Operational Strategies for Customer Service: A Gatekeeper Framework

Maqbool Dada
Johns Hopkins University

Brett Hathaway
Brigham Young Univerisity

Evgeny Kagan
Johns Hopkins University

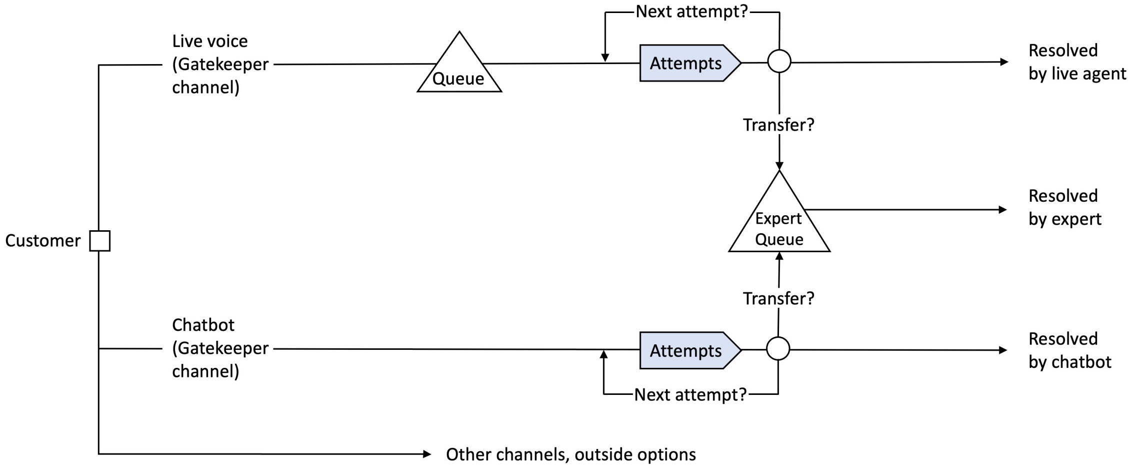
Abstract: Customer service has evolved beyond in-person visits and phone calls to include live chat, AI chatbots and social media, among other contact options. Service providers typically refer to these contact modalities as “channels”. Within each channel, customer service agents are tasked with managing and resolving a stream of inbound service requests. Each request involves milestones where the agent must decide whether to keep assisting the customer or to transfer them to a more skilled – and often costlier – provider. To understand how this request resolution process should be managed, we develop a model in which each channel is represented as a gatekeeper system. We characterize the structure of the optimal gatekeeping policy and identify conditions under which threshold policies yield the optimal solution. We then turn to the broader question of strategic selection of an appropriate mix of service channels. We examine the channel mix problem for the special case where the service provider chooses among three channel architectures: live-agent-only, chatbot-only or both, and show that, in equilibrium, all three architectures may emerge as optimal.

Key words: Gatekeepers, Service Design, Chatbots, Dynamic Programming

1 Introduction

The customer service channels available today vary along many dimensions, including synchronicity (immediate interaction with a chatbot or live agent vs. delayed resolution via email or social media), modality (text-based vs. voice), and the type of service provider (ranging from self-service options like FAQs to live human agents to AI-driven chatbots). Digital channels, particularly chatbots and social media, have seen a significant increase in adoption, with reports indicating a double-digit adoption increase from 2018 to 2020 (Saleseforce 2020). These channels often meet the increased customer expectations for fast and flexible support, while helping reduce call volumes and lower operational costs. At the same time, customers may resent dealing with an automated technology unable to provide a more personal service experience (Aircall 2023). This tension presents firms with several strategic and operational challenges, as firms seek to leverage the cost savings offered by service automation while maintaining high levels of customer satisfaction and loyalty.

Figure 1 Typical Customer Service Desk Architecture Implemented by BlackBeltHelp



To better understand how multiple channels are managed in the customer service setting, we approached BlackBeltHelp, a third-party provider of customer service solutions for the higher education sector. We interviewed BlackBeltHelp executives and engineers in November 2022. The following insights emerged from our conversations:

- **Multichannel architecture:** Different BlackBeltHelp clients choose different channel architectures. Some clients opt only for live channels. Other clients opt for a mix of live and AI-driven channels. BlackBeltHelp even reported at least one client who opted only for the chatbot channel. The client chooses which of these channels to offer and BlackBeltHelp provides the necessary platform, training, staffing, and software development to integrate the channels in one platform. A representation of a two-channel architecture commonly implemented by BlackBeltHelp is shown in Figure 1.
- **Gatekeeper layers:** Customer request resolution is rarely a one-shot process – correctly identifying the type of request and attempting to resolve it is an iterative process that involves milestones or steps codified by the client. While BlackBeltHelp is able to handle the majority of customer requests, a subset of them are ultimately routed to the client (dubbed “expert” in Figure 1). Thus, BlackBeltHelp employees and chatbots almost invariably serve as gatekeepers to the expert workers at the client organization.
- **Chatbot channel has a unique performance and cost structure:** Clients are increasingly choosing to include a chatbot channel into their architecture. For a fixed development cost (plus minimal maintenance costs), the bot is capable of handling a virtually unlimited number of customer inquiries. Rather than paying by resolved request, clients only pay for a software engineer to train the chatbot. Training the bot involves feeding it selected types of requests from a question database, beginning with the most frequently encountered inquiries and gradually incorporating more complex or less common scenarios.

Our interactions with BlackBeltHelp suggest an increasingly complex range of operational strategies for designing customer service – specifically, which channels to offer, and how to integrate them. Our study focuses on developing an analytical framework to address these questions. In particular, we examine the macro-level question of the overall system architecture (*What is the right mix of channels?*), as well as the more micro-level question of within-channel service design (*When and how should transfers be performed?*).

To help guide our modeling choices, we first present two empirical findings in §2.3. To motivate our macro-level question of channel architecture selection, we perform a comprehensive, web-based scan of customer service channels used in industry and find substantial heterogeneity of architectures, both across and within industries. To motivate our micro-level problem of within-channel service design, we perform a survey of customer attitudes towards transfers. The main result of this survey is that not only the presence, but also the type of transfer matters. Specifically, the survey data suggest that customers prefer more personalized transfers (often referred to as "warm transfers" in industry), where the transferring agent provides context and passes on the relevant information to the receiving (expert) agent before leaving the call. Conversely, customers have a strong aversion against the more common "cold transfer", where the customer is transferred without any context or personalization. Indeed, the preference for 'warm transfers" holds even when such transfers result in longer waiting times.

In §3 we begin by analyzing the micro-problem of optimal gatekeeping, i.e., the question of optimal transfer policies of a single channel within the service architecture.We characterize the service resolution process as a sequential S-attempt process. Each of the S resolution attempts has its own processing time and success probability. At the end of an attempt, if the customer issue has not been resolved, the agent handling the request chooses one of three actions: 1) *continue* to the next attempt, 2) *warm transfer*, wherein the agent accompanies the customer and facilitates the transition to the next channel, or 3) *cold transfer*, wherein the agent drops the customer into the next channel.

The firm's objective is to determine optimal transfer policies for the above problem, while accounting for congestion. Specifically, we consider customer traffic (upstream congestion) and expert availability (downstream congestion). Further, channels may have limited operating hours. To account for these factors, we formulate the decision problem as a finite-horizon stochastic dynamic program. The solution to this problem includes a carefully specified set of terminal conditions that make the optimal resolution policy stationary, thereby accommodating both 24/7 operations and finite work shifts. In §4, stationarity is exploited to further structure the optimal policy, which is shown to depend on congestion and resolution attempt.

The optimal policy is computationally complex and does not necessarily have a threshold structure. However, limiting the search to only include threshold policies offers a more intuitive and computationally efficient alternative that runs in polynomial time. Numerical experiments demonstrate that a suitably designed threshold policy performs nearly as well as the optimal policy, and under a simple sufficient condition, which can be interpreted as a variant of the shortest-processing-time (SPT) rule, it is provably optimal.

Having formulated and solved the micro-problem of optimal gatekeeping, we turn to the macro-problem of choosing the overall architecture of the service system (§5), where each channel is modeled as a separate gatekeeper process. The macro-problem consists of choosing which gatekeeper channels to offer (human, chatbot, both), while accounting for the optimal resolution policies within each channel. To do so, we first re-examine the gatekeeping problem when a chatbot, rather than a live agent, acts as a gatekeeper. Chatbot systems have a unique cost structure (fixed costs of development, rather than variable costs of staffing), face virtually no upstream congestion, and their actions are more limited compared to human gatekeepers, all of which significantly simplifies the problem. We then characterize the effects of the channel architecture and resolution policies on service quality, which in turn determine customer demand (arrivals to each channel) as a response to the channel-specific performance measures announced by the firm. Finally, we present a numerical example that shows that, depending on the problem parameters, all three architectures (live-agent-only, chatbot-only, both) may emerge as optimal, mirroring the variety of service channel architectures observed in industry.

Our work makes two key contributions. First, we develop a comprehensive model of optimal gatekeeping. While gatekeepers are becoming a more prevalent lens for studying service and healthcare systems (Shumsky and Pinker 2003, Freeman et al. 2017, Hathaway et al. 2023), optimal gatekeeper behavior remains understudied. Our results bridge an important gap in theoretical development in this area. Second, we integrate chatbots into a broader service operations framework. As chatbot adoption continues to rise (Cohen 2018, Sheehan et al. 2020, Senawi et al. 2023), it is becoming increasingly important to develop analytical approaches to better understand their value (and limitations) in customer service.

2 Overview, Literature and Empirical Foundations

2.1 Overview

At a high level, the firm's service design problem is to choose service channels based on their contributions to the firm's profit, which depend on demand and operating cost. If we restrict the choices to two focal channels, a live-agent channel and a chatbot channel, and denote their profit contributions by π^{agent} and π^{bot}, then the firm's profit-maximization problem is given by:

$$\max_{\mathcal{A},\mathcal{B},D^{agent},D^{bot}} \mathcal{A}\cdot\pi^{agent}(D^{agent},\lambda^{agent}(D^{agent},D^{bot}))+\mathcal{B}\cdot\pi^{bot}(D^{bot},\lambda^{bot}(D^{agent},D^{bot})), \tag{1}$$

where $\mathcal{A}$ ($\mathcal{B}$) are indicator variables that denote whether the live-agent (bot) channel is offered, λ^{agent} (λ^{bot}) are the arrival rates to each channel, and D^{agent} (D^{bot}) are the live (bot) channel resolution policies, such as when and how to initiate transfers under different congestion patterns. The firm selects the optimal channels ($\mathcal{A},\mathcal{B}$), and how these channels should operate (D^{agent},D^{bot}). Utility-maximizing customers observe these choices and respond by choosing which channel (if any) to join, with their collective decisions determining λ^{agent} and λ^{bot}.

In the remainder of §2 we review related research and present several pieces of empirical evidence that help support our modeling choices. In §3 we introduce the micro-problem of optimally setting D^{agent}, i.e., the resolution policy for channels staffed by live agents. In §4 we characterize the structure of the optimal policy for this problem. In §5 we show how the problem differs for the bot, i.e., how the firm should set D^{bot}, examine the problem from the customers' perspective and characterize λ^{agent} and λ^{bot}, and conclude with a numerical example that illustrates that the firm should jointly choose $\mathcal{A}, \mathcal{B}, D^{agent}$ and D^{bot} while accounting for customer response to those choices.

2.2 Related Research

Channel Architecture

Optimal channel mix problems have been studied in the retail context, where the integration of physical and digital channels has become common practice (Brynjolfsson et al. 2013, Gao and Su 2017, Gallino and Moreno 2019). Multichannel strategies are especially common in the food service industry, where congestion plays a central role (Gao and Su 2018, Feldman et al. 2023). Multichannel environments have also been studied in marketing with a focus on understanding customer experiences in different channels (Ansari et al. 2008, Lund and Marinova 2014, Lemon and Verhoef 2016). No studies that we are aware of focus specifically on channel selection in the customer service domain.

Resolution Process

The channel resolution policies (D^{agent}, D^{bot}) set the protocols for managing incoming customer requests; in particular, how much time and effort the agent (or chatbot) should spend on each request, and how and when requests should be transferred to an expert. Each channel therefore acts as a gatekeeper system (Shumsky and Pinker 2003, Hasija et al. 2005, Lee et al. 2012), in which simple requests are handled by lower-cost workers (or chatbots), while more complex tasks are reserved for higher-cost specialists or experts. Hathaway et al. (2023) examine experimentally whether gatekeepers correctly incorporate incentives to cold transfer while accounting for upstream congestion present in such systems. Freeman et al. (2017) and Batt and Terwiesch (2017) study work-sharing behaviors in healthcare. Also related are Alizamir et al. (2013) and Kremer and de Véricourt (2023), who examine optimal stopping in sequential diagnostic processes under congestion. Our gatekeeping model differs from the existing research in that we (a) expand the action set to include multiple types of transfers, and (b) incorporate upstream and downstream congestion.

Channel Joining Behavior

We model channel demand, λ^{agent} and λ^{bot}, as an endogenous response to the firm's service offering. Our work is therefore related to the literature on optimal routing models that incorporate customer decisions (Armony and Maglaras 2004, De Véricourt and Zhou 2005, Gans and Zhou 2007), as well as to the broader behavioral literature on queue joining (Naor 1969, Kremer and Debo 2016, Allon and Kremer 2018). Gatekeeper systems, i.e., systems in which customers are frequently transferred, may make the wait more uncertain as well as more fragmented, both of which may reduce channel uptake (Leclerc et al. 1995, Kumar and Dada 2021, Flicker and Hannigan 2022). Abrupt transitions within service experiences, which are more likely in gatekeeper systems, may be particularly memorable and can negatively affect customer satisfaction and channel uptake (Lee 2008, Das Gupta et al. 2016). Our model of customer preferences explicitly accounts for such behavioral factors affecting customer response.

AI Chatbots in Customer Service

While AI-driven chatbots are becoming increasingly capable of solving complex problems, there remain significant barriers to their adoption in customer service. A key barrier is that chatbot interactions often result in a transfer, necessitating a complete restart of the service process with a human provider (Kagan et al. 2023). Our model will refer to such transfers as *cold* transfers, as opposed to more seamless and personalized *warm* transfers, which chatbots are typically unable to provide (Senawi et al. 2023). Separately, customers may dislike interactions with an algorithmic (rather than human) service provider, and that may reduce their willingness to use the chatbot option (Dietvorst et al. 2015, Luo et al. 2019, Sheehan et al. 2020, Kagan et al. 2023). Algorithmic attitudes are incorporated into our model of customer preferences via a technology aversion parameter, affecting channel demands λ^{agent} and λ^{bot} in (1).

2.3 Empirical Foundations

Before continuing with our model and analysis, we briefly describe two pieces of empirical evidence that help guide our modeling choices.

Industry Scan

To better understand channel architecture choices used in practice, we examined the "Contact Us" page of each of the Fortune 100 companies (100 top revenue-generating companies within the United States). Our industry scan reveals a wide variety of available channels (see EC.1.1. for details). In Table 1, we restrict our attention to voice, live chat, and chatbot - the three prevalent channels that allow for synchronous communication between customer and server. Each company that offers two or more of these channels is categorized as "multichannel", with approximately one third of the companies falling into this category. Notably, there is considerable variation across sectors. For example, 70% of retail companies provide multiple synchronous channels, compared to just 10% in the energy sector. These differences in channel adoption suggest that firms strategically tailor customer service channels to meet distinct operational and customer service needs. We will examine the relevant trade-offs related to within-channel design in §3-4, as well as the broader channel mix decisions in §5.

Table 1 2022 Fortune 100 Company Service Channel Architectures by Sector

Sector	Count	% Multichannel	% Voice	% Live Chat	% Chatbot
Energy	10	10%	80%	10%	10%
Financials	22	32%	95%	18%	27%
Health Care	16	31%	100%	13%	19%
Retail	10	70%	100%	60%	70%
Technology	10	55%	91%	45%	36%
Other	31	29%	77%	23%	13%
Total	100	33%	89%	25%	25%

Cold vs. Warm Transfers: Survey

One common approach to counteract the negative service quality effects of transfers is to ensure a more continuous transition by accompanying the customer during the transfer and by passing on the information to the next server. This approach is often referred to as a *warm* transfer in industry (Moneypenny 2021). While there are many anecdotal accounts of warm transfers increasing customer satisfaction (Senawi et al. 2023, Maya 2023), such transfers have not been studied in the academic literature.

To better understand customer attitudes towards warm versus cold transfers, we conducted an online survey on Prolific ($N = 202$, see EC.1.2. for details). The majority of respondents reported having experienced both cold and warm types of transfers in the past (90.09% and 69.31%, respectively). Respondents were also asked to rate their satisfaction with each type of transfer on a 5-point Likert scale. The results indicated an overwhelming preference for warm transfers, with average satisfaction scores of 3.80 for warm transfers compared to 2.53 for cold transfers (paired t-test, $p \ll 0.01$). Furthermore, when presented with a choice between the two types of transfers, a notable 71.78% of respondents favored warm transfers over cold ones, despite being informed that warm transfers might involve additional waiting time. These findings suggest a strong customer preference for continuity and personalization in service transitions, offered by warm transfers. In §3-4 we will consider both cold and warm transfer types in our model of customer service design.

