On-demand Service Sharing via Collective Dynamic Pricing

Mustafa Dogan
University of East Anglia, Norwich, England

Alexandre Jacquillat
Sloan School of Management and Operations Research Center, MIT, Cambridge, MA

Problem definition. This paper studies an on-demand service sharing problem, motivated by emerging operating models in ride-sharing, food delivery and made-to-order manufacturing. Time-sensitive customers arrive dynamically onto a platform with heterogenous willingness to pay and private information. The platform can serve each customer individually or pool customers together, giving rise to interdependencies between customers and over time. This goal is to optimize who to serve, when, and at what price. **Methodology/results.** We formulate a dynamic allocation and pricing mechanism to maximize the platform's expected discounted profits, subject to incentive compatibility and individual rationality constraints. We prove that the problem can be decomposed via dynamic programming, based on the novel notion of *collective virtual value* defined as the marginal revenue that the platform can extract from all customers. The optimal mechanism follows a simple, easily-implementable index rule: service is provided whenever the collective virtual value exceeds a threshold that decreases with the number of available suppliers. **Managerial implications.** Service sharing enables temporal discrimination: the platform provides immediate or delayed services based on customers' own willingness to pay but also on the time of their requests and demand from other customers. In practice, on-demand service sharing can be managed via a dynamic menu to offer differentiated service levels and prices, trading off cost-minimization, demand-supply management, and discriminatory objectives. Our results show that even simple dynamic menus can outperform benchmarks based on posted prices, and can lead to win-win outcomes for the platform and consumers.

Key words: Dynamic mechanism design, On-demand platforms, Service sharing, Non-rival goods.

1. Introduction

The rise of digital platforms has led to new operating models based on on-demand service sharing across service and manufacturing domains. Instead of serving each customer individually, some platforms can provide shared services at no, or moderate, cost increases.

Example 1 (On-demand urban mobility). Ride-pooling operators offer shared trips to customers traveling along similar routes. Similarly, food delivery companies can pool orders to minimize trips to the kitchens or the stores. Shared services create economies of scale by relying on multi-customer routes, but may also increase wait times to enable customer pooling. This tension has historically created difficulties for ride-pooling providers to offer attractive service offers.[1]

[1] see, e.g., https://www.theatlantic.com/technology/archive/2022/07/uberx-share-carpooling-ride-app-cost/661483/, https://www.bloomberg.com/news/articles/2023-05-11/lyft-will-discontinue-pooled-rides-roll-out-new-features

Example 2 (Made-to-order manufacturing). In sectors such as automobile, fashion and home design, made-to-order production allows customization while reducing inventories. These operations need to determine when to process orders to balance short lead times via immediate order processing, versus efficiency and sustainability gains obtained via batching orders.

These examples are all different: shared services may or may not be capacitated; sharing may or may not create inconvenience for customers; shared services may or may not come at extra costs for the platform; etc. Yet, all share two core features: customers request services dynamically, and the operator can serve each customers individually or together. Thus, the platform needs to determine who to serve, when and at what price. These decisions need to balance three objectives: (i) cost minimization (sharing can create economies of scale), (ii) demand-supply management (sharing can save capacity to serve future demand), and (iii) price discrimination (sharing adds a degree of freedom to tailor service offers, wait times and prices across heterogeneous customers).

This paper proposes a dynamic allocation and pricing mechanism for on-demand service sharing. Time-sensitive customers arrive stochastically with heterogeneous willingness to pay and private information. The platform can provide individual or shared services. This creates a trade-off between holding customers in queue to leverage future sharing opportunities versus providing timely services upon customer arrivals—potentially at a higher price. We design the most general direct mechanism in this environment: customers reveal their valuation upon arrival, and the platform optimizes an allocation and payment rule subject to incentive compatibility and individual rationality constraints. This can equivalently be interpreted as the platform offering a dynamic menu of differentiated service offerings and prices, each option being tailored to an agent type

We first derive insights in a two-customer setting (Section 3). This model approximates instances where a small pool of customers can share a service, such as ride-pooling and some last-mile delivery operations. In Section 4, we discuss managerial insights and practical implications; we also study extensions with supply-side restrictions, and with added cost to the platform and added disutility to customers from shared services. We then generalize the problem with an unlimited number of customers and suppliers (Section 5); to retain tractability in a stationary environment, we assume that each service is uncapacitated. This model approximates instances where service can be shared by a larger group of customers, such as some last-mile deliveries, and made-to-order manufacturing.

This environment features a mechanism design problem with perishable non-rival goods and cost externalities. Perishability stems from the on-demand environment and customers' time-sensitivity. Non-rivalry stems from the sharing option: serving a customer does not preclude serving others. Cost externalities reflect the economies of scale resulting from shared services. In this environment, on-demand service sharing creates interdependencies across customers and over time, in that the platform specifies a service offer to any customer contingent on future demand realizations—that is,

on all possible request times and valuations of future customers. In turn, the platform's subsequent decisions need to comply with the level of service promised to earlier customers. To circumvent this technical challenge, we identify structural features of the optimal allocation and pricing rule to decompose the platform's problem into a sequence of sub-problems via dynamic programming.

Our main technical result is that the platform's service decisions are governed by the *collective virtual value*, which we define as the surplus that the platform can extract from all customers present at any time in an incentive-compatible manner. This notion generalizes that of virtual value from Myerson (1981) with perishable non-rival goods. Our result uncovers that the collective virtual value provides a sufficient statistic to capture the demand-side history of the system. In the two-customer setting, we show it via an explicit case analysis; in the more stationary case, we prove that the optimal mechanism can be decomposed via dynamic programming with a state space comprising the collective virtual value and the number of suppliers. This decomposition reveals that the optimal allocation rule follows a simple and easily-implementable index rule: the platform serves all customers with a positive virtual value each time the collective virtual value exceeds a threshold, which decreases with the number of available suppliers. The collective virtual value can potentially be applied to other dynamic allocation environments where time-sensitive agents can be served jointly, such as made-to-order manufacturing, cloud computing or inventory bundling.

Service sharing leads to two opposite effects: it can increase wait times, as the platform can hold customers in queue to pool them with future customers; but it yields free-riding benefits, as customers with moderate willingness to pay would not have been served by themselves but can now receive a shared service. Still, sharing is not blindly leveraged, in that the platform may strategically reject requests from customers with low willingness to pay to charge a higher price to other customers. The optimal mechanism leverages service sharing for temporal discrimination by providing immediate vs. delayed services based on customers' willingness to pay and the system's history. As opposed to classical dynamic pricing in which customers can only be served immediately (Stokey 1979), sharing enables the platform to strategically delay service to some customers until they can be pooled with other customers or until more suppliers become available. In the extension where sharing comes at an extra cost or a smaller utility (Section 4.3.2), the mechanism combines discrimination in terms of service timing and service type (individual vs. shared service).

The inter-customer and inter-temporal dependencies underlying the optimal mechanism give rise to a *collective dynamic pricing* structure. Specifically, the service received by any customer and the corresponding payment do not merely depend on the customer's own willingness to pay, but also on the time of the request and demand from other customers. All else equal, a customer is more likely to receive a service if their own willingness to pay is larger but also if other customers have a higher willingness to pay; moreover, a customer is more likely to receive a service if other

customers have been waiting for a shorter shorter amount of time. Notably, expected wait times and payments depend on the state of the system, leading to non-monotonicities—a customer with a higher valuation can wait longer and/or be charged a lower price than another one.

In practice, the mechanism can be implemented via a menu of service offers dynamically updated when customers and suppliers arrive onto the platform. For instance, the menu could include an immediate service at a high price and several sharing options corresponding to different wait times and prices (e.g., high-priority, medium-priority and low-priority offers). We conduct a comprehensive performance assessment showing that even a discrete approximation of the optimal menu can result in significant profit upsides versus benchmark posted-price mechanisms; moreover, a welfare analysis suggests that it can increase total surplus and even create win-win outcomes for the platform and consumers. Ultimately, this paper suggests that on-demand platforms can leverage service sharing strategically to provide differentiated service guarantees, wait times and prices—balancing cost minimization, demand-supply management and discriminatory objectives.

2. Literature Review

On-demand platforms. An extensive literature studies on-demand platforms (Hu 2019). For instance, Cachon et al. (2017) compared static and dynamic pricing. Taylor (2017) modeled a strategic queuing setting where customers balance prices and wait times. Hu and Zhou (2022) and Chen and Hu (2020) incorporated demand-supply matching into pricing and service. Other balancing mechanisms include spatial pricing (Bimpikis et al. 2019) and vehicle repositioning (Braverman et al. 2019). On-demand operations have also been studied in assemble-to-order manufacturing, albeit without customer incentives and differentiated offers (Xu and Li 2007, Yu et al. 2024).

Several studies have focused on on-demand service sharing in ride-pooling. One body of work developed optimization algorithms to group riders together and assign them to drivers, trading off service quality, in-vehicle detours, and passenger walking (Lobel and Martin 2024, Zhang et al. 2023, Martin et al. 2021). More closely related to our paper, Hu et al. (2020) showed that revenue-maximizing platforms may induce fewer shared rides than welfare-maximizing outcomes. Yan et al. (2020) studied a dynamic waiting mechanism that varies waiting and walking before dispatch. Ke et al. (2020) optimized monopoly pricing for individual and pooling services to avoid the wild good chase phenomenon. Jacob and Roet-Green (2021) designed a two-option menu with distinct prices for individual rides and for delayed or shared rides in a queuing environment with strategic suppliers. Wang and Zhang (2022) jointly optimized posted prices for individual and shared services along with driver wages, based on their synergies. Taylor (2024) generalized that model with network effects and a disutility for sharing and uncovered surprising interaction effects between shared-ride efficiencies, customers' time sensitivities, and labor costs.

Recently, Karaenke et al. (2023) designed ex-post prices for individual and shared services, based on the actual cost of the pools. Similarly, Yan et al. (2024) proposed adaptive prices that vary depending on the ex-post efficiency of the match. As in our setting, Yan et al. (2024) consider time-sensitive customers, heterogeneous willingness to pay, and private information. Our paper characterizes the optimal dynamic mechanism in this environment, which would be implemented via a dynamic menu of individual and shared service options rather than posted prices.

Mechanism design. Our paper relates to mechanism design with heterogeneous, time-sensitive customers and private information. Several studies designed static mechanisms in queuing environments with price- and time-sensitive customers. Afeche (2013) identified strategic delays to prioritize impatient customers and delay patient customers for discriminatory purposes. Afèche and Mendelson (2004) characterized revenue-maximizing and socially optimal mechanisms in a priority auction under a generalized delay cost structure. Katta and Sethuraman (2005) and Afèche and Pavlin (2016) optimized scheduling policies and menus of prices and lead times in a setting with discrete customer types, heterogeneous valuations and heterogeneous delay costs. They found that pooling multiple customer types into a single service class can be optimal to manage lead time differentiation. Maglaras et al. (2017) extended strategic delays to multi-server queues.

Our paper also considers a revenue maximization setting with heterogeneous valuations, customer choice, and service design. In our setting, the latter component involves optimizing the timing of individual versus shared services, as opposed to queue priority classes. Moreover, we propose a dynamic mechanism, which optimizes a probabilistic pricing and allocation rule based on the state of the system. Dynamic mechanism design has been applied to capacity planning (Oh and Özer 2013), online advertising (Balseiro et al. 2021), corporate social responsibility (Wang et al. 2016), carpooling (Amin et al. 2023), etc. In traditional dynamic monopoly pricing, the optimal price path remains constant over time (Stokey 1979). However, the firm can leverage service timing as a discriminatory lever under varying demand (Board 2008), heterogeneous price-sensitive and time-sensitive customers (Besbes and Lobel 2015), and differentiated time preferences (Golrezaei et al. 2018). Dynamic menus of prices and lead-times have been proposed in queuing systems, using an aggregate demand function (Çelik and Maglaras 2008), a two-class model (Ata and Olsen 2013), and a welfare-maximization objective (Akan et al. 2012). Motivated by "wait and save" offerings in ride-sharing, Abhishek et al. (2019) proposed a dynamic menu of prices and wait times to manage demand-capacity imbalances and customer heterogeneity. Our paper identifies service sharing as a new lever to induce temporal discrimination by serving some customers immediately and individually but providing delayed, shared services to others.

Non-rival goods. Service sharing creates a non-rival environment because consumption does not preclude consumption by customers. The management of public goods involves non-rival goods with non-excludability (see, e.g., Samuelson 1954, Bergstrom et al. 1986, Andreoni 1990). In our setting, however, the platform can exclude some customers from the service. Dreze (1980) and Moulin (1994) designed a mechanism to optimize the allocation and production of non-rival goods subject to partial or full exclusion, such as membership clubs, software and cable TV. Maniquet and Sprumont (2004, 2005) addressed issues of fairness in this context. Our problem involves allocating perishable non-rival goods stemming from on-demand services and time-sensitive customers.

3. Model with Two Customers and Unrestricted Supply Capacity

We first study a model with two customers (agents) to isolate the trade-off between the cost-minimization and discrimination. From the revelation principle, it is without loss of generality to consider an incentive-compatible *direct* mechanism, in which each agent reports private information and the platform designs service options for each one (Myerson 1981). This can be interpreted as the platform offering a menu of options designed for each agent type. All proofs are in EC.1.

3.1. Model

Environment. We consider a continuous-time horizon with discount rate $r > 0$. Two agents arrive dynamically onto a platform to request a service. Agent 1 arrives at time 0, and Agent 2 arrives at time $\tau > 0$ following an exponential distribution with rate $\lambda \in \Re_+$. We define the *type* of each agent as their valuation for an immediate service. Both agents are risk neutral and time-sensitive, with decay rate $\delta > 0$. Thus, if Agent $i = 1, 2$ arrives onto the platform at time τ_i with type θ_i and is served at time $t \geq \tau_i$, they derive a discounted utility of $e^{-\delta(t-\tau_i)}\theta_i$ at time t, or $e^{-(r+\delta)(t-\tau_i)}\theta_i$ at time τ_i. Agent types realize from a common continuous distribution $f(\cdot)$ over $[\underline{\theta}, \overline{\theta}]$, with cumulative distribution function $F(\cdot)$. The distribution is publicly known, but each agent's type is private information. We assume that $f(\cdot)$ satisfies the monotone hazard rate condition.

Assumption 1. *The hazard rate function,* $\frac{f(\theta)}{1-F(\theta)}$*, is non-decreasing in* θ*.*

The platform can serve agents individually or together, at cost $c > 0$. The cost-minimization objective creates incentives to hold Agent 1 in queue to provide a shared service, whereas the discriminatory objective creates incentives to provide differentiated services to charge higher prices to high-valuation customers. This section relies on three assumptions: (i) two suppliers are available on the platform at time 0; (ii) service sharing induces no disutility; and (iii) service sharing induces no extra cost for the platform. We relax these assumptions in Section 4.3.

