



Electronique

Energie Electrique

Automatique

Master 2 3EA, Parcours RoVA

Master 2 Info, Parcours SDD



Systemes Robotiques Hétérogènes et Coopératifs

UPJV, Département EEA

Fabio MORBIDI

Laboratoire MIS

Équipe Perception Robotique

E-mail : `fabio.morbidi@u-picardie.fr`

CM : Mercredi et Vendredi 13h30-16h30, salle CURI 8

TD, TP : salle TP203/204

AU 2025-2026

Protocole de consensus

Extensions et applications
(graphes non orientés)



Extensions du protocole de consensus

1. Retards temporels uniformes

- On considère la dynamique de consensus *uniformément retardée* sur un graphe non orienté \mathcal{G} , décrite par l'équation :

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

pour quelque $\tau > 0$. Ce protocole retardé parvient au consensus en moyenne si et seulement si le retard satisfait :

$$\tau < \frac{\pi}{2 \lambda_n(\mathbf{L}(\mathcal{G}))}$$

où $\lambda_n(\mathbf{L}(\mathcal{G}))$ est la valeur propre *la plus forte* de la matrice laplacienne du graphe

➔ **Compromis** entre la *vitesse de convergence* et la *tolerance aux retards uniformes* dans les échanges d'information sur le réseau



- “Consensus problems in networks of agents with switching topology and time-delays”, R. Olfati-Saber, R.M. Murray, IEEE Trans. Automat. Contr., vol. 49, n. 9, pp. 1520-1533, 2004

Extensions du protocole de consensus

2. Agents double intégrateur

- On considère la dynamique du *second ordre* suivante pour l'agent i :

$$\ddot{\mathbf{x}}_i = \mathbf{u}_i \quad \text{ou de façon} \quad \begin{bmatrix} \dot{\mathbf{x}}_i \\ \dot{\mathbf{v}}_i \end{bmatrix} = \begin{bmatrix} \mathbf{0}_m & \mathbf{I}_m \\ \mathbf{0}_m & \mathbf{0}_m \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{v}_i \end{bmatrix} + \begin{bmatrix} \mathbf{0}_m \\ \mathbf{I}_m \end{bmatrix} \mathbf{u}_i$$

équivalente
(représentation d'état)

où $\mathbf{x}_i \in \mathbb{R}^m$ et $\mathbf{v}_i \in \mathbb{R}^m$ sont, respectivement, la *position* et la *vitesse* de l'agent i par rapport à un certain repère absolu et $\mathbf{u}_i \in \mathbb{R}^m$ est l'*entrée de commande* (accélération), avec $m \in \{2, 3\}$.

- Inspirés par le problème du rendez-vous, on peut définir l'entrée de commande suivante pour l'agent i :

$$\mathbf{u}_i = \sum_{j \in \mathcal{N}(i)} [(\mathbf{x}_j - \mathbf{x}_i) + \gamma(\mathbf{v}_j - \mathbf{v}_i)], \quad i \in \{1, \dots, n\}$$

où γ est un gain positif



- “On Consensus Algorithms for Double-Integrator Dynamics”, W. Ren, IEEE Trans. Automat. Contr., vol. 53, n. 6, pp. 1503-1509, 2008

Extensions du protocole de consensus

3. Formulation à temps discret

- La **forme itérative** du protocole de consensus en *temps discret* est la suivante, $k \in \{0, 1, 2, \dots\}$:

$$x_i(k+1) = x_i(k) + \epsilon \sum_{j \in \mathcal{N}(i)} (x_j(k) - x_i(k)), \quad i \in \{1, \dots, n\}$$

où $0 < \epsilon < 1/d_{\max}$ est l'incrément et $d_{\max} = \max_i d(v_i)$ est le *degré maximum* du graphe non orienté \mathcal{G}

- On peut écrire de façon compacte les dynamiques collectives des n agents comme suit :

$$\mathbf{x}(k+1) = \mathbf{P} \mathbf{x}(k)$$

où $\mathbf{x} = [x_1, \dots, x_n]^T$ et $\mathbf{P} = \mathbf{I}_n - \epsilon \mathbf{L}$



Extensions du protocole de consensus

3. Formulation à temps discret

- \mathbf{P} est dite **matrice de Perron** du graphe, avec paramètre ϵ
- \mathbf{P} est une **matrice stochastique** (à droite), c'est-à-dire la somme des éléments sur chaque ligne est égale à 1 :

$$\mathbf{P} \mathbf{1} = \mathbf{1}$$

- Les conditions sur le graphe \mathcal{G} pour parvenir au consensus en temps discret, sont *les mêmes* qu'en temps continu
- La *vitesse de convergence* vers l'ensemble de consensus \mathcal{A} est dictée cette fois-ci par :

$$\mu_2 = 1 - \epsilon \lambda_2(\mathbf{L})$$

la *deuxième valeur propre la plus forte* de \mathbf{P}

Pour plus de détails sur le protocole de consensus à temps discret et ses liens avec la théorie des **chaînes de Markov**, voir l'article :

- “*Consensus and Cooperation in Networked Multi-Agent Systems*”, R. Olfati-Saber, J.A. Fax, R.M. Murray, in Proc. IEEE, vol. 95, n. 1, pp. 215-233, 2007



Oskar Perron
(1880 – 1975)

