BUILD-TO-ORDER:

ENDOGENOUS SUPPLY IN CENTRALIZED MECHANISMS*

Andrew Ferdowsian†

Kwok Hao Lee‡

Luther Yap§

This version: June 30, 2026

How should a planner choose the composition of scarce goods when an application both claims a good and reports the applicant's private type? We study a dynamic matching model in which agents prefer one of several good types, choose queues, and receive goods by lottery within queues. The planner controls supply over time, minimizing mismatch and unassignment. In Markovian lottery mechanisms, responsive supply creates an incentive problem: supplying only the currently over-demanded good encourages agents to join the popular queue to avoid waiting. The optimal mechanism in this class therefore sometimes supplies the under-demanded type, despite risking temporary unassignment. Batching applications increases market thickness, helping the planner equalize expected waiting times across queues.

Keywords: Dynamic Matching, Market Design, Waiting Lists.

JEL: C72, C73, D47, D61, D82.

*We are especially grateful to Leeat Yariv for her advice and support. We also thank Adam Kapor, Alan Wei, Alessandro Lizzeri, Can Urgun, Daniel McGee, Erez Yoeli, Joseph Ruggiero, Kate Ho, Moshe Hoffman, Neil Thakral, Nick Arnosti, Pietro Ortoleva, Qiang Fu, Sylvain Chassang, Teck-Yong Tan, Thomas Gresik, Tim Wang, and audiences at Princeton University, Stony Brook University, National University of Singapore, University of Texas: Dallas, and the EEA for their helpful suggestions and feedback. Last, we acknowledge the financial support of the Dietrich Economic Theory Center and the Princeton Economics Department.

†University of Notre Dame, aferdows@nd.edu

‡Corresponding Author—National University of Singapore: Strategy and Policy, kwokhao@nus.edu.sg

§National University of Singapore: Economics, lyap@nus.edu.sg

1 Motivation

When scarce goods cannot be allocated by market-clearing prices, institutions often ration goods by lottery. Much attention has been placed on how best to allocate a limited supply of goods,[1] but an equally important question is *what type* of goods to offer in the first place. Public housing authorities must choose where to locate new projects; schools must allocate funding across programs such as bilingual or special education; and food banks choose which foods to procure. Many such bodies allocate by lottery for fairness (public housing in Singapore, Turkey, and Amsterdam; special-education seats; the diversity-visa green-card lottery; etc.). In choosing what to offer, a designer must both minimize waste and match agents to the goods they want. How should the designer determine the *types* of good provided in a dynamic environment? Applications play three roles: they determine who competes for current goods, convey private information, and determine the market state to which the committed supply rule responds. Because agents trade off applying for a desired good against expected wait time, queue choices become strategic. Through the lens of an endogenous supply lottery model, we demonstrate two insights. First, because queue choices are strategic, the planner may optimally supply some goods with low current demand in order to keep applications informative. Second, batching allocation cycles thickens the applicant pool before supply decisions are made, helping the planner choose a better mix of goods.

In our model, at the beginning of each period, the designer observes applications from previous periods, then allocates new goods between potential types. Simultaneously, an agent arrives with a private type, corresponding to the good that the agent prefers. Newly arrived agents select one type of good to apply for. Unlike standard mechanism design, where the designer observes types within a truthful mechanism, in line with real-world lotteries, our designer only observes applications to queues. When a queue has more applicants than available goods, available goods of that type must be *randomly* allocated among agents in that queue. We assume that first-in-first-out (FIFO) rationing is infeasible, either due to policy constraints or limited consumer commitment.[2]

[1] See, e.g., Akbarpour, Li, and Gharan (2020), Arnosti and Shi (2020), and Baccara, Lee, and Yariv (2020).

[2] In many settings, FIFO is infeasible because of procedural fairness (e.g., to avoid prioritizing agents with flexible time), having to implement a complex system of priorities, or limited consumer commitment giving rise to oversupply.

If an agent is allocated a good, both exit the market. Otherwise, the agent pays a waiting cost and applies again in the next period. The designer aims to minimize a convex combination of two types of inefficiency: allocation and unassignment. Allocation inefficiency captures the designer's desire to match goods to agent of the same type. Unassignment inefficiency reflects the designer's aim to minimize the number of agents that remain unmatched in a given period.[3]

We characterize the designer's optimal mechanism when supply is endogenous. We have two key results. We first show that supplying underdemanded goods, i.e., those less represented in observed queues of agents, is crucial to ensure incoming agents apply sincerely. Our second key result is that higher levels of demand can improve allocative efficiency when supply is endogenous. In contrast, under a balanced fixed-supply benchmark, allocative efficiency is reduced when demand increases. We show that the second result implies that batching multiple applications together reduces the aggregate uncertainty households face, thereby increasing their willingness to apply sincerely, improving the quality of matches.

Why should the planner supply underdemanded goods? Waiting makes applications strategic: an agent may apply for a less preferred good today rather than wait for her preferred option. Under exogenous supply, one queue could become so long that agents of that type switch to the underdemanded good. Under endogenous supply, the binding constraint runs the other way: the planner is tempted to supply only the high-demand good, which is certain to be taken up, but doing so makes agents who prefer rarely supplied goods join the popular queue. The designer trades off responding to observed queue imbalances today against keeping future applications informative.[4]

Our next contribution is to add to a growing literature on thickness in dynamic markets by characterizing when batching application cycles is optimal. In Section 4 we show that, under endogenous supply, responsive mechanisms work better when competition is higher.[5] Greater

[3] Allocation inefficiency is standard in the literature; and unassigned goods are expensive to maintain, while necessarily imposing waiting costs on agents.

[4] This logic is Coasean in spirit, although our environment is not a durable-goods monopoly: current choices depend on expectations of the designer's future actions. Here, the anticipated future action is not a price cut but a supply response, and the distortion is not delayed purchase but the loss of informativeness in queue choices.

[5] A responsive mechanism is one where different preferences of incoming households lead to different allocations, i.e., the designer responds to preferences.

competition lets the designer equalize expected wait times across queues and so makes agents more willing to apply sincerely. Under a fixed-supply benchmark, the opposite conclusion holds: competition inflates wait times and worsens efficiency. We use this insight to show that the designer can benefit from generating thickness by batching applications.

The same trade-offs arise in many settings where a planner chooses the supply of a good with incomplete preference information. In course allocation, electives are substitutable and faculty can be reassigned, so schools decide year over year which electives to offer and how many seats to cap; Budish and Cantillon (2012) show that students strategically report preferences in such mechanisms. In food assistance, Prendergast (2016) and Altmann (2024) study Feeding America's points-based market, through which food banks express relative preferences across food types and the charity learns which types to solicit.[6]

Related Literature

A large literature designs mechanisms to allocate scarce resources, from kidney exchange to school choice, but typically takes the supply of those resources as fixed: for instance, a kidney-allocation designer cannot choose the blood types of the organs entering the system. We study the complementary problem in which a central agency controls both supply and allocation (for instance, what types of apartments are built and how they are assigned) and show the optimal mechanism and its comparative statics differ from the setting of exogenous supply.

This paper speaks to the theoretical literature on matching and queuing, much of which stems from optimal student assignment to schools (Abdulkadiroğlu and Sönmez 2003).[7] We are most closely related to work on optimal dynamic matching, in which agents are "born" in sequence and trade off taking their best option at birth against waiting for a better match (Baccara, Lee, and Yariv 2020). Akbarpour, Li, and Gharan (2020) shows that a designer may prefer to build market thickness over matching myopically, an insight we show carries over to endogenous supply. The

[6] We note Feeding America uses a virtual points budget rather than monetary willingness to pay; hence, it is not directly applicable to settings where allocation by willingness to pay is ruled out on equity grounds.

[7] See Abdulkadiroğlu and Sönmez (2013) for a survey.

queuing literature similarly studies a fixed supply of goods (Agarwal et al. 2019; Shi 2022), and Thakral (2019) shows an ex-post efficient mechanism may fail to exist when supply is stochastic. In all of these papers, supply is exogenous; we let the designer control the types of arriving goods.

A useful benchmark is Leshno (2022), which characterizes optimal queueing mechanisms under exogenous supply. Endogenous supply changes the designer's problem: queue choices are no longer only claims on allocation, but also reports to which future supply responds. Other papers study dynamic allocation under private information without transfers: Verdier and Reeling (2022) show that repeatedly allocating bear-hunting licenses to the same individuals can improve matching (an impractical approach here, since agents need only one good), and Guo and Hörner (2020) consider repeated allocation to a single agent with fluctuating valuations. Relatedly, Galichon and Hsieh (2018) treat money burning—time spent waiting—as a non-transferable numeraire.[8]

2 Model

We develop a parsimonious model of a lottery mechanism with endogenous supply of goods. Our results survive a diverse range of model generalizations, which we discuss in Section 6. Time is discrete with an infinite horizon, $t \in \{0,1,2,\ldots\}$. Each period, one agent arrives.[9] Agents are born with preferences over the goods. The planner has no goods available at $t = 0$, but supplies one good in every subsequent period. Goods and agents can be of $|\Theta| = 2$ types, $\Theta = \{A,B\}$.[10] We use θ_t to denote the type of the arriving agent in period t, and ϕ_t to denote the good supplied in that period. Each agent's type is drawn with uniform probability from Θ. Agent types are private information, unknown to the designer. An agent matched to a good of the same type receives utility h. An agent matched to a good with a different type receives (positive) utility $l < h$.

There is one queue for each type of good, denoted as queue A and queue B. At the beginning of

[8] Our companion work (**lee2026dynamic**) takes an empirical approach to a distinct question: how limited consumer commitment impacts welfare and distributional effects of a primary market for public housing in Singapore, and whether market design solutions can improve applicant outcomes.

[9] In Section 4 we investigate the effects of competition; by permiting multiple agents to arrive at $t = 0$ and allowing the designer to wait for multiple agents to arrive after $t = 0$.

[10] Appendix B generalizes to an infinite type space with heterogeneous preference intensities.

each period, agents are informed of the number of agents and goods in each queue. An incoming agent chooses the queue she wishes to enter. We denote the queue choice of the period-t agent by $d_t \in \{A, B\}$. At the beginning of time, the designer commits to a supply rule (the mechanism defined below) that sets the probability with which $\phi_t = A$, conditional on the currently present agents and goods.[11] Each period, one agent arrives and simultaneously chooses a queue while the designer supplies a good; goods are then allocated within queues by uniform lottery, and unmatched agents pay a wait cost, $c > 0$, and remain. Figure 1 in Appendix **??** summarizes the timing.

An unmatched agent incurs a per-period flow cost of waiting, c, and remains in her queue.[12] We assume waiting costs are linear, as in related work on dynamic matching (e.g., Ashlagi et al. 2018; Baccara, Lee, and Yariv 2020; Leshno 2022). The designer is thus agnostic to arrival times, rendering the state space tractable.[13]

Two features of the environment are essential. First, the designer cannot elicit agent preferences through means outside the allocation mechanism. Second, and central to our analysis, we restrict the designer to *queue-anonymous* mechanisms:

Assumption 1 (Queue-anonymity)**.** *The designer treats agents within a queue identically: if k agents are in queue ϕ and $m \leq k$ type-ϕ goods are available, each agent receives one with equal probability m/k.*

This restriction is the crux of our analysis. Queue-anonymity requires that agents be treated without regard to seniority. Among other mechanisms, this rules out first-in-first-out (FIFO) priority queues. Were FIFO permitted, the designer could trivially achieve the first-best in every parameter region without ever observing types, by supplying a good to match the report of the entering agent. But FIFO relies on a form of commitment that applicants in these settings lack: an applicant who reaches the front but has since found an outside option can simply decline the

[11] This commitment timing reflects settings in which supply must be planned in advance of observing applications, as when (public) housing projects are planned before launch, or when the mix of MBA classes is determined before students register.

