Batching and Greedy Policies: How Good Are They in Dynamic Matching?

Myungeun Eom Alejandro Toriello

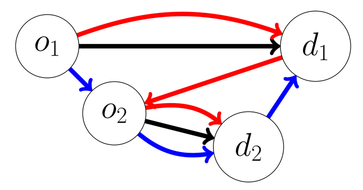
Georgia Institute of Technology, H. Milton Stewart School of Industrial and Systems Engineering

Link to paper

Motivation

Ride-sharing marketplace

- Trip requests dynamically appear over time
- A platform assigns these requests to carriers
- Carriers want to avoid detours
- Platform may "match" requests to reduce total traveling distance



Reward = traveling distance saved ($\sum - \sum$)

Practical policies: Batching and Greedy

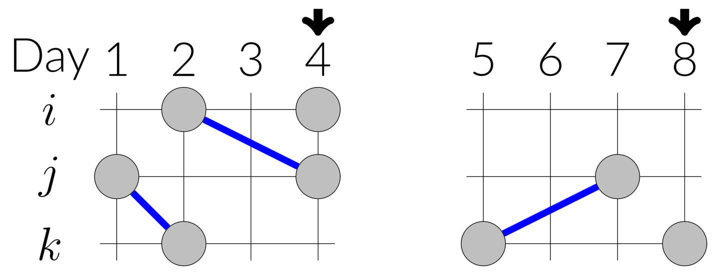
- Limited number of studies analyzing their performance (Aouad and Saritaç, 2022; Ashlagi et al., 2022; Kerimov et al., 2023)

Problem Definition

- Node type set $N = \{1, \dots, n\}$
- Bernoulli arrival probability $p_i \in (0, 1]$, $i \in N$
- Match reward $w_{ij} \geq 0$
- Sojourn period $\tau \in \mathbf{N}$
- Objective: to maximize long-run average reward

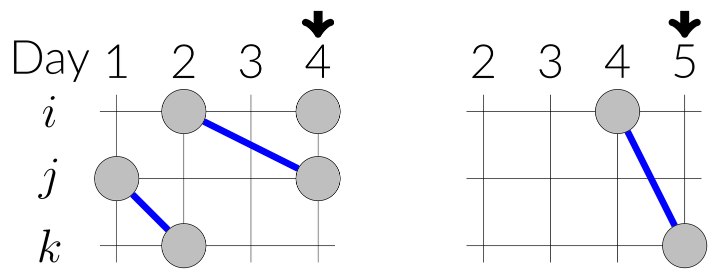
Batching Policy

It solves a max-reward matching and clears out the system every τ periods



(Modified) Greedy Policy

It solves a max-reward matching (with a restricted set of nodes) every period

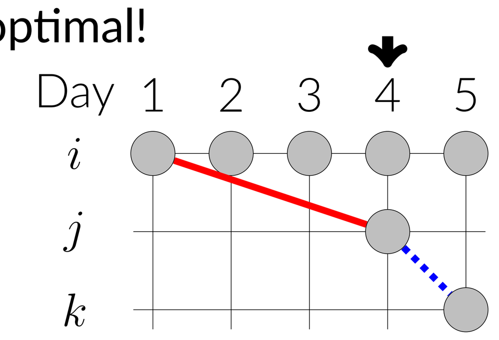


Simple greedy may not be optimal!

$\tau = 4$

$w_{ij} = w_{ik} \approx 0, w_{jk} = 1$

$p_i = 1, p_j = p_k = p < 1$



Optimal policy is to match only j and k whenever available

Contribution

$H(\tau)$: long-run average reward of policy H with sojourn period τ

- $B(\tau) \geq OPT(\tau) - \mathcal{O}(1/\sqrt{\tau})$
- $B(\tau) \geq (1-\epsilon)OPT(\tau) - \mathcal{O}(e^{-\tau\epsilon^2})$ for small $\epsilon > 0$
- $G(\tau) \geq OPT(\tau) - \mathcal{O}(1/\tau)$
- $G(\tau) \geq (1-\epsilon)OPT(\tau) - \mathcal{O}((1+\epsilon)^{-\tau})$ for small $\epsilon > 0$

The only necessary assumption for node arrival distribuion is bounded support.

