



# Appariements, polytopes et dons d'organes

Céline Comte — [c.m.comte@tue.nl](mailto:c.m.comte@tue.nl)

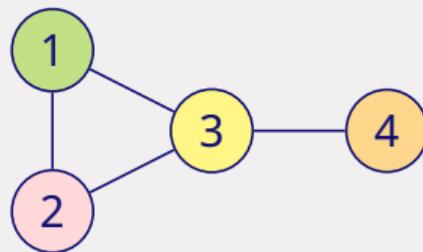
Université Technologique d'Eindhoven

Travail réalisé avec Fabien Mathieu (SwapCard) et Ana Bušić (Inria – ENS)

# Problème d'appariement

**Grphe**  $G$  non-orienté, connexe, sans boucle

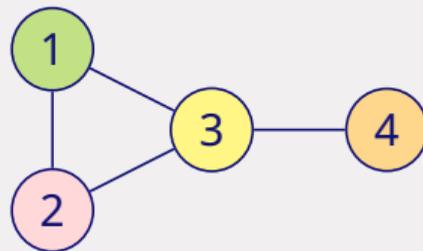
- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



# Problème d'appariement **dynamique et aléatoire**

**Grphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



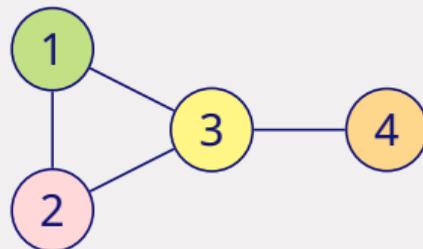
# Problème d'appariement **dynamique et aléatoire**

**Graphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés

Les éléments de la **classe**  $i$

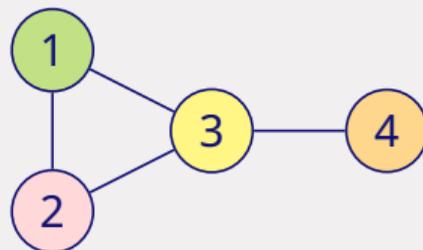
- arrivent selon un processus de Poisson d'intensité  $\mu_i$
- sont compatibles avec les éléments des classes voisines



# Problème d'appariement **dynamique et aléatoire**

**Graphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



Les éléments de la **classe i**

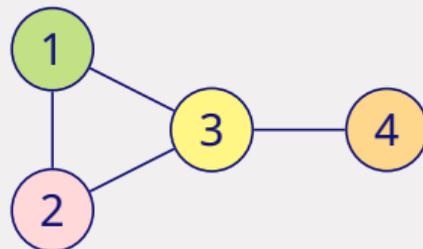
- arrivent selon un processus de Poisson d'intensité  $\mu_i$
- sont compatibles avec les éléments des classes voisines



# Problème d'appariement **dynamique et aléatoire**

**Graphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



Les éléments de la **classe i**

- arrivent selon un processus de Poisson d'intensité  $\mu_i$
- sont compatibles avec les éléments des classes voisines

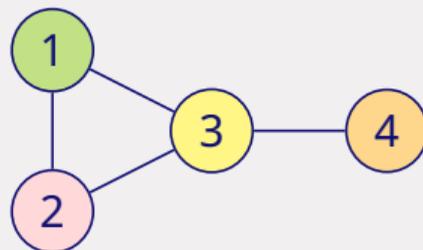
Arrivée



# Problème d'appariement **dynamique et aléatoire**

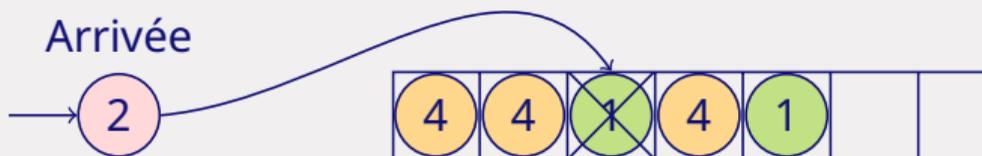
**Graphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



Les éléments de la **classe i**

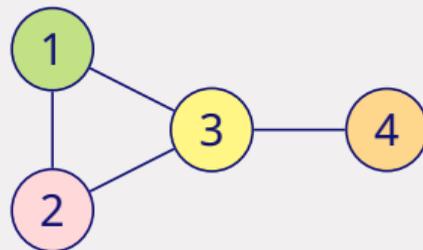
- arrivent selon un processus de Poisson d'intensité  $\mu_i$
- sont compatibles avec les éléments des classes voisines



# Problème d'appariement **dynamique et aléatoire**

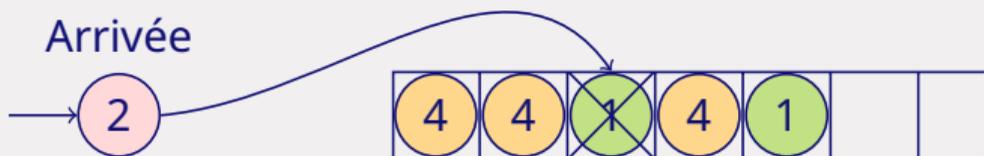
**Graphe**  $G$  non-orienté, connexe, sans boucle

- Sommets  $V = \{1, 2, \dots, n\} \rightarrow$  **classes** d'éléments
- Arêtes  $E = \{1, 2, \dots, m\} \rightarrow$  appariements autorisés



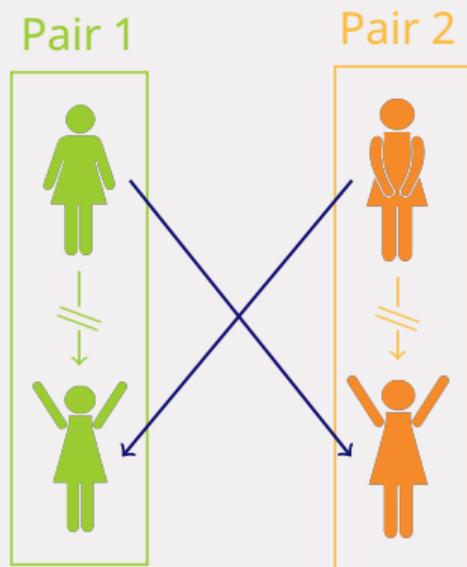
Les éléments de la **classe i**

- arrivent selon un processus de Poisson d'intensité  $\mu_i$
- sont compatibles avec les éléments des classes voisines

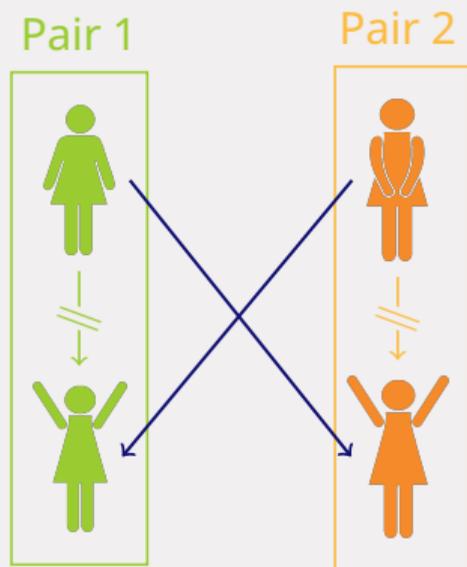


Pour une politique donnée  
 $\rightarrow$  Processus de Markov

## Application : Don croisé de reins

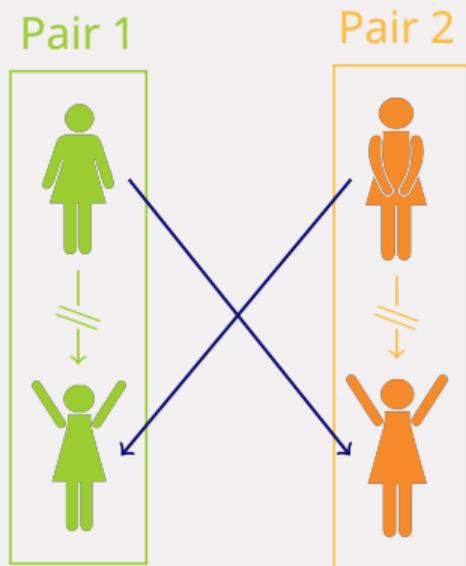


## Application : Don croisé de reins



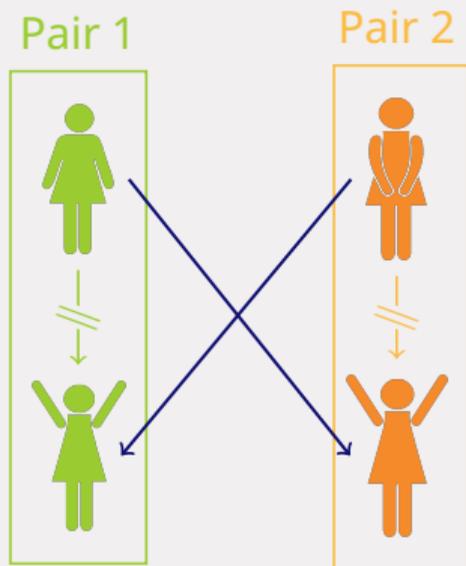
- **Élément** → Couple donneur-receveur

## Application : Don croisé de reins



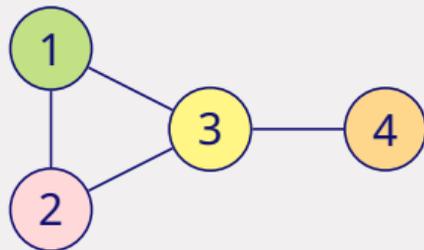
- **Élément** → Couple donneur-receveur
- **Contraintes strictes** → Groupe sanguin

## Application : Don croisé de reins



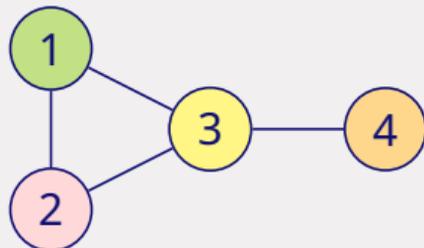
- **Élément** → Couple donneur-receveur
  - **Contraintes strictes** → Groupe sanguin
  - **Contraintes souples**  
→ Antigènes de leucocytes humains (HLA)
- Point bio :** Notre corps utilise les protéines définies par le HLA pour reconnaître ses cellules.

