


SICEセミナー「モデル予測制御理論と協調制御の新展開」

Tokyo Institute of Technology

SICE セミナー
「実践的な制御系設計の最前線
-ポストロバスト制御理論と応用の新展開-」
「モデル予測制御理論と協調制御と協調制御の新展開」

第 I 部


2006年10月27日
電気通信大学スカイオフィス

東京工業大学 藤田政之

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

*** アウトライン**

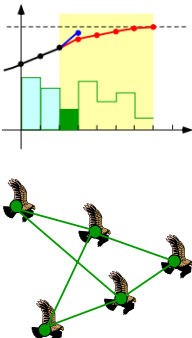
第 I 部

- 非線形モデル予測制御
 - 非線形モデル予測制御の安定性
 - 非線形モデル予測制御の応用例

第 II 部

協調制御

- 合意 (consensus) 問題
- 群れ (Flocking)



Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

はじめに

モデル予測制御の成立過程

- プロセスの現場での経験的制御
- 適応制御 (一般化予測制御) からの発展
- 最適制御 (optimal control) からの発展

↓

有限時間区間 (finite horizon) の最適制御を、時刻が進むにつれて評価時間区間を先へずらして進めていく

receding horizon (moving horizon)
の考え方に基づくアドバンスな最適化制御

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

最適制御とモデル予測制御

有限時間 (finite horizon) 最適制御
最適性

無限時間 (infinite horizon) 最適制御
最適性, 安定性 ($t = \infty$)

モデル予測制御 (receding horizon control)
最適性, 安定性??

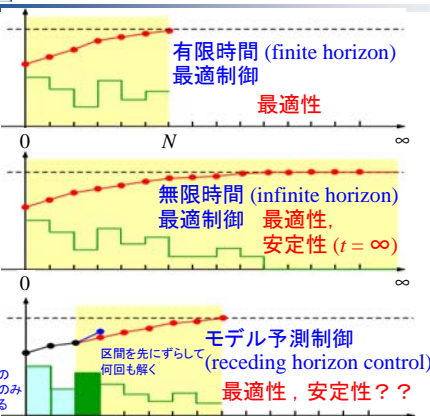


Fig: アポロ15号 "NASA"

Fig: F-8 "F-8"

Fig: 車

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

モデル予測制御を適用するシステム

非線形システム (制御対象)

$$x(k+1) = f(x(k), u(k)) \quad (1) \quad f(0,0) = 0$$

例 (ロボットマニピュレータ)


$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau$$


Fig: ロボットマニピュレータ

線形システム (制御対象)

$$x(k+1) = Ax(k) + Bu(k)$$

例 (バネマスダンパ系)

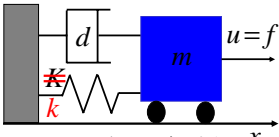
$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{d}{m} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u$$


Fig: バネマスダンパ系

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

入力と状態に関する制約

入力と状態に関する制約

入力制約 $u(k) \in \mathcal{U}$

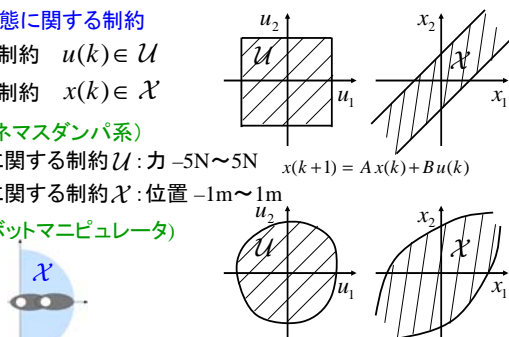
状態制約 $x(k) \in \mathcal{X}$

例 (バネマスダンパ系)

入力に関する制約 \mathcal{U} : 力 $-5N \sim 5N$ $x(k+1) = Ax(k) + Bu(k)$

状態に関する制約 \mathcal{X} : 位置 $-1m \sim 1m$

例 (ロボットマニピュレータ)



入力に関する制約 \mathcal{U} : 入力電圧 $-10V \sim 10V$

状態に関する制約 \mathcal{X} : 関節角度 $-90^\circ \sim 90^\circ$

Tokyo Institute of Technology Fujita Laboratory

SICEセミナー「モデル予測制御理論と協調制御の展開」

ステージコストと終端コスト

コスト (評価関数)

$$V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N))$$

$l(x(i), u(i))$: ステージコスト (stage cost)
 $F(x(N))$: 終端コスト (terminal cost)
 $l(x(i), u(i)) > 0, l(0,0) = 0, F(x(N)) > 0$

有限の N ステップ区間での制御問題

例 線形システムの評価関数

ステージコスト $l(x(i), u(i)) = x(i)^T Q x(i) + u(i)^T R u(i)$ $Q \geq 0, R > 0$
 終端コスト $F(x(N)) = x(N)^T P_f x(N)$

制約条件付きの最適制御問題

制約条件付きの最適制御問題 (optimal control problem)

$$\mathcal{P}_N(x) : \min_u \{ V_N(x, u) \mid u \in \mathcal{U}_N(x) \} \quad (2)$$

$\mathcal{U}_N(x)$: 入力と状態に関する制約を満たす入力集合

最適入力 $x : i = 0$ における状態 $x^o(0; x)$
 $u^o(x) = \{ u^o(0; x), u^o(1; x), \dots, u^o(N-1; x) \} \quad (3)$

最適状態軌道 (trajectory)
 $x^o(x) = \{ x^o(0; x), x^o(1; x), \dots, x^o(N-1; x), x^o(N; x) \}$
 $= f(x^o(N-1; x), u(N-1; x))$

最適コスト (value function)
 $V_N^o(x) = V_N(x, u^o(x))$

モデル予測制御とは

モデル予測制御では
 最適入力系列の最初の操作量 $u^o(0; x)$ のみを制御対象に適用
 制御則 $K_N(x) := u^o(0; x)$ $u^o(x) = \{ u^o(0; x), u^o(1; x), \dots, u^o(N-1; x) \}$

最適制御問題における N ステップの区間を一つ先に進めながら、最適制御問題を繰り返し解く。

receding horizon (moving horizon) $u = K_N(x)$ という入力を繰り返しかわえる。

モデル予測制御

安定性のための終端制約集合と終端コスト

閉ループ系の安定性

非線形システム
 モデル予測制御

終端制約集合 X_f
 終端等式制約 $x(N) = 0$ Kwon and Pearson (1977)
 $X_f = \{0\}$

緩和 $x(N) \in X_f$
 X_f は原点近傍

終端コスト $F(x(N))$
 安定性 \rightarrow 終端制約集合 X_f

評価関数 $V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N))$

$F(x(N)) = x(N)^T P_f x(N)$
 P_f が大 \rightarrow 安定性 \rightarrow 終端コスト $F(x(N))$
 P_f が小 \rightarrow 安定性 \rightarrow 終端コスト $F(x(N))$

線形システムの無限時間最適制御の安定性

離散時間線形システム $x(k+1) = Ax(k) + Bu(k)$

評価関数 (∞) $V_\infty(x, u) = \sum_{i=0}^{\infty} (x^T(i) Q x(i) + u^T(i) R u(i))$

最適入力 $u^o(k) = Kx(k)$ $K = -(R + B^T P B)^{-1} B^T P A$
 (安定化制御則)
 代数リカッチ方程式 $P = A^T P A - A^T P B (R + B^T P B)^{-1} B^T P A + Q$

最適コスト $V_\infty^o(k) = V_\infty(x(k), u^o(k)) = x(k)^T P x(k)$

安定性 $\Delta V_\infty^o = V_\infty^o(k+1) - V_\infty^o(k)$
 $= x(k+1)^T P x(k+1) - x(k)^T P x(k)$
 $= -l(x(k), u(k)) \leq 0$ (等号は $x(k) = u(k) = 0$ のみ)

最適コスト $V_\infty^o(k)$ をリアプノフ関数とみれば、安定性がいえる。
 最適入力 $u^o(k) = Kx(k)$ は安定化制御則となっている。

線形システムのモデル予測制御の安定性

線形システム $x(k+1) = Ax(k) + Bu(k)$ $x(0) = x_0$

入力制約 $u(k) \in \mathcal{U}$ $k = 0, \dots, N-1$
 状態制約 $x(k) \in \mathcal{X}$ $k = 0, \dots, N$
 終端制約 $x(N) \in \mathcal{X}_f \subset \mathcal{X}$

評価関数 $V_N(x, u) = \sum_{i=0}^{N-1} (x^T(i) Q x(i) + u^T(i) R u(i)) + x^T(N) P_f x(N)$
 ステージコスト 終端コスト

閉ループ系の安定性をいうのに重要な要素 (線形システム)

- 終端制約集合 \mathcal{X}_f は最大出力許容集合 \mathcal{O}
- 終端ステップにおける安定化制御則の存在 $\kappa_f(x) = Kx$
 $K = -(R + B^T P_f B)^{-1} B^T P_f A$
- 終端コスト $F(x(N)) = x^T(N) P_f x(N)$ $\Delta F(x) + l(x, \kappa_f(x)) \leq 0$

SICEセミナー「モデル予測制御理論と協調制御の展開」

非線形モデル予測制御の安定性の準備

Tokyo Institute of Technology

終端制約 (terminal constraint)
 終端の第 N ステップにおける状態の制約
 $x(N) \in X_f \subset \mathcal{X}$
 X_f : 終端制約集合 (terminal constraint set)

仮定: 非線形システム (1) に対して, (局所的な) 安定化制御則 (stabilizing control) が存在する
 $u = \kappa_f(x)$

状態が x のときに入力 u を加えた次のステップの状態
 $x^+ = f(x, u)$

状態が x から x^+ へ変化したときの関数 $\phi(x, u)$
 $\Delta\phi(x, u) := \phi(x^+) - \phi(x)$

13 Fujita Laboratories

非線形モデル予測制御の安定条件

Tokyo Institute of Technology

非線形モデル予測制御の安定条件

A1: $X_f \subset \mathcal{X}$, X_f は閉集合, $0 \in X_f$
 A2: $\kappa_f(x) \in \mathcal{U}$, $\forall x \in X_f$
 A3: $f(x, \kappa_f(x)) \in X_f$, $\forall x \in X_f$
 A4: $[\Delta F + I](x, \kappa_f(x)) \leq 0$, $\forall x \in X_f$

A3: 終端制約集合 X_f が正の不変集合 (positive invariant set)
 A4: 終端コスト $F(x)$ は (局所的な) リアプノフ関数 (control Lyapunov function) となる

A1-A3: 入力と状態に関する制約および終端制約が満たされている

↓

可能解の存在性 (feasibility)

14 Fujita Laboratories

非線形モデル予測制御の安定条件の略証

Tokyo Institute of Technology

非線形モデル予測制御の安定条件の略証 (3)

(2) 最適制御問題 (7) と対応する最適入力 (8) に対して, 1ステップ経過した時点で新たに x^+ を初期値とする N ステップの最適制御問題 $\mathcal{P}_N(x^+)$ を考える.

