Greedy Matching of Impatient Agents

The Role of Inventory

Angela Kohlenberg

Blood transfusion

Adoption

Ride-sharing

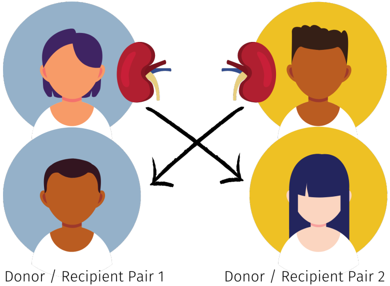
All agents (customers/demand, servers/supply) arrive and depart dynamically

Decisions:

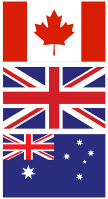
- **Who participates?** (centralized vs. decentralized, pricing, etc.)
- **What are the matching "rules"?** (strategic market design: agent compatibility, incentives, etc.)
- **How should agents be matched in real-time?** (timing, priorities, etc.) [My research]

Tension in dynamic matching: Immediate (greedy) versus delayed matching

Paired kidney exchange



Match timing



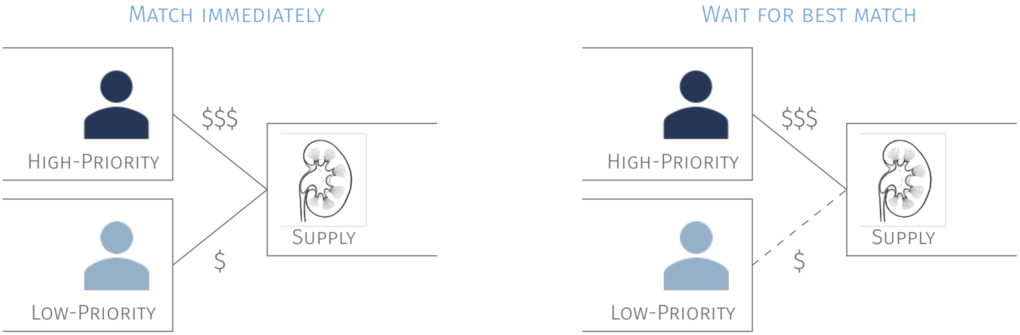
Every 4 months

Every 3 months

Every 3 months

- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching

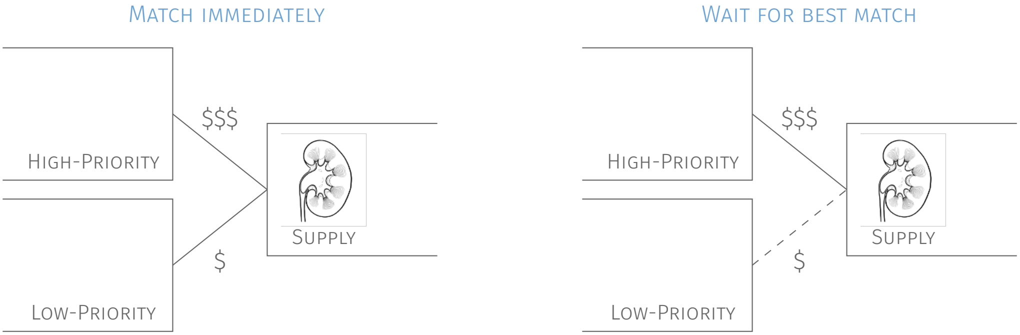


- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching

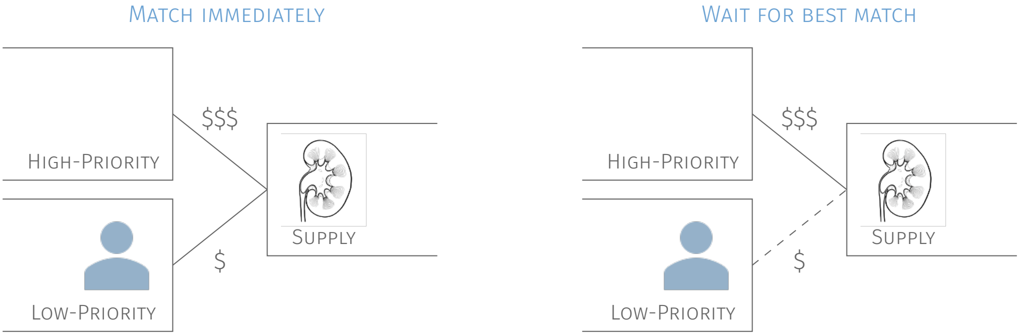
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching



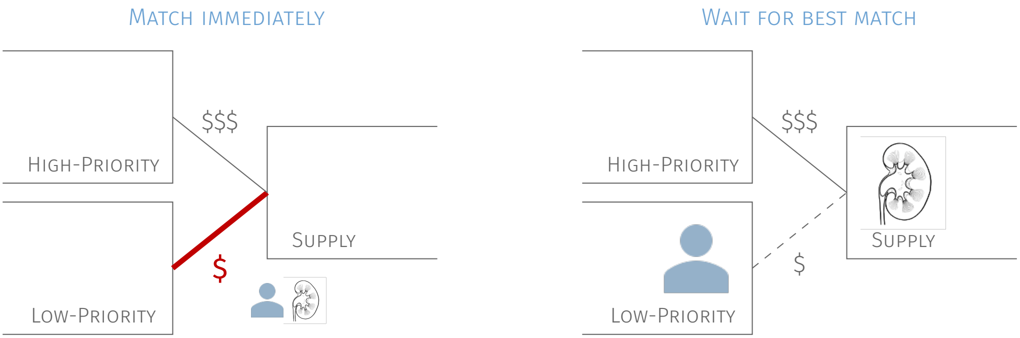
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching



- **Intuition:** Wait to match $\Rightarrow$ Thicker matching market $\Rightarrow$ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching



- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching

- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching

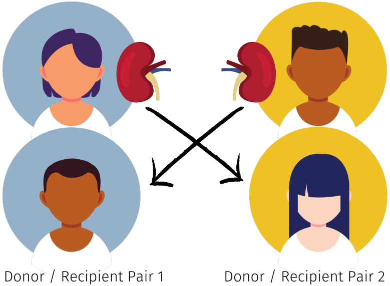
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches

Tension in dynamic matching: Immediate (greedy) versus delayed matching

- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches
- **But, agents do not wait forever...**

Tension in dynamic matching: Immediate (greedy) versus delayed matching

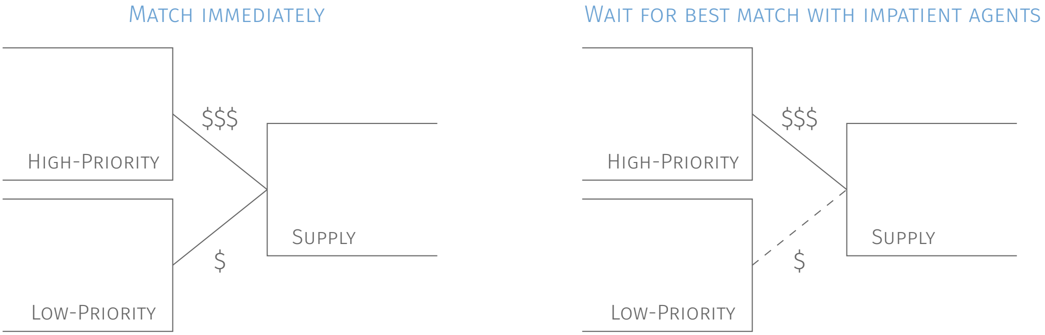
Paired kidney exchange



Match timing

- **Intuition:** Wait to match $\Rightarrow$ Thicker matching market $\Rightarrow$ Better matches
- But, agents do not wait forever...

