Stochastic Dynamic Matching

A Mixed Graph-Theory and Linear-Algebra Approach

Céline Comte
TU/e & CNRS

Fabien Mathieu
LINCS & Swapcard

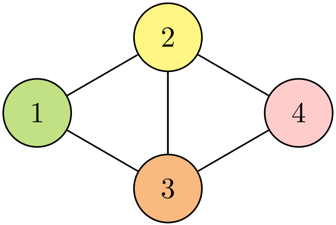
Ana Bušić
Inria & DI ENS

Informs APS – June 28, 2023

Matching problem

Graph G undirected, connected, without loop

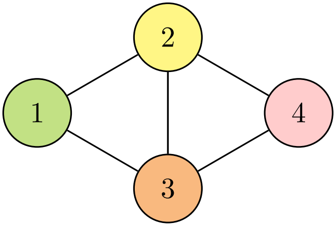
- Nodes $V = \{1, 2, \dots, n\}$ → items
- Edges $E = \{1, 2, \dots, m\}$ → possible matches



Matching problem

Graph G undirected, connected, without loop, non-bipartite

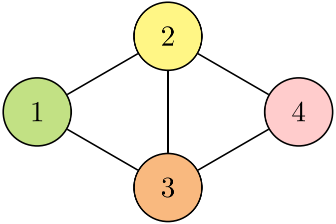
- Nodes $V = \{1, 2, \ldots, n\} \to$ items
- Edges $E = \{1, 2, \ldots, m\} \to$ possible matches



Online stochastic matching problem

Graph G undirected, connected, without loop, non-bipartite

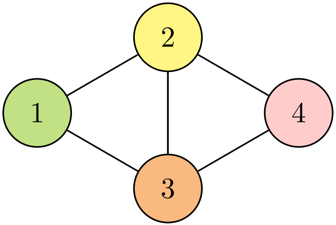
- Nodes $V = \{1, 2, \ldots, n\} \rightarrow$ item classes
- Edges $E = \{1, 2, \ldots, m\} \rightarrow$ possible matches



Online stochastic matching problem

Graph G undirected, connected, without loop, non-bipartite

- Nodes $V = \{1, 2, \ldots, n\} \to$ item classes
- Edges $E = \{1, 2, \ldots, m\} \to$ possible matches



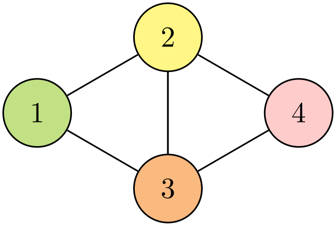
Class-i items:

- arrive according to a Poisson process with rate λ_i
- can be matched with items of neighbor classes

Online stochastic matching problem

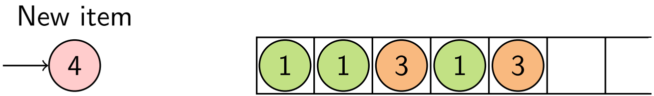
Graph G undirected, connected, without loop, non-bipartite

- Nodes $V = \{1, 2, \ldots, n\} \rightarrow$ item classes
- Edges $E = \{1, 2, \ldots, m\} \rightarrow$ possible matches



Class-i items:

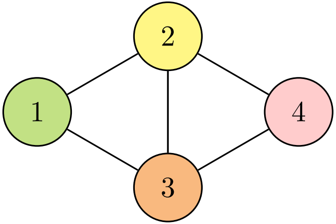
- arrive according to a Poisson process with rate λ_i
- can be matched with items of neighbor classes



Online stochastic matching problem

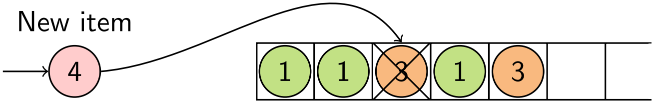
Graph G undirected, connected, without loop, non-bipartite

- Nodes $V = \{1, 2, \dots, n\} \rightarrow$ item classes
- Edges $E = \{1, 2, \dots, m\} \rightarrow$ possible matches



Class-i items:

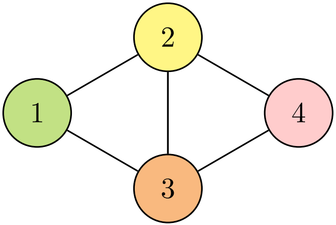
- arrive according to a Poisson process with rate λ_i
- can be matched with items of neighbor classes



Online stochastic matching problem

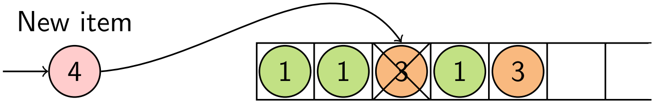
Graph G undirected, connected, without loop, non-bipartite