制約条件つき最適制御問題に対する入力

$\tilde{u}(x) := \{u^o(1; x), \dots, u^o(N-1; x), \kappa_f(x^o(N; x))\}$ (16)

$\kappa_f(x^o(N; x)) \in \mathcal{U}$ $f(x^o(N; x), \kappa_f(x^o(N; x))) \in X_f \subset \mathcal{X}$

15 Fujita Laboratories

非線形モデル予測制御の安定条件の略証

Tokyo Institute of Technology

16 Fujita Laboratories

非線形モデル予測制御の安定条件の略証

Tokyo Institute of Technology

17 Fujita Laboratories

非線形モデル予測制御の安定条件の略証

Tokyo Institute of Technology

コスト (評価関数) $V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N))$ (17)

$V_N(x^+, \tilde{u}(x)) = \sum_{i=0}^{N-1} l(x^o(i; x), u^o(i; x)) + F(x^o(N; x))$

$= l(x^o(0; x), u^o(0; x)) + F(x^o(N; x)) + \dots + l(x^o(N-1; x), u^o(N-1; x)) + F(f(x^o(N; x), \kappa_f(x^o(N; x))))$

18 Fujita Laboratories

SICEセミナー「モデル予測制御理論と協調制御の展開」

非線形モデル予測制御の安定条件の略証

入力列 $\tilde{u}(x)$

$$V_N(x^+, \tilde{u}(x)) = V_N^o(x(0;x), u^o(0;x)) - l(x^o(0;x), u^o(0;x)) - F(x^o(N;x))$$

$$= V_N^o(x) = x = \kappa_N(x)$$

$$+ l(x^o(N;x), \kappa_f(x^o(N;x))) + F(f(x^o(N;x), \kappa_f(x^o(N;x))))$$

$$= V_N^o(x) - l(x, \kappa_N(x)) + \Delta F(x^o(N;x), \kappa_f(x^o(N;x)))$$

$$+ l(x^o(N;x), \kappa_f(x^o(N;x)))$$

$\therefore \Delta F(x, u) := F(x^+) - F(x)$

非線形モデル予測制御の安定条件の略証

入力列 $\tilde{u}(x)$

$$V_N(x^+, \tilde{u}(x)) = V_N^o(x) - l(x, \kappa_N(x)) + \Delta F(x^o(N;x), \kappa_f(x^o(N;x)))$$

$$+ l(x^o(N;x), \kappa_f(x^o(N;x)))$$

$$= V_N^o(x) - l(x, \kappa_N(x)) + [\Delta F + l](x^o(N;x), \kappa_f(x^o(N;x)))$$

$$\leq V_N^o(x) - l(x, \kappa_N(x)) \stackrel{(18)}{\leq} 0$$

\therefore モデル予測制御の安定条件 A4 : $[\Delta F + l](x, \kappa_f(x)) \leq 0, \forall x \in X$

非線形モデル予測制御の安定条件の略証

入力列 $\tilde{u}(x)$

$$\therefore V_N(x^+, \tilde{u}(x)) \leq V_N^o(x) - l(x, \kappa_N(x)) \stackrel{(18)}{\leq} 0$$

一方 $\tilde{u}(x) := \{u^o(1;x), \dots, u^o(N-1;x), \kappa_f(x^o(N;x))\}$
 最適入力とは限らない

$$\therefore V_N(x^+) \leq V_N(x^+, \tilde{u}(x)) \stackrel{(19)}{\leq} 0$$

$$\therefore V_N^o(x^+) \leq V_N(x^+, \tilde{u}(x)) \leq V_N^o(x) - l(x, \kappa_N(x))$$

非線形モデル予測制御の安定条件の略証

入力列 $\tilde{u}(x)$

$$\therefore V_N(x^+) \leq V_N^o(x) - l(x, \kappa_N(x))$$

$$\Delta V_N^o(x, \kappa_N(x)) \leq -l(x, \kappa_N(x)) \stackrel{(18)}{\leq} 0 \therefore \Delta V_N^o(x, u) := V_N^o(x^+) - V_N^o(x)$$

value function $V_N^o(\cdot) > 0$
 value function $V_N^o(\cdot)$ をリアプノフ関数候補として用い、
 時間の経過とともに $V_N^o(\cdot)$ の値が単調減少し原点 $x = 0$ へ

閉ループ系の漸近安定性

終端コストの意味づけ

無限時間 (infinite horizon) 最適制御
 最適性, 安定性 ($t = \infty$)

モデル予測制御 (receding horizon control)

コスト (評価関数)
 $V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N))$

$F(x(N))$: 終端コスト (terminal cost)

$F(x(N))$: $N \sim \infty$ までのコストを込める \rightarrow 無限時間最適制御の安定性

終端コストの意味づけ

例: 線形の場合
 システム $x(k+1) = Ax(k) + Bu(k)$
 $y(k) = Cx(k)$

評価関数
 $V_N(x, u) = \sum_{i=0}^{N-1} \{x^T(i)Qx(i) + u^T(i)Ru(i)\} + x^T(N)P_f x(N)$

どのように決めるのか?

モデル予測制御の安定条件 A4:
 $[\Delta F + l](x, \kappa_f(x)) \leq 0, \forall x \in X_f$ より求められる条件である

Nステップ以降 $\kappa_f(x) = 0$ として A4 を満たす終端コストを考える
 操作量は 0

リアプノフ方程式 $A_f^T P_f A_f + Q_f = P_f$ を満たす $A_f := A + BK_f$
 $Q_f := Q + K_f^T R K_f$
 $\kappa_f(x) = K_f x$

$$P_f = \sum_{k=0}^{\infty} (A_f^k)^T Q_f A_f^k$$
 を用いると $\kappa_f(x) = K_f x$

* 実際にはリアプノフ方程式を直接計算する

SICEセミナー「モデル予測制御理論と協調制御の展開」

終端コストの意味づけ

評価関数

$$V_N(x, u) = \sum_{i=0}^{N-1} \{x^T(i)Qx(i) + u^T(i)Ru(i)\} + x^T(N)P_f x(N)$$

$$= \sum_{i=0}^{N-1} \{x^T(i)Qx(i) + u^T(i)Ru(i)\} + \sum_{i=N}^{\infty} \{x^T(i)Qx(i) + u^T(i)Ru(i)\}$$

$$= \sum_{i=0}^{\infty} \{x^T(i)Qx(i) + u^T(i)Ru(i)\}$$

つまり
終端コスト $F(x(N))$ に $N \sim \infty$ までのコストに込めること

無限時間 (infinite horizon) 最適制御の安定性

終端コストに control Lyapunov function を用いる

A. Jadbabaie, J. Yu and J. Hauser, "Unconstrained Receding-Horizon Control of Nonlinear Systems," *IEEE Trans. Automatic Control*, Vol.46, No.5, pp.776-783, 2001.

無限時間最適制御と有限時間最適制御

離散時間線形システム $x(k+1) = Ax(k) + Bu(k)$

無限時間最適制御の評価関数

$$V_{\infty}(x, u) = \sum_{i=0}^{\infty} (x^T(i)Qx(i) + u^T(i)Ru(i))$$

最適入力 $u^o(k) = Kx(k)$
 $K = -(R + B^T P B)^{-1} B^T P A$

モデル予測制御の評価関数

$$V_N(x, u) = \sum_{i=0}^{N-1} (x^T(i)Qx(i) + u^T(i)Ru(i)) + x(N)^T P_f x(N)$$

$N \sim \infty$ の最適なコスト

終端に込める!

制約条件 $x(N) \in \mathcal{X}_f$ (終端制約)

\mathcal{X}_f における安定化入力
 $K_f(x) = Kx(k)$
 $K = -(R + B^T P_f B)^{-1} B^T P_f A$

その他の安定性の証明方法

安定 コスト (評価関数)

$$V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N))$$

終端コストは用いない

安定

安定

安定

安定性を満足するまで Horizon の長さをのばしていく。

G. Grimm, M. J. Messina, S. E. Tuna and A. R. Teel, "Model Predictive Control: For Want of a Local Control Lyapunov Function, All is Not Lost," *IEEE Trans. Automatic Control*, Vol.50, No.5, pp.546-558, 2005.

