

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の紹介」

第 I 部



FL06_22_1

山田 照樹



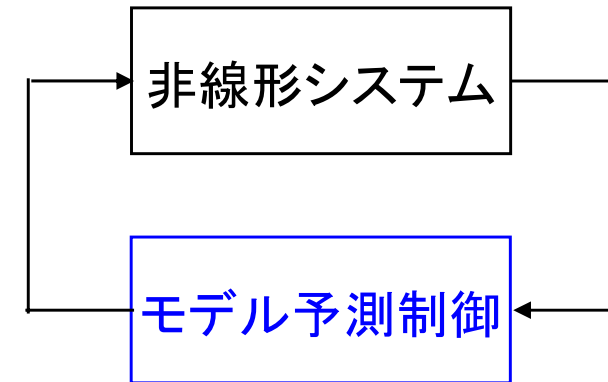
非線形モデル予測制御の安定条件

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



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および終端制約が満たされている




可能解の存在性 (feasibility)



安定条件の下での安定性の証明の流れ1


安定条件A1からA4がなりたつとき

評価関数の最適なコスト $V_N^o(x)$ がリアプノフ関数となることを示す。

- $V_N^o(x) > 0$  評価関数の置き方より明らか。
- $\Delta V_N^o(x) = \boxed{V_N^o(x^+)} - V_N^o(x) \leq 0$ を示したい。

最適入力列 $u^o(x) = \{ \underbrace{u^o(0;x)}_{\text{blue}}, \boxed{u^o(1;x), \dots, u^o(N-1;x)} \}$

$\underline{x^+} = f(x, \underline{u^o(0;x)})$  初期ステップの入力だけシステムに加える

$\boxed{V_N^o(x^+)}$ の評価が必要  x^+ を初期状態とする最適制御問題で可解な入力列 \tilde{u} を考える。

$\tilde{u}(x^+) = \{ \boxed{u^o(0;x^+), u^o(1;x^+), \dots, u^o(N-2;x^+)} \}, \boxed{} \}$

$u^o(1;x), \dots, u^o(N-1;x)$ に等しい。



\tilde{u} が可解となるためには？



安定条件の下での安定性の証明の流れ2

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

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

$= x$

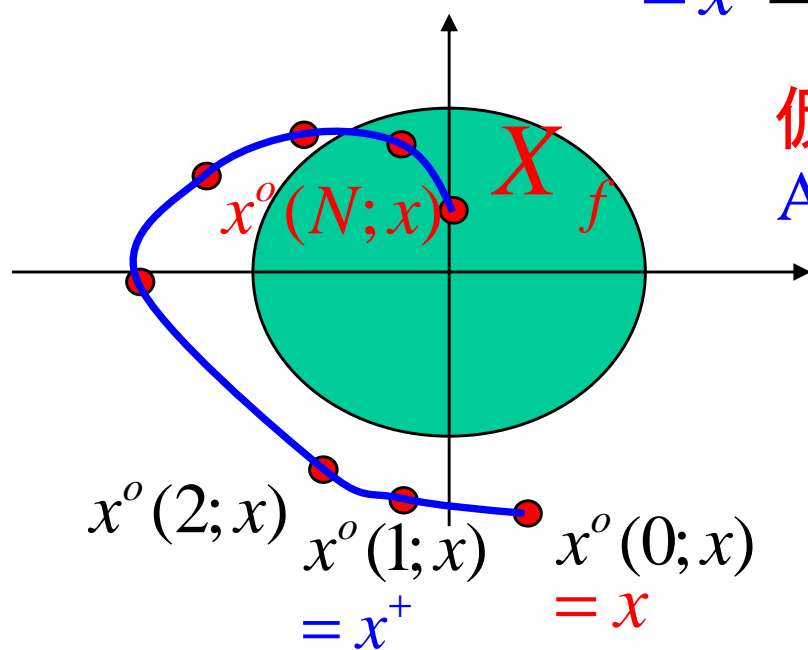
$= x^+ = f(x, u^o(0;x))$

終端制約条件を満たす.

$$x^o(N;x) \in X_f$$

仮定

A1からA4がなりたつ.



$$A2 \text{ より } \kappa_f(x^o(N;x)) \in \mathcal{U}$$

A3 より

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

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

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

は可解な最適入力列のひとつ.



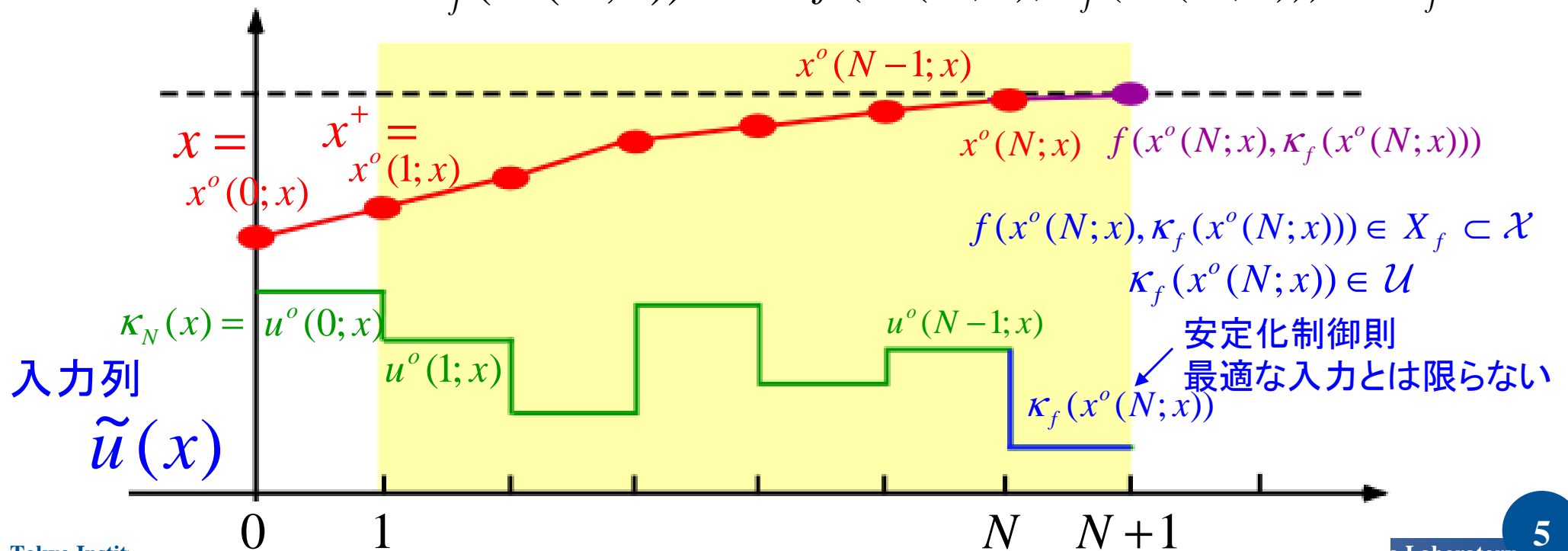
非線形モデル予測制御の安定条件の略証

最適制御問題(2)と対応する最適入力(3)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

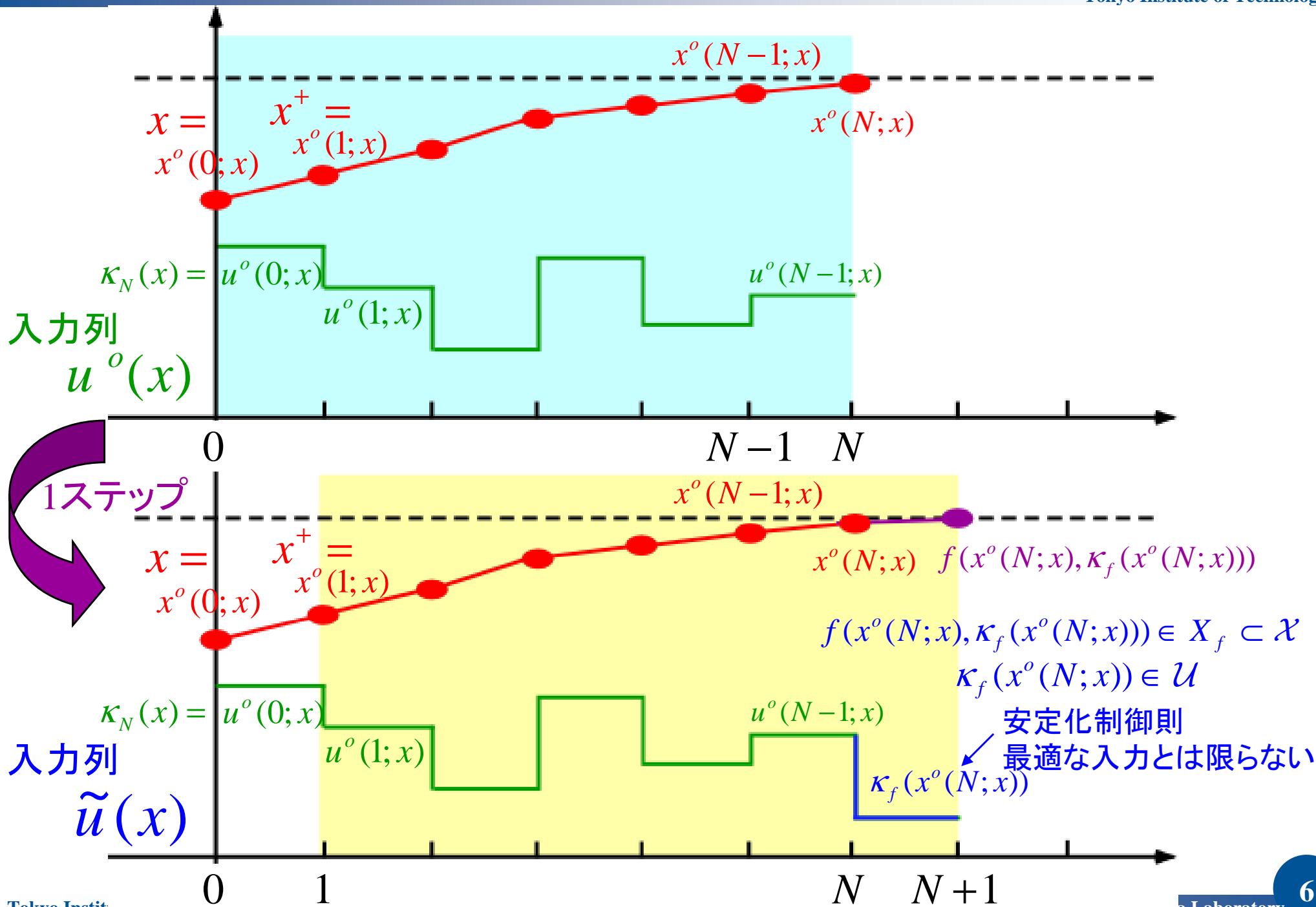
$$\tilde{u}(x) := \left\{ \underbrace{u^o(1; x), \dots, u^o(N-1; x)}_{\text{最適入力}}, \kappa_f(x^o(N; x)) \right\}$$

