Optimization and control of dynamic matching systems

Pascal Moyal
IECL, Université de Lorraine

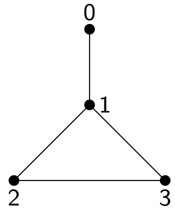
Joint works with Loïc Jean (IECL), Jocelyn Begeot (IECL), Irène Marcovici (Univ. Rouen) and Céline Comte (CNRS).

13ème Atelier d'évaluation de performances - Toulouse, 03/12/2024

Outline

(General) Dynamic matching model

Fix a simple connected graph $G = (V, E)$,



- Items of the various classes in V arrive one by one; their class (r.v. A) is drawn following μ on V.
- Any incoming item is matched, if possible **and profitable**, with a compatible item present in the system.
  Otherwise it is stored in a buffer;
- If several possible matches are possible, the incoming item follows a given matching policy ϕ.

(General) Dynamic matching model

Usual types of **greedy** matching policies:

- 'Priority' type;
- 'Class-uniform': visit the compatible classes in a uniform random order, and pick an item of the first non-empty one.
- 'Match the Longest' (ML), 'Match the Shortest' (MS),...
- Max-weight-type (MW, including ML): if an j-item enters the system, then it chooses a i-item for her match, where j is drawn uniformly from the set

  $$\text{Argmax}_{i\in E(j)\,:\,x(i)>0}\big(x(i)+r_{i,j}\big),$$

  for a fixed set of rewards $r_{i,j}$ on the edges of G.

- FCFM, LCFM, etc.

(General) Dynamic matching model

Bipartite dynamic matching

- The compatibility graph is bipartite: $G = (V_1 \cup V_2, E)$.
- Arrivals occur *pairwise*, by arrivals of the type $(v_1, v_2) \in V_1 \times V_2$.
- Same matching rules as above.

Applications

- **Healthcare systems**: Organ transplants systems, Blood banks... (**bipartite** graphs);
- **Healthcare systems**: Kidney cross-transplants (**general** graphs).
- **Matching interfaces**: Job search, Public Housing allocations (**bipartite** graphs);
- On-line dating (**general** graphs);
- **Collaborative economy**: Peer-to-peer sharing platforms, BlaBlaCar, Uberdrive, Bike-sharing...(**general** graphs);
- Assemble-to-order systems (**general** graphs and **hypergraphs**).
- ...

Bipartite dynamic matching model

1. R. Caldentey, E.H. Kaplan, and G. Weiss. "FCFS Infinite bipartite matching of servers and customers". *Adv. Appl. Probab*, 41(3):695–730, 2009.
2. Visschers, J., Adan, I., and Weiss. "A product form solution to a system with multi-type jobs and multi-type servers." *Queueing Syst.* **70**(3): 269-298, 2012.
3. A. Bušić, V. Gupta, and J. Mairesse. "Stability of the bipartite matching model". *Adv. Appl. Probab.*, 45(2):351–378, 2013.
4. B. Buke, H Chen. Fluid and diffusion approximations of probabilistic matching systems. *Queueing Syst.* **86**(1-2): 1–33, 2017.
5. I. Adan, A. Bušić, J. Mairesse and G. Weiss. Reversibility and further properties of the FCFM Bipartite matching model. *Math. Oper. Research* **43**(2): 598–621, 2018.

General dynamic matching model

1. J. Mairesse and P. Moyal. "Stability of the stochastic matching model", *Journ. Appl. Probab.* **53**(4): 1064-1077, 2016.
2. P. Moyal and O. Perry. "On the instability of matching queues", *Annals Appl. Probab.* **27**(6): 3385-343, 2017.
3. C. Comte, F. Mathieu and A. Bušić. "Stochastic dynamic matching: A mixed graph- theory and linear algebra approach". *ArXiv math/PR: 2112.14457*, 2021.
4. C. Comte. Stochastic non-bipartite matching models and order-independent loss queues. *Stoch. Models* **38**(1): 1–36, 2022.
5. M. Jonckheere, P. Moyal, C. Ramírez and N. Soprano-Loto. "Generalized max-weight policies in stochastic matching", *Stoch. Systems* **13**: 1–19, 2023.

