

情報指向の agent を用いた 可動センサネットワークによる, 目標追従問題

藤田研究室

古関 裕道

2006/7/10

1 はじめに

本レポートでは, Olfati-Saber の文献 [1] に基づき, Distributed Kalman Filter(以下 DKF) を導入した可動式センサネットワークによる目標追従問題を扱う. 可動センサによる目標 (ターゲット) の追従を行う時, 各 agent が自ら得られる情報をより改善するように動くなら, flocking が達成されることが知られている. また agent が flock の状態にあるならば, DKF を用いた cooperative filtering を用いて追従性能を向上することが可能になる. 以上の事実により, 集団として得られる情報を最大値にする様に各 agent が協調して動く時, その動きは flocking algorithm を用いることによって達成されることを証明できる.

本レポートでは, この証明の第一段階として, 各 agent の algorithm について扱う. まず情報指向の複数の agent によって目標追従を行おうとすると, flocking が達成されることを示す. 次に文献 [2] で紹介されている, 任意の次元における目標追従を意識した flocking algorithm を紹介し, その algorithm を導入するメリットについて示していく.

2 目標追従と Flocking

文献 [2] では, ある m 次元の空間内を移動する複数台の agent が flocking を達成するための制御則についていくつか提案している. この節では, センサネットワークによる目標追従問題を想定して, この文献で提案されている制御則の一つを紹介する.

2.1 問題設定

各 agent 間の関係について, 以下の仮定が与えられている.

- 各 agent は, 他の agent と通信することのできる有限距離 (finite interaction range) r を持ち, その距離内に含まれる場合のみ互いの存在を認識できる.
- 初期状態では, 各 agent 間の距離が r 以上ある. つまり初期状態では通信できない.
- 各 agent は, 自分以外の agent もターゲットに追従していることを知らない.
- ターゲットに関するセンサデータはお互いに共有しない
- すべての agent は, ターゲットとの距離 $\rho(\geq \rho_0)$ が縮まる程精度が高くなり, その動きには影響されなくなるというセンサのモデルを共有している. (ここで, $\rho_0(\ll r/2)$ は, agent の衝突安全距離である)

以上の仮定より, 各 agent は目標の動きをより正確に把握するために, ターゲットに対して近づこうとする.

この, 共通の目標に対してそれぞれの情報精度を高めようとする結果として, agent は目標周辺で rendezvous のような動きをすることになる. 一方で実際にこの動きをさせようとする, 各 agent はターゲットに近づこうとする動きをしつつも, 互いの衝突を避ける動きをする必要が生じる. 文献 [2] には, この各 agent における衝突回避と rendezvous の動きの組み合わせによって flocking が達成されることが書かれている.

2.2 flocking のための制御則

この節では、まず flocking algorithm を導出する上で必要な事項を確認する。次に、文献 [2] で紹介されている 2 種類の flocking algorithm を紹介し、ターゲットに関する推定値を用いることのメリットを示していく。

2.2.1 Topology of Flocks

n 台の agent のダイナミクスは、

$$\dot{q}_i = p_i \quad (1)$$

$$\dot{p}_i = f_i \quad (2)$$

と表される。ここで、 q_i, p_i, f_i は (絶対座標系での) agent i の位置、速度、制御入力 (力) とする。agent の近傍を、次のように定義する。

$$N_i = \{j \in V : \|q_j - q_i\| < r\} \quad (3)$$

ここで、 r は以下の仮定で定めるような agent の通信可能距離で、 N_i は通信距離にある agent の集合である。

2.2.2 Geometry of Flocks

実際に flock が現れたかを視覚的に捕らえるために、flock 内の agent が lattice-type (格子状) の構造になるように設計する。各 agent はグラフ構造 $G(q)$ にしたがって、近傍にある他の agent と等距離で接続されるようにする。この距離を保つために、agent は次の数式で表される拘束を受ける。

$$\|q_j - q_i\| = d, \quad \forall j \in N_i(q) \quad (4)$$

Definition 1 (α -lattice) 上式の拘束を満たす状態 q の集合を α -lattice という。ここで $d, \kappa (= r/d)$ をそれぞれ lattice の scale, ratio という。