3 Live Agent Model

In this section we focus on the live-agent channel and develop a dynamic programming model for how the firm should set the resolution policy for this channel (D^{agent} in (1)). This channel is staffed by a number of agents who are trained to follow the same protocol determined by the firm. The admission control is organized such that each agent receives an equitable load, making the system scalable in the number of agents. Hence, it is convenient to simply model the channel as a single representative live agent (which we will refer to as the *gatekeeper*) receiving a stream of incoming customer requests. The agent receives a flat payment and acts in the interest of the firm. The agent can be an internal employee, a worker in an outsourced call center, or can be employed by a third-party provider, such as BlackBeltHelp (§1).

The key service process decision facing the firm is to determine, given current congestion patterns, how long the agent should continue working with each customer, and when and how that customer should be transferred to an expert service provider. The trade-off from the firm's perspective is between allowing its agents (gatekeepers) to spend more time with each customer, leading to fewer transfers and higher quality service, vs. instructing its gatekeepers to transfer earlier (and do so in a cost-effective manner).

3.1 Resolution Process

The agent operates in a shift of T discrete periods and is paid at a rate of c^{wage} per unit time. At the beginning of each period $t = 1, \cdots, T-1$, a customer arrives with probability q. If the agent is busy, the customer is not admitted and leaves the system. If the agent is available, the customer is admitted for service and service starts at $t+1$.[1] The firm receives revenue of r for each admitted customer when the customer issue is resolved and the transaction is closed.

The resolution process is represented by an ordered list of potential solutions $s \in \{1, 2, ..., S\}$, where each attempt s is characterized by the probability that it resolves the request, p_s, and its handling time, τ_s. A similar checklist approach also used in Alizamir et al. (2013) and Kremer and de Véricourt (2023) to represent diagnostic test ordering in the healthcare setting. Without loss of generality, one of the S attempts is guaranteed to resolve the request ($\sum p_s = 1$). The focal decision of the firm is *how many* attempts the agent will make before transferring the customer to the expert. In particular, upon completion of attempt s, one of two outcomes may occur: (1) attempt s resolves the request or (2) attempt s does not resolve the request and the agent follows the firm's resolution policy by performing one of three actions:

Continue: The agent makes the next attempt $(s+1)$. This expends τ_{s+1} time units, and resolves the request with conditional probability ρ_{s+1}.

Warm Transfer: The agent warm transfers the customer to an expert. In practice, this may include introducing the customer to the expert, providing context, or passing on any relevant customer information to avoid repetition. A warm transfer costs the firm c^w for expert compensation, where c^w represents a transfer payment to a different division of the same firm where the expert is employed, or a fee to the third party that employs the expert. The duration of the warm transfer depends on whether an expert is immediately available to receive the customer. In each period, an expert is available $(A = 1)$ with probability a and unavailable

[1] The model can be extended to include queueing effects as a consequence of congestion but not doing so allows us to keep the essential decision in focus.

($A = 0$) with probability $1 - a$. Once the warm transfer begins, the agent expends τ_w periods performing the transfer. Following the warm transfer, the agent remains idle if the customer queue is empty ($Q = 0$), or begins handling a new request if the queue is nonempty ($Q = 1$).

Cold Transfer: The agent cold transfers the customer to an expert, who resolves the request. A cold transfer costs the firm c^c for expert compensation, where $c^c > c^w$ since the agent simply drops the customer into the expert queue, requiring the expert to spend more time with the customer. If the queue is empty ($Q = 0$), the agent is idle until another request arrives; otherwise, the queue is nonempty ($Q = 1$) and the agent immediately begins handling a new request.

The decision repeats until each request is either resolved or transferred to the expert. After each resolved (or transferred) request, the agent begins serving the next customer as soon as one becomes available. Both congestion states (Q and A) are observable. Thus, the key trade-off at each decision point is whether, given current congestion, to dedicate more agent time to the focal customer to increase the chances of resolving their request, or to process a higher volume of customers by referring them to the expert, thereby freeing up agent capacity.

3.2 Dynamic Programming (DP) Formulation

To formulate the DP for the above problem, we need to define the state of the system in period $t = 1, \cdots, T$. To that end, let $X \in \{0^i, 0^w, 1, 2, \cdots, S\}$ denote the agent's status, where $X \in \{1, 2, \cdots, S\}$ indicates that the agent just finished attempt s, $X = 0^i$ indicates an auxiliary state in which the agent was idle in the previous period waiting for a new request to process, and $X = 0^w$ indicates the other auxiliary state in which the agent has initiated a warm transfer but was waiting with the customer in the previous period because no expert was available to receive the customer. Thus, the state of the system is fully specified by the 3-tuple (X, Q, A). At time t, the objective is to choose the action that maximizes expected net revenue over time t to T, exclusive of the "sunk" agent wage, which need not be considered. Denote the optimal value function at $t = 1, \cdots, T$ by $V_t(X, Q, A)$, and the terminal value vector by $V_{T+1}(X, Q, A)$.

Before writing the value function for each of the $4(S+2)$ states, we make two observations. First, if the agent was idle waiting for a new request to arrive in the previous period ($X = 0^i$), then the availability of the expert is irrelevant since the agent will either continue to be idle (if $Q = 0$) or begin working on the first attempt of a new request (if $Q = 1$). Hence, the value function does not depend on A and can be conveniently denoted by $V_t(0^i, Q, -)$. Second, if the agent has initiated a warm transfer and was waiting for an expert to become available ($X = 0^w$), then the queue state is irrelevant since the agent will either continue to wait for the expert (if $A = 0$) or begin handing the request off to the expert (if $A = 1$). Hence, the value function does not depend on Q and can be conveniently denoted by $V_t(0^w, -, A)$.

These simplifications allow us to write the value function for $t = 1, \cdots, T$ as follows:

$$V_t(X,Q,A) = \begin{cases} (1-q)V_{t+1}(0^i,0,-) + qV_{t+1}(0^i,1,-), & \text{if } X = 0^i, Q = 0, \quad (2.1) \\ (1-q)(1-a)V_{t+\tau_1}(1,0,0) + q(1-a)V_{t+\tau_1}(1,1,0) + \cdots & \\ \quad (1-q)aV_{t+\tau_1}(1,0,1) + qaV_{t+\tau_1}(1,1,1), & \text{if } X = 0^i, Q = 1, \quad (2.2) \\ (1-a)V_{t+1}(0^w,-,0) + aV_{t+1}(0^w,-,1), & \text{if } X = 0^w, A = 0, \quad (2.3) \\ (1-q)V_{t+\tau_w}(0^i,0,-) + qV_{t+\tau_w}(0^i,1,-), & \text{if } X = 0^w, A = 1, \quad (2.4) \\ \rho_X(r + V_t(0^i,Q,-)) + (1-\rho_X)\big[\max\{\mathcal{N}_t(X), \cdots & \\ \quad -(c^w - r) + V_t(0^w,-,A), -(c^c - r) + V_t(0^i,Q,-)\}\big], & \text{if } 1 \le X < S, \quad (2.5) \\ r + V_t(0^i,Q,A), & \text{if } X = S, \quad (2.6) \end{cases}$$

where $\mathcal{N}_t(X)$ in (2.5) is the value-to-go for making attempt $X+1$ (continuing) after attempt X fails in period t. This is given by

$$\mathcal{N}_t(X) = (1-q)(1-a)V_{t+\tau_{X+1}}(X+1,0,0) + q(1-a)V_{t+\tau_{X+1}}(X+1,1,0) + \cdots$$
$$(1-q)aV_{t+\tau_{X+1}}(X+1,0,1) + qaV_{t+\tau_{X+1}}(X+1,1,1). \tag{3}$$

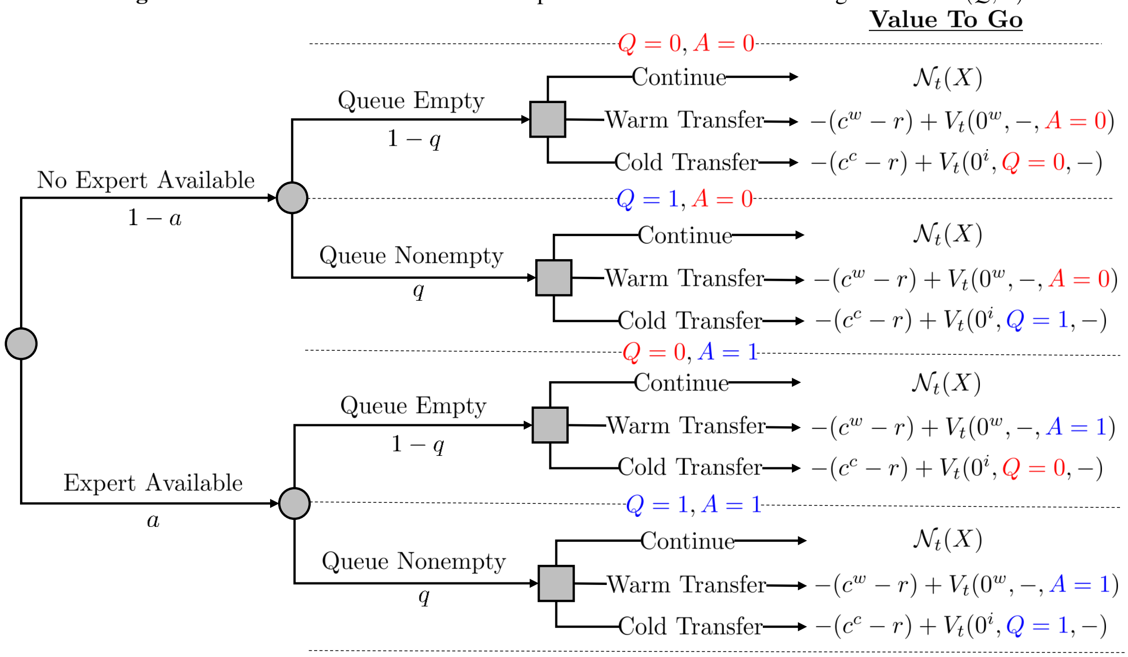
In (3) each of the four terms on the right side of the inequality is the probability of being in a given congestion state (Q,A) after completing attempt $X+1$, multiplied by the value-to-go of being in that congestion state upon completion.

Equation (2.1) assures that if the agent was idle during period t because the queue was empty, the agent begins period $t+1$ idle, whereas (2.2) assures that if the queue was nonempty, the agent began service. Equation (2.3) assures that if the agent was idle during period t waiting for an expert, the agent begins period $t+1$ idle, whereas (2.4) assures that if the expert became available, the agent spends τ_w periods performing the handoff and is idle at the beginning of period $t+\tau_w$. Equation (2.6) guarantees that if the final potential solution was attempted ($X = S$), the firm received revenue r, and the agent moved to the next customer as soon as one became available.

The key trade-off is shown in (2.5). In this case, with (conditional) probability ρ_X, attempt $X = s$ resolves the request, the firm receives the revenue r and the firm instructs the agent on how to proceed optimally. With complementary probability $1-\rho_X$, attempt $X = s$ did not resolve the request. So, to maximize expected future value, the agent must either make the next attempt $X+1$ (the first argument of the max operator), warm transfer and incur the warm transfer cost of c^w (the second argument), or transfer the request and incur the cold transfer cost of c^c (the third argument). Note that in all cases, the firm eventually receives revenue r from the customer. Finally, while we can choose $V_{T+1}(X,Q,A)$ (the terminal conditions) in many ways, we defer their specification to later as they play a critical role in characterizing the structure of the optimal policy.

The essence of the problem is captured intuitively by Figure 2, in which circles represent random nodes and squares represent decision nodes. The agent has just failed to resolve the request after making attempt

Figure 2 Value-To-Go After Failed Attempt X For Each Action And Congestion State (Q,A)



X. The state is defined by downstream congestion (A) and the presence of customers in queue (Q). For each (Q,A) combination, one of the three actions is optimal: continuing, resulting in the value-to-go of $\mathcal{N}_t(X)$, warm transferring, resulting in a net revenue of $r-c^w$, and cold transferring, resulting in a net revenue of $r-c^c$, with either transfer option leading to a transition to a new state.

Several observations are in order. First, because Q and A are time-independent, $\mathcal{N}_t(X)$ depends only on X, i.e., how many unsuccessful attempts have been completed. Thus, the value of continuing is constant across the congestion states. Second, to visualize upstream congestion, we highlight instances where $Q=1$ in blue and instances where $Q=0$ in red. When comparing the top two decisions (($Q=0, A=0$) and ($Q=1, A=0$)), the only difference between the value-to-go for each action is that it is greater for transferring when $Q=1$ than when $Q=0$, because transferring when the queue is empty leads to nonproductive idle time. The same holds when comparing the bottom two decisions. Third, to visualize downstream congestion, we highlight instances where $A=1$ in blue and instances where $A=0$ in red. Analogous to upstream congestion, when comparing congestion in the first and third decisions, the only difference between the value-to-go for each action is that it is greater for warm transferring when $A=1$ than when $A=0$, because when there is no expert available, warm transferring requires the agent to wait idly until one becomes available. The same holds when comparing the second and fourth decisions. We next use these observations to derive local properties that help simplify the structure of the optimal policy.

3.3 Local Properties

The dynamic program in (2.1) - (2.6) generates a large number of candidate policies. Since there are four possible states, each with a choice of three actions, at each X and t, the strategy space consists of $3^4=81$

possible action vectors. Ordinarily, such a large number of possible policies would be daunting, even if the problem were stationary. Fortunately, we are able to reduce complexity by finding a set of dominance rules that eliminate all but 11 of the 81 decision rule vectors. The proof of this and all subsequent results are in EC.2.

THEOREM 1A *If in period t, under a given A it is optimal to warm transfer when the queue is nonempty ($Q = 1$), then it is also optimal to warm transfer when the queue is empty ($Q = 0$). If in period t, under a given A it is optimal for the agent to continue when the queue is nonempty ($Q = 1$), then it is also optimal to continue when the queue is empty ($Q = 0$).*

THEOREM 1B *If in period t, under a given Q it is optimal to cold transfer when an expert is available ($A = 1$), then it is also optimal to cold transfer when no expert is available ($A = 0$). If in period t, under a given Q it is optimal for the agent to continue when an expert is available ($A = 1$), then it is also optimal to continue when no expert is available ($A = 0$).*

Theorem 1A relies on the fact that neither the payoff of warm transferring nor the payoff of making the next attempt depends on the current queue state (Q), and the payoff of cold transferring is lower when the queue is empty ($Q = 0$). Analogously, Theorem 1B relies on the fact that neither the payoff of cold transferring nor the payoff of making the next attempt depends on current expert availability (A), and the payoff of warm transferring is lower when no expert is available ($A = 0$).

Table 2 lists the admissible decision rules that remain after eliminating all decision rules that do not satisfy the dominance conditions in Theorems 1A and 1B. The first rule is C(ontinue), which is to continue with the next attempt irrespective of congestion state (Q, A). The next four are H(ybrid) rules, in which (depending on the state) the agent may continue, warm transfer, or cold transfer. The final six are T(ransfer) rules, in which (depending on the state) the agent always either warm transfers or cold transfers. From the customer perspective (which we will discuss in §5) the rules also map to service quality. For example, rule C is the pure single-channel strategy as the customer remains with the agent for the next attempt, regardless of congestion. In contrast, rules T1 through T5 correspond to lower service quality and transfer the customer, with the transfer mode depending on the state. Among the T rules, T5 is the rule with the lowest service quality, as the customer is always cold transferred, regardless of state.

3.4 Terminal Conditions and Stationarity

While there are many ways to specify terminal conditions, we tailor them such that the policy is stationary over T, making the optimal policy independent of the amount of time left in the agent's shift. This also allows us to characterize both finite-horizon shifts and 24/7 operations using the same policy.

Let $G(X, Q, A)$ be a constant used in defining the value function under the optimal stationary policy at state (X, Q, A). Since there are four congestion states for $X \in \{1, 2, \cdots, S\}$ and we effectively collapse the

Table 2 Admissible Decision Rules

highest service quality

State:	$(Q=0,A=0)$	$(Q=1,A=0)$	$(Q=0,A=1)$	$(Q=1,A=1)$
Label				
C	Continue	Continue	Continue	Continue
H1	Continue	Continue	Warm Transfer	Warm Transfer
H2	Continue	Cold Transfer	Warm Transfer	Warm Transfer
H3	Continue	Cold Transfer	Warm Transfer	Transfer
H4	Continue	Cold Transfer	Continue	Cold Transfer
T1	Warm Transfer	Warm Transfer	Warm Transfer	Warm Transfer
T2	Warm Transfer	Cold Transfer	Warm Transfer	Warm Transfer
T3w	Cold Transfer	Cold Transfer	Warm Transfer	Warm Transfer
T3c	Warm Transfer	Cold Transfer	Warm Transfer	Cold Transfer
T4	Cold Transfer	Cold Transfer	Warm Transfer	Cold Transfer
T5	Cold Transfer	Cold Transfer	Cold Transfer	Cold Transfer

lowest service quality

four congestion states into two each for $X = 0^i$ in (2.1) and for $X = 0^w$ in (2.4), there are $4S + 2 + 2 = 4S + 4 = 4(S+1)$ constants to be specified, which we refer to as the terminal conditions. Also, let R^* be the revenue (net of agent and expert costs) per unit time under the optimal stationary policy. Then we define the solution to the dynamic program as

$$V_t(X,Q,A) = R^*(T+1-t) + G(X,Q,A). \tag{4}$$

The term $R^*(T+1-t)$ is the pro-rated payment for the time remaining in the program, and the constant captures the expected incremental value of being in a given congestion state (Q,A), given the agent's current status (X). In the proof of Theorem 2 below we specify the conditions necessary to induce stationarity, and refer to them as *Type F* conditions, since they are affine functions of R^*. This leads us to the following:

THEOREM 2 *Under terminal conditions of Type F the optimal resolution policy is stationary, i.e., the optimal decision rule after a given failed attempt does not depend on t.*

The economic meaning of these terminal conditions is as follows (EC.2 contains detailed derivation). For a given t, the value of (4) increases in the number of failed attempts. This is because the request is getting closer to successful resolution. Moreover, for each X, $G(X,Q,A)$ is lowest for $(Q=0,A=0)$ and highest for $(Q=1,A=1)$. This is because there is potential value in the queue being nonempty (since the agent may immediately begin processing a request) and potential value in an expert being available (since the agent may immediately begin performing a warm transfer). Also, the difference between $G(0^i,1,-)$ and $G(0^i,0,-)$ is R^*/q, the incremental expected opportunity cost of waiting for the next customer to arrive. Analogously, the difference between $G(0^w,-,1)$ and $G(0^w,-,0)$ is R^*/a, the incremental expected opportunity cost of waiting for the expert to become available. This allows us to price out the incremental value of congestion, leading to the structural results in §4.