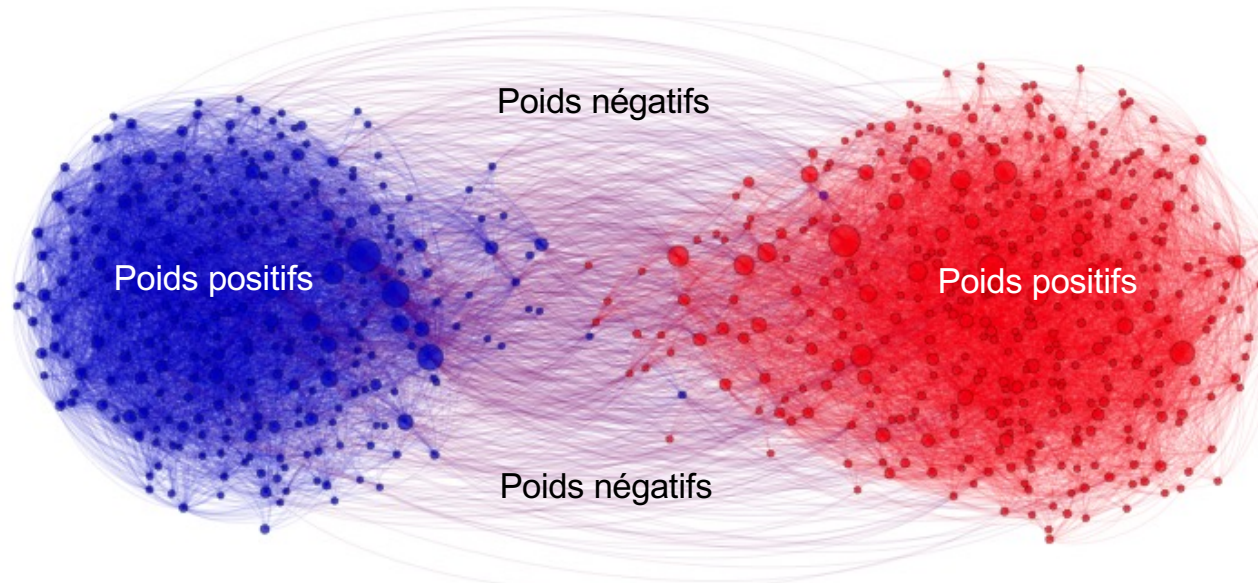


Extensions du protocole de consensus

4. Interactions antagonistes

- On peut modéliser des *interactions antagonistes* (par exemple, amis vs ennemis dans un réseau social), avec des **poids négatifs** w_{ij} dans le graphe de communication
- On parle alors de **graphes signés**

Exemple : Visualisation de blogs politiques aux États-Unis en 2014.
Parti démocrate de Barack Obama (bleu), parti républicain (rouge)



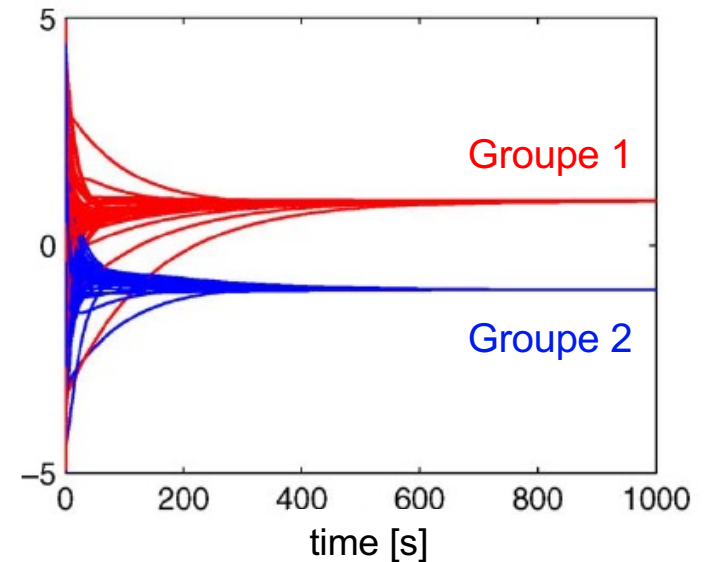
1490 blogs (sommets) et 19025 liens (arêtes) entre les blogs



Extensions du protocole de consensus

4. Interactions antagonistes

- Sur des *graphes signés*, les agents peuvent converger vers la valeur de consensus. Cette valeur est la même pour tous à l'exception du signe : on parle alors de "*consensus biparti*"
- On peut parvenir au consensus biparti si le graphe est *structurellement balancé*. Ça veut dire que tous les cycles dans le graphe sont positifs, à savoir ils contiennent un *nombre pair d'arêtes avec poids négatifs*



Consensus biparti
(par ex. polarisation des opinions
dans deux groups : $n = 1000$)

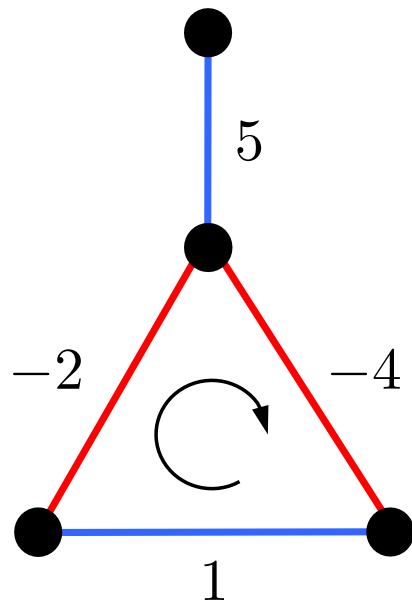
- "Consensus Problems on Networks With Antagonistic Interactions", C. Altafini, IEEE Trans. Automat. Contr., vol. 58, n. 4, pp. 935-946, 2013
- "Predictable Dynamics of Opinion Forming for Networks with Antagonistic Interactions", C. Altafini, G. Lini, IEEE Trans. Automat. Contr., vol. 60, n. 2, pp. 342-357, 2015



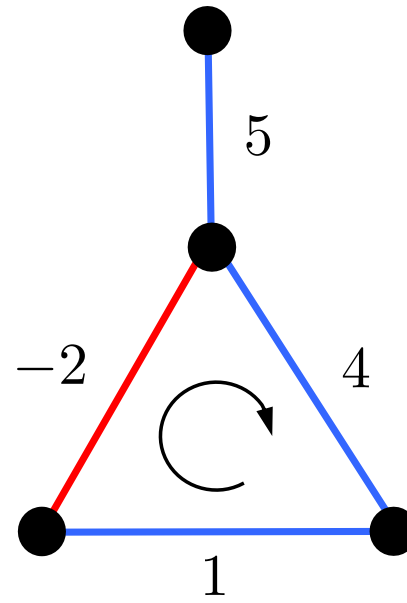
Extensions du protocole de consensus

4. Interactions antagonistes

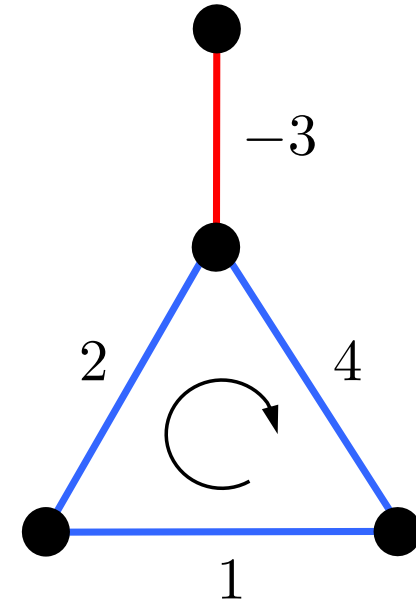
Exemples (graphes signés avec 4 sommets) :