A. Jadbabaie and J. Hauser, "On the Stability of Receding Horizon Control With a General Terminal Cost," *IEEE Trans. Automatic Control*, Vol.50, No.5, pp.674-678, 2005.

* アウトライン

第 I 部

- 非線形モデル予測制御
 - 非線形モデル予測制御の安定性
 - 非線形モデル予測制御の応用例

第 II 部

協調制御

- 合意 (consensus) 問題
- 群れ (Flocking)

制御システムの階層構造

制御システムの階層構造

システム全体の目標

上位レベルでの最適化 (自己組織・適応)

最適化制御

直接(ローカル)制御

制御対象

問題点

制御手法

モデル予測制御

最適化

ロバスト制御

フィードバック

設定値

操作量

制約(拘束), 故障

不確かさ

Fig: モデル予測制御のブロック線図

非線形モデル予測制御の応用例

非線形モデル予測制御の応用例

- Hover Craft (大阪大学)
- Visual Feedback System (東工大)

モデル予測制御
プロセス制御の分野ではない
"速い" 制御対象への適用へ!!

Fig: Hover Craft

Fig: Visual Feedback System

SICEセミナー「モデル予測制御理論と協調制御の展開」

モデル予測制御の応用例 (Hover Craft)

Tokyo Institute of Technology

• Hover Craft

制御目標
 推進力、移動できる範囲に制約がある条件下で Hover Craftの位置、姿勢を目標値に収束させる。

システム

$$\frac{dx}{dt} = \begin{bmatrix} -\sin x_3 & -\sin x_3 \\ \cos x_3 & \cos x_3 \\ 1/L & -1/L \end{bmatrix} u$$

状態 $x = [x_1, x_2, x_3]^T$
 入力 $u = [u_1, u_2]^T$




Fig : Hover Craft

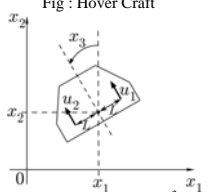


Fig : Hover Craft モデル

T. Ohothuka and A. Kodama, "Automation Code Generation System for Nonlinear Receding Horizon Control," 計測自動制御学会論文集, Vol.38, No.7, pp.617-623 2002. http://www-newton.mech.eng.osaka-u.ac.jp/~ohtsuka/paper/CCA02ohp_hover.pdf

モデル予測制御の応用例 (Hover Craft)

Tokyo Institute of Technology

入力と状態に関する制約

入力に関する制約 U : ファンの推進力 $-u_{\max} \sim u_{\max}$
 状態に関する制約 \mathcal{X} : 移動できる範囲 $-x_{\max} \sim x_{\max}$

コスト (評価関数)

$$J = \phi(x(T)) + \int_t^{t+T} L(x(t'), u(t')) dt'$$

$L = \frac{1}{2} \{ (p(t) - x(t))^T Q (p(t) - x(t)) + u^T u \}$: ステージコスト (stage cost)
 $\phi = \frac{1}{2} (p(t) - x(t))^T S_f (p(t) - x(t))$: 終端コスト (terminal cost)
 $Q > 0, S_f > 0$
 $p(t) = [0 \ 0 \ \frac{\pi}{2}]$: 目標値
 $T(t) = T_f (1 - e^{-\sigma t})$: 評価区間

計算アルゴリズム
C/GMRES

モデル予測制御の応用例 (Hover Craft)

Tokyo Institute of Technology



$\pi/2$ [rad] CCW Rotated Position

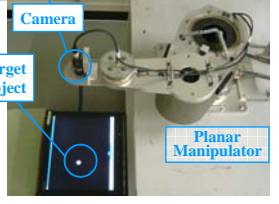
<http://www-newton.mech.eng.osaka-u.ac.jp/~ohtsuka/research.htm>

モデル予測制御の応用例 (Visual Feedback Control)

Tokyo Institute of Technology

• Visual Feedback Control

制御目標
 視覚情報を用いて、マニピュレータの手先を観測対象の真上に一致させる。



システム

$$\begin{cases} \dot{\xi} = -M^{-1}(q)C(q, \dot{q})\xi + M^{-1}(q)u \\ \dot{f} = -\frac{s\lambda}{z_{wp}} R_{wc}^T J_b \xi - \left(\frac{s\lambda}{z_{wp}} R_{wc}^T J_b \omega_b^T R_{wc} + R_{wc}^T \dot{R}_{wc} \right) f \end{cases}$$

Fig : Visual Feedback System

$Q > 0 \ R^{-1} > 0$
 $V(x)$: エネルギー関数

コスト (評価関数)

$$J_{RHC}(u, t) = \int_t^{t+T} l(x(\tau), u(\tau)) d\tau + M(x(t+T))$$

$l(x(\tau), u(\tau)) = 4x^T Q x + u^T R^{-1} u$: ステージコスト (stage cost)
 $M(x(t+T)) = 4V(x(t+T))$: 終端コスト (terminal cost)

モデル予測制御の応用例 (Visual Feedback Control)

Tokyo Institute of Technology



* 非線形モデル予測制御の応用例

Tokyo Institute of Technology

非線形モデル予測制御の応用例

- Caltech ducted fan (California Institute of Technology)
- Alice - the DARPA Grand Challenge (California Institute of Technology)
- F-16 - DARPA SEC program (University of Minnesota) (University of California, Berkeley)

モデル予測制御
経路生成など上位の制御への適用




Fig: Caltech ducted fan




Fig: Alice




Fig: F-16

SICEセミナー「モデル予測制御理論と協調制御の展開」

*** モデル予測制御の応用例 (Caltech ducted fan)**

Caltech Institute of Technology

Caltech ducted fan

制御目標
 入力(推進力)の大きさに制約がある条件下で、中央のスタンドを軸に回転運動を行いつつ、上下に動きながら翼の位置、姿勢を目標値に収束させる。

システム

$$\begin{cases} m\ddot{x} + F_{x_a} - F_{x_b} \cos\theta - F_{z_b} \sin\theta = 0 \\ m\ddot{z} + F_{z_a} + F_{x_b} \sin\theta - F_{z_b} \cos\theta = mg \\ J\ddot{\theta} - M_a + \frac{1}{r_s} I_p \Omega \dot{x} \cos\theta - F_{z_b} l_\tau = 0 \end{cases}$$

状態 $x = [x, z, \theta, \dot{x}, \dot{z}, \dot{\theta}]^T$
 入力 $u = [F_{x_b}, F_{z_b}]^T$

Fig: Caltech ducted fan
 ducted fanの慣性モーメント J
 プロペラの慣性モーメント I_p

Fig: 翼の断面図

<http://www.cds.caltech.edu/~7Emilam/research/res.htm>

37 Fujita Laboratory

*** モデル予測制御の応用例 (Caltech ducted fan)**

Caltech Institute of Technology

入力と状態に関する制約

入力に関する制約 \mathcal{U} : 与える力 F_{x_b} 0 [N] ~ 13 [N]
 F_{z_b} -6.5 [N] ~ 6.5 [N]

状態に関する制約 \mathcal{X} : 高さ z 0 [m] ~ 2.5 [m]

コスト (評価関数)

$$J(x) = \int_0^T q(x(\tau), u(\tau)) d\tau + V(x(T))$$

$$q(x(t), u(t)) = \frac{1}{2} x_{err}^T(t) Q x_{err}(t) + \frac{1}{2} u_{err}^T(t) R u_{err}(t)$$

: ステージコスト (stage cost)
 $V(x(T)) = \frac{1}{2} x_{err}^T(T) P x_{err}(T)$: 終端コスト (terminal cost)

$x_{err} \equiv x - x_{eq}$ $u_{err} \equiv u - u_{eq}$: 偏差 $Q > 0, R > 0$
 $x_{eq} = [x_{cmd}, z_{cmd}, \theta_{cmd}, 0, 0, 0]^T$ $u_{eq} = [mg, 0]^T$: 平衡状態

M. B. Milam, R. Franz, J. E. Hauser and R. M. Murray, "Receding horizon control of vectored thrust flight Experiment," *IEE Proc. Control Theory Appl.*, Vol.152, No.3, pp.340-348, 2005.

38 Fujita Laboratory

*** モデル予測制御の応用例 (Caltech ducted fan)**

Caltech Institute of Technology

安定性 → 終端コストにCLF (Control Lyapunov Function)

CLF: つぎを満たす正定関数 $V: R^n \rightarrow R_+$
 $\inf_u [\dot{V}(x, u) + q(x, u)] \leq 0$

終端コスト $V(x(T)) = \frac{1}{2} x_{err}^T(T) P x_{err}(T)$ のCLFの求め方

1. システムを平衡点近傍で線形化 → LQRに基づくCLF
 計算時間からホライズンを長く取れる場合に有効
 ducted fan では $T \geq 0.6$

平衡点

ホライズン長い → $x_{err}^o(T)$ が平衡点近傍へ
 LQRに基づくCLF ≒ コストの最適値 J^o

2. 状態変数をパラメータとしLMIsを解く → LPVに基づくCLF
 (Linear Matrix inequalities)
 計算時間からホライズンを長く取れない場合に有効

39 Fujita Laboratory

*** モデル予測制御の応用例 (Caltech ducted fan)**

Caltech Institute of Technology

Differential Flatnessによる最適化

State space $x(t), u(t)$
 $x(0), u(0)$

Flat space $z = [z_1, z_2, \dots, z_q]^T$
 フラット出力 $z = A(x, u, u^{(1)}, \dots, u^{(r)})$

Software
 NTG (Nonlinear Trajectory Generation)
 State space 動的な空間
 $\dot{x}(t) = f(x(t), u(t))$
 $u(t) \in \bar{U}$

