



Electronique

Energie Electrique

Automatique

Master 2 3EA, EC35 - Parcours RoVA



# Systemes Robotiques Hétérogènes et Coopératifs

UPJV, Département EEA

**Fabio MORBIDI**

Laboratoire MIS

Équipe Perception Robotique

E-mail : [fabio.morbidi@u-picardie.fr](mailto:fabio.morbidi@u-picardie.fr)

CM, TD: Lundi et Mercredi 13h30-16h30, salle CURI 305

TP: Mercredi 13h30-16h30, salle TP204

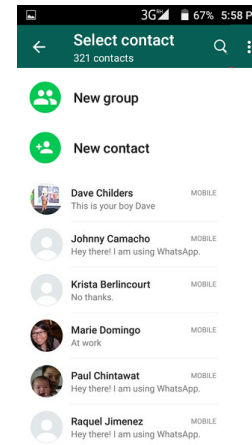
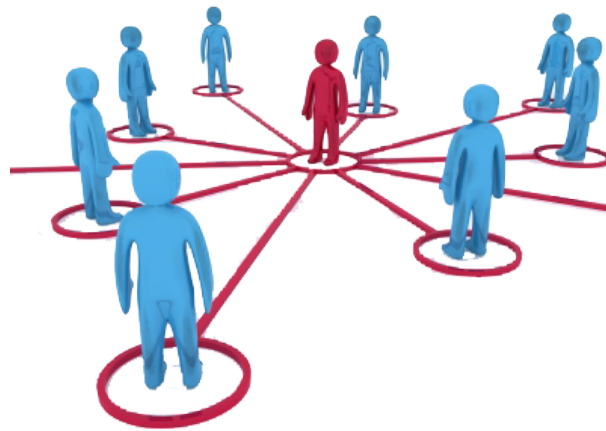
# Plan du cours

1. Systèmes multi-agents: introduction
2. Théorie des graphes
3. Systèmes dynamiques connectés en réseaux: protocole de consensus
4. Problèmes en robotique mobile coopérative



# Introduction

- Le problème de l'**accord** ("agreement") est l'un des problèmes les plus importants pour la coordination d'un système multi-agents  
 ➔ Un groupe d'agents doit *parvenir à un accord* sur la valeur d'un *état partagé*



- Nous étudierons les *dynamiques* du *protocole de consensus* sur des **réseaux statiques non orientés et orientés**
- Nous verrons la **dépendance mutuelle** entre les *propriétés de convergence* du protocole de consensus et les *attributs structurels* du réseau sous-jacent



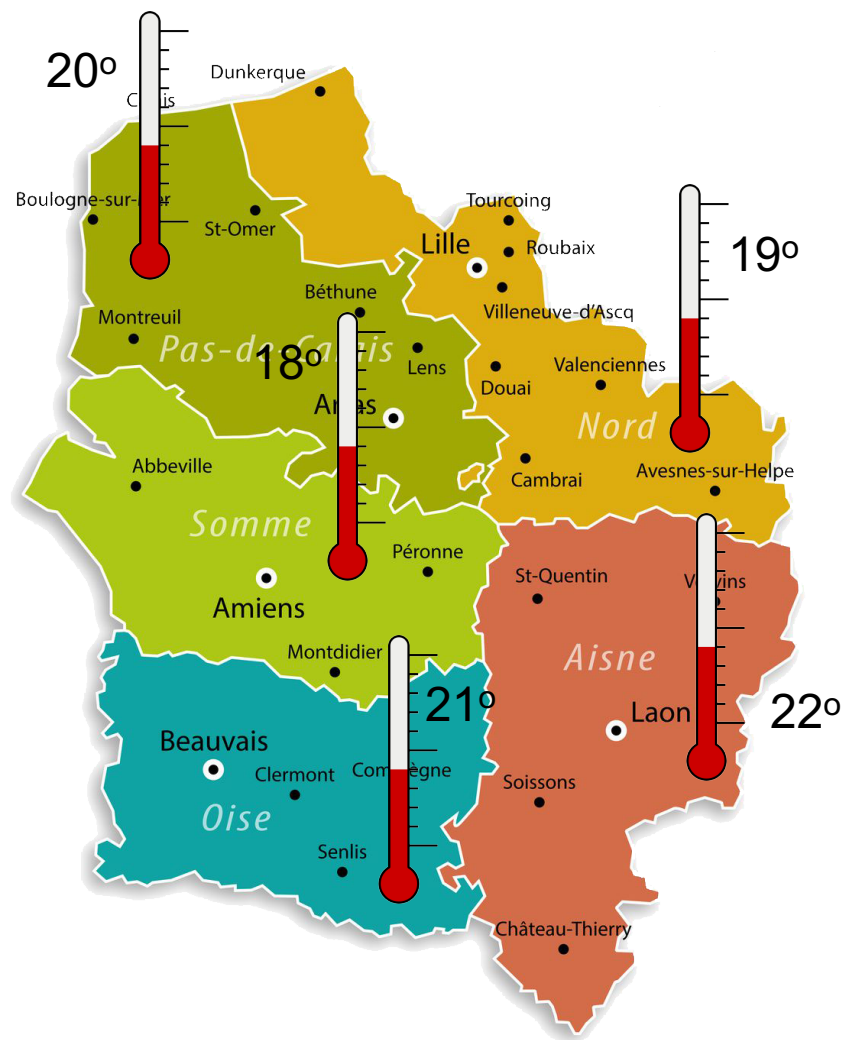
# Protocole de consensus

## Graphes non orientés



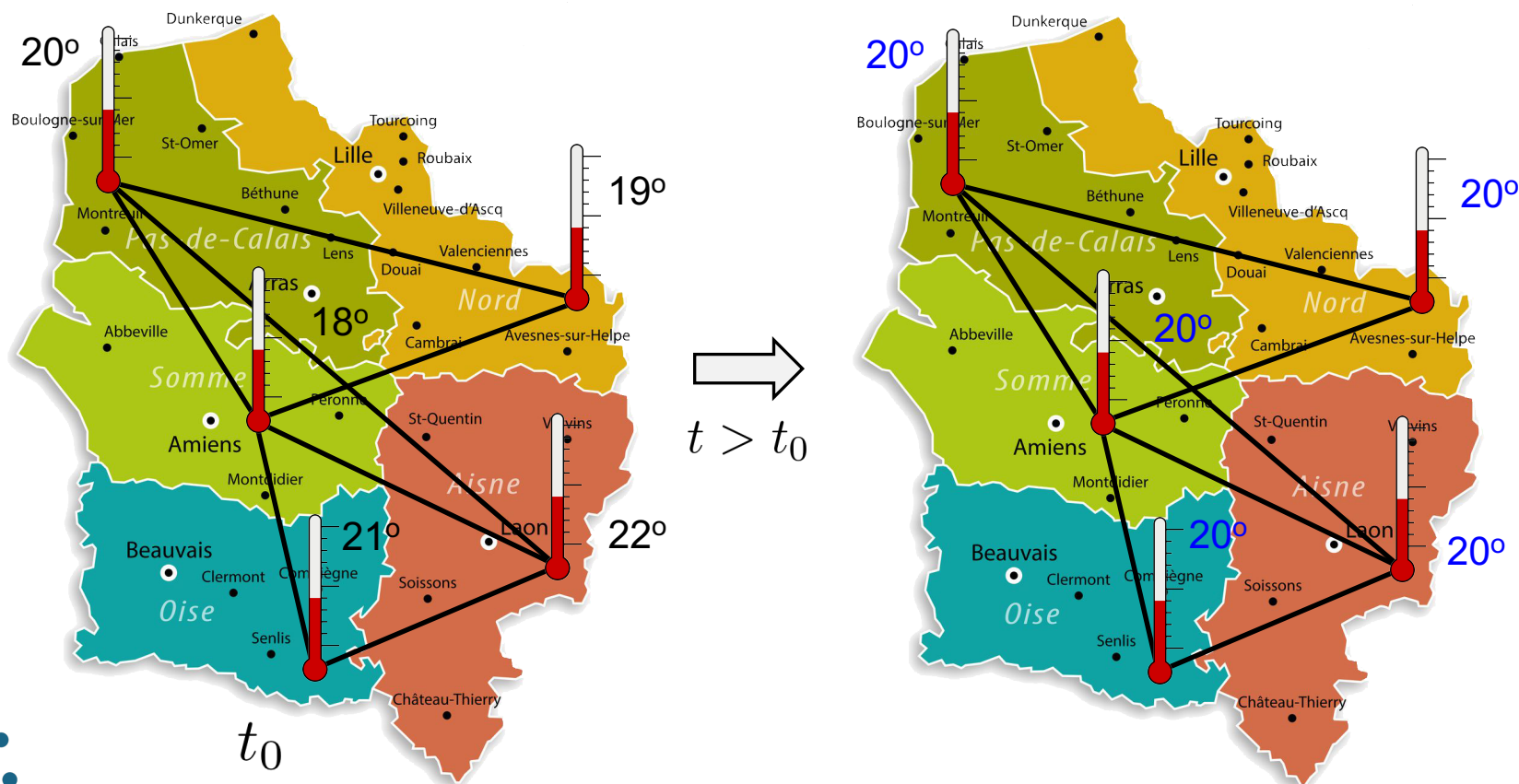
# Protocole de consensus: un exemple motivant

