \ldots, n \) and \( j = 1, \ldots, m_i \), if \( t^i_j \in B_i \), then \( t^i_{j+1} - t^i_j \leq p/h \).

(ii) The length of the ending time between a callback episode and a following offline period (with busy periods of the online queue in-between) exceeds \( p/h \). In other words, for \( i = 1, \ldots, n \) and \( j = 1, \ldots, m_i \), if \( t^i_j \in O_i \), then \( t^i_j - d_i = t^i_j - t^i_1 \geq p/h \).

Proof. (i) We prove this by contradiction. Suppose that \( t^i_{j+1} - t^i_j > p/h \). Therefore, the following holds: For \( s \in (0, p/h] \),

\[
\bar{q}(t^i_j) - \Gamma(t^i_j, t^i_j + s) = \bar{q}(t^i_j + s) > 0,
\]

where both equalities follows from the fact that the system is in a busy period of a no-callback episode. Thus, it follows from this inequality and the definition of the fluid lookahead policy that
the system should still be in the callback episode at time $t^i_j$. This contradicts the fact that the system is the no-callback episode at time $t^i_j$.

(ii) We proceed by contradiction. Suppose that there exists $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m_i\}$ such that $t^i_j - d_i < p/h$. Since the system starts an offline period at $t^i_j$, there exists small $\epsilon_1 > 0$ such that $(t^i_j, t^i_j + \epsilon_1)$ are still in the offline sub-period. Therefore $\lambda(t) \leq \mu$ for $t \in (t^i_j, t^i_j + \epsilon_1)$. Letting $\delta_1 = -\Gamma(t^i_j, t^i_j + \epsilon_1)$, we have that $\delta_1 > 0$ because $\lambda(t)$ cannot equal to $\mu$ on the interval $(t^i_j, t^i_j + \epsilon_1)$.

Denoting $M = \sup_{u \in [d_i - \epsilon_2, d_i]} \lambda(u)$, we let $\epsilon_2 > 0$ be a small number such that $\epsilon_2 < \delta_1/M$. Thus, the following holds:

$$
\bar{q}(d_i - \epsilon_2) - \Gamma(d_i - \epsilon_2, t^i_j + \epsilon_1) = \bar{q}(d_i - \epsilon_2) - \Gamma(d_i - \epsilon_2, d_i) - \Gamma(d_i, t^i_j + \epsilon_1) \\
= \bar{q}(d_i - \epsilon_2) + \int_{d_i - \epsilon_2}^{d_i} (\lambda(u) - \bar{\mu}(u) - \alpha(u)) \, du \\
+ \int_{d_i - \epsilon_2}^{d_i} (\bar{\mu}(u) + f(o)(\alpha(u)) - \mu) \, du - \Gamma(d_i, t^i_j + \epsilon_1) \\
\leq \bar{q}(d_i) + M\epsilon_2 - \Gamma(d_i, t^i_j + \epsilon_1) \\
= M\epsilon_2 + \bar{q}(t^i_j) - \Gamma(t^i_j, t^i_j + \epsilon_1) \\
\leq \bar{q}(t^i_j) = 0.
$$

The inequality in the fourth line follows from that $f_o(\bar{\alpha}(u)) \leq \bar{f}_o(\lambda(u)) \leq M$ and $\bar{\mu}(u) \leq \mu$ for $u \in [d_i - \epsilon_2, d_i]$. The equality in the fifth line follows from the fact that $\bar{\alpha}(u) = 0$ and $\bar{\mu}(u) = \mu$ for $u \in (d_i, t^i_j)$. The last inequality follows from the definition of $\epsilon_2$. Since by the contradicting assumption that $t^i_j - d_i < p/h$, we can let $\epsilon_1$ and $\epsilon_2$ be small enough such that

$$
t^i_j + \epsilon_1 - (d_i - \epsilon_2) = (t^i_j - d_i) + \epsilon_1 + \epsilon_2 < p/h.
$$

Thus, it follows from Definition 5 that the system is in the no-callback episode at time $d_i - \epsilon_1$. This contradicts the fact that $d_i$ is the time when the callback episode ends and the system switches to a no-callback episode. Therefore, we must have that $t^i_j - d_i \geq p/h$. \(\square\)

Lastly, we show that the derivative $\bar{q}'(t) = \lambda(t) - \alpha(t) - \mu(t)$ (for $t \in [c_i, d_i]$) is continuous when the system switches from a no-callback episode to a callback episode. However, it is possible that this derivative jumps when the system switches from the callback episode to the no-callback episode because the callback offer rate $\alpha(t)$ can jump to zero at these occurrences.
Lemma 15. When a no-callback episode ends and the system switches to a callback episode, the arrival rate equals to the service capacity, i.e. $\bar{\lambda}(c_{i+1}) = \mu$ for $i = 1, \ldots, n - 1$. Moreover, the derivative $\bar{q}'(t) = \bar{\lambda}(t) - f_1(\alpha(t)) - \mu(t)$ is continuous at the switching time $c_{i+1}$ for $i = 1, \ldots, n - 1$.

Proof. We first fix an $i$ and show that $\bar{\lambda}(c_{i+1}) = \mu$. Let $\delta > 0$ be a small constant such that the system is in the no-callback episode during $[c_{i+1} - \delta, c_{i+1})$ and that the system is in the callback episode during $[c_{i+1}, c_{i+1} + \delta]$. We show that $\bar{\lambda}(c_{i+1}) = \mu$ by showing that $\bar{\lambda}(t) \leq \mu$ for $t \in [c_{i+1} - \delta, c_{i+1})$ and $\bar{\lambda}(t) \geq \mu$ for $t \in [c_{i+1}, c_{i+1} + \delta]$.

We first show that $\bar{\lambda}(t) \leq \mu$ for $t \in [c_{i+1} - \delta, c_{i+1})$. Note that $\bar{q}(c_{i+1}) = 0$ because the system either is in an offline sub-period of the no-callback episode or ends a busy sub-period at time $c_{i+1}$. If the system is in an offline sub-period, it must be that $\bar{\lambda}(t) \leq \mu$ for a small $\delta$ such that $[c_{i+1} - \delta, c_{i+1})$ is in an offline sub-period. If the system ends a busy sub-period at time $c_{i+1}$, it must be that the online queue length decreases to zero at time $c_{i+1}$. Therefore, for $\delta$ small enough, we have that for $t \in [c_{i+1} - \delta, c_{i+1})$,

$$\bar{q}'(t) = \bar{\lambda}(t) - f_1(\alpha(t)) - \mu(t) = \bar{\lambda}(t) - \mu \leq 0,$$

where the second equality follows from $\alpha(t) = 0$ and $\mu(t) = \mu$ because the system is in a busy sub-period in the no-callback episode. Thus, we have that $\bar{\lambda}(t) \leq \mu$ for $t \in [c_{i+1} - \delta, c_{i+1})$.

Next, we show that $\bar{\lambda}(t) \geq \mu$ for $t \in [c_{i+1}, c_{i+1} + \delta]$. The system starts the callback episode either with an empty sub-period or a busy sub-period because $\bar{q}(c_{i+1}) = 0$. If the system starts with the callback episode with a busy sub-period, we have that $\bar{q}'(t) > 0$ for $t \in [c_{i+1}, c_{i+1} + \delta]$ and a small $\delta$. Therefore, the following holds:

$$0 \leq \bar{q}'(t) = \bar{\lambda}(t) - f_1(\alpha(t)) - \mu(t) = \bar{\lambda}(t) - f_1(\alpha(t)) - \mu \leq \bar{\lambda}(t) - \mu.$$

In other words, $\bar{\lambda}(t) \geq \mu$ for $t \in [c_{i+1}, c_{i+1} + \delta]$. If the system starts the callback episode with an empty sub-period, then the call center manager serves the incoming customers at rate $\mu$ and routes $\bar{\lambda}(t) - \mu$ customers to the offline queue by offering the callback episode to $f_1(\bar{\lambda} - \mu)$ customers. This implicitly assumes that $\bar{\lambda}(t) \geq \mu$ for $t \in [c_{i+1}, c_{i+1} + \delta]$.

In sum, we show that $\bar{\lambda}(t) \leq \mu$ for $t \in [c_{i+1} - \delta, c_{i+1})$ and $\bar{\lambda}(t) \geq \mu$ for $t \in [c_{i+1}, c_{i+1} + \delta]$ for a small constant $\delta > 0$. By the continuity of the arrival rate $\bar{\lambda}(\cdot)$, we have that $\bar{\lambda}(c_{i+1}) = \mu$.

We end the proof to show that $\bar{q}'(t)$ is continuous at $c_{i+1}$. Since the system is in the no-callback
episode during \([c_i+1-\delta, c_{i+1})\), we have that \(\bar{q}'(t) = \bar{\lambda}(t) - \mu(t)\). If the no-callback episode ends with a busy sub-period, then the following holds:

\[
\bar{q}'(t) = \bar{\lambda}(t) - \mu(t) = \bar{\lambda}(t) - \mu \to 0 \quad \text{as} \quad t \to c_{i+1}. 
\]

If the no-callback episode ends with an offline sub-period, then \(\bar{q}'(t) = 0\) as \(t \to c_{i+1}^{-}\). In both scenarios, we have that the left limit of \(\bar{q}'(t)\) is zero.

On the other hand, we have to show that the right limit of \(\bar{q}'(t)\) is zero as well. If the callback episode starts with an empty sub-period, the right limit of \(\bar{q}'(t)\) is zero because the online queue does not increase. If the callback episode starts with a busy sub-period, then the following holds:

\[
0 \geq \bar{q}'(t) = \bar{\lambda}(t) - f_t(\bar{\lambda}(t)) - \mu \leq \bar{\lambda}(t) - \mu \to 0 \quad \text{as} \quad t \to c_{i+1}. 
\]

Therefore, we have that the right limit of \(\bar{q}'(t)\) equals to zero as well. This shows that \(\bar{q}'(t)\) is continuous at time \(c_{i+1}\).

The complete characterization of the online queue dynamics. Recall that the current busy period of the system \([t_0, \bar{T}]\) always ends with a no-callback episode; see Definition 5. The following lemma further shows that the last no-callback episode ends with an offline sub-period.

**Lemma 16.** The last no-callback episode ends with an offline period, i.e. \(t_{n_m}^n \in O_n\).

**Proof.** We proceed by contradiction. Suppose the busy period ends with a busy period of the online queue. Thus, when the last busy period of the online queue starts, the system has no offline customers as well as an empty online queue. This contradicts the definition of \(\bar{T}\). Thus, the system must end with an offline period.

Figure 12 shows an example of a given arrival rate in the fluid model. The system starts with a busy sub-period of the callback episode. Then the system switches to a busy sub-period of a no-callback episode when the callback offer rate jumps down to zero. Note that the busy period of the online queue does not end when the busy sub-period of the callback episode ends. Instead, the system switches to a no-callback episode. The system stays in the no-callback episode, consisting of two busy sub-periods and an offline sub-period, then switches to a callback episode. Note that when the system switches from an empty sub-period to a busy sub-period of the callback episode, the callback offer rate changes smoothly; see Lemma 11. However, the callback offer rate jumps downward when the system switches from a busy to an empty sub-period of the callback episode.
This example also shows that a callback episode can end with either a busy or an empty sub-period. In addition, it also shows that a no-callback episode always starts with a busy sub-period block. Moreover, the current busy period of the system always ends with an offline sub-period of the no-callback episode.

Figure 12: The dynamics of $\tilde{q}(\cdot)$ for the given arrival rate $\tilde{\lambda}(\cdot)$. Let $p/h = 2$ minutes $= 120$ seconds. In this example, $t_0 = 0$ and $T = 684$ seconds. The first three panels plot the dynamics of the arrival rate $\tilde{\lambda}(\cdot)$, the callback offer rate $\alpha(\cdot)$ and the rate to serve the online queue $\mu(\cdot)$ under the lookahead policy (with units /min. The last panel plot the resulting dynamics of the online queue.

Table 4 summarizes all possible switching patterns between different sub-periods, i.e. the busy and empty sub-periods with the callback episodes and the busy and offline sub-periods within the no-callback episodes.

