

# SICEセミナー「モデル予測制御理論の新展開」

SICE セミナー  
「実践的な制御系設計  
-ポストロバスト制御理論の新展開と応用最前線-」  
  
「モデル予測制御理論の新展開」

  
2005年11月11日  
電気通信大学スカイオフィス  
  
東京工業大学 藤田政之

## 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**

↓

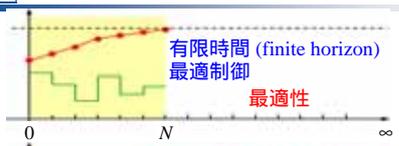
有限時間区間 (finite horizon) の最適制御を、時刻が進むにつれて評価時間区間を先へずらして進めていく

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

Fig: 化学プラント  
大嶋, 関, "モデル予測制御-V" [9]

## 最適制御とモデル予測制御

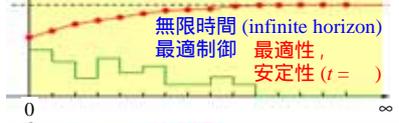
有限時間 (finite horizon) 最適制御



最適性

Fig: アポロ15号 "NASA"

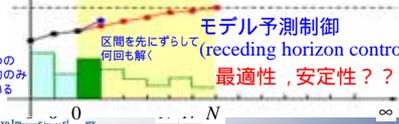
無限時間 (infinite horizon) 最適制御



最適性, 安定性 (t = infinity)

Fig: F-8 "[2]"

モデル予測制御 (receding horizon control)



最適性, 安定性??

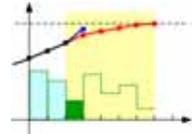
初めの入力のみ用いる

区間を先にずらして何回も解く

Fig: 車

## アウトライン

1. はじめに
2. 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
3. ハイブリッドシステムのモデル予測制御
4. おわりに



## 非線形システムと入力と状態に関する制約

非線形システム (制御対象)

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

$$y(k) = h(x(k)) \quad (2)$$

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 2.1: ロボットマニピュレータ

入力と状態に関する制約

入力  $u(k) \in U$  (3)

状態  $x(k) \in X$  (4)

例 (ロボットマニピュレータ)

入力に関する制約 U : 入力電圧 -10V ~ 10V

状態に関する制約 X : 関節角度 -90° ~ 90°

## ステージコストと終端コスト

コスト (評価関数)

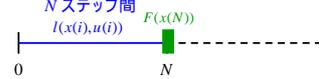
$$V_N(x, u) = \sum_{i=0}^{N-1} l(x(i), u(i)) + F(x(N)) \quad (5)$$

$l(x(i), u(i))$  : 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 (ロボットマニピュレータ)

ステージコスト  $l(x(i), u(i)) = x(i)^T Q x(i) + u(i)^T R u(i)$

終端コスト  $F(x(N)) = x^T(N) P_f x(N)$

$Q \geq 0, R > 0$

### 終端制約

終端制約 (terminal constraint)

終端の第  $N$  ステップにおける状態の制約

$$x(N) \in X_f \subset X \quad (6)$$

$X_f$ : 終端制約集合 (terminal constraint set)

例 (ロボットマニピュレータ)

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

終端制約  $X_f$ : 関節角度  $-5^\circ \sim 5^\circ$

### 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)

$$P_N(x) : \min_u \{V_N(x, u) | u \in U_N(x)\} \quad (7)$$

$U_N(x)$ : 入力と状態に関する制約 (3), (4) および終端制約 (6) を  $N$  ステップ区間すべてで満たす入力集合

最適入力  $x: i=0$  における状態  $x^o(0; x)$

$$u^{\text{optimal}}(x) = \{u^o(0; x), u^o(1; x), \dots, u^o(N-1; x)\} \quad (8)$$

最適な状態軌道 (trajectory)

$$x^o(x) = \{x^o(0; x), x^o(1; x), \dots, x^o(N-1; x), x^o(N; x)\} \quad (9)$$

$$= f(x^o(N-1; x), u(N-1; x))$$

最適なコスト (value function)

$$V_N^o(x) = V_N(x, u^o(x)) \quad (10)$$

### モデル予測制御とは

モデル予測制御では

最適入力系列 (8) の最初の操作量  $u^o(0; x)$  のみを制御対象に適用

制御則  $\kappa_N(x) := u^o(0; x) \quad (11)$

最適制御問題 (7) における  $N$  ステップの区間を一つ先を進めながら、制御則 (11) の手続きを繰り返し解く。

↓

receding horizon (moving horizon)

モデル予測制御

$u = \kappa_N(x) \quad (12)$  という入力を繰り返しかわる。

### 非線形モデル予測制御の安定性の準備

閉ループ系の安定性

非線形システム

モデル予測制御

Fig 2.1: ロボットマニピュレータ

状態が  $x$  のときに入力  $u$  を加えた次のステップの状態

$$x^+ = f(x, u) \quad (13)$$

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

$$\Delta\phi(x, u) := \phi(x^+) - \phi(x) \quad (14)$$

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

$$u = \kappa_f^*(x) \quad (15)$$

### 非線形モデル予測制御の安定条件

非線形モデル予測制御の安定条件

A1:  $X_f \subset X, X_f$  は閉集合,  $0 \in X_f$

A2:  $\kappa_f(x) \in U, \forall x \in X_f$

A3:  $f(x, \kappa_f(x)) \in X_f, \forall x \in X_f$

A4:  $[\Delta F + l](x, \kappa_f(x)) \leq 0, \forall x \in X_f$

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 (3),(4) および終端制約 (8) が満たされている

↓

可能解の存在性 (feasibility)

↓

最大出力許容集合 (maximal output admissible set)