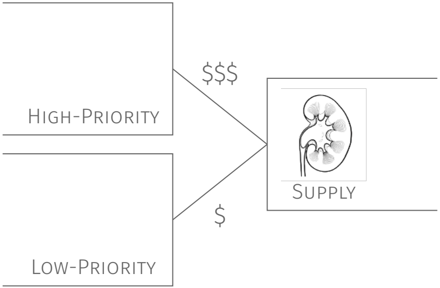
Tension in dynamic matching: Immediate (greedy) versus delayed matching



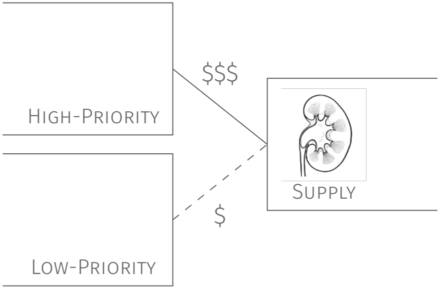
- **Intuition:** Wait to match $\Rightarrow$ Thicker matching market $\Rightarrow$ Better matches
- **But, agents do not wait forever...**

Tension in dynamic matching: Immediate (greedy) versus delayed matching

Match immediately



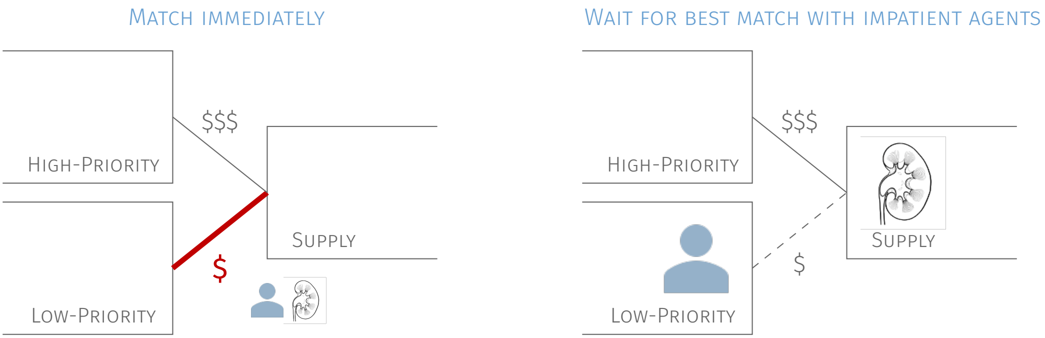
Wait for best match with impatient agents



- **Intuition:** Wait to match $\Rightarrow$ Thicker matching market $\Rightarrow$ Better matches
- **But, agents do not wait forever...**

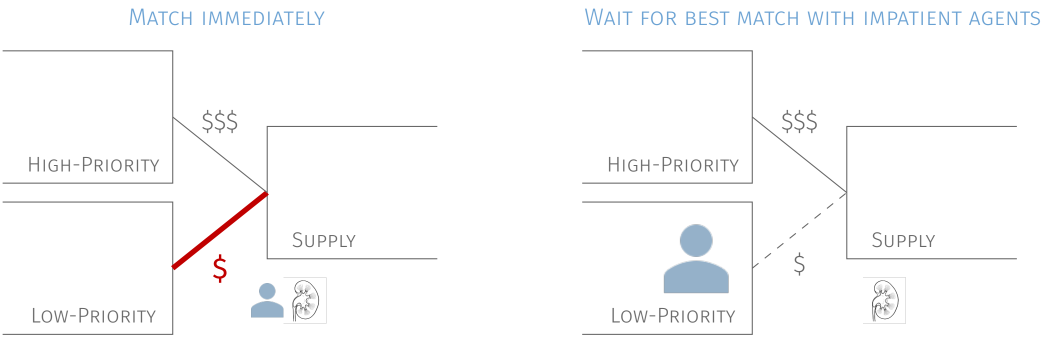
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches
- **But, agents do not wait forever...**

Tension in dynamic matching: Immediate (greedy) versus delayed matching



- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches
- **But, agents do not wait forever...**

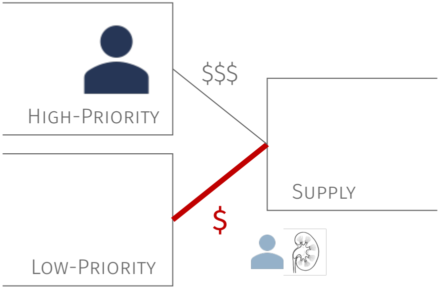
Tension in dynamic matching: Immediate (greedy) versus delayed matching



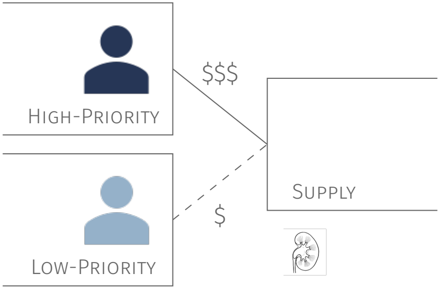
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches
- **But, agents do not wait forever...**

Tension in dynamic matching: Immediate (greedy) versus delayed matching

Match immediately



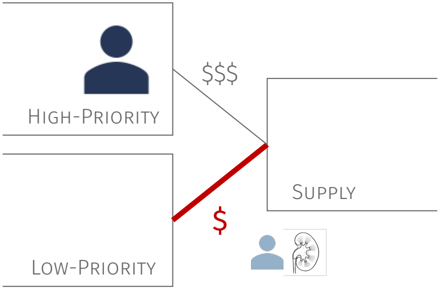
Wait for best match with impatient agents



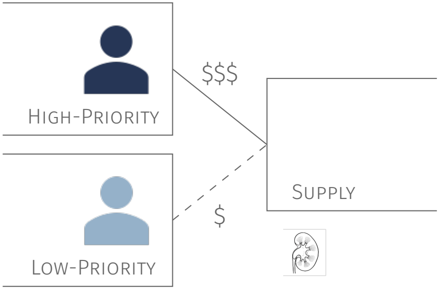
- **Intuition:** Wait to match ⇒ Thicker matching market ⇒ Better matches
- **But, agents do not wait forever...**

Tension in dynamic matching: Immediate (greedy) versus delayed matching

Match immediately



Wait for best match with impatient agents



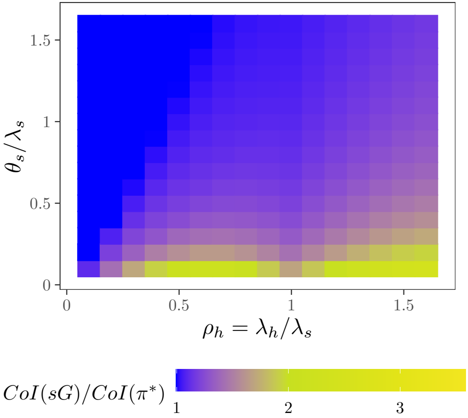
- Ignoring impatience may result in suboptimal decisions

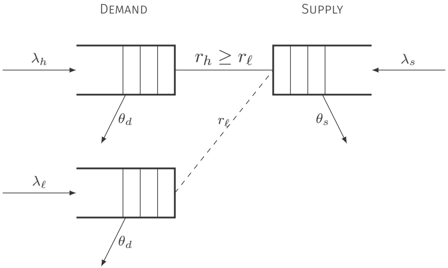
Tension in dynamic matching: Immediate (greedy) versus delayed matching

- Ignoring impatience may result in suboptimal decisions
- What is the optimal timing?

MATCH IMMEDIATELY $[sG]$

$\lambda_s = 10, \lambda_\ell = 10, \theta_d = 2, r_\ell = 1, r_h - r_\ell = 10$





PARAMETER VECTOR:

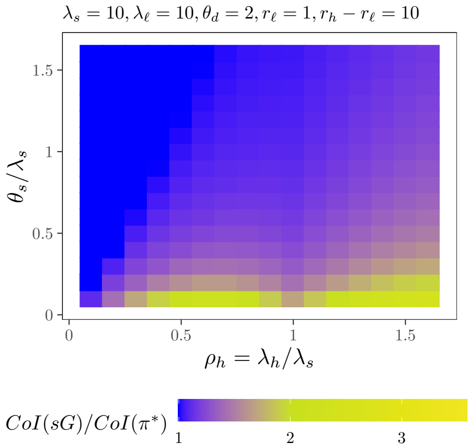
$$p = (\lambda_h,\ \lambda_\ell,\ \lambda_s,\ \theta_d,\ \theta_s,\ r_h,\ r_\ell)$$

OBJECTIVE: MINIMIZE THE REWARD LOSS DUE TO ABANDONMENT, COST-OF-IMPATIENCE $\mathsf{CoI}(\pi, p)$

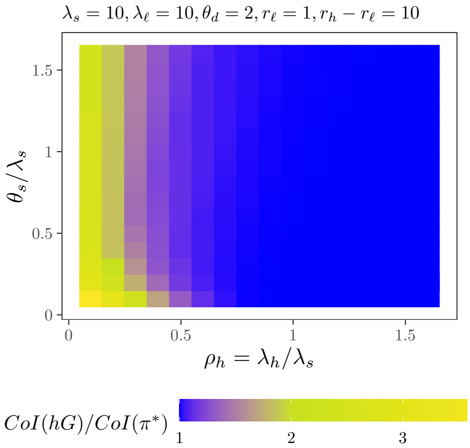
$$\pi^*(p) = \arg\min_{\pi} \mathsf{CoI}(\pi, p)$$

The "extreme" policies: Performance relative to optimal

Match immediately [sG]

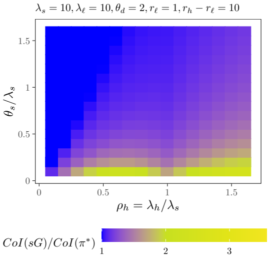


Wait for best match [hG]