We end this subsection by another observation of the online queue dynamics. Let $C$ denote a callback episode, which may constitute of multiple empty and busy sub-periods. If we take the starting time of each busy sub-period of the no-callback episode as a “regeneration” point, there are four possible paths to reach the next regeneration point:

(i) (B)B... The current busy sub-period of the no-callback episode is followed immediately by another busy sub-period of the no-callback episode.
Table 4: The possible switching patterns between different sub-periods. The sub-period $a$ in each row can be followed by a sub-period $b$ in each column. If the value of the cell is NA, then sub-period $a$ cannot be followed by sub-period $b$. If the value of the cell is $*$ or $**$, then sub-period $a$ can be followed by sub-period $b$. The derivative of $\bar{q}(\cdot)$ may jump when the system switches from sub-period $a$ to $b$. If the value of the cell is $**$, then sub-period $a$ can be followed by sub-period $b$ where the derivative of $\bar{q}(\cdot)$ changes continuously; see Lemmas 11 and 15.

<table>
<thead>
<tr>
<th></th>
<th>CB</th>
<th>CE</th>
<th>B</th>
<th>O</th>
</tr>
</thead>
<tbody>
<tr>
<td>CB</td>
<td>N/A</td>
<td>*</td>
<td>*</td>
<td>N/A</td>
</tr>
<tr>
<td>CE</td>
<td>**</td>
<td>N/A</td>
<td>*</td>
<td>N/A</td>
</tr>
<tr>
<td>B</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>*</td>
</tr>
<tr>
<td>O</td>
<td>**</td>
<td>**</td>
<td>**</td>
<td>N/A</td>
</tr>
</tbody>
</table>

(ii) (BC)B... The current busy sub-period of the no-callback episode is followed by a callback episode. The callback episode then is followed by a new busy sub-period of the no-callback episode.

(iii) (BO)B... The current busy sub-period of the no-callback episode is followed by an offline sub-period of the no-callback episode. This offline sub-period of the no-callback episode is then followed by a busy sub-period of the no-callback episode.

(iv) (BOC)B... The current busy sub-period of the no-callback episode is followed by an offline period of the no-callback episode. This offline period is then followed by a callback episode, which is followed by a busy sub-period of a new no-callback episode.

We can view these four possible paths as the building blocks to characterize the dynamics of the online queue. The system starts with either a callback or a no-callback episode. If it starts with a callback episode, it switches to a no-callback episode at time $d_1$. Once the system enters the no-callback episode, the dynamics of the online queue can be viewed as the combinations (with repetition) of the four types of building blocks. In addition, the period $[t_0, \bar{T}]$ always ends with a no-callback episode, which ends with the building block BO.

C.2 Proof of Proposition 2

This subsection proves Proposition 2, i.e. the optimality of the fluid lookahead policy, using the characterization of the online queue dynamics under the fluid lookahead policy provided in Section C.1.
Note that the callback offer rate \( \alpha(\cdot) \) only enters the optimal control problem \((P)\) through the acceptance function \( f_t(\cdot) \). To simplify the notation, we let \( \bar{\alpha}(t) = f_t(\tilde{\alpha}(t)) \) denote the actual rate of customers who are routed to the offline queue at time \( t \). Note that \( \bar{\alpha}(t) \in [0, f_t(\bar{\lambda}(t))] \) for \( t \in [t_0, \bar{T}] \).

In addition, recall that \( \bar{\alpha}(t) \) and \( \bar{\mu}(t) \) are restricted to be piecewise continuous functions. Thus, \( \bar{\alpha}(t) \) is piecewise continuous as well because we assume \( f_t(\tilde{\alpha}) \) to be continuous in both \( t \) and \( \tilde{\alpha} \). Letting \( \tilde{\pi} = \{ (\bar{\alpha}(t), \bar{\mu}(t)), t \in [t_0, \bar{T}] \} \) be an admissible policy such that \( \bar{\alpha}(t) \in [0, f_t(\bar{\lambda}(t))] \) and \( \bar{\mu}(t) \in [0, \mu] \) and \( \tilde{\Pi} \) be the set of all such admissible policies, we can write the optimal control problem \((P)\) equivalently as follows:

\[
\min_{\tilde{\pi} \in \tilde{\Pi}} \int_{t_0}^{\bar{T}} (h\bar{q}(t) + p\bar{\alpha}(t)) \, dt,
\]

subject to \( \bar{q}'(t) = \bar{\lambda}(t) - \bar{\alpha}(t) - \bar{\mu}(t), t \in [t_0, \bar{T}], \)

\( \bar{q}(t) \geq 0, t \in [t_0, \bar{T}], \)

\( \bar{q}(t_0) = \bar{q}_0 \) and \( \bar{q}(\bar{T}) = 0. \)

After obtaining the optimal solution \( \{ \bar{\alpha}(t) : t \in [t_0, \bar{T}] \} \), we calculate the optimal callback offer rate by inverting the acceptance function, i.e. \( \bar{\alpha}^*(t) = f_t^{-1}(\tilde{\alpha}^*(t)) \) for \( t \in [t_0, \bar{T}] \).

To facilitate the analysis to follow, we let \( H(\bar{q}, \bar{\alpha}, \bar{\mu}, l, t) \) denote the Hamiltonian of the optimal control problem \((\tilde{P})\), given as follows:

\[
H(\bar{q}, \bar{\alpha}, \bar{\mu}, l, t) = -(hq + p\bar{\alpha}) + l(\bar{\lambda}(t) - \bar{\alpha} - \bar{\mu}).
\]

Note that we write \( -(hq + p\bar{\alpha}) \) in the Hamiltonian because \((\tilde{P})\) is a minimization problem instead of a maximization one. The following proposition provides a sufficient condition for verifying the optimality of a given policy for the optimal control problem \((\tilde{P})\).

**Proposition 4.** Let \( \tilde{\pi}^* \in \tilde{\Pi} \) be an admissible policy of problem \((\tilde{P})\). In addition, let \( \bar{q}^*(t) \) \( (t \in [t_0, \bar{T}] \) be the resulting online queue length process under the policy \( \tilde{\pi}^* \). Let \( l(t) \) \( (t \in [t_0, \bar{T}] \) be a piecewise continuous and piecewise continuously differentiable function with kinks or jumps at times \( \tau_1, \ldots, \tau_k \). In addition, let \( m(t) \) \( (t \in [t_0, \bar{T}] \) be a nonnegative and piecewise continuous function.

\[ ^{22}\text{The function } l(t) \text{ may or may not have jumps at } \tau_1, \ldots, \tau_k. \text{ If it jumps at } \tau_i \text{ (for } i = 1, \ldots, k), \text{ the condition (41) has to be satisfied.} \]
function and $\beta_1, \ldots, \beta_k$ be non-negative constants. If the functions $\tilde{\alpha}^*(t), \tilde{\mu}^*(t), \tilde{q}^*(t), l(t), m(t)$ and the constants $\beta_1, \ldots, \beta_k$ satisfy the following conditions:

$$H(q^*(t), \tilde{\alpha}^*(t), \tilde{\mu}^*(t), l(t), t) = \max_{\tilde{\alpha} \in [0, f_1(\lambda(t))], \tilde{\mu} \in [0, \mu]} H(q^*(t), \tilde{\alpha}, \tilde{\mu}, l(t), t), \quad t \in [t_0, T],$$  \hspace{1cm} (37)

$$m(t) = 0 \text{ if } \tilde{q}^*(t) > 0, \quad t \in [t_0, T],$$  \hspace{1cm} (38)

$$l'(t) = h - m(t), \quad t \in [t_0, T]\backslash\{\tau_1, \ldots, \tau_k\},$$  \hspace{1cm} (39)

$$l(\tau_i -) - l(\tau_i) = \beta_i, \quad i = 1, \ldots, k,$$  \hspace{1cm} (40)

if $\beta_i > 0$, then $\bar{q}(\tau_i) = 0$ and $\tilde{\lambda}(t) - \bar{\mu}(t) - \tilde{\alpha}(t)$ is continuous at $\tau_i, \quad i = 1, \ldots, k,$  \hspace{1cm} (41)

then $\tilde{\pi}^*$ is the optimal policy for $(\tilde{P})$.

**Proof.** We follow Theorem 1 of Chapter 5 of Seierstad and Sydsaeter (1987) to establish the sufficient condition. Seierstad and Sydsaeter (1987) consider a problem in which both the state variables and control variables are multi-dimensional. Since the state variable in our problem is one-dimensional, we simplify the original problem state in Seierstad and Sydsaeter (1987) and state the problem here. Consider finding the piecewise continuous control $u(t) = (u_1(t), u_2(t)) \in U_t$, where $U_t$ is a fixed set for $t \in [t_0, t_1]$, to optimize the following problem:

$$\max_u \int_{t_0}^{t_1} f_0(x(t), u_1(t), u_2(t), t) \, dt,$$

s.t. $x'(t) = f_1(x(t), u_1(t), u_2(t), t)$

$g(x(t), t) \geq 0, \quad j = 1, \ldots, s,$

$u(t) \in U_t, \quad x(t_0) = x_0$ and $x(t_1) = x_1.$

The sufficient condition provided in Seierstad and Sydsaeter (1987) in Theorem 1 of Chapter 5 as well as note 1 is re-stated here.

**Theorem 1.** (Chapter 5 of Seierstad and Sydsaeter (1987)) Let $(x^*(t), u^*(t))$ be an admissible pair for the optimal control problem. Assume that there exists a function $p(t)$, which is piecewise continuous and piecewise continuously differentiable with (potential) jump discontinuities at $t_0 < \tau_1 < \ldots < \tau_N \leq t_1$. Assume further that there exist a piecewise continuous function $\gamma(t)$ and
numbers $\beta_1, \ldots, \beta_N$ such that the following conditions are satisfied: For $t \in [t_0, t_1)$,

\begin{align}
    u^*(t) &\text{ maximizes } f_0(x^*(t), u_1, u_2, t) + p(t) f_1(x^*(t), u_1, u_2, t) \text{ for } u \in U_t, \quad (42) \\
    m(t) &\geq 0 \text{ and } m(t) = 0 \text{ if } g(x^*(t), t) > 0, \ t \in [t_0, \bar{T}], \quad (43) \\
    p'(t) &= -\frac{\partial f_0(x^*(t), u(t), t)}{\partial x} - p(t) \frac{\partial f_1(x^*(t), u(t), t)}{\partial x} - m(t) \frac{\partial g(x^*(t), t)}{\partial x}, \ t \in [t_0, \bar{T}] \setminus \{\tau_1, \ldots, \tau_N\}, \quad (44) \\
    p(\tau_i) - p(\tau_i) &= \beta_i \frac{\partial g(x^*(t), t)}{\partial x}, \ i = 1, \ldots, N, \quad (45) \\
    \beta_i &\geq 0 \text{ and if } \beta_i > 0, \text{ then } g(x^*(\tau_i), x^*(\tau_i)) = 0 \\
    \text{and } \frac{\partial g(x^*(t), t)}{\partial x} f_1(x^*(t), u_1^*(t), u_2^*(t), t) \text{ is continuous at } \tau_i, \ i = 1, \ldots, N. \quad (46)
\end{align}

Then $(x^*(t), u^*(t))$ solves the optimal control problem presented.

We first translate the optimal control problem (P) using the notation in Seierstad and Sydsaeter (1987). We then verify that conditions (42)-(46) are satisfied if conditions (37)-(41) are satisfied. Let $x(t) = \tilde{q}(t), u(t) = (\tilde{\alpha}(t), \tilde{\mu}(t)), p(t) = l(t), N = k, f_0(x, u_1, u_2, t) = -(hx + pu_1), f_1(x, u_1, u_2, t) = \tilde{\lambda}(t) - u_1 - u_2$ and $g(x, t) = x$. Therefore, condition (42) is equivalent to

$$(\tilde{\alpha}^*(t), \tilde{\mu}^*(t)) \text{ maximizes } -(h\tilde{q}^*(t) + pu_1) + l(t)(\tilde{\lambda}(t) - u_1 - u_2),$$

which is equivalent to condition (37). Moreover, it follows from condition (38) and the assumption that $m(t)$ is non-negative, that condition (43) is satisfied. In addition, condition (44) is equivalent to

\begin{align}
    l'(t) &= h - m(t), \ t \in [t_0, \bar{T}] \setminus \{\tau_1, \ldots, \tau_N\}, \quad (47)
\end{align}

which is equivalent to condition (39). In addition, it is immediate that conditions (45)-(46) are equivalent to conditions (40)-(41).

The rest of this section shows that the fluid system under the fluid lookahead policy (and the non-idling service policy) solves problem (P) by verifying conditions (37)-(41). To this end, we construct functions $l(t)$ and $m(t)$ and positive constants $\beta_1, \ldots, \beta_k$ and show that they satisfy conditions (37)-(41).