Structurellement balancé



Non structurellement
balancé



Structurellement balancé



Extensions du protocole de consensus

5. Contrôle de formations de robots

- **Problème** : contrôler un groupe de robots de type intégrateur pour obtenir une *formation invariante aux translations*, définie par

- Graphe de formation $\mathcal{G}_f = (V, E_f)$
- Ensemble associé de *positions cibles* $\Xi = \{\xi_1, \xi_2, \dots, \xi_n\}$

- Soit $\mathbf{x}_i \in \mathbb{R}^m$ la *position* du robot i . Le contrôleur de formation doit assurer que pour quelque $\boldsymbol{\tau} \in \mathbb{R}^m$,

$$\mathbf{x}_i = \xi_i + \boldsymbol{\tau}, \quad \forall i \in \{1, \dots, n\}.$$

- Le graphe $\mathcal{G} = (V, E)$, appelé **graphe d'interaction**, est utilisé pour décrire la structure du *réseau de communication* entre les robots
- **Objectif** : contrôler les robots de telle sorte que $\|\mathbf{x}_i(t) - \mathbf{x}_j(t)\|$ converge asymptotiquement vers la distance prédéfinie d_{ij} pour tous les i, j tels que $\{v_i, v_j\} \in E_f$



Extensions du protocole de consensus

5. Contrôle de formations de robots

Théorème :

Soit $\mathcal{G}_f = (V, E_f)$ le graphe *connexe* de formation, Ξ l'ensemble de positions cibles et $\mathcal{N}_f(i)$ le voisinage du robot i sur \mathcal{G}_f .

Si le graphe d'interaction $\mathcal{G} = (V, E)$ est tel que $E_f \subseteq E$, l'entrée de commande :

$$\mathbf{u}_i(t) = \sum_{j \in \mathcal{N}_f(i)} (\mathbf{x}_j(t) - \mathbf{x}_i(t)) - (\xi_j - \xi_i), \quad i \in \{1, \dots, n\}$$

permet aux robots de converger asymptotiquement vers un *déplacement constant* par rapport aux positions cibles, c'est-à-dire, pour tout i nous avons que :

$$\lim_{t \rightarrow \infty} \mathbf{x}_i(t) - \xi_i = \boldsymbol{\tau}.$$

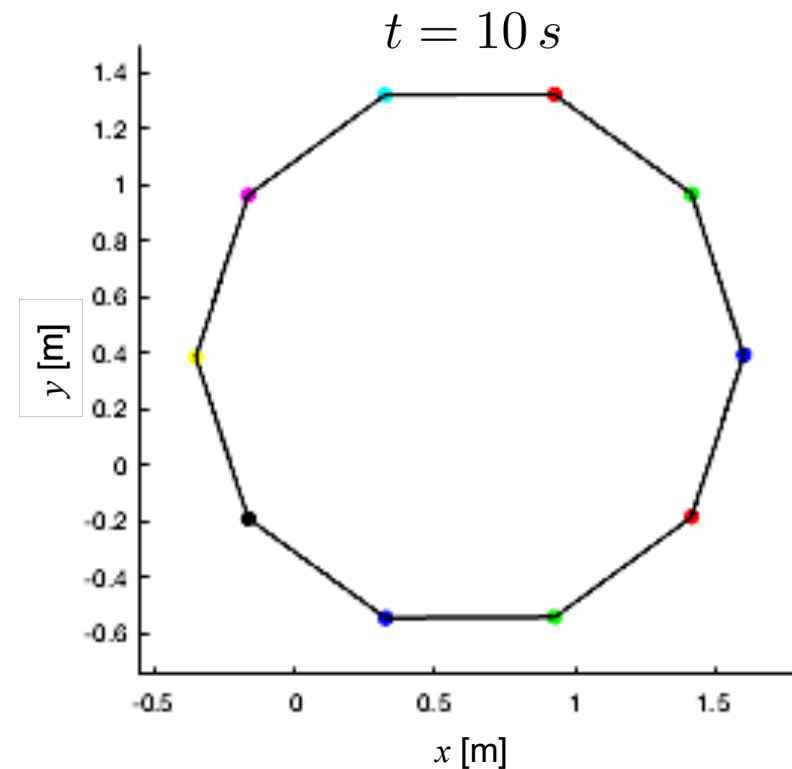
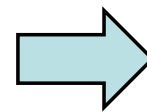
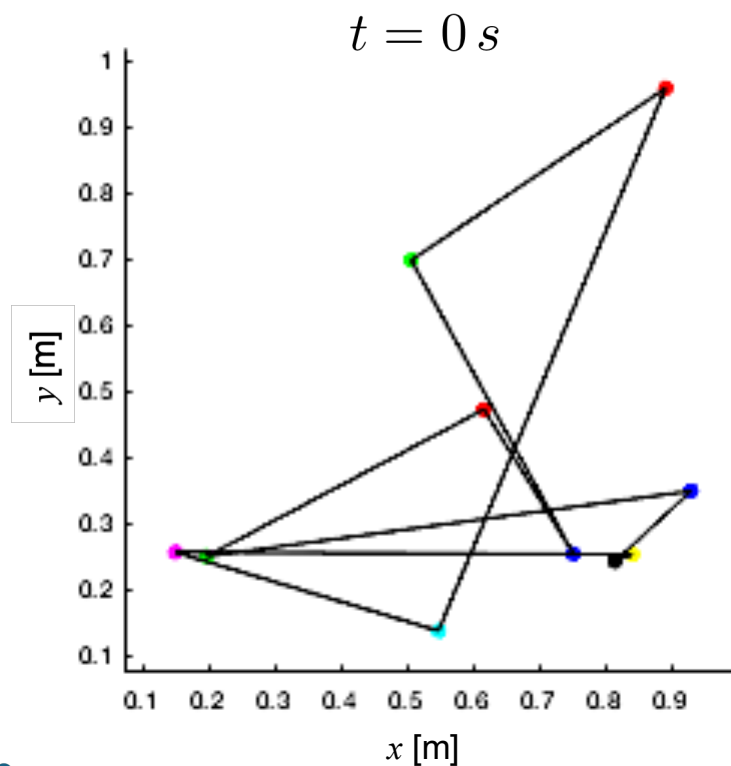
- “*Formation Control: Approaches for Distributed Agents*”, H.-S. Ahn, vol. 205, Studies in Systems, Decision and Control, Springer, 2020



Extensions du protocole de consensus

Exemple : Formation dans l'espace 2D ($m = 2$)