- Nodes $V = \{1, 2, \ldots, n\} \to$ item classes
- Edges $E = \{1, 2, \ldots, m\} \to$ possible matches



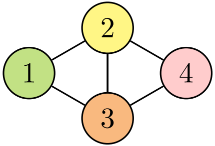
Class-i items:

- arrive according to a Poisson process with rate λ_i
- can be matched with items of neighbor classes

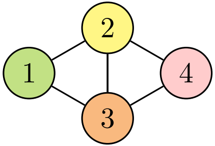


- If the policy is given, we obtain a Markov chain
- We assume stabilizability

Matching rate

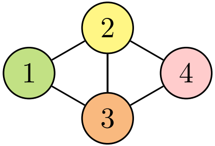


Matching rate



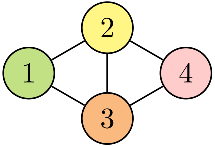
- **Matching rate** μ_k along edge $k = \{i, j\}$:
  mean number of matches per time unit between classes i and j.

Matching rate



- **Matching rate** μ_k along edge $k = \{i, j\}$:
  mean number of matches per time unit between classes i and j.
- Matching rates are particularly interesting:
  - They give intuition about the long-term impact of the matching policy.

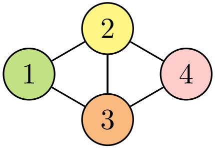
Matching rate



- **Matching rate** μ_k along edge $k = \{i, j\}$:
  mean number of matches per time unit between classes i and j.

- Matching rates are particularly interesting:
  - They give intuition about the long-term impact of the matching policy.
  - We often want to maximize a reward $\sum_{k \in E} r_k \mu_k$ that depends on these matching rates.

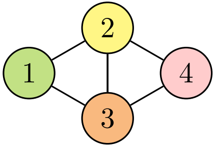
Matching rate



- **Matching rate** μ_k along edge $k = \{i, j\}$:
  mean number of matches per time unit between classes i and j.

- Matching rates are particularly interesting:
  - They give intuition about the long-term impact of the matching policy.
  - We often want to maximize a reward $\sum_{k \in E} r_k \mu_k$ that depends on these matching rates.
  - These matching rates may be the input of a “second-level” system.

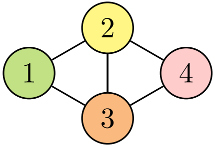
Matching rate



- **Matching rate μ_k** along edge $k = \{i, j\}$:
  mean number of matches per time unit between classes i and j.
- Matching rates are particularly interesting:
  - They give intuition about the long-term impact of the matching policy.
  - We often want to maximize a reward $\sum_{k \in E} r_k \mu_k$ that depends on these matching rates.
  - These matching rates may be the input of a "second-level" system.

Given a graph $G = (V, E)$ and an arrival-rate vector $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$, what is the set of "achievable" matching-rate vectors $\mu = (\mu_1, \mu_2, \ldots, \mu_m)$?

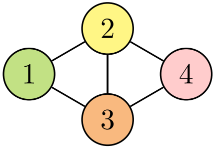
Conservation law



The matching rates satisfy the **conservation law**

$$\lambda_i = \sum_{k \in E_i} \mu_k, \quad i \in \{1, 2, \ldots, n\}.$$

Conservation law

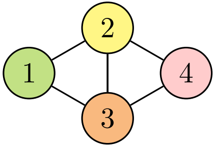


The matching rates satisfy the **conservation law**

$$\lambda_i = \sum_{k \in E_i} \mu_k, \quad i \in \{1, 2, \ldots, n\}.$$

$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

Conservation law



The matching rates satisfy the **conservation law**

$$\lambda_i = \sum_{k \in E_i} \mu_k, \quad i \in \{1, 2, \ldots, n\},$$

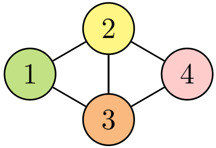
that is, in matrix form,

$$\lambda = A\mu,$$

where A is the **incidence matrix** of the graph G.

$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

Conservation law



The matching rates satisfy the **conservation law**

$$\lambda_i = \sum_{k \in E_i} \mu_k, \quad i \in \{1, 2, \ldots, n\},$$

$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

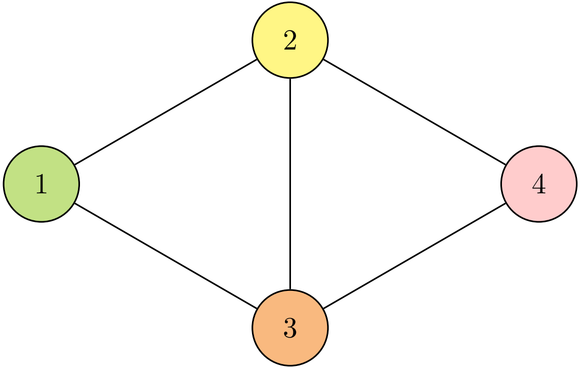
that is, in matrix form,

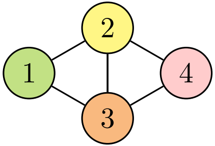
$$\lambda = A\mu,$$

where A is the **incidence matrix** of the graph G.