Definition 1 introduces the notion of *virtual value*, defined as the surplus that can be extracted from an agent of type θ in an incentive-compatible mechanism (Myerson 1981). This can also be interpreted it as an agent-specific marginal revenue curve (Bulow and Roberts 1989).

DEFINITION 1. The virtual value of an agent of type θ is given by $\varphi(\theta) = \theta - \frac{1-F(\theta)}{f(\theta)}$

We denote $\theta_0 = \inf\{\theta \in [\underline{\theta}, \overline{\theta}] \mid \varphi(\theta) \geq 0\}$ and $\theta_c = \inf\{\theta \in [\underline{\theta}, \overline{\theta}] \mid \varphi(\theta) \geq c\}$. For clarity, we assume that $\theta_0 \in (\underline{\theta}, \overline{\theta})$ and $\theta_c \in (\underline{\theta}, \overline{\theta})$, so $\varphi(\theta_0) = 0$ and $\varphi(\theta_c) = c$.

REMARK 1. In the absence of the sharing option, an agent of type θ receives an immediate service if $\theta \geq \theta_c$ (at price θ_c) and no service otherwise (Stokey 1979).

Decisions. The platform commits to a mechanism at time $t = 0$, which specifies an allocation and a payment rule to Agent 1 at $t = 0$ and a subsequent allocation and pricing rule for Agent 2 at $t = \tau$ (Figure 1). Since agents are risk neutral, we define an expected payment at the time of arrival—we provide later on an equivalent payment rule satisfying ex post individual rationality.

- Agent 1 reports their type θ_1 at time $t = 0$; the mechanism specifies an expected payment $p_1(\theta_1)$ and a time $T_1(\theta_1)$ when Agent 1 is served individually if Agent 2 has not arrived yet.
- Agent 2 reports their type θ_2 at time $\tau > 0$; the mechanism specifies an expected payment $p_2^\tau(\theta_1, \theta_2)$. If $\tau \geq T_1(\theta_1)$ (i.e., Agent 1 has left), the mechanism specifies $T_2^\tau(\theta_1, \theta_2)$ such that Agent 2 is served at time $\tau + T_2^\tau(\theta_1, \theta_2)$. Otherwise, the mechanism specifies $T_1^\tau(\theta_1, \theta_2)$, $T_2^\tau(\theta_1, \theta_2)$, and $T_{12}^\tau(\theta_1, \theta_2)$, such that both agents are served together at time $\tau + T_{12}^\tau(\theta_1, \theta_2)$, Agent 1 is served individually at time $\tau + T_1^\tau(\theta_1, \theta_2)$, and Agent 2 is served individually at time $\tau + T_2^\tau(\theta_1, \theta_2)$. For consistency, we impose that $T_1^\tau(\theta_1, \theta_2) = \infty$ or $T_{12}^\tau(\theta_1, \theta_2) = \infty$, and that $T_2^\tau(\theta_1, \theta_2) = \infty$ or $T_{12}^\tau(\theta_1, \theta_2) = \infty$ (i.e., each agent can be served at most once).

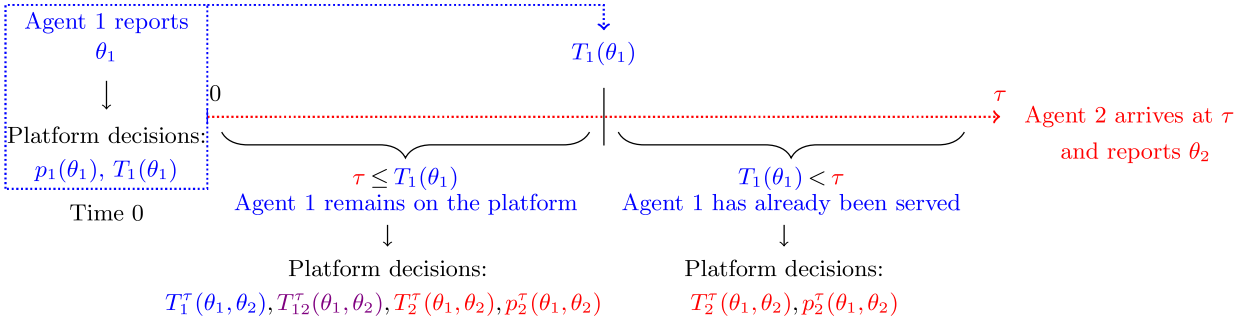


Figure 1 **Sequence of events. Components specific to Agent 1 (resp. Agent 2, both) in blue (resp. red, purple).**

The platform's commitment creates inter-agent and inter-temporal dependencies, because the allocation and pricing rule offered to Agent 1 needs to account for all possible realizations of uncertainty (i.e., arrival time τ and type θ_2). Vice versa, the allocation and pricing rule offered to Agent 2 needs to be consistent with the promises made to Agent 1 at time $t = 0$.

Payoffs. We denote by $U_1(\theta_1)$ the expected discounted payoff of Agent 1 at time $t = 0$, and by $\beta_1(\theta_1)$ the expected discount factor when Agent 1 receives a service. These are given by:

$$U_1(\theta_1) = \theta_1 \beta_1(\theta_1) - p_1(\theta_1), \quad \text{where:} \tag{1}$$

$$\beta_1(\theta_1) = e^{-(r+\delta+\lambda)T_1(\theta_1)} + \int_0^{T_1(\theta_1)} \lambda e^{-\lambda\tau} \int_{\underline{\theta}}^{\overline{\theta}} \left(e^{-(r+\delta)\left(\tau + T_{12}^{\tau}(\theta_1,\theta_2)\right)} + e^{-(r+\delta)\left(\tau + T_1^{\tau}(\theta_1,\theta_2)\right)} \right) f(\theta_2) d\theta_2 d\tau.$$

The first term of $\beta_1(\theta_1)$ captures the cases where Agent 1 is served at $T_1(\theta_1)$ before Agent 2 arrives. The second term captures instances where Agent 2 arrives before $T_1(\theta_1)$; it accounts for whether Agent 1 is served individually at $T_1^{\tau}(\theta_1,\theta_2)$ or together with Agent 2 at $T_{12}^{\tau}(\theta_1,\theta_2)$, while taking the expectation over all possible arrival times $0 \le \tau \le T_1(\theta_1)$ and over all possible types θ_2 of Agent 2.

Similarly, let $U_2^{\tau}(\theta_1,\theta_2)$ denote the expected discounted payoff of Agent 2 at time $t=\tau$, with $\beta_2^{\tau}(\theta_1,\theta_2)$ tracking the expected discount factor at time $T_{12}^{\tau}(\theta_1,\theta_2)$ if Agent 2 receives a shared service or at time $T_2^{\tau}(\theta_1,\theta_2)$ if Agent 2 receives an individual service.

$$U_2^{\tau}(\theta_1,\theta_2) = \theta_2 \beta_2^{\tau}(\theta_1,\theta_2) - p_2^{\tau}(\theta_1,\theta_2), \text{ where } \beta_2^{\tau}(\theta_1,\theta_2) = e^{-(r+\delta)T_{12}^{\tau}(\theta_1,\theta_2)} + e^{-(r+\delta)T_2^{\tau}(\theta_1,\theta_2)} \tag{2}$$

Profits. Let $\Pi^{\tau}(\theta_1)$ denote the platform's expected discounted profit upon Agent 2's arrival. It is defined as the payment minus the discounted cost, taken in expectation over Agent 2's type θ_2:

$$\Pi^{\tau}(\theta_1) = \int_{\underline{\theta}}^{\overline{\theta}} \left[p_2^{\tau}(\theta_1,\theta_2) - c\left(e^{-rT_{12}^{\tau}(\theta_1,\theta_2)} + e^{-rT_1^{\tau}(\theta_1,\theta_2)} + e^{-rT_2^{\tau}(\theta_1,\theta_2)} \right) \right] f(\theta_2) d\theta_2. \tag{3}$$

Let Π be the platform's expected discounted profit at time 0, comprising the revenue collected from Agent 1, the cost of any service provided before τ, and the expected discounted profit at τ. This expression is taken in expectation over Agent 1's type θ_1 at time 0.

$$\Pi = \int_{\underline{\theta}}^{\overline{\theta}} \left[p_1(\theta_1) - c e^{-(r+\lambda)T_1(\theta_1)} + \int_0^{\infty} \lambda e^{-(r+\lambda)\tau} d\tau \Pi^{\tau}(\theta_1) \right] f(\theta_1) d\theta_1. \tag{4}$$

The platform maximizes its expected discounted profit, subject to incentive compatibility and individual rationality constraints. The corresponding problem, referred to as $(\mathcal{P})$, is given by:

$$\max_{\substack{p_1, p_2^{\tau} \\ T_1, T_1^{\tau}, T_2^{\tau}, T_{12}^{\tau}}} \quad \Pi \qquad \text{s.t. } (IC_1),\ (IC_2),\ (IR_1),\ (IR_2), \tag{$\mathcal{P}$}$$

where:
$$\beta_1(\theta_1)\theta_1 - p_1(\theta_1) \ge \beta_1(\theta_1')\theta_1 - p_1(\theta_1'), \quad \forall \theta_1, \theta_1' \in [\underline{\theta}, \overline{\theta}]. \tag{IC_1}$$

$$\beta_2^{\tau}(\theta_1,\theta_2)\theta_2 - p_2^{\tau}(\theta_1,\theta_2) \ge \beta_2^{\tau}(\theta_1,\theta_2')\theta_2 - p_2^{\tau}(\theta_1,\theta_2'), \quad \forall \theta_1, \theta_2, \theta_2' \in [\underline{\theta}, \overline{\theta}],\ \forall \tau > 0. \tag{IC_2}$$

$$\beta_1(\theta_1)\theta_1 - p_1(\theta_1) \ge 0, \quad \forall \theta_1 \in [\underline{\theta}, \overline{\theta}]. \tag{IR_1}$$

$$\beta_2^{\tau}(\theta_1,\theta_2)\theta_2 - p_2^{\tau}(\theta_1,\theta_2) \ge 0, \quad \forall \theta_1, \theta_2 \in [\underline{\theta}, \overline{\theta}],\ \forall \tau > 0. \tag{IR_2}$$

3.2. Optimal Solution

We obtain the monotonicity and envelope conditions analogous to that of Myerson (1981) to eliminate payment terms and reformulate Problem $(\mathcal{P})$, using the virtual value of Agent 1.

LEMMA 1. *Problem* $(\mathcal{P})$ *is equivalent to:*

$$\max_{T_1,T_1^\tau,T_2^\tau,T_{12}^\tau} \quad \widehat{\Pi} = \int_{\underline{\theta}}^{\overline{\theta}} \left(e^{-(r+\lambda)T_1(\theta_1)} \left(e^{-\delta T_1(\theta_1)} \varphi(\theta_1) - c \right) + \int_0^\infty \lambda e^{-(r+\lambda)\tau} \widehat{\Pi}^\tau(\theta_1) \right) f(\theta_1) d\theta_1 \tag{5}$$

$$s.t. \quad \beta_1(\theta_1) \text{ is increasing in } \theta_1 \in [\underline{\theta}, \overline{\theta}] \tag{6}$$

$$\beta_2^\tau(\theta_1,\theta_2) \text{ is increasing in } \theta_2 \in [\underline{\theta}, \overline{\theta}],\ \forall \tau > 0,\ \forall \theta_1 \in [\underline{\theta}, \overline{\theta}], \tag{7}$$

$$\text{where: } \widehat{\Pi}^\tau(\theta_1) = \int_{\underline{\theta}}^{\overline{\theta}} e^{-rT_{12}^\tau(\theta_1,\theta_2)} \left(e^{-\delta\left(\tau + T_{12}^\tau(\theta_1,\theta_2)\right)} \varphi(\theta_1) + e^{-\delta T_{12}^\tau(\theta_1,\theta_2)} \varphi(\theta_2) - c \right) f(\theta_2) d\theta_2 \tag{8}$$

$$+ \int_{\underline{\theta}}^{\overline{\theta}} \left[e^{-rT_1^\tau(\theta_1,\theta_2)} \left(e^{-\delta\left(\tau + T_1^\tau(\theta_1,\theta_2)\right)} \varphi(\theta_1) - c \right) + e^{-rT_2^\tau(\theta_1,\theta_2)} \left(e^{-\delta\left(T_2^\tau(\theta_1,\theta_2)\right)} \varphi(\theta_2) - c \right) \right] f(\theta_2) d\theta_2.$$

Theorem 1 and Corollary 1 elicit the optimal allocation and payment rules. These results show that service is only provided at time $t = 0$ and/or $t = \tau$, so delaying service can only be beneficial to create sharing opportunities. At time $t = 0$, Agent 1 gets served individually if and only if their type exceeds a threshold $\zeta \geq \theta_c$. At time $t = \tau$, the service and Agent 2's payment depend on both agents' types and the elapsed time τ. If $\theta_1 \geq \theta_c$ and τ is small enough (so that $e^{-\delta\tau}\varphi(\theta_1) \geq c$), Agent 1 is guaranteed to be served; this results in a shared service if $\theta_2 \geq \theta_0$ and an individual service otherwise. If $e^{-\delta\tau}\varphi(\theta_1) < c$, both agents are served together if $\theta_1 \geq \theta_0$ and $e^{-\delta\tau}\varphi(\theta_1) + \varphi(\theta_2) \geq c$; Agent 2 gets served individually if $\theta_2 \geq \theta_c$ and $\theta_1 < \theta_0$; and no one is served otherwise.

THEOREM 1. *There exists* $\zeta \geq \theta_c$ *such that the optimal solution to Problem* $(\mathcal{P})$ *satisfies:*

1. *If* $\theta_1 \geq \zeta$, $T_1(\theta_1) = 0$. *For each* $\tau > 0$, $T_2^\tau(\theta_1,\theta_2) = 0$ *if* $\theta_2 \geq \theta_c$, *and* $T_2^\tau(\theta_1,\theta_2) = \infty$ *otherwise.*
2. *If* $\theta_1 < \zeta$, $T_1(\theta_1) = \infty$. *Let* $\phi^\tau = \max\left\{e^{-\delta\tau}\varphi(\theta_1) + \varphi(\theta_2) - c, e^{-\delta\tau}\varphi(\theta_1) - c, \varphi(\theta_2) - c, 0\right\}$. *Then: (i)* $T_{12}^\tau(\theta_1,\theta_2) = 0$ *if* $\phi^\tau = e^{-\delta\tau}\varphi(\theta_1) + \varphi(\theta_2) - c$; *(ii)* $T_1^\tau(\theta_1,\theta_2) = 0$ *if* $\phi^\tau = e^{-\delta\tau}\varphi(\theta_1) - c$; *(iii)* $T_2^\tau(\theta_1,\theta_2) = 0$ *if* $\phi^\tau = \varphi(\theta_2) - c$; *and (iv)* $T_1^\tau(\theta_1,\theta_2) = T_2^\tau(\theta_1,\theta_2) = T_{12}^\tau(\theta_1,\theta_2) = \infty$ *if* $\phi^\tau = 0$.

