An Equilibrium Solver for A Dynamic Queueing Game*

Denise Cerna† Chiwei Yan‡ Hongyao Ma§

August 24, 2026

Abstract

Consider the dispatch of ridesharing trips to drivers lining up at airports, or the allocation of deceased donor organs to patients in transplant wait lists. In such dynamic queueing games, heterogeneity in short-lived items combined with agents' discretion to decline induce substantial cherrypicking, resulting in high rates of unfulfilled trips and discarded organs. Existing research typically focuses on easy-to-analyze dispatch policies, sometimes making fluid assumptions for tractability. In this work, we introduce a solver for general dispatch policies with closed-form updates, which computes agents' equilibrium acceptance, entry, and reneging strategies as functions of queue length and queue position. We prove that a Markov perfect equilibrium exists for any dispatch policy that never assigns the same item to an agent more than once. Moreover, our solver converges in finitely many iterations under policies where items are offered monotonically down the queue and dispatch probabilities depend only on queue position, not queue length. We further show that our solver naturally extends to parallel dispatching, where items may be offered to multiple agents simultaneously, and costly inspection, where agents pay to learn an item's type before deciding whether to accept it. Through extensive simulations, we show that the solver recovers known equilibria for theoretically analyzable policies, offers insights into more complex policies better suited for practice, evaluates the robustness of different policies under model misspecification, and elucidates the competitive dynamics between platforms in the presence of driver multihoming and rider dual-apping.

1 Introduction

Matching items to agents in a queue is a fundamental challenge at the intersection of operations and economics. Examples include ridesharing platforms assigning rider trips to drivers waiting in virtual queues at an airport, and allocating deceased donor organs to patients on the national transplant waiting list. These problems share common features: (1) agents are *strategic* and have the option to *decline* undesirable items offered to them; and (2) items are *short-lived* and *heterogeneous*. As a consequence, agents' cherry-picking can result in significant inefficiencies including unfulfilled

*The authors would like to thank Etienne Ackermann, Philipp Afèche, Nick Arnosti, Itai Ashlagi, Francisco Castro, Federico Echenique, Nikhil Garg, Sergey Gitlin, Chris Golinski, Izzy Grosof, Jake Marcinek, Hamid Nazerzadeh, Peng Shi, Ran Snitkovsky, Sylvia Zheng, and participants at conferences and seminars for their valuable comments and discussions. Cerna and Yan acknowledge support from the National Science Foundation under Award Number 2517861. Yan also gratefully acknowledges Eurandom's support for attending the *Product-Form Probability Distributions* workshop and the valuable discussions with its participants.

†Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720, USA. Email: denise_cerna@berkeley.edu

‡Department of Industrial Engineering and Operations Research, University of California, Berkeley, CA 94720, USA. Email: chiwei@berkeley.edu.

§Decision, Risk, and Operations Division, Graduate School of Business, Columbia University, New York, NY, 10027, USA. Email: hongyao.ma@columbia.edu.

rides and wasted organs. For example, ridesharing passengers have very limited patience, often willing to wait only a few minutes for a match. Furthermore, trip destinations introduce significant heterogeneity in value—trips from an airport can range from $10 to over $100. As drivers can freely accept or decline trips, after ridesharing platforms like Uber allowed drivers to view trip destinations and earnings,[1] the trip completion rate plummeted to 60% at major airports like SFO and LAX [San Francisco Chronicle, 2021], as drivers increasingly cherry-picked high-earning trips, resulting in unfulfilled trips and long wait times. As another example, the viability of deceased donor organs, measured by cold ischemia time—the period an organ remains viable in cold storage—ranges from 4–6 hours for hearts and lungs to 24–36 hours for kidneys [Medscape, 2023]. Organ quality also varies significantly, influenced by factors such as donor age, medical history, and cause of death [Mohan et al., 2018]. Patients are free to accept or decline organ offers, and the rate of discarded deceased donor kidneys in the U.S. has averaged around one-fifth and has risen to 25–30% recently [Wood et al., 2025], much of which is due to patients strategically declining organs in search of higher quality kidneys [Agarwal et al., 2021, Ashlagi et al., 2025].

Previous work on this problem has developed analyzable dispatch policies to improve efficiency. In the context of deceased donor kidneys, Su and Zenios [2004] notably showed that under a FIFO (First-In-First-Out) dispatch policy, patients' selective acceptance behavior can result in significant organ wastage. To address this, they proposed a hybrid policy that combines FIFO and LIFO (Last-In-First-Out) dispatching to optimize organ utilization. More recently, Shi and Yin [2022] introduced a policy that prioritizes patients newly added to the waitlist. In practice, many organ allocation agencies rely heavily on simulation models to evaluate alternative dispatch rules. However, these simulations typically neglect the strategic responses of patients who adapt their acceptance decisions in reaction to changes in dispatch policies,[2] likely because there is not yet a simple, general methodology for computing equilibrium acceptance behavior under arbitrary dispatch policies.

Similarly, in the context of ridesharing, Castro et al. [2021] show that FIFO dispatching at airports can lead to market inefficiencies. To address this, they propose a class of randomized FIFO dispatch policies that achieve second-best outcomes by partitioning the queue into bins and dispatching trips in a randomized, bin-by-bin manner. Intuitively, this policy improves efficiency because (i) the option to skip the rest of the queue incentivizes drivers toward the end of the queue to accept lower-paying trips, accelerating dispatch and reducing rider cancellations; and (ii) randomizing over multiple drivers substantially increases the cost of cherry-picking. For analytical tractability, however, their model assumes a fluid approximation in which queue lengths remain constant at equilibrium. The dispatch policy is constructed statically and does not account for real-time fluctuations in queue length. Moreover, their analysis requires that bins are constructed in a specific way aligned with the equilibrium outcome, and they do not characterize equilibrium behavior for arbitrarily defined bins. These assumptions limit its direct applicability to dynamic, real-world environments. Therefore, it is essential to explore policies that can adapt to the stochastic fluctuations of queue length in practical settings. For instance, consider a simple dynamic variation of the static randomized FIFO policy discussed previously. In this adaptive approach, for example,

[1] https://www.uber.com/newsroom/only-on-uber/, accessed February 9, 2025.

[2] For example, the latest simulation software OASIM, developed by the Scientific Registry of Transplant Recipients (SRTR) to support the Organ Procurement and Transplantation Network (OPTN) in the U.S., explicitly states in its manual: "For future simulation requests, the same placement mechanism will be used for all proposed policies. This introduces an important assumption: the accept/decline behavior of candidates is invariant across policies (i.e., the probability of acceptance under different allocation policies is reasonably approximated by the same logistic regression model[s]). This assumption is likely not true in practice." More details are available at https://www.srtr.org/media/1838/ki2024_01_request_report_website.pdf, accessed June 14, 2025.

whenever a new trip request arrives, if the queue is currently 240 drivers long, simply divide it into 12 bins of 20 drivers each,[3] and the request is randomly assigned to a driver down the queue bin by bin. Compared to its static counterpart, this adaptive policy is more practical as it eliminates the need for estimating hard-to-determine inputs such as arrival rates and payoffs of trips. Moreover, it naturally adjusts to the dynamic nature of queue length, making it better suited for real-world applications. However, there is currently no understanding in the literature of how to compute the equilibrium under such a policy.

Our work is thus directly motivated by the strong need to compute an equilibrium for a general dispatch policy in a stochastic environment, and aims to fill this important gap. This task is challenging because agents' strategies depend not only on their position in the queue but also on the overall queue length. These strategies, in turn, influence the stochastic evolution of the queue as new agents arrive and existing ones either accept trips and exit or renege. This dynamic feedback loop—where queue length and agent strategies continuously shape one another—requires specialized analytical techniques, which form the core focus of this paper. We now provide a brief overview of our framework and summarize our main contributions.

1.1 Our Results

In this work, we develop an analytical solver with *closed-form updates* that computes agents' equilibrium strategies under general dispatch policies.

Model. We study a stochastic and dynamic queueing game in which items and agents arrive according to stationary Poisson processes. Agents are homogeneous, while items vary by type. Upon arrival, agents strategically decide whether to join at the tail of the queue based on its current length, and later, whether to accept or reject an item when it is offered to them to maximize their expected utility—defined as the expected value of the accepted item minus the expected waiting cost. Additionally, agents in the queue can renege both *exogenously* (due to unexpected departure) and *endogenously* (due to negative continuation payoff). Items have finite patience, meaning that they can be re-dispatched to other agents in the queue if declined, but will expire eventually if declined too many times. Agents are allowed to decline items freely without penalties and have full information regarding the state of the queue and the types of items being offered, among others. Our framework encompasses a general class of dispatch policies that map each item's type and dispatch history to a probability distribution over potential queue positions.

Main Results. We begin by formalizing an equilibrium concept of our game based on Markov perfect equilibrium and characterizing the equilibrium conditions through fixed-point equations defined by agents' continuation payoffs. We further show that, for any given dispatch policy and strategy profile, these payoffs can be computed in closed form. We then prove that a Markov perfect equilibrium (MPE) strategy always exists under any dispatch policy that does not dispatch any item to the same agent more than once. Leveraging this fixed-point equilibrium characterization, we develop an algorithm that computes equilibria for arbitrary dispatch policies using entirely closed-form updates. For a broad class of dispatch policies—specifically, those with queue-length-independent dispatch probabilities and items offered monotonically down the queue—we prove that our solver converges to a Markov perfect equilibrium in a finite number of iterations from any initial strategy.

[3] If each declined dispatch takes 10 seconds and riders begin canceling their requests after 2 minutes, that amounts to roughly 12 dispatch attempts before a typical cancellation.

Through extensive simulations using ridesharing data from Chicago O'Hare International Airport, we demonstrate that our solver exhibits robust convergence, even for policies lacking theoretical guarantees. It not only recovers known results for analytically tractable policies but also uncovers new insights beyond prior analyses. Notably, the solver successfully converges for the dynamic implementation of the randomized FIFO policy discussed earlier, confirming its superior performance. As previously noted, the dynamic randomized FIFO mechanism does not require knowledge of model parameters such as arrival rates or payoffs of trips. Leveraging our solver's capabilities, we further evaluate scenarios with mis-specified trip and driver arrival rates—a common challenge in practice due to imperfect forecasts. Our results show that the dynamic randomized FIFO policy consistently delivers robust performance under these conditions, outperforming alternative policies. This proof-of-concept demonstrates that our solver is a valuable tool for market designers, enabling the exploration and optimization of more complex policies tailored to real-world applications.

Extension to Parallel Dispatching. We conclude by showing that our framework naturally extends to settings where an item may be offered to *multiple* agents *simultaneously*. In cases where multiple agents accept the same item, ties are broken according to a predefined rule. This framework enables equilibrium computation across various practical settings, including recent ridesharing dispatch products such as Uber's Trip Radar[4] and Lyft's Rider Finder, which broadcast ride requests to multiple drivers, as well as the UK's Kidney Fast-Track Scheme [Callaghan et al., 2017], which accelerates organ allocation by offering organs at risk of nonuse to multiple patients. Perhaps surprisingly, it also supports the analysis of competing ridesharing platforms at the same airport that share a portion of rider demand (i.e., rider dual-apping) and the full driver supply (i.e., driver multihoming), as examined in detail in Section 6.1.1.

1.2 Related Work

Our work is mainly related to two streams of literature: queueing games and equilibrium computation.

Queueing Games. The study of agents' strategic behavior in queueing systems was pioneered by Naor [1969], who analyzed a single-server system with an observable queue. He first demonstrated the negative externalities from waiting: when agents make self-interested decisions on whether to join a FIFO queue, in equilibrium more agents line up in the queue in comparison to the socially optimal outcome. See also the related early work by Yechiali [1971], as well as David and Yechiali [1985], who considered a closely related setting in which agents are offered items with random rewards in order of their position in the queue. Since then, a large body of subsequent work has studied how to align incentives and improve system efficiency in various settings (see Hassin and Haviv 2003 and Hassin 2016 for comprehensive reviews). Without the use of monetary transfers, a significant amount of work explores the idea of using last-in-first-out (LIFO) inspired queueing discipline to achieve the socially optimal outcome since when the agent who has waited the longest in the queue decides whether to leave, she imposes no externality on any current or future agents [Hassin, 1985, Su and Zenios, 2004, Haviv and Oz, 2016, Scarsini and Shmaya, 2024]. Nikzad and Strack [2024] propose random allocation to balance equity and efficiency. Castro et al. [2021] analyze a class of randomized FIFO mechanisms. These mechanisms organize the queue into carefully constructed bins, ensuring that agents remain incentive-compatible when accepting specific items

[4] https://www.uber.com/blog/trip-radar/

assigned to certain bins. Through this design, they demonstrate that the mechanism attains the second-best outcome. However, their analysis is limited to a fluid model, where the equilibrium queue length remains constant. Furthermore, they do not provide equilibrium analysis for randomized FIFO policies with arbitrarily constructed bins. Existing research also compares FIFO with other randomized allocation rules across various settings, highlighting their efficiency. However, much of this analysis is conducted under restrictive assumptions about queueing dynamics—for instance, assuming that queues are either perpetually overloaded with a constant length [Bloch and Cantala, 2017, Schummer, 2021] or maintain a minimum length [Leshno, 2022]. In contrast, our work imposes minimal assumptions on both queueing dynamics and dispatch policies. We compute equilibrium outcomes for a broad class of dispatch policies, providing a more general and flexible framework. This facilitates the exploration and understanding of practically relevant dispatch policies that are challenging to analyze theoretically. Finally, Che and Tercieux [2021] examines the role of information design in enhancing efficiency and establishing the optimality of FIFO in such settings. In contrast, our work focuses on a full information setting, where the state of the queue and the types of items offered are fully disclosed to all agents.

Equilibrium Computation. The study of equilibrium existence and computation traces back to the seminal work of Nash [1951], and often relies on fixed-point theorems (e.g., Kakutani, or related results by Debreu, Glicksberg, and Fan), structural properties such as supermodularity (e.g., Milgrom and Roberts 1990; Topkis 1998; Yao 1995), or reformulations as optimization problems—such as linear programming for zero-sum games (e.g., Adler 2013; von Stengel 2024) and potential games [Monderer and Shapley, 1996]. Our dynamic queueing game is related to stochastic (Markov) games [Shapley, 1953], but it fundamentally differs in that the number of players is infinite and fluctuates dynamically as agents enter and exit the queue. This dynamic population renders standard equilibrium computation methods for Markov games—such as policy or value iteration—inapplicable without significant adaptation. Instead, we exploit the specific structure of our game and propose a fixed-point characterization of equilibrium, drawing inspiration from techniques in congestion games (e.g., Smith 1983). Our work advances the literature beyond existing results in congestion games: whereas prior approaches typically assume continuity via predefined latency functions, we derive a closed-form expression for continuation payoffs under arbitrary dispatch policies and strategy profiles. Finally, in the related literature of computing or approximating equilibria of dynamic queueing games, Ravner and Snitkovsky [2024] notably propose a stochastic approximation algorithm to compute a symmetric Nash equilibrium strategy with a finite action space, utilizing simulation. However, this approach is not tractable for our setting due to the prohibitively large number of state-dependent actions.

2 Preliminaries

We study a continuous time model where heterogeneous items are dispatched to agents waiting in a queue. Consider, for example, the dispatch of trips with different destinations to ridesharing drivers lining up at the airport, or the allocations of deceased donor organs to patients on the transplant waitlist. Let $[L] \triangleq \{1, 2, \ldots, L\}$ denote the set of $L \in \mathbb{Z}_{>0}$ item types. The arrivals of items of each type $\ell \in [L]$ follow a stationary Poisson process with rate $\mu_\ell > 0$, and the *payoff* of type ℓ items is $w_\ell \in \mathbb{R}$. Without loss of generality, we assume that the items are ordered such that $w_1 > w_2 > \ldots > w_\ell$. Items need to be dispatched to agents in the queue upon arrival, and may be dispatched again after being declined. We assume that all items share the same *patience level* $p \in \mathbb{Z}_{>0}$, meaning that an item expires and can no longer be dispatched after being declined by p

agents in the queue.[5]

Agents are homogeneous, and arrive according to a stationary Poisson process with rate $\lambda > 0$. Upon arrival, agents may decide whether to join at the tail of the queue based on the length of the queue,[6] and we assume that agents who choose to not join the queue get a payoff that is normalized to be zero.[7] After joining the queue, an agent incurs a *waiting cost* of $c_w \in \mathbb{R}$ (which is allowed to be negative) for each unit of time she spends waiting in the queue. Moreover, each agent in the queue reneges *exogenously* with a rate $\theta > 0$,[8] modeling the possibility that ridesharing drivers might need to leave the airport queue due to unexpected personal obligations, or patients may become too sick to receive a transplant.

Additionally, we also allow for *endogenous* reneging, i.e., an agent currently in the queue may choose to leave the queue based on the queue length and her queue position. This models, for example, a ridesharing driver leaving for the city without a rider when she expects poor outcomes from continuing to wait in the airport queue. Finally, when offered an item, agents can decide whether to accept the item based on its payoff, the agent's position in the queue, and the current queue length. Agents are strategic, and seek to optimize their expected *net payoff*, i.e., the payoff from the item they accept (if any) minus their waiting cost. We assume that agents do not have preferences over the items on the basis of anything other than the payoff.

We focus on complete information settings that are fully *transparent* and *flexible*, meaning that (i) the arrival rates, payoffs from items, and agents' cost of time are common knowledge among agents; (ii) at any point of time, all agents know the length of the queue and their positions in the queue; (iii) agents are not penalized for any actions they take, and have the option to decline any number of dispatches without losing their positions in the queue.Additionally, we assume that the length of the queue is upper bounded by an arbitrarily large number $\bar{Q} \in \mathbb{Z}_{>0}$. This models, for example, the finite capacity of airport waiting lots for ridesharing drivers, and with agent exogeneous reneging this is almost without loss.

2.1 Dynamic Dispatch Policies

Upon arrival, or after being declined by the agents, items are offered to agents in the queue based on a *dispatch policy*, denoted as π. If an agent accepts the item, they leave the queue and collect the corresponding payoff. If the item is rejected, it is dispatched again until (i) an agent accepts, (ii) the item expires after being declined p times, or (iii) the policy decides not to offer the item again to any agent. A dispatch policy π is specified by a set of probabilities for each queue position to receive the item, which can depend on the current length of the queue, the type of the item, and how many times the item had already been dispatched. Additionally, a policy can take into consideration the realized *dispatch history*, i.e., which positions in the queue were offered (and then rejected) the item in the earlier dispatches.

[5] The patience level represents, for example, the amount of the time ridesharing customers are willing to wait for a match before canceling their trip request, or the period of time organs can be preserved in cold storage. We assume homogeneous patience for simplicity of notation; our framework can be generalized to settings where patience levels are heterogeneous, as well as when items may expire with some probability after each decline.

[6] Most of our technical results continue to hold even when agents are not required to join the tail of the queue. The exception is that the closed-form expression for the continuation payoff in Lemma 1 is replaced by the solution to a system of linear equations. See Section 3.1 and Appendix D.2 for more details.

[7] If the outside option is nonzero, it can be normalized by subtracting the outside option value from the payoff of each item ℓ.

[8] We can include the case $\theta = 0$ as long as the continuation payoff is well defined, which is true under most reasonable dispatch policies and strategy profiles, but fails, for example, under a pathological strategy profile in which all agents reject every item, leading to a continuation payoff of negative infinity. By contrast, for any $\theta > 0$, the continuation payoff is guaranteed to be well defined under any dispatch policy and strategy profile.

Formally, with $[n] \triangleq \{1, 2, \ldots, n\}$ for any positive integer $n \in \mathbb{Z}_{>0}$, we use $Q \in [\bar{Q}]$ and $q \in [Q]$ to denote the *queue length* and a *queue position*. When the length of the queue is Q, positions $q = 1$ and $q = Q$ correspond to the *head* and the *tail* of the queue, respectively. Additionally, for each $k \in [p]$, let $h^{(k)} = (h_1^{(k)}, h_2^{(k)}, \ldots, h_k^{(k)}) \in [1, \ldots, \bar{Q}]^k$ represent the *dispatch history* of an item's first k dispatches, such that $h_{k'}^{(k)} \in [\bar{Q}]$ is the queue position to which the k'th dispatch was offered. A dispatch policy π is specified by the dispatch probabilities

$$\left(f_\ell^{(k)}\left(q, Q; h^{(k-1)}\right)\right)_{\ell \in [L],\ k \in [p],\ q \in [\bar{Q}],\ Q \in [\bar{Q}],\ h^{(k-1)} \in \mathcal{H}^{(k-1)},} \tag{1}$$

where $\mathcal{H}^{(k)} \triangleq [\bar{Q}]^k$ denotes the set of all possible dispatch histories after an item's first k dispatches.

For each item type $\ell \in [L]$, dispatch $k \in [p]$, queue length $Q \in [\bar{Q}]$ and history $h^{(k-1)} \in \mathcal{H}^{(k-1)}$ for the first $k-1$ dispatches, the probability for each queue position $q \in [\bar{Q}]$ to receive the item $f_\ell^{(k)}(q, Q; h^{(k-1)})$ satisfies $f_\ell^{(k)}(q, Q; h^{(k-1)}) \in [0, 1]$, and we must have $\sum_{q \in [\bar{Q}]} f_\ell^{(k)}(q, Q; h^{(k-1)}) \leq 1$. Note that it is possible for the item to not be dispatched again after the first $k-1$ dispatches, which happens with probability $1 - \sum_{q \in [\bar{Q}]} f_\ell^{(k)}(q, Q; h^{(k-1)})$. Also note that we allow items to be dispatched to queue positions where *no agent is present* (i.e., where $q > Q$), in which case the item is automatically declined for that dispatch. If the dispatch policy is *history-independent*, we omit $h^{(k-1)}$ and refer to the probability for position q to receive the item as $f_\ell^{(k)}(q, Q)$.

We now present two example dispatch policies, both of which are history-independent; accordingly, we omit the history from the notation.

Example 1 (Strict FIFO Dispatch)**.** Under the Strict FIFO dispatch policy, the k^{th} time an item is dispatched, it is offered to the agent at position k in the queue with probability 1. Formally, for all $\ell \in [L]$, $Q \in [\bar{Q}]$, $q \in [\bar{Q}]$ and each $k \in [p]$,

$$f_\ell^{(k)}(q, Q) = \begin{cases} 1, & \text{if } q = k, \\ 0, & \text{otherwise.} \end{cases}$$

Example 2 (Pure Random Dispatch)**.** Under the pure random dispatch policy, for each dispatch, every item is offered to all agents in the queue uniformly at random. Formally, for all $\ell \in [L]$, $Q \in [\bar{Q}]$, $q \in [\bar{Q}]$ and each $k \in [p]$,

$$f_\ell^{(k)}(q, Q) = \begin{cases} 1/Q, & \text{if } q \leq Q, \\ 0, & \text{otherwise.} \end{cases}$$

2.2 Agent Strategies

We use the tuple (q, Q) to describe the *state* of an agent at position $q \in [Q]$ in the queue, when the length of the queue is $Q \in [\bar{Q}]$. An agent's *strategy* $\sigma \triangleq (\alpha, \phi)$ consists of an *acceptance* strategy α and a *queue-joining* strategy ϕ:

- At each state (q, Q), $\alpha_\ell(q, Q) \in [0, 1]$ is the probability with which the agent accepts item $\ell \in [L]$ when it is offered;
- For each $Q \in [\bar{Q}]$, $\phi(Q-1) \in [0, 1]$ is the probability with which an agent joins the queue upon arrival when the queue length is $Q-1$ (in which case the agent takes position Q).

Note that $\phi(\bar{Q}) = 0$ always holds since the length of the queue is capped at $\bar{Q}$, and $\phi(0)$ represents the joining probability when the arriving agent sees the queue empty.

Endogenous Reneging. In many practical settings, it is possible that agents in the queue may choose to leave because the continuation payoff from remaining in the queue is low. Instead of introducing a separate strategy, we incorporate the possibility of such *endogenous reneging* via an additional type 0 item with payoff $w_0 = 0$ and arrival rate $\mu_0 = \rho\bar{Q}$ for some $\rho \geq 0$. The item is dispatched according to

$$f_0^{(k)}(q, Q) = \begin{cases} 1/\bar{Q}, & \text{if } k = 1, \\ 0, & \text{otherwise,} \end{cases}$$

meaning every agent in the queue receives type 0 items (i.e., considers the option to renege) at rate ρ. In this way, the acceptance strategy for the type 0 items determines whether to renege or to remain in the queue. When acting optimally, the agents' continuation payoff is lower bounded by $-c_w/\rho$, which can be made arbitrarily close to 0 via setting a very larger ρ.

In the rest of the paper, $\mathcal{L} \triangleq \{0, 1, 2, \ldots, L\}$ denotes the *augmented* set of items that includes the decision to renege, and $\mathcal{S}$ denotes the set of all possible strategies an agent may adopt.

3 Markov Perfect Equilibrium

In this section, we define the *Markov Perfect Equilibrium* (MPE) under the model introduced in the previous section. For any dispatch policy that does not offer any item to the same agent more than once, we provide a set of easy-to-check conditions that determine whether a strategy forms an MPE, and prove that there always exists a (potentially randomized) strategy that forms an MPE.

We start from equilibrium definition. Let $V(\sigma, \sigma_-, q, Q)$ denote the expected continuation payoff of an agent at position q in the queue, when (i) the queue length is Q, (ii) the agent adopts strategy σ, and (iii) every other agent, including those who have not yet arrived, adopts strategy σ_-.[9]

Definition 1. A strategy $\sigma^* = (\alpha^*, \phi^*)$ forms a symmetric *Markov perfect equilibrium* (MPE) if

$$V(\sigma^*, \sigma^*, q, Q) \geq V(\sigma, \sigma^*, q, Q), \qquad \forall \sigma \in \mathcal{S},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{2}$$

$$V(\sigma^*, \sigma^*, Q, Q) \geq 0, \qquad \forall Q \in [\bar{Q}] : \phi^*(Q-1) > 0, \tag{3}$$

$$V(\sigma^*, \sigma^*, Q, Q) \leq 0, \qquad \forall Q \in [\bar{Q}] : \phi^*(Q-1) < 1. \tag{4}$$

Condition (2) requires that given any queue length $Q \in [\bar{Q}]$, at any queue position $q \in [Q]$, the agent's continuation payoff is weakly higher than that under any possible strategy $\sigma \in \mathcal{S}$. Conditions (3) and (4) require that the agent joins the queue if and only if doing so leads to non-negative continuation payoffs. The optimal joining conditions are stated separately because $V(\sigma^*, \sigma^*, q, Q)$ represents the continuation payoff of agents who are *already in the queue*, therefore (2) only guarantees that the strategy is optimal in terms of the acceptance decisions (which also includes the decision to renege if endogenous reneging is allowed, as discussed in Section 2).

Given our focus on symmetric equilibria where all agents adopt the same strategy, when the strategy σ is clear from the context and adopted by all agents, we simplify the notation and use $V(q, Q)$ to denote the continuation payoff of an agent at position q when the queue length is Q.

It is non-trivial to evaluate the continuation payoff under general dispatch policies and arbitrary agent strategies. Our first result establishes that a *closed-form* solution for the continuation payoff V exists for *any* dispatch policy π, and *any* strategy σ adopted by all agents.

[9] Note that this does not provide notation for the continuation payoff of an agent, where the rest of the agents adopt multiple different strategies. This work focuses on symmetric equilibria, where all agents adopt the same strategy.

3.1 Evaluating the Continuation Payoff

To compute the continuation payoff $V = (V(q,Q))_{Q\in[\bar{Q}],q\in[Q]}$ given strategy $\sigma = (\alpha,\phi)$ adopted by all agents and dispatch policy π specified by dispatch probabilities f (see (1)), we start from defining a number of useful rate terms induced by σ and f.