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

Example: Diamond graph

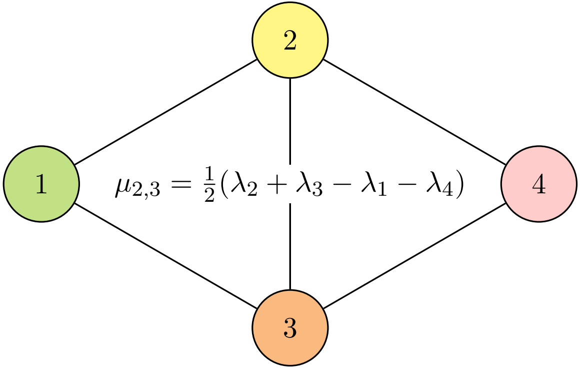


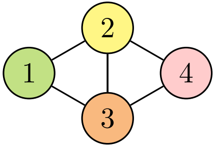


$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

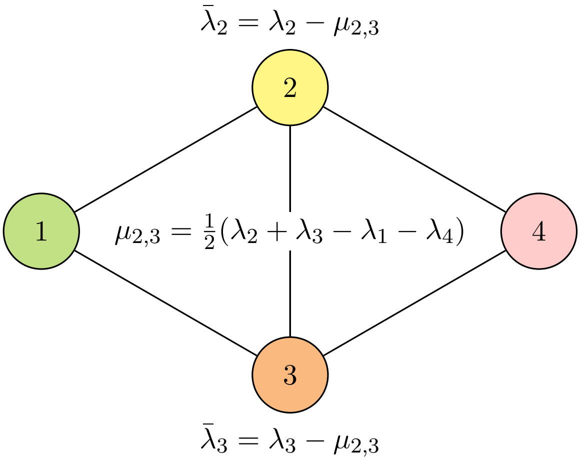
Example: Diamond graph

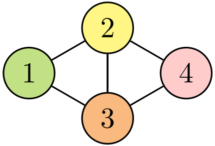




$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

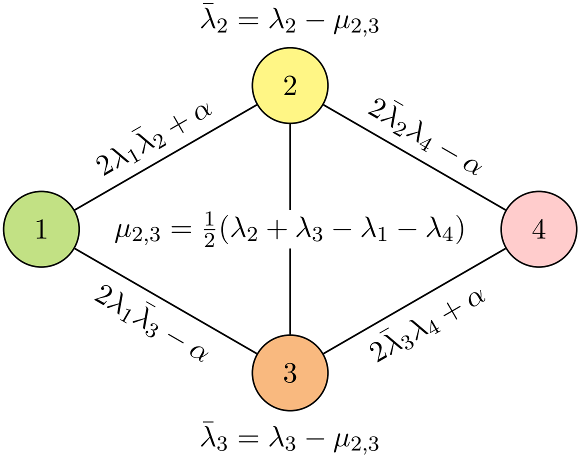


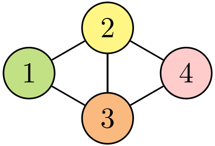


$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

Example: Diamond graph

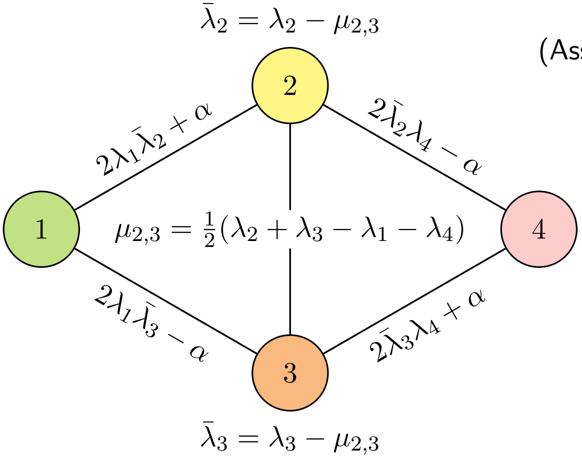




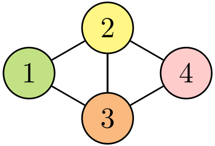
$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

Example: Diamond graph



(Assuming $\lambda_1 + \lambda_4 = \frac{1}{2}$)