# Taux d'appariement



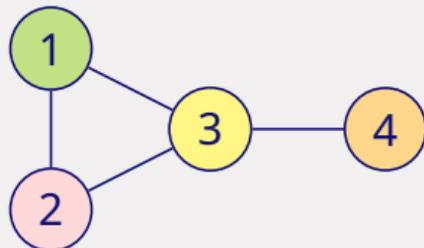
- **Taux d'appariement**  $\lambda_k$  le long de l'arête  $k = \{i, j\}$ :  
nombre moyen d'appariements par unité de temps entre les classes  $i$  et  $j$ .

# Taux d'appariement



- **Taux d'appariement**  $\lambda_k$  le long de l'arête  $k = \{i, j\}$ :  
nombre moyen d'appariements par unité de temps entre les classes  $i$  et  $j$ .
- Nous nous intéressons aux taux d'appariement car :
  - On veut souvent optimiser une fonction de coût qui dépend de ces taux.
  - Ils donnent une intuition sur l'impact long-terme de la politique d'appariement.

# Taux d'appariement



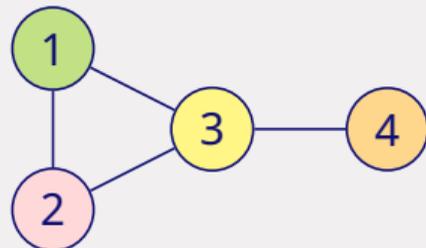
- **Taux d'appariement**  $\lambda_k$  le long de l'arête  $k = \{i, j\}$ :  
nombre moyen d'appariements par unité de temps entre les classes  $i$  et  $j$ .
- Nous nous intéressons aux taux d'appariement car :
  - On veut souvent optimiser une fonction de coût qui dépend de ces taux.
  - Ils donnent une intuition sur l'impact long-terme de la politique d'appariement.

Étant donné un graphe  $G = (V, E)$  et un vecteur  $\mu = (\mu_1, \mu_2, \dots, \mu_n)$ ,  
quel est l'ensemble des vecteurs  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$  atteignables ?

## Loi de conservation

Les taux d'appariement satisfont  
la **loi de conservation**

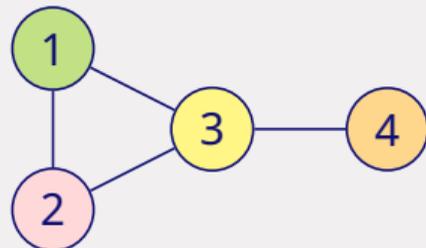
$$\sum_{k \in E_i} \lambda_k = \mu_i, \quad i \in \{1, 2, \dots, n\}.$$



## Loi de conservation

Les taux d'appariement satisfont  
la **loi de conservation**

$$\sum_{k \in E_i} \lambda_k = \mu_i, \quad i \in \{1, 2, \dots, n\}.$$



$$\left\{ \begin{array}{l} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{array} \right.$$

## Loi de conservation

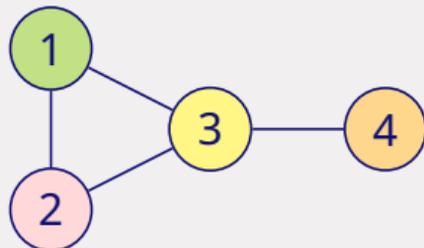
Les taux d'appariement satisfont la **loi de conservation**

$$\sum_{k \in E_i} \lambda_k = \mu_i, \quad i \in \{1, 2, \dots, n\},$$

c'est-à-dire, sous forme matricielle,

$$A\lambda = \mu,$$

où  $A$  est la matrice d'incidence du graphe de compatibilité  $G$ .



$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{cases}$$

## Loi de conservation

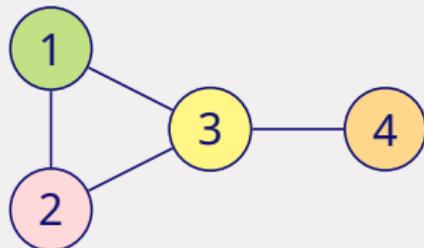
Les taux d'appariement satisfont la **loi de conservation**

$$\sum_{k \in E_i} \lambda_k = \mu_i, \quad i \in \{1, 2, \dots, n\},$$

c'est-à-dire, sous forme matricielle,

$$A\lambda = \mu,$$

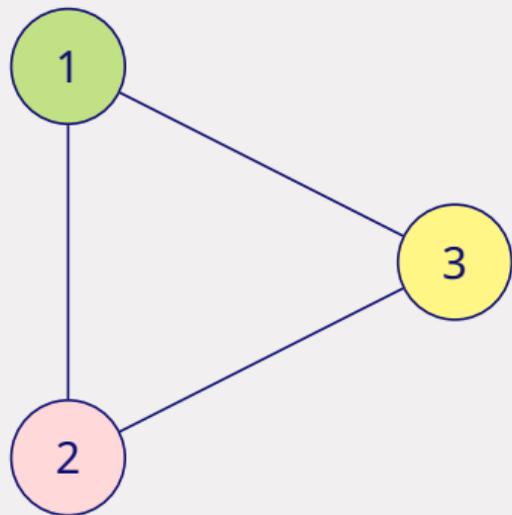
où  $A$  est la matrice d'incidence du graphe de compatibilité  $G$ .



$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

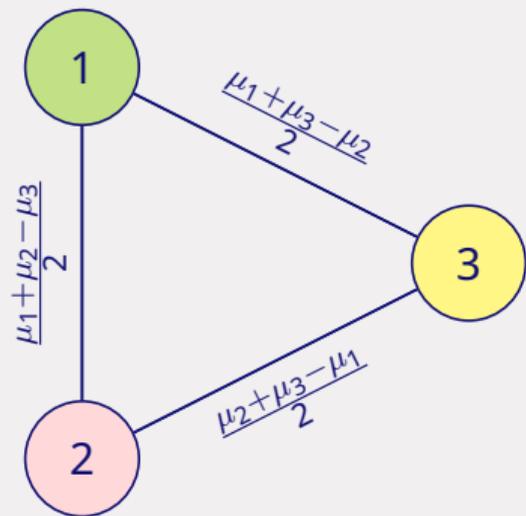
## Exemple : Graphe triangle



$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} = \mu_3 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}$$

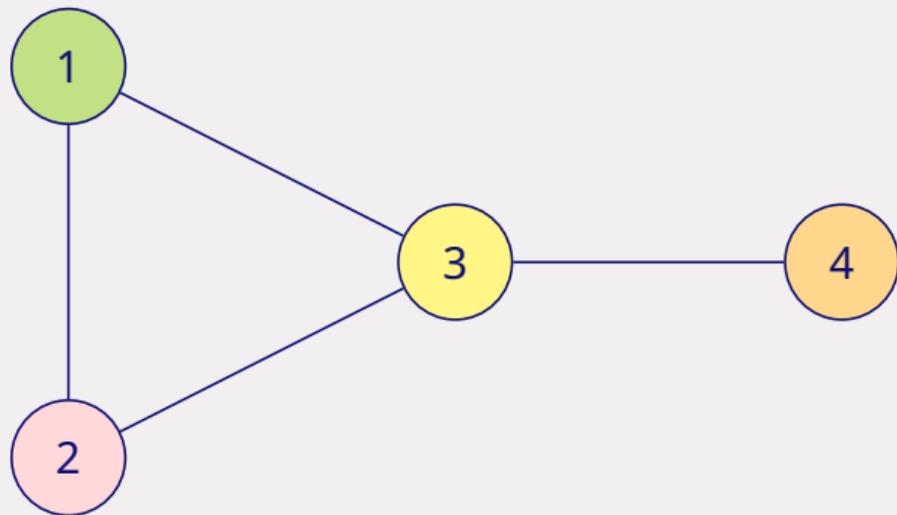
## Exemple : Graphe triangle



$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} = \mu_3 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}$$