4 Optimal Policy Under Stationarity

In this section we use the terminal conditions from Theorem 2 to further structure the optimal policy. While these simplifications provide some insight, the optimal policy remains computationally burdensome.

To provide a polynomial-time alternative, we examine the performance of *threshold* policies, which are both intuitive and fast to compute. We identify conditions under which threshold policies are optimal and calculate the optimality gap when they are not.

4.1 Additional Structure

Here we use the terminal conditions formulated in the proof of Theorem 2 to reduce the number of admissible decision rules for a given problem instance from 11 down to 3 or 4. The key is identifying when a cold transfer is preferred to a warm transfer for a given (Q,A). As in the proof of Theorem 3, we proceed by substituting the terminal conditions in (4) into the transfer arguments of the max operator in (2.5). Then, a cold transfer is preferred to a warm transfer for a given (Q,A) if

$$G(0^i,Q,-)-G(0^w,-,A)\geq c^c-c^w.$$

Furthermore, substituting in $G(0^i,Q,-)$ and $G(0^w,-,A)$ from the proof of Theorem 2 yields

$$R^*[\tau_w-1+(1-A)/a+Q/q]\geq c^c-c^w.$$

Adding and subtracting $1/q$ to the left-hand side and simplifying yields

$$R^*[\tau_w+(1-q)/q+(1-A)/a]-R^*[(1-Q)/q]\geq c^c-c^w.$$

In the inequalities above, the right-hand side is the incremental savings from performing a warm transfer over a cold transfer (recall that $c^w<c^c$ as the expert requires less time to resolve a warm transfer). For a cold transfer to be preferred, these savings must exceed the foregone opportunity cost that a warm transfer induces relative to a cold transfer. This is given by the profit-adjusted difference between the first and second terms in brackets on the left-hand side. The first term in brackets is the expected time before the agent will recommence service after a warm transfer and the second term is the analogous expected time after a cold transfer. This intuitive condition can be rearranged to show that a warm transfer is preferred to a cold transfer when the optimal profit R^* is greater than

$$R^*\geq(c^c-c^w)/[\tau_w-1+(1-A)/a+Q/a]\equiv\mathcal{R}(Q,A).$$

These intuitive insights are formalized in Theorem 3 below.

THEOREM 3a *For a given (Q,A) and any X, there exists a positive constant $\mathcal{R}(Q,A)$ such that when $R^*<\mathcal{R}(Q,A)$, a warm transfer is preferred to a cold transfer; otherwise, $R^*\geq\mathcal{R}(Q,A)$, and a cold transfer is preferred to a warm transfer.*

THEOREM 3b R^* *is such that* $0\leq\mathcal{R}(1,0)\leq\mathcal{R}(1,1)\leq\mathcal{R}(0,0)\leq\mathcal{R}(0,1)$*, if* $1/q\leq 1/a$*; otherwise,* $0\leq\mathcal{R}(1,0)\leq\mathcal{R}(0,0)\leq\mathcal{R}(1,1)\leq\mathcal{R}(0,1)$.

THEOREM 3c *For a given problem instance, in an optimal resolution policy there are only up to four admissible decision rules: the continue rule, one or two of the four hybrid rules, and one of the six transfer rules.*

An implication of Theorem 3a is that for each congestion state (Q,A), if the problem is not resolved at attempt X, the agent has two actions: continue or transfer to an expert; if transferring, then for every X, the transfer mode (cold transfer or warm transfer) is the same. In this sense, the choice is binary, but which of these choices is optimal may depend on (Q,A). However, Theorem 3a is sufficient to provide further structure since the constants $\mathcal{R}(Q,A)$ can be ordered as in Theorem 3b; this ordering depends on the relative values of q and a. Thus, for a given problem instance, there are five cases defined by ranges where R^* may lie relative to the four $\mathcal{R}(Q,A)$ constants. Each range has a different set of preferred (more economical) transfer methods across the congestion states (Q,A). Finally, Theorem 3c follows by determining, for each range, which of the 11 decision rules derived from Theorem 1 (listed in Table 2) are potentially optimal under stationarity by admitting only the rules where, for each (Q,A), the optimal action is either continue or to transfer the customer under the method that is more economical in that range. To elaborate on Theorem 3, we include Table 3. For ease of exposition, we only present the case where $1/q > 1/a$ (see proof of Theorem 3 for $1/q \leq 1/a$ case).

Table 3a provides the ranges, with the value of R^* increasing from left to right. Table 3b lists the preferred transfer method for each congestion state. For example, in Case 1, R^* is sufficiently low such that it is more economical to warm transfer the customer, regardless of congestion state. However, in Case 2, R^* is sufficiently high such that it is more economical to cold transfer if the queue is empty and there is no expert available ($Q = 1, A = 0$), but sufficiently low such that it is more economical to warm transfer in the remaining congestion states. More generally, when R^* is low, the revenue losses due to the inability to serve new arrivals are low. However, as R^* increases, these revenue losses increase relative to the costs of cold transfers, resulting in cold transferring becoming increasingly preferred. At the extreme, in Case 5 it is more economical to cold transfer in all states.

Finally, given the preferred transfer type, we can construct Table 3c, where we indicate, by range, which decision rules are admissible. We highlight the two corner cases. Case 1 contains problem scenarios where the firm's optimal resolution policy is to provide the most personalized service possible. So, after a given failure, the admissible transfer rule is T1, in which the agent must perform a warm transfer, regardless of Q and A, and the admissible hybrid rule is H1, wherein the agent performs a warm transfer if there is expert availability ($A = 1$). On the other extreme is Case 5, where the firm provides the least-personalized service, concentrating on throughput instead. In this case, the only admissible transfer rule is T5, wherein the agent must cold transfer the customer, regardless of Q and A, and the only admissible hybrid rule is H4, wherein the agent cold transfers contingent on the queue being nonempty ($Q = 1$).

Table 3 Ranges, Preferred Transfer Method, and Admissible Decision Rules Over Cases of R^* $(1/q > 1/a)$

Table 3a: Ranges

Case	Case 1	Case 2	Case 3w	Case 4	Case 5
R^*	$R^* < \mathcal{R}(1,0)$	$\mathcal{R}(1,0) \leq R^* < \mathcal{R}(1,1)$	$\mathcal{R}(1,1) \leq R^* < \mathcal{R}(0,0)$	$\mathcal{R}(0,0) \leq R^* < \mathcal{R}(0,1)$	$\mathcal{R}(0,1) \leq R^*$

Table 3b: Preferred Transfer Method

Q	A	Case 1	Case 2	Case 3w	Case 4	Case 5
0	0	Warm	Warm	Cold	Cold	Cold
1	0	Warm	Cold	Cold	Cold	Cold
0	1	Warm	Warm	Warm	Warm	Cold
1	1	Warm	Warm	Warm	Cold	Cold

Table 3c: Admissible Decision Rules

Label	Rule	Case 1	Case 2	Case 3w	Case 4	Case 5
C	[n,n,n,n]	✓	✓	✓	✓	✓
H1	[n,n,w,w]	✓	✓	✓		
H2	[n,c,w,w]		✓	✓		
H3	[n,c,w,c]				✓	
H4	[n,c,n,c]				✓	✓
T1	[w,w,w,w]	✓				
T2	[w,c,w,w]		✓			
T3w	[c,c,w,w]			✓		
T4	[c,c,w,c]				✓	
T5	[c,c,c,c]					✓

n = continue, w = warm transfer, c = cold transfer

4.2 Operationalization

Having reduced the optimal policy search to a manageable number of decision rules (four or fewer per instance), we now consider the implications of operationalizing them for a specific instance of the cases presented in Table 3. An immediate consequence of Theorem 3c is that if we knew R^* for a given problem instance, we would know which case from Table 3 applies and could therefore find the optimal policy by calculating the profit per unit time for each possible policy within that case and choosing the maximum. Because we do not know the value of R^* a priori, we have to examine each admissible case separately.

Initially, it would appear that since there are three possible decision rules in Cases 1 and 5, and four possible decision rules in Cases 2, 3w, and 4, the computational complexity of evaluating all of the possible optimal policies within each case would be $O(3^S)$ and $O(4^S)$, respectively. Therefore, it would appear that running time would be $O(4^S)$. However, by applying the conditions in Table 3c, the running time can be shown to be $O(3^S)$. This follows since, in each of these cases, it is possible to aggregate at least two of the congestion states. For example, in Case 1, each action only depends on the value of A; similarly, in Case 5, each action depends only on the value of Q. Hence, the problem can be collapsed into just two states so that running time would be $O(2^S)$. In the remaining cases, only two of the four states can be aggregated. For example, in Case 4, for each of the four admissible decision rules, the action is the same for $(Q = 1, A = 0)$ and $(Q = 1, A = 1)$; the two remaining cases are similar. We formalize this logic in the following corollary.

COROLLARY 1 *Under a given problem instance with S attempts, the running time to find the optimal policy is* $O(3^S)$.

4.2.1 Threshold Policy Performance

Although our structural analysis has reduced computational complexity to $O(3^S)$, the complexity still grows exponentially in S. As a polynomial-time alternative to the optimal algorithm, we examined performance using a threshold policy. A policy has a threshold structure when, for a given (Q,A), if the policy is to transfer after failed attempt X, then it would also be to transfer for all subsequent failed attempts. Intuitively, under a threshold policy, the decision-making sequence for a given request is: for a given congestion state, continue until a specified number of attempts; thereafter, perform the transfer type specified for that congestion state. From a practical standpoint, this would allow for easy management communication of the policy structure and eliminate the burden of retaining a decision rule for each attempt/congestion-state combination.

In EC.3 we measure the performance of threshold policies as a heuristic solution for 1,200,000 randomly generated problem instances, each for $S = 3$, $S = 4$, $S = 5$, and $S = 6$. Table 4 summarizes the relevant performance measures from our analysis.

Table 4 Performance Measures of Threshold Policy Heuristic

	Number of Potential Solutions (S)			
Performance Measure	$S = 3$	$S = 4$	$S = 5$	$S = 6$
% Instances Threshold Optimal	97.46%	96.58%	96.02%	95.62%
Average Optimality Gap	0.005%	0.006%	0.006%	0.005%
Maximum Optimality Gap	1.888%	1.818%	1.634%	1.575%

In over 95% of the instances, a threshold policy is optimal, with an average optimality gap of no greater than 0.006%. Notably, while the fraction of instances where a threshold policy is optimal is decreasing in S, the optimality gap remains stable; this is because the optimality gap in instances where a threshold policy was not optimal is decreasing in S. Additionally, worst-case performance measured by the maximum optimality gap decreases in S, suggesting that the threshold heuristic performance improves as the computational burden for finding the optimal policy increases. Overall, this analysis suggests that threshold policies achieve optimal or close-to-optimal performance in the majority of scenarios.

4.2.2 Optimality of Threshold Policies

To build intuition for the main result on threshold policies (Theorem 4) we first consider a simpler variant of the problem, which is akin to a stopping problem. In this simpler variant only warm transfers are available, and the states (Q,A) are not observable to the decision-maker (the result is analogous if instead only cold transfers are available). Consider the case when a threshold policy is optimal. When this is the case, label $X = M$ as the last failed attempt after which the continue action is optimal; thereafter, the optimal policy is to warm transfer. Then, we use Theorems 2 and 3 to i) identify a sufficient condition for the optimality of threshold policies; ii) frame the induction hypothesis; and, iii) start the recursive inductive proof.

We start by focusing on state X at time t by comparing $W_t(X)$, the optimal value-to-go when a warm transfer is performed, with $C_t(X)$, the optimal value-to-go when the continue action is taken. Then, it follows from Theorem 3 that under stationarity, independent of X, $W_t(X)$ is given by

$$W_t(-) = -c^w + R^*\{T+1-t-(1-a)/a - \tau_w - (1-q)/q)\}, \tag{5}$$

where the final three terms with negative coefficients account for expected time expended waiting for the expert, performing the handoff, and then expected waiting time for the next customer arrival. Specifically, when $X = M$, since a warm transfer is optimal when $X = M+1$, using (5) to substitute for $W_{t+\tau_M}(-)$, we get:

$$\begin{aligned} C_t(M) &= \rho_M\{r + R^*(T+1-t-\tau_M)\} + (1-\rho_M)W_{t+\tau_M}(-) \\ &= \rho_M[r + c^w - R^*\{(1-a)/a + \tau_w + (1-q)/q)\}] - c^w - R^*\{(1-a)/a + \tau_w + (1-q)/q)\} + R^*\{T+1-t-\tau_M\} \\ &\equiv \rho_M D - c^w - R^*\{(1-a)/a + \tau_w + (1-q)/q)\} + R^*(T+1-t) - R^*\tau_M \\ &= \rho_M D - R^*\tau_M + W_t(-) \end{aligned} \tag{6}$$

Since the continue action is optimal when $X = M$ at time t, it then follows that

$$\Delta_t(M) = C_t(M) - W_t(-) = \rho_M D - R^*\tau_M = (\rho_M R^*)(D/R^* - \tau_M/\rho_M) \geq 0. \tag{7}$$

Numerical examples show that $\Delta_t(M)$ is not monotone in M, creating challenges in establishing conditions for the optimality of threshold policies. Fortunately, we are able to leverage the property that $\Delta_t(M)$ may be written in product form, with the first term $(\rho_M R^*)$ capturing the modulating effect of R^* and the local conditional probability of success ρ_M. The second term (D/R^*) has a constant D that captures parameters independent of M, while the last term (τ_M/ρ_M) contains only operational parameters that depend on M. To that end, using (7) we can write (the lower bound of) the risk-return adjusted benefit of continuing at M as:

$$\Theta_t(M) = \Delta_t(M)/(\rho_M R^*) = (D/R^* - \tau_M/\rho_M) \geq 0. \tag{8}$$

Now that we have expressed the optimality condition when $X = M$, we use induction to establish the optimality of a threshold policy. To initiate the induction, we first have to establish that if it is optimal to continue if $X = M$, then it is also optimal to continue when $X = M-1$ at time $t - \tau_{M-1}$. In other words, we need to show that $C_{t-\tau_{M-1}}(M-1) - W_{t-\tau_{(M-1)}}(-) \geq 0$. Establishing this result directly requires that the optimal value-to-go $V_t(M)$ be available, which requires, among other factors, that the value for R^* is known. Unfortunately, R^* is not available a priori. Nevertheless, it turns out that if we assume that the operating parameters are so ordered that

$$\tau_X/\rho_X - \tau_{X-1}/\rho_{X-1} \geq 0 \text{ for } X = 2, 3, \cdots, S, \tag{9}$$

then the optimality of the threshold policy can be established by using $W_t(-)$ as a lower bound for $V_t(M)$. To see this, notice that

$$\begin{aligned}
\Delta_{t-\tau_{M-1}}(M-1) &= C_{t-\tau_{M-1}}(M-1) - W_{t-\tau M-1}(-) \\
&= \rho_{M-1}\{r + R^*(T+1-t) + (1-\rho_{M-1})V_t(M)\} - W_{t-\tau_{M-1}}(-) \\
&\geq \rho_{M-1}\{r + R^*(T+1-t) + (1-\rho_{M-1})W_t(M)\} - W_{t-\tau_{M-1}}(-) \\
&= \rho_{M-1}D – R^*\tau_{M-1} + W_t(-) - W_{t-\tau_{M-1}}(-) \\
&= (\rho_{M-1}R^*)(D/R^* - \tau_{M-1}/\rho_{M-1}) \equiv (\rho_{M-1}R^*)\Theta_{t-\tau_{M-1}}(M-1) \\
&\geq (\rho_{M-1}R^*)(D/R^* – \tau_M/\rho_M) = (\rho_{M-1}R^*)\Theta_t(M) \geq 0. \qquad (10)
\end{aligned}$$

In the derivation above, the first inequality follows because a lower bound for $V_t(M)$ is evoked, the second inequality follows since we assumed that $\tau_M/\rho_M – \tau_{M-1}/\rho_{M-1} \geq 0$ and the last inequality follows because of (8), which is a consequence of the optimality of the continue action when $X = M$. One implication of the above is that risk-return adjusted quantities $\Theta_t(X)$ are monotone decreasing, (recall that the $\Delta_t(X)$'s are not necessarily monotone), which gives enough regularity to establish that the threshold policy is optimal.