Flat space 低次元化 静的な空間
 $(x, u) = B(z, z^{(1)}, \dots, z^{(s)})$

フラット出力の最適化によって、軌道計画とそのために必要な入力が求まる。

実時間軌道生成

$z_1 = \sum_{i=1}^{p_1} B_{i,k_1}(t) C_1^i$ $\min F(y)$ s.t. $lb \leq c(y) \leq ub$
 $z_2 = \sum_{i=1}^{p_2} B_{i,k_2}(t) C_2^i$ $y = (C_1, C_2, \dots, C_q)$ 係数の最適化
 (計算時間の短縮)
 $z_q = \sum_{i=1}^{p_q} B_{i,k_q}(t) C_q^i$ B-spline 基底関数

40 Fujita Laboratory

*** モデル予測制御の応用例 (Caltech ducted fan)**

Caltech Institute of Technology



<http://www.cds.caltech.edu/~murray/>

41 Fujita Laboratory

*** Real-Time Trajectory Generation**

Caltech Institute of Technology

Real-Time Trajectory Generation (リアルタイムでの経路生成)

目的
 現在の位置から目的地までの通過可能な経路をリアルタイムで生成すること

制約

- 通過可能経路の生成
- スピード
- 障害物回避
- ステアリング etc

問題点

- 計算時間
- NTG Software Library
- C/GMRES
- 最適化計算の可解性

Global | Local

モデル予測制御

Planning

Fig: Planningのブロック線図

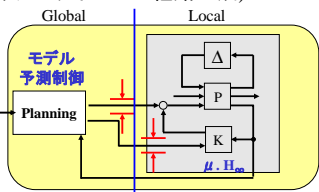



Fig: Alice (Team Caltech)



<http://www.parody.org/2005/pictures/darpa/index.html>

42 Fujita Laboratory

SICEセミナー「モデル予測制御理論と協調制御の展開」

*** Alice**
Tokyo Institute of Technology

Real-Time Trajectory Generationの使用例
Racing to win the DARPA Grand Challenge
URL : <http://www.darpa.mil/grandchallenge/index.asp>
Team Caltech : Alice URL: <http://team.caltech.edu/index.html>
2005 Ford E-350 Van
5台のカメラ
・2組のステレオカメラ+ roadfinding



Fig. Alice
5つのレーダー
2つのGPS

*** Alice**
Tokyo Institute of Technology



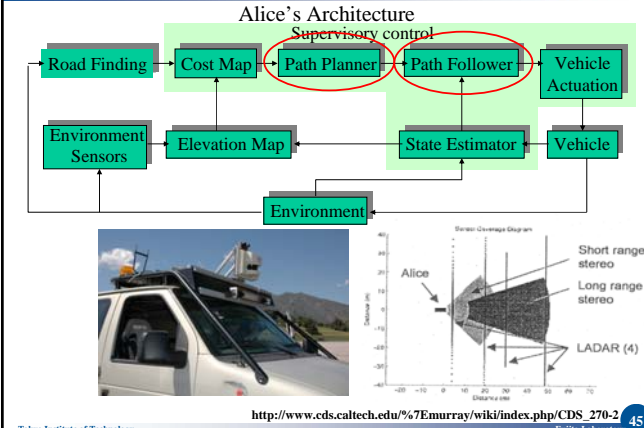
<http://team.caltech.edu/index.html>

*** Alice**
Tokyo Institute of Technology

Alice's Architecture
Supervisory control

```

    graph TD
      subgraph Supervisory_control
        RF[Road Finding] --> CM[Cost Map]
        CM --> PP[Path Planner]
        PP --> PF[Path Follower]
        PF --> VA[Vehicle Actuation]
      end
      ES[Environment Sensors] --> EM[Elevation Map]
      EM --> CM
      ES --> SE[State Estimator]
      SE --> PF
      SE --> V[Vehicle]
      V --> SE
      V --> ES
      E[Environment] --> ES
      E --> EM
      E --> SE
  
```



http://www.cds.caltech.edu/~Emurray/wiki/index.php/CDS_270-2

*** Alice**
Tokyo Institute of Technology

Real-Time Trajectory Generation

$$\arg \min \int_t^{t+T} L(x,u) d\tau + V(x(T))$$

ex.

$$L(x,u) = 1 \quad V(x(T)) = 0$$

$$\arg \min \int_t^{t+T} L(x,u) d\tau + V(x(T)) = \min T$$

所要時間の最小化
入力最小化

$$L(x,u) = u_1^2 + u_2^2$$

s.t

ダイナミクス

$$\begin{cases} \dot{x} = v \cos \theta \\ \dot{y} = v \sin \theta \\ \dot{\theta} = \frac{v}{L} \tan \phi \\ \dot{\phi} = u_1 \\ \dot{v} = u_2 \end{cases}$$

制約

$$\begin{cases} \phi \in [\phi_{\min}, \phi_{\max}] \\ u_1 \in [u_{1\min}, u_{1\max}] \\ u_2 \in [u_{2\min}, u_{2\max}] \\ v \in (0, v_{\max}] \end{cases}$$

障害物回避

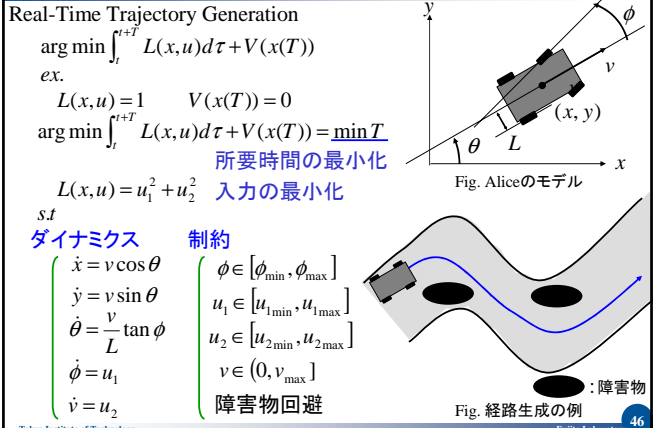
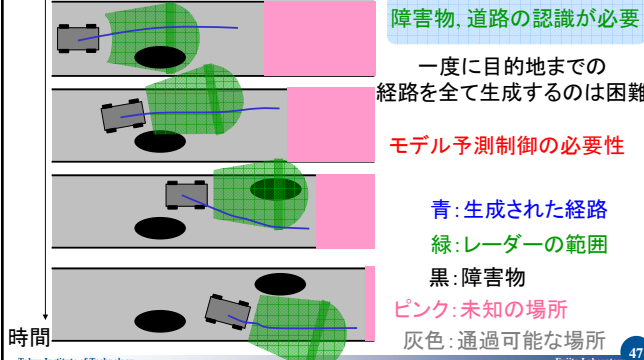


Fig. Aliceのモデル
Fig. 経路生成の例

*** Alice**
Tokyo Institute of Technology

Real-Time Trajectory Generation
モデル予測制御 オンラインで徐々に経路生成

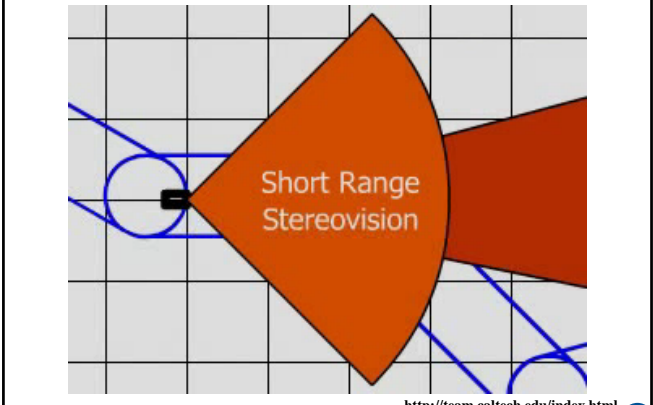


障害物, 道路の認識が必要
一度に目的地までの経路を全て生成するのは困難
モデル予測制御の必要性

青: 生成された経路
緑: レーダーの範囲
黒: 障害物
ピンク: 未知の場所
灰色: 通過可能な場所

時間

*** Alice**
Tokyo Institute of Technology




Short Range Stereovision

<http://team.caltech.edu/index.html>

SICEセミナー「モデル予測制御理論と協調制御の展開」

* モデル予測制御の応用例 (F-16)

Tokyo Institute of Technology



非線形モデル $x_{NL} = [h, \theta, V_t, \alpha, q]^T$
 $\dot{x}_{NL} = f(x_{NL}, \delta)$

線形化・離散化

$x(k+1) = A_k x(k) + B_k r(k)$
 $y(k) = C_k x(k) + D_k r(k)$

パラメータベクトル $\rho(k)$ $\rho(k) = [\bar{q}(k) M(k)]^T$

$A_k = A(\rho(k))$ $B_k = B(\rho(k))$
 $C_k = C(\rho(k))$ $D_k = D(\rho(k))$

$\bar{q}(k)$: 動圧 (正面から受ける空気の圧力)
 $M(k)$: マッハ数 (音速に対する飛行機速度)