つぎに、式 (4) にきわめて近い状態 q' の集合を表現するために、次の不等式を用いる。

$$-\delta \leq \|q_j - q_i\| - d \leq \delta \quad (5)$$

この不等式を満たす状態 q' の集合を、quasi α -lattice と呼ぶ。

2.2.3 σ -Norms and Smooth Adjacency Elements

flock がなめらかに収束するような potential 関数と近傍の隣接行列を作るために、次のような σ ノルムを定義する。ただし、 σ -norm は通常使われる 2-norm や ∞ -norm の類ではない。m 次ベクトルをある負ではない実数へ写像する関数 ($\mathbb{R}_{\geq 0}^m \rightarrow \mathbb{R}_{\geq 0}$) として定義する。

$$\|z\|_\sigma = \frac{1}{\epsilon} [\sqrt{1 + \epsilon \|z\|^2} - 1] \quad (6)$$

ここで、 $\epsilon > 0$ とし $\|z\|_\sigma$ の勾配 $\nabla \|z\|_\sigma$ は

$$\sigma_\epsilon(z) = \frac{z}{\sqrt{1 + \epsilon \|z\|^2}} = \frac{z}{1 + \epsilon \|z\|_\sigma} \quad (7)$$

で与えられる。この新しいノルムを定義した理由は、 $\|z\|_\sigma$ は $\|z\|$ とは違い $z = 0$ で微分可能だからである。この特徴は後で滑らかに収束する potential 関数を作成する際に用いる。 σ -norm とその勾配の例を、それぞれ Fig.1, Fig.2 に示す。

次に 0 から 1 の間を滑らかに変化する、スカラー関数である bump function $\rho_h(z)$ を導入する。この bump function は滑らかに収束する potential 関数と隣接行列を作成する際に用いる。ここでは、次式であらわさえる bump function を用いる。

$$\rho_h(z) = \begin{cases} 1, & z \in [0, h) \\ \frac{1}{2} [1 + \cos(\pi \frac{z-h}{1-h})], & z \in [h, 1] \\ 0, & otherwise \end{cases} \quad (8)$$