Outline

Stability problem

The **stability region** $\textsc{Stab}(G, \phi)$, is the set of probability measures μ on V rendering the system stable.

Natural necessary condition on μ

$\textsc{Stab}(G, \phi)$ is included in the set

$$\textsc{NCond}(G) := \left\{ \mu : \ \mu(I) < \mu(E(I)) \text{ for all independent sets } I \right\}.$$

Maximality of the stability region

- G is said *stabilizable* if

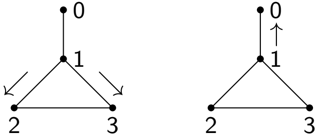
$$\textsc{Stab}(G, \phi) \neq \emptyset \text{ for some } \phi.$$

- ϕ is said *maximal* on a stabilizable graph G if

$$\textsc{Stab}(G, \phi) = \textsc{Ncond}(G).$$

- G is said *maximal* if it is stabilizable, and any ϕ is maximal on G.

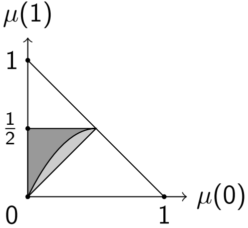
Dependence on the matching policy: example of the 'Paw graph'



Policy A Policy B

Stability regions

B (whole triangle) is maximal whereas A (light grey zone) is not: if $\mu(2) = \mu(3)$,



Main stability results

Theorems

For any connected graph G,

(i) [G is stabilizable] $\iff$ [G is non bipartite];

(ii) [G non bipartite] $\iff$ [Any MW-type policy (and thus, ML) is maximal on G];

(iii) [G non bipartite] $\iff$ [FCFM is maximal on G] (*and the stationary distribution has a product form*).

Outline

Optimization problem

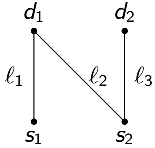
- Suppose now that there is a *cost*, or a *reward* associated to the model:
  1. **Matching rewards**: The various matches are associated to a given reward, e.g. a $\{i,j\}$-matching yields the reward $r_{i,j}$.
  2. **Holding costs**: The waiting times of items in line are associated to given costs: e.g. one time unit for a i-item costs c_i.
  3. **Loss costs**: Items have due dates. The reneging of e.g. items of class i costs d_i.
- We aim at constructing an optimal matching policy to solve Optimization problem 2.
- For this we allow **non-greedy** policies.

Related literature: Optimization of matching models

- Gurvich, I., and Ward, A. (2014). On the dynamic control of matching queues. *Stochastic Systems*, **4**(2): 1–45:
  Lower bound for the long-run cumulative holding costs.
- Ana Bušić and Sean Meyn. Approximate optimality with bounded regret in dynamic matching models. *ACM Sigmetrics Performance Evaluation Review*, 43(2):75–77, 2015: *Approximately optimal policy with bounded regret, in the heavy-traffic regime (holding costs).*
- Nazari, M., and Stolyar, A.L. (2019). Reward maximization in general dynamic matching systems. *Queueing Systems: Theory and Applications* **91**(1): 143–170:
  Optimality of a greedy primal-dual algorithm for the long-term average matching rewards.
- Süleyman Kerimov, Itai Ashlagi, and Itai Gurvich. On the optimality of greedy policies in dynamic matching. *Operations Research*, 2023:
  Hindsight optimality of greedy policies (matching rewards).

Dynamic programming for the bipartite matching model

Consider the following 'N-graph', with two supply classes and two demands classes:

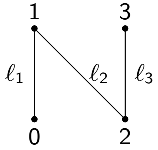


- Items enter by pairs (one supply - one demand);
- Arrival rates are fixed;
- Threshold-type policies are optimal: *do not match any (d_1, s_2) pair until there are K such pairs in the buffer*.

- A. Cadas, A. Bušić and J. Doncel. Optimal control of dynamic bipartite matching models. in *Proceedings of the 12th EAI International Conference on Performance Evaluation Methodologies and Tools*: 39–46, 2019.