COROLLARY 1. *The expected payment rule is obtained from the envelope conditions as follows:*

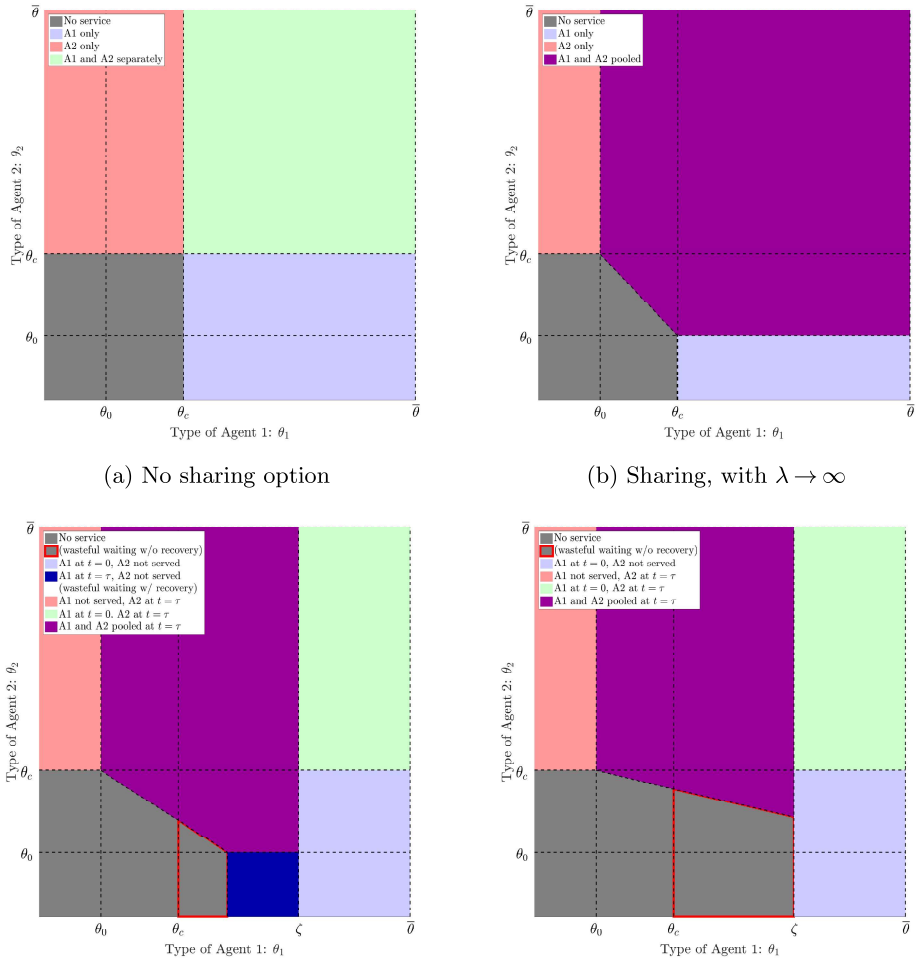
$$p_1(\theta_1) = \beta_1(\theta_1)\theta_1 - \int_{\underline{\theta}}^{\theta_1} \beta_1(\tilde{\theta_1}) d\tilde{\theta_1}, \qquad \forall \theta_1 \in [\underline{\theta}, \overline{\theta}]$$

$$p_2^\tau(\theta_1,\theta_2) = \beta_2^\tau(\theta_1,\theta_2)\theta_2 - \int_{\underline{\theta}}^{\theta_2} \beta_2^\tau(\theta_1,\tilde{\theta_2}) d\tilde{\theta_2}, \qquad \forall \theta_1,\theta_2 \in [\underline{\theta}, \overline{\theta}],\ \forall \tau > 0$$

In Appendix EC.1.3, we compute performance metrics under uniform $f(\cdot)$, which provide visibility into operations (probability of service), consumer surplus, producer surplus, and social welfare.

3.3. Illustration of the Optimal Mechanism, and Managerial Insights

Allocation rule. The optimal allocation rule is depicted in Figure 2 in a θ_1-θ_2 space. Figure 2a shows a benchmark in the absence of sharing, in which each agent i gets served if $\theta_i \geq \theta_c$ (Remark 1). Figure 2b considers simultaneous arrivals ($\lambda \to \infty$). Figures 2c and 2d depict instances with small and large values of τ. Let us describe the optimal allocation rule region by region depending on θ_1:



(a) No sharing option

(b) Sharing, with $\lambda \to \infty$

(c) Sharing, with small τ

(d) Sharing, with large τ

Figure 2 Visualization of the optimal allocation rule.

Region 1. When $\theta_1 < \theta_0$, Agent 1 leaves because the platform cannot extract a positive payment. Similar to the no-sharing scenario, Agent 2 gets served at time τ if and only if $\theta_2 \geq \theta_c$ (at price θ_c).

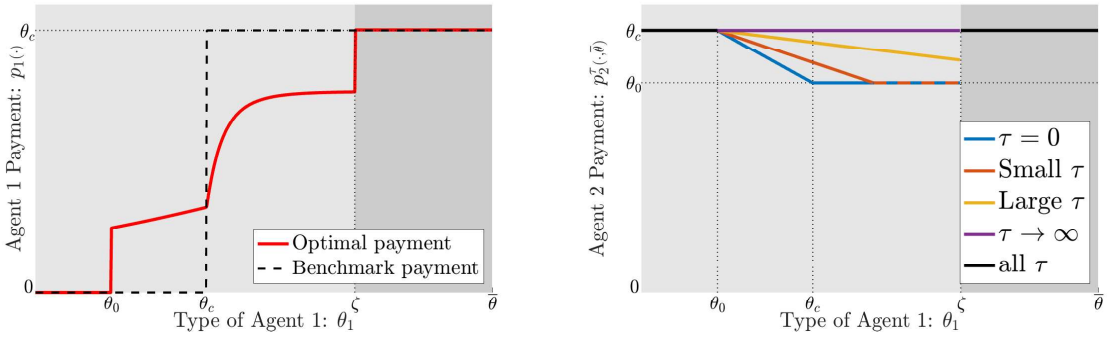
Region 2. When $\theta_0 \leq \theta_1 < \theta_c$, Agent 1 would not have been served individually but is now held in queue. If $\theta_2 \geq \theta_c$, the platform would have served Agent 2 individually but can now serve both requests together to increase revenue. When $\theta_2 < \theta_c$, the platform would have served neither agent but can now provide a profitable shared service if $e^{-\delta\tau}\varphi(\theta_1) + \varphi(\theta_2) \geq c$.

Region 3. When $\theta_c \leq \theta_1 < \zeta$, Agent 1 would have been served by themselves but is again held in queue. When $\theta_2 \geq \theta_c$, the platform can save cost via a shared service rather than two individual services. When $\theta_0 \leq \theta_2 < \theta_c$, the platform would have served Agent 1 only but can now serve both requests together to increase revenue. In that region, sharing may induce *wasteful waiting* when the platform fails to capitalize on the sharing opportunity upon Agent 2's arrival. If Agent 2 arrives shortly after Agent 1 ($e^{-\delta\tau}\varphi(\theta_1) \geq c$) with a low value ($\theta_2 < \theta_0$), Agent 1 will still be served individually at time τ; this is referred to as *wasteful waiting with recovery* in

Figure 2c. As τ grows larger and if θ_2 is small (so that $e^{-\delta\tau}\varphi(\theta_1)+\varphi(\theta_2)<c$), no service is provided at time τ; this is referred to as *wasteful waiting without recovery* in Figures 2c and 2d. As we shall see in Section 5, the wasteful waiting without recovery outcome is specific to the two-agent setting; otherwise, any agent with a non-negative virtual value will get served.

Region 4. When $\theta_1 \geq \zeta$, the discriminatory incentives outweigh the cost-minimization incentives. Agent 1 gets served individually at time 0, and Agent 2 gets served at time τ if and only if $\theta_2 \geq \theta_c$.

Payment rule. Without sharing, the payment rule is a step function: 0 if the valuation is less than θ_c, and θ_c otherwise (Stokey 1979). The payment for Agent 2 at time τ follows a similar step-function structure, since the menu always consists of two options: immediate service (shared or individual) or no service. In contrast, Agent 1's payment at time 0 is not a step function of θ_1, as the menu includes infinitely many options. Figure 3 plots the payment rule for Agent 1 (Figure 3a) and Agent 2 (Figure 3b) as functions of θ_1.



(a) Agent 1 (b) Agent 2 (of type $\overline{\theta}$)

Figure 3 Expected payment of Agent 1 at time $t=0$ and of Agent 2 at time $t=\tau$, as a function of θ_1.

As expected, the payment of Agent 1 is flat when $\theta_1 < \theta_0$ (because they do not get served) and when $\theta_1 \geq \zeta$ (because they get served immediately). In-between, the expected payment increases with $\theta_1 \in [\theta_0, \zeta]$, because a higher valuation increases the likelihood that Agent 1 will receive a delayed service. This increasing payment rule indicates that service sharing induces temporal discrimination, by adjusting the service guarantee and the payment extracted from Agent 1.

In contrast, Agent 2's payment is a step function, because they either get served immediately or not at all (see Figure 2). Specifically, the price charged to Agent 2 is the lowest valuation that justifies a service. When $\theta_1 < \theta_0$ and $\theta_1 \geq \zeta$, Agent 1 has left the platform (Regions 1 or 4) and Agent 2 is thus charged a price of θ_c. Otherwise, when sharing is possible, the price is lower than θ_c. If $e^{-\delta\tau}\varphi(\theta_1) \geq c$, Agent 1 is guaranteed to be served, so Agent 2 will be charged a price of θ_0. In-between, Agent 1's discounted virtual value at time τ is positive but below c, Agent 2 is served if their valuation exceeds a threshold, equal to $\varphi^{-1}(c - e^{-\delta\tau}\varphi(\theta_1))$; that threshold increases with τ and decreases with θ_1, and so does Agent 2's payment.

Managerial insights: collective dynamic pricing. We now use the structure of the allocation and payment rules to derive insights on the management of on-demand shared services.

The platform leverages sharing in several ways. In Region 3, it can save costs by pooling both agents together when both would have been served individually. It can also increase revenue without providing more services, by serving both agents together when otherwise only Agent 1 (Region 3) or Agent 2 (Region 2) would have been served. And it can provide more profitable services altogether by serving both agents together when none would have been served by themselves (Region 2).

Sharing has two opposite effects on customers: wait times vs. service. A negative effect on wait times arises at time 0 in Region 3, as Agent 1 would have been served immediately but is now delayed. A positive effect on service arises when both agents are present at time τ, in which case agents can receive a shared service when they would not have been served by themselves.

The sharing option is not blindly leveraged. The platform may elect to forego the sharing option. For example, the platform only serves Agent 2 at time τ if $\theta_1 < \theta_0$ and $\theta_2 \geq \theta_c$, and it only serves Agent 1 if $e^{-\delta\tau}\varphi(\theta_1) \geq c$ and $\theta_2 < \theta_0$. In these cases, the platform strategically rejects requests for discriminatory purposes, to extract a higher payment from agents with a higher willingness to pay.

Service sharing induces temporal discrimination. In dynamic pricing with commitment, customers are either served immediately or unserved, so time is not used for discrimination in the absence of sharing (Remark 1). The sharing option creates an intermediate category for Agent 1: customers who may receive a delayed service, depending on their valuation and on future demand—with a payment rule increasing in Agent 1's willingness to pay. Thus, service sharing provides an additional discriminatory lever for the platform to differentiate service provision over time.

The mechanism features collective dynamic pricing. The allocation and payment rules exhibit inter-agent and inter-temporal dependencies, giving rise to a collective dynamic pricing structure. All else equal, an agent is more likely to receive a service if their own type is larger but also if the other agent's type is larger and if both agents arrive closer together. Moreover, the payment of Agent 2 increases with their arrival time and decreases with Agent 1's valuation.

In particular, the inter-agent and inter-temporal dependencies suggest that the optimal mechanism cannot be implemented by means of posted prices—fixed, take-it-or-leave-it prices set in advance without personalization. In Section 4.1, we show that the collective dynamic pricing mechanism significantly improves the platform's profit as compared to posted prices.

The allocation rule can be encapsulated with a sufficient statistic, called collective virtual value. We define the collective virtual value $\Phi(t)$ as the maximum surplus that the platform can extract from all agents at time t through an incentive compatible mechanism—extending the notion of

virtual value from Myerson (1981). The collective virtual value decays between agent arrivals, but jumps upward when Agent 2 arrives if their virtual value is positive. It is given by:

$$\Phi(t) = \begin{cases} 0 & \text{if } t < \tau \text{ and } t > T_1(\theta_1), \\ \max\left(e^{-\delta t}\varphi(\theta_1), 0\right) & \text{if } t < \tau \text{ and } t \leq T_1(\theta_1), \\ \max\left(e^{-\delta(t-\tau)}\varphi(\theta_2), 0\right) & \text{if } t \geq \tau \text{ and } t > T_1(\theta_1), \\ \max\left(e^{-\delta t}\varphi(\theta_1) + e^{-\delta(t-\tau)}\varphi(\theta_2), e^{-\delta t}\varphi(\theta_1), e^{-\delta(t-\tau)}\varphi(\theta_2), 0\right) & \text{if } t \geq \tau \text{ and } t \leq T_1(\theta_1). \end{cases} \tag{9}$$

Specifically, the collective virtual value is equal to the sum of the discounted virtual values of all agents on the platform with non-negative individual virtual values. This expression captures the inter-agent dependencies created by service sharing: the platform's revenue opportunities depend on the collective characteristics of the group. Still, the collective virtual value excludes agents with negative individual virtual values, so that low-value agents do not undermine the revenue extracted from higher-value agents. Finally, each agent's contribution is discounted over time to account for customers' time-sensitivity. With the lens of Bulow and Roberts (1989), it can be viewed as the aggregate discounted marginal revenue generated by the group of agents on the platform.