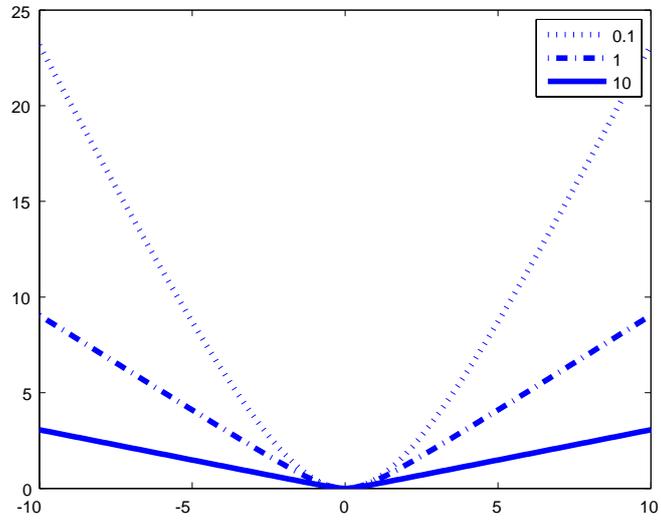


Fig.1: σ -norm $\|z\|_\sigma$

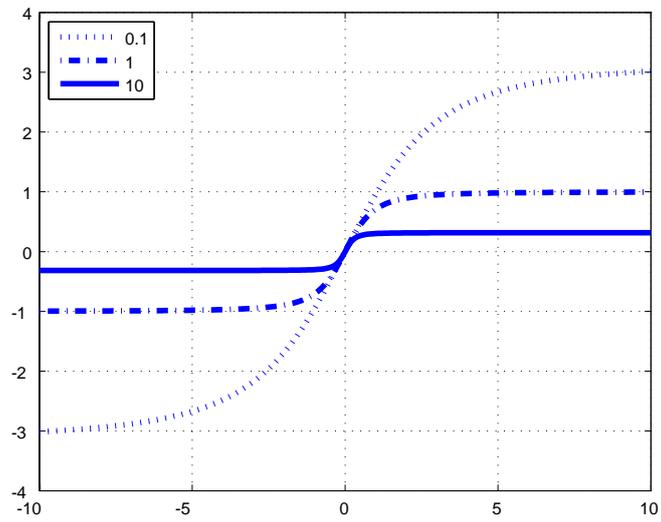


Fig.2: gradient $\sigma_\epsilon(z) = \nabla \|z\|_\sigma$

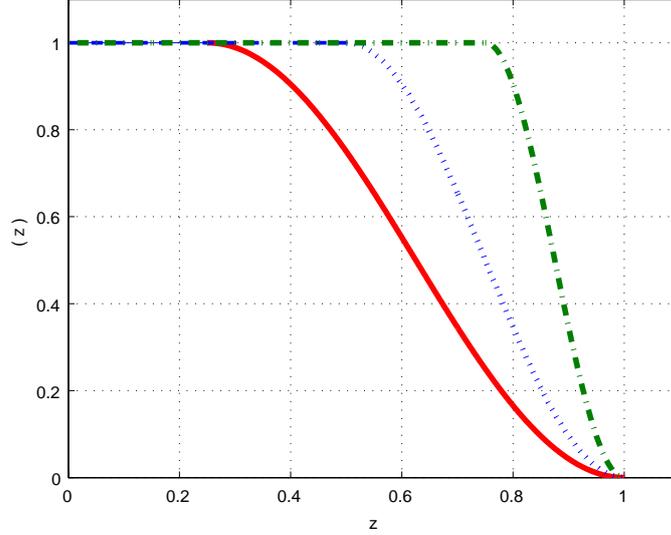


Fig.3: Bump function.(実線 $h=0.25$, 点線 $h=0.5$, 一点鎖線 $h=0.75$)

ここで $h \in (0, 1)$ とする . bump function の例を Fig.3 に示す .

この bump function を用いると , 隣接行列 $A(q)$ の各要素を次のように定義することが出来る .

$$a_{ij}(q) = \rho_h(\|q_j - q_i\|_\sigma / r_\alpha) \in [0, 1], \quad j \neq i \quad (9)$$

ここで $r_\alpha = \|r\|_\sigma$ とし , 任意の a, q に関して $a_{ii}(q) = 0$ とする . $h = 1$ とすると , $\rho_h(z)$ は $[0, 1]$ で 1, それ以外で 0 となる標示関数となる . 標示関数は近傍に関する隣接行列を 0 と 1 とで表現する際に用いる .

2.2.4 Collective Potential Functions

agent の集合の potential 関数 $V(q)$ は , 負ではない関数 $V : \mathbb{R}^{mn} \rightarrow \mathbb{R}_{\geq 0}$ であり , 式 (4) で表される拘束の解が $V(q)$ の停留点となる特徴を持つ . このレポート内では , 二方向のスカラー potential による , エネルギー偏差の滑らかな関数を potential 関数とする . この特徴は後で flocking algorithm に拡張する際に用います .

$\psi(z) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ を , r を有限範囲とし , $z = d$ を最小点に持つ二方向の attractive/repulsive potential とする . この時 ,

$$\varphi(q) = \frac{1}{2} \sum_i \sum_{j \neq i} \psi(\|q_j - q_i\|) \quad (10)$$

は , $q_j = q_i$ を満たす特異点で微分不可能な potential 関数となる . この微分不可能な点があるという問題に対処するため , 式 (4) を σ -norm で書き直した次の拘束を導入する .

$$\|q_j - q_i\|_\sigma = d_\alpha, \quad \forall j \in N_i(q) \quad (11)$$

ここで , $d_\alpha = \|d\|_\sigma$ である . この拘束によって , 次式で表されるような滑らかな potential 関数が得られる .

$$V(q) = \frac{1}{2} \sum_i \sum_{j \neq i} \psi_\alpha(\|q_j - q_i\|_\sigma) \quad (12)$$

ここで , $\psi_\alpha(z)$ は式 (14) で表され , $r_\alpha = \|r\|_\sigma$ を有限範囲とし , $z = d_\alpha$ を最小点に持つ二方向の attractive/repulsive potential とする .