First, we use $g_\ell^{(k)}(q,Q)$ to denote the probability for position $q\in[\bar{Q}]$ to receive an item of type $\ell\in\mathcal{L}$ during its k^{th} dispatch, when the queue length is $Q\in[\bar{Q}]$. For the first dispatch, we have

$$g_\ell^{(1)}(q,Q) = f_\ell^{(1)}(q,Q).$$

For any dispatch history $h^{(1)}\in\mathcal{H}^{(1)}$, the probability that the first dispatch is received and rejected by the agent at position $h_1^{(1)}$ is given by $f_\ell^{(1)}(h_1^{(1)},Q)(1-\alpha_\ell(h_1^{(1)},Q))$, hence

$$g_\ell^{(2)}(q,Q) = \sum_{h^{(1)}\in\mathcal{H}^{(1)}} f_\ell^{(2)}(q,Q;h^{(1)}) f_\ell^{(1)}(h_1^{(1)},Q)\left(1-\alpha_\ell(h_1^{(1)},Q)\right).$$

Generally, for any $k\geq 2$, $Q\in[\bar{Q}]$, $q\in[Q]$, and $\ell\in\mathcal{L}$, we have

$$g_\ell^{(k)}(q,Q) = \sum_{h^{(k-1)}\in\mathcal{H}^{(k-1)}} f_\ell^{(k)}(q,Q;h^{(k-1)}) \underbrace{\prod_{k'=1}^{k-1} f_\ell^{(k')}(h_{k'}^{(k-1)},Q;h_{1:k'-1}^{(k-1)})(1-\alpha_\ell(h_{k'}^{(k-1)},Q))}_{\text{dispatched to and declined by each position in } h^{(k-1)}}. \tag{5}$$

Here, $h_{1:k'-1}^{(k-1)}$ includes the first $k'-1$ dispatch positions given dispatch history $h^{(k-1)}$. With this, we can compute the rate at which each position $q\in[\bar{Q}]$ receives items of type $\ell\in\mathcal{L}$ (across all dispatches), when the length of the queue is $Q\in[\bar{Q}]$:

$$\nu_\ell(q,Q) \triangleq \mu_\ell \sum_{k=1}^{p} g_\ell^{(k)}(q,Q). \tag{6}$$

Additionally, we define the following rate terms for all $Q\in[\bar{Q}]$ and $q\in[Q]$, corresponding to events taking places at different positions in the queue:

$$r^0(q,Q) \triangleq \sum_{\ell\in\mathcal{L}} \nu_\ell(q,Q)\alpha_\ell(q,Q) + \theta, \tag{7}$$

$$r^+(q,Q) \triangleq \sum_{q'=1}^{q-1} r^0(q',Q), \tag{8}$$

$$r^-(q,Q) \triangleq \sum_{q'=q+1}^{Q} r^0(q',Q), \tag{9}$$

Intuitively, $r^0(q,Q)$ defined in (7) is the rate at which the agent at position q in the queue leaves the queue when the length of the queue is Q, i.e., the rate at which the agent receives and accepts an item, plus the rate of agent reneging exogenously. $r^+(q,Q)$ and $r^-(q,Q)$ given by (8) and (9) correspond to the rates at which agents in positions ahead of or behind position q leave the queue, respectively. Finally, $r(Q)$ represents the total rate at which the queue length changes, which combines the rate for any agent in the queue to leave, and the rate at which new agents arrive and join the queue at the tail.

$$r(Q) \triangleq \sum_{q'=1}^{Q} r^0(q',Q) + \lambda\phi(Q). \tag{10}$$

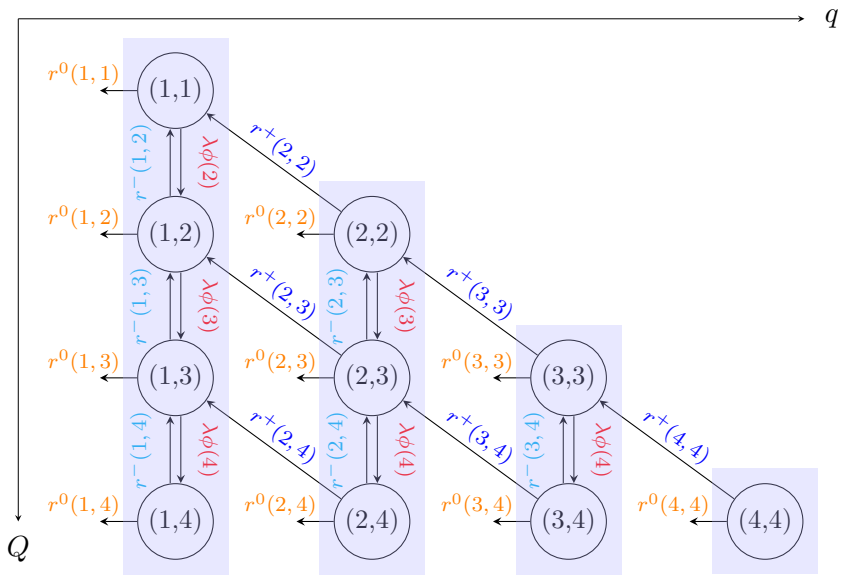


Figure 1: Illustration of the state transition diagram (of an agent) for $\bar{Q} = 4$.

Note that $r(Q) = r^0(q,Q) + r^+(q,Q) + r^-(q,Q) + \lambda\phi(Q)$ for all $q \in [Q]$.

By Poisson thinning, with these rate terms defined, the continuation payoffs $V = \big(V(q,Q)\big)_{Q\in[\bar{Q}],\, q\in[Q]}$ satisfy the following system of equations.

$$V(q,Q) = \underbrace{-\frac{c_w}{r(Q)}}_{\text{waiting cost}} + \underbrace{\frac{\sum_{\ell\in\mathcal{L}} \nu_\ell(q,Q)\alpha_\ell(q,Q)w_\ell}{r(Q)}}_{\text{payoff from accepted items}} + \underbrace{\frac{r^+(q,Q)}{r(Q)}\cdot V(q-1,Q-1)}_{\text{an agent in front leaves the queue}}$$
$$+ \underbrace{\frac{r^-(q,Q)}{r(Q)}\cdot V(q,Q-1)}_{\text{an agent behind leaves the queue}} + \underbrace{\frac{\lambda\phi(Q)}{r(Q)}\cdot V(q,Q+1)}_{\substack{\text{a new agent joins}\\ \text{at the tail of the queue}}}, \quad \forall Q\in[\bar{Q}],\ \forall q\in[Q], \tag{11}$$

with boundary conditions $V(0,\cdot) = V(\cdot,0) = V(\cdot,\bar{Q}+1) = 0$.[10] Equation (11) captures the discrete-event dynamics of the underlying continuous-time Markov chain, and requires solving a linear system with $\mathcal{O}(\bar{Q}^2)$ variables and $\mathcal{O}(\bar{Q}^2)$ equations. This typically results in a computational complexity of $\mathcal{O}(\bar{Q}^6)$—which may become prohibitive as $\bar{Q}$ increases. Remarkably, we show in the following lemma that a closed-form solution exists for *any* strategy and dispatch policy, resulting in only $\mathcal{O}(\bar{Q}^2)$ complexity.

Lemma 1 (Continuation Payoff Evaluation)**.** Under any dispatch policy, given any strategy adopted by all agents, the continuation payoff V can be computed via recursion

$$V(q,\bar{Q}) = a_{(q,\bar{Q})} + c_{(q,\bar{Q})}V(q-1,\bar{Q}-1), \qquad \forall q\in[\bar{Q}],$$
$$V(q,Q) = a_{(q,Q)} + b_{(q,Q)}V(q,Q+1) + c_{(q,Q)}V(q-1,Q-1), \qquad \forall Q\in[\bar{Q}-1],\ \forall q\in[Q],$$

where the coefficients $\big(a_{(q,Q)}, b_{(q,Q)}, c_{(q,Q)}\big)_{Q\in[\bar{Q}],q\in[Q]}$, depend on the strategy and the dispatch policy, and can be computed in closed forms.

[10] The event of exogenous reneging, which occurs with probability $\theta/r(Q)$, does not appear in the Bellman equation (11) because it yields a payoff of zero.

The proof of this lemma and the expressions of $(a_{(q,Q)}, b_{(q,Q)}, c_{(q,Q)})_{Q\in[\bar{Q}],q\in[Q]}$ are provided in Appendix B.1. The key observation is that agents *cannot* move backward in the queue, which enables us to compute $V(q,Q)$ recursively starting from $q=1$. Figure 1 provides an illustration of the state transition diagram (of an agent) for $\bar{Q}=4$. For the head of the queue $q=1$, the transition rates between states for different queue lengths Q resemble those of a birth-death process, resulting in a system of linear equations that yields closed-form solutions for $V(1,Q)$ for all $Q\in[\bar{Q}]$. Once $V(q,Q)$ is obtained for all $Q\in[\bar{Q}]$, the expressions for $V(q+1,Q)$ can be evaluated analogously. The complete procedure is given in Algorithm 4 in Appendix B.1. It is worth noting that $V(\sigma,\sigma_-,q,Q)$ can be evaluated in closed form in a similar manner when the focal agent's strategy σ differs from that of the other agents σ_-. For any strategy and dispatch policy, we can also analyze other performance metrics in closed forms, including the expected and steady-state distribution queue length, item fulfillment rate, and variance of agent payoffs. Further details and numerical examples on these performance metrics are provided in Section 5, while the theoretical framework for computing the variance is presented in Appendix B.9.

The closed-form solution for the continuation payoffs enable both the characterization and the computation of equilibria, which we address in the following sections. We note that the recursion in Lemma 1 applies to cases in which agents join at the tail of the queue. There exist extensions that violate this assumption. In such cases, one can still evaluate the continuation payoff by solving an appropriate system of linear equations akin to (11). In Appendix D.2, we discuss and experiment with an extension in which agents may be placed at the front of the queue upon arrival, as originally proposed by Su and Zenios [2004].

3.2 Existence of MPE

We now provide a set of sufficient conditions for a strategy to form a symmetric MPE under any dispatch policy that does not offer an item to the same agent more than once. The same characterization also allows us to establish that an MPE always exists under such policies.

Consider any strategy $\sigma^*=(\alpha^*,\phi^*)$, and the corresponding continuation payoff when σ^* is adopted by all agents $V^* \triangleq (V(\sigma^*,\sigma^*,q,Q))_{Q\in[\bar{Q}],q\in[Q]}$. The equilibrium conditions (3) and (4) on the queue-joining decisions are straightforward to verify once V^* is computed via Lemma 1. To check condition (2), we must formalize the *best-response* problem faced by a focal agent in the queue regarding whether to accept items of each type, given that strategy σ^* is adopted by the rest of the agents (including those arriving later in time).

Intuitively, an agent in the queue decides dynamically whether to accept their received items given the evolving queue length and their queue positions. This is a continuous-time infinite-horizon Markov decision process (MDP) with an absorbing state—the agent maximizes her expected payoff until receiving and accepting an item, or reneging from the queue. Let $\mathcal{A}_{q,Q}$ denote the set of feasible acceptance strategies that coincide with α^* in all states except (q,Q), i.e.,

$$\mathcal{A}_{q,Q}(\alpha^*) \triangleq \left\{ \alpha \;\middle|\; \alpha_\ell(q,Q)\in[0,1],\ \forall \ell\in\mathcal{L};\ \alpha_\ell(q',Q')=\alpha^*_\ell(q',Q'),\ \forall (q',Q')\neq(q,Q) \right\},$$

we can write the Bellman optimality equation for all $Q\in[\bar{Q}]$ and all $q\in[Q]$ as,

$$V^*(q,Q) = \underbrace{-\frac{c_w}{M}}_{\text{waiting cost}} + \max_{\alpha\in\mathcal{A}_{q,Q}(\alpha^*)} \left\{ \sum_{\ell\in\mathcal{L}} \left[\underbrace{\frac{\nu_\ell(q,Q)\alpha_\ell(q,Q)w_\ell}{M}}_{\text{accept type } \ell \text{ item}} \right] + \underbrace{\frac{r^+(q,Q)}{M} V^*(q-1,Q-1)}_{\text{agents in front accept or renege}} \right.$$

$$+ \underbrace{\frac{r^-(q,Q)}{M}V^*(q,Q-1)}_{\text{agents behind accept or renege}} + \underbrace{\frac{(M-r(Q))}{M}V^*(q,Q)}_{\text{no event happens}}\Bigg\} + \underbrace{\frac{\lambda\phi^*(Q)}{M}V^*(q,Q+1)}_{\text{new agent joins}}, \tag{12}$$

where $r(Q)$ as defined in (10) takes ϕ^* as the queue-joining strategy, and $M \triangleq \sum_{\ell\in\mathcal{L}}\mu_\ell + \lambda + \bar{Q}\theta$ is an upper bound on the total rate for any event to happen, and is weakly larger than $r(Q)$ for all $Q \in [\bar{Q}]$. Instead of dividing by $r(Q)$ as in (11), the constant M is introduced as part of the uniformization technique used to turn the continuous-time MDP into an equivalent discrete-time MDP (e.g., see Chapter 5.8 of Ross 1995).

Multiplying both sides by M and rearranging the terms, the agent's best response problem (12) regarding the acceptance decisions at state (q,Q) can be written as:[11]

$$\max_{\alpha\in\mathcal{A}_{q,Q}(\alpha^*)} \quad H_{q,Q}(\alpha;V^*) \tag{BR}$$

where

$$H_{q,Q}(\alpha;V^*) \triangleq \sum_{\ell\in\mathcal{L}} \underbrace{\nu_\ell(q,Q)\alpha_\ell(q,Q)\big(w_\ell - V^*(q,Q)\big)}_{\text{(i) accept type } \ell \text{ item}} + \underbrace{r^+(q,Q)\big(V^*(q-1,Q-1)-V^*(q,Q)\big)}_{\text{(ii) agent in front leaves the queue}}$$
$$+ \underbrace{r^-(q,Q)\big(V^*(q,Q-1)-V^*(q,Q)\big)}_{\text{(iii) agent behind leaves the queue}}. \tag{13}$$

The three components of $H_{q,Q}(\alpha;V^*)$ in (13) highlight the tradeoffs an agent faces when deciding whether to accept an item:

(i) For each $\ell \in \mathcal{L}$, $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ corresponds to the rate at which the agent receives and accepts items of this type, and $w_\ell - V^*(q,Q)$ reflects the tradeoff between leaving the queue with payoff w_ℓ vs remaining in the queue and receiving the continuation payoff $V^*(q,Q)$.

(ii) $r^+(q,Q)$ is the rate at which some agent in front of position q leaves, in which case the focal agent moves forward and receives a continuation payoff of $V^*(q-1,Q-1)$ instead of $V^*(q,Q)$.

(iii) $r^-(q,Q)$ represents the rate at which an agent behind position q in the queue leaves, leading to a continuation payoff of $V^*(q,Q-1)$ instead of $V^*(q,Q)$ for the focal agent.

Parts (ii) and (iii) are included in (13) since the focal agent's acceptance strategies $(\alpha_\ell(q,Q))_{\ell\in\mathcal{L}}$ affect both rates of state changes $r^+(q,Q)$ and $r^-(q,Q)$: when the agent declines an item, the item may be accepted by some agent in front or behind position q in the queue in one of the subsequent dispatches. Such "cascading effects" therefore need to be taken into consideration as the agent decided whether to accept each received item.

Given the above characterization of the best-response problem, we now provide a set of necessary and sufficient conditions for a strategy σ^* to form a symmetric MPE, when the dispatch policy satisfies the following very mild assumption.

Assumption 1. A dispatch policy does not offer any item to the same agent more than once. That is, for each item type $\ell \in \mathcal{L}$, for each dispatch $k \in [p]$, under any queue length $Q \in [\bar{Q}]$ and all prior histories $h^{(k-1)} \in \mathcal{H}^{(k-1)}$, $f_\ell^{(k)}(q,Q;h^{(k-1)}) = 0$ for any position $q \in h^{(k-1)}$ that have already been offered the item.

[11] To see this, note that after multiplying both sides with M, the Bellman optimality equation (12) can be rewritten as $\theta V^*(q,Q) = -c_w + \lambda\phi^*(Q)\big(V^*(q,Q+1)-V^*(q,Q)\big) + \max_{\alpha\in\mathcal{A}_{q,Q}(\alpha^*)}\{\sum_{\ell\in\mathcal{L}}\nu_\ell(q,Q)\alpha_\ell(q,Q)\big(w_\ell - V^*(q,Q)\big) + r^+(q,Q)\big(V^*(q-1,Q-1)-V^*(q,Q)\big) + r^-(q,Q)\big(V^*(q,Q-1)-V^*(q,Q)\big)\}$.

We can verify that the Strict FIFO dispatch policy discussed in Example 1 satisfies Assumption 1, while the pure random dispatch policy from Example 2 does not. It is, however, straightforward to adapt the pure random policy: for each dispatch, we can re-draw another agent uniformly at random until an agent who had not yet received the item is selected.

Lemma 2 (Equilibrium Conditions)**.** Under Assumption 1, i.e., when the dispatch policy does not offer any item to the same agent more than once, a strategy $\sigma^* = (\alpha^*, \phi^*)$ forms a symmetric MPE if and only if σ^* and its corresponding continuation payoff V^* satisfy

$$\alpha_\ell^*(q,Q) = P_{[0,1]}\left[\alpha_\ell^*(q,Q) + \frac{\partial H_{q,Q}(\alpha^*;V^*)}{\partial \alpha_\ell(q,Q)}\right], \qquad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{14}$$

$$\phi^*(Q-1) = P_{[0,1]}\big[\phi^*(Q-1) + V^*(Q,Q)\big], \qquad \forall Q \in [\bar{Q}]. \tag{15}$$

Moreover, $(\partial H_{q,Q}(\alpha;V)/\partial \alpha_\ell(q,Q))_{\ell\in\mathcal{L},Q\in[\bar{Q}],q\in[Q]}$ can be evaluated in closed form.

Here, $P_{[0,1]}[\cdot]$ represents the projection operator onto $[0,1]$. It is worth noting that when computing $\partial H_{q,Q}(\alpha;V)/\partial\alpha_\ell(q,Q)$, we do not differentiate V with respect to $\alpha_\ell(q,Q)$. The full proof of this lemma is provided in Appendix B.2. We derive in Appendix B.3 the expressions of the partial derivatives of $(\partial H_{q,Q}(\alpha;V)/\partial\alpha_\ell(q,Q))_{\ell\in\mathcal{L},Q\in[\bar{Q}],q\in[Q]}$, which apply to any dispatch policy and do not rely on Assumption 1.

Intuitively, (15) is equivalent to (3) and (4), and guarantees that the queue joining decision ϕ^* is optimal when the rest of the agents adopt strategy σ^*. Equation (14) corresponds to the first-order (necessary) conditions of the agent's best-response problem (BR) regarding the acceptance strategies. For general dispatch policies, $H_{q,Q}(\alpha;V^*)$ as defined in (13) may not be concave in $\alpha_\ell(q,Q)$ since the rate terms $\nu_\ell(q,Q)$, $r^+(q,Q)$ and $r^-(q,Q)$ are linear combinations of the $g_\ell^{(k)}(q,Q)$'s, and the probability of receiving an item during its k^{th} dispatch $g_\ell^{(k)}(q,Q)$ given by (5) can be a polynomial in $\alpha_\ell(q,Q)$ of degree at most $k-1$. When no position receives an item more than once, however, we can prove that $\nu_\ell(q,Q)$ is independent to $\alpha_\ell(q,Q)$ and each $g_\ell^{(k)}(q,Q)$ is at most linear in $\alpha_\ell(q,Q)$. Consequently, every $H_{q,Q}(\alpha;V^*)$ is linear in $\alpha_\ell(q,Q)$, the first-order conditions become sufficient and imply the optimality for (BR), and requirement (2) for MPE is also satisfied.

Theorem 1 (Existence of MPE)**.** *There exists a symmetric Markov perfect equilibrium under any dispatch policy satisfying Assumption 1, i.e., when the policy does not dispatch an item to the same agent more than once.*

We prove this theorem in Appendix B.4. Briefly, consider the equilibrium conditions (14) and (15) as fixed-point equations of the strategy σ, where V is the continuation payoff corresponding to $\sigma = (\alpha, \phi)$. Applying Brouwer's fixed-point theorem, we know that a solution to the fixed-point equations must exist, since the continuation payoff V is continuous in the strategy σ (Lemma 1), and the partial derivative $\partial H_{q,Q}(\alpha;V)/\partial\alpha_\ell(q,Q)$ is continuous in both the acceptance strategy α and the continuation payoff V (Appendix B.3).

Given the existence of MPEs established in Theorem 1, it is natual to ask whether the equilibrium is unique. In Appendix C.3, we show via an example that there may exist multiple MPEs with substantially different outcomes, even under the most simple dispatch policies such as Strict FIFO dispatching.

One may also wonder whether there always exists deterministic (i.e., all acceptance and queue-joining probabilities taking value 0 or 1) equilibrium strategies, especially given the linearity of $H_{q,Q}(\alpha;V)$. In general, randomized strategies may still be necessary, since the expression for $H_{q,Q}(\alpha;V)$ depends on V, which itself is shaped by the strategy.

For general dispatch policies that violate Assumption 1, $H_{q,Q}(\alpha; V)$ may not be concave in α so the first-order conditions do not necessarily guarantee optimality. However, the proof of Theorem 1 also implies that (i) equations (14) and (15) are necessary conditions for MPE, and (ii) a strategy satisfying these necessary conditions always exist under any dispatch policy. The algorithms we propose compute such strategies, after which we can numerically verify whether they form optimal solutions to the best-response problem (BR).

3.3 Simple Dispatch Policies

In the rest of this section, we consider a family of *simple dispatch policies* under which the equilibrium conditions can be significantly simplified. The simplified equilibrium conditions also motivate an approximate projection algorithm that speeds up our equilibrium computation, which we discuss in detail in Section 4.

Definition 2 (Simple Dispatch Policy)**.** A simple dispatch policy is one that satisfies:

1. *Queue-length independence.* For each dispatch $k \in [p]$ and all prior dispatch histories $h^{(k-1)} \in \mathcal{H}^{(k-1)}$, the probability for any position $q \in [\bar{Q}]$ to receive items of type $\ell \in \mathcal{L}$ is independent to the length of the queue, i.e.,

$$f_\ell^{(k)}(q, Q; h^{(k-1)}) = f_\ell^{(k)}(q, \bar{Q}; h^{(k-1)}), \quad \forall Q \in \{q, q+1, \ldots, \bar{Q}\}.$$

2. *Monotonic dispatch.* No item is offered to any agent positioned in front of those who have already received it.Formally, for each dispatch $k \in \{2, 3, \ldots, p\}$, each item type $\ell \in [L]$, and all prior dispatch histories $h^{(k-1)} \in \mathcal{H}^{(k-1)}$,

$$f_\ell^{(k)}(q, Q; h^{(k-1)}) = 0, \quad \forall q < h_{k-1}^{(k-1)}, \ \forall Q \in [\bar{Q}].$$

Note that simple dispatch policies may not necessarily satisfy Assumption 1.[12] We can verify that Strict FIFO defined in Example 1 is a simple dispatch policy, while the pure random dispatch in Example 2 is not. The term *simple* refers to the strategic environment this class of policies induce (discussed below), rather than the dispatch policies themselves. Indeed, this class of simple dispatch policies includes sophisticated history-dependent mechanisms such as the Geometric Hopping policy introduced in Definition 3 in Appendix C.1.1, which (i) upon the arrival of each item, draws the first dispatch position from a geometric distribution, and (ii) after each decline, draws another geometric random variable, adds it to the previous dispatch position, and sends the item to the position equals the sum.[13]

With the two properties of a simple dispatch policy, the outcome of an agent at some position q in the queue is not affected by the length of the queue or the actions of those in positions $q' > q$ in the queue. This allows us to substantially simplify the equilibrium conditions from Lemma 2.

[12] Monotonic dispatch requires that if an item is sent to an agent at $q = 3$ and it is rejected, agents at $q = 1$ and 2 will not be eligible to receive the item in subsequent dispatches. However, the same agent at $q = 3$ can be offered the item multiple times before any agent at positions $q \geq 3$ receives the item.

[13] Simple dispatch policy also encompasses the static randomized FIFO policy introduced in Castro et al. [2021], where p static bins are constructed independently of the queue lengths, and items are dispatched sequentially bin by bin, with each item assigned uniformly at random within each bin. This differs from the dynamic version of the randomized FIFO policy discussed in Section 1 (and formally defined in Definition 4) where the p bins are constructed dynamically based on the current queue length. The Dynamic Randomized FIFO policy is therefore *not* a simple dispatch policy.

Corollary 1. An MPE always exists under simple dispatch policies. In this setting, a strategy $\sigma^* = (\alpha^*, \phi^*)$ constitutes a symmetric MPE if the complementarity conditions are satisfied by σ^* together with its associated continuation payoff V^*:

$$V^*(q,Q) < w_\ell \implies \alpha^*_\ell(q,Q) = 1, \quad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{16}$$
$$V^*(q,Q) > w_\ell \implies \alpha^*_\ell(q,Q) = 0, \quad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{17}$$
$$V^*(Q,Q) > 0 \implies \phi^*(Q-1) = 1, \quad \forall Q \in [\bar{Q}], \tag{18}$$
$$V^*(Q,Q) < 0 \implies \phi^*(Q-1) = 0, \quad \forall Q \in [\bar{Q}]. \tag{19}$$

The proof of this result is provided in Appendix B.6. An example of the equilibrium outcome under the Strict FIFO dispatch policy, illustrating the application of Corollary 1, is provided in Appendix C.2.

Intuitively, this simplification arises because, under simple policies, the "cascading effects" of the focal agent's decision—where another agent ahead or behind would immediately accept the item if the focal agent declines—are eliminated. Specifically, under simple policies, an agent's acceptance strategy does not affect her rate of moving forward in the queue (i.e., $\partial r^+(q,Q)/\partial \alpha_\ell(q,Q) = 0$), because agents ahead of position q in the queue will never receive the item again once the focal agent receives it. Additionally, $V^*(q, Q-1) = V^*(q,Q)$ since the dispatch probabilities do not depend on the queue length. As a result, the best-response problem (BR) reduces to

$$\max_{\alpha \in \mathcal{A}_{q,Q}(\alpha^*)} \quad \sum_{\ell \in \mathcal{L}} \nu_\ell(q,Q)\alpha_\ell(q,Q)\big(w_\ell - V^*(q,Q)\big). \tag{20}$$

We further show that $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ is non-decreasing in $\alpha_\ell(q,Q)$ for any dispatch policy (see Lemma 3),[14] implying that an agent will always accept items whose immediate payoff exceeds the continuation payoff, i.e., $w_\ell > V^*(q,Q)$, and will reject if $w_\ell < V^*(q,Q)$. To see why (16)–(19) admit a solution, we examine the equivalent fixed-point equations by considering V as a function of α:

$$\alpha_\ell(q,Q) = P_{[0,1]}\big[\alpha_\ell(q,Q) + (w_\ell - V(q,Q))\big], \qquad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{21}$$
$$\phi(Q-1) = P_{[0,1]}\big[\phi(Q-1) + V(Q,Q)\big], \qquad \forall Q \in [\bar{Q}], \tag{22}$$

It can be shown that any solution of (21) and (22) satisfy (16)–(19). By the continuity of V with respect to α, these equations admit a solution by Brouwer's fixed-point theorem, ensuring the existence of an MPE under simple dispatch policies. It is also worth noting that even for general, non-simple dispatch policies, (21) and (22) *always* admit a solution. In this context, such a solution can be viewed as an *approximate* equilibrium, in which agents disregard the possibility and impact of immediate state changes after declining an offered item—effectively ignoring the second and third terms in the objective $H_{q,Q}(\alpha; V^*)$ of the full best-response problem (13), leading to the same simplification as in (20).