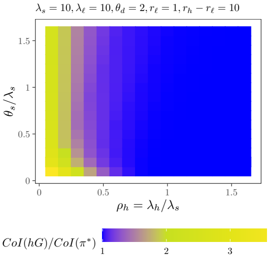


The "extreme" policies: Performance relative to optimal

Match immediately $[sG]$



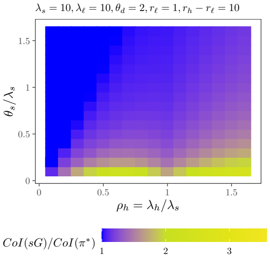
Wait for best match $[hG]$



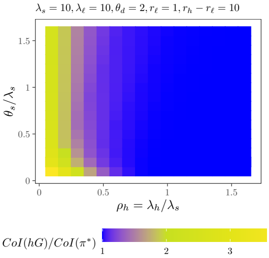
Simply **always** or **never** performing low-reward matches is near-optimal in "many" settings $\Rightarrow$ extends to more complex networks

The "extreme" policies: Performance relative to optimal

Match immediately $[sG]$



Wait for best match $[hG]$



- Simply **always** or **never** performing low-reward matches is near-optimal in "many" settings
- **When, which one, and why** $\Rightarrow$ key determinants of optimal timing

Talk outline

Research & talk overview

Model, approach, and main results

Detailed approach and intuition: Three-agent $\Rightarrow$ full network

Proof outline

Related literature

BLOOD TRANSFUSION

ADOPTION

RIDE-SHARING

IMPATIENT AGENTS:

1. AFFECT PERFORMANCE AND OPTIMAL CONTROL
   - [Performance] **The Cost of Impatience in Dynamic Matching**, A. Kohlenberg and I. Gurvich, *Management Science (2025), 71(4): 2751-3636.*
   - [Control] **Greedy Matching of Impatient Agents**, A. Kohlenberg, *Under review*
2. COMPLICATE DECISION-MAKING
   - **Universal (in parameters) “scaling laws”:** closed-form approximations of reward loss due to abandonment $\Rightarrow$ two-sided queues

BLOOD TRANSFUSION

ADOPTION

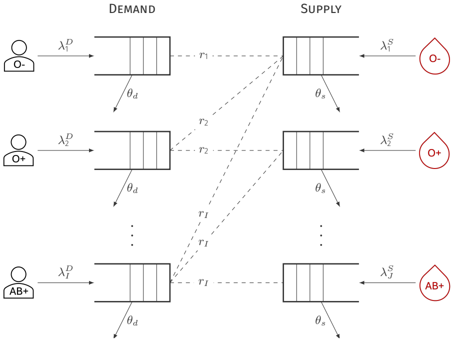
RIDE-SHARING

IMPATIENT AGENTS:

1. AFFECT PERFORMANCE AND OPTIMAL CONTROL
   - [Performance] **The Cost of Impatience in Dynamic Matching**, A. Kohlenberg and I. Gurvich, *Management Science (2025), 71(4): 2751-3636.*
   - [Control] **Greedy Matching of Impatient Agents**, A. Kohlenberg, *Under review*

> BUT, OPTIMAL MATCH TIMING IS NOT COMPLICATED $\Rightarrow$ STATIC NETWORK DESIGN + GREEDY MATCHING [THIS TALK]

Overview: Static network design + Greedy matching



- **Example:** Blood allocation

What: Select subset to match $\Rightarrow$ immediately match selected, never match others

- Match decision does not depend on state at time t (e.g., inventory levels)
- Knapsack-like algorithm identifies subset to match

When: Near optimal in many settings $\Rightarrow$ characterized by "Limited inventory" or "Imbalanced inventory cost"

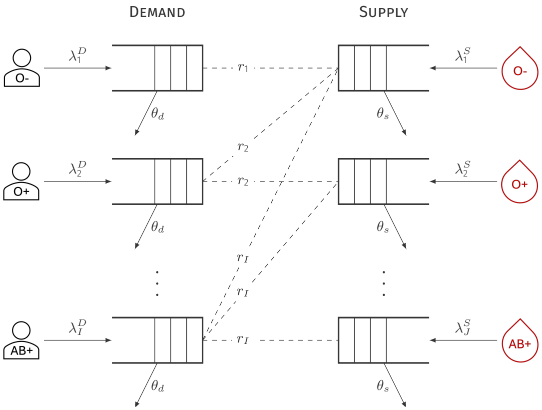
- **When not:** Arrival rates are controllable rather than exogenous

Why: Abandonment limits control via match timing

Talk outline

- Research & talk overview
- Model, approach, and main results
- Detailed approach and intuition: Three-agent $\Rightarrow$ full network
- Proof outline
- Related literature

Two-sided network with dynamic arrivals and departures



Bipartite Graph $(\mathcal{I}, \mathcal{J}, \mathcal{E})$: Demand set $\mathcal{I} = [I]$, supply set $\mathcal{J} = [J]$, edge set $\mathcal{E} \subseteq \mathcal{I} \times \mathcal{J}$

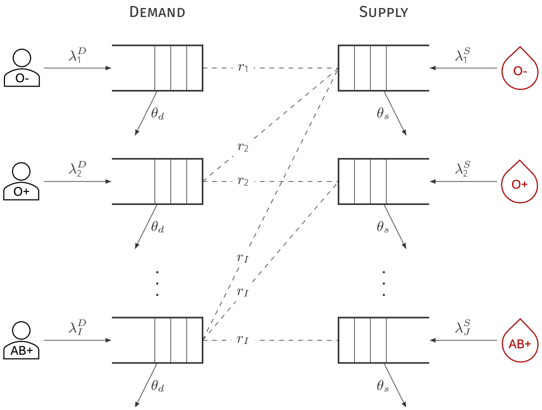
Arrivals: Poisson process with rate λ_i^D for $i \in \mathcal{I}$ and λ_j^S for $j \in \mathcal{J}$

Impatience: Exponential patience with rate θ_d for all $i \in \mathcal{I}$ and θ_s for all $j \in \mathcal{J}$

Rewards: Match $(i, j) \in \mathcal{E}$ yields reward r_i (demand-dependent)

$\Rightarrow$ **Known match prioritization** (for fully-connected networks); **focus is on match timing**

- **Example:** Blood allocation



- **Example:** Blood allocation

Bipartite Graph $(\mathcal{I}, \mathcal{J}, \mathcal{E})$: Demand set $\mathcal{I} = [I]$, supply set $\mathcal{J} = [J]$, edge set $\mathcal{E} \subseteq \mathcal{I} \times \mathcal{J}$

Arrivals: Poisson process with rate λ_i^D for $i \in \mathcal{I}$ and λ_j^S for $j \in \mathcal{J}$

Impatience: Exponential patience with rate θ_d for all $i \in \mathcal{I}$ and θ_s for all $j \in \mathcal{J}$

Rewards: Match $(i, j) \in \mathcal{E}$ yields reward r_i (demand-dependent)

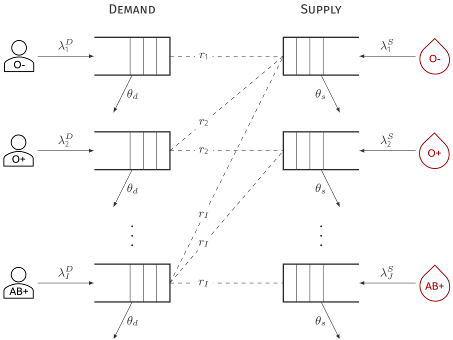
Parameter vector, p:

$$p = (\lambda_i^D, \lambda_j^S, \theta_d, \theta_s, r_i, \forall i \in \mathcal{I}, j \in \mathcal{J})$$

Parameter set, $\mathcal{M}$:

$$p \in \mathcal{M} = [0, \infty)^{2I+J+2}$$

(with some restrictions)



- **Example:** Blood allocation

For any: Bipartite graph $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ and parameters p

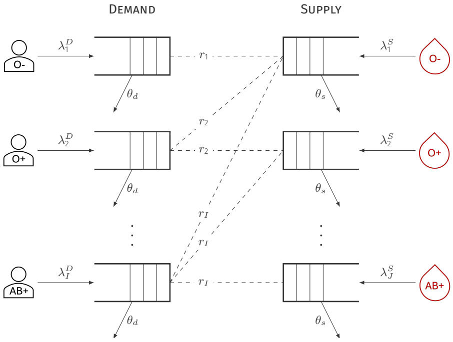
Matching policy, π:

Specify matches to perform at all time $t \geq 0$ based on number of agents in each queue, $Q_i^D(t), Q_j^S(t),\ \forall i \in \mathcal{I}, j \in \mathcal{J}$

Long-run average reward, $v^\pi(p)$:

$$v^\pi(p) = \liminf_{t\uparrow\infty} \frac{1}{t}\mathbb{E}\left[\sum_{(i,j)\in\mathcal{E}} r_i M_{ij}^\pi(t)\right] = \sum_{(i,j)\in\mathcal{E}} r_i m_{ij}^\pi \ ,$$

$M_{ij}^\pi(t)$ is the number of match-(i,j) performed by policy π at time t; m_{ij}^π is the match rate.