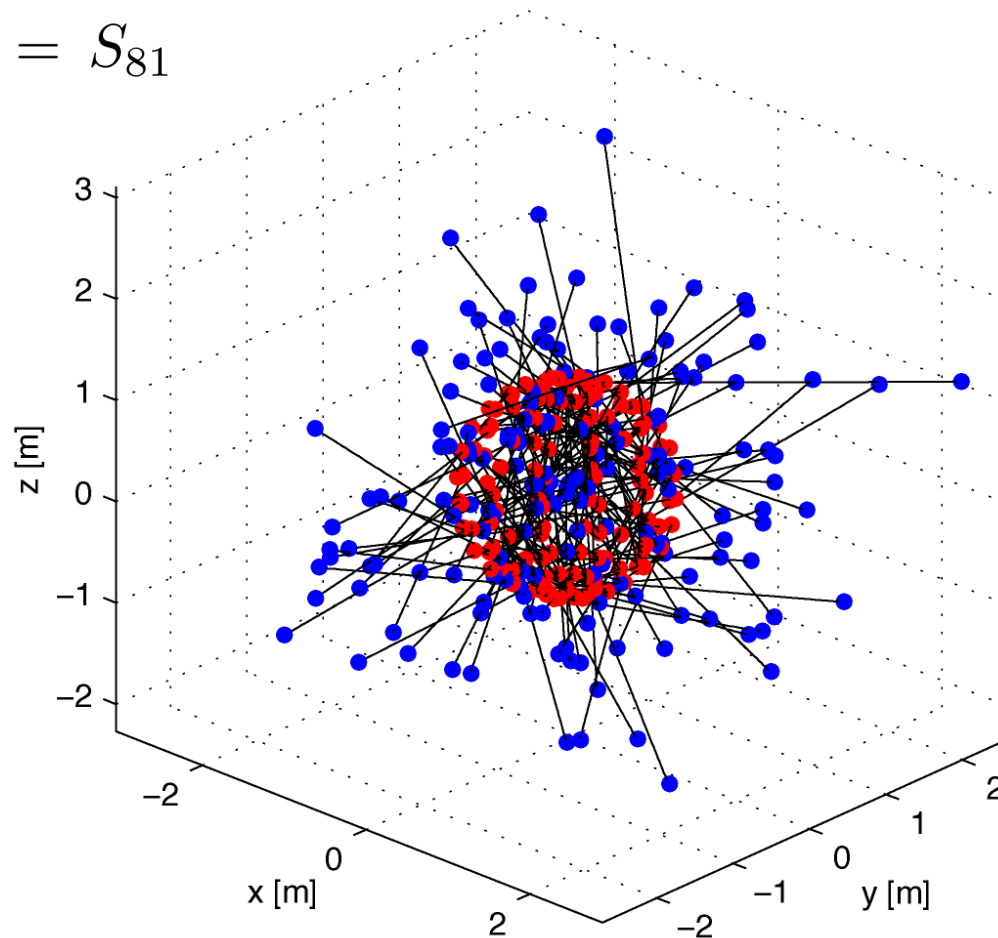
- Ensemble de positions cibles Ξ : 10 points équidistants sur le *circle unitaire*
- Pour plus de simplicité, on a choisi $\mathcal{G}_f = \mathcal{G}$. Dans les figures : $\mathcal{G}_f = C_{10}$



Extensions du protocole de consensus

Exemple : Formation dans l'espace 3D ($m = 3$)

- Ensemble de positions cibles Ξ : 81 points espacés de façon régulière sur la surface de la *sphère unitaire*
- $\mathcal{G}_f = \mathcal{G} = S_{81}$



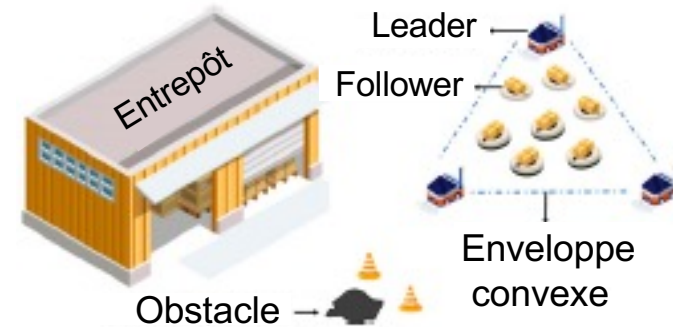
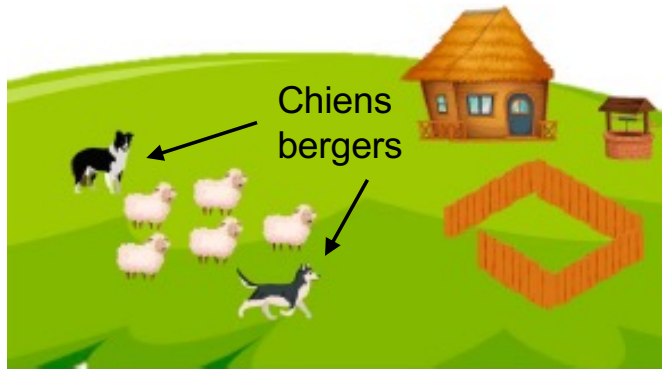
Bleu : positions initiales

Rouge : positions finales



Extensions du protocole de consensus

6. Contrôle de confinement ("Containment control")

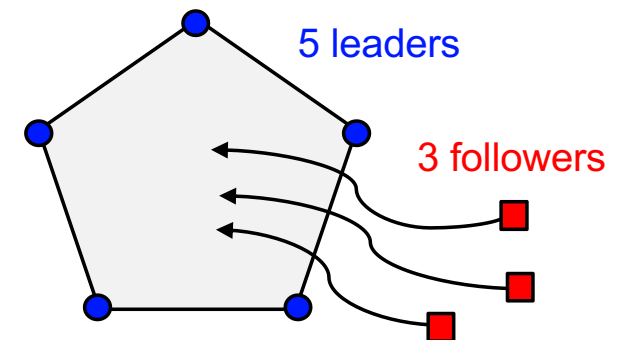


m leaders (ils n'ont pas de voisins)
 $n - m$ followers (ils ont des voisins)
 On fait l'hypothèse que $m < n$

Les robots communiquent sur le digraphe $\mathcal{D} = (V, E)$ avec $V = V_L \cup V_F$

▪ Leaders stationnaires :

Objectif de la commande : garantir la convergence asymptotique des followers (à partir de toute condition initiale) vers l'*enveloppe convexe* définie par les leaders