4 Equilibrium Computation

In this section, we present an iterative gradient projection algorithm with closed-form updates for computing a Markov perfect equilibrium under arbitrary dispatch policies. For simple dispatch policies, we establish global and finite convergence guarantees and show that the update direction reduces to a form that only requires the computation of continuation payoffs. Additionally, this

[14] This is non-trivial because $\partial \nu_\ell(q,Q)/\partial \alpha_\ell(q,Q) \le 0$.

Algorithm 1 Gradient Projection Algorithm

Input: Model parameters, initial strategy $\sigma^{(0)}$, stepsize $(\gamma^{(t)})_{t=0}^{\infty}$, termination tolerance $\varepsilon > 0$, maximum iterations $T \in \mathbb{Z}_{>0}$.
Output: Strategy σ.
Initialize $t \leftarrow 0$.
while $t < T$ **do**
  Solve $V^{(t)}$ corresponding to $\sigma^{(t)}$ using Lemma 1.
  if $\mathrm{Loss}(\alpha^{(t)}, V^{(t)}) < \varepsilon$ **and** $\mathrm{Loss}(\phi^{(t)}, V^{(t)}) < \varepsilon$ **then**
    break
  end if
  $\alpha_\ell^{(t+1)}(q,Q) \leftarrow P_{[0,1]}\left[\alpha_\ell^{(t)}(q,Q) + \gamma^{(t)} \frac{\partial H_{q,Q}(\alpha^{(t)}; V^{(t)})}{\partial \alpha_\ell(q,Q)}\right]$ for all $Q \in [\bar{Q}]$, $q \in [Q]$, $\ell \in \mathcal{L}$.
  $\phi^{(t+1)}(Q-1) \leftarrow P_{[0,1]}\left[\phi^{(t)}(Q-1) + \gamma^{(t)} V^{(t)}(Q,Q)\right]$ for all $Q \in [\bar{Q}]$.
  Increment $t \leftarrow t+1$.
end while
return $\sigma^{(t)}$.

simplified procedure can be applied to general dispatch policies as an efficient approximation, yielding substantial computational savings.

Our baseline solver is detailed in Algorithm 1. The update direction for the acceptance strategy is determined by the gradient of the agent's best-response problem in eq. (BR), motivated by seeking the fixed point characterized in eq. (14) at convergence. The update for the joining decision is based on the continuation payoff after joining at the tail of the queue, and is similarly motivated by eq. (15). If joining the queue results in a positive continuation payoff, we increase the joining probability.

We design the termination criteria to ensure that the the equilibrium conditions from Lemma 2 are approximately satisfied. Specifically, with $(x)^+ \triangleq \max\{x, 0\}$ and $(x)^- \triangleq \max\{-x, 0\}$ indicating the positive and negative parts of any number $x \in \mathbb{R}$, $\mathrm{Loss}(\alpha, V)$ and $\mathrm{Loss}(\phi, V)$ defined as follows quantify the maximum deviation (across item types, queue lengths, and queue positions) from the equilibrium conditions for the acceptance and joining strategies, respectively.

$$\mathrm{Loss}(\alpha, V) \triangleq \max_{Q \in [\bar{Q}], q \in [Q], \ell \in \mathcal{L}} \left\{ (1 - \alpha_\ell(q,Q)) \left(\frac{\partial H_{q,Q}(\alpha; V)}{\partial \alpha_\ell(q,Q)} \right)^+ + \alpha_\ell(q,Q) \left(\frac{\partial H_{q,Q}(\alpha; V)}{\partial \alpha_\ell(q,Q)} \right)^- \right\}, \tag{23}$$

$$\mathrm{Loss}(\phi, V) \triangleq \max_{Q \in [\bar{Q}]} \left\{ (1 - \phi(Q-1))(V(Q,Q))^+ + \phi(Q-1)(V(Q,Q))^- \right\}. \tag{24}$$

Though in closed form, the evaluation of the gradient $\partial H_{q,Q}(\alpha^{(t)}; V^{(t)}) / \partial \alpha_\ell(q,Q)$ can be computationally intensive, as it requires quantifying how the acceptance probability $\alpha_\ell(q,Q)$ affects the rates at which each position $q' \in [Q]$ receives items of type $\ell \in \mathcal{L}$ across all p dispatches (see Appendix B.3 for detailed derivations). We address this issue in the next subsection.

4.1 An Approximate Gradient Projection Algorithm

In this subsection, we discuss a simplified projection algorithm as described in Algorithm 2, where the termination criteria on the acceptance strategy is given by

$$\mathrm{Loss}(\alpha, V) = \max_{Q \in [\bar{Q}], q \in [Q], \ell \in \mathcal{L}} \left\{ (1 - \alpha_\ell(q,Q)) \left(w_\ell - V(q,Q)\right)^+ + \alpha_\ell(q,Q) \left(w_\ell - V(q,Q)\right)^- \right\}.$$

Algorithm 2 Approximate Gradient Projection Algorithm

Input: Model parameters, initial strategy $\sigma^{(0)}$, stepsize $(\gamma^{(t)})_{t=0}^{\infty}$, termination tolerance $\varepsilon > 0$, maximum iterations $T \in \mathbb{Z}_{>0}$.
Output: Strategy σ.
Initialize $t \leftarrow 0$.
while $t < T$ **do**
  Solve $V^{(t)}$ corresponding to $\sigma^{(t)}$ using Lemma 1.
  if $\mathrm{Loss}(\alpha^{(t)}, V^{(t)}) < \varepsilon$ **and** $\mathrm{Loss}(\phi^{(t)}, V^{(t)}) < \varepsilon$ **then**
    break
  end if
  $\alpha_\ell^{(t+1)}(q,Q) \leftarrow P_{[0,1]}\left[\alpha_\ell^{(t)}(q,Q) + \gamma^{(t)}(w_\ell - V^{(t)}(q,Q))\right]$ for all $Q \in [\bar{Q}]$, $q \in [Q]$, $\ell \in \mathcal{L}$.
  $\phi^{(t+1)}(Q-1) \leftarrow P_{[0,1]}\left[\phi^{(t)}(Q-1) + \gamma^{(t)}(V^{(t)}(Q,Q))\right]$ for all $Q \in [\bar{Q}]$.
  Increment $t \leftarrow t+1$.
end while
return $\sigma^{(t)}$.

In contrast to the gradient projection algorithm (Algorithm 1), this projection algorithm (Algorithm 2) updates each acceptance strategy $\alpha_\ell(q,Q)$ by simply comparing the immediate payoff from accepting the job w_ℓ with the continuation payoff $V(q,Q)$. When the payoff w_ℓ is strictly higher (or lower) than $V(q,Q)$, the acceptance probability is increased (decreased). Algorithm 2 offers significant computational advantages by eliminating the need for gradient operations. It is motivated by the complementarity conditions in Corollary 1 and the simplified fixed-point equations (21) and (22) which characterize an approximate equilibrium in which agents disregard the (typically second-order) effects of other agents immediately accepting items they have declined—specifically, the second and third terms in the objective $H_{q,Q}(\alpha)$ in (BR) (see the discussion in Section 3.3). This approximation is justified when the continuation payoffs at (q,Q) and neighboring states $(q-1,Q-1)$ and $(q,Q-1)$ are similar in equilibrium. For example, under a purely random dispatch policy with a large queue, reducing the total queue length by one or moving up by one position has a negligible impact on the agent's continuation payoff. In such settings, Algorithm 2 is expected to provide an accurate and efficient equilibrium approximation. As evidence, Appendix D.4 shows that under a pure random dispatch policy, both algorithms yield very similar performance metrics and continuation payoffs upon convergence.[15] Moreover, as noted in Section 3.3, under simple dispatch policies, Algorithms 1 and 2 become essentially equivalent, since the second and third terms in $H_{q,Q}(\alpha; V^*)$ (eq. (13)) vanish.[16]

[15] However, this does not always hold. In Appendix C.4, we present a (pathological) counterexample showing that accounting for the immediate effects of declining items can be crucial for identifying the correct update direction and ensuring convergence to the proper equilibrium strategy.

[16] Under simple dispatch policies, line 9 in Algorithm 1 reduces to

$$\alpha_\ell^{(t+1)}(q,Q) \leftarrow P_{[0,1]}\left[\alpha_\ell^{(t)}(q,Q) + \gamma^{(t)}\left(\nu_\ell(q,Q) + \frac{\partial \nu_\ell(q,Q)}{\partial \alpha_\ell(q,Q)}\right)(w_\ell - V^{(t)}(q,Q))\right].$$

By Lemma 3, $\nu_\ell(q,Q) + \partial\nu_\ell(q,Q)/\partial\alpha_\ell(q,Q) \geq 0$, so the only difference from line 9 in Algorithm 2 is the step size.

Convergence. Both Algorithms 1 and 2 are related to the projection-based methods for complementarity problems [Facchinei and Pang, 2007].[17] However, existing results on the convergence of such methods do not guarantee convergence in our setting. For example, the projection-based methods in Section 12.1.1 of Facchinei and Pang [2007] require additional monotonicity assumptions that are not satisfied in our context, even under simple dispatch policies. Nevertheless, by leveraging the structure of our game, we establish a convergence guarantee for simple dispatch policies.

Theorem 2. *Under any stepsize rule such that* $\sum_{t=0}^{+\infty} \gamma^{(t)} = +\infty$*, Algorithm 2 converges to an equilibrium strategy within a finite number of iterations for simple dispatch policies for any initial strategy* $\sigma^{(0)}$*.*

The proof of this theorem is provided in Appendix B.7. We show that the continuation payoff V is monotonically increasing in each iteration t, and proceed by solving for the continuation payoff sequentially by queue position, starting from the head of the queue. By queue-length independence we get the same $V(q, Q)$ for any queue length Q when q is fixed. We show that once the continuation payoff for an agent at $q = 1$ converges in a finite number of iterations, with a termination slack of $\varepsilon = 0$, the agent at $q = 2$ will undergo a similar process. Using induction and the monotonic dispatch property, this argument can be generalized to all of the queue positions $q \in [\bar{Q}]$.

5 Numerical Results

This section presents counterfactual simulation results based on the driver airport queue setting studied in Castro et al. [2021], where agents (drivers) in the queue wait to be dispatched for items (trips). Unlike the prior work, however, our model maintains stochastic assumptions rather than relying on fluid approximations. This distinction is crucial, as the inherent randomness in arrival and service processes leads to a stochastically evolving queue length. Accounting for these fluctuations enables us to analyze (practical) dispatch policies that must adapt to the actual queue length. Notably, we find that our solver converges for the Dynamic Randomized FIFO policy described in Section 1, a practical alternative that cannot be analyzed using previous fluid analyses.

Dispatch Policies. The dispatch policies we analyze in this section include:

- *Strict FIFO:* A formal statement is provided in Example 1.
- *Direct FIFO:* Each item is offered to the first position in the queue where it is accepted in equilibrium. An illustrative calculation is given in Appendix C.2.
- *Pure Random:* Each item is assigned to an agent in the queue uniformly at random.
- *Dynamic Randomized FIFO (DRFIFO):* The queue is partitioned into p subsets, each containing approximately the same number of agents. An item is offered uniformly at random to an agent in the first bin; if rejected, the process continues sequentially with the next bin. A formal description of this policy is provided in Appendix C.1.2.

[17] In our context, we can reformulate the fixed-point equations (14) and (15) or (21) and (22) as a *nonlinear complementarity problem (NCP)*, where the goal is to find a vector x such that:

$$F(x) \geq 0, \quad x \geq 0, \quad \text{and} \quad x^\intercal F(x) = 0,$$

for some nonlinear function $F(x)$.

- *LIFO:* Each item is first offered to the agent at the tail of the queue. If rejected, the offer proceeds sequentially up the queue.

Performance Metrics. We evaluate the performance of the aforementioned dispatch policies at Chicago O'Hare International Airport using six key metrics: equilibrium net revenue, equilibrium trip throughput, expected queue length, expected waiting time, average driver earnings, and standard deviation of driver earnings. Platform throughput is defined as the number of dispatches completed per unit time by agents in the queue, while net revenue is the total payoff from fulfilled dispatches per unit time minus the opportunity cost incurred by agents waiting in the queue; this opportunity cost is set equal to the agents' waiting cost. The equilibrium expected queue length and waiting time are obtained from the steady-state queue-length distribution under equilibrium. We note that this steady-state distribution is derived using the stationary distribution of a birth-death process.

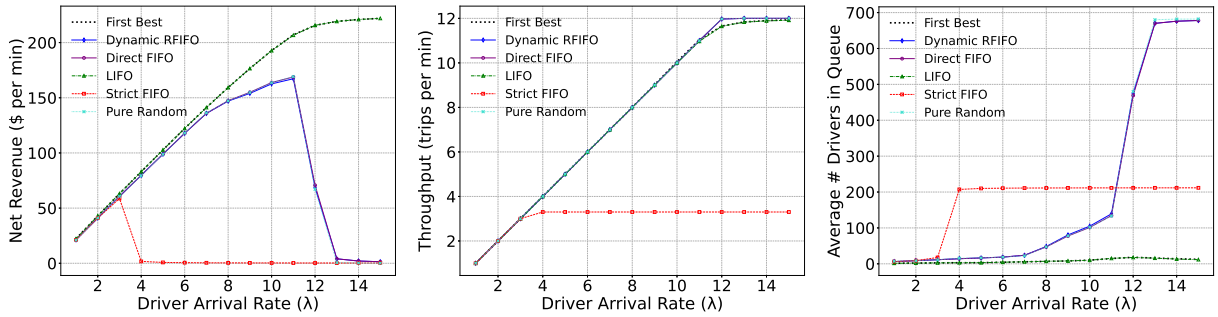
Standard Deviation of Earnings To derive the standard deviation of driver earnings, we recognize that an agent's terminal payoff equals the earnings from the job they accept, net of the waiting costs accumulated prior to acceptance. This observation lets us construct a stopped martingale: the agent's continuation value minus accumulated waiting costs, stopped at the (random) time the agent reaches the terminal state, at which point the payoff becomes deterministic. Under this construction, the variance of earnings conditional on the entry state equals the expected quadratic variation accrued up to the time the focal agent exits the queue. This quadratic variation decomposes into the squared increments across state transitions and the contributions from absorbing events in which the agent accepts a job and leaves the queue. The remaining quadratic variation at each state then satisfies a system of linear equations, with transition probabilities specified analogously to those in the system used to solve for the continuation payoff $V(q, Q)$ in Section 3.1. In Appendix B.9, we show that, for any strategy and dispatch policy, both the expected agent earnings and the variance of earnings can be computed in closed form within this framework.

Implementation. For simple dispatch policies (Strict and Direct FIFO), we apply the simple projection method described in Algorithm 2. For other dispatch policies, we adopt a hybrid approach for efficiency: we first run the simple projection Algorithm 2 to convergence to obtain a warm start, and then further refine the solution using Algorithm 1 until convergence. The convergence tolerance ε is set to be 10^{-4}. Throughout, we use a constant step size and random strategy initialization. See more details of our computational experiments in Appendix D.3. A GitHub repository replicating all our experiments is publicly available at: https://github.com/denise-cerna/queue-equilibrium-solver.

5.1 Varying Driver Supply

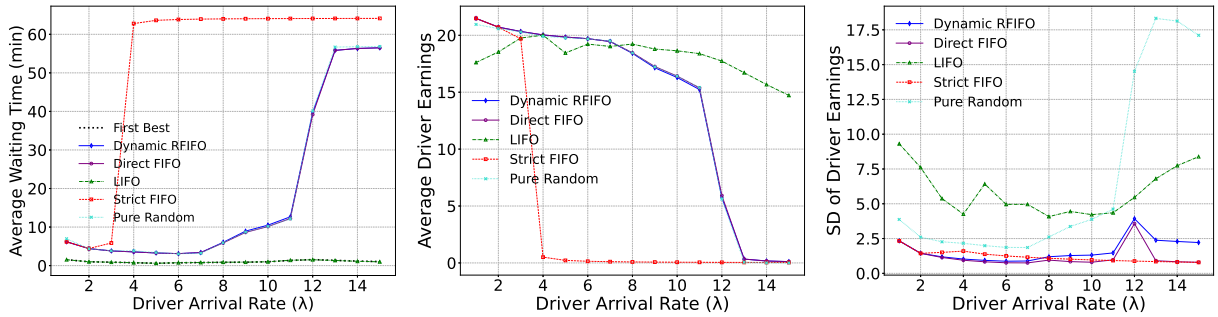
The Economy. To estimate the distribution of trips and net earnings across destinations, we utilize publicly available trip-level data from ridesharing platforms, as provided by the City of Chicago.[18] This dataset includes information such as trip fares (rounded to the nearest $2.50), origins and destinations at the Census Tract level, and trip start and end times rounded to 15-minute intervals. Since Chicago comprises 801 census tracts, we consider each tract as a potential destination (trip type). We assume a waiting cost of $c_w = 1/3$, which corresponds to an average

[18] https://data.cityofchicago.org/Transportation/Transportation-Network-Providers-Trips-2018-2022-/m6dm-c72p/about_data



(a) Net revenue. (b) Trip throughput. (c) Expected queue length.

Figure 2: Equilibrium net revenue, trip throughput, and expected queue length as the arrival rate of drivers varies at Chicago O'Hare.



(a) Expected waiting time. (b) Average driver earnings. (c) SD of driver earnings.

Figure 3: Expected waiting time, average driver earnings, and standard deviation (SD) of driver earnings as the arrival rate of drivers varies at Chicago O'Hare.

city driver earning $20 per hour. Using this assumption, along with data on average fares and trip durations, we estimate net earnings for each destination. We determine the fraction of trips for each destination based on the realized trip volume from the data. The rider's patience level is modeled as $p = 12$, assuming that a driver decline takes an average of 10 seconds and that riders are only willing to wait for 2 minutes for a match. We set the exogenous reneging rate of drivers to be negligible ($\theta \approx 0$), while allowing for endogenous reneging.

For experiments, we assume that the arrival rate of trips per minute is fixed, at $\sum_{\ell \in [L]} \mu_\ell = 12$. This roughly corresponds to the rate of completed trips during early evening hours on weekdays in the dataset. The distribution of trip rates and earnings in this data is displayed in Appendix D.1. We analyze the performance of various dispatch policies as we vary the driver arrival rate λ from 1 to 15. This captures the differences in performance as the system changes from being under-supplied to over-supplied. The performance under various dispatch policies are shown in Figure 2 and Figure 3. We see that the Strict FIFO, Direct FIFO, and Pure Random policies closely resemble the equilibrium assuming fluid approximation that was presented in Castro et al. [2021].

First Best. The *first-best* outcome is defined as the maximum net revenue attainable by the system if driver behavior—specifically, whether to accept trips or join the queue—can be fully controlled. In the fluid setting the queue is empty under the first-best of Castro et al. [2021].

However, in the stochastic setting, the queue is not always empty. The derivation of the first-best outcome, detailed in Appendix B.8, involves solving an infinite-horizon average-reward Markov Decision Process (MDP), where the state is defined by the total queue length Q. At each step, the platform decides (i) whether to dispatch an item ℓ and (ii) whether to admit an agent into position $Q+1$ in the queue. It is not necessary to track the individual state (q, Q) in formulating the first-best, as the platform has full control and can direct any agent in the queue to accept or decline an item. Under a different setting, Su and Zenios [2004] showed that LIFO is provably optimal and can achieve this first-best outcome. We can see from our experiments that LIFO does, in fact, follow the first-best performance metrics. Interestingly, as shown in Figure 2b, the first-best policy does not achieve the highest throughput—Dynamic Randomized FIFO, Direct FIFO, and Pure Random all yield higher throughput. By contrast, in the fluid analysis, the first-best net-revenue policy also achieves first-best throughput. This is because the first-best policy limits the number of agents in the queue to reduce waiting costs (see the significantly shorter queue length under the first-best policy in Figure 2c), thereby offering less buffer to absorb the randomness in trip arrivals.

Performance of More Complex Dispatch Policies. The solver developed in this paper offers a significant opportunity to analyze and evaluate more complex and dynamic dispatch policies that are highly practical yet challenging to study theoretically. In our experiments, we specifically look at the Dynamic Randomized FIFO policy discussed in Section 1. The prior work by Castro et al. [2021], using fluid approximations, was only able to model a static version of dispatch bin formation—fixing the set of drivers at certain queue positions (regardless of whether they were actually present in the queue) to be considered as candidates for receiving items in each dispatch. The results in Figure 2 and Figure 3 confirm that adapting the bin size to the observed queue length achieves similar performance, while being a simple, lightweight and undemanding policy that does not require any knowledge of the arrival rates of various item types. Remarkably, we observe that the standard deviation of driver earnings under Dynamic Randomized FIFO is similar to that of Direct FIFO and substantially lower than that of Pure Random dispatching. This confirms that Dynamic Randomized FIFO is able to strike a balance between performance across key metrics and fairness among drivers.

We also find that LIFO yields a high standard deviation of driver earnings, even though its net revenue performance matches the first-best benchmark. Nevertheless, in the oversupplied regime it still outperforms random dispatching with lower earning variances. The key difference is that under random dispatching, agents refrain from entering once the queue becomes too long; yet those who do enter tend to remain in the system and consequently experience substantial earnings volatility. Under LIFO, by contrast, nearly all agents choose to enter initially, but once their waiting time becomes excessive, their continuation payoff turns negative and they optimally exit the queue, which limits the variance attributable to waiting costs.

5.2 Mis-specified Dispatch Policies

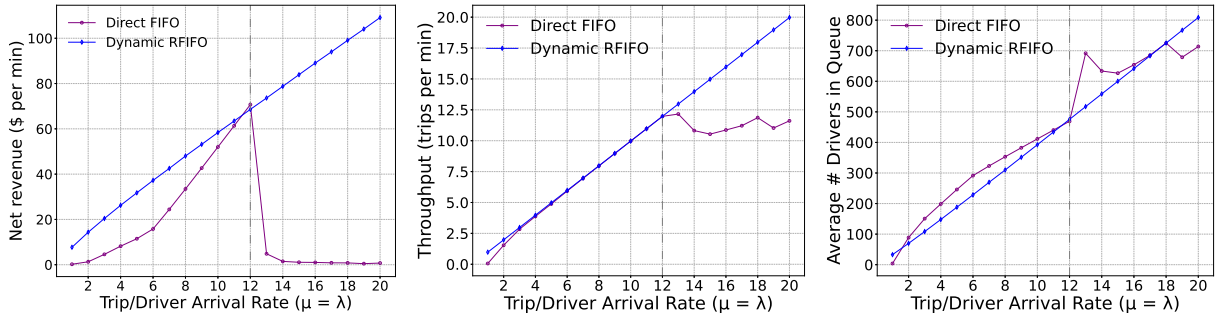
In the fluid setting of Castro et al. [2021], equilibrium outcomes are hard to compute when the model is mis-specified, since the policies themselves are constructed carefully for each economy with the intended equilibrium outcome in mind. Mis-specification, however, is inevitable due to errors in estimating the arrival rates of different item types and the payoffs of different items.

As an example, the Direct FIFO dispatch policy depends critically on assigning item types to specific queue positions based on these estimations. Yet, even under the fluid model it is still challenging to derive how the how equilibrium outcomes shift when the items are no longer sent

precisely to the "right places". This poses a major limitation, since practitioners need insight into the sensitivity of different policies when real world parameters deviate from the model assumptions.

Since our solver can be applied to any dispatch rule for any economy, we are able to study the robustness of different policies by comparing a (potentially) mis-specified Direct FIFO policy to the Dynamic Randomized FIFO policy that uses *no* information of the economy. In this experiment, we fix the direct FIFO policy designed based on an *assumed* item arrival rate of 12.

The rest follows the setup in Section 5 of the submission. We then apply our solver to instances where the *actual* item arrival rate equals the driver arrival rate and varies from 1 to 20. The computed equilibrium outcome is presented in Figure 4.



(a) Net revenue. (b) Trip throughput. (c) Expected queue length.

Figure 4: Equilibrium outcomes as the arrival rate of items varies at Chicago O'Hare.

When the market is over-supplied (i.e., when the item arrival rate is smaller than 12, the driver arrival rate), net revenue under both policies are close to zero since drivers will keep joining the queue until the net payoff from joining the queue is precisely zero. When the arrival rate of items is substantially higher than 12, Dynamic Randomized FIFO adapts and performs well. Direct FIFO, on the other hand, exhibits substantial sensitivity because items are now dispatched "too close" to the head of the queue, leading to some items being rejected by the drivers. As a result, the queue becomes effectively over-supplied once the amount of items that reach drivers who are willing to accept them drops below driver supply. In this case, we again observe long queues and close to zero net revenue.

6 Extensions

This section develops two extensions that broaden the applicability of our framework. We first allow items to be offered to multiple agents simultaneously and then introduce costly inspection, where agents incur a cost before learning an item's type. We can compute equilibria in both settings using adaptations of Algorithm 2.

6.1 Parallel Dispatching

In this section, we extend our framework by applying our solver to dispatching policies that present an item to *multiple* agents *simultaneously*, i.e., parallel dispatching. Such policies are quite popular in practice. For instance, Uber's recent Trip Radar product offers a trip request to several drivers at once, complementing the traditional exclusive offers that dispatch a request to a single driver only; similarly, in the UK, organs that have been declined more than three times are offered to all eligible patients at once through the Kidney Fast-Track Scheme [Callaghan et al., 2017].

We assume that parallel dispatches are generated by a collection of dispatch generators, indexed by $[m] \triangleq \{1, 2, \ldots, m\}$ for some $m \in \mathbb{Z}_{>0}$. Each generator is characterized by a set of probabilities that specify, for each queue position, the likelihood of receiving the item. These probabilities may depend on the current queue length, the item type, and the number of times the item has already been dispatched. Moreover, each dispatching "stream" of a generator can be history-dependent, but the streams are conditionally independent given the dispatch history. With some abuse of notation, for each $k \in [p]$, let $h^{(k)} = \left(h^{(k)}_{i,k'}\right)_{i\in[m],\, k'\in[k]} \in \mathcal{H}^{(k)} \triangleq [\bar{Q}]^{m\times k}$ denote the *dispatch history* of an item's first k dispatches across all generators, where $h^{(k)}_{i,k'} \in [\bar{Q}]$ specifies the queue position to which the k'th dispatch ($k' \le k$) was offered by generator i. A parallel dispatch policy π^{PAR} is specified by the collection of dispatch probabilities

$$\left(f^{(i,k)}_{\ell}\left(q, Q; h^{(k-1)}\right)\right)_{i\in[m],\ \ell\in[L],\ k\in[p],\ q\in[\bar{Q}],\ Q\in[\bar{Q}],\ h^{(k-1)}\in\mathcal{H}^{(k-1)}}. \tag{25}$$

For instance, Example 3 formally defines a dispatching policy in which items are offered to agents in fixed sequential batches. Since this policy is history-independent, we suppress the history in the notation.