### 非線形モデル予測制御の安定条件の略証

非線形モデル予測制御の安定条件の略証

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  $P_N(x^+)$  を考える。

制約条件付き最適制御問題に対する入力

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

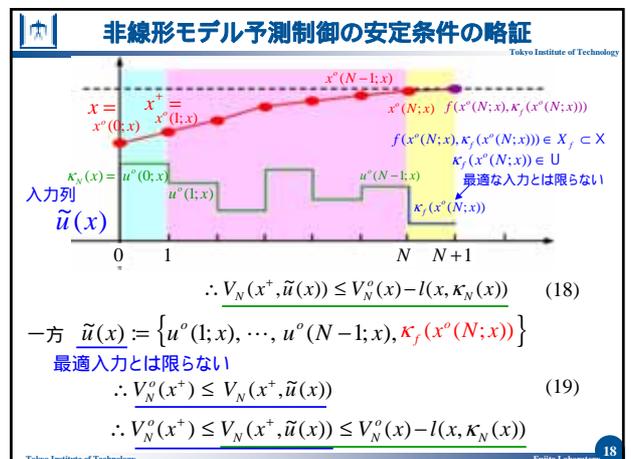
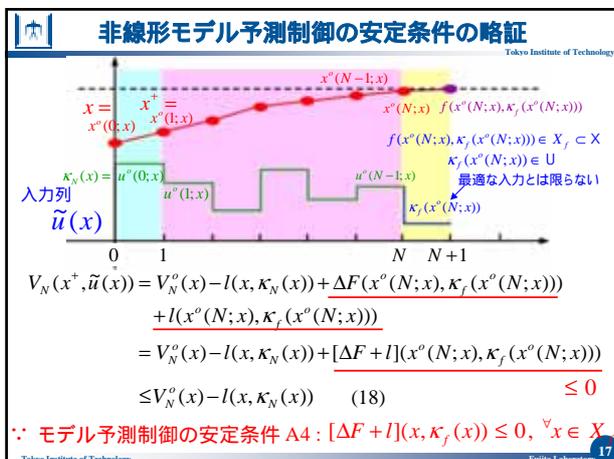
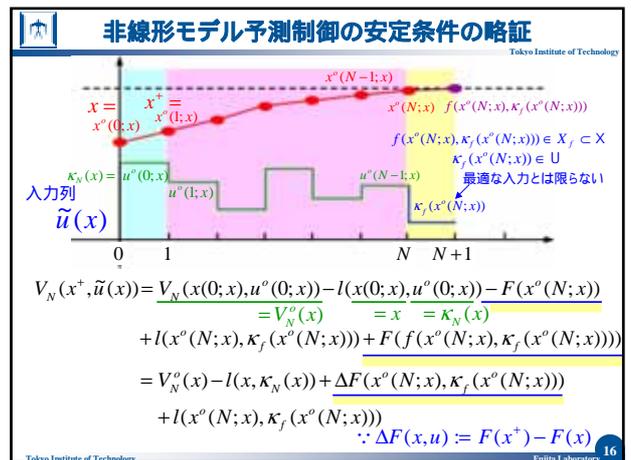
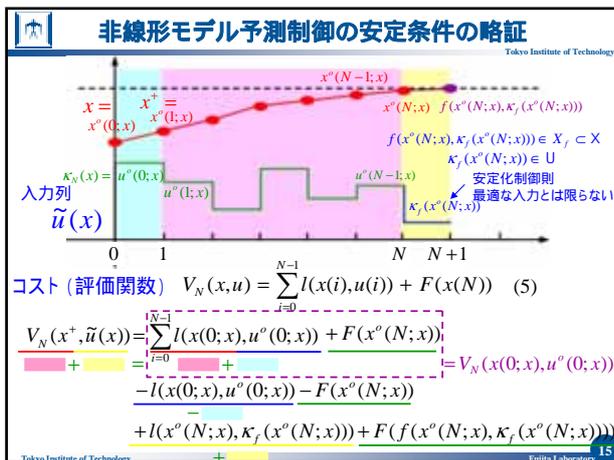
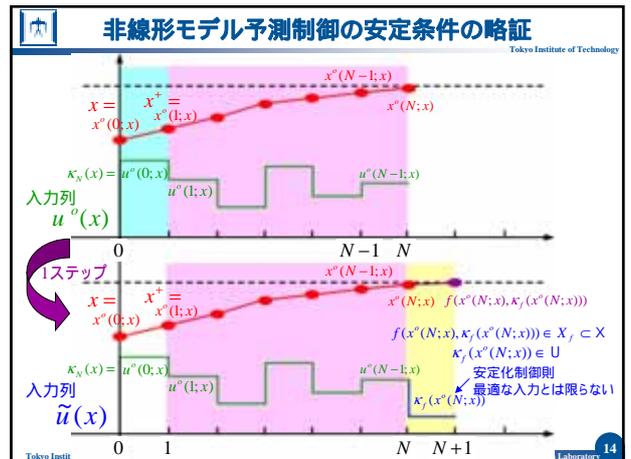
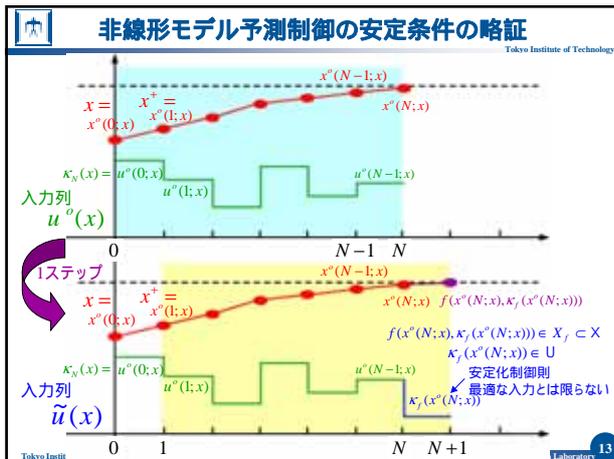
$\kappa_f(x^o(N; x)) \in U, f(x^o(N; x), \kappa_f(x^o(N; x))) \in X_f \subset X$

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

$\kappa_f(x^o(N; x)) \in U$

安定化制御則

最適入力とは限らない



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

### 非線形モデル予測制御の安定条件の略証

$x = x^*(0;x)$   
 $x^*(1;x)$   
 $x^*(N;x)$   
 $x^*(N-1;x)$   
 $f(x^*(N;x), \kappa_f(x^*(N;x)))$   
 $f(x^*(N-1;x), \kappa_f(x^*(N-1;x))) \in X_f \subset X$   
 $\kappa_f(x^*(N;x)) \in U$   
 最適な入力とは限らない  
 $\kappa_f(x^*(N;x))$   
 $u^*(0;x)$   
 $u^*(1;x)$   
 $u^*(N-1;x)$   
 $\kappa_f(x^*(N;x))$   
 $\tilde{u}(x)$   
 $0$   $1$   $N$   $N+1$

$\therefore V_N^o(x^+) \leq V_N^o(x) - l(x, \kappa_N(x))$   
 $\Delta V_N^o(x, \kappa_N(x)) \leq -l(x, \kappa_N(x))$  (21)  $\therefore \Delta V_N^o(x, u) := V_N^o(x^+) - V_N^o(x)$   
 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 → 安定性

### 終端コストの意味づけ

例: 線形の場合  
 システム  $x(k+1) = Ax(k) + Bu(k)$   
 $y(k) = Cx(k)$  どのように決めるのか?

評価関数  

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

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

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

\*実際にはリアプノフ方程式を直接計算する

### 終端コストの意味づけ

評価関数  

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

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

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

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**

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.  
 終端コストに control Lyapunov function を用いる

### 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.

### 非線形モデル予測制御のロバスト安定条件

時間領域において外乱が加わる不確かなシステム  
 $x^+ = f(x, u, w)$  (22)  
 $z = h(x)$  (23)

$w(k)$ : 不確かさなどに起因する外乱  
 $w(k) \in W(x(k), u(k))$  (24)

非線形モデル予測制御のロバスト安定条件  
 A1:  $X_f \subset X, X_f$  は閉集合,  $0 \in X_f$   
 A2:  $\kappa_f(x) \in U, \forall x \in X_f$   
 A3:  $f(x, \kappa_f(x), w) \in X_f, \forall x \in X_f, \forall w \in W(x, \kappa_f(x))$   
 A4:  $[\Delta F + l](x, \kappa_f(x), w) \leq 0, \forall x \in X_f, \forall w \in W(x, \kappa_f(x))$

終端コスト  $F(x)$  が (局所的な) ロバスト制御リアプノフ関数 (robust control Lyapunov function) となる  

$$[\Delta V_N^o + l](x, \kappa_N(x), w) \leq 0$$

# SICEセミナー「モデル予測制御理論の新展開」

## 非線形モデル予測制御のロバスト性

制御則の構成法

- ・ 開ループ min-max 型
- ・ 閉ループ min-max 型 (フィードバック min-max 型)
- ・  $H_\infty$  モデル予測制御

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

微分ゲーム理論 (game theory)

最適制御

- ・ 最適レギュレータが有していた好ましい安定余裕 (inherent robustness)

↓

モデル予測制御

- ・ 逆最適性の観点からの研究 (inverse optimality)

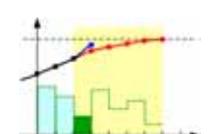
持続的な外乱が付加される場合

- ・ 入力状態間安定性 (input-to-state stability, ISS)

Tokyo Institute of Technology Fujita Laboratory 25

## アウトライン

1. はじめに
2. 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
3. ハイブリッドシステムのモデル予測制御
4. おわりに



Tokyo Institute of Technology Fujita Laboratory 26

## 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

- ・ Caltech ducted fan (California Institute of Technology)
- ・ 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" 制御対象への適応へ!!



Fig 2.2 : Caltech ducted fan    Fig 2.3 : Hover Craft    Fig 2.4 : Visual Feedback System

Tokyo Institute of Technology Fujita Laboratory 27

## モデル予測制御の応用例 (Caltech ducted fan)

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

$$\begin{cases} m\ddot{x} + F_{x_s} - F_{x_b} \cos \theta - F_{z_b} \sin \theta = 0 \\ m\ddot{z} + F_{z_s} + F_{x_b} \sin \theta - F_{z_b} \cos \theta = mg_{\text{off}} \\ J\ddot{\theta} - M_a + \frac{1}{r_s} I_p \Omega \dot{x} \cos \theta - F_{z_b} l_r = 0 \end{cases}$$

状態  $x = [x, z, \theta, \dot{x}, \dot{z}, \dot{\theta}]^T$

入力  $u = [F_{x_b}, F_{z_b}]^T$

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



Fig 2.2 : 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  $I_p$

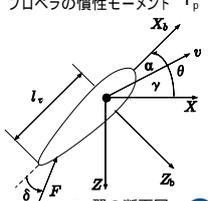


Fig 2.5 : 翼の断面図

Tokyo Institute of Technology Fujita Laboratory 28

## モデル予測制御の応用例 (Caltech ducted fan)

入力と状態に関する制約

入力に関する制約  $U$ : 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  $X$ : 高さ  $Z$  0 [m] ~ 2.5 [m]

コスト (評価関数)

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

Fig 2.6 : 中央スタンド

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

: ステージコスト (stage cost)

$$V(x(T)) = \frac{1}{2} x_{\text{err}}^T(T) P x_{\text{err}}(T)$$

: 終端コスト (terminal cost)

