

Flocking in Fixed Networks

藤田研究室

小林尚斗

平成18年6月26日

1 はじめに

本レポートは、文献 [1], [2] で提起されている、時不変の topology を持つ複数台の agent による加速度入力の flocking 問題についてまとめる。まずはじめに2章で文献 [3] の前半部分を参考に consensus 問題を軽く紹介する。3章で本レポートで扱う flocking の問題設定をする。そして4章で flocking を達成するための制御則の提案とその証明をし、5章でシミュレーション結果を行う。また、専門用語の説明を付録に載せる。

2 Consensus 問題の紹介

まず文献 [3] の前半部分を参考に consensus 問題について紹介していく。ここで consensus とは複数台の agent のある状態量がすべて一致することを言う。これから具体的に例をあげて説明していく。

頂点の集合 \mathcal{V} と辺の集合 \mathcal{E} からなる無向・連結のグラフ $G = (\mathcal{V}, \mathcal{E})$ で表されるネットワークシステムについて考える。このシステムはダイナミクスが

$$\dot{x}_i = u_i \quad (1)$$

である N 台の agent からなるものとする。 x_i を agent i のある状態量、例えばここでは位置を表すものとし、 u_i が agent i に加えられる入力とする。このシステムにおいて consensus が達成されるということは $x_1 = x_2 = \dots = x_N$ となることである。これはつまり $x = \alpha \mathbf{1}$ となることである。ここで $\alpha \in \mathcal{R}$ は collective decision と呼ばれグループで決定される値である。次に agent i の近傍を $\mathcal{N}_i = \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$ と定義する。また $A = [a_{ij}]$ をグラフ G の隣接行列 (*1) とする。これを用いて次のような入力を考える。

$$\dot{x}_i = \sum_{j \in \mathcal{N}_i} a_{ij} (x_j(t) - x_i(t)) \quad (2)$$

これはすべての agent が近傍の agent と情報を通信し、ある同じ値へ収束するための consensus アルゴリズムとなっている。また、

$$\alpha = \frac{1}{n} \sum_i x_i(0) \quad (3)$$

となることが知られている。つまり式 (2) を入力するとすべての agent は各 agent の初期状態の平均値に収束することが知られている。この特性は average-consensus algorithm と呼ばれている。また式 (2) のダイナミクスは

$$\dot{x} = -Lx \quad (4)$$

と表すことができる。ここで $x = \{x_1, \dots, x_N\}$ であり、 L はグラフ G のグラフラプラジアンで

$$L \triangleq D - A \quad (5)$$

で定義されるものである。ここで D は次数行列 (*2) である。 L は $\sum_j l_{ij} = 0$ つまり $L\mathbf{1} = 0$ を満たしている。これから L が 0 固有値を持ち、その固有ベクトルが $\mathbf{1} = [1, \dots, 1]^T$ であることが分かる。

式 (1)(及び式 (3)) の収束の証明をする。まず

$$x^T Lx = \sum_{j \sim i} (x_j - x_i)^2 \quad (6)$$

をポテンシャル関数として考えると

$$x^T Lx \geq 0 \quad (7)$$

が分かる。次に時間微分すると

$$\frac{d}{dt}(x^T Lx) = \frac{1}{2}(\dot{x}^T Lx + x^T L\dot{x}) \quad (8)$$

$$= \frac{1}{2}(-x^T L^T Lx - x^T L Lx) \quad (9)$$

$$= -(Lx)^T (Lx) \leq 0 \quad (10)$$

なので LaSalle の不変原理を用いて、いかなる x も $\frac{d}{dt}(x^T Lx) = 0$ を満たす集合の中で一番大きな部分集合、すなわち $\alpha 1$ に収束することが証明される。