有限範囲を持つ滑らかな potential 関数は, $z \geq r_\alpha$ のすべての点で 0 になるような action 関数 $\phi_\alpha(q)$ を積分することで得られる. action 関数は次のように定義される.

$$\begin{aligned}\phi_\alpha(q) &= \rho_h(z/r_\alpha)\phi(z - d_\alpha) \\ \phi(z) &= \frac{1}{2}[(a+b)\sigma_1(z+c) + (a-b)]\end{aligned}\tag{13}$$

ここで, $\sigma_1 = z/\sqrt{1+z^2}$ とし, $\phi(z)$ は, パラメータが, $0 < a \leq b, c = |a-b|/\sqrt{4ab}$ で $\phi(0) = 0$ が保証されるような関数である. この関数により, 式 (12) 中の $\psi_\alpha(z)$ は次のように定義される.

$$\psi_\alpha(z) = \int_{d_\alpha}^z \phi_\alpha(s) ds\tag{14}$$

action 関数の例を Fig.4 に示す. (ただし $r = \infty$)

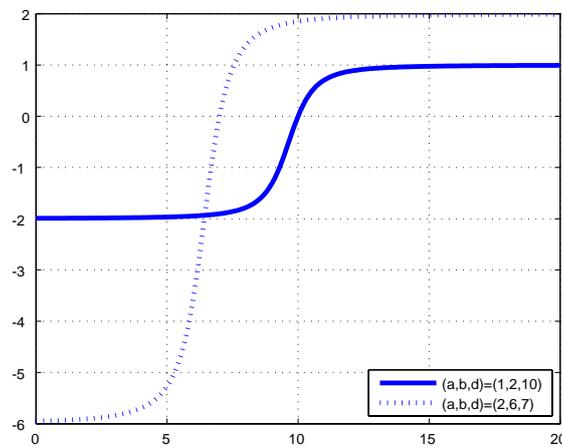


Fig.4: Action function $\phi_\alpha(p)$

文献 [2] で紹介されていた, この potential 関数の例を Fig. 5 に示す.

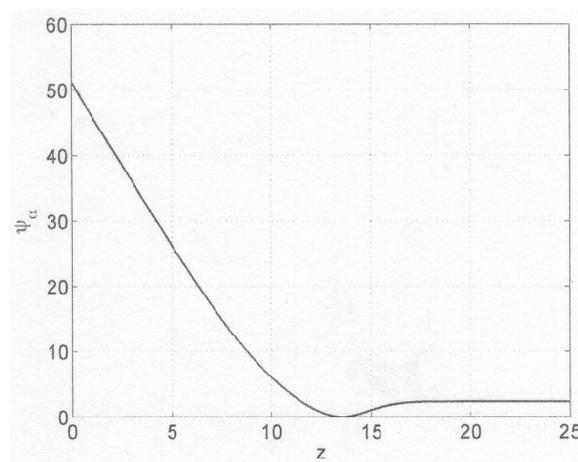


Fig.5: Smooth pairwise potential $\psi_\alpha(z)$

2.2.5 Consensus on Proximity Nets and Graph Laplacian

Graph Laplacian は, flock 内の agent の速度を一致させようとするとき必要になる .n 次元で隣接行列が $A = [a_{ij}]$ を持つグラフ構造 G を考える . G の次数行列は, 対角成分が $\sum_{j=1}^n a_{ij}$ となるような対角行列 $\Delta = \Delta(A)$ である . Graph Laplacian $L = [l_{ij}]$ は $n \times n$ の行列で, 次のように定義される .

$$L = \Delta(A) - A \quad (15)$$

L は常に $\mathbf{1}_n = (1, \dots, 1)^T$ を固有ベクトルとして持ち, このときの固有値は, $\lambda_1 = 0$ となる . Graph Laplacian は, 以下に挙げるような特徴を持つ .

Lemma 1 G を, 負ではない隣接行列 $A = A^T$ を持つ, n 次の無向グラフ (*undirected graph*) とする . この時以下の特徴を持つ .

1. L は, 次の二乗和の特徴を満たす半正定の行列になる .

$$z^T L z = \frac{1}{2} \sum_{(i,j) \in \varepsilon} a_{ij} (z_j - z_i)^2, \quad z \in \mathbb{R}^n \quad (16)$$

2. G が連結 (*connected*) であるならば, $\text{rank}(L) = n - 1$

3. G を連結 (*connected*) とすると,

$$\lambda_2(L) = \min_{z \perp \mathbf{1}_n} \frac{z^T L z}{\|z\|^2} \quad (17)$$

文献 [3] では, Consensus の収束速度を決める protocol が $\lambda_2 > 0$ に強く影響されることが紹介されている . この Consensus の収束速度を決める protocol は, 速度を一致させる項を持つすべての flocking algorithm に登場する . 特に次式で定義されるような m 次元の Graph Laplacian を用いる .