We can reinterpret the optimal mechanism as a time-dependent index rule: the platform provides a service any time collective virtual value exceeds a threshold (equal to $\varphi(\zeta)$ for $t < \tau$ and to c for $t \geq \tau$), and includes the contributing agents. In case 1 of Theorem 1, the virtual value of Agent 1 exceeds the threshold and receives an individual service. In 2.(i), $e^{-\delta\tau}\varphi(\theta_1) \geq 0$, $\varphi(\theta_2) \geq 0$ and $\Phi(\tau) \geq c$, so agents receive a shared service. In 2.(ii), $e^{-\delta\tau}\varphi(\theta_1) \geq c$ but $\varphi(\theta_2) < 0$, so Agent 1 receives an individual service. In 2.(iii), $e^{-\delta\tau}\varphi(\theta_1) < c$ but $\varphi(\theta_2) \geq 0$, so Agent 2 receives an individual service. In 2.(iv), $\Phi(\tau) < c$ and no service is provided. Section 5 generalizes these findings.

4. Assessment, Implementation, and Extensions

We estimate the key performance metrics of the optimal mechanism numerically, using our theoretical results from Section 3. Throughout the numerical analysis, we use a uniform type distribution $f(\cdot)$ to mitigate discretization errors by leveraging intermediate closed-form expressions, and to elicit posted-prices benchmarks in closed form. We use these results to perform a welfare assessment against the posted-prices benchmarks in Section 4.1 and to evaluate the performance of a discrete approximation of the optimal menu in Section 4.2. Finally, in Section 4.3, we extend our two-agent model to derive theoretical results in the presence of stochastic supply and of differentiated agent utilities and costs of service across the individual and shared service options.

4.1. Performance Assessment

We compare the collective dynamic pricing mechanism to three posted-prices benchmarks (we formalize them and derive closed-form solutions in Section 1 of the online supplement):[2]

[2] Available at https://mitsloan-php.s3.amazonaws.com/wp-faculty/sites/136/2025/09/28220917/supplement.pdf

1. A uniform price for individual services only: the platform charges θ_c to each agent.
2. A uniform price for shared services only: the platform charges a fixed price p to each agent. A service is provided if and only if both agents are available and willing to pay p.
3. Differentiated prices for shared and individual services: the platform charges p_I for individual services and p_S for shared services. The shared service is only available if both agents simultaneously opt for it. Unlike our mechanism, posted prices do not create inter-agent dependencies: service is shared if both agents are *independently* willing to pay p_S (Proposition 1).

PROPOSITION 1. *There exist $\chi(p_I, p_S)$ such that Agent 1 purchases an individual service at $t = 0$ if $\theta_1 \geq \chi(p_I, p_S)$. When $\theta_1 < \chi(p_I, p_S)$, (i) Agents 1 and 2 purchase a shared service at time τ if $\theta_1 \geq e^{\delta\tau} p_S$ and $\theta_2 \geq p_S$; (ii) Agent 1 purchases an individual service if $\theta_1 \geq e^{\delta\tau} p_I$ and $\theta_2 < p_S$; and (iii) Agent 2 purchases an individual service if $\theta_1 < e^{\delta\tau} p_S$ and $\theta_2 \geq p_I$.*

Table 1 shows that the shared-only mechanism performs poorly by requiring synergistic arrivals (large θ_1, large θ_2, and small τ). When both c and λ are high, it can pool both customers, but otherwise it induces high profit losses. It is therefore critical for the platform to retain the option to offer individual services. Then, the hybrid posted-price mechanism improves upon the single-option mechanism. The benefits can be significant, especially with a high cost (which amplifies the benefits of sharing) and a high arrival rate (which amplifies the incidence of sharing). Most importantly, the dynamic mechanism developed in this paper can provide profit improvements as compared to posted prices—by up to 12%. The gains are stronger with a smaller arrival rate, which reinforces the impact of tailoring service offers to customers based on their own type and the future events.

The welfare analysis provides additional insights. The optimal mechanism consistently outperforms the individual-only benchmark across all performance metrics, showing that service sharing can generate win-win outcomes: higher profits for the platform and greater utility for customers. Interestingly, the probability of service can be higher for Agent 1 in some instances (e.g., with a small arrival rate and a small cost of service provision, which promote individual services for Agent 1); and it can also be higher for Agent 2 in some other cases (e.g., with a larger arrival rate and a high cost of service provision, which promote shared services). However, the optimal mechanism consistently leads to larger relative improvements in Agent 2's utility than Agent 1's, uncovering free-riding benefits for Agent 2 due to the inter-agent dependencies exploited in our mechanism.

Finally, the optimal mechanism has a disparate impact on both agents as compared to the hybrid mechanism. As noted above, the optimal mechanism can improve the platform's profit by up to 12%. It also uniformly improves Agent 1's service probability and utility by tailoring services to their valuation, resulting in a smaller incidence of wasteful waiting. At the same time, it can deteriorate outcomes for Agent 2, especially with intermediate values of λ and small values of

Table 1 Welfare analysis and performance assessment.

λ	c	Method	Profit	Prob. A1	Prob. A2	Utility A1	Utility A2	Surplus
Low	Low	Individual	(base)	0.375	0.375	(base)	(base)	(base)
		Shared	-51.86%	0.297	0.297	-7.61%	+36.13%	-29.07%
		Hybrid	+1.00%	0.379	0.386	+2.65%	+7.47%	+2.38%
		Optimal	+2.67%	0.405	0.389	+5.20%	+8.70%	+4.12%
Low	Medium	Individual	(base)	0.300	0.300	(base)	(base)	(base)
		Shared	-46.65%	0.231	0.231	+2.79%	+51.81%	-21.72%
		Hybrid	+4.19%	0.308	0.326	+9.10%	+23.70%	+8.32%
		Optimal	+8.60%	0.340	0.333	+16.79%	+27.34%	+13.13%
Low	High	Individual	(base)	0.200	0.200	(base)	(base)	(base)
		Shared	-28.26%	0.158	0.158	+38.33%	+104.85%	+5.50%
		Hybrid	+18.31%	0.208	0.249	+35.50%	+87.36%	+32.88%
		Optimal	+32.19%	0.246	0.264	+63.32%	+97.33%	+48.34%
High	Low	Individual	(base)	0.375	0.375	(base)	(base)	(base)
		Shared	-37.08%	0.333	0.333	+17.78%	+43.71%	-14.46%
		Hybrid	+3.88%	0.394	0.426	+11.90%	+33.61%	+10.03%
		Optimal	+6.59%	0.419	0.420	+13.22%	+28.23%	+11.29%
High	Medium	Individual	(base)	0.300	0.300	(base)	(base)	(base)
		Shared	-26.77%	0.273	0.273	+37.08%	+69.19%	-0.11%
		Hybrid	+15.49%	0.333	0.401	+34.54%	+92.19%	+31.42%
		Optimal	+19.89%	0.362	0.378	+39.72%	+67.24%	+31.06%
High	High	Individual	(base)	0.200	0.200	(base)	(base)	(base)
		Shared	+5.70%	0.203	0.203	+97.89%	+148.45%	+44.91%
		Hybrid	+49.70%	0.245	0.314	+92.09%	+183.10%	+78.97%
		Optimal	+59.21%	0.277	0.299	+118.03%	+160.08%	+85.81%
∞	Low	Individual	(base)	0.375	0.375	(base)	(base)	(base)
		Shared	-29.42%	0.341	0.341	+41.33%	+41.33%	-5.84%
		Hybrid	+16.95%	0.430	0.430	+34.14%	+34.15%	+22.68%
		Optimal	+16.95%	0.430	0.430	+34.14%	+34.15%	+22.68%
∞	Medium	Individual	(base)	0.300	0.300	(base)	(base)	(base)
		Shared	-15.72%	0.285	0.285	+68.75%	+68.75%	+12.44%
		Ind. & Shared	+34.96%	0.381	0.381	+70.13%	+70.18%	+46.69%
		Optimal	+34.96%	0.381	0.381	+70.13%	+70.18%	+46.69%
∞	High	Individual	(base)	0.200	0.200	(base)	(base)	(base)
		Shared	+27.04%	0.218	0.218	+153.74%	+153.74%	+69.27%
		Hybrid	+86.46%	0.306	0.306	+173.48%	+173.66%	+115.50%
		Optimal	+86.46%	0.306	0.306	+173.48%	+173.66%	+115.50%

Parameter values: $r = 0.01$; $\delta = 0.1$; $c \in \{0.25, 0.4, 0.6\}$; $\lambda \in \{0.25, 0.75, +\infty\}$; uniform type distribution.

δ (which strengthen sharing under the posted-price benchmark). The resulting impact on social welfare is generally positive; in some cases with an intermediate value of λ and a small value of δ, the optimal mechanism can result in a slightly smaller surplus than hybrid posted prices (by less than 1% in our experiments); still, it increases total surplus in most cases (by up to 10%).

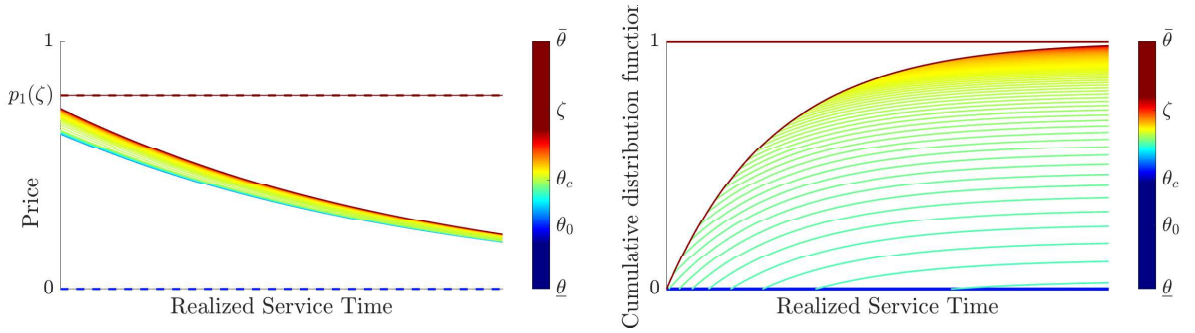
4.2. Practical Implementation

The collective dynamic pricing mechanism can be implemented by means of a dynamic menu specifying a set of service and payment options, depending on the state of the system at the time of the agent's arrival (for Agent 2) and contingent on the future dynamics of the system (for Agent 1).

This dynamic menu departs from the no-sharing benchmark, which would merely rely on a posted price θ_c. Instead, the optimal mechanism relies on an uncountable number of service options for Agent 1 specifying the service rule for each value of τ and each value of θ_2.

Ex post payment rules. Recall that model defined *expected* payments $p_1(\theta_1)$ and $p_2^\tau(\theta_1,\theta_2)$. The mechanism ensures interim individual rationality at the time of reporting, but not ex post individual rationality at the time of service for Agent 1. In practice, it would be undesirable to charge a high price upfront but to then offer no service (if Agent 2's valuation is too low) or a highly delayed service (if Agent 2 arrives too late). Nevertheless, the platform can implement the optimal payment rule without violating ex post individual rationality, for instance by charging a price $\frac{e^{-\delta t}}{\beta_1(\theta_1)} \cdot p_1(\theta_1)$ to Agent 1 when receiving a service at time t. This payment rule does not alter the agent's expected utility and the platform's expected profit; it also guarantees that Agent 1 only makes a payment when served, and provides a discount as a function of the waiting time.

This payment rule is illustrated in Figure 4 along with service times. The figure indicates a price $p_1(\zeta)$ and no delay when $\theta_1 \geq \zeta$, and a price of 0 and an infinite delay when $\theta_1 \leq \theta_0$. In-between, the price and delay of each service become stochastic. Stochastic domination patterns indicate that agents with a higher willingness to pay are more likely to receive a faster service—and more likely to receive a service altogether. Accordingly, the price paid increases with the agent's willingness to pay but decreases with service delay. In other words, the proposed menu tailors service offers and prices to manage customer heterogeneity, while providing a discount based on service delay.



(a) Ex-post prices (b) Cumulative distribution of service time

Figure 4 Ex-post individually rational prices and distribution of service times, as a function of Agent 1's type.

A discretized menu. In practice, it can be challenging to offer a menu with an uncountable number of options. Instead, the collective dynamic pricing mechanism could be approximated via a limited menu. We propose a discretized menu with up to $K+2$ service options for Agent 1: a no-service option at a price of 0, for types $\theta_1 < \theta_0$; an immediate service option at a price $p_1(\zeta)$, if $\zeta < 1$, for types $\theta_1 \geq \zeta$; and K shared-service options corresponding to stochastic options in the

optimal menu for discretized types $\theta_k^D = \theta_0 + \frac{k-1}{K-1} \cdot (\min\{\zeta, 1\} - \theta_0)$ for $k = 1, \cdots, K$. Therefore, the discretized mechanism is simply the submenu: $\left\{(\beta_1(\theta_1), p_1(\theta_1)) : \theta_1 \in \{\underline{\theta}, \theta_1^D, \cdots, \theta_K^D, \overline{\theta}\}\right\}$.

Agent 1 selects the option that maximizes their utility. Specifically, Agent 1 selects item k if and only if $\beta_1(\theta_k^D)\theta_1 - p_1(\theta_k^D) > \beta_1(\theta_{k+1}^D)\theta_1 - p_1(\theta_{k+1}^D)$ and $\beta_1(\theta_k^D)\theta_1 - p_1(\theta_k^D) \geq \beta_1(\theta_{k-1}^D)\theta_1 - p_1(\theta_{k-1}^D)$. Thus, the set $[\theta_0, \min\{\zeta, 1\}]$ can be partitioned into sub-intervals such that Agent 1 chooses sharing option k if and only if $\theta_1 \in [\psi_{k-1}, \psi_k)$, with $\theta_0 = \psi_0 < \psi_1 < \cdots < \psi_{K-1} < \psi_K = \min\{\zeta, 1\}$ such that:

$$\beta_1(\theta_k^D)\psi_k - p_1(\theta_k^D) = \beta_1(\theta_{k+1}^D)\psi_k - p_1(\theta_{k+1}^D), \text{ i.e., } \psi_k = \frac{p_1(\theta_{k+1}^D) - p_1(\theta_k^D)}{\beta_1(\theta_{k+1}^D) - \beta_1(\theta_k^D)}.$$

By design, the options corresponding to a type θ_0 (i.e., the lowest-priority sharing option under the optimal mechanism) and to a type $\min\{\zeta, 1\}$ (i.e., the highest-priority sharing option) are always included in the discretized menu. This ensures that the overall partition remains intact between agents selecting no service, shared services and immediate services. Between θ_0 and $\min\{\zeta, 1\}$, however, the discretized mechanism induces a coarser menu, as illustrated in Figure 5.