Replicating this logic recursively establishes the optimality of the threshold policy under this sufficient condition. The sufficient condition $\tau_X/\rho_X - \tau_{X-1}/\rho_{X-1} \geq 0$ for $X = 2, 3, \cdots, S$ is remarkable because it is independent of all economic parameters of the problem. It may be interpreted as a variant of the weighted shortest processing time rule (WSPT in Chapter 9 of Pinedo (1995)). However, we weight the processing times by conditional probabilities. Thus, our condition may be viewed as a conditional risk-adjusted version of WSPT.

In our special variant, since no state variables are given, it may be hard to envisage a practical setting in which a non-threshold policy could arise. After all, once a customer is transferred, there is no possibility of continuing. However, if for some reason a customer were not transferred at the first instance when it was optimal, it is possible that continue is optimal in subsequent decisions. Such instances could and do arise when there are multiple states. For example, at X it may be optimal to transfer when $Q = 1$ but continue when $Q = 0$. So, it can happen that the customer transitions to $X + 1$, and there, the agent finds an unresolved issue when $Q = 1$. If the policy is threshold for $Q = 1$, then the customer would by definition be transferred at this step.

Since we have four congestion states characterized by binary values for Q and A, the direct approach we used to get (6) may not be used, since in states other than the focal state being considered at M, the optimal action may be to continue not only at M but also continue at $M + 1$. Nevertheless, we can find a sufficient condition for the optimality of threshold policies that is closely related to (9) by using: an upper bound for $\mathcal{N}_t(M + 1)$, an upper bound on $C_t(M)$ that mimics (6) and relies on Theorem 3, and a lower bound for $\mathcal{N}_\iota(M+1)$ that we use to get a condition that mimics (9). This sufficient condition comes in two variants, C-WSPT-a and C-WSPT-q. Which one applies depends on the relative ranking of a and q. Since this development is less insightful, its details are deferred to the proof of Theorem 4 in EC.2; however, the result is formalized next:

THEOREM 4 *A threshold policy is an optimal solution if either of the following conditions holds:*

(i) Condition C-WSPT-a: If $1/a > 1/q$ *and for* $X = 2, 3, \cdots, S$,
$\tau_X/\rho_X - \tau_{X-1}/\rho_{X-1} \leq (1/a)/\rho_X + [(1-a)/a](1-\rho_{X-1})/\rho_{X-1}$*; or*

(ii) Condition C-WSPT-q: If $1/a \leq 1/q$ *and for* $X = 2, 3, \cdots, S$,
$\tau_X/\rho_X - \tau_{X-1}/\rho_{X-1} \leq (1/q)/\rho_X + [(1-q)/q](1-\rho_{X-1})/\rho_{X-1}$.

As in (9), each of the two conditions in Theorem 4 compares conditional risk-return adjusted processing times. However, unlike in (9), (Q,A) are observed; hence, there are contingencies that must be accommodated. In particular, for threshold policies to be optimal, the weighted processing times must be sufficiently apart. Specifically, in each case the first term on the right-hand side can be interpreted as an "information premium" that is charged because we cannot know a priori at step $X-1$ whether continuing will be optimal at step X. Similarly, there is a "congestion" premium of either $q/((1-q)$ or $a/(1-a)$ since we cannot know ahead of time whether a transfer will entail a wait. Despite the added complexity from accounting for the state (Q,A), Theorem 4 establishes that a variant of a weighted shortest processing time provides an intuitive sufficient condition for optimality of threshold policies.

5 Channel Selection: Two-channel Case

In this section, we turn our attention to the broader problem of designing a channel architecture, i.e., selecting the optimal combination of service channels to maximize the firm's profit. To maintain focus, we limit our analysis to a two-channel system with homogeneous customers, where the firm chooses between offering a live-agent channel, a chatbot channel, or both. While the presented approach can be easily generalized to accommodate more than two channels or heterogeneous customers, we omit these extensions for the sake of brevity and clarity. As in the previous sections, we prioritize intuition and key results, with technical details provided in EC.4 - EC.7.

5.1 Overview

To examine the firm's problem of selecting a channel architecture, we revisit the profit-maximization problem from (1) in §2.1:

$$\max_{\mathcal{A},\mathcal{B},D^{agent},p^{bot}_{succ}} \mathcal{A}\cdot\pi^{agent}(D^{agent},\lambda^{agent}(D^{agent},p^{bot}_{succ})) + \mathcal{B}\cdot\pi^{bot}(p^{bot}_{succ},\lambda^{bot}(D^{agent},p^{bot}_{succ})), \tag{11}$$

We have already characterized the live-agent resolution policy D^{agent} in §3-4. Note, however, that we have substituted the resolution policy D^{bot} from (1) with its chatbot-specific equivalent, a single scalar measure p^{bot}_{succ}. In §5.2 we will show how the firm would optimally set p^{bot}_{succ}.

While the problem in (11) is formulated from the firm's profit-maximization perspective, the firm must balance both cost and quality considerations. This is because the channel policies D^{agent} and p^{bot}_{succ} give rise to channel-specific performance measures (such as transfer probabilities and expected service times), which in turn affect service quality. We characterize these effects in §5.3, where we solve the customers' channel choice problem and calculate demand rates for each channel $(\lambda^{agent}, \lambda^{bot})$ as a response to exogenously given service performance measures. Then, in §5.4 we reconcile the firm's and the customers' problems by endogenizing the arrival rates to each channel. To do so, we set the demand rates expected by the firm equal to the arrival rates resultant from aggregate customer decisions (setting $\lambda^{agent} = q^{agent}, \lambda^{bot} = q^{bot}$) and solve (11) by evaluating it over all channel mix combinations and optimized resolution policies within each channel. Finally, in §5.5 we show in our numerical illustration that all three possible architectures – live-agent-only, chatbot-only, or both – can emerge as optimal, and examine how the macro-level design problem of selecting the architecture interacts with the micro-level choice of within-channel resolution policies.

5.2 The Chatbot Channel

Our interviews with BlackBeltHelp (described in §1) suggest that AI developers program chatbots to make a limited number of attempts to diagnose and resolve each request before transferring the customer to a human expert. This is similar to the S-attempt approach that we used to model the live-agent channel (§3). In the case of the chatbot channel, the developer conducts a cost-benefit analysis to choose which of the S attempts to program the chatbot to make, and to what degree of reliability. Since the chatbot's opportunity cost of time is negligible, it continues to make attempts until it has exhausted all attempts for a given request before transferring the customer. As a result, the chatbot's resolution policy is entirely characterized by its resolution probability, p^{bot}_{succ}, where the development cost function is increasing and convex in p^{bot}_{succ}. This functional form is directly supported by the extensive computer science literature discussing the diminishing returns of training domain-specific chatbots (Hoffmann et al. 2022, and references there).

The chatbot's resolution policy is further simplified by the limited nature of its action set (relative to that of the live agent). In particular, if the chatbot fails to resolve the request, the customer *must* be cold transferred to the expert. This is because the chatbot cannot perform warm transfers. Given that the chatbot will make up to all programmed attempts before transferring to the expert, it is as if the chatbot is programmed as a gatekeeper whose optimal policy is static in that it performs the sequential request resolution procedure (from §3) and then transfers upon failure. Note that such a static protocol is a special instance of Case 5 of Theorem 3 in Table 3. Finally, from the customer's perspective, there is no waiting time to access the chatbot. The chatbot admits all customers that arrive and they are served immediately without waiting.

5.3 Customer Channel Choice Model

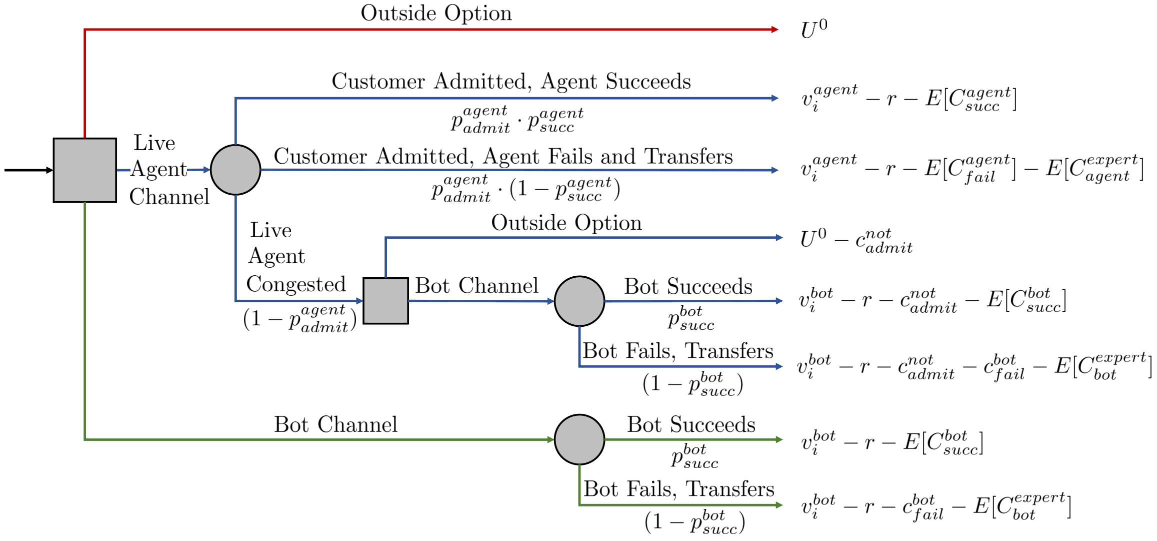
The live-agent and chatbot policies $(D^{agent}, p^{bot}_{succ})$ give rise to channel performance measures; specifically, the customers' admission probability (in the live-agent channel), transfer probabilities, and expected service times. Customers respond to these measures by choosing the channel (if any) that maximizes their utility. The resultant demand model is shown in Figure 3.

Each customer i chooses between joining the live-agent channel, the chatbot channel, and a fixed outside option U^0. From the customer's perspective, the outcome and the amount of time spent in each channel are uncertain. We represent this uncertainty with the circles (random nodes) in Figure 3. The live-agent channel has three possible outcomes: the agent succeeds, the agent transfers the customer to the expert, and the agent channel is congested. In the latter case, the customer is left with the choice to leave the system or to contact the chatbot. (Consistent with §3, customers that are not admitted do not wait). The chatbot channel is analogous, with the difference being that it is never congested. Customers act on their idiosyncratic technology preferences (channel-specific service valuations v_i^{agent} and v_i^{bot}). Their responses also depend on potential aversion to transfers (included in the $E[C^{expert}]$ terms), the relative comparison of waiting costs ($E[C]$ terms), and possible disutility of not being admitted to the live-agent channel (c^{not}_{admit}). In EC.4 we fully specify the utilities and solve for the demand rates for each channel, λ^{agent} and λ^{bot}, as a function of performance measures communicated to customers.

5.4 Channel Architecture Problem

We have so far assumed that the firm's problem (§3-4, §5.2) and the customers' problem (§5.3) are solved independently. That is, the firm's problem is solved under exogenously given arrival rates (q^{agent}, q^{bot}), while the channel choice model in §5.3 is solved under exogenously given performance measures $(D^{agent}, p^{bot}_{succ})$.

Figure 3 Customer Channel Choice Model


Outside Option
U^0
Customer Admitted, Agent Succeeds
$p^{agent}_{admit} \cdot p^{agent}_{succ}$
$v_i^{agent} - r - E[C^{agent}_{succ}]$
Live Agent Channel
Customer Admitted, Agent Fails and Transfers
$p^{agent}_{admit} \cdot (1 - p^{agent}_{succ})$
$v_i^{agent} - r - E[C^{agent}_{fail}] - E[C^{expert}_{agent}]$
Outside Option
$U^0 - c^{not}_{admit}$
Live Agent Congested
$(1 - p^{agent}_{admit})$
Bot Channel
Bot Succeeds
p^{bot}_{succ}
$v_i^{bot} - r - c^{not}_{admit} - E[C^{bot}_{succ}]$
Bot Fails, Transfers
$(1 - p^{bot}_{succ})$
$v_i^{bot} - r - c^{not}_{admit} - c^{bot}_{fail} - E[C^{expert}_{bot}]$
Bot Channel
Bot Succeeds
p^{bot}_{succ}
$v_i^{bot} - r - E[C^{bot}_{succ}]$
Bot Fails, Transfers
$(1 - p^{bot}_{succ})$
$v_i^{bot} - r - c^{bot}_{fail} - E[C^{expert}_{bot}]$

Linking the two problems requires that we endogenize channel demand such that we end up at a "rational expectations" equilibrium in which the firm delivers that which is promised to the customer, who in turn anticipates the fulfillment of this promise. We do this by solving for $\lambda^{agent} = q^{agent}$ and $\lambda^{bot} = q^{bot}$ such that arrival rates from the resolution model and demand rates from the customer choice model are set equal under a given choice of resolution policies $(D^{agent}, p^{bot}_{succ})$. To build managerial intuition, we do this for the special case of uniformly distributed service valuations (v^{agent}_i, v^{bot}_i). See EC.5 for details.

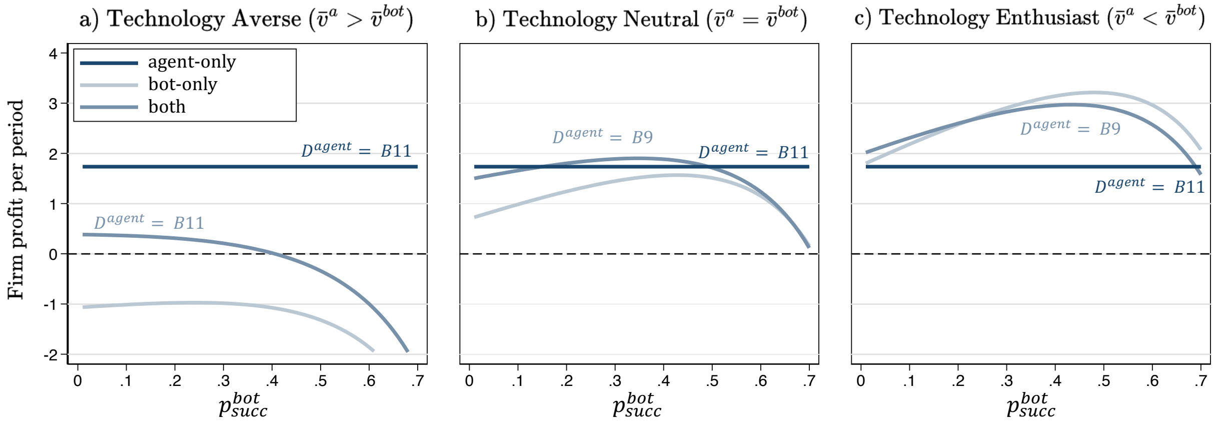
Once the profit maximization problem in (11) is fully specified, it can be solved by evaluating three decisions: 1) which live-agent resolution policy to employ (D^{agent}), 2) to what extent the chatbot is trained (p^{bot}_{succ}) and, 3) which channels to offer $(\mathcal{A}, \mathcal{B})$. In EC.6 we formulate the firm's profit function in (11) under the above inputs. In essence, for each architecture (agent-only, bot-only, both), the firm iterates through each admissible live-agent resolution policy from Theorem 3 and/or each potential chatbot success probability to find the input(s) that optimally balance channel demand, chatbot training costs, and expert resolution costs. Then the firm selects the most profitable of the three architectures.

5.5 Numerical Illustration

The following example illustrates the key trade-offs involved in choosing an appropriate channel architecture. To ensure that the example was most insightful, we evaluated diverse sets of parameters and functional forms before settling on a minimally-complex set of choices. While the remaining parameters and functions are described in detail in EC.7, we set $S = 2$ and ran three scenarios that differ in consumer attitudes toward chatbot technology; namely, technology-averse, technology-neutral and technology-enthusiastic.

Figure 4 shows the firm's profit per unit time under each architecture across p^{bot}_{succ} for each of the three technology acceptance levels. In all cases, the darkest blue line represents the agent-only architecture, which is a natural benchmark since its performance is necessarily independent of p^{bot}_{succ}. Unsurprisingly, when customers are technology-averse (panel a), the chatbot-only architecture (lightest-blue) is least profitable, while

Figure 4 Numerical Illustration



the agent-only is most profitable. When customer attitudes towards the chatbot technology are more neutral (panel b), the profitability of the chatbot-only architecture improves but it does not dominate the agent-only architecture. Nevertheless, the multichannel architecture (medium blue) is optimal for intermediate values of p_{succ}^{bot}. Finally, when consumers are technology enthusiasts (panel c), the agent-only architecture is always dominated and under sufficiently high values of p_{succ}^{bot}, the bot-only architecture emerges as optimal. Thus, depending on the success probability and attitude toward technology, each of the three architectures can be the most profitable.

Finally, recall that the optimal architecture depends on the optimal resolution policies, which in turn can differ by architecture. Consider the technology-neutral case (panel b). Under the agent-only architecture the optimal policy is T5, i.e., to cold transfer irrespective of the queue state (see Table 2). However, under the multichannel architecture, the policy is T3w, i.e., to warm transfer in some states. Since the presence of the bot decreases demand for the agent, the arrival rate to the agent decreases. With some of the customer inquiries diverted to the bot, the agent now has more time to offer customers this more personalized interaction. The resultant increase in service quality attracts some of the chatbot users back to the live agent, which lowers the demand for the bot. The sum of these effects leads to a lower training investment in the chatbot, since optimal investment is increasing in chatbot demand. Together, these comparisons highlight that firms need to consider both the strategic implications of AI development investments and the indirect effects on service delivery processes, in order to balance service quality and costs.