A **general** dynamic matching system on the 'N'-graph

Consider again the following 'N-graph':



Our contribution

We extend the latter result to the general matching model (single arrivals)... in which case the situation is a bit more intricate.

Stationary policies

- Class-detail Markov chain of the system:

$$\mathbb{N}^4 - \text{valued sequence}\,(X_n) := ((X_n(0), X_n(1), X_n(2), X_n(3)))\,;$$

- *Admissible policy*: a sequence of Markovian decision rules $(u_n)_{n\in\mathbb{N}}$s.t. for all n,

$$X_{n+1} = u_n(X_n) + A_{n+1}.$$

- *Stationary policy* π: a constant sequence $(u_n)_{n\in\mathbb{N}} \equiv (u)_{n\in\mathbb{N}}$.

Matching on a state $\mathbf{x}$

Le $\mathbf{x}$ be a state of the Markov chain.

- Let $\mathbf{M_x}$ be the set of *matchings* $\mathbf{m}$ on the buffer represented by $\mathbf{x}$ ($\hookrightarrow \mathbf{M_x} = \emptyset$, if π is greedy).
- Let $\mathbf{x} - \mathbf{m}$ be the resulting state after executing matching $\mathbf{x} \in \mathbf{M_x}$.

Settings (II)

Cost function

Linear mapping c of the form

$$c : \left\{ \begin{array}{ccc} \mathbb{N}^4 & \longrightarrow & \mathbb{R}_+ \\ \mathbf{x} := (x_0, x_1, x_2, x_3) & \longmapsto & c_0 x_0 + c_1 x_1 + c_2 x_2 + c_3 x_3 \end{array} \right. ,$$

for $c_0, c_1, c_2, c_3 \geq 0$, and s.t. $c_2 \leq c_0$ and $c_1 \leq c_3$.

Discounted cost problem

For a discount factor $\gamma \in (0,1)$, for any state $\mathbf{x}$,

$$v_\gamma^\pi(\mathbf{x}) = \sum_{n=0}^{+\infty} \gamma^n \mathbb{E}_{\mathbf{x}}^\pi [c(X_n)].$$

We aim at determining the value function, as a mapping of the form

$$v_\gamma : \mathbf{x} \longmapsto \inf_{\pi \in \mathrm{ADM}} v_\gamma^\pi(\mathbf{x}).$$

Settings (III)

Dynamic programming operator

For any real mapping v on $\mathbb{N}^4$ and $\mathbf{x} \in \mathbb{N}^4$, denote

$$L^{\gamma}_{\mathbf{m}} v(\mathbf{x}) = c(\mathbf{x}) + \gamma \mathbb{E}\left[v(\mathbf{x} - \mathbf{m} + A)\right], \text{ for all } \mathbf{m} \in \mathbf{M}_{\mathbf{x}};$$

$$L^{\gamma} v(\mathbf{x}) = \min_{\mathbf{m} \in \mathbf{M}_{\mathbf{x}}} L^{\gamma}_{\mathbf{m}} v(\mathbf{x}) = c(\mathbf{x}) + \gamma \min_{\mathbf{m} \in \mathbf{M}_{\mathbf{x}}} \mathbb{E}\left[v(\mathbf{x} - \mathbf{m} + A)\right].$$

Bellman equation

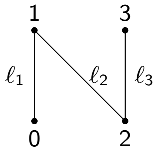
For any γ, the value function v_{γ} solves the fixed point equation

$$v_{\gamma}(\mathbf{x}) = L^{\gamma} v_{\gamma}(\mathbf{x}), \text{ for any state } \mathbf{x}.$$

Threshold-type policy on the 'N-graph'

Good-sense threshold policy:

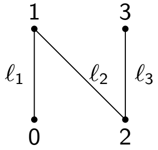
"Do not match any ℓ_2 edge until there are too many such pairs in the buffer."



Threshold-type policy on the ‘N-graph’

More precisely:

Match all possible ℓ_1 and ℓ_3 edges, before matching a certain amount $k_{t_{i(\mathbf{x})}}(\mathbf{x})$ of ℓ_2 edges, according to a threshold $t_{i(\mathbf{x})}$ that depends on the difference between the number of remaining items 0 and 2 in x.