$x_{\text{err}} \equiv x - x_{\text{eq}} \quad u_{\text{err}} \equiv u - u_{\text{eq}} \quad$  : 偏差     $Q > 0, R > 0$

$x_{\text{eq}} = [x_{\text{cmd}}, z_{\text{cmd}}, \theta_{\text{cmd}}, 0, 0, 0]^T \quad u_{\text{eq}} = [mg_{\text{off}}, 0]^T$  : 平衡状態

M. B. Milan, 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.

Tokyo Institute of Technology Fujita Laboratory 29

## モデル予測制御の応用例 (Caltech ducted fan)



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

Tokyo Institute of Technology Fujita Laboratory 30

# SICEセミナー「モデル予測制御理論の新展開」

## 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 2.3 : Hover Craft

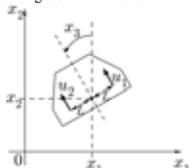


Fig 2.7 : Hover Craft モデル

T. Ohtsuka 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/CGA02ohp.html>

31

## 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  $X$  : 移動できる範囲  $-x_{\max} \sim x_{\max}$

**コスト (評価関数)**

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

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

計算アルゴリズム  
C/GMRES

32

## モデル予測制御の応用例 (Hover craft)

Tokyo Institute of Technology



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

33

## 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**

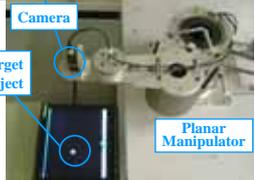
$$\begin{cases} \dot{\xi} = -M^{-1}(q)C(q, \dot{q})\xi + M^{-1}(q)u \\ \dot{f} = -\frac{s\lambda}{z_{wo}} R_{wc}^T J_b \xi - \left( \frac{s\lambda}{z_{wo}} R_{wc}^T J_b \alpha J_b^T R_{wc} + R_{wc}^T \dot{R}_{wc} \right) f \end{cases}$$

Fig 2.4 : Visual Feedback System

**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)



34

## モデル予測制御の応用例 (Visual Feedback Control)

Tokyo Institute of Technology



35

## 制御システムの階層構造

Tokyo Institute of Technology

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

制約(拘束), 故障, 不確かさ, 問題点

設定値, 操作量

モデル予測制御 (最適化)

フィードバック (ロボスト制御)

制御手法

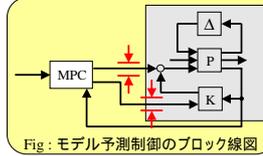


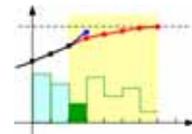
Fig : モデル予測制御のブロック線図

36

# SICEセミナー「モデル予測制御理論の新展開」

## アウトライン

- はじめに
- 非線形モデル予測制御
- ハイブリッドシステムのモデル予測制御**
- おわりに



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**

Tokyo Institute of Technology Fujita Laboratory 37

## ハイブリッドシステムとは

人間が介在して、人間の勘、度胸(決断)、経験に左右されるシステム

川口忠雄, 制御理論適用の実際, システムと制御チュートリアル講座制御工学へのガイド・ライン, 日本自動制御協会編, 1985.

貝塚洋, 原川哲美, 山岸健太郎, 鉄鋼プロセスにおけるホロン型ハイブリッドシステム, 計測と制御, vol. 26, no. 1, 1987.

...

新しいパラダイムの構築が必要

**ハイブリッドシステム**  
+  
**モデル予測制御**



Tokyo Institute of Technology Fujita Laboratory

## ハイブリッドシステム

トランスミッション

離散的な命令 (R, N, 1, 2, 3, 4, 5) + 連続的なダイナミクス (速度, トルク)



Fig.3.1 トランスミッション

プラント

バルブ, スイッチ + 連続的なダイナミクス (開閉, ON OFF)



Fig.3.2 プラント

論理を組み込んだシステム

**ハイブリッドシステム**

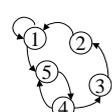
離散事象システムと連続時間ダイナミカルシステムが混在し相互影響しているシステム

Tokyo Institute of Technology Fujita Laboratory 39

## ハイブリッドシステムとは

計算機科学 (論理オートマトン) + 制御理論 (ダイナミカルシステム)

If A Then B



**ハイブリッドシステム**

$$S = (X, U, \varphi)$$

$$X = \{1, 2, 3, 4, 5\}$$

$$U = \{A, B, C\}$$

$$\varphi: X \times U \rightarrow X$$

システム:  $u(t) \rightarrow \text{system} \rightarrow y(t)$

$$\begin{cases} \frac{dx(t)}{dt} = f(x(t), u(t)) \\ y(t) = g(x(t), u(t)) \end{cases}$$

$x \in R^n, u \in R^m, y \in R^p$

Tokyo Institute of Technology Fujita Laboratory 40

## ハイブリッドシステムのクラス

ハイブリッドシステムの表現

- ハイブリッドオートマトン
- ハイブリッドベトリネット
- Mixed Logical Dynamical(MLD)システム
- Piecewise Affine(PWA)システム**
- Linear Complementarity(LC)システム
- Extended Linear Complementarity(ELC)システム
- Max-Min-Puls-Scaling(MMPS)システム

**等価性**

Tokyo Institute of Technology Fujita Laboratory 41

## 例題: Piecewise Affine System

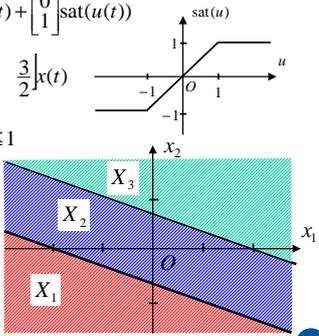
入力飽和を持つシステム

$$x(t+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \text{sat}(u(t))$$

状態フィードバック

$$u(t) = -\begin{bmatrix} 1 & 3 \\ 2 & 2 \end{bmatrix} x(t)$$

飽和

$$\text{sat}(u) = \begin{cases} 1 & u \geq 1 \\ u & -1 \leq u \leq 1 \\ -1 & u \leq -1 \end{cases}$$


$X_1: x(t+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$X_2: x(t+1) = \begin{bmatrix} 1 & 2 \\ -1/2 & -1/2 \end{bmatrix} x(t)$

$X_3: x(t+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ -1 \end{bmatrix}$

Tokyo Institute of Technology Fujita Laboratory 42

# SICEセミナー「モデル予測制御理論の新展開」

## Piecewise Affine System

(PWA) Piecewise Affine System

$$\begin{aligned} x_{k+1} &= A_i x_k + B_i u_k + f_i \\ y_k &= C_i x_k + D_i u_k + g_i \end{aligned} \quad \text{if } \begin{bmatrix} x_k \\ u_k \end{bmatrix} \in P_i$$

- 非線形要素の存在による動特性の変化

Sontag, T-AC 1981  
 Johansson and Rantzer, T-AC 1998  
 Rantzer and Johansson, T-AC 2000  
 Johansson, Springer-Verlag, 2003

## 命題計算(propositional calculus)

命題論理(propositional logic)  
 Tyler and Morari, Automatica 1999

if **A** then **B** 定性的な表現

[事実] 命題論理  $[f(x) \leq 0] \leftrightarrow [\delta = 1]$   $\delta$ : 0-1変数

$$[f(x) \leq 0] \leftrightarrow [\delta = 1] \text{ が真} \Leftrightarrow \begin{cases} f(x) \leq M(1-\delta) \\ f(x) \geq \varepsilon + (m-\varepsilon)\delta \end{cases}$$

$M = \max_{x \in Z} f(x), m = \min_{x \in Z} f(x)$   
 $\varepsilon$ : 小さい正の数

命題論理  $\Leftrightarrow$  混合整数線形不等式

## Mixed Logical Dynamical System

(MLD)

$$\begin{aligned} x(t+1) &= Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t) \\ y(t) &= Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t) \\ E_2 \delta(t) + E_3 z(t) &\leq E_4 u(t) + E_5 x(t) + E_5 \end{aligned}$$

- 0-1変数:  $\delta \in \{0,1\}^r$
- 補助変数:  $z \in R^c$
- 混合整数不等式

Bemporad and Morari, Automatica 1999

## ハイブリッドシステム表現の等価性

Heemels, Schutter, and Bemporad, Automatica 2001

- Mixed Logical Dynamical(MLD)システム
- Piecewise Affine(PWA)システム
- Linear Complementarity(LC)システム
- Extended Linear Complementarity(ELC)システム
- Max-Min-Puls-Scaling(MMPS)システム