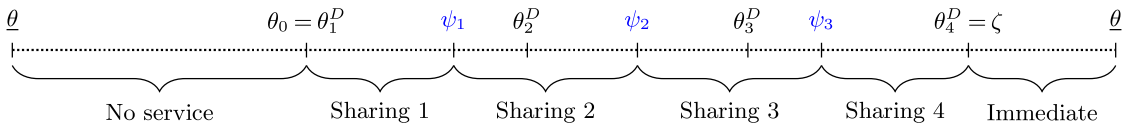
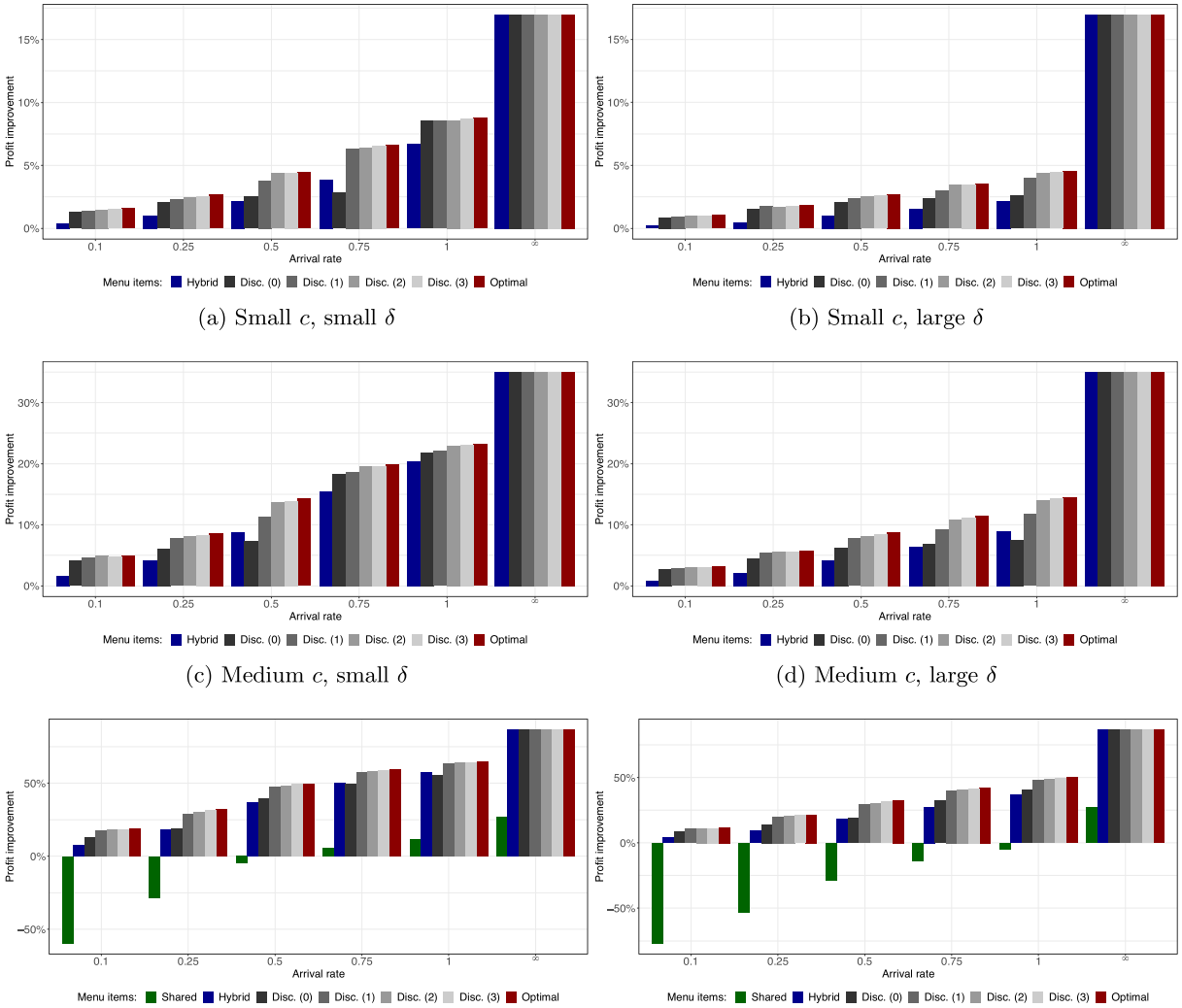


Figure 5 Description of the discretized mechanism ($\zeta < 1$, $K = 4$).

Note that the discretized menu provides a feasible mechanism with up to $K + 2$ service options for Agent 1 that ensures individual rationality (i.e., each agent derives a non-negative utility) and incentive compatibility (i.e., each agent selects the service option that maximizes their own utility). This menu is not guaranteed to be the optimal such mechanism with up to $K + 2$ service options, so the results below can be seen as a conservative characterization of a discretized mechanism.

Figure 6 compares the profits of the optimal mechanism, the discretized menu, and the three benchmarks (Section 4.1). The figure underscores that the discretized menus provide strong approximations of the optimal menu. In most instances, even the sparsest menu with 2 sharing options yields profit improvements as compared to the hybrid posted-price mechanism. Then, menus with 3 or 4 sharing options result in close-to-optimal outcomes across a wide range of instances (different arrival rates, costs of service, and willingness to wait). In fact, the impact of additional sharing options is concave, indicating non-increasing returns as the menu becomes increasingly granular. These observations suggest that the benefits of the optimal mechanism do not merely stem from the continuous menu of options based on all future contingencies at the time of each agent's arrival; rather, most of these gains can be captured via a simple menu featuring a few sharing options.



(a) Small c, small δ

(b) Small c, large δ

(c) Medium c, small δ

(d) Medium c, large δ

(e) Large c, small δ

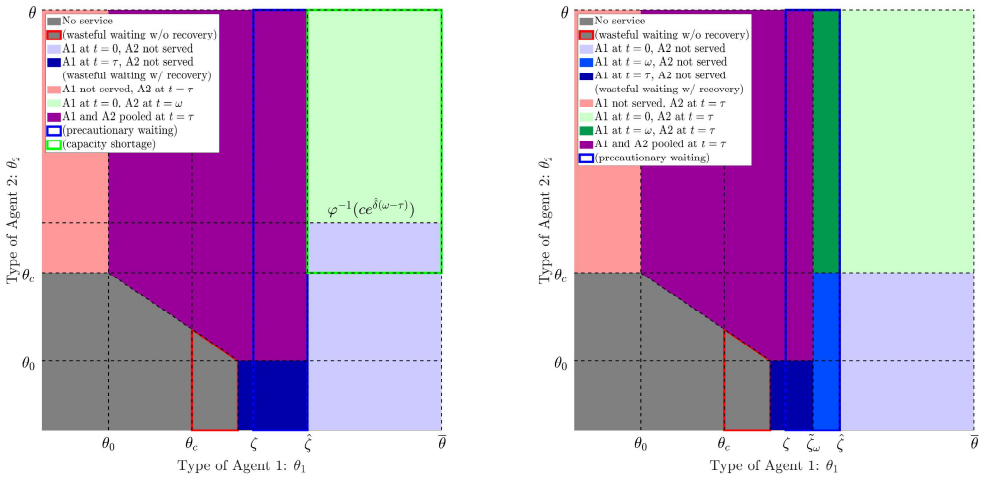
(f) Large c, large δ

Figure 6 Profit comparison. All numbers are relative to the individual-only mechanism. Parameter values: $r = 0.01$**;** $\delta \in \{0.1, 0.2\}$**;** $c \in \{0.25, 0.4, 0.6\}$**;** $\lambda \in \{0.1, 0.25, 0.5, 0.75, +\infty\}$**; uniform type distribution.**

For instance, with $K = 3$, the platform could display a menu upon a customer's request with: (i) an immediate service for \$20; (ii) a high-priority shared service with an expected wait time of 5 minutes for \$15; (iii) a medium-priority shared service with an expected wait time of 10 minutes for \$12; and (iv) a low-priority shared service with an expected wait time of 15 minutes for \$10. Such service differentiation is consistent with some menus available in ride-sharing for example (e.g., priority pickup, wait and save) and could be easily be integrated into common user interfaces. As our results show, even a small menu could lead to significant performance improvements.

4.3. Robustness and extensions

4.3.1. Stochastic supply One supplier is present at time $t = 0$ but a second one arrives at time $\omega > 0$, following a Poisson process with rate $\mu \in \Re_+$. This setting adds demand-supply management objectives to cost-minimization and discrimination. Serving Agent 1 at time 0 creates opportunity costs by creating a potential supply shortage when Agent 2 arrives. This leads to stricter allocation rule at time $t = 0$: Agent 1 is served if θ_1 exceeds a threshold $\hat{\zeta} \geq \zeta$. When $\zeta \leq \theta_1 < \hat{\zeta}$, the platform would have served Agent 1 with sufficient capacity but now delays service to prevent supply shortages, which we refer to as *precautionary waiting*. If the second supplier arrives before Agent 2 (i.e., $\omega < \tau$), the platform may "update" its decision by serving Agent 1 as long as $e^{-\delta\omega}\varphi(\theta_1) \geq \varphi(\zeta)$, reflecting the valuation decay by time ω. The mechanism is formalized in EC.2.1, and the optimal allocation rule is characterized in Theorem EC.1 and Figure 7.

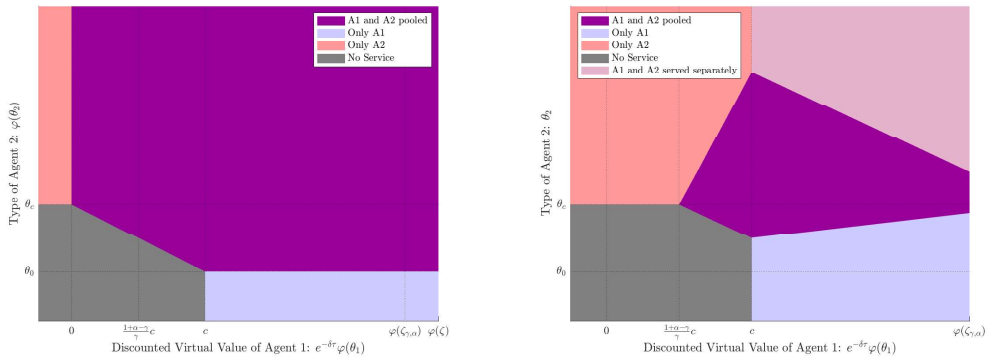


(a) $\tau < \omega$ (b) $\tau > \omega$

Figure 7 Optimal allocation rule with stochastic supply.

This result strengthens our insights regarding the two opposite effects of sharing and the collective dynamic pricing structure. On the one hand, supply restrictions can increase waiting for Agent 1 (if $\zeta \leq \theta_1 < \hat{\zeta}$) and for Agent 2 (in case of a supply shortage). But they may also strengthen sharing opportunities if $\zeta \leq \theta_1 < \hat{\zeta}$ and $\theta_0 \leq \theta_2 < \theta_c$. Similarly, supply-side restrictions enhance the effects of temporal discrimination, since the platform can differentiate service levels more granularly as a function of Agent 1's type. The mechanism can still be implemented via a time-dependent index rule: service is provided if and only if the collective virtual value (Equation (9)) exceeds a threshold. However, the threshold now decreases from $\varphi(\hat{\zeta})$ to $\varphi(\tilde{\zeta}_\omega)$ if $\omega < \tau$, meaning that the platform can implement the mechanism by updating the collective virtual value based on customer demand (to reflect increases and decreases in the marginal revenue) and the service threshold based on supplier availability (to reflect different opportunity costs). We generalize these findings in Section 5.

4.3.2. Disutility from sharing and higher costs for shared services. Service sharing decreases each agent's valuation by a factor $\gamma \in (0,1)$ and increases the platform's cost by a factor $\alpha \in [0,1)$. In ride-pooling, γ and α reflect the longer routes required to serve two customers, and $\alpha < 1$ captures economies of scale. This setting increases the costs of sharing—either the direct cost to the platform, or the indirect cost from the discriminatory objectives due to higher disutilities. In fact, when $\gamma \leq \frac{1+\alpha}{2}$, sharing is never a viable option, and the outcome reduces to the benchmark case with only individual services. The mechanism is formalized in EC.2.2, and the optimal allocation rule is characterized in Theorem EC.2 and Figure 8. At time 0, the threshold for Agent 1 to be served increases as service sharing becomes more attractive (higher γ or lower α). As compared to the baseline setting, service at time τ can feature two individual services, and can exclude Agent 1 even if their discounted virtual value is positive. Specifically: (i) if $e^{-\delta\tau}\varphi(\theta_1) \geq c$, the platform either serves Agent 1 individually, both agents together, or both agents separately; (ii) if $e^{-\delta\tau}\varphi(\theta_1) \in \left[\frac{1+\alpha-\gamma}{\gamma}c, c\right)$, the platform either serves Agent 2 individually, both agents together, or no one; and (iii) if $e^{-\delta\tau}\varphi(\theta_1) < \frac{1+\alpha-\gamma}{\gamma}c$, the platform either serves Agent 2 individually or no one.



(a) Baseline setting: $\gamma = 1$, $\alpha = 0$. (b) Sharing less attractive: $\gamma < 1$ and $\alpha > 0$.

Figure 8 Optimal service decisions at time τ when agent 1 is still on the platform.

This policy shows that the incidence of sharing is not monotonic in Agent 1's discounted virtual value—that is, with their wait time τ or valuation θ_1. If $e^{-\delta\tau}\varphi(\theta_1) \in \left[\frac{1+\alpha-\gamma}{\gamma}c, c\right)$, a higher discounted virtual value leads to more sharing for the platform to extract more revenue. If $e^{-\delta\tau}\varphi(\theta_1) \geq c$, however, the platform extracts a higher payment from an individual service to Agent 1 as their discounted virtual value increases, leading to a lower incidence of sharing (due to the possibility of serving both agents separately). Similarly, the level of service for Agent 2 is not monotonic in Agent 1's discounted virtual value. When $\theta_2 < \theta_c$, a higher discounted virtual value from Agent 1 first induces a shared service rather than no service at all, but then leads the platform to forego Agent 2's request and serve Agent 1 individually. When $\theta_2 > \theta_c$, a higher discounted virtual value

from Agent 1 first induces the platform to provide a shared service, with a negative effect on Agent 2's level of service, but then leads the platform to switch to two individual services.

These results reinforce the inter-agent and inter-temporal dependencies and the effect of temporal discrimination, as well as uncover more complex non-monotonic interactions. In particular, agents feature both complementarity and substitutability, in that service sharing can bring economies of scale to the platform but can also bring lower profits. These interactions enable the platform to design a more complex mechanism that implements both temporal discrimination across agents—by providing timely vs. delayed services—and service type differentiation—by providing individual vs. shared services. In practice, this could therefore lead to a menu that would differentiate between, for instance, an offer with an expected wait time of 5 minutes and a high probability of sharing for \$15, vs. an offer with an expected wait time of 5 minutes and a low probability of sharing for \$18.