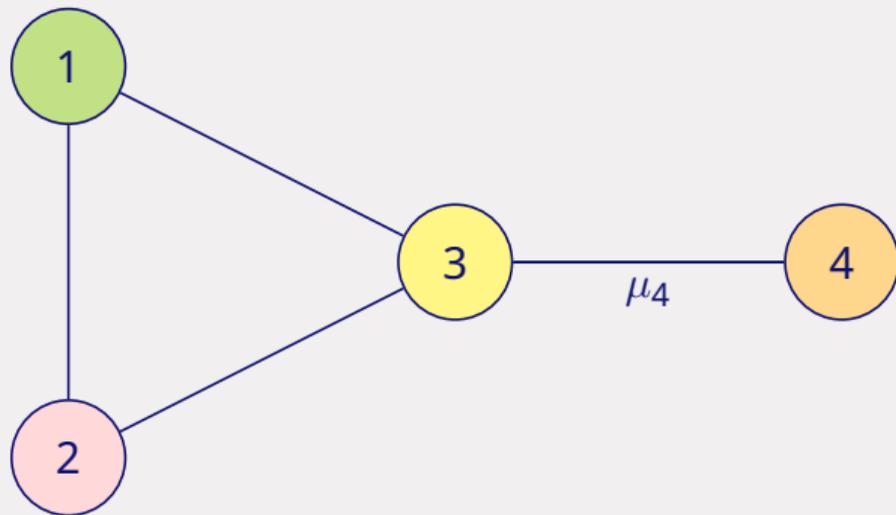
## Exemple : Graphe patte



$$\left\{ \begin{array}{l} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{array} \right.$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

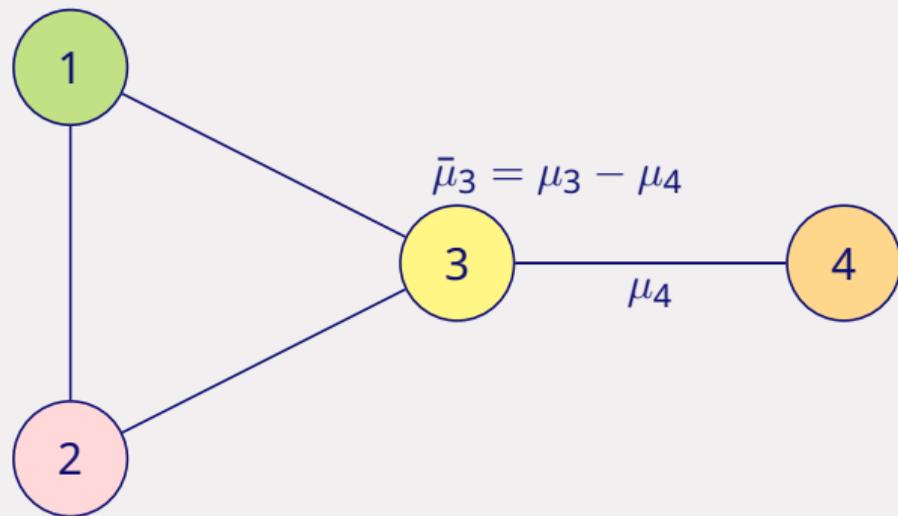
## Exemple : Graphe patte



$$\left\{ \begin{array}{l} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{array} \right.$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

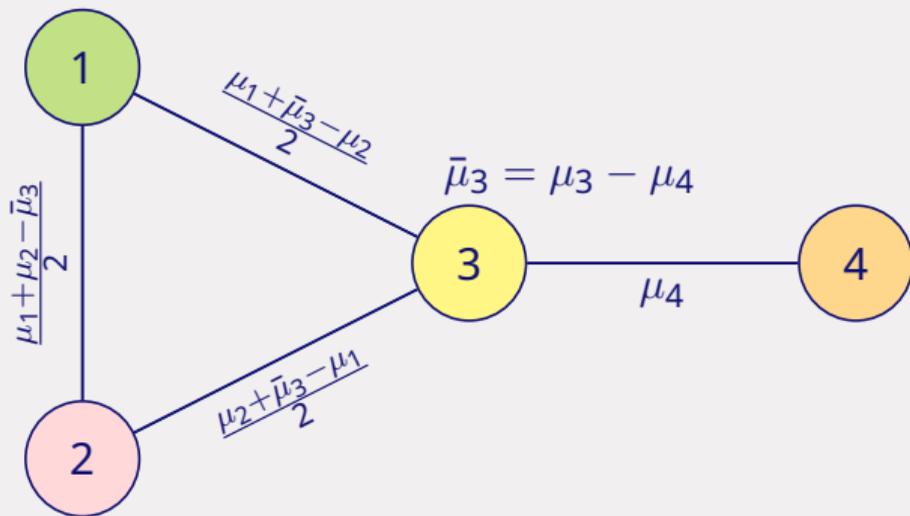
## Exemple : Graphe patte



$$\left\{ \begin{array}{l} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{array} \right.$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

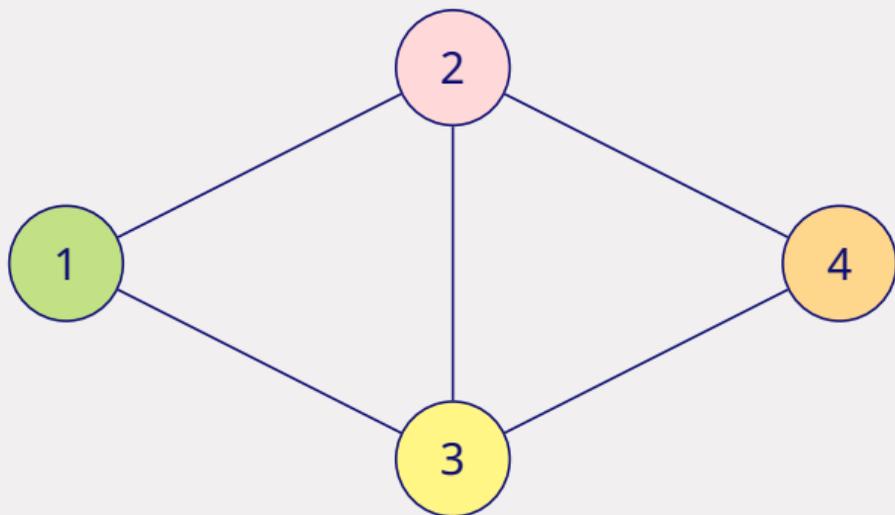
## Exemple : Graphe patte



$$\left\{ \begin{array}{l} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{3,4} = \mu_4 \end{array} \right.$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

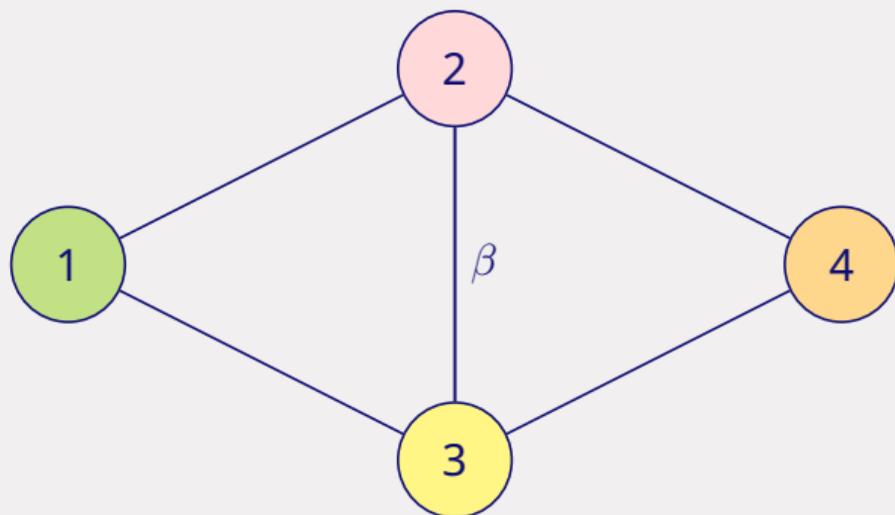
## Exemple : Graphe diamant



$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} + \lambda_{2,4} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{2,4} + \lambda_{3,4} = \mu_4 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{2,4} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