Example 3 (Sequential Batch Dispatching)**.** Consider a sequential batch dispatching process in which an item is first offered to all agents in positions $1, \ldots, n$. In the next dispatch, the item is offered to agents in positions $n+1, \ldots, 2n$, and so on, until either an agent accepts the item or the patience limit is reached. To model this, we construct m dispatch generators, where for all $i \in [m]$ and $\ell \in [L]$, $Q \in [\bar{Q}]$, $q \in [\bar{Q}]$ and each $k \in [p]$,

$$f^{(i,k)}_{\ell}(q, Q) = \begin{cases} 1, & \text{if } q = (k-1)n + i, \\ 0, & \text{otherwise.} \end{cases}$$

Similarly to Section 3.1, the continuation payoffs $V = \left(V(q, Q)\right)_{Q\in[\bar{Q}],\, q\in[Q]}$ can still be computed in closed form under a strategy $\sigma = (\alpha, \phi)$ adopted by all agents and a parallel dispatch policy π^{PAR} specified by the dispatch probabilities in (25). In Appendix A.1, we introduce auxiliary terms that characterize the rates at which a focal agent advances, exits the queue, or observes changes in queue length. These rates depend on which agents are offered items and on their acceptance decisions. A key challenge under parallel dispatching is that multiple agents may accept the same item, creating ambiguity over which agent ultimately receives it and leaves the queue. To resolve this, we impose a tie-breaking rule, such as selecting an accepting agent uniformly at random or assigning the item to the accepting agent with the lowest queue position. Further details on the evaluation of $V(q, Q)$ and the tie-breaking mechanism are provided in Appendix A.1.

For equilibrium computation, the gradient-based Algorithm 1 still applies in this case. However, evaluating the gradient is substantially more complex (though it still admits a closed-form recursion similar to that in Appendix B.3), since the tie-breaking rule introduces an additional layer of dependence on the strategy. In the experiments below, we apply Algorithm 2 to approximate the equilibrium, motivated by empirical evidence in Appendix D.4 showing that the approximate solution is generally very close to the one produced by Algorithm 1. As discussed in Section 4.1, this approximation assumes that agents ignore the possibility of immediate state changes when taking actions.

6.1.1 Modeling Platform Competition via Parallel Dispatching

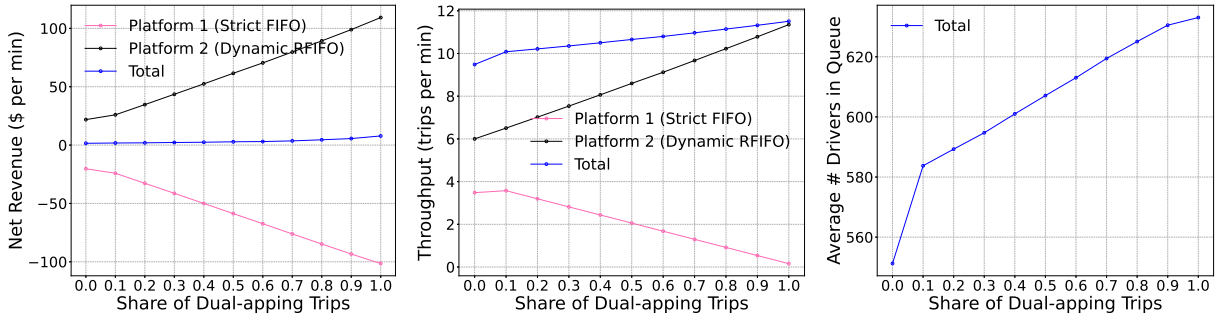
Rather than illustrating our parallel dispatching framework within a single platform or organization, this section demonstrates how it can, perhaps surprisingly, be applied to model competition

among ridesharing platforms at an airport with rider dual-apping and driver multihoming, thereby complementing the insights presented in Section 5.

Consider two ridesharing platforms, each operating under distinct dispatch policies. Platform 1 adopts Strict FIFO, while Platform 2 employs Dynamic Randomized FIFO. At an airport, drivers often multihome by joining the waiting queues of both platforms and may receive trip offers from either dispatch system. We model three types of trips: (i) those exclusive to Platform 1, dispatched under Strict FIFO; (ii) those exclusive to Platform 2, dispatched under Dynamic Randomized FIFO; and (iii) dual-apping trips, where riders request rides through both platforms. Dispatching for trips in (iii) can thus be modeled as a parallel dispatch policy, where each trip is processed by two independent dispatch generators—Strict FIFO and Dynamic Randomized FIFO. After agents indicate whether they accept or decline, a random tie-breaking rule determines which driver (and corresponding platform) ultimately receives the trip. It is important to note that, unlike the nonparallel dispatch policy in Section 2.1, where a trip is fulfilled as long as it is accepted within p trials, in the competition setting the platform that secures a driver in fewer trials gains the advantage—explicitly competing on fulfillment speed.

We apply Algorithm 2 to compute the equilibrium in the same setting as the Chicago O'Hare experiments in Section 5, varying the share of type (iii) multihoming riders from 0 to 1, under equal trip and driver arrival rates ($\mu = \lambda = 12$). The remaining market share (type (i) and (ii) riders) as well as the driver waiting costs are assumed to be evenly split between the two platforms.

Figure 5 illustrates the equilibrium performance metrics. While the aggregate net revenue across both platforms remains positive, Platform 1 consistently incurs *negative* and *declining* net revenue as the share of dual-apping trips grows. This arises because, under finite patience, lower-earning jobs are never dispatched under Strict FIFO alone. In contrast, Dynamic Randomized FIFO reallocates these jobs to later positions in the queue where they are more likely to be accepted, which lengthens the queue as more drivers join relative to Strict FIFO alone. Platform 2 benefits and its net revenue correspondingly increases. These dynamics underscore that the speed at which each platform matches riders with drivers willing to accept trips is critical for overall performance. As dual-apping trips become more prevalent, Platform 1's throughput declines due to intensified competition for the limited set of jobs that can be dispatched within riders' patience windows, whereas Platform 2's throughput steadily rises as it captures a growing share of trip requests.



(a) Net revenue. (b) Trip throughput. (c) Expected queue length.

Figure 5: Equilibrium outcomes as the share of dual-apping trips increases.

6.2 Costly Inspection

The model so far assumes that once an item is offered, the receiving agent observes enough information to decide whether or not to accept it at no additional cost. In many applications, however, an

offer first arrives as an uninformative notification. The agent must then devote attention, time, or other resources to inspect the item before learning whether it is worth accepting. You can imagine, for example, surgeons spending time examining the quality of a kidney, or airport drivers having to spend consistent energy to monitor trip offers.

We incorporate this consideration by adding an inspection decision to the offer timing. At state (q, Q), an agent who receives an offer notification inspects with probability $\delta(q, Q) \in [0, 1]$. Inspection costs $c_{\mathrm{I}} \geq 0$ and reveals the item type $\ell \in [L]$. Conditional on inspection and learning type ℓ, the agent accepts with probability $\alpha_\ell(q, Q)$. An agent who does not inspect is treated as declining the offer. Thus, the probability that the agent both inspects and accepts a type-ℓ item can be written as

$$\xi_\ell(q, Q) \triangleq \delta(q, Q)\alpha_\ell(q, Q).$$

The inspection decision is not type-dependent because the type is not observed before inspection. Consequently, the feasible strategy set at each state can be represented in terms of (ξ, δ), as

$$\mathcal{K}_{q,Q} \triangleq \{(\xi, \delta) : 0 \leq \delta \leq 1, \quad 0 \leq \xi_\ell \leq \delta \text{ for all } \ell \in [L]\}. \tag{26}$$

When $\delta(q, Q) > 0$, the acceptance probabilities are recovered as $\alpha_\ell(q, Q) = \xi_\ell(q, Q)/\delta(q, Q)$. When $\delta(q, Q) = 0$, the acceptance probabilities are irrelevant.

The continuation-payoff calculation changes in two ways. First, whenever an item is accepted with probability $\alpha_\ell(q, Q)$ in the baseline model, the inspection model uses the *effective acceptance* probability $\xi_\ell(q, Q)$. Second, inspection incurs a cost whenever the agent receives and inspects an offer, regardless of whether the agent later accepts the item or wins any tie-breaking procedure in a parallel dispatch setting. The type-0 endogenous reneging item does *not* require inspection.

For computation, we extend the approximate gradient approach in Algorithm 2 to include costly inspection. Intuitively, the new update direction considers the tradeoff between the expected earnings from always inspecting at a given state and the corresponding inspection cost that is incurred. The procedure evaluates continuation payoffs after accounting for inspection costs and, if using parallel dispatch, the probability of winning the tie-breaking procedure. It separately updates the type-0 reneging decision, jointly updates effective acceptance and inspection so that effective acceptance cannot exceed inspection, and then updates joining. Appendix A.2 provides the corresponding rate calculations, fixed-point equilibrium conditions, and equilibrium computation algorithm.

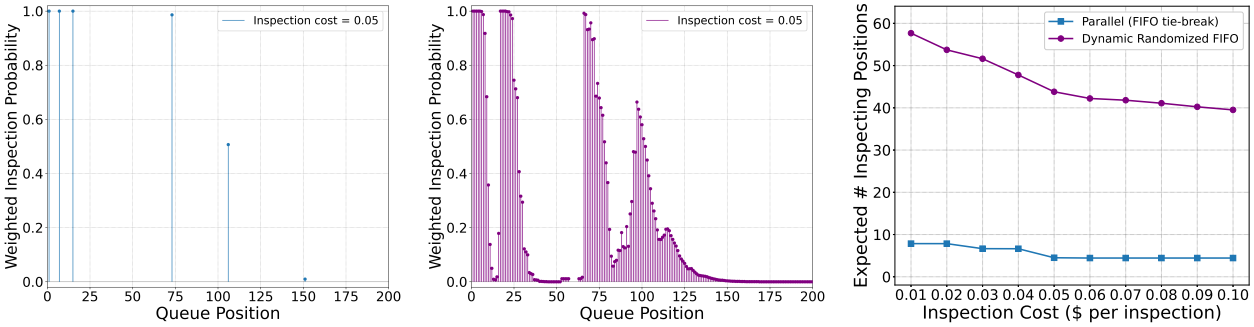
6.2.1 Varying Inspection Cost

We illustrate the inspection model using a full-queue blasting policy, under which each arriving item is announced in parallel to every agent in the queue. If multiple agents inspect and accept, the item is assigned to the accepting agent with the smallest queue position. We compare this policy with the sequential, single-recipient dynamic randomized FIFO dispatching policy. We use the inspection-aware approximate gradient projection algorithm in Algorithm 3 in Appendix A.2 to compute the approximate-equilibrium strategies and outcomes.

In Figure 6a and Figure 6b, we plot, for each queue position q, the weighted inspection probability $\sum_{Q \geq q} \pi(Q)\delta(q, Q) = \Pr(Q \geq q)\,\mathbb{E}[\delta(q, Q) \mid Q \geq q]$, where $\pi(Q)$ is the steady-state probability of queue length Q, for an inspection cost of 0.05. In other words, the weighted inspection probability at position q equals the probability that the queue length is at least q multiplied by the average inspection probability conditional on the queue length being at least q. This is meant to reflect a small inspection probability for queue lengths that are unlikely to occur. Note that $\delta(q, Q) = 0$ for infeasible states where $Q < q$.

In Figure 6c, we plot the total expected number of inspecting positions in the system for each inspection cost. Finally, in Figure 6d, Figure 6e, and Figure 6f, we plot the equilibrium net revenue, trip throughput, and expected queue length under the two dispatching policies for inspection costs ranging from 0.01 to 0.10. For all experiments we use a driver arrival rate of $\lambda = 10$ and a trip arrival rate of $\mu = 12$.

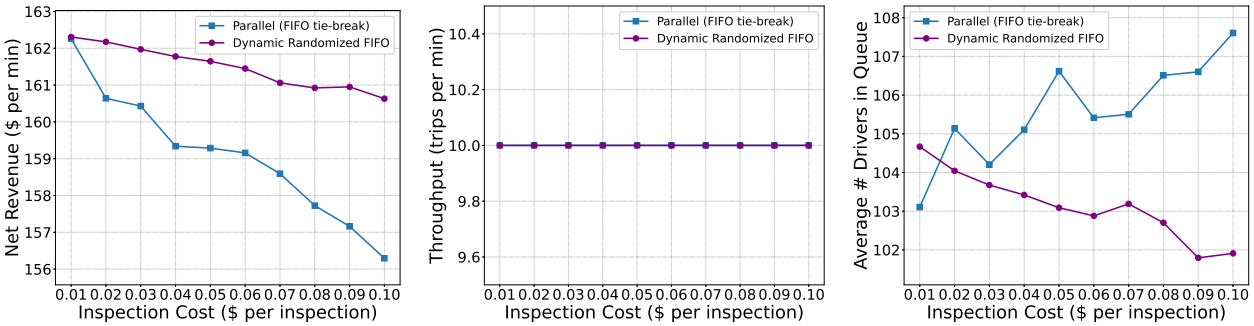
Notice that the weighted inspection profile under parallel dispatching is concentrated at relatively few positions. This example highlights a coordination problem rather than a policy recommendation: the equilibrium may place inspection only at very specific queue positions (see the relatively small number of inspection positions in Figure 6c). Such finely position-dependent inspection behavior is unlikely to arise in practice when a platform or allocation system broadcasts offers to many agents.



(a) Parallel dispatch inspection profile.

(b) Dynamic randomized FIFO inspection profile.

(c) Expected number of inspecting positions.



(d) Net revenue.

(e) Trip throughput.

(f) Expected queue length.

Figure 6: Inspection strategies and approximate-equilibrium outcomes under full-queue parallel dispatching with FIFO tie-breaking and sequential dynamic Randomized FIFO. Panels (a) and (b) use an inspection cost of \$0.05 per notification; panels (c)–(f) vary the inspection cost.

Furthermore, we see that, intuitively, the overall net revenue and expected number of agents actually inspecting decrease as the inspection cost increases for both dispatching policies. The sequential dynamic Randomized FIFO produces only a slightly higher net revenue than parallel dispatching at the same inspection cost. This comparison suggests that the approximate equilibrium under full-queue dispatching is not inherently inefficient, however it is unrealistic to achieve in practice since there are few agents who should actually be inspecting and they must do so in the right positions.

7 Concluding Remarks

We present an analytical solver for a broad and practically significant class of dynamic queueing games in which agents make entry, acceptance, and reneging decisions, addressing key challenges in managing strategic behavior in applications such as rideshare dispatching and deceased-donor organ allocation. The solver computes equilibrium strategies with closed-form updates under general dispatch rules that capture both queue length and history dependence. Through extensive simulations, we validate its accuracy by recovering known theoretical results and demonstrating its ability to analyze complex yet operationally critical policies—such as the Dynamic Randomized FIFO policy—that are previously unexplored and analytically intractable. We further illustrate the solver's versatility by evaluating the robustness of various policies under model misspecification and extending it to study platform competition and costly inspection strategies through parallel dispatching. Together, these results position the solver as a powerful tool for market designers, enabling the exploration and optimization of sophisticated policies tailored to real-world applications.

References

Ilan Adler. The equivalence of linear programs and zero-sum games. *International Journal of Game Theory*, 42:165–177, 2013.

Nikhil Agarwal, Itai Ashlagi, Michael A Rees, Paulo Somaini, and Daniel Waldinger. Equilibrium allocations under alternative waitlist designs: Evidence from deceased donor kidneys. *Econometrica*, 89(1):37–76, 2021.

Itai Ashlagi, Ravi Jagadeesan, and Pengyu Qian. Congested waiting lists and organ allocation. *Available at SSRN 5338317*, 2025.

Francis Bloch and David Cantala. Dynamic assignment of objects to queuing agents. *American Economic Journal: Microeconomics*, 9(1):88–122, 2017.

Chris J Callaghan, Lisa Mumford, Laura Pankhurst, Richard J Baker, J Andrew Bradley, and Christopher JE Watson. Early outcomes of the new uk deceased donor kidney fast-track offering scheme. *Transplantation*, 101(12):2888–2897, 2017.

Francisco Castro, Hongyao Ma, Hamid Nazerzadeh, and Chiwei Yan. Randomized FIFO mechanisms, 2021. URL https://arxiv.org/abs/2111.10706.

Yeon-Koo Che and Olivier Tercieux. Optimal queue design. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 312–313, 2021.

Israel David and Uri Yechiali. A time-dependent stopping problem with application to live organ transplants. *Operations Research*, 33(3):491–504, 1985.

F. Facchinei and J.S. Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Springer Series in Operations Research and Financial Engineering. Springer New York, 2007. ISBN 9780387218144. URL https://books.google.com/books?id=qekMDFCRYEUC.

Guillermo Gallego and Huseyin Topaloglu. *Revenue Management and Pricing Analytics*, volume 209. Springer, 2019.

Refael Hassin. On the optimality of first come last served queues. *Econometrica*, 53(1), 1985.

Refael Hassin. *Rational queueing*. CRC press, 2016.

Refael Hassin and Moshe Haviv. *To queue or not to queue: Equilibrium behavior in queueing systems*, volume 59. Springer Science & Business Media, 2003.

Moshe Haviv and Binyamin Oz. Regulating an observable M/M/1 queue. *Operations Research Letters*, 44(2):196–198, 2016.

Jacob D Leshno. Dynamic matching in overloaded waiting lists. *American Economic Review*, 112 (12):3876–3910, 2022.

Medscape. Organ preservation, 2023. URL https://emedicine.medscape.com/article/431140-overview. Accessed: 2025-02-08.

Paul Milgrom and John Roberts. Rationalizability, learning, and equilibrium in games with strategic complementarities. *Econometrica: Journal of the Econometric Society*, pages 1255–1277, 1990.

Sumit Mohan, Mariana C Chiles, Rachel E Patzer, Stephen O Pastan, S Ali Husain, Dustin J Carpenter, Geoffrey K Dube, R John Crew, Lloyd E Ratner, and David J Cohen. Factors leading to the discard of deceased donor kidneys in the united states. *Kidney international*, 94 (1):187–198, 2018.

Dov Monderer and Lloyd S Shapley. Potential games. *Games and Economic Behavior*, 14(1): 124–143, 1996.

Pinhas Naor. The regulation of queue size by levying tolls. *Econometrica*, pages 15–24, 1969.

John Nash. Non-cooperative games. *Annals of Mathematics*, 54(2):286–295, 1951. doi: 10.2307/1969529. URL https://doi.org/10.2307/1969529. Accessed 11 Feb. 2025.

Afshin Nikzad and Philipp Strack. Equity and efficiency in dynamic matching: Extreme waitlist policies. *Management Science*, 70(8):5187–5207, 2024.

M.L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 2005. ISBN 9780471727828. URL https://books.google.com/books?id=Y-gmAQAAIAAJ.

Liron Ravner and Ran I Snitkovsky. Stochastic approximation of symmetric Nash equilibria in queueing games. *Operations Research*, 72(6):2698–2725, 2024.

Sheldon M Ross. *Stochastic Processes*. John Wiley & Sons, 1995.

San Francisco Chronicle. Uber may stop letting drivers see destinations in California after losing court battle, 2021. URL https://www.sfchronicle.com/business/article/Uber-may-stop-letting-drivers-see-destinations-16078491.php. Accessed: 2025-02-08.

Marco Scarsini and Eran Shmaya. Optimal queueing regimes. In *Proceedings of the 25th ACM Conference on Economics and Computation*, EC '24, page 580, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400707049. doi: 10.1145/3670865.3673580. URL https://doi.org/10.1145/3670865.3673580.

James Schummer. Influencing waiting lists. *Journal of Economic Theory*, 195:105263, 2021.

Lloyd S Shapley. Stochastic games. *Proceedings of the National Academy of Sciences*, 39(10): 1095–1100, 1953.

Peng Shi and Junxiong Yin. Eliminating waste in cadaveric organ allocation. *USC Marshall School of Business Research Paper Sponsored by iORB*, 2022.

Mike J Smith. The existence and calculation of traffic equilibria. *Transportation Research Part B: Methodological*, 17(4):291–303, 1983.

Xuanming Su and Stefanos Zenios. Patient choice in kidney allocation: The role of the queueing discipline. *Manufacturing & Service Operations Management*, 6(4):280–301, 2004.

Donald M Topkis. *Supermodularity and complementarity*. Princeton University Press, 1998.

Bernhard von Stengel. Zero-sum games and linear programming duality. *Mathematics of Operations Research*, 49(2):1091–1108, 2024.

Nicholas L Wood, Douglas N VanDerwerken, Allan B Massie, Dorry L Segev, Jon J Snyder, and Sommer E Gentry. Diagnosing the recent decrease in utilization of deceased donor kidneys. *Transplantation*, 109(3):496–503, 2025.

David D. Yao. S-modular games, with queueing applications. *Queueing systems*, 21(3-4):449–475, 1995. ISSN 0257-0130.

Uri Yechiali. On optimal balking rules and toll charges in the GI/M/1 queuing process. *Operations Research*, 19(2):349–370, 1971.

Appendix

Appendix A presents the technical details and proofs for the parallel-dispatching and costly-inspection extensions in Section 6. Appendix B provides proofs and auxiliary results for the baseline model. Additional examples and discussions are included in Appendix C. Finally, we include additional numerical results and simulation details in Appendix D.

A Extensions

A.1 Parallel Dispatching

In this section, we provide the technical details required to adapt our solver to parallel dispatching. The procedure for computing the continuation payoffs $V = (V(q,Q))_{Q\in[\bar{Q}],\, q\in[Q]}$ under the strategy $\sigma = (\alpha, \phi)$ adopted by all agents and the dispatch policy π^{PAR} closely resembles the case where items are offered to a single agent at a time (Section 3.1). The key distinction is that, in parallel dispatching, multiple agents may accept the item within the same dispatch, in which case the platform must determine which agent ultimately receives it. To capture this feature, in addition to defining the corresponding rate terms as in (7)–(10), we introduce a tie-breaking rule.

Let $d_\ell^{(i,k)}(q,Q)$ denote the probability that the agent at position (q,Q) wins the dispatch and receives the item through generator i, conditional on the agent accepting the item during the k^{th} dispatch. Intuitively, this term represents the likelihood that an agent actually obtains the item once they choose to accept it on a given dispatch. While the definition is broad, we provide exact formulations under two natural tie-breaking policies. First, Example 4 specifies a FIFO-based rule, in which the item is awarded to the accepting agent with the smallest queue position—equivalently, the probability that no agent in front accepts the item. Second, Example 5 specifies a random tie-breaking rule, in which one of the accepting agents is selected uniformly at random.

Example 4 (FIFO Tie-Breaking)**.** The probability that an item ℓ, dispatched to an agent in state (q,Q) during the k^{th} dispatch attempt by generator i, is not accepted by any higher-priority agent across all other generators is

$$d_\ell^{(i,k)}(q,Q) \;=\; \prod_{j\in[m],j\neq i} \left(1 - \sum_{q'=1}^{q-1} f_\ell^{(j,k)}(q',Q;h^{(k-1)})\,\alpha_\ell(q',Q)\right),$$

$\forall i \in [m]$, $\forall \ell \in [L]$, $\forall k \in [p]$, $\forall q \in [Q]$, $\forall Q \in [\bar{Q}]$, $\forall h^{(k-1)} \in \mathcal{H}^{(k-1)}$.

Example 5 (Random Tie-Breaking)**.** Let $A_\ell^{(k)}(Q)$ denote the total number of agents, across all dispatch generators, who accept item ℓ during the k^{th} dispatch. This random variable follows a Poisson binomial distribution with success probabilities $p_i = \sum_{q'=1}^{Q} f_\ell^{(i,k)}(q',Q;h^{(k-1)})\,\alpha_\ell(q',Q)$ for $i \in [m]$. Similarly, let $A_\ell^{-i,(k)}(Q)$ denote the total number of agents who accept item ℓ during the k^{th} dispatch across all dispatch generators except the focal generator i. With random tie-breaking,

$$\begin{aligned}
&d_\ell^{(i,k)}(q,Q)\\
&= \mathbb{E}\left[\frac{1}{1+A_\ell^{-i,(k)}(Q)}\right]\\
&= \mathbb{E}\left[\int_0^1 t^{A_\ell^{-i,(k)}(Q)}\,dt\right] \qquad\qquad (\text{since } \tfrac{1}{1+a} = \textstyle\int_0^1 t^a\,dt \text{ for } a \geq 0)
\end{aligned}$$

$$= \int_0^1 \mathbb{E}\Big[t^{A_\ell^{-i,(k)}(Q)}\Big]\, dt \qquad \text{(Tonelli's Theorem due to non-negativity)}$$

$$= \int_0^1 \prod_{j\in[m],j\neq i} \mathbb{E}[t^{X_j}]\, dt \qquad \text{(independence, } A_\ell^{-i,(k)}(Q) = \sum_{j\neq i} X_j \text{ where } X_j \sim \text{Bernoulli}(p_j))$$

$$= \int_0^1 \prod_{j\in[m],j\neq i} \big((1-p_j) + p_j t\big)\, dt, \quad \forall i \in [m],\, \forall \ell \in [L],\, \forall k \in [p],\, \forall q \in [Q],\, \forall Q \in [\bar{Q}].$$

Next, we use $g_\ell^{(i,k)}(q,Q)$ to denote the probability for position $q \in [\bar{Q}]$ to receive an item of type $\ell \in \mathcal{L}$ through dispatch generator i during the k^{th} dispatch, when the queue length is $Q \in [\bar{Q}]$. For each independent generator i we start with:

$$g_\ell^{(i,1)}(q,Q) = f_\ell^{(i,1)}(q,Q).$$

Since an item must now survive all dispatching streams from the previous $k-1$ dispatches, we can state the following. For any $i \in [m]$, $k \geq 2$, $Q \in [\bar{Q}]$, $q \in [Q]$, and $\ell \in \mathcal{L}$, we have:

$$g_\ell^{(i,k)}(q,Q) = \sum_{h^{(k-1)} \in \mathcal{H}^{(k-1)}} f_\ell^{(i,k)}(q,Q;h^{(k-1)}) \left[\prod_{j\in[m]} \Bigg(\underbrace{\prod_{k'=1}^{k-1} f_\ell^{(j,k')}\big(h_{j,k'}^{(k-1)}, Q; h_{1:m,1:k'-1}^{(k-1)}\big)\big(1-\alpha_\ell(h_{j,k'}^{(k-1)},Q)\big)}_{\text{declined at each position across all generators in } h^{(k-1)}} \Bigg) \right].$$

When the policy is history-independent, this reduces to:

$$g_\ell^{(i,k)}(q,Q) = f_\ell^{(i,k)}(q,Q) \prod_{j\in[m]} \left(\sum_{q'\in[Q]} g_\ell^{(j,k-1)}(q',Q)(1-\alpha_\ell(q',Q)) \right).$$

By combining the tie-breaking rule with the probability of being offered a job defined above, we can express the rate at which an agent in state (q,Q) receives items of type ℓ from dispatch generator i as follows:

$$\tau_\ell^i(q,Q) = \mu_\ell \sum_{k\in[p]} g_\ell^{(i,k)}(q,Q) d_\ell^{(i,k)}(q,Q).$$

The total rate experienced by an agent in state (q,Q), denoted previously by $\nu_\ell(q,Q)$, is given by the sum of these rates over all independent dispatch streams.

$$\tau_\ell(q,Q) = \sum_{i\in[m]} \tau_\ell^i(q,Q).$$

Additionally, we adjust the following rate terms for all $Q \in [\bar{Q}]$ and $q \in [Q]$, corresponding to events occurring at different positions in the queue, analogous to (6)–(10):

$$r^0(q,Q) = \nu_0(q,Q)\alpha_0(q,Q) + \sum_{\ell\in[L]} \tau_\ell(q,Q)\alpha_\ell(q,Q) + \theta,$$

$$r^+(q,Q) = \sum_{q'=1}^{q-1} r^0(q',Q),$$

$$r^-(q,Q) = \sum_{q'=q+1}^{Q} r^0(q',Q),$$
$$r(Q) = \sum_{q=1}^{Q} r^0(q,Q) + \lambda\phi(Q) = r^0(q,Q) + r^+(q,Q) + r^-(q,Q) + \lambda\phi(Q).$$

Notice that $r^0(q,Q)$, the rate at which agents in state (q,Q) receive and accept items or renege, uses $\tau_\ell(q,Q)$ for ordinary items to account for tie-breaking and $\nu_0(q,Q)$ for the type-0 reneging option. Once these rate terms are computed, the evaluation of V proceeds as in Lemma 1.