5. Generalized Mechanism with Unlimited Numbers of Agents

We generalize collective dynamic pricing to a stationary setting with an unlimited number of agents and suppliers. For tractability, we assume that service sharing induces no disutility and no extra cost; thus, the problem relies on the exact same dynamics as in the two-agent case with customer and supplier arrivals (Sections 3 and 4.3.1). To retain tractability and stationarity, we assume that each service is uncapacitated. Our main result is proved in Appendix A; others are proved in EC.3.

5.1. Mechanism Description

Environment. An unrestricted number of agents (indexed by $i \in \mathbb{N}$) and suppliers (indexed by $j \in \mathbb{N}$) arrive onto the platform at rates $\lambda \in \Re_+$ and $\mu \in \Re_+$, respectively. The platform can serve any number of agents at once, at cost c. For each Agent i, we denote their arrival time by τ_i and their valuation by θ_i. All valuations are independently drawn from distribution $f(\cdot)$. For each supplier j, we denote by ω_j their arrival time. Without loss of generality, we assume that $\tau_1 \leq \tau_2 \leq \cdots$ and $\omega_1 \leq \omega_2 \leq \cdots$. We consider a discount rate $r > 0$ and a valuation decay rate $\delta > 0$.

Let S_t denote the state of the system at time t, in a state space $\mathcal{S}$. We denote by $m(S_t)$ (resp., $n(S_t)$) the number of suppliers (resp., agents) in state $S_t \in \mathcal{S}$. Each state $S_t = (\Omega_t, \Sigma_t, \Theta_t)$ stores:

(i) the arrival times of the suppliers $\Omega_t = \{\omega_1, \cdots \omega_{m(S_t)}\}$ when $m(S_t) \geq 1$ ($\Omega_t = \emptyset$ otherwise);
(ii) the arrival times of the agents $\Sigma_t = \{\tau_1, \cdots \tau_{n(S_t)}\}$ when $n(S_t) \geq 1$ ($\Sigma_t = \emptyset$ otherwise); and
(iii) the types of the agents $\Theta_t = \{\theta_1, \cdots, \theta_{n(S_t)}\}$ when $n(S_t) \geq 1$ ($\Theta_t = \emptyset$ otherwise);

When no agent is present, the arrival times of the suppliers are irrelevant, so we simply denote by $S_\emptyset(m)$ the state with $m \in \mathbb{N}$ suppliers but no agent. Lemma 2 shows that an agent is either included in the earliest service following their arrival or not at all. Indeed, this does not alter their incentives and does not decrease the platform's expected discounted profit (if anything, it may reduce the cost of service provision). We can therefore assume that, whenever a service is provided, all unserved agents leave the platform and the state transitions from S_t to $S_\emptyset(m(S_t) - 1)$.

LEMMA 2. *If a service is provided at time t, it is without loss of generality to assume that the state transitions from S_t to $S_\emptyset(m(S_t)-1)$.*

As in the two-agent setting, services can only be provided when an agent or a supplier arrives. For small $dt > 0$, we denote $t^- = t - dt$ and $t^+ = t + dt$. For $t' > t$, we denote by $\sigma^I_{t',\theta}(S_t) = (\Omega_t, \Sigma_t \cup \{t'\}, \Theta_t \cup \{\theta\})$ and $\sigma^J_{t'}(S_t) = (\Omega_t \cup \{t'\}, \Sigma_t, \Theta_t)$ the state at time t' following the arrival of an agent of type θ and the arrival of a supplier. Per Lemma 2, state transitions are given by:

(1) When an agent of type θ arrives at time t, the state transitions from S_{t^-} to $S_t = \sigma^I_{t,\theta}(S_{t^-})$.
(2) When a supplier arrives at time t, the state transitions from S_{t^-} to $S_t = \sigma^J_t(S_{t^-})$.
(3) When a service is provided at time t, the state transitions from S_t to $S_\emptyset(m(S_t)-1)$.

We denote by $\mathcal{S}_\emptyset$ the subset of the state space $\mathcal{S}$ when no agent is present on the platform; by $\mathcal{S}_I$ the subset of $\mathcal{S}$ at times when an agent arrives onto the platform; and by $\mathcal{S}_J$ the subset of the state space $\mathcal{S}$ when a supplier arrives and at least one agent is present on the platform:

$$\mathcal{S}_\emptyset = \{S_\emptyset(m),\ m \in \mathbb{N}\},$$
$$\mathcal{S}_I = \left\{S_t \in \mathcal{S} \mid \exists S_{t^-} \in \mathcal{S}, \exists \theta \in [\underline{\theta}, \overline{\theta}],\ \text{such that } S_t = \sigma^I_{t,\theta}(S_{t^-})\right\},$$
$$\mathcal{S}_J = \left\{S_t \in \mathcal{S} \mid \exists S_{t^-} \in \mathcal{S} \setminus \mathcal{S}_\emptyset,\ \text{such that } S_t = \sigma^J_t(S_{t^-})\right\}.$$

The mechanism determines an allocation and pricing rule to each agent at the time of arrival, based on the system's history and future evolution. The mechanism needs to incorporate all possible future service decisions, based on the stochastic arrivals of agents and suppliers. Vice versa, the platform's future decisions need to be consistent with the service guaranteed to Agent i. To capture these interdependencies across agents and over time, our proofs rely on mappings that relate any state S_t to all previous states in the history and to all possible future states (Appendix A).

Decisions. Lemma 3 shows that the platform can guarantee a probability of service to each agent based on their type θ_i and on the state of the system at time τ_i (although the time of service depends on the subsequent arrivals of agents and suppliers).

LEMMA 3. *Without loss of generality, an agent of type θ arriving onto the platform at time τ is specified a constant probability $q(S_\tau)$ of being included in the earliest service after time τ.*

Thus, the platform faces an optimal stopping problem. At each time $t = \tau_i$ or $t = \omega_j$, the platform decides whether or not to provide a service, based on the state $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$. Once a service is provided, each Agent i present on the platform will be served with probability $d(S_{\tau_i})$, the system transitions to state $S_\emptyset(m(S_t)-1)$ and the subsequent problem faced by the platform is then equivalent to the original one. As a result, time can be "reset" any time a service is provided and "re-started" when the first subsequent agent arrives. We then index time by 0 whenever the first agent arrives onto the platform following a service provision (that is, $\tau_1 = 0$).

Therefore, the allocation and pricing rule can be characterized by three mappings:

$p(S_t) \in \Re_+$: expected payment when an agent with type θ arrives in state $S_t = \sigma^I_{t,\theta}(S_{t^-}) \in \mathcal{S}_I$

$q(S_t) \in \Re_+$: probability that an agent arriving in state $S_t = \sigma^I_{t,\theta}(S_{t^-}) \in \mathcal{S}_I$ will be served

$d(S_t) \in \Re_+$: probability that the platform provides a service at time t in state $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$

Payoffs. For an incoming agent at time t, we denote by $\beta(S_t)$ the expected discount factor at the time of service, and by $U(S_t) = \beta(S_t)\theta - p(S_t)$ the expected discounted payoff (EC.3.3).

Profit. Let $\Pi(S_t)$ denote the platform's expected discounted *future* profit in state S_t. By definition, this expression captures the payment from the incoming agent, if any, the cost of service provision, if any, and the future expected discounted profits. However, it does not include the payments from earlier agents. It is given as follows for $S_t \in \mathcal{S}_\emptyset \cup \mathcal{S}_I \cup \mathcal{S}_J$.

– When $S_t = S_\emptyset(m) \in \mathcal{S}_\emptyset$, recall that we "re-start" time with the arrival of the first agent. The system transitions to a state of the form $\sigma^I_{0,\theta}(S_t)$ at rate λ, and to $S_\emptyset(m+1)$ at rate μ, so:

$$\Pi(S_\emptyset(m)) = \int_0^\infty \lambda e^{-(r+\lambda+\mu)\tau} \int_{\underline{\theta}}^{\overline{\theta}} \Pi\left(\sigma^I_{0,\theta}(S_\emptyset(m))\right) f(\theta) d\theta d\tau + \int_0^\infty \mu e^{-(r+\lambda+\mu)\omega} \Pi\left(S_\emptyset(m+1)\right) d\omega.$$

– When $S_t \in \mathcal{S}_I$, the expected discounted profit consists of: (i) the expected payment received from the incoming agent, (ii) cost c, if service is provided, and (iii) future profits. If a service is provided, the system transitions to $S_\emptyset(m(S_t)-1)$. Otherwise, it transitions to a state of the form $\sigma^I_{\tau,\theta}(S_t)$ at rate λ, and to a state of the form $\sigma^J_\omega(S_t)$ at rate μ. Hence:

$$\begin{aligned}\Pi(S_t) = {} & p(S_t) + d(S_t)\left[-c + \Pi\left(S_\emptyset(m(S_t)-1)\right)\right] \\ & + (1-d(S_t))\left[\int_t^\infty \lambda e^{-(r+\lambda+\mu)(\tau-t)} \int_{\underline{\theta}}^{\overline{\theta}} \Pi\left(\sigma^I_{\tau,\theta}(S_t)\right) f(\theta) d\theta d\tau + \int_t^\infty \mu e^{-(r+\lambda+\mu)(\omega-t)} \Pi\left(\sigma^J_\omega(S_t)\right) d\omega\right].\end{aligned}$$

– When $S_t \in \mathcal{S}_J$, the profit function is similar but no payment is received. We have:

$$\begin{aligned}\Pi(S_t) = {} & d(S_t)\left[-c + \Pi\left(S_\emptyset(m(S_t)-1)\right)\right] \\ & + (1-d(S_t))\left[\int_t^\infty \lambda e^{-(r+\lambda+\mu)(\tau-t)} \int_{\underline{\theta}}^{\overline{\theta}} \Pi\left(\sigma^I_{\tau,\theta}(S_t)\right) f(\theta) d\theta d\tau + \int_t^\infty \mu e^{-(r+\lambda+\mu)(\omega-t)} \Pi\left(\sigma^J_\omega(S_t)\right) d\omega\right].\end{aligned}$$

At time 0, the platform determines which agents to serve and when, for every possible sequence of arrivals of agents and suppliers and for any number of suppliers $m \in \mathbb{N}$. Specifically, the platform maximizes its expected discounted profit in state $S_\emptyset(m) \in \mathcal{S}_\emptyset$, subject to incentive compatibility and individual rationality constraints in any subsequent state $S_t \in \mathcal{S}_I$.

$$\max_{\boldsymbol{p},\boldsymbol{q},\boldsymbol{d}} \quad \Pi(S_\emptyset(m)) \qquad \text{subject to } (IC_G), (IR_G), \text{ where:}$$

$$\beta\left(\sigma^I_{\tau,\theta}(S_{\tau-})\right)\theta - p\left(\sigma^I_{\tau,\theta}(S_{\tau-})\right) \geq \beta\left(\sigma^I_{\tau,\theta'}(S_{\tau-})\right)\theta - p\left(\sigma^I_{\tau,\theta'}(S_{\tau-})\right), \forall \theta, \theta' \in [\underline{\theta}, \overline{\theta}], \forall \tau \geq 0, \forall S_{\tau-} \in \mathcal{S}, \qquad (IC_G)$$

$$\beta\left(\sigma^I_{\tau,\theta}(S_{\tau-})\right)\theta - p\left(\sigma^I_{\tau,\theta}(S_{\tau-})\right) \geq 0, \quad \forall \theta \in [\underline{\theta}, \overline{\theta}], \forall \tau \geq 0, \forall S_{\tau-} \in \mathcal{S}. \qquad (IR_G)$$

5.2. Problem Decomposition based on the Collective Virtual Value

The main complexity is that the platform's service decision does not only impact the system's future evolution, but also the allocation and pricing rule applied to prior agents. Back to our two-agent example, the service options presented to Agent 1 are contingent on the arrival time and the type of Agent 2; thus, once Agent 2 arrives at time τ, the platform must honor the commitments made to Agent 1 back at time 0. The same dynamics hold in the generalized mechanism: the platform cannot simply optimize $\Pi(S_t)$ in a forward-looking manner but also needs to respect the commitments embedded in the service options offered to earlier agents. The inter-agent and inter-temporal dependencies raise technical challenges because they are not amenable to a direct decomposition; yet, Theorem 2 proves that the problem can be decomposed into a dynamic program.

Lemma 4 in Appendix A shows that agents receive a service with probability 1 or 0, depending on whether their individual virtual value is positive or negative. In other words, an agent is served (at the time of the earliest service provision following their arrival, per Lemma 2) if and only if they contribute to the collective virtual value. Unlike in Section 3, the optimal policy no longer feature wasteful waiting because the platform will *always* be able to capitalize on future sharing opportunities after any agent's arrival. This difference in outcomes arises from the stationary dynamics in the generalized model versus the stochastic termination in the two-agent model.

We now turn to our main result, showing that the platform's stopping decision is entirely governed by: (i) the number of available suppliers, and (ii) the agents' collective virtual value—still defined as the surplus that the platform can extract from all agents in an incentive compatible manner. These variables are sufficient statistics that capture the entire system history—i.e., suppliers' arrivals, agents' arrival times and valuations, and previous level-of-service guarantees.

THEOREM 2. *For each state $S_t \in \mathcal{S}$, let $\Phi(S_t)$ denote the collective virtual value, defined as:*

$$\Phi(S_t) = \sum_{i=1}^{n(S_t)} e^{-\delta(t-\tau_i)} \varphi^+(\theta_i), \quad \textit{where } \varphi^+(\theta) = \max\{\varphi(\theta), 0\}.$$

The problem can be cast as a dynamic program with a state variable comprising the number of suppliers and the collective virtual value. The value function satisfies, for each $m \geq 1, \Phi \geq 0$:

$$V(0,\Phi) = \int_0^\infty \lambda e^{-(r+\lambda+\mu)t} \int_{\underline{\theta}}^{\overline{\theta}} V\left(0, e^{-\delta t}\Phi + \varphi^+(\theta)\right) f(\theta) d\theta dt + \int_0^\infty \mu e^{-(r+\lambda+\mu)t} V\left(1, e^{-\delta t}\Phi\right) dt.$$

$$V(m,\Phi) = \max_d \quad d\left[\Phi - c + V(m-1,0)\right] + (1-d)\left[\int_0^\infty \lambda e^{-(r+\lambda+\mu)t} \int_{\underline{\theta}}^{\overline{\theta}} V\left(m, e^{-\delta t}\Phi + \varphi^+(\theta)\right) f(\theta) d\theta dt + \int_0^\infty \mu e^{-(r+\lambda+\mu)t} V\left(m+1, e^{-\delta t}\Phi\right) dt\right].$$

The optimal policy captures the platform's optimal stopping decision, and is denoted by $d^*(m,\Phi)$.