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



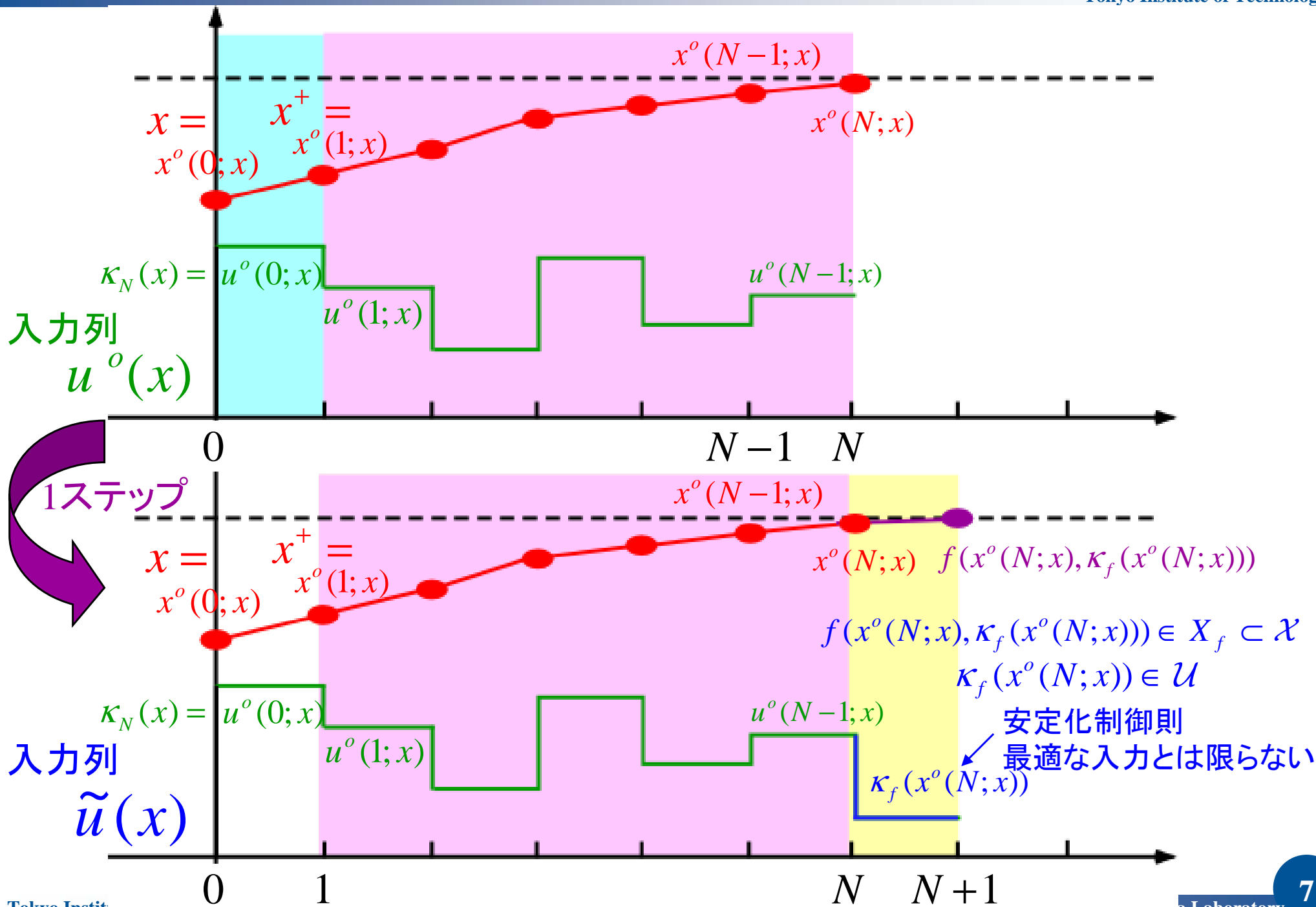


非線形モデル予測制御の安定条件の略証



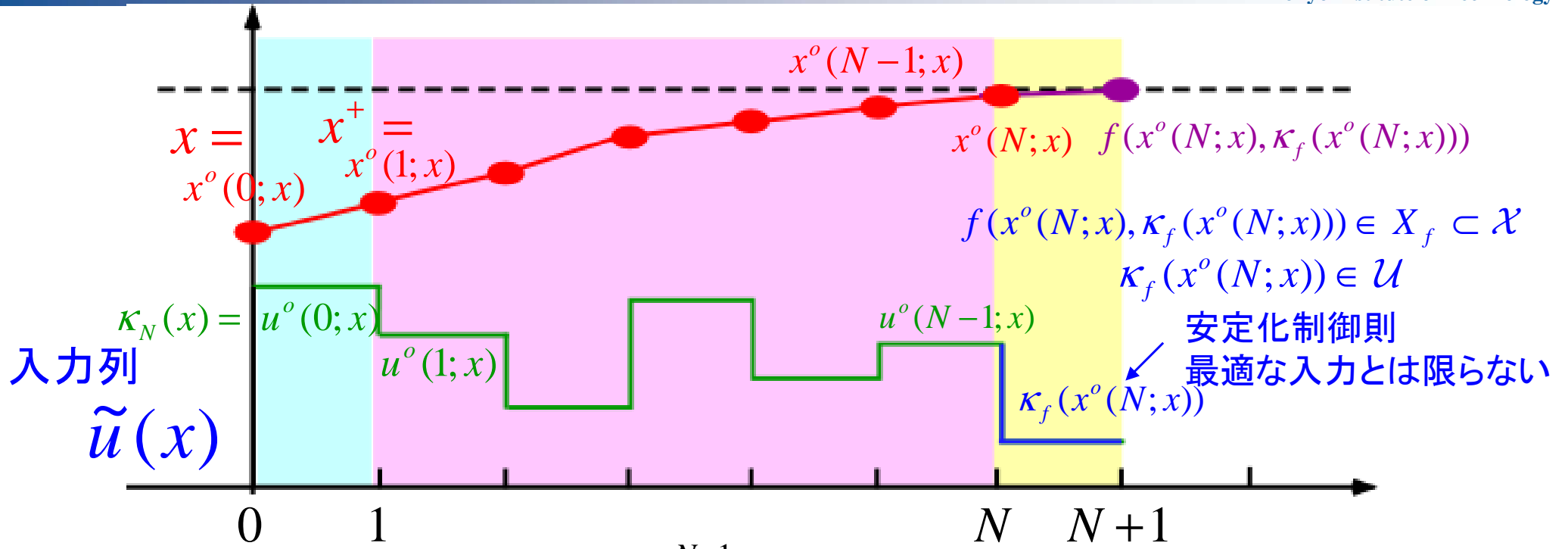


非線形モデル予測制御の安定条件の略証





非線形モデル予測制御の安定条件の略証



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

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

+

=

=

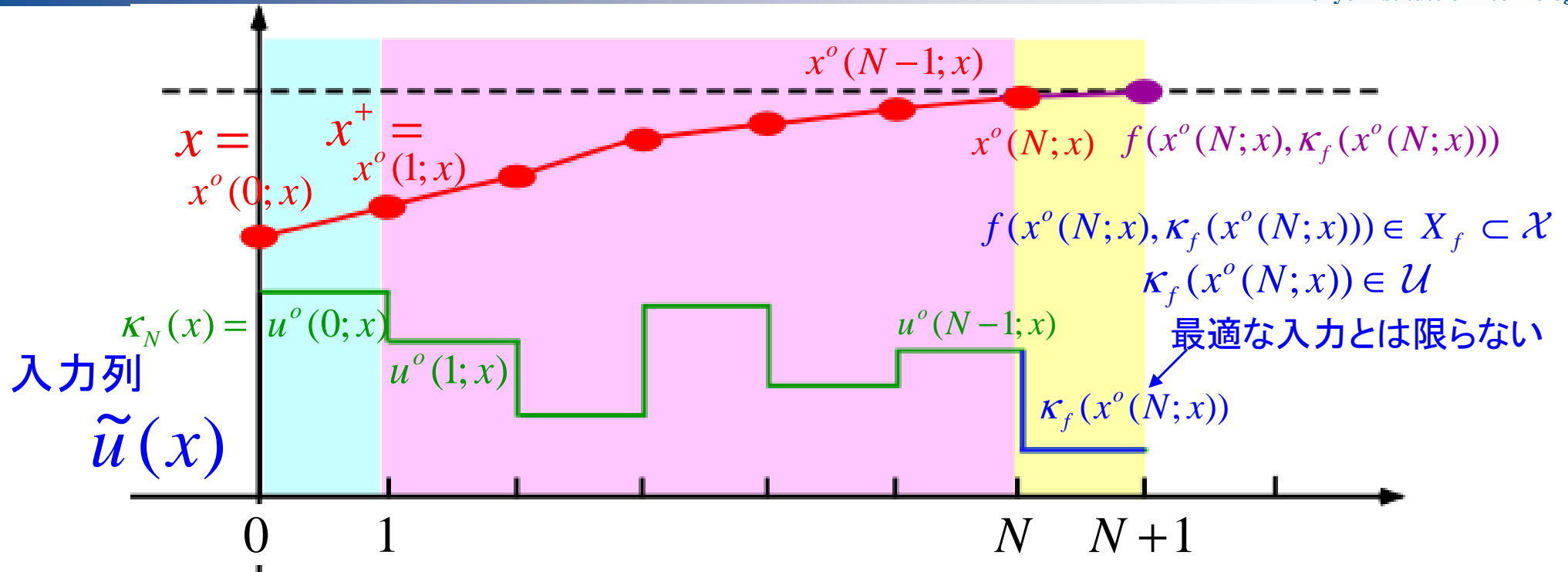
 $V_N(x(0; x), u^o(0; x))$

$$- \underbrace{l(x^o(0; x), u^o(0; x))}_{\text{cyan}} - \underbrace{F(x^o(N; x))}_{\text{green}}$$

$$+ \underbrace{l(x^o(N; x), \kappa_f(x^o(N; x)))}_{\text{yellow}} + \underbrace{F(f(x^o(N; x), \kappa_f(x^o(N; x))))}_{\text{green}}$$



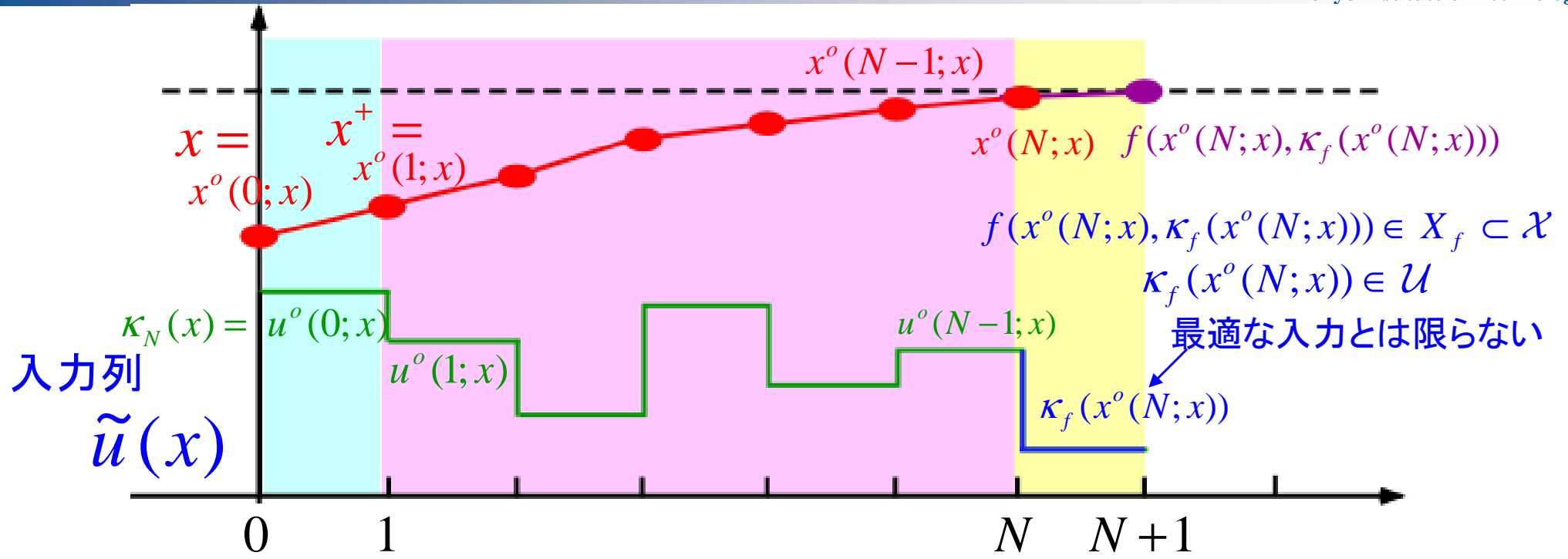
非線形モデル予測制御の安定条件の略証



$$\begin{aligned}
 V_N(x^+, \tilde{u}(x)) &= \underbrace{V_N(x(0; x), u^o(0; x))}_{= V_N^o(x)} - \underbrace{l(x^o(0; x), u^o(0; x))}_{= x} - \underbrace{F(x^o(N; x))}_{= \kappa_N(x)} \\
 &+ \underbrace{l(x^o(N; x), \kappa_f(x^o(N; x))) + F(f(x^o(N; x), \kappa_f(x^o(N; x))))}_{\text{terminal cost}} \\
 &= V_N^o(x) - l(x, \kappa_N(x)) + \underbrace{\Delta F(x^o(N; x), \kappa_f(x^o(N; x)))}_{\text{terminal cost difference}} \\
 &+ l(x^o(N; x), \kappa_f(x^o(N; x))) \\
 &\quad \because \Delta F(x, u) := F(x^+) - F(x)
 \end{aligned}$$