## Exemple : Graphe diamant

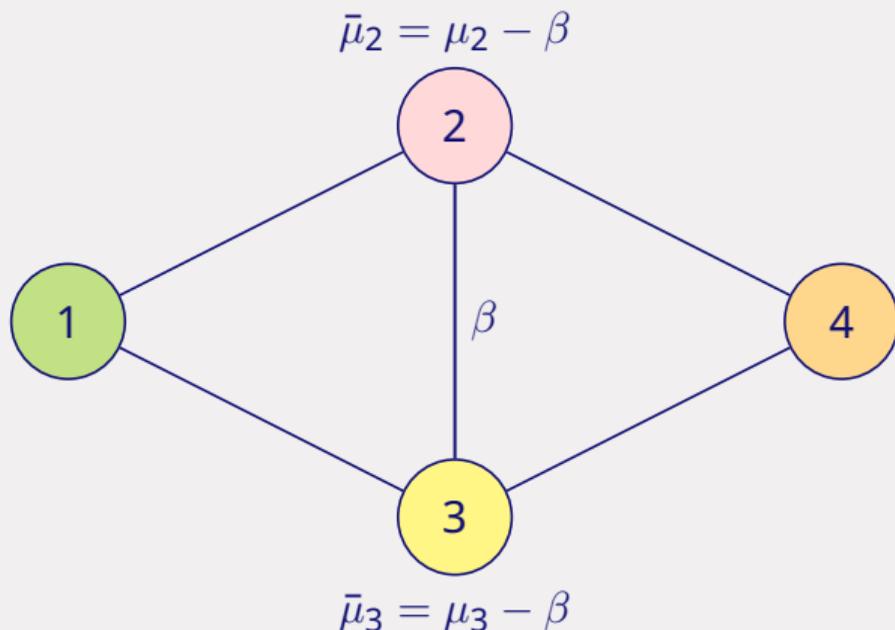


$$\beta = \frac{1}{2}(\mu_2 + \mu_3 - \mu_1 - \mu_4)$$

$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} + \lambda_{2,4} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{2,4} + \lambda_{3,4} = \mu_4 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{2,4} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

## Exemple : Graphe diamant

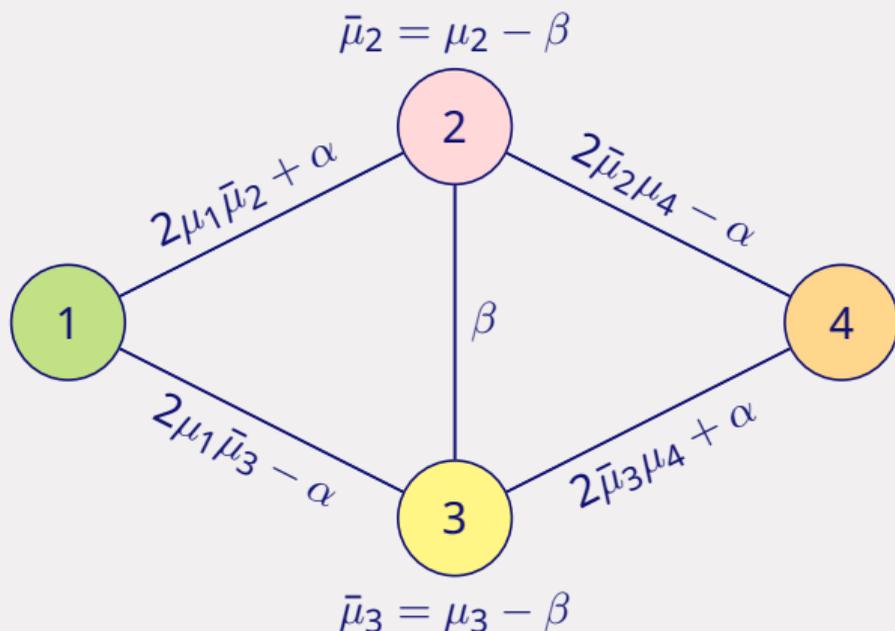


$$\beta = \frac{1}{2}(\mu_2 + \mu_3 - \mu_1 - \mu_4)$$
$$\mu_1 + \mu_4 = \bar{\mu}_2 + \bar{\mu}_3 = \frac{1}{2}$$

$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} + \lambda_{2,4} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{2,4} + \lambda_{3,4} = \mu_4 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{2,4} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

## Exemple : Graphe diamant



$$\beta = \frac{1}{2}(\mu_2 + \mu_3 - \mu_1 - \mu_4)$$

$$\mu_1 + \mu_4 = \bar{\mu}_2 + \bar{\mu}_3 = \frac{1}{2}$$

$$\begin{cases} \lambda_{1,2} + \lambda_{1,3} = \mu_1 \\ \lambda_{1,2} + \lambda_{2,3} + \lambda_{2,4} = \mu_2 \\ \lambda_{1,3} + \lambda_{2,3} + \lambda_{3,4} = \mu_3 \\ \lambda_{2,4} + \lambda_{3,4} = \mu_4 \end{cases}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{2,4} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

# Injectivité, surjectivité, bijectivité

## Injectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est injective
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a au plus une solution
- Le graphe  $G$  contient au plus un cycle, et ce cycle est de longueur impaire

# Injectivité, surjectivité, bijectivité

## Injectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **injective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au plus** une solution
- Le graphe  $G$  contient **au plus un cycle**, et ce cycle est de longueur impaire

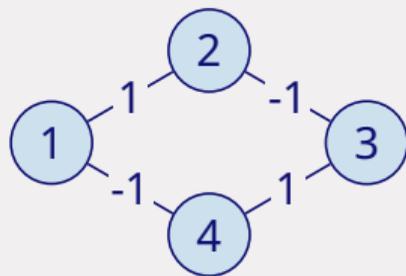
Intuition : Comment construire un vecteur appartenant au noyau de  $A$  ?

# Injectivité, surjectivité, bijectivité

## Injectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **injective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au plus** une solution
- Le graphe  $G$  contient **au plus un cycle**, et ce cycle est de longueur impaire

Intuition : Comment construire un vecteur appartenant au noyau de  $A$  ?

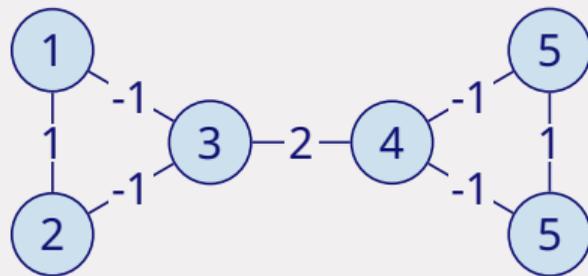
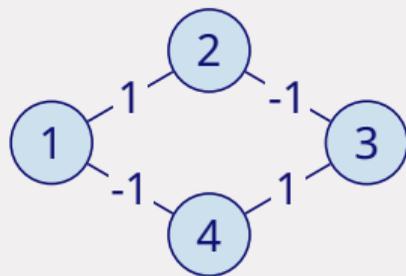


# Injectivité, surjectivité, bijectivité

## Injectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **injective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au plus** une solution
- Le graphe  $G$  contient **au plus un cycle**, et ce cycle est de longueur impaire

Intuition : Comment construire un vecteur appartenant au noyau de  $A$  ?

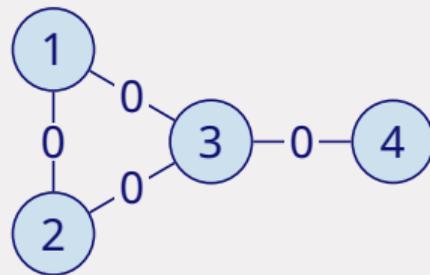
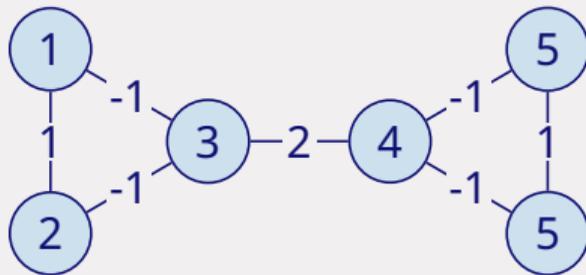
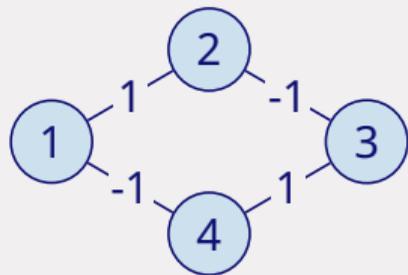


# Injectivité, surjectivité, bijectivité

## Injectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **injective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au plus** une solution
- Le graphe  $G$  contient **au plus un cycle**, et ce cycle est de longueur **impaire**

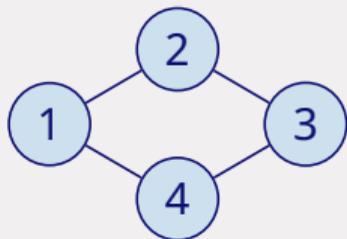
Intuition : Comment construire un vecteur appartenant au noyau de  $A$  ?