## ハイブリッドシステム表現の変換

ハイブリッドシステム  
線形システム + 論理

モデリング  
論理表現

HYSDEL

hys2pwa

MLD システム

PWA システム

付録参照

## 応用例

ETH(Zurich)のグループ  
 車のtraction force(静摩擦力)のコントロール

traction forceを高くすると  
 加速力がUP  
 滑りにくなる

拘束  
 タイヤが滑らない  
 安定性を保つ

Adaptive Cruise Control  
 前方を走る車を考慮

拘束  
 衝突回避

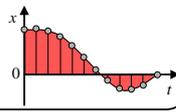
**マルチパラメトリック計画法を用いた最適制御**

状態方程式  $x(t+1) = Ax(t) + Bu(t)$   
 $y(t) = Cx(t) + Du(t)$

最適制御問題

$$J = \sum_{k=0}^{N-1} (x_{t+k}^T Q x_{t+k} + u_{t+k}^T R u_{t+k}) + x_{t+N}^T P x_{t+N}$$

線形システムに対し、評価関数  $J$  を最小にする制御問題



最適レギュレータ(LQR)  $P(t)$ : リカッチ方程式の解  
 $u(t) = Kx(t)$   $K = -(R + B^T P(t+1)B)^{-1} B^T P(t+1)A$

**拘束を有する線形離散時間システムの最適制御**

状態方程式  $x(t+1) = Ax(t) + Bu(t)$   
 $y(t) = Cx(t) + Du(t)$

拘束条件  $x_{\min} \leq x_{t+k} \leq x_{\max}$   
 $u_{\min} \leq u_{t+k} \leq u_{\max}$

最適制御問題

$$\min_{\{u_t, \dots, u_{t+N-1}\}} \sum_{k=0}^{N-1} (x_{t+k}^T Q x_{t+k} + u_{t+k}^T R u_{t+k}) + x_{t+N}^T P x_{t+N}$$

マルチパラメトリック計画問題  
 マルチパラメトリック計画問題の最適解は、区分的アファインな状態フィードバックで与えられる

状態  $x(t)$  をパラメータとする陽な形の(Explicit) 制御則  
 $u(t) = K_i x(t) + h_i \quad \text{if } x \in X_i$   
 A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos (Automatica, 2002)

**例題: マルチパラメトリック計画法を用いた最適制御**

線形システム

$$x(t+1) = \begin{bmatrix} 0.7326 & -0.0861 \\ 0.1722 & 0.999 \end{bmatrix} x(t) + \begin{bmatrix} 0.0609 \\ 0.0064 \end{bmatrix} u(t)$$

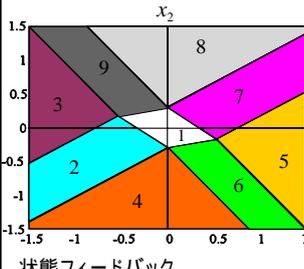
$$y(t) = [0 \quad 1.4142] x(t)$$

拘束条件:  $-2 \leq u(t) \leq 2$

最適制御問題

$$\min_{\{u_t, u_{t+1}\}} \sum_{k=0}^1 \left( x_{t+k}^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_{t+k} + 0.01 u_{t+k}^2 \right) + x_{t+2}^T \begin{bmatrix} 3.0485 & -2.5055 \\ -2.5055 & 12.9916 \end{bmatrix} x_{t+2}$$

**例題: マルチパラメトリック計画法を用いた最適制御**

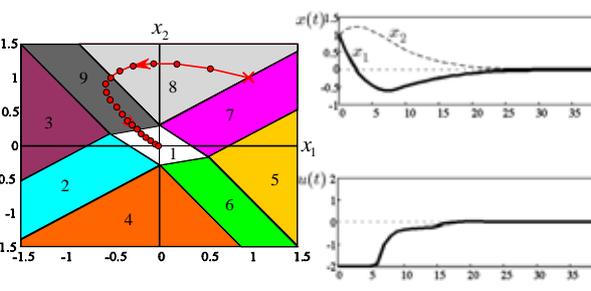


$u(t) =$

- Region 1  $[-5.9220 \quad -6.883]x$
- Region 2,3,4  $2.0000$
- Region 5,7,8  $-2.0000$
- Region 6  $[-6.5159 \quad -4.6953]x + 0.6423$
- Region 9  $[-6.5159 \quad -4.6953]x - 0.6423$

状態フィードバック  $u(t) = Kx(t)$   
 区分的アファインな状態フィードバック  $u(t) = K_i x(t) + h_i \quad \text{if } x \in X_i$   
 拘束条件:  $-2 \leq u(t) \leq 2$

**例題: システムの応答 (初期値  $x_0 = [1 \ 1]^T$ )**



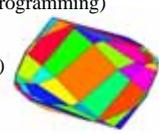
**Multi-Parametric Toolbox**

Multi-Parametric Toolbox (MPT) ver.2.04  
 スイス連邦工科大 IfA  
 URL <http://control.ee.ethz.ch/~mpt/>

数理計画問題

- 線形計画問題 (Linear Programming)
- 2次計画問題 (Quadratic Programming)
- 混合整数計画問題 (Mixed Integer Programming)
- マルチパラメトリック計画問題 (Multi-Parametric Programming)

オフライン計算で最適解が得られる



付録参照

### ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology Fujita Laboratory

**MLDシステムのモデル予測制御**  
Dynamicalシステム+命題論理

**PWAシステムのモデル予測制御**  
区分的アファインなコントローラ  
+  
区分的アファインなシステム

等価

$$X_1: x(t+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$X_2: x(t+1) = \begin{bmatrix} -1/2 & -2 \\ -1/2 & -2 \end{bmatrix} x(t)$$

$$X_3: x(t+1) = \begin{bmatrix} 1 & 2 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u(t)$$

Tokyo Institute of Technology Fujita Laboratory 55

### ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology Fujita Laboratory

**最適制御問題**

$$\min_{\{u_t, \dots, u_{t+N-1}\}} J(u_0^{N-1}, x(t)) := \sum_{k=0}^{N-1} \{ \|Q_1(u_k - u_e)\|_\infty + \|Q_2(\delta_{t+k} - \delta_e)\|_\infty + \|Q_3(z_{t+k} - z_e)\|_\infty + \|Q_4(x_{t+k} - x_e)\|_\infty + \|Q_5(y_{t+k} - y_e)\|_\infty \}$$

subject to

$$\begin{cases} x_{t+k+l} = Ax_{t+k} + B_1 u_{t+k} + B_2 \delta_{t+k} + B_3 z_{t+k} & \text{MLDシステム} \\ y_{t+k} = Cx_{t+k} + D_1 u_{t+k} + D_2 \delta_{t+k} + D_3 z_{t+k} & \text{(ハイブリッドシステム)} \\ E_2 \delta_{t+k} + E_3 z_{t+k} \leq E_1 u_{t+k} + E_4 x_{t+k} + E_5 & \text{制約条件} \\ x_{\min} \leq x_{t+k} \leq x_{\max} \\ u_{\min} \leq u_{t+k} \leq u_{\max} \\ x_{t+N} = x_e & \text{終端制約条件} \end{cases}$$

→ **混合整数計画(Mixed Integer Programming)**  
最適解  $\{u_t, \dots, u_{t+N-1}\}$  の最初の要素を入力として適用する

Tokyo Institute of Technology Fujita Laboratory 56

### ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology Fujita Laboratory

**最適制御問題**

$$\min_{\{u_t, \dots, u_{N-1}\}} \sum_{k=0}^{N-1} (u_k^T R u_k + x_k^T Q x_k) + x_N^T Q_f x_N^T$$

subject to

$$\begin{cases} x(k+1) = Ax(k) + B\mu(k) + f_i & \text{PWAシステム (ハイブリッドシステム)} \\ y(k) = C_i x(k) + D_i \mu(k) + g_i \\ x_k \in X \\ u_k \in U & \text{制約条件} \\ x(N) = X_{\text{set}} & \text{終端集合} \end{cases}$$