Benchmark: Fluid LP Relaxation

$$\max_{z \geq 0} \left\{ \sum_{i,j \in N} w_{ij} z_{ij} : \sum_{j \in N} z_{ij} \leq p_i, \forall i \in N \right\}$$

z_{ij}: average matching frequency of a pair $\{i, j\}$

Analysis For a Single Pair $\{i, j\}$

Batching

- Matching frequency: $\mathbf{E}[\min\{A_i, A_j\}]/\tau$ (A_i: # of type i arrivals in τ periods)
- By using basic probability inequalities, we show

$$\min\{p_i, p_j\} - \mathbf{E}[\min\{A_i, A_j\}]/\tau = \begin{cases} \mathcal{O}(1/\sqrt{\tau}) & \text{if } p_i = p_j \\ \mathcal{O}(e^{-\tau(p_j - p_i)^2}) & \text{else} \end{cases}$$

Greedy

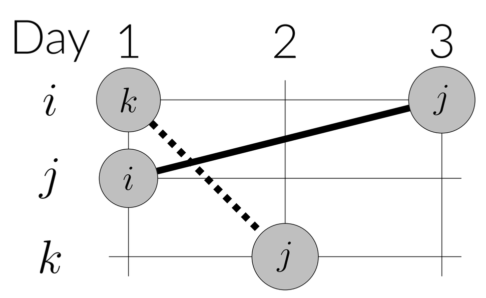
- Analyze a discrete-time Markov chain with state defined to be the sequence of number of (node type, remaining period) combinations
- Matching frequency: $\sum_S \pi(S)\alpha(S)$ (π: stationary dist., α: # of matches)
- By solving balance equations of the chain, we show

$$\sum_S \pi(S)\alpha(S) = \begin{cases} p - \mathcal{O}(1/\tau) & \text{if } p_i = p_j = p \\ p_i - \mathcal{O}(q^{2\tau}) & \text{if } p_i < p_j, q = \frac{1-p_j}{1-p_i} < 1 \end{cases}$$

Analysis For The General Case

1. Randomization

- $\bar{z}_{ij} = z^*_{ij} / \sum_{k \in N} z^*_{ik}$ (normalized, asymmetric) given optimal solution z^*
- Assign arriving type i node to sub-type (i, j) with probability $\bar{z}_{ij}$

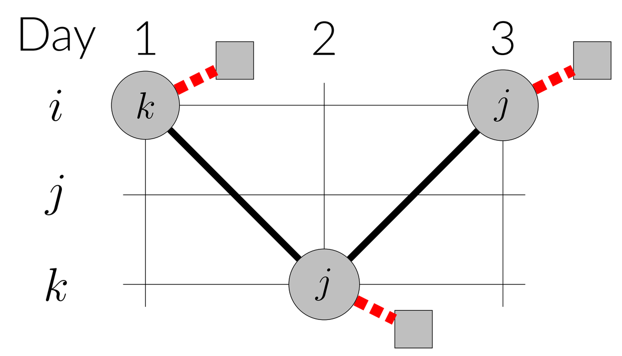


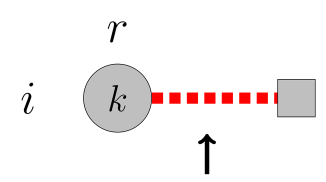
Randomized batching/greedy: Assume edges are compatible only if their sub-types also match

→ Asymptotically optimal!