6 Conclusion

Building on empirical findings (interviews, industry scan and customer survey), we develop an analytical framework for designing and managing a multichannel network for customer service delivery. Within this framework, each channel is modeled as a gatekeeper system in which agents (or chatbots) act as gatekeepers receiving a stream of incoming customer requests. Requests are represented by an ordered list of potential solutions, where each solution has a known probability of solving the problem and a time it takes. Since we can induce stationarity by imposing appropriate terminal conditions, our formulation accommodates limited operating hours (e.g., urgent care clinics) as well as 24/7 operations (e.g., airline reservation desks). While the optimal control policy for this finite-horizon dynamic program can be complicated, we are able to show that under intuitively appealing conditions, a computationally efficient threshold policy is optimal.

Our analytical results reveal that the optimal control policy can have an appealing managerial structure. One such admissible policy is for the gatekeeper to continue troubleshooting the service request until it is resolved. We interpret such a policy as representative of a high-end, full-service firm, in which customers receive comprehensive concierge assistance from a single provider. In other policies, transfers may occur at some point during the troubleshooting process. One such mode of transferring is a warm transfer to an expert; by handing off the customer, the gatekeeper provides a more integrative, seamless experience for the customer, which (as our customer sentiment survey in §2 showed) is the transfer mode preferred by customers. However, when opportunity costs are high or the gatekeeper channel is congested, the optimal resolution policy may be to cold transfer, leaving it to the customer to restart their service request from scratch.

To focus on first-order trade-offs we have simplified certain aspects of the problem. We modeled upstream congestion as a waiting room that can accommodate at most one customer ($Q = 1$ or $Q = 0$) who abandons the service facility if not seen right away. Analogously, when transferring, the gatekeeper either finds the expert available or not ($A = 1$ or $A = 0$); but, unlike customers, the gatekeeper patiently waits until served. Given its underlying Markovian structure, our congestion model could be enriched by adding finite waiting rooms both upstream and downstream, but at the expense of increasing the number of admissible policies. Further, our model assumed that gatekeepers act in alignment with the firm. Examining incentive design for individual agents (gatekeepers), as well as contracting problems that arise between AI developers and their customer-facing clients, may offer interesting variations. Further extensions include settings with more than two channels, settings with heterogeneous customer requests, as well as settings with uncertain processing times.

Our research incorporates the effects of AI chatbots on service channel design, with a particular focus on the customer service sector. Looking forward, it would be valuable to examine how the gains from AI can be split in the most efficient as well as the most welfare-maximizing manner. Beyond customer service, our gatekeeper framework can be used as a building block to further advance our understanding of how AI chatbots can be leveraged in other customer-facing domains, such as healthcare, banking and education. Although these domains are still in the early stages of adopting chatbots, they offer significant potential for improving service quality and reducing costs.

References

Aircall. 2023. Customer service automation: pros, pitfalls, and best practices. URL `https://aircall.io/blog/customer-happiness/customer-service-automation/`. Accessed: 2023-11-24.

Alizamir, Saed, Francis De Véricourt, Peng Sun. 2013. Diagnostic accuracy under congestion. *Management Science*, 59 (1), 157-171.

Allon, Gad, Mirko Kremer. 2018. Behavioral foundations of queueing systems. *The Handbook of Behavioral Operations*, 9325.

Ansari, Asim, Carl F Mela, Scott A Neslin. 2008. Customer channel migration. *Journal of Marketing Research*, 45 (1), 60-76. doi:10.1509/jmkr.45.1.060.

Armony, Mor, Constantinos Maglaras. 2004. On customer contact centers with a call-back option: Customer decisions, routing rules, and system design. *Operations Research*, 52 (2), 271-292.

Batt, Robert J, Christian Terwiesch. 2017. Early task initiation and other load-adaptive mechanisms in the emergency department. *Management Science*, 63 (11), 3531-3551.

Brynjolfsson, Erik, Yu Jeffrey Hu, Mohammad S Rahman. 2013. *Competing in the age of omnichannel retailing*, vol. 1. MIT Cambridge.

Cohen, Maxime C. 2018. Big data and service operations. *Production and Operations Management*, 27 (9), 1709-1723.

Das Gupta, Aparupa, Uday S Karmarkar, Guillaume Roels. 2016. The design of experiential services with acclimation and memory decay: Optimal sequence and duration. *Management Science*, 62 (5), 1278-1296.

De Véricourt, Francis, Yong-Pin Zhou. 2005. Managing response time in a call-routing problem with service failure. *Operations Research*, 53 (6), 968-981.

Dietvorst, Berkeley J, Joseph P Simmons, Cade Massey. 2015. Algorithm aversion: people erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General*, 144 (1), 114.

Feldman, Pnina, Andrew E Frazelle, Robert Swinney. 2023. Managing relationships between restaurants and food delivery platforms: Conflict, contracts, and coordination. *Management Science*, 69 (2), 812-823.

Flicker, Blair, Charlie Hannigan. 2022. On people's utility over wait fundamentals and information.

Freeman, Michael, Nicos Savva, Stefan Scholtes. 2017. Gatekeepers at work: An empirical analysis of a maternity unit. *Management Science*, 63 (10), 3147-3167.

Gallino, Santiago, Antonio Moreno. 2019. *Operations in an omnichannel world*. Springer.

Gans, Noah, Yong-Pin Zhou. 2007. Call-routing schemes for call-center outsourcing. *Manufacturing & Service Operations Management*, 9 (1), 33-50.

Gao, Fei, Xuanming Su. 2017. Omnichannel retail operations with buy-online-and-pick-up-in-store. *Management Science*, 63 (8), 2478-2492.

Gao, Fei, Xuanming Su. 2018. Omnichannel service operations with online and offline self-order technologies. *Management Science*, 64 (8), 3595-3608.

Hasija, Sameer, Edieal J Pinker, Robert A Shumsky. 2005. Staffing and routing in a two-tier call centre. *International Journal of Operational Research*, 1 (1/2), 8-29.

Hathaway, Brett A, Evgeny Kagan, Maqbool Dada. 2023. The gatekeeper's dilemma:"when should i transfer this customer?". *Operations Research*, 71 (3), 843-859.

Hoffmann, Jordan, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, .

Kagan, Evgeny, Maqbool Dada, Brett Hathaway. 2023. Ai chatbots in customer service: Adoption hurdles and simple remedies. *Available at SSRN*, .

Kremer, Mirko, Francis de Véricourt. 2023. Mismanaging diagnostic accuracy under congestion. *Operations Research*, 71 (3), 895-916.

Kremer, Mirko, Laurens Debo. 2016. Inferring quality from wait time. *Management Science*, 62 (10), 3023-3038.

Kumar, Piyush, Maqbool Dada. 2021. Investigating the impact of service line formats on satisfaction with waiting. *International Journal of Research in Marketing*, 38 (4), 974-993.

Leclerc, France, Bernd H Schmitt, Laurette Dube. 1995. Waiting time and decision making: Is time like money? *Journal of consumer research*, 22 (1), 110-119.

Lee, Hsiao-Hui, Edieal J Pinker, Robert A Shumsky. 2012. Outsourcing a two-level service process. *Management Science*, 58 (8), 1569-1584.

Lee, Sampson. 2008. To create advocates, you have to differentiate your organization. *CustomerThink*, URL `https://customerthink.com/create_advocates_differentiate/`. Accessed: 12/02/2023.

Lemon, Katherine N, Peter C Verhoef. 2016. Understanding customer experience throughout the customer sourney. *Journal of Marketing*, 80 (6), 69-96. doi:10.1509/jm.15.0420.

Lund, Donald J, Detelina Marinova. 2014. Managing revenue across retail channels: The interplay of service performance and direct marketing. *Journal of Marketing*, 78 (2), 23-39. doi:10.1509/jm.13.0220.

Luo, Xueming, Siliang Tong, Zheng Fang, Zhe Qu. 2019. Frontiers: Machines vs. humans: The impact of artificial intelligence chatbot disclosure on customer purchases. *Marketing Science*, 38 (6), 937-947.

Maya, Viktoriya. 2023. Warm transfer: What it is, best practices and more. `https://customersfirstacademy.com/warm-transfer-in-customer-service/`. Accessed: 2023-12-06.

Moneypenny. 2021. What is a warm transfer? URL `https://www.moneypenny.com/us/resources/blog/what-is-a-warm-transfer/`. [Online; accessed 27-Jan-2023].

Naor, Pinhas. 1969. The regulation of queue size by levying tolls. *Econometrica: journal of the Econometric Society*, 15-24.

Pinedo, Michael. 1995. Scheduling. theory, algorithms and systems. *ISBN0-13-706757-7*, .

Saleseforce. 2020. Digital customer service channels. URL `https://www.salesforce.com/blog/digital-customer-service-channels/`. Accessed: 2023-11-24.

Senawi, Dounia, Tim McDougal, Jaden Herrin, Leah Yousif, Michael Kottwitz. 2023. New realities drive new models for contact center transformation. Technical Report, Deloitte Consulting LLP. URL `https://www.deloittedigital.com/content/dam/deloittedigital/us/documents/offerings/offering-20230426-gcs-survey-report.pdf`. Accessed: 2023-12-06.

Sheehan, Ben, Hyun Seung Jin, Udo Gottlieb. 2020. Customer service chatbots: Anthropomorphism and adoption. *Journal of Business Research*, 115 14-24.

Shumsky, Robert A, Edieal J Pinker. 2003. Gatekeepers and referrals in services. *Management Science*, 49 (7), 839-856.

E-Companion to: *Operational Strategies for Customer Service*

EC.1 Empirical Foundations

EC.1.1 Fortune 100 Scan

Our scan of the customer service channel architecture of the 2022 Fortune 100 companies was conducted between 9/11/2023 and 9/25/2023 by research assistants (RA's). For each corporate website, an RA searched through the "Contact Us" or "Help" section for various service channels. To focus on channels that are easily accessible to customers, RA's limited their search to no more than three mouse clicks from the landing page. Each company was assigned to a sector based on the Fortune 100 classification and the data was aggregated into Table 1. Results for all 100 companies are available from the authors.

EC.1.2 Customer Survey

The survey was conducted in July 2023 on the Prolific platform (`www.prolific.co`). A total of 202 respondents were recruited from the US-based population (49% female, average age: 38). All respondents received a show up payment of $3.00 and an additional payment of $2.00 at the end of the study. The survey included a panel of questions related to customer attitudes to customer service offerings. Here we report on the questions related to transfers. The questions, along with a summary of responses, are reproduced below.

Imagine the following situation. You need assistance from customer service, for example, to fix your WiFi service, or your phone plan. You have called and connected with a live agent who has attempted to resolve your problem. But the agent must now transfer you to a different queue which is staffed by experts who can solve the issue. The agent may transfer you to the expert queue using one of two methods:

> *Cold Transfer: The agent puts you into the expert queue and immediately disconnects. After possibly waiting for an expert to answer, you introduce yourself to the expert, and re-explain your issue to the expert. The expert then resolves your problem.*
>
> *Warm Transfer: The agent puts you into the expert queue but stays with you on the call. After possibly waiting for an expert to answer, the agent introduces you to the expert, and explains your issue to the expert on your behalf and then leaves the call. The expert then resolves your problem.*

Please answer the following questions:

Q1: *Given the choice between the following two alternatives, what would you choose?* (Cold: 28.22% warm: 71.78%)

Q2: *Aiming for 1-2 sentences, explain which transfer method you prefer and why.* [Respondents enter text]

Q3: *Have you ever been cold transferred?* (Yes: 90.09%, No: 9.91%)

Q4A: *If so, think back to the last time you were cold transferred. What was your satisfaction with the cold transfer experience?* (Average: 2.54 out of 5)

Q4B: *If not, what do you think your satisfaction with the cold transfer experience would be?* (Average: 2.45 out of 5)

Q4 A+B: Average (weighted by number of respondents in Q4A and Q4B): 2.53 out of 5

Q5: *Have you ever been warm transferred?* (Yes: 69.31%, No: 30.69%)

Q6A: *If so, think back to the last time you were warm transferred. What was your satisfaction with the warm transfer experience?* (Average: 3.94 out of 5)

Q6B: *If not, what do you think your satisfaction with the warm transfer experience would be?* Average: 3.47 out of 5)

Q6 A+B: Average (weighted by number of respondents in Q4A and Q4B): 3.80 out of 5

EC.2 Proofs

Proof of Theorem 1A. Focusing on the arguments of the max operator in (2.5), we observe that the value of continuing (the first argument) and the value of warm transferring (the second argument) do not depend on the current queue state (Q). However, the value of cold transferring (the third argument) depends on Q. This allows us to partially rank the payoffs of cold transferring, depending on Q and A, with the following lemma:

LEMMA 1A *For any t, $V_t(0^i,1,-) \; - \; V_t(0^i,0,-) \geq 0$.*

Proof of Lemma 1A. From (2.1),

$$(1-q)V_{t+1}(0^i,0,-)+qV_{t+1}(0^i,1,-)=V_t(0^i,0,-).$$

Expanding the first term gives

$$V_{t+1}(0^i,0,-)-qV_{t+1}(0^i,0,-)+qV_{t+1}(0^i,1,-)=V_t(0^i,0,-).$$

Subtracting the first term from both sides and factoring the left-hand side gives

$$q(V_{t+1}(0^i,1,-)-V_{t+1}(0^i,0,-))=V_t(0^i,0,-)-V_{t+1}(0^i,0,-).$$

Finally, dividing both sides by q gives

$$V_{t+1}(0^i,1,-)-V_{t+1}(0^i,0,-)=(V_t(0^i,0,-)-V_{t+1}(0^i,0,-))/q \geq 0.$$

Here the inequality follows because expected benefits must be non-decreasing as the number of periods increases. Then an induction argument completes the proof of Lemma 1A.

Returning to the proof of Theorem 1A, suppose that, for a fixed A and X, it is optimal to warm transfer when $Q = 1$. It follows from (2.5) that

$$-(c^w - r) + V_t(0^w, -, A) \geq \mathcal{N}_t(X), \tag{EC.1}$$

and

$$-(c^w - r) + V_t(0^w, -, A) \geq -(c^c - r) + V_t(0^i, Q = 1, -). \tag{EC.2}$$

Then (EC.1) also holds when $Q = 0$ because none of the values depend on Q. Further, (EC.2) also holds if $Q = 0$ because: 1) the value of the left side does not depend on Q and 2) from Lemma 1A, the value of the right side will be no greater when $Q = 0$ than when $Q = 1$, since $-(c^c - r) + V_t(0^i, 1, -) \geq -(c^c - r) + V_t(0^i, 0, -)$.

This same approach can be also be used to show that if it is optimal to continue when $Q = 1$, then it is optimal to continue when $Q = 0$. Specifically, this can be done by swapping the left and right sides in (EC.1) and replacing the left side of (EC.2) with the value-to-go function $\mathcal{N}_t(X)$.

Proof of Theorem 1B. Focusing on the arguments of the max operator in (2.5), we observe that the value of continuing (the first argument) and the value of cold transferring (the third argument) do not depend on current expert availability (A). However, the value of warm transferring (the second argument) does depend on A. This allows us to partially rank the payoffs of warm transferring, depending on Q and A, with the following lemma:

LEMMA 1B *For any t, $V_t(0^w, -, 1) \; - \; V_t(0^w, -, 0) \geq 0$.*

Proof of Lemma 1B. Using the same approach as Lemma 1A, we can rearrange (2.3) to be

$$V_{t+1}(0^w, -, 1) - V_{t+1}(0^w, -, 0) = (V_t(0^w, -, 0) - V_{t+1}(0^w, -, 0))/a \geq 0$$

Again, the inequality follows because expected benefits must be non-decreasing as the number of periods increases. Then an induction argument completes the proof.

Returning to the proof of Theorem 1B, suppose that, for a fixed Q and X, it is optimal to cold transfer when $A = 1$. It follows from (2.5) that

$$-(c^c - r) + V_t(0^i, Q, -) \geq \mathcal{N}_t(X) \tag{EC.3}$$

and

$$-(c^c - r) + V_t(0^i, Q, -) \geq -(c^w - r) + V_t(0^w, -, A = 1). \tag{EC.4}$$

Then (EC.3) also holds when $A = 0$ because none of the values depend on A. Further, (EC.4) also holds if $A = 0$ because: 1) the value of the left side does not depend on A, and 2) from Lemma 1B, the value of the right side will be no greater when $A = 0$ than when $A = 1$, since $-(c^w - r) + V_t(0^w, -, 1) \geq -(c^w - r) + V_t(0^w, -, 0)$.

This same approach can be also be used to show that if it is optimal to continue when $A = 1$, then it is optimal to continue when $A = 0$. Specifically, this can be done by swapping the left and right sides in (EC.3) and replacing the left side of (EC.4) with the value-to-go function $\mathcal{N}_t(X)$.