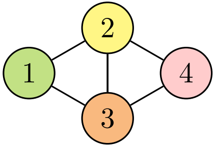
$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

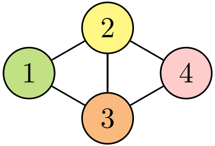
Overview

- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	
Non-negative solutions	
Achievable solutions	



Overview



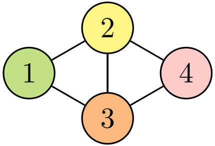
- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	Affine space
Non-negative solutions	
Achievable solutions	

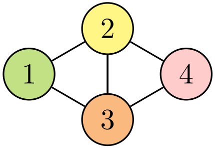
Overview

- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	Affine space
Non-negative solutions	Convex polytope
Achievable solutions	



Overview

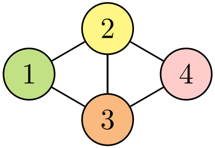


- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	Affine space of dimension d
Non-negative solutions	Convex polytope of dimension d
Achievable solutions	

with $d =$ (number of edges) $-$ (number of nodes)

Overview

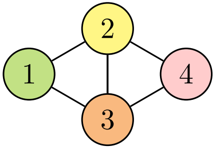


- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	Affine space of dimension d
Non-negative solutions	Convex polytope of dimension d
Achievable solutions	Convex set $\supseteq$ (interior of the polytope + some faces)

with $d =$ (number of edges) $-$ (number of nodes)

Overview



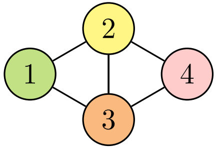
- **Solutions of the conservation law $\lambda = A\mu$**

All solutions	Affine space of dimension d
Non-negative solutions	Convex polytope of dimension d
Achievable solutions	Convex set $\supseteq$ (interior of the polytope + some faces)

  with $d =$ (number of edges) $-$ (number of nodes)

- **Contributions**
  - Characterization of surjective/injective/bijective graphs G.
  - Stabilizability conditions that are easier to verify.
  - Almost complete characterization of the set of achievable matching rate vectors.
  - Filtering (resp. semi-filtering) policies to achieve (resp. approach) vertices of the polytope.

Affine space of all solutions of $\lambda = A\mu$



We define

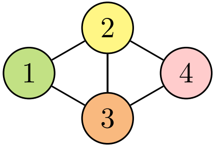
- μ° = a particular solution, built using the pseudo-inverse of A
- d = (number of edges) − (number of nodes)
- $\{b_1, b_2, \ldots, b_d\}$ = basis of kernel(A), built using a spanning tree of G (Doob, 1973)

Proposition

The solution set Π *of the equation* $\lambda = A\mu$ *is the* **d-dimensional affine space**

$$\Pi = \left\{ \mu^\circ + \alpha_1 b_1 + \alpha_2 b_2 + \ldots + \alpha_d b_d : (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{R}^d \right\}.$$

Affine space of all solutions of $\lambda = A\mu$



We define

- μ° = a particular solution, built using the pseudo-inverse of A
- d = (number of edges) − (number of nodes)
- $\{b_1, b_2, \ldots, b_d\}$ = basis of kernel(A), built using a spanning tree of G (Doob, 1973)

Proposition

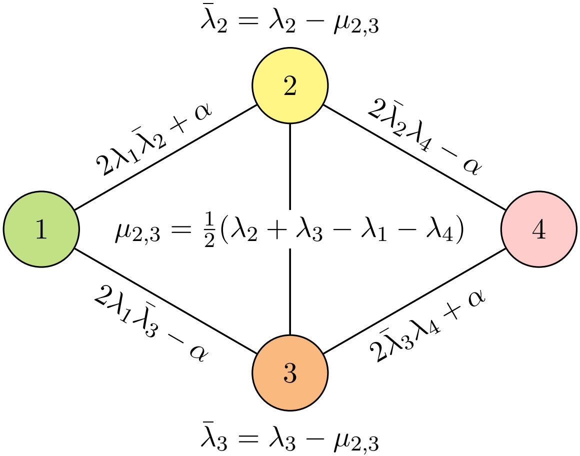
The solution set Π of the equation $\lambda = A\mu$ is the **d-dimensional affine space**

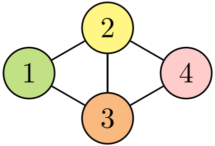
$$\Pi = \left\{ \mu^\circ + \alpha_1 b_1 + \alpha_2 b_2 + \ldots + \alpha_d b_d : (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{R}^d \right\}.$$

Two coordinate systems:

- **Edge coordinates:** $\mu = (\mu_1, \mu_2, \ldots, \mu_m) \in \mathbb{R}^m$
- **Kernel coordinates:** $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{R}^d$