- **EXAMPLE:** BLOOD ALLOCATION

FOR ANY: BIPARTITE GRAPH $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ AND PARAMETERS p

OBJECTIVE: DETERMINE POLICY $\pi^*(p)$ THAT MINIMIZES THE **COST-OF-IMPATIENCE, CoI(π, p)**

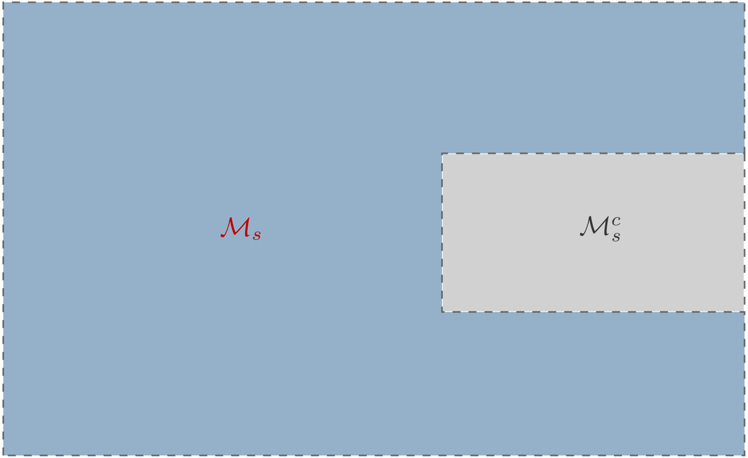
$$\text{CoI}(\pi, p) = \text{Optimal no-abandonment reward, } \bar{v}(p) - \text{Long-run average reward, } v^{\pi}(p)$$

$$\pi^*(p) = \arg\min_{\pi} \text{CoI}(\pi, p) = \arg\min_{\pi} \{\bar{v}(p) - v^{\pi}(p)\}$$

NO-ABANDONMENT UPPER BOUND, $\bar{v}(p)$:

$$\bar{v}(p) = \begin{cases} \max & \sum_{(i,j)\in\mathcal{E}} r_i m_{ij} \\ \text{s.t.} & \sum_{j\in\mathcal{S}_i} m_{ij} \le \lambda_i^D, \quad i \in \mathcal{I} \\ & \sum_{i\in\mathcal{D}_j} m_{ij} \le \lambda_j^S, \quad j \in \mathcal{J} \\ & m_{ij} \ge 0, \quad (i,j) \in \mathcal{E}\,. \end{cases}$$

Approach: Partition parameter set $\mathcal{M}$ + Scale optimality on $\mathcal{M}_s \subseteq \mathcal{M}$



All parameter combinations $p \in \mathcal{M} = [0, \infty)^{2I+J+2}$

Approach: Partition parameter set $\mathcal{M}$ + Scale optimality on $\mathcal{M}_s \subseteq \mathcal{M}$

A policy π is scale optimal on set $\mathcal{M}_s$ if:

$$\sup_{p \in \mathcal{M}_s} \frac{\mathrm{CoI}(\pi, p)}{\mathrm{CoI}(\pi^*, p)} \leq \Gamma \, ,$$

for a parameter-independent constant $\Gamma \geq 1$

All parameter combinations $p \in \mathcal{M} = [0, \infty)^{2I+J+2}$

Approach: Partition parameter set $\mathcal{M}$ + Scale optimality on $\mathcal{M}_s \subseteq \mathcal{M}$

A policy π is scale optimal on set $\mathcal{M}_s$ if:

$$\sup_{p \in \mathcal{M}_s} \frac{\text{CoI}(\pi,p)}{\text{CoI}(\pi^*,p)} \leq \Gamma\ ,$$

for a parameter-independent constant $\Gamma \geq 1$

π is not scale optimal on set $\mathcal{M}_s^c$ if:

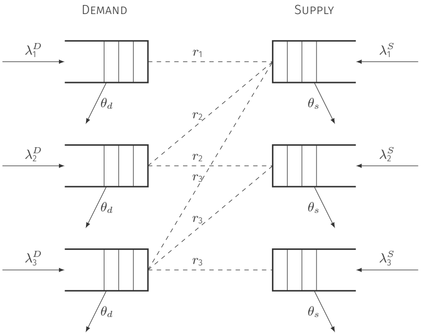
$$\exists p^n \in \mathcal{M}_s^c,$$

$$\text{CoI}(\pi, p^n)/\text{CoI}(\pi^{*,n}, p^n) \to \infty$$

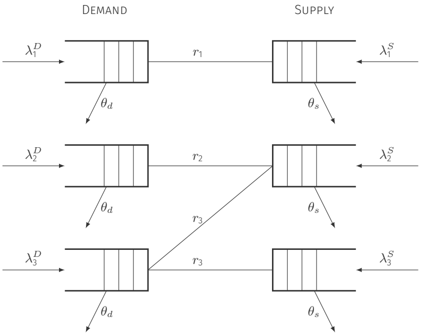
All parameter combinations $p \in \mathcal{M} = [0, \infty)^{2I+J+2}$

Family of Restricted Greedy policies: Greedy policy on $\mathcal{E}_r \subseteq \mathcal{E}$

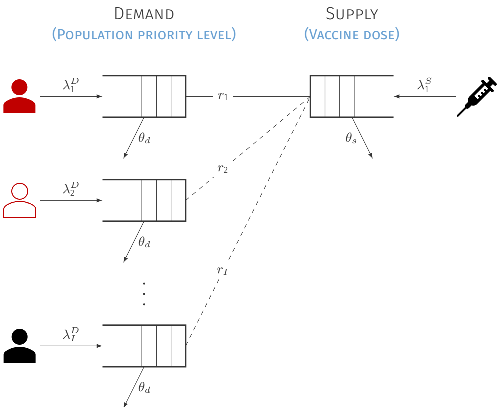
Original network $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ and parameters p



Optimized network $(\mathcal{I}, \mathcal{J}, \mathcal{E}_r^*(p))$



- **Restricted Greedy, RG($\mathcal{E}_r$):** Immediately match $(i,j) \in \mathcal{E}_r$ if $Q_i^D(t), Q_j^S(t) \geq 1$; never match $(i,j) \in \mathcal{E} \backslash \mathcal{E}_r$
- **Prioritize:** **Highest-reward** if $\mathcal{E}_r^c$ is fully-connected, or **Longest-queue** if $\mathcal{E}_r^c$ is partially connected, where $\mathcal{E}_r = \bigcup_{c \in [C]} \mathcal{E}_r^c$



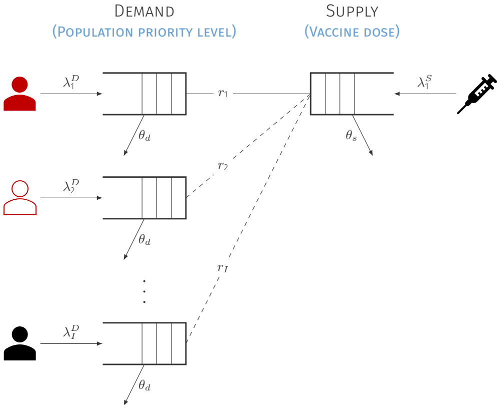
Input: Network $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ and parameters p

Output: Subset of edges $\mathcal{E}_r^*(p) \subseteq \mathcal{E}$

Iterative process:

1. Add highest-reward demand type to “knapsack”
2. Decision based on **closed-form parameter conditions**: Add next demand type?
3. Stop if next type is rejected

Knapsack-like algorithm to identify optimal $\mathcal{E}_r^*(p)$



Input: Network $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ and parameters p

Output: Subset of edges $\mathcal{E}_r^*(p) \subseteq \mathcal{E}$

Iterative process:

1. Add highest-reward demand type to "knapsack"
2. Decision based on **closed-form parameter conditions**: Add next demand type?
3. Stop if next type is rejected

Parameters with non-empty output: $\mathcal{M}_{RG} = \{p : \mathcal{E}_r^*(p) \neq \emptyset\}$

MAIN RESULT: RG($\mathcal{E}_r$), FOR SOME $\mathcal{E}_r \subseteq \mathcal{E}$, IS SCALE OPTIMAL ON $\mathcal{M}_{RG}$

$\mathcal{M}_{RG}$: RG ON $\mathcal{E}_r^*(p)$ IS SCALE OPTIMAL

$$\sup_{p\in\mathcal{M}_{RG}} \frac{\text{CoI}(RG,p)}{\text{CoI}(\pi^*,p)} \leq \Gamma$$

$\mathcal{M}_{RG}^c$: RG ON ALL $\mathcal{E}_r \subseteq \mathcal{E}$ IS NOT SCALE OPTIMAL