3 Flocking の問題設定

この章では文献 [1], [2] で扱われている問題について述べる。まず時不変、無向、連結のグラフ G で表される N 台の agent からなる集団を考える。個々の agent は次のような 2 次のダイナミクスで表されるとする。

$$\dot{r}_i = v_i \quad (11)$$

$$\dot{v}_i = u_i \quad i = 1, \dots, N \quad (12)$$

ここで $r_i = (x_i, y_i)^T$ は agent i の位置を表し、 $v_i = (\dot{x}_i, \dot{y}_i)^T$, $u_i = (u_x, u_y)^T$ はそれぞれ速度と加速度入力を表す。また、 $r_{ij} = r_i - r_j$ とする。 u_i は次のようなものを考える。

$$u_i = \alpha_i + a_i \quad (13)$$

上式の α_i はすべての agent の速度ベクトルを同じスピード、同じ方向にするための項である。 a_i は 4.1 節で定義するようなポテンシャル関数 V_i を小さくする方向のベクトルと考えられ、それゆえ衝突回避 (separation) やグループのまとまり (cohesion) に寄与するものと考えられる。これについては 4 章で詳しく述べる。文献 [1], [2] の見解によると、Reynolds の flocking に対する考えは、”すべての agent が同じ速度ベクトルを持ち、agent 間の距離が希望の距離になっており、またその過程において衝突が起こらないとき、mobile agent の集団は flocking している” というものである。すなわち $v_{ij} = \alpha 1$, $\|r_{ij}\| > 0$ ($\forall t > 0$ and $\forall (i, j) \in N \times N$) の 2 つを満たすとき flocking が達成されたといえる。ここでの問題はつまり、時不変の topology を持つシステムにおいてこの flocking を達成する新しい制御入力 u_i を考えることである。

4 制御則の提案とその証明

この章では前述のとおり、時不変、無向、連結の topology を持つシステムにおいて flocking を達成する制御入力について述べ、証明していく。

4.1 制御則の提案

まず、agent i に対するその新しい制御入力を次のように定義する。

$$u_i = - \underbrace{\sum_{j \in \mathcal{N}_i} (v_i - v_j)}_{\alpha_i} - \underbrace{\sum_{j \in \mathcal{N}_i} \nabla_{r_i} V_{ij}}_{a_i} \quad (14)$$

ここで \mathcal{N}_i は agent i の近傍の agent 集合を表し, V_{ij} は近傍の agent 間の距離に依存するポテンシャル関数である. V_{ij} を次のように定義する.

定義 1 ポテンシャル V_{ij} は微分可能, 非負, そして有界でなく, agent i, j 間の距離 $\|r_{ij}\|$ に関する関数で, 次のようなものである.

1. $V_{ij}(\|r_{ij}\|) \rightarrow \infty$ as $\|r_{ij}\| \rightarrow 0$,
2. $\|r_{ij}\|$ が希望の値になった時 V_{ij} は唯一の最小値をとる.

この定義より, 個々の agent 間のポテンシャル関数 V_{ij} の最小化は集団の中での cohesion(まとまり) と separation(分離) を意味することが分かる. 文献 [1], [2] ではこのようなポテンシャル関数として (4.1) に示すものを例にあげている. また, agent i のポテンシャル V_i は次式で定義される.

$$V_i \triangleq \sum_{j \in \mathcal{N}_i} V_{ij}(\|r_{ij}\|) \quad (15)$$

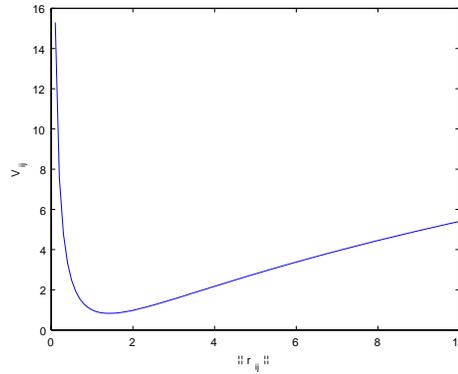


Fig.1 ポテンシャル関数 V_{ij} の例: $V_{ij} = \log^2(\|r_{ij}\|) + \frac{1}{\|r_{ij}\|}$

4.2 時不変, 無向, 連結の topology の下での flocking と証明

システムのグラフ G が時不変, 無向, 連結 (connected) であるとき, LaSalle の不変原理を用いて式 (11)-(14) に従う agent の閉ループシステムは flocking となるという次に示す定理 1 が証明できる. (厳密にはすべての agent 間での衝突回避については証明されていない. これを満たす条件は後に系 3 に示す)

定理 1 ダイナミクス (11), (12) を持ち, 入力 (14) であり, そのグラフが時不変, 無向, 連結であるような N 台の mobile agents からなるシステムのすべての agent の速度ベクトルは漸近的に等しくなり, システムはすべての agent のポテンシャルが局地的に最小化される姿勢に近づいていく.