Example: Diamond graph

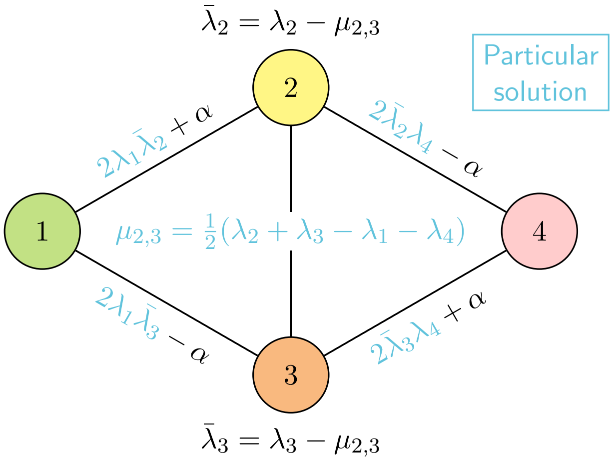


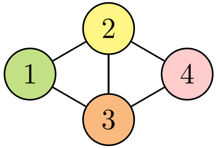


$$
\begin{cases}
\lambda_1 = \mu_{1,2} + \mu_{1,3} \\
\lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\
\lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\
\lambda_4 = \mu_{2,4} + \mu_{3,4}
\end{cases}
$$

$$
\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix}
=
\begin{bmatrix}
1 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1
\end{bmatrix}
\begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}
$$

Example: Diamond graph

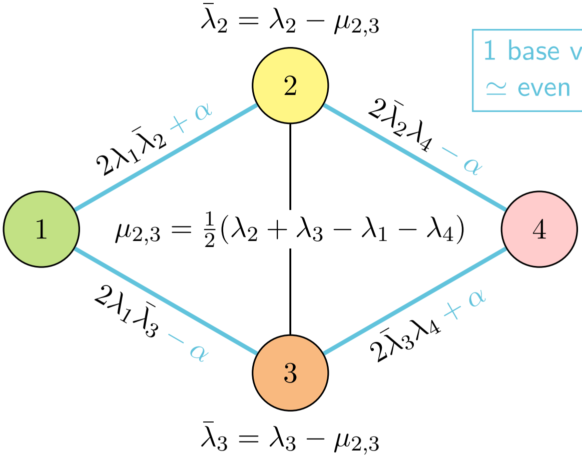




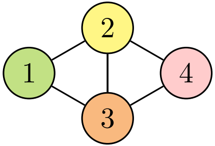
$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

Example: Diamond graph



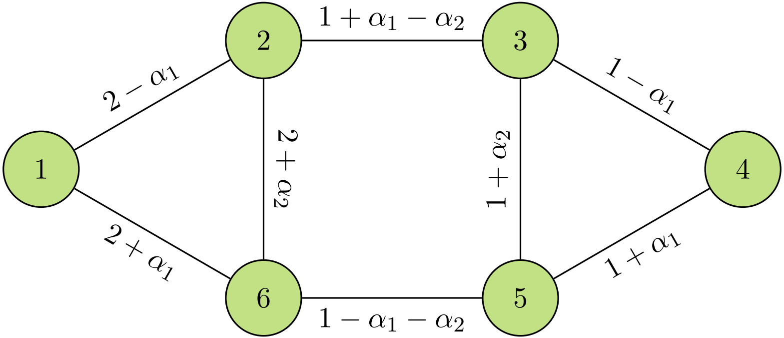
1 base vector
$\simeq$ even cycle



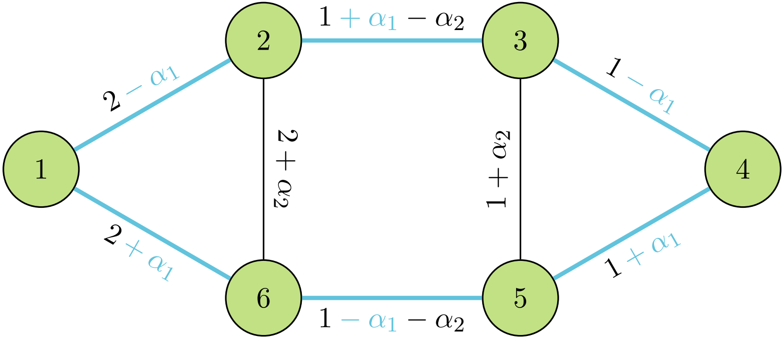
$$\begin{cases} \lambda_1 = \mu_{1,2} + \mu_{1,3} \\ \lambda_2 = \mu_{1,2} + \mu_{2,3} + \mu_{2,4} \\ \lambda_3 = \mu_{1,3} + \mu_{2,3} + \mu_{3,4} \\ \lambda_4 = \mu_{2,4} + \mu_{3,4} \end{cases}$$