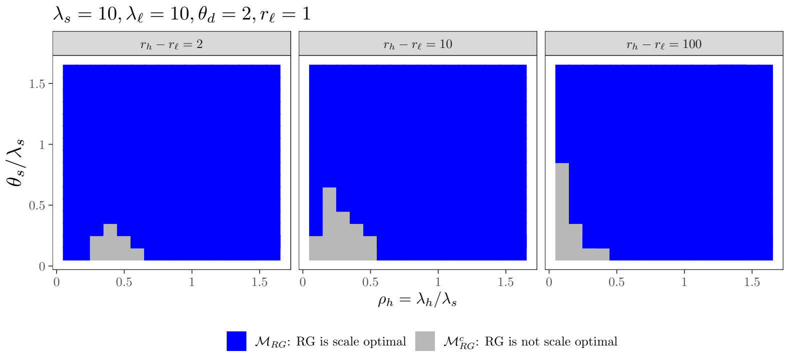
$$\forall \mathcal{E}_r \subseteq \mathcal{E},\ \exists p^n \in \mathcal{M}_{RG}^c$$
$$\text{CoI}(RG,p^n)/\text{CoI}(\pi^{*,n},p^n) \to \infty$$

ALL PARAMETER COMBINATIONS $p \in \mathcal{M} = [0,\infty)^{2I+J+2}$

SHOWN NEXT:

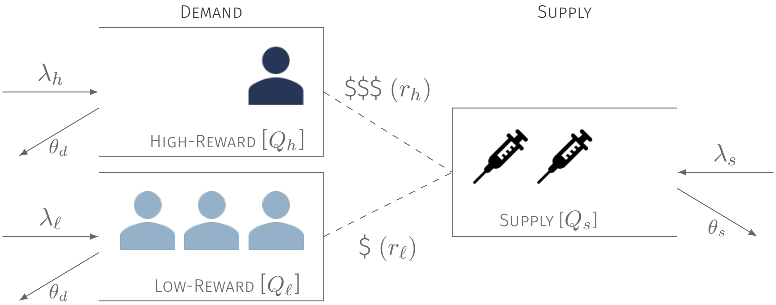
- $\mathcal{M}_{RG}$ IS "LARGE"
- RG ON $\mathcal{E}_r^*(p)$ IS "VERY CLOSE" TO OPTIMAL
- $\mathcal{M}_{RG}$ AND $\mathcal{E}_r^*(p)$ ARE EXPLICITLY DEFINED BY **CLOSED-FORM, INTERPRETABLE** EXPRESSIONS OF p

Numerical results: Three-agent network



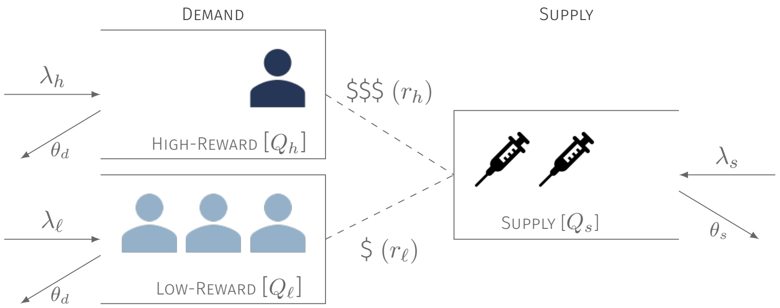
For 2,304 different parameter vectors p, representing wide range of $\mathcal{M}$:

- **86%** are in $\mathcal{M}_{RG}$
- For $p \in \mathcal{M}_{RG}$, CoI(RG, p) is within **1%** of the optimal CoI(π^*, p) on average (stdev **3%**, max **32%**; max hG or sG over all p is **1,383%**)



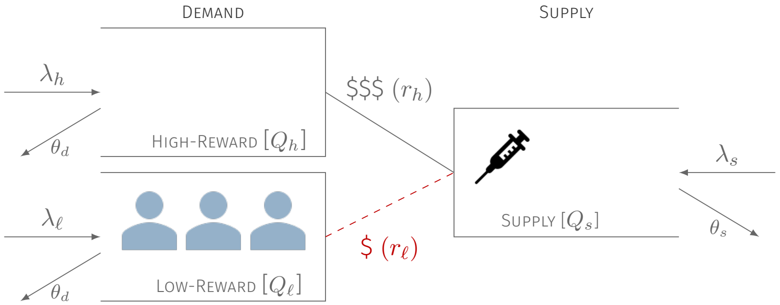
- **PARAMETERS:** $\hat{p} = (\lambda_h, \lambda_\ell, \lambda_s, \theta_d, \theta_s, r_h, r_\ell)$
- **OPTIMAL NO-ABANDONMENT REWARD:** $\bar{v}(\hat{p}) = r_h\lambda_h + r_\ell(\lambda_s - \lambda_h)$ [IF $\lambda_h \leq \lambda_s \leq \lambda_h + \lambda_\ell$]

The three-agent network: No-abandonment upper bound



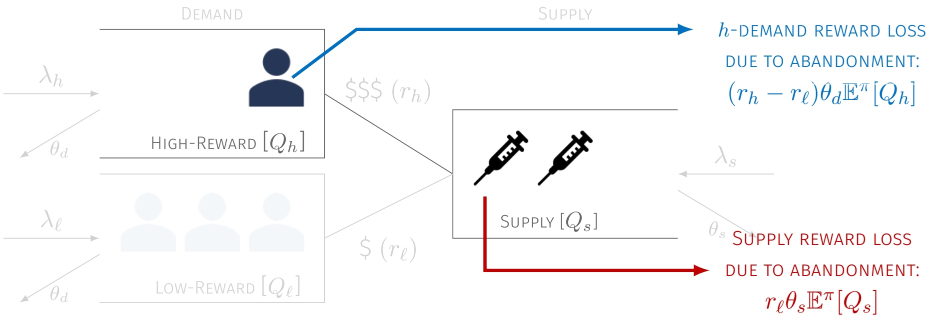
- Parameters: $\hat{p} = (\lambda_h, \lambda_\ell, \lambda_s, \theta_d, \theta_s, r_h, r_\ell)$
- Optimal no-abandonment reward: $\bar{v}(\hat{p}) = r_h\lambda_h + r_\ell(\lambda_s - \lambda_h)$ [if $\lambda_h \leq \lambda_s \leq \lambda_h + \lambda_\ell$]
- Average reward under policy π with impatient agents: $v^\pi(\hat{p}) \leq \bar{v}(\hat{p})$

Objective: Minimize the cost-of-impatience



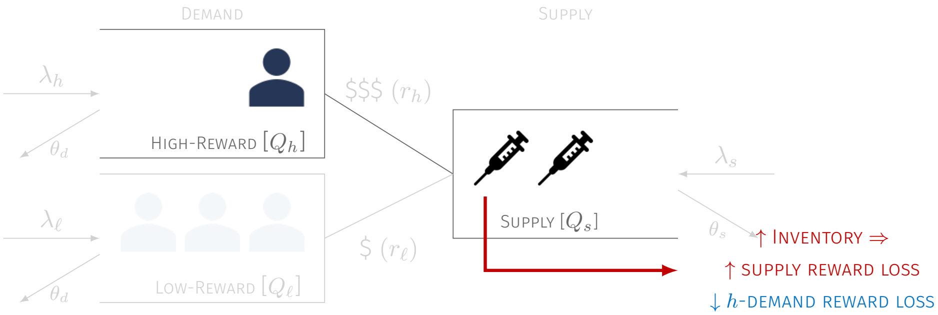
> CoI$(\pi, \hat{p})$ = Optimal no-abandonment reward, $\bar{v}(\hat{p})$ − Average reward under policy π, $v^{\pi}(\hat{p})$

Objective: Minimize the cost-of-impatience



$$
\begin{aligned}
\text{CoI}(\pi, \hat{p}) &= \text{Optimal no-abandonment reward, } \bar{v}(\hat{p}) - \text{ Average reward under policy } \pi,\ v^\pi(\hat{p}) \\
&= \{r_h\lambda_h + r_\ell(\lambda_s - \lambda_h)\} - \{r_h(\lambda_h - \theta_d\mathbb{E}^\pi[Q_h]) + r_\ell(\lambda_s - \lambda_h - \theta_s\mathbb{E}^\pi[Q_s] + \theta_d\mathbb{E}^\pi[Q_h])\} \\
&= (r_h - r_\ell)\theta_d\mathbb{E}^\pi[Q_h] + r_\ell\theta_s\mathbb{E}^\pi[Q_s]
\end{aligned}
$$

Objective: Minimize the cost-of-impatience



Ideal control: Adjust match timing $\Rightarrow$ Adjust inventory (waiting supply) $\Rightarrow$ Balance loss

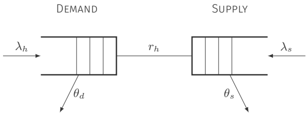
$$\underbrace{(r_h - r_\ell)\theta_d \mathbb{E}^\pi[Q_h]}_{} \approx \underbrace{r_\ell \theta_s \mathbb{E}^\pi[Q_s]}_{}$$

$\downarrow$ Wait for high-reward match
$\uparrow$ Match immediately

$\uparrow$ Wait for high-reward match
$\downarrow$ Match immediately

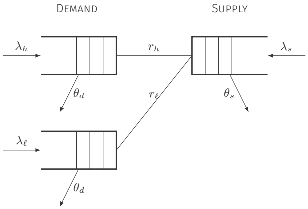
RG ON $\mathcal{E}_r = \{(h,s)\}$ $[hG]$:

ALWAYS WAIT FOR HIGH-REWARD
(ALWAYS PRIORITIZE h-DEMAND)



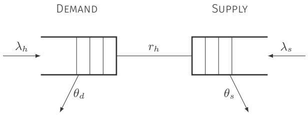
RG ON $\mathcal{E}_r = \{(h,s),(\ell,s)\}$ $[sG]$:

ALWAYS MATCH IMMEDIATELY
(ALWAYS PRIORITIZE SUPPLY)



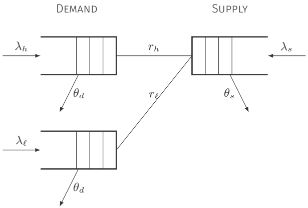
RG on $\mathcal{E}_r = \{(h,s)\}$ [hG]:

Always wait for high-reward (always prioritize h-demand)



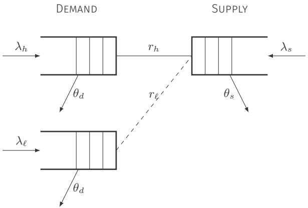
RG on $\mathcal{E}_r = \{(h,s),(\ell,s)\}$ [sG]:

Always match immediately (always prioritize supply)



Non-greedy (e.g., threshold):

Occasionally match (ℓ, s) (prioritization changes)

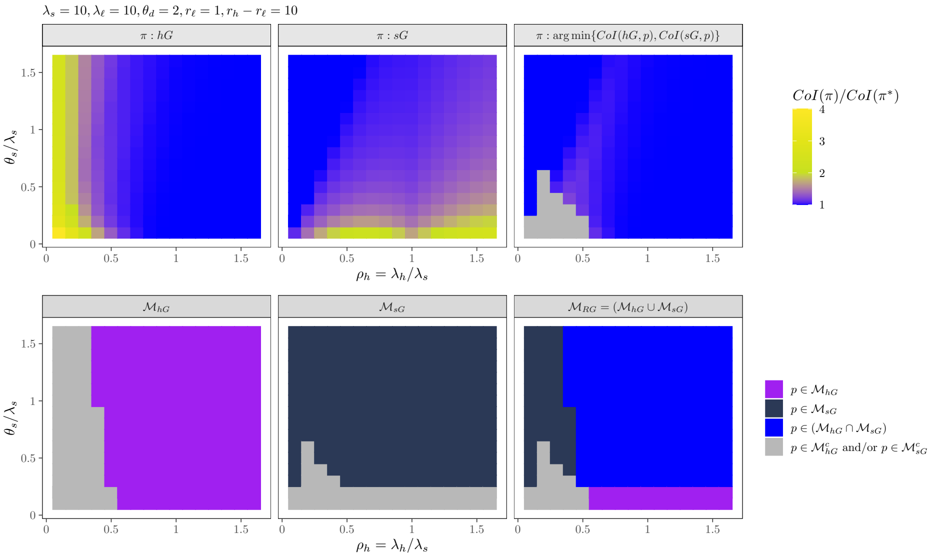


$$\mathcal{M}_{hG} = \left\{ \hat{p} : \underbrace{\left(\sqrt{\frac{\lambda_s}{\theta_s}}(1-\rho_h) \le 1 \text{ and } \theta_d \ge \theta_s \right)}_{\text{LIMITED INVENTORY AND IMPATIENT DEMAND}} \quad \text{or} \quad \underbrace{\left(r_\ell \theta_s \mathbb{E}^{hG}[Q_s] \le (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] \right)}_{\text{LOW INVENTORY COST}} \right\}$$

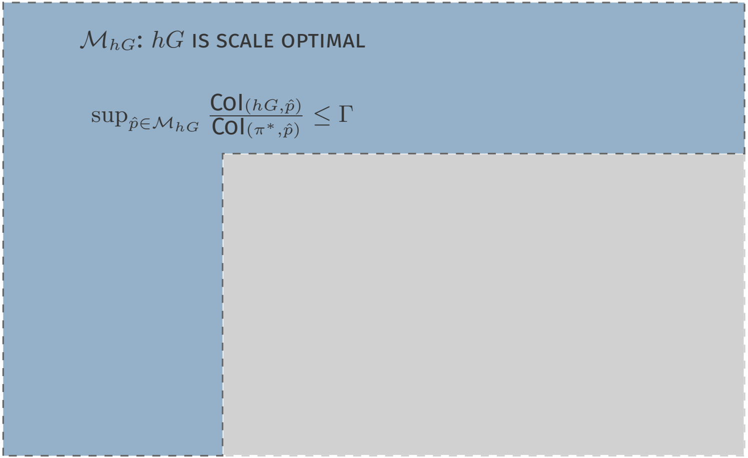
$$\mathcal{M}_{sG} = \left\{ \hat{p} : \underbrace{\left(\sqrt{\frac{\lambda_s}{\theta_s}}(1-\rho_h) \le 1 \text{ and } \theta_s \ge \theta_d \right)}_{\text{LIMITED INVENTORY AND IMPATIENT SUPPLY}} \quad \text{or} \quad \underbrace{\left(r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \ge (r_h - r_\ell)\theta_d \mathbb{E}^{sG}[Q_h] \right)}_{\text{HIGH INVENTORY COST}} \right\}$$

CLOSED-FORM APPROXIMATIONS OF $\mathbb{E}^{sG}[Q_s], \mathbb{E}^{hG}[Q_h]$ BASED ON TWO-SIDED QUEUE SCALING LAWS IN KOHLENBERG AND GURVICH (2025)

Numerical results



Scale optimality of hG and sG



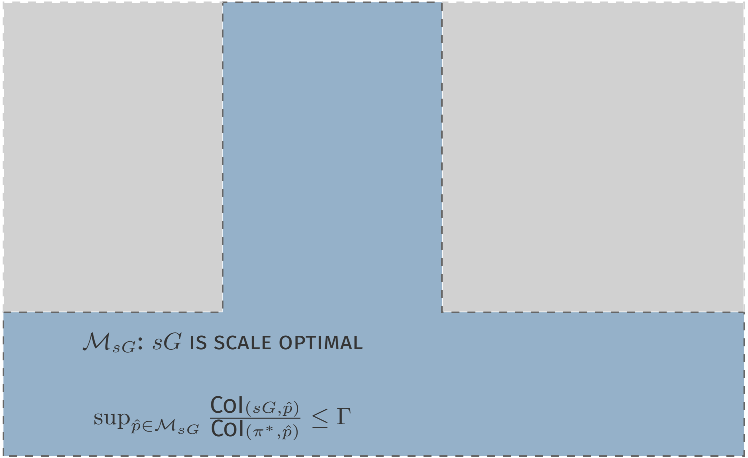
All parameter combinations $\hat{p} \in \mathcal{M} = [0, \infty)^7$

Numerical results (2,304 $\hat{p}$):

For $\hat{p} \in \mathcal{M}_{hG}$, CoI$(hG, \hat{p})$ is within **3%** of optimal CoI$(\pi^*, \hat{p})$ on average (stdev **10%**)

For $\hat{p} \in \mathcal{M}_{hG}^c$, CoI$(hG, \hat{p})$ is within **173%** of optimal CoI$(\pi^*, \hat{p})$ on average (stdev **185%**)

Scale optimality of hG and sG



All parameter combinations $\hat{p} \in \mathcal{M} = [0, \infty)^7$

Numerical results (2,304 $\hat{p}$):

For $\hat{p} \in \mathcal{M}_{sG}$, CoI$(sG, \hat{p})$ is within **18%** of optimal CoI$(\pi^*, \hat{p})$ on average (stdev **23%**)

For $\hat{p} \in \mathcal{M}^c_{sG}$, CoI$(sG, \hat{p})$ is within **71%** of optimal CoI$(\pi^*, \hat{p})$ on average (stdev **55%**)

Scale optimality of hG and sG

$\mathcal{M}_{RG} = (\mathcal{M}_{hG} \cup \mathcal{M}_{sG})$: RG on $\mathcal{E}_r^*(\hat{p})$ is scale optimal

$\mathcal{M}_{RG}^c$: RG on all $\mathcal{E}_r \subseteq \mathcal{E}$ is not scale optimal

$$\forall \mathcal{E}_r \subseteq \mathcal{E},\ \exists \hat{p}^n \in \mathcal{M}_{RG}^c$$

$$\text{CoI}(RG, \hat{p}^n)/\text{CoI}(\pi^{*,n}, \hat{p}^n) \rightarrow \infty$$

All parameter combinations $\hat{p} \in \mathcal{M} = [0, \infty)^7$

Numerical results (2,304 $\hat{p}$):

- **86%** of $\hat{p}$ in $\mathcal{M}_{RG}$
- For $\hat{p} \in \mathcal{M}_{RG}$, CoI$(RG, \hat{p})$ is within **1%** of optimal CoI$(\pi^*, \hat{p})$ on average (stdev **3%**)