- "Laplacian Sheep: A Hybrid, Stop-Go Policy for Leader-Based Containment Control", G. Ferrari-Trecate, M. Egerstedt, A. Buffa, M. Ji, Hybrid Systems: Comp. Control, pp. 212-226, 2006

Extensions du protocole de consensus

Leaders stationnaires :

Pour n robots de type intégrateur ($\dot{x}_i = u_i$), on choisit l'entrée de commande :

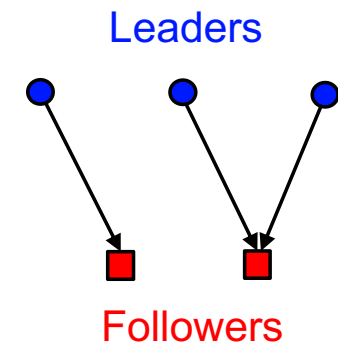
$$u_i(t) = \begin{cases} 0 & \text{si } i \in V_L \\ \sum_{j \in V} a_{ij} (x_j(t) - x_i(t)) & \text{si } i \in V_F \end{cases}$$

où a_{ij} est l'élément (i, j) de la matrice d'adjacence \mathbf{A} du digraphe \mathcal{D}

Remarque : $x_j, j \in V_L$ est *constant*, car les leaders sont statiques

Théorème :

Les followers convergent vers l'enveloppe convexe définie par les leaders à partir de toute condition initiale $x_i(0), i \in V_F$ si et seulement si le digraphe \mathcal{D} a un *arbre orienté enraciné uni*, c'est-à-dire, pour chaque follower existe au moins un leader qui a un chemin direct vers ce follower



- "Containment Control with Multiple Stationary or Dynamic Leaders Under a Directed Interaction Graph", Y. Cao, W. Ren, in Proc. 48th IEEE Conf. Dec. Contr., pp. 3014-3019, 2009

Extensions du protocole de consensus

- **Leaders dynamiques :**

Objectif de la commande : garantir la convergence asymptotique des followers (à partir de toute condition initiale) vers l'*enveloppe convexe dynamique* définie par les leaders

- Pour n robots de type intégrateur, on choisit l'entrée de commande :

$$u_i(t) = \begin{cases} v_i(t) & \text{si } i \in V_L \\ \sum_{j \in V} a_{ij} (x_j(t) - x_i(t)) + \beta \operatorname{sgn} \left[\sum_{j \in V} a_{ij} (x_j(t) - x_i(t)) \right] & \text{si } i \in V_F \end{cases}$$

où $v_i(t)$ est la *vitesse variable dans le temps* du leader $i \in V_L$,
 $\operatorname{sgn}(\cdot)$ est la fonction signe et $\beta > 0$.

Remarque :

Dans la loi de commande précédente, les followers n'ont pas besoin de *mesures de vitesse*



- "Containment Control with Multiple Stationary or Dynamic Leaders Under a Directed Interaction Graph", Y. Cao, W. Ren, in Proc. 48th IEEE Conf. Dec. Contr., pp. 3014-3019, 2009

Extensions du protocole de consensus

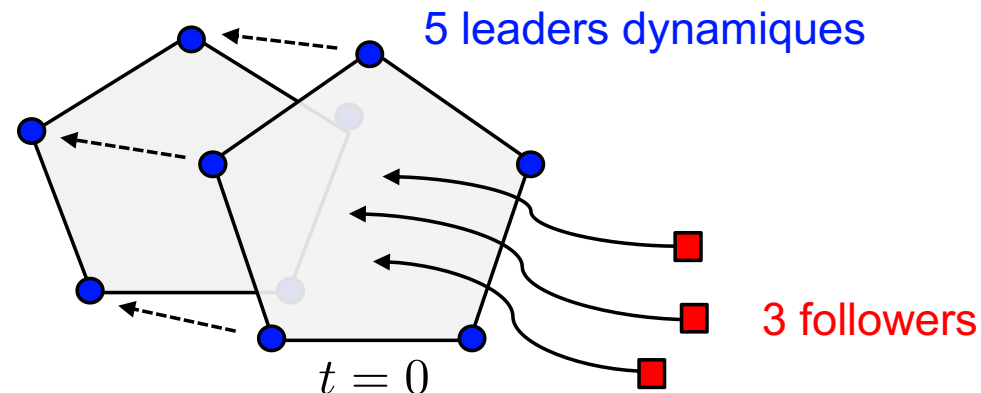
Leaders dynamiques :

Théorème :

Supposons que

$$\beta > \sup_{i \in V_L} |v_i(t)|.$$

Alors les followers convergent vers l'enveloppe convexe dynamique définie par les leaders à partir de toute condition initiale $x_i(0)$, $i \in V_F$, si et seulement si le digraphe \mathcal{D} a un *arbre orienté enraciné uni*.