- Un groupe de capteurs mesure la *température* en Hauts-de-France



# Protocole de consensus: un exemple motivant

- En utilisant un **réseau de communication** pour échanger les informations, les capteurs doivent parvenir à un accord sur la **seule valeur** (par exemple 20°) qui représente la température de la région Hauts-de-France
- Dans ce but, le groupe de capteurs doit mettre en œuvre un **protocole** sur le réseau, lui permettant de parvenir à un accord sur la valeur commune mesurée

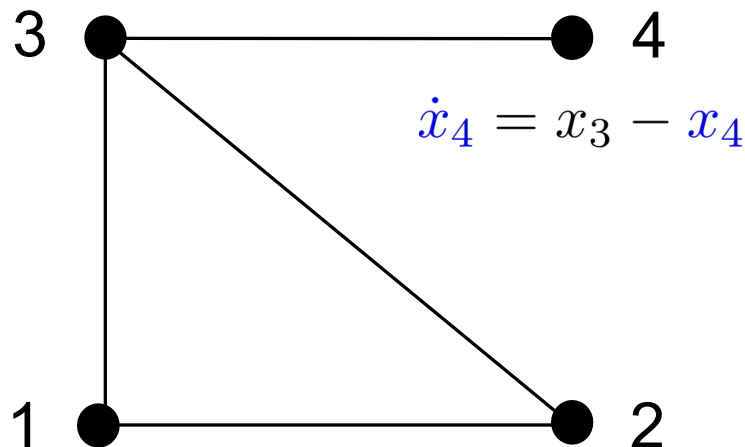


# Protocole de consensus

- Le protocole de consensus implique  $n$  **unités dynamiques** connectées par des liaisons où les *informations relatives* circulent
- On fait l'hypothèse que le **taux de variation** de l'état de chaque unité (la température dans notre exemple) est régi par la somme des états relatifs par rapport aux unités voisines

Exemple:

$$\dot{x}_3 = (x_1 - x_3) + (x_2 - x_3) + (x_4 - x_3)$$



$$\dot{x}_1 = (x_2 - x_1) + (x_3 - x_1) \quad \dot{x}_2 = (x_1 - x_2) + (x_3 - x_2)$$



# Protocole de consensus

- Si l'*état scalaire* du sommet  $i$  est noté  $x_i \in \mathbb{R}$ , alors nous avons que:

$$\dot{x}_i(t) = \sum_{j \in \mathcal{N}(i)} (x_j(t) - x_i(t)), \quad i \in \{1, \dots, n\},$$

où  $\mathcal{N}(i)$  est le voisinage du sommet  $i$  dans le réseau (c'est-à-dire l'ensemble des sommets adjacents à  $i$ )

- Nous pouvons récrire le *système global*, sous forme vectorielle, comme suit:

$$\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathcal{G}) \mathbf{x}(t) \quad \text{dynamique de consensus}$$

- $\mathbf{L}(\mathcal{G})$  est la matrice laplacienne du réseau d'interaction des agents  $\mathcal{G}$
- $\mathbf{x}(t) \triangleq [x_1(t), \dots, x_n(t)]^T \in \mathbb{R}^n$ .





# Protocole de consensus: exemple électrique

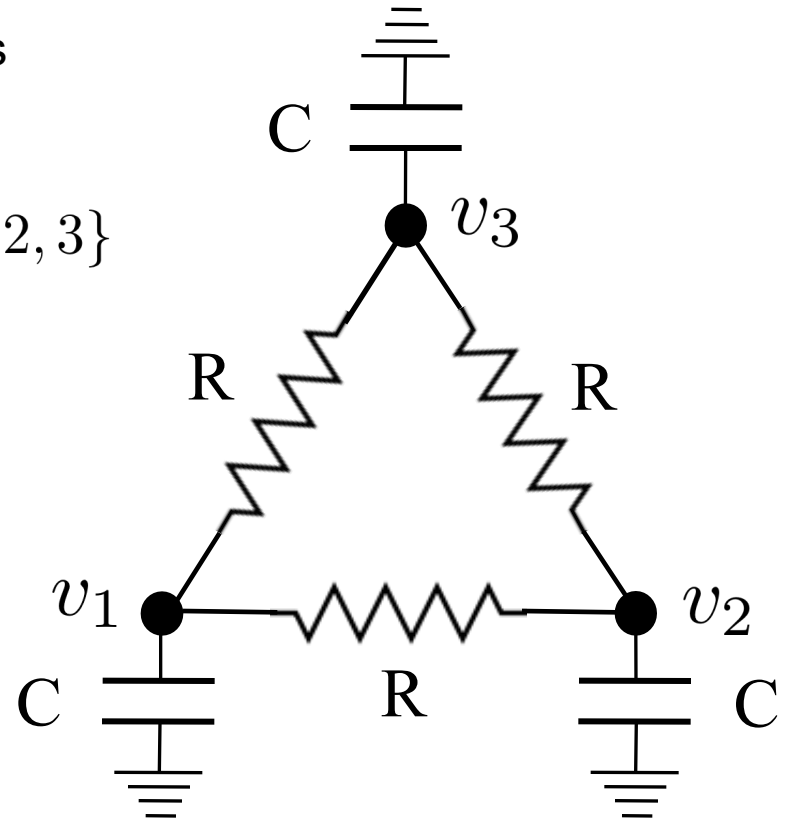
## Exemple (Circuit RC):

- Soient  $R = 1 \Omega$  et  $C = 1 F$
- Si on applique la loi de Kirchhoff des nœuds et des mailles, nous obtenons:

$$\dot{v}_i(t) = \sum_{j \in \mathcal{N}(i)} (v_j(t) - v_i(t)), \quad i \in \{1, 2, 3\}$$

qui décrit la dynamique des tensions sur les trois condensateurs

$\mathcal{N}(i)$  représente l'ensemble des nœuds dans le circuit qui sont connectés au nœud  $i$  par une résistance



# Protocole de consensus: états non scalaires

Le **produit de Kronecker** de deux matrices  $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{m \times n}$  et  $\mathbf{B} = [b_{ij}] \in \mathbb{R}^{p \times q}$ , est la matrice par blocs de taille  $mp \times nq$ , définie par:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}.$$