**Construction of the adjoint function $l(t)$ and verification of condition (37).** The adjoint function $l(t)$ ($t \in [t_0, \bar{T}]$) is constructed to characterize the marginal cost of extra customers
added to the online queue at time $t$. If the system is perturbed by adding an infinitesimal amount of extra customers, the call center managers can choose either to offer the callback option (to other customers who arrive to the system) to mitigate the impact of the extra customers or to let the extra customers stay in online queue and incur extra holding costs. Since the call center manager makes the callback offer decisions optimally under the optimal solution of $(\tilde{P})$, the adjoint function $l(t)$ equals to the minimum value resulting from these two choices. In what follows, we define two auxiliary functions $b(\cdot)$ and $e(\cdot)$ to facilitate the characterization of the marginal costs of these two choices. To be specific, we use the function $b(\cdot)$ to characterize the marginal cost of using the callback option to mitigate the impact of the additional customers. We define the function $b(t)$ to be the best time to offer the callback option if extra customers are added to the online queue at time $t$. In what follows, we first construct the function $b(t)$ and then argue why $b(t)$ is the “best time” to offer the callback option. In addition, we define a function $e(\cdot)$ to characterize the marginal cost if the call center manager does nothing and incurs extra holding cost due to the additional customers. The function $e(\cdot)$ helps characterize the time when the extra customers can be served.

First, for $t \in [t_0, \bar{T})$, we let $b(t)$ denote the best time the call center manager can offer the callback option to more customers if extra customers are added to the online queue at time $t$. If the online queue is empty at time $t$, then the system is either in an offline sub-period (in a no-callback episode) or an empty sub-period (in a callback episode) when the system does not offer the callback option to all arriving customers. (Recall that we cannot have $\tilde{\lambda}(t) - \mu = f_t(\tilde{\lambda}(t))$ identically on any interval by assumption.) In both cases, $b(t) = t$ is the best time to offer the callback option because offering it at time $t$ lowers the online queue length back to zero as soon as the extra customer are added. Moreover, doing it sooner would not help because the online queue is already empty at time $t$.

On the other hand, if the online queue is not empty, then the system is in a busy sub-period of either a callback or a no-callback episode. Recall that a busy sub-period of the no-callback episode can follow another busy sub-period of the no-callback episode, whereas a busy sub-period of the callback episode has to follow an empty sub-period of the callback episode or a no-callback episode, i.e. it cannot follow a busy sub-period of the callback episode by definition. We first consider the case when the system is in a busy sub-period of a callback episode at time $t$. To facilitate the definition of $b(t)$, we note that the beginning of the current sub-period always coincides with the
beginning of a busy period of the online queue; see Table 4 and Lemma 13. In other words, the
current busy sub-period starts either at time $t_0$ or with an empty online queue. Thus, offering the
callback option before the current sub-period will not affect the dynamics of the online queue of the
current sub-period. Therefore, it is sub-optimal to do so because it only adds extra cost for sending
more customers to the offline queue without saving the holding cost of the extra customers.Since
the system is in a busy sub-period of a callback episode, it has already been offering the callback
option to all arriving customers during this sub-period. Thus, the best time to offer the callback
option is the end of the current busy sub-period, as it is the first time when the system can offer the
callback option to more incoming customers. If the system is in a busy sub-period of a no-callback
episode at time $t$, then it either follows a busy sub-period in a callback episode or starts with an
empty online queue. In either case, the best time to offer the callback option is the beginning of
the current busy sub-period of the no-callback episode, because doing it at that time also results
in a holding cost reduction. Moreover, doing it sooner is either infeasible or sub-optimal. To be
specific, if the busy sub-period of the no-callback episode follows a busy sub-period in a callback
episode, it is infeasible to offer the callback option sooner because the system has already offered
the callback option to all customers. If the busy sub-period starts with an empty online queue,
offering the callback option before the current sub-period does not affect the dynamics of the online
queue in the current sub-period because it has already started with an empty online queue. To
summarize, the auxiliary function $b(t)$ (for $t \in [t_0, \bar{T}]$) is given as follows:

$$b(t) = \begin{cases} 
s_{j+1}^i, & \text{if } t \in [s_j^i, s_{j+1}^i) \text{ and } s_j^i \in CB_i \text{ for } i = 1, \ldots, n, j = 1, \ldots, k_i, \\
t, & \text{if } t \in [s_j^i, s_{j+1}^i) \text{ and } s_j^i \in CE_i \text{ for } i = 1, \ldots, n, j = 1, \ldots, k_i, \\
t_j^i, & \text{if } t \in [t_j^i, t_{j+1}^i) \text{ and } t_j^i \in B_i \text{ for } i = 1, \ldots, n, j = 1, \ldots, m_i, \\
t, & \text{if } t \in [t_j^i, t_{j+1}^i) \text{ and } t_j^i \in O_i \text{ for } i = 1, \ldots, n, j = 1, \ldots, m_i. 
\end{cases}$$

(47)

Note from Equation (47) that the function $b(t)$ is piecewise linear with slope 1 or zero and with
jump discontinuities when a busy period of the online queue ends.

Next, we define the second auxiliary function $e(t)$ as the next time (after time $t$) when the
system is in an offline period. Thus, if the system is in an offline period, then the function $e(t)$
equals to $t$. Otherwise, the value of $e(t)$ is the starting time of the next offline period. In other
words, the function \( e(t) \) can be characterized as follows: For \( i = 1, \ldots, m \) and \( t \in [t_{i-1}, t_i) \),

\[
e(t) = \begin{cases} 
  t, & \text{if } t \in [t_i^j, t_{i+1}^j) \text{ and } t_i^j \in O_i \text{ for some } i = 1, \ldots, n, j = 1, \ldots, m_i, \\
  \inf \{ t_i^j \geq t : t_i^j \in O_i, i = 1, \ldots, n, j = 1, \ldots, m_i \}, & \text{otherwise.}
\end{cases}
\]  

(48)

Note that the function \( e(\cdot) \) is piece-wise linear with slope 1 or zero and with jump discontinuities when an offline period ends.

We next construct the adjoint function \( l(\cdot) \) using these two auxiliary functions. Let \( l(\cdot) \) (for \( t \in [t_0, T] \)) be a RCLL function defined as follows:

\[
l(t) = \max \{-p + h(t - b(t)), -h(e(t) - t)\}.
\]  

(49)

The adjoint function \( l(t) \) is the marginal cost of extra customers added to the online queue at time \( t \). In particular, if extra \( \epsilon \) customers are added to the online queue at time \( t \), the system incurs an extra cost of \(-l(t)\epsilon\). Note that the extra \( \epsilon \) customers are not new arrivals but are the small perturbation added to the system state, i.e. the online queue length \( \bar{q}(t) \).

The call center manager can choose whether to use the callback option to mitigate the cost impact of the small perturbation or not. The call center manager can also choose to let the extra \( \epsilon \) customers stay in the online queue. Note that the system allocates its full capacity to the online queue except for the offline periods. Thus, it does not have excess capacity to deplete the perturbation \( \epsilon \) until the next offline period starts. Thus, the cost incurred by the perturbation \( \epsilon \) without using the callback option is \( h(e(t) - t)\epsilon \).

Consequently, the adjoint function \( l(t) \) defined in Equation (49) represents the marginal cost incurred by the perturbation at time \( t \) if the call center manager chooses to offer the callback option optimally in response to the perturbation.

Next, we verify condition (37). To facilitate the verification of this condition, the following lemma provides the characterization of the adjoint function \( l(\cdot) \). In particular, it provides the value of the adjoint function \( l(\cdot) \) in different sub-periods.

**Lemma 17.** The following holds:

(i) (The busy sub-period in a callback episode.) If \( t \in [s_i^j, s_{i+1}^j) \) and \( s_i^j \in CB_i \) for \( i = 1, \ldots, n \) and \( j = 1, \ldots, k_i \), then \( l(t) = -p - h(s_{i+1}^j - t) \). In addition, \( l(t) \leq -p \) and \( l'(t) = h \).
(ii) (The empty sub-period in a callback episode.) If $t \in [s^i_j, s^i_{j+1})$ and $s^i_j \in CB_i$ for $i = 1, \ldots, n$ and $j = 1, \ldots, k_i$, then $l(t) = -p$ and $l'(t) = 0$.

(iii) (The busy sub-period of the online queue.) If $t \in [t^i_j, t^i_{j+1})$ and $t^i_j \in B_i$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m_i$, then $l(t) = ht - \min\{p + ht^i_j, he(t^i_j)\}$. In addition, it holds that $l(t) \in [-p, 0]$ and $l'(t) = h$ for $t \in (t^i_j, t^i_{j+1})$.

(iv) (The offline sub-period of the online queue.) If $t \in [t^i_j, t^i_{j+1})$ and $t^i_j \in O_i$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m_i$, then $l(t) = 0$ and $l'(t) = 0$.

**Proof.** (i) If $t \in [s^i_j, s^i_{j+1})$ and $s^i_j \in CB_i$, it follows from Equations (47)-(48) that $l(t) = \max\{-p + h(t - s^i_{j+1}), -h(e(t) - t)\} = ht - \min\{p + hs^i_{j+1}, he(t)\}$. It follows from Lemma 14 that the length of the ending time between a callback episode and the next offline period must exceed $p/h$. Since $s^i_{j+1}$ may or may not be the end of the current callback episode, we have that $e(t) - s^i_{j+1} \geq p/h$. Therefore, $l(t) = -p - h(s^i_{j+1} - t)$. It is immediate that $l'(t) = h$. In addition, since $t \leq s^i_{j+1}$, we have that $l(t) \leq -p$.

(ii) If $t \in [s^i_j, s^i_{j+1})$ and $s^i_j \in CE_i$, it follows from Equations (47)-(48) that $l(t) = \max\{-p, -h(e(t) - t)\}$. It follows from Lemma 14 that $e(t) - t \geq p/h$ because $t \leq d_i$. Therefore, we have that $l(t) = -p$. It is immediate that $l'(t) = 0$.

(iii) If $t \in [t^i_j, t^i_{j+1})$ and $t^i_j \in B_i$ for $i = 1, \ldots, n$ and $j = 1, \ldots, m_i$, it follows from it follows from Equations (47)-(48) that

$$l(t) = \max\{-p + h(t - t^i_j), -h(e(t) - t)\}$$

$$= \max\{-p + h(t - t^i_j), -h(e(t^i_j) - t)\}$$

$$= ht - \min\{p + ht^i_j, he(t^i_j)\},$$

where the second equality follows from the fact that $e(t) = e(t^i_j)$ because the system is in a busy sub-period during $[t^i_j, t^i_{j+1})$. Since the values in the min function are two constants, $l'(t) = h$. We complete the proof of (iii) by showing that $l(t) \in [-p, 0]$. The following holds:

$$l(t) = ht - \min\{p + ht^i_j, he(t^i_j)\} \geq ht - p - ht^i_j = -p + h(t - t^i_j) \geq -p,$$

where the last inequality holds because $t \geq t^i_j$. In addition, the following holds:

$$h(t - e(t^i_j)) = h(t - e(t)) \leq 0 \quad \text{and} \quad -p + h(t - t^i_j) \leq 0,$$

74
where the first inequality holds because \( t \leq e(t) \) by definition and the second inequality follows from (i) of Lemma 14. Thus, we have that \( l(t) \leq 0 \).

(iv) If \( t \in [t^i_j, t^i_{j+1}] \) and \( t^i_j \in O_i \) for \( i = 1, \ldots, n \) and \( j = 1, \ldots, m_i \), it follows from Equations (47)-(48) that \( l(t) = \max\{-p, 0\} = 0 \). Thus, it is immediate that \( l'(t) = 0 \).

