

Title	拘束条件を有する系に対するスイッチング制御に関する研究
Author(s)	亀井, 泉寿
Citation	
Issue Date	1999-03
Type	Thesis or Dissertation
Text version	author
URL	<a href="http://hdl.handle.net/10119/1255">http://hdl.handle.net/10119/1255</a>
Rights	
Description	Supervisor: 藤田 政之, 情報科学研究科, 修士

修士論文

拘束条件を有する系に対するスイッチング制御  
に関する研究

指導教官 藤田政之 助教授

北陸先端科学技術大学院大学  
情報科学研究科情報システム学専攻

亀井泉寿

1999年2月15日



# 目次

第1章	序論	1
1.1	制御系に存在する拘束条件	1
1.2	従来研究	2
1.3	本論文の目的と構成	3
第2章	数学的準備	5
2.1	はじめに	5
2.2	凸多面体の射影	5
2.3	冗長な拘束式の消去法	8
第3章	拘束条件を有する制御系の解析	9
3.1	はじめに	9
3.2	問題設定	9
3.3	最大 CPI 集合	11
第4章	状態フィードバックに基づく拘束システムのスイッチング制御	13
4.1	はじめに	13
4.2	問題設定	14
4.3	最大 CPI 集合	15
4.4	補償器の切り換え	15
4.5	例題	18
4.6	問題点	23
第5章	Set-Valued オブザーバの構成	25
5.1	はじめに	25
5.2	Set-Valued オブザーバ	26
5.3	SVO の構成法 I - 外乱を考慮しない場合	27
5.3.1	問題設定	27
5.3.2	推定アルゴリズム	28
5.3.3	例題	31
5.4	SVO の構成法 II - 外乱を考慮する場合	33
5.4.1	問題設定	33
5.4.2	推定アルゴリズム	34
5.4.3	例題	38

第 6 章	出力フィードバックに基づく拘束システムのスイッチング制御	41
6.1	はじめに	41
6.2	問題設定	42
6.3	最大 CPI 集合	43
6.4	補償器の切り換え	43
6.5	例題	47
第 7 章	結論	49
7.1	本研究のまとめ	49
7.2	今後の課題	50
参考文献		51
付 録 A		53
A.1	定理 5.1 の証明	53
謝辞		59

# 第1章 序論

## 1.1 制御系に存在する拘束条件

実際の制御対象に対して補償器を設計する場合、与えられた安定性や制御性能に対する仕様の実現を、Fig. 1.1 の枠組で考える。

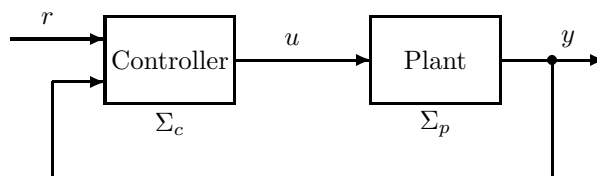


Fig. 1.1: Closed loop system

しかしながら、実際の制御系は、アクチュエータの性能限界による入力信号の制限、制御対象の保護のために要求される動作状態の制限など、多くの拘束条件が存在する‘拘束システム’である (Fig. 1.2). バルブの全開、全閉、モータの回転速度の限界、対象の強度等がその例である。

拘束条件が制御系に与える影響としては、補償器により要求される制御入力  $u_c$  と制御対象への実際の入力  $u$  とのギャップによる制御性能の劣化をはじめ、場合によっては系の不安定化を引き起こすことが知られている。このことは、制御系の解析や設計は、Fig. 1.1 で与えられる枠組で行うのではなく、Fig. 1.2 の枠組で行うべきであることを意味している。

工学の実状では、その対処法として、システム設計上での性能目標の軽減や、より強力なアクチュエータの使用というような方法で解決されることがしばしばある。しかしながら、これらは問題の解決法として本質的であるとは決して言えない。したがって、拘束条件の存在を陽に考慮した制御系設計法の確立が、実用的にも理論的にも重要な課題となっている。

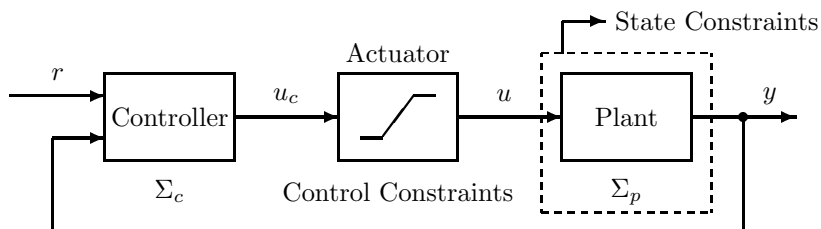


Fig. 1.2: Closed loop system with state and control constraints

## 1.2 従来研究

拘束システムに対する制御系設計を考える場合、まず考えられる方法は、拘束条件の達成を保証する補償器を設計するというアプローチである。しかしながら、実際補償器の設計においては、制御性能の向上、不確かさに対するロバスト性等、種々の設計仕様の達成を要求される。そのため、さらに拘束条件の達成を要求することは問題を複雑にし、実用的であるとはいえない。これに対し、拘束条件が存在しないものとして補償器を設計し、その後、拘束条件の達成を目的とした付加的な補償機構を構成するアプローチが考えられる。この場合、補償器の設計過程において拘束条件という厄介な制約を考慮することなく、種々の優れた線形制御理論により制御性能を追求できることから実用的であるといえ、これまでにこの観点から多くの研究がなされている。

アプローチの代表的なものとして、制御入力に対する制限が原因となり制御量が大きくオーバーシュートするウィンドアップ現象 [14] に対処するアンチウィンドアップ手法 [16, 17] がある。アンチウィンドアップ手法の基本的なアイデアは、入力制限を経た制御入力と補償器の出力が常に一致するように補償器のダイナミクスを修正する点にある。アンチウィンドアップ手法は補償器とは独立に構成され比較的構成が容易であるために、制御入力に制限のあるシステムの制御に広く用いられている。しかしながら、入力制限以外の拘束条件により引き起こされる問題には対処できない点が問題の一つとして挙げられる。

同様のアプローチとして、正の不変集合 (positively invariant set) と状態拘束集合 (state constraint set) の概念を応用した制御系設計法が数多く提案されている。正の不変集合は、制御系の状態変数を常にその領域内に留めるという性質により特徴づけられる。また状態拘束集合は、出力に対して課せられた拘束条件をみだす状態変数の全体からなる。この二つの集合の性質を兼ねあわせもつ最大の正の不変集合に制御系の初期状態が属している限り、制御系は拘束を破ることなく振る舞うことが保証される。その応用としてエラーガバナ (error governor) [2], リファレンスガバナ (reference governor) [3, 7] 等の付加的な補償機構が提案されている。エラーガバナは補償器の入力の整形を目的として、制御系に付加される機構である。リファレンスガバナは外部からの参照入力の整形を目的として制御系に付加される機構である。いずれも補償器の出力に影響を与える信号を整形することにより、制御系の状態変数を、常に最大の正の不変集合の内部に留め続ける。それにより拘束は達成される。この集合の性質に基づいた他の制御手法として、最近ではスイッチング制御によるアプローチが注目され大きな効果がえられている [6, 8, 9]。スイッチング制御は、複数の補償器を線形制御理論の枠組で構成し、これらを付加的な補償機構であるスーパーバイザにより切り換えながら適用する制御手法である。この制御則により、拘束システムを広い動作範囲で安定化できるとともに、制御性能の劣化を抑制することが可能となる。しかしながらスイッチング制御則を構成する個々の補償器を具体的にどのように設計するかについては、状態フィードバック制御など一部の場を除いて [6, 8], 一般には有効な指針が示されておらず課題となっている。またこのような状態空間に規定されたある部分集合に基づいた制御系設計法はいずれも状態変数の正確な情報が不可欠であり、この点も問題の一つである。

## 1.3 本論文の目的と構成

本論文では、拘束を有する系の制御系設計法としてスイッチング制御に注目する。スイッチング制御則は状態空間に規定されるある部分集合の性質に基づいているため、制御系の状態が観測可能であるという仮定が必要不可欠である。これに対し本論文では、オブザーバを用いて状態を推定し、その推定値に基づいて補償器を切り換えるスイッチング制御系の実現を目的とする。

本論文を構成する各章の概要は以下に示すとおりである。

2章では、本論文を構成するうえでの数学的準備を行う。まず本論文で用いる記号について述べる。つぎに凸多面体の射影および冗長な拘束式の消去法について述べる。本論文では状態空間のある部分集合を有限個の線形拘束式で表現することが多々ある。そのなかで特に、オブザーバの構成法および切り換えのアルゴリズムの導出の際に、凸多面体の射影は重要な役割を果たす。また拘束式に基づいて計算機上に凸多面体を表現する際、冗長な拘束式の存在は取り扱うデータが増加するため、オンライン処理を考えれば好ましくない。そのためそれらを消去する方法を考える。

3章では、制御系が拘束条件を破ることなく動作するための条件について述べる。まず、制御系の状態変数を常にその領域内に留めるという性質により特徴付けられる正の不変集合、出力に対して課せられた拘束条件をみだす状態変数全体からなる状態拘束集合、これら二つの性質を兼ね合わせた集合、CPI集合 (constrained positively invariant set) の構成について述べ、さらに任意のCPI集合を内部に含む最大CPI集合について述べる。制御系の初期状態が最大CPI集合に属していれば、制御系は拘束条件を破ることなく動作することが保証される。この最大CPI集合は、以降でスイッチング制御則を考えるうえで重要な基盤となる。

4章では、拘束を有する系の制御系設計法としてスイッチング制御によるアプローチについて述べる。その目的は、拘束条件の達成と制御性能の劣化の抑制という相反する問題を解決することである。切り換えられる補償器は安全性を重視し性能を犠牲にした補償器から、性能を重視し安全性を犠牲にした補償器まで複数個構成される。スーパーバイザは現在の制御対象の状態変数と対応する最大CPI集合の情報から、安全でかつ最も制御性能の良い補償器を選択する。それにより拘束条件は満足され、同時に制御性能の劣化も抑制される。ただし、状態変数が観測可能であるという仮定が必要である。

5章では、状態の推定機構として Set-Valued オブザーバの構成法について述べる。Set-Valued オブザーバは、対象のダイナミクスとえられる入出力データから、各時点で想定される状態変数すべてを集合として推定する点に特徴を有する。したがってダイナミクスのモデル化誤差を考えなければ、必ず状態の真値は推定された集合の内部に属するため、正確な状態変数の情報が要求される制御には有効であるといえる。ここでは外乱を考慮しない場合と考慮した場合の二種類のオブザーバを構成しシミュレーションによりその有効性を確認する。特に外乱を考慮しない場合は有限時間での真値への収束が保証される。

6章では、状態の推定機構として Set-Valued オブザーバを適用し、その推定値に基づいて補償器を切り換えるスイッチング制御則の実現を試みる。4章で扱った同じ例題に対しシミュレーションにより有効性を確認する。状態フィードバックに基づいたスイッチング制御則と異なる点は、スーパーバイザが切り換えるの指針として、Set-Valued オブザーバにより推定された制御対象の状態変数の集合と最大CPI集合の情報を用いる点である。指針として用いる状態変数が真値ではなく真値を含む集合として与えられることで、切り換えのタイミングの遅れる等の影響が考えられるが、状態変数を観測可能とした場合と同様に、拘束条件の達成と制御性能の劣



化の抑制が同時に保証される.

7章では, 本研究で得られた結果を要約して述べる.

記号:  $A^T$  は行列  $A$  の転置行列,  $A^{-1}$  は行列  $A$  の逆行列をあらわす.  $A \in \mathcal{R}^{m \times n}$  に対して,  $A_{(i,:)}$  は行列  $A$  の第  $i$  行ベクトルを,  $b \in \mathcal{R}^m$  に対して,  $b_{(j)}$  はベクトル  $b$  の  $j$  番目の要素をあらわす. また  $\mathcal{Z}^+$  は非負の整数全体をあらわす.  $0$  は要素が零の行列,  $I$  は単位行列をあらわす. また,  $\|x\|$  は  $x$  のユークリッド ノルムをあらわす.