非線形モデル予測制御の安定条件の略証

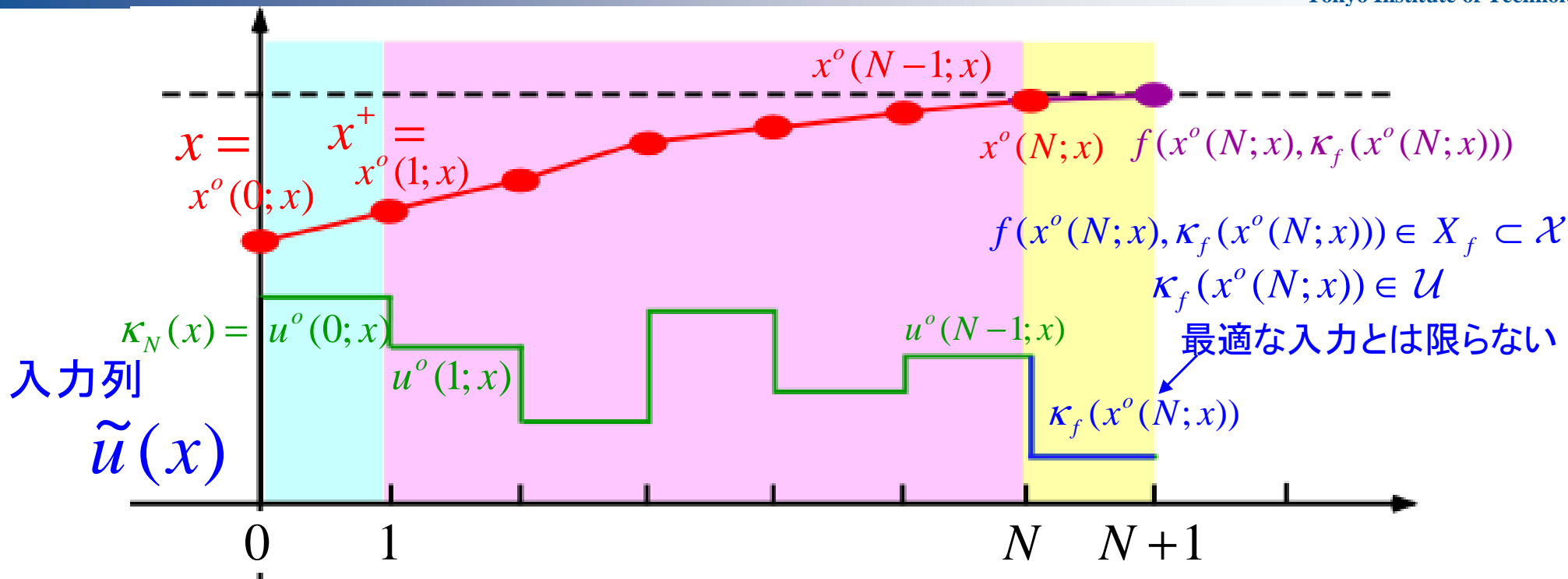


$$\begin{aligned}
 V_N(x^+, \tilde{u}(x)) &= V_N^o(x) - l(x, \kappa_N(x)) + \underline{\Delta F(x^o(N; x), \kappa_f(x^o(N; x)))} \\
 &\quad + \underline{l(x^o(N; x), \kappa_f(x^o(N; x)))} \\
 &= V_N^o(x) - l(x, \kappa_N(x)) + \underline{[\Delta F + l](x^o(N; x), \kappa_f(x^o(N; x)))} \\
 &\leq V_N^o(x) - l(x, \kappa_N(x)) \leq 0
 \end{aligned}$$

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



非線形モデル予測制御の安定条件の略証



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

一方 $\underline{\tilde{u}(x)} := \{u^o(1;x), \dots, u^o(N-1;x), \kappa_f(x^o(N;x))\}$

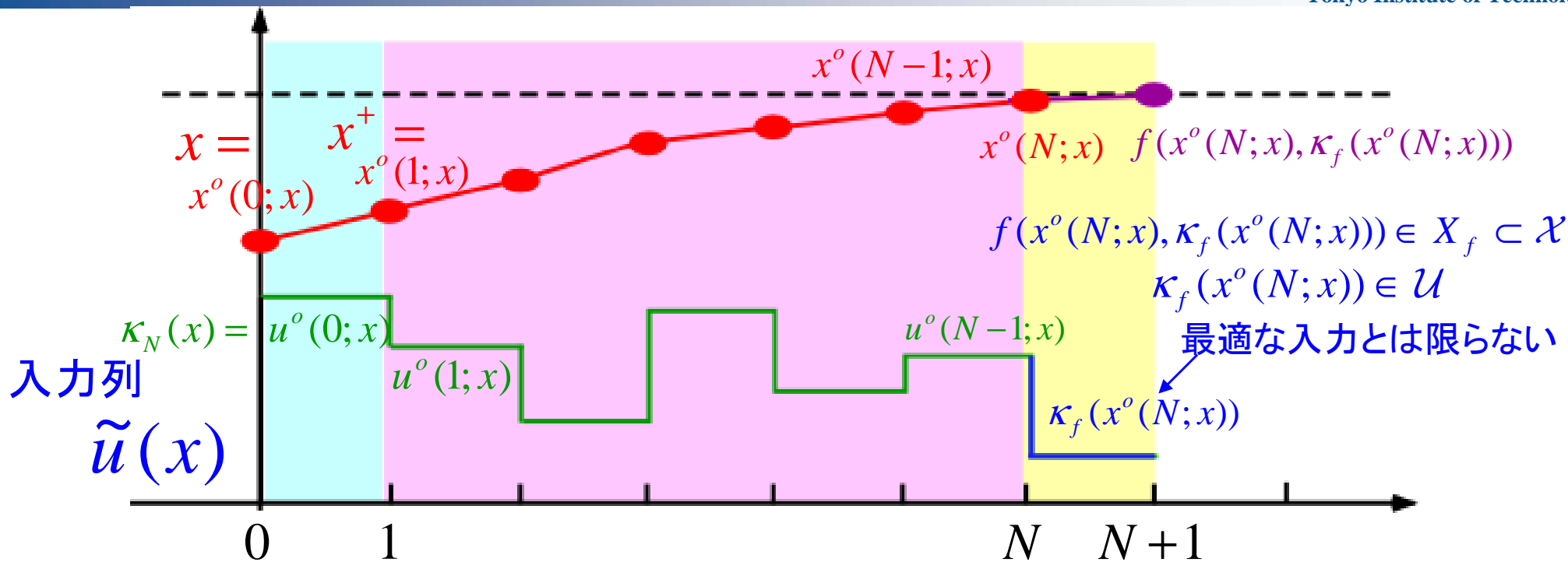
最適入力とは限らない

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

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



非線形モデル予測制御の安定条件の略証



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

$$\Delta V_N^o(x, \kappa_N(x)) \leq -l(x, \kappa_N(x)) \quad \because \Delta V_N^o(x, u) := V_N^o(x^+) - V_N^o(x)$$

value function $V_N^o(\cdot) > 0$

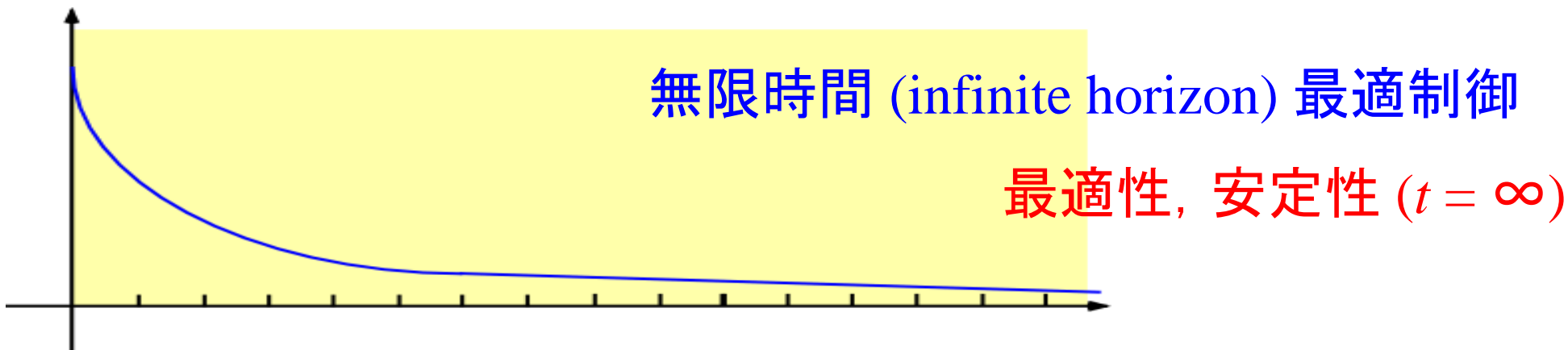
value function $V_N^o(\cdot)$ をリアプノフ関数候補として用い、
 時間の経過とともに $V_N^o(\cdot)$ の値が単調減少し原点 $x=0$ へ



閉ループ系の漸近安定性



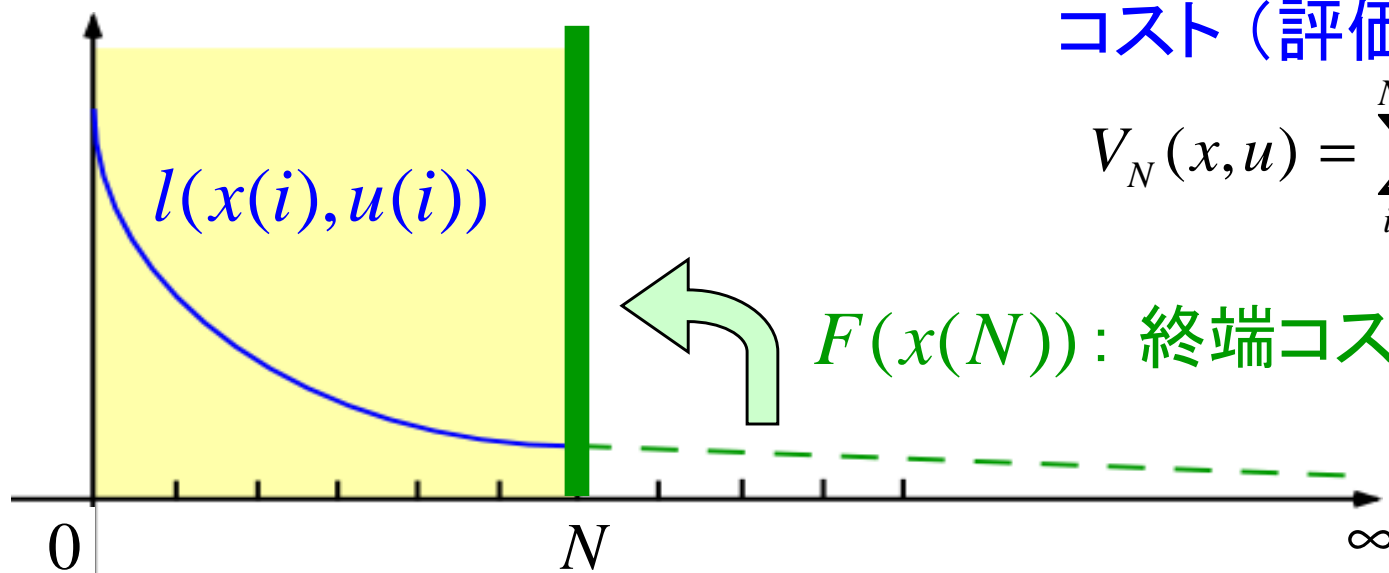
終端コストの意味づけ



モデル予測制御 (receding horizon control)

コスト (評価関数)

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



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



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

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

モデル予測制御の安定条件 A4:

$$[\Delta F + l](x, \kappa_f(x)) \leq 0, \forall x \in X_f \text{ より求められる条件である}$$

Nステップ以降
操作量は0

$\kappa_f(x) = 0$ としてA4を満たす終端コストを考える

リアプノフ方程式 $A_f^T P_f A_f + Q_f = P_f$ を満たす

$$\begin{aligned} A_f &:= A + BK_f \\ Q_f &:= Q + K_f^T R K_f \\ \kappa_f(x) &:= K_f x \end{aligned}$$

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

* 実際にはリアプノフ方程式を直接計算する



評価関数

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

$P_f = \sum_{k=0}^{\infty} (A_f^k)^T Q_f A_f^k$

0 ~ ∞ までの評価関数

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



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

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

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

$$\text{最適入力 } u^o(k) = Kx(k)$$

$$K = -(R + B^T P B)^{-1} B^T P A$$

$N \sim \infty$ の最適なコスト

モデル予測制御の評価関数

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

終端に
込める!