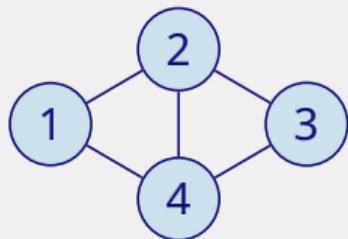
# Injectivité, surjectivité, bijectivité

## Injectivité

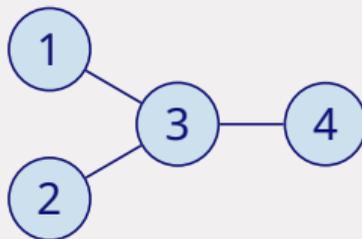
- L'application linéaire  $\lambda \mapsto A\lambda$  est **injective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au plus** une solution
- Le graphe  $G$  contient **au plus un cycle**, et ce cycle est de longueur **impaire**



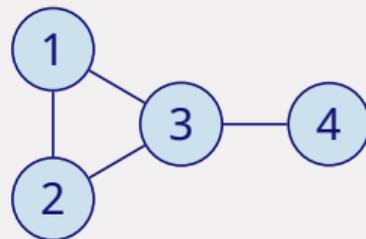
(a) Non



(b) Non



(c) Oui



(d) Oui

# Injectivité, surjectivité, bijectivité

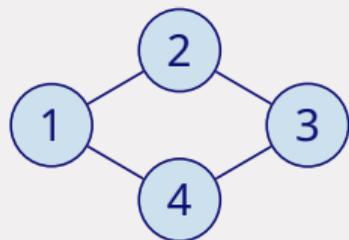
## Surjectivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **surjective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au moins** une solution
- Le graphe  $G$  contient **au moins un cycle de longueur impaire** (i.e., non-biparti)

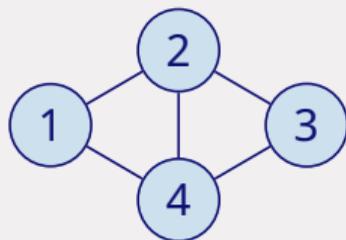
# Injectivité, surjectivité, bijectivité

## Surjectivité

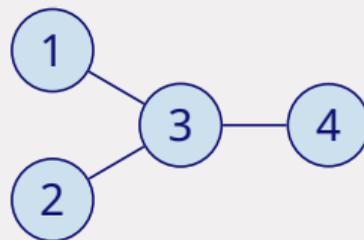
- L'application linéaire  $\lambda \mapsto A\lambda$  est **surjective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **au moins** une solution
- Le graphe  $G$  contient **au moins un cycle de longueur impaire** (i.e., non-biparti)



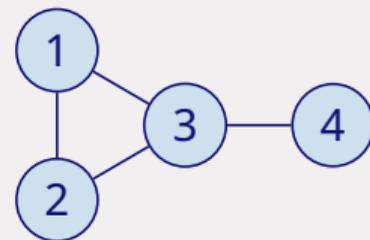
(a) Non



(b) Oui



(c) Non



(d) Oui

# Injectivité, surjectivité, bijectivité

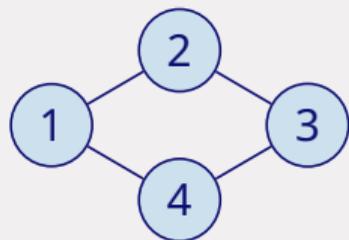
## Bijektivité

- L'application linéaire  $\lambda \mapsto A\lambda$  est **bijective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **exactement** une solution
- Le graphe  $G$  contient **exactement un cycle**, et ce cycle est de longueur impaire

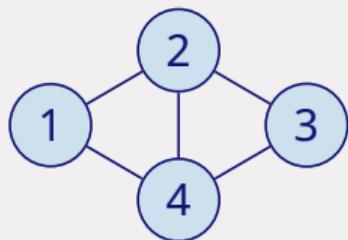
# Injectivité, surjectivité, bijectivité

## Bijektivité

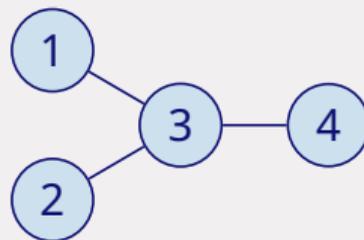
- L'application linéaire  $\lambda \mapsto A\lambda$  est **bijective**
- Pour chaque  $\mu \in \mathbb{R}^n$ , l'équation  $A\lambda = \mu$  a **exactement** une solution
- Le graphe  $G$  contient **exactement un cycle**, et ce cycle est de longueur impaire



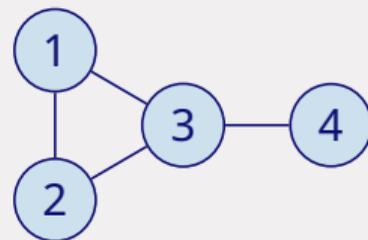
(a) Rien



(b) Exclusivement  
surjectif

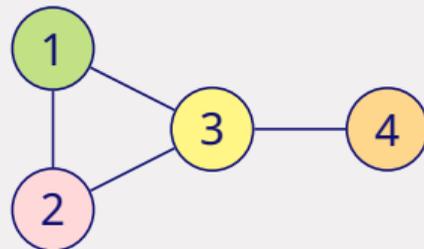


(c) Exclusivement  
injectif



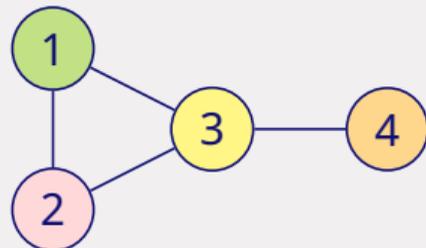
(d) Bijectif

# Stabilisabilité



$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

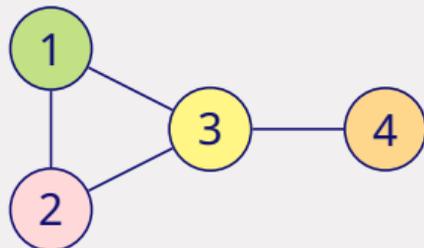
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

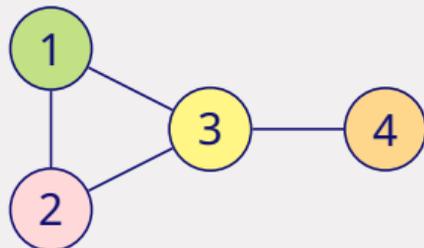
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si **la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$**   
😊 Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

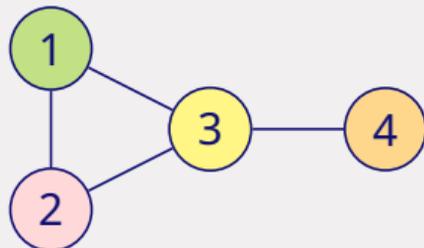
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si **la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$** 
  - 😊 Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$
  - 😊 En général, la complexité est polynomiale en  $n$  et  $m$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

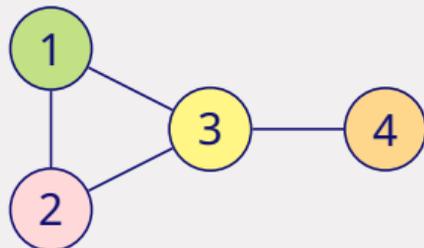
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si **la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$** 
  - 😊 Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$
  - 😊 En général, la complexité est polynomiale en  $n$  et  $m$
  - 😊 Le système est stable sous la politique la-plus-longue-d'abord

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

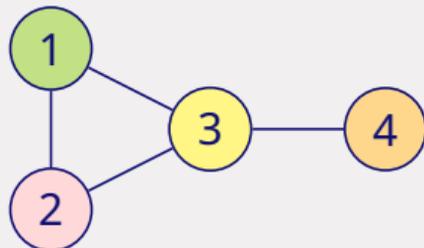
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$ 
  - 😊 Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$
  - 😊 En général, la complexité est polynomiale en  $n$  et  $m$
  - 😊 Le système est stable sous la politique la-plus-longue-d'abord
- Un graphe de compatibilité  $G$  est **stabilisable** si et seulement si  $G$  is non-biparti

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

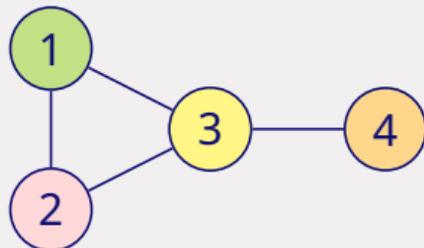
# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si **la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$** 
  - 😊 Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$
  - 😊 En général, la complexité est polynomiale en  $n$  et  $m$
  - 😊 Le système est stable sous la politique la-plus-longue-d'abord
- Un graphe de compatibilité  $G$  est **stabilisable** si et seulement si  **$G$  est surjectif**

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

# Stabilisabilité



- Un problème d'appariement  $(G, \mu)$  est **stabilisable** si et seulement si **la loi de conservation  $A\lambda = \mu$  a une solution  $\lambda > 0$** 
  - ☺ Si  $G$  est bijectif, il suffit de vérifier si  $A^{-1}\mu > 0$
  - ☺ En général, la complexité est polynomiale en  $n$  et  $m$
  - ☺ Le système est stable sous la politique la-plus-longue-d'abord
- Un graphe de compatibilité  $G$  est **stabilisable** si et seulement si  **$G$  est surjectif**
  - ☺ Le rang de la matrice  $A$  est  $n$   
Le noyau de  $A$  est de dimension  $d = m - n$   
(d'après le théorème du rang)

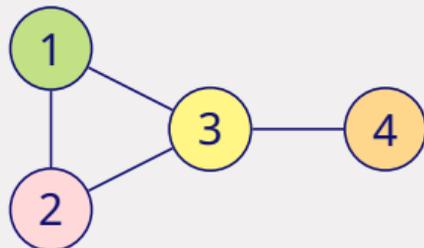
$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \lambda_{1,2} \\ \lambda_{1,3} \\ \lambda_{2,3} \\ \lambda_{3,4} \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \\ \mu_4 \end{bmatrix}$$

## Solutions de la loi de conservation

- Espace affine de dimension  $d = m - n$ :

$$\Lambda = \left\{ \lambda^\circ + \alpha_1 b_1 + \alpha_2 b_2 + \dots + \alpha_d b_d : \alpha \in \mathbb{R}^d \right\}$$

où  $\lambda^\circ$  est une solution quelconque de la loi de conservation  
et  $\{b_1, b_2, \dots, b_d\}$  est une base du noyau de  $A$ .