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mu_{1,2} \\ \mu_{1,3} \\ \mu_{2,3} \\ \mu_{2,4} \\ \mu_{3,4} \end{bmatrix}$$

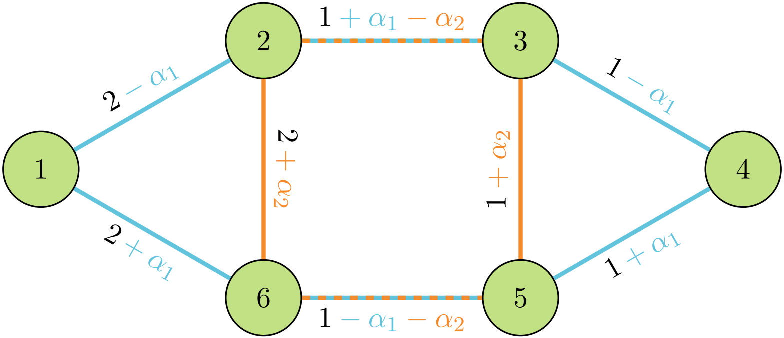
Example: Codomino graph



Example: Codomino graph



Example: Codomino graph



Convex polytope of non-negative solutions of $\lambda = A\mu$

Corollary

The set $\Pi_{\geq 0}$ *of non-negative solutions of* $\lambda = A\mu$ *is the* **d-dimensional convex polytope**

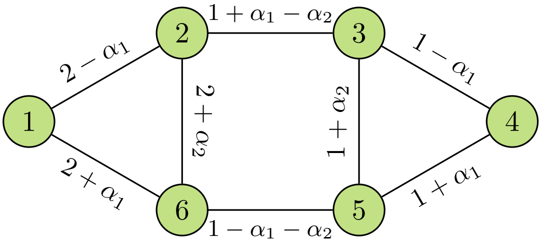
$$\Pi_{\geq 0} = \Pi \cap \mathbb{R}^d_{\geq 0} \simeq \left\{ \alpha \in \mathbb{R}^d : \mu^\circ + \alpha_1 b_1 + \ldots + \alpha_d b_d \geq 0 \right\}.$$

Convex polytope of non-negative solutions of $\lambda = A\mu$

Corollary

The set $\Pi_{\geq 0}$ of non-negative solutions of $\lambda = A\mu$ is the **d-dimensional convex polytope**

$$\Pi_{\geq 0} = \Pi \cap \mathbb{R}^d_{\geq 0} \simeq \left\{\alpha \in \mathbb{R}^d : \mu^\circ + \alpha_1 b_1 + \ldots + \alpha_d b_d \geq 0\right\}.$$



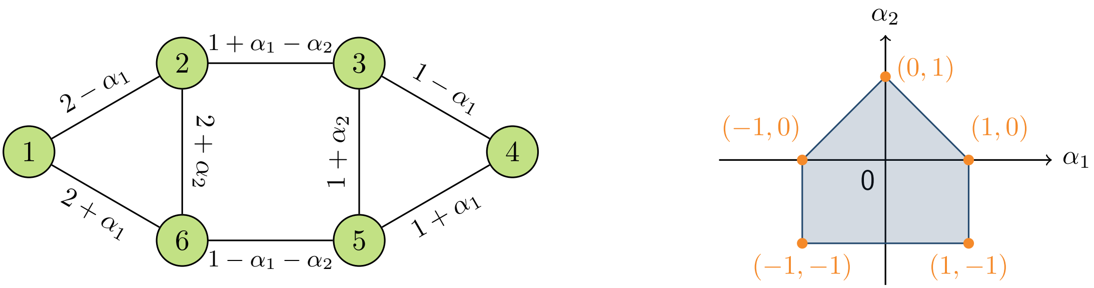
(a) General solution of $\lambda = A\mu$

Convex polytope of non-negative solutions of $\lambda = A\mu$

Corollary

The set $\Pi_{\geq 0}$ *of non-negative solutions of* $\lambda = A\mu$ *is the* **d-dimensional convex polytope**

$$\Pi_{\geq 0} = \Pi \cap \mathbb{R}^d_{\geq 0} \simeq \left\{ \alpha \in \mathbb{R}^d : \mu^\circ + \alpha_1 b_1 + \ldots + \alpha_d b_d \geq 0 \right\}.$$



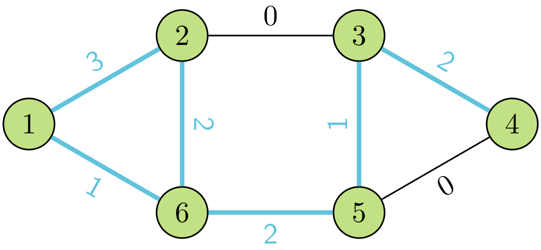
(a) General solution of $\lambda = A\mu$ (b) Convex polytope of non-negative solutions