Next we show that condition (37) holds. The following table summarizes the values of the adjoint function and the lookahead policy for various period. Note that the Hamiltonian \( H(\cdot) \) is linear in both \( \tilde{\alpha} \) and \( \tilde{\mu} \). Letting \((\tilde{\alpha}^*, \mu^*)\) denote one pair of values that optimize the right-hand side of (37), we have that the following holds: For \( t \in [t_0, \bar{T}] \),

<table>
<thead>
<tr>
<th>Type of period</th>
<th>( l(t) )</th>
<th>( f_t(\alpha(t)) )</th>
<th>( \bar{\mu}(t) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Busy period in a callback episode</td>
<td>((-\infty, -p))</td>
<td>(f_t(\lambda(t)))</td>
<td>(\bar{\mu})</td>
</tr>
<tr>
<td>Empty period in a callback episode</td>
<td>(-p)</td>
<td>(\min(\lambda(t) - \bar{\mu}, f_t(\lambda(t))))</td>
<td>(\bar{\mu})</td>
</tr>
<tr>
<td>Busy period in a no-callback episode</td>
<td>((0, -p))</td>
<td>(0)</td>
<td>(\bar{\mu})</td>
</tr>
<tr>
<td>Offline period in a no-callback episode</td>
<td>(0)</td>
<td>(0)</td>
<td>(\min(\bar{\mu}, \lambda(t)))</td>
</tr>
</tbody>
</table>

Table 5: Summary of the values of the adjoint function \( l(t) \) and the callback and service rates under the fluid lookahead policy.

\[
\begin{align*}
\tilde{\alpha}^* &= 0, & \text{if } l(t) > -p, \\
\tilde{\alpha}^* &\in [0, f_t(\lambda(t))], & \text{if } l(t) = -p, \\
\tilde{\alpha}^* &= f_t(\lambda(t)), & \text{if } l(t) < -p,
\end{align*}
\]

and

\[
\begin{align*}
\mu^* &= 0, & \text{if } l(t) > 0, \\
\mu^* &\in [0, \bar{\mu}], & \text{if } l(t) = 0, \\
\mu^* &= \bar{\mu}, & \text{if } l(t) < 0.
\end{align*}
\]

It follows immediately from Table 5 and Equation (50)-(51) that the callback offer rate and service rate \((\tilde{\alpha}(t), \bar{\mu}(t))\) satisfies condition (37).

**Corollary 3.** Condition (37) holds. In particular, the following holds: For \( t \in [t_0, \bar{T}] \),

\[
H(\bar{q}(t), \tilde{\alpha}(t), \bar{\mu}(t), l(t), t) = \max_{\tilde{\alpha} \in [0, f_t(\lambda(t))], \bar{\mu} \in [0, \bar{\mu}]} H(\bar{q}^*(t), \tilde{\alpha}, \bar{\mu}, l(t), t),
\]

where \( \tilde{\alpha}(t) = f_t(\alpha(t)) \).

**Construction of the auxiliary function** \( m(t) \) **and verification of conditions** (38)-(39).
Lemma 17 characterizes the adjoint function \( l(t) \) in different sub-periods. In particular, the adjoint function \( l(t) \) is a piecewise linear function with kinks or jumps at times when a sub-period ends. Thus, the adjoint function is differentiable except at those kinks and jumps. We define \( \tau_1, \ldots, \tau_k \) be the time when a sub-period ends, i.e. \( c_1 = s_{11}, s_{12}, \ldots, s_{k1} = d_1 = t_{11}, t_{12}, \ldots, t_{m1} = c_2, \ldots, c_{n+1} = \bar{T} \). In addition, we define the function \( m(\cdot) \) as follows: For \( t \in [t_0, \bar{T}] \),

\[
m(t) = - \lim_{s \to t^+} l'(s) + h, \tag{52}
\]

where \( x^+ \) define as the right limit of \( x \). In particular, the function \( m(t) \) is defined using the right limit of the derivative of the adjoint function \( l(t) \). Condition (39) holds automatically by the construction of the auxiliary function \( m(\cdot) \). Note that \( -l'(t) \) is either \(-h\) or zero. Thus, the function \( m(t) \) is non-negative. It follows from (i) and (iii) of Lemma 17 immediately that \( l'(t) \) equals to \( h \) when the online queue is not empty. In other words, \( m(t) = 0 \) if the online queue is not empty. This proves that condition (38) holds.

**Corollary 4.** If \( \bar{q}(t) > 0 \), then \( m(t) = 0 \), i.e. condition (38) holds.

**Jump discontinuities and verification of conditions** (40)-(41). Since the adjoint function \( l(\cdot) \) is piecewise linear with kinks and jumps only when each sub-period ends, we restrict our analysis to the switching times \( \tau_1, \ldots, \tau_k \) of the sub-periods. Figure 13 summarizes all scenarios of how the adjoint function \( l(t) \) changes as the system switches between different sub-periods.

Let \( \beta_1, \ldots, \beta_k \) be the constants defined using Equation (40). It follows immediately from Figure 13 that these constants \( \beta_1, \ldots, \beta_k \) are either zero or strictly positive. The following table gives indices to the seven cases when the jump discontinuities occur in Figure 13. By verifying the online

<table>
<thead>
<tr>
<th>Index</th>
<th>Sub-period Switching Scenario</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>CE ( \rightarrow ) CB</td>
</tr>
<tr>
<td>II</td>
<td>B ( \rightarrow ) CB</td>
</tr>
<tr>
<td>III</td>
<td>B ( \rightarrow ) CE</td>
</tr>
<tr>
<td>IV</td>
<td>B ( \rightarrow ) B</td>
</tr>
<tr>
<td>V</td>
<td>O ( \rightarrow ) CB</td>
</tr>
<tr>
<td>VI</td>
<td>O ( \rightarrow ) CE</td>
</tr>
<tr>
<td>VII</td>
<td>O ( \rightarrow ) B</td>
</tr>
</tbody>
</table>

Table 6: The indices of the scenarios when the adjoint function \( l(t) \) has jump discontinuities.
$i = 1, \ldots, k$) is strictly positive, then the first part of condition (41) holds.

**Corollary 5.** If $\beta_i > 0$, then $\bar{q}(\tau_i) = 0$ for $i = 1, \ldots, k$.

**Proof.** Note that the system switches from a sub-period when the online queue is empty to another sub-period in cases I, V, VI and VII. Thus, it suffices to verify cases II, III and IV. In both cases II and III, the no-callback episode ends and switches to a callback episode. It follows from Lemma 13 that the online queue is empty when the busy sub-period of the no-callback episode ends. By construction of the busy sub-periods of the no-callback episode, the online queue is empty in case IV.

Lastly, we check the second part of condition (41) that the value of $\bar{\lambda}(t) - \mu(t) - f_t(\alpha(t))$ changes continuously when the adjoint function $l(t)$ jumps. Corollary 6 verifies all seven cases.

**Corollary 6.** If $\beta_i > 0$, then $\bar{\lambda}(t) - \mu(t) - f_t(\alpha(t))$ is continuous at $\tau_i$ for $i = 1, \ldots, k$.

**Proof.** It follows from Lemma 15 that $\bar{q}'(t) = \bar{\lambda}(t) - \mu(t) - f_t(\alpha(t))$ for cases II, III, V and VI. Thus, it suffices to show that it is continuous in cases I, IV and VII.

Case I. Note that the service rate $\mu(t) = \mu$ in both the empty and busy sub-periods of the callback episode. It follows from Lemma 11 that the callback offer rate $\alpha(t)$ is continuous when
the system switches from an empty sub-period to a busy sub-period of the callback episode. Thus, it follows from the continuity of \( f_t(\cdot) \) and \( \bar{\lambda}(\cdot) \) that \( \bar{\lambda}(t) - \mu(t) - f_t(\alpha(t)) \) is continuous.

Case IV. Note that \( \mu(t) = \mu \) and \( \alpha(t) = f_t(\alpha(t)) = 0 \) in both busy sub-periods of the no-callback episode. Therefore, it follows from the continuity of the arrival rate \( \bar{\lambda}(\cdot) \) that \( \bar{\lambda}(t) - \mu(t) - f_t(\alpha(t)) \) is continuous.

Case VII. Note that \( \alpha(t) = 0 \) in the no-callback episode. Thus, it suffices to show that \( \bar{\lambda}(t) - \mu(t) \) is continuous. Note that \( \bar{\lambda}(t) - \mu(t) = 0 \) and \( \bar{\lambda}(t) \leq \mu \) when the system is in the offline sub-period. Thus, the left limit of \( \bar{\lambda}(t) - \mu(t) \) at the switching time is zero. When the system switching to a busy sub-period from an offline sub-period, it must be that \( \bar{q}'(t) = \bar{\lambda}(t) - \mu(t) = \bar{\lambda}(t) - \mu \geq 0 \). This implies that \( \lambda(t) \geq \mu \) at the switching time. Therefore, we have that \( \lambda(t) = \mu \) at the switching time, implying that the right limit of \( \lambda(t) - \mu \) at the switching time is zero as well. Thus, we have that \( \bar{\lambda}(t) - \mu(t) - f_t(\alpha(t)) \) is continuous at the switching time.

It follows from Corollaries 3-6 that the fluid lookahead policy solves problem (P).

Figure 14 shows the derivate \( \bar{q}'(\cdot) \) and the adjoint function \( l(\cdot) \) associated with the example in Figure 12. Either the derivative \( \bar{q}'(\cdot) \) of the online queue or the adjoint function \( l(\cdot) \) jumps when the system switches to a new sub-period. However, the two functions never jump at the same time; see Corollary 6. In addition, the adjoint function can only jump downward.
D Proof of Proposition 3

To facilitate the analysis to follow, we define an auxiliary function $\tilde{q}(\cdot)$ as follows

$$\tilde{q}(t) = \max_{s \in [0, p/h]} \int_{t}^{t+s} (\mu - \bar{\lambda}(u)) \, du = \max_{s \in [0, p/h]} \Gamma(t, t + s).$$

Substituting Equation (24) into the definition of the auxiliary function $\Gamma(t, s)$, we obtain the follows:

$$\Gamma(t_0, t_0 + s) = (\mu - b)s - \frac{\bar{\lambda}_0 - b}{a} \left(1 - e^{-as}\right).$$

In addition, let $\tilde{\Gamma}(s) = \Gamma(t_0, t_0 + s)$. Note that if $\bar{\lambda}_0 \leq b$, then $\tilde{\Gamma}'(s) = (\mu - b) - (\bar{\lambda}_0 - b)e^{-as} \geq 0.$

Thus, $\tilde{q}(t_0) = \tilde{\Gamma}(p/h)$. If $\bar{\lambda}_0 > b$, then $\tilde{\Gamma}''(s) = a(\bar{\lambda}_0 - b)\exp(-as) > 0$, i.e. the function $\tilde{\Gamma}(s)$ is a strictly convex function. In both cases, the function $\tilde{q}(t)$ is simplified to the following:

$$\tilde{q}(t_0) = \max \{0, \tilde{\Gamma}(p/h)\} = \max \left\{0, (\mu - b)p/h - \frac{\bar{\lambda}_0 - b}{a} \left(1 - e^{-ap/h}\right)\right\}. \quad (53)$$

To characterize the fluid lookahead policy at time $t_0$, we consider the following two: $\tilde{\Gamma}(p/h) \geq 0$ and $\tilde{\Gamma}(p/h) < 0.

**Case 1:** $\tilde{\Gamma}(p/h) \geq 0$. In particular, we have that $\tilde{q}(t_0) = \tilde{\Gamma}(p/h)$. Thus, if $\tilde{q}(t_0) > \tilde{\Gamma}(p/h) = \tilde{q}(t_0)$, then it is immediate that $\tilde{q}(t_0) - \tilde{\Gamma}(s) \geq \tilde{q}(t_0) - \tilde{q}(t_0) > 0$ for all $s \in (0, p/h]$. Thus, it follows from Definition 5 the system is in a callback episode. If $\tilde{q}(t_0) \leq \tilde{\Gamma}(p/h)$, then it is immediate that $\tilde{q}(t_0) - \tilde{\Gamma}(p/h) \leq 0$ holds, i.e. the system is in a no-callback episode. In sum, the system is in the no-callback episode if and only if $\tilde{q}_0 > \tilde{\Gamma}(p/h).

**Case 2:** $\tilde{\Gamma}(p/h) < 0$. Note that the condition that $\tilde{q}_0 > f(p/h)$ is always satisfied in this case.

In this case, we have that $\tilde{q}(t_0) = 0$. In addition, $\tilde{q}(t_0) = 0 > \tilde{\Gamma}(s)$ for all $s \in (0, p/h]$ because the maximum of the function $\tilde{\Gamma}(s)$ over $[0, p/h]$ is achieved at zero and that $\tilde{\Gamma}(\cdot)$ is a strictly convex function. Thus, we have that $\tilde{q}_0 \geq 0 > \tilde{\Gamma}(s)$ for $s \in (0, p/h]$. In this case, the system is always in the callback episode, regardless of the value of the online queue length $\tilde{q}_0$.

The discussion of both cases shows that the system offer the callback option if and only $\tilde{q}_0 > \tilde{\Gamma}(p/h)$. By substituting the explicit value of $\tilde{\Gamma}(p/h)$ and rearranging the terms, we obtain the following condition: The system offers the callback option to customers arriving at time $t_0$ if and only if

$$\tilde{q}_0 + \frac{1 - e^{-ap/h}}{a} \bar{\lambda}_0 > \left(\frac{(\mu - b)p}{h} + b \frac{(1 - e^{-ap/h})}{a}\right).$$
Extending the condition to a general time $t$ completes the proof.

\section{Supplementary Numerical Results}

This section provides supplementary results for the analysis in Section 6.