Commande Matlab:  
`kron`

## Propriétés du produit de Kronecker

Le produit de Kronecker est *bilinéaire* et *associatif* :

$$\mathbf{A} \otimes (\mathbf{B} + \mathbf{C}) = \mathbf{A} \otimes \mathbf{B} + \mathbf{A} \otimes \mathbf{C},$$

$$(\mathbf{B} + \mathbf{C}) \otimes \mathbf{A} = \mathbf{B} \otimes \mathbf{A} + \mathbf{C} \otimes \mathbf{A},$$

$$(k\mathbf{A}) \otimes \mathbf{B} = \mathbf{A} \otimes (k\mathbf{B}) = k(\mathbf{A} \otimes \mathbf{B}),$$

$$(\mathbf{A} \otimes \mathbf{B}) \otimes \mathbf{C} = \mathbf{A} \otimes (\mathbf{B} \otimes \mathbf{C}),$$

$$\mathbf{A} \otimes \mathbf{0} = \mathbf{0} \otimes \mathbf{A} = \mathbf{0},$$

où  $\mathbf{A}$ ,  $\mathbf{B}$  et  $\mathbf{C}$  sont des matrices et  $k$  est un scalaire. Toutefois, il n'est *pas commutatif*, à savoir, en général:  $\mathbf{A} \otimes \mathbf{B} \neq \mathbf{B} \otimes \mathbf{A}$



# Protocole de consensus: états non scalaires

Exemples:

$$a) \begin{bmatrix} 2 \\ 3 \end{bmatrix} \otimes \begin{bmatrix} 2 & -1 & 0 \\ 4 & 1 & 2 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 4 & -2 & 0 \\ 8 & 2 & 4 \\ 0 & -2 & 0 \\ 6 & -3 & 0 \\ 12 & 3 & 6 \\ 0 & -3 & 0 \end{bmatrix}$$

$$b) \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \begin{bmatrix} 2 & -1 & 0 \\ 4 & 1 & 2 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & -1 & 0 & 4 & -2 & 0 \\ 4 & 1 & 2 & 8 & 2 & 4 \\ 0 & -1 & 0 & 0 & -2 & 0 \\ 6 & -3 & 0 & 8 & -4 & 0 \\ 12 & 3 & 6 & 16 & 4 & 8 \\ 0 & -3 & 0 & 0 & -4 & 0 \end{bmatrix}$$

$$c) \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \mathbf{I}_2 = \begin{bmatrix} 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \\ 3 & 0 & 4 & 0 \\ 0 & 3 & 0 & 4 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \otimes \mathbf{I}_3 = ?$$



# Protocole de consensus: états non scalaires

Si  $\mathbf{x}_i \in \mathbb{R}^s$ ,  $s > 1$  (par ex. la position 2D d'un robot ou les mesures de température, humidité et pression d'un capteur), on peut encore écrire la dynamique de consensus de façon compacte en utilisant le **produit de Kronecker**

En effet, nous avons simplement:

$$\dot{\mathbf{x}}(t) = (-\mathbf{L}(\mathcal{G}) \otimes \mathbf{I}_s) \mathbf{x}(t),$$

où

$\mathbf{I}_s$  est la matrice identité de taille  $s \times s$

$$\mathbf{x} = [\mathbf{x}_1^T, \dots, \mathbf{x}_n^T]^T \in \mathbb{R}^{ns}.$$

Exemple:

$$-\mathbf{L}(C_3) = \begin{bmatrix} -2 & 1 & 1 \\ 1 & -2 & 1 \\ 1 & 1 & -2 \end{bmatrix}, \quad -\mathbf{L}(C_3) \otimes \mathbf{I}_2 = \begin{bmatrix} -2 & 0 & 1 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 & 0 & 1 \\ 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 \\ 1 & 0 & 1 & 0 & -2 & 0 \\ 0 & 1 & 0 & 1 & 0 & -2 \end{bmatrix}$$



# Ensemble de consensus

- Dans les exemples précédents, nous avons observé que l'état de chaque sommet dans le réseau est “**tiré**” vers les états des sommets voisins
- Est-ce que tous les sommets parviendront à une **moyenne pondérée** de leurs états initiaux, qui correspond aussi à un **point fixe** de la dynamique collective ?

**Définition** (Ensemble de *consensus*)

L'**ensemble de consensus**  $\mathcal{A} \subseteq \mathbb{R}^n$  est le sous-espace  $\text{span}\{\mathbf{1}\}$ , c'est-à-dire:

$$\mathcal{A} = \{\mathbf{x} \in \mathbb{R}^n \mid x_i = x_j, \text{ pour tous les } i, j\}.$$



# Ensemble de consensus

- Dans les exemples précédents, nous avons observé que l'état de chaque sommet dans le réseau est “**tiré**” vers les états des sommets voisins
- Est-ce que tous les sommets parviendront à une **moyenne pondérée** de leurs états initiaux, qui correspond aussi à un **point fixe** de la dynamique collective ?

---

## Définition (Ensemble de *consensus*)

L'**ensemble de consensus**  $\mathcal{A} \subseteq \mathbb{R}^n$  est le sous-espace  $\text{span}\{\mathbf{1}\}$ , c'est-à-dire:

$$\mathcal{A} = \{\mathbf{x} \in \mathbb{R}^n \mid x_i = x_j, \text{ pour tous les } i, j\}.$$

## Exemples:

$$[2, 2, \dots, 2]^T \in \mathcal{A}, \quad [-7, -7, \dots, -7]^T \in \mathcal{A}.$$

En d'autres termes,  $\mathcal{A}$  contient tous les vecteurs de la forme  $\alpha\mathbf{1}$ ,  $\alpha \in \mathbb{R}$



# Rappel: stabilité d'un système LTI

- Soit le système linéaire invariant dans le temps (LTI) à temps continu:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$$

où

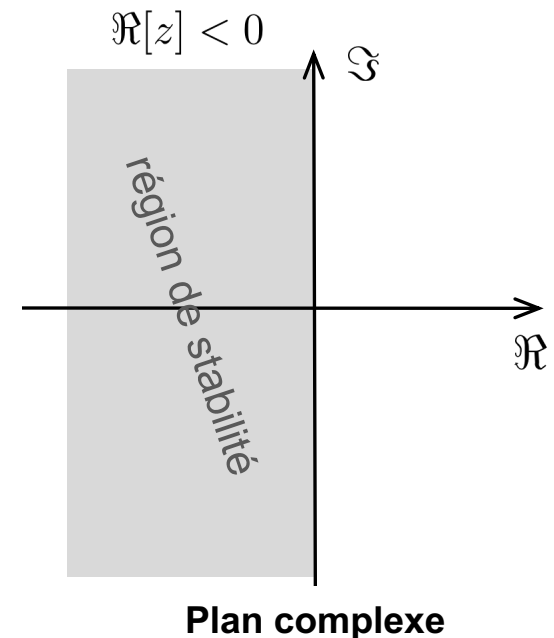
$\mathbf{x} \in \mathbb{R}^n$  : état du système avec condition initiale  $\mathbf{x}(0) = \mathbf{x}_0$

$\mathbf{u} \in \mathbb{R}^m$  : entrée du système

$\mathbf{A} \in \mathbb{R}^{n \times n}$ ,  $\mathbf{B} \in \mathbb{R}^{n \times m}$  : matrices constantes

Le système est dit:

- Asymptotiquement stable** si et seulement si toutes les valeurs propres de  $\mathbf{A}$  se trouvent dans la région de stabilité (grisée)
- Unstable** si au moins une valeur propre de  $\mathbf{A}$  est à l'extérieur de la région de stabilité
- Marginalement stable** si toutes les valeurs propres de  $\mathbf{A}$  se trouvent à l'intérieur de la région de stabilité ou sur son bord (l'axe imaginaire), et celles qui sont sur le bord sont *simples* (c.-à-d. multiplicité algébrique = multiplicité géométrique)



# Vers le consensus: graphes non orientés

Le spectre de la matrice laplacienne d'un graphe *connexe* non orienté est:

$$0 = \lambda_1(\mathbf{L}(\mathcal{G})) < \lambda_2(\mathbf{L}(\mathcal{G})) \leq \dots \leq \lambda_n(\mathbf{L}(\mathcal{G})),$$

et  $\mathbf{1} = [1, 1, \dots, 1]^T$  est le *vecteur propre* qui correspond à la *valeur propre zéro*.