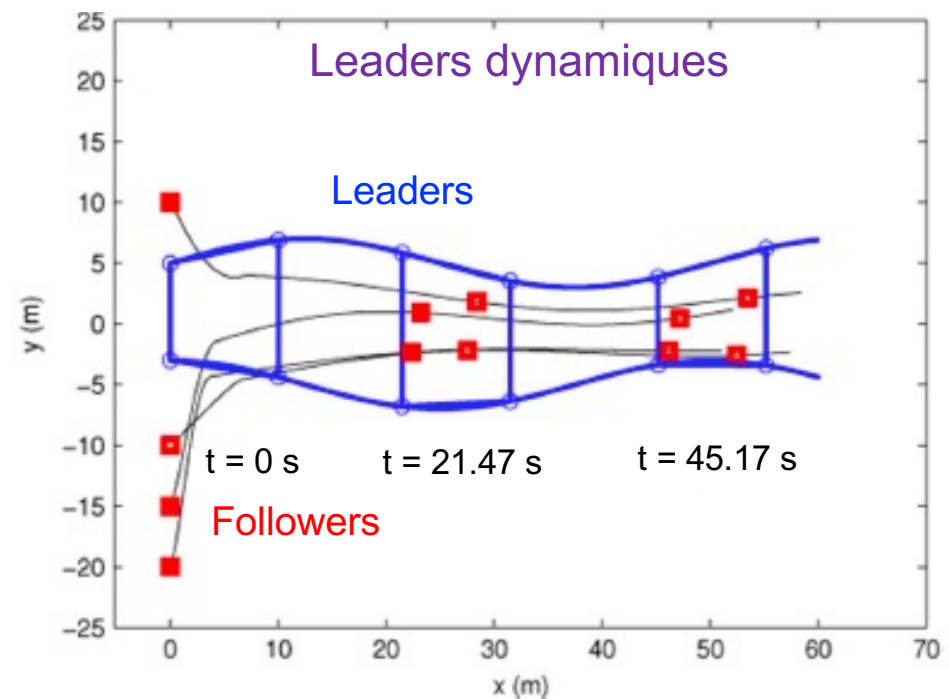
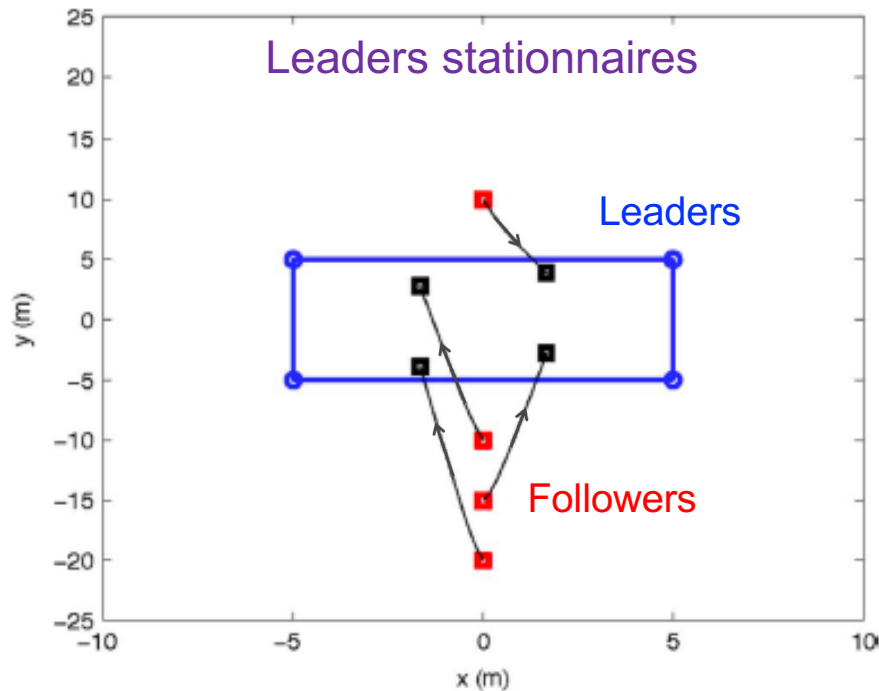


- "Containment Control with Multiple Stationary or Dynamic Leaders Under a Directed Interaction Graph", Y. Cao, W. Ren, in Proc. 48th IEEE Conf. Dec. Contr., pp. 3014-3019, 2009

Extensions du protocole de consensus

Simulations :

- Évolution des robots (4 leaders et 4 followers) dans l'espace 2D
- Modèle dynamique des robots : **double intégrateur**



- “Distributed containment control for multiple autonomous vehicles with double-integrator dynamics: Algorithms and experiments”, Y. Cao, D. Stuart, W. Ren, Z. Meng, IEEE Trans. Control Syst. Technol., vol. 19, n. 4, pp. 929-938, 2011



Extensions du protocole de consensus

7. Protocoles de consensus *non linéaires* (I)

- Le *modèle de Kuramoto* pour n oscillateurs couplés, en interaction sur un réseau \mathcal{G} est défini par :

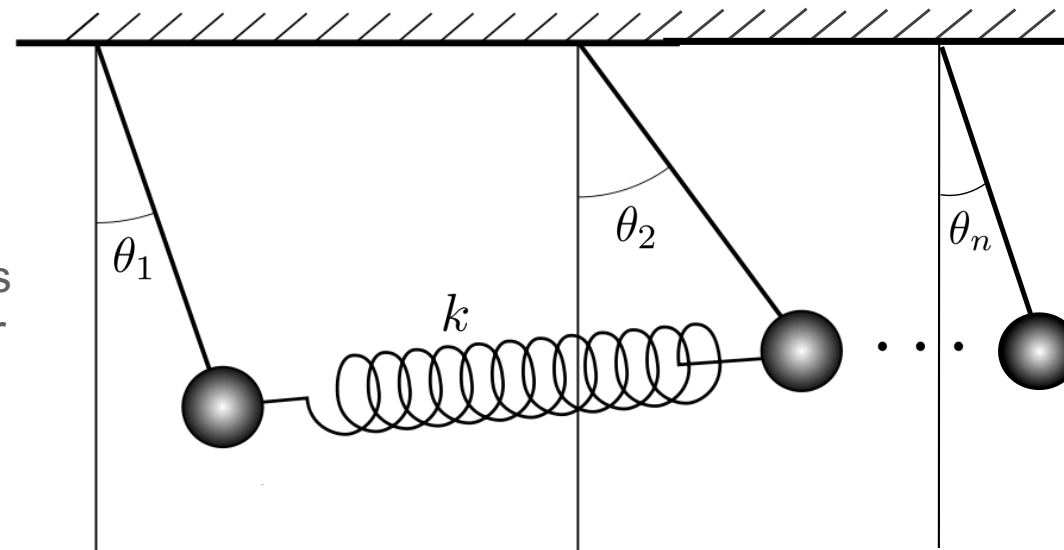
$$\dot{\theta}_i(t) = k \sum_{j \in \mathcal{N}(i)} \sin(\theta_j(t) - \theta_i(t)), \quad i \in \{1, \dots, n\},$$

où θ_i est la *phase* de l'oscillateur i et $k > 0$ est la *force de couplage* entre les oscillateurs



Yoshiaki Kuramoto
(1940 -)

Analogie mécanique
du modèle de Kuramoto :
pendules inversés couplés
(k = constante de raideur
des ressorts)



Extensions du protocole de consensus

- On peut réécrire sous forme matricielle les lois d'interaction non linéaires entre les oscillateurs, comme :

$$\dot{\boldsymbol{\theta}}(t) = -k \mathbf{D}(\mathcal{G}) \sin(\mathbf{D}^T(\mathcal{G}) \boldsymbol{\theta}(t)),$$

où $\boldsymbol{\theta}(t) = [\theta_1(t), \dots, \theta_n(t)]^T$, $\mathbf{D}(\mathcal{G})$ est la matrice d'incidence du graphe \mathcal{G} et on définit :

$$\sin \boldsymbol{\theta} \triangleq [\sin \theta_1, \sin \theta_2, \dots, \sin \theta_n]^T.$$

- On peut montrer (par exemple, avec des outils de *théorie de la passivité*) que pour un graphe *connexe* \mathcal{G} , pour tout $k > 0$ et presque pour toute condition initiale $\boldsymbol{\theta}(0)$, le mouvement des oscillateurs sera **synchronisé**. En outre, la vitesse de convergence vers l'état de synchronisation est dictée par la valeur de Fiedler $\lambda_2(\mathbf{L}(\mathcal{G}))$