\subsection{Solving the optimal non-anticipating policy}

This subsection constructs the associated Markov decision process (MDP) to compute the optimal non-anticipating callback policy. By imposing the Markovian assumption, the dynamics of the online queue can be characterized by a continuous time Markov chain, with the two-dimensional state descriptor: the online queue length and current arrival rate. The online queue length takes values in non-negative integers whereas the arrival rate can takes values of all non-negative real numbers. To formulate a MDP that can be solved numerically, we take two major steps: discretizing the time by uniformization and discretizing the state space of the arrival rate process.

\textbf{Uniformization of the continuous time Markov chain.} We first convert the continuous-time MDP into a discrete-time MDP by uniformization. In the queueing system with fixed arrival and service rates, the uniformization rate is usually the sum of the arrival and service rates. In our problem, the arrival rate process follows a Markov process. Prior to the uniformization, we first truncate the arrival rate process and restrict the arrival rate to live on the space of $[0, \lambda_M]$, where the constant $\lambda_M$ takes a large value such that the probability of the stationary distribution of the arrival rate process greater than $\lambda_M$ is negligible. Thus, we assume that the uniformization rate of the process is $\lambda_M + \mu$. In particular, we assume that all events happen at the discrete decision epochs $T_k$ (for $k \geq 1$) and the time difference between two discrete decision epochs follows an exponential distribution with rate $\lambda_M + \mu$.

\textbf{Discretizing the state space of the arrival rate process.} Recall that the arrival rate process follows the SDE in Equation (23) and lives in the space of $[0, \infty)$. We then discretize the interval $[0, \lambda_M]$. To be specific, we assume that the discretized arrival rate process, denote by $\hat{\lambda}(k)$ for $k \geq 1$, can only take value of $0, \delta_\lambda, \ldots, M_\lambda \delta_\lambda$ where $M_\lambda \delta_\lambda = \lambda_M$. We let $\delta_\lambda = 0.01$ in the computation.

Next, we compute the transition matrix of the discretized arrival rate process. The distribution of future values of a CIR process given that $\lambda(t) = \lambda_0$ can be computed in closed form, which is...
given as follows: For \( s \geq 0 \),

\[
\lambda(t + s) \sim Y_s/2c_s. 
\] (54)

where \( c_s = 2a/(1 - e^{-as})\sigma^2 \) and \( Y_s \) is a non-central Chi-squared distribution with the degree of freedom equaling \( 4ab/\sigma \) and non-centrality parameter \( 2c_s\lambda_0e^{-as} \). We use this closed form characterization to compute the transition matrix of the discretized arrival rate process. Recall that after the uniformization, the time between two discrete decision epochs in the discrete-time MDP follows an exponential distribution with rate \( \lambda M + \mu \). Thus, the transition probability of the discretized arrival rate process, denoted by \( \tilde{p}(i,j) \), is given as follows: For \( k = 1, 2, \ldots, i = 0, \ldots, M_\lambda \) and \( j = 0, \ldots, M_\lambda - 1 \),

\[
\tilde{p}(i,j) = \Pr(\tilde{\lambda}(T_k) = i\delta_\lambda, \tilde{\lambda}(T_{k+1}) = j\delta_\lambda)
\]

\[
= \int_0^\infty \Pr(\lambda(T_k + s) \in [(j - 0.5)\delta_\lambda, (j + 0.5)\delta_\lambda]|\lambda(T_k) = i\delta_\lambda) \exp(-(\lambda M + \mu)s) \, ds
\]

and

\[
\Pr(\tilde{\lambda}(T_k) = i\delta_\lambda, \tilde{\lambda}(T_{k+1}) = M_\lambda\delta_\lambda)
\]

\[
= \int_0^\infty \Pr(\lambda(T_k + s) \in [(M_\lambda - 0.5)\delta_\lambda, \infty)|\lambda(T_k) = i\delta_\lambda) \exp(-(\lambda M + \mu)s) \, ds.
\]

Since this transition matrix is independent of the callback decisions, we pre-compute it before solving the optimization problem to compute the optimal callback policy.

**State transition probabilities.** After discretizing the time by uniformization and the state space of the arrival rate process, we construct a discrete-time MDP which can be described as follows:

- **State space:** \((q, \tilde{\lambda}, a)\) where \( q \in \{0, 1, \ldots\} \), \( \tilde{\lambda} \in \{0, \delta_\lambda, \ldots, M_\lambda\delta_\lambda\} \) and \( a \in \{0, 1\} \). The first dimension of the state \( q \) denotes the online queue length (right before the potential customer joining the online queue), while the second dimension of the state \( \tilde{\lambda} \) denotes the current arrival rate. The last dimension \( a \) denotes whether the type of event happens at the current decision epoch is an arrival or not. If there is a new arrival at the current decision epoch, we let \( a = 1 \); otherwise, we let \( a = 0 \). The call center manager only makes the callback decision.

- **Action space:** \( u(q, \tilde{\lambda}, a) \in \{0, 1\} \), where \( u = 0 \) if the call center manager lets the incoming customer join the online queue, whereas \( u = 1 \) if the call center manager offers the callback.
option to the incoming customer and routes him to the offline queue. Since the callback decision is made only when there is an incoming customer, we let \( u(q, \tilde{\lambda}, a) = 0 \) if \( a = 0 \).

Figure 15 characterizes the state-transition probabilities of the MDP. We then solve the optimal callback policy by solving the associated linear programming problem of this discrete-time MDP; see Chapter 5.5 of Bertsekas (2012).

**E.2 Supplementary numerical results in Section 6.1**

**The range of \( p/h \) values.** In the simulation studies in Section 6.1, we choose the range of \( p/h \in \{5, 10, 20, 50\} \) to study a broad spectrum of call center performance. In particular, we are interested in the trade-off between the performance of the online queue and the fraction of customers routed to the offline queue. Note that as the value of \( p/h \) decreases, more customers are routed to the offline queue. Meanwhile, the performance of the online queue improves as a result of routing more customers to the offline queue. Thus, choosing the \( p/h \) value is equivalent to finding the tradeoff between the fraction of offline customers and the performance of the online queue. In the simulation studies in Section 6.1, we choose the range of \( p/h \in \{5, 10, 20, 50\} \) to study a broad spectrum of call center performance as shown in Figure 3. Note that the service rate was normalized to \( \mu = 1 \) in Section 6.1. Consequently, \( p/h \) can be interpreted as the expected number of service completions in \( p/h \) time units.
The relative differences of the costs under the LH, MDP and line policies. The following table shows the relative differences of the costs under the LH, MDP and line policies. Note that the line policy still performs well compared to the MDP policy.

<table>
<thead>
<tr>
<th>% of rejection</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost diff.</td>
<td>23.22%</td>
<td>22.58%</td>
<td>21.73%</td>
<td>20.65%</td>
<td>18.57%</td>
<td>15.46%</td>
<td>10.79%</td>
<td>5.63%</td>
<td>1.61%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>% of rejection</th>
<th>10%</th>
<th>20%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
<th>60%</th>
<th>70%</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost diff.</td>
<td>0.56%</td>
<td>0.77%</td>
<td>0.82%</td>
<td>0.75%</td>
<td>0.88%</td>
<td>0.82%</td>
<td>0.84%</td>
<td>0.93%</td>
<td>0.96%</td>
</tr>
</tbody>
</table>

Table 7: The relative differences of the costs under the LH, MDP and line policies.

E.3 The Value of the Callback Option

This section analyzes of the value of the callback option under the lookahead policy. We start with a general framework to characterize the value of the callback option under the lookahead policy. Then we discuss three specific examples and provide characterizations of the value of the callback option for these systems. The three examples include the exact analysis of an M/D/1 queue and the approximate analysis of an M/G/1 queue and a system with deterministic but periodic arrivals. Note that all analysis in this appendix assumes that customers always accept the callback option if they are offered one.

We start the analysis with a general framework to characterize the value of the callback option under the lookahead policy. Recall that the system seeks to minimize the following objective function:

\[
\limsup_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ pA_2(t) + h \int_0^t Q_1(s) \, ds \right].
\]

Instead of analyzing the original characterization of the original objective function, we derive an alternative characterization of it. To be specific, instead of integrating the cost incurred over the time, we sum up the total cost incurred for each customer. Let \( A_2(t) \) denote the set of customers routed to the offline queue up to time \( t \) (Thus, \( A_2(t) = |A_2(t)| \)). When customer \( i \) arrives at time \( t \), he is routed to either the online queue or the offline queue. If she is routed to the offline queue, i.e. \( i \in A_2(t) \), a penalty of \( p \) is incurred. If he is routed to the online queue, i.e. \( i \notin A_2(t) \), then the system incurs a holding cost which is linear in her waiting time \( w_i \). In sum, the objective function
can be written alternatively as follows:

$$\limsup_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ \sum_{i=1}^{A(t)} p \mathbb{I}_{\{i \in A_2(t)\}} + hw_i \left(1 - \mathbb{I}_{\{i \in A_2(t)\}}\right) \right]$$

where $\mathbb{I}_{\{\cdot\}}$ is the indicator function. In addition, this equation can be rewritten in the integral form equivalently as follows:

$$\limsup_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ \int_0^t \left(p \mathbb{I}_{\{A(s) \in A_2(s)\}} + hw_{A(s)} \left(1 - \mathbb{I}_{\{A(s) \in A_2(s)\}}\right)\right) \ dA(s) \right]. \quad (55)$$

Recall that under the $p/h$-lookahead policy, an incoming customer is routed to the offline queue if and only if her waiting time (under LCFS) is greater than $p/h$, i.e.

$$\{A(t) \in A_2(t)\} = \{w_{A(t)} > p/h\}.$$ 

Substituting this into Equation (55), we have that under the optimal lookahead policy, the objective function is given as follows:

$$\limsup_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ \int_0^t \left(p \mathbb{I}_{\{w_{A(s)} > p/h\}} + hw_{A(s)} \left(1 - \mathbb{I}_{\{w_{A(s)} > p/h\}}\right)\right) \ dA(s) \right]. \quad (56)$$

If the system does not offer the callback option, then the offline queue is always empty, i.e. $A_2(t) = 0$ for $t \geq 0$. The value of the callback option is the difference between the long-term average cost of a system offering the callback option and that of a system without the callback option. As we discuss in the complete foresight analysis, the long term average waiting time of the online queue does not depend on the service discipline within the online queue. Thus, we assume that the customers in the system without the callback option will be served in the exact same order as if they are in the system with lookahead policy. Thus, the objective function is given as follows:

$$\limsup_{t \to \infty} \frac{1}{t} \mathbb{E} \left[ \int_0^t hw_{A(s)} \ dA(s) \right]. \quad (57)$$

Subtracting this by Equation (56), we obtain the difference between a system with and without
the callback option as follows:

\[
\limsup_{t \to \infty} \frac{1}{t} h \mathbb{E} \left[ \int_0^t (h w_A(s) - p)^+ \mathbb{1}_{\{w_A(s) > p/h\}} \, dA(s) \right]
\]

\[
= \limsup_{t \to \infty} \frac{1}{t} h \mathbb{E} \left[ \int_0^t (h w_A(s) - p)^+ \, dA(s) \right]
\]

\[
= \limsup_{t \to \infty} \frac{1}{t} h \mathbb{E} \left[ \int_0^t \mathbb{E}[h w_A(s) - p]^+ \, dA(s) \right],
\]

where \( x^+ = \max(x, 0) \) and the last equality follows from Fubini’s Theorem. The rest of this subsection seeks to characterize this value, which facilitates the characterization of the relative value \( \eta^{LH} \) of the callback option, for three specific examples. In particular, we are interested in how the relative value \( \eta^{LH} \) of the callback option under the lookahead policy changes as the variability of the arrival process changes.

**Example 1: M/D/1 queue.** Consider a queueing system in which the arrival process follows a Poisson process with arrival rate \( \lambda \). In addition, let \( \bar{t} \) be the service time of each customers. We assume that the system is underloaded, i.e. \( \rho = \lambda \bar{t} < 1 \). Since the arrival process follows a Poisson process, the variability of the arrival process increases as the traffic intensity increases. Thus, we fix \( \bar{t} \) and study how the relative value \( \eta^{LH} \) of the callback option changes as the traffic intensity increases. Note that the system cost increases with the traffic intensity, naturally increasing the absolute value of the callback option. Thus, for a more meaningful analysis, we focus on the relative value of the callback option.