Considérons la **factorisation spectrale** de la matrice laplacienne ( $\mathbf{L}(\mathcal{G})$  est une *matrice symétrique*):

$$\mathbf{L}(\mathcal{G}) = \mathbf{U} \mathbf{\Lambda}(\mathcal{G}) \mathbf{U}^T,$$

où

$$\mathbf{\Lambda}(\mathcal{G}) = \text{diag}(\lambda_1(\mathcal{G}), \dots, \lambda_n(\mathcal{G})),$$

et

$$\mathbf{U} = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \cdots \quad \mathbf{u}_n],$$

est une matrice  $n \times n$  dont les colonnes sont les vecteurs propres *normalisés* et mutuellement orthogonaux de  $\mathbf{L}(\mathcal{G})$ , avec  $\mathbf{u}_1 = (1/\sqrt{n}) \mathbf{1}$

**Remarque:** pour plus de simplicité, par la suite on écrira  $\lambda_i(\mathcal{G})$  au lieu de  $\lambda_i(\mathbf{L}(\mathcal{G}))$





# Vers le consensus: graphes non orientés

- La solution de l'équation différentielle  $\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathcal{G}) \mathbf{x}(t)$ , avec état initial  $\mathbf{x}(0) = \mathbf{x}_0$ , est:

$$\mathbf{x}(t) = e^{-\mathbf{L}(\mathcal{G})t} \mathbf{x}_0.$$

Si nous utilisons la *factorisation spectrale* de la matrice laplacienne, nous avons:

$$\begin{aligned} \mathbf{x}(t) &= e^{-\mathbf{L}(\mathcal{G})t} \mathbf{x}_0 \\ &= e^{-(\mathbf{U}\mathbf{\Lambda}(\mathcal{G})\mathbf{U}^T)t} \mathbf{x}_0 \\ &= \mathbf{U}e^{-\mathbf{\Lambda}(\mathcal{G})t} \mathbf{U}^T \mathbf{x}_0 \\ &= (e^{-\lambda_1(\mathcal{G})t} \mathbf{u}_1 \mathbf{u}_1^T + e^{-\lambda_2(\mathcal{G})t} \mathbf{u}_2 \mathbf{u}_2^T + \dots + e^{-\lambda_n(\mathcal{G})t} \mathbf{u}_n \mathbf{u}_n^T) \mathbf{x}_0 \\ &\stackrel{\ominus}{=} e^{-\lambda_1(\mathcal{G})t} (\mathbf{u}_1^T \mathbf{x}_0) \mathbf{u}_1 + e^{-\lambda_2(\mathcal{G})t} (\mathbf{u}_2^T \mathbf{x}_0) \mathbf{u}_2 + \dots + e^{-\lambda_n(\mathcal{G})t} (\mathbf{u}_n^T \mathbf{x}_0) \mathbf{u}_n. \end{aligned}$$

↖ Décomposition selon chaque "axe propre"



# Vers le consensus: graphes non orientés

## Théorème

Soit  $\mathcal{G}$  un *graphe connexe*. Alors, le protocole  $\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathcal{G}) \mathbf{x}(t)$  converge vers l'ensemble de consensus  $\mathcal{A}$  avec une vitesse de convergence dictée par  $\lambda_2(\mathbf{L}(\mathcal{G}))$

## Preuve

La preuve est une conséquence directe de l'équation,

$$\mathbf{x}(t) = e^{-\lambda_1(\mathcal{G})t} (\mathbf{u}_1^T \mathbf{x}_0) \mathbf{u}_1 + e^{-\lambda_2(\mathcal{G})t} (\mathbf{u}_2^T \mathbf{x}_0) \mathbf{u}_2 + \dots + e^{-\lambda_n(\mathcal{G})t} (\mathbf{u}_n^T \mathbf{x}_0) \mathbf{u}_n,$$

si on observe que pour un graphe connexe  $\lambda_i(\mathcal{G}) > 0$ , pour  $i \geq 2$ ; comme toujours,  $\lambda_1(\mathcal{G}) = 0$ . Donc,

$$\lim_{t \rightarrow \infty} \mathbf{x}(t) = (\mathbf{u}_1^T \mathbf{x}_0) \mathbf{u}_1 = \frac{\mathbf{1}^T \mathbf{x}_0}{n} \mathbf{1} = \left( \frac{1}{n} \sum_{i=1}^n x_i(0) \right) \mathbf{1} \quad \text{Consensus en moyenne}$$

et par conséquent  $\mathbf{x}(t) \rightarrow \mathcal{A}$ . Comme  $\lambda_2(\mathcal{G})$  est la valeur propre positive la plus faible de la matrice laplacienne, elle détermine le *mode de convergence le plus lent* dans la limite ci-dessus. ■

# Vers le consensus: graphes non orientés

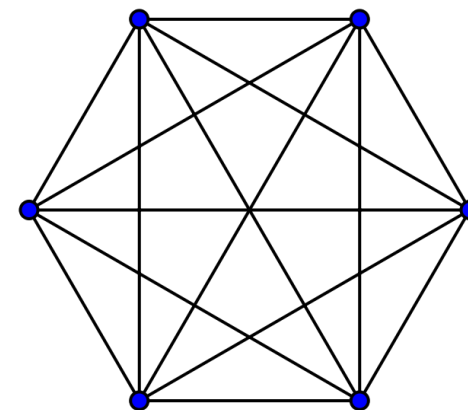
Exemples ( $n = 6$ ):

## Graphe complet

Le spectre de la matrice laplacienne de  $K_6$  est:

$$\{0, 6, 6, 6, 6, 6\}$$

Donc  $\lambda_2(K_6) = 6$

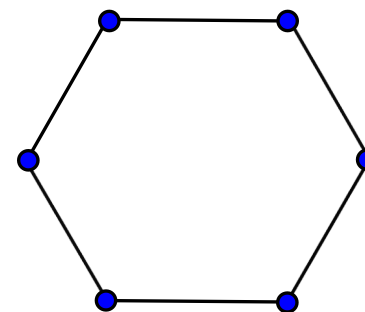


## Graphe cycle

Les valeurs propres de la matrice laplacienne de  $C_6$  sont:

$$2 - 2 \cos \left( \frac{\pi k}{3} \right), \quad k \in \{0, 1, 2, 3, 4, 5\}$$

Donc  $\lambda_2(C_6) = 1$



**Remarque:**  $\lambda_2(K_6) > \lambda_2(C_6)$



# Vers le consensus: remarque I

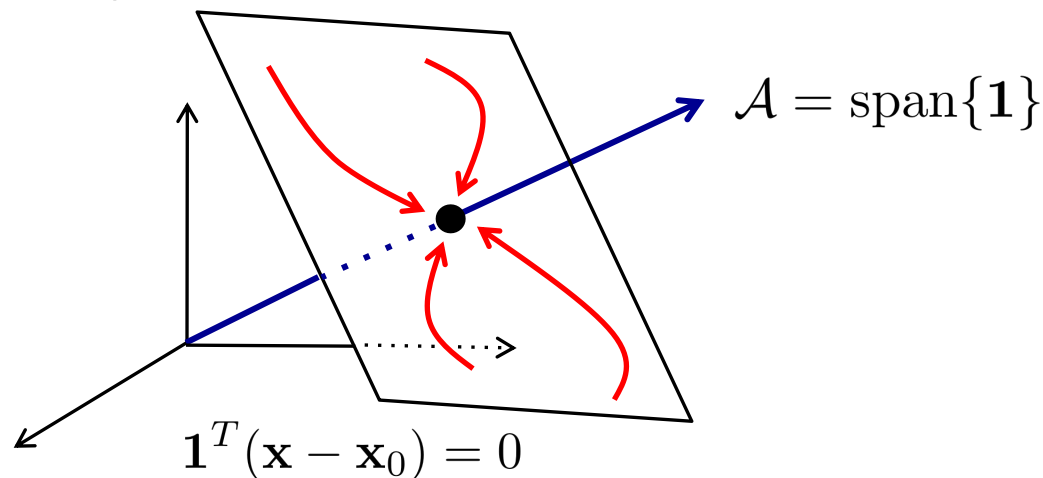
Il est à noter que:

$$\frac{d}{dt} \left( \frac{\mathbf{1}^T \mathbf{x}(t)}{n} \right) = \frac{\mathbf{1}^T (-\mathbf{L}(\mathcal{G}) \mathbf{x}(t))}{n} = -\frac{\mathbf{x}^T(t) \mathbf{L}(\mathcal{G}) \mathbf{1}}{n} = 0.$$

Donc, la quantité

$$\frac{\mathbf{1}^T \mathbf{x}(t)}{n} = \frac{1}{n} \sum_{i=1}^n x_i(t), \quad \forall t \geq 0,$$

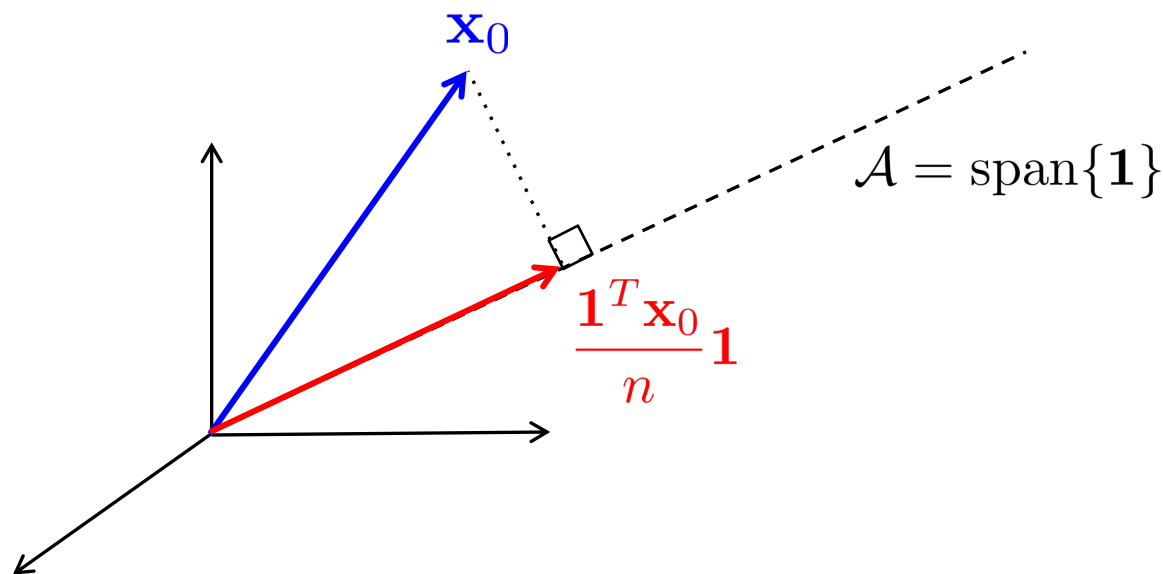
c'est-à-dire le **barycentre** des états des agents, est une **constante du mouvement** pour la dynamique de consensus  $\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathcal{G}) \mathbf{x}(t)$  (à savoir, elle est *conservée* pendant l'évolution du système)



## Vers le consensus: remarque II

La trajectoire d'état générée par le protocole de consensus converge vers la **projection** de l'état initial  $\mathbf{x}_0$  sur le sous-espace de consensus  $\mathcal{A}$ . En effet:

$$\arg \min_{\mathbf{x} \in \mathcal{A}} \|\mathbf{x} - \mathbf{x}_0\|_2 = \frac{\mathbf{1}^T \mathbf{x}_0}{\mathbf{1}^T \mathbf{1}} \mathbf{1} = \frac{\mathbf{1}^T \mathbf{x}_0}{n} \mathbf{1}.$$

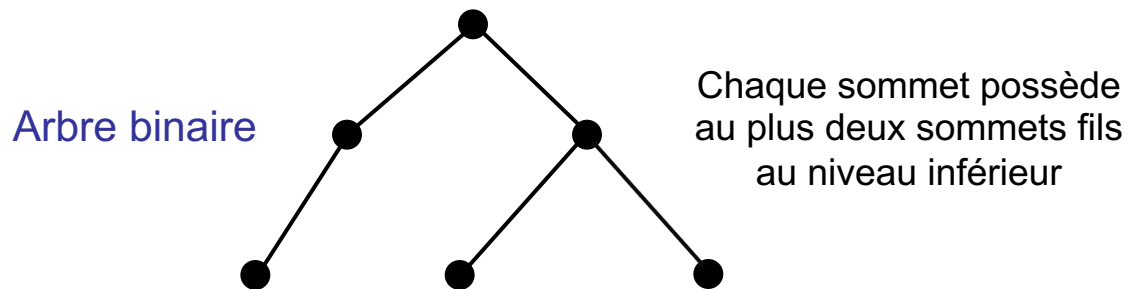
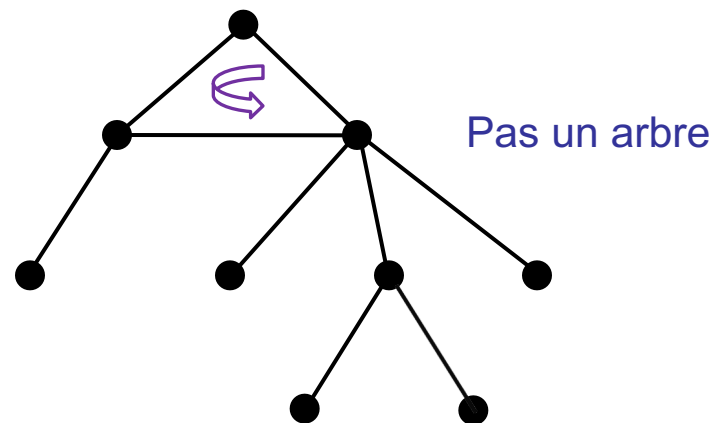
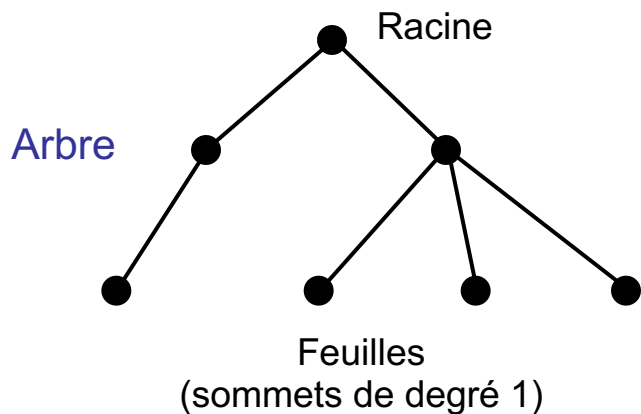