$x(N) \in \mathcal{X}_f$ (終端制約)

制約条件

$$u(k) \in \mathcal{U} \quad k=0, \dots, N-1$$

$$x(k) \in \mathcal{X} \quad k=0, \dots, N$$

$$x(N) \in \mathcal{X}_f \subset \mathcal{X}$$

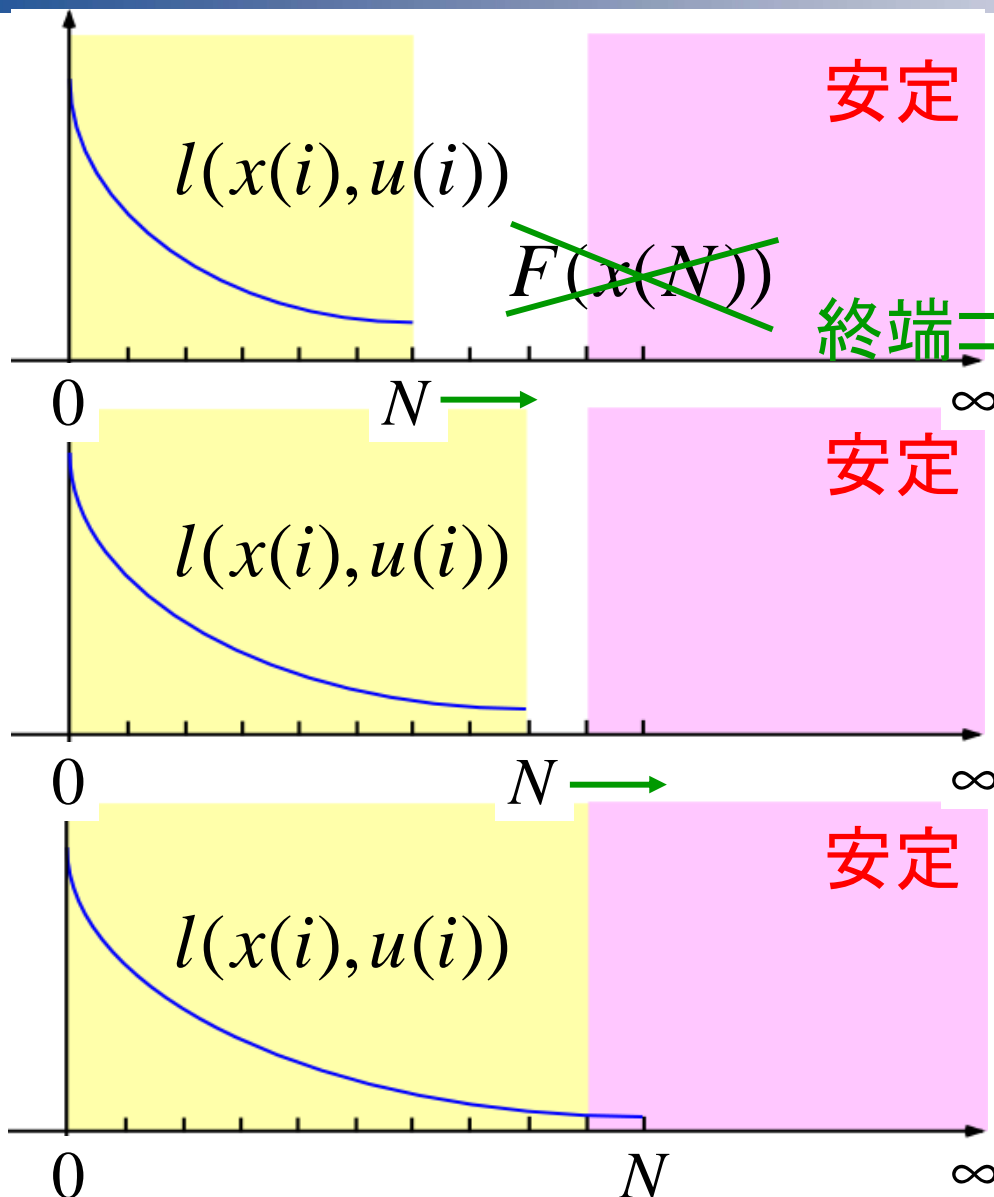
\mathcal{X}_f における安定化入力

$$K_f(x) = Kx(k)$$

$$K = -(R + B^T P_f B)^{-1} B^T P_f A$$



その他の安定性の証明方法



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

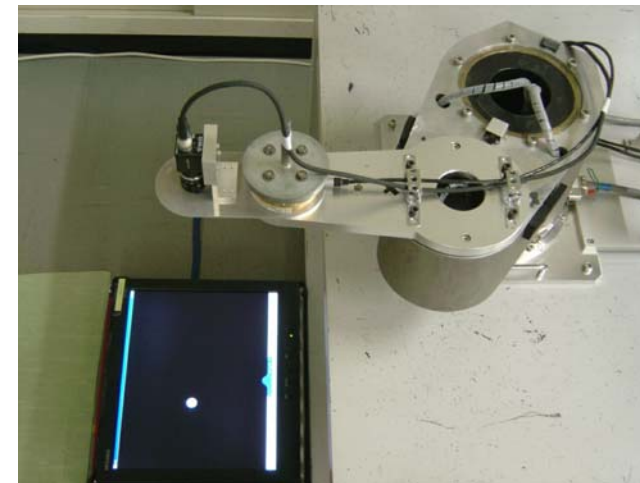


Fig : Visual Feedback System



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

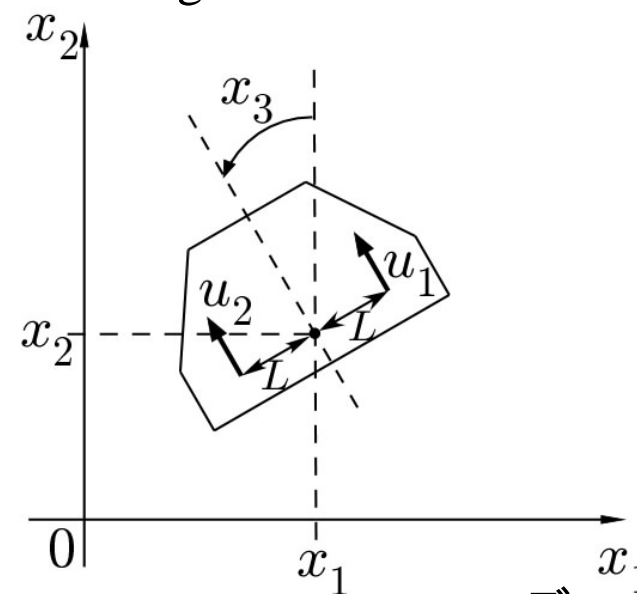


Fig : Hover Craft モデル

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.



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)

$$J = \underbrace{\varphi(x(T))}_{\text{終端コスト}} + \int_t^{t+T} \underbrace{L(x(t'), u(t'))}_{\text{ステージコスト}} dt'$$

$$\underline{L} = \frac{1}{2} \left\{ (p(t) - x(t))^T Q (p(t) - x(t)) + u^T u \right\} : \text{ステージコスト (stage cost)}$$

$$\underline{\varphi} = \frac{1}{2} (p(t) - x(t))^T S_f (p(t) - x(t)) : \text{終端コスト (terminal cost)}$$

$$Q > 0, \quad S_f > 0$$

$$p(t) = \left[0 \quad 0 \quad \frac{\pi}{2} \right] : \text{目標値}$$

$$T(t) = T_f (1 - e^{-\alpha t}) : \text{評価区間}$$

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



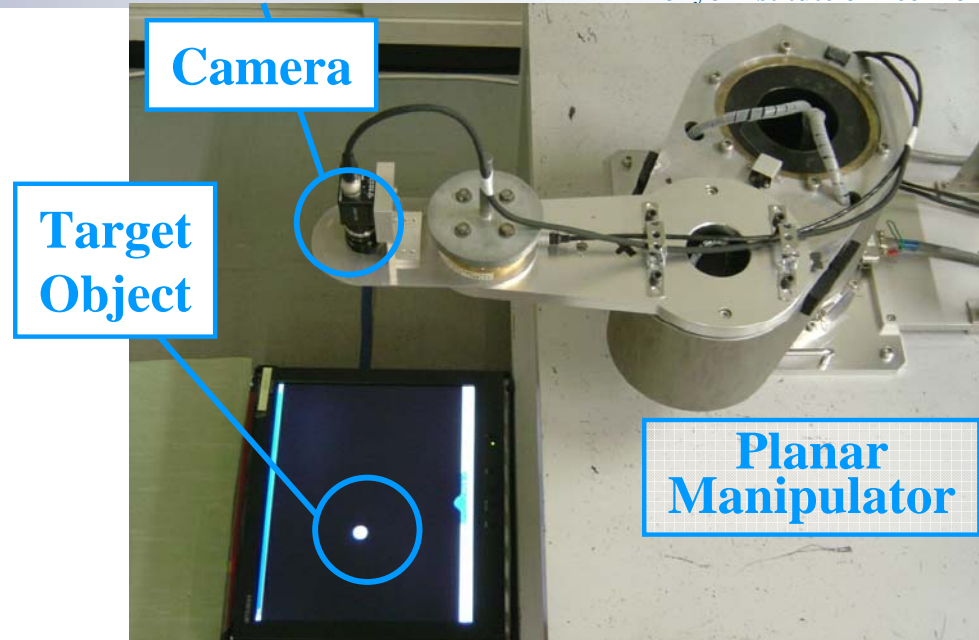
<http://www-newton.mech.eng.osaka-u.ac.jp/~ohtsuka/reseach.htm>



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 : Visual Feedback System

$Q > 0 \quad R^{-1} > 0$

$V(x)$: エネルギー関数

コスト (評価関数)