## 第2章 数学的準備

### 2.1 はじめに

本章では、本論文を構成するうえで必要となる数学的準備を行う。本論文では状態空間のある部分集合を有限個の線形拘束式で表現することが多々ある。そのなかで特に、オブザーバの構成法およびスイッチング制御則の導出の際に、凸多面体の射影は重要な役割を果たす。また拘束式に基づいて計算機上に凸多面体を表現する際、冗長な拘束式の存在は取り扱うデータが増加するため、オンライン処理を考えれば好ましくない。そのためそれらを消去する方法を考える。

### 2.2 凸多面体の射影

問題 2.1.  $Set(M, m)$  は、 $\{x_p \in \mathcal{R}^n : Mx_p \leq m\}$ ,  $M \in \mathcal{R}^{l \times n}$ ,  $m \in \mathcal{R}^l$  で定義される凸多面体であるものとする。

以下で表現される集合  $S$  を考える。

$$S = \{x_p \in \mathcal{R}^n : M_1 x_p + M_2 w \leq m_1, \exists w \in \mathcal{R}\}, M_1 \in \mathcal{R}^{r \times n}, M_2 \in \mathcal{R}^r, m \in \mathcal{R}^r \quad (2.1)$$

このとき、(2.1) を  $Set(\bar{M}, \bar{m})$  と同様の形、すなわち

$$S = Set(\bar{M}, \bar{m}) \quad (= \{x_p \in \mathcal{R}^n : \bar{M}x_p \leq \bar{m}\}, \bar{M} \in \mathcal{R}^{\bar{l} \times n}, \bar{m} \in \mathcal{R}^{\bar{l}})$$

のように変形せよ。

この問題を考える場合、(2.1) の条件式から不要な変数  $w$  を消去する操作を行う必要がある。ここでは、Fourier-Motzkin の消去法 [4, 13, 15] に基づく解法を示す。

アルゴリズム 2.1.  $(\bar{M}, \bar{m})$  の導出アルゴリズム

Data:  $M_1, M_2, m_1$

Step1: まずつぎの集合を構成する。

$$I^0 = \{i : M_{2(i,:)} = 0\}, \quad I^+ = \{i : M_{2(i,:)} > 0\}, \quad I^- = \{i : M_{2(i,:)} < 0\}.$$

さらに各集合の元の個数 (cardinality) を、つぎのように定義する。

$$s^0 = \text{card}(I^0), \quad s^+ = \text{card}(I^+), \quad s^- = \text{card}(I^-).$$

Step2:  $C = [M_1 \quad -m_1] \in \mathcal{R}^{r \times (n+1)}$  とする。このとき、行列  $[\bar{M} \quad -\bar{m}] \in \mathcal{R}^{\bar{l} \times (n+1)}$ ,  $\bar{l} = s^0 + s^+ + s^-$  の各行はつぎで与えられる。

$$\{C_{(i,:)} : i \in I^0\}, \quad \{M_{2(i,:)}C_{(j,:)} - M_{2(j,:)}C_{(i,:)} : i \in I^+, j \in I^-\}$$

このアルゴリズムにより, 不要な変数  $w$  を消去することが可能となる. 以上の操作によりえられる行列の組  $(\bar{M}, \bar{m})$  の集合を, あらためてつぎのように定義する.

$$\text{Rack}[(M_1 \ M_2), m_1] = \{(\bar{M}, \bar{m}) \in \mathcal{R}^{\bar{l} \times n} \times \mathcal{R}^{\bar{l}} : S = \text{Set}(\bar{M}, \bar{m})\} \quad (2.2)$$

不要な変数  $w$  が  $k$  次のベクトルである場合は, 同様の操作  $\text{Rack}[\cdot]$  を  $k$  回繰り返すことにより消去が可能となる. すなわち,

$$\begin{aligned} \text{Rack}^2[(M_1 \ M_2), m_1] &= \text{Rack}[\text{Rack}[(M_1 \ M_2), m_1]] \\ &\dots \\ \text{Rack}^k[(M_1 \ M_2), m_1] &= \text{Rack}[\text{Rack}^{k-1}[(M_1 \ M_2), m_1]] \end{aligned}$$

と順に定義すれば,

$$S = \{x_p \in \mathcal{R}^n : M_1 x_p + M_2 w \leq m_1, \exists w \in \mathcal{R}^k\}$$

に対して,  $\text{Rack}^k[(M_1 \ M_2), m_1]$  を求めることにより,  $S$  を  $\text{Set}(\bar{M}, \bar{m})$  の形で表現可能になる. また  $\text{Rack}^k[(M_1 \ M_2), m_1]$  を求めることは, もともと  $n+k$  次元であった凸多面体を,  $n$  次元の凸多面体に射影する操作であると言い換えることができる.

注意 2.1. このアルゴリズムにより得られる  $(\bar{M}, \bar{m})$  で規定される線形拘束式のなかには,  $x_p$  の集合を表現するのに不必要な拘束式が存在する可能性がある. 冗長な拘束式は計算機上に凸多面体を実現する場合, 取り扱うデータ数が増加するため好ましくない. そのため, 冗長な拘束式を取り除く有効な方法を考える必要がある. この方法については次節で考察する.

例題 2.1. 5つの方程式により囲まれるつぎの凸多面体を考える (Fig. 2.1).  
図の凸多面体における  $x_p$  の集合は, つぎの式によって表現される.

$$S = \{x_p \in \mathcal{R}^n : M_1 x_p + M_2 w \leq m_1, \exists w \in \mathcal{R}\} \quad (2.3)$$

ただし, 各行列はつぎで与えられる.

$$M_1 = \begin{bmatrix} -2 & 1 & 1 & 5 & -4 \end{bmatrix}^T, \quad M_2 = \begin{bmatrix} 1 & 0 & 4 & -12 & -1 \end{bmatrix}^T, \quad m_1 = \begin{bmatrix} 6 & 8 & 28 & 60 & 0 \end{bmatrix}^T.$$

前述のアルゴリズムにしたがって, 式 (2.3) から不要な変数  $w$  を消去し,  $\text{Set}(M, m) = \{x_p \in \mathcal{R}, Mx_p \leq m\}$  の形に変形する.

Step1:  $I^0 = \{2\}$ ,  $I^+ = \{1, 3\}$ ,  $I^- = \{4, 5\}$ .  $s^0 = 1$ ,  $s^+ = 2$ ,  $s^- = 2$ .

$$\text{Step2: } C = \begin{bmatrix} M_1 & -m_1 \end{bmatrix} = \begin{bmatrix} -2 & -6 \\ 1 & 8 \\ 1 & -7 \\ 5 & -60 \\ -4 & 0 \end{bmatrix}.$$

$\begin{bmatrix} M & -m \end{bmatrix}$  の各行の要素は,

$$C_{(2,:)} = \begin{bmatrix} 1 & -8 \end{bmatrix},$$

$$M_{2(1,:)}C_{(4,:)} - M_{2(4,:)}C_{(1,:)} = \begin{bmatrix} -19 & -132 \end{bmatrix}, \quad M_{2(1,:)}C_{(5,:)} - M_{2(5,:)}C_{(1,:)} = \begin{bmatrix} -6 & -6 \end{bmatrix},$$

$M_{2(3,:)}C_{(4,:)} - M_{2(4,:)}C_{(3,:)} = \begin{bmatrix} 32 & -576 \end{bmatrix}$ ,  $M_{2(3,:)}C_{(5,:)} - M_{2(5,:)}C_{(3,:)} = \begin{bmatrix} -15 & -28 \end{bmatrix}$  より,

$$\begin{bmatrix} M & -m \end{bmatrix} = \begin{bmatrix} 1 & -8 \\ -19 & -132 \\ -6 & -6 \\ 32 & -576 \\ -15 & -28 \end{bmatrix} \quad (2.4)$$

となる.

以上より,  $Set(M, m) = \{x_p \in \mathcal{R}, Mx_p \leq m\}$  のように, 変数  $x_p$  のみに関する拘束式に変形できる. これを解くと,  $x_p$  の範囲は  $-1 \leq x_p \leq 8$  となり, もともとの凸多面体を  $x_p$  軸に射影した場合の  $x_p$  の範囲と一致する (Fig. 2.1). ただし, 式 (2.4) において  $x_p$  を表現するのに必要であるのは, 1, 3 行目の要素のみであり, それ以外の 3 行は冗長な拘束式である.

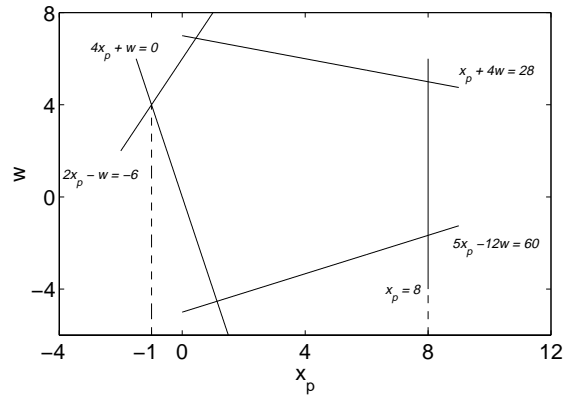


Fig. 2.1: Projections of convex sets

## 2.3 冗長な拘束式の消去法

$M \in \mathcal{R}^{l \times n}$ ,  $m \in \mathcal{R}^l$  に対して,  $S = \text{Set}(M, m) = \{x_p \in \mathcal{R}^n : Mx_p \leq m\}$  を考える.  $(M, m)$  で規定される線形拘束式の中から, ある拘束式を取り除いても実行可能領域 (与えられた全ての拘束式を満足する領域)  $S$  が変化しない場合, その拘束式は冗長である. 冗長な拘束式は計算機上に凸多面体を表現する場合, 取り扱うデータ数が増加するため好ましくない. 特に前節で述べた  $\text{Rack}[\cdot]$  演算を要する場合, もともと  $l$  本の拘束式が最悪ケースで  $(\frac{l}{2})^2$  本に増加してしまう. したがって冗長な拘束式を取り除く方法を考える.

仮に,  $M_{(i,:)}x_p \leq m_{(i)}$  が冗長な拘束式であるとする. このとき全体の拘束式から  $M_{(i,:)}x_p \leq m_{(i)}$  を取り除いてできる拘束式の組を  $\hat{M}x_p \leq \hat{m}$  とすると,  $S = \text{Set}(\hat{M}, \hat{m})$  が成り立つ.

この考えに基づいて冗長な拘束式を消去するアルゴリズムを考える.

### アルゴリズム 2.2. 冗長な拘束式の消去アルゴリズム

**Data:**  $M \in \mathcal{R}^{l \times n}$ ,  $m \in \mathcal{R}^l$

**Step1:**  $i = 1$

**Step2:**  $i > l$  ならば終了.

$Mx_p \leq m$  から  $M_{(i,:)}x_p \leq m_{(i)}$  を取り除いてできる拘束式の組を  $\hat{M}x_p \leq \hat{m}$  とする.

**Step3:** 線形計画問題

$$h = \text{maximize } (M_{(i,:)}x_p), \quad \text{subject to } \hat{M}x_p \leq \hat{m}$$

を解く. このとき,  $h \leq m_{(i)}$  ならば  $M_{(i,:)}x_p \leq m_{(i)}$  は冗長な拘束式である.  $i$  行目を取り除き,  $i + 1$  行目以下を繰り上げ新たな  $M, m$  を構成する.  $l = l - 1$  として Step2 へ戻る.  $h > m_{(i)}$  ならば  $M_{(i,:)}x_p \leq m_{(i)}$  は必要な拘束式である.  $i = i + 1$  として Step2 に戻る.

このアルゴリズムにより冗長な拘束式を消去することが可能となる. 以降, 与えられた  $(M, m)$  に対しこの操作を適用することを,  $\text{Elim}(M, m)$  とあらわすものとする.