A.2 Costly Inspection

This section records the technical details associated with the costly inspection extension in Section 6.2. The strategy can be described as (α,ϕ,δ), where $\delta(q,Q)$ is the probability of inspecting an uninformative notification and $\alpha_\ell(q,Q)$ is the probability of accepting conditional on inspection and learning the type $\ell \in [L]$. For computation, however, we use the effective acceptance probability $\xi_\ell(q,Q) = \delta(q,Q)\alpha_\ell(q,Q)$ and the statewise feasible set $\mathcal{K}_{q,Q}$ in (26).

Going forward, to generalize to both single (non-parallel) and parallel dispatching, recall that $\tau_\ell(q,Q)$ denotes the rate at which the focal agent receives a type-ℓ item conditional on accepting it, after any tie-breaking. Under an ordinary single-recipient (non-parallel) policy, we have $\tau_\ell(q,Q) = \nu_\ell(q,Q)$. We can now redefine the following rate terms:

$$\omega^{\mathrm{I}}(q,Q) \triangleq \sum_{\ell\in[L]} \tau_\ell(q,Q)\xi_\ell(q,Q)w_\ell,$$
$$\kappa_{\mathrm{I}}(q,Q) \triangleq c_{\mathrm{I}}\delta(q,Q) \sum_{\ell\in[L]} \nu_\ell(q,Q),$$
$$r^0_{\mathrm{I}}(q,Q) \triangleq \theta + \nu_0(q,Q)\alpha_0(q,Q) + \sum_{\ell\in[L]} \tau_\ell(q,Q)\xi_\ell(q,Q).$$

Inspection costs depend on $\nu_\ell(q,Q)$ because an agent pays the cost after receiving and inspecting a notification, even if another accepting agent wins the tie-breaking procedure. Earnings and departures instead depend on $\tau_\ell(q,Q)$ because they occur only when the focal agent receives the item. Rejected inspections and lost tie-breaking opportunities leave the focal agent in the same queue state but still generate inspection costs while the agent remains there.

The term $\nu_0(q,Q)\alpha_0(q,Q)$ accounts for endogenous reneging, which does not require inspection or tie-breaking. Define $r^+_{\mathrm{I}}(q,Q)$, $r^-_{\mathrm{I}}(q,Q)$, and $r_{\mathrm{I}}(Q)$ analogously to (8)–(10). For types that require inspection, these calculations replace $\alpha_\ell(q,Q)$ with $\xi_\ell(q,Q)$ and use the tie-breaking-adjusted offer rate. The continuation payoff then satisfies

$$\begin{aligned} V(q,Q) = \frac{\omega^{\mathrm{I}}(q,Q) - c_w - \kappa_{\mathrm{I}}(q,Q)}{r_{\mathrm{I}}(Q)} + \frac{r^+_{\mathrm{I}}(q,Q)}{r_{\mathrm{I}}(Q)}V(q-1,Q-1) \\ + \frac{r^-_{\mathrm{I}}(q,Q)}{r_{\mathrm{I}}(Q)}V(q,Q-1) + \frac{\lambda\phi(Q)}{r_{\mathrm{I}}(Q)}V(q,Q+1). \end{aligned} \tag{27}$$

The boundary conditions are unchanged from (11).

At each state, the focal agent's best-response problem is to choose how often to inspect and which types to accept after inspection. Increasing $\delta(q,Q)$ expands the feasible range of every

$\xi_\ell(q,Q)$, while increasing $\xi_\ell(q,Q)$ raises the probability of accepting type ℓ. The set $\mathcal{K}_{q,Q}$ captures this link between the inspection and acceptance decisions.

For computation, following the same approximation used by Algorithm 2, we define the approximate payoff gradient by

$$\widehat{D}^{\xi}_{\ell,q,Q} \triangleq \tau_\ell(q,Q)\big(w_\ell - V(q,Q)\big). \tag{28}$$

This gradient holds the continuation payoff fixed when measuring the gain from increasing effective acceptance. It disregards the immediate effect of a rejection on other agents' departures and therefore avoids computing how transition rates change with the focal agent's effective acceptance probability. The cost of one additional unit of inspection probability is

$$C_{\mathrm{I}}(q,Q) \triangleq c_{\mathrm{I}} \sum_{\ell\in[L]} \nu_\ell(q,Q).$$

The resulting approximate best-response problem is

$$\max_{\substack{\alpha_0(q,Q)\in[0,1]\\(\xi(q,Q),\delta(q,Q))\in\mathcal{K}_{q,Q}}} \left\{ \nu_0(q,Q)\alpha_0(q,Q)\big(w_0 - V(q,Q)\big) + \sum_{\ell\in[L]} \widehat{D}^{\xi}_{\ell,q,Q}\xi_\ell(q,Q) - C_{\mathrm{I}}(q,Q)\delta(q,Q) \right\}.$$

As in Algorithm 2, the best-response problem is exact for simple dispatch policies and serves as an approximation for general policies.

Inspection-Aware Approximate Equilibrium. Write $\xi_{[L]}(q,Q) = (\xi_\ell(q,Q))_{\ell\in[L]}$. We call a strategy an inspection-aware approximate equilibrium when it satisfies the following fixed point equations:

$$\begin{aligned} \alpha_0^*(q,Q) &= P_{[0,1]}\left[\alpha_0^*(q,Q) + w_0 - V^*(q,Q)\right], \\ \begin{pmatrix} \xi^*_{[L]}(q,Q) \\ \delta^*(q,Q) \end{pmatrix} &= \mathsf{P}_{\mathcal{K}_{q,Q}}\left[\begin{pmatrix} \xi^*_{[L]}(q,Q) \\ \delta^*(q,Q) \end{pmatrix} + \begin{pmatrix} (\widehat{D}^{\xi,*}_{\ell,q,Q})_{\ell\in[L]} \\ -C^*_{\mathrm{I}}(q,Q) \end{pmatrix} \right]. \end{aligned} \tag{29}$$

These conditions must hold at every state together with the unchanged joining condition (15). This fixed-point equation combines the gradient direction in (28) with the joint feasible set $\mathcal{K}_{q,Q}$. For a general dispatch policy, it is an approximate equilibrium and is not asserted to satisfy the exact statewise best-response conditions.

The unconstrained inspection coordinate is decreasing because its direct gradient contains only the inspection cost. However, profitable item types generate positive proposed changes in their effective acceptance probabilities. If a proposed $\xi_\ell(q,Q)$ exceeds $\delta(q,Q)$, it violates the constraint $\xi_\ell(q,Q) \le \delta(q,Q)$. The joint projection finds the closest feasible pair by potentially both lowering the proposed $\xi_\ell(q,Q)$ and raising $\delta(q,Q)$. Thus, profitable acceptance opportunities affect the inspection decision indirectly through the feasibility constraint.

Loss Functions. Conditional on a given inspection probability $\delta(q,Q)$, feasibility restricts each $\xi_\ell(q,Q)$ to the interval $[0,\delta(q,Q)]$. The effective-acceptance loss used to construct the stopping rule in Algorithm 3 is defined at state (q,Q) as

$$\mathrm{Loss}_\xi(q,Q) \triangleq \sum_{\ell\in[L]} \Big[\big(\delta(q,Q) - \xi_\ell(q,Q)\big)\big(\widehat{D}^{\xi}_{\ell,q,Q}\big)^+ + \xi_\ell(q,Q)\big(\widehat{D}^{\xi}_{\ell,q,Q}\big)^- \Big]. \tag{30}$$

When $\widehat{D}^{\xi}_{\ell,q,Q} > 0$, the payoff of the approximate best-response problem increases with $\xi_\ell(q,Q)$ and the best response is $\xi_\ell(q,Q) = \delta(q,Q)$. The first term in (30) measures the payoff forgone by

leaving the difference $\delta(q,Q)-\xi_\ell(q,Q)$ unused. When $\widehat{D}^{\xi}_{\ell,q,Q}<0$, the approximate best-response problem instead sets $\xi_\ell(q,Q)=0$, and the second term measures the loss from the current positive effective-acceptance probability.

Increasing $\delta(q,Q)$ relaxes the upper bound on every profitable $\xi_\ell(q,Q)$ by the same amount. After optimizing effective acceptance conditional on inspection, the benefit from one additional unit of inspection probability is $\sum_{\ell\in[L]}(\widehat{D}^{\xi}_{\ell,q,Q})^+$, while its cost is $C_{\mathrm{I}}(q,Q)$. The inspection loss is therefore

$$\mathrm{Loss}_\delta(q,Q)\triangleq\big(1-\delta(q,Q)\big)\left(\sum_{\ell\in[L]}\big(\widehat{D}^{\xi}_{\ell,q,Q}\big)^+-C_{\mathrm{I}}(q,Q)\right)^+ +\delta(q,Q)\left(\sum_{\ell\in[L]}\big(\widehat{D}^{\xi}_{\ell,q,Q}\big)^+-C_{\mathrm{I}}(q,Q)\right)^- .$$

The type-0 loss at state (q,Q) is

$$\mathrm{Loss}_0(q,Q)\triangleq\big(1-\alpha_0(q,Q)\big)\big(w_0-V(q,Q)\big)^+ +\alpha_0(q,Q)\big(w_0-V(q,Q)\big)^-. \tag{31}$$

Adding the type-0 loss in (31) to the effective-acceptance and inspection losses, define the aggregate loss by

$$\mathrm{Loss}_{\mathrm{I}}(\alpha_0,\xi,\delta,V)\triangleq\max_{Q\in[\bar{Q}],\,q\in[Q]}\left\{\mathrm{Loss}_\xi(q,Q)+\mathrm{Loss}_\delta(q,Q)+\mathrm{Loss}_0(q,Q)\right\}.$$

Using the unchanged joining loss in (24), the inspection-aware algorithm terminates when

$$\mathrm{Loss}_{\mathrm{I}}\big(\alpha_0^{(t)},\xi^{(t)},\delta^{(t)},V^{(t)}\big)\le\varepsilon \qquad\text{and}\qquad \mathrm{Loss}\big(\phi^{(t)},V^{(t)}\big)\le\varepsilon.$$

For a general dispatch policy, this stopping rule measures convergence to the inspection-aware approximate equilibrium in (29).

Algorithm 3 Inspection-Aware Approximate Gradient Projection Algorithm

Input: Model parameters, initial strategy $\sigma^{\mathrm{I},(0)}=(\alpha_0^{(0)},\xi^{(0)},\delta^{(0)},\phi^{(0)})$, stepsize $(\gamma^{(t)})_{t=0}^{\infty}$, termination tolerance $\varepsilon>0$, maximum iterations $T\in\mathbb{Z}_{>0}$.
Output: Strategy σ^{I}.
Initialize $t\leftarrow 0$.
while $t<T$ **do**
  Solve $V^{(t)}$ corresponding to $\sigma^{\mathrm{I},(t)}$ using (27).
  Compute $\widehat{D}^{\xi,(t)}_{\ell,q,Q}$ and $C_{\mathrm{I}}^{(t)}(q,Q)$ for all $Q\in[\bar{Q}]$, $q\in[Q]$, and $\ell\in[L]$ using the definitions above.
  if $\mathrm{Loss}_{\mathrm{I}}(\alpha_0^{(t)},\xi^{(t)},\delta^{(t)},V^{(t)})\le\varepsilon$ **and** $\mathrm{Loss}(\phi^{(t)},V^{(t)})\le\varepsilon$ **then**
    break
  end if
  $\alpha_0^{(t+1)}(q,Q)\leftarrow P_{[0,1]}\left[\alpha_0^{(t)}(q,Q)+\gamma^{(t)}\big(w_0-V^{(t)}(q,Q)\big)\right]$ for all $Q\in[\bar{Q}]$ and $q\in[Q]$.
  $\begin{pmatrix}\xi_{[L]}^{(t+1)}(q,Q)\\ \delta^{(t+1)}(q,Q)\end{pmatrix}\leftarrow \mathsf{P}_{\mathcal{K}_{q,Q}}\left[\begin{pmatrix}\xi_{[L]}^{(t)}(q,Q)\\ \delta^{(t)}(q,Q)\end{pmatrix}+\gamma^{(t)}\begin{pmatrix}(\widehat{D}^{\xi,(t)}_{\ell,q,Q})_{\ell\in[L]}\\ -C_{\mathrm{I}}^{(t)}(q,Q)\end{pmatrix}\right]$ for all $Q\in[\bar{Q}]$ and $q\in[Q]$.
  $\phi^{(t+1)}(Q-1)\leftarrow P_{[0,1]}\left[\phi^{(t)}(Q-1)+\gamma^{(t)}V^{(t)}(Q,Q)\right]$ for all $Q\in[\bar{Q}]$.
  Increment $t\leftarrow t+1$.
end while
return $\sigma^{\mathrm{I},(t)}$.

B Proofs

Contents

B.1 Proof of Lemma 1 . . . 35
B.2 Proof of Lemma 2 . . . 38
B.3 Closed-form Expressions for $\partial H_{q,Q}(\alpha; V)/\partial \alpha_\ell(q,Q)$ . . . 40
B.4 Proof of Theorem 1 . . . 41
B.5 Monotonicity of $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ . . . 41
B.6 Proof of Corollary 1 . . . 42
B.7 Proof of Theorem 2 . . . 43
B.8 First-Best Outcome . . . 46
B.9 Deriving the Expectation and Variance of Agent Payoffs . . . 47

B.1 Proof of Lemma 1

The proof of the lemma uses the following notation to represent the earning rate obtained by an agent at position $q \in [Q]$ when the queue length is $Q \in [\bar{Q}]$:

$$\omega(q,Q) = \sum_{\ell \in \mathcal{L}} \nu_\ell(q,Q)\alpha_\ell(q,Q)w_\ell.$$

Here, the rates $(\nu_\ell(q,Q))_{\ell\in\mathcal{L},Q\in[\bar{Q}],q\in[Q]}$, as defined in (6), correspond to the rates at which an agent at (q,Q) receives items of each type. Note that $\omega(q,Q)$ also accounts for the possibility of accepting a type 0 item (i.e., endogenous reneging), as discussed in Section 2.2.

Lemma 1 (Continuation Payoff Evaluation)**.** Under any dispatch policy, given any strategy adopted by all agents, the continuation payoff V can be computed via recursion

$$\begin{aligned} V(q,\bar{Q}) &= a_{(q,\bar{Q})} + c_{(q,\bar{Q})}V(q-1,\bar{Q}-1), && \forall q \in [\bar{Q}], \\ V(q,Q) &= a_{(q,Q)} + b_{(q,Q)}V(q,Q+1) + c_{(q,Q)}V(q-1,Q-1), && \forall Q \in [\bar{Q}-1],\ \forall q \in [Q], \end{aligned}$$

where the coefficients $\left(a_{(q,Q)}, b_{(q,Q)}, c_{(q,Q)}\right)_{Q\in[\bar{Q}],q\in[Q]}$, depend on the strategy and the dispatch policy, and can be computed in closed forms.

Proof. Deriving a closed form for the continuation payoff $V(q,Q)$ stems from the important observation that *it is not possible for an agent to move backwards in the queue*, which suggests that we can solve $V(q,Q)$ in a recursive manner from the head $(q=1)$ to the tail $(q=Q)$ of the queue.

Head of the queue $q=1$. We start from the agent at the head of the queue $q=1$. The only possible events include the agent accepts an item, an agent behind $q=1$ left the queue, or a new agent joins the queue. Equation (11) can therefore be simplified, and the agent's continuation payoffs at different queue lengths must satisfy:

$$\begin{aligned} V(1,1) &= \left(\frac{\omega(1,1)}{r(1)} - \frac{c_w}{r(1)}\right) + \frac{\phi(1)\lambda}{r(1)} \cdot V(1,2), \\ V(1,2) &= \left(\frac{\omega(1,2)}{r(2)} - \frac{c_w}{r(2)}\right) + \frac{\phi(2)\lambda}{r(2)} \cdot V(1,3) + \frac{r^-(1,2)}{r(2)} \cdot V(1,1), \\ &\vdots \end{aligned}$$

$$V(1,\bar{Q}-1)=\left(\frac{\omega(1,\bar{Q}-1)}{r(\bar{Q}-1)}-\frac{c_w}{r(\bar{Q}-1)}\right)+\frac{\phi(\bar{Q}-1)\lambda}{r(\bar{Q}-1)}\cdot V(1,\bar{Q})+\frac{r^-(1,\bar{Q}-1)}{r(\bar{Q}-1)}\cdot V(1,\bar{Q}-2),$$
$$V(1,\bar{Q})=\left(\frac{\omega(1,\bar{Q})}{r(\bar{Q})}-\frac{c_w}{r(\bar{Q})}\right)+\frac{r^-(1,\bar{Q})}{r(\bar{Q})}\cdot V(1,\bar{Q}-1).$$

This is roughly a birth-death process, with the option to exit at every state. We can verify that the above system of linear equations has the following closed-form solution:

$$V(1,\bar{Q})=a_{(1,\bar{Q})},$$
$$V(1,Q)=a_{(1,Q)}+b_{(1,Q)}V(1,Q+1),\quad \forall Q\in[\bar{Q}-1],$$

where the coefficients $(a_{1,Q},b_{1,Q})_{Q\in[\bar{Q}]}$ are of the form

$$a_{(1,1)}=\frac{\omega(1,1)-c_w}{r(1)},$$
$$b_{(1,1)}=\frac{\phi(1)\lambda}{r(1)},$$
$$a_{(1,Q)}=\frac{(\omega(1,Q)-c_w)+a_{(1,Q-1)}r^-(1,Q)}{r(Q)-r^-(1,Q)b_{(1,Q-1)}},\qquad \forall Q>1,$$
$$b_{(1,Q)}=\frac{\phi(Q)\lambda}{r(Q)-r^-(1,Q)b_{(1,Q-1)}},\qquad \forall Q>1.$$

When the exogenous renege rate $\theta>0$, we know from (7)–(10) that the rate $r(Q)$ is strictly positive for all $Q\in[\bar{Q}]$, and that $r(Q)>r^-(q,Q)$ holds for any $q\in[Q]$. As a result, the coefficients $(a_{1,Q},b_{1,Q})_{Q\in[\bar{Q}]}$ are well defined if $b_{1,Q}<1$ for all $Q\in[\bar{Q}]$, which we establish later in Claim 1. This completes the derivation of $V(1,Q)$ for all $Q\in[\bar{Q}]$.

Any queue position $q>1$. We now proceed to the computation of $(V(q,Q))_{Q\in\{q,\ldots,\bar{Q}\}}$ for some $q>1$, assuming that $(V(q',Q))_{q'<q,Q\in\{q',\ldots,\bar{Q}\}}$ have already been derived. The Bellman equation (11) for this q and all $Q\geq q$ are given by:

$$V(q,q)=\left(\frac{\omega(q,q)}{r(q)}+\frac{r^+(q,q)\cdot V(q-1,q-1)}{r(q)}-\frac{c_w}{r(q)}\right)+\frac{\phi(q)\lambda}{r(q)}\cdot V(1,q+1),$$
$$V(q,q+1)=\left(\frac{\omega(q,q+1)}{r(q+1)}+\frac{r^+(q,q+1)\cdot V(q-1,q)}{r(q+1)}-\frac{c_w}{r(q+1)}\right)+\frac{\phi(q+1)\lambda}{r(q+1)}\cdot V(q,q+2)$$
$$+\frac{r^-(q,q+1)}{r(q+1)}\cdot V(q,q),$$
$$\vdots$$
$$V(q,\bar{Q}-1)=\left(\frac{\omega(q,\bar{Q}-1)}{r(\bar{Q}-1)}+\frac{r^+(q,\bar{Q}-1)\cdot V(q-1,\bar{Q}-2)}{r(\bar{Q}-1)}-\frac{c_w}{r(\bar{Q}-1)}\right)+\frac{\phi(\bar{Q}-1)\lambda}{r(\bar{Q}-1)}\cdot V(q,\bar{Q})$$
$$+\frac{r^-(q,\bar{Q}-1)}{r(\bar{Q}-1)}\cdot V(q,\bar{Q}-2),$$
$$V(q,\bar{Q})=\left(\frac{\omega(q,\bar{Q})}{r(\bar{Q})}+\frac{r^+(q,\bar{Q})\cdot V(q-1,\bar{Q}-1)}{r(\bar{Q})}-\frac{c_w}{r(\bar{Q})}\right)+\frac{r^-(q,\bar{Q})}{r(\bar{Q})}\cdot V(q,\bar{Q}-1).$$

Then, similar to the $q=1$ case, we can solve the linear system, and write an agent's payoff $V(q,Q)$ at state (q,Q) as follows:

$$V(q,\bar{Q})=a_{(q,\bar{Q})}+c_{(q,\bar{Q})}V(q-1,\bar{Q}-1),$$

$$V(q,Q) = a_{(q,Q)} + b_{(q,Q)}V(q,Q+1) + c_{(q,Q)}V(q-1,Q-1), \qquad \forall Q \in \{q, q+1, \dots, \bar{Q}-1\}.$$

Note that in comparison to the $q = 1$ case, we now have terms corresponding to the possibility of the agent moving forward to position $q-1$ in the queue. Similarly to before, we can verify that the initial parameters for (q,q) are given by:

$$\begin{aligned} a_{(q,q)} &= \frac{\omega(q,q) - c_w}{r(q)}, \\ b_{(q,q)} &= \frac{\phi(q)\lambda}{r(q)}, \\ c_{(q,q)} &= \frac{r^+(q,q)}{r(q)}. \end{aligned}$$

For every subsequent $Q > q$, the parameters are given recursively as:

$$\begin{aligned} a_{(q,Q)} &= \frac{[\omega(q,Q) - c_w] + a_{(q,Q-1)}r^-(q,Q)}{r(Q) - r^-(q,Q)b_{(q,Q-1)}}, \\ b_{(q,Q)} &= \frac{\phi(Q)\lambda}{r(Q) - r^-(q,Q)b_{(q,Q-1)}}, \\ c_{(q,Q)} &= \frac{r^+(q,Q)}{r(Q) - r^-(q,Q)b_{(q,Q-1)}}. \end{aligned}$$

Note that with boundary conditions $V(0,\cdot) = V(\cdot,0) = V(\cdot,\bar{Q}+1) = 0$, the expressions for $q > 1$ also applies for the $q = 0$ case, and these gives us the closed-form solutions as stated in the lemma. Furthermore, when $Q = q$, the expression for the coefficients simplifies due to the absence of any agents behind in the queue.

What is left to show is that the coefficients $\left(a_{(q,Q)}, b_{(q,Q)}, c_{(q,Q)}\right)_{Q\in\{q,q+1,\dots,\bar{Q}\}}$ are well defined for any $q \in [\bar{Q}]$. For one of the parameters to be undefined, it would have to be the case that $r(Q) = r^-(q,Q)b_{(q,Q-1)}$ so that the denominator takes value zero. Since $r^-(q,Q) < r(Q)$ always holds, as we have discussed earlier, this would imply that $b_{(q,Q-1)} > 1$. The following claim proves that this is not possible.

Claim 1. $b_{(q,Q)} < 1$ for all $q \in [\bar{Q}]$, and all $Q \in \{q, q+1, \dots, \bar{Q}\}$.

Proof. Consider any fixed $q \in [\bar{Q}]$. We proceed by induction on Q starting from $Q = q$. First, we can see that $b_{(q,q)} < 1$ always holds because given eq. (10), $r(q) \geq \lambda\phi(q) + q\theta > \lambda\phi(q)$ when $\theta > 0$.

Now, for the purpose of induction, suppose that $b_{(q,Q)} < 1$ for some $Q \geq q$. $b_{(q,Q+1)}$ is of the form

$$b_{(q,Q+1)} = \frac{\phi(Q+1)\lambda}{r(Q+1) - r^-(q,Q+1)b_{(q,Q)}}.$$

Assume towards a contradiction that $b_{(q,Q+1)} \geq 1$. We must have

$$\begin{aligned} & \phi(Q+1)\lambda \geq r(Q+1) - r^-(q,Q+1)b_{(q,Q)} \\ \implies \quad & 0 \geq r^-(q,Q+1) + r^+(q,Q+1) + r^0(q,Q+1) + \phi(Q+1)\lambda - \phi(Q+1)\lambda - r^-(q,Q+1)b_{(q,Q)} \\ \implies \quad & 0 \geq r^0(q,Q+1) + r^+(q,Q+1) + r^-(q,Q+1)(1 - b_{(q,Q)}). \end{aligned}$$

Since $b_{q,Q} < 1$ holds by our induction assumption, the inequality above can only be true if all of the rates $r^0(q,Q+1) = r^+(q,Q+1) = r^-(q,Q+1) = 0$. However, $r^0(q,Q+1) + r^+(q,Q+$

$1) + r^-(q, Q+1) \geq (Q+1)\theta > 0$ according to (7)–(9). This is a contradiction, thus $b_{(q,Q)} < 1$ for all $q \in [\bar{Q}]$ and all $Q \in \{q, q+1, \ldots, \bar{Q}\}$ and the coefficients $a_{(q,Q)}$, $b_{(q,Q)}$ and $c_{(q,Q)}$ are all well defined. □

Algorithm 4 Compute the continuation payoffs

Initialization: $V(0,\cdot) \leftarrow 0,\ V(\cdot,0) \leftarrow 0,\ V(\cdot,\bar{Q}+1) \leftarrow 0.$
Input: $((a_{q,Q},\, b_{q,Q},\, c_{q,Q}))_{Q\in[\bar{Q}],q\in[Q]}$.
for $q = 1$ **to** $\bar{Q}$ **do**
  for $Q = \bar{Q}$ **downto** q **do**
    if $Q = \bar{Q}$ **then**
      $V(q,Q) \leftarrow a_{(q,Q)} + c_{(q,Q)}\, V(q-1, Q-1).$
    else
      $V(q,Q) \leftarrow a_{(q,Q)} + b_{(q,Q)}\, V(q,Q+1) + c_{(q,Q)}\, V(q-1,Q-1).$
    end if
  end for
end for
return $(V(q,Q))_{Q\in[\bar{Q}],\forall q\in[Q]}$.

This completes the proof of Lemma 1. The full recursive procedure for computing V is detailed in Algorithm 4. The proof also shows that each $V(q,Q)$ is a rational function of σ, with all denominators being nonzero for every $\sigma \in \mathcal{S}$. Therefore, each $V(q,Q)$ is continuous in σ. □

B.2 Proof of Lemma 2

Lemma 2 (Equilibrium Conditions)**.** Under Assumption 1, i.e., when the dispatch policy does not offer any item to the same agent more than once, a strategy $\sigma^* = (\alpha^*, \phi^*)$ forms a symmetric MPE if and only if σ^* and its corresponding continuation payoff V^* satisfy

$$\alpha^*_\ell(q,Q) = P_{[0,1]}\left[\alpha^*_\ell(q,Q) + \frac{\partial H_{q,Q}(\alpha^*; V^*)}{\partial \alpha_\ell(q,Q)}\right], \qquad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{14}$$

$$\phi^*(Q-1) = P_{[0,1]}\big[\phi^*(Q-1) + V^*(Q,Q)\big], \qquad \forall Q \in [\bar{Q}]. \tag{15}$$

Moreover, $(\partial H_{q,Q}(\alpha; V)/\partial \alpha_\ell(q,Q))_{\ell\in\mathcal{L},Q\in[\bar{Q}],q\in[Q]}$ can be evaluated in closed form.