Specifically, Theorem 2 decomposes the problem into a sequence of sub-problems solved at times of customer and supplier arrivals, while maintaining consistency with level-of-service guarantees over time. Stated differently, Theorem 2 elicits the value function V that the platform maximizes at any time (which differs from $\Pi(S_t)$ due to inter-agent and inter-temporal dependencies). In the Bellman equation, the first term accrues $\Phi - c$ any time a service is provided and encodes the subsequent transition to $S_\emptyset(m-1)$; the second term computes the expected value function given the system transitions and the decay in collective virtual value when a service is not provided.

This reformulation shows that the optimal policy function satisfies $d^*(m,\Phi) \in \{0,1\}$, that is, the platform either provides a service with probability 1 or probability 0. We further express the pricing rule that comes from the allocation policy in Appendix A.

5.3. Characterization of the Optimal Mechanism

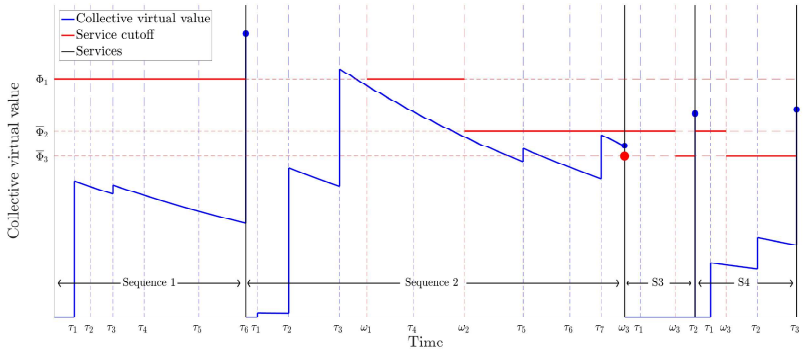
Using the technical results above, Theorem 3 shows that the platform provides a service (to all agents with a non-negative virtual value, per Lemma 4) if and only if the collective virtual value exceeds a cutoff $\overline{\Phi}_m$. Moreover, the cutoff $\overline{\Phi}_m$ is non-increasing in the number of suppliers m.

THEOREM 3. *For $m \in \mathbb{N}$, there exists $\overline{\Phi}_m$ such that $\overline{\Phi}_m \geq \overline{\Phi}_{m+1}$, and: $d^*(m,\Phi) = 1 \iff \Phi \geq \overline{\Phi}_m$.*

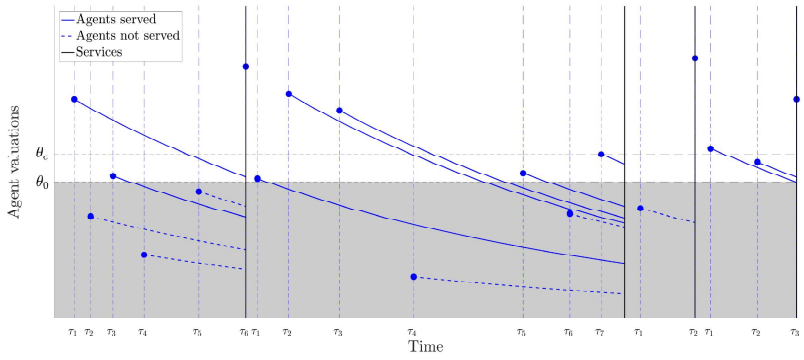
Figure 9 illustrates these dynamics over four sequences, starting with one supplier and zero agent. In Sequence 1, Agents 1, 3 and 6 are served when Agent 6 arrives. Sequence 2 starts with a supply shortage; the platform then waits until three suppliers and seven agents arrive to serve five agents together. Then, the platform serves Agent 2 in Sequence 3 and three agents in Sequence 4. For each sequence, the figure reports the collective virtual value $\Phi(t)$ and the cutoff $\overline{\Phi}_m$ (Figure 9a), the valuation of the agents (Figure 9b) and their expected payments (Figure 9c).

Theorem 3 and Figure 9 extend our insights from the two-agent setting. First, the allocation rule exhibits a simple and easily-implementable structure, which provides a service as soon as the collective virtual value exceeds the cutoff. The collective virtual value decays at rate δ, and exhibits discontinuous jumps when an agent arrives with a type higher than θ_0. Moreover, the cutoff decreases any time a supplier arrives, reflecting the smaller opportunity cost of providing a service (with an infinite number of suppliers, the cutoffs in red would remain constant). Thus, service occurs either when an agent (Sequences 1, 3 and 4) or a supplier (Sequence 2) arrives. The optimal allocation rule exhibits a double monotonic structure: all else equal, the platform is more likely to provide services with more suppliers and with a higher collective virtual value.

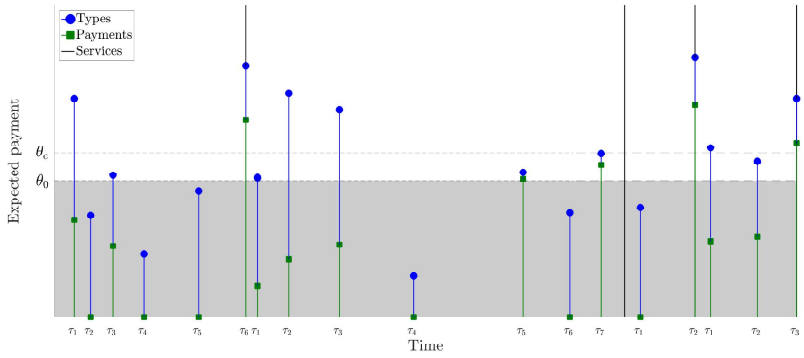
Furthermore, the sharing option still has a negative impact on wait times but a positive impact on service. In Sequence 1, Agent 1 would have received an immediate service in the absence of a sharing option (because $\theta_1 > \theta_c$); however, their valuation remains lower than the cutoff with a single supplier so the platform holds them in queue to pool them with future customers. In contrast,



(a) Collective virtual values and thresholds



(b) Agent valuations over time



(c) Agent types vs. expected payments

Figure 9 Dynamics of service sharing, starting with one supplier and no agent on the platform at time 0.

Agent 3 would not have been served on their own (because $\theta_3 < \theta_c$) but is now included in a shared service because they contribute to the collective virtual value ($\theta_3 > \theta_0$). Still, the sharing option is not blindly leveraged as the platform foregoes requests from other agents with a negative virtual value (e.g., Agents 2, 4 and 5 in Sequence 1) to extract a higher overall revenue.

These dynamics also underscore the collective dynamic pricing structure of the mechanism. Recall that whether or not Agent i receives a service depends only on their own type θ_i. However, the wait time and expected payment do not only depend on the agent's own valuation, but also on

the state of the system at the time of arrival. All else equal, the higher the collective virtual value and/or the more suppliers are present on the platform, the lower the expected wait time and the larger the expected payment. These dynamics give rise to three additional observations:

- Wait times are not monotonic with agent valuations. In Sequence 2, Agent 5 has a lower valuation upon arrival than Agent 2 but receives a service faster, because the collective virtual value has increased and two suppliers have arrived between τ_2 and τ_5.
- The optimal mechanism features allocative inefficiencies: the agents that are served may not be the ones with the highest willingness to pay. In Sequence 1, Agent 3 receives a service because $\theta_3 \geq \theta_0$ but Agent 5 does not because $\theta_5 < \theta_0$. But since Agent 3 arrives earlier than Agent 5, their valuation has decayed by a larger amount when service is provided (at time τ_6), so that $e^{-\delta(\tau_6-\tau_3)}\theta_3 < e^{-\delta(\tau_6-\tau_5)}\theta_5$. As a result, service is provided at time τ_6 to Agent 3 but not to Agent 5 although Agent 3's willingness to pay is lower than Agent 5's at that instant.
- Expected payments are not monotonic with agent valuations. In Sequence 2, Agent 7 has a lower valuation than Agent 2 but their expected payment is higher, because the collective virtual value has increased and two suppliers have arrived between τ_2 and τ_7.

Implementation. As in Section 3, collective dynamic pricing can be implemented via a dynamic menu updated based on the number of available suppliers and the collective virtual value. The menu specifies a set of service and payment options to each incoming customer contingent on all future sequences. As earlier, it can rely on a payment rule that satisfies ex post individual rationality; an agent arriving onto the platform at time t in state S_t with a type $\theta \geq \theta_0$ will be guaranteed to receive a service and will be charged $\frac{e^{-\delta t}}{b(S_\tau)} \cdot p(S_t)$ upon getting served at time $\tau \geq t$. Moreover, the menu can be approximated with a discretized set of options that can ben easily integrated into user interfaces as opposed relying on uncountably many options. Per our results in Section 4.2, this discretized menu can yield close-to-optimal benefits, outperforming posted-prices benchmarks.

6. Conclusion

This paper proposes an allocation and pricing mechanism for on-demand service sharing with heterogeneous, time-sensitive customers and private information. This environment trades off holding customers to provide a shared service versus serving customers immediately at a higher price. These decisions need to balance cost minimization, demand-supply management, and price discrimination objectives. The mechanism determines who to serve, when and at what price. More broadly, this problem can be cast as a mechanism to allocate perishable non-rival goods with cost externalities.

Service sharing creates inter-agent and inter-temporal dependencies: at any point, the platform specifies a service guarantee that is contingent on the future dynamics of the system, while complying with the service guarantees offered to earlier customers. Despite these interdependencies, we

proved that the platform's problem can be decomposed into a dynamic program, using the novel notion of collective virtual value—defined as the revenue that the platform can extract from all customers given incentive compatibility. The optimal mechanism follows an easily-implementable index rule: the platform provides a service each time the collective virtual value exceeds a threshold, which decreases with the number of available suppliers. This result yields several managerial insights. In particular, the platform can leverage service sharing to induce temporal discrimination across heterogeneous customers, by creating service offers with differentiated prices and differentiated wait times. In turn, the service received by any customer depends on their own willingness to pay, but also on their time of arrival and other customers' valuations. We refer to the resulting mechanism as *collective dynamic pricing*. Numerical results showed that this mechanism can provide significant gains for the platform and even increase consumer surplus; moreover, most of these benefits can be captured from discretized menus that can be easily implemented in user interfaces.

This paper opens research avenues on perishable non-rival goods. One question lies in characterizing the optimal mechanism in the stationary environment with infinitely many customers and suppliers (Section 5) where shared services comes with higher costs or lower utilities (as in Section 4.3.2) and with capacity constraints. Another question lies in theoretically analyzing the performance of discretized mechanisms. Still, this paper uncovers opportunities to manage on-demand service sharing via a dynamic menu with differentiated services, wait times and prices.

References

Abhishek V, Dogan M, Jacquillat A (2019) Strategic Timing and Dynamic Pricing in On-demand Platforms. *Working paper* .

Afeche P (2013) Incentive-compatible revenue management in queueing systems: Optimal strategic delay. *Manufacturing & Service Operations Management* 15(3):423–443.

Afèche P, Mendelson H (2004) Pricing and priority auctions in queueing systems with a generalized delay cost structure. *Management science* 50(7):869–882.

Afèche P, Pavlin M (2016) Optimal Price/Lead-time Menus for Queueing Systems with Customer Choice: Segmentation, Pooling, and Strategic Delay. *Management Science* 62(8):2412–2436.

Akan M, Ata Bş, Olsen T (2012) Congestion-based lead-time quotation for heterogenous customers with convex-concave delay costs: Optimality of a cost-balancing policy based on convex hull functions. *Operations research* 60(6):1505–1519.

Amin S, Jaillet P, Pulyassary H, Wu M (2023) Market design for dynamic pricing and pooling in capacitated networks. *arXiv preprint arXiv:2307.03994* .

Andreoni J (1990) Impure altruism and donations to public goods: A theory of warm-glow giving. *The economic journal* 100(401):464–477.

Ata B, Olsen TL (2013) Congestion-based leadtime quotation and pricing for revenue maximization with heterogeneous customers. *Queueing Systems* 73:35–78.

Balseiro SR, Candogan O, Gurkan H (2021) Multistage intermediation in display advertising. *Manufacturing & Service Operations Management* 23(3):714–730.

Bergstrom T, Blume L, Varian H (1986) On the private provision of public goods. *Journal of public economics* 29(1):25–49.

Bertsekas D (2012) *Dynamic Programming and Optimal Control*, volume II (Athena Scientific), 4th edition.

Besbes O, Lobel I (2015) Intertemporal price discrimination: Structure and computation of optimal policies. *Management Science* 61(1):92–110.

Bimpikis K, Candogan O, Saban D (2019) Spatial pricing in ride-sharing networks. *Operations Research* 67(3):744–769.

Board S (2008) Durable-goods monopoly with varying demand. *The Review of Economic Studies* 75(2):391–413.

Braverman A, Dai JG, Liu X, Ying L (2019) Empty-car routing in ridesharing systems. *Operations Research* 67(5):1437–1452.

Bulow J, Roberts J (1989) The simple economics of optimal auctions. *Journal of political economy* 97(5):1060–1090.

Cachon G, Daniels K, Lobel R (2017) The role of surge pricing on a service platform with self-scheduling capacity. *Manufacturing & Service Operations Management* 19(3):368–384.

Çelik S, Maglaras C (2008) Dynamic pricing and lead-time quotation for a multiclass make-to-order queue. *Management Science* 54(6):1132–1146.

Chen Y, Hu M (2020) Pricing and matching with forward-looking buyers and sellers. *Manufacturing & Service Operations Management* 22(4):717–734.

Dreze JH (1980) Public goods with exclusion. *Journal of Public Economics* 13(1):5–24.

Golrezaei N, Nazerzadeh H, Randhawa R (2018) Dynamic pricing for heterogeneous time-sensitive customers. *Manufacturing & Service Operations Management* 22(3):562–581.

Hu M (2019) *Sharing economy: making supply meet demand* (Springer).

Hu M, Wang J, Wen H (2020) Share or solo? individual and social choices in ride-hailing. *Available at SSRN: https://ssrn.com/abstract=3675050* .

Hu M, Zhou Y (2022) Dynamic type matching. *Manufacturing & Service Operations Management* 24(1):125–142.

Jacob J, Roet-Green R (2021) Ride solo or pool: Designing price-service menus for a ride-sharing platform. *European Journal of Operational Research* 295(3):1008–1024.