証明 まず, 次のような非負の関数を考える.

$$W(r, v) = \frac{1}{2} \sum_{i=1}^N (V_i + v_i^T v_i) \quad (16)$$

ここで $r = (r_1, \dots, r_N)^T$, $v = (v_1, \dots, v_N)^T$ である. この関数はすべての agent のポテンシャル V_i と運動エネルギーを足し合わせたものであると考えられる. この $W(r, v)$ を用いて次の compact(*3) な集合を考える.

$$\Omega = \{(r, v) | W(r, v) \leq c\} \quad (17)$$

ここで $W(r, u)$ の時間微分は

$$\dot{W}(r, v) = \frac{1}{2} \sum_{i=1}^N \dot{V}_i - \sum_{i=1}^N v_i^T \left(\sum_{j \in \mathcal{N}_i} (v_i - v_j) + \nabla_{r_i} V_i \right) \quad (18)$$

である。一方 V_{ij} は $\|r_{ij}\|$ の関数なので i と j を入れ替えても同じ値となる。この対称性を用いて

$$\frac{1}{2} \sum_{i=1}^N \dot{V}_i = \sum_{j \sim i} r_{ij}^T \nabla_{r_{ij}} V_{ij} = \sum_{j \sim i} (r_i^T \nabla_{r_{ij}} V_{ij} - r_j^T \nabla_{r_{ij}} V_{ij}) = \sum_{j \sim i} (r_i^T \nabla_{r_i} V_{ij} + r_j^T \nabla_{r_j} V_{ij}) = \sum_{i=1}^N r_i^T \nabla_{r_i} V_i \quad (19)$$

これより式 (18) は

$$\dot{W}(r, v) = \sum_{i=1}^N v_i^T \nabla_{r_i} V_i - \sum_{i=1}^N v_i^T \left(\sum_{j \in \mathcal{N}_i} (v_i - v_j) + \nabla_{r_i} V_i \right) = - \sum_{i=1}^N v_i^T \sum_{j \in \mathcal{N}_i} (v_i - v_j) = -v^T (L \otimes I_2) v \quad (20)$$

と書き直せる。 L はこのシステムのグラフの Laplacian で、 \otimes は Kronecker matrix product(*4) を表す。これより $\dot{W}(r, u)$ は次のように x, y 成分に分けて考えられる。また無向・連結のグラフを考えているので式 (6) と同様に変形でき、

$$\begin{aligned} \dot{W}(r, u) &= -v_x^T L v_x - v_y^T L v_y \quad (21) \\ &= - \sum_{j \sim i} (v_{jx} - v_{ix})^2 - \sum_{j \sim i} (v_{jy} - v_{iy})^2 \leq 0 \quad (22) \end{aligned}$$

ここで $v_x = \{v_{1x}, \dots, v_{N_x}\}^T, v_y = \{v_{1y}, \dots, v_{N_y}\}^T$ である。上式は v_x, v_y がどちらも $\text{span}\{1\} (= \{\alpha \mathbf{1} | \alpha \in R\})$ に属するときのみ $\dot{W} = 0$ になるということの意味している。ここで v_x, v_y が $\text{span}\{1\}$ に属するというのはつまりすべての agent の速度が等しくなるということである。LaSalle の不変原理をこのシステムに適用すると、システムの初期状態が Ω の中にあるならば、 $\dot{W}(r, v) \leq 0$ より Ω は正不変であり、 (r, v) は $S = \{(r, v) | \dot{W}(r, v) = 0\}$ の中の最大の不变集合に収束することが分かる。なおここで c を大きくとることによりシステムの初期状態を Ω に含ませることができることに注意すると、このシステムは必ず漸的にすべての agent の速度が等しくなることが分かる。

以降は S の中で十分時間が経過して安定な状態において agent の速度は変化しない。すなわち $\dot{v} = 0$ であるということと、また agent i のポテンシャル関数 V_i が局所的に最小化されることを導く。