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

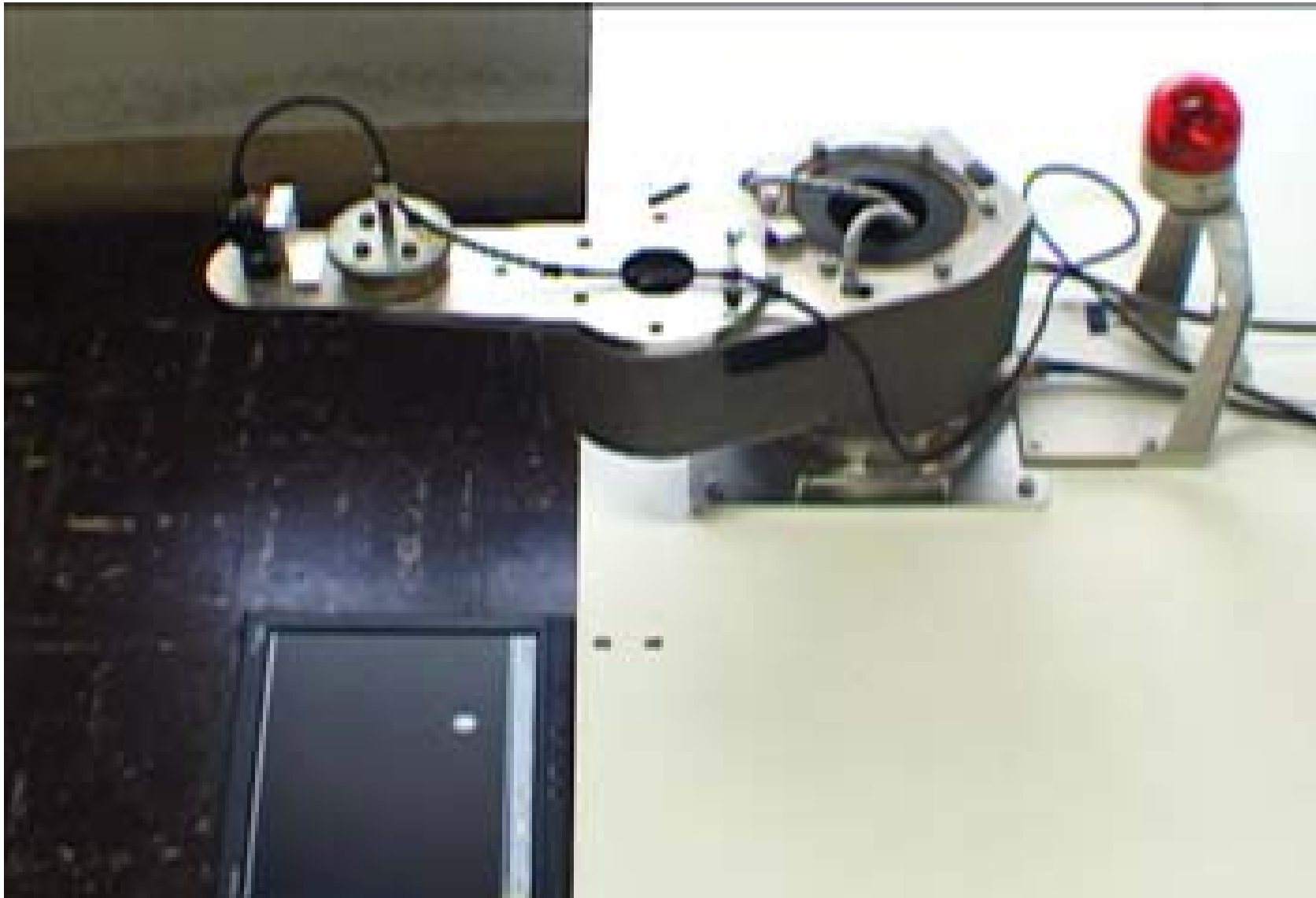
$l(x(\tau), u(\tau)) = 4x^T Qx + u^T R^{-1}u$: ステージコスト (stage cost)

$M(x(t+T)) = 4V(x(t+T))$: 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





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

制約(拘束), 故障

設定値

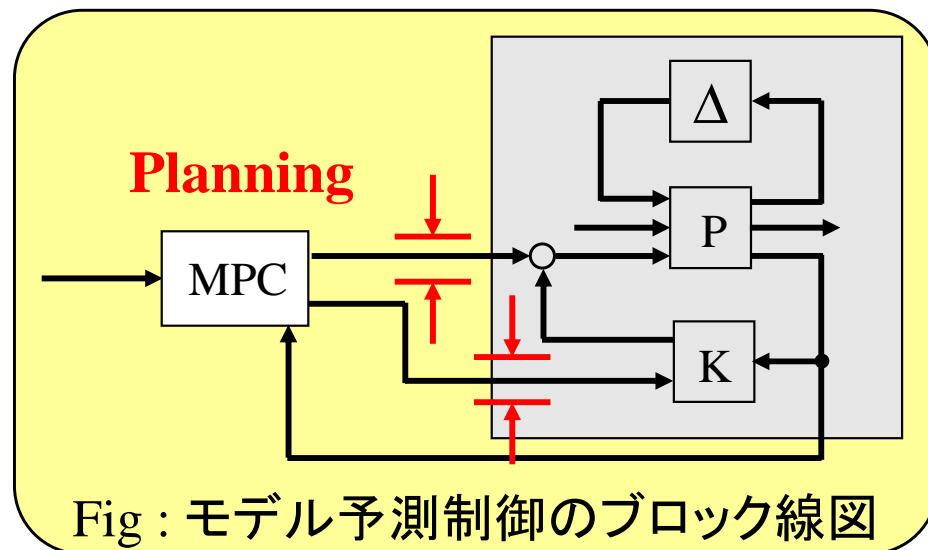
直接(ローカル)制御

不確かさ

操作量

問題点

制御対象



制御システムの構造,
評価関数, パラメータ値

モデル予測制御

最適化

ロバスト制御

フィードバック

制御手法



* 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

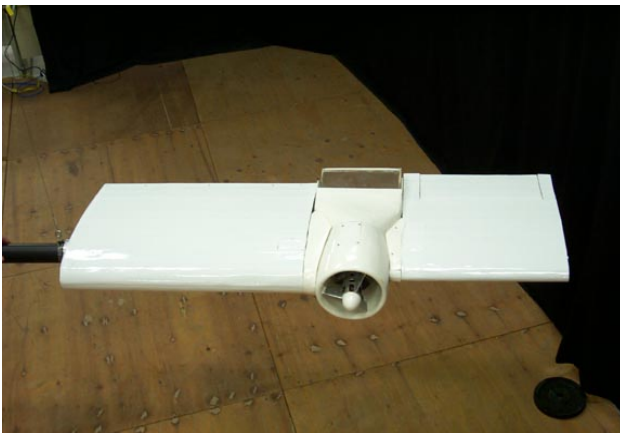


Fig: Caltech ducted fan



Fig: Alice



Fig: F-16



* 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

$$\begin{cases} m\ddot{x} + F_{X_a} - F_{X_b} \cos \theta - F_{Z_b} \sin \theta = 0 \\ m\ddot{z} + F_{Z_a} + F_{X_b} \sin \theta - F_{Z_b} \cos \theta = mg \\ J\ddot{\theta} - M_a + \frac{1}{r_s} I_p \Omega \dot{x} \cos \theta - F_{Z_b} l_\tau = 0 \end{cases}$$

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

入力 $u = [F_{X_b}, F_{Z_b}]^T$

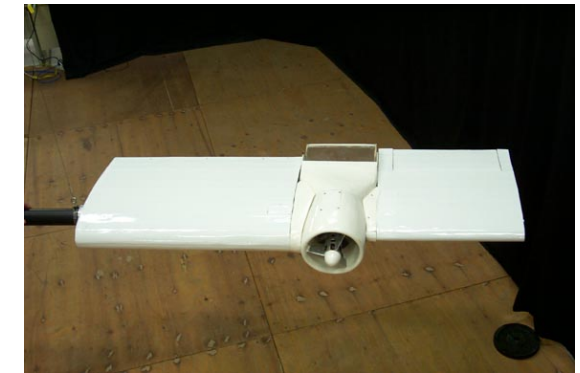


Fig : Caltech ducted fan

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

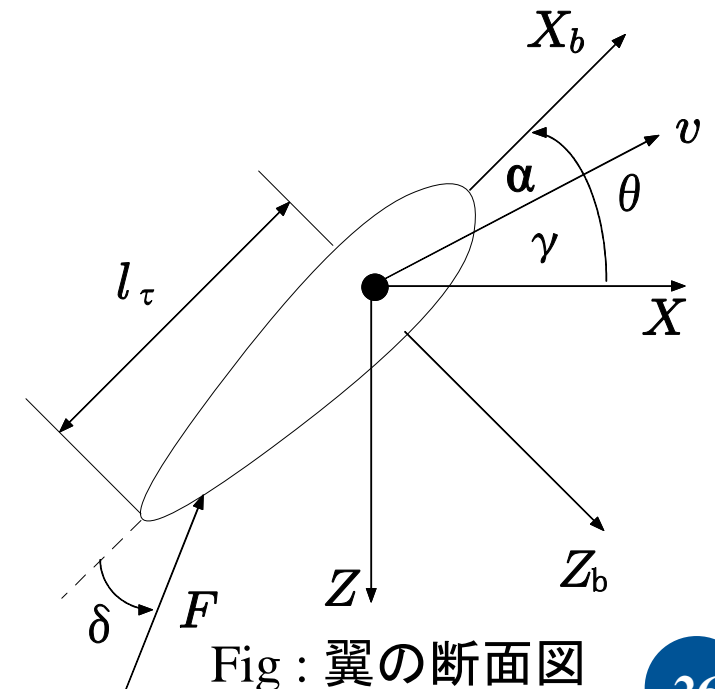


Fig : 翼の断面図

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



* 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 \mathcal{U} : 与える力 F_{X_b} 0 [N] \sim 13 [N]
 F_{Z_b} -6.5 [N] \sim 6.5 [N]

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

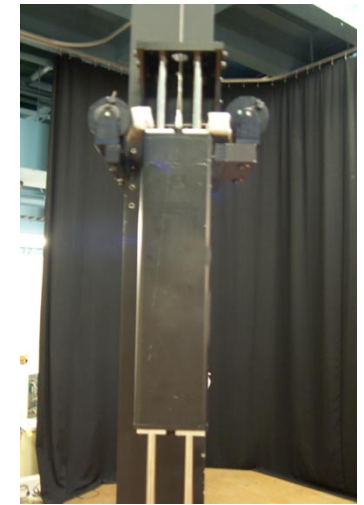


Fig : 中央スタンド

コスト (評価関数)

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

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

: ステージコスト (stage cost)

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

: 終端コスト (terminal cost)

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

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

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.



* モデル予測制御の応用例 (Caltech ducted fan)

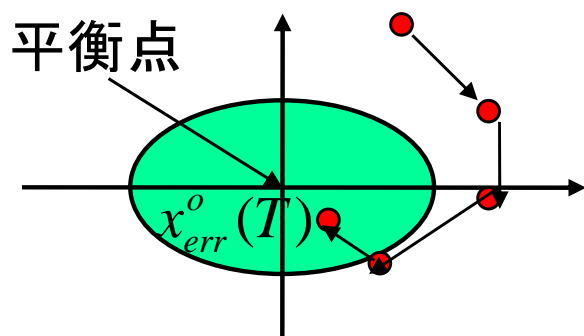
安定性 \longrightarrow 終端コストにCLF (Control Lyapunov Function)

$$\text{CLF: つぎを満たす正定関数 } V: R^n \rightarrow R_+$$

$$\inf_u \left[\dot{V}(x, u) + q(x, u) \right] \leq 0$$

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 \longrightarrow 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 $T \geq 0.6$



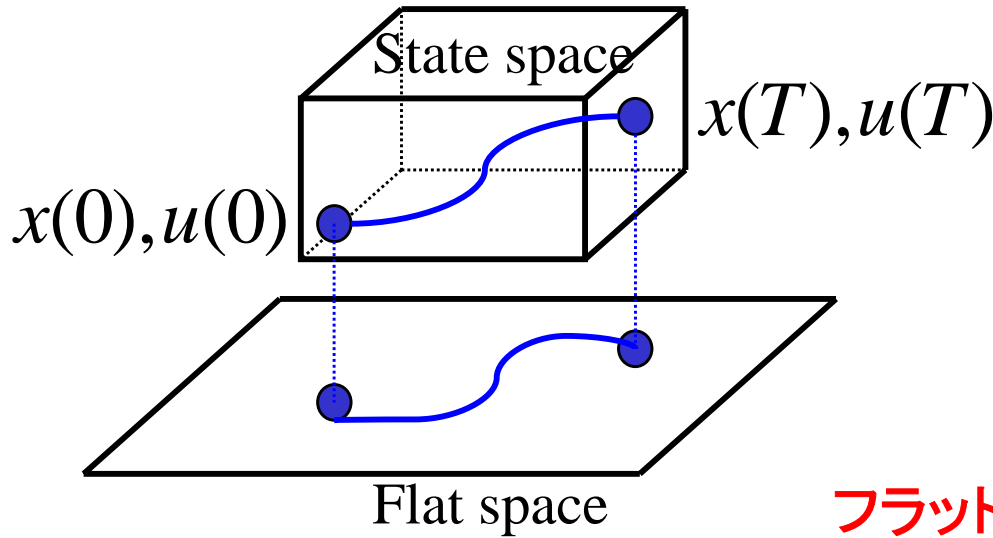
ホライズン長い \longrightarrow $x_{err}^o(T)$ が **平衡点近傍へ**
 LQRに基づくCLF \cong コストの最適値 J_∞^o