[12] Appendix **??** shows our mechanisms remain optimal when agents may switch queues between periods.

[13] Exponential costs would require tracking each agent's time of arrival.

offered good, so a priority queue generates stale offers, refusals, and vacancy risk rather than clean one-period matching. (In public housing, applicants routinely decline units offered when their number is called.)[14]

We use $s^t = (s_A^t, s_B^t) \in \mathbb{Z}^2$ to denote the net demand of the market in the beginning of period t. For $s_\phi^t > 0$, s_ϕ^t denotes the number of agents in queue ϕ. Otherwise, queue ϕ has no agents and $-s_\phi^t$ denotes the number of unassigned type-ϕ goods. We will focus on Markovian mechanisms, mechanisms which condition period by period on payoff-relevant state variables.

Definition 1. *A **mechanism** μ is $\Phi^A : \mathbb{Z}^2 \to [0,1]$, which maps the state to the probability a type-A good is supplied.*

Transfers are not allowed. This restriction reflects the same equity considerations that motivate rationing by lottery rather than price (e.g., in public housing, school seats, and food assistance). Analytically, the absence of transfers leaves waiting time as the only instrument that can screen agents, so wait time plays the role of a non-transferable numeraire.

Equilibrium concept. The designer has commitment power and declares the mechanism μ at the beginning of time. The mechanism μ induces a queue-choice game: agents observe the current public state, take the committed supply rule and other agents' queue-choice strategies as given, and choose a queue to maximize expected utility. Agent's wait times are therefore equilibrium objects, not primitives of the mechanism. We do not solve a direct-revelation mechanism-design problem; incentive constraints enter through agents' equilibrium queue choices. When an agent is indifferent across queues, ties are broken toward the queue matching her own type. We will call applications to a queue matching an agent's own type *sincere applications*. In Section 3.2, Lemma 3 establishes the steady-state selection result needed to evaluate optimal mechanisms.

In state s, $W_\phi^\mu(s)$ denotes the expected wait time for an incoming agent that enters queue ϕ. At risk of confusion we will differentiate this initial expected wait time from $w_\phi^\mu(s)$, the expected

[14] Consistent with this account, Singapore's pre-2001 seniority-based Registration for Flats System left the housing authority holding vacant stock during the Asian Financial Crisis, after which it moved to the current per-cycle lottery.

wait time at the beginning of the period for an agent *already in queue* ϕ. Note that $w_\phi^\mu(s)$ involves uncertainty over the incoming agent's type, while $W_\phi^\mu(s)$ has already resolved this uncertainty. We will drop the reference to the mechanism when the context poses no confusion.

The expected utility in state $s = (s_A, s_B)$ for an agent that enters queue ϕ is the match benefit from a type-ϕ good minus expected waiting costs from queue ϕ:

$$U(\phi, \theta, s) = l + \mathbb{1}_{\{\phi=\theta\}}(h - l) - cW_\phi(s). \tag{1}$$

When an agent enters the queue of her type, we will say she has *applied sincerely*. An agent only applies sincerely if doing so maximizes her utility:

$$\theta \in \arg\max_d \; U(d, \theta, s). \tag{2}$$

The designer's objective depends on two elements: the quality of matches and the frequency of unassignment. Since there is always one more agent than available goods, there will always be a minimum of one unassigned agent in every period. To normalize the measure of inefficiency, we only consider unassignment above the baseline of one agent. In this context, counting the number of unallocated goods is equivalent to counting the number of excess unmatched agents. We will use the two measures interchangeably, except when investigating batching in Section 4.2 where the difference between the two measures is relevant.

Because we do not know exactly how the designer ranks these two sources of inefficiency in practice, we characterize the general solution to the designer's problem. The designer aims to minimize total inefficiency, with weights α and $1 - \alpha$ placed on allocation and unassignment inefficiency respectively. A realized match is allocatively inefficient when the matched agent's type differs from the type of the good she receives (a *mismatch*); m_t, the level of allocation inefficiency in period t, counts such mismatches. Similarly, we define the unassignment inefficiency in state $s^t = (s_A, s_B)$ as $v_t \equiv \max\{s_A, s_B\} - 1$, the normalized number of unallocated goods. A mechanism

is evaluated by the *average inefficiency* it creates:

$$V(\mu) \equiv \lim_{T\to\infty} \sup \frac{1}{T}\mathbb{E}_\mu \left[\sum_{t=1}^{T} \alpha m_t(\mu) + (1-\alpha) v_t(\mu)\right]. \tag{3}$$

Definition 2. *Consider the class of queue-anonymous lottery mechanisms. A mechanism μ^* in this class is **optimal** if:*

$$V(\mu^*) = \inf_{\mu} V(\mu),$$

where the infimum is taken over all mechanisms in the class.

Equation 1 defines $V(\mu)$ for an arbitrary mechanism. Under Assumption 1 (stated below in Section 3.2), however, Lemma 3 will show that every optimal mechanism generates at least one steady state, and that if there are multiple steady states of an optimal mechanism, they are equivalent up to relabeling the queues, so they yield the same value of V. We therefore evaluate inefficiency at the steady state: letting $m(\mu)$ and $v(\mu)$ denote allocation and unassignment inefficiency in a steady state of μ, the designer's problem becomes

$$\min_{\mu} \ \alpha m(\mu) + (1-\alpha) v(\mu). \tag{4}$$

We note that this framework encapsulates utilitarianism, but can also address more general environments. A designer wishing to maximize agent utility would place a weight of $h-l$ on allocation inefficiency, and a weight of c on unassignment inefficiency to represent the agent cost of unassignment (then normalize the sum of weights to 1 by dividing both weights by $h-l+c$).

3 Results

When is it optimal to disregard agent preferences when choosing the type of good to supply? In this section, we provide context with the complete information benchmark: the designer supplies precisely the desired good. We then characterize optimal designer strategy under endogenous

supply: "oversupplying" the underdemanded good can motivate sincere applications.

3.1 First-Best Benchmark

The first-best eliminates both sources of inefficiency: all matched agents receive their preferred good, no goods are left unused, and the only unmatched agent is the unavoidable one-agent backlog. When the designer has complete information, this result is immediate. The designer assigns each arriving agent to her preferred queue and supplies the good matching the non-empty queue. Thus, on the equilibrium path, $\Phi^A(1,0) = 1$ and $\Phi^A(0,1) = 0$, the system remains in states $(1,0)$ and $(0,1)$, and $m = v = 0$.

Under private information, the same allocation is implementable only if agents are willing to apply sincerely. The relevant incentive constraint compares the match benefit from entering one's preferred queue with the induced difference in expected wait times.

Lemma 1. *A type-A agent prefers to apply sincerely if and only if* $\frac{h-l}{c} \geq W_A(s) - W_B(s)$.

The proof of Lemma 1 and all future proofs are relegated to the appendix. The left-hand side is the benefit-cost ratio $\gamma \equiv \frac{h-l}{c}$. A higher value of γ means that agents are more willing to wait for a preferred good.

The only binding incentive constraint under the first-best mechanism arises for the type whose preferred queue is currently empty. In state $(1,0)$, for example, a type-B agent trades off a better match from entering queue B against a shorter wait from entering queue A. Under the first-best mechanism, the expected waits are $W_A(1,0) = \frac{2}{3}$ and $W_B(1,0) = \frac{4}{3}$. Hence, sincere application requires $\gamma \geq W_B(1,0) - W_A(1,0) = \frac{2}{3}$.

Proposition 1 (First-best benchmark)**.** *Under complete information, the first best achieves* $m = v = 0$*. Under private information, a zero-inefficiency outcome is implementable within the mechanism class of Definition* **??** *if and only if* $\gamma \geq \frac{2}{3}$.

Thus, when $\gamma \geq \frac{2}{3}$, the designer can eliminate inefficiency by responding to the currently non-empty queue: supply the type-A good in state $(1,0)$ and the type-B good in state $(0,1)$. When

$\gamma < \frac{2}{3}$, that responsive policy makes the under-demanded type unwilling to apply sincerely. The remainder of the paper studies the optimal policy in this incentive-constrained region.

3.2 When the First-Best Cannot be Implemented

We show the existence and effective uniqueness of steady states for optimal mechanisms (unique up to relabeling the queues) then characterize the optimum conditional on the benefit-cost ratio γ. The class of optimal mechanisms is rich: it comprises pooling applicants and supply on one good; the one-parameter family of responsive two-state mechanisms introduced below; and, once we allow batching in Section 4, the batching mechanisms. As α and γ vary, the optimum moves across these, each being optimal on distinct parameter regions (Theorems 1 and 2).

Assumption 2 (Impatience)**.** $\gamma < \frac{2}{3}$.

Assume now that agents are impatient enough that a mechanism designed to eliminate all inefficiency fails to motivate sincere applications. Importantly, Assumption 1 sharply restricts agent behavior in state $(2,-1)$. In state $(2,-1)$[15], all agents prefer to enter queue B: Under any mechanism, the maximum wait for an agent that enters queue B is 0 because a type-B good is available. In contrast, the minimum wait for an agent that enters queue A is $\frac{2}{3} + \frac{2}{3} \cdot \frac{1}{2} = 1$, which arises when the designer always supplies type-A goods and all incoming agents apply to queue B. Combined with Lemma 1, a type-A agent prefers to apply sincerely only if $\gamma \geq 1$, which violates Assumption 1. Therefore, in state $(2,-1)$, by Assumption 1, any incoming agent will enter queue B regardless of her type. This statement also holds for any state with more than two agents in queue A. The expected wait from queue A in such a state is strictly larger than in state $(2,-1)$, while the expected wait from queue B remains 0.

Lemma 2. *Under Assumption 1, if* $\max\{s_A, s_B\} > 1$*, in state* (s_A, s_B)*, either type-A or type-B agents do not apply sincerely.*

[15] Where appropriate, we refer to queue-symmetric states as one state, i.e., $(2,-1)$ and $(-1,2)$ are reduced to $(2,-1)$.

The intuition from the above result is: in extreme states, i.e., when one queue is long and the other has an unallocated good, all entering agents prefer to take the available good. Therefore, the system experiences negative feedback and tends towards states in which demand is less imbalanced. It follows that the state space of any mechanism's steady state is finite. In particular, any optimal mechanism generates a steady state with a finite state space. Furthermore, since it cannot be optimal to remain permanently in state $(2,-1)$ or state $(-1,2)$,[16] the steady-state must be recurrent unless it never transitions between $(1,0)$ or $(0,1)$. Any mechanisms that fail to transition between $(1,0)$ and $(0,1)$ have equal inefficiency levels. Then, because the steady state is finite and the state space is recurrent, the steady state is *unique up to relabeling the queues*.

Before stating the result, we define the term *queue symmetric*. A steady state, $\mathbb{S}$, is queue symmetric if there exists a permutation $\pi : \Theta \to \Theta$ such that the probabilities of any two states, $s, s' \in \mathbb{S}$, are equal under the permutation $\pi(s) = s'$ (i.e., equivalent up to relabeling queues). We show that any optimal mechanism generates at least one steady state, and furthermore, that outcomes are effectively unique under such mechanisms:

Lemma 3.

1. *Any Markovian mechanism μ has at least one steady-state associated with μ.*

2. *If an optimal mechanism μ generates multiple steady states, the steady states are queue symmetric.*

Lemma 3 implies that the average level of inefficiency is well defined. Either the steady state is unique, or the two possible steady states feature equivalent levels of inefficiency. As such, we will proceed by evaluating mechanisms using the average level of inefficiency in any steady state.