Proof of Theorem 2. We need to find $G(X, Q, A)$, for $X \in \{1, 2, \cdots, S\}$ in each of the four congestion states (Q, A) and $G(0^i, Q, -)$ for $X = 0^i$ in each of the two queue states $(Q = 0, Q = 1)$, along with $G(0^w, -, A)$ for $X = 0^w$ in each expert availability state $(A = 0, A = 1)$. For convenient exposition, the derivation steps in this proof are named after the sub-equation of (2) in the dynamic programming formulation that we use to construct the constant:

- **Step 0 - Normalize** $G(0^i, 0, -)$**:** Without loss of generality, we normalize $G(0^i, 0, -)$ to 0.
- **Step 2.1 - Specify** $G(0^i, 1, -)$**:** Substituting (4) into (2.1), we obtain $G(0^i, 1, -) = R^*/q$.
- **Step 2.4 - Specify** $G(0^w, -, 1)$**:** Substituting (4) into (2.4), we obtain

$$V_t(0^w, -, 1) = R^*(T + 1 - t) + G(0^w, -, 1) =$$

$$R^*(T + 1 - t) - R^*\tau_w + (1 - q)G(0^i, 0, -) + qG(0^i, 1, -), \text{ or,}$$

$$R^*(T + 1 - t) + G(0^w, -, 1) = R^*(T + 1 - t) - R^*\tau_w + R^*.$$

  Solving for $G(0^w, -, 1)$, we obtain $= G(0^w, -, 1) = -R^*(\tau_w - 1)$

- **Step 2.3 - Specify** $G(0^w, -, 0)$**:** Substituting (4) into (2.3), we obtain

$$G(0^w, -, 0) = -R^*(a(\tau_w - 1) + 1)/a.$$

- **Step 2.6 - Specify** $G(S, 0, 0), G(S, 1, 0), G(S, 0, 1), G(S, 1, 1)$**:** Substituting the previously-derived constants into (2.6), we obtain $G(S, 0, 0) = G(S, 0, 1) = G(S, 0, -) = r$, and $G(S, 1, 0) = G(S, 1, 1) = G(S, 1, -) = r + R^*/q$.
- **Step 2.5 - Specify** $G(X, Q, A)$**, for** $X = 1, \cdots, S - 1$**:** Since we do not know the optimal policy, we will proceed as if R^* is known and recursively find these values by working down from $X = S - 1$. At each X in the recursion these expressions depend on which of the three actions is optimal when the issue is not resolved after failing attempt X and which of the four congestion states (Q, A) the system is in. Since we only know the optimal policy for a specific set of parameters, we will solve for each constant for each possible action. Working backwards also allows us to express the payoff from continuing at time t after a given failure X as: $R^*(T + 1 - t) + N_{X+1}$, where, for $X = S - 1, \cdots, 1$,

$$N_{X+1} = -R^*\tau_{X+1} + (1 - q)(1 - a)G(X + 1, 0, 0) + q(1 - a)G(X + 1, 1, 0) + \cdots$$

$$(1-q)aG(X+1,0,1)+qaG(X+1,1,1). \qquad \text{(EC.5)}$$

Then substituting the above constants into (2.5), we obtain

$$G(X,Q,A)=\begin{cases}\rho_X(r+G(0^i,Q,-))+(1-\rho_X)N_{X+1}, & \text{if continue is optimal,}\\ \rho_X(r+G(0^i,Q,-))+(1-\rho_X)(G(0^w,-,A)\text{–}(c^w-r)), & \text{if warm transfer is optimal,}\\ \rho_X(r+G(0^i,Q,-))+(1-\rho_X)(G(0^i,Q,-)\text{–}(c^c-r)), & \text{if cold transfer is optimal.}\end{cases}$$

Finally, note that in each of the three expressions above, the first term is the same, and the second terms are all multiplied by $(1-\rho_X)$. Hence, the optimal decision rule simplifies to

$$\operatorname{argmax}\{N_{X+1},-(c^c-r)+G(0^i,Q,-),-(c^w-r)+G(0^w,-,A)\}. \qquad \text{(EC.6)}$$

- **Step 2.2 - Solve for R^*:** The values of the constants from (2.5) depend on which of the three actions is optimal for each of the four congestion states. Regardless, once we have completed the recursion, as a function of R^*, we would have populated all $4(S+1)$ values of the constants using all but (2.2). Finally, substituting the constants into (2.2) and simplifying, we have

$$R^*/q=-R^*\tau_1+(1-q)(1-a)G(1,0,0)+q(1-a)G(1,1,0)+(1-q)aG(1,0,1)+qaG(1,1,1)$$

  This is a linear function of R^* and hence may be solved uniquely for R^*.

Proof of Theorem 3a. From (2.5), cold transfer is preferred to warm transfer if

$$[-(c^c-r)+V_t(0^i,Q,A)]-[-(c^w-r)+V_t(0^w,Q,A)]\geq 0;\ \text{or,}$$

$$V_t(0^i,Q,A)-V_t(0^w,Q,A)\geq c^c-c^w>0.$$

By substituting the terminal conditions from (4) into the above and reducing, we obtain that cold transfer is preferred to warm transfer for a given congestion state (Q,A) if

$$G(0^i,Q,-)\text{–}G(0^w,-,A)\geq c^c-c^w>0. \qquad \text{(EC.7)}$$

Note that each side of the inequality is independent of X, meaning that the ranking depends only on the congestion state. Substituting the constants from the proof of Theorem 2 into (EC.7), we solve this inequality for R^* over each of the four congestion states (Q,A) as follows:

$$R^*\geq\begin{cases}(c^c-c^w)/(\tau_w-1+\frac{1}{a})\equiv\mathcal{R}(0,0), & \text{if } Q=0,A=0,\\ (c^c-c^w)/(\tau_w-1+\frac{1}{a}+\frac{1}{q})\equiv\mathcal{R}(1,0), & \text{if } Q=1,A=0,\\ (c^c-c^w)/(\tau_w-1)\equiv\mathcal{R}(0,1), & \text{if } Q=0,A=1,\\ (c^c-c^w)/(\tau_w-1+\frac{1}{q})\equiv\mathcal{R}(1,1), & \text{if } Q=1,A=1.\end{cases}$$

Proof of Theorem 3b. Because the numerator in each $\mathcal{R}(\cdot,\cdot)$ is the same, then comparing the denominators allows us to order them. $\mathcal{R}(1,0)$ is the minimum since its denominator is the largest ($\tau_w-1+\frac{1}{a}+\frac{1}{q}$). $\mathcal{R}(0,1)$

is the maximum since its denominator is the smallest ($\tau_w - 1$). Finally, $\mathcal{R}(0,0)$ and $\mathcal{R}(1,1)$ fall in between since their denominators are $\tau_w - 1 + \frac{1}{a}$ and $\tau_w - 1 + \frac{1}{q}$, respectively. Specifically, we have $0 \le \mathcal{R}(1,0) \le \mathcal{R}(1,1) \le \mathcal{R}(0,0) \le \mathcal{R}(0,1)$, if $1/q \le 1/a$; otherwise, $0 \le \mathcal{R}(1,0) \le \mathcal{R}(0,0) \le \mathcal{R}(1,1) \le \mathcal{R}(0,1)$

Proof of Theorem 3c. First, denote the minimum of the $\mathcal{R}(\cdot,\cdot)$ by $\mathcal{R}_1$, the second-lowest of the above by $\mathcal{R}_2$, the third-lowest by $\mathcal{R}_3$, and the maximum by $\mathcal{R}_4$. Note that $\mathcal{R}_1$ is always $\mathcal{R}(1,0)$, $\mathcal{R}_4$ is always $\mathcal{R}(0,1)$, but $\mathcal{R}_2$ ($\mathcal{R}_3$) may be either $\mathcal{R}(0,0)$ or $\mathcal{R}(1,1)$, depending on the values of $1/a$ and $1/q$. Thus, there are five regions where R^* may be located: 1) $R^* < \mathcal{R}_1$, 2) $\mathcal{R}_1 \le R^* < \mathcal{R}_2$, 3) $\mathcal{R}_2 \le R^* < \mathcal{R}_3$, 4) $\mathcal{R}_3 \le R^* < \mathcal{R}_4$, and 5) $\mathcal{R}_4 \le R^*$. Within each of these five regions, for a given (Q,A), only one transfer method (cold transfer or warm transfer) will be preferred across all failures (X). Hence, for each region, we can determine which of the 11 decision rules from Table 2 are potentially optimal under stationarity by admitting only the vectors where, for each (Q,A), the optimal action is either to continue or to move the customer to the expert using the method that is more economical. By inspection of Table 3c, we see that for a given problem instance, in an optimal policy there are only four admissible decision rule vectors: the continue rule, at most, two hybrid rules, and one transfer rule. Note that Table 3 is for the case of $1/q \ge 1/a$. For completeness, we include the $1/q < 1/a$ case below in Table EC.1.

Table EC.1 Ranges, Preferred Transfer Method, and Admissible Decision Rules Over Cases of R^* $(1/q \le 1/a)$

Table 3a: Ranges

Case	Case 1	Case 2	Case 3c	Case 4	Case 5
R^*	$R^* < \mathcal{R}(1,0)$	$\mathcal{R}(1,0) \le R^* < \mathcal{R}(0,0)$	$\mathcal{R}(0,0) \le R^* < \mathcal{R}(1,1)$	$\mathcal{R}(1,1) \le R^* < \mathcal{R}(0,1)$	$\mathcal{R}(0,1) \le R^*$

Table 3b: Preferred Transfer Method

Q	A	Case 1	Case 2	Case 3c	Case 4	Case 5
0	0	Warm	Warm	Warm	Cold	Cold
1	0	Warm	Cold	Cold	Cold	Cold
0	1	Warm	Warm	Warm	Warm	Cold
1	1	Warm	Warm	Cold	Cold	Cold

Table 3c: Admissible Decision Rules

Label	Rule	Case 1	Case 2	Case 3c	Case 4	Case 5
C	[n,n,n,n]	✓	✓	✓	✓	✓
H1	[n,n,w,w]	✓	✓	✓		
H2	[n,c,w,w]		✓	✓		
H3	[n,c,w,c]				✓	
H4	[n,c,n,c]				✓	✓
T1	[w,w,w,w]	✓				
T2	[w,c,w,w]		✓			
T3c	[w,c,w,c]			✓		
T4	[c,c,w,c]				✓	
T5	[c,c,c,c]					✓

n = continue, w = warm transfer, c = cold transfer

Proof of Theorem 4. We only have to consider decisions after failed attempts ($1 \le X < S$) so the DP depends on (2.5) only. Let $X = M$ be the last attempt where continuing is optimal. Then we will use the following relationships:

1. The expected value of continuing at X is greater than or equal to the value of continuing at X, transferring at $X+1$ and then following the optimal policy thereafter.
2. When Cases 1, 2 or 3w apply, the expected value of transferring is less than or equal to $-c^w - R^*\tau_w - R^*(1-q)/q + R^*(T+1-t)$. This is because warm transfer dominates cold transfer when $(Q=1, A=1)$. Thus, when $X = M+1$, this is an upper bound on the value of the optimal decision. Conversely, the expected value of transferring is greater than or equal to $-c^w - R^*\tau_w - R^*(1-q)/q - R^*/a + R^*(T+1-t)$, because when $(Q=0, A=0)$, warm transfer dominates cold transfer in Cases 1 and 2 and is dominated by cold transfer in Case 3w. Hence, the optimal decision can be no worse than a warm transfer and thus serves as a valid lower bound.
3. Analogously, when Cases 3c, 4 or 5 apply, the expected value of transferring is less than or equal to $-c^c + R^*(T+1-t)$. This is because cold transfer dominates warm transfer when $(Q=1, A=1)$. Thus, when $X = M+1$, this is an upper bound on the value of the optimal decision. Conversely, the expected value of transferring is greater than or equal to $-c^c - R^*/q + R^*(T+1-t)$, because when $(Q=0, A=0)$, cold transfer dominates warm transfer in Cases 4 and 5 and is dominated by warm transfer in Case 3c. Hence, the optimal decision can be no worse than a cold transfer and thus serves as a valid lower bound.

Cases 3c, 4 or 5

When $X = M$ is the last attempt where the continue policy is optimal, then when $(Q=1, A=1)$, cold transfer is preferred to warm transfer so that

$$-c^c \geq -c^w - R^*\tau_w - R^*(1-q)/q.$$

Suppose that we continue in attempt $X-1$. Then denoting the value of cold transferring by $L_t(X-1, Q, A)$, from (2.5) we get

$$C_t(X-1,Q,A) - L_t(X-1,Q,A) \geq \rho_{X-1}(r - R^*(1-q)/q + R^*(T+1-t-\tau_{X-1})) + \cdots$$
$$(1-\rho_{X-1})\{-c^c - R^*(1-q)/q + R^*(T+1-t-\tau_{X-1})\} - \{-c^c + R^*(T+1-t)\}.$$

The inequality follows since we have a lower bound for the optimal action for X (the first term in braces) and an upper bound for the transfer action for $X-1$ independent of Q and A (the second term in braces). Rearrangement gives

$$C_t(X-1,Q,A) - L_t(X-1,Q,A) \geq \rho_{X-1}(r + c^c) + \{-(R^*(1-q)/q - R^*\tau_{X-1}\}$$
$$= R^*\rho_{X-1}\{(r+c^c)/R^* - (\tau_{X-1} + (1-q)/q)/\rho_{X-1}\}.$$

This implies that

$$\{C_t(X-1,Q,A) - L_t(X-1,Q,A)\}/\{R^*\rho_{X-1}\} \geq \{(r+c^c)/R^* - (\tau_{X-1} + (1-q)/q)/\rho_{X-1}\}$$

$$\equiv \{D^C/R^* - (\tau_{X-1} + (1-q)/q)/\rho_{X-1}\}.$$

Lemma 2 *If* $(\tau_{X-1} + (1-q)/q)/\rho_{X-1} \leq (\tau_X + (1-q)/q)/\rho_X$ *for* $X = 2,\cdots,S$ *then* $\{C_t(X-1,Q,A)–L_t(X-1,Q,A)\}/\{R^*\rho_{X-1}\} \geq \{C_t(X,Q,A)–L_t(X,Q,A)\}/\{R^*\rho_X\}$ *And, if* $\{C_t(X-1,Q,A)–L_t(X-1,Q,A)\}/\{R^*\rho_{X-1}\} \geq 0$*, then it is optimal to continue at attempts* $1,\cdots,X-1$.

Let us now assume that $X = M$ is the last attempt for which it is optimal to continue for a given (Q,A). Then for attempt $M+1$

$$-c^c \geq -c^w–R^*\tau_w–R^*(1-q)/q \text{ and } -c^c + R^*(T+1-t-\tau_{U-1}) \geq \mathcal{N}_t(M+1),$$

where the RHS of the second inequality is the value of continuing. Note that this relies on the upper and lower bounds previously established in point 3 at the beginning of the proof. Then

$$0 \leq C_t(M,Q,A)–L_t(M,Q,A) \leq \rho_M(r–R^*(1-q)/q + R^*(T+1-t–\tau_M)) + \cdots$$

$$(1–\rho_M)\{-c^c–R^*(1-q)/q + R^*(T+1-t–\tau_M)\}–\{-c^c–R^*/q + R^*(T+1-t)\}.$$

The last inequality follows since we have an upper bound for the optimal action when $X = M+1$ and a lower bound for the transfer action when $X = M+1$, independent of Q and A. Rearrangement gives

$$C_t(M,Q,A)–L_t(M,Q,A) \geq \rho_M(r+c^c) + \{-\rho_M R^*(1-q)/q - R^*\tau_M + R^*/q\}$$

$$= R^*\rho_M\{(r+c^c)/R^* - ((\tau_M - 1/q)/\rho_M + (1-q)/q)\} \equiv R^*\rho_M\{D^C/R^* - ((\tau_M - 1/q)/\rho_M + (1-q)/q)\}.$$

Now note by inspection that

Lemma 3 $(\tau_X–1/q)/\rho_X + (1-q)/q) \leq \tau_X/\rho_X + (1-q)/q) \leq (\tau_X + (1-q)/q))/\rho_X$.

Theorem *If* $(\tau_{X-1} + (1-q)/q)/\rho_{X-1} \leq (\tau_X - 1/q)/\rho_X + (1-q)/q)$ *for* $X = 2,\cdots,S$ *and* $X = M$ *is the last failed attempt for which it is optimal to continue for a given* (Q,A)*, then* $\{C_t(X-1,Q,A)–L_t(X-1,Q,A)\}/R^*\rho_{X-1} \geq 0$ *so that a threshold policy is optimal for* (Q,A).

The proof follows since the continue action is optimal for $X = M$ given (Q,A), and $D^C/R^* - (\tau_M + (1-q)/q)/\rho_M \geq 0$. Thus, the result follows from Lemma 2.

Interpretation: $(\tau_{X-1} + (1-q)/q)/\rho_{X-1} \leq (\tau_X - 1/q)/\rho_X + (1-q)/q)$ can be rewritten as

$$\tau_X/\rho_X - \tau_{X-1}/\rho_{X-1} \leq (1-q)/q)(1-\rho_{X-1})/\rho_{X-1} + (1/q)/\rho_X.$$

The first term on the RHS is a "premium" for continuing at X and the second term is the premium for not knowing whether to continue at $X+1$ when $(Q=1, A=1)$ is not true.

The proof for cases 1,2 and 3a is analogous except we use the fact that $-c^c < -c^w–R^*\tau_w–R^*(1-q)/q$, which we established through the upper and lower bounds provided in point 2 at the beginning of the proof. In any given problem instance, which of the two inequalities in Theorem 4 are sufficient depends on whether $1/a$ is greater than $1/q$.