When is RG near-optimal? Limited inventory or Imbalanced costs

$$\mathcal{M}_{RG} = \Bigg\{ \hat{p} : \underbrace{\left(\sqrt{\frac{\lambda_s}{\theta_s}}(1-\rho_h) \leq 1 \right)}_{\text{Limited inventory}} \text{ or }$$

$$\underbrace{\left(r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \geq (r_h - r_\ell)\theta_d \mathbb{E}^{sG}[Q_h] \right)}_{\text{High inventory cost}} \text{ or } \underbrace{\left(r_\ell \theta_s \mathbb{E}^{hG}[Q_s] \leq (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] \right)}_{\text{Low inventory cost}} \Bigg\}$$

Limited inventory: h-demand might have to wait, even if all supply is reserved for it

- Supply arrival rate relative to impatience rate and excess capacity, $\rho_h = \lambda_h / \lambda_s$ (match rate)

When is RG near-optimal? Limited inventory or Imbalanced costs

$$\mathcal{M}_{RG} = \left\{ \hat{p} : \underbrace{\left(\sqrt{\frac{\lambda_s}{\theta_s}} (1 - \rho_h) \leq 1 \right)}_{\text{Limited inventory}} \text{ or } \right.$$

$$\left. \underbrace{\left(r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \geq (r_h - r_\ell) \theta_d \mathbb{E}^{sG}[Q_h] \right)}_{\text{High inventory cost}} \text{ or } \underbrace{\left(r_\ell \theta_s \mathbb{E}^{hG}[Q_s] \leq (r_h - r_\ell) \theta_d \mathbb{E}^{hG}[Q_h] \right)}_{\text{Low inventory cost}} \right\}$$

> **Imbalanced inventory cost:** "Cost" of inventory is always higher (lower) than "benefit"
>
> - Relative supply and h-demand impatience rates, arrival rates, reward loss per abandonment

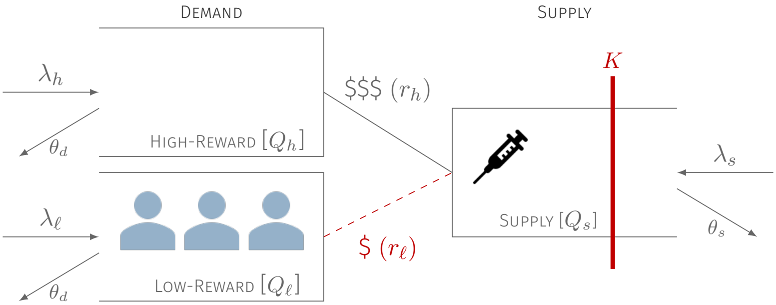
Why is RG near-optimal? Abandonment limits control via match timing

Ideal control: adjust match timing $\Rightarrow$ adjust inventory levels $\Rightarrow$ balance reward loss

$$\underbrace{(r_h - r_\ell)\theta_d \mathbb{E}^\pi[Q_h]}_{\substack{\downarrow \text{ Wait for high-reward match} \\ \uparrow \text{ Match immediately}}} \approx \underbrace{r_\ell \theta_s \mathbb{E}^\pi[Q_s]}_{\substack{\uparrow \text{ Wait for high-reward match} \\ \downarrow \text{ Match immediately}}}$$

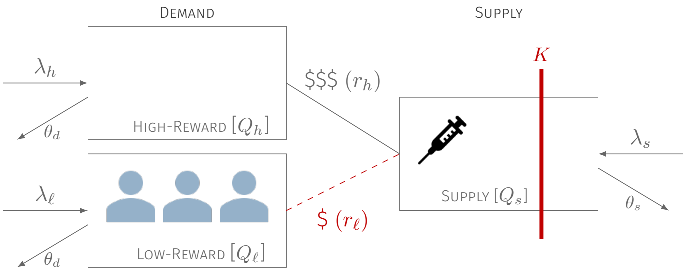
- **Limited ability to control inventory** [Limited inventory]: Inventory levels driven primarily by abandonment and arrival rates (matches)
- **One side "dominates" reward loss** [Imbalanced cost]: Always prioritize one side

Why is RG near-optimal? Threshold policy example



- Threshold K is set so that: $(r_h - r_\ell)\theta_d \mathbb{E}^\pi[Q_h] \approx r_\ell \theta_s \mathbb{E}^\pi[Q_s]$

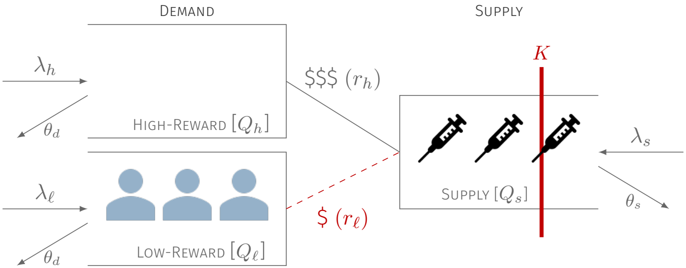
Why is RG near-optimal? Threshold policy example



Threshold does not affect match rate if:

> **Supply is always below K** or ℓ-demand is not available when it exceeds K
>
> - Supply abandons quickly $(\theta_s > \lambda_s)$
> - Demand abandons quickly $(\theta_d > \lambda_h)$
> - Supply matches h-demand quickly $(\lambda_s \approx \lambda_h)$
> - h-demand is very valuable $(r_h - r_\ell \gg r_\ell)$

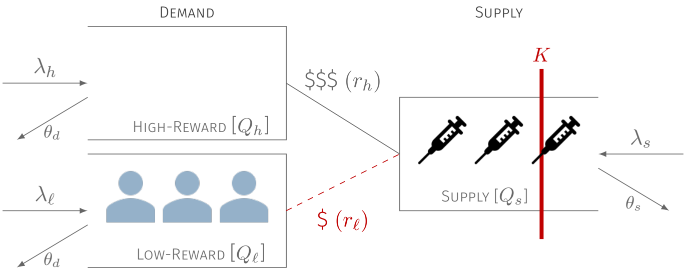
Why is RG near-optimal? Threshold policy example



Threshold does not affect match rate if:

> **Supply always exceeds K**
>
> - Supply abandons slowly ($\theta_s < \lambda_s$)
> - h-demand abandons slowly ($\theta_d < \lambda_h$)
> - Supply matches slowly
> - h-demand is not very valuable ($r_h - r_\ell \approx r_\ell$)

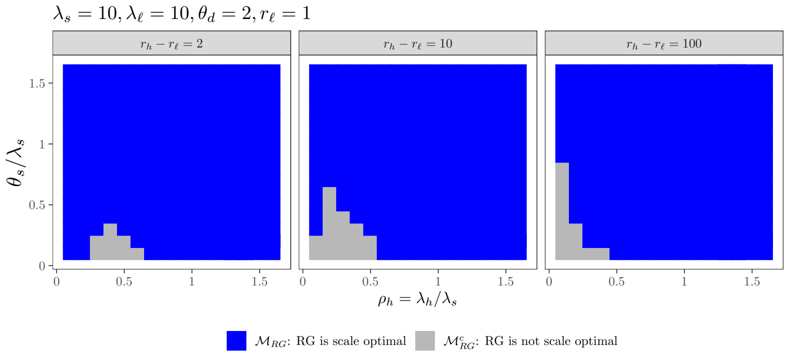
When does RG NOT perform well? Arrival rates are controllable rather than exogenous



Threshold DOES affect match rate if parameters are aligned "just right":

> Supply occasionally falls below and exceeds K, and can match with ℓ-demand when it exceeds K
>
> - Supply is patient "enough"
> - There is much more supply than h-demand, and less than both demand types ($\lambda_h \ll \lambda_s \ll \lambda_h + \lambda_\ell$)
> - h-demand is valuable "enough" but not too valuable
> - Demand is impatient "enough" but not too impatient

WHEN DOES RG NOT PERFORM WELL? ARRIVAL RATES ARE CONTROLLABLE RATHER THAN EXOGENOUS



- **EXAMPLE:** VERY SMALL QUANTITY OF VALUABLE h-DEMAND, AND VERY HIGH QUANTITY OF ℓ-DEMAND ($\lambda_h \ll \lambda_s \ll \lambda_\ell$ AND $\theta_s < \lambda_s$)
- **EXAMPLE:** CONTROLLABLE ARRIVAL OR IMPATIENCE RATES

The full network: Knapsack-like algorithm for network design

Input

Bipartite network $(\mathcal{I}, \mathcal{J}, \mathcal{E})$ and parameters p

Knapsack-like algorithm

Iteration $k \in [I]$ with parameters $\hat{p}^k$:

(1) $\hat{p}^k \in \mathcal{M}_{hG}$: Reject

(2) $\hat{p}^k \in \mathcal{M}_{sG}$: Add

(3) $\mathcal{E}_r^*(p) = \emptyset$

At each iteration, form three-agent network and add or drop agents based on $\mathcal{M}_{hG}$ and $\mathcal{M}_{sG}$