$$\hat{L} = L \otimes I_m \quad (18)$$

ここで, \otimes はクロネッカー積を表す . この多次元 Laplacian は, 次の二乗和の特徴を満たす .

$$z^T \hat{L} z = \frac{1}{2} \sum_{(i,j) \in \varepsilon} a_{ij} \|z_j - z_i\|^2, \quad z \in \mathbb{R}^{mn} \quad (19)$$

ここで, $z = \text{col}(z_1, z_2, \dots, z_n)$ で $z_i \in \mathbb{R}^m$ である .

2.2.6 Flocking Algorithms

ダイナミクスが $\dot{q}_i = u_i$ の集合を α -agent, agent の収束目標 (ターゲット) を表す架空の agent を γ -agent と表すことにする . flock 内の α -agent の第一目標は, 近傍に存在する他の α -agent と α -lattice を形作ることである . 障害物がない空間での flocking の場合, 各 α -agent にへ加えられる入力は, 次の三つの項からなる .

$$u_i = f_i^g + f_i^d + f_i^\gamma \quad (20)$$

ここで, $f_i^g = -\nabla_{q_i} V(q)$ は agent 間の位置勾配に基づく項, f_i^d は速度の consensus を達成するための項, f_i^γ は, ターゲットの位置・速度の推定値に基づき構成されるフィードバック項である . 以下, 文献 [2] で紹介されている二種類の具体的な flocking algorithm を示す .

Algorithm 1 $u_i = u_i^\alpha$ として,

$$u_i^\alpha = \underbrace{\sum_{j \in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma) \mathbf{n}_{ij}}_{\text{gradient-based term}} + \underbrace{\sum_{j \in N_i} a_{ij}(q)(p_j - p_i)}_{\text{consensus term}} \quad (21)$$

ここで, $\mathbf{n}_{ij} = \sigma_\epsilon(q_j - q_i) = \frac{q_j - q_i}{\sqrt{1 + \epsilon \|q_j - q_i\|^2}}$ は, q_i から q_j への単位ベクトル, $\epsilon \in (0, 1)$ は, σ -norm の固定パラメータである .

Algorithm 1 は, 集団としての目標を持たず, 二つの α -agent 間での相互作用にのみで状態が決まるので, (α, α) flocking として知られています.

Algorithm 2 $u_i = u_i^\alpha + u_i^\gamma$ すなわち

$$u_i = \sum_{j \in N_i} \phi_\alpha(\|q_j - q_i\|_\sigma) \mathbf{n}_{ij} + \sum_{j \in N_i} a_{ij}(q)(p_j - p_i) + f_i^\alpha(q_i, p_i, q_r, p_r) \quad (22)$$

ここで, u_i^γ は, *navigational feedback* であり, 次の式で与えられる.

$$u_i^\gamma := f_i^\alpha(q_i, p_i, q_r, p_r) = -c_1(q_i - q_r) - c_2(p_i - p_r), \quad c_1, c_2 > 0 \quad (23)$$

ここで $(q_r, p_r) \in \mathbb{R}^m \times \mathbb{R}^m$ は, γ -agent(ターゲット)の状態である.

Algorithm 2 において, γ -agent は集団の目標地点となるので, moving rendezvous point と考えることができる.

Algorithm 1 と Algorithm 2 とは見た目はよく似ているが, agent が実際にとる動きには大きな違いが生じる.

式 (21) に従う flocking の動きは, 初期配置のみで決まってしまう. 特に偏りなく配置された, 数多く agent ($n > 10$) に式 (21) を適用しようとする, agent 間で α -lattice を形成しようとするが, 全体としては Fig.6 のような fragmentation が発生してしまうことがある. この fragmentation は, flocking が陥る問題点である. これに対して, 式 (22) に従う flocking は, この fragmentation が発生することはない.

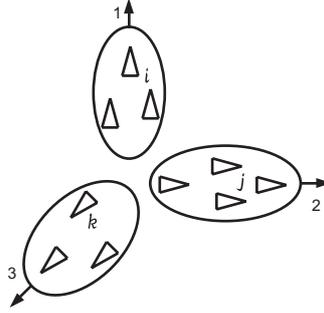


Fig.6: Fragmentation phenomenon.