EC.3 Threshold Policy Performance

To test the performance of threshold policies as a heuristic solution, we randomly generated 1,200,000 problem instances each for $S = 3$, $S = 4$, $S = 5$, and $S = 6$. In each problem instance, we found the admissible policy with the maximum profit per unit time (R^*) and compared it with that of the best-performing threshold policy (R^*_{TH}). We then calculated the optimality gap in each instance as $OG = 1 - R^*_{TH}/R^*$, and collected salient performance measures displayed in Table 4 in §4.2.1 ($OG = 0$ equates to a threshold policy being optimal). The following procedure was used to generate the random instances:

- **Initialize:** Set $S = 6$. Draw the handling time of each attempt ($\tau_s \in S$) as an integer-valued uniform random variable between 10 and 30, and the conditional resolution probability of all but the final attempt ($\rho_s \in s < S$) as random numbers (recall that the final attempt is guaranteed to resolve the request, i.e., $\rho_S = 1$). Join these two vectors to create a handling-time/resolution-probability matrix. Repeat this process to generate 1000 random matrices.
- **For each Matrix and for each** $S = \{3,4,5,6\}$**:**
  - —**Trim Matrix (If Necessary):** If $S < 6$, then trim the HTRP matrix by removing entries greater than S and by setting $\rho_S = 1$.
  - —**Find** *OG* **Across Grid:** Set $r = 100$ and calculate OG across a grid of operational and cost parameters as follows: $q = \{0.2, 0.4, 0.6, 0.8, 1\}$, $a = \{0.2, 0.4, 0.6, 0.8, 1\}$, $\tau_w = \{1,2,3\}$, $(c^w, c^c) = \{(40,50),\ (30,50),\ (20,50),\ (10,50),\ (0,50),\ (30,40),\ (20,40),\ (10,40),\ (0,40),\ (20,30),\ (10,30),\ (0,30),\ (10,20),\ (0,20),\ (0,10),\ (0,0)\}$.

Given that there were 1000 random matrices, 5 parameterizations of q, 5 parameterizations of a, 3 parameterizations of τ_w, and 16 parameterizations of (c^w, c^c), this resulted in 1,200,000 random problem instances for each of $S = 6$, $S = 5$, $S = 4$, and $S = 3$. ($1000 \times 5 \times 5 \times 3 \times 16$).

EC.4 Customer Channel Choice Model

EC.4.1 Utility Specifications

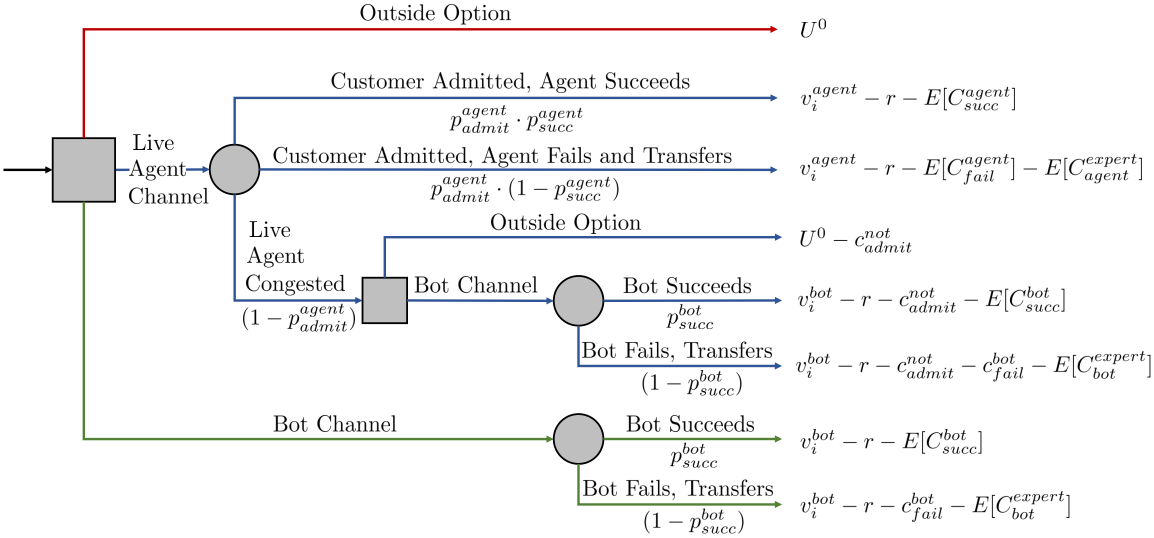
To determine the demand (arrival rate) for each channel, we measure the utility of a service as the value provided by that service net the waiting cost to receive service. Consider customer i. If the live agent (chatbot) is chosen, customer i receives v_i^{agent} (v_i^{bot}) once service is completed. The magnitudes of v_i^{agent} and v_i^{bot} can be reflective of the importance or urgency of the request; the difference between v_i^{agent} and v_i^{bot} can be interpreted as their willingness to use chatbot technology.

Since the underlying system is a network of queues, at each stage, the customer has an expected wait followed by an expected service. Thus, for a given $provider \in \{agent, bot, expert\}$, the expected disutility of waiting can be computed as:

$$E[C^{provider}] = c^{wait} \cdot E[W^{provider}] + c^{provider} \cdot E[H^{provider}],$$

where c^{wait} is the per period disutility of waiting, $E[W^{provider}]$ is the expected number of periods waiting for the provider (live agent, bot or expert), $c^{provider}$ is the per period disutility of being served by the provider, and $E[H^{provider}]$ is the expected number of periods being handled by the provider.

Figure EC.1 Customer Channel Choice Model


Outside Option
U^0
Customer Admitted, Agent Succeeds
$p^{agent}_{admit} \cdot p^{agent}_{succ}$
$v_i^{agent} - r - E[C^{agent}_{succ}]$
Live Agent Channel
Customer Admitted, Agent Fails and Transfers
$p^{agent}_{admit} \cdot (1 - p^{agent}_{succ})$
$v_i^{agent} - r - E[C^{agent}_{fail}] - E[C^{expert}_{agent}]$
Live Agent Congested
$(1 - p^{agent}_{admit})$
Outside Option
$U^0 - c^{not}_{admit}$
Bot Channel
Bot Succeeds
p^{bot}_{succ}
$v_i^{bot} - r - c^{not}_{admit} - E[C^{bot}_{succ}]$
Bot Fails, Transfers
$(1 - p^{bot}_{succ})$
$v_i^{bot} - r - c^{not}_{admit} - c^{bot}_{fail} - E[C^{expert}_{bot}]$
Bot Channel
Bot Succeeds
p^{bot}_{succ}
$v_i^{bot} - r - E[C^{bot}_{succ}]$
Bot Fails, Transfers
$(1 - p^{bot}_{succ})$
$v_i^{bot} - r - c^{bot}_{fail} - E[C^{expert}_{bot}]$

Consider first the initial channel choice represented by the leftmost square in Figure EC.1. If the outside option is chosen, the customer does not present for service and obtains U^0. Otherwise, additional calculations are required. We start with the chatbot channel. Denote by $E[C^{bot}_{succ}]$ the expected customer cost of being served by the bot if the bot succeeds, by c^{bot}_{fail} the cost of being served by the bot if the bot fails, and by $E[C^{expert}_{bot}]$ the cost of being served by the expert if the bot fails. Note that c^{bot}_{fail} is a constant since there is no stochastic waiting time for the bot and the time spent on a making all possible attempts before transferring is deterministic. Then the resulting expectation is a weighted average of two random outcomes and is given by

$$U_i^{bot} = v_i^{bot} - r - p^{bot}_{succ} \cdot E[C^{bot}_{succ}] - (1 - p^{bot}_{succ}) \cdot (c^{bot}_{fail} + E[C^{expert}_{bot}]) \equiv v_i^{bot} - L^{bot}.$$

Recall that p^{bot}_{succ} is the probability that the request will be resolved by the bot.

Similarly, if the agent channel is chosen and the customer is admitted (with probability p^{agent}_{admit}), then the resulting pathways mimic those of the bot channel. However, with probability $1 - p^{agent}_{admit}$ the channel is congested and the customer is not admitted. In that case, the customer incurs the cost of non-admission, c^{not}_{admit}, which represents the psychological cost (or the hassle cost) of trying but failing to access a service provider. Denote by $E[C^{agent}_{succ}]$ the expected customer cost of being served by the agent if the agent succeeds, by $E[C^{agent}_{fail}]$ the expected cost of being served by the agent if the agent fails, and by $E[C^{expert}_{agent}]$ the expected cost of being served by the expert if the agent fails. The per unit cost of waiting with the expert is generally higher than with the live agent (or chatbot), reflecting the added disutility incurred by the customer entering an additional waiting episode with a second provider. Then the expected utility of entering the agent channel is given by

$$\begin{aligned} U_i^{agent} = {} & p_{admit}^{agent}\cdot\left[v_i^{agent} - r - p_{succ}^{agent}\cdot E[C_{succ}^{agent}] - (1-p_{succ}^{agent})\cdot(E[C_{fail}^{agent}]+E[C_{agent}^{expert}])\right]\cdots \\ & +(1-p_{admit}^{agent})\cdot\left[-c_{admit}^{not}+\max\{U_i^{bot},U^0\}\right]\cdots \\ = {} & p_{admit}^{agent}\cdot\left[v_i^{agent} - r - p_{succ}^{agent}\cdot E[C_{succ}^{agent}] - (1-p_{succ}^{agent})\cdot(E[C_{fail}^{agent}]+E[C_{agent}^{expert}])\cdots\right. \\ & \left.-(1-p_{admit}^{agent})/p_{admit}^{agent}\cdot c_{admit}^{not}-\max\{U_i^{bot},U^0\}\right]+\max\{U_i^{bot},U^0\}\equiv\cdots \\ & p_{admit}^{agent}\cdot\left[v_i^{agent}-L^{agent}-\max\{U_i^{bot},U^0\}\right]+\max\{U_i^{bot},U^0\}, \end{aligned}$$

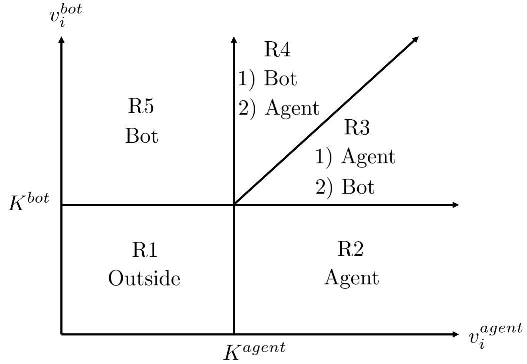
where p_{succ}^{agent} is the probability that the request will be resolved by the agent.

EC.4.2 Calculating Demand

To determine the proportion of traffic going to each channel, we compare U^{agent}, U^{bot} and U^0. We will be aided by Figure EC.2, in which the service values v_i^{agent} and v_i^{bot} are plotted on the x- and y-axes; to avoid trivialities, we have assumed that these utilities are non-negative. We partition the $\mathbb{R}^+$ quadrant into five mutually exclusive and collectively exhaustive regions that represent all realizations of (v_i^{agent}, v_i^{bot}). We begin by finding the regions in which $U_i^{bot} > U^0$. Notice that $U_i^{bot} > U^0$ implies that $v_i^{bot} - L^{bot} > U^0$ or

$$v_i^{bot} > L^{bot} + U^0 \equiv K^{bot}.$$

Figure EC.2 Channel Choice Regions



The inequality above is met when (v_i^{agent}, v_i^{bot}) falls in regions R3, R4 or R5. It is easy to see that customers whose (v_i^{agent}, v_i^{bot}) falls in these three regions prefer the bot channel to the outside channel. Analogous to the bot channel, the customer preference for the agent channel over the outside option is independent of the parameters of the bot channel. However, in this case, it requires a little effort to discern. Let us first consider the case when $\max\{U_i^{bot},U^0\} = U^0$. Then,

$$U_i^{agent} = p_{admit}^{agent}\cdot\left[v_i^{agent}-L^{agent}-\max\{U_i^{bot},U^0\}\right]+U^0 = p_{admit}^{agent}\cdot\left[v_i^{agent}-L^{agent}-U^0\right]+U^0.$$

In such instances, $U_i^{agent} > U^0$ implies that

$$U_i^{agent} = p_{admit}^{agent} \cdot \left[v_i^{agent} - L^{agent} - U^0\right] + U^0 > U^0,$$

so that $v_i^{agent} - L^{agent} - U^0 > 0$ or $v_i^{agent} > L^{agent} + U^0 \equiv K^{agent}$, which is met when (v_i^{agent}, v_i^{bot}) falls in regions R2, R3 or R4. More specifically, in R5 the bot channel is preferred to the agent channel and the outside option, whereas in R2 the agent channel is preferred to the outside option, which is preferred to the bot channel; hence, a customer in R2 will attempt to join the agent channel and then, if not admitted, will seek the outside option. We can also conclude that it is only when (v_i^{agent}, v_i^{bot}) falls in R1 that the outside option is preferred by customers. Now it just remains to determine the choice made by customers whose (v_i^{agent}, v_i^{bot}) falls in R3 or R4, two regions in which the outside option is dominated. One consequence is that $\max\{U_i^{bot}, U^0\} = U_i^{bot}$ Then,

$$U_i^{agent} = p_{admit}^{agent} \cdot \left[v_i^{agent} - L^{agent} - \max\{U_i^{bot}, U^0\}\right] + U^0 = p_{admit}^{agent} \cdot \left[v_i^{agent} - L^{agent} - U_i^{bot}\right] + U_i^{bot}.$$

In such instances, $U_i^{agent} > U_i^{bot}$ implies that

$$U_i^{agent} = p_{admit}^{agent} \cdot \left[v_i^{agent} - L^{agent} - U_i^{bot}\right] + U_i^{bot} > U_i^{bot},$$

so that $v_i^{agent} - L^{agent} > U_i^{bot} = v_i^{bot} - L^{bot}$. Or,

$$v_i^{bot} - L^{bot} - U^0 = v_i^{bot} - K^{bot} > v_i^{agent} - L^{agent} - U^0 \equiv v_i^{agent} - K^{agent}.$$

Or,

$$v_i^{bot} > v_i^{agent} - L^{agent} - U^0 \equiv v_i^{agent} + K^{bot} - K^{agent}.$$

Since the half-line $v_i^{bot} = v_i^{agent} + K^{bot} - K^{agent}$ is precisely the equation of the half-line that divides R3 from R4, we can conclude that the bot channel is preferred over the agent channel in R4 and the opposite holds in R3. In the case of R3, the customer initially tries to join the agent channel and, if not admitted, joins the bot channel.

We are now able to fully characterize the proportion of customers in each channel. Let $f(v_i^{agent}, v_i^{bot})$ be the joint probability density function of (v_i^{agent}, v_i^{bot}) and let $p(\cdot)$ denote the probability that (v_i^{agent}, v_i^{bot}) falls in a given region (R1 through R5). Then:

PROPOSITION 1 *When $f(v_i^{agent}, v_i^{bot})$ is the joint pdf of (v_i^{agent}, v_i^{bot}), then*

i. *The probability that a random customer first chooses the outside channel is p(R1),*

ii. *The probability that a random customer first chooses the live-agent channel is p(R2) + p(R3),*

iii. *The probability that a random customer first chooses the chatbot channel is p(R4) + p(R5).*

Proposition 1 is quite general since it applies to an arbitrary distribution of (v_i^{agent}, v_i^{bot}). Importantly, since the constant K^{bot} depends only on p_{succ}^{bot} and K^{agent} depends on q^{agent} under a given resolution policy D^{agent}, we can readily express $p(\mathrm{R1})$ to $p(\mathrm{R5})$ as functions of these primitives. Thus, using the five regions as building blocks, customer demand, which we denote by λ^{bot} and λ^{agent}, may be parameterized on these same primitives $(p_{succ}^{bot}, D^{agent}, q^{agent})$. Since the performance measures in each channel are also implicit functions of these same primitives, this will be shown in EC.5 to be sufficient to determine demand such that customers arrive at rates anticipated by the firm and the service desk delivers service at the performance measures promised by the firm to its customers. Thus, the firm's service desk design problem only requires for each architecture and admissible resolution policy D^{agent}, finding the optimal bot success probability p_{succ}^{bot} and then choosing that combination which returns the highest expected profit for the firm.

EC.5 Equilibrium Demand

We provide the general method for how to find the equilibrium arrival rates (q^{agent}, q^{bot}) to each channel and the calculations for when service valuations are drawn from an iid uniform distribution, which we use in the numerical illustration in §5.5. Without loss of generality, we assume that the service valuations v_i^{agent} and v_i^{bot} are uniformly distributed from 0 to $\bar{v}_{agent}$ and from 0 to $\bar{v}_{bot}$. As discussed in EC.4, we can parameterize λ^{bot} and K^{bot} on p_{succ}^{bot} as $\lambda^{bot}(p_{succ}^{bot})$ and $K^{bot}(p_{succ}^{bot})$. Likewise, we can parameterize λ^{agent} and K^{agent} on D^{agent} and q^{agent} as $\lambda^{agent}(D^{agent}, q^{agent})$ and $K^{agent}(D^{agent}, q^{agent})$. Also, we find it convenient to denote $\bar{v}_{bot} - K^{bot}(p_{succ}^{bot})$ by $\Delta^{bot}(p_{succ}^{bot})$ and $\bar{v}_{agent} - K^{agent}(D^{agent}, q^{agent})$ by $\Delta^{agent}(D^{agent}, q^{agent})$. We solve for the equilibrium arrival rates for each architecture:

EC.5.1 Bot-Only

When only the chatbot channel is offered, it naturally follows that $q^{agent} = \lambda^{agent} \equiv 0$. Then, for a given p_{succ}^{bot},

$$q^{bot} = \lambda^{bot}(p_{succ}^{bot}) = p(\mathrm{R3}) + p(\mathrm{R4}) + p(\mathrm{R5}),$$

and a simple computation solves for the equilibrium solution.