Convex polytope of non-negative solutions of $\lambda = A\mu$

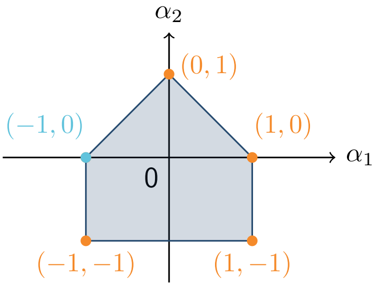
Corollary

The set $\Pi_{\geq 0}$ *of non-negative solutions of* $\lambda = A\mu$ *is the* **d-dimensional convex polytope**

$$\Pi_{\geq 0} = \Pi \cap \mathbb{R}^d_{\geq 0} \simeq \left\{ \alpha \in \mathbb{R}^d : \mu^\circ + \alpha_1 b_1 + \ldots + \alpha_d b_d \geq 0 \right\}.$$



(a) General solution of $\lambda = A\mu$



(b) Convex polytope of non-negative solutions

Achievable solutions of $\lambda = A\mu$

Theorem

Let μ be a vertex of $\Pi_{\geq 0}$ and $p =$ number of (strictly) positive coordinates of μ.

Achievable solutions of $\lambda = A\mu$

Theorem

Let μ be a vertex of $\Pi_{\geq 0}$ and $p =$ number of (strictly) positive coordinates of μ.

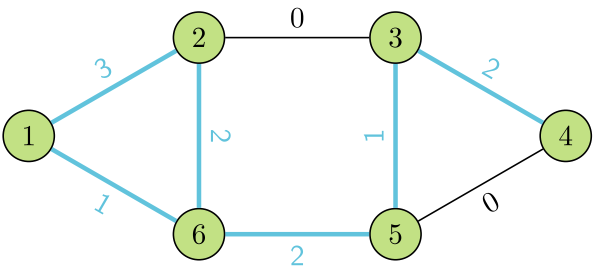
- *If $p =$ number of nodes in G, then μ can be achieved by a stable policy: for instance, a "filtering" match-the-longest policy applied on the support of μ.*

Achievable solutions of $\lambda = A\mu$

Theorem

Let μ be a vertex of $\Pi_{\geq 0}$ and $p =$ number of (strictly) positive coordinates of μ.

- *If $p =$ number of nodes in G, then μ can be achieved by a stable policy: for instance, a "filtering" match-the-longest policy applied on the support of μ.*

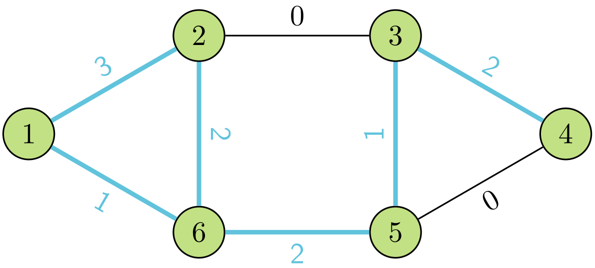


Achievable solutions of $\lambda = A\mu$

Theorem

Let μ be a vertex of $\Pi_{\geq 0}$ and p = number of (strictly) positive coordinates of μ.

- *If p = number of nodes in G, then μ can be achieved by a stable policy: for instance, a "filtering" match-the-longest policy applied on the support of μ.*
- *If $p <$ number of nodes in G, then μ cannot be achieved by a stable policy.*



Non-achievable solutions of $\lambda = A\mu$

Conjecture

Let μ *be a vertex of* $\Pi_{\geq 0}$.

Non-achievable solutions of $\lambda = A\mu$

Conjecture

Let μ be a vertex of $\Pi_{\geq 0}$. For each $k \in \mathbb{N}$, let Φ_k denote the following "semi-filtering" policy:

- *If longest queue size $< k$, apply the filtering match-the-longest policy on the support of μ.*
- *Otherwise, apply the (non-filtering) match-the-longest policy.*

Non-achievable solutions of $\lambda = A\mu$

Conjecture

Let μ be a vertex of $\Pi_{\geq 0}$. For each $k \in \mathbb{N}$, let Φ_k denote the following "semi-filtering" policy:

- *If longest queue size $< k$, apply the filtering match-the-longest policy on the support of μ.*
- *Otherwise, apply the (non-filtering) match-the-longest policy.*

Φ_k is stable for each $k \in \mathbb{N}$, and $\lim_{k \to +\infty} \mu(\Phi_k) = \mu$.