→ **マルチパラメトリック計画問題**

**モデル予測制御則**

$$u(t) = [0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0] f(x(t)), \quad x \in X_i$$

$$f(x(t)) = F_i x(t) + G_i \text{ if } x \in X_i$$

Tokyo Institute of Technology Fujita Laboratory 57

### モデル予測の比較

Tokyo Institute of Technology Fujita Laboratory

**MLDシステムのモデル予測制御**

→ **混合整数計画問題(Mixed Integer Programming)**

- on-line計算(繰り返し計算)
- 計算時間: 大

⇔ 等価

**PWAシステムのモデル予測制御**

→ **マルチパラメトリック線形計画問題(mp-LP)**

$$\min_p \{ f_c^T p_c + f_d^T p_d \}$$

$$G_c p_c + G_d p_d \leq S + F \xi(t)$$

- off-line計算
- 計算時間: 小

Tokyo Institute of Technology Fujita Laboratory 58

### 例題1: ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology Fujita Laboratory

**ハイブリッドシステム**

$$x(t+1) = 0.8 \begin{bmatrix} \cos \alpha(t) & -\sin \alpha(t) \\ \sin \alpha(t) & \cos \alpha(t) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$\alpha(t) = \begin{cases} \pi/3 & \text{if } [1 \ 0]x(t) \geq 0 \\ -\pi/3 & \text{if } [1 \ 0]x(t) < 0 \end{cases}$$

拘束条件:  $x(t) \in [-10, 10] \times [-10, 10]$   
 $u(t) \in [-1, 1]$

**最適制御問題**

$$\min_{\{u_t, \dots, u_{N-1}\}} \sum_{k=0}^{N-1} (u_k^T R u_k + x_k^T Q x_k) + x_N^T Q_f x_N^T$$

$R = 1$   
 $N = 5$   
 $Q = I_2$   
 $Q_f = 0$

Tokyo Institute of Technology Fujita Laboratory 59

### 例題1: ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology Fujita Laboratory

**Multi-Parametric Toolbox**

ハイブリッドシステム (線形システム + 論理) → モデリング 論理表現 → Multi-Parametric Toolbox

Multi-Parametric Toolbox → PWAシステム (最適制御 (モデル予測制御))

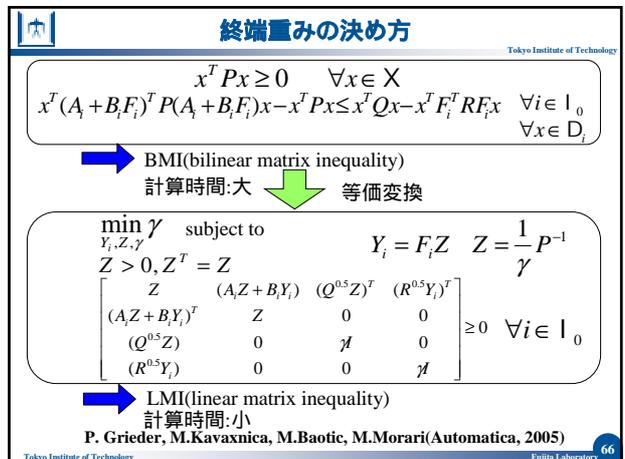
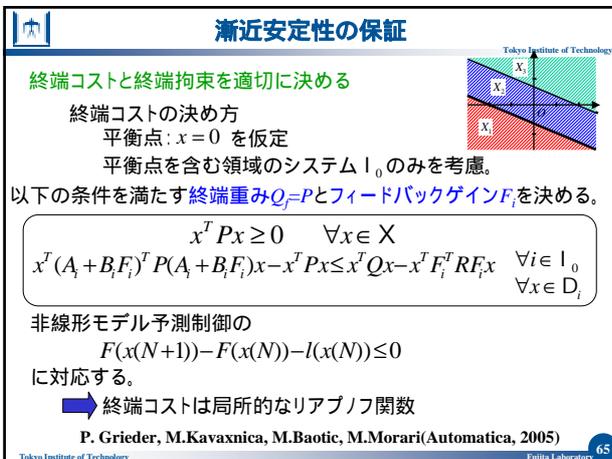
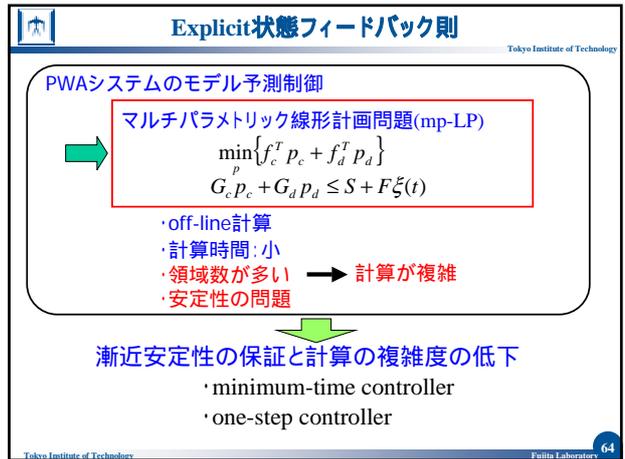
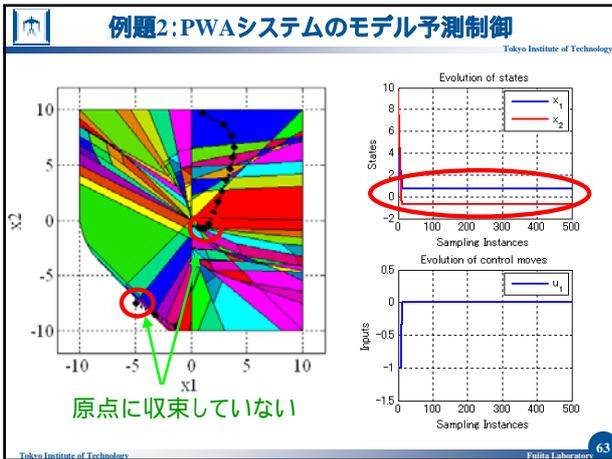
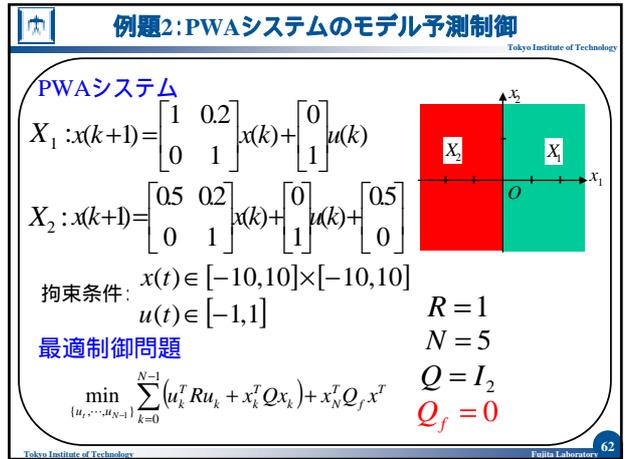
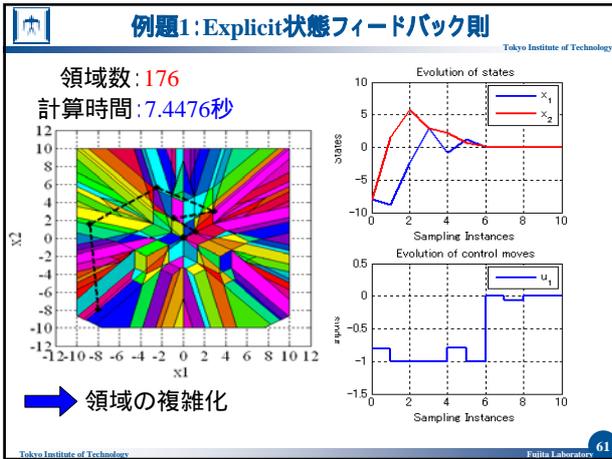
Multi-Parametric Toolbox → MLDシステム (hys2pwa)

区分的アファインな制御則  $u(t) = K_i x(t) + h_i \text{ if } x \in X_i$

- 複雑度の軽減, 近似最適解の構成
- 安定性
- 実行可能性

付録参照

Tokyo Institute of Technology Fujita Laboratory 60



### ■ 終端集合の決め方

Tokyo Institute of Technology

$$\min_{Y_i, Z, \gamma} \gamma \quad \text{subject to} \quad Y_i = F_i Z \quad Z = \frac{1}{\gamma} P^{-1}$$

$$Z > 0, Z^T = Z$$

$$\begin{bmatrix} Z & (A_i Z + B_i Y_i) & (Q^{0.5} Z)^T & (R^{0.5} Y_i)^T \\ (A_i Z + B_i Y_i)^T & Z & 0 & 0 \\ (Q^{0.5} Z) & 0 & \not\# & 0 \\ (R^{0.5} Y_i) & 0 & 0 & \not\# \end{bmatrix} \geq 0 \quad \forall i \in I_0$$