S の中においては agent の速度のダイナミクスは (14) において第一項が 0 になるので、 $\begin{bmatrix} \vdots \\ \nabla_{r_{ij}} V_{ij} \\ \vdots \end{bmatrix}$ をうまくとれば

$$\dot{v} = - \begin{bmatrix} \nabla_{r_1} V_1 \\ \vdots \\ \nabla_{r_N} V_N \end{bmatrix} = -(B \otimes I_2) \begin{bmatrix} \vdots \\ \nabla_{r_{ij}} V_{ij} \\ \vdots \end{bmatrix} \quad (23)$$

と変形することができる。ここで B は接続行列 (*5) である。これを x, y 成分それぞれに書き下すと

$$\dot{v}_x = -B[\nabla_{r_{ij}} V_{ij}]_x \quad \dot{v}_y = -B[\nabla_{r_{ij}} V_{ij}]_y \quad (24)$$

これより \dot{v}_x, \dot{v}_y はどちらも接続行列の右からある 1 列の縦ベクトルを掛けた形になっていることが分かる。ここで接続行列の $B^T \mathbf{1} = 0$ という性質を用いて

$$\dot{v}_x^T \mathbf{1} = -[\nabla_{r_{ij}} V_{ij}]^T B^T \mathbf{1} = 0 \quad (25)$$

ここで $\dot{v}_x^T \mathbf{1}$ は \dot{v}_x と $\mathbf{1}$ の内積であり、内積が 0 であるので \dot{v}_x は $\mathbf{1}$ と直交することが分かる。 \dot{v}_y も同様である。すなわち

$$\dot{v}_x, \dot{v}_y \in \text{span}\{1\}^\perp \quad (26)$$

が成り立つ。しかし一方 S の中では

$$v_x, v_y \in \text{span}\{1\} \Rightarrow \dot{v}_x, \dot{v}_y \in \text{span}\{1\} \quad (27)$$

上の2式は次のもとでないと矛盾する。

$$\dot{v}_x, \dot{v}_y \in \text{span}\{1\} \cap \text{span}\{1\}^\perp \equiv \{0\} \quad (28)$$

よって安定状態で agent の速度は変化しないことが分かる。また式 (23) より
$$\begin{bmatrix} \nabla_{r_1} V_1 \\ \vdots \\ \nabla_{r_N} V_N \end{bmatrix} = 0$$
 となるので, agent i の

ポテンシャル V_i が局所的に最小化されることも分かる。

系 1 (Distance setpoint stabilization) もしシステムを表すグラフが tree(*6) ならば, 近傍関係にある agent 間の距離は求めている距離に安定させることができる。

証明

もしグラフが tree ならば, その接続行列 B はフルランク (*7) なので次式が成り立つ。

$$(B \otimes I_2) \begin{bmatrix} \vdots \\ \nabla_{r_{ij}} V_{ij} \\ \vdots \end{bmatrix} = 0 \Rightarrow \begin{bmatrix} \vdots \\ \nabla_{r_{ij}} V_{ij} \\ \vdots \end{bmatrix} = 0, \quad (29)$$

ここで r_d を V_{ij} が最小値をとるような値だとすると, $\frac{\partial V_{ij}}{\partial \|r_{ij}\|} = 0$ は $\|r_{ij}\| = r_d$ を意味し, これより近傍関係にある agent 間の距離を希望の距離にすることができることが分かる。

系 2 (Convergence speed) グラフラプラシアン of 2 番目に小さい固有値 (algebraic connectivity) が大きくなるにつれ, 速度の同調のスピードは加速されていく。

証明

文献 [1] 参照。

系 3 (Collision avoidance in fixed networks) もし topology が complete なグラフになっているとすると, すべての agent 間での衝突回避が保障される。

証明

文献 [1] 参照。

5 シミュレーション

この章では図 (2), (3) に示すようなダイナミクス (11), (12) を持つ 4 台の agent からなるシステムに式 (14) を入力として加えた場合についてそれぞれシミュレーションする。また今回は, 図 (4.1) に示すポテンシャル関数

$$V_{ij} = \log^2(\|r_{ij}\|) + \frac{1}{\|r_{ij}\|} \quad (30)$$

を用いる。

まず, どちらの場合も各 agent の初期状態を次のようにおく。

$$r_1 = (-3, 5)^T \quad v_1 = (1, 4)^T \quad r_2 = (1, 3)^T \quad v_2 = (-3, 7)^T \quad (31)$$

$$r_3 = (3, 7)^T \quad v_3 = (2, -2)^T \quad r_4 = (-3, -6)^T \quad v_4 = (-5, -3)^T \quad (32)$$