Importantly, when Assumption 1 holds, behavior in state $(1,0)$ is tightly regulated. Suppose the designer attempted to avoid unallocated goods through always supplying a good of type A, i.e., $\Phi^A(1,0) = 1$. Then, incoming agents optimally respond by entering queue A irrespective of their type. Since in every period one type-A good is supplied and one agent enters queue A, the state

[16] Such a mechanism would be dominated by a mechanism that always remains in state $(1,0)$.

remains in $(1,0)$ indefinitely. We will refer to the described mechanism as the **pooling mechanism**, μ_p.[17] Under the pooling mechanism, without loss of generality, $\Phi^A(s) = 1$ and $d_t = A$.

Since half of the entering agents are of type A, the level of allocation inefficiency is $\frac{1}{2}$ and the level of vacancy inefficiency is 0.

By Assumption 1, any mechanism that always supplies a good of the type matching the non-empty queue causes all agents to prefer said non-empty queue. Then, the state will never change, because every period a new agent enters the queue matching the one that the good is supplied for. To prevent this behavior, the designer must supply the less desirable good with positive probability. This insight cautions against endogenous supply policy that responds myopically to observed queue imbalances: agents are incentivized to not apply sincerely. Furthermore, a naïve imputation of demand through observing queuing decisions, without a deeper understanding of the decision problem faced by the applicant, overstates how satisfied agents are with the current policy regime.

We proceed by determining the shape of an optimal mechanism with allocation inefficiency below $\frac{1}{2}$. In order to improve overall efficiency, we must have $\Phi^A(1,0) \neq 1$. In particular, $\Phi^A(1,0)$ must be low enough to incentivize type-B agents to apply sincerely. In state $(2,-1)$, the designer always supplies type-A goods to minimize vacancy inefficiency, since it cannot increase the incentive for agents to apply sincerely.

Consider the following mechanism. In state $(1,0)$, the designer supplies a type-B good with probability q. In state $(2,-1)$, the designer always supplies a type-A good. There are never more agents in queue A than in state $(2,-1)$, since agents always enter queue B when $s = (2,-1)$ according to Lemma 2. Similarly, $\Phi^A(0,1) = q$ and $\Phi^A(-1,2) = 0$. We refer to this mechanism as the **two-state mechanism with parameter** q. Formally, the *two-state mechanism with parameter* q, μ_q, sets $\Phi^A(1,0) = 1-q$ and $\Phi^A(2,-1) = 1$.[18]

Proposition 2. *Under Assumption 1, the optimal mechanism within the class of Definition* **??** *takes*

[17] There are technically several mechanisms that result in similar allocations and equivalent levels of efficiency. For the sake of exposition, we focus on the pooling mechanism described in the main text.

[18] As an aside, a mechanism could potentially sometimes supply a type-B good when the state is $(2,-1)$. It turns out that the optimal mechanism never does so. In the appendix, we formally show that supplying "the wrong good" in extreme states never increases the extent to which agents apply sincerely.

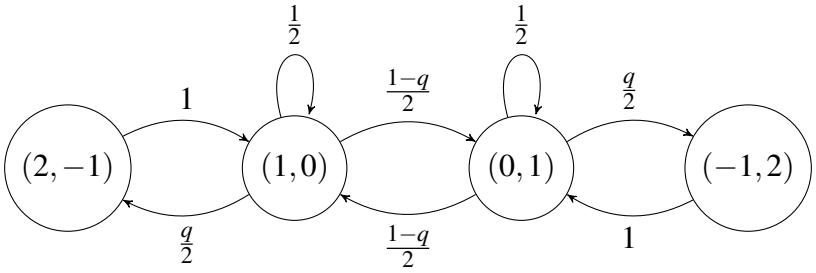

Figure 1: *Finite state automaton depicting the two-state mechanism, μ_q.*
Notes*: Under the two-state mechanism, in extreme states, the designer supplies a good of the type of the long queue. In states $(0,1)$ and $(1,0)$, the designer supplies a good "of the wrong type" with probability q. Arrows denote transition probabilities.*

on one of two forms. Either the pooling mechanism is optimal, or there exists q^* such that μ_{q^*} is optimal.

The probability q governs a trade-off. Raising q lengthens the expected wait $w_A(1,0)$ in the popular queue, which is precisely what makes an incoming type-B agent willing to apply sincerely rather than crowd into queue A; but it also raises the chance of drifting into the costly extreme state $(2,-1)$. The designer therefore picks the smallest q consistent with sincere application, defined implicitly by setting the type-B wait-time difference in Lemma 1 equal to γ:

$$\gamma = \frac{1-q}{2}(1+w_A(1,0)) - q(1+w_A(2,-1)). \tag{5}$$

Solving the equilibrium wait-time recursions (relegated to the appendix) and substituting gives the optimal q in closed form:

Proposition 3. $q^*(\gamma) = \frac{3\gamma-2}{2(\gamma-3)}$.

In the rest of this paper, when we refer to μ_q without specifying q, it is understood that q is optimally chosen, i.e., $q = q^*(\gamma)$. At this q^*, a type-A agent in state $(1,0)$ strictly prefers queue A (the wait-time difference flips sign); symmetrically, agents apply sincerely in state $(0,1)$.

State $(2,-1)$ is the only source of inefficiency under μ_q: there a type-A arrival is mismatched and a good waits unallocated, whereas in state $(1,0)$ all agents apply sincerely. Solving the steady-state equations (again in the appendix) shows the chain spends a fraction $\frac{q^*}{2+q^*}$ of the time in

$(2,-1)$, so allocation inefficiency is $\frac{q^*}{2(2+q^*)}$ and unassignment inefficiency is $\frac{q^*}{2+q^*}$. Thus exactly two mechanisms are candidates for the optimum: pooling and the two-state mechanism; the proof of Theorem 1 shows no other mechanism improves on both. Which of the two wins depends on α, the weight on allocation inefficiency relative to unassignment inefficiency.

Theorem 1. *Let Assumption 1 hold (i.e., agents are sufficiently impatient):*

For $\alpha \leq \frac{2-3\gamma}{8-5\gamma}$, pooling is optimal. For $\alpha \geq \frac{2-3\gamma}{8-5\gamma}$, the two-state mechanism is optimal.

Theorem 1 shows that either μ_p or μ_q must be optimal. The cutoff is decreasing in agents' benefit-cost ratio:

$$\frac{\partial}{\partial \gamma}\left(\frac{2-3\gamma}{8-5\gamma}\right) = \frac{-14}{(8-5\gamma)^2} < 0.$$

Therefore, if the two-state mechanism is optimal for some $\underline{\gamma}$, it is optimal for all $\gamma \geq \underline{\gamma}$. The economic intuition is that μ_q engenders sincere applications, while pooling ignores preference information entirely. As agents place more value on receiving their preferred good relative to waiting, the designer can induce sincere applications with a smaller probability of supplying the under-demanded good, so the region in which endogenous supply improves on pooling expands.

A useful benchmark is the utilitarian objective that weights allocation and unassignment losses by agents' own payoff parameters: $\alpha = \frac{h-l}{h-l+c}$. Then, substituting into the cutoff in Theorem 1 implies that the two-state mechanism is optimal if and only if $\gamma \geq \frac{9-\sqrt{65}}{4} \approx .234$; otherwise, pooling is optimal. Thus, even a utilitarian designer need not always elicit preferences. When waiting costs are sufficiently high relative to match quality, a mechanism that ignores preferences can dominate; when match quality matters enough, endogenous supply becomes valuable because it makes sincere applications informative.

As an example of this insight, we consider why different public housing authorities worldwide adopt different mechanisms. When the alternative to being matched is severe, that is, c is large, e.g., homelessness in the U.S., a preference-ignoring "take-it-or-leave-it" mechanism can be optimal, consistent with Arnosti and Shi (2019). Where the alternative is milder, as in Singapore, where applicants can rent or live with family, eliciting preferences through endogenous supply could be

worthwhile. Therefore, we caution that differences in such mechanisms may not be design failures: they could be optimal designer responses to different waiting costs.

4 Competition and Market Thickness

Having characterized the optimum without batching, we proceed by considering the impact of competition on gains from endogenous supply. Our key result is that the designer can improve on the above mechanisms by batching several applications together. To gain an intuition for why batching works, we first show in Section 4.1 that competition lowers the γ threshold for sincere application when supply is endogenous—by shrinking the *difference* in expected wait times across queues—a result absent under the balanced fixed-supply benchmark.

We use two notions of demand that differ in their timing. A queue is *competitive* if an agent expects to face rivals for the same good, whether now or in later periods; it is *thick* if many of those rivals apply at the *same* time. A continuously-open waitlist is competitive but thin: at any instant a lone applicant queues, yet she knows many others will arrive before she is matched. A single synchronized application window (a housing launch, or the batched cycles we study below) is thick. Thickness thus implies competition, but the converse may not hold.

We show that competition is undesirable under the balanced fixed-supply benchmark: it has a multiplicative effect on expected wait times. Importantly, under this benchmark competition must increase the expected difference in wait times. Then, under this benchmark, high levels of competition dissuade agents from applying sincerely. On the other hand, when supply is endogenous, competition gives the designer more flexibility to equalize expected wait times across queues. Additional policy flexibility dominates the multiplicative effect of competition on wait times, thus improving efficiency in certain parameter regions (though overall welfare still decreases). In particular, thickness is highly desirable for the designer. We show that the designer can artificially generate thickness by batching several applications together. Furthermore, when the designer prioritizes sincere applications, i.e., when α is large, it is optimal for the designer to batch.

4.1 Oversubscription

We consider a natural form of competition, oversubscription: more agents apply than there are goods available. We model oversubscription by letting N agents arrive in period 0 (instead of one), with one agent arriving each period thereafter; the previous sections involve the case where $N = 2$. Larger values of N raise wait times across the board, but also expand the first-best region. The reason is that, in any intermediate state $(k, N-k)$ where both types are present, the designer is now free to choose the supply probability $\Phi^A(k, N-k)$, whereas at $N = 2$ no such freedom existed. For $N > 2$, this added flexibility always improves the designer's ability to equalize wait times across queues and thus to implement the first-best; Appendix **??** characterizes the optimal supply probabilities and the associated range of γ.

Two forces are therefore at play. Increasing competition exacerbates the loss from missing a match, because waiting times rise; but it also gives the designer more free variables with which to equalize wait times across reports, making agents more willing to apply sincerely. To see that the second force is what distinguishes endogenous supply, contrast it with the balanced fixed-supply benchmark, exogenous supply, modeled by $\Phi^A(s) = \frac{1}{2}$ in every state. Such a mechanism can never achieve the first-best, and (as we show in Appendix **??**) the minimal γ sustaining sincere application *rises* with competition, exactly the opposite of the endogenous case. Figure 4 displays the two thresholds: thickening the market helps the designer when she controls supply and hurts her when she does not.