# Vers le consensus: remarque III

## Définition (*Arbre*)

Un **arbre** est un graphe connexe sans cycles (circuits)

Exemples:



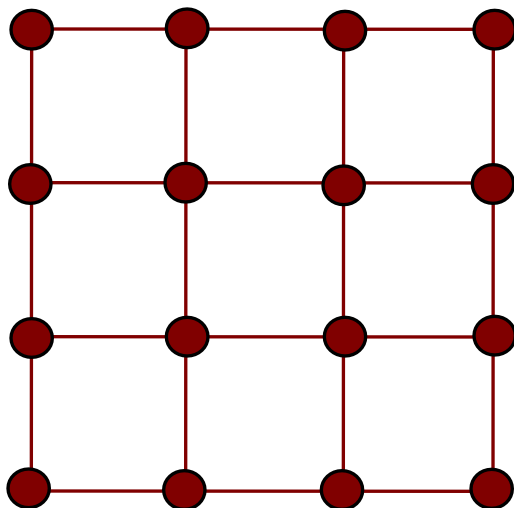
# Vers le consensus: remarque III

**Définition** (*Arbre couvrant* ou “*spanning tree*“)

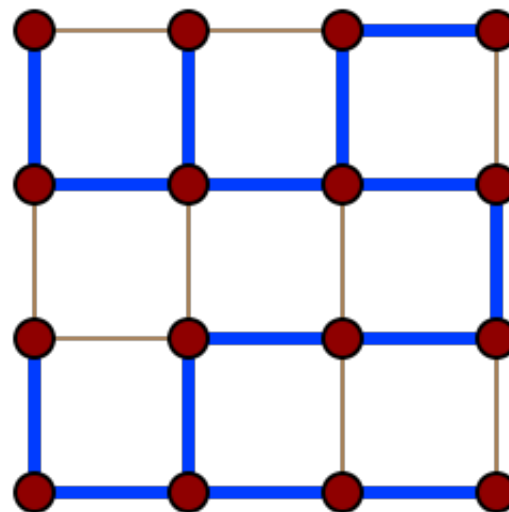
Un **arbre couvrant** d'un graphe connexe  $\mathcal{G}$  est un arbre composé de **tous les sommets** et de certaines (ou éventuellement toutes) les arêtes de  $\mathcal{G}$

En d'autres termes, un arbre couvrant de  $\mathcal{G}$  est un sous-ensemble des arêtes de  $\mathcal{G}$  formant un arbre qui couvre tous les sommets

Exemple:



Graphe grille  $\mathcal{G}$



**Bleu:** arbre couvrant du graphe  $\mathcal{G}$



# Vers le consensus: remarque III

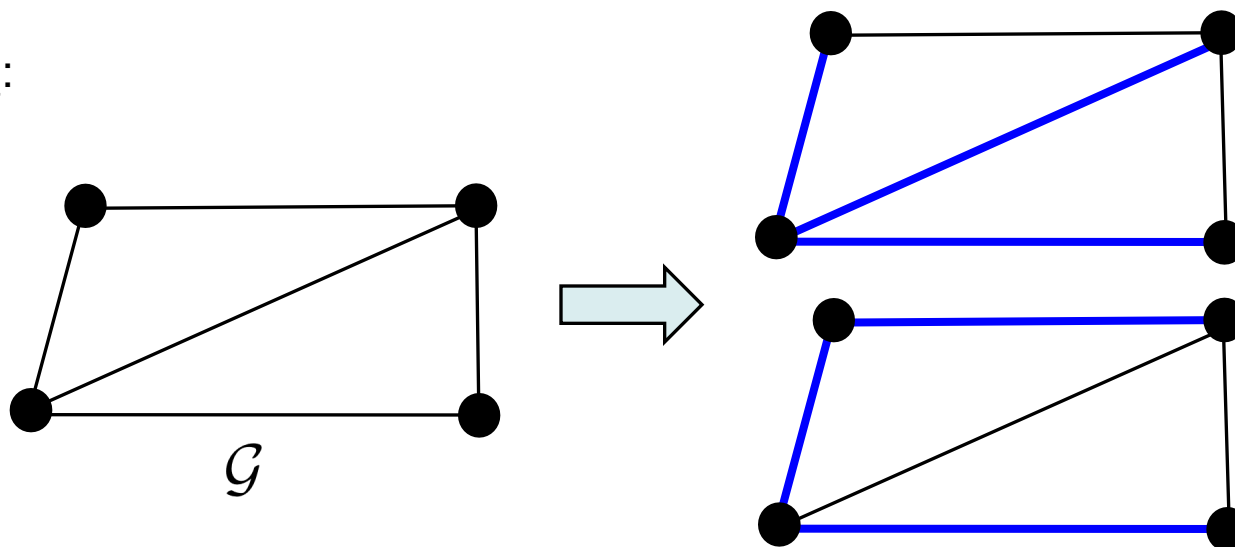
## Proposition

Une condition *nécessaire* et *suffisante* pour la convergence du protocole de consensus

$$\dot{\mathbf{x}}(t) = -\mathbf{L}(\mathcal{G}) \mathbf{x}(t),$$

vers l'ensemble de consensus  $\mathcal{A}$  pour toute condition initiale  $\mathbf{x}_0$ , est que le graphe  $\mathcal{G}$  contient un **arbre couvrant**

Exemple:



Deux possibles  
arbres couvrants  
de  $\mathcal{G}$  (bleus)





# Exemple 1: problème du rendez-vous

## Protocole de consensus “en action”

- $n$  robots mobiles avec un modèle dynamique de type *integrateur*

$$\dot{\mathbf{x}}_i = \mathbf{u}_i, \quad i \in \{1, \dots, n\},$$

doivent se rencontrer au même endroit. Ici,  $\mathbf{x}_i = [x_i, y_i]^T$  indique la position du robot  $i$  dans le plan

- Le point de rendez-vous **n'est pas connu** au préalable et les robots **n'ont pas accès** à leurs *positions absolues* (pas de GPS embarqué)
- Les robots peuvent uniquement mesurer les **déplacements relatifs** par rapport aux robots voisins

**Problème:** comment définir l'entrée de commande  $\mathbf{u}_i$  du robot  $i$  ?



# Exemple 1: problème du rendez-vous

## Protocole de consensus “en action”

- $n$  robots mobiles avec un modèle dynamique de type *integrateur*

$$\dot{\mathbf{x}}_i = \mathbf{u}_i, \quad i \in \{1, \dots, n\},$$

doivent se rencontrer au même endroit. Ici,  $\mathbf{x}_i = [x_i, y_i]^T$  indique la position du robot  $i$  dans le plan

- Le point de rendez-vous **n'est pas connu** au préalable et les robots **n'ont pas accès** à leurs *positions absolues* (pas de GPS embarqué)
- Les robots peuvent uniquement mesurer les **déplacements relatifs** par rapport aux robots voisins