2. 状態変数をパラメータとしLMIsを解く. \longrightarrow 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



* モデル予測制御の応用例 (Caltech ducted fan)

Differential Flatnessによる最適化



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

フラット出力 $z = A(x, u, u^{(1)}, \dots, u^{(r)})$

$$(x, u) = B(z, z^{(1)}, \dots, z^{(s)})$$

フラット出力 $z = [z_1, z_2, \dots, z_q]^T$

$$z_1 = \sum_{i=1}^{p_1} B_{i,k_1}(t) C_i^1 \quad \min_{y \in R^M} F(y) \quad s.t. \quad lb \leq c(y) \leq ub$$

$$z_2 = \sum_{i=1}^{p_2} B_{i,k_2}(t) C_i^2 \quad y = (C_i^1, C_i^2, \dots, C_i^q)$$

係数

係数の最適化

(計算時間の短縮)

$$z_q = \sum_{i=1}^{p_q} B_{i,k_q}(t) C_i^q$$

B-spline 基底関数

フラット出力の最適化によって、軌道計画とそのため必要な入力が求まる。



実時間軌道生成



* モデル予測制御の応用例 (Caltech ducted fan)

Tokyo Institute of Technology



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



* Real-Time Trajectory Generation

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

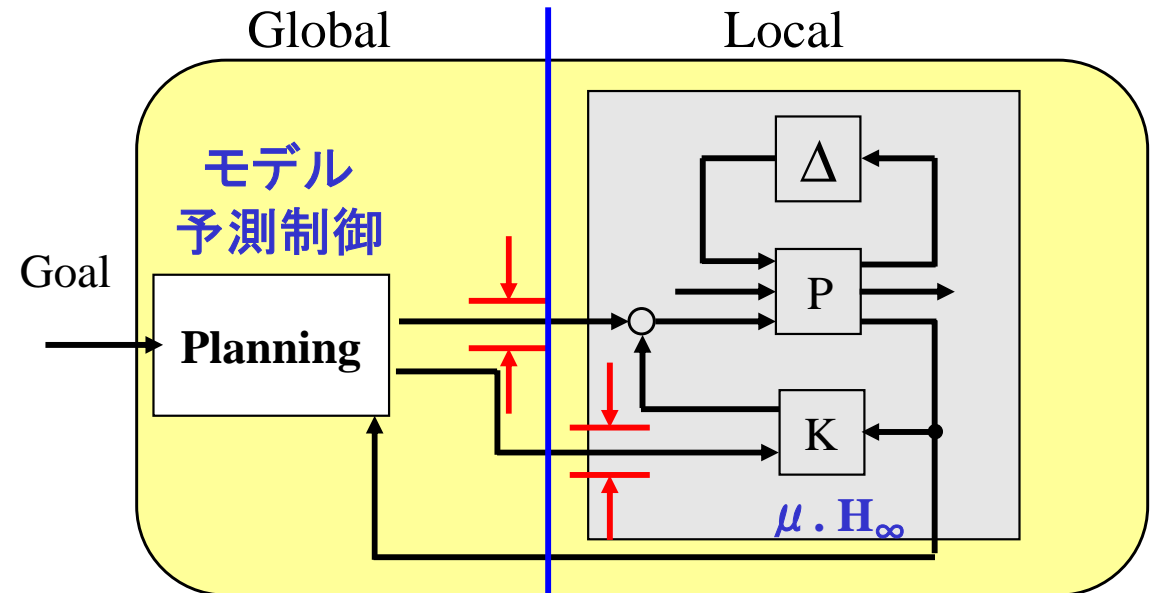


Fig. Planningのブロック線図



Fig. Alice (Team Caltech)

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



* Alice

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

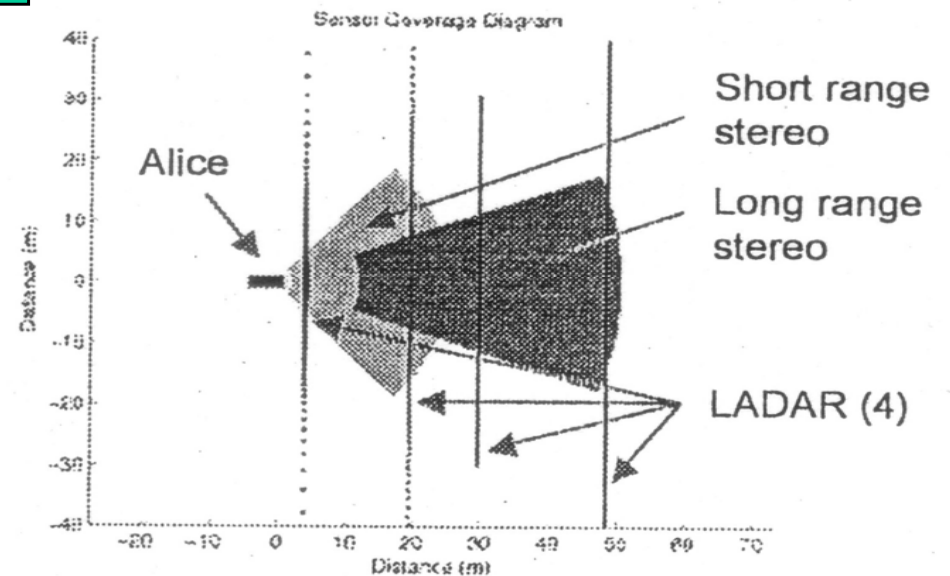
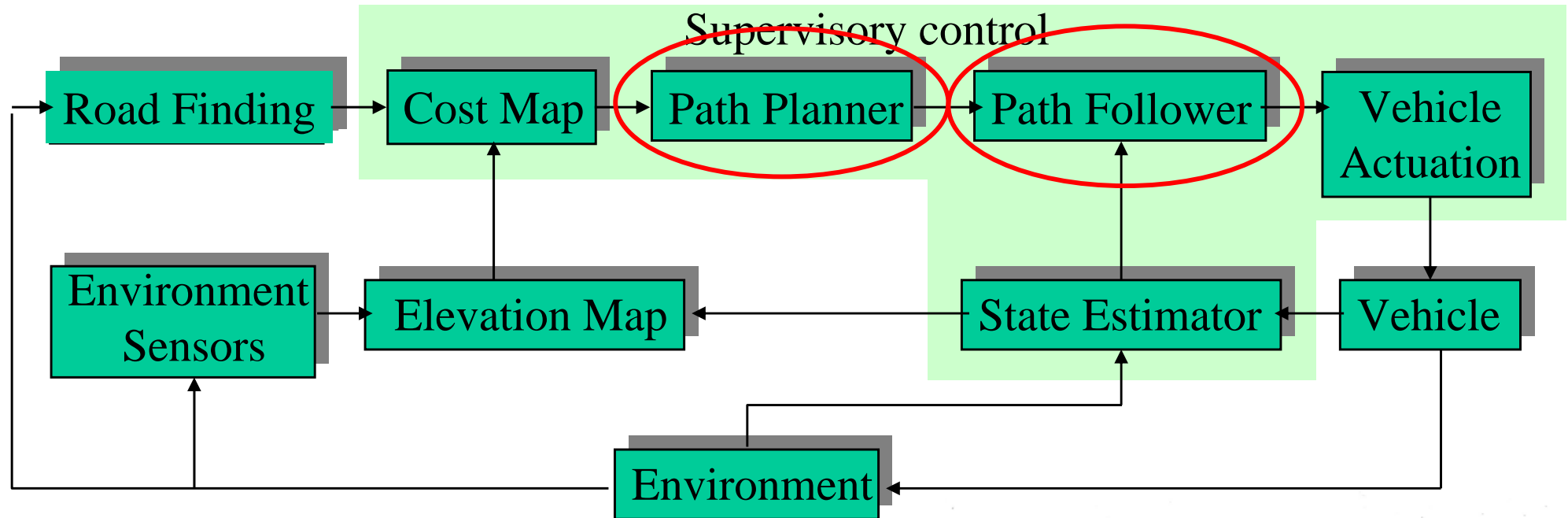


<http://team.caltech.edu/index.html>



* Alice

Alice's Architecture





Real-Time Trajectory Generation

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

ex.