Proof. The proof establishes the equivalence between the MPE definitions in (2), (3), and (4), and the fixed-point equations (14) and (15).

- (2) $\iff$ (14)

  We first show that (2) $\implies$ (14) by demonstrating that the latter follows from the KKT conditions of the best-response problem (BR), which are necessary for optimality.

  Specifically, given the continuation payoff V^* corresponding to strategy α^*, the best-response objective function $H_{q,Q}(\alpha; V^*)$ as defined in (13) is a polynomial function in $\alpha_\ell(q,Q), \ell \in \mathcal{L}$ thus smooth. Since (BR) has linear constraints, the optimal solution $\alpha^\star$ of (BR) (it exists by extreme value theorem) along with Lagrange multipliers $\beta_\ell(q,Q)$ and $\gamma_\ell(q,Q)$ must satisfy the following Karush–Kuhn–Tucker (KKT) conditions.

– Stationarity

$$\frac{\partial H_{q,Q}(\alpha^\star; V^*)}{\partial \alpha_\ell(q,Q)} + \beta_\ell(q,Q) - \gamma_\ell(q,Q) = 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}.$$

– Primal feasibility

$$\alpha^\star_\ell(q,Q) \in [0,1], \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}.$$

– Dual feasibility

$$\beta_\ell(q,Q),\ \gamma_\ell(q,Q) \geq 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}.$$

– Complementary slackness

$$\beta_\ell(q,Q)\alpha^\star_\ell(q,Q) = 0,\ \gamma_\ell(q,Q)(1-\alpha^\star_\ell(q,Q)) = 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}.$$

It can be seen that these KKT conditions imply

$$\begin{aligned}
\alpha^\star_\ell(q,Q) \in (0,1) \quad &\Longrightarrow \quad \frac{\partial H_{q,Q}(\alpha^\star; V^*)}{\partial \alpha_\ell(q,Q)} = 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}, \\
\alpha^\star_\ell(q,Q) = 1 \quad &\Longrightarrow \quad \frac{\partial H_{q,Q}(\alpha^\star; V^*)}{\partial \alpha_\ell(q,Q)} \geq 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L}, \\
\alpha^\star_\ell(q,Q) = 0 \quad &\Longrightarrow \quad \frac{\partial H_{q,Q}(\alpha^\star; V^*)}{\partial \alpha_\ell(q,Q)} \leq 0, \quad \forall Q \in [\bar{Q}],\ \forall q \in [Q],\ \forall \ell \in \mathcal{L},
\end{aligned}$$

which further imply (14).

For the reverse direction (14) $\Longrightarrow$ (2), we show that the best-response problem (BR) is a concave maximization problem; therefore, the KKT conditions are not only necessary but also sufficient for optimality. In fact, under Assumption 1, we show that $H_{q,Q}(\alpha; V^*)$ is *linear* in $\alpha_\ell(q,Q)$ for all $\ell \in \mathcal{L}$. To see this, observe that

– for part (i) of $H_{q,Q}(\alpha; V^*)$, $\nu_\ell(q,Q)$ does *not* depend on $\alpha_\ell(q,Q)$. This is because $\nu_\ell(q,Q)$ is a linear combination of the $g^{(k)}_\ell(q,Q)$'s as derived in (5). $g^{(k)}_\ell(q,Q)$ does not depend on $\alpha_\ell(q,Q)$ since for any dispatch history $h^{(k-1)} \in \mathcal{H}^{(k-1)}$ such that $q \in h^{(k-1)}$, we must have $f^{(k)}_\ell(q,Q;h^{(k-1)}) = 0$ given Assumption 1;

– for parts (ii) and (iii) of $H_{q,Q}(\alpha; V^*)$, $r^+(q,Q)$ and $r^-(q,Q)$ are at most linear in $\alpha_\ell(q,Q)$. This is because the two rate terms depend on $\nu_\ell(q',Q)$ for $q' \neq q$, which is a linear combination of the $g^{(k)}_\ell(q',Q)$'s. From (5) we can see that each $g^{(k)}_\ell(q',Q)$ is at most linear in $\alpha_\ell(q,Q)$ since under Assumption 1, any dispatch history $h^{(k-1)} \in \mathcal{H}^{(k-1)}$ with non-zero probability can contain q at most once.

- (3), (4) $\Longleftrightarrow$ (15)

  Consider the following three cases.

  1: $\phi^*(Q-1) \in (0,1)$.
  It is not hard to verify in this case that (3), (4) $\Longleftrightarrow$ $V^*(Q,Q) = 0$ $\Longleftrightarrow$ (15).

  2: $\phi^*(Q-1) = 1$.
  It is not hard to verify in this case that (3), (4) $\Longleftrightarrow$ $V^*(Q,Q) \geq 0$ $\Longleftrightarrow$ (15).

  3: $\phi^*(Q-1) = 0$.
  It is not hard to verify in this case that (3), (4) $\Longleftrightarrow$ $V^*(Q,Q) \leq 0$ $\Longleftrightarrow$ (15).

This completes the proof of Lemma 2. □

B.3 Closed-form Expressions for $\partial H_{q,Q}(\alpha; V)/\partial \alpha_\ell(q, Q)$

In this section, we provide the closed-form expressions for the partial derivatives of $H_{q,Q}(\alpha; V)$ (as defined in eq. (13)) with respect to $\alpha_\ell(q, Q)$ for each $\ell \in \mathcal{L}$. These expressions are valid for any general dispatch policy and do not rely on Assumption 1 (i.e., that no item is dispatched to the same agent more than once).

To take the partial derivative of $H_{q,Q}(\alpha; V^*)$ w.r.t. $\alpha_\ell(q, Q)$ for a state (q, Q) and an item type $\ell \in \mathcal{L}$, we first observe that in the expression of $H_{q,Q}(\alpha; V)$ given by eq. (13), both w_ℓ and V are constants that are independent to $\alpha_\ell(q, Q)$. Also observe that both $\nu_{\ell'}(q, Q)$ and $\alpha_{\ell'}(q, Q)$ are independent to $\alpha_\ell(q, Q)$ when $\ell' \neq \ell$. Therefore, we only need to derive the partial derivatives of the following three terms w.r.t. $\alpha_\ell(q, Q)$:

(i) $\nu_\ell(q, Q)\alpha_\ell(q, Q)$,

(ii) $r^+(q, Q)$, and

(iii) $r^-(q, Q)$.

We start with part (i), $\nu_\ell(q, Q)\alpha_\ell(q, Q)$. For convenience of notation, define

$$z_\ell(q, Q) \triangleq \nu_\ell(q, Q)\alpha_\ell(q, Q), \ \forall Q \in [\bar{Q}], \forall q \in [Q].$$

Recall from (6) that $\nu_\ell(q, Q) = \mu_\ell \sum_{k=1}^{p} g_\ell^{(k)}(q, Q)$. We have

$$\begin{aligned}\frac{\partial z_\ell(q, Q)}{\partial \alpha_\ell(q, Q)} &= \nu_\ell(q, Q) + \frac{\partial \nu_\ell(q, Q)}{\partial \alpha_\ell(q, Q)}\alpha_\ell(q, Q) \\ &= \mu_\ell \sum_{k=1}^{p} g_\ell^{(k)}(q, Q) + \left(\mu_\ell \sum_{k=1}^{p} \frac{\partial g_\ell^{(k)}(q, Q)}{\partial \alpha_\ell(q, Q)}\right) \alpha_\ell(q, Q).\end{aligned}$$

What is left to compute is the partial derivatives of $g_\ell^{(k)}(q, Q)$ w.r.t. the acceptance probabilities for each $k \in [p]$. Recall that $g_\ell^{(k)}(q, Q)$ given by (5) is the probability of seeing an item ℓ during its k^{th} dispatch at state (q, Q). As a function of $\alpha_\ell(\bar{q}, Q)$ for any $\bar{q} \in [Q]$, $g_\ell^{(k)}(q, Q)$ is a polynomial of degree at most $k - 1$, and we can compute its derivative w.r.t. $\alpha_\ell(\bar{q}, Q)$ using the chain rule.

This calculation can be further simplified for history-independent policies. Specifically, when the dispatch policy is history-independent, the expression in (5) can be rewritten in the following recursive form:

$$\begin{aligned} g_\ell^{(1)}(q, Q) &= f_\ell^{(1)}(q, Q), \\ g_\ell^{(2)}(q, Q) &= f_\ell^{(2)}(q, Q) \sum_{q'=1}^{Q} g_\ell^{(1)}(q', Q)(1 - \alpha_\ell(q', Q)), \\ &\vdots \\ g_\ell^{(k)}(q, Q) &= f_\ell^{(k)}(q, Q) \sum_{q'=1}^{Q} g_\ell^{(k-1)}(q', Q)(1 - \alpha_\ell(q', Q)).\end{aligned}$$

We can then take the partial derivatives of these probabilities at a state (q, Q) with respect to the acceptance strategy at any queue position $\bar{q} \leq Q$ in a similar recursive way.

$$\frac{\partial g_\ell^{(1)}(q, Q)}{\partial \alpha_\ell(\bar{q}, Q)} = 0,$$

$$\frac{\partial g_\ell^{(2)}(q,Q)}{\partial \alpha_\ell(\bar{q},Q)} = -f_\ell^{(2)}(q,Q) g_\ell^{(1)}(\bar{q},Q),$$

$$\vdots$$

$$\frac{\partial g_\ell^{(k)}(q,Q)}{\partial \alpha_\ell(\bar{q},Q)} = f_\ell^{(k)}(q,Q) \left[-g_\ell^{(k-1)}(\bar{q},Q) + \sum_{q'=1}^{Q} (1-\alpha_\ell(q',Q)) \cdot \frac{\partial g_\ell^{(k-1)}(q',Q)}{\partial \alpha_\ell(\bar{q},Q)} \right].$$

This completes our derivation for part (i).

For parts (ii) and (iii), observe that $r^+(q,Q)$ and $r^-(q,Q)$ as defined in (8) and (9) are both linear combinations of $\nu_\ell(q',Q)\alpha_\ell(q',Q)$ for $q' \neq q$. $\alpha_\ell(q',Q)$ is independent of $\alpha_\ell(q,Q)$ when $q' \neq q$, thus the partial derivatives $\partial r^+(q,Q)/\partial \alpha_\ell(q,Q)$ and $\partial r^-(q,Q)/\partial \alpha_\ell(q,Q)$ can be computed by taking $\partial \nu_\ell(q',Q)/\partial \alpha_\ell(q,Q)$, which again only depends on the derivative of the g terms, as we have discussed above.

B.4 Proof of Theorem 1

Theorem 1 (Existence of MPE)**.** *There exists a symmetric Markov perfect equilibrium under any dispatch policy satisfying Assumption 1, i.e., when the policy does not dispatch an item to the same agent more than once.*

Proof. We prove this theorem by showing that under any dispatch policy (even for those that violate Assumption 1), the first-order conditions (14) and (15) must admit a solution. We establish this existence by applying Brouwer's fixed-point theorem, treating (14) and (15) as fixed-point equations of the strategy $\sigma = (\alpha, \phi)$, where V depends on σ in the way that it's the continuation payoff corresponding to σ.

Recall from Appendix B.3 that $z_\ell(q,Q) \triangleq \nu_\ell(q,Q)\alpha_\ell(q,Q)$. The partial derivative of H w.r.t. α can be decomposed as:

$$\begin{aligned}\frac{\partial H_{q,Q}(\alpha;V)}{\partial \alpha_\ell(q,Q)} =& \sum_{\ell \in \mathcal{L}} \frac{\partial z_\ell(q,Q)}{\partial \alpha_\ell(q,Q)} \big(w_\ell - V(q,Q)\big) + \frac{\partial r^+(q,Q)}{\partial \alpha_\ell(q,Q)} \big(V(q-1,Q-1) - V(q,Q)\big) \\ &+ \frac{\partial r^-(q,Q)}{\partial \alpha_\ell(q,Q)} \big(V(q,Q-1) - V(q,Q)\big).\end{aligned}$$

Recall also from Appendix B.3 that the partial derivatives $\partial z_\ell(q,Q)/\partial \alpha_\ell(q,Q)$, $\partial r^+(q,Q)/\partial \alpha_\ell(q,Q)$, and $\partial r^-(q,Q)/\partial \alpha_\ell(q,Q)$ are all polynomial functions of α. Moreover, by Lemma 1, the continuation payoff V is continuous in σ. It follows that $\partial H_{q,Q}(\alpha;V)/\partial \alpha_\ell(q,Q)$ is also continuous in σ. Since the queue length is bounded by $\bar{Q}$, the strategy space $\mathcal{S}$ is compact and convex. The projection operator onto a bounded convex set is continuous, and the composition of continuous functions is continuous. Therefore, by Brouwer's fixed-point theorem, the fixed-point equations (14) and (15) always admit a solution. □

B.5 Monotonicity of $\nu_\ell(q,Q)\alpha_\ell(q,Q)$

We show that $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ is non-decreasing in $\alpha_\ell(q,Q)$ under any dispatch policy.

Lemma 3. For any dispatch policy π, $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ is non-decreasing in $\alpha_\ell(q,Q)$.

Proof. We prove this result using a sample path argument. Consider any arbitrary sample path of a dispatch history consisting of p dispatches, denoted by $h^{(p)}$. We simulate the outcome by first sampling a sequence of p numbers independently and uniformly from $[0, 1]$, denoted by $u^{(p)}$. Put $\alpha^{(p)} = [\alpha_\ell(q, Q) : q \in h^{(p)}]$. Consider the following algorithm of sampling the dispatching outcome given a dispatch history $h^{(p)}$.

Algorithm 5 Simulate Dispatching Outcomes

Require: Sequences $h^{(p)}$, $u^{(p)}$ and $\alpha^{(p)}$.
1: **for** $k \leftarrow 1$ **to** p **do**
2: **if** $\alpha_k^{(p)} > u_k^{(p)}$ **then**
3: **return** `accepted by agent` $h_k^{(p)}$
4: **end if**
5: **end for**
6: **return** `item lost`

Intuitively, item ℓ will be accepted by the first agent in $h^{(p)}$ whose acceptance probability exceeds the corresponding sampled uniform random variable. Now consider the following scenarios.

- The focal agent at position q is *not* in $h^{(p)}$, and $\alpha_\ell(q, Q)$ clearly does affect the dispatch outcome.
- The focal agent at position q appears in $h^{(p)}$, possibly multiple times. Suppose item ℓ is accepted by an agent ahead of the first occurrence of q. In this case, $\alpha_\ell(q, Q)$ does not influence the dispatch outcome—the item will still be accepted by the same agent.
- The focal agent at position q appears in $h^{(p)}$, potentially multiple times. If item ℓ is accepted by agent q (not necessarily at the first occurrence), then increasing $\alpha_\ell(q, Q)$ ensures the item is still accepted by agent q, possibly at an earlier occurrence in $h^{(p)}$.
- The focal agent at position q appears in $h^{(p)}$, possibly multiple times. Suppose item ℓ is accepted by an agent after the first occurrence of q in $h^{(p)}$. In this case, increasing $\alpha_\ell(q, Q)$ may change the outcome so that the item is instead accepted by agent q at an earlier position in $h^{(p)}$.
- The focal agent at position q appears in $h^{(p)}$, possibly multiple times. Suppose item ℓ is lost because no one accepts in $h^{(p)}$. Increasing $\alpha_\ell(q, Q)$ may change the outcome, resulting in the item being accepted by agent q in $h^{(p)}$.

These discussions imply that increasing $\alpha_\ell(q, Q)$ can only increase the chance of agent q getting item ℓ. This concludes the proof. □

B.6 Proof of Corollary 1

Corollary 1. An MPE always exists under simple dispatch policies. In this setting, a strategy $\sigma^* = (\alpha^*, \phi^*)$ constitutes a symmetric MPE if the complementarity conditions are satisfied by σ^* together with its associated continuation payoff V^*:

$$V^*(q, Q) < w_\ell \implies \alpha_\ell^*(q, Q) = 1, \quad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{16}$$

$$V^*(q, Q) > w_\ell \implies \alpha_\ell^*(q, Q) = 0, \quad \forall \ell \in \mathcal{L},\ \forall Q \in [\bar{Q}],\ \forall q \in [Q], \tag{17}$$

$$V^*(Q,Q) > 0 \quad \Longrightarrow \quad \phi^*(Q-1) = 1, \quad \forall Q \in [\bar{Q}], \tag{18}$$
$$V^*(Q,Q) < 0 \quad \Longrightarrow \quad \phi^*(Q-1) = 0, \quad \forall Q \in [\bar{Q}]. \tag{19}$$

Proof. Under simple policies, $\partial r^+(q,Q)/\partial \alpha_\ell(q,Q) = 0$ because agents ahead will never receive the item again once the focal agent declines it; similarly, $V(q,Q-1) = V(q,Q)$ since the dispatch probability does not depend on the queue length. As a result, (BR) reduces to

$$\max_{\alpha_\ell(q,Q)\in[0,1],\, \ell\in\mathcal{L}} \sum_{\ell\in\mathcal{L}} \nu_\ell(q,Q)\alpha_\ell(q,Q)\big(w_\ell - V(q,Q)\big). \tag{32}$$

By Lemma 3, we know that $\nu_\ell(q,Q)\alpha_\ell(q,Q)$ is non-decreasing in $\alpha_\ell(q,Q)$. This implies that an optimal solution to (32) takes the following form.

$$\alpha^*_\ell(q,Q) = \begin{cases} 1, & \text{if } w_\ell > V(q,Q), \\ \text{any value in } [0,1], & \text{if } w_\ell = V(q,Q), \\ 0, & \text{if } w_\ell < V(q,Q), \end{cases}$$

for all $\ell \in \mathcal{L}$. As a consequence, any solution satisfying (16) and (17) is an optimal solution to (32) and forms an optimal acceptance strategy satisfying the Bellman equation (12), thus satisfying the MPE definition (2).

Using a similar argument in the second part of the proof of Lemma 2, we can show that conditions (3) and (4) are equivalent to (18) and (19). Moreover, any solution of (21) and (22) satisfy (16)–(19). By Lemma 1, $V(q,Q)$ is a continuous function of σ. Since the queue length is bounded by $\bar{Q}$, the strategy space $\mathcal{S}$ is compact and convex. The projection operator onto a bounded convex set is continuous, and the composition of continuous functions is continuous. Therefore, by Brouwer's fixed-point theorem, the fixed-point equations (21) and (22) always admit a solution. This completes the proof of Corollary 1. □

B.7 Proof of Theorem 2

We start by analyzing the head of the queue. By the two properties of simple dispatch policies, for any item $\ell \in \mathcal{L}$, we must have $\nu_\ell(1,Q) = \nu_\ell(1,\cdot)$, $\forall Q \in [\bar{Q}]$ for some constant $\nu_\ell(1,\cdot)$ regardless of the agent's strategy. This suggests that the equilibrium $V^*(1,\cdot)$ is the optimal value of the following (single-agent) optimization problem and is queue-length independent as well.

$$V^*(1,\cdot) = \max_{\alpha(1,\cdot)\in[0,1]^{L+1}} \frac{\sum_{\ell\in\mathcal{L}} w_\ell \cdot \nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot) - c_w}{\sum_{\ell\in\mathcal{L}} \nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot) + \theta}. \tag{33}$$

Based on a standard result in choice-based revenue management literature, we have the following claim regarding an optimal solution of (33). Interestingly, this is also consistent with the complementarity conditions in Corollary 1.

Claim 2. Any optimal solution of problem (33) takes the form of

$$\alpha^*_\ell(1,\cdot) = \begin{cases} 1, & \text{if } w_\ell > V^*(1,\cdot), \\ \text{any value in } [0,1], & \text{if } w_\ell = V^*(1,\cdot), \\ 0, & \text{if } w_\ell < V^*(1,\cdot), \end{cases}$$

for all $\ell \in \mathcal{L}$.

Proof. A similar proof appears in Theorem 5.1 of Gallego and Topaloglu [2019]. Equation (33) suggests

$$V^*(1,\cdot) \geq \frac{\sum_{\ell\in\mathcal{L}} w_\ell \cdot \nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot) - c_w}{\sum_{\ell\in\mathcal{L}} \nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot)+\theta}, \quad \forall \alpha(1,\cdot)\in[0,1]^{L+1}.$$

Equivalently, this can be written as

$$\theta V^*(1,\cdot) \geq \sum_{\ell\in\mathcal{L}} (w_\ell - V^*(1,\cdot))\cdot\nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot) - c_w, \quad \forall \alpha(1,\cdot)\in[0,1]^{L+1}.$$

This implies that there is an equivalent optimization formulation to (33):

$$\max_{\alpha(1,\cdot)\in[0,1]^L} \sum_{\ell\in\mathcal{L}} (w_\ell - V^*(1,\cdot))\cdot\nu_\ell(1,\cdot)\cdot\alpha_\ell(1,\cdot) - c_w. \tag{34}$$

It is clear to see any optimal solution to (34) satisfies the conditions in the statement. □

For any queue length $Q\in[\bar{Q}]$, we now show that $V^{(t)}(1,Q)$ is monotonically increasing with respect to t, regardless of the initial strategy $\sigma^{(0)}$ used. Using (33) we can write

$$V^{(t+1)}(1,Q) = \frac{\sum_{\ell\in\mathcal{L}} w_\ell\cdot\alpha_\ell^{(t+1)}(1,Q)\cdot\nu_\ell(1,Q) - c_w}{\sum_{\ell\in\mathcal{L}}\alpha_\ell^{(t+1)}(1,Q)\cdot\nu_\ell(1,Q)+\theta}. \tag{35}$$

Taking the difference between $V^{(t+1)}(1,Q) - V^{(t)}(1,Q)$, we get

$$V^{(t+1)}(1,Q) - V^{(t)}(1,Q) = \frac{\sum_{\ell\in\mathcal{L}}(w_\ell - V^{(t)}(1,Q))\cdot\alpha_\ell^{(t+1)}(1,Q)\cdot\nu_\ell(1,Q) - c_w - V^{(t)}(1,Q)\cdot\theta}{\sum_{\ell\in\mathcal{L}}\alpha_\ell^{(t+1)}(1,Q)\cdot\nu_\ell(1,Q)+\theta}.$$

Since,

$$\begin{aligned}
&V^{(t)}(1,Q) = \frac{\sum_{\ell\in\mathcal{L}} w_\ell\cdot\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q) - c_w}{\sum_{\ell\in\mathcal{L}}\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q)+\theta}\\
\implies &\frac{\sum_{\ell\in\mathcal{L}}(w_\ell - V^{(t)}(1,Q))\cdot\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q) - c_w - V^{(t)}(1,Q)\cdot\theta}{\sum_{\ell\in\mathcal{L}}\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q)+\theta} = 0\\
\implies &\sum_{\ell\in\mathcal{L}}(w_\ell - V^{(t)}(1,Q))\cdot\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q) - c_w - V^{(t)}(1,Q)\cdot\theta = 0.
\end{aligned}$$

Notice that in each iteration of Algorithm 2, we increase $\alpha_\ell(1,Q)$ for items with $w_\ell > V^{(t)}(1,Q)$ and decrease $\alpha_\ell(1,Q)$ for items with $w_i < V^{(t)}(1,Q)$. So, it must be the case that

$$\begin{aligned}
&\sum_{\ell\in\mathcal{L}}(w_\ell - V^{(t)}(1,Q))\cdot\alpha_\ell^{(t+1)}(1,Q)\cdot\nu_\ell(1,Q)\\
\geq &\sum_{\ell\in\mathcal{L}}(w_\ell - V^{(t)}(1,Q))\cdot\alpha_\ell^{(t)}(1,Q)\cdot\nu_\ell(1,Q) = c_w + V^{(t)}(1,Q)\cdot\theta\\
\implies &V^{(t+1)}(1,Q)\geq V^{(t)}(1,Q).
\end{aligned}$$

This concludes the proof of $V^{(t)}(1,Q)$ being non-decreasing in t for any $Q\in[\bar{Q}]$. Since $V^{(t)}(1,Q)$ is upper bounded by w_1, the sequence of $V^{(t)}(1,Q)$ must converge. We now show it converges to $V^*(1,\cdot)$ in a finite number of iterations. We do so by showing $\alpha_l^{(t)}(1,Q)$ converges to a solution satisfying the optimality conditions in Claim 2 in a finite number of iterations.

- **Case 1:** $\alpha_l^{(t)}(1,Q)$ converges to 1 if $w_\ell > V^*(1,Q)$ in a finite number of iterations.

  We know that $V^{(t)}(1,Q) \le V^*(1,\cdot) < w_\ell$ for any $t \ge 0$. Let $\delta = \min_{\ell\in\mathcal{L}, w_\ell > V^*(1,\cdot)}\{w_\ell - V^*(1,\cdot)\}$. It follows that $\delta > 0$. So, $\alpha_\ell^{(t)}(1,Q)$ must increase by at least $\gamma^{(t)}\delta$ in any iteration t. As long as $\sum_{t=0}^{+\infty}\gamma^{(t)} = +\infty$, $\alpha_\ell^{(t)}(1,\cdot)$ must converge to 1 within a finite number of iterations.

- **Case 2:** $\alpha_l^{(t)}(1,Q)$ converges to 0 if $w_\ell < V^*$ in a finite number of iterations.

  Suppose all items with $w_\ell > V^*(1,\cdot)$ are already accepted with probability 1 at some finite iteration t. We start from the worst paying item that is accepted with a strictly positive probability at iteration t. Let's call it $w_{\ell'}$. We claim that $V^{(t)}(1,Q) > w_{\ell'}$. To see that, let $\mathcal{L}^* = \{\ell \in \mathcal{L} : w_\ell > V^*(1,\cdot)\}$. Consider a strategy that accepts everything in $\mathcal{L}^*$ and one other item ℓ' such that $w_{\ell'} < V^*(1,\cdot)$ with acceptance probability $\alpha_{\ell'}(1,Q) = \varepsilon > 0$. We have the value of this strategy

  $$\begin{aligned}
  &\frac{\sum_{\ell\in\mathcal{L}^*} w_\ell\nu_\ell(1,Q) + w_{\ell'}\nu_{\ell'}(1,Q)\varepsilon - c_w}{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q) + \nu_{\ell'}(1,Q)\varepsilon + \theta}\\
  =&\frac{\sum_{\ell\in\mathcal{L}^*} w_\ell\nu_\ell(1,Q) - c_w}{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q) + \nu_{\ell'}(1,Q)\varepsilon + \theta} + \frac{w_{\ell'}\nu_{\ell'}(1,Q)\varepsilon}{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q) + \nu_{\ell'}(1,Q)\varepsilon + \theta}\\
  =&\frac{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q)}{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q) + \nu_{\ell'}(1,Q)\varepsilon}V^*(1,\cdot) + \frac{\nu_{\ell'}(1,Q)\varepsilon}{\sum_{\ell\in\mathcal{L}^*}\nu_\ell(1,Q) + \nu_{\ell'}(1,Q)\varepsilon + \theta}w_{\ell'}\\
  >&w_{\ell'}.
  \end{aligned}$$

  Define $\delta = V^{(t)}(1,Q) - w_{\ell'}$. We know that $\alpha_{\ell'}^{(t)}(1,Q)$ drops by at least $\delta\gamma^{(t)}$ in all iterations $t' \ge t$ since $V^{(t)}(1,Q)$ is non-decreasing. Once $\alpha_{\ell'}^{(t)}(1,Q)$ hits zero, we will move on to apply the same argument to the new worst item that is accepted with a strictly positive probability.

- **Case 3:** $\alpha_l^{(t)}(1,Q)$ converges in a finite number of iterations if $w_\ell = V^*$.