### **Solution:**

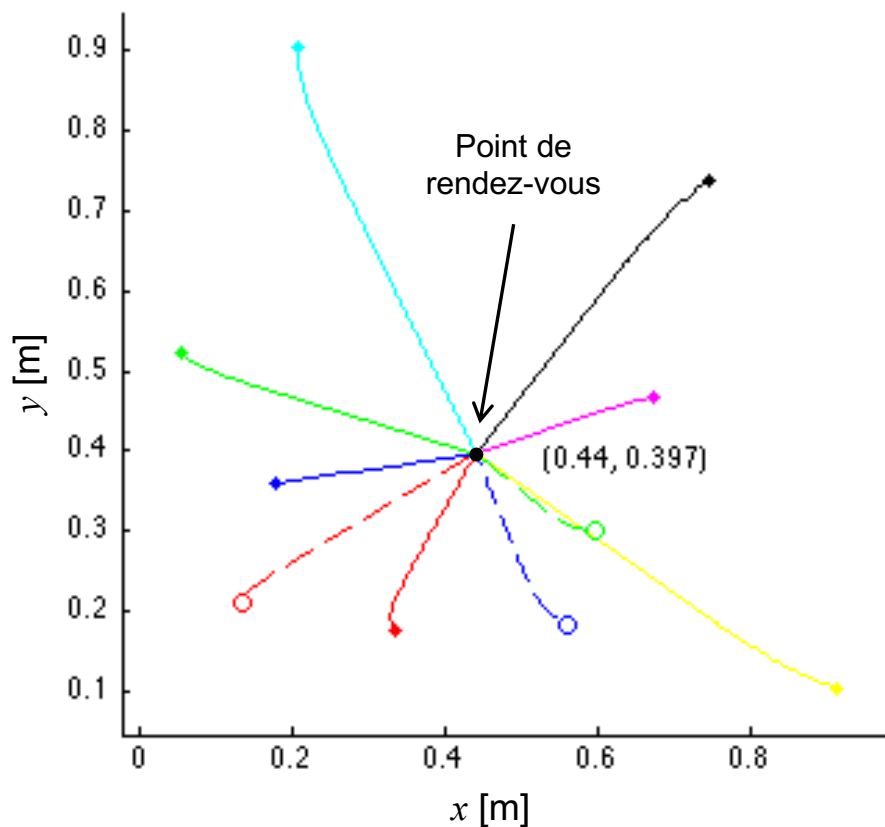
Choisir l'*entrée de commande* suivante pour le robot  $i$ :

$$\mathbf{u}_i(t) = \sum_{j \in \mathcal{N}(i)} (\mathbf{x}_j(t) - \mathbf{x}_i(t))$$

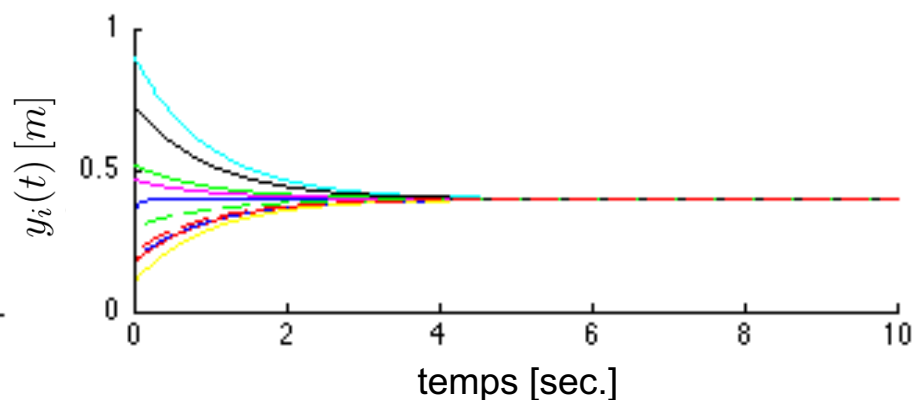
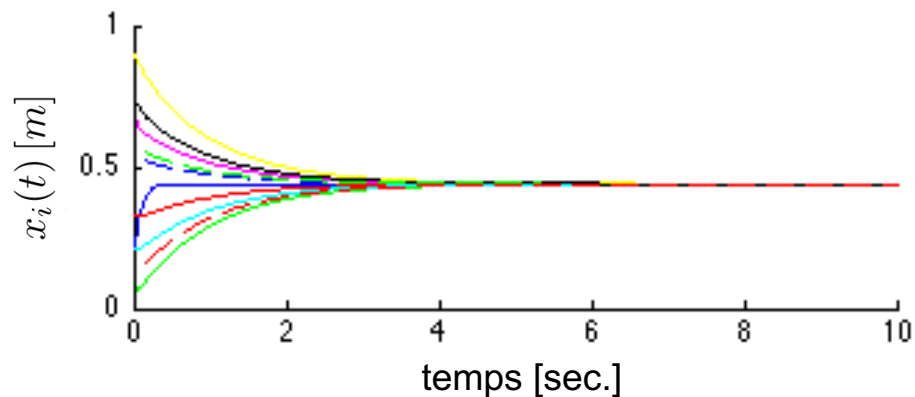


# Exemple 1: problème du rendez-vous

- Simulation Matlab (Topologie du graphe:  $S_{10}$ )



Trajectoire 2D des 10 robots



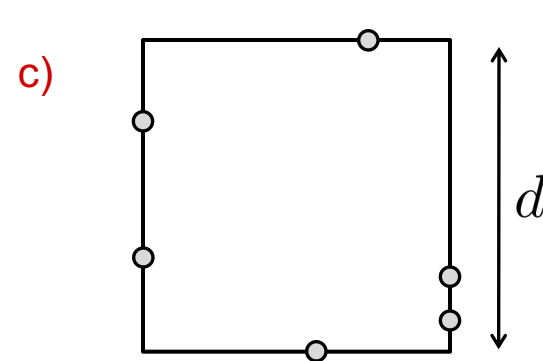
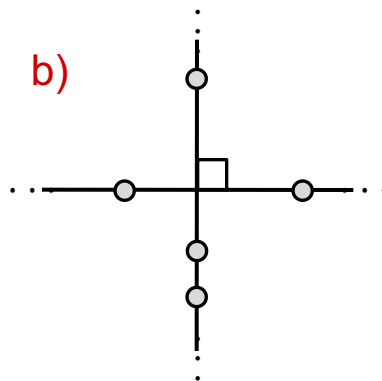
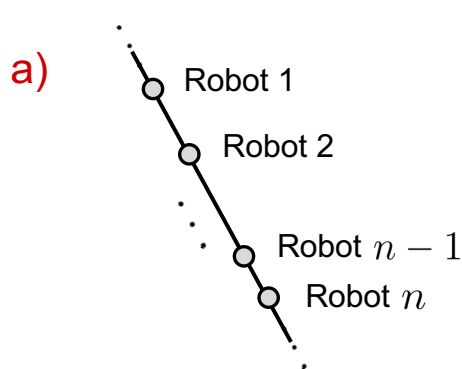
Évolution temporelle des coordonnées  $x$  et  $y$  des 10 robots



# Exemple 1: problème du rendez-vous

## Exercice [Matlab]:

1. Choisir des topologies du graphe de communication  $\mathcal{G}$  différentes et étudier la *vitesse de convergence* vers le point de rendez-vous
2. Écrire l'*expression mathématique* du point de rendez-vous
3. Étudier le comportement des robots lorsque ils se trouvent tous, à l'*instant initial*  $t_0$ , sur :
  - a) La même droite
  - b) Deux droites qui se coupent à angle droit
  - c) Les quatre côtés de longueur  $d$  d'un carré. Est-ce que les trajectoires des robots sortiront à l'extérieur du carré ?