そして時刻 $t = 0[s]$ $t = 300[s]$ までシミュレーションする。その結果を図 (4)-(7) に示す。これらを見ると graph1, graph2 のどちらの場合もすべての agent の速度が $(-1.25, 1.5)^T$ に収束しており, これは各 agent の速度の初期値の平均と等しくなっている。

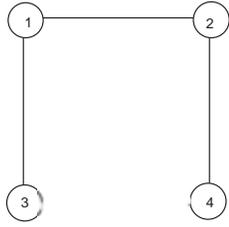


Fig.2 graph1(tree)

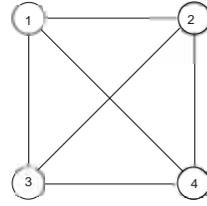


Fig.3 graph2

系 1 で述べた, tree の場合 (graph1) は agent 間の距離を望ましい距離にすることができることを確認するため,十分に時間が経過した際の軌跡に注目したのが図 (8) であり, その比較のため tree でない場合 (graph2) の十分に時間が経過した際の軌跡を示したものが図 (9) である. graph1 での 300[s] での $\|r_{ij}\|$ をそれぞれ求めるとすべて 1.421 で一致した. この値は式 (30) に示す評価関数 V_{ij} を最小とする $\|r_{ij}\|$ の値 $\|r_{ij}\| = 1.421$ と等しくなっている. これより系 1 の妥当性が確認された. 一方 graph2 での 300[s] での $\|r_{ij}\|$ は図 (9) からばらつきがあることが分かる.

また, graph1, graph2 の 2 番目に小さい固有値 (algebraic connectivity) はそれぞれ 0.5858, 4 となっており, graph2 のほうが大きくなっている. そこで図 (5), (7) を比較してみると, graph2 の場合のほうがわずかではあるが graph1 より早く速度が同調していることが分かる. これより系 2 の妥当性が確認された.

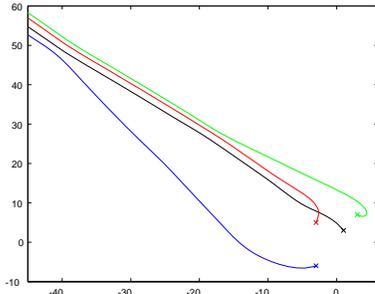


Fig.4 r(graph1)

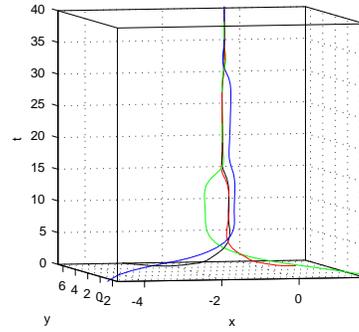


Fig.5 v(graph1)

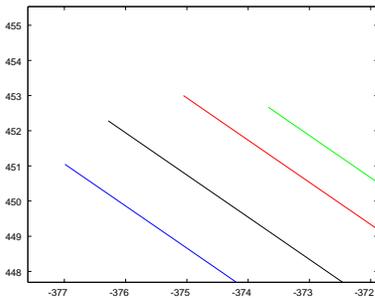


Fig.8 300[s] 経過したときの r(graph1)

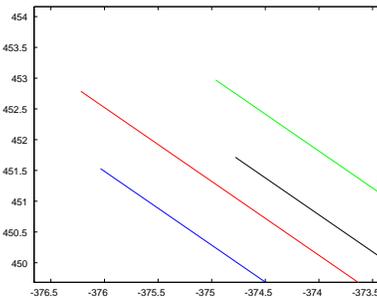


Fig.9 300[s] 経過したときの r(graph2)

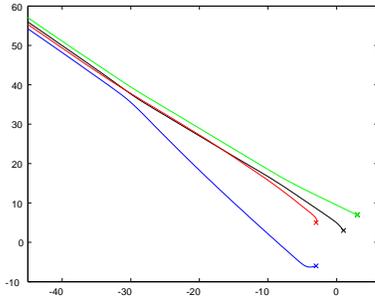


Fig.6 r(graph2)