# 第3章 拘束条件を有する制御系の解析

## 3.1 はじめに

実際の制御系では、アクチュエータの性能の限界や制御対象の保護の観点から、飽和制限等、制御入力や制御対象の状態に対して様々な拘束条件が存在する。拘束条件が破られることは、システムの挙動に悪影響を与え、制御性能の劣化を引き起こすだけでなく、場合によっては系が不安定化されることを意味する。このように、制御系に存在する拘束条件は、制御系を実際に稼働させるうえで大きな問題となっている。

拘束条件を有する制御系の解析法に状態拘束集合と正の不変集合の概念を用いるものがある。状態拘束集合は、出力に関して課せられた拘束条件をみたす状態変数全体からなる。また正の不変集合は、状態変数を常にその領域内に留めるという性質により特徴づけられる。これら二つの性質を兼ね合わせた、状態拘束集合に含まれる最大の正の不変集合に関して様々な文献で議論がなされている [2, 3, 5, 6, 7, 8]。制御系の初期状態がこの最大の正の不変集合に属しているかぎり、制御系は拘束を破ることなく振る舞うことが保証される。

本章では、外部入力のない線形離散時間システムが拘束を破ることなく振る舞うことを保証する条件について述べる。

## 3.2 問題設定

拘束を有するつぎの線形離散時間システムを考える (Fig. 3.1)。

$$\Sigma \quad \begin{aligned} x(k+1) &= Ax(k) \\ z_0(k) &= C_0x(k) \end{aligned} \quad (3.1)$$

ここで  $x \in \mathcal{R}^n$  は状態、 $z_0 \in \mathcal{R}^p$  は拘束条件を考えるために設けられた変数であり、 $z_0(k) \in Z \subset \mathcal{R}^p$  の拘束を有するものとする。集合  $Z$  としては、つぎのような凸多面体を考える。

$$Z = \{z_0 \in \mathcal{R}^p : Fz_0 \leq f\}, \quad F \in \mathcal{R}^{r \times p}, \quad f \in \mathcal{R}^r. \quad (3.2)$$

$A, C_0$  は適当な大きさの定数行列である。

仮定 3.1. 閉ループ系  $\Sigma$  は漸近安定 ( $A$  の固有値が単位円内部)、 $(C_0, A)$  は可観測であるとする。また、 $z_0(k)$  が十分小さい場合には拘束が破られないという観点から、 $0 \in \text{int}Z$  を仮定する。さらに  $Z$  は有界であるとする。

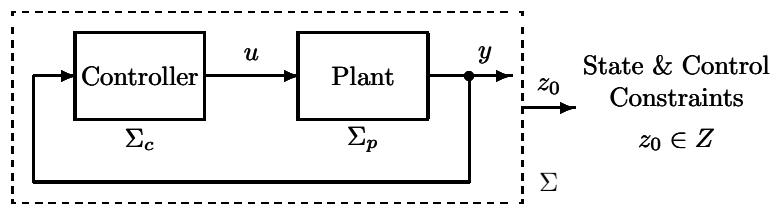


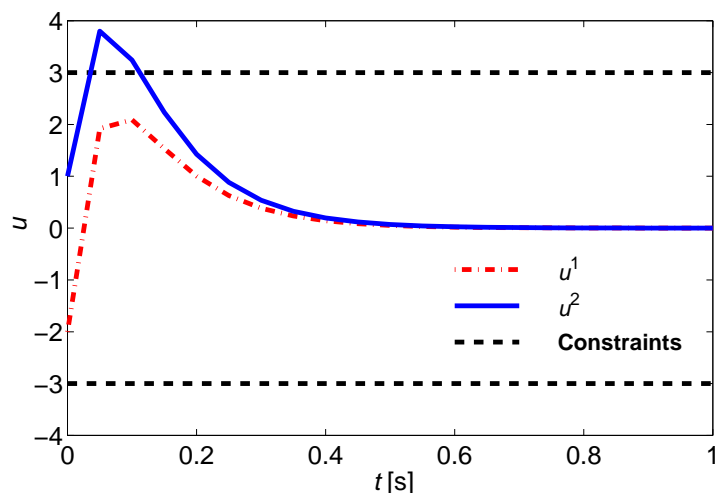
Fig. 3.1: Constrained system

例題 3.1. つぎの状態フィードバック系と2つの初期状態  $x_0^1, x_0^2$  を考える.

$$x(k+1) = \begin{bmatrix} -0.6 & 0.1 \\ 0.3 & 0.4 \end{bmatrix} x(k) + \begin{bmatrix} -1.0 \\ 0.5 \end{bmatrix} u(k), \quad x(k) = \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

$$u(k) = - \begin{bmatrix} 1.0 & -0.2 \end{bmatrix} x(k), \quad x_0^1 = \begin{bmatrix} 7 \\ 25 \end{bmatrix}, \quad x_0^2 = \begin{bmatrix} 5 \\ 30 \end{bmatrix},$$

ただし制御入力には,  $|u| \leq 3$  の拘束があるものとする. すなわち, (3.2)において,  $M = \begin{bmatrix} -1 & 1 \end{bmatrix}^T$ ,  $m = \begin{bmatrix} 3 & 3 \end{bmatrix}^T$  とすることにより表される. 各初期状態に対する制御入力の時間変化を, Fig. 3.2 に示した. 初期状態  $x_0^1$  に対する応答  $u^1$  は, 制御入力に対する拘束を破ることなく振る舞うが,  $x_0^2$  に対する応答  $u^2$  は, 拘束を破ってしまうことがわかる. この例から拘束条件がみたされるためには, 初期状態に対する条件を考える必要があるといえる.

Fig. 3.2: Control input  $u(k)$

### 3.3 最大 CPI 集合

本節では, 制御系が与えられた拘束条件を常に満足しながら動作する条件を明らかにし, これが最大 CPI 集合と呼ばれる状態空間上に構成される部分集合により達成されることを示す.

まず拘束条件を考える観点からつぎの集合を定義する.

定義 3.1. ( 状態拘束集合 )

状態拘束集合をつぎのように定義する.

$$X_{C_0Z} = \{ x \in \mathcal{R}^n \mid C_0x \in Z \}. \quad (3.3)$$

状態拘束集合は出力に対して課せられた拘束条件をみたす状態変数の全体からなる集合である. 制御系が拘束を破ることなく動作するためには, その状態変数を状態拘束集合の内部に留めておくことが必要となる.

そこでつぎに正の不変集合の概念を導入する. 集合  $S \subset \mathcal{R}$  は原点を内点にもつとする. このとき任意の初期状態  $x(0) \in S$  に対する応答が常に  $S$  の内部に留まり続けるならば,  $S$  は正の不変集合と呼ばれる.

正の不変集合の中で, 特に  $S \subseteq X_{C_0Z}$  が成立するとき,  $S$  は CPI(constrained positively invariant) 集合と呼ばれる. このとき正の不変集合の性質から, 任意の初期状態  $x(0) \in S$  に対して常に  $x(k) \in S$ , すなわち  $x(k) \in X_{C_0Z}$  が達成され制御系は拘束を破ることなく振る舞う.

一般に, CPI 集合は複数存在する. CPI 集合が, ‘制御系が拘束を破ることなく振る舞う状態変数の領域である’ という点から, なるべく大きな領域を明らかにすることが望ましい. そこで任意の CPI 集合を内部に含む最大 CPI 集合をつぎのように定義する.

定義 3.2. ( 最大 CPI 集合 ) [5]

$x_0 = x(0)$  とする.  $\Sigma$  に対する最大 CPI 集合  $O_\infty$  をつぎのように定義する.

$$O_\infty = \{ x_0 \in \mathcal{R}^n \mid z_0(k) = C_0A^kx_0 \in Z, \forall k \in \mathcal{Z}^+ \}. \quad (3.4)$$

注意 3.1. 最大 CPI 集合は,  $\Sigma$  に対するすべての CPI 集合を内部に含む最大の CPI 集合である. よって任意の初期状態  $x(0) \in O_\infty$  に対して閉ループ系  $\Sigma$  の応答は常に  $x(k) \in O_\infty$  かつ  $z_0(k) \in Z$  をみたす. 言い換えれば,  $\Sigma$  が常に拘束条件をみたすための必要十分条件は,  $x(0) \in O_\infty$  が成立することである.

注意 3.2. 最大 CPI 集合  $O_\infty$  は有界な凸多面体であり, 線形計画法を利用した数値計算により構成され, つぎのような形で表現される.

$$O_\infty = \{ x_0 \in \mathcal{R}^n : M_0x_0 \leq m_0 \}, \quad M_0 \in \mathcal{R}^{g \times n}, \quad m_0 \in \mathcal{R}^g. \quad (3.5)$$

ここで  $M_0, m_0$  は, 凸多面体  $O_\infty$  を規定する線形拘束式を表現する行列であり,  $g$  は線形拘束式の数である. なおここでの不等式は, ベクトルの各要素ごとに成立しているものとする. 最大 CPI 集合の性質, および構成法の詳細については, 文献 [2, 5] 等で議論されている.



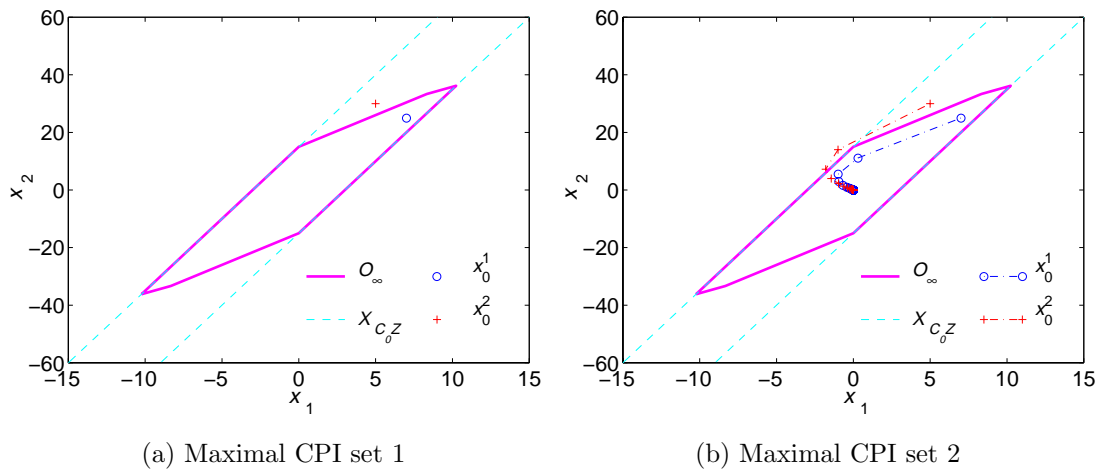


Fig. 3.3: Maximal CPI set

例題 3.2. 例題 3.1 で考えた状態フィードバック系に対し, 最大 CPI 集合を構成し, 再度閉ループ系の振る舞いについて検証する.

最大 CPI 集合  $O_\infty$  は, Fig. 3.3(a) 上で実線で囲まれる凸多面体となる. 破線で囲まれる領域は状態拘束集合  $X_{C_0Z}$  を表す. また, ‘o’, ‘+’ は各々, 初期状態  $x_0^1 = [7 \ 25]^T$ ,  $x_0^2 = [5 \ 30]^T$  を表す.  $x_0^2$  を初期状態とする応答が拘束をみたさなかったのは,  $x_0^2$  が  $X_{C_0Z}$  には属しているが,  $O_\infty$  には属さない状態であったためである.  $X_{C_0Z}$  は, 最大 CPI 集合  $O_\infty$  の重要な性質のひとつである, 動特性に関する不変性, すなわち

$$x(k) \in O_\infty \implies x(k+1) \in O_\infty \quad (3.6)$$

を有していないため, 拘束条件の達成を常に保証することはできない. 実際に,  $x_0^1, x_0^2$  から始まるトラジェクトリ -  $x_k^1, x_k^2$  を描くと, Fig. 3.3(b) に示すように,  $x_0^2$  の場合, 拘束条件を満足する安全な領域から外れてしまうのが確認できる.

制御系が拘束を破ることなく動作するための必要十分条件は, その初期状態が最大 CPI 集合に属していることである. 次章では, この最大 CPI 集合の概念を基盤として提案されている制御系設計法について述べる.