  For this case, suppose that cases 1 and 2 have already converged. If there exists $\ell \in \mathcal{L}$ s.t. $w_\ell = V^*$, then at this point $V^{(t)}(1,Q) = V^*$ according to Claim 2. From this point onward, the acceptance probability α_ℓ will no longer change given that the direction of the strategy updates is constructed to be $V^{(t)}(1,Q) - w_\ell$.

This completes the proof of convergence in a finite number of steps for $q = 1$ and any $Q \in [\bar{Q}]$.

Once the strategy at $q = 1$ with queue length Q has converged according to the logic above at time t, the rates $\{\nu_\ell^{(t)}(2,Q)\}_{\ell\in\mathcal{L}}$ become fixed for all subsequent iterations $t' \ge t$. This again follows from item 2 of Definition 2, since items can only be sent to agents monotonically down the queue. Consequently, the rate for the driver to move forward to $q = 1$ not longer changes. We can now apply similar reasoning to the convergence of strategies for $q = 2$. Once the states at $q = 2$ converge, we can repeat the reasoning for $q = 3$, and so on until we complete the process up to $q = \bar{Q}$. To be a bit more specific, suppose that $V^{(t)}(1,Q), \cdots, V^{(t)}(q,Q)$ all converge after a finite number of iterations for some $q \ge 1$, and let $V^*(1,Q), \cdots, V^*(q,Q)$ to be the converged value. We now analyze the monotonicity of $V^{(t)}(q+1,Q)$. We have

$$V^{(t+1)}(q+1,Q) = \frac{\sum_{\ell\in\mathcal{L}} w_\ell \cdot \nu_\ell(q+1,Q)\cdot\alpha_\ell^{(t+1)}(q+1,Q) + r^+(q+1,Q)V^{(t+1)}(q,Q) - c_w}{\sum_{\ell\in\mathcal{L}}\nu_\ell(q+1,Q)\cdot\alpha_\ell^{(t+1)}(q+1,Q) + r^+(q+1,Q) + \theta}.$$

Note that since $V^{(t)}(1,Q),\cdots,V^{(t)}(q,Q)$ are all converged, $\nu_l(q+1,Q)$ and $r^+(q+1,Q)$ remain constants from time t onward. By treating $c'=c_w-r^+(q+1,Q)V^{(t+1)}(q,Q)$ and $\theta'=r^+(q+1,Q)+\theta$, we get the same structure as the one we analyzed in (35). So all results for $V(1,Q)$ directly apply.

Because conditions in Claim 2 are the same as the complementarity conditions (16) and (17) in Corollary 1, we just proved that the acceptance strategy $\alpha_l^{(t)}(q,Q)$ converged satisfy (16) and (17). What remains to be demonstrated is that the joining strategy $\phi^{(t)}(Q)$ also converges to an equilibrium that satisfies (18) and (19) in a finite number of iterations. This can be seen in a quite straightforward manner. Once $V^{(t)}(Q,Q)$ converges to $V^*(Q,Q)$ for some $Q\in[\bar{Q}]$ in a finite number of iterations, consider the following scenarios:

- **Case 1:** $V^*(Q,Q)>0$

  This suggests that $\phi^{(t)}(Q-1)$ must increase by exactly $\gamma^{(t)}V^*(Q,Q)$ in any iteration after t. As long as $\sum_{t=0}^{+\infty}\gamma^{(t)}=+\infty$, $\phi^{(t)}(Q-1)$ will converge to 1 in a finite number of iterations.

- **Case 2:** $V^*(Q,Q)<0$

  This suggests that $\phi^{(t)}(Q-1)$ must decrease by exactly $-\gamma^{(t)}V^*(Q,Q)$ in any iteration after t. As long as $\sum_{t=0}^{+\infty}\gamma^{(t)}=+\infty$, $\phi^{(t)}(Q-1)$ will converge to 0 in a finite number of iterations.

- **Case 3:** $V^*(Q,Q)=0$

  This suggests that $\phi^{(t)}(Q-1)$ will remain constant after t.

These directly imply that the converged $\phi^{(t)}(Q-1)$ satisfies (18) and (19). Then by Corollary 1, we complete the proof that Algorithm 2 converges to an MPE in a finite number of iterations regardless of the initial strategy $\vec{\sigma}^{(0)}$.

B.8 First-Best Outcome

We solve for the first-best net revenue using the following infinite-horizon average-reward MDP:

$$V(Q)+g=\frac{1}{M}\Big[\sum_{\ell\in\mathcal{L}}\mu_\ell\underbrace{\max\{w_\ell+V(Q-1),\,V(Q)\}}_{\text{item dispatch decision}}+\lambda\underbrace{\max\{V(Q),\,V(Q+1)\}}_{\text{admission control decision}}$$
$$+\underbrace{Q\,\theta\,V(Q-1)}_{\text{exogenous reneging}}+\underbrace{(M-\Delta(Q))\,V(Q)}_{\text{null event}}-\underbrace{c_w\,Q}_{\text{waiting cost}}\Big],$$

where M is a uniformization constant such that $M\geq\sum_{\ell\in\mathcal{L}}\mu_\ell+\lambda+\theta\bar{Q}$ and g is the optimal net revenue per period. One of four events occurs: (i) an item of type ℓ arrives with rate μ_ℓ and we choose between dispatching (gain w_ℓ and move to $Q-1$) or rejecting (stay at Q); (ii) an agent arrives at rate λ and we choose between blocking (stay at Q) or admitting (move to $Q+1$); (iii) exogenous reneging at rate θQ moves the state to $Q-1$; or (iv) a null event of rate $M-\Delta(Q)$ leaves system in state Q.

To solve this MDP, we use relative value iteration to maximize the long-run average reward (see Chapter 8 of Puterman 2005). Once the algorithm converges, we extract the optimal policy, which then allows us to compute the steady-state distribution and various performance metrics, including throughput, average waiting time, and queue length.

B.9 Deriving the Expectation and Variance of Agent Payoffs

First, we introduce the following notation to discuss transitions between adjacent states. Fix an initial state s_0, which denotes the state of the system at the moment an agent arrives. When the agent joins the end of the queue after observing a queue length of $Q-1$, we have $s_0 = (Q, Q)$. Let Y be the random variable representing the net payoff of an agent who joins the queue at state s_0, defined as the final earnings received upon accepting an item minus the total waiting cost incurred before exiting the system. Let $s_k = (q_k, Q_k)$ denote the state occupied by the agent after the k-th state transition. Similarly, let Δ_k denote the cumulative time elapsed from the moment the agent joins at s_0 to the time of the k-th transition. Both s_k and Δ_k are random variables, since they depend on the stochastic arrivals and departures of items and agents, as well as on the dispatch policy and the agents' strategies. Let τ denote the random index of the transition at which the agent exits the queue, either by accepting an item and departing or by reneging. Note that τ is a stopping time, since its value is determined by the natural filtration up to that point and does not depend on future information.

Let $(\mathcal{F}_k)_{k\geq 0}$ be the natural filtration generated by the embedded jump chain $(s_k)_{k\geq 0}$ together with the realized waiting times up to transition k, namely $(\Delta_k)_{k\geq 0}$. Conditional on the information contained in this filtration, the agent's expected net payoff is given by

$$M(s_k) := V(q_k, Q_k) - c_w \Delta_k.$$

Note that, conditional on $\mathcal{F}_k$, $M(s_k)$ is a realized quantity. Intuitively, $M(s_k)$ represents the agent's expected total payoff after following the realized sample path up to state s_k. This is because $V(q_k, Q_k)$ is the expected continuation payoff from state (q_k, Q_k) onward, while $c_w \Delta_k$ is the waiting cost already incurred by the agent up to that point. We now construct a martingale based on $M(s_k)$.

Lemma 4 (Martingale representation)**.** Fix a strategy under a given dispatching policy. Define the stopped process

$$M_k := \begin{cases} M(s_k), & k < \tau, \\ Y, & k \geq \tau. \end{cases}$$

Then $(M_k)_{k\geq 0}$ is a martingale with respect to the filtration $(\mathcal{F}_k)_{k\geq 0}$.

Proof. First, we show the martingale property by proving that:

$$\mathbb{E}\left[M_{k+1} \mid \mathcal{F}_k\right] = M_k.$$

- If $k \geq \tau$, both sides are equal to Y and the conclusion follows.
- If $k < \tau$, then we evaluate the martingale property as follows:

$$\begin{aligned}
\mathbb{E}\left[M_{k+1} \mid \mathcal{F}_k\right] =& \mathbb{E}\left[M(s_{k+1}) \mid \mathcal{F}_k, \tau > k+1\right] \mathbb{P}\left[\tau > k+1\right] + \mathbb{E}\left[Y \mid \mathcal{F}_k, \tau = k+1\right] \mathbb{P}\left[\tau = k+1\right] \\
=& \mathbb{E}\left[V(q_{k+1}, Q_{k+1}) - c_w \Delta_{k+1} \mid \mathcal{F}_k, \tau > k+1\right] \mathbb{P}\left[\tau > k+1\right] \\
&+ \mathbb{E}\left[Y \mid \mathcal{F}_k, \tau = k+1\right] \mathbb{P}\left[\tau = k+1\right] \\
=& \mathbb{E}[V(q_{k+1}, Q_{k+1}) - c_w \Delta_k - c_w(\Delta_{k+1} - \Delta_k) \mid \mathcal{F}_k, \tau > k+1] \mathbb{P}\left[\tau > k+1\right] \\
&+ \mathbb{E}\left[Y \mid \mathcal{F}_k, \tau = k+1\right] \mathbb{P}\left[\tau = k+1\right] \\
=& \left(\mathbb{E}[V(q_{k+1}, Q_{k+1}) - c_w(\Delta_{k+1} - \Delta_k) \mid \mathcal{F}_k, \tau > k+1] - c_w \Delta_k\right) \mathbb{P}\left[\tau > k+1\right] \\
&+ \mathbb{E}\left[Y \mid \mathcal{F}_k, \tau = k+1\right] \mathbb{P}\left[\tau = k+1\right]
\end{aligned}$$

$$
\begin{aligned}
&= \mathbb{E}[V(q_{k+1}, Q_{k+1}) - c_w(\Delta_{k+1} - \Delta_k) \mid \mathcal{F}_k, \tau > k+1]\,\mathbb{P}\left[\tau > k+1\right] \\
&\quad + \left(\mathbb{E}\left[Y \mid \mathcal{F}_k, \tau = k+1\right] + c_w\Delta_k\right)\mathbb{P}\left[\tau = k+1\right] \\
&\quad - c_w\Delta_k\mathbb{P}\left[\tau > k+1\right] - c_w\Delta_k\mathbb{P}\left[\tau = k+1\right] \\
&= V(q_k, Q_k) - c_w\Delta_k \\
&= M(s_k) \\
&= M_k,
\end{aligned}
$$

where the third-to-last equality follows from the definition of $V(q_k, Q_k)$ as the conditional expectation of the terminal payoff net of all future waiting costs, conditional on starting from state (q_k, Q_k).

Next, we verify that the process is integrable. Let Δ_τ denote the time until the focal agent exits the queue. Then, for every k, we have

$$
\mathbb{E}[|M_k|] \le \mathbb{E}[|V(q_k, Q_k)|] + \mathbb{E}[|c_w\Delta_k|] \le w_1 + c_w\mathbb{E}[\Delta_\tau] \le w_1 + \frac{c_w}{\theta} < \infty.
$$

The bound by w_1 follows from the fact that the agent's positive payoff cannot exceed the reward of the highest-paying item. Moreover, since the model includes an exogenous reneging rate $\theta \ge 0$, the exit time Δ_τ is stochastically dominated by an exponential random variable with rate θ, which implies the stated bound. □

Given the martingale defined above, we now characterize the variance of agent earnings conditional on joining the queue at a given state using quadratic variation. To this end, we establish the following lemma.

Lemma 5 (Variance via Quadratic Variation)**.** Let $\Delta M_k := M_k - M_{k-1}$ for $k \ge 1$. Then, for any entry state s_0,

$$
\mathrm{Var}(Y \mid s_0, D = J) = \mathbb{E}\big[[M]_\tau \mid s_0, D = J\big],
$$

where $D \in \{J, N\}$ denotes the entry decision (join or not join), and the quadratic variation of the stopped martingale is defined by

$$
[M]_\tau := \sum_{k=1}^{\tau} (\Delta M_k)^2.
$$

Proof. In this lemma, we focus on the case in which the focal agent joins, and therefore suppress the conditioning on $D = J$. Since $(M_k)_{k\ge 0}$ is a martingale with $M_0 = M(s_0)$ and terminal value $M_\tau = Y$, we get:

$$
\begin{aligned}
&\mathrm{Var}(Y \mid s_0) \\
&= \mathbb{E}[Y^2 \mid s_0] - (\mathbb{E}[Y \mid s_0])^2 \\
&= \mathbb{E}\left[\left(M(s_0) + \sum_{k=1}^{\tau} \Delta M_k\right)^2 \,\middle|\, s_0\right] - M(s_0)^2 \\
&= \mathbb{E}\left[M(s_0)^2 + 2M(s_0)\left(\sum_{k=1}^{\tau} \Delta M_k\right) + \left(\sum_{k=1}^{\tau} \Delta M_k\right)^2 \,\middle|\, s_0\right] - M(s_0)^2 \\
&= 2M(s_0)\mathbb{E}\left[\sum_{k=1}^{\tau} \Delta M_k \,\middle|\, s_0\right] + \mathbb{E}\left[\left(\sum_{k=1}^{\tau} \Delta M_k\right)^2 \,\middle|\, s_0\right]
\end{aligned}
$$

$$
\begin{aligned}
&= 2M(s_0)\mathbb{E}\left[Y - M(s_0) \mid s_0\right] + \mathbb{E}\left[\sum_{\substack{k,j=1 \\ k\neq j}}^{\tau} \Delta M_k \Delta M_j \;\middle|\; s_0\right] + \mathbb{E}\left[\sum_{k=1}^{\tau} (\Delta M_k)^2 \;\middle|\; s_0\right] \qquad (36)\\
&= \mathbb{E}\left[\sum_{k=1}^{\tau} (\Delta M_k)^2 \;\middle|\; s_0\right]\\
&= \mathbb{E}\big[[M]_\tau \mid s_0\big],
\end{aligned}
$$

which yields the stated equality. The expectation is taken over both the randomness in the sample paths starting from s_0 and the randomness in the terminal earnings collected at time τ. We now show that the first two terms in eq. (36) are equal to zero.

Claim 3. $2M(s_0)\mathbb{E}\left[Y - M(s_0) \mid s_0\right] = 0.$

Proof. To show this property, we invoke the optional stopping theorem. We have already shown that M_k is a martingale and we know that τ is a stopping time by definition, since it denotes the index of the state where an agent leaves the queue. We know that $\mathbb{E}[\tau] \leq \frac{M}{\theta} < \infty$ since τ is stochastically dominated by a geometric random variable with parameter $\frac{\theta}{M}$ where M is defined in the same way as the uniformization constant of Section 3.2. Intuitively, $\frac{\theta}{M}$ corresponds to the smallest possible probability of leaving the queue. Next, we show that there exists a constant C such that $\mathbb{E}[|M_{k+1} - M_k| \mid \mathcal{F}_k] \leq C$ almost surely for all $k < \tau$.

To establish this property, we invoke the optional stopping theorem. We have already shown that $(M_k)_{k\geq 0}$ is a martingale, and by definition τ is a stopping time, since it is the index of the state at which the agent leaves the queue. Moreover, $\mathbb{E}[\tau] \leq M/\theta < \infty$ because τ is stochastically dominated by a geometric random variable with parameter θ/M, where M is defined analogously to the uniformization constant in Section 3.2. Intuitively, θ/M is the smallest possible probability that the agent leaves the queue at a given step. It therefore remains to show that there exists a constant C such that $\mathbb{E}[|M_{k+1} - M_k| \mid \mathcal{F}_k] \leq C$ almost surely for all $k < \tau$.

$$
\begin{aligned}
\mathbb{E}[|M_{k+1} - M_k| \mid \mathcal{F}_k] =& \mathbb{E}[|V(s_{k+1}) - V(s_k) - c_w(\Delta_{k+1} - \Delta_k)| \mid \mathcal{F}_k, \tau > k+1]\mathbb{P}\left[\tau > k+1\right]\\
&+ \mathbb{E}[|Y - V(s_k) - c_w(\Delta_\tau - \Delta_k)| \mid \mathcal{F}_k, \tau = k+1]\mathbb{P}\left[\tau = k+1\right]\\
\leq& 2w_1 + \frac{c_w}{\theta}.
\end{aligned}
$$

Thus, the standard conditions for the optional stopping theorem are satisfied, and we conclude that $\mathbb{E}[Y \mid s_0] = M(s_0)$. The remainder of the claim then follows immediately. □

Claim 4. The process $(M_k)_{k\geq 0}$ has orthogonal increments. In other words:

$$
\mathbb{E}[S_\tau] := \mathbb{E}\left[\sum_{\substack{k,j=1 \\ k\neq j}}^{\tau} \Delta M_k \Delta M_j \;\middle|\; s_0\right] = 0.
$$

Proof. First, we rewrite $\mathbb{E}[S_\tau]$ using an indicator representation.

$$
\mathbb{E}[S_\tau] = \mathbb{E}\left[\sum_{\substack{k,j=1 \\ k\neq j}}^{\tau} \Delta M_k \Delta M_j \;\middle|\; s_0\right]
$$

$$= \mathbb{E}\left[2 \left(\sum_{1 \le k < j < \infty} \Delta M_k \Delta M_j \mathbb{1}(\tau \ge j) \right) \,\middle|\, s_0 \right]$$
$$= \mathbb{E}\left[2 \lim_{n \to \infty} \sum_{1 \le k < j < n} \Delta M_k \Delta M_j \mathbb{1}(\tau \ge j) \,\middle|\, s_0 \right].$$

We prove that the limit and expectation may be interchanged for each of the two terms on the right-hand side separately. First, for every n,

$$|M_{\tau_n} - M_0| \le |M_{\tau_n}| + |M_0| \le w_1 + c_w \Delta_\tau + M_0.$$

Here M_0 is nonnegative, since an agent joins the queue only if the continuation payoff at the entry state is nonnegative. Using the inequality $(a+b+c)^2 \le 3(a^2+b^2+c^2)$, we obtain

$$(M_{\tau_n} - M_0)^2 \le (w_1 + c_w \Delta_\tau + M_0)^2 \le 3w_1^2 + 3c_w^2 \Delta_\tau^2 + 3M_0^2.$$

Define

$$Z := 3w_1^2 + 3c_w^2 \Delta_\tau^2 + 3M_0^2.$$

Taking conditional expectations gives

$$\mathbb{E}\big[(M_{\tau_n} - M_0)^2 \mid s_0\big] \le \mathbb{E}[Z \mid s_0] \le 3w_1^2 + \frac{6c_w^2}{\theta^2} + 3M_0^2 < \infty.$$

Since $\tau_n = \tau \wedge n$ increases to τ, it follows that $M_{\tau_n} \to M_\tau$ almost surely. Because Z is integrable, the dominated convergence theorem yields

$$\mathbb{E}\big[(M_{\tau_n} - M_0)^2 \mid s_0\big] \longrightarrow \mathbb{E}\big[(M_\tau - M_0)^2 \mid s_0\big].$$

Next, observe that every term in the quadratic variation sum is nonnegative. Hence, $\sum_{j \le \tau_n} (\Delta M_j)^2$ is non-decreasing in n and converges almost surely to $\sum_{j \le \tau} (\Delta M_j)^2$. Therefore, by the monotone convergence theorem, we may interchange the limit and expectation for the quadratic variation term, which yields

$$\mathbb{E}\left[\sum_{j \le \tau_n} (\Delta M_j)^2 \right] \longrightarrow \mathbb{E}\left[\sum_{j \le \tau} (\Delta M_j)^2 \right].$$

Combining the two convergences above, we obtain

$$\mathbb{E}[S_{\tau_n} \mid s_0] \longrightarrow \mathbb{E}[S_\tau \mid s_0].$$

Since $\mathbb{E}[S_{\tau_n} \mid s_0] = 0$ for every n, it follows that

$$\mathbb{E}[S_\tau \mid s_0] = \lim_{n \to \infty} \mathbb{E}[S_{\tau_n} \mid s_0] = 0.$$

□

This concludes the proof of Lemma 5. □

We now show that, under any dispatch policy and for any strategy profile adopted by the agents, the expected remaining quadratic variation upon entering a given state can be computed recursively. This recursion mirrors the continuation-payoff recursion in Appendix B.1, but with a different effective reward term.

Let us define the variance of earnings, conditional on joining the queue at state (q, Q), for all $Q \in [\bar{Q}]$ and all $q \in [Q]$, as

$$W(q,Q) := \mathrm{Var}(Y \mid s_0 = (q,Q),\, D = J) = \mathbb{E}[[M]_\tau \mid s_0 = (q,Q),\, D = J]\,,$$

where the second equality follows from Lemma 5.

We define the edge cost associated with each state transition by

$$e_{s_k \to s_{k+1}} := \mathbb{E}\Big[\big(M_{k+1} - M_k\big)^2 \,\Big|\, s_k\Big]\,, \qquad e_{s_k \to s_\tau} := \mathbb{E}\Big[\big(M_\tau - M_k\big)^2 \,\Big|\, s_k, \tau = k+1\Big]\,.$$

For a transition to an adjacent state, we have

$$\begin{aligned} e_{s_k \to s_{k+1}} &= \mathbb{E}\Big[\big(V(s_{k+1}) - V(s_k) - c_w(\Delta_{k+1} - \Delta_k)\big)^2 \,\Big|\, s_k\Big] \\ &= \big(V(s_{k+1}) - V(s_k)\big)^2 - 2\,\big(V(s_{k+1}) - V(s_k)\big)\,\frac{c_w}{r(s_k)} + \frac{2c_w^2}{r(s_k)^2}, \end{aligned}$$

where we use that the waiting time spent at state s_k is exponentially distributed with rate $r(s_k)$, the total transition rate out of state s_k.

For absorption at state s_k, where the agent accepts a job of type $\ell \in \mathcal{L}$, we similarly obtain

$$\begin{aligned} e_{s_k \to s_\tau} &= \mathbb{E}\left[\sum_{\ell \in \mathcal{L}} \frac{\nu_\ell(s_k)\alpha_\ell(s_k)}{\sum_{\ell' \in \mathcal{L}} \nu_{\ell'}(s_k)\alpha_{\ell'}(s_k)}\,\big(w_\ell - V(s_k) - c_w(\Delta_\tau - \Delta_k)\big)^2 \,\middle|\, s_k\right] \\ &= \sum_{\ell \in \mathcal{L}} \frac{\nu_\ell(s_k)\alpha_\ell(s_k)}{\sum_{\ell' \in \mathcal{L}} \nu_{\ell'}(s_k)\alpha_{\ell'}(s_k)} \left((w_\ell - V(s_k))^2 - 2(w_\ell - V(s_k))\frac{c_w}{r(s_k)} + \frac{2c_w^2}{r(s_k)^2} \right), \end{aligned}$$

where the second line again uses the fact that the waiting time at state s_k is exponential with rate $r(s_k)$. In this case, the expected squared increment upon absorption is computed by averaging over the type of job received and accepted, conditional on the event that the agent leaves the queue.

After characterizing all edge costs, and using the same transition probabilities as in Section 3.1, the recursion for $W(q,Q)$ is given by

$$\begin{aligned} W(q,Q) = \frac{r^0(q,Q)}{r(Q)}\, e_{(q,Q)\to s_\tau} + \frac{r^+(q,Q)}{r(Q)}\Big(e_{(q,Q)\to(q-1,Q-1)} + W(q-1,Q-1)\Big) \\ + \frac{r^-(q,Q)}{r(Q)}\Big(e_{(q,Q)\to(q,Q-1)} + W(q,Q-1)\Big) + \frac{\lambda\phi(Q)}{r(Q)}\Big(e_{(q,Q)\to(q,Q+1)} + W(q,Q+1)\Big). \end{aligned}$$

This recursion has the same structure as that in Appendix B.1. The key difference is that, instead of using the immediate earnings rate as the one-step reward, we now use the expected edge cost, obtained by weighting each edge cost by its corresponding transition rate. Accordingly, define

$$\begin{aligned} \omega(q,Q) := r^0(q,Q)e_{(q,Q)\to s_\tau} + r^+(q,Q)e_{(q,Q)\to(q-1,Q-1)} \\ + r^-(q,Q)e_{(q,Q)\to(q,Q-1)} + \lambda\phi(Q)e_{(q,Q)\to(q,Q+1)}. \end{aligned}$$

Then the argument in Appendix B.1, including the derivation of all closed-form coefficients, carries over directly to the variance recursion.

Next, we apply the law of total variance to characterize the variance of driver earnings, accounting for whether the driver joins the queue. Let $s_0 \sim \pi$ denote the steady-state distribution over entry states, and let $\Pr(D = J \mid s_0) = \phi(s_0)$ denote the joining strategy. We further define the overall probability that an arriving agent joins the queue by

$$p := \Pr(D = J) = \mathbb{E}_{s_0 \sim \pi}[\phi(s_0)].$$

In the calculations that follow, it is useful to interpret the continuation payoff as the expected terminal earnings conditional on entering the queue at state s_0. Accordingly, we have

$$\mathbb{E}[Y \mid s_0 = s,\, D = J] = V(s).$$

If $D = N$, then $Y = 0$ deterministically, and hence

$$\mathbb{E}[Y \mid D = N] = 0, \qquad \mathrm{Var}(Y \mid D = N) = 0.$$

Conditional on joining, we have

$$\mathbb{E}[Y \mid s_0,\, D = J] = V(s_0), \qquad \mathrm{Var}(Y \mid s_0,\, D = J) = W(s_0).$$

Applying the law of total variance conditional on $D = J$, we obtain

$$\begin{aligned}\mathrm{Var}(Y \mid D = J) &= \mathrm{Var}\big(\mathbb{E}[Y \mid s_0, D = J] \mid D = J\big) + \mathbb{E}\big[\mathrm{Var}(Y \mid s_0, D = J) \mid D = J\big] \\ &= \mathrm{Var}_{s_0 \sim \pi'}\big(V(s_0)\big) + \mathbb{E}_{s_0 \sim \pi'}\big[W(s_0)\big],\end{aligned}$$

where the variance and expectation are taken with respect to the conditional distribution

$$\pi'(s_0) := \pi(s_0 \mid D = J) = \frac{\phi(s_0)\pi(s_0)}{p}.$$

We next apply the law of total variance once more, this time with respect to the joining decision D. This gives

$$\mathrm{Var}(Y) = \mathrm{Var}\big(\mathbb{E}[Y \mid D]\big) + \mathbb{E}\big[\mathrm{Var}(Y \mid D)\big].$$

For the first term,

$$\begin{aligned}\mathrm{Var}\big(\mathbb{E}[Y \mid D]\big) &= \mathbb{E}\left[\big(\mathbb{E}[Y \mid D]\big)^2\right] - \big(\mathbb{E}\left[\mathbb{E}[Y \mid D]\right]\big)^2 \\ &= \Big[p\big(\mathbb{E}_{s_0 \sim \pi'}[V(s_0)]\big)^2 + (1-p)\cdot 0^2\Big] - \left[p\,\mathbb{E}_{s_0 \sim \pi'}[V(s_0)] + (1-p)\cdot 0\right]^2 \\ &= p(1-p)\big(\mathbb{E}_{s_0 \sim \pi'}[V(s_0)]\big)^2.\end{aligned}$$

For the second term,

$$\begin{aligned}\mathbb{E}\big[\mathrm{Var}(Y \mid D)\big] &= p\mathrm{Var}(Y \mid D = J) + (1-p)\mathrm{Var}(Y \mid D = N) \\ &= p\left(\mathrm{Var}_{s_0 \sim \pi'}\big(V(s_0)\big) + \mathbb{E}_{s_0 \sim \pi'}\big[W(s_0)\big]\right).\end{aligned}$$

Combining the two expressions yields

$$\mathrm{Var}(Y) = p(1-p)\big(\mathbb{E}_{s_0 \sim \pi'}[V(s_0)]\big)^2 + p\left(\mathrm{Var}_{s_0 \sim \pi'}\big(V(s_0)\big) + \mathbb{E}_{s_0 \sim \pi'}\big[W(s_0)\big]\right).$$

Finally, the expected driver earnings are given by

$$\mathbb{E}[Y] = p\,\mathbb{E}_{s_0 \sim \pi'}[V(s_0)] + (1-p)\cdot 0 = p\,\mathbb{E}_{s_0 \sim \pi'}[V(s_0)].$$

C Additional Examples and Discussions

Contents

C.1 Dispatch Policies . . . 53
C.1.1 Geometric Hopping . . . 53
C.1.2 Dynamic Randomized FIFO . . . 53
C.2 Equilibrium under Strict FIFO . . . 54
C.3 Multiple Equilibria . . . 55
C.4 Impact of Cascading Effects . . . 55

C.1 Dispatch Policies

In this section, we provide formal definitions for a number of additional dispatch policies discussed in the body of the paper.