→ 終端重み  $Q_f = P$  とフィードバックゲイン  $F_i$  が決定

**終端集合**

PWAの閉ループ系  $x(k+1) = (A_i + B_i F_i)x(k)$  に対し  
 拘束を満たす不変集合(最大出力許容集合)  $O_\infty^{PWA}$   
 を終端集合とする

→ 終端重み  $Q_f$  と終端集合  $O_\infty^{PWA}$  により漸近安定を保証

P. Grieder, M.Kavavnicca, M.Baotic, M.Morari(Automatica, 2005)

Tokyo Institute of Technology Fujita Laboratory 67

### ■ 漸近安定性

Tokyo Institute of Technology

定理

PWAシステム  $x(k+1) = A_i x(k) + B_i u(k) + f_i$   
 に対する最適制御問題

$$J_N^*(x(0)) = \min_{\{u_i, \dots, u_{N-1}\}} \sum_{k=0}^{N-1} (u_k^T R u_k + x_k^T Q x_k) + x_N^T Q_f x^T$$

に関して、以下の最適化

$$\min_{Y_i, Z, \gamma} \gamma \quad \text{subject to} \quad Y_i = F_i Z \quad Z = \frac{1}{\gamma} P^{-1}$$

$$Z > 0, Z^T = Z$$

$$\begin{bmatrix} Z & (A_i Z + B_i Y_i) & (Q^{0.5} Z)^T & (R^{0.5} Y_i)^T \\ (A_i Z + B_i Y_i)^T & Z & 0 & 0 \\ (Q^{0.5} Z) & 0 & \not\# & 0 \\ (R^{0.5} Y_i) & 0 & 0 & \not\# \end{bmatrix} \geq 0 \quad \forall i \in I_0$$

により求まる  $P$  を終端重み、閉ループ系  $x(k+1) = (A_i + B_i F_i)x(k)$   
 に対する拘束を満たす不変集合  $O_\infty^{PWA}$  を終端集合とするとシステムは**漸近安定**となる。

P. Grieder, M.Kavavnicca, M.Baotic, M.Morari(Automatica, 2005)

Tokyo Institute of Technology Fujita Laboratory 68

### ■ low complexity

Tokyo Institute of Technology

領域の複雑度の低下

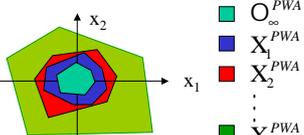
- minimum-time controller
- 問題が解ける範囲を考える
- $N$ ステップで不変集合  $O_\infty^{PWA}$  へ到達できる初期状態の集合を  $X_\infty^{PWA}$  とする。
- $X_\infty^{PWA}$ :  $O_\infty^{PWA}$  へ到達できる最大の初期状態集合
- $O_\infty^{PWA}$  から逆に  $X_\infty^{PWA}$  を Multi-Parametric 計画法で計算する

**複雑度低下の理由**

初期値が決まると  $O_\infty^{PWA}$  へ収束を保障する最少のホライズンが決まる。

→ 必要最小限のホライズンでできる。

P. Grieder, M.Kavavnicca, M.Baotic, M.Morari(Automatica, 2005)



- $O_\infty^{PWA}$
- $X_1^{PWA}$
- $X_2^{PWA}$
- ⋮
- $X_N^{PWA}$

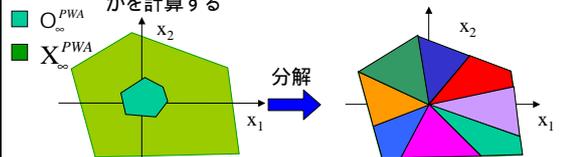
Tokyo Institute of Technology Fujita Laboratory 69

### ■ low complexity

Tokyo Institute of Technology

領域の複雑度の低下

- one-step controller
- $X_\infty^{PWA}$  を計算する
- ホライズンは1に固定 → 複雑度低下の理由
- $X_\infty^{PWA}$  をより小さな多角形に分解
- Multi-Parametric 計画法で次にどの領域に行けばよいかを計算する



P. Grieder, M.Kavavnicca, M.Baotic, M.Morari(Automatica, 2005)

Tokyo Institute of Technology Fujita Laboratory 70

### ■ 例題: ハイブリッドシステムのモデル予測制御

Tokyo Institute of Technology

ハイブリッドシステム

$$x(t+1) = 0.8 \begin{bmatrix} \cos \alpha(t) & -\sin \alpha(t) \\ \sin \alpha(t) & \cos \alpha(t) \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$\alpha(t) = \begin{cases} \pi/3 & \text{if } [1 \ 0]x(t) \geq 0 \\ -\pi/3 & \text{if } [1 \ 0]x(t) < 0 \end{cases}$$

拘束条件:  $x(t) \in [-10, 10] \times [-10, 10]$   
 $u(t) \in [-1, 1]$

最適制御問題

$$\min_{\{u_i, \dots, u_{N-1}\}} \sum_{k=0}^{N-1} (u_k^T R u_k + x_k^T Q x_k) + x_N^T Q_f x^T$$

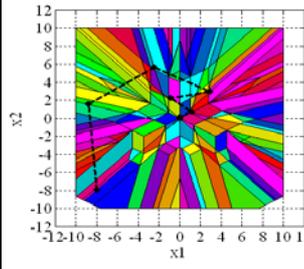
$R = 1$   
 $N = 5$   
 $Q = I_2$   
 $Q_f = 0$

Tokyo Institute of Technology Fujita Laboratory 71

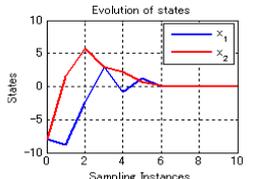
### ■ 例題: Explicit状態フィードバック則

Tokyo Institute of Technology

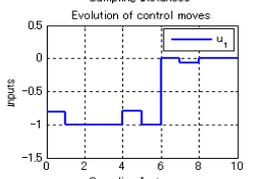
領域数: 176  
 計算時間: 7.4476秒



Evolution of states

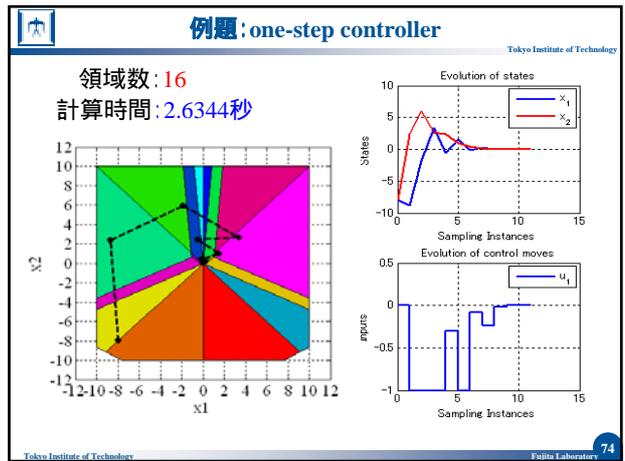
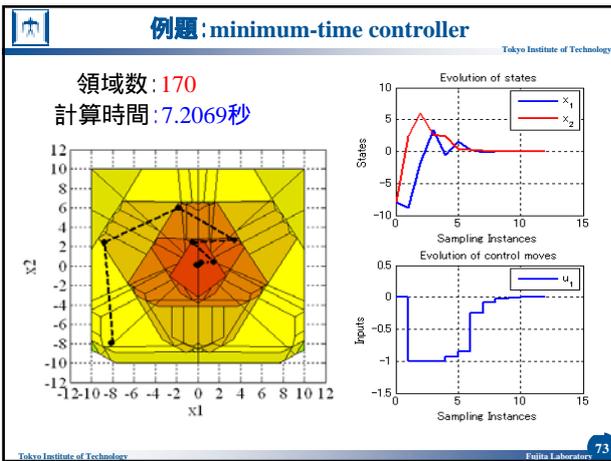


Evolution of control moves



Tokyo Institute of Technology Fujita Laboratory 72

# SICEセミナー「モデル予測制御理論の新展開」



### ハイブリッドシステムの安定性解析

基本はLyapunovの定理

問題はLyapunov関数の計算

- Common quadratic Lyapunov function
- Common sum-of-squares Lyapunov function
- Piecewise affine Lyapunov function
- Piecewise quadratic Lyapunov function