# 第4章 状態フィードバックに基づく拘束システムのスイッチング制御

## 4.1 はじめに

実際の制御系には、アクチュエータの飽和要素や制御対象の保護のための制限など多くの拘束条件が存在する。拘束条件が破られることは、システムの挙動に悪影響を与え制御性能の劣化を引き起こす。したがって補償器の設計においては、拘束条件を満足すると同時に、要求される制御性能を達成するという困難な問題を解決しなければならない。

このような問題に対する制御手法として、近年、スイッチング制御によるアプローチが注目されている [6, 8, 9]。スイッチング制御は、線形制御理論の枠組で複数の補償器を構成し、これらを切り換えることにより拘束条件による影響の軽減を目指すものである (Fig. 4.1)。ここでの特徴として、個々の補償器を構成する際に、その補償器を適用することが許される状態空間の領域を同時に構成することが挙げられる。スイッチング制御則は、この状態空間の領域に基づいた補償器の切り換えにより、拘束条件を有する制御系を広い状態空間の領域で安定化し、また制御性能の劣化を抑制する。

本章では、文献 [6] で提案されたスイッチング制御則の構成法をもとに、状態フィードバックに基づくスイッチング制御の概要と問題点について考察する。この制御則の特徴は切り換えのアルゴリズムが状態空間上の最大 CPI 集合の性質に基づいている点にある。

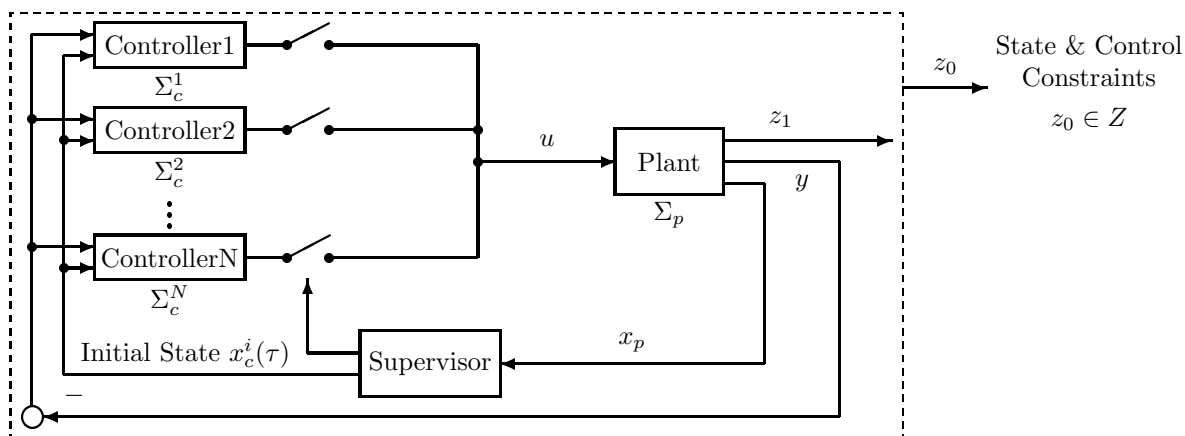


Fig. 4.1: Switching control

## 4.2 問題設定

本章では, 制御対象  $\Sigma_p$  に対して,  $N$  個の補償器  $\Sigma_c^1, \Sigma_c^2, \dots, \Sigma_c^N$  を構成し, それらのスイッチングに基づいたレギュレータ問題を考える (Fig. 4.1).

つぎの制御対象  $\Sigma_p$  を考える.

$$\begin{aligned} \Sigma_p \quad x_p(k+1) &= A_p x_p(k) + B_p u(k) \\ z_1(k) &= C_{p1} x_p(k) \\ y(k) &= C_{p2} x_p(k) \end{aligned} \quad (4.1)$$

ここで  $x_p \in \mathcal{R}^{n_p}$  は制御対象の状態,  $u \in \mathcal{R}^m$  は制御入力,  $z_1 \in \mathcal{R}$  は被制御量である. また  $y \in \mathcal{R}^{n_y}$  は観測出力であり,  $A_p, B_p, C_{p1}, C_{p2}$  は適当な大きさの定数行列である.

制御対象  $\Sigma_p$  に適用される補償器として, ここではつぎの線形フィードバック補償器  $\Sigma_c^i$  を考える.

$$\begin{aligned} \Sigma_c^i \quad x_c^i(k+1) &= A_c^i x_c^i(k) + B_c^i y(k) \\ u(k) &= C_c^i x_c^i(k) + D_c^i y(k), \quad i = 1, \dots, N \end{aligned} \quad (4.2)$$

ここで  $x_c^i \in \mathcal{R}^{n_{ci}}$  は補償器の状態,  $A_c^i, B_c^i, C_c^i, D_c^i$  は適当な大きさの定数行列である. 各補償器の次元  $n_{ci}$  は必ずしも等しくある必要はない. また静的な補償器に関しては, 4.2 において, 次数  $n_{ci} = 0$ ,  $u(k) = D_c^i y(k)$  と考えれば適用可能である.

注意 4.1. 各補償器はすでに構成されており, 特にその系列  $\Sigma_c^i$ ,  $i = 1, \dots, N$  は順に制御性能が向上するように与えられているとする. したがって,  $\Sigma_c^N$  が最も望ましい制御性能を有する. スwitching制御則を構成する個々の補償器の設計に関しては, 状態フィードバック制御など一部の場 [6, 8] を除いて, 一般には有効な指針が示されておらず今後の研究課題となっている.

$\Sigma_p, \Sigma_c^i$  により構成される閉ループ系  $\Sigma^i$  はつぎのようになる.

$$\begin{aligned} \Sigma^i \quad x^i(k+1) &= A^i x^i(k) \\ z_1(k) &= C x^i(k) \end{aligned} \quad (4.3)$$

ただしここで  $x = \begin{bmatrix} x_p^T & (x_c^i)^T \end{bmatrix}^T \in \mathcal{R}^{n^i}$ ,  $n^i := n_p + n_{ci}$  であり, 各行列はつぎで与えられる.

$$A^i = \begin{bmatrix} A_p + B_p D_c^i C_{p2} & B_p C_c^i \\ B_c^i C_{p2} & A_c^i \end{bmatrix} \in \mathcal{R}^{n \times n} \quad C = \begin{bmatrix} C_{p1} & \mathbf{0} \end{bmatrix} \in \mathcal{R}^{n_y \times n}$$

つぎに閉ループ系  $\Sigma^i$  に対する拘束条件を表現するために以下で与えられる変数  $z_0(k)$  を設ける.

$$\begin{aligned} z_0(k) &= C_{0p} x_p(k) + D_0 u(k) \\ &= C_0^i x^i(k) \in Z \subset \mathcal{R}^p \end{aligned} \quad (4.4)$$

集合  $Z$  としては, つぎのような凸多面体を考える.

$$Z = \{z_0 \in \mathcal{R}^p : F z_0 \leq f\}, \quad F \in \mathcal{R}^{r \times p}, \quad f \in \mathcal{R}^r. \quad (4.5)$$

仮定 4.1. 閉ループ系  $\Sigma^i$  は漸近安定 ( $A^i$  の固有値が単位円内部),  $(C_0^i, A^i)$  は可観測であるとする. また,  $z_0(k)$  が十分小さい場合には拘束が破られないという観点から,  $0 \in \text{int} Z$  を仮定する. さらに  $Z$  は有界であるとする.

### 4.3 最大 CPI 集合

前章でふれたように、制御系が拘束を破ることなく動作するための必要十分条件は、その初期状態が最大 CPI 集合に属していることである。

まず (4.3) の各閉ループ系  $\Sigma^i$  に対して最大 CPI 集合をつぎのように定義する。

定義 4.1. ( 最大 CPI 集合 )

$x_0^i = x^i(0)$  とする.  $\Sigma^i$  に対する最大 CPI 集合  $O_\infty^i$  をつぎのように定義する.

$$O_\infty^i = \{ x_0^i \in \mathcal{R}^{n^i} \mid z_0(k) = C_0^i (A^i)^k x_0^i \in Z, \forall k \in \mathcal{Z}^+ \}, \quad i = 1, \dots, N. \quad (4.6)$$

注意 4.2.  $O_\infty^i \neq \emptyset$ ,  $i = 1, \dots, N$  とする.  $O_\infty^i$  は  $\Sigma_c^i$  に対するすべての CPI 集合を内部に含む最大の CPI 集合である. すなわち  $O_\infty^i$  は、各閉ループ系  $\Sigma^i$  が安全に動作できる最も大きな領域を与える. 任意の初期状態  $x^i(0) \in O_\infty^i$  に対して閉ループ系  $\Sigma^i$  の応答は常に  $x^i(k) \in O_\infty^i$ , かつ  $z_0(k) \in Z$  をみたす. 言い換えれば、 $\Sigma^i$  が常に拘束をみたすための必要十分条件は、 $x^i(0) \in O_\infty^i$  が成立することである.

注意 4.3. 最大 CPI 集合  $O_\infty^i$  は有界な凸多面体であり、線形計画法を利用した数値計算により構成され、つぎのような形で表現される [6, 8].

$$O_\infty^i = \{ x_0^i \in \mathcal{R}^{n^i} : M_0^i x_0^i \leq m_0^i \}, \quad M_0^i \in \mathcal{R}^{g^i \times n^i}, \quad m_0^i \in \mathcal{R}^{g^i}. \quad (4.7)$$

ここで  $M_0^i$ ,  $m_0^i$  は、凸多面体  $O_\infty^i$  を規定する線形拘束式を表現する行列であり、 $g^i$  は線形拘束式の数である. なおここでの不等式は、ベクトルの各要素ごとに成立しているものとする.

本稿で述べるスイッチング制御は、この  $O_\infty^i$  を指針として補償器の切り換えを達成するものである. 次節では補償器の切り換えについて考える.

### 4.4 補償器の切り換え

まず補償器の切り換えについて考えるうえで重要な、つぎの仮定をおく.

仮定 4.2. 制御対象  $\Sigma_p$  の状態  $x_p(k)$  は観測可能である.

補償器を切り換えるスーパーバイザは、最大 CPI 集合  $O_\infty^i$  が与える情報と制御対象の状態  $x_p(k)$  に基づき、適用可能な補償器のうち最も制御性能の優れているものを各時刻で選択する. したがって、スーパーバイザの具体的な役割は、

- 使用する補償器の選定
- 新たに使用する補償器の初期状態の決定

をオンラインで実行することである. これらについて考える.

## (補償器の選定)

補償器の選定は各時点における制御対象の状態  $x_p(k)$  の値に依存する. 基本概念は, 現在の状態  $x_p(k)$  に対しどの補償器を使用すれば, 安全でかつ最も制御性能の良い閉ループ系を構成できるかである.

これまで述べたように, 安全な制御の判断基準は最大 CPI 集合にある. 各  $O_\infty^i$  は制御対象と補償器の次数を合わせた  $n_p + n_{ci}$  次元の空間で定義されている. そこでまずこれを以下のように制御対象の状態空間へ射影する.

$$O_{\infty p}^i = \{ x_p \in \mathcal{R}^{n_p} \mid \exists x_c^i \in \mathcal{R}^{n_{ci}}, \begin{bmatrix} x_p \\ x_c^i \end{bmatrix} \in O_\infty^i \}, \quad i = 1, \dots, N \quad (4.8)$$

各  $O_\infty^i$  が凸多面体であるため, これを射影した  $O_{\infty p}^i$  もまた凸多面体である.  $O_{\infty p}^i$  の具体的な表現は, 2章で述べた Fourier-Motzkin の消去法に基づいて与えることが可能である. この  $O_{\infty p}^i$  を用いると, 補償器  $\Sigma_c^i$  が適用可能となる条件が,  $x_p(k) \in O_{\infty p}^i$  と表現される.

つぎに,

$$O_{\infty p} = \bigcup_{i=1}^N O_{\infty p}^i$$

とする.  $O_{\infty p}$  は, 適用可能となる補償器が少なくとも一つは存在する制御対象  $\Sigma_p$  の状態すべてから成っている.  $x_p \in O_{\infty p}$  に対して,  $I(x_p) = \{ i \mid x_p \in O_{\infty p}^i \}$  を定義すれば, この集合は状態  $x_p$  に対して適用可能な補償器のインデックスをすべて集めたものとなる. 補償器は順に制御性能が向上するように構成されているため,  $I(x_p(k))$  のなかから最も大きな値が選択されることが望ましい. したがってスーパーバイザは, 以下の方針で使用する補償器をオンラインで選択する.

$$i(k) = \max \{ i \mid x_p(k) \in O_{\infty p}^i \}. \quad (4.9)$$

スーパーバイザは  $i(\tau) \neq i(\tau - 1)$  が成立した時刻  $\tau$  のみで補償器を切り換える. 切り換えが実行される際には, 新たに適用される補償器  $\Sigma_c^i$  の初期状態  $x_c^{i(\tau)}(\tau)$  を同時に決定することが必要となる.

## (補償器の初期状態の決定)

新たに適用される補償器  $\Sigma_c^i$  の初期状態  $x_c^{i(\tau)}(\tau)$  の決定において,  $x_c^{i(\tau)}(\tau)$  がみたさなければならぬ最低限の条件は

$$\begin{bmatrix} x_p(\tau) \\ x_c^{i(\tau)}(\tau) \end{bmatrix} \in O_\infty^i \quad (4.10)$$

である.  $O_\infty^i$  の定義 (4.8) より, このような  $x_c^{i(\tau)}(\tau)$  は必ず存在し, またこの条件をみたすならば全体の閉ループ系は拘束条件を破ることなく動作する. 本稿での制御目的は初期状態  $x_p(0)$  の影響を速やかに減衰させることであるため,  $\|x_c^i\|$  を最小化することを考え, 以下の二次計画問題を解き, 条件をみたす  $x_c^i$  を初期状態として与える.

$$\text{minimize } (x_c^i)^T x_c^i \quad \text{subject to } M_2^i x_c^i \leq m_0^i - M_1^i x_p(\tau) \quad (4.11)$$

ここでの線形拘束式は (4.7) で,

$$M_0^i = [ M_1^i \ M_2^i ] \in \mathcal{R}^{g^i \times n^i}, \quad M_1^i \in \mathcal{R}^{g^i \times n_p}, \quad M_2^i \in \mathcal{R}^{g^i \times n_{ci}}.$$

と定義したものである. ここでえられる  $x_c^i$  は条件 (4.10) をみたすものである.