C.1.1 Geometric Hopping

We formally define the geometric hopping policy that was briefly discussed in Section 3.3.

Definition 3 (Geometric Hopping)**.** The first time an item is dispatched, generate a queue position by sampling from a geometric distribution as $x \sim geom(\zeta)$ for some parameter $\zeta \in (0, 1]$.

$$f_\ell^{(1)}(q, Q) = \begin{cases} 1 & \text{if } q = x, \\ 0, & \text{otherwise.} \end{cases}$$

If the item is not accepted, we add q to $h_1^{(1)}$ and sample again $x \sim geom(\zeta)$.

$$f_\ell^{(2)}(q, Q) = \begin{cases} 1 & \text{if } q = h_1^{(1)} + x, \\ 0, & \text{otherwise.} \end{cases}$$

The process above generalizes to the following dispatch policy:

$$f_\ell^{(k)}(q, Q) = \begin{cases} 1 & \text{if } q = h_{k-1}^{(k-1)} + x, \\ 0, & \text{otherwise.} \end{cases}$$

To make sure we yield a simple dispatch policy, we can choose the parameter of the geometric distribution to be fixed. Alternatively, we could set it in a way that depends on the length of the queue. For instance, we can require that the means of the random variables are Q/p. This would still satisfy monotonicity, but it would no longer be queue-length independent.

C.1.2 Dynamic Randomized FIFO

Under the Dynamic Randomized FIFO (DRFIFO) dispatch policy, we (almost) evenly partition the current queue of agents into p "bins". The k^{th} time each item is dispatched, it is sent to an agent in the k^{th} bin, uniformly at random. Formally, given any queue length Q and patience level p, we define

$$z_{Q,p} \triangleq \left\lfloor \frac{Q}{p} \right\rfloor,$$

$$r_{Q,p} \triangleq Q \bmod p.$$

We distribute the remainder $r_{Q,p}$ equally across the first $r_{Q,p}$ bins, so these bins each contains $z_{Q,p}+1$ queue positions, and the rest of the $p - r_{Q,p}$ bins each contain $z_{Q,p}$ queue positions. The set of bins $\{b^{(k)}\}_{k=1,\dots,p}$ are therefore given by

$$b^{(k)} = \{(k-1)(z_{Q,p}+1)+1, (k-1)(z_{Q,p}+1)+2, \dots, k(z_{Q,p}+1)\}, \qquad \text{if } k \le r_{Q,p}, \tag{37}$$

$$\begin{aligned} b^{(k)} = \{&r_{Q,p}(z_{Q,p}+1) + (k - r_{Q,p} - 1)z_{Q,p} + 1, \\ &r_{Q,p}(z_{Q,p}+1) + (k - r_{Q,p} - 1)z_{Q,p} + 2, \dots, \\ &r_{Q,p}(z_{Q,p}+1) + (k - r_{Q,p})z_{Q,p}\}, \end{aligned} \qquad \text{if } k > r_{Q,p}. \tag{38}$$

Definition 4 (Dynamic Randomized FIFO)**.** At any queue length $Q \le \bar{Q}$, the Dynamic Randomized FIFO mechanism divides the queue into p bins according to (37) and (38). For each $k \in [p]$, the k^{th} dispatch is specified by

$$f_\ell^{(k)}(q,Q) = \begin{cases} 1/(z_{Q,p}+1), & \text{if } q \le Q,\ k \le r_{Q,p},\ q \in b^{(k)}, \\ 1/z_{Q,p}, & \text{if } q \le Q,\ r_{Q,p} < k \le p,\ q \in b^{(k)}, \\ 0, & \text{otherwise,} \end{cases}$$

for all $\ell \in \mathcal{L}$.

Note that if $Q \le p$, this dispatch policy simplifies to Strict FIFO.

C.2 Equilibrium under Strict FIFO

Example 6 (Equilibrium under Strict FIFO)**.** Consider the Strict FIFO dispatch policy (Example 1) and an economy where agents arrive at rate $\lambda = 5$ and have waiting cost $c_w = 1/3$ per unit of time. There are three item types, with $w_1 = 75$, $\mu_1 = 1$, $w_2 = 25$, $\mu_2 = 6$, and $w_3 = 15$, $\mu_3 = 3$.

Assume for now that $p = +\infty$, i.e., items do not expire. Consider a strategy where the agent never reneges, accepts only type 1 items at $1 \le q < 150$, accepts type 1 and type 2 items if $150 \le q < 360$, accepts all three item types at $360 \le q < 720$, and joins the queue if and only if the length is at most 720 (after the agent joined). We apply Corollary 1 and show that the strategy forms an MPE.

First observe that under Strict FIFO dispatch, the rate at which the driver at the head of the queue ($q = 1$) receives items of each type does not depend on the length of the queue or the strategy adopted by the rest of the agents. It takes the driver at $q = 1$ an average of $1/\mu_1$ units of time to receive a type 1 item, thus the continuation payoff for $q = 1$ at any $Q \ge 1$ is

$$V(1,Q) = w_1 - c_w/\mu_1 = 74.67.$$

We can verify that (16), (17), (18) and (19) are all satisfied, hence the strategy is best-responding at $q = 1$ for all $Q \ge 1$. Now consider queue positions $1 < q < 150$. These agents do not receive any item they would accept (since all type 1 items are accepted when dispatched to $q = 1$), but they move forward at a rate of μ_1. As a result, it take them $(q-1)/\mu_1$ units of time to reach the head of the queue, at which point they receive $V(1,Q)$. Therefore, regardless of the queue length Q,

$$V(q,Q) = V(1,Q) - c_w(q-1)/\mu_1 = w_1 - cq/\mu_1 = 75 - q/3,\ \forall q < 150.$$

Similarly, we can compute that at $q = 150$, the continuation payoff is exactly $V(q,Q) = 25$, thus (16), (17), (18) and (19) are satisfied for all $q \le 150$. The same exercise can be repeated for all queue positions, verifying that the strategy indeed forms an MPE.

Now consider the more realistic setting where items expire after $p = 12$ dispatches. In this case, we can verify that agents at positions $q < 150$ in the queue are still only unwilling to accept type 1 items. This means that all items of types 2 and 3 are never accepted, leading to the market failure under Strict FIFO dispatch as similarly illustrated in Example 1 in Castro et al. [2021].

C.3 Multiple Equilibria

We provide an example of multiple equilibria. Consider the scenario under the Strict FIFO dispatch policy where the parameters take the following values:

$$w_1 = 1,000,000 + 5, \quad \mu_1 = \frac{1}{1,000,000}, \quad w_2 = 5, \quad \mu_2 = 1,000,000,$$

$$\lambda = 1,000,000, \quad c_w = 1, \quad p = 1, \quad \bar{Q} = 2.$$

Under Strict FIFO, taking item 1 with probability 1 and item 2 with any probability is a best response at the head of the queue. However, the resulting continuation payoff at position $(2, 2)$ can be very different.

Strategy 1. Consider the strategy where, at the head of the queue, $q = 1$, an agent takes both items with probability 1. So, $\alpha_1(1, \cdot) = 1$ and $\alpha_2(1, \cdot) = 1$. The resulting continuation payoff is $V(2, 2) \approx 5$ since there is nearly no waiting time. Agents move swiftly from state $(2, 2)$ to $(1, 1)$.

Strategy 2. Now, consider the alternative strategy $\alpha_1(1, \cdot) = 1$ and $\alpha_2(1, \cdot) = 0$. The waiting cost becomes too high and a driver will never join at $q = 2$ since $V(2, 2) < 0$.

One can verify that both of these strategies satisfy the complementarity conditions (Corollary 1), yet the outcome of each equilibrium is very different. Throughput under strategy 1 is $1,000,000$ while strategy 2 results in a throughput of $1/1,000,000$. Net revenue similarly differs from $5,000,000$ to 5 respectively.

C.4 Impact of Cascading Effects

We provide an example where the "cascading effects" of the focal agent's decision (see Section 3.3 for more details) have a large impact on a state's continuation payoff and, consequently, the gradient-based Algorithm 1 and the simple projection Algorithm 2 yield very different results.

Consider an economy where agents always join, there are only two items, and the maximum queue length is 2. Suppose item 1 is dispatched uniformly at random and item 2 is dispatched via Strict FIFO where the parameters take the following values:

$$w_1 = 2,000,000, \quad \mu_1 = \frac{1}{1,000,000}, \quad w_2 = 500,000, \quad \mu_2 = \frac{1}{1,000,000,000},$$

$$\lambda = \frac{1}{1,000,000}, \quad c_w = 1, \quad p_1 = 1, \quad p_2 = 2, \quad \bar{Q} = 2.$$

Now, suppose that the acceptance strategy is:

$$\alpha_1(1, 1) = 1, \quad \alpha_1(1, 2) = 1, \quad \alpha_1(2, 2) = 0,$$

$$\alpha_2(1, 1) = 0, \quad \alpha_2(1, 2) = 0, \quad \alpha_2(2, 2) = 1.$$

Using the closed-form evaluation in Lemma 1, notice that $V(1, 1) \approx 1,000,000$ since the agent arrival rate λ and the item 2 arrival rate μ_2 are both very small. Similarly, $V(1, 2) \approx 2,000$, since

it takes $2,000,000$ units of time for the agent to get a type 1 item. In the Algorithm 2 update, this suggests that the strategy should increase to $\alpha_2(1,2) = 1$ since the earning $w_2 \gg V(1,2)$.

However, we observe that if the agent declined the item instead, it would then be sent to $q = 2$, where it would be accepted. Thus, the agent's state would immediately change to $(1,1)$ where $w_2 < V(1,1)$, so it is actually better for the agent to decline the item instead. This comparison is captured in the update direction for Algorithm 1, which puts weight on both the terms $V(1,1) - V(1,2)$ and $w_2 - V(1,2)$. Taking the first order condition of eq. (BR) with respect to $\alpha_2(1,2)$ in this way yields a negative direction.

This result is largely due to the fact that $V(q,Q)$ varies by a large amount for $Q = 1$ and $Q = 2$ and that item 2 can be dispatched multiple times. In consequence, the strategy updates in this scenario are heavily reliant on the cascading effects of the focal agent's decision.

D Additional Numerical Results

Contents

D.1 Trip Data 56
D.2 Front-of-Queue Joining 56
D.3 Additional Details of the Computational Experiments in Section 5 58
D.4 Comparison between Algorithms 1 and 2 58

D.1 Trip Data

Figure 7 shows the distribution of earnings and trip rates at Chicago O'Hare used in our computational experiments.

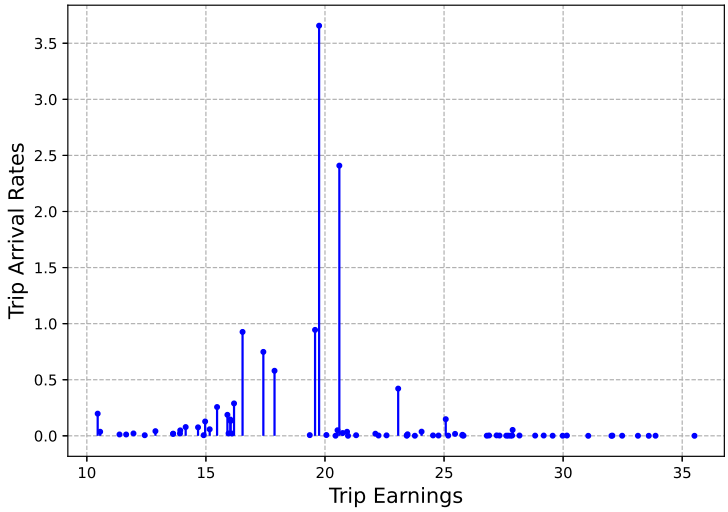


Figure 7: Distribution of earnings and trip rate data at Chicago O'Hare.

D.2 Front-of-Queue Joining

In this section, we extend our original framework to allow new agents to join at the head of the queue upon arrival. The continuation payoff V in this case can still be evaluated by solving a

system of equations rather than using the recursive form in Lemma 1.

Analogous to the family of "absolute priority rules" introduced in Su and Zenios [2004], we model the possibility of joining at the front of the queue with probability κ, and at the end of the queue with probability $(1-\kappa)$. Intuitively, under a Strict FIFO dispatching policy, this can be interpreted as randomly assigning priority to newly arriving agents. This implies that the continuation payoff now includes an additional term representing a transition to state $V(q+1, Q+1)$ occurring at rate $\kappa\lambda$:

$$V(q,Q) = \underbrace{\left(\frac{\sum_{\ell\in\mathcal{L}} \nu_\ell(q,Q)\,\alpha_\ell(q,Q)\,w_\ell}{r(Q)} - \frac{c_w}{r(Q)}\right)}_{\substack{\text{payoff from accepted items}\\ \text{minus waiting cost}}} + \underbrace{\frac{r^+(q,Q)}{r(Q)}\cdot V(q-1,Q-1)}_{\text{an agent in front leaves the queue}}$$
$$+ \frac{\lambda\phi(Q)}{r(Q)}\left(\underbrace{(1-\kappa)\cdot V(q,Q+1)}_{\text{a new agent joins at tail}} + \underbrace{\kappa\cdot V(q+1,Q+1)}_{\text{a new agent joins at head}}\right)$$
$$+ \underbrace{\frac{r^-(q,Q)}{r(Q)}\cdot V(q,Q-1)}_{\text{an agent behind leaves the queue}}, \qquad \forall Q\in[\bar{Q}],\ \forall q\in[Q].$$

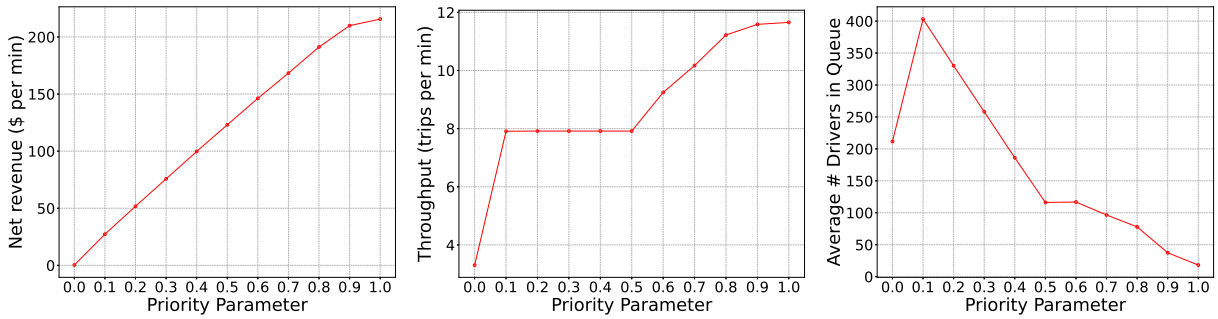
Additionally, we update the joining strategy toward the weighted average $\kappa V(1,Q+1)+(1-\kappa)V(Q+1,Q+1)$. This leads to the following modified fixed-point equation, replacing (15):

$$\phi(Q-1) = P_{[0,1]}\big[\phi(Q-1)+\kappa V(1,Q)+(1-\kappa)V(Q,Q)\big], \qquad \forall Q\in[\bar{Q}],$$

as well as the following update, which replaces line 10 in Algorithms 1 and 2,

$$\phi^{(t+1)}(Q-1) \leftarrow P_{[0,1]}\left[\phi^{(t)}(Q-1)+\gamma^{(t)}\left(\kappa V^{(t)}(1,Q)+(1-\kappa)V^{(t)}(Q,Q)\right)\right], \qquad \forall Q\in[\bar{Q}].$$

Figure 8 reports the equilibrium performance metrics as the priority parameter p varies from 0 (Strict FIFO) to 1 (Strict LIFO), assuming equal agent and job arrival rates of 12, with the same setting in Section 5.



(a) Net revenue. (b) Trip throughput. (c) Expected queue length.

Figure 8: Equilibrium outcomes for $\mu = \lambda = 12$ as the probability of new agents joining at the front changes.

Intuitively, net revenue increases with the priority parameter because agents become less selective as the risk of losing their advantageous position in the queue grows. In our simulation, trip throughput begins to level off when the priority parameter ranges from 0.2 to 0.6. This plateau

primarily stems from discontinuities in the trip-earnings distribution. As illustrated in Figure 7 of Appendix D.1, certain trips have extremely low arrival rates, creating intervals where no trips with the corresponding earnings are observed. Consequently, the continuation payoff for the agent in the $p = 12$th position varies little as κ increases from 0.2 to 0.6, leading to only marginal changes in trip acceptance behavior and, therefore, leaving throughput largely unchanged.

D.3 Additional Details of the Computational Experiments in Section 5

In this subsection, we provide additional details on the numerical experiments presented in Section 5. For simple dispatch policies such as strict FIFO and direct FIFO, we employed Algorithm 2, which is both straightforward to implement and exact for these cases. We used a relatively large step size of 10,000 and observed convergence within at most six iterations for strict FIFO, and within three to four iterations for direct FIFO. (By Theorem 2, any sufficiently large step-size sequence guarantees the convergence of Algorithm 2 to an MPE.)

Since Algorithm 1 is relatively more computationally expensive per iteration, general dispatch policies that require this procedure were initialized with a warm start using an approximate equilibrium strategy obtained upon convergence of Algorithm 2. This provides a better initialization for Algorithm 1, whereas Algorithm 2 was always initialized with a randomly generated strategy profile.

Although we did not follow a formal rule for selecting step sizes, our choices were guided by intuition. As shown in Table 1, Table 2, and Table 3, we report the parameters used to generate the experiments for the dispatch policies discussed in Section 5. We also provide the number of iterations required for the warm start using Algorithm 2, as well as the additional iterations needed for the convergence of Algorithm 1. Broadly, the joining strategy requires a much smaller step size than the acceptance strategy, for which a value of 1 proved sufficient. In contrast, the acceptance strategy involves greater variation in step sizes. While we do not claim these to be optimal choices, they were effective and computationally efficient for (re)producing our reported results.

For Dynamic Randomized FIFO, a large step size performs well, consistent with the intuition that this policy behaves as a hybrid between Strict FIFO and Pure Random policies. For LIFO, a range of step sizes yields reasonable performance. Since the required $\bar{Q}$ is not excessively large, we used a step size of 10 with a momentum parameter, although larger values may further improve performance under certain driver arrival rates. Similarly, for Pure Random, a moderate step size combined with momentum helps mitigate oscillations. In oversupplied settings, where $\bar{Q}$ can be large, increasing the step size for the gradient-based algorithm tends to accelerate convergence. Overall, because the process typically begins from a well-initialized solution obtained from the Algorithm 2's warm start, Algorithm 1 often converges within only a few iterations.

D.4 Comparison between Algorithms 1 and 2

To further compare the performance of Algorithm 1 and Algorithm 2, Table 4 reports their respective performance metrics upon convergence in the O'Hare simulations (see Section 5) under the Pure Random dispatch policy. As reported in Table 1, Algorithm 2 was implemented with a constant step size of 20, while Algorithm 1 used a constant step size of 200. For both algorithms, a momentum parameter of 0.9 was applied to mitigate oscillations and accelerate convergence.

The loss metrics correspond to the strategy obtained from Algorithm 2 upon convergence, evaluated using the loss function associated with Algorithm 1 (see eq. (23) and eq. (24)). The results indicate that, in many cases, the approximate gradient projection algorithm converges to a solution that performs comparably well under the gradient-based criterion. Notably, even when the

λ	**Algorithm 2**			**Algorithm 1**			Momentum	$\bar{Q}$
	Iter.	α step	ϕ step	Iter.	α step	ϕ step		
1	240	20	1	1	200	1	0.9	150
2	93	20	1	1	200	1	0.9	150
3	76	20	1	1	200	1	0.9	150
4	71	20	1	88	200	1	0.9	150
5	82	20	1	1	200	1	0.9	150
6	27	20	1	84	200	1	0.9	150
7	28	20	1	60	200	1	0.9	150
8	39	20	1	1	200	1	0.9	150
9	230	20	1	59	200	1	0.9	200
10	76	20	1	155	200	1	0.9	200
11	110	20	1	1	200	1	0.9	300
12	370	20	1	1	200	1	0.9	800
13	307	20	1	1	200	1	0.9	800
14	445	20	1	1	200	1	0.9	800
15	261	20	1	1	200	1	0.9	800

Table 1: Details of executing Algorithm 1 and Algorithm 2 under Pure Random.

λ	**Algorithm 2**			**Algorithm 1**			Momentum	$\bar{Q}$
	Iter.	α step	ϕ step	Iter.	α step	ϕ step		
1	302	10	1	282	10	1	0.9	200
2	203	10	1	226	10	1	0.9	200
3	41	10	1	1	10	1	0.9	200
4	254	10	1	293	10	1	0.9	200
5	243	10	1	1	10	1	0.9	200
6	45	10	1	1	10	1	0.9	200
7	81	10	1	1	10	1	0.9	200
8	162	10	1	1	10	1	0.9	200
9	93	10	1	1	10	1	0.9	200
10	62	10	1	1	10	1	0.9	200
11	60	10	1	1	10	1	0.9	200
12	110	10	1	1	10	1	0.9	200
13	36	10	1	1	10	1	0.9	200
14	68	10	1	1	10	1	0.9	200
15	69	10	1	1	10	1	0.9	200

Table 2: Details of executing Algorithm 1 and Algorithm 2 under LIFO.

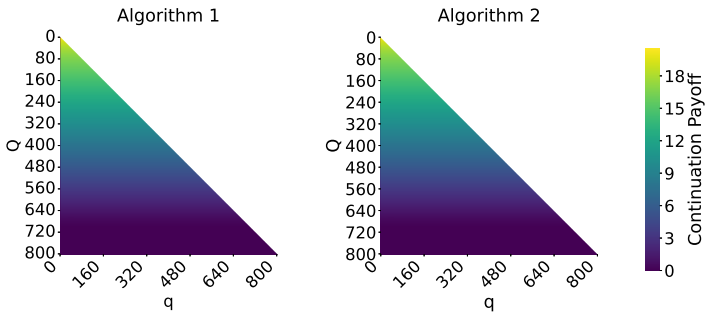
loss slightly exceeds the termination threshold of 10^{-4}, the resulting performance metrics remain nearly indistinguishable.

In Figure 9, we further compare the equilibrium continuation payoffs generated by the two algorithms under random dispatching, assuming equal trip and driver arrival rates ($\mu = \lambda = 12$). The results indicate that both algorithms yield nearly identical continuation payoffs. The payoff remains uniform across agents within the queue and decreases as the queue length grows. The discrepancies between the exact values are minimal, with only slightly greater variation observed at longer queue lengths.

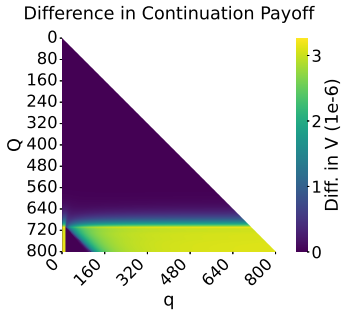
Computation time depends primarily on the chosen value of $\bar{Q}$, which determines the size of the state space. For instance, in oversupplied settings, when $\bar{Q} = 800$, a single iteration of Algorithm 2 takes seconds, while an iteration of Algorithm 1 takes minutes. This difference is smaller in under-supplied settings, although still prevalent.

λ	**Algorithm 2**			**Algorithm 1**			Momentum	$\bar{Q}$
	Iter.	α step	ϕ step	Iter.	α step	ϕ step		
1	12	1000	1	8	1000	1	0	150
2	16	1000	1	10	1000	1	0	150
3	12	1000	1	8	1000	1	0	150
4	12	1000	1	9	1000	1	0	150
5	15	1000	1	6	1000	1	0	150
6	11	1000	1	15	1000	1	0	150
7	11	1000	1	10	1000	1	0	150
8	10	1000	1	6	1000	1	0	150
9	10	1000	1	3	1000	1	0	200
10	10	1000	1	5	1000	1	0	200
11	11	1000	1	4	1000	1	0	300
12	10	1000	1	3	1000	1	0	800
13	12	1000	1	5	1000	1	0	800
14	15	1000	1	13	1000	1	0	800
15	11	1000	1	6	1000	1	0	800

Table 3: Details of executing Algorithm 1 and Algorithm 2 under Dynamic Randomized FIFO.



(a) Equilibrium Continuation Payoffs



(b) Diff. in Continuation Payoffs ($1e^{-6}$)

Figure 9: Heatmap comparison of the equilibrium continuation payoff $V(q, Q)$ obtained upon convergence of both algorithms under the Pure Random dispatch policy, with $\mu = \lambda = 12$.

λ	Net Revenue		Throughput		Exp. Queue Length		Loss Metrics	
	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Alg. 1	Alg. 2	Loss(α)	Loss(ϕ)
1	20.9821	20.9821	1.00	1.00	7.0007	7.0007	0.000093	0.000000
2	41.0665	41.0665	2.00	2.00	8.7480	8.7480	0.000010	0.000000
3	60.6757	60.6757	3.00	3.00	11.5036	11.5036	0.000029	0.000000
4	79.4456	79.4532	4.00	4.00	15.3464	15.3230	0.000111	0.000000
5	98.7306	98.7306	5.00	5.00	16.8344	16.8344	0.000059	0.000000
6	117.9475	118.0080	6.00	6.00	18.3883	18.2097	0.000177	0.000000
7	136.2769	136.2771	7.00	7.00	22.4934	22.4928	0.000586	0.000000
8	147.3070	147.3070	8.00	8.00	47.3845	47.3845	0.000048	0.000000
9	154.8014	154.7994	9.00	9.00	78.0939	78.1000	0.001024	0.000000
10	163.7815	163.7881	10.00	10.00	101.4545	101.4346	0.000477	0.000000
11	169.2664	169.2667	11.00	11.00	133.2069	133.2062	0.000093	0.000000
12	67.0717	67.0717	11.97	11.97	480.0935	480.0935	0.000085	0.000000
13	0.6619	0.6619	12.00	12.00	680.1725	680.1725	0.000076	0.000079
14	0.1178	0.1178	12.00	12.00	681.8046	681.8046	0.000057	0.000094
15	0.0444	0.0444	12.00	12.00	682.0250	682.0250	0.000081	0.000092

Table 4: Comparison of Algorithm 1 and Algorithm 2 under Pure Random for varying driver arrival rates λ. The loss metrics correspond to the strategy produced by Algorithm 2 upon convergence, evaluated using the loss function associated with Algorithm 1 (eq. (23) and eq. (24)).