4.2 Batching

Of course, as oversubscription increases, agents are made worse off due to increased wait times. Indeed, our optimality measure penalizes inefficiencies, rather than measuring agent surplus, weakening the comparison across different levels of oversubscription. Furthermore, in practice, the designer does not control agent demand. Nonetheless, the designer can (and does) manipulate the timing of applications. The designer could have agents apply quarterly or annually, instead of

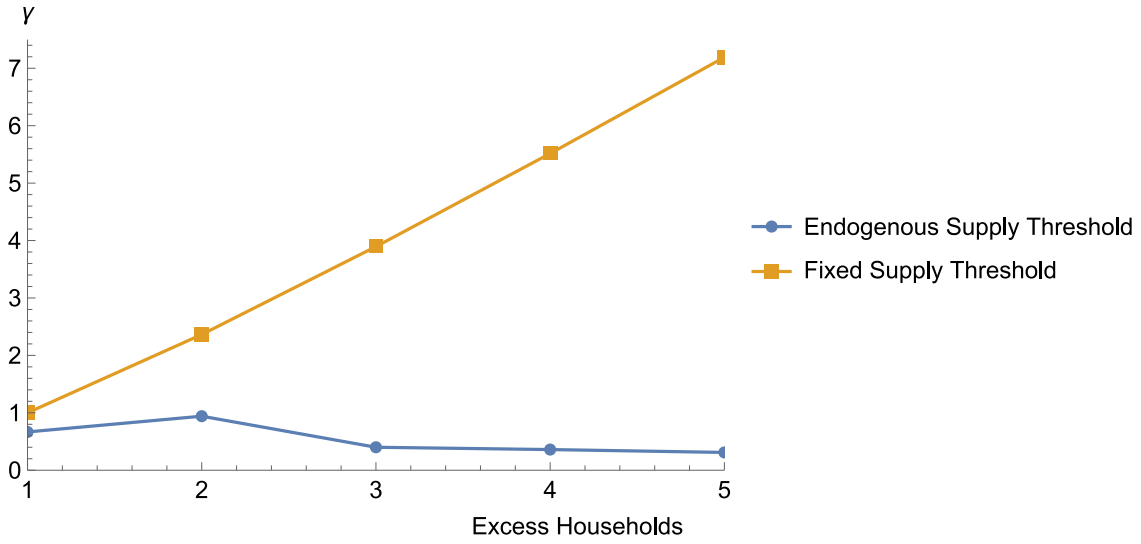

Figure 2: *Lower bounds for the benefit-cost ratio, γ, under endogenous supply and the balanced fixed-supply benchmark.*
Notes*: The x-axis displays the number of excess agents, while the y-axis displays the lower bound on γ. Endogenous Supply Threshold refers to the lower bound on γ such that the expected inefficiency is* 0*. Fixed Supply Threshold refers to the lower bound on γ under which agents are willing to apply sincerely so long as no goods are unmatched. The two curves use* different *criteria: the endogenous curve is a zero-inefficiency threshold, whereas the fixed-supply curve uses the weaker sincere-application criterion. Applying the weaker criterion to the fixed-supply benchmark makes the figure understate the disadvantage of that benchmark relative to a zero-inefficiency criterion within it.*

period-by-period.[19] Through delaying the timing of applications, the designer increases the level of unassignment in the short run, as incoming agents must wait until the next application cycle to enter the market. However, we show that the corresponding increase in market thickness improves the designer's ability to incentivize sincere applications. This lever provides the designer with a trade-off: increase the level of unassignment to improve the quality of matchings.

We extend the model so the designer also chooses the cycle length T. Agents arriving in periods that are not multiples of T wait (type still hidden, flow cost still paid) until the next multiple of T, when all queued-but-unentered agents simultaneously choose queues and the designer releases T stockpiled goods at once. The designer's strategy thus becomes a distribution over the T supplied types, $\Phi^A : S \to \Delta\{A,B\}^T$, chosen before observing reports, and she minimizes the same objective $\alpha m + (1-\alpha)v$. Unassignment and unallocated goods are no longer equivalent, since agents keep

[19]For instance, in 2024, Singapore moved from four to three annual public housing launch cycles, illustrating that planners can adjust the timing of applications as well as the composition of supplied goods.

arriving while goods are withheld; we retain the original unassignment measure, which charges batching a minimal $\frac{T-1}{2}$ and so stacks the deck against batching. Even so, batching can be optimal when the designer cares enough about match quality.

We describe the optimal $T = 2$ mechanism and the condition under which it dominates. Wherever the $T = 2$ mechanism is feasible—for $\gamma \geq \bar{\gamma}_2 \approx .294$, the threshold derived below—it dominates any mechanism with longer cycles, $T \geq 3$ (Lemma **??**). The relevant comparison is therefore between $T = 2$ and the non-batched ($T = 1$) mechanisms, which Theorem 2 resolves.

To begin, suppose the designer aimed to ensure all agents applied sincerely, $m = 0$. In state $(1,0)$, in order to properly incentivize agents, the designer randomizes between supplying one good of each type or supplying two type-A goods. Let $\Phi^A(1,0) = [(q,(1,1)),(1-q,(2,0))]$. In state $(2,-1)$, the designer always supplies two type-A goods, $\Phi^A(2,-1) = [(1,(2,0))]$. We can then compute expected wait times conditional upon the state. These in turn allow the expected wait times conditional on reports to be computed. By Lemma 1 the difference must be bounded by γ in order for agents to apply sincerely.

In the appendix, we compute the difference in wait times with respect to q. Then, we can solve for the optimal level of q that minimizes the difference in wait times. We find the minimal difference is given by $\gamma \approx .294$, which we will denote by $\bar{\gamma}_2$. Hence, the batching mechanism with $T = 2$ induces sincere applications for $\gamma \geq \bar{\gamma}_2$. Notably, batching attains a substantive improvement in motivating sincere applications relative to $T = 1$, which required $\gamma \geq \frac{2}{3}$.

Since the mechanism achieves zero allocation inefficiency, its only cost is unassignment from two sources: the unavoidable $\frac{1}{2}$ that batching with $T = 2$ imposes, and the time spent in the two extreme states $(2,-1)$ and $(-1,2)$. Because each arrival is equally likely to be either type, the steady-state probability of these two queue-symmetric extreme states is independent of q; the appendix computes it to be $\frac{1}{4}$. The total unassignment inefficiency is therefore $\frac{1}{2} + \frac{1}{4} = \frac{3}{4}$. Crucially, the $T = 2$ mechanism already attains zero allocation inefficiency, so a longer cycle has no allocation margin left to improve; yet any $T \geq 3$ pays a deterministic batching cost of at least $\frac{T-1}{2} \geq 1$, strictly above the $\frac{3}{4}$ that $T = 2$ incurs. Hence the $T = 2$ mechanism dominates every $T \geq 3$ mecha-

nism for *every* α and every $\gamma \geq \bar{\gamma}_2$; we record this as Lemma **??**. It remains to rule out a dominating mechanism at $T = 2$, which would also have to beat the pooling mechanism (zero unassignment but $\frac{1}{2}$ allocation inefficiency).

A superior mechanism with $T = 2$ must generate a lower level of unassignment inefficiency. At the same time, it must improve upon the allocation inefficiency generated by the pooling mechanism. However, in order to improve upon the inefficiency of the pooling mechanism, agents must not be matched uniformly. In our proof, we show that no mechanism can do both. The key tension is that mechanisms that improve the level of unassignment inefficiency do so at the cost of increasing allocation inefficiency. However, due to the inherent $\frac{1}{2}$ unassignment inefficiency due to setting $T = 2$, such mechanisms are dominated either by the previously defined $T = 2$ batching mechanism or by the pooling mechanism for all α.

As an aside, we note that for $\gamma \in (.294, \frac{2}{3})$, the optimal mechanism depends on α, the weight on allocative efficiency. When α is small, the pooling mechanism is optimal. As α increases, μ_q becomes optimal. Finally, when α is large, the batching mechanism is optimal.

Theorem 2. *When $\gamma \in (.294, \frac{2}{3})$, the optimal mechanism for $\alpha < \frac{2-3\gamma}{8-5\gamma}$ is the pooling mechanism. For $\alpha \in \left[\frac{2-3\gamma}{8-5\gamma}, \frac{34-9\gamma}{38-15\gamma}\right]$ it is μ_q. Last, for $\alpha > \frac{34-9\gamma}{38-15\gamma}$ the optimal mechanism involves $T = 2$ batching.*

The key implication of Theorem 2 is that batching is only a useful tool when allocation is a greater concern than unassignment. If so, then despite a higher temporary level of unassignment, batching can drastically improve the quality of matches by increasing the thickness of the market.

5 Extensions

The baseline stylizes several elements to isolate the forces of endogenous supply; the appendix shows the results are robust to relaxing them. We allow a continuum of preference intensities (Appendix B), so that each agent has her own benefit-cost ratio γ_i. A threshold in γ_i again separates agents who apply sincerely from those who do not, and the optimal mechanism mirrors the two-

state mechanism of the main text. We permit agents to switch queues across cycles, as they can in many such settings (Appendix **??**), and show the optimal mechanisms are unchanged, because an incoming agent is always better informed than those already queued, so any willingness to switch would push all incoming agents to the same queue and raise mismatch. Finally, we compare the optimal endogenous mechanisms against outcomes under exogenous supply (Appendix D), where the designer cannot choose good types: the wait-minimizing Load Independent Expected Wait policy of Leshno (2022) (optimal in his setting) is only viable in the $\gamma \geq 2$ domain, a region in which the first-best could already be implemented under a lottery with endogenous supply.

6 Discussion

The central lesson of our analysis is that, when supply is endogenous, an application is two things at once: a claim on an available good and a report of the applicant's private type, which affects future supply through the committed rule. Because agents understand that today's applications shape tomorrow's supply, a designer who responds myopically to observed queue imbalances invites strategic, preference-mismatched applications; simple counts of applications then understate how much true preferences have shifted. Recognizing this dual role reshapes optimal design. The designer should sometimes supply the under-demanded good to keep applications informative, and can raise match quality by thickening the market, for instance, by delaying and batching application cycles.

This lesson contrasts with much of mechanism design, which allocates goods arriving exogenously. When the designer can adjust the inflow of goods, as for housing, transportation, and food, the gains from doing so may be large — larger than a naïve comparison of waitlist lengths would suggest, since that comparison neglects how exogenous supply pushes agents to strategize. Endogenous supply is more than an additional instrument: it changes the incentives agents face.

References

Abdulkadiroğlu, Atila, Parag A Pathak, and Alvin E Roth (2009). "Strategy-proofness versus efficiency in matching with indifferences: Redesigning the NYC high school match". In: *American Economic Review* 99.5, pp. 1954–1978.

Abdulkadiroğlu, Atila and Tayfun Sönmez (2003). "School choice: A mechanism design approach". In: *American Economic Review* 93.3, pp. 729–747.

— (2013). "Matching markets: Theory and practice". In: *Advances in Economics and Econometrics* 1, pp. 3–47.

Agarwal, Nikhil, Itai Ashlagi, Michael A Rees, Paulo J Somaini, and Daniel C Waldinger (2019). *An empirical framework for sequential assignment: The allocation of deceased donor kidneys*. National Bureau of Economic Research.

Akbarpour, Mohammad, Shengwu Li, and Shayan Oveis Gharan (2020). "Thickness and information in dynamic matching markets". In: *Journal of Political Economy* 128.3, pp. 783–815.

Altmann, Sam M (2024). "Choice, Welfare, and Market Design". In: *mimeo*.

Arnosti, Nick and Peng Shi (2019). "How (Not) to Allocate Affordable Housing". In: *AEA Papers and Proceedings*. Vol. 109, pp. 204–08.

— (2020). "Design of Lotteries and Wait-Lists for Affordable Housing Allocation". In: *Management Science*.

Ashlagi, Itai, Maximilien Burq, Patrick Jaillet, and Amin Saberi (2018). "Maximizing efficiency in dynamic matching markets". In: *arXiv preprint arXiv:1803.01285*.

Baccara, Mariagiovanna, SangMok Lee, and Leeat Yariv (2020). "Optimal dynamic matching". In: *Theoretical Economics* 15.3, pp. 1221–1278.

Budish, Eric and Estelle Cantillon (2012). "The multi-unit assignment problem: Theory and evidence from course allocation at Harvard". In: *American Economic Review* 102.5, pp. 2237–71.