注意 4.4. 一般に, (4.10) をみたす初期状態は複数存在し, またこの選択が制御入力の高きさに直接関係することを考えれば, 全体の制御性能に大きな影響を与えることは明らかである. 残念ながら現在のところ, 全体の制御性能までを考慮した初期状態の有力な決定法は提案されておらず, 今後の考察が必要な課題となっている [9].

注意 4.5. 新たに適用される補償器が静的なもの ( $n_{ci} = 0$ ) である場合は初期化の必要は生じない.

以上から補償器の切り換えに関してつぎの定理がえられる.

定理 4.1. 仮定 4.1,  $x_p(0) \in O_{\infty p}$  が成立しているとする. このとき, 制御対象  $\Sigma_p$ , 補償器  $\Sigma_c^i$ ,  $i = 1, \dots, N$  および (4.9), (4.10) のスイッチング制御則により構成される閉ループ系は漸近安定である. またこのとき, 拘束条件  $z_0(k) \in Z$  はすべての時刻において達成される.

(証明) 拘束条件が存在しない場合, 仮定 4.1 より各閉ループ系  $\Sigma^i$  は漸近安定である. また最大 CPI 集合の性質から,  $x_p(k) \in O_{\infty p}^i \rightarrow x_p(k+1) \in O_{\infty p}^i$  が成立する. 以上より, いま時刻  $t$  において,  $x_p(t) \in O_{\infty p}^i$  であったとすると,  $x_p(\tau) \in O_{\infty p}^{i+1}$  なる  $\tau > t$  が必ず存在し,  $\Sigma_c^i \rightarrow \Sigma_c^{i+1}$  の補償器の切り換えが生じる. したがって, スwitching 制御則 (4.9), (4.10) により, 任意の初期状態  $x_p(0) \in O_{\infty p}$  に対する応答は収束し, またこのとき, 適用される補償器  $\Sigma_c^i$  のインデックスの値  $i(k)$  は単調に増加するのみである. さらに, 拘束条件  $z_0(k) \in Z$  はすべての時刻において達成される.

## 4.5 例題

### 例題 4.1. 2次の制御対象

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -0.01 & -0.04 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0.01 \end{bmatrix} u(t), \quad x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

$$z_1(t) = y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t).$$

を考える. ただし制御入力には,  $|u| \leq 2$  の拘束があるものとする.  
使用する補償器はつぎの4種類である.

$$\begin{aligned} \Sigma_c^1 & \quad u(t) = -y(t) \\ \Sigma_c^2 & \quad \dot{x}_c^2(t) = -0.51962 x_c^2(t) + 1.0392 y(t) \\ & \quad u(t) = 5 x_c^2(t) - 15 y(t) \\ \Sigma_c^3 & \quad \dot{x}_c^3(t) = -1.7156 x_c^3(t) + 6.1743 y(t) \\ & \quad u(t) = 15 x_c^3(t) - 68.9837 y(t) \\ \Sigma_c^4 & \quad \dot{x}_c^4(t) = -2.1445 x_c^4(t) + 7.7179 y(t) \\ & \quad u(t) = 45 x_c^4(t) - 206.951 y(t) \end{aligned}$$

各補償器は順に性能が向上するよう構成されている. 各補償器を用いた制御系の周波数応答お

よび制御対象の初期状態  $x_p(0)^T = [1.5 \ 0]^T$  に対する初期値応答を各々 Fig. 4.2, Fig. 4.3 に示す. Fig. 4.3 より,  $\Sigma_c^1$  以外の補償器を用いる場合収束は速くなるが, 制御入力は入力制限を破っているのが確認できる. その度合は性能を追求すればするほど大きい.

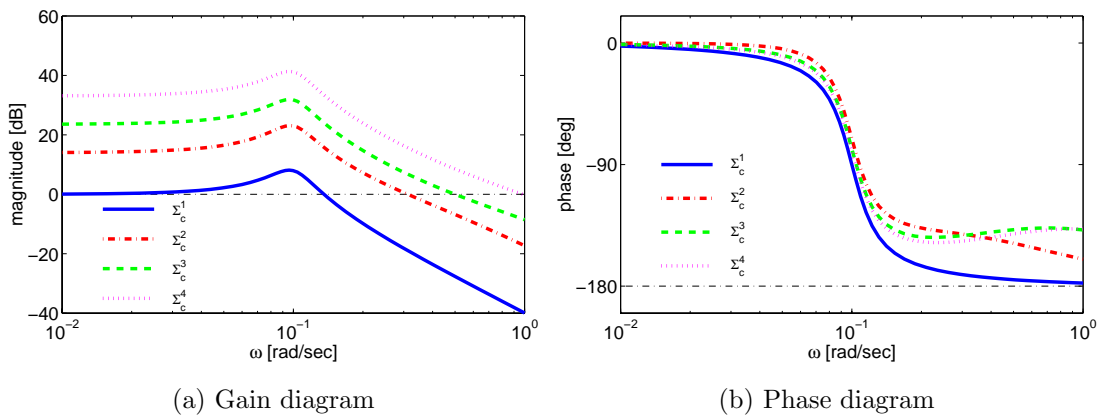
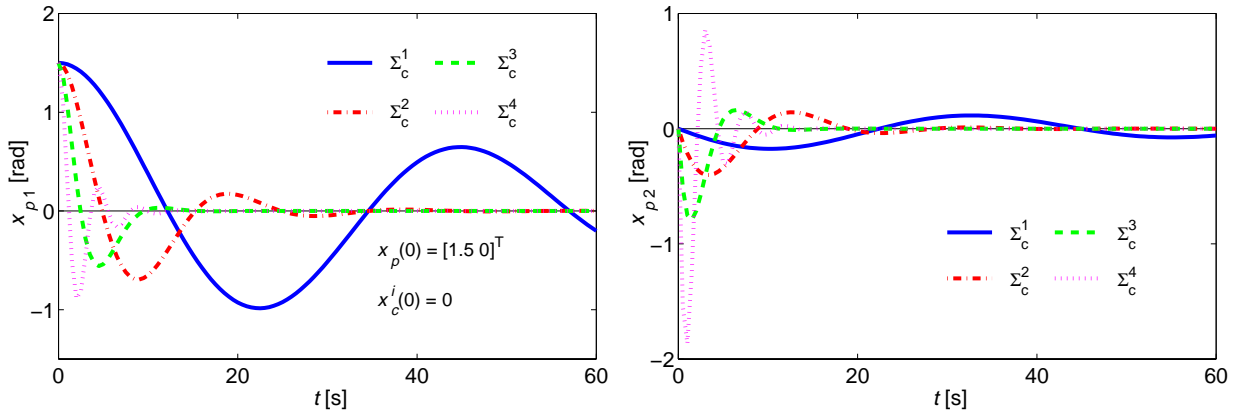
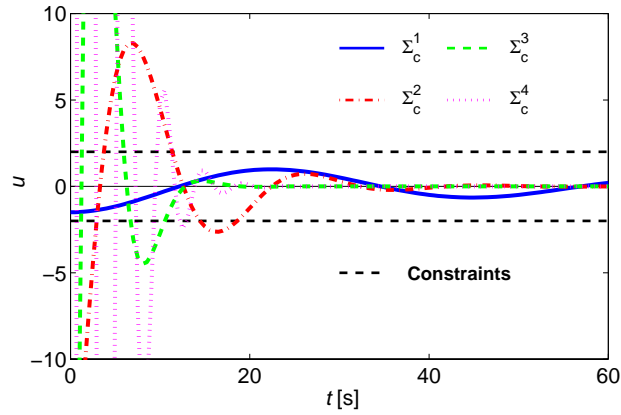


Fig. 4.2: Bode diagram

(a) State variable  $x_{p1}(=z_1)$ (b) State variable  $x_{p2}$ (c) Control input  $u$ Fig. 4.3: Initial condition response  $\Sigma^i$ ,  $i = 1, \dots, 4$ 

サンプリング時間  $T = 0.5[s]$  の零次ホールドで得られる離散時間系に対し、スイッチング制御系を構成する。まず離散化により得られる制御対象および補償器の状態空間表現はつぎのようになる。

$$\Sigma_p \quad x_p(k+1) = \begin{bmatrix} 0.99876 & 0.49483 \\ -0.00495 & 0.97897 \end{bmatrix} x_p(k) + \begin{bmatrix} 0.00124 \\ 0.00495 \end{bmatrix} u(k)$$

$$z_1(k) = y(k) = \begin{bmatrix} 1 & 0 \end{bmatrix} x_p(k), \quad x_p(k) = \begin{bmatrix} x_{p1}(k) \\ x_{p2}(k) \end{bmatrix}$$

$$\Sigma_c^1 \quad u(k) = -y(k)$$

$$\Sigma_c^2 \quad \begin{aligned} x_c^2(k+1) &= 0.7712 x_c^2(k) + 0.4576 y(k) \\ u(k) &= 5 x_c^2(k) - 15 y(k) \end{aligned}$$

$$\Sigma_c^3 \quad \begin{aligned} x_c^3(k+1) &= 0.42409 x_c^3(k) + 2.07264 y(k) \\ u(k) &= 15 x_c^3(k) - 68.9837 y(k) \end{aligned}$$

$$\Sigma_c^4 \quad \begin{aligned} x_c^4(k+1) &= 0.34224 x_c^4(k) + 2.36723 y(k) \\ u(k) &= 45 x_c^4(k) - 206.951 y(k) \end{aligned}$$



各補償器で構成される閉ループ系  $\Sigma^i$  に対する最大CPI集合  $O_\infty^i$  を, 制御対象の状態空間へ射影した  $O_{\infty p}^i$  を Fig. 4.4 に示す. 制御性能の向上とともに拘束条件を満足する安全な領域が狭くなっている. したがって安定化を保証する領域の拡大と制御性能の向上はトレードオフの関係にあることが確認できる.

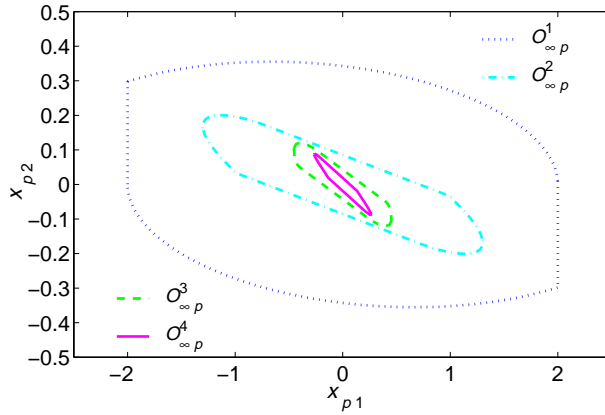


Fig. 4.4:  $O_{\infty p}^i$ ,  $i = 1, \dots, 4$

初期状態  $x_p(0)^T = [1.5 \ 0]^T$  に対する状態変数の軌跡を Fig. 4.5 に, 状態変数の変位および制御入力の時応答を Fig. 4.6, Fig. 4.7 に示す. 各図ともにスイッチング制御によるものと, 補償器  $\Sigma_c^1$  のみを用いた場合を比較している. Fig. 4.7 は Fig. 4.6 の  $0 \sim 40$  [s] 間を拡大したもので切り換えのタイミングも表示してある. スイッチング制御則により応答が改善され, かつ拘束条件も満足されているのが確認できる. スイッチング制御則は, 入力制限を破らない範囲で大きな制御入力を要求している.