Fig: F-16
 $y = \begin{bmatrix} h \\ V_t \end{bmatrix}$ $r = \begin{bmatrix} \delta_{th\ cmd} \\ q_{cmd} \end{bmatrix}$

h : 高度
 θ : ピッチ角
 V_t : 対気速度
 α : 迎え角
 q : ピッチ角速度

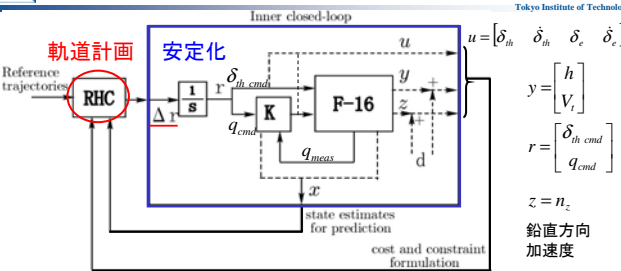
F-16への入力 $u = [\delta_{th} \ \delta_{\dot{\theta}} \ \delta_c \ \delta_{\dot{c}}]^T$
 δ_{th} : 推力 δ_c : 昇降舵角
 (速度制御) (姿勢制御)

students.ceid.upatras.gr/
 -kouroupa/Images/F16.jpg

49

* モデル予測制御の応用例 (F-16)

Tokyo Institute of Technology



軌道計画 (RHC) → 安定化 (Inner closed-loop) → F-16

Reference trajectories r → Δr → $\frac{1}{s}$ → r → K → q_{cmd} → F-16

state estimates for prediction $\hat{y}(k+i|k)$

cost and constraint formulation

outer-loop: モデル予測制御による目標軌道に対する最適な入力の生成
 inner-loop: 線形化されたシステムに対して2自由度制御系の目標値への追従制御

制御目標: 入力や状態の制約を満たしつつ、目標の高度・速度に保つ、 $y \rightarrow y_{ref}$

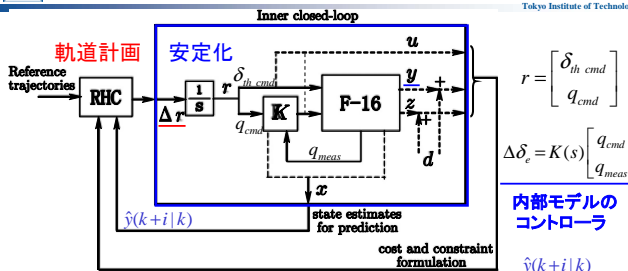
入力制約 $1000 \leq \delta_{th} \leq 19000$ (lb)
 $-10000 \leq \delta_c \leq 10000$ (lb/s)
 $-25 \leq \delta_{\dot{c}} \leq 25$ (°)
 $-60 \leq \delta_{\dot{c}} \leq 60$ (°/s)

鉛直方向加速度

50

* モデル予測制御の応用例 (F-16)

Tokyo Institute of Technology



軌道計画 (RHC) → 安定化 (Inner closed-loop) → F-16

Reference trajectories r → Δr → $\frac{1}{s}$ → r → K → q_{cmd} → F-16

state estimates for prediction $\hat{y}(k+i|k)$

cost and constraint formulation

内部モデルのシステム (linear prediction model)

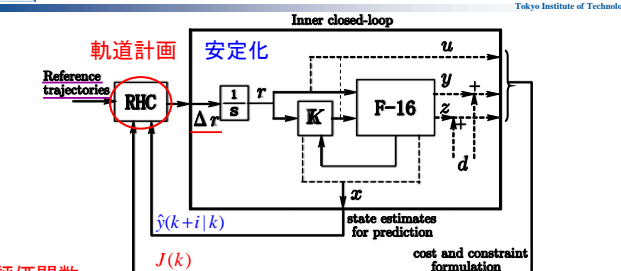
$\hat{\xi}(k+1) = A_k \hat{\xi}(k) + B_k \Delta r(k)$
 $\hat{\xi}(k) = \begin{bmatrix} \hat{x}(k) \\ \hat{d}(k) \end{bmatrix}$ $\hat{w}(k) = \begin{bmatrix} \hat{y}(k) \\ \hat{z}(k) \end{bmatrix}$
 $\hat{w}(k) = C_k \hat{\xi}(k) + D_k \Delta r(k)$
 $\hat{d}(k+1) = \hat{d}(k) + \frac{y(k) - \hat{y}(k|k-1)}{z(k) - \hat{z}(k|k-1)}$

内部モデルのコントローラ $\Delta \delta_c = K(s) \begin{bmatrix} q_{cmd} \\ q_{meas} \end{bmatrix}$

51

* モデル予測制御の応用例 (F-16)

Tokyo Institute of Technology



軌道計画 (RHC) → 安定化 (Inner closed-loop) → F-16

Reference trajectories r → Δr → $\frac{1}{s}$ → r → K → q_{cmd} → F-16

state estimates for prediction $\hat{y}(k+i|k)$

cost and constraint formulation

評価関数 $J(k) = \sum_{i=1}^{H_c} \|\hat{y}(k+i|k) - y_{ref}(k+i|k)\|_Q^2 + \sum_{i=0}^{H_p-1} \|\Delta r(k+i|k)\|_R^2 + \rho \epsilon$

$i=0, \delta_{th}, 2\delta_{th}, \dots, H_c-1$ 制御ホライゾン
 $i=1, 2, \dots, H_p$ 予測出力ホライゾン

52

最新のモデル予測制御の研究

Tokyo Institute of Technology

最新のモデル予測制御の研究

- 非線形モデル予測制御 (nonlinear model predictive control)
- ロバストモデル予測制御 (robust model predictive control)
- ‘速い’システム (fast systems) のモデル予測制御
- ハイブリッドシステム (hybrid systems) のモデル予測制御
- Trajectory Generation
- Path Planning

URL: http://www.fl.ctrl.titech.ac.jp/paper/2006/SICE_seminar_06Fujita.pdf

53

* 制御理論の最前線

Tokyo Institute of Technology

45th IEEE Conference on Decision and Control, 2006



45th IEEE Conference on Decision and Control, San Diego, CA, USA, Dec. 13-15, 2006

Workshop Rolf Findeisen
 “Model Predictive Control for Fast Nonlinear Systems: Existing Approaches, Challenges and Applications”


Plenary lecture A. Stephen Morse
 “Multi-Agent Formations and Sensor Networks”

54

SICEセミナー「モデル予測制御理論と協調制御の新展開」

Tokyo Institute of Technology

SICE セミナー
「実践的な制御系設計の最前線
-ポストロバスト制御理論と応用の新展開-」
「モデル予測制御理論と協調制御と協調制御の新展開」

第II部


2006年10月27日
電気通信大学スカイオフィス

東京工業大学 藤田政之

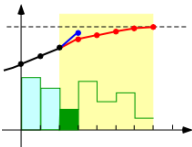
Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

*** アウトライン**

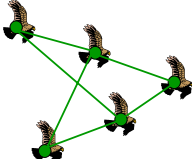
第I部

- ✓ 非線形モデル予測制御
 - ・ 非線形モデル予測制御の安定性
 - ・ 非線形モデル予測制御の応用例



第II部

- ✚ 協調制御
 - ・ 合意 (consensus) 問題
 - ・ 群れ (Flocking)



Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

*** はじめに**

協調制御: Cooperative Control

- 合意: Consensus
- 群れ: Flocking

応用

- フォーメーション制御 (Formation Control)
- ロボットセンサネットワーク (Robotic Sensor Network)
- 交通管制 (Traffic Control)

上位階層の制御問題



<http://www.grampus.co.jp/report/images/warasa.jpg>
http://sotoasobi.seesaa.net/media/img_20050401T074306125.jpg

http://www.ec.kth.se/~kallej/control_over_wireless_benelux06.pdf
<http://www.path.berkeley.edu/>

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

はじめに

協調制御と分散制御

- 協調制御: 各エージェントの**目的**が他のエージェントに関連して決まる
- 分散制御: 各エージェントの**制御入力**が他のエージェントに関連して決まる

協調・分散制御の例

- 自律分散システム
- センサネットワーク
- ITS/管制
- フォーメーション制御
- ランデブー (Rendezvous)
- 合意 (Consensus)
- 群れ (Flocking) ...

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

合意 (Consensus) 問題

エージェント (制御対象)

$$\dot{x} = u$$

$$x \in R \quad u \in R$$

エージェントモデルの例

運動学モデル (速度入力)

動的モデル

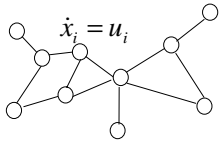
- 2重積分系 $\dot{q} = p \quad \dot{p} = u$
- 状態方程式 $\dot{x} = Ax + Bu$
- 非線形系 $\dot{x} = f(x) + g(x)u$

Tokyo Institute of Technology Fujita Laboratory

Tokyo Institute of Technology

合意 (Consensus) 問題

複数のエージェント (n 台) いる状況

$$\begin{matrix} \dot{x}_1 = u_1 \\ \dot{x}_2 = u_2 \\ \vdots \\ \dot{x}_n = u_n \end{matrix} \Rightarrow \begin{matrix} \dot{x}_i = u_i \quad (i=1, \dots, n) \\ x_i \in R \quad u_i \in R \end{matrix}$$


分散制御を考える上で重要なこと

どのエージェントの情報が利用できるか?