Galichon, Alfred and Yu-Wei Hsieh (2018). "Aggregate stable matching with money burning". In: *Available at SSRN 2887732*.

Guo, Yingni and Johannes Hörner (2020). "Dynamic allocation without money". In: *TSE Working Paper*.

Lee, Kwok Hao, Andrew Ferdowsian, Yiying Tan, and Luther Yap (2024). "Public housing at scale". In: *Mimeo*.

Leshno, Jacob D (2022). "Dynamic matching in overloaded waiting lists". In: *American Economic Review* 112.12, pp. 3876–3910.

Prendergast, Canice (2016). "The Allocation of Food to Food Banks." In: *EAI Endorsed Trans. Serious Games* 3.10, e4.

Shi, Peng (2022). "Optimal priority-based allocation mechanisms". In: *Management Science* 68.1, pp. 171–188.

Thakral, Neil (2019). "Matching with stochastic arrival". In: *AEA Papers and Proceedings*. Vol. 109, pp. 209–12.

Verdier, Valentin and Carson Reeling (2022). "Welfare effects of dynamic matching: An empirical analysis". In: *The Review of Economic Studies* 89.2, pp. 1008–1037.

Wong, Maisy (2013). "Estimating ethnic preferences using ethnic housing quotas in Singapore". In: *Review of Economic Studies* 80.3, pp. 1178–1214.

— (2014). "Estimating the distortionary effects of ethnic quotas in Singapore using housing transactions". In: *Journal of Public Economics* 115, pp. 131–145.

Appendices

A Market Timing

Figure 1 summarizes the timing of the market described in Section 2.

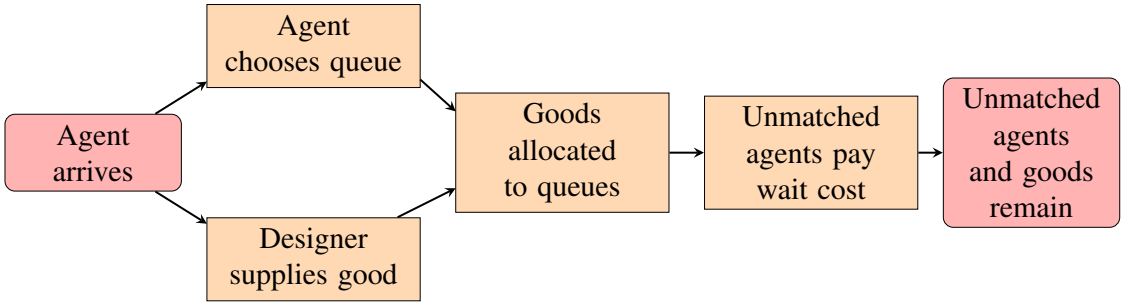

Figure 3: *Market Timing.*
Notes*: One agent arrives. Simultaneously, the agent chooses a queue to join, while the designer decides what kind of good to supply. Goods are allocated to agents in each queue by uniform random lottery. Unmatched agents pay a wait cost and remain in their queues till the next period.*

B Oversubscription: Construction

This appendix gives the construction underlying the oversubscription discussion of Section 4.1. At $t = 0$ all agents privately and simultaneously choose a queue. Agent types are private, but agents observe the queues other agents have entered and know the state of the market.

As with μ_{fb}, a mechanism that achieves 0 inefficiency cannot allow unallocated goods and must incentivize sincere application. When the state is $(N,0)$ the designer must supply a type-A good. In any intermediate state $(k,N-k)$ where both types are present, she can freely choose $\Phi^A(k,N-k)$, the added latitude (absent at $N = 2$) that lets her equalize wait times across queues.

We find the optimal supply probabilities and the associated restriction on γ. The optimal mechanism randomizes: it sometimes supplies the less-demanded good, pushing the state toward a more extreme level. To preserve the first-best the mechanism cannot supply a good of the less-demanded type when no agents wait in the corresponding queue, which limits the range of γ over which the first-best is implementable. Except in the most extreme states $s = (N,0)$ and $s = (0,N)$, both good

types are supplied with positive probability. Figure 3 displays the optimal mechanism conditional on the number of excess agents; in general the probability the less desired good is supplied exceeds the fraction of agents in the corresponding queue.[20]

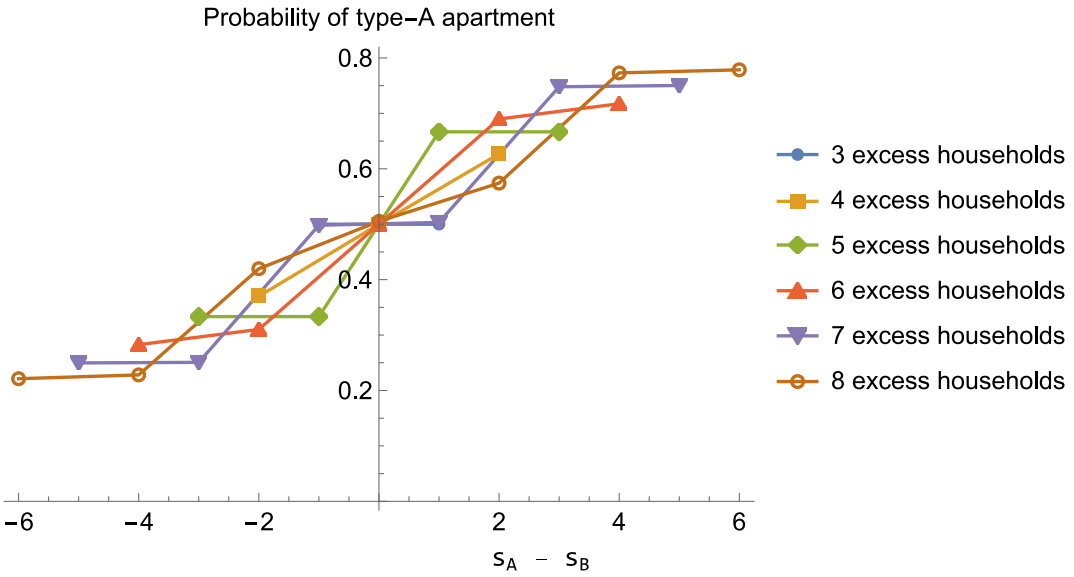

Figure 4: *The probability that a type-A good is supplied under the optimal mechanism.*
Notes*: The x-axis depicts the number of agents in queue A minus the number in queue B. The y-axis is the probability that the designer supplies a type-A good in the corresponding state. States* $(0,N)$ *and* $(N,0)$ *are omitted, their corresponding y-values are* 0 *and* 1 *respectively.*

To isolate the role of supply flexibility, consider the balanced fixed-supply benchmark; then $\Phi^A(s) = \frac{1}{2}$ in every state and agents are still free to choose queues. This mechanism can never achieve the first-best, so we evaluate it against the weaker requirement that agents apply sincerely until a good goes unallocated (states up to $(N+1,-1)$); imposing only this weaker criterion on the fixed-supply benchmark, rather than the zero-inefficiency criterion used for endogenous supply, makes the comparison conservative against our own claim. As Figure 4 shows, the minimal γ sustaining sincere application *rises* with competition under the fixed-supply benchmark—because thickness raises wait times—while it falls under endogenous supply, where the designer's added control reduces the *difference* in wait times across queues.

[20] We solve explicitly for the optimal mechanism when $N < 5$, and numerically compute it when $N \geq 5$.

C Proofs

Proof of Lemma 1. In state s a type-A agent prefers to apply sincerely if and only if $u(d = A, \theta = A, s) \geq u(d = B, \theta = A, s)$. Rearranging the utility function yields:

$$u(d = A, \theta = A, s) \geq u(d = B, \theta = A, s)$$
$$h - cW_A(s) \geq l - cW_B(s) \implies \frac{h-l}{c} \geq W_A(s) - W_B(s).$$

An identical computation holds for type-B agents. The claim follows. □

Proof of Proposition 1. As shown in the text, within the mechanism class of Definition **??** the first-best mechanism is the only one that achieves 0 inefficiency. Under the first-best mechanism, the expected wait times for incoming agents in state $(1,0)$ are:

$$W_B(1,0) = \sum_{i=0}^{\infty} \left(\frac{1}{2} \cdot \frac{1}{2}\right)^i = \frac{4}{3},$$
$$W_A(1,0) = \sum_{i=0}^{\infty} \frac{1}{2}\left(\frac{1}{4}\right)^i = \frac{2}{3}.$$

Because $W_B(1,0) > W_A(1,0)$, type-A agents will always be willing to apply sincerely in state $(1,0)$. Lemma 1 implies that type-B agents will be willing to apply sincerely if $\gamma \geq W_B(1,0) - W_A(1,0) = \frac{2}{3}$. By design, the mechanism is symmetric in states $(1,0)$ and $(0,1)$. Therefore, in state $(0,1)$, type-B agents are always willing to apply sincerely, and type-A agents are willing to apply sincerely if $\gamma \geq \frac{2}{3}$. The mechanism never leaves the set of states $\{(1,0),(0,1)\}$, which implies that no other constraints are relevant. Then, when $\gamma \geq \frac{2}{3}$, the first-best mechanism is implementable, and if $\gamma < \frac{2}{3}$, no mechanism in this class achieves 0 inefficiency. □

Proof of Lemma 2. The paper proves the case when $\max\{s_A, s_B\} = 2$. It remains to show that the claim holds when $\max\{s_A, s_B\} > 2$. We show that a lower bound on a state's expected wait time is monotonically increasing in the state's imbalance, and that lower bound is greater than 1 for $\max\{s_A, s_B\} = 2$. Suppose $\max\{s_A, s_B\} = n$, then, in the initial period where an agent enters

the long queue, she has a $\frac{n}{n+1}$ chance of not receiving a good. If she does not receive a good, in the following period, the best-case for her expected wait time is if the incoming agent enters the short queue, while a good is added to the long queue, leading to a $\frac{n-1}{n}$ chance of receiving the good. Iterating implies that her lower bound can be written as: $\frac{n}{n+1}(1+\frac{n-1}{n}(1+\ldots\frac{2}{3}(1+1/2)))$. Then, letting this lower bound when $\max\{s_A,s_B\}=n$ be represented by k_n, we have $k_n = \frac{n}{n+1}(1+k_{n-1})$. Comparing k_n and k_{n-1} shows that $k_n > k_{n-1}$ if $1 > \frac{1}{n}(k_{n-1})$. However, k_{n-1} must be less than n, because at worst goods could be allocated to the same queue n times while all incoming agents enter the other queue. Therefore, the lower bound is increasing as a function of $\max\{s_A,s_B\}$. Combining the fact that lower bound for the expected wait time is an increasing function of $\max\{s_A,s_B\}$ with the fact that the lower bound is already 1 when $\max\{s_A,s_B\}=2$ implies that minimal expected wait is greater than 1 for any state where $\max\{s_A,s_B\}\geq 2$.

Last, note that the expected wait upon entering the under-filled queue is always 0 when $\max\{s_A,s_B\} > 1$. It follows that the difference in expected waits must be strictly greater than 1 whenever $\max\{s_A,s_B\} > 2$ for *any* mechanism, and by Lemma 1 the claim holds. □

Proof of Lemma 3.

1. To begin, Lemma 2 implies that in state $(2,-1)$ all incoming agents strictly prefer to enter queue B. Then, for any mechanism, the state space is bounded by $(2,-1)$ and $(-1,2)$. A dynamic system with a finite number of states must be recurrent. Hence, by standard results in dynamics, at least one steady state exists.