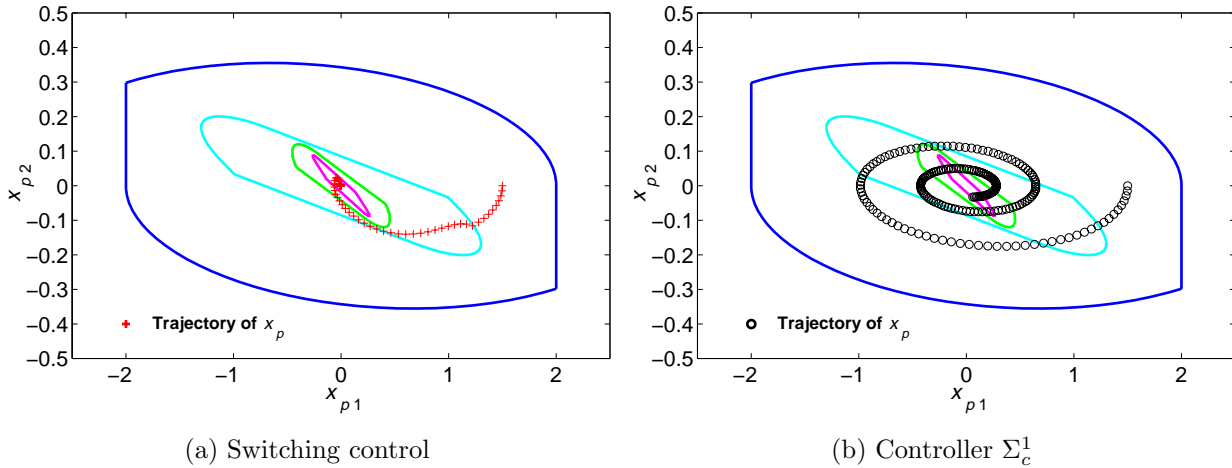


Fig. 4.5: State trajectories

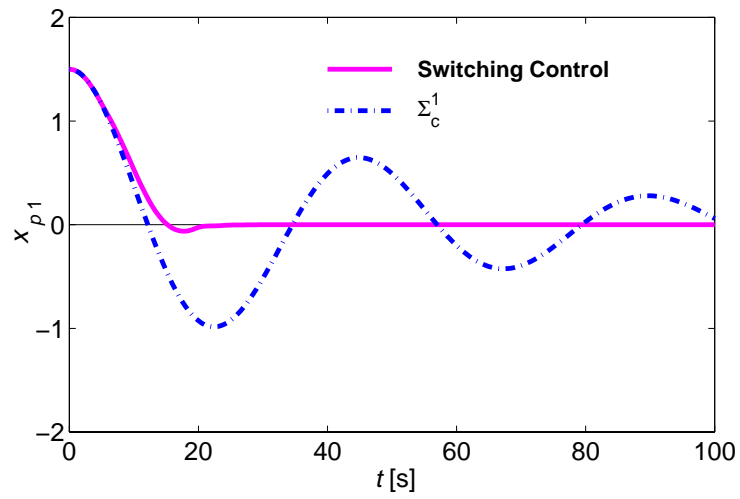
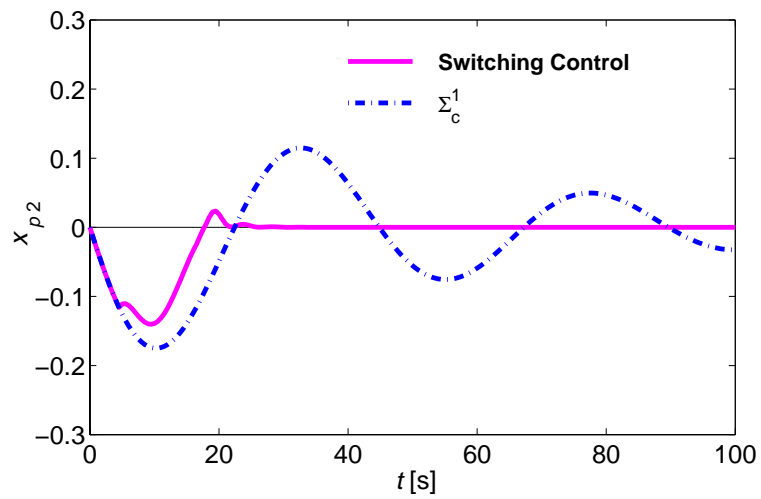
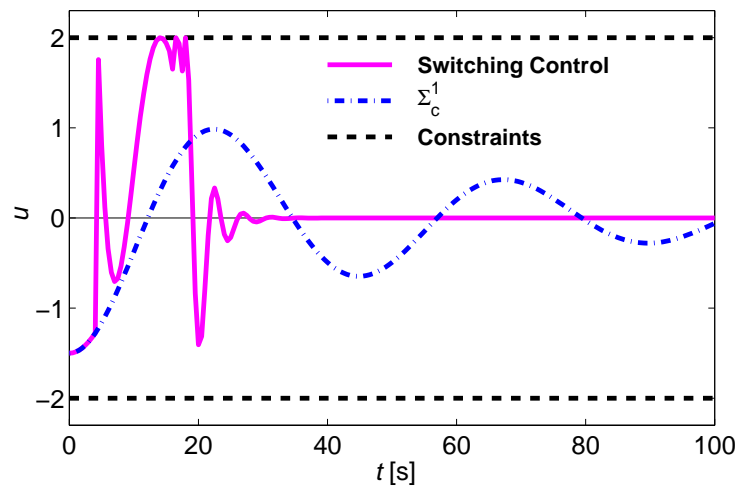
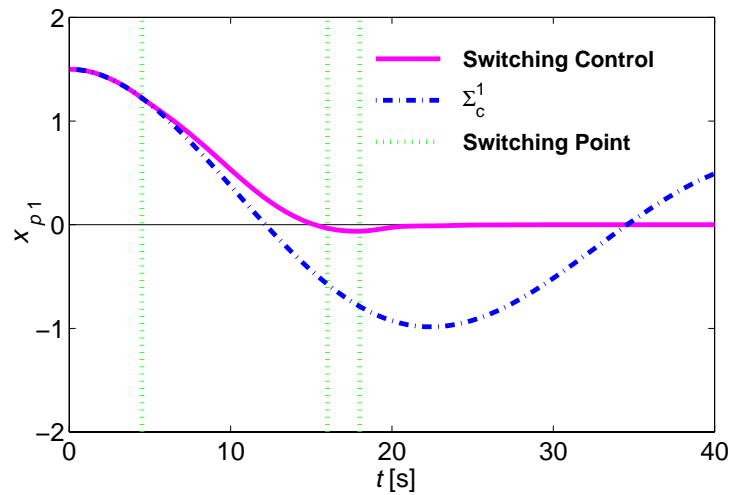
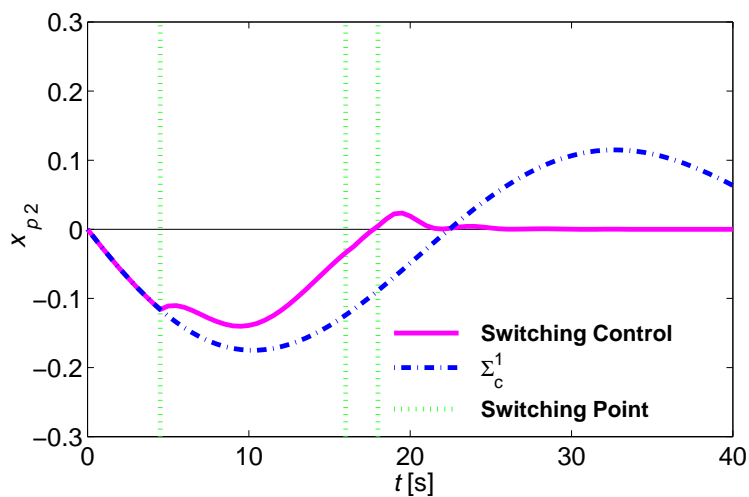
(a) State variable  $x_{p1}(=z_1)$ (b) State variable  $x_{p2}$ (c) Control input  $u$ 

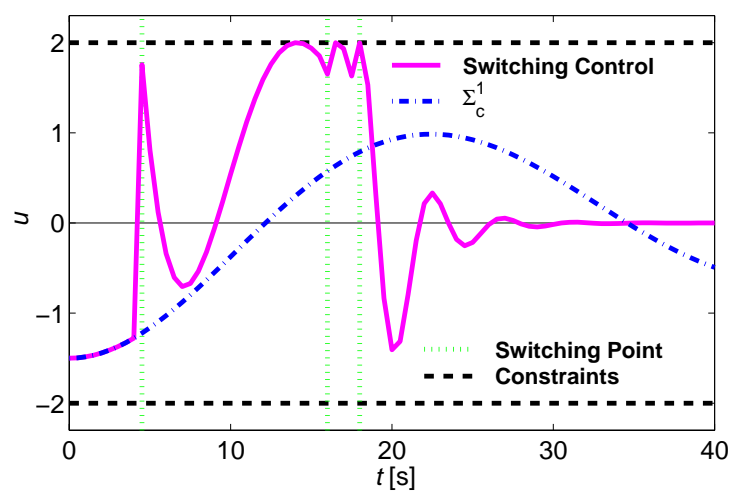
Fig. 4.6: Initial condition response (Switching control)



(a) State variable  $x_{p1}(=z_1)$



(b) State variable  $x_{p2}$



(c) Control input  $u$

Fig. 4.7: Blowup of Fig.4.6 ( $0 \leq k \leq 40$ )

## 4.6 問題点

本章で述べたスイッチング制御のアルゴリズムは、適用可能な補償器のインデックスをすべて集めた集合である  $I(x_p) = \{ i \mid x_p \in O_{\infty p}^i \}$  に基づくものであり、制御対象の全状態変数を正確に観測可能であるという仮定が必要不可欠である (仮定 4.2)。適用する補償器を決める、すなわち制御入力  $u$  を決めるためのフィードバック情報として状態変数を適用できるという仮定は、情報量という点では完璧なフィードバック制御の実現が期待できる。本論文で述べるスイッチング制御を含め、正の不変集合の概念を基盤としたこれまでに数多く提案されている制御系設計法に関してもこの仮定の重要度は大きい。しかし実システムを考えた場合、すべての状態変数を測定することが困難な場合や経済的な制約から、これを測定することが好ましくない場合も多く、一部の状態変数が観測出力として測定できる場合がほとんどである。したがってこの仮定が一つの弱点でもある。このことから以降ではこの仮定を必要としない、本論文の主題である出力フィードバックに基づくスイッチング制御則の実現を目指す。

状態変数の観測に関する問題に対しては、制御対象の入出力から状態変数を推定するオブザーバ (状態観測器) を用いることにより近似的に状態フィードバックを実現する方法が一般的である。センサをモデルを用いた推定計算に置き換えることができる点は強みでもあるが、スイッチング制御等を考える場合は正確な状態変数の情報が必要とされるため、単に推定値の真値への収束を保証するオブザーバでは、推定値と真値の関係が明確に与えられないため正確な制御が保証されない。それは同時に拘束条件の達成が保証できないことを意味する。

そこで、次章ではスイッチング制御系への適用を考慮したうえでのオブザーバの設計について考える。



# 第5章 Set-Valued オブザーバの構成

## 5.1 はじめに

拘束条件を有する系に対する制御系設計法としては、前章で述べたスイッチング制御や、その他にはリファレンスガバナ等によるアプローチがこれまでに提案されている [2, 3, 6, 7, 9]. これらの特徴は、状態空間上に構成されたある部分集合の性質を基盤として、補償器の切り換えや参照入力の整形をおこなう点にある。このため、状態変数が観測できるという仮定が必要不可欠となる。しかし、実際のシステムを考えた場合、その仮定が成り立たない場合が多く、一部の状態変数が観測出力として測定できる場合がほとんどである。

状態変数の観測に関する問題に対しては、制御対象の入出力から状態変数を推定するオブザーバ (状態観測器) を用いることにより近似的に状態フィードバックを実現する方法が一般的である。しかしながら、実際スイッチング制御のように状態空間上のある領域に基づいた制御則を考える場合、単に推定値の真値への収束を保証するオブザーバでは、推定値と真値の関係が明確に与えられないため正確な制御が保証されない。それは同時に拘束条件の達成が保証されないことを意味する。

Shamma ら [10, 11, 12] は状態推定機構として、Set-Valued オブザーバ (SVO) の構成法を示している。その特徴は、システムのダイナミクスに基づき、各時点で想定される状態変数すべてを集合として推定する点にある。推定される集合は、線形拘束式の組で規定される凸多面体として与えられる。したがって状態変数の真値は、推定された集合の内部に属することが保証されるため、正確な状態変数の情報が要求される制御には有効であるといえる。

本章ではまず、SVO の意義について考える。そのうえで SVO の構成法を述べる。本章で述べる SVO は、以降のスイッチング制御系への適用を考慮し、Fig. 5.1 に示すように、制御対象の入出力およびダイナミクスに基づき制御対象の状態変数のみを推定するものとして議論を進める。また制御系に対する外乱を考慮しない場合と考慮する場合の二種類についてその構成法を示す。さらに、構成した SVO の有効性をシミュレーションにより検証し、考察をおこなう。

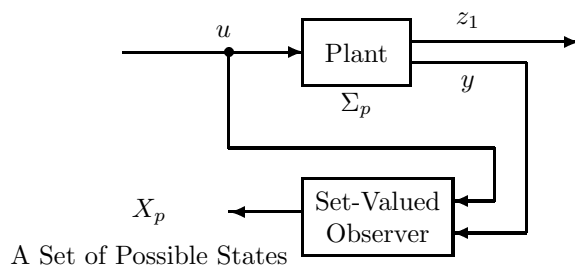


Fig. 5.1: Set-Valued observer

## 5.2 Set-Valued オブザーバ

具体例として, 前章で述べた最大CPI集合に基づいたスイッチング制御を用いて Set-Valued オブザーバ (SVO) の意義を考える.