Main result

Theorem (Jean, M' 2024+)

Under the ongoing assumptions, there exists an optimal stationary policy of the threshold type.

Main result

Theorem (Jean, M' 2024+)

Under the ongoing assumptions, there exists an optimal stationary policy of the threshold type.

About (in-)stability

- Despite instability ($\Leftarrow$ **bipartite graph**), the series

$$\sum_{n=0}^{N} \gamma^n \mathbb{E}_{\mathbf{x}}^{\pi}\left[c(X_n)\right]$$

converges for all π.

- This is not the case for the *average cost problem*

$$v_{\gamma}^{\pi}(\mathbf{x}) = \lim_{N \to +\infty} \frac{1}{N} \sum_{n=0}^{N} \mathbb{E}_{\mathbf{x}}^{\pi}\left[c(X_n)\right].$$

(*contrary to the bipartite matching model with pairwise arrivals*).

Main tool for proof

Theorem 6.11.3 (Puterman, 2014)

Suppose that

(i) There exists a mild function $w\colon \mathbb{N}^4 \to \mathbb{R}_+$, such that for all v in

$$V_w = \left\{ v\colon \mathbb{N}^4 \to \mathbb{R}_+ \;:\; \sup_{x \in \mathbb{N}^4} \frac{v(\mathbf{x})}{w(\mathbf{x})} < \infty \right\},$$

there exists a Markovian decision rule u such that $L^\gamma v = L^\gamma_{u(.)} v$.

(ii) There exist two sets $\mathscr{V}$ and $\mathscr{D}$ such that

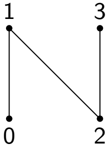
1. $\mathscr{V}$ is stable under L^γ and under pointwise convergence;
2. For $v \in \mathscr{V}$ there exists a deterministic Markovian decision rule $u \in \mathscr{D}$ such that

$$L^\gamma v = L^\gamma_{u(.)} v.$$

Then, there exists an optimal stationary policy $\pi^\star$ structured by a single Markovian decision rule $u^\star \in \mathscr{D}$.

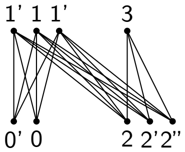
Natural extension

The above result holds if we replace the ‘N-graph’ by any blow-up of the ‘N-graph’, and the cost function is defined accordingly.



Natural extension

The above result holds if we replace the ‘N-graph’ by any blow-up of the ‘N-graph’, and the cost function is defined accordingly.



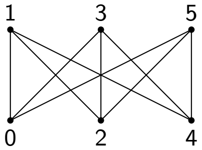
Ongoing work and open questions

- **Determining** or **approximating** the value of the optimal threshold?
- **Learning** the threshold ?
- $\hookrightarrow$ *Done in some cases by (Cadas et al., 2021) for the bipartite model.*

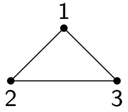
Ongoing work and open questions

Is **Greedy** an optimal policy for the **complete graph**?

- True in the bipartite case (obvious);

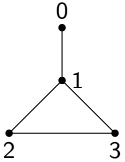


- **... not so obvious** in the general case.



Ongoing work and open questions

Optimal threshold policy for the **paw graph**?



- L. Jean and P. Moyal. "Dynamic programming for the stochastic matching model on general graphs: the case of the 'N-graph'." ArXiv preprint math/PR 2402.01803 (2024).

Outline

An access control problem

- We saw that the system is stable if ran e.g. by FCFM or MW, if μ belongs to the fundamental region

$$\textsc{Ncond}(G) = \{\mu \in \mathscr{M}(V) \,:\, \mu(I) < \mu(E(I)) \text{ for all independent sets } I\}\,.$$

- Suppose that the matching policy is fixed, equal to the above. How do we *construct* a measure μ able to stabilize the system, i.e. a measure $\mu \in \textsc{Ncond}(G)$?