2. For the steady state to fail to be unique, there must be an absorbing state separating either $(2,-1)$ from $(1,0)$ or $(1,0)$ from $(0,1)$ (or symmetrically, (0, 1) from (-1, 2)). A mechanism that remains in $(2,-1)$ (or (-1, 2)) indefinitely cannot be optimal, since such a mechanism would have both unassignment inefficiency higher than the pooling mechanism, and a equal level of allocation inefficiency. On the other hand, a mechanism that fails to transition between $(1,0)$ and $(0,1)$ implies uniqueness up to queue-symmetric steady states, unless the mechanism has different steady states when beginning with a type-A agent as opposed to a

type-B agent. However, this state-based non-uniqueness cannot be optimal due to the symmetry of the problem. Suppose a mechanism generated two steady states that were not queue symmetric, with differing levels of efficiency. Then, because both must involve equilibrium behavior of the part of the agent, the designer could simply use a mechanism which generates an outcome equivalent to that of the steady state with lower inefficiency for both queues. Through doing so, agent behavior must still be equilibrium behavior, and inefficiency would have been lowered, proving that the original mechanism was suboptimal.

□

Remark 1. *When the designer utilizes μ_q with parameter q^* an incoming type-A agent in state $(1,0)$ does not have an incentive to deviate.*

The following simple result aids in the proof of Proposition 2.

Lemma 4. *The maximal equilibrium allocation inefficiency is $\frac{1}{2}$.*

Proof of Lemma 5. Let $W_A(s) - W_B(s) < \gamma$ for some state s, then each type-A agent strictly prefers to enter queue A. As such, allocation inefficiency in s is at most $\frac{1}{2}$ since all type-A agents apply sincerely. When $W_B(s) - W_A(s) < \gamma$ a similar argument holds for type-B agents. Then, in any state, allocation inefficiency $\leq \frac{1}{2}$. Last, since overall allocation inefficiency is a weighted average of the allocation inefficiency in each state, the overall allocation inefficiency $\leq \frac{1}{2}$. □

Proof of Proposition 2. Lemma 2 allows us to focus on mechanisms with state spaces restricted to states $(2,-1)$ through $(-1,2)$. Since incoming agents always enter the short queue in $(2,-1)$ and $(-1,2)$, the state can never exceed $(2,-1)$ or $(-1,2)$. In state $(2,-1)$, allocation and unassignment inefficiencies are $\frac{1}{2}$ and 1. Note that both values are weakly larger than inefficiency in state $(1,0)$ by Lemmas 2 and 5. It follows that the designer aims to minimize the proportion of time spent in states $(2,-1)$ and $(-1,2)$.

Importantly, if allocation inefficiency is lower than $\frac{1}{2}$, the difference in expected wait times in states $(1,0)$ and $(0,1)$ can be at most γ. That is, $|W_A(1,0) - W_B(1,0)| \leq \gamma$, otherwise type-$A$ agents or type-B agents strictly prefer to not apply sincerely.

We then optimize over all such possible mechanisms, and show that μ_q minimizes inefficiency. Firstly, note that if a type-B agent in state $(1,0)$ has a strict preferences for appling sincerely, $\Phi^A(1,0)$ can be increased while ensuring type-B still has incentive to continue applying sincerely. Furthermore, increasing $\Phi^A(1,0)$ reduces the probability that the mechanism enters state $(2,-1)$. Then, the optimal mechanism has $\gamma = W_B(1,0) - W_A(1,0)$.

We proceed by considering a mechanism more general than μ_q. There are two primary differences. First, the designer sometimes supplies the wrong good in state $(2,-1)$, that is $\Phi^A(2,-1)$ is not necessarily 1. Second, type-B agents are incentivized to apply sincerely with probability below 1 in state $(1,0)$. Let $x^B(1,0) = \Pr(\text{a type-B agent enters queue A in state (1, 0)})$.

The level of inefficiency generated by this mechanism is

$$V(x,\Phi) = \frac{2(1+x)q - \alpha(q + x(-2+q+2\Phi))}{2(2+q+xq-2\Phi)},$$

where we write $x \equiv x^B(1,0)$ and $\Phi \equiv 1 - \Phi^A(2,-1)$ for brevity. The following lemma shows that both can be set to 0.

Lemma 5. *Under Assumption 1, if the pooling mechanism is not optimal, then the optimal mechanism sets* $x^B(1,0) = 0$ *and* $\Phi^A(2,-1) = 1$.

Proof. On the admissible domain $x, \Phi, \alpha \in [0,1]$ and $q \in (0,1]$, the denominator $2(2+q+xq-2\Phi)$ is positive (it vanishes only at the degenerate corner $q = 0$, $\Phi = 1$), so I is differentiable there and the sign of each partial derivative is the sign of its numerator (the denominator enters squared). Differentiation gives

$$\frac{\partial I}{\partial \Phi} = \frac{-q\,(x+1)\left(\alpha(x+1)-2\right)}{(2+q+xq-2\Phi)^2}, \qquad \frac{\partial I}{\partial x} = \frac{-2\,(\Phi-1)\left(\alpha(1-\Phi)+q\right)}{(2+q+xq-2\Phi)^2}.$$

Both numerators are nonnegative: in the first, $q, (x+1) \geq 0$ and $\alpha(x+1) - 2 \leq 1 \cdot 2 - 2 = 0$; in the second, $-(\Phi - 1) \geq 0$ and $\alpha(1-\Phi) + q \geq 0$. Hence I is nondecreasing in both Φ and x, and is minimized at $x = \Phi = 0$. □

Then, Lemma 6 implies that the last variable to consider is the value of q. As shown above, q must be minimized subject to the constraint that type-B agents in state $(1,0)$ apply sincerely. Therefore, μ_q is optimal anytime type-B agents in state $(1,0)$ have incentive to apply sincerely. Then, given the previous computation of q^*, μ_q is optimal. □

We now determine the wait times for μ_q. They solve

$$w_A(1,0) = \frac{1}{2}\cdot q[1+w_A(1,0)] + \frac{1}{2}\cdot q[1+w_A(2,-1)] + \frac{1-q}{4}\cdot[1+w_A(1,0)],$$
$$w_A(2,-1) = \frac{1}{2}[1+w_A(1,0)],$$

so $w_A(1,0) = \frac{4q+1}{3-2q}$ and $w_A(2,-1) = \frac{2+q}{3-2q}$. In order for agents to apply sincerely, Lemma 1 implies the following constraints on γ:

$$\gamma \geq (1-q)[1+w_A(1,0)] - [q(1+w_A(2,-1)) + \frac{1-q}{2}(1+w_A(1,0))],$$
$$\gamma \geq \frac{1-q}{2}(1+w_A(1,0)) - q(1+w_A(2,-1)).$$

Lemma 6. *Under μ_q, the level of allocation inefficiency is $\frac{2-3\gamma}{14(2-\gamma)}$ and the level of unassignment inefficiency is $\frac{2-3\gamma}{7(2-\gamma)}$.*

Proof of Lemma 7. Since μ_q only generates inefficiency in states $(2,-1)$ and $(-1,2)$, the total inefficiency is directly proportional to the proportion of time spent in those two states. Under the optimal mechanism, the proportion of time in states $(2,-1)$ and $(-1,2)$ is $\frac{2-3\gamma}{7(2-\gamma)}$. With $\frac{1}{2}$ probability, an agent fails to apply sincerely, and there is always an unallocated good in $(2,-1)$ or $(-1,2)$. Therefore, the allocation and vacancy inefficiencies are given by $\frac{2-3\gamma}{14(2-\gamma)}$ and $\frac{2-3\gamma}{7(2-\gamma)}$. □

Proof of Proposition **??**. The argument in the main text nearly suffices a complete proof of the claim. The only lacking feature is that the choice of q need not exactly render type-B agents indifferent in state $(1,0)$; larger values of q would also convince agents to apply sincerely. The upper limit for q is the point at which type-A agents in state $(1,0)$ prefer to enter queue B. The

constraint compares the benefit-cost ratio to a function of q:

$$\gamma \geq q(1+w_A(2,-1)) - \frac{1-q}{2}(1+w_A(1,0)). \tag{6}$$

Solving Equation 3 for q implies that $q = \frac{3\gamma+2}{2(\gamma+3)}$. Then, agents apply sincerely in state $(1,0)$ when $q \in \left[\frac{3\gamma-2}{2(\gamma-3)}, \frac{3\gamma+2}{2(\gamma+3)}\right]$. We utilize the previously calculated inefficiency values by taking the derivative of inefficiency with respect to q. Unsurprisingly, both derivatives are positive: increasing the probability that the undesired good is supplied, increases inefficiency. The claim follows. □

Proof of Theorem 1. Proposition 2 shows that either μ_q or μ_p is optimal. We proceed by comparing the inefficiencies generated. Lemma 7 directly provides the individual inefficiencies for μ_q. Summing them with weights from the designer's objective implies that total inefficiency is:

$$\frac{\alpha}{2}\frac{2-3\gamma}{7(2-\gamma)} + (1-\alpha)\frac{2-3\gamma}{7(2-\gamma)} = \left(1-\frac{\alpha}{2}\right)\frac{2-3\gamma}{7(2-\gamma)}.$$

The level of inefficiency under the pooling mechanism is $\alpha\frac{1}{2}$; there is no vacancy inefficiency and half of the agents fail to apply sincerely. Comparing the two and solving for α yields the threshold $\frac{2-3\gamma}{8-5\gamma}$. □

The following lemma settles the choice of cycle length *wherever the $T=2$ mechanism is feasible*: in that region no longer cycle beats $T=2$, so the only live comparison is $T=2$ versus the non-batched ($T=1$) mechanisms. This is what justifies restricting attention to $T \in \{1,2\}$ in the proof of Theorem 2. We do not characterize the optimal T for $\gamma < \bar{\gamma}_2$; see the remark after the proof.

Lemma 7. *Fix any $\alpha \in [0,1]$ and any $\gamma \geq \bar{\gamma}_2$, where $\bar{\gamma}_2 \approx .294$ is the threshold at which the $T=2$ batching mechanism induces sincere applications. Within the class of Definition **??**, no mechanism with cycle length $T \geq 3$ attains a strictly lower value of the objective $\alpha m + (1-\alpha)v$ than the $T=2$ batching mechanism. Consequently, for $\gamma \geq \bar{\gamma}_2$ the designer's optimal cycle length lies in $\{1,2\}$.*

Proof. Both m and v are nonnegative, so $\alpha m + (1-\alpha)v \geq (1-\alpha)v$ for every mechanism, and the objective of any mechanism is bounded below by its unassignment component scaled by $(1-\alpha)$.

Consider any mechanism with cycle length $T \geq 3$. Because goods are released only at multiples of T while one agent arrives each period, agents who arrive in the $T-1$ intermediate periods of a cycle are unassigned for the remainder of that cycle. Averaging over arrival positions within a cycle, this contributes a deterministic unassignment of $\frac{T-1}{2}$, independent of the supply rule Φ^A and of the realized states; this is the $\frac{T-1}{2}$ floor noted in the main text. For $T \geq 3$ this floor is at least $\frac{3-1}{2} = 1$. So unassignment inefficiency is a minimum of 1 for every $T \geq 3$ mechanism, and overall inefficiency is at least $(1-\alpha) \cdot 1$.