まず, ある制御対象  $\Sigma_p$  に対して, 2つの補償器  $\Sigma_c^1, \Sigma_c^2$  が順に性能が向上するよう構成されている制御系を考える. このとき各閉ループ系  $\Sigma^1, \Sigma^2$  に対する最大CPI集合  $O_\infty^1, O_\infty^2$  を制御対象の状態空間上に射影したものが, Fig. 5.2の凸多面体  $O_{\infty p}^1, O_{\infty p}^2$  で与えられていると仮定する. すなわち制御対象の状態変数  $x_p$  が  $O_{\infty p}^1$  のみに属している間は補償器  $\Sigma_c^1$  を使用し,  $x_p$  が  $O_{\infty p}^2$  に属した瞬間に補償器を  $\Sigma_c^2$  に切り換えることにより, 拘束条件が保証され安全な, かつ補償器  $\Sigma_c^1$  のみを使用するよりも向上した性能の制御が達成される. このような制御の達成には, 制御対象の状態変数  $x_p$  の正確な情報が不可欠である.  $x_p$  が  $O_{\infty p}^2$  に属していない場合に補償器を  $\Sigma_c^2$  に切り換えれば, 拘束が破られるおそれがあるためである. 状態変数の推定値  $\hat{x}_p$  の真値  $x_p$  への収束は保証するが, 真値と推定値との関係は明確には与えないオブザーバを用いる場合, Fig. 5.2(a)のような状況が予想され, 切り換えのタイミングがずれ拘束が破られる危険性がある. これに対し, 制御対象の入出力およびそのダイナミクスから, 想定される状態変数全体の集合  $X_p$  を推定する SVO を適用することを考える. この場合は, ダイナミクスのモデル化誤差を考慮しなければ集合  $X_p$  がかならず真値  $x_p$  を含む. したがって Fig. 5.2(b)のように, 集合  $X_p$  が  $O_{\infty p}^2$  に完全に属するまで補償器を切り換えなければ, 前述のような危険性は生じない.

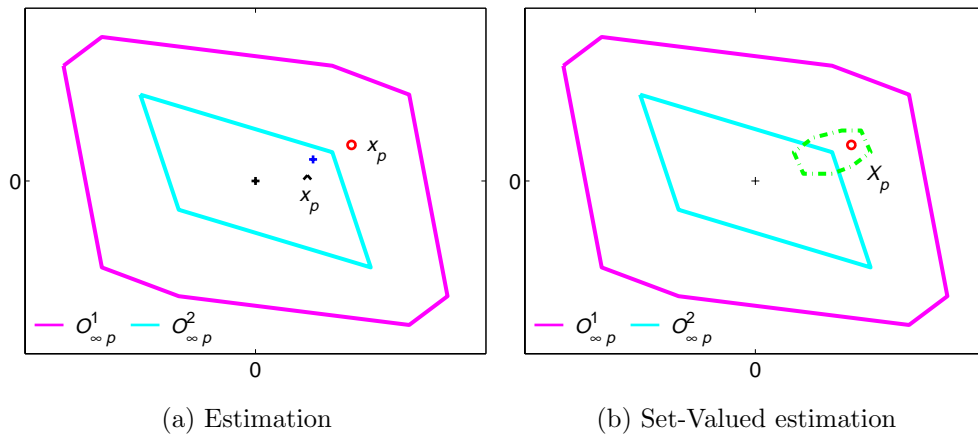


Fig. 5.2: Concept of Set-Valued estimation

## 5.3 SVO の構成法 I - 外乱を考慮しない場合

### 5.3.1 問題設定

本節では、外乱を考慮しないものとして制御対象の状態を推定する SVO の構成法について述べる (Fig. 5.1).

つぎの制御対象  $\Sigma_p$  を考える.

$$\begin{aligned} x_p(k+1) &= A_p x_p(k) + B_p u(k) \\ \Sigma_p \quad z_1(k) &= C_{p1} x_p(k) \\ y(k) &= C_{p2} x_p(k) \end{aligned} \quad (5.1)$$

ここで  $x_p \in \mathcal{R}^{n_p}$  は制御対象の状態,  $u \in \mathcal{R}^m$  は制御入力,  $z_1 \in \mathcal{R}$  は被制御量である. また  $y \in \mathcal{R}^{n_y}$  は観測出力であり,  $A_p, B_p, C_{p1}, C_{p2}$  は適当な大きさの定数行列である.

オブザーバを構成するうえで、つぎの仮定をおく.

仮定 5.1.  $B_{\mathcal{R}^{n_p}}$  を  $\mathcal{R}^{n_p}$  全体あるいはつぎで定義される必ずしも有界とは限らない凸多面体とする.

$$B_{\mathcal{R}^{n_p}} = \{x_p \in \mathcal{R}^{n_p} : S_1 x_p \leq S_2\}, \quad S_1 \in \mathcal{R}^{f \times n_p}, \quad S_2 \in \mathcal{R}^f \quad (5.2)$$

このとき制御対象の初期状態は  $x_p(0) \in B_{\mathcal{R}^{n_p}}$  であるとする.

SVO は、ある時点  $k$  で考えられうるすべての状態を推定するオブザーバである. そこでその状態全体の集合を、 $X_p(k)$  と定義する. するとこのとき  $X_p(k)$  はつぎの二種類の集合の共通集合としてあらわすことができる.

まず、時点  $k-1$  までの情報に基づいて、考えられうる状態  $x_p$  全体の集合をつぎのようにならわす.

$$X_{p\text{pre}}(k) = \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(k-1) + B_p u(k-1), \exists x_p(k-1) \in X_p(k-1)\} \quad (5.3)$$

つぎに、時点  $k$  の観測出力  $y(k)$  の情報に基づいて、考えられうる状態  $x_p$  全体の集合をつぎのようにならわす.

$$\tilde{X}_p(y(k)) = \{x_p \in \mathcal{R}^{n_p} : y(k) = C_{p2} x_p\}. \quad (5.4)$$

(5.3), (5.4) の条件を同時にみたす  $x_p$  の集合が時点  $k$  で考えられうるすべての状態の集合  $X_p(k)$  である.

以上をふまえ、初期状態から順に状態の存在可能な範囲を考えていく.



## 5.3.2 推定アルゴリズム

- $k = 0$ ,  $(x_p(0))$  の推定

$x_p(0)$  の範囲は (5.2) と  $y(0)$  から, つぎであらわされる.

$$\begin{aligned} X_p(0) &= B_{\mathcal{R}^{n_p}} \cap \{x_p \in \mathcal{R}^{n_p} : y(0) = C_{p2}x_p\} \\ &= B_{\mathcal{R}^{n_p}} \cap \tilde{X}_p(y(0)) \end{aligned} \quad (5.5)$$

ここで,  $\tilde{X}_p(y(0))$  は,

$$\tilde{X}_p(y(0)) = \text{Set}(\tilde{M}_c, \tilde{m}(0)), \quad \tilde{M}_c = \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \quad \tilde{m}(0) = \begin{bmatrix} -y(0) \\ y(0) \end{bmatrix}.$$

と変形できるので, 結果的に  $X_p(0)$  は,

$$X_p(0) = \text{Set}(M(0), m(0)), \quad M(0) = \begin{bmatrix} S_1 \\ \tilde{M}_c \end{bmatrix}, \quad m(0) = \begin{bmatrix} S_2 \\ \tilde{m}(0) \end{bmatrix}.$$

の形であらわすことができる. これにより,  $x_p(0)$  の存在範囲が規定される.

- $k = 1$ ,  $(x_p(1))$  の推定

$x_p(1)$  の範囲はつぎであらわされる.

$$\begin{aligned} X_p(1) &= \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(0) + B_p u(0), \exists x_p(0) \in X_p(0)\} \\ &\quad \cap \{x_p \in \mathcal{R}^{n_p} : y(1) = C_{p2}x_p\} \\ &= X_{p\text{pre}}(1) \cap \tilde{X}_p(y(1)) \end{aligned} \quad (5.6)$$

ここで,  $X_{p\text{pre}}(1)$  は,  $x_p(0) \in X_p(0)$  を満足する  $x_p(0)$  と,  $u(0)$  の値により考えられうる状態全体の集合を意味する.

$A_p$  が正則であれば,

$$X_{p\text{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : A_p^{-1}x_p - A_p^{-1}B_p u(0) = x_p(0) \in X_p(0)\}$$

と変形できるので, 結果的に,

$$X_{p\text{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : M(0)A_p^{-1}x_p \leq m(0) + M(0)A_p^{-1}B_p u(0)\}$$

とあらわせる.

$\tilde{X}_p(y(1))$  に関しては,

$$\tilde{X}_p(y(1)) = \text{Set}(\tilde{M}_c, \tilde{m}(1)), \quad \tilde{M}_c = \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \quad \tilde{m}(1) = \begin{bmatrix} -y(1) \\ y(1) \end{bmatrix}.$$

と変形できるので, 結果的に  $X_p(1)$  は,

$$X_p(1) = \text{Set}(M(1), m(1)), \quad M(1) = \begin{bmatrix} M(0)A_p^{-1} \\ \tilde{M}_c \end{bmatrix}, \quad m(1) = \begin{bmatrix} m(0) + M(0)A_p^{-1}B_p u(0) \\ \tilde{m}(1) \end{bmatrix}.$$

の形であらわすことができる.

$A_p$  が非正則の場合は,

$$X_{p_{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : x_p - A_p x_p(0) = B_p u(0), \exists x_p(0) \in X_p(0)\}$$

より, 結果的に  $X_p(1)$  は,

$$X_p(1) = \{x_p \in \mathcal{R}^{n_p} : M_1(1)x_p + M_2(1)x_p(0) \leq m_1(1)\}$$

$$M_1(1) = \begin{bmatrix} I \\ -I \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(1) = \begin{bmatrix} -A_p \\ A_p \\ M(0) \\ \mathbf{0} \end{bmatrix}, \quad m_1(1) = \begin{bmatrix} B_p u(0) \\ -B_p u(0) \\ m(0) \\ \tilde{m}(1) \end{bmatrix}.$$

となる. この場合は, 変数  $x_p(0)$  を消去するために,  $Rack[\cdot]$  演算を  $n_p$  回おこなえば,  $X_p(1) = Set(M(1), m(1))$  の形に変形できる.

以上より,  $x_p(1)$  の存在範囲が規定される.

同様の操作を繰り返せば, 時点  $t$  における状態の範囲はつぎのように与えることができる.

- $k = t$ , ( $x_p(t)$  の推定)

$x_p(t)$  の範囲はつぎであらわされる.

$$\begin{aligned} X_p(t) &= \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(t-1) + B_p u(t-1), \exists x_p(t-1) \in X_p(t-1)\} \\ &\cap \{x_p \in \mathcal{R}^{n_p} : y(t) = C_{p2} x_p\} \\ &= X_{p_{pre}}(t) \cap \tilde{X}_p(y(t)) \end{aligned} \quad (5.7)$$

$\tilde{X}_p(y(t))$  に関しては,

$$\tilde{X}_p(y(t)) = Set(\tilde{M}_c, \tilde{m}(t)), \quad \tilde{M}_c = \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \quad \tilde{m}(t) = \begin{bmatrix} -y(t) \\ y(t) \end{bmatrix}.$$

と変形できるので, 結果的に  $X_p(t)$  は,

$A_p$  が正則であれば,

$$X_p(t) = Set(M(t), m(t)),$$

$$M(t) = \begin{bmatrix} M(t-1)A_p^{-1} \\ \tilde{M}_c \end{bmatrix}, \quad m(t) = \begin{bmatrix} m(t-1) + M(t-1)A_p^{-1}B_p u(t-1) \\ \tilde{m}(t) \end{bmatrix}.$$

の形であらわすことができる.