Note that the waiting time (under LCFS) has the same distribution of the length of the busy period, i.e. \( w_i \sim B \) (for \( i \geq 1 \)), where \( B \) is is the length of the busy period. Thus, the right-hand side of Equation (58) can be simplified as follows:

\[
\limsup_{t \to \infty} \frac{1}{t} h \mathbb{E} \left[ \int_0^t \mathbb{E}[h w_A(s) - p]^+ \, dA(s) \right]
\]

\[
= \limsup_{t \to \infty} \frac{1}{t} h \mathbb{E} \left[ \int_0^t \mathbb{E}[h B_\rho - p]^+ \, dA(s) \right]
\]

\[
= \limsup_{t \to \infty} \frac{\mathbb{E}[A(t)]}{t} \mathbb{E}[h B_\rho - p]^+
\]

\[
= \lambda \mathbb{E}[h B_\rho - p]^+.
\]

where \( B_\rho \) denote the busy period of this M/D/1 queue. Borel (1942) derives the probability of \( n \)
customers served within one busy period, which is given as follows:

\[ p_n = \frac{e^{-n\lambda t} (n\lambda t)^{n-1}}{n!} \quad \text{for} \quad n \geq 1. \tag{60} \]

Since the service times are deterministic, it is easy to see that \( P(B_{\rho} = nt) = p_n \). Thus, it follows from Equations (57)-(58) that the relative value \( \eta^{LH}(\rho) \) (which depends on the traffic intensity \( \rho \)), of the callback option is given as follows:

\[ \eta^{LH}(\rho) = \lambda \frac{E[hB_{\rho} - p]}{E[hB_{\rho}]} = \frac{E[hB_{\rho}] - p}{E[hB_{\rho}]} \]

The following proposition states the monotonicity property of the relative value \( \eta^{LH}(\rho) \).

**Proposition 5.** The relative value of the callback option \( \eta^{LH}(\rho) \) increases in the traffic intensity \( \rho \) for \( \rho \in (0, 1) \).

**Proof.** We first prove the monotonicity property. In other words, we want to prove the the relative value of the callback is increasing in the arrival rate (with the service rate fixed). To be specific, we want to show that if \( 0 < \rho < \rho' < 1 \) where \( \rho' = \lambda' t \), then \( \eta^{LH}(\rho) \leq \eta^{LH}(\rho') \). To facilitate the proof, we use Theorem 1.B.12 of Shaked and Shanthikumar (2007), which is stated as follows:

**Theorem 1.B.12 (Shaked and Shanthikumar (2007))** Let \( X \) and \( Y \) be two independent random variables. The following conditions are equivalent:

(a) \( X \leq_{hr} Y \), where \( \leq_{hr} \) denote smaller than or equal to in the hazard rate order.

(b) \( E[\alpha(X)]E[\beta(Y)] \leq E[\alpha(Y)]E[\beta(X)] \) for all functions \( \alpha \) and \( \beta \) for which the expectations exist and such that \( \beta \) is nonnegative and \( \alpha/\beta \) and \( \beta \) are increasing.

By substituting \( \beta(x) = x \) into Theorem 1.B.12, we obtain that if \( X \leq_{hr} Y \), then the following holds:

\[ \frac{E[\alpha(X)]}{E[X]} \leq \frac{E[\alpha(Y)]}{E[Y]} \tag{61} \]

if \( \alpha(\cdot) \) is an increasing convex function with \( \alpha(0) = 0 \). More specifically, we can substitute \( \alpha(x) = (x - p/h)^+ \) into the equation. Thus, in order to show that \( \eta^{LH}(\rho) \leq \eta^{LH}(\rho') \) for all \( \rho < \rho' < 1 \), it suffices to show that \( B_{\rho} \leq_{hr} B_{\rho'} \) for \( \rho < \rho' \).

Theorem 1.C.1 in Shaked and Shanthikumar (2007) states that \( X \leq_{tr} Y \), then \( X \leq_{hr} Y \) where \( \leq_{tr} \) denote smaller than or equal to in the likelihood ratio order. Thus, we can instead to verify
the condition that $B_\rho \leq_{tr} B_{\rho'}$ for $\rho < \rho'$, i.e. $\lambda < \lambda'$, which is equivalent to show that

$$lr_n = \frac{\mathbb{P}(B_\rho = n\bar{t})}{\mathbb{P}(B_{\rho'} = n\bar{t})}$$
decreases in $n$ for $n \geq 1$;

see Chapter 1.C of Shaked and Shanthikumar (2007). By substituting Equation (60) into the definition of the likelihood ratio $lr_n$, we have the following:

$$lr_n = \frac{\mathbb{P}(B_\rho = n\bar{t})}{\mathbb{P}(B_{\rho'} = n\bar{t})} = e^{n(\lambda' - \lambda)\bar{t}} \left( \frac{\lambda}{\lambda'} \right)^{n-1} \text{ for } n \geq 1. \quad (62)$$

Thus, the following holds:

$$\frac{lr_{n+1}}{lr_n} = e^{(\lambda' - \lambda)\bar{t}} \frac{\lambda}{\lambda'} = \frac{e^{-\lambda t}(\lambda t)}{e^{-\lambda' t}(\lambda' t)} < 1 \text{ for } n \geq 1. \quad (63)$$

The last inequality holds because $e^{-x}x$ is increasing in $x$ for $x \in [0, 1]$ and that $\lambda \bar{t} < \lambda' \bar{t} < 1$. This inequality shows that $lr_n$ is decreasing in $n$, i.e. $B_\rho \leq_{tr} B_{\rho'}$. This completes the proof of the conjecture that $\eta^{LH}(\rho) \leq \eta^{LH}(\rho')$ for $0 < \rho < \rho' < 1$. \qed

**Example 2: M/G/1 Queue in Heavy Traffic.** The analysis of the M/G/1 system uses the waiting time approximation provided in Abate and Whitt (1997). We first summarize the approximation in Abate and Whitt (1997) and then prove a result that is similar to Proposition 5 for the M/G/1 queue. To be specific, we assume that the arrival process of the system follows a Poisson process with the arrival rate $\lambda$. The service times of the customers are i.i.d. with a general distribution, whose cdf is denoted by $G$. We assume that the average service time is $1/\mu$ and denote $m_2(G)$ as the second moment of the random variable with a distribution $G$. We are interested in studying the distribution of the waiting time of a customer under LCFS (or equivalently, the distribution of the busy period) for this system. For this analysis, we fix the mean service time and study how the system evolves as we change the traffic intensity $\rho = \lambda/\mu$ and the second moment $m_2(G)$ of the service time.

To facilitate the analysis to follow, let $F_\rho(\cdot)$ denote the cdf of the waiting time (under LCFS) with the arrival rate $\rho$. In addition, let $F_\rho^c(\cdot)$ denote the complementary cdf. The following theorem (Theorem 2.1 of Abate and Whitt (1997)) provides the characterization of the complementary cdf $F_\rho^c(\cdot)$ in heavy traffic.

**Theorem 2.1.** Abate and Whitt (1997) For each $t > 0$, the following holds:

$$\lim_{\rho \to 1} 2(1 - \rho)^{-1} F_\rho^c(tm_2(G)/(1 - \rho)^2) = h_1(t),$$

87
where

\[ h_1(t) = \frac{2\phi(\sqrt{t})}{\sqrt{t}} - 2(1 - \Phi(\sqrt{t})). \]

It follows from Theorem 2.1 of Abate and Whitt (1997) that an approximation of the complementary cdf \( F_c^\rho(\cdot) \) is given as follows:

\[ F_{c, \rho,m_2}^2(G)(t) \approx \tilde{F}_{c, \rho,m_2}^2(G)(t) = 1 - \rho h_1(t(1 - \rho)^2/m_2(G)). \]

We let \( \tilde{w}_{\rho,m_2}(G) \) denote the random variable with the complementary cdf \( \tilde{F}_{c, \rho,m_2}^2(G)(\cdot) \). Thus, its pdf, denoted by \( \tilde{f}_{\rho,m_2}(G)(\cdot) \) is given as follows:

\[ \tilde{f}_{\rho,m_2}(G)(t) = -\left( \tilde{F}_{c, \rho,m_2}^2(G)(t) \right)' = \sqrt{\frac{m_2(G)}{2t\sqrt{t}}} \exp\left\{ -\frac{(1 - \rho)^2t}{2m_2(G)} \right\}. \quad (64) \]

We use \( \tilde{w}_{\rho,m_2}(G) \) to approximate the waiting time term that appears in Equation (58). Thus, the approximated value of the callback option using the approximated waiting time \( \tilde{w}_\rho \) is simplified as follows:

\[
\begin{align*}
\limsup_{t \to \infty} \frac{1}{t} h \mathbb{E}\left[ \int_0^t \mathbb{E}[h w_A(s) - p]^+ dA(s) \right] \\
\approx \limsup_{t \to \infty} \frac{1}{t} h \mathbb{E}\left[ \int_0^t \mathbb{E}[h \tilde{w}_{\rho,m_2}(G) - p]^+ dA(s) \right] \\
= \limsup_{t \to \infty} \frac{\mathbb{E}[A(t)]}{t} \mathbb{E}[h \tilde{w}_{\rho,m_2}(G) - p]^+ \\
= \lambda \mathbb{E}[h \tilde{w}_{\rho,m_2}(G) - p]^+.
\end{align*}
\]

Furthermore, we define the approximated relative value of the callback option, denoted by \( \tilde{\eta}(\rho, m_2(G)) \) depending on \( \rho \) and \( m_2(G) \), which is given as follows:

\[ \tilde{\eta}(\rho, m_2(G)) = \frac{\mathbb{E}[h \tilde{w}_{\rho,m_2}(G) - p]^+}{\mathbb{E}[h \tilde{w}_{\rho,m_2}(G)]}. \]

The following proposition states the monotonicity property of the approximated relative value \( \tilde{\eta}(\rho, m_2(G)) \) in the traffic intensity \( \rho \) and the second moment \( m_2(G) \) of the service time distribution.

**Proposition 6.** The approximated relative value of the callback option \( \tilde{\eta}(\rho, m_2(G)) \) is increasing in \( \rho \) and \( m_2(G) \) (for \( \rho \in (0,1) \)).

**Proof.** Similar to the proof of Proposition 5, it suffices to prove that the approximated waiting time \( \tilde{w}_{\rho,m_2}(G) \) is increasing in \( \rho \) and \( m_2(G) \) in the likelihood ratio order. To be specific, we first fix
\( m_2(G) \) and show that \( \tilde{w}_{\rho_1,m_2(G)} \leq_{lr} \tilde{w}_{\rho_2,m_2(G)} \) for \( 0 < \rho_1 < \rho_2 < 1 \). This is equivalent to show that for \( 0 < \rho_1 < \rho_2 < 1 \),
\[
\frac{\tilde{f}_{\rho_1,m_2(G)}(t)}{\tilde{f}_{\rho_2,m_2(G)}(t)} \text{ is decreasing in } t.
\]

Substituting \( \rho_1 \) and \( \rho_2 \) into Equation (64), we have that the following holds:
\[
\frac{\tilde{f}_{\rho_1,m_2(G)}(t)}{\tilde{f}_{\rho_2,m_2(G)}(t)} = \exp \left\{ -\left[(1 - \rho_1)^2 - (1 - \rho_2)^2\right]t/2m_2(G) \right\}.
\]

Since \( 1 - \rho_1 > 1 - \rho_2 \), the likelihood ratio of the two random variables is decreasing in \( t \). Thus, we have shown that \( \tilde{w}_{\rho_1,m_2(G)} \leq_{lr} \tilde{w}_{\rho_2,m_2(G)} \).

We next fix \( \rho \) and show that that \( \tilde{w}_{\rho,m_2(G)} \leq_{lr} \tilde{w}_{\rho,m'_2(G)} \) for \( 0 < m_2(G) < m'_2(G) \). This is equivalent to show that for \( 0 < m_2(G) < m'_2(G) \),
\[
\frac{\tilde{f}_{\rho,m_2(G)}(t)}{\tilde{f}_{\rho,m'_2(G)}(t)} \text{ is decreasing in } t.
\]

Substituting \( m_2(G) \) and \( m'_2(G) \) into Equation (64), we have that the following holds:
\[
\frac{\tilde{f}_{\rho,m_2(G)}(t)}{\tilde{f}_{\rho,m'_2(G)}(t)} = \sqrt{m_2(G)}\sqrt{m'_2(G)} \exp \left\{ -(1 - \rho)^2t \left[ \frac{1}{2m_2(G)} - \frac{1}{2m'_2(G)} \right] \right\}.
\]

Since \( 1/m_2(G) > 1/m'_2(G) \), the likelihood ratio of the two random variables is decreasing in \( t \). Thus, we have shown that \( \tilde{w}_{\rho,m_2(G)} \leq_{lr} \tilde{w}_{\rho,m'_2(G)} \). This completes the proof.