Now compare against the $T = 2$ batching mechanism of Theorem 2, which by hypothesis is feasible (it induces sincere applications whenever $\gamma \geq \bar{\gamma}_2$) and attains $m = 0$ and $v = \frac{3}{4}$, hence objective $(1-\alpha) \cdot \frac{3}{4}$. Two features make this an unconditional comparison. First, on the allocation margin the $T = 2$ mechanism already attains the global lower bound $m = 0$, so no longer cycle can improve match quality: the only thing a larger T could buy is lower unassignment, and we have just shown it instead *raises* the unassignment floor. Second, the gap is in unassignment alone, so it does not depend on how the designer weights the two margins:

$$\underbrace{(1-\alpha) \cdot 1}_{\text{any } T \geq 3} > \underbrace{(1-\alpha) \cdot \tfrac{3}{4}}_{T=2} \qquad \text{for every } \alpha \in [0,1).$$

(At $\alpha = 1$ the designer ignores unassignment entirely; then every mechanism that attains $m = 0$, including $T = 2$, is optimal, and no $T \geq 3$ mechanism does strictly better.) Thus no $T \geq 3$ mechanism strictly beats $T = 2$, for any α and any $\gamma \geq \bar{\gamma}_2$. □

Proof of Theorem 2. In order to show that the $T = 2$ batching mechanism is optimal, we proceed in a similar manner to the proof of the optimality of the q^* mechanism. By Lemma **??**, whenever the $T = 2$ mechanism encourages agents to apply sincerely, it dominates every mechanism with $T \geq 3$, for every α. It therefore remains only to compare $T = 2$ against the $T = 1$ mechanisms.

Since we have already characterized the optimal mechanisms with $T = 1$, we proceed by prov-

ing the $T = 2$ mechanism is optimal among all mechanisms with $T = 2$. Note that for reasons similar to that in the q^* mechanism, it is never optimal to supply type-B goods in state $(2,-1)$. Formally, the problem is the following:

$$V(\mu) = \min_{\Phi^A(1,0)\in\Delta\{0,1\}^2} \alpha m + (1-\alpha)v(\mu).$$

Standard optimization techniques imply that $\Phi^A(1,0)[0,2] = 0$. Last, we determine the optimal value for q under the $T = 2$ mechanism. We can compute expected wait times for an agent already in queue A as the solution to the following pair of equations:

$$\begin{aligned} w_A(1,0) &= \frac{1}{4}\left[\frac{2q}{3}(2+w_A(2,-1)) + (1-q)\frac{1}{3}(2+w_A(1,0))\right] + \frac{1}{2}\cdot\frac{q}{2}(2+w_A(1,0)), \\ w_A(2,-1) &= \frac{1}{4}\cdot\frac{1}{2}(2+w_A(2,-1)) + \frac{1}{2}\cdot\frac{1}{3}(2+w_A(1,0)). \end{aligned}$$

Algebra yields the following solutions with respect to q:

$$w_A(1,0) = \frac{42+196q}{231-50q}; \quad w_A(2,-1) = \frac{162+4q}{231-50q}.$$

Conditional on the state, the difference in wait times is:

$$\begin{aligned} w_A(1,0) - w_B(1,0) &= \left(\frac{1}{6}\right)\frac{1725-2357q-62q^2}{231-50q} \\ w_A(2,-1) - w_B(2,-1) &= \frac{453-142q}{1386-300q} \end{aligned}$$

This batching mechanism minimizes the maximum of the two differences when $q = \frac{3}{124}(-833 + \sqrt{753905})$, implying that agents apply sincerely for γ above $\frac{-453+\frac{213}{62}(-833+\sqrt{753905})}{6(-231+\frac{75}{62}(-833+\sqrt{753905}))} \approx .294$.

Finally, we compute the state-dependent component of unassignment inefficiency: the time the chain spends in the extreme states. Order the states as $(2,-1)$, $(1,0)$, $(0,1)$, and $(-1,2)$. The

transition matrix under the $T = 2$ mechanism is

$$P = \begin{pmatrix} 1/4 & 1/2 & 1/4 & 0 \\ q/4 & (1+q)/4 & (2-q)/4 & (1-q)/4 \\ (1-q)/4 & (2-q)/4 & (1+q)/4 & q/4 \\ 0 & 1/4 & 1/2 & 1/4 \end{pmatrix}.$$

By the queue symmetry of the problem, the stationary distribution takes the form (e, r, r, e). The balance equation for the extreme state and the normalization give

$$\left\{ e = \frac{e}{4} + \frac{r}{4},\ e + r = \frac{1}{2} \right\} \quad \Longrightarrow \quad \left\{ e = \frac{1}{8},\ r = \frac{3}{8} \right\}.$$

Hence, the two extreme states together carry stationary probability $2e = \frac{1}{4}$, as the main text asserts.

□

D Preference Intensity

In the main paper, we assumed that agents were one of two types, A or B. Both types valued matched and mismatched goods comparably (h, l), and shared wait costs c; they only differed in the types of goods that they preferred. We now consider what happens if agents have differing intensities of preferences. Agent types are denoted by θ_i, where $\theta \in \{A, B\}$ and $i \in \{1, 2, \ldots, K\}$. Each type's utility is characterized by three parameters: h_i, l_i, and c_i. Agents of type θ prefer that variety of good and receive $h_i \geq l_i$, while facing wait cost c_i. Without loss of generality, we order the types by increasing value of $\gamma_i = \frac{h_i - l_i}{c_i}$.[21] We assume that in each period the incoming agent's type is drawn uniformly across all possible types.

To begin, we observe an analogue of Lemma 1 for the generalized type space. Agent i is only willing to apply sincerely if the increase in expected wait time is below γ_i. The proof follows

[21] We also assume each type has a unique value of γ_i. If two types have the same θ and γ_i we treat them as a single type.

directly from the proof of Lemma 1.

Lemma 8. *A type-A_i agent prefers to apply sincerely if and only if $\gamma_i \geq W_A(s) - W_B(s)$.*

Therefore, for the remainder of this section, we identify types with their respective values of γ_i.

D.1 Strong and Weak Preferences

To begin, we let $K = 2$. First, suppose that $\gamma_1 \geq 2/3$, which implies that $\gamma_2 \geq 2/3$. Then, all types are willing to apply sincerely under the first-best mechanism, which is optimal.

Next, suppose that $\gamma_1 < 2/3$. As shown in the main body of the paper, the first-best mechanism then fails to motivate agents to apply sincerely. Consider the two-state mechanism and choose q such that type-A_1 agents are willing to apply sincerely in state $(1,0)$. This two-state mechanism automatically generates an equilibrium. As shown in the main text, $W_A(1,0) > W_B(1,0)$ for any value of q consistent with the two-state mechanism. Therefore, there is no worry that type-A_1 agents will fail to apply sincerely in state $(1,0)$. Furthermore, in state $(2,-1)$, Lemma 2 implies that all agents will enter queue B.

It remains to be determined when the two-state mechanism is optimal. There are two other mechanisms that require consideration. First, recall the pooling mechanism, which has been previously considered; second, we introduce a "blended" mechanism, in which q is chosen such that type-B_2 agents apply sincerely in state $(1,0)$ while type-B_1 agents do not. We will refer to this mechanism as the *mixed mechanism*. Under the mixed mechanism, agents still always enter queue B in state $(2,-1)$. However, only type B_2 agents enter queue B in state $(1,0)$, while agents of types A_2, A_1, and B_1 all enter queue A instead.

The level of inefficiency under the mixed mechanism can be computed as follows:

$$u_{mm} = \frac{3(-5+\gamma+\sqrt{41-30\gamma+\gamma^2})(1-\alpha/2)+4\alpha}{1+3\gamma+3\sqrt{41-30\gamma+\gamma^2}}.$$

We can then compare inefficiencies under the mixed mechanism to those under the two-state mechanism. Notably, for $\alpha > 1/2$ and any values of γ_1 and γ_2, the mixed mechanism is strictly less

efficient than the two-state mechanism: In the mixed mechanism, in every state, there is a minimum of $1/4$ matching inefficiency. In contrast, under the two-state mechanism, all agents apply sincerely in state $(1,0)$, and the probability of state $(2,-1)$ can be kept sufficiently low. Indeed, the maximum probability of $(2,-1)$, comes when $\gamma = 0$, and is still only $1/7$.

However, as α decreases, the relative value of the mixed mechanism increases. Indeed, when γ_1 and γ_2 are sufficiently far apart, the mixed mechanism benefits from type-B_1 agents not applying sincerely in state $(1,0)$, decreasing the value of q needed for the correct agents to apply sincerely.

D.2 Continuum of Agent Types

The results in the previous section generalize to when $K = \infty$, and there is a continuous distribution G over each agent's realization of γ_i. Let $G(x)$ indicate the probability that an agent's value of γ is below x. We require that $G(0) = 0$. Agents can be divided into two groups: those who intend to apply sincerely in state $(1,0)$, and those who do not. These groups are characterized by a threshold t: agents with γ_i above the threshold apply sincerely, while those with γ_i below the threshold do not. Then, the mass of sincere applicants is $1 - G(t)$. We denote this mixed mechanism by μ_t.

Theorem 3. *For any distribution G, there exists $t_G \in [0,1]$ such that μ_{t_G} is optimal.*

Proof. The designer's objective $\alpha m(\mu_{t_G}) + (1-\alpha)v(\mu_{t_G})$ is a continuous function of t_G, since the steady-state probabilities are continuous in t_G and m and v are continuous in those probabilities. By the Weierstrass extreme value theorem, the objective attains a minimum on the compact set $[0,1]$. By Lemma 2, the state never exceeds $(2,-1)$ or $(-1,2)$ and all agents apply to queues independently of their types in those states, so choosing $t \in [0,1]$ exhausts the generalized Markovian strategy space; the minimizing t_G therefore defines an optimal mechanism in that class. □

In the previous setting, we characterized when the two-state mechanism and the pooling mechanism were optimal. These previously optimal mechanisms admit simple structures: the mixed mechanism with thresholds of $G(0)$ and $G(1)$ respectively. When the threshold is $G(0)$, agents always apply sincerely in state $(1,0)$; when the threshold is $G(1)$, agents never apply sincerely.

E Queue Hopping

In practice, between application cycles, agents can freely switch queues in many lottery mechanisms; permitting such behavior does not change the optimality of the mechanisms in the main text. In fact, under those mechanisms, no agent wishes to switch: the incoming agent always has more information than agents already queued. Thus, if a queued agent were willing to switch, every incoming agent would strictly prefer the queue she switched to. Therefore, mismatch, and hence inefficiency, is high under any stationary mechanism that induces switching. This result applies to stationary markets with continual arrivals; in a static market, swapping could be part of an optimal mechanism.[22]

The model is identical to our baseline model with the exception of the following alteration: In each period, all present agents simultaneously also choose a queue; we denote the period-τ choice of the period-t agent by d_t^τ. Once agents can switch, their types become payoff-relevant and they could in principle track others' types. We rule this behavior out as unrealistic: we assume applicants observe queue lengths, not agent-by-agent histories.

Assumption 3. *Agents' strategies are a function of the state.*

When the first-best mechanism is implementable in the original model, $\gamma \geq \frac{2}{3}$, it remains optimal in the new stationary setting: truthful reporting still gives the current agent the matching good, while switching queues forfeits the current allocation and induces the designer to supply the wrong good next period. We therefore focus on the setting where the first-best cannot be implemented, i.e., when Assumption 1 holds.

The new incentive constraints implied by the ability to switch are never violated by the pooling mechanism. Switching queues merely ensures that the agent cannot receive a good in the given period, and will not change the types of goods the designer supplies in the future.

We show that the natural translation of Section 3.1's two-state mechanism continues to be an equilibrium in agent strategies. As before, μ_q is optimal whenever the pooling mechanism or

[22] Consider a simple static setting with two agents and one good of each type. If both agents initially apply to the same queue, the agent that loses the resulting lottery would prefer to switch queues.

first-best mechanism are not optimal.

Proposition 4. *Agents never switch their queues under μ_q.*

If neither pooling nor the first-best mechanisms are optimal, then μ_q is optimal.