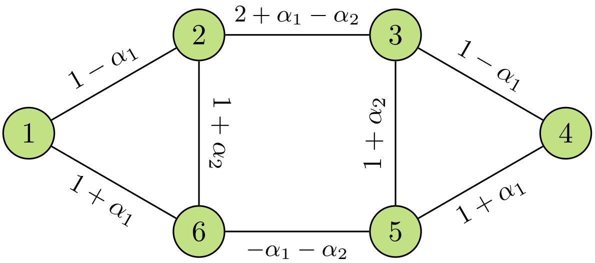
Non-achievable solutions of $\lambda = A\mu$

Conjecture

Let μ be a vertex of $\Pi_{\geq 0}$. For each $k \in \mathbb{N}$, let Φ_k denote the following "semi-filtering" policy:

- *If longest queue size $< k$, apply the filtering match-the-longest policy on the support of μ.*
- *Otherwise, apply the (non-filtering) match-the-longest policy.*

Φ_k is stable for each $k \in \mathbb{N}$, and $\lim_{k \to +\infty} \mu(\Phi_k) = \mu$.



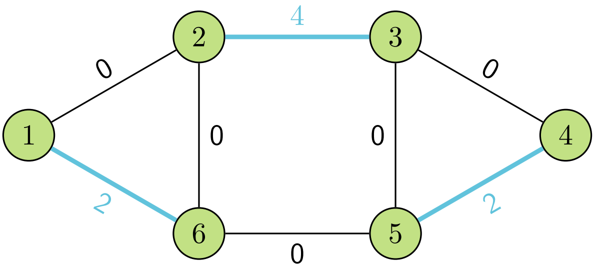
Non-achievable solutions of $\lambda = A\mu$

Conjecture

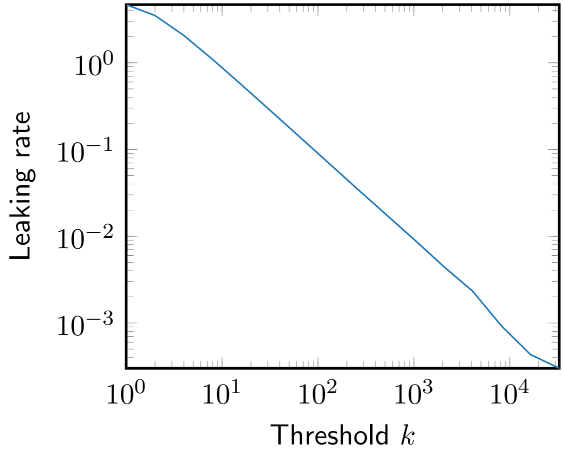
Let μ be a vertex of $\Pi_{\geq 0}$. For each $k \in \mathbb{N}$, let Φ_k denote the following "semi-filtering" policy:

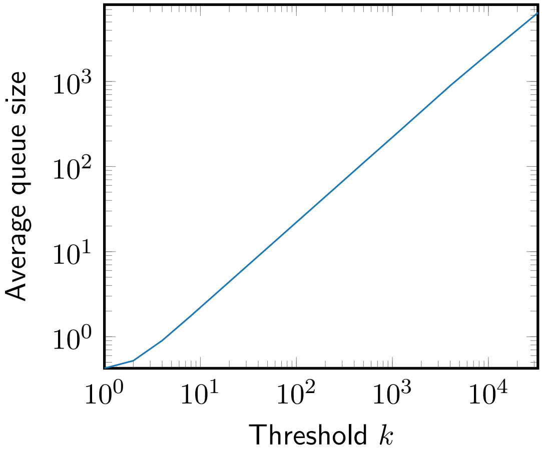
- *If longest queue size $< k$, apply the filtering match-the-longest policy on the support of μ.*
- *Otherwise, apply the (non-filtering) match-the-longest policy.*

Φ_k is stable for each $k \in \mathbb{N}$, and $\lim_{k \to +\infty} \mu(\Phi_k) = \mu$.



Numerical results: Performance of semi-filtering policies





Conclusion

Contributions

- Characterization of surjective/injective/bijective graphs.
- Stabilizability conditions that are easier to verify.
- Almost complete characterization of the set of achievable matching rate vectors.
- Filtering (resp. semi-filtering) policies to achieve (resp. approach) vertices of the polytope.

Conclusion

Contributions

- Characterization of surjective/injective/bijective graphs.
- Stabilizability conditions that are easier to verify.
- Almost complete characterization of the set of achievable matching rate vectors.
- Filtering (resp. semi-filtering) policies to achieve (resp. approach) vertices of the polytope.

Future works

- More realistic model: hypergraph? reneging?
- What it the arrival rates and/or the graph structure are unknown?