Fig.3.3 Piecewise quadratic Lyapunov function

Fig.3.4 Piecewise affine Lyapunov function

P. Biswas, P.Grieder, J.Lofberg, M.Morari(IFAC, 2005)

75

### 応用例

ETH(Zurich)のグループ  
Electronic throttle

Smart Damping Materials

76

### 分散型の制御

制御目的  
入力大きさに制約がある条件下で複数台のVehicleがFormationを形成しながら目的地にたどり着くこと

制約  
入力大きさ 障害物回避 衝突回避

問題の大規模化  
分散制御 + Networked Control の必要性

Fig.3.6 ブルーインパルス  
<http://www.jda.go.jp/jasdf/gallery/bluet2.htm>

77

### Distributed Receding Horizon Control

コントローラ

- Distributed Receding Horizon (分散型モデル予測制御)  
基本的な考えは一般のモデル予測と同じ
- (Centralized) Receding Horizon  
全てピークルの状態が計測可能  
Masterが存在
- Distributed Receding Horizon  
通信できるピークルの状態のみ獲得可能  
個々が入力を決定

Networked Control System  
分散化の問題点  
通信時間によるむだ時間の発生 → 不安定化を導く  
ノイズなどによるバケットロス

→ Distributed + Networkedの一体化が必要

78

# SICEセミナー「モデル予測制御理論の新展開」

## Distributed Receding Horizon Control

Tokyo Institute of Technology

分散制御への二つのアプローチ

連続系の非線形システムのモデル予測制御

William B. Dunbar and Richard M. Murray,  
 'Distributed Receding Horizon Control with  
 Application to Multi-Vehicle Formation Stabilization,'  
 Accepted to *Automatica*, June, 2004. Revision submitted May, 2005.

離散系のハイブリッドモデル予測制御

T. Keviczky, F. Borrelli and G. J. Balas  
 'A Study on Decentralized Receding Horizon Control for  
 Decoupled Systems'  
 Technical Report, University of Minnesota, Minneapolis. March 2004.

通信の問題に関して触れられていない  
**Networked Control System**への拡張が必要

Tokyo Institute of Technology Fujita Laboratory 79

## Distributed Receding Horizon Control

Tokyo Institute of Technology

連続系の非線形システムのモデル予測制御

個々のシステムは**連続な非線形システム**  
 フォーメーションの核が存在  
 通信相手は固定  
**衝突回避、障害物回避不可能**

最適化

$$J^*(x(t_k)) = \min_{u(t_k)} \int_{t_k}^{t_k+T} \sum_{j \in N_i} L(x_j(s), x(s), u(s)) ds + \gamma \|x(t_k+T) - x^c\|_P^2$$

Subject to

$$\begin{cases} \dot{x}_i = f(x_i(\tau), u_i(\tau)) & L(x_i, x, u): \text{評価関数} \\ \dot{x}_{-i} = f(x_{-i}(\tau), u_{-i}(\tau)) & x_i: \text{自分の状態} \quad u_i: \text{自分の入力} \\ x_i \in X_i & \text{拘束条件} \quad x_{-i}: \text{相手の状態} \quad u_{-i}: \text{相手の入力} \\ u_i \in U_i & N_i: \text{近傍} \quad P: \text{終端重み} \\ x_i(T) \in \Omega_i(\varepsilon_i) & \text{終端集合} \quad \gamma: \text{定数} \quad T: \text{ホライゾン} \end{cases}$$

William B. Dunbar and Richard M. Murray, (*Automatica*, 2005)

Tokyo Institute of Technology Fujita Laboratory 80

## Distributed Receding Horizon Control

Tokyo Institute of Technology

離散系のハイブリッドモデル予測制御

個々のシステムは**離散系の非線形システム**  
 個々のシステムが目的地へ移動  
 通信相手は流動的  
**衝突回避、障害物回避可能**

最適化

$$J_N^*(x_1^j, \bar{x}_1^j) = \min_{U_1^j} \sum_{k=0}^{N-1} l^j(x_{k+1}^j, u_{k+1}^j, x_{k+1}^j, u_{k+1}^j) + l_N^j(x_{N+1}^j, x_{N+1}^j)$$

Subject to

$$\begin{cases} x_{k+1}^j = f^j(x_k^j, u_{k+1}^j) & x_{k+1}^j \in X^j \quad u_{k+1}^j \in U^j \\ x_{k+1}^j = f^j(x_k^j, u_{k+1}^j) & x_{k+1}^j \in X^j \quad u_{k+1}^j \in U^j \quad j \in N_i \\ g^{i,j}(x_{k+1}^i, u_{k+1}^i, x_{k+1}^j, u_{k+1}^j) \leq 0 & j \in N_i \quad \text{拘束条件} \\ x_{N+1}^j \in X_f^j & x_{N+1}^j \in X_f^j \quad \text{終端集合} \\ x_{0,j} \in x_1^j & x_{0,j} \in x_1^j \quad \text{初期条件} \end{cases}$$

T. Keviczky, F. Borrelli and G. J. Balas (Technical Report, University of Minnesota, 2004)

Tokyo Institute of Technology Fujita Laboratory 81

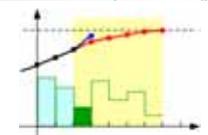
## アウトライン

Tokyo Institute of Technology

- はじめに
- 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
- ハイブリッドシステムのモデル予測制御
- おわりに

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



Tokyo Institute of Technology Fujita Laboratory 82

## 付録

Tokyo Institute of Technology

HYSTEL

Multi-Parametric Tool box

Tokyo Institute of Technology Fujita Laboratory 83

## HYSDEL

Tokyo Institute of Technology

Hybrid System Description Language (HYSDEL) ver.2.0.5

<http://control.ee.ethz.ch/~hybrid/hysdel/>

ハイブリッドシステム (Hybrid Discrete Automaton)

- 線形ダイナミクス
- オートマトン
- If-then-else 規則
- 命題論理

➡ **Mixed Logical Dynamical (MLD) システム**

$$\begin{aligned} x(t+1) &= Ax(t) + B_1u(t) + B_2\delta(t) + B_3z(t) \\ y(t) &= Cx(t) + D_1u(t) + D_2\delta(t) + D_3z(t) \\ E_2\delta(t) + E_3z(t) &\leq E_1u(t) + E_4x(t) + E_5 \end{aligned}$$

Tokyo Institute of Technology Fujita Laboratory 84

# SICEセミナー「モデル予測制御理論の新展開」

**HYSDEL** Tokyo Institute of Technology Fujita Laboratory 85

```

SYSTEM car{
INTERFACE{
STATE{ }           状態
INPUT{ }           入力
OUTPUT{ }          出力
PARAMETER{ }      パラメータ
}
IMPLEMENTATION{
CONTINUOUS{ }     ダイナミクス
OUTPUT{ }         出力
MUST{ }           拘束条件
AUX{ }            補助変数
AD{ }             AD e.g.  $x \leq 0 \Leftrightarrow \delta = 1$ 
DA{ IF THEN }     DA e.g. IF  $\delta = 1$  THEN  $x = 1000$ 
LOGIC{ }          命題論理
}
}
    
```

**HYSDEL** Tokyo Institute of Technology Fujita Laboratory 86

F.D. Torrisi and A. Bemporad,  
HYSDEL – A Tool for Generating Computational Hybrid Models for Analysis and Synthesis Problems, Technical Report: AUT 02-03, ETH, Zurich, 2002.

```

SYSTEM car{
INTERFACE{
STATE{
REAL position [-1000, 1000];
REAL speed [-50*1000/3600, 220*1000/3600];
}
INPUT{
REAL torque [-300,300];
REAL F_brake [0,9000];
REAL slope [0, 1];
BOOL gear1, gear2, gear3, gear4, gear5, gearR;
}
OUTPUT{
REAL position_y, speed_y, w_y;
}
PARAMETER{
REAL mass = 1020;
REAL Ts = 0.5;
REAL g = 9.8;
REAL beta_friction =25;
REAL Rgear1 = 3.7272;
REAL Rgear2 = 2.048;
}
}
    
```

5段ギアシステム (プログラムの一部)

位置  $p(t+1) = p(t) + T_s v(t)$

速度  $v(t+1) = f(t)$

ギアに依存した非線形関数



**HYSDEL** Tokyo Institute of Technology Fujita Laboratory 87

```

IMPLEMENTATION{
AUX{ REAL Fe1, Fe2, Fe3, Fe4, Fe5, FeR, w1, w2, w3, w4, w5,
      BOOL dPWL1, dPWL2, dPWL3, dPWL4; }
AD{
dPWL1 = wPWL1 - (w1 + w2 + w3 + w4 + w5 + wR) <= 0;
}
DA{
Fe1 = {IF gear1 THEN torque / speed_factor * Rgear1;
w1 = {IF gear1 THEN speed / speed_factor * Rgear1;
DCel = {IF dPWL1 THEN (aPWL2 - aPWL1) + (bPWL2 - bPWL1) * (w1 + w2 + w3 + w4 + wR);
CONTINUOUS{
position = position + Ts * speed;
speed = speed + Ts / mass * (Fe1 + Fe2 + Fe3 + Fe4 + Fe5 + FeR - F_brake - beta_friction * speed) - g * slope;
OUTPUT{
position_y = position;
speed_y = speed;
w_y = w1 + w2 + w3 + w4 + w5 + wR;
MUST{
-w1 <= -wemin; w1 <= wemax; -w2 <= -wemin; w2 <= wemax;
-torque - (aPWL1 + bPWL1 * (w1 + w2 + w3 + w4 + w5 + wR)) <= 0;
}
}
    
```