文献 [1] では, この文献 [2] で紹介されている Algorithm 2 の (q_r, p_r) を, 各 α -agent から測定した γ -agent の推定状態値 $(\hat{q}_{i,\gamma}, \hat{p}_{i,\gamma})$ に置き換え, 次の algorithm を提案している.

Algorithm 3

$$\dot{q}_i = p_i \quad (24)$$

$$\dot{p}_i = \sum_{j \in N_i} f_{ij} \mathbf{n}_{ij} + \sum_{j \in N_i} a_{ij}(q)(q_j - p_i) + f_i^\gamma \quad (25)$$

ここで $f_{ij} = \phi_\alpha(\|q_j - q_i\|_\sigma)$. また, $(\hat{q}_{i,\gamma}, \hat{p}_{i,\gamma})$ を γ -agent の推定状態値として

$$f_i^\gamma = -c_1(q_i - \hat{q}_{i,\gamma}) - c_2(p_i - \hat{p}_{i,\gamma}), \quad c_1, c_2 > 0 \quad (26)$$

文献 [1] では, このアルゴリズムを用いることで, すべての agent が漸近的にターゲットの推定値を目標として consensus を達成できると仮定して, 次の提案を行っている.

Proposition 1 n 台の agent が Algorithm 3 にしたがって動き, それぞれがターゲットの位置と速度 $(q_{i,\gamma}, p_{i,\gamma})$ をノイズを含んだ値で測定していると仮定する. そして有限時間 $T > 0$ 後に, すべての agent がターゲットの推定状態値 $(\hat{q}_{i,\gamma}, \hat{p}_{i,\gamma})$ を目標値として consensus を達成でき, この時の $\hat{q}_{i,\gamma} = \bar{q}, \forall i$ が成り立つとすると,

1. agent は, flocking を達成する
2. グラフ構造は, 不変になる
3. agent が形作る構成は, quasi α -lattice になる. つまり agent 間の距離が $d \pm \delta$ になる.

文献 [1] で紹介されている, agent が Algorithm 3 にしたがって flocking を達成していく simulation 結果を Fig.7 に示す.

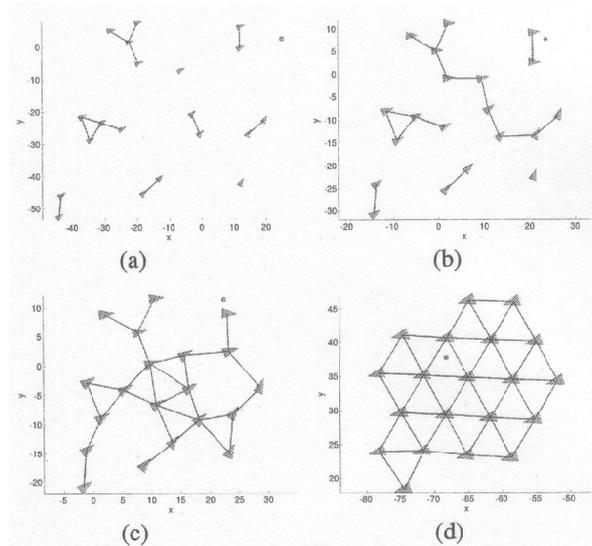


Fig.7: Emergence of flock of 20 sensors (triangles) cooperative tracking of mobile target (square)

3 終わりに

本レポートでは, R.Olfati-Saber の文献 [1] について扱った. 今回は, ターゲットの追従を行うための Flocking Algorithm についてのみを扱ったので, そこで必要になる状態推定のための Micro-Kalman Filter や, Consensus Filter について今後詳しく考えていきたい. また, 自分自身で Simulation を行い, Algorithm 3 や Micro-Kalman Filter の有効性を確認したい.

参考文献

- [1] R.Olfati-Saber. "Distributed Tracking for Mobile Sensor Networks with Information-Driven Mobility," *The 45th IEEE Conference on Decision and Control*, Dec. 2006.
- [2] R.Olfati-Saber. "Flocking for Multi-Agent Dynamic System: Algorithms and Theory," *IEEE Trans. on Automatic Control*, 51(3):401–420, Mar. 2006
- [3] R.Olfati-Saber. "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Automatic Control*, vol.49, no.9, pp.1520–1533, Sep 2004.