グラフ: G エージェント間の結合をあらわしたもの

近傍: N_i エージェント i が利用できる情報をもつエージェントの集合

Tokyo Institute of Technology Fujita Laboratory

SICEセミナー「モデル予測制御理論と協調制御の展開」

グラフ理論

Tokyo Institute of Technology

グラフ理論
 グラフ $G = (V(G), E(G))$:
 点あるいは頂点とよばれる有限集合 $V(G)$ と、辺と呼ばれる $V(G)$ の相異なる2つの要素の非順序対集合 $E(G)$ とから構成されるもの
 頂点集合 $V(G) := \{1, \dots, n\}$
 辺集合 $E(G) \subseteq V(G) \times V(G)$

近傍 N_i : ある点の隣接点の集合

グラフの例

$V(G) = \{1, 2, 3, 4, 5\}$
 $E(G) = \{(1,2), (2,3), (2,4), (3,5), (4,5)\}$
 $N_1 = \{2\}$
 $N_2 = \{1, 3, 4\} \dots$

$V(G) = \{1, 2, 3, 4\}$
 $E(G) = \{(1,2), (2,3), (3,4), (4,5)\}$
 $N_1 = \{2\}$
 $N_2 = \{1, 3\} \dots$

61

グラフ理論

Tokyo Institute of Technology

グラフ理論
 グラフの種類

- 無向グラフ: グラフの辺に矢印がないグラフ
- 有向グラフ: グラフの辺に矢印があるグラフ
- 連結グラフ: ひとつにまとまったグラフ

無向グラフ 有向グラフ 非連結グラフ

平衡グラフ

62

制御目的

Tokyo Institute of Technology

制御目的
 合意(Consensus)を達成すること
 すべてのエージェントの状態がある値に一致すること
 $x_1 = x_2 = \dots = x_n = \alpha \quad t \rightarrow \infty$
 $x_i = x_j = \alpha \quad t \rightarrow \infty \quad \forall i, j$

α : 一致する場所
 制御目的が周りのエージェントと関係: **協調制御**

例 ヘリコプターの高さ制御

63

制御入力

Tokyo Institute of Technology

各エージェントに加える入力(制御入力)

$$u_i = - \sum_{j \in N_i} (x_i - x_j) \quad \text{分散制御}$$

自分以外のエージェントの状態との差
 近傍のエージェントとの差

$u = -Lx$ L : グラフラプラシアン

例

$$u_1 = - \sum_{j \in N_1} (x_i - x_j) = -(x_1 - x_2) \quad N_1 = \{2\}$$

$$u_2 = - \sum_{j \in N_2} (x_i - x_j) = -(x_2 - x_1) - (x_2 - x_3) - (x_2 - x_4) \quad N_2 = \{1, 3, 4\}$$

$$u_3 = - \sum_{j \in N_3} (x_i - x_j) = -(x_3 - x_2) - (x_3 - x_5) \quad N_3 = \{2, 5\}$$

$$u_4 = - \sum_{j \in N_4} (x_i - x_j) = -(x_4 - x_2) - (x_4 - x_5) \quad N_4 = \{2, 5\}$$

$$u_5 = - \sum_{j \in N_5} (x_i - x_j) = -(x_5 - x_3) - (x_5 - x_4) \quad N_5 = \{3, 4\}$$

64

グラフ理論

Tokyo Institute of Technology

グラフ理論
 グラフを特徴づける行列

- 隣接行列(adjacency matrix)
- 次数行列(degree matrix)
- グラフラプラシアン(graph laplacian)

隣接行列(adjacency matrix): A

$$A := a_{ij} \begin{cases} 1: \text{頂点 } i \text{ と } j \text{ が隣接している} \\ 0: \text{その他} \end{cases}$$

例

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

65

グラフ理論

Tokyo Institute of Technology

グラフ理論
 グラフを特徴づける行列

- 次数行列(degree matrix): D

$$D := \begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_n \end{bmatrix}$$

d_i : 頂点 i の次数
 次数: 無向グラフの各点に辺で結ばれている頂点の個数

例

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

66

SICEセミナー「モデル予測制御理論と協調制御の展開」

グラフ理論 Tokyo Institute of Technology

グラフ理論
 グラフを特徴づける行列
 * グラフラプラシアン(graph laplacian): L

$L := D - A$ D : 次数行列 A : 隣接行列

例

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

$$L = D - A = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 3 & -1 & -1 & 0 \\ 0 & -1 & 2 & 0 & -1 \\ 0 & -1 & 0 & 2 & -1 \\ 0 & 0 & -1 & -1 & 2 \end{bmatrix}$$

Tokyo Institute of Technology Fujita Laboratory 67

グラフ理論 Tokyo Institute of Technology

グラフ理論
 グラフを特徴づける行列
 * グラフラプラシアン(graph laplacian): L

$L := D - A$ D : 次数行列 A : 隣接行列

例

$$D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$L = D - A = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

Tokyo Institute of Technology Fujita Laboratory 68

クロネッカー積 Tokyo Institute of Technology

クロネッカー積
 $A \in R^{m \times n}$ $B \in R^{p \times q}$ についてクロネッカー積を次のように定義する.

$$A \otimes B := \begin{bmatrix} a_{11}B & a_{12}B & \dots & a_{1n}B \\ a_{21}B & a_{22}B & \dots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \dots & a_{mn}B \end{bmatrix} \in R^{mp \times nq}$$

例

$$A = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

$$B = I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A \otimes B = \begin{bmatrix} I_3 & -I_3 & 0I_3 \\ -I_3 & 2I_3 & -I_3 \\ 0I_3 & -I_3 & I_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 2 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}$$

Tokyo Institute of Technology Fujita Laboratory 69

閉ループ系 Tokyo Institute of Technology

閉ループ系
 $\begin{cases} \dot{x}_i = u_i \\ u_i = -\sum_{j \in N_i} (x_i - x_j) \end{cases}$ すべてのエージェントについてまとめると

$$\dot{x}_i = -\sum_{j \in N_i} (x_i - x_j) \Rightarrow \dot{x} = -Lx \quad x = [x_1, x_2, \dots, x_n]^T$$

例

$$u_1 = -\sum_{j \in N_1} (x_1 - x_j) = -(x_1 - x_2) \quad N_1 = \{2\}$$

$$u_2 = -\sum_{j \in N_2} (x_2 - x_j) = -(x_2 - x_1) - (x_2 - x_3) \quad N_2 = \{1, 3\}$$

$$u_3 = -\sum_{j \in N_3} (x_3 - x_j) = -(x_3 - x_2) \quad N_3 = \{2\}$$

$$\begin{cases} \dot{x}_1 = -(x_1 - x_2) \\ \dot{x}_2 = -(x_2 - x_1) - (x_2 - x_3) \\ \dot{x}_3 = -(x_3 - x_2) \end{cases} \Rightarrow \dot{x} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = -Lx$$

Tokyo Institute of Technology Fujita Laboratory 70

グラフ理論 Tokyo Institute of Technology

グラフ理論
 グラフラプラシアンの性質
 * グラフラプラシアン: L

無向グラフの場合, 準正定・対称行列となる.
 $L^T = L \quad L \geq 0$

無向グラフの場合, 二次形式が状態の偏差の二乗和となる.
 $x^T Lx = \frac{1}{2} \sum_{i=1}^n \sum_{j \in N_i} (x_i - x_j)^2 \geq 0$

連結グラフの場合, $\text{rank } L = n - 1$
 どのようなグラフにおいても, 次の式が成り立つ

$$L\bar{1} = L \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \quad \bar{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

Tokyo Institute of Technology Fujita Laboratory 71

グラフ理論 Tokyo Institute of Technology

グラフ理論
 グラフラプラシアンの性質
例

$$L = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

無向グラフの場合, 二次形式が状態の偏差の二乗和となる.

$$x^T Lx = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$$= x_1^2 + 2x_2^2 + x_3^2 - 2x_1x_2 - 2x_2x_3$$

$$= x_1^2 - 2x_1x_2 + x_2^2 + x_2^2 - 2x_2x_3 + x_3^2$$

$$= (x_1 - x_2)^2 + (x_2 - x_3)^2$$

$$= \frac{1}{2}(x_1 - x_2)^2 + \frac{1}{2}(x_2 - x_1)^2 + \frac{1}{2}(x_2 - x_3)^2 + \frac{1}{2}(x_3 - x_2)^2$$

$$= \frac{1}{2} \sum_{i=1}^3 \sum_{j \in N_i} (x_i - x_j)^2$$

Tokyo Institute of Technology Fujita Laboratory 72

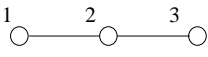
SICEセミナー「モデル予測制御理論と協調制御の展開」

グラフ理論

Tokyo Institute of Technology

グラフ理論

グラフラプラジアンの性質

例 $L = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$ 

どのようなグラフにおいても、次の式が成り立つ

$$L\bar{1} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$L\bar{1} = L \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1-1 \\ -1+2-1 \\ -1+1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Tokyo Institute of Technology Fujita Laboratories 73

仮定

Tokyo Institute of Technology

仮定

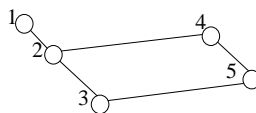
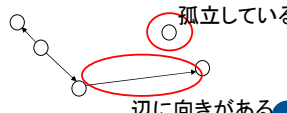
グラフは無向, 連結, 固定

➡ グラフラプラジアン L は次のような性質を持つ

$$\text{rank } L = n-1 \quad L \geq 0 \quad L^T = L$$

仮定の工学的な意味

- 無向: エージェントの通信が常に双方向
- 連結: 必ずどれかのエージェントと通信を行う
- 固定: 通信相手が固定

例 無向, 連結  仮定を満たさない例 有向, 非連結  孤立している

Tokyo Institute of Technology Fujita Laboratories 74

証明

Tokyo Institute of Technology

収束性

閉ループ系 $\dot{x} = -Lx$ は $t \rightarrow \infty$ で $x = \alpha [1 \ \dots \ 1]^T$ に収束する

証明

ステップ1 $V(x) \geq 0$ を決める.

$$V(x) = \frac{1}{2} x^T Lx = \frac{1}{2} \sum_{i=1}^n \sum_{j \in N_i} \|x_i - x_j\|^2$$