Karaenke P, Schiffer M, Waldherr S (2023) On the benefits of ex-post pricing for ride-pooling. *Transportation Research Part C: Emerging Technologies* 155:104290.

Katta AK, Sethuraman J (2005) Pricing strategies and service differentiation in queues—a profit maximization perspective, working Paper.

Ke J, Yang H, Li X, Wang H, Ye J (2020) Pricing and equilibrium in on-demand ride-pooling markets. *Transportation Research Part B: Methodological* 139:411–431.

Lobel I, Martin S (2024) Detours in shared rides. *Management Science* .

Maglaras C, Yao J, Zeevi A (2017) Optimal price and delay differentiation in large-scale queueing systems. *Management Science* 64(5):2427–2444.

Maniquet F, Sprumont Y (2004) Fair production and allocation of an excludable nonrival good. *Econometrica* 72(2):627–640.

Maniquet F, Sprumont Y (2005) Welfare egalitarianism in non-rival environments. *Journal of Economic Theory* 120(2):155–174.

Martin S, Taylor SJ, Yan J (2021) Trading flexibility for adoption: Dynamic versus static walking in ridesharing. *Available at SSRN 3984476* .

Moulin H (1994) Serial cost-sharing of excludable public goods. *The Review of Economic Studies* 61(2):305–325.

Myerson R (1981) Optimal auction design. *Mathematics of operations research* 6(1):58–73.

Oh S, Özer Ö (2013) Mechanism design for capacity planning under dynamic evolutions of asymmetric demand forecasts. *Management Science* 59(4):987–1007.

Samuelson PA (1954) The pure theory of public expenditure. *The review of economics and statistics* 387–389.

Stokey N (1979) Intertemporal price discrimination. *The Quarterly Journal of Economics* 355–371.

Taylor T (2017) On-demand service platforms. *Manufacturing & Service Operations Management* in press.

Taylor TA (2024) Shared-ride efficiency of ride-hailing platforms. *Manufacturing & Service Operations Management* 26(5):1945–1961.

Wang S, Sun P, de Véricourt F (2016) Inducing environmental disclosures: A dynamic mechanism design approach. *Operations Research* 64(2):371–389.

Wang X, Zhang R (2022) Carpool services for ride-sharing platforms: Price and welfare implications. *Naval Research Logistics (NRL)* 69(4):550–565.

Xu SH, Li Z (2007) Managing a single-product assemble-to-order system with technology innovations. *Management Science* 53(9):1467–1485.

Yan C, Yan J, Shen Y (2024) Pricing shared rides. *Available at SSRN* .

Yan C, Zhu H, Korolko N, Woodard D (2020) Dynamic pricing and matching in ride-hailing platforms. *Naval Research Logistics (NRL)* 67(8):705–724.

Yu Y, Deng T, Song JS (2024) Conditional lead-time flexibility in an assemble-to-order system. *Manufacturing & Service Operations Management* .

Zhang W, Jacquillat A, Wang K, Wang S (2023) Routing optimization with vehicle–customer coordination. *Management Science* 69(11):6876–6897.

Appendix A: Details on the generalized mechanism, and proof of Theorem 2

Preliminaries. In state $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$ at time t, we denote by $H_p(S_t) \subset \mathcal{S}_I \cup \mathcal{S}_J$ the set of prior states visited between time 0 and time t, and by $H_f(S_t)$ the set of future states that can be visited after time t. Recall that the main difficulty in the generalized mechanism lies in ensuring consistency between the service offer provided at any time and the service promises made on to other agents. Thus, the mappings define any service offer in $S_t \in \mathcal{S}_I$ contingent on all future states in $H_f(S_t)$, while complying with service-level guarantees made in previous states in $H_p(S_t)$. There is a unique sequence of events leading to S_t, so $H_p(S_t)$ is finite, whereas the set $H_f(S_t)$ is uncountable. By definition, $H_p(S_t) \cap H_f(S_t) = \{S_t\}$. They are given by:

$$H_p(S_t) = \{S' \in \mathcal{S}_I \cup \mathcal{S}_J \mid \exists t' \in \Omega_t \cup \Sigma_t \text{ such that } S' = \xi_{t'}(S_t)\},$$
$$H_f(S_t) = \{S' \in \mathcal{S}_I \cup \mathcal{S}_J \mid S' = (\Omega', \Sigma', \Theta') \text{ such that } t \in \Omega' \cup \Sigma' \text{ and } S_t = \xi_t(S')\}.$$

where $\xi_{t'}(S_t)$ denotes the projection of state S_t at time $t' \leq t$:

$$\xi_{t'}(S_t) = (\Omega_t \cap [0,t'], \Sigma_t \cap [0,t'], \{\theta_i \in \Theta_t \mid \tau_i \leq t'\}), \quad \text{for all } S_t = (\Omega_t, \Sigma_t, \Theta_t) \in \mathcal{S}.$$

For $t < t'$ and $S_t, S_{t'} \in \mathcal{S}_I \cup \mathcal{S}_J$, we denote the probability density of reaching state $S_{t'} \in H_f(S_t)$ from state S_t by $h(S_t, S_{t'})$. Unlike $H_p(S_t)$ and $H_f(S_t)$, the mapping $\boldsymbol{h}$ defines a conditional density function that depends on the platform's optimal stopping decision $\boldsymbol{d}$. We define the mapping $\boldsymbol{h}$ recursively:

$$h\left(S_t, \sigma^I_{\tau,\theta}(S_t)\right) = (1 - d(S_t))\lambda e^{-(\lambda+\mu)(\tau-t)} f(\theta), \qquad \forall \tau \geq t, \forall \theta \in [\underline{\theta}, \overline{\theta}],$$
$$h\left(S_t, \sigma^J_{\omega}(S_t)\right) = (1 - d(S_t))\mu e^{-(\lambda+\mu)(\omega-t)}, \qquad \forall \omega \geq t.$$

We can re-write the profit function in each state $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$ as a function of the mapping $\boldsymbol{h}$ and of the platform's stopping decision $\boldsymbol{d}$. Lemma 4 (proved in EC.3.4) expresses $\Pi(S_t)$ by means of virtual value of the incoming agent and future agents, thus eliminating the pricing terms. This lemma also captures the interdependencies between the decisions at different times, by expressing $\Pi(S_t)$ in terms of $\Pi(S_{t'})$'s for $S_{t'} \in H_f(S_t)$—which shows the effect of $d(S_{t'})$ on $\Pi(S_t)$.

Lemma 4. *For every $\tau \geq 0$ and $S_{\tau^-} \in \mathcal{S}$, $\left(\sigma^I_{\tau,\theta}(S_{\tau^-})\right) = 1$ if $\varphi(\theta) \geq 0$, and 0 otherwise. Moreover, defining $\varphi^+(\theta) = \max\{\varphi(\theta), 0\}$, we can rewrite $\Pi(S_t)$ for each $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$ as follows:*

$$\Pi(S_t) = d(S_t)\left[\mathbb{1}(S_t \in \mathcal{S}_I)\varphi^+(\theta_{n(S_t)}) + \Pi(S_\emptyset(m(S_t)-1)) - c\right] \tag{10}$$
$$+ \int_{H_f(S_t)\setminus\{S_t\}} e^{-r(t'-t)} h(S_t, S_{t'}) \left(\Pi(S_{t'}) + d(S_{t'}) \sum_{S_l \in \left(H_f(S_t) \cap H_p(S_{t'^-}) \cap \mathcal{S}_I\right)} e^{-\delta(t'-l)} \varphi^+(\theta_{n(S_l)})\right) dS_{t'}.$$

Proof of Theorem 2. To see the cumulative effects of all stopping decisions $\{d(S_t),\ S_t \in \mathcal{S}_I \cup \mathcal{S}_J\}$, we re-write its objective function as follows, for any given $m \in \mathbb{N}$:

$$\Pi(S_\emptyset(m)) = \int_{\underline{\theta}}^{\overline{\theta}} \Pi(\sigma^I_{0,\theta}(S_\emptyset(m)))f(\theta)d\theta.$$

For $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$, let $\theta_1(S_t)$ be the type of the first agent included in S_t, i.e., the first element in Θ_t, and the unique element in $[\underline{\theta}, \overline{\theta}]$ such that $\sigma^I_{0,\theta_1(S_t)}(S_\emptyset(m)) \in H_p(S_t)$. Denoting $\overline{S}_0 = \sigma^I_{0,\theta_1(S_t)}(S_\emptyset(m))$, we have:

$$\begin{aligned}\Pi(\overline{S}_0) &= d(\overline{S}_0)\left[\varphi^+(\theta) - c + \Pi(S_\emptyset(m(\overline{S}_0)-1))\right] \\ &+ \int_{H_f(\overline{S}_0)\setminus\{\overline{S}_0\}} e^{-rt'} h(\overline{S}_0, S_{t'}) \left(\Pi(S_{t'}) + d(S_{t'})\left(\sum_{S_l \in \left(H_f(\overline{S}_0)\cap H_p(S_{t'-})\cap \mathcal{S}_I\right)} e^{-\delta(t'-l)}\varphi^+(\theta_{n(S_l)})\right)\right) dS_{t'}.\end{aligned} \tag{11}$$

The stopping decision $d(S_t)$ maximizes the following quantity, denoted by $\overline{V}(S_t)$:

$$\overline{V}(S_t) = \Pi(S_t) + d(S_t)\left(\sum_{S_l \in \left(H_f(\overline{S}_0)\cap H_p(S_{t-})\cap \mathcal{S}_I\right)} e^{-\delta(t-l)}\varphi^+(\theta_{n(S_l)})\right)$$

Note that the collective virtual value of the agents present in the platform in state S_t is given by:

$$\Phi(S_t) = \sum_{S_l \in \left(H_f(\overline{S}_0)\cap H_p(S_t)\cap \mathcal{S}_I\right)} e^{-\delta(t-l)}\varphi^+(\theta_{n(S_l)}).$$

We can re-write: $\overline{V}(S_t) = \max_{d\in[0,1]} \quad dX_{stop}(S_t) + (1-d)X_{cont}(S_t)$, where

$$\begin{aligned}X_{stop}(S_t) &= \Phi(S_t) - c + \Pi(S_\emptyset(m(S_t)-1)), \\ X_{cont}(S_t) &= \int_{H_f(S_t)\setminus\{S_t\}} e^{-r(t'-t)} h(S_{t^+}, S_{t'}) \left[\Pi(S_{t'}) + d(S_{t'})\Phi(S_{t'-})\right] dS_{t'}.\end{aligned}$$

First, note that $\overline{V}(S_t)$ is achieved when $d \in \{0,1\}$, so the following holds for each $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$:

$$d(S_t) = \begin{cases} 1 & \text{if } X_{stop}(S_t) \geq X_{cont}(S_t), \\ 0 & \text{if } X_{stop}(S_t) < X_{cont}(S_t). \end{cases}$$

Moreover, $X_{cont}(S_t)$ can also be written recursively as follows by leveraging the definition of $\boldsymbol{h}$:

$$\begin{aligned}X_{cont}(S_t) &= \int_t^\infty \int_{\underline{\theta}}^{\overline{\theta}} \lambda e^{-(r+\lambda+\mu)(\tau-t)} \left[e^{-\delta(\tau-t)} b(\sigma^I_{\tau,\theta}(S_t))\Phi(S_t) + \Pi\left(\sigma^I_{\tau,\theta}(S_t)\right)\right] f(\theta)d\theta d\tau \\ &+ \int_t^\infty \mu e^{-(r+\lambda+\mu)(\omega-t)} \left[e^{-\delta(\omega-t)} b(\sigma^J_\omega(S_t))\Phi(S_t) + \Pi\left(\sigma^J_\omega(S_t)\right)\right] d\omega.\end{aligned}$$

This expression yields a maximization problem that governs the platform's decision at time t in state S_t. This shows that $m(S_t)$ and $\Phi(S_t)$ are sufficient statistics governing the platform's decision. Moreover, this expression provides a dynamic programming decomposition of the platform's problem, which captures the system's history and its future dynamics. We denote by $V(m,\Phi)$ the value function, so $V(m(S_t),\Phi(S_t)) = \overline{V}(S_t)$ for each $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$. The Bellman equation is:

$$V(0,\Phi) = \int_0^\infty \lambda e^{-(r+\lambda+\mu)t} \int_{\underline{\theta}}^{\overline{\theta}} V\left(0, e^{-\delta t}\Phi + \varphi^+(\theta)\right) f(\theta)d\theta dt + \int_0^\infty \mu e^{-(r+\lambda+\mu)t} V\left(1, e^{-\delta t}\Phi\right) dt.$$

$$V(m,\Phi) = \max\Bigg\{\Phi - c + V(m-1,0);$$
$$\int_0^\infty \lambda e^{-(r+\lambda+\mu)t} \int_{\underline{\theta}}^{\overline{\theta}} V\left(m, e^{-\delta t}\Phi + \varphi^+(\theta)\right) f(\theta) d\theta dt + \int_0^\infty \mu e^{-(r+\lambda+\mu)t} V\left(m+1, e^{-\delta t}\Phi\right) dt\Bigg\}.$$

This completes the proof of Theorem 2. □

Payment rule. The optimal policy characterizes the optimal stopping decision $d(S_t) = d^*(m(S_t),\Phi(S_t))$ for all $S_t \in \mathcal{S}_I \cup \mathcal{S}_J$. Similarly, $b(S_t)$ is equal to $b(S_t) = b^*(m(S_t),\Phi(S_t))$, where:

$$\text{if } d^*(m,\Phi)=1: \qquad b^*(m,\Phi) = 1$$
$$\text{if } d^*(m,\Phi)=0: \qquad b^*(m,\Phi) = \int_0^\infty \lambda e^{-(r+\lambda+\mu+\delta)t} \int_{\underline{\theta}}^{\overline{\theta}} b^*\left(m, e^{-\delta t}\Phi + \varphi^+(\theta)\right) f(\theta) d\theta dt$$
$$+ \int_0^\infty \mu e^{-(r+\lambda+\mu+\delta)t} b^*\left(m+1, e^{-\delta t}\Phi\right)$$

Last, the allocation rule and discount factor yield the payment rule using the envelope condition.

Corollary 2. *The payment of a θ-type agent arriving at time t in state S_{t^-} satisfies:*

$$p(\sigma^I_{t,\theta}(S_{t^-})) = \begin{cases} 0 & \text{if } \theta < \theta_0, \\ b^*\left(m(S_{t^-}), \Phi(S_{t^-}) + \varphi(\theta)\right)\theta - \int_{\underline{\theta}}^{\theta} b^*\left(m(S_{t^-}), \Phi(S_{t^-}) + \varphi^+(\tilde{\theta})\right) d\tilde{\theta} & \text{if } \theta \geq \theta_0. \end{cases}$$