**Example 3: A deterministic queue with a periodic arrival rate.** This example studies a fluid model when the arrival rate follows a periodic deterministic function. To be specific, we assume that the arrival rate \( \lambda(t) \) follows: For \( n = 1, 2, \ldots \),
\[
\lambda(t) = \begin{cases} 
\lambda + \sigma, & t \in [nT, nT + 1/2T), \\
\lambda - \sigma, & t \in [nT + 1/2T, (n + 1)T),
\end{cases}
\]
where \( \lambda, \sigma \) and \( T \) are positive constants such that \( \lambda < \mu < \lambda + \sigma \) and \( \sigma \leq \lambda \). In addition, the cumulative number of customers arriving by time \( t \) is given as follows:
\[
A(t) = \int_0^t \lambda(s) \, ds.
\]

It suffices to study one cycle of the system where the queue is empty at time \( t = 0 \). The right-hand side of Equation (58) simplifies to the following:
\[
\frac{1}{T} \int_0^T \lambda(t)(hw(t) - p)^+ \, dt,
\]
where \( w(t) \) is the waiting time (under LCFS) of a customer who arrives at time \( t \), which is given as follows:

\[
w(t) = \begin{cases} 
\frac{\sigma(T-2t)}{\mu+\sigma-\lambda}, & t \in [0, T/2], \\
0, & t \in [T/2, T]. 
\end{cases}
\]

Substituting this into Equation (65), we obtain that the value of the callback option is:

\[
\frac{1}{T} \int_0^T \lambda(t)(hw(t) - p)^+ \, dt = \begin{cases} 
\frac{(\mu+\sigma-\lambda)(\lambda+\sigma-\mu)}{4\sigma T} \left[ \left( \frac{\sigma T}{\mu+\sigma-\lambda} \right)^2 - \left( \frac{\mu}{\sigma} \right)^2 \right], & T > \frac{(\mu+\sigma-\lambda)p}{\sigma h}, \\
0, & \text{otherwise.}
\end{cases}
\]

Since the second case is not interesting, we focus on the case when \( T > (\mu + \sigma - \lambda)p/\sigma h \). Note that the total cost without the callback option can be derived easily and is given as follows:

\[
\frac{1}{T} \int_0^T \lambda(t)hw(t) \, dt = \frac{h\sigma(\lambda + \sigma - \mu)T}{4(\mu + \sigma - \lambda)}. 
\]

Thus, the relative value of the callback option \( \eta^{LH}(\lambda, \sigma) \) depending on \( \lambda \) and \( \sigma \), is given as follows:

\[
\eta^{LH}(\lambda, \sigma)(\lambda, \sigma) = 1 - \left( \frac{(\mu + \sigma - \lambda)p}{\sigma T h} \right)^2.
\]

The following proposition shows the monotonicity property of \( \eta^{LH}(\lambda, \sigma) \) on both \( \lambda \) and \( \sigma \).

**Proposition 7.** The relative value of the callback option \( \eta^{LH}(\lambda, \sigma) \) increases in both the average arrival rate \( \lambda \) and volatility \( \sigma \).

**Proof.** This is easy to show by verifying the derivatives of the function \( \eta^{LH}(\cdot, \cdot) \) with respect to \( \lambda \) and \( \sigma \):

\[
\frac{\partial \eta^{LH}}{\partial \lambda} = \frac{2(\mu + \sigma - \lambda)p^2}{(\sigma T h)^2} > 0,
\]

\[
\frac{\partial \eta^{LH}}{\partial \sigma} = \frac{2(\mu + \sigma - \lambda)(\mu - \lambda)p^2}{(\sigma T h)^2\sigma} > 0.
\]

\[ \square \]
E.4 Statistics and Parameter estimation of the US bank data set

This subsection provides the description of and the key procedure for estimating the primitives of the simulation study from the dataset.

We use the individual call level data of a US bank call center to study the system with the callback option. To be specific, we analyze the call arrival data of brokerage customers in February 2003. To eliminate the day-of-week effect, we focus on those customers who arrive during the peak hours (9am-2pm) in the weekdays and request the service from the agents. There are \( D = 19 \) days of workdays in February 2003. In addition, we pick the time unit, denoted by \( \delta \), to be 10 seconds, i.e. \( \delta = 10 \) seconds. Since we focus on the peak hours (9am - 2pm), we have that there are \( T = 3000 \) time units in each day. The summary statistics for this portion of the data are given in Table 8.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td># of observations</td>
<td>38,392</td>
</tr>
<tr>
<td>Average waiting time (sec)</td>
<td>21.38</td>
</tr>
<tr>
<td>Average service time (sec)</td>
<td>233.94</td>
</tr>
<tr>
<td>% of abandonments</td>
<td>2.77%</td>
</tr>
</tbody>
</table>

Table 8: The summary statistics of the data.

Although we focus on the arrivals during the peak hours in the weekdays, the arrival process still has the hour-of-day effect. To incorporate the hour-of-day effect, we modify the model of the arrival rate process in Equation (23) by scaling the arrival rate process by the hour-of-day effect. To be specific, we assume that the arrival process follows a Poisson process with its intensity follows

\[
\lambda(t) = c(t)x(t), \quad \text{for} \quad t \in [0, T]
\]

where \( c(t) \) (for \( t \geq 0 \)) is a deterministic function and \( x(t) \) is the diffusion process which follows\(^{23}\)

\[
dx(t) = \alpha(1 - x(t)) \, dt + \sigma \sqrt{x(t)} \, dW(t), \quad t \in [0, T].
\]

In each day, the realized diffusion process \( x_d(t) \) (for \( d = 1, \ldots, D \)) draws one sample path from the diffusion process \( x(t) \). We follow Glynn et al. (2018) and assume that the deterministic

\(^{23}\)We normalize the constant \( b \) in Equation (23) to be one, so the long-term mean of the diffusion process \( x(t) \) is one, i.e. \( \lim_{t \to \infty} \mathbb{E}[x(t)] = 1. \)
hour-of-day effect function $c(t)$ is a constant within each 30-minute interval, i.e. for $i = 1, 2, \ldots, 10$,

$$c(t) = \theta_i, \quad \text{for} \quad t \in [(i - 1)\Delta, i\Delta),$$

where $\Delta = 300$. Thus, the parameters to estimate are $\Xi = (a, \sigma, \Theta)$ where $\Theta = (\theta_1, \ldots, \theta_{10})$.

We assume that the arrival process follows a Poisson process with its intensity following a CIR process. We use the Bayesian approach to estimate the parameters which characterize the arrival process, via Markov Chain Monte Carlo (MCMC) method\textsuperscript{24}, i.e. the constants $a$, $b$ and $\sigma$ in Equation (23). The detailed steps to estimate the parameters are provided at the end this section. Next, we calibrate the number of agents to replicate the current performance of the call center without the callback option. We simulate the system and vary the number of agents using the arrival processes with the estimated parameters $a$, $b$ and $\sigma$, the empirical service time and abandonment time distributions\textsuperscript{25}. We use the exact simulation method proposed in Giesecke et al. (2011) for a Poisson process with stochastic intensity to simulate the arrival process. We compare the simulated average waiting time and fraction of abandoning customers with the data and pick the number of agents to be 30.

In the discrete event simulation in Section 6, we use the parameters $a$ and $\sigma$ estimated from the dataset but ignore the hour-of-day effect. Instead, we scale the diffusion process $x(t)$ by the average arrival rate over the peak hours to obtain the arrival rate. We make this assumption because we assume that the number of agents is a constant over the peak hours, which ignores the hour-of-day effect in the staffing level. To be more specific, we simulate the system and vary the number of agents using the arrival processes with the estimated parameters $a$ and $\sigma^2$, the empirical service time and abandonment time distributions. We compare the simulated average waiting time and fraction of abandoning customers with the data and pick the number of agents.

The rest of this section describes the detailed steps to estimate the parameters from the dataset. Note that the observed data is the counts of the incoming customers\textsuperscript{26}. Let $Y_{k,d}$ (for $k = 1, \ldots, T$)

\textsuperscript{24}Zhang (2013) uses the same approach to estimate the parameters for a similar model.

\textsuperscript{25}To be specific, we use the Kaplan-Meier estimate to estimate the abandonment time distribution. As observed in Brown et al. (2005), the Kaplan-Meier estimate may be biased under heavy center. Therefore, we assume that the abandonment time follows the exponential distribution and follow Brown et al. (2005) to use the first quartile of the Kaplan-Meier estimate of the cumulative distribution function to estimate the hazard rate.

\textsuperscript{26}The data set recorded the arrival time of each customer at the accuracy of one second. Therefore, there may be multiple arrivals within one second.
and \( d = 1, \ldots, D \) denote the observed number of arrivals within in the \( k \)-th time unit in day \( d \). Since we assume that the arrival process follows a Poisson process given its intensity \( \lambda(t) \), the following holds: For \( k = 1, \ldots, T \) and \( d = 1, \ldots, D \),

\[
P(Y_{k,d} = j | x_{k,d}, \Xi) = \left( \frac{\int_{k-1}^{k} c(t)x_d(t) \, dt}{j} \right)^j \exp \left( - \int_{k-1}^{k} c(t)x_d(t) \, dt \right) \approx \left( \frac{\theta_{I(k)x_{k,d}}}{j} \right)^j \exp(-\theta_{I(k)x_{k,d}}),
\]

where \( I(k) = i \) if \( c(k) = \theta_i \). The goal is to estimate the parameters \( \Xi \) given the observations \( Y = (Y_{k,d}) \) (for \( k = 1, \ldots, T \) and \( d = 1, \ldots, D \)).

We use a Bayesian approach to estimate the parameters \( \Xi \). To be specific, we assume that the prior distribution of the parameters \( a, b \) and \( \sigma^2 \) follow a Gaussian, Gamma and inverse Gamma distribution, respectively. We pick the specific conjugate prior distributions because their associated posterior distributions are easy to compute. In particular, the posterior distribution of the parameters \( a, b \) and \( \sigma^2 \) follow a Gaussian, Gamma and inverse Gamma distribution, respectively; see Lemma 18. The closed-form characterization of posterior distribution accelerates the computation of estimating the parameters. We first draw the initial samples of the parameters from the prior distributions. We then implement the Gibbs sampler to generate new samples of the parameters until the sampled parameters converge. Lastly, we use the mean of the sampled parameters as their estimates. We then simulate the joint posterior distributions given the observations, i.e. \( P(\Xi | Y) \). In other words, we run a Monte Carlo simulation and sample the parameters \( \Xi \) which follow the posterior distribution \( P(\Xi | Y) \). The following assumption provides the prior distributions of the parameters \( \Xi \).

**Assumption 1.** The prior of \( \Xi \) are given by the follows:

\[
a \sim \mathcal{N}(\mu_a, \sigma^2_a)
\]

\[
\sigma^2 \sim \text{InverseGamma}(c_\sigma, d_\sigma)
\]

\[
\theta_i \sim \text{Gamma}(c_i, d_i), \ i = 1, \ldots, 10,
\]

where \( c_\sigma \) and \( c_i \) are the shape parameters of the inverse Gamma distribution and Gamma dis-
The joint posterior distribution of the parameters $\Xi$ is difficult to compute. Therefore, it is difficult to sample the parameters $\Xi$ from its joint posterior distribution directly. However, the Gibbs sampler provides an iterative approach to sample the joint posterior distribution via sampling the marginal posterior distributions of the parameters. To be specific, the Gibbs sampler draws one sample of the parameters to be estimated in each iteration and update the marginal posterior distribution of each parameter given the current sample. The sequence of the samples constitutes a Markov chain, whose stationary distribution is the joint posterior distribution of the parameters to be estimated; see Gamerman (2006) for more discussion. In the rest of this section, we first derive the marginal posterior distribution of the parameters $\Xi$. Then we describe the specific steps to implement the Gibbs sampler. In addition, we report the estimates of the parameters at the end of this section.

Prior to deriving the marginal posterior distributions of the parameters, we follow Zhang (2013) and approximate the one-step transition distribution of the Markov Chain $x_{1,d}, \ldots, x_{T,d}$ (for $d = 1, \ldots, D$) by the Euler discretization scheme of the stochastic differential equation (66). To be specific, let $f(\cdot|x_k,d,\xi)$ denote the probability density function of the value of $x_{k+1,d}$ given the values of $x_k,d$ and the parameters. Thus, the pdf $f(\cdot)$ can be approximated as follows:

$$f(x_{k+1,d}|x_k,d,\xi) \approx \frac{1}{\sqrt{2\pi \sigma^2 x_k,d}} \exp \left(-\frac{(x_{k+1,d} - a(1 - x_k,d))^2}{2\sigma^2 x_k,d}\right).$$

(68)

From now on, we replace the posterior distributions of $Y_{k,d}|x_k,d,\xi$ and $x_{k+1,d}|x_k,d,\xi$ with the approximation (67)-(68). The following lemma provides the marginal posterior distributions of the parameters $\Xi$. 