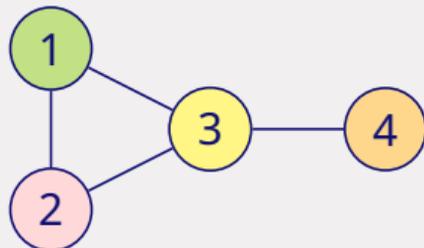
## Solutions de la loi de conservation

- Espace affine de dimension  $d = m - n$ :

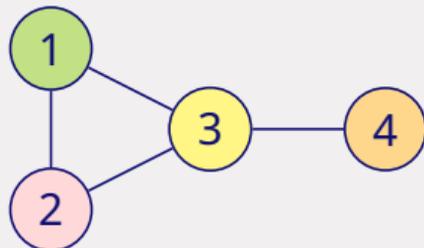
$$\Lambda = \left\{ \lambda^\circ + \alpha_1 b_1 + \alpha_2 b_2 + \dots + \alpha_d b_d : \alpha \in \mathbb{R}^d \right\}$$

où  $\lambda^\circ$  est une solution quelconque de la loi de conservation  
et  $\{b_1, b_2, \dots, b_d\}$  est une base du noyau de  $A$ .

- Un algorithme de (Doob, 1973) permet de construire une base du noyau de  $A$ .



## Solutions de la loi de conservation



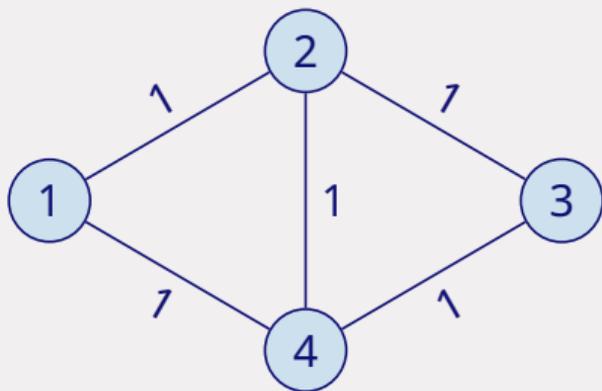
- Espace affine de dimension  $d = m - n$ :

$$\Lambda = \left\{ \lambda^\circ + \alpha_1 b_1 + \alpha_2 b_2 + \dots + \alpha_d b_d : \alpha \in \mathbb{R}^d \right\}$$

où  $\lambda^\circ$  est une solution quelconque de la loi de conservation  
et  $\{b_1, b_2, \dots, b_d\}$  est une base du noyau de  $A$ .

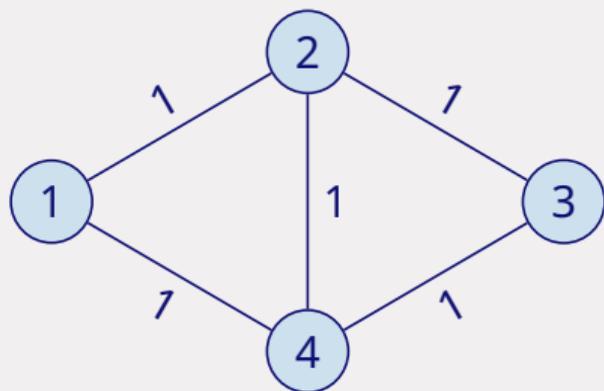
- Un algorithme de (Doob, 1973) permet de construire une base du noyau de  $A$ .
- On utilise deux systèmes de coordonnées :
  - Coordonnées **arêtes**  $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m) \in \mathbb{R}^m$ .
  - Coordonnées **noyau**  $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_d) \in \mathbb{R}^d$ .

## Exemple : Graphe diamant



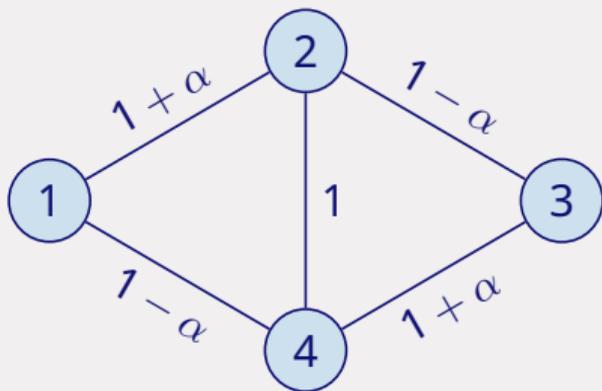
- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution

## Exemple : Graphe diamant



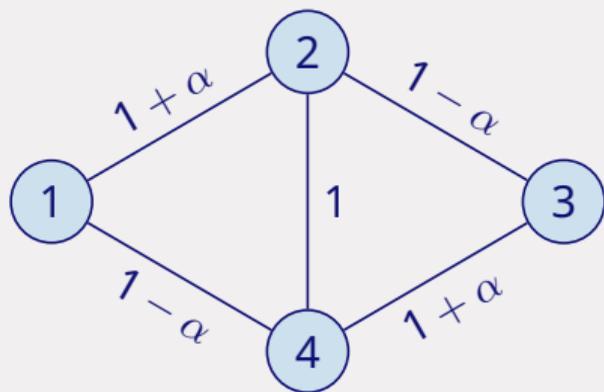
- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$

## Exemple : Graphe diamant



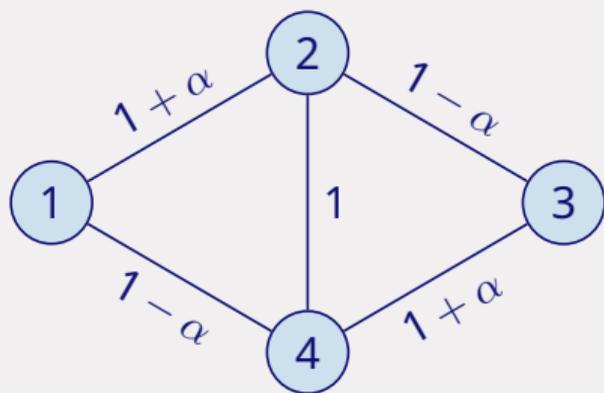
- Taux d'arrivée égaux au degré  
 $\rightarrow \lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$
- Solution générale paramétrée par  $\alpha = \alpha_1$

## Exemple : Graphe diamant



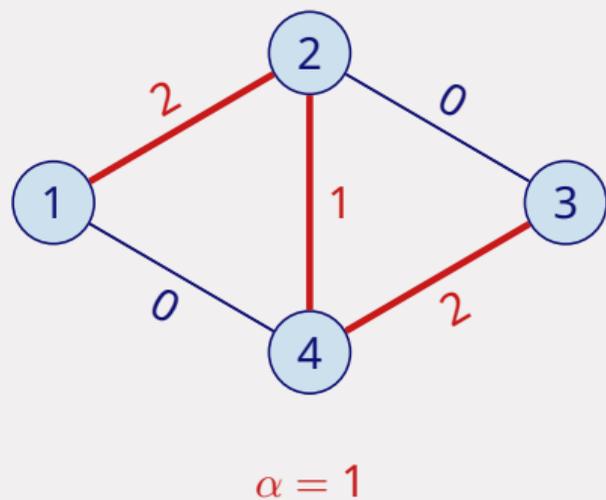
- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$
- Solution générale paramétrée par  $\alpha = \alpha_1$
- Solutions positives avec  $-1 \leq \alpha \leq 1$

## Exemple : Graphe diamant



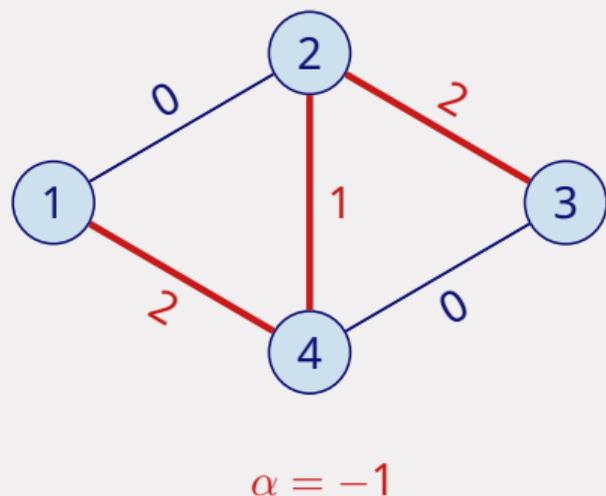
- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$
- Solution générale paramétrée par  $\alpha = \alpha_1$
- Solutions positives avec  $-1 \leq \alpha \leq 1$
- Extrémités → certaines arêtes sont annulées