By inspection of Figure EC.2, under the uniform distribution, this is given by

$$q^{bot} = \frac{\Delta^{bot}(p_{succ}^{bot})}{\bar{v}_{bot}}.$$

Since q^{bot} is not an input of $\Delta^{bot}(p_{succ}^{bot})$, a simple computation of the above at p_{succ}^{bot} yields the equilibrium solution.

EC.5.2 Agent-Only

When only the agent channel is offered, it naturally follows that $q^{bot} = \lambda^{bot} \equiv 0$ and the bot success probability p^{bot}_{succ} plays no role. Then, for a given D^{agent} and q^{agent},

$$q^{agent} = \lambda^{agent}(D^{agent}, q^{agent}) = p(\mathrm{R2}) + p(\mathrm{R3}) + p(\mathrm{R4}),$$

which is analogous to the bot-only case, but with $p(\mathrm{R5})$ swapped with $P(\mathrm{R2})$. Since the sum of probabilities here depends on performance measures of the agent channel, each of which is a functions of q^{agent}, some computational effort may be needed to solve for the equilibrium solution.

For the uniform case, by inspection of Figure EC.2, this is given by

$$q^{agent} = \frac{\Delta^{agent}(D^{agent}, q^{agent})}{\bar{v}_{agent}}.$$

In this case, since q^{agent} is an input of $\Delta^{agent}(D^{agent}, q^{agent})$ through the performance measures, then q^{agent} must be solved for by finding the value of q^{agent} that sets the left and right sides equal. Because each of these performance measures can be expressed as a ratio of polynomial functions of q^{agent}, the equilibrium q^{agent} can be readily solved for numerically.

EC.5.3 Multichannel

When both channels are offered, we must proceed in two steps. First, since demand now depends on the performance measures of both channels, λ^{agent} now also depends on p^{bot}_{succ} and can be parameterized as $\lambda^{agent}(D^{agent}, p^{bot}_{succ}, q^{agent})$. More interestingly, the two channels now compete for customers that fall in R3 and R4. From Proposition 2 we can see that, in addition to customers in R2, the agent channel will directly win those in R3 so that

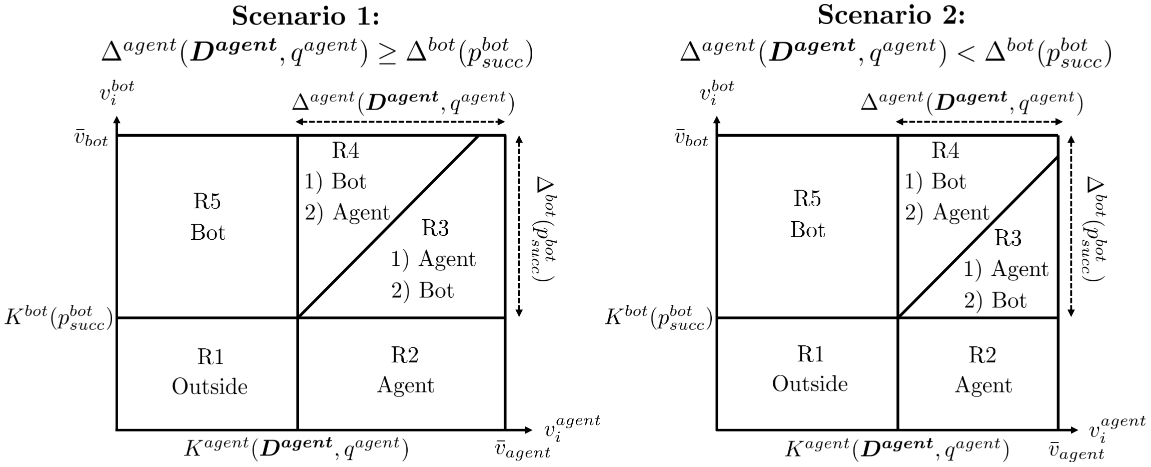
$$q^{agent} = \lambda^{agent}(D^{agent}, p^{bot}_{succ}, q^{agent}) = p(\mathrm{R2}) + p(\mathrm{R3}).$$

However, determining the demand for the bot requires an additional consideration. In addition to the customers in R4 and R5 that can be found from Proposition 2, the bot channel gets the portion $(1 - p^{agent}_{admit})$ of demand in R3 that was not admitted for service due to congestion at the agent channel. This probability depends on the resolution policy D^{agent} and the equilibrium agent demand q^{agent} and can therefore be parameterized as $p^{agent}_{admit}(D^{agent}, q^{agent})$. Thus, bot demand can be parameterized also on $p^{agent}_{admit}(D^{agent}, q^{agent})$ as $\lambda(p^{bot}_{succ}, p^{agent}_{admit}(D^{agent}, q^{agent}))$, and the equilibrium bot demand can be computed as

$$q^{bot} = \lambda^{bot}(p^{bot}_{succ}, p^{agent}_{admit}(D^{agent}, q^{agent})) = (1 - p^{agent}_{admit}(D^{agent}, q^{agent})) \cdot p(\mathrm{R3}) + p(\mathrm{R4}) + p(\mathrm{R5}).$$

Under the multichannel architecture, the shapes of R3 and R4 depend on whether the half-line dividing the two regions intersects $\bar{v}_{bot}$ on the vertical axis or $\bar{v}_{agent}$ on the horizontal axis. As we illustrate in Figure

Figure EC.3 Channel Choice Regions Uniform



Scenario 1: $\Delta^{agent}(\boldsymbol{D^{agent}}, q^{agent}) \geq \Delta^{bot}(p^{bot}_{succ})$

Scenario 2: $\Delta^{agent}(\boldsymbol{D^{agent}}, q^{agent}) < \Delta^{bot}(p^{bot}_{succ})$

v_i^{bot}
$\Delta^{agent}(\boldsymbol{D^{agent}}, q^{agent})$
$\bar{v}_{bot}$
R4
1) Bot
2) Agent
R5
Bot
R3
1) Agent
2) Bot
$\Delta^{bot}(p^{bot}_{succ})$
$K^{bot}(p^{bot}_{succ})$
R1
Outside
R2
Agent
v_i^{agent}
$K^{agent}(\boldsymbol{D^{agent}}, q^{agent})$
$\bar{v}_{agent}$

EC.3, which of these two scenarios holds depends on the value of $\Delta^{bot}(p^{bot}_{succ})$ relative to $\Delta^{agent}(D^{agent}, q^{agent})$. We derive the arrival rates (q^{agent}, q^{bot}) under each scenario.

Scenario 1 ($\Delta^{agent}(D^{agent}, q^{agent}) \geq \Delta^{bot}(p^{bot}_{succ})$): We have

$$q^{agent} = \lambda^{agent}(D^{agent}, p^{bot}_{succ}, q^{agent}) = p(\text{R2}) + p(\text{R3}).$$

By inspection of Scenario 1 in Figure EC.3, this is given by

$$q^{agent} = \frac{k^{bot}(p^{bot}_{succ}) \cdot \Delta^{agent}(D^{agent}, q^{agent}) + \Delta^{bot}(p^{bot}_{succ}) \cdot (\Delta^{agent}(D^{agent}, q^{agent}) - \frac{\Delta^{bot}(p^{bot}_{succ})}{2})}{\bar{v}_{agent} \cdot \bar{v}_{bot}}.$$

As was the case under the agent-only architecture, q^{agent} is solved for by finding the value of q^{agent} that sets the left and right sides equal.

Having solved for q^{agent}, the admission probability $(p^{agent}_{admit}(D^{agent}, q^{agent})$ under policy D^{agent} can be calculated via substitution. This admission probability is used in calculating q^{bot} since the bot channel gets the portion of demand $(1 - p^{agent}_{admit}(D^{agent}, q^{agent}))$ in R3 that was not admitted for service at the agent channel. In particular, we have

$$q^{bot} = \lambda^{bot}(p^{bot}_{succ}, p^{agent}_{admit}(D^{agent}, q^{agent})) = (1 - p^{agent}_{admit}(D^{agent}, q^{agent})) \cdot p(\text{R3}) + p(\text{R4}) + p(\text{R5}).$$

By inspection of Scenario 1 in Figure EC.3, this is given by

$$q^{bot} = \frac{(1 - p^{agent}_{admit}(D^{agent}, q^{agent})) \cdot \Delta^{bot}(p^{bot}_{succ}) \cdot (\Delta^{agent}(D^{agent}, q^{agent}) - \frac{\Delta^{bot}(p^{bot}_{succ})}{2})}{\bar{v}_{agent} \cdot \bar{v}_{bot}} + \cdots$$

$$\frac{\Delta^{bot}(p^{bot}_{succ}) \cdot (K^{agent}(D^{agent}, q^{agent}) + \frac{\Delta^{bot}(p^{bot}_{succ})}{2})}{\bar{v}_{agent} \cdot \bar{v}_{bot}},$$

where the first term is the spillover from the customers not admitted to the live-agent channel $((1 - p^{agent}_{admit}(D^{agent}, q^{agent})) \cdot p(\text{R3}))$ and the second term is the direct arrivals to the bot channel $(p(\text{R4}) + p(\text{R5}))$.

Since q^{agent} has already been solved for, then simply computing the above at p^{bot}_{succ} yields the equilibrium solution.

Scenario 2 ($\Delta^{agent}(D^{agent}, q^{agent}) < \Delta^{bot}(p^{bot}_{succ})$): By inspection of Scenario 2 in Figure EC.3, q^{agent} is given by

$$q^{agent} = \frac{\Delta^{agent}(D^{agent}, q^{agent}) \cdot (K^{bot}(p^{bot}_{succ}) + \frac{\Delta^{agent}(D^{agent}, q^{agent})}{2})}{\bar{v}_{agent} \cdot \bar{v}_{bot}},$$

and q^{bot} is given by

$$q^{bot} = \frac{(1 - p^{agent}_{admit}(D^{agent}, q^{agent})) \cdot \frac{\Delta^{agent}(D^{agent}, q^{agent})^2}{2}}{\bar{v}_{agent} \cdot \bar{v}_{bot}} + \cdots$$

$$\frac{K^{agent}(D^{agent}, q^{agent}) \cdot \Delta^{bot}(p^{bot}_{succ}) + \Delta^{agent}(D^{agent}, q^{agent}) \cdot (\Delta^{bot}(p^{bot}_{succ}) - \frac{\Delta^{agent}(D^{agent}, q^{agent})}{2})}{\bar{v}_{agent} \cdot \bar{v}_{bot}}.$$

Finally, we remark that the boundary condition that divides Scenario 1 from Scenario 2 ($\Delta^{agent}(D^{agent}, q^{agent}) \geq \Delta^{bot}(p^{bot}_{succ})$) itself depends on q^{agent} through $\Delta^{agent}(D^{agent}, q^{agent})$. Hence, it is not known in advance which condition holds. This can be resolved by first assuming that Scenario 1 holds, solving for q^{agent} under Scenario 1, substituting q^{agent} into $\Delta^{agent}(D^{agent}, q^{agent})$, and then testing whether the scenario condition holds. If so, then continue on to calculate q^{bot} under Scenario 1. If not, perform the procedure under Scenario 2.

EC.6 Channel Architecture Problem

Now that we have shown how to calculate equilibrium demand, we turn our attention to the firm's overarching customer service design problem.

EC.6.1 Profit Measure

Our analysis so far has focused on a single live agent (chatbot) handling customer requests, and at most one customer arriving in each time period. However, most service desks employ multiple agents, and may have more than one customer requesting service in any given time period. A pragmatic approach is to think of there being multiple service desks, one per agent, each of which is randomly assigned customers by a routing protocol, so that in any period t no more than one customer presents for service. This can be assured by making the period interval sufficiently small relative to the market size which would ultimately imply that a Poisson process generated arrivals, a standard approach in queuing models of customer service. Assuming a homogeneous agent pool and scaling up processing times yield similar reconciliation. To evaluate different channel architectures, it is therefore sufficient to focus on the firm's profit margin per unit time.

EC.6.2 Live-Agent Channel Contribution

The firm's channel profit in the live-agent channel consists of revenue less agent wages and expert compensation. Specifically, the firm receives a revenue of r for each customer served. The revenue r can represent a direct payment made upon completion of a task-oriented request, for example, a student wanting to register and pay for a course or a program; alternatively, r can represent the loss of "goodwill" that the firm would incur if the customer request remains unresolved. The firm also has the following expenditures: live-agent wages, c^{wage}, expert fees of c^w (c^c) for each warm transferred (cold transferred) customer.

Recall that by inducing stationarity through our terminal conditions, we can directly calculate the performance measures that characterize the performance of the live-agent channel under a given resolution policy. Then parameterizing on D^{agent} and $\lambda^{agent}(D^{agent}, p^{bot}_{succ})$, these measures include (among others) the customer admission probability, $p^{agent}_{admit}(D^{agent}, \lambda^{agent}(D^{agent}, p^{bot}_{succ}))$, and the respective conditional probabilities, $\rho^w(D^{agent}, \lambda^{agent}(D^{agent}, p^{bot}_{succ}))$ and $\rho^c(D^{agent}, \lambda^{agent}(D^{agent}, p^{bot}_{succ}))$, that the customer will be warm transferred or cold transferred. These three performance measures under each admissible D^{agent} for the 2-solution ($S = 2$) problem are available upon request from the authors. For exposition, we simplify notation by suppressing all arguments and write the profit contribution of the live-agent channel as

$$\pi^{agent} = \lambda^{agent} \cdot p^{agent}_{admit} \cdot (r - \rho^w \cdot c^w - \rho^c \cdot c^c) - c^{wage}. \tag{EC.8}$$

Per period, λ^{agent} customers arrive to the channel and p^{agent}_{admit} of the arrivals are admitted. The firm then receives r for each admitted customer, but pays an expert c^w with conditional probability ρ^w and c^c with conditional probability ρ^c to complete resolution. Finally, the agent receives an expected wage of c^{wage}.

EC.6.3 Chatbot Channel Contribution

Since the chatbot is scalable, there is no waiting and all customers are admitted to service. Therefore, the chatbot channel contribution can be written as

$$\pi^{bot} = \lambda^{bot} \cdot (r - (1 - p^{bot}_{succ}) \cdot c^c) - c^{bot}(p^{bot}_{succ}), \tag{EC.9}$$

where $c^{bot}(p^{bot}_{succ})$ is the amortized per period chatbot development cost as a function of p^{bot}_{succ}.

EC.7 Numerical Illustration Parameters

We limit the number of potential solutions to 2 ($S = 2$). The firm receives a fee of 20 per processed request ($r = 20$). The first attempt takes the agent two periods ($\tau_1 = 2$) and resolves the request with probability 0.7 ($\rho_1 = 0.7$). The second takes five periods ($\tau_2 = 5$) and resolves with conditional probability 1 ($\rho_2 = 1$). If the first attempt fails, an expert is available with probability 0.8 ($a = 0.8$). If the request is cold transferred, the expert takes two periods to resolve it upon receipt. If the request is warm transferred, the handoff between the agent and the expert takes 1 period ($\tau_w = 1$) and the expert takes one period to resolve it thereafter. The firm pays the expert 6 per period to process requests, resulting in a cost to the firm of 12 for each cold transfer and warm transfer ($c^c = c^w = 12$). The agent's wage is 3 per period ($c^{wage} = 3$).

For purposes of this numerical illustration, we take the chatbot processing times as exogenous. While the chatbot cannot attempt the second solution, it can be trained to be up to as proficient as the live agent in solving the first attempt ($0 < p^{bot}_{succ} \leq \rho_1$), which takes two periods ($\tau^{bot}_{succ} = \tau^{bot}_{fail} = 2$). The amortized cost of training the bot is convex in p^{bot}_{succ} and is given by $c^{bot}(p^{bot}_{succ}) = 2/(1 - p^{bot}_{succ})$.

With respect to the customer utility parameters, customers incur a disutility of 1 for each period spent with the agent or bot, and a non-admission cost of 1 if not admitted to the live-agent channel ($c^{not}_{admit} = 1$). To reflect the lower perceived service quality of being transferred, the per period disutility increases to 1.25 for a warm transfer and to 2 for a cold transfer. Moreover, we assume that customer service valuations v^{agent}_i and v^{bot}_i are uniformly distributed from 0 to $\bar{v}^{agent}$ and from 0 to $\bar{v}^{bot}$, and set $\bar{v}^{agent}$ to 40. Finally, to test different levels of technology acceptance, we run three scenarios: 1) $\bar{v}^{bot} = 30 < 40 = \bar{v}^{agent}$ (technology averse), 2) $\bar{v}^{bot} = 40 = \bar{v}^{agent}$ (technology neutral), 3) $\bar{v}^{bot} = 50 > 40 = \bar{v}^{agent}$ (technology enthusiast).