Weighted measures

Definition

Let $G = (V, E)$ be a graph. For any family of weights α on the edges of G, we define the associated positive measure on nodes $\mu^\alpha \in \mathscr{M}(V)$, by

$$\mu^\alpha(i) := \sum_{j \in E(i)} \alpha_{i,j}, \quad i \in V,$$

and $\bar{\mu}^\alpha$ is the associated probability measure. The set of such *weighted probability measures* is denoted by $\mathscr{W}(G) := \{\bar{\mu}^\alpha : \alpha \in \mathcal{M}(E)\}$.

Main result

Theorem

The set $\mathscr{W}(G)$ (and thus, the set of invariant probability measures for reversible random walks on V) coincides with

- The set $\mathrm{NCOND}(G)$, if G is not a bipartite graph;
- The set

  $$\mathrm{NCOND}_2(G) = \left\{ \mu \in \mathcal{M}(V) : \left\{ \begin{array}{l} \forall\, I \in \mathbb{I}(V) \setminus \{V_1, V_2\},\ \mu(I) < \mu(E(I)) \\ \mu(V_1) = \mu(V_2) \end{array} \right. \right\}$$

  if G is a bipartite graph of bipartition $V = V_1 \cup V_2$.

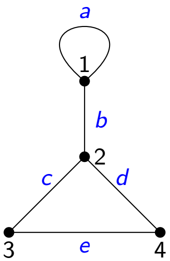
Interest for admission control

- Set a maximal stable policy (FCFM, ML, MW,...).
- To stabilize the system it is sufficient to tune μ so as to belong to $\textsc{Ncond}(G)$.

Interest for admission control

- Set a maximal stable policy (FCFM, ML, MW,...).
- To stabilize the system it is sufficient to tune μ so as to belong to $\textsc{Ncond}(G)$. ... But checking this is of order $O(N^3)$ complexity!
- For this, it is sufficient (and necessary!) to 'throw' a family of weights on the edges of the graph, and then constructing the corresponding weighted measure $\bar{\mu}^{\alpha}$.

Example



Example

Then, we have

$$\begin{cases} \bar{\mu}^\alpha(1) & = \frac{a+b}{a+2b+2c+2d+2e} \\ \bar{\mu}^\alpha(2) & = \frac{b+c+d}{a+2b+2c+2d+2e} \\ \bar{\mu}^\alpha(3) & = \frac{c+e}{a+2b+2c+2d+2e} \\ \bar{\mu}^\alpha(2) & = \frac{d+e}{a+2b+2c+2d+2e} \end{cases}$$

Then, $\bar{\mu}^\alpha \in \textsc{Ncond}(G)$, and is invariant for the reversible Markov chain on $\{1,2,3,4\}$ having transitions

$$\begin{cases} P(1,1) & = \frac{a}{a+b};\ P(1,2) = \frac{b}{a+b}; \\ P(2,1) & = \frac{b}{b+c+d};\ P(2,3) = \frac{c}{b+c+d};\ P(2,4) = \frac{d}{b+c+d} \\ P(3,2) & = \frac{c}{c+e};\ P(3,4) = \frac{e}{c+e}; \\ P(4,2) & = \frac{d}{d+e};\ P(4,3) = \frac{e}{d+e}. \end{cases}$$

Sketch of proof: $\mathscr{W}(G) \subset \mathscr{N}(G)$ or $\mathscr{N}_2(G)$

- Proved directly by hand;
- Already done in Comte (2021) for simple non-bipartite graphs and $\mathscr{N}(G)$.

Sketch of proof: $\mathscr{W}(G) \supset \mathscr{N}(G)$ or $\mathscr{N}_2(G)$

- A: incidence matrix of the compatibility graph.
- **Farkas Lemma**: One, and only one of the following linear systems admits solutions:
  1. the system $A\mathbf{x} = b$, for $\mathbf{x}$ indexed by E satisfying $\mathbf{x} \geq 0$ component-wise ;
  2. the system ${}^tA\mathbf{y} \geq 0$, for $\mathbf{y}$ indexed by V and satisfying ${}^tb\mathbf{y} < 0$ component-wise.