Output

Subset of edges $\mathcal{E}_r^*(p) \subseteq \mathcal{E}$

$\mathcal{E}_r^*(p) \neq \emptyset$

Policy prescription

Restricted greedy on $\mathcal{E}_r^*(p)$

Priority to:

(a) Highest-reward if fully-connected

(b) Longest queue if partially-connected

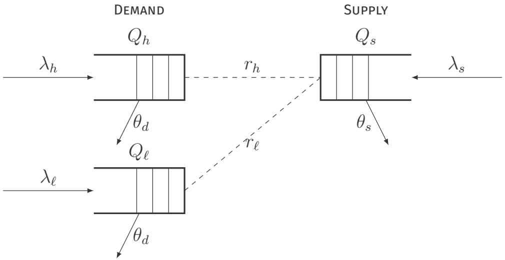
Key proof step: Reduction to a two-sided queue

1. Three-agent network
    1.1 Lower bound: closed-form expression

$\mathbb{E}^{\pi}[Q_s]$ and $\mathbb{E}^{\pi}[Q_s]$: Multi-dimensional Markov chain

Lower bound on CoI$(\pi^*, \hat{p})$.

$$\begin{aligned}\text{CoI}(\pi^*, \hat{p}) &= (r_h - r_\ell)\theta_d \mathbb{E}^{\pi^*}[Q_h] + r_\ell \theta_s \mathbb{E}^{\pi^*}[Q_s] \\ &\geq (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] + r_\ell \theta_s \mathbb{E}^{sG}[Q_s]\end{aligned}$$



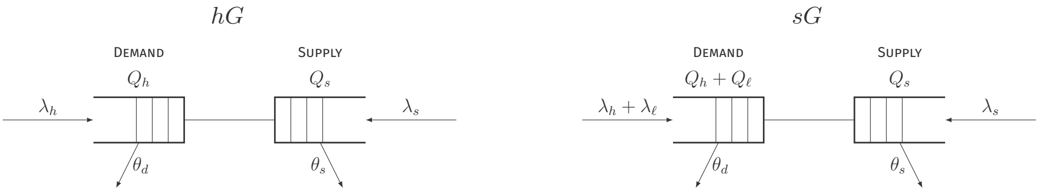
Key proof step: Reduction to a two-sided queue

1. Three-agent network
   1.1 Lower bound: closed-form expression

Lower bound on CoI$(\pi^*, \hat{p})$.

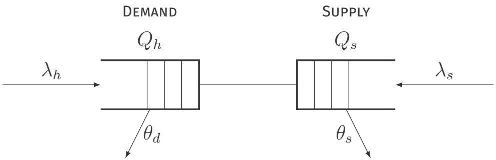
$$\begin{aligned}\mathrm{CoI}(\pi^*, \hat{p}) &= (r_h - r_\ell)\theta_d \mathbb{E}^{\pi^*}[Q_h] + r_\ell \theta_s \mathbb{E}^{\pi^*}[Q_s] \\ &\geq (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] + r_\ell \theta_s \mathbb{E}^{sG}[Q_s]\end{aligned}$$

Two-sided queues under hG and sG:



Key proof step: Reduction to a two-sided queue

1. Three-agent network
   1.1 Lower bound: closed-form expression



(Kohlenberg and Gurvich, 2025, Theorem 1 (informal), Universal Two-Sided Queue Approximation).

For "all" parameter combinations $\hat{p}$,

$$\mathbb{E}^{hG}[Q_h] = \Theta\left(\min\left\{\frac{\rho_h}{1-\rho_h}, \sqrt{\frac{\lambda_h}{\theta_d}}\right\}\right.$$
$$\left.\left(1 + \left[1 + \frac{\rho_h}{1-\rho_h}\frac{\theta_d}{\lambda_h}\min\left\{\frac{\rho_h}{1-\rho_h}, \sqrt{\frac{\lambda_h}{\theta_d}}\right\}\right]\sqrt{\frac{\lambda_s}{\theta_s}(1-\rho_h)e^{\frac{\lambda_s}{\theta_s}(1-\rho_h)^2\mathcal{H}(\rho_h)}}\right)^{-1}\right)$$

where $\rho_h = \lambda_h/\lambda_s < 1$ and $\mathcal{H}(\rho_h) = \sum_{n=1}^{\infty}\frac{1}{n(n+1)}(1-\rho_h)^{n-1}$.

1. Three-agent network
    1.1 Lower bound: closed-form expression
    1.2 Upper bound: conditions under which hG or sG achieves lower bound

Upper bound on CoI$(hG, \hat{p})$ and CoI$(sG, \hat{p})$.

For all $\hat{p} \in \mathcal{M}_{hG}$, $r_\ell \theta_s \mathbb{E}^{hG}[Q_s] \leq \Gamma (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h]$, so that

$$\text{CoI}(hG, \hat{p}) = (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] + r_\ell \theta_s \mathbb{E}^{hG}[Q_s] \leq \Gamma \left((r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] + r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \right)$$

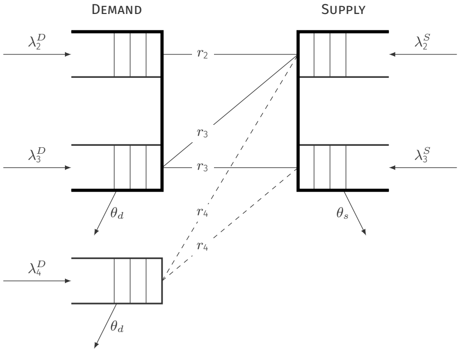
For all $\hat{p} \in \mathcal{M}_{sG}$, $(r_h - r_\ell)\theta_d \mathbb{E}^{sG}[Q_h] \leq \Gamma \max \left\{ (r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h],\ r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \right\}$, so that

$$\text{CoI}(sG, \hat{p}) = (r_h - r_\ell)\theta_d \mathbb{E}^{sG}[Q_h] + r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \leq \Gamma \left((r_h - r_\ell)\theta_d \mathbb{E}^{hG}[Q_h] + r_\ell \theta_s \mathbb{E}^{sG}[Q_s] \right)$$

Key proof step: Reduction to a two-sided queue

1. Three-agent network
   1.1 Lower bound: closed-form expression
   1.2 Upper bound: conditions under which hG or sG achieves lower bound
2. Full network
   Edge set $\mathcal{E}_r \subseteq \mathcal{E}$: Add to or remove from three-agent network, based on conditions $\mathcal{M}_{hG}, \mathcal{M}_{sG}$

- **Fully connected network:** Two-sided queues
- **Partially connected network:** Lyapunov function argument + match longest-queue policy $\Rightarrow$ pooling



Talk outline

Optimality of Greedy Policies - Infinite Patience or Zero Patience

- **Greedy matching policy** is optimal in LARGE MARKETS (Ashlagi et al., 2023) and in suitably preprocessed two-way networks (Kerimov et al., 2025; Gupta, 2024)
- **Greedy policy strictly outperforms batch matching policy (waiting to match)** in some settings (Ashlagi et al., 2023)

No abandonment / Specific set of parameters

Optimal Algorithms - Random Abandonment

- **Batch matching policy is optimal in high-volume regime**: arrival rates scaled up and all other parameters held constant (Aveklouris et al., 2024)

Specific set of parameters

- **Batch matching is arbitrarily bad in some settings** (Aouad and Saritaç, 2022)
- **Randomized matching policy** achieves constant approximation of upper bound on AVERAGE REWARD (Aouad and Saritaç, 2022; Collina et al., 2020)

Single policy for all sets of parameters

Thank you!

Extra slides

Extending to original network

[plain,c,noframenumbering]

Extra slides

Extending to original network

Aouad, A. and Saritaç, Ö. (2022). Dynamic stochastic matching under limited time. *Operations Research*, 70(4):2349–2383.

Ashlagi, I., Nikzad, A., and Strack, P. (2023). Matching in dynamic imbalanced markets. *The Review of Economic Studies*, 90(3):1084–1124.

Aveklouris, A., DeValve, L., Stock, M., and Ward, A. (2024). Matching impatient and heterogeneous demand and supply. *Operations Research*, page forthcoming.

Collina, N., Immorlica, N., Leyton-Brown, K., Lucier, B., and Newman, N. (2020). Dynamic weighted matching with heterogeneous arrival and departure rates. In *Web and Internet Economics: 16th International Conference, WINE 2020, Beijing, China, December 7–11, 2020, Proceedings 16*, pages 17–30. Springer.

Gupta, V. (2024). Greedy algorithm for multiway matching with bounded regret. *Operations Research*, 72(3):1139–1155.

Kerimov, S., Ashlagi, I., and Gurvich, I. (2025). On the optimality of greedy policies in dynamic matching. *Operations Research*, 73(1):560–582.

Kohlenberg, A. and Gurvich, I. (2025). The cost of impatience in dynamic matching: Scaling laws and operating regimes. *Management Science*, 71(4):3303–3319.