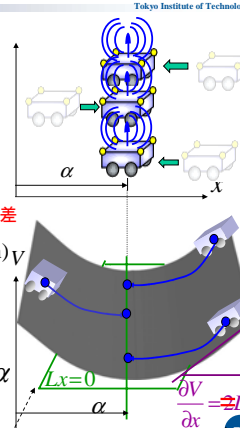
と決める = エージェント間の偏差

$V(x)$: 不一致関数 (disagreement function) V

エージェント間の偏差が大きい $\leftrightarrow V(x)$ が大きい

エージェント間の偏差が小さい $\leftrightarrow V(x)$ が小さい

$V(x) = 0$ となるのは $x_1 = x_2 = \dots = x_n = \alpha$ のみ \leftrightarrow 制御目標達成



Tokyo Institute of Technology Fujita Laboratories 75

証明

Tokyo Institute of Technology

証明

ステップ2 $\dot{V}(x) \leq 0$ どうか調べる

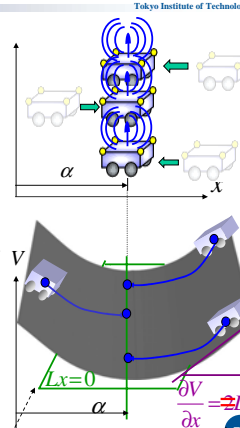
$$\begin{aligned} \dot{V}(x) &= \frac{1}{2} \dot{x}^T Lx + \frac{1}{2} x^T L \dot{x} \\ &= -x^T L Lx \quad \dot{x} = -Lx \\ &= -(Lx)^T Lx \leq 0 \end{aligned}$$

$$\dot{V}(x) = \nabla V(x)^T \dot{x} = \frac{\partial V(x)}{\partial x} \dot{x}$$

$$\frac{\partial V(x)}{\partial x} = \frac{\partial}{\partial x} \left(\frac{1}{2} x^T Lx \right) = Lx = -u \quad u = -Lx$$

したがって閉ループ系 $\dot{x} = -\nabla V(x)$

エージェントをポテンシャルの坂を下るように進めると見ることができる。



Tokyo Institute of Technology Fujita Laboratories 76

証明

Tokyo Institute of Technology

ステップ3 $\dot{V}(x) = 0$ となる x を求める

$-(Lx)^T Lx = 0$ (ポテンシャルの傾きが0)

$$Lx = 0$$

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \alpha \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \quad \text{rank } L = n-1 \quad L\bar{1} = 0$$

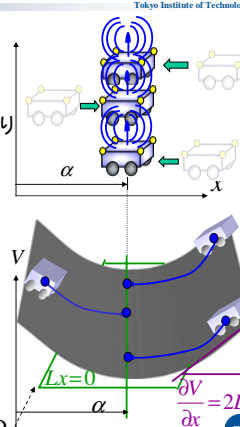
$x_1 = x_2 = \dots = x_n = \alpha$

ステップ4 不変集合を求める

$x_1 = x_2 = \dots = x_n = \alpha$ のとき $u = -Lx = 0$ より $x_1 = x_2 = \dots = x_n = \alpha$ が不変集合

すべてのエージェントが同じ位置に収束

Q.E.D.



Tokyo Institute of Technology Fujita Laboratories 77

証明

Tokyo Institute of Technology

すべてのエージェントが同じ位置に収束

どこに?

合意 (Consensus) の場所 各エージェントの初期状態の平均値

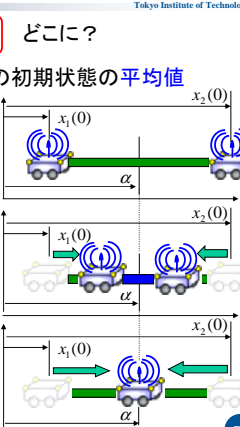
$$\alpha = \frac{1}{n} \sum_{i=1}^n x_i(0) \quad \text{average consensus}$$

証明 各エージェントの状態の平均値が 時間に対して不変である ことを使う

不変量 $\frac{1}{n} \sum_{i=1}^n x_i = \frac{1}{n} [1 \ \dots \ 1] \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \frac{1}{n} [1 \ \dots \ 1] x$

これを微分すると $\frac{1}{n} [1 \ \dots \ 1] \dot{x} \xrightarrow{\text{green}} -\frac{1}{n} [1 \ \dots \ 1] Lx = 0$

よって, 平均値は時間で変化しない。

$$\frac{1}{n} \sum_{i=1}^n x_i(0) = \frac{1}{n} \sum_{i=1}^n \alpha = \alpha \quad \alpha = \frac{1}{n} \sum_{i=1}^n x_i(0)$$


Tokyo Institute of Technology Fujita Laboratories 78

SICEセミナー「モデル予測制御理論と協調制御の展開」

例題1

Tokyo Institute of Technology

5台のエージェント

$$\begin{cases} \dot{x}_1 = u_1 \\ \dot{x}_2 = u_2 \\ \dot{x}_3 = u_3 \\ \dot{x}_4 = u_4 \\ \dot{x}_5 = u_5 \end{cases}$$

グラフ

接続行列
隣接

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

次数行列

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

グラフラプラシアン
 $L := D - A$

$$L = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

79

例題1

Tokyo Institute of Technology

制御入力

$$u = -Lx = - \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

$u_1 = -\sum_{j \in N_1} (x_i - x_j) = -(x_1 - x_2)$ $N_1 = \{2\}$

$u_2 = -\sum_{j \in N_2} (x_i - x_j) = -(x_2 - x_1) - (x_2 - x_3)$ $N_2 = \{1,3\}$

$u_3 = -\sum_{j \in N_3} (x_i - x_j) = -(x_3 - x_2) - (x_3 - x_4)$ $N_3 = \{2,4\}$

$u_4 = -\sum_{j \in N_4} (x_i - x_j) = -(x_4 - x_3) - (x_4 - x_5)$ $N_4 = \{3,5\}$

$u_5 = -\sum_{j \in N_5} (x_i - x_j) = -(x_5 - x_4)$ $N_5 = \{4\}$

80

シミュレーション1

Tokyo Institute of Technology

初期状態

$$\begin{cases} x_1(0) = 10 \\ x_2(0) = 5 \\ x_3(0) = -20 \\ x_4(0) = -3.25 \\ x_5(0) = 2.25 \end{cases}$$

$\alpha = \frac{1}{n} \sum_{i=1}^n x_i(0)$ $n=5$

$$= \frac{1}{5} (x_1(0) + x_2(0) + x_3(0) + x_4(0) + x_5(0))$$

$$= -1.2$$

81

例題2

Tokyo Institute of Technology

5台のエージェント

$$\begin{cases} \dot{x}_1 = u_1 \\ \dot{x}_2 = u_2 \\ \dot{x}_3 = u_3 \\ \dot{x}_4 = u_4 \\ \dot{x}_5 = u_5 \end{cases}$$

グラフ

接続行列
隣接

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

次数行列

$$D = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

グラフラプラシアン
 $L := D - A$

$$L = \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ -1 & 0 & -1 & 3 & -1 \\ 0 & -1 & 0 & -1 & 2 \end{bmatrix}$$

82

例題2

Tokyo Institute of Technology

制御入力

$$u = -Lx = - \begin{bmatrix} 2 & -1 & 0 & -1 & 0 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 2 & -1 & 0 \\ -1 & 0 & -1 & 3 & -1 \\ 0 & -1 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

$u_1 = -\sum_{j \in N_1} (x_i - x_j) = -(x_1 - x_2) - (x_1 - x_4)$ $N_1 = \{2,4\}$

$u_2 = -\sum_{j \in N_2} (x_i - x_j) = -(x_2 - x_1) - (x_2 - x_3) - (x_2 - x_5)$ $N_2 = \{1,3,5\}$

$u_3 = -\sum_{j \in N_3} (x_i - x_j) = -(x_3 - x_2) - (x_3 - x_4)$ $N_3 = \{2,4\}$

$u_4 = -\sum_{j \in N_4} (x_i - x_j) = -(x_4 - x_3) - (x_4 - x_5) - (x_4 - x_1)$ $N_4 = \{3,5,1\}$

$u_5 = -\sum_{j \in N_5} (x_i - x_j) = -(x_5 - x_4) - (x_5 - x_2)$ $N_5 = \{4,2\}$

83

シミュレーション2

Tokyo Institute of Technology

初期状態

$$\begin{cases} x_1(0) = 10 \\ x_2(0) = 5 \\ x_3(0) = -20 \\ x_4(0) = -3.25 \\ x_5(0) = 2.25 \end{cases}$$

$\alpha = \frac{1}{n} \sum_{i=1}^n x_i(0)$ $n=5$

$$= \frac{1}{5} (x_1(0) + x_2(0) + x_3(0) + x_4(0) + x_5(0))$$

$$= -1.2$$

84

SICEセミナー「モデル予測制御理論と協調制御の展開」

収束性の早さとむだ時間の許容値

Tokyo Institute of Technology

合意(Consensus)問題の性質

- 収束性の速さ
 - グラフラプラシアンの2番目に小さい固有値 λ_2 に依存する (グラフラプラシアンの1番目に小さい固有値は常に0)
 - $\|\delta(t)\| \leq \|\delta(0)\| e^{-\lambda_2 t}$ (固有値が大きいほど収束が早い)
 - $\delta(t) := x - \alpha 1$ 合意の場所までの偏差
 - α : 合意(Consensus)の場所
- むだ時間の許容値
 - エージェント間の情報のやり取りに遅れがある場合
 - $u_i(t) = -\sum_{j \in N_i} (x_i(t) - x_j(t)) \rightarrow u_i(t) = -\sum_{j \in N_i} (x_i(t - \tau) - x_j(t - \tau))$
 - τ : むだ時間
 - むだ時間の許容値 $0 \leq \tau \leq \frac{\pi}{2\lambda_{\max}}$ (固有値が大きいほど許容値が少ない)
 - グラフラプラシアンの最も大きい固有値 λ_{\max} に依存