# Exemple 1: problème du rendez-vous

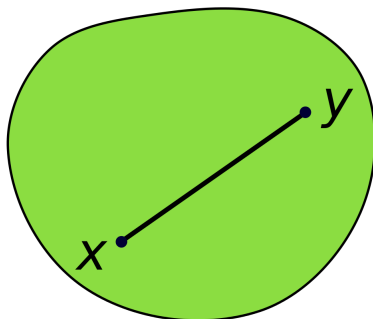
## Exercice [Matlab]:

Pour le point 3 de l'exercice, on rappelle les définitions suivantes:

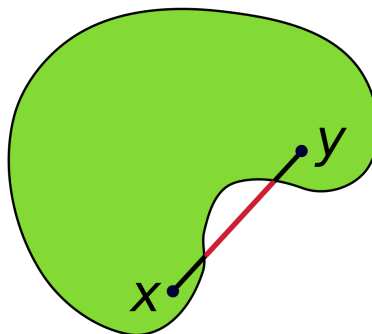
1. Un ensemble  $C$  est dit **convexe** lorsque, pour tous  $x, y \in C$ , le segment  $[x, y]$  est tout entier contenu dans  $C$ , c'est-à-dire:

$$\forall x, y \in C, \forall \alpha \in [0, 1], \alpha x + (1 - \alpha)y \in C$$

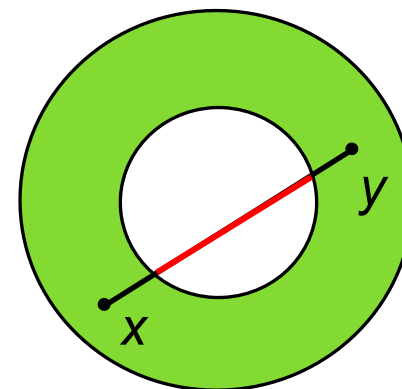
Exemples:



Convexe



Non convexe



Non convexe

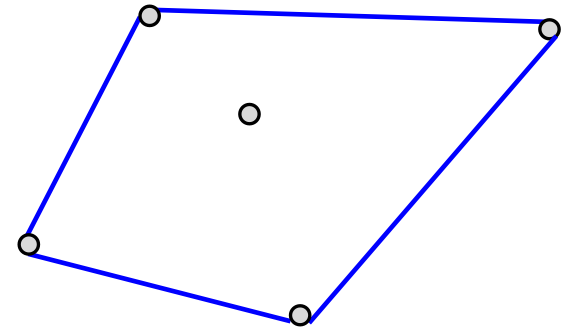
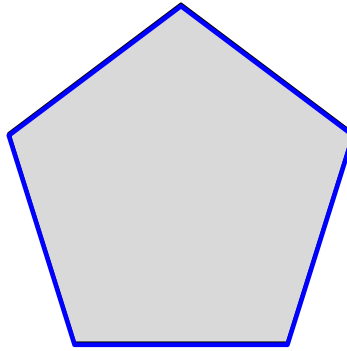
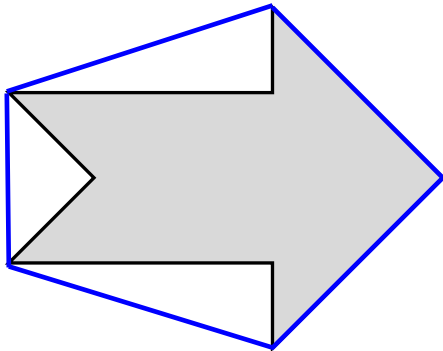


# Exemple 1: problème du rendez-vous

## Exercice [Matlab]:

2. L'**enveloppe convexe** ("convex hull") d'un objet ou d'un regroupement d'objets géométriques est l'ensemble convexe **le plus petit** parmi ceux qui le contiennent

Exemples:



- Objet(s): **gris**
- Enveloppe convexe: **ensemble bleu**



## Exemple 2: poursuite cyclique ( $n$ -bug problem)

- Considérons  $n$  robots mobiles avec un modèle dynamique de type *integrateur*

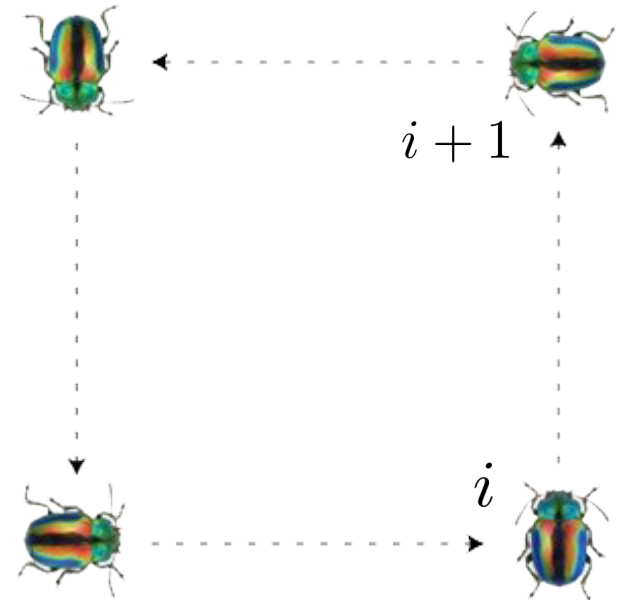
$$\dot{\mathbf{x}}_i = \mathbf{u}_i, \quad i \in \{1, \dots, n\},$$

et supposons que le robot  $i$  doit **poursuivre le** robot  $i + 1$  modulo  $n$ .  
Comme auparavant  $\mathbf{x}_i \in \mathbb{R}^2$  indique la position du robot  $i$  dans le plan

- Nous pouvons choisir l'*entrée de commande* suivante pour le robot  $i$ :

$$\mathbf{u}_i(t) = k (\mathbf{x}_{i+1}(t) - \mathbf{x}_i(t)),$$

où  $k$  est un *gain positif*. Plus  $k$  est grand plus la "force d'attraction" entre deux robots sera intense



## Exemple 2: poursuite cyclique ( $n$ -bug problem)

- Cela nous amène à nouveau au système dynamique suivant (protocole de consensus):

$$\dot{\mathbf{x}}(t) = (-\mathbf{L}(\mathcal{G}) \otimes \mathbf{I}_2) \mathbf{x}(t),$$

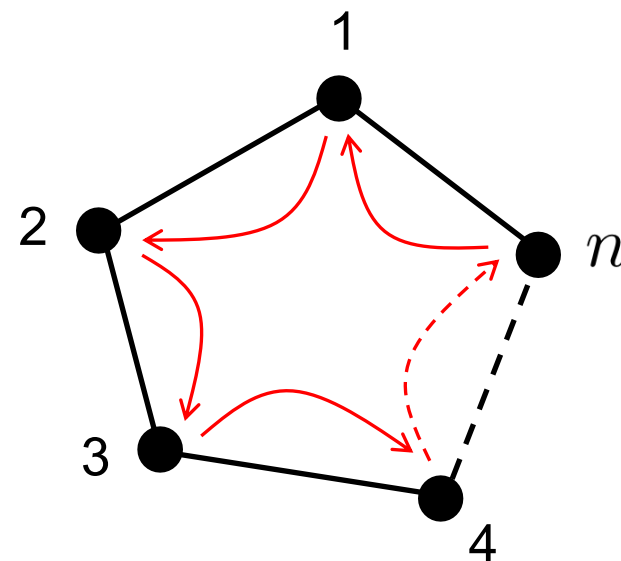
où  $\mathbf{x}(t) = [\mathbf{x}_1^T(t), \dots, \mathbf{x}_n^T(t)]^T$  et

$$\mathbf{L}(\mathcal{G}) = \begin{bmatrix} k & -k & 0 & \dots & 0 & 0 \\ 0 & k & -k & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & k & -k \\ -k & 0 & 0 & \dots & 0 & k \end{bmatrix}$$



Matrice circulante

(la ligne  $i + 1$  est obtenue à partir de la ligne  $i$  par **décalage circulaire**)



Pour plus de détails, voir le livre “*Flocking and Rendezvous in Distributed Robotics*”,  
B.A. Francis, M. Maggiore, SpringerBriefs in Electrical and Computer Engineering, Springer 2016





# Exemple 2: poursuite cyclique ( $n$ -bug problem)

- Trajectoires des robots (en bleu):