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

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

所要時間の最小化

$$L(x, u) = u_1^2 + u_2^2 \quad \text{入力の最小化}$$

s.t

ダイナミクス

制約

$$\dot{x} = v \cos \theta$$

$$\dot{y} = v \sin \theta$$

$$\dot{\theta} = \frac{v}{L} \tan \phi$$

$$\dot{\phi} = u_1$$

$$\dot{v} = u_2$$

$$\phi \in [\phi_{\min}, \phi_{\max}]$$

$$u_1 \in [u_{1\min}, u_{1\max}]$$

$$u_2 \in [u_{2\min}, u_{2\max}]$$

$$v \in (0, v_{\max}]$$

障害物回避

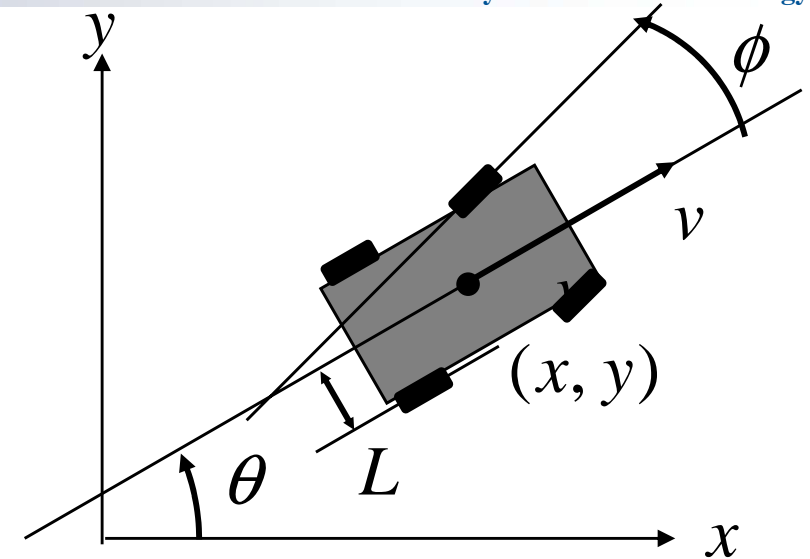


Fig. Aliceのモデル

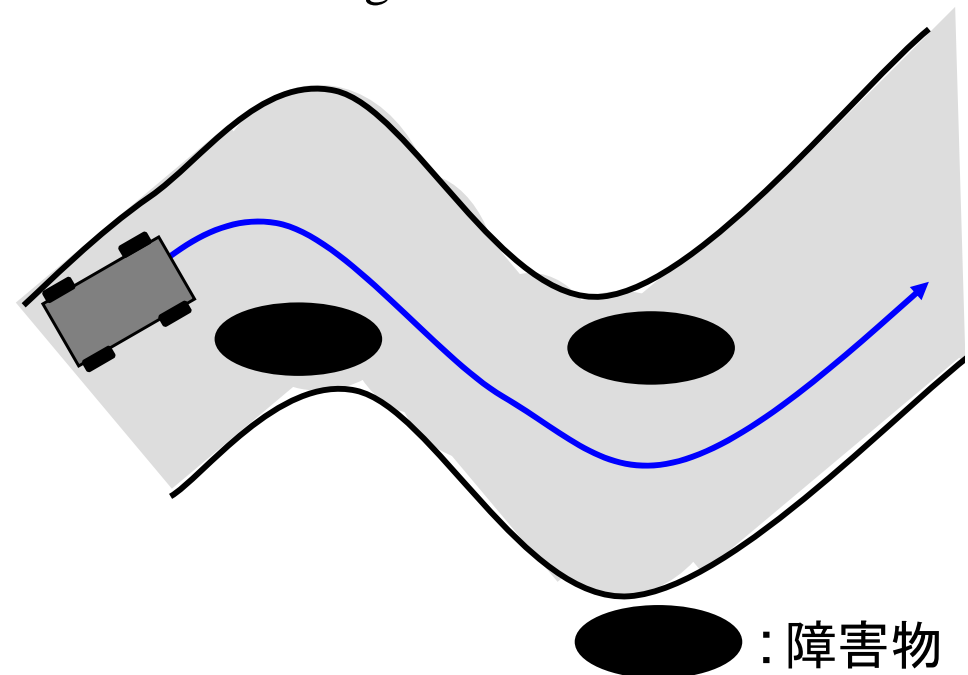
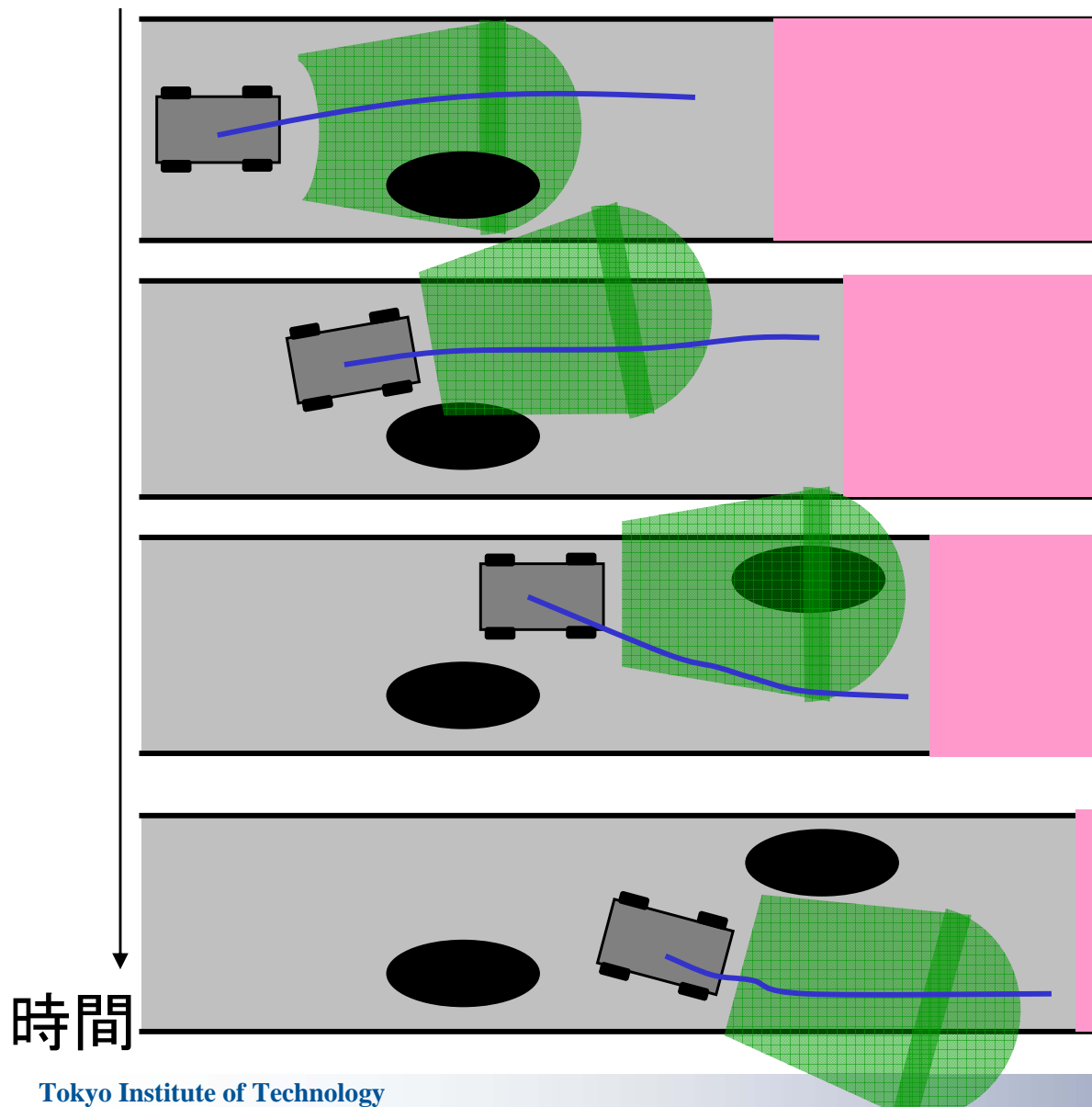


Fig. 経路生成の例



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

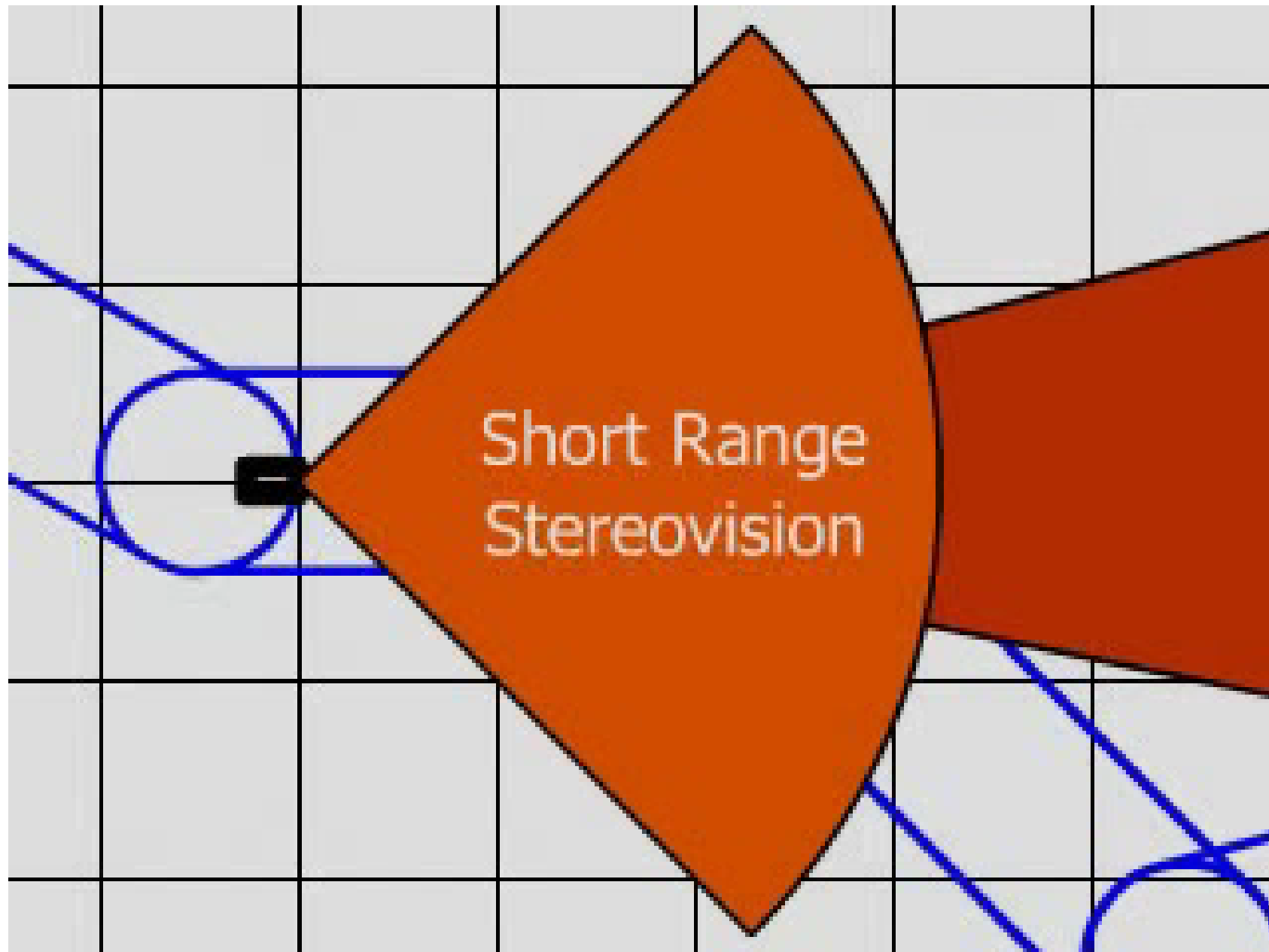
黒: 障害物

ピンク: 未知の場所

灰色: 通過可能な場所



* Alice



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

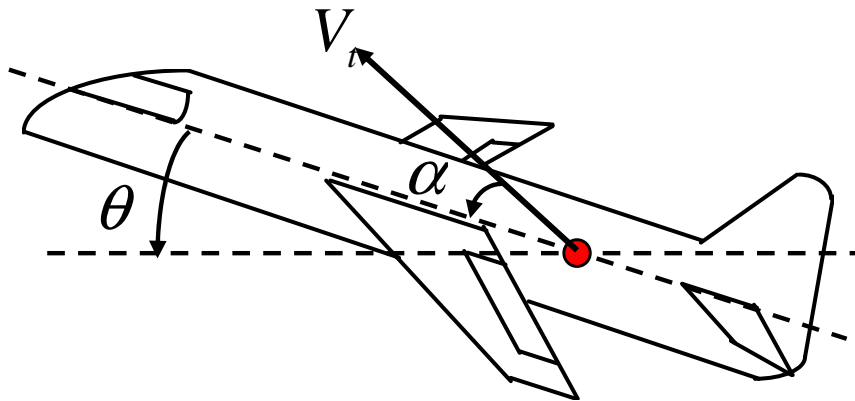


* モデル予測制御の応用例 (F-16)



Fig: F-16

$$y = \begin{bmatrix} h \\ V_t \end{bmatrix} \quad r = \begin{bmatrix} \delta_{th \text{ cmd}} \\ q_{cmd} \end{bmatrix}$$



F-16への入力

$$u = \begin{bmatrix} \delta_{th} & \dot{\delta}_{th} & \delta_e & \dot{\delta}_e \end{bmatrix}^T$$

δ_{th} : 推力 δ_e : 昇降舵角
 (速度制御) (姿勢制御)

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

$$x(k+1) = A_k x(k) + B_k r(k)$$

$$\begin{bmatrix} y(k) \\ z(k) \\ u(k) \end{bmatrix} = C_k x(k) + D_k r(k)$$

h : 高度
 θ : ピッチ角
 V_t : 対気速度
 α : 迎え角
 $-10^\circ \leq \alpha \leq 45^\circ$
 q : ピッチ角速度

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

$$A_k = A(\rho(k)) \quad B_k = B(\rho(k))$$

$$C_k = C(\rho(k)) \quad D_k = D(\rho(k))$$

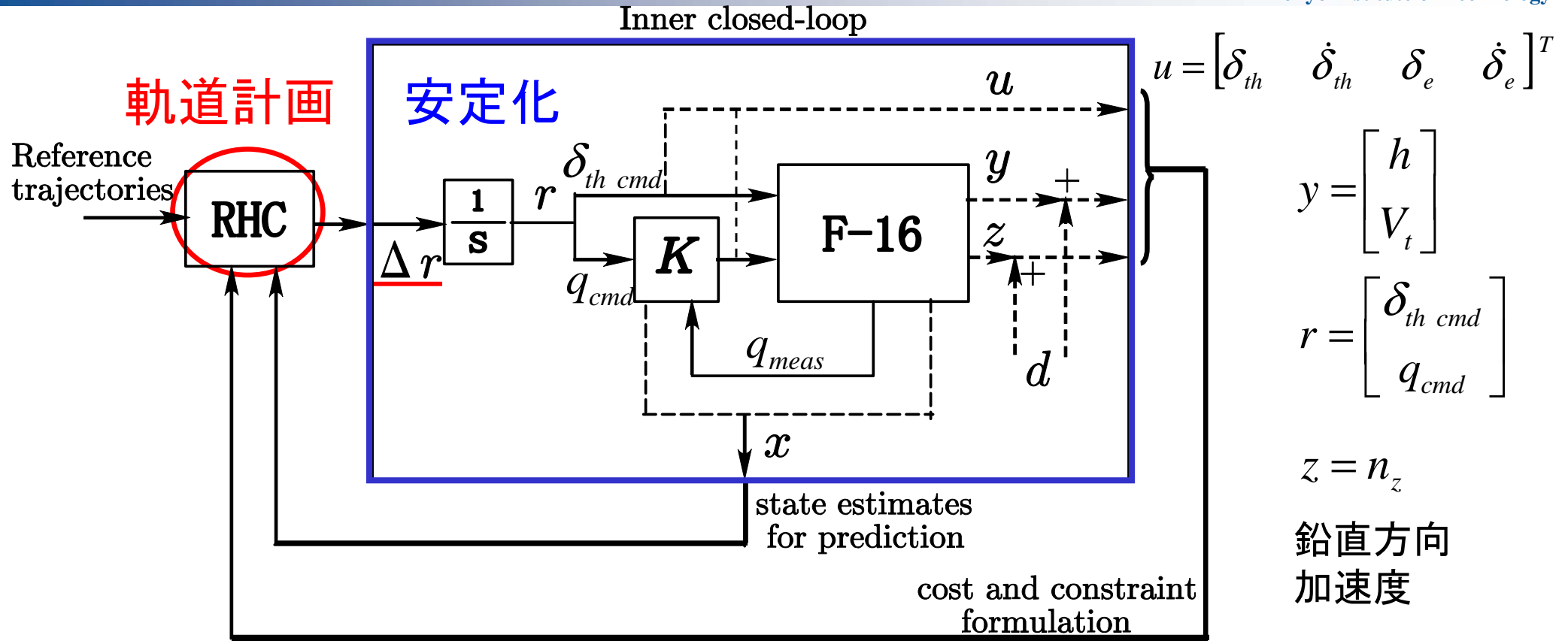
$\bar{q}(k)$: 動圧 (正面から受ける空気の圧力)