85 Fujita Laboratory

シミュレーションの比較

Tokyo Institute of Technology

シミュレーションの比較

シミュレーション1 $\lambda_2 = 0.3820$

シミュレーション2 $\lambda_2 = 2$

グラフが変わっても合意(Consensus)の場所は変わらない
グラフが密になるほど収束が早い

86 Fujita Laboratory

むだ時間

Tokyo Institute of Technology

しかし 情報のやり取りにむだ時間がある場合

$0 \leq \tau \leq \frac{\pi}{2\lambda_{\max}} = 0.434$ (left) $0 \leq \tau \leq \frac{\pi}{2\lambda_{\max}} = 0.314$ (right)

$\tau = 0.4$ の場合

収束の早さ ← → むだ時間の影響
トレードオフ

87 Fujita Laboratory

連続時間と離散時間の合意(Consensus)問題

Tokyo Institute of Technology

連続時間と離散時間の比較

	連続時間	離散時間
システム	$\dot{x} = u$	$x(k+1) = x(k) + u(k)$
重要な行列	グラフラプラシアン L	確率行列(ペロン行列)* P
入力	$u = -Lx$	$u(k) = -\varepsilon Lx(k)$ $\varepsilon \in (0, d_{\max}^{-1})$ d_{\max} : グラフの最大次数
合意(Consensus)の場所	平均値 $\frac{1}{n} \sum_{i=1}^n x_i(0)$	平均値 $\frac{1}{n} \sum_{i=1}^n x_i(0)$

* 確率行列 各列の要素の和が1に等しい正方の非負行列

88 Fujita Laboratory

グラフラプラシアンとペロン行列の種類

Tokyo Institute of Technology

グラフラプラシアンとペロン行列の種類

	グラフラプラシアン L	ペロン行列 P
R. O. Saber and R. M. Murray	$D - A$	$I - \varepsilon(D - A)$ $\varepsilon \in (0, d_{\max}^{-1})$
J. A. Fax and R. M. Murray	$I - D^{-1}A$	$D^{-1}A$
A. Jadbabaie, J. Lin and A. S. Morse	$I - (I + D)^{-1}(I + A)$	$(I + D)^{-1}(I + A)$

89 Fujita Laboratory

その他の合意(Consensus)問題

Tokyo Institute of Technology

平均値合意(average consensus)以外の合意(Consensus)問題

	合意(Consensus)の場所	各エージェントの入力
算術的 (Arithmetic)	$\frac{1}{n} \sum_{i=1}^n x_i(0)$	$u_i = -\sum_{j \in N_i} (x_i - x_j)$
幾何学的 (Geometric)	$\sqrt[n]{\prod_{i=1}^n x_i(0)}$	$u_i = -x_i \sum_{j \in N_i} (x_i - x_j)$
調和的 (harmonic)	$\frac{n}{\sum_{i=1}^n \frac{1}{x_i(0)}}$	$u_i = x_i^2 \sum_{j \in N_i} (x_i - x_j)$
P 次平均 (mean of order p)	$\sqrt[p]{\sum_{i=1}^n \frac{1}{n} x_i(0)^p}$	$u_i = -\frac{1}{p x_i^{p-1}} \sum_{j \in N_i} (x_i - x_j)$

90 Fujita Laboratory

SICEセミナー「モデル予測制御理論と協調制御の展開」

*** アウトライン**

第 I 部

- ✓ 非線形モデル予測制御
 - ・ 非線形モデル予測制御の安定性
 - ・ 非線形モデル予測制御の応用例

第 II 部

- ✚ 協調制御
 - ・ 合意 (consensus) 問題
 - ・ 群れ (Flocking)

Tokyo Institute of Technology
Fuji Laboratory 91

*** 群れ (Flocking)**

Flock : 羊・ヤキ・鳥などの群れ, 群れをなすこと

➡ Flocking : 全てのエージェントが同じ速度で動くこと

<http://www.yasukawa.com/diving/kohama5/photo.html>

Tokyo Institute of Technology
Fuji Laboratory 92

*** 群れ (Flocking)**

エージェント (制御対象)

$$\begin{cases} \dot{x} = v \cos \theta & \dot{x} : x \text{ 軸方向の速度} & \omega : \text{角速度} \\ \dot{y} = v \sin \theta & \dot{y} : y \text{ 軸方向の速度} & v : \text{速さ} \\ \dot{\theta} = \omega & & \theta : \text{進行方向} \end{cases}$$

例 2輪車両 d : 車輪の半径

$$\begin{cases} \dot{x} = v \cos \theta & \omega_R : \text{右の車輪の回転速度} \\ \dot{y} = v \sin \theta & \omega_L : \text{左の車輪の回転速度} \\ \dot{\theta} = \omega & r : \text{車輪間の距離} \end{cases}$$

$$v = \frac{r\omega_R + r\omega_L}{2} \quad \omega = \frac{r\omega_R - r\omega_L}{2d}$$

Tokyo Institute of Technology
Fuji Laboratory 93

*** 群れ (Flocking)**

複数 (n 台) のエージェントがいる状況

$$\begin{cases} \dot{x}_1 = v_1 \cos \theta_1 \\ \dot{y}_1 = v_1 \sin \theta_1 \\ \dot{\theta}_1 = \omega_1 \end{cases} \cdots \begin{cases} \dot{x}_n = v_n \cos \theta_n \\ \dot{y}_n = v_n \sin \theta_n \\ \dot{\theta}_n = \omega_n \end{cases} \Rightarrow \begin{cases} \dot{x}_i = v_i \cos \theta_i \\ \dot{y}_i = v_i \sin \theta_i \\ \dot{\theta}_i = \omega_i \end{cases} \quad (i=1, \dots, n)$$

例 車の縦列走行

PATH (California Partners for Advanced Transit and Highways)
<http://www.path.berkeley.edu/PATH/Publications/Videos/>

エージェント間の結合: グラフを用いる

仮定 グラフは無向, 固定, 連結

Tokyo Institute of Technology
Fuji Laboratory 94

*** 群れ (Flocking)**

制御目的

群れ (Flocking) を達成すること

➡ 全てのエージェントの速度が等しくなること

$$\begin{aligned} \dot{x}_1 = \dot{x}_2 = \dots = \dot{x}_n & \Rightarrow \dot{x}_i = \dot{x}_j \\ \dot{y}_1 = \dot{y}_2 = \dots = \dot{y}_n & \Rightarrow \dot{y}_i = \dot{y}_j \quad \forall i, j \end{aligned}$$

制御目標が周りのエージェントとの兼ね合い: 協調制御

例 車の縦列走行

仮定

速さは全てのエージェントに対して一定と仮定
全てのエージェントの速度が等しくなること

全てのエージェントの向きが等しくなること

$$\theta_i = \theta_j \quad \forall i, j \quad \text{姿勢整列 (Alignment)}$$

Tokyo Institute of Technology
Fuji Laboratory 95

*** 群れ (Flocking)**

制御入力

$$\omega_i = - \sum_{j \in N_i} \sin(\theta_i - \theta_j) \quad \text{分散制御}$$

自分以外のエージェントの状態を使用
近傍のエージェントとの差

$$\omega = -B \sin(B^T \theta) \quad B : \text{接続行列}$$

例

$$\begin{aligned} \omega_1 &= - \sum_{j \in N_1} \sin(\theta_1 - \theta_j) = - \sin(\theta_1 - \theta_2) \quad N_1 = \{2\} \\ \omega_2 &= - \sum_{j \in N_2} \sin(\theta_2 - \theta_j) = - \sin(\theta_2 - \theta_1) - \sin(\theta_2 - \theta_3) - \sin(\theta_2 - \theta_4) \quad N_2 = \{1, 3, 4\} \\ \omega_3 &= - \sum_{j \in N_3} \sin(\theta_3 - \theta_j) = - \sin(\theta_3 - \theta_2) - \sin(\theta_3 - \theta_4) \quad N_3 = \{2, 4\} \\ \omega_4 &= - \sum_{j \in N_4} \sin(\theta_4 - \theta_j) = - \sin(\theta_4 - \theta_2) - \sin(\theta_4 - \theta_3) \quad N_4 = \{2, 3\} \end{aligned}$$

Tokyo Institute of Technology
Fuji Laboratory 96

SICEセミナー「モデル予測制御理論と協調制御の展開」

おわりに

Tokyo Institute of Technology

協調・分散制御が目指すところ


ダイナミクスの複雑度 ↑

一つのエージェント (一つのシステム)

- ハイブリッドシステム
- 不確かなシステム
- 非線形システム

など

複雑ネットワーク上のシステム
協調・分散制御
(例 ロボットセンサネットワーク)



● 被覆(Coverage)

● 合意(Consensus)

● 群れ(Flocking) など

マルチエージェントシステム

相互結合の複雑度 →


97

Tokyo Institute of Technology Fujita Laboratories

* これからの研究の発展


Tokyo Institute of Technology

協調・分散制御へ




アポロ11号

(LQG 制御)

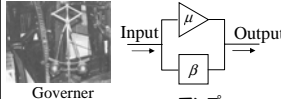


現代制御



Alice


モデル予測制御



Governor

Input → μ → Output

アンブ



F-8 Air Craft

古典制御

Robust Control (ロバスト制御)

98

Tokyo Institute of Technology Fujita Laboratories