Pour plus de détails, voir :

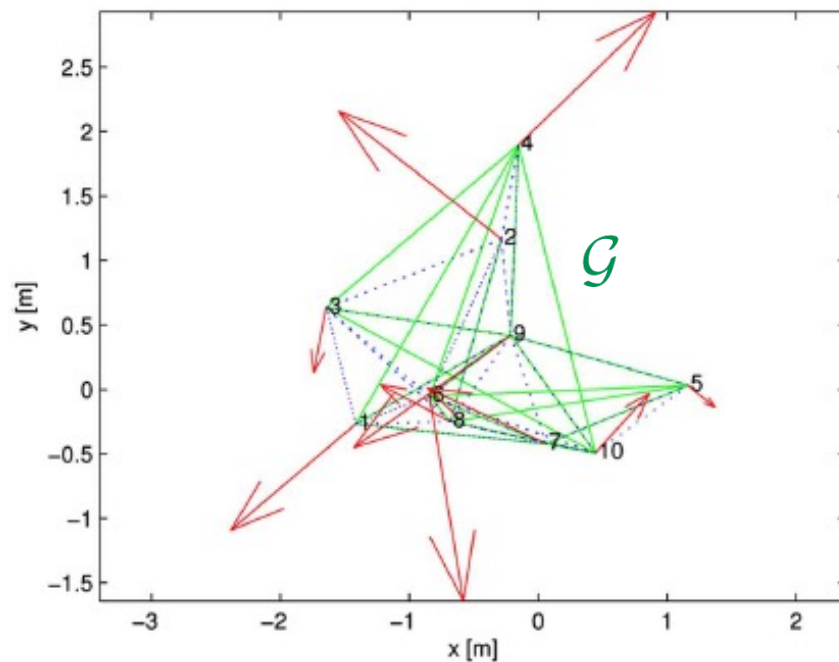
- “On the Stability of the Kuramoto Model of Coupled Nonlinear Oscillators”, A. Jadbabaie, N. Motee, M. Barahona, in Proc. American Control Conf., pp. 4296-4301, 2004
- “From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators”, S.H. Strogatz, Physica D: Nonlinear Phenomena, vol. 143, n. 1, pp. 1-20, 2000
- “Stability conditions for cluster synchronization in networks of heterogeneous Kuramoto oscillators”, T. Menara, G. Baggio, D.S. Bassett, F. Pasqualetti, IEEE Trans. Contr. Netw. Syst. vol. 7, n. 1, pp. 302-314, 2019



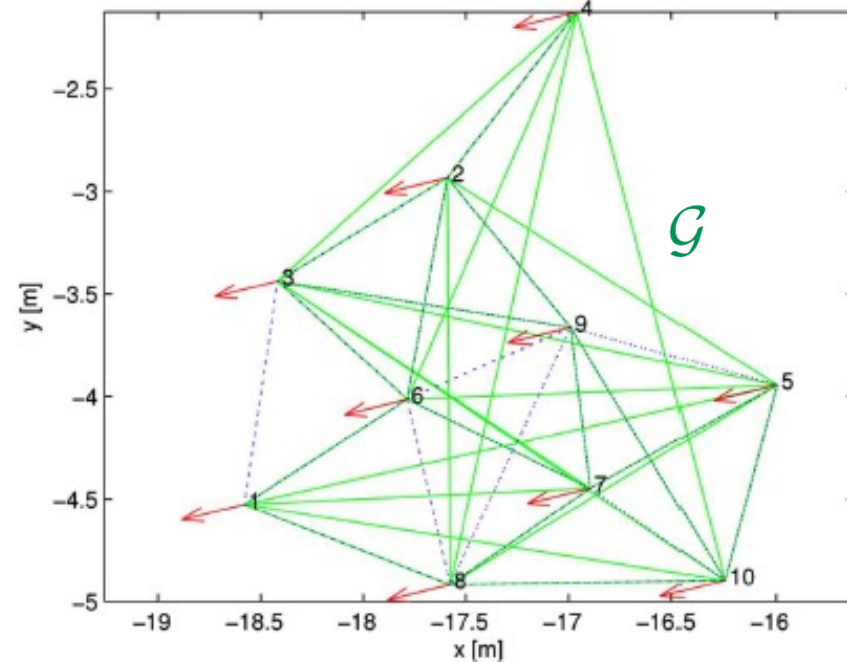
Extensions du protocole de consensus

- Avec des robots de type **unicycle**, on peut utiliser ce mécanisme de synchronisation pour le **flocking**, à savoir pour aligner leurs vecteurs de vitesse

Configuration initiale



Flocking (vecteurs de vitesse alignés : en rouge)



Pour plus de détails, voir :

- “*Distributed Geodesic Control Laws for Flocking of Nonholonomic Agents*”, N. Moshtagh, A. Jadbabaie, IEEE Trans. Automat. Contr., vol. 52, n. 4, pp. 681-686, 2007
- “*Stabilization of Planar Collective Motion With Limited Communication*” R. Sepulchre, D.A. Paley, N.E. Leonard, IEEE Trans. Automat. Contr., vol. 53, n. 3, pp. 706-719, 2008



Extensions du protocole de consensus

7. Protocoles de consensus *non linéaires* (II)

- La **moyenne pondérée d'ordre r** de n nombres réels positifs x_1, \dots, x_n est définie comme :

$$M_r(x_1, \dots, x_n) = \left(\sum_{i=1}^n k_i x_i^r \right)^{1/r}$$

où les poids positifs k_i sont tels que $k_1 + k_2 + \dots + k_n = 1$.