5段ギアシステム (プログラムの一部)

**HYSDEL** Tokyo Institute of Technology Fujita Laboratory 88

```

IMPLEMENTATION{
SYSTEM car{
INTERFACE{
STATE{
REAL position [-1000, 1000];
REAL speed [-50*1000/3600, 220*1000/3600];
}
INPUT{
REAL torque [-300,300];
REAL F_brake [0,9000];
REAL slope [0, 1];
BOOL gear1, gear2, gear3, gear4, gear5, gearR;
}
OUTPUT{
REAL position_y, speed_y, w_y;
}
PARAMETER{
REAL mass = 1020;
REAL Ts = 0.5;
REAL g = 9.8;
REAL beta_friction =25;
REAL Rgear1 = 3.7272;
REAL Rgear2 = 2.048;
}
}
}
    
```

コンパイル

Mixed Logical Dynamical (MLD) システム

**HYSDEL** Tokyo Institute of Technology Fujita Laboratory 89

Piecewise Affine (PWA) Plugin: hys2pwa 1.1.3

Mixed Logical Dynamical システム

$$x(t+1) = Ax(t) + B_1 u(t) + B_2 \delta(t) + B_3 z(t)$$

$$y(t) = Cx(t) + D_1 u(t) + D_2 \delta(t) + D_3 z(t)$$

$$E_2 \delta(t) + E_3 z(t) \leq E_1 u(t) + E_4 x(t) + E_5$$

↓ hys2pwa

<http://control.ee.ethz.ch/~hybrid/hysdel/hysdel.msql>

Piecewise Affine システム

$$x(t+1) = A_i x(t) + B_i u(t) + f_i \quad \text{if} \quad \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \in D_i, \quad i \in I$$

**Multi-Parametric Toolbox** Tokyo Institute of Technology Fujita Laboratory 90

Multi-Parametric Toolbox (MPT) ver.2.04

スイス連邦工科大 IfA  
URL <http://control.ee.ethz.ch/~mpt/>

- MATLAB
- LP solver
  - Linprog (MATLAB)
  - CDD (Free)
  - CPLEX (Commercial)
  - NAC solver (Commercial)
- SeDuMi

**有限時間最適制御問題**

線形離散時間システム

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

$$y(t) = Cx(t) + Du(t)$$

ノルム  $\|Qx\|_1 = |Qx|$   
 $\|Qx\|_2 = x^T Q x$   
 $\|Qx\|_\infty = \max|Qx|$

有限時間最適制御問題

$$J_N^*(x(0)) = \min_{u_0, \dots, u_{N-1}} \|Q_f x_N\|_l + \sum_{k=0}^{N-1} \|Ru_k\|_l + \|Qx_k\|_l \quad l \in \{1, 2, \infty\}$$

制約条件  $\begin{cases} x_k \in X, & \forall k \in \{1, \dots, N\} \\ u_k \in U, & \forall k \in \{1, \dots, N-1\} \\ x_N \in X_{set} \\ x_{k+1} = Ax_k + Bu_k, & x(0) = x_0, \quad \forall k \in \{1, \dots, N\} \end{cases}$

**有限時間最適制御問題**

有限時間最適制御問題

$$J_N^*(x(0)) = \min_{u_0, \dots, u_{N-1}} \|Q_f x_N\|_l + \sum_{k=0}^{N-1} \|Ru_k\|_l + \|Qx_k\|_l$$

制約条件  $\begin{cases} x_k \in X, & \forall k \in \{1, \dots, N\} \\ u_k \in U, & \forall k \in \{1, \dots, N-1\} \\ x_N \in X_{set} \\ x_{k+1} = Ax_k + Bu_k, & x(0) = x_0, \quad \forall k \in \{1, \dots, N\} \end{cases}$

$$J_N^*(x(0)) = x^T(0)Yx(0) + \min_{U_N} \{U_N^T H U_N + x^T(0)F U_N\}$$

制約条件  $GU_N \leq W + Ex(0)$

入力列  $U_N = [u_0^T, \dots, u_{N-1}^T]^T$

**有限時間最適制御問題**

$$J_N^*(x(0)) = x^T(0)Yx(0) + \min_{U_N} \{U_N^T H U_N + x^T(0)F U_N\}$$

制約条件  $GU_N \leq W + Ex(0)$

入力列  $U_N = [u_0^T, \dots, u_{N-1}^T]^T$

**定理 (陽な形の(Explicit)コントローラ)**

有限時間最適制御問題が可解であれば、最適入力列は

$$U_N^*(x(0)) = F_r x(0) + G_r \quad \text{if } x(0) \in P_r = \{x \in R^n \mid H_r x \leq K_r\}$$

という区分的アファインな制御則で与えられる。

A. Bemporad, M. Morari, V. Dua and E.N. Pistikopoulos (*Automatica*, 2002)  
 F. Borrelli (*Constrained Optimal Control of Linear and Hybrid Systems*, 2003)

**Example 1**

システム

$$x(t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t)$$

問題

$$\min_{u_0, \dots, u_4} \sum_{k=0}^4 \left\| \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k \right\|_2 + \|u_k\|_2$$

Subject to  $\begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x(t) \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix}$   
 $-1 \leq u(t) \leq 1$

**Example 1**

システム

$$x(t+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u(t)$$

$$\begin{bmatrix} -5 \\ -5 \end{bmatrix} \leq x(t) \leq \begin{bmatrix} 5 \\ 5 \end{bmatrix}$$

$$-1 \leq u(t) \leq 1$$

システムの記述

```
sysStruct.A=[1 1;0 1];
sysStruct.B=[0;1];
sysStruct.C=[1 0;0 1];
sysStruct.D=[0;0];
sysStruct.umin=-1;
sysStruct.umax=1;
sysStruct.ymin=[-5 -5]';
sysStruct.ymax=[5 5]';
```

構造体 sysStruct

**Example 1**

問題

$$\min_{u_0, \dots, u_4} \sum_{k=0}^4 \left\| \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k \right\|_2 + \|u_k\|_2$$

最適制御問題の記述

```
probStruct.norm=2;
probStruct.Q=eye(2);
probStruct.R=1;
probStruct.N=5;
probStruct.xObounds0=0;
probStruct.subopt_lev=0;
probStruct.P_N=0;
probStruct.Tconstraint=0;
```

構造体 probStruct

# SICEセミナー「モデル予測制御理論の新展開」

**Example 1**

システム 構造体 sysStruct  
 問題 構造体 probStruct

最適制御問題を解く

```
[ctrlStruct]=mpt_Control(sysStruct,probStruct);
```

構造体 ctrlStruct

$$U_N^*(x(0)) = F_r x(0) + G_r, \text{ if } x(0) \in P_r = \{x \in R^n \mid H_r x \leq K_r\}$$

Tokyo Institute of Technology Fujita Laboratory 97

**Example 1**

構造体 ctrlStruct

領域を描く

```
mpt_plotPartition(ctrlStruct)
```

Controller partition with 13 regions.

Tokyo Institute of Technology Fujita Laboratory 98

**Example 1**

複雑度を軽減する

```
probStruct.subopt_lev=2;
```

Controller partition with 3 regions.

Tokyo Institute of Technology Fujita Laboratory 99

**Example 1**

最適トラジェクトリ(入力)を計算する

```
[X,U,Y,D,cost,trajectory,feasible] = mpt_computeTrajectory(ctrlStruct,[8;-4],10)
```

Tokyo Institute of Technology Fujita Laboratory 100

**Example 1**

最適トラジェクトリを描く

```
[X,U,cost,trajectory]=mpt_plotTrajectory(ctrlStruct)
```

Closed-Loop Trajectory for initial state [7.9563, -4.0268]

Tokyo Institute of Technology Fujita Laboratory 101

**GUI**

GUIを使って設計をする

MATLABのcommand windowで'mpt\_studio'

Tokyo Institute of Technology Fujita Laboratory 102