94
Lemma 18. If the parameters $\Xi$ follow the distributions given next:

\[
\begin{align*}
  a &\sim \mathcal{N}(\tilde{\mu}_a, \tilde{\sigma}_a^2) \\
  \sigma^2 &\sim \text{InverseGamma}(\tilde{c}_\sigma, \tilde{d}_\sigma) \\
  \theta_i &\sim \Gamma(\tilde{c}_i, \tilde{d}_i), \; i = 1, \ldots, 10,
\end{align*}
\]

then under the approximation (67)-(68), the following holds: For $i = 1, \ldots, 10$,

\[
\begin{align*}
  a|\sigma^2, \Theta, X, Y &\sim \mathcal{N}(B/A, 1/A) \quad (69) \\
  \sigma^2|a, \Theta, X, Y &\sim \text{InverseGamma}(\tilde{c}_\sigma+D(T-1)/2, \tilde{d}_\sigma), \quad (70) \\
  \theta_i|a, \sigma^2, X, Y &\sim \Gamma(\tilde{c}_i + \sum_{d=1}^{D} \sum_{k=(i-1)\Delta+1}^{T} Y_{k,d}, \tilde{d}_i + \sum_{d=1}^{D} \sum_{k=(i-1)\Delta+1}^{T} x_{k,d} ), \quad (71)
\end{align*}
\]

where $A$, $B$, and $\tilde{d}_\sigma$ are given as follows:

\[
\begin{align*}
  A &= \frac{1}{\tilde{\sigma}_a^2} + \frac{1}{\sigma^2} \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(1-x_{k,d})^2}{x_{k,d}}, \\
  B &= \frac{\tilde{\mu}_a}{\tilde{\sigma}_a^2} + \frac{1}{\sigma^2} \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(1-x_{k,d})x_{k+1,d}}{x_{k,d}}, \\
  \tilde{d}_\sigma &= \tilde{d}_\sigma + \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(x_{k+1,d} - a(1-x_{k,d}))^2}{2x_{k,d}}.
\end{align*}
\]

Proof. We first prove Equation (69). Substituting the priority distribution of the parameter $a$ into the posterior distribution of $a|\sigma^2, \Theta, X, Y$, we obtain its pdf as follows:

\[
\begin{align*}
  f(a|\sigma^2, \Theta, X, Y) &\propto f(\Xi, X, Y) = P(Y|\Xi, X) f(X|\Xi) f(\Xi) \\
  &\propto f(X|\Xi) f(a) \propto f(a) \prod_{d=1}^{D} \prod_{k=1}^{T-1} f(x_{k+1,d}|x_{k,d}, \Xi) \\
  &= \frac{1}{\sqrt{2\pi\tilde{\sigma}_a^2}} \exp \left( \frac{(a - \tilde{\mu}_a)^2}{2\tilde{\sigma}_a^2} \right) \prod_{d=1}^{D} \prod_{k=1}^{T-1} \frac{1}{\sqrt{2\pi\sigma^2 x_{k,d}}} \exp \left( \frac{-(x_{k+1,d} - a(1-x_{k,d}))^2}{2\sigma^2 x_{k,d}} \right) \\
  &\propto \exp \left[ a \left( \frac{\tilde{\mu}_a}{\tilde{\sigma}_a^2} + \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(1-x_{k,d})x_{k+1,d}}{\sigma^2 x_{k,d}} \right) - a^2 \left( \frac{1}{\tilde{\sigma}_a^2} + \frac{1}{\sigma^2} \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(1-x_{k,d})^2}{x_{k,d}} \right) \right].
\end{align*}
\]
The second line follows from Equation (67) that conditioning on the process $X$, the arrival count $Y$ does not depend on the parameter $a$, so it is a constant independent of $a$. The third line follows from Equation (68). Thus, it follows from the last line that $a|\sigma^2, \Theta, X, Y$ follows a Normal distribution with mean $B/A$ and standard deviation $1/A$.

We prove Equation (70) by substituting the prior distribution of $\sigma^2$ into the posterior distribution of $\sigma^2|a, \Theta, X, Y$. To be specific, the following holds:

$$f(\sigma^2|a, \Theta, X, Y)$$

$$\propto f(\Xi, X, Y) = P(Y|\Xi, X)f(X|\Xi)f(\Xi)$$

$$\propto f(X|\Xi)f(\sigma^2) \propto f(\sigma^2) \prod_{d=1}^{D} \prod_{k=1}^{T-1} f(x_{k+1,d}|x_{k,d}, \Xi)$$

$$\propto \frac{\tilde{c}_\sigma^d}{\Gamma(\tilde{c}_\sigma^d)} (\sigma^2)^{-\tilde{c}_\sigma-1} \exp \left( -\frac{\tilde{d}_\sigma}{\sigma^2} \prod_{d=1}^{D} \prod_{k=1}^{T-1} \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left( -\frac{(x_{k+1,d} - a(1 - x_{k,d}))^2}{2\sigma^2 x_{k,d}} \right) \right)$$

$$\propto (\sigma^2)^{-\tilde{c}_\sigma-1-D(T-1)/2} \exp \left[ -\frac{1}{\sigma^2} \left( \tilde{d}_\sigma + \sum_{d=1}^{D} \sum_{k=1}^{T-1} \frac{(x_{k+1,d} - a(1 - x_{k,d}))^2}{2x_{k,d}} \right) \right].$$

The second line follows from Equation (67) that conditioning on the process $X$, the arrival count $Y$ does not depend on the parameter $\sigma^2$, so it is a constant independent of the value of $\sigma^2$. The third line follows from Equation (68). Therefore, the last line implies that $\sigma^2|a, \Theta, X, Y \sim \text{InverseGamma}(\tilde{c}_\sigma + D(T - 1)/2, \tilde{d}_\sigma)$.

We end the proof by showing Equation (71). The following holds:

$$f(\theta_i|a, \sigma^2, X, Y)$$

$$\propto f(\Xi, X, Y) = P(Y|\Xi, X)f(X|\Xi)f(\Xi)$$

$$\propto f(Y|\theta_i, X)f(\theta_i) \propto f(\theta_i) \prod_{d=1}^{D} \prod_{k=1}^{i\Delta} \mathbb{P}(Y_{k,d}|x_{k,d}, \theta_i)$$

$$\propto \frac{\tilde{c}_{\theta_i}^{i\Delta} - \tilde{c}_\theta}{\Gamma(\tilde{c}_\theta)} \tilde{d}_\theta^{-\tilde{c}_\theta} \exp(-\tilde{d}_\theta \theta_i) \prod_{d=1}^{D} \prod_{k=1}^{i\Delta} \frac{(\theta_i x_{k,d})^j}{j} \exp(-\theta_i x_{k,d})$$

$$\propto \frac{\tilde{c}_\theta + \sum_{d=1}^{D} \sum_{k=1}^{i\Delta} Y_{k,d}}{\Gamma(\tilde{c}_\theta)} \exp \left( -\theta_i \left( \tilde{d}_\theta + \sum_{d=1}^{D} \sum_{k=1}^{i\Delta} x_{k,d} \right) \right).$$
The second line follows from the fact that the diffusion process \( x(t) \) and thus its discretization \( X \) is independent of the hour-of-day effect \( \theta_i \). This proves Equation (71) and thus completes the proof. \( \square \)

Moreover, we can also compute the posterior distribution of the Markov process \( x_{k,d} \) (for \( k = 1, \ldots, T \) and \( d = 1, \ldots, D \)) given the parameters \( \Xi \) and the observed data \( Y \). Let \( X = (x_{k,d}) \) denote the underlying arrival rate process guiding the arrival process. In addition, let \( X_{-(k,d)} = \{x_{k',d'} : k' \neq k \text{ and } d' \neq d\} \). It follows from Equation (11) in Zhang (2013) that

\[
 f(x_{k,d}|X_{-(k,d)}, Y, \Xi) \propto f(x_{k,d}|x_{k-1,d}, \Xi) f(x_{k+1,d}|x_{k,d}, \Xi) \mathbb{P}(Y_{k,d}|x_{i,d}, \Xi) \tag{72}
\]

The right-hand side of Equation (72) is calculated by substituting Equations (67)-(68). The Metropolis-Hastings algorithm can be applied to generate the samples of \( x_{k,d} \) using Equation (72); see Chapter 6 of Gamerman (2006).

Letting \( J \) denote the the total number of samples of the Gibbs sampler, the steps of implementing the Gibbs Sampler are given as follows:

1. Initialize the initial sample of \((\Xi, X)\) at \((\Xi^{(0)}, X^{(0)})\).

2. Given \((\Xi^{(j)}, X^{(j)})\), simulate \((\Xi^{(j+1)}, X^{(j+1)})\) as follows: For \( j = 0, 1, \ldots, J - 1, \)
   
   (a) Draw a sample of \( a^{(j+1)} \) from the Gaussian posterior conditional on the values of \((\sigma^{(j)}|^2, \Theta^{(j)}, X^{(j)}, Y)\) given by Equation (69).

   (b) Draw a sample of \( (\sigma^{(j+1)}|^2 \) from the inverse Gamma posterior conditional on \((a^{(j)}, \Theta^{(j)}, X^{(j)}, Y)\)

   given by Equation (70).

   (c) Draw a sample of \( \theta_i^{(j+1)} \) from the Gamma distribution conditional on the values of \((a^{(j)}, (\sigma^{(j)}|^2, X^{(j)}, Y))\) given by Equation (71).

   (d) Draw a sample of \( x_{k,d}^{(j+1)} \) from the posterior distribution conditional on the value of \((\Xi^{(j)}, X_{-(i,d)}^{(j)}, Y)\) by Equation (72) with the Metropolis-Hastings algorithm.

There is no specific rule to determine the number of samplers \( J \). We draw \( J = 2,000,000 \) samples and trace the plots of the samples of the unknown parameters \( \Xi \) to ensure that the samples have converged. We discard the first half of the samples and use the mean of the second half of the
samples as the estimates of the unknown parameters $\Xi$. In addition, we use various initial values of the parameters $\Xi$ and do not observe the dependency of the estimates on the initial value. The estimates of the unknown parameters $\Xi$ are given in Table 9. The parameters $a$ and $\sigma$ characterizes the dynamics of the underlying diffusion process $x(t)$ that rules the arrival rate process $\lambda(t)^{27}$. The parameters $\Theta = (\theta_1, \ldots, \theta_{10})$ capture the hour-of-day effect.

Table 9: The estimates of the parameters $a$, $\sigma$ and $\Theta$.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a$</td>
<td>8.73e-3</td>
<td>4.27e-4</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>1.82e-2</td>
<td>3.29e-5</td>
</tr>
<tr>
<td>$\theta_1$</td>
<td>0.488</td>
<td>0.97e-2</td>
</tr>
<tr>
<td>$\theta_2$</td>
<td>0.712</td>
<td>1.46e-2</td>
</tr>
<tr>
<td>$\theta_3$</td>
<td>0.754</td>
<td>1.53e-2</td>
</tr>
<tr>
<td>$\theta_4$</td>
<td>0.768</td>
<td>1.60e-2</td>
</tr>
<tr>
<td>$\theta_5$</td>
<td>0.752</td>
<td>1.51e-2</td>
</tr>
<tr>
<td>$\theta_6$</td>
<td>0.663</td>
<td>1.37e-2</td>
</tr>
<tr>
<td>$\theta_7$</td>
<td>0.591</td>
<td>1.22e-2</td>
</tr>
<tr>
<td>$\theta_8$</td>
<td>0.581</td>
<td>1.17e-2</td>
</tr>
<tr>
<td>$\theta_9$</td>
<td>0.612</td>
<td>1.23e-2</td>
</tr>
<tr>
<td>$\theta_{10}$</td>
<td>0.623</td>
<td>1.27e-2</td>
</tr>
</tbody>
</table>

References


---

$^{27}$Note that $\mathbb{E}[x(t)|x(0)] = x(0)e^{-at} + (1 - e^{-at})$. Note that $1/a \approx 115$ time units, which is equivalent to 19 minutes. Thus, the pikes induced by the variation of $x(t)$ lasts at the order of 20-30 minutes.