- Selon la valeur de r , nous avons :

$M_{-\infty}$ Minimum

M_{-1} Moyenne harmonique

M_0 Moyenne géométrique

M_1 **Moyenne arithmétique pondérée**: si $k_1 = \dots = k_n = \frac{1}{n}$, $M_1 = \frac{1}{n} \sum_{i=1}^n x_i$

M_2 Moyenne quadratique (RMS, "Root Mean Square")

M_{∞} Maximum



Extensions du protocole de consensus

Exemple :

Soient

$$x_1 = 1, \quad x_2 = 3, \quad x_3 = 6, \quad x_4 = \frac{1}{2}, \quad x_5 = 5$$

avec poids :

$$k_1 = k_2 = \dots = k_5 = \frac{1}{5}$$

Nous avons que :

$$M_{-\infty}(x_1, \dots, x_5) = \lim_{r \rightarrow -\infty} M_r(x_1, \dots, x_5) = \min\{x_1, \dots, x_5\} = \frac{1}{2}$$

$$M_{-1}(x_1, \dots, x_5) = \frac{1}{k_1 x_1^{-1} + \dots + k_5 x_5^{-1}} = 1.3514 \quad (\text{M. harmonique})$$

$$M_0(x_1, \dots, x_5) = x_1^{k_1} x_2^{k_2} \dots x_5^{k_5} = 2.1411 \quad (\text{M. géométrique})$$



Extensions du protocole de consensus

Exemple :

Soient

$$x_1 = 1, \quad x_2 = 3, \quad x_3 = 6, \quad x_4 = \frac{1}{2}, \quad x_5 = 5$$

avec poids :

$$k_1 = k_2 = \dots = k_5 = \frac{1}{5}$$

Nous avons que :

$$M_1(x_1, \dots, x_5) = \frac{1}{5} (x_1 + x_2 + \dots + x_5) = 3.1 \quad (\text{M. arithmétique})$$

$$M_2(x_1, \dots, x_5) = \sqrt{k_1 x_1^2 + k_2 x_2^2 + \dots + k_5 x_5^2} = 3.7749 \quad (\text{M. quadratique})$$

$$M_\infty(x_1, \dots, x_5) = \lim_{r \rightarrow \infty} M_r(x_1, \dots, x_5) = \max\{x_1, \dots, x_5\} = 6$$



Extensions du protocole de consensus

7. Protocoles de consensus non linéaires (II)

- On peut définir le **protocole de consensus à moyenne pondérée d'ordre r** sur le graphe \mathcal{G} comme suit :

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

Remarque : pour $r = 1$ et $k_1 = \dots = k_n = 1/n$, nous retrouvons le *protocole de consensus classique* (à une constante multiplicative près)

Pour plus de détails sur les propriétés de convergence de ce protocole non linéaire, voir :

- “*Non-linear protocols for optimal distributed consensus in networks of dynamic agents*”, D. Bauso, L. Giarré, R. Pesenti, Syst. Control Lett., vol. 55, n. 11, pp. 918-928, 2006
- “*Distributed algorithms for reaching consensus on general functions*”, J. Cortés, Automatica, vol. 44, n. 3, pp. 726-737, 2008



Extensions du protocole de consensus

8. Potentiel laplacien

- Le **potentiel laplacien** associé à un graphe non orienté \mathcal{G} est défini par :

$$\Phi_{\mathcal{G}}(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{L} \mathbf{x}$$

Proposition :

Le potentiel laplacien d'un graphe \mathcal{G} est *non négatif* et il satisfait l'identité :

$$\mathbf{x}^T \mathbf{L} \mathbf{x} = \sum_{(i,j) \in E^o} (x_i - x_j)^2$$

où E^o est l'ensemble des arêtes du digraphe \mathcal{G}^o obtenu en associant une orientation arbitraire aux arêtes de \mathcal{G} . En outre, pour un *graphe connexe*

$$\Phi_{\mathcal{G}}(\mathbf{x}) = 0$$

si et seulement si $x_i = x_j, \forall i, j$.



Extensions du protocole de consensus

8. Potentiel laplacien

- On peut observer que :

$$\dot{\mathbf{x}} = -\mathbf{L} \mathbf{x} = -\nabla \Phi_{\mathcal{G}}(\mathbf{x})$$

donc on peut interpréter le vecteur $\mathbf{x}(t)$ comme la solution d'un *système gradient* (“gradient flow”) associé au potentiel laplacien $\Phi_{\mathcal{G}}(\mathbf{x})$

Pour plus de détails et extensions, voir :

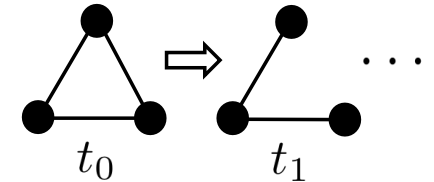
- “*Consensus problems in networks of agents with switching topology and time-delays*”, R. Olfati-Saber, R.M. Murray, IEEE Trans. Automat. Contr., vol. 49, n. 9, pp. 1520-1533, 2004
- “*Finite-time convergent gradient flows with applications to network consensus*”, J. Cortés, Automatica, vol. 42, n. 11, pp. 1993-2000, 2006



Extensions du protocole de consensus

Autres extensions dans la littérature :

- Graphes de communication *variables dans le temps* : $\mathcal{G} = \mathcal{G}(t)$
- Communication *asynchrone* (self- ou event-triggered, gossip, etc.)
- Communication *quantifiée* ou *ensemble fini* de valeurs admissibles
- Communication *imparfaite* : défaillance des arêtes et/ou bruit dans le canal de comm.
- Graphes *aléatoires* (par ex. modèle de Erdős-Rényi)
- Consensus *dynamique* (“dynamic average consensus”) : moyenne d’entrées variables



- “Consensus seeking in multiagent systems under dynamically changing interaction topologies”, W. Ren, R.W. Beard, IEEE Trans. Autom. Control, vol. 50, n. 5, pp. 655-661, 2005
- “Distributed event-triggered control for multi-agent systems”, D.V. Dimarogonas, E. Frazzoli, K.H. Johansson, IEEE Trans. Autom. Control, vol. 57, n. 5, pp. 1291-1297, 2011
- “Average consensus on networks with quantized communication”, P. Frasca, R. Carli, F. Fagnani, S. Zampieri, Int. J. Robust Nonlinear Control, vol. 19, n. 16, pp. 1787-1816, 2009
- “Consensus networks over finite fields”, F. Pasqualetti, D. Borra, F. Bullo, Automatica, vol. 50, n. 2, pp. 349-358, 2014
- “Distributed consensus algorithms in sensor networks with imperfect communication: link failures and channel noise”, S. Kar, J. Moura, IEEE Trans. Signal Proces., vol. 57, n. 1, pp. 355-369, 2009
- “A necessary and sufficient condition for consensus over random networks”, A. Tahbaz-Salehi, A. Jadbabaie, IEEE Trans. Autom. Control, vol. 53, n. 3, pp. 791-795, 2008
- “Tutorial on Dynamic Average Consensus: The Problem, its Applications, and the Algorithms”, S.S. Kia, B. Van Scoy, J. Cortés, R.A. Freeman, K.M. Lynch, S. Martinez, IEEE Contr. Syst. Mag., vol. 39, n. 3, pp. 40-72, 2019