## Exemple : Graphe diamant



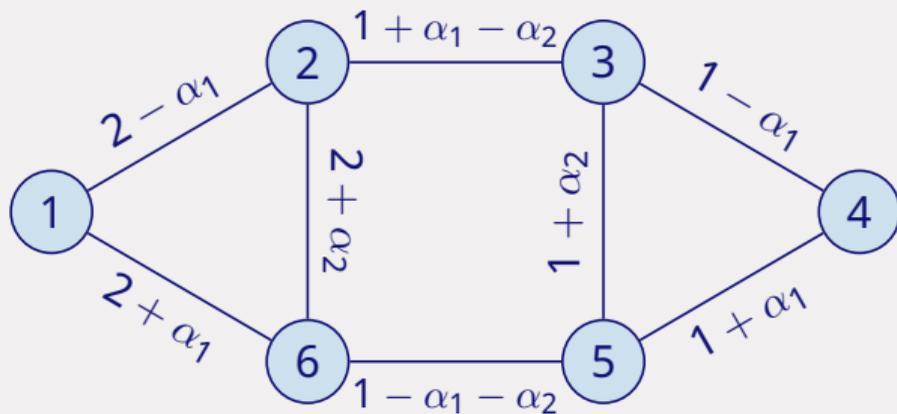
- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$
- Solution générale paramétrée par  $\alpha = \alpha_1$
- Solutions positives avec  $-1 \leq \alpha \leq 1$
- Extrémités → certaines arêtes sont annulées

## Exemple : Graphe diamant

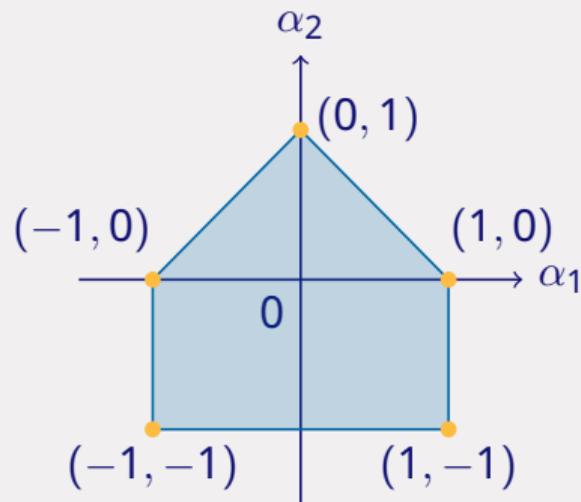
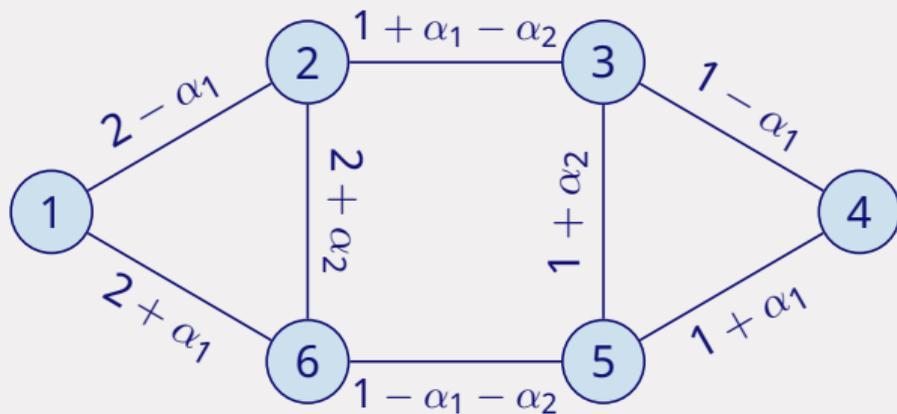


- Taux d'arrivée égaux au degré  
→  $\lambda^\circ = (1, 1, 1, 1, 1)$  est solution
- Dimension du noyau  $d = m - n = 1$
- Solution générale paramétrée par  $\alpha = \alpha_1$
- Solutions positives avec  $-1 \leq \alpha \leq 1$
- Extrémités → certaines arêtes sont annulées

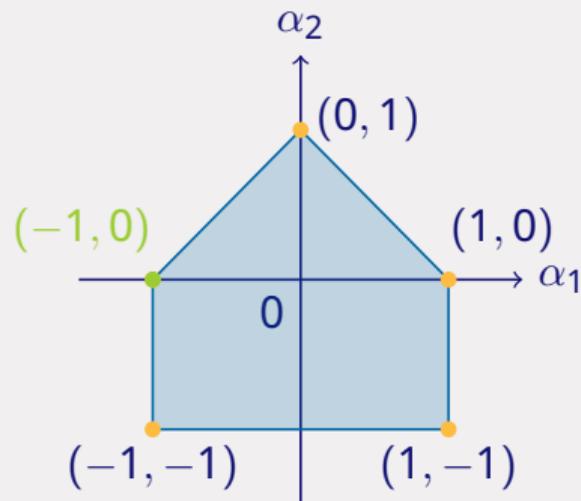
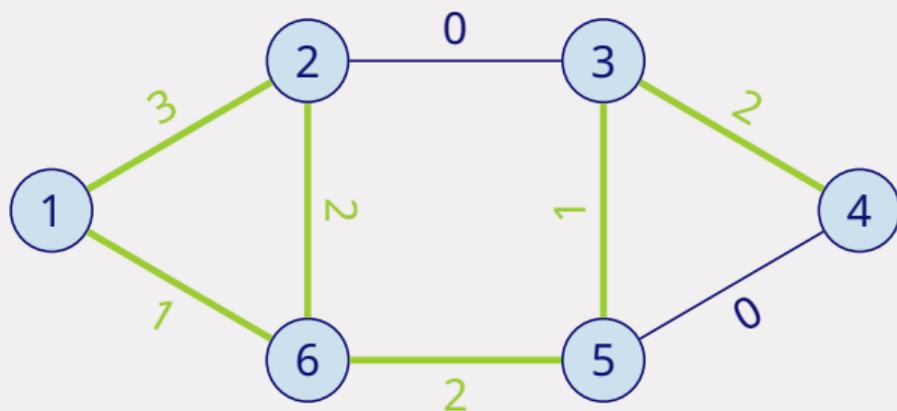
## Exemple : Graphe co-domino



## Exemple : Graphe co-domino



## Exemple : Graphe co-domino



## Solutions positives de la loi de conservation

- Polytope convexe de dimension  $d$ :

$$\Lambda_{\geq 0} = \Lambda \cap \mathbb{R}_+^m \approx \left\{ \alpha \in \mathbb{R}^d : \lambda^\circ + \alpha_1 \mathbf{b}_1 + \alpha_2 \mathbf{b}_2 + \dots + \alpha_d \mathbf{b}_d \geq \mathbf{0} \right\}.$$

## Solutions positives de la loi de conservation

- Polytope convexe de dimension  $d$ :

$$\Lambda_{\geq 0} = \Lambda \cap \mathbb{R}_+^m \approx \left\{ \alpha \in \mathbb{R}^d : \lambda^\circ + \alpha_1 \mathbf{b}_1 + \alpha_2 \mathbf{b}_2 + \dots + \alpha_d \mathbf{b}_d \geq \mathbf{0} \right\}.$$

- Le sous-graphe restreint au support d'un sommet de  $\Lambda_{\geq 0}$  est injectif

## Solutions positives de la loi de conservation

- Polytope convexe de dimension  $d$ :

$$\Lambda_{\geq 0} = \Lambda \cap \mathbb{R}_+^m \approx \left\{ \alpha \in \mathbb{R}^d : \lambda^\circ + \alpha_1 \mathbf{b}_1 + \alpha_2 \mathbf{b}_2 + \dots + \alpha_d \mathbf{b}_d \geq \mathbf{0} \right\}.$$

- Le sous-graphe restreint au support d'un sommet de  $\Lambda_{\geq 0}$  est injectif:
  - Si ce sous-graphe est bijectif :  
Le sommet est **atteignable** avec une politique **filtrante** stable.