Sketch of proof: $\mathscr{W}(G) \supset \mathscr{N}_2(G)$

- Network-flow problem.

Some related results: Access control

- Begeot, J., Marcovici, I. and Moyal, P. (2023) “Stability regions of systems with compatibilities, and ubiquitous measures on graphs”, *Queueing Systems: Theory and Applications*, **103**: 275–312.
- C. Comte, F. Mathieu, S. Varma and A. Bušić. Online stochastic matching: A polytope perspective. *arXiv preprint* arXiv:2112.14457v5, 2024.

Outline

State space

Let V^* be the free monoid associated to V, and

$$\mathcal{C} = \Big\{ \mathbf{c} \in V^* \ : \ \forall (i,j) \in E, \ |c|_i |c|_j = 0 \Big\}.$$

Buffer detail

At any arrival time point t,

$$C_t = \mathbf{c} = c_1 c_2 ... c_q \in \mathcal{C},$$

where c_j = class of the i-th oldest item in line.

Class detail

At any t,

$$X_t = [C_t] := (|c|_i)_{i \in V} \, \mathcal{X} \subset \mathbb{N}^{|V|} \in$$

Assumptions

- Continuous-time model;
- FCFM policy: *First Come, First Matched*;
- For any buffer detail $\mathbf{c}$, the arrival process of class i-items is
  - $\lambda(i)$, if i has a niehgboring class in $\mathbf{c}$,
  - $\gamma_i([\mathbf{c}])\lambda(i)$, else, for a set of γ_i's satisfying

$$\gamma_i([\mathbf{c}])\gamma_j([\mathbf{c}] + e_i) = \gamma_i([\mathbf{c}] + e_i)\gamma_j([\mathbf{c}]), \quad [\mathbf{c}] \in \mathcal{X}, \quad i, j \in [\![1, n]\!],$$

which is equivalent to saying that for some mapping Γ on $\mathcal{X}$,

$$\gamma_i(\mathbf{c}) = \frac{\Gamma(|\mathbf{c}| + e_i)}{\Gamma(|\mathbf{c}|)} \in [0, 1], \quad \mathbf{c} \in \mathcal{C}, \quad i \in [\![1, n]\!].$$

Examples of balanced access controls

- Decentralized case:

$$\Gamma(x) = \prod_{i=1}^{n} \prod_{\ell=0}^{x_i-1} \gamma_i(\ell) \text{ with } \gamma_i : \{0,1,2,\ldots\} \to (0,1).$$

- Power-laws:

$$\Gamma(x) = \prod_{i=1}^{n} {\gamma_i}^{\varphi_i(x_i)} \text{ with } 0 < \gamma_i < 1 \text{ and } \varphi_i : \{0,1,2,\ldots\} \to (0,+\infty).$$

- Semi-centralized case:

$$\Gamma(x) = \Big(\prod_{i=1}^{n} {\gamma_i}^{x_i}\Big)\Big(\prod_{\substack{i,j=1\\ i\neq j}}^{n} {\gamma_{i,j}}^{x_i x_j}\Big) \text{ with } 0 < \gamma_i < 1 \text{ and } 0 < \gamma_{i,j} < 1.$$

- ...

Score-aware policy gradient

- We aim at optimizing the long-run average reward

$$
\begin{aligned}
v_\Gamma(\theta) &= \lim_{T\to\infty} \mathbb{E}_\Gamma\left(\frac{1}{T}\int_0^T R(C_t)dt\right)\\
&= \sum_{\mathbf{c}\in\mathcal{C}}\sum_{a\in\mathcal{A}}\sum_{r\in\mathcal{R}} r\mathbb{P}(r|\mathbf{c},a)\gamma([\mathbf{c}]|\theta)\pi_\Gamma(\mathbf{c}|\theta),
\end{aligned}
$$

  where
  - $R(C_t)\in\mathcal{R}$ is the reward at time t;
  - $\mathcal{A}=\{\text{enter},\text{not enter}\}$ is the set of actions;
  - π_Γ is the stationary distribution of the CTMC (C_t) under the access control γ.
- The gradient of the mapping v_Γ thus satisfies

$$
\begin{aligned}
\nabla v_\Gamma(\theta) = \sum_{\mathbf{c}\in\mathcal{C}}\sum_{a\in\mathcal{A}}\sum_{r\in\mathcal{R}} & r\mathbb{P}(r|\mathbf{c},a)\gamma([\mathbf{c}]|\theta)\pi(\mathbf{c}|\theta)\\
& \times\left(\nabla\log\pi_\Gamma(\mathbf{c}|\theta)+\nabla\log\gamma([\mathbf{c}]|\theta)\right).
\end{aligned}
$$