2. Policy Comparison

- Batching dominates randomized batching
- Modified greedy policy dominates randomized greedy





Reward = Reserve price defined by sub-type and remaining period

Extension: Impatient Nodes

Unmatched node of type i abandons the system w.p. $d_i(\tau)$ in each period
Our main convergence results continue to hold for

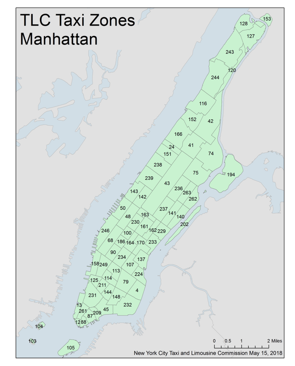
- Batching policy if $d_i(\tau) = \mathcal{O}(\tau^{-\beta})$ for all $i \in N$ and some $\beta > 1$
- Greedy policy if $d_i(\tau) = o(1)$ for all $i \in N$

→ Batching is slightly more vulnerable to impatience than Greedy

Computational Study

Ride-sharing marketplace

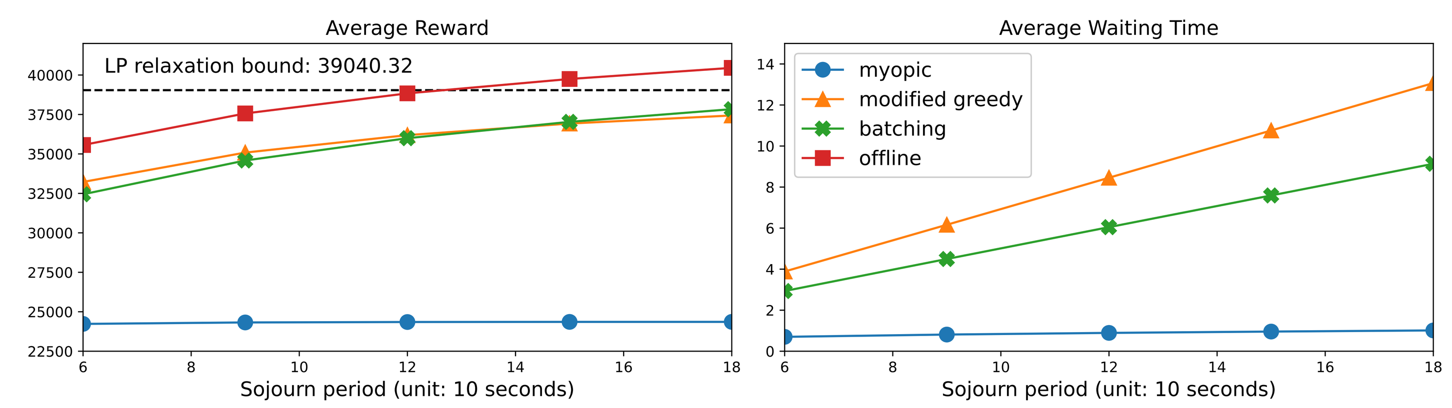
- Manhattan, New York City (69 taxi zones)
- For-hire vehicle record, NYC Open Data platform (February 2020, 6-10 am, Mon to Fri except holidays)
- Match reward for a request pair: traveling miles saved



Simulation parameters

- Policies: myopic, (modified) greedy, batching, offline benchmark
- Time unit: 10 seconds, Time horizon: 1 hour (360 periods), 200 replications
- Sojourn $\tau \in \{6, 9, 12, 15, 18\}$

Result



Greedy/Batching objectives are close to LP bound with small sojourn period!

Future Directions

- Constant departure probability
- Match rewards depending on nodes' waiting times
- Node type-dependent random sojourn period

References

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

Ashlagi, I., Burq, M., Dutta, C., Jaillet, P., Saberi, A., and Sholley, C. (2022). Edge weighted online windowed matching. *Mathematics of Operations Research*, 48(2):999–1016.

Kerimov, S., Ashlagi, I., and Gurvich, I. (2023). Dynamic matching: Characterizing and achieving constant regret. *Management Science*.

NYC Taxi and Limousine Commission (2023). Taxi zone map - manhattan. `https://www.nyc.gov/assets/tlc/images/content/pages/about/taxi_zone_map_manhattan.jpg`. Accessed: 2023-06-21.