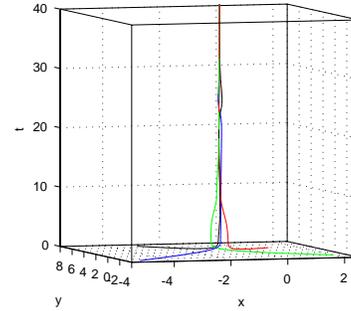


Fig.7 v(graph2)

6 おわりに

本レポートでは文献 [1], [2] で提案されている flocking の制御則を紹介した。今回は T_EX や Matlab の使い方を習得するという目的が大きかったので、文献の内容の理解にかける時間より資料作成のための時間のほうが長くなってしまった。これからは文献の理解にもっと長く時間を費やすことができるようにしていきたい。また、これから大学院入試が終わるまでは入試に向けての勉強をし、入試が終わってからは consensus 問題の基礎的な問題をきちんと理解できるようになるために、なるべく多くの文献にふれていこうと考えている。

7 付録

(*1) 隣接行列 (adjacency matrix) : 隣接行列 $A = [a_{ij}]$ とは agent j が agent i の近傍である ($j \in \mathcal{N}_i$) とき $a_{ij} = 1$ とし、それ以外が 0 であるような $n \times n$ の行列である。

(*2) 次数行列 (valency(degree) matrix) : 次数行列 $D = [d_{ii}]$ とは対角成分 d_{ii} に agent i の近傍の数を持ち、それ以外が 0 であるような $N \times N$ の行列である。

(*3) compact : 有界で閉じた集合ことを compact という。

(*4) Kronecker matrix product : Kronecker matrix product とは、行列 P, Q が $P : m \times n, Q : r \times s$ のとき

$$P \otimes Q = \begin{bmatrix} p_{11}Q & \dots & p_{1n}Q \\ \vdots & \ddots & \vdots \\ p_{m1}Q & \dots & p_{mn}Q \end{bmatrix} \quad (33)$$

となる演算子である。

(*5) 接続行列 (incidence matrix) : 接続行列とは agent i に edge j が入ってくるとき $b_{ij} = -1$, edge j が出ていくとき $b_{ij} = 1$ となるような $N \times \text{thenumberofedges}$ の行列である。

(*6) tree : グラフの任意の 2 つの agent が 1 つの path により連結されるとき、そのグラフを tree という。

(*7) tree のグラフの接続行列がフルランクであることを証明する。

頂点の数が p , 辺の数が q であるようなグラフ G に対して次の定理が成り立つ。

定理: グラフ G が ρ 個の連結成分を持つ時, G の接続行列 B の階数は $\text{rank}B = p - \rho$ である。

ここで tree のグラフは連結なグラフなので連結成分は 1 個しか持たない ($\rho = 1$)。すなわち $\text{rank}B = p - 1$ となる。また tree のグラフに対して次の定理が成り立つ。

定理: G が tree ならば $q = p - 1$ である。

接続行列 B は $p \times q$ の行列である. ここでは特に $q = p - 1$ なので B は $p \times p - 1$ の行列であることが分かる. またここで $\text{rank} B = p - 1$ であるので, これらより tree のグラフの接続行列はフルランクであることが証明される.

参考文献

- [1] Herbert G. Tanner, Ali Jadbabaie and George J. Pappas, “Flocking in Fixed and Switching Networks,” *IEEE Trans. on Automatic Control*, submitted,
- [2] Herbert G. Tanner, Ali Jadbabaie, and George J. Pappas, “Stable Flocking of Mobile Agents, Part 1: Fixed Topology,” *Proc. of IEEE Conference on Decision and Control*, Vol. 2, pp. 2016–2021, 2003.
- [3] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray, “Consensus and Cooperation in Networked Multi-Agent Systems,” *Proc. of the IEEE*, 2006.
- [4] 五十嵐裕司, “consensus 問題の性質,” ゼミ資料 *FL06-07-2*, 2006.
- [5] Hassan K. Khalil, “Nonlinear Systems Third Edition,” *Prentice Hall*, 2001.