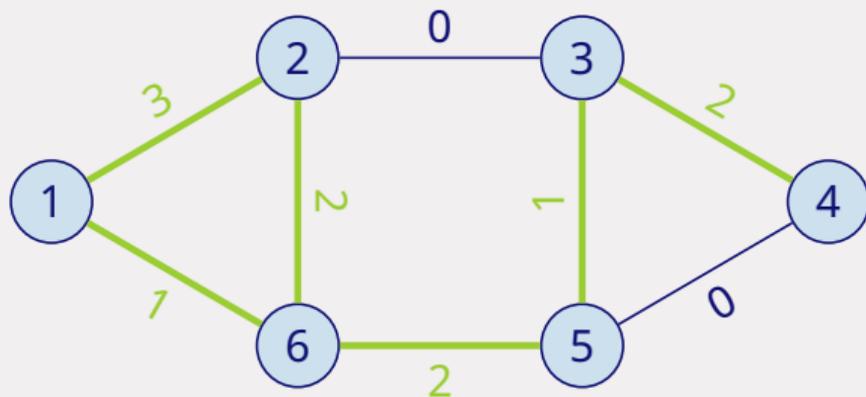
## Solutions positives de la loi de conservation

- Polytope convexe de dimension  $d$ :

$$\Lambda_{\geq 0} = \Lambda \cap \mathbb{R}_+^m \approx \left\{ \alpha \in \mathbb{R}^d : \lambda^\circ + \alpha_1 \mathbf{b}_1 + \alpha_2 \mathbf{b}_2 + \dots + \alpha_d \mathbf{b}_d \geq \mathbf{0} \right\}.$$

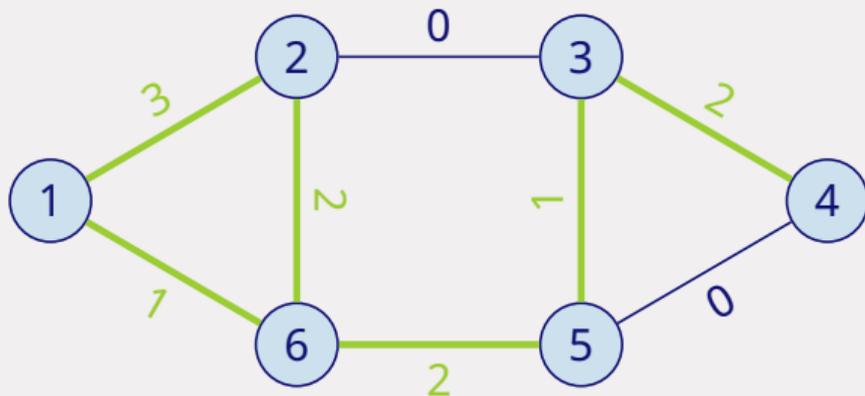
- Le sous-graphe restreint au support d'un sommet de  $\Lambda_{\geq 0}$  est injectif:
  - Si ce sous-graphe est bijectif :  
Le sommet est **atteignable** avec une politique **filtrante** stable.
  - Si ce sous-graphe est strictement injectif :  
Le sommet est **approchable** avec une politique **filtrante à seuil** stable.

## Exemple : Graphe co-domino

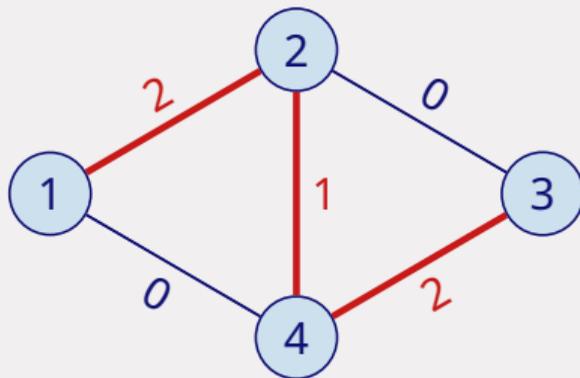


Ce sommet est atteignable  
avec une politique filtrante

## Exemple : Graphe co-domino



Ce sommet est atteignable  
avec une politique filtrante



Ce sommet est approchable  
avec une politique filtrante à seuil

## Solutions **atteignables** de la loi de conservation

L'ensemble des solutions atteignables est **convexe**.

→ On peut atteindre l'intérieur de  $\Lambda_{\geq 0}$ , plus certaines de ses faces.

## Solutions **atteignables** de la loi de conservation

L'ensemble des solutions atteignables est **convexe**.

→ On peut atteindre l'intérieur de  $\Lambda_{\geq 0}$ , plus certaines de ses faces.

**Politique gloutonne ou non-gloutonne ?**

## Solutions **atteignables** de la loi de conservation

L'ensemble des solutions atteignables est **convexe**.

→ On peut atteindre l'intérieur de  $\Lambda_{\geq 0}$ , plus certaines de ses faces.

### **Politique gloutonne ou non-gloutonne ?**

- Il existe toujours une politique gloutonne qui stabilise le système. (Connu.)

## Solutions **atteignables** de la loi de conservation

L'ensemble des solutions atteignables est **convexe**.

→ On peut atteindre l'intérieur de  $\Lambda_{\geq 0}$ , plus certaines de ses faces.

### **Politique gloutonne ou non-gloutonne ?**

- Il existe toujours une politique gloutonne qui stabilise le système. (Connu.)
- Les politique gloutonnes ne permettent pas (en général) d'atteindre tous les taux d'appariement atteignables.

## Solutions **atteignables** de la loi de conservation

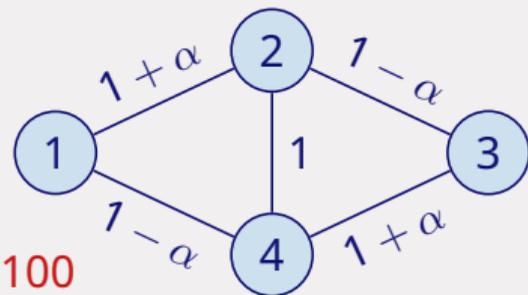
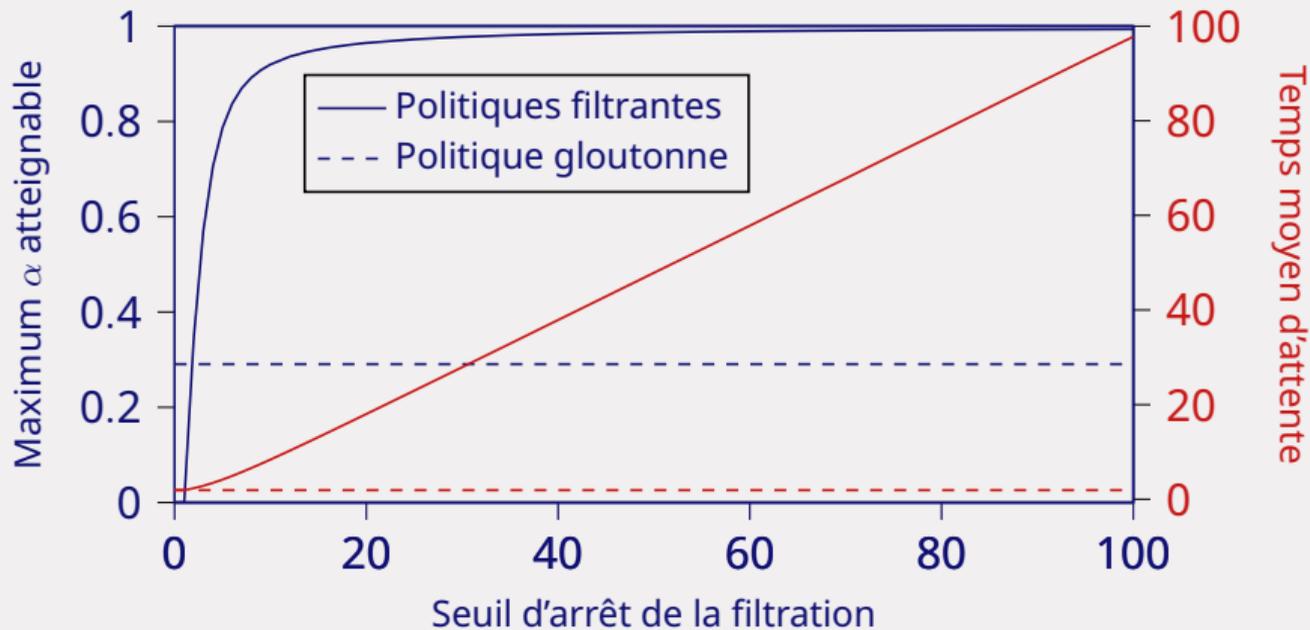
L'ensemble des solutions atteignables est **convexe**.

→ On peut atteindre l'intérieur de  $\Lambda_{\geq 0}$ , plus certaines de ses faces.

### Politique gloutonne ou non-gloutonne ?

- Il existe toujours une politique gloutonne qui stabilise le système. (Connu.)
- Les politique gloutonnes ne permettent pas (en général) d'atteindre tous les taux d'appariement atteignables.
- Pour ça, on a besoin de politiques non-gloutonnes (e.g., filtrantes).

## Résultats numériques



# Conclusion

## Contributions

- Caractérisation des graphes surjectifs/injectifs/bijectifs.
- Condition de stabilisabilité plus simple à vérifier.
- Caractérisation de l'ensemble des taux d'appariement atteignables.
- Politiques filtrantes (à seuil) pour atteindre (s'approcher) des sommets.

# Conclusion

## Contributions

- Caractérisation des graphes surjectifs/injectifs/bijectifs.
- Condition de stabilisabilité plus simple à vérifier.
- Caractérisation de l'ensemble des taux d'appariement atteignables.
- Politiques filtrantes (à seuil) pour atteindre (s'approcher) des sommets.

## Travaux futurs

- Modèle plus réaliste : hypergraphe ? abandons ?
- Et si les taux d'arrivée et/ou la structure du graphe sont inconnus ?