Proof. We first show formally that no agent wishes to change its queue under μ_q on the equilibrium path. We translate μ_q to the current setting by fixing the designer's strategy and incoming agent's strategies. Old agents reenter their queue in every period. Notably, in state $(2,-1)$ incoming agents report type B independently of their type. In addition, we require that agents do not swap the queue they have entered in later periods. This requirement generates two new constraints, one for each possible state.

In state $(1,0)$, it is easy to see that the type-A agent does not wish to switch queues. The probability with which a good of type B is supplied, q, was selected to render incoming type-B agents indifferent between the two queues. Relative to incoming type-B agents, current type-A agents expect less competition in queue A and prefer to match correctly. Then, if an incoming type-B agent is indifferent, current type-A agents strictly prefer to remain in queue A.

Similarly, in state $(2,-1)$ current type-A agents expect the same wait time independent of the queue they select. All incoming agents select queue B and the designer always supplies a type-A good. By remaining in queue A, they compete with one other agent for one good: the other agent from the previous period. By switching to queue B, they compete with one other agent for one good as well: in this case, the incoming agent. Furthermore, in the event the agent does not receive a good in the current period, they prefer state $(1,0)$ to state $(0,1)$.

We now prove μ_q is still optimal. Proposition 2 implies that μ_q is optimal among mechanisms where agents remain in their queues. Next, suppose a mechanism involved agents swapping queues with probability k, in state $(2,-1)$, where $0<k<1$. The willingness to randomize implies that present agents are indifferent between the two queues. Such a mechanism must fail to improve upon μ_q with respect to allocative inefficiency in state $(2,-1)$. To see why, note that under the two state mechanism, agents always sincerely apply except when in state $(2,-1)$ in which case they are immediately matched and exit the market. It remains to show that such a mechanism cannot

reduce the proportion of time spent in state $(2,-1)$ through swapping in either state.

Suppose a mechanism involved agents swapping queues in state $(1,0)$ with probability $0 < k < \frac{1}{2}$, while incoming agents apply sincerely. The willingness of a present agent to randomize implies the agent is indifferent between queues. However, the agent knows there is a $\frac{1}{2}$ chance that the incoming agent is of type-B and enters queue B. However, the incoming agent of type A then must strictly prefer to enter queue B. To see why incoming agents prefer to enter queue B, note that there is a $k < \frac{1}{2}$ chance that the present agent enters queue B. If the present agent was indifferent between the two queues, then the incoming agent must have a strict preference for queue B. Then, allocative inefficiency under such a mechanism is equal to that of the pooling mechanism.

Lastly, suppose instead that $\frac{1}{2} \leq k < 1$. For agents to be willing to switch queues, either $\Phi^A(1,0) < \Phi^B(1,0)$ or incoming agents are not applying sincerely. In both cases, allocative inefficiency compares to that of the pooling mechanism, which by assumption is suboptimal. □

F Optimal Exogenous Mechanism

Leshno (ibid.) studies exogenous-supply mechanisms that minimize insincere applications over an arbitrary queueing rule, whereas we restrict attention to lottery mechanisms. His comparison applies in the $\gamma \geq 2$ domain; by Section 3.1, endogenous supply already attains zero inefficiency in that range, while the optimal exogenous mechanism at $\gamma = 2$ sustains only a queue length of 1. His theorem 2 characterizes the misallocation minimizing policy: Load Independent Expected Wait (LIEW). This is distinct from the balanced fixed-supply lottery benchmark we analyze below, which sustains a responsive threshold-2 equilibrium only for $\gamma \in [\frac{5}{2}, \frac{10}{3}]$; both make the same qualitative point that in these exogenous benchmarks the necessary benefit-cost ratio sits far above the $\frac{2}{3}$ that endogenous supply needs, so the two values 2 and $\frac{5}{2}$ are not competing estimates of one quantity.

Under the LIEW policy in this parameter region, all agents expect to wait 2 periods upon entering a non-empty queue. This policy minimizes the average wait time, but nonetheless generates

insincere applications whenever the queue stretches past 1 agent in length. Theorem 2 in Leshno (ibid.) shows that the policy generates a misallocation rate of:

$$m = \frac{2p(1-p)}{(1-p)k + pk + 1},$$

where $k = 2p\gamma - 1 = 1$. In our setting, there is an equal arrival rate of both type-A and type-B agents, so $p = 1/2$. Then, the overall misallocation rate is $m = 1/4$.[23]

F.1 Exogenous Supply Benchmark

We now fix the queueing rule and consider purely exogenous supply, $\Phi^A(s) = \frac{1}{2}$ in every state (the "balanced" setting) so that incoming agents and goods are equally split across types and perfect allocation remains feasible in the long run. This is the natural symmetric benchmark: with symmetric arrivals, a type-blind supply process must carry long-run marginal rate $\frac{1}{2}$ for each type if perfect allocation is to remain feasible. Other type-blind processes (a biased constant rate, or a deterministic or temporally correlated supply calendar) lie outside this benchmark; what distinguishes all of them from endogenous supply is that none can condition the supplied type on the current queue imbalance, which is the operative force below.

The first agent applies sincerely, since both queues have equal wait times. Thereafter, whenever queue B is empty the wait in queue A strictly exceeds that in queue B, so type-B agents always apply sincerely; a type-A agent's choice turns on the wait-time difference, and because supply is exogenous the queue-A wait is increasing in its length. Equilibrium profiles are therefore **threshold strategies**: for some κ, a type-θ agent in state $(s_\theta, s_{\theta'})$ enters queue θ if and only if $s_\theta < \kappa$.

When all agents use the same threshold κ, we denote the expected wait time at the start of the period in state s for an agent in queue ϕ by $w_\phi^\kappa(s)$. In equilibrium, κ must be large enough to ensure that incoming agents no longer wish to apply sincerely when the state reaches $(\kappa, -(\kappa-1))$. When $\kappa > 1$, in equilibrium, two constraints must hold. When there are κ agents in either queue,

[23] Unallocated goods also occur under LIEW, but since LIEW is not designed to reduce unallocated goods, to provide a fairer comparison, we only consider efficiency losses arising from misallocation.

incoming agents must prefer the empty queue. Second, when there are $\kappa-1$ agents in either queue, incoming agents must prefer to apply sincerely. Lemma 1 then implies the following inequalities that relate the threshold κ to the benefit-cost ratio γ:

$$\gamma \leq \frac{\kappa}{2(\kappa+1)}\left[1+w_A^{\kappa}(\kappa,-(\kappa-1))\right]+\frac{1+w_A^{\kappa}(\kappa+1,-\kappa)}{2},$$
$$\gamma \geq \frac{\kappa-1}{2\kappa}\left[1+w_A^{\kappa}(\kappa-1,-(\kappa-2))\right]+\frac{1+w_A^{\kappa}(\kappa,-(\kappa-1))}{2}.$$

The wait times $w_A^{\kappa}(s)$ are the solution to a linear system of $\kappa+1$ equations. We focus on the case $\kappa=2$ to illustrate a point of comparison with the first-best mechanism. To begin, we compute the solution to the system of equations, yielding wait times of $w_A^2(2,-1)=\frac{5}{2}$, $w_A^2(1,0)=2$, and $w_A^2(3,-2)=\frac{10}{3}$. Substituting these values into the above equations implies the following:

Lemma 9 (Minimal Exogenous Equilibrium)**.** *If $\gamma\in[\frac{5}{2},\frac{10}{3}]$ and supply is exogenous, the strategy profile where all agents use a threshold of 2 is an equilibrium.*

Proof of Lemma 9. Under a threshold-m profile, the start-of-period wait times $w_A^m(k,-(k-1))$ solve a finite linear system obtained from the transition matrix, with the probability of an immediate match and the continuation states determined by $\Phi^A(s)=\frac{1}{2}$. For $m=2$ this system is

$$w_A^2(2,-1)=\frac{1}{2}\cdot\frac{1}{2}(1+w_A^2(1,0))+\frac{1}{2}(1+w_A^2(2,-1)),$$
$$w_A^2(1,0)=\frac{1}{2}\left[\frac{1}{2}\cdot\frac{1}{2}(1+w_A^2(1,0))\right]+\frac{1}{2}\left[\frac{1}{2}(1+w_A^2(2,-1))+\frac{1}{2}(1+w_A^2(1,0))\right],$$

with solution $w_A^2(2,-1)=\frac{5}{2}$ and $w_A^2(1,0)=2$; the off-path value $w_A^2(3,-2)=\frac{10}{3}$ follows similarly. Since $\kappa=2$ is an equilibrium only if both $IC_{\kappa}(\kappa)$ and $IC_{\kappa}(\kappa-1)$ hold, the threshold-2 equilibrium requires $\gamma\in[\frac{5}{2},\frac{10}{3}]$. □

As the threshold κ rises, the lowest benefit-to-cost ratio γ that can be sustained in equilibrium also rises. To see why, observe that as the number of agents in a queue increases, each agent expects to wait for a longer period of time in that queue. Critically, when the state is $(\kappa-1,-(\kappa-2))$, the incentives of an incoming type-A agent determine the binding lower bound on γ.

Lemma 10 (Increased threshold requires higher benefit-cost ratio)**.** *As κ increases, the minimum γ under which the threshold κ strategy profile is an equilibrium also increases.*

Proof of Lemma **??**. Raising the threshold from $\kappa-1$ to κ raises wait times: the two profiles differ when a queue holds $\kappa-1$ agents and a same-type agent arrives, who enters the empty queue and matches immediately under the lower threshold but joins the full queue under the higher one, lengthening everyone's wait. Hence $w_A^{\kappa}(s) > w_A^{\kappa-1}(s)$ and $w_A^{\kappa}(s+1) > w_A^{\kappa}(s)$, so $w_A^{\kappa}(\kappa,-(\kappa-1)) > w_A^{\kappa-1}(\kappa-1,-(\kappa-2))$ and $w_A^{\kappa}(\kappa+1,-\kappa) > w_A^{\kappa-1}(\kappa,-(\kappa-1))$. Comparing $IC_{\kappa}(\kappa-1)$ with $IC_{\kappa-1}(\kappa-2)$ then shows the binding lower bound on γ is larger at threshold κ. □

We focus on $\kappa=2$, the minimal threshold that generates a responsive equilibrium. Suppose $\kappa=1$ was the threshold in some equilibrium. Then, in state $(1,0)$, agents enter queue B, no matter their type. Such an equilibrium is not responsive and is equivalent to allocating goods independently of type.

For contrast, if the designer had controlled the supply of goods, the first-best could have been implemented when $\gamma \geq \frac{2}{3}$. Furthermore, the level of allocation inefficiency is higher for equilibria with $\kappa \geq 2$ relative to the first-best implementation. We compute the level of allocation inefficiency under exogenous supply when $\kappa=2$. The resulting steady state has a frequency in states $((1,0),(2,-1))$ of $(\frac{2}{3},\frac{1}{3})$ generating an inefficiency level of $\frac{\alpha}{6}+\frac{1-\alpha}{3}$. This inefficiency is directly increasing in the proportion of time spent in state $(2,-1)$. State $(2,-1)$ inherently contains an unallocated good and furthermore, agents do not apply sincerely. By comparison, when $\gamma \geq \frac{2}{3}$, there is 0 inefficiency when supply is endogenous. The contrast is therefore sharp within this benchmark: the balanced fixed-supply benchmark cannot sustain any responsive equilibrium until γ reaches $\frac{5}{2}$, and by Lemma **??** that requirement only rises with competition, whereas endogenous supply reaches the first-best at $\gamma \geq \frac{2}{3}$ and, as oversubscription thickens the market, the threshold for the first-best *falls* (Figure 4).

This analysis suggests that the ability to control the supply of goods is important for the designer. Even when only a single agent is in a queue, agents are tempted to apply insincerely.