Score-aware policy gradient

Procedure:

1. Fix the observation times t_i's and the step sizes α_i's;
2. Start from a point in the state space and a base parameter θ_0.
3. At each observation time t_m:
   1. Take an action (enter/not enter) following γ given θ_m, and observe the next state;
   2. Update

      $$\theta_{m+1} = \theta_m + \alpha_m F(\mathbf{c}_m, m),$$

      for F following the ascending direction of the above gradient.

Example: Admission control in a M/M/1 queue

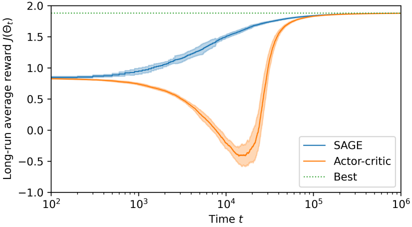


Figure: Bipartite matching model

- Comte, C., Jonckheere, M., Sanders, J., and Senen-Cerda, A. (2023). "Score-Aware Policy-Gradient Methods and Performance Guarantees using Local Lyapunov Conditions: Applications to Product-Form Stochastic Networks and Queueing Systems." *ArXiv math.PR/2312.02804*

Stationary measure: product form

Theorem

Under stability conditions, the stationary measures have the form

$$\pi(\mathbf{c}) = \pi(\varnothing) \prod_{p=1}^{\ell} \frac{\gamma_{c_p}([c_1 \cdots c_{p-1}])\lambda(c_p)}{\lambda\left(V\left(c_1, \ldots, c_p\right)\right)}, \quad \mathbf{c} = c_1 c_2 \ldots c_\ell \in \mathcal{C} \setminus \{\varnothing\}.$$

Stationary measure: product form

Theorem

Under stability conditions, the stationary measures have the form

$$\pi(\mathbf{c}) = \pi(\varnothing) \prod_{p=1}^{\ell} \frac{\gamma_{c_p}([c_1 \cdots c_{p-1}])\lambda(c_p)}{\lambda\left(V\left(c_1, \ldots, c_p\right)\right)}, \quad \mathbf{c} = c_1 c_2 \ldots c_\ell \in \mathcal{C} \setminus \{\varnothing\}.$$

- Already known for unconstrained arrivals;
- Generalizes to discrete-time models and/or finite buffer system and/or memoryless reneging;
- **Key tool**: These are **Order-Independent queues**.

Consequence: explicit expression for the gradient

Corollary

The gradient

$$\nabla \log \pi_{\Gamma}(\mathbf{c}|\theta) = \nabla \log \Gamma([\mathbf{c}] \mid \theta) - \mathbb{E}\left(\nabla \log \Gamma([C] \mid \theta)\right), \quad \mathbf{c} \in \mathcal{C},$$

where C is distributed according to the stationary distribution $\pi_{\Gamma}(\cdot|\theta)$.

Consequence: explicit expression for the gradient

Corollary

The gradient

$$\nabla \log \pi_\Gamma(\mathbf{c}|\theta) = \nabla \log \Gamma([\mathbf{c}] \mid \theta) - \mathbb{E}\left(\nabla \log \Gamma([C] \mid \theta)\right), \quad \mathbf{c} \in \mathcal{C},$$

where C is distributed according to the stationary distribution $\pi_\Gamma(\cdot|\theta)$.

Ongoing work and perspectives

- Optimization of the access control for matching rewards?
- ... for holding costs?
- ... for loss costs in the case of reneging?

Merci