$M(k)$: マッハ数 (音速に対する飛行機の速度)

[students.ceid.upatras.gr/
~kouroupa/Images/F16.jpg](http://students.ceid.upatras.gr/~kouroupa/Images/F16.jpg)



* モデル予測制御の応用例 (F-16)



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 **ポンド**
目標値への追従制御

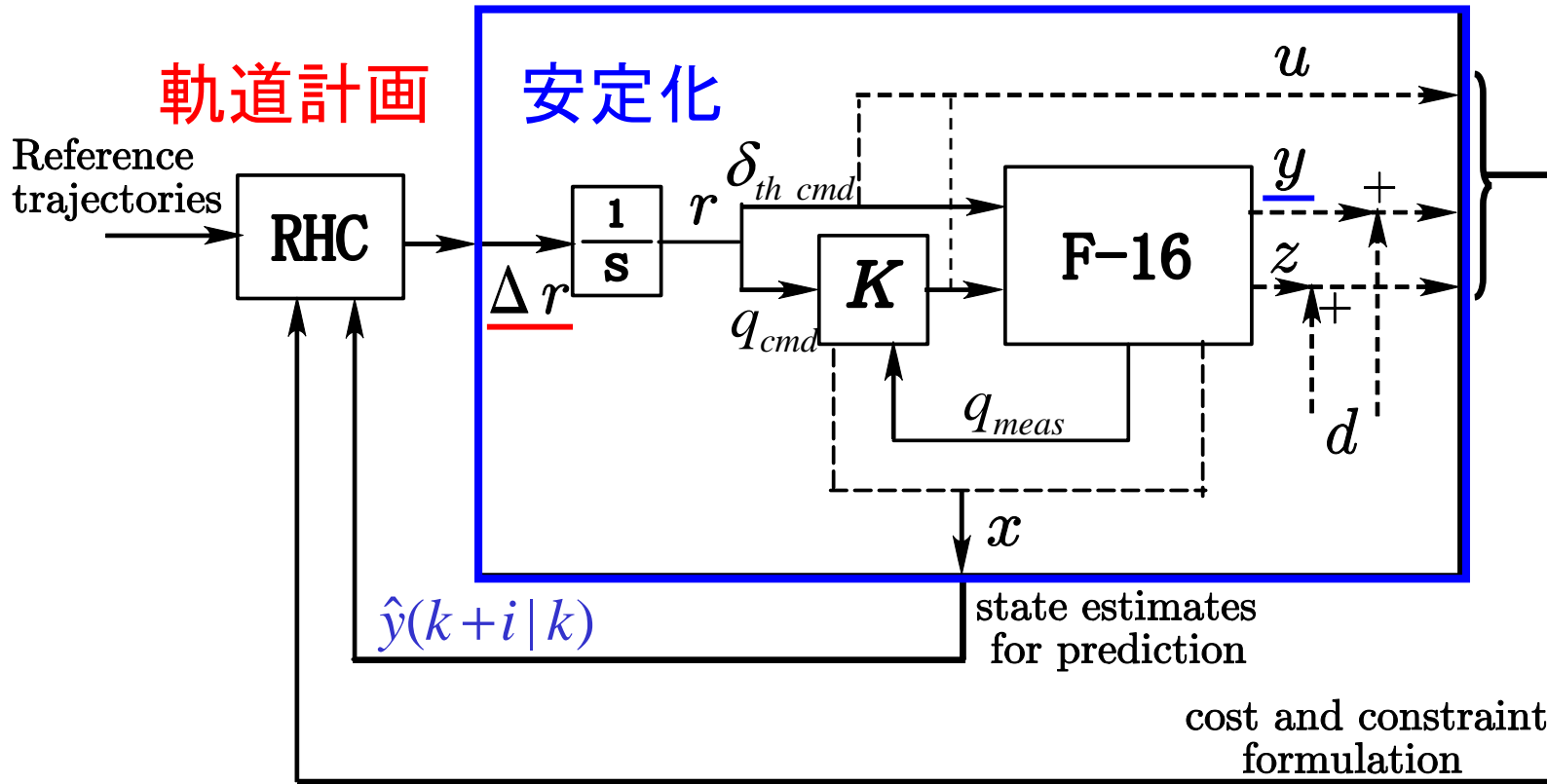
制御目標: 入力や状態の制約を満たしつつ、
目標の高度・速度に保つ, $y \rightarrow y_{ref}$

- 入力制約 $1000 \leq \delta_{th} \leq 19000$ (lb)
- $-10000 \leq \dot{\delta}_{th} \leq 10000$ (lb / s)
- $-25 \leq \delta_e \leq 25$ (°)
- $-60 \leq \dot{\delta}_e \leq 60$ (° / s)



* モデル予測制御の応用例 (F-16)

Inner closed-loop



$$r = \begin{bmatrix} \delta_{th\ cmd} \\ q_{cmd} \end{bmatrix}$$

$$\Delta \delta_e = K(s) \begin{bmatrix} q_{cmd} \\ q_{meas} \end{bmatrix}$$

内部モデルの
コントローラ

$$\hat{y}(k+i|k)$$

kステップまでの
出力に基づく
iステップ先の
予測出力

内部モデルのシステム (linear prediction model)

$$\hat{\xi}(k+1) = \mathbf{A}_k \hat{\xi}(k) + \mathbf{B}_k \Delta r(k)$$

$$\hat{w}(k) = \mathbf{C}_k \hat{\xi}(k) + \mathbf{D}_k \Delta r(k)$$

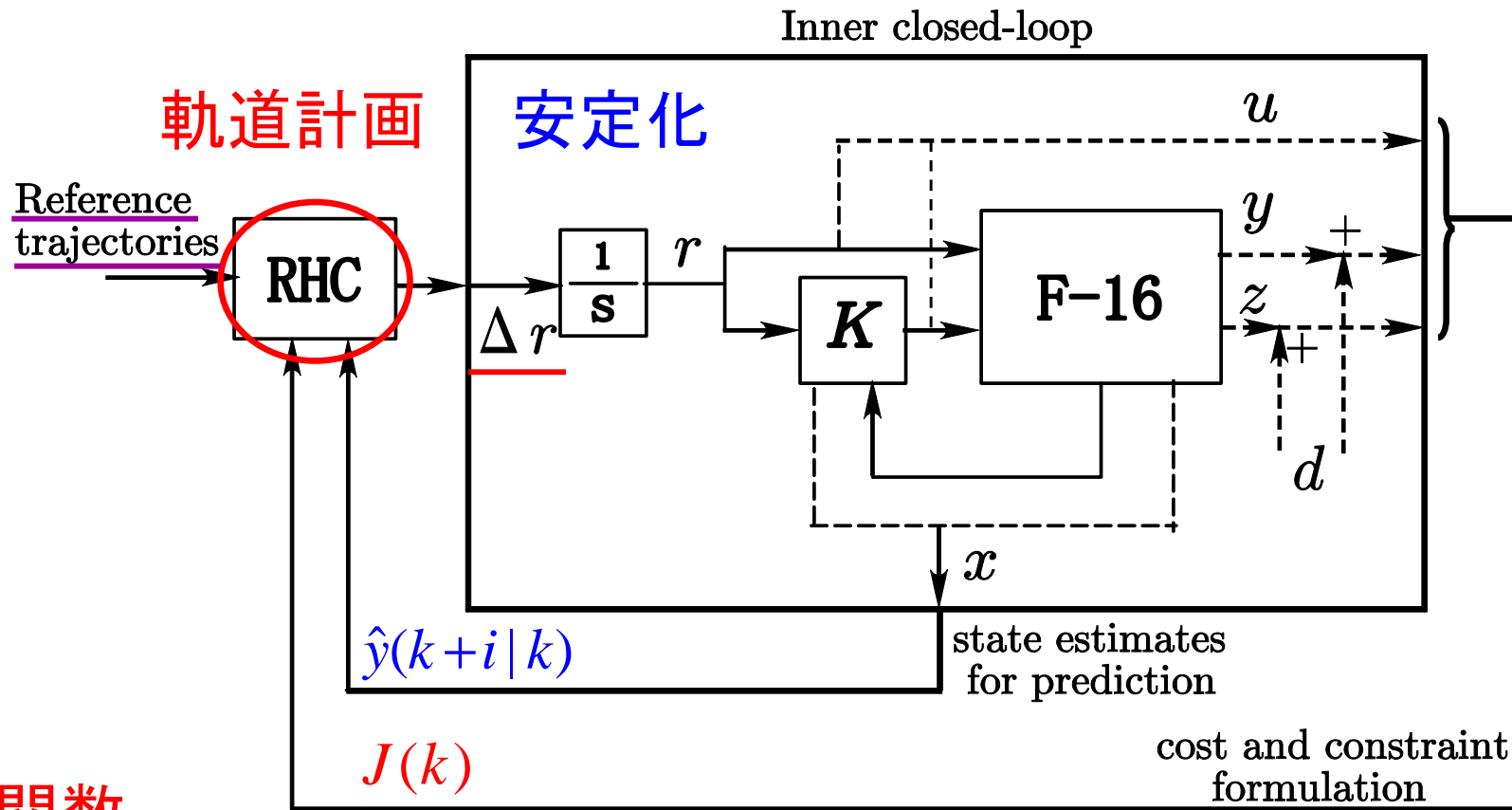
$$\hat{\xi}(k) = \begin{bmatrix} \hat{x}(k) \\ \hat{d}(k) \\ r(k-1) \end{bmatrix} \quad \hat{w}(k) = \begin{bmatrix} \hat{y}(k) \\ \hat{z}(k) \\ \hat{u}(k) \end{bmatrix}$$

○ は $\rho(k) = [\bar{q}(k) M(k)]^T$ に依存

$$\hat{d}(k+1) = \hat{d}(k) + \begin{bmatrix} y(k) - \hat{y}(k|k-1) \\ z(k) - \hat{z}(k|k-1) \end{bmatrix}$$



* モデル予測制御の応用例 (F-16)



評価関数

$$J(k) = \sum_{i=1}^{H_p} \left\| \underbrace{\hat{y}(k+i|k)}_{\text{予測出力}} - \underbrace{y_{ref}(k+i|k)}_{\text{目標出力}} \right\|_Q^2 \xrightarrow{\text{最適化}} \underbrace{\Delta r(k+i|k)}_{\text{内部モデル}} \quad i = 0, \delta_{H_c}, 2\delta_{H_c}, \dots, H_c - 1$$

H_c 制御ホライゾン

$$+ \sum_{i=0}^{H_c-1} \left\| \underbrace{\Delta r(k+i|k)}_{\text{内部モデル}} \right\|_R^2 + \rho \varepsilon$$

H_p 予測出力ホライゾン



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