$A_p$  が非正則の場合は,

$$X_p(t) = \{x_p \in \mathcal{R}^{n_p} : M_1(t)x_p + M_2(t)x_p(t-1) \leq m_1(t)\}$$

$$M_1(t) = \begin{bmatrix} I \\ -I \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(t) = \begin{bmatrix} -A_p \\ A_p \\ M(t-1) \\ \mathbf{0} \end{bmatrix}, \quad m_1(t) = \begin{bmatrix} B_p u(t-1) \\ -B_p u(t-1) \\ m(t-1) \\ \tilde{m}(t) \end{bmatrix}.$$

となる. この場合は, 変数  $x_p(t-1)$  を消去するために,  $Rack[\cdot]$  演算を  $n_p$  回おこない,  $Rack^{n_p}[(M_1(t) \ M_2(t)), m_1(t)]$  を求めれば,  $X_p(t) = Set(M(t), m(t))$  の形に変形できる.

以上より,  $x_p(t)$  の存在範囲が規定される.

したがって推定則はつぎのようにまとめられる.

### アルゴリズム 5.1. $X_p(k)$ の推定アルゴリズム

$k = 0$ :

$$\begin{aligned} X_p(0) &= B_{\mathcal{R}^{n_p}} \cap \tilde{X}_p(y(0)) \\ &= Set(M(0), m(0)) \\ (M(0), m(0)) &= \left[ \begin{pmatrix} S_1 \\ \tilde{M}_c \end{pmatrix}, \begin{pmatrix} S_2 \\ \tilde{m}(0) \end{pmatrix} \right], \tilde{M}_c = \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \tilde{m}(0) = \begin{bmatrix} -y(0) \\ y(0) \end{bmatrix}. \end{aligned}$$

$k = t > 0$ :

$$\begin{aligned} X_p(t) &= X_{p_{pre}}(t) \cap \tilde{X}_p(y(t)) \\ &= Set(M(t), m(t)) \end{aligned}$$

$\langle A_p$  が正則  $\rangle$

$$\begin{aligned} (M(t), m(t)) &= \left[ \begin{pmatrix} M(t-1)A_p^{-1} \\ \tilde{M}_c \end{pmatrix}, \begin{pmatrix} m(t-1) + M(t-1)A_p^{-1}B_p u(t-1) \\ \tilde{m}(t) \end{pmatrix} \right], \\ \tilde{M}_c &= \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \tilde{m}(t) = \begin{bmatrix} -y(t) \\ y(t) \end{bmatrix}. \end{aligned}$$

$\langle A_p$  が非正則  $\rangle$

$$\begin{aligned} (M(t), m(t)) &\in Rack^{n_p} \left[ \begin{pmatrix} I & -A_p \\ -I & A_p \\ \mathbf{0} & M(t-1) \\ \tilde{M}_c & \mathbf{0} \end{pmatrix}, \begin{pmatrix} B_p u(t-1) \\ -B_p u(t-1) \\ m(t-1) \\ \tilde{m}(t) \end{pmatrix} \right], \\ \tilde{M}_c &= \begin{bmatrix} -C_{p2} \\ C_{p2} \end{bmatrix}, \tilde{m}(t) = \begin{bmatrix} -y(t) \\ y(t) \end{bmatrix}. \end{aligned}$$

注意 5.1. SVO の構成アルゴリズムは, 対象の安定性, 可観測性については全く言及しない. ただし, 可観測性の仮定は, 制御系を構成するうえで有効なつぎの定理を導く.

定理 5.1. (5.1) であらわされる制御対象に対して,  $(C_{p2}, A_p)$  が可観測であるとする. このとき,  $n_p - 1$  時点での SVO による推定値は, 一点からなる集合  $X_p(n_p - 1) = \{\hat{x}_p(n_p - 1)\}$  に収束し, かつ  $\hat{x}_p(n_p - 1) = x_p(n_p - 1)$  が成立する. (証明は付録参照)

注意 5.2. 上の定理より,  $n_p - 1$  時点以降で SVO が与える推定値は, 状態の真値  $x_p(\tau)$ ,  $\tau \geq n_p - 1$  に常に一致する.

### 5.3.3 例題

例題 5.1. [DC モータ, ギア, シャフトで結合された負荷からなる位置サーボ機構] [1]  
 制御対象 (Fig. 5.3) はつぎの状態空間モデルで与えられる.

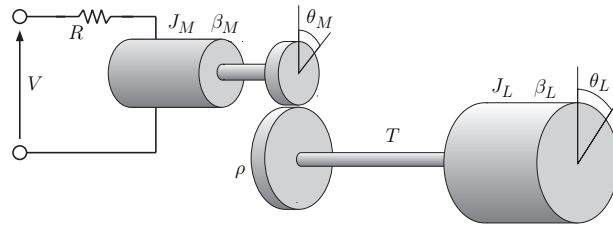


Fig. 5.3: Servomechanism model

$$\frac{d}{dt} \begin{bmatrix} \theta_L \\ \dot{\theta}_L \\ \theta_M \\ \dot{\theta}_M \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_\theta}{J_L} & -\frac{\beta_L}{J_L} & \frac{k_\theta}{\rho J_L} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{k_\theta}{\rho J_M} & 0 & -\frac{k_\theta}{\rho^2 J_M} & -\frac{\beta_M + K_T^2/R}{J_M} \end{bmatrix} \begin{bmatrix} \theta_L \\ \dot{\theta}_L \\ \theta_M \\ \dot{\theta}_M \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{K_T}{R J_M} \end{bmatrix} V$$

$$z_1(t) = y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_L \\ \dot{\theta}_L \\ \theta_M \\ \dot{\theta}_M \end{bmatrix}$$

サンプリング時間  $T = 0.1[s]$  の零次ホールドによって離散系に変換したモデルを用いる。  
 また, 入力を  $r - \theta_L$ , 出力を  $V$  とする補償器はつぎで与えられるものとする.

$$\Sigma^c \frac{1000(9.7929z^3 - 2.1860z^2 - 7.2663z + 2.5556)}{10z^4 - 2.7282z^3 - 3.5585z^2 - 1.3029z - 0.0853}$$

$\Sigma^p$ ,  $\Sigma^c$  からなる閉ループ系に, Set-Valued オブザーバを構成し制御対象の状態,

$$x_p = \begin{bmatrix} x_{p1} & x_{p2} & x_{p3} & x_{p4} \end{bmatrix}^T = \begin{bmatrix} \theta_L & \dot{\theta}_L & \theta_M & \dot{\theta}_M \end{bmatrix}^T$$

を推定した結果を Fig. 5.4 に示す.

Fig. 5.4 において,  $x_{pi}$  は状態  $x_p$  の各要素の真値をあらわし, Max, Min は状態  $x_p$  の各要素の推定値  $\hat{x}_{pi}$  の最大値, 最小値をあらわす. また Reference は制御系に加える参照入力 (負荷の角度目標値  $\pi/6$  [rad]) をあらわす. 推定の初期条件は  $|x_{pi}| \leq 1$  とした.

シミュレーションより, 状態  $x_p$  の各要素に対する推定は,  $x_{p1}$  を除いてすべて時点  $k = 3$  ( $0.3[s]$ ) で真値に収束することが確かめられた.  $x_{p1}$  に関しては, 出力  $y$  として観測できるために, 時点  $k = 0$  ( $[0.0[s]$ ) で一致することが分かる.

また, 各時点における線形拘束式の数を Table 5.1 に示す. Elim[.] により冗長な拘束式を消去した場合と消去しない場合とを比較すると, 前者では高々15本であるのに対し, 後者は  $10+2k$  ( $k$  は時点) の割合で増加していくのが確認できる.

Table 5.1: Number of linear constraints in  $X_p$ 

k	0	1	2	3	4	5	6	7	8	9	10
Elim(M,m)	8	8	7	8	10	11	13	15	13	12	13
(M,m)	10	12	14	16	18	20	22	24	26	28	30

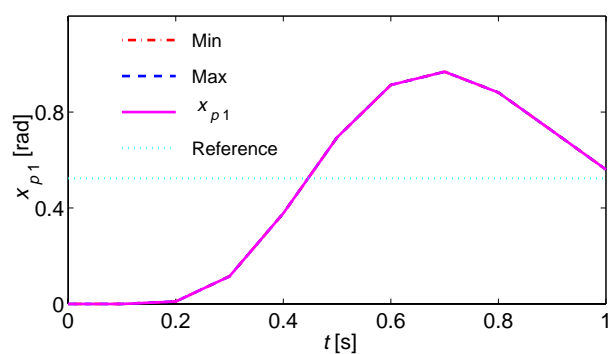
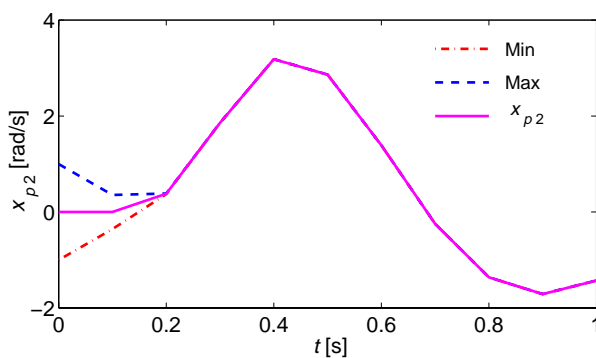
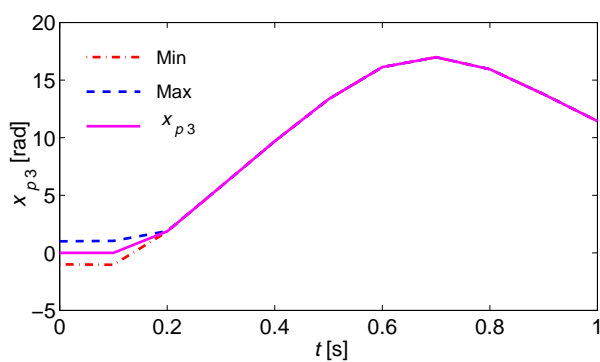
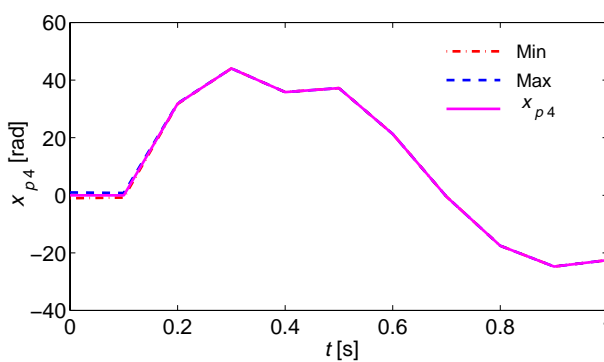
(a) State variable:  $x_{p1} (= \theta_L)$ (b) State variable:  $x_{p2} (= \dot{\theta}_L)$ (c) State variable:  $x_{p3} (= \theta_M)$ (d) State variable:  $x_{p4} (= \dot{\theta}_M)$ 

Fig. 5.4: Simulation results of position servomechanism model

## 5.4 SVO の構成法 II - 外乱を考慮する場合

### 5.4.1 問題設定

本節では、外乱を考慮するものとして制御対象の状態を推定する SVO の構成法について述べる (Fig. 5.5).

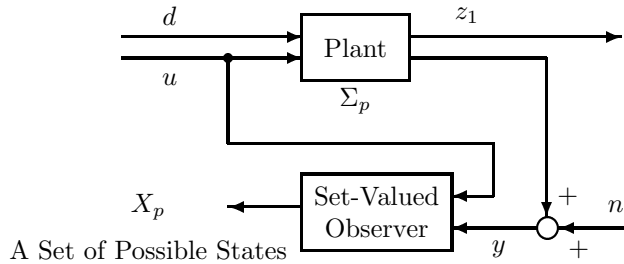


Fig. 5.5: Set-Valued observer

つぎの制御対象  $\Sigma_p$  を考える.

$$\begin{aligned} \Sigma_p \quad x_p(k+1) &= A_p x_p(k) + B_{p1} u(k) + B_{p2} d(k) \\ z_1(k) &= C_{p1} x_p(k) \\ y(k) &= C_{p2} x_p(k) + n(k) \end{aligned} \quad (5.8)$$

ここで  $x_p \in \mathcal{R}^{n_p}$  は制御対象の状態,  $u \in \mathcal{R}^m$  は制御入力,  $z_1 \in \mathcal{R}$  は被制御量,  $y \in \mathcal{R}^{n_y}$  は観測出力である. また,  $d \in \mathcal{R}^{n_d}$  は制御対象への外乱,  $n \in \mathcal{R}^{n_n}$  は観測雑音であり,  $A_p, B_{p1}, B_{p2}, C_{p1}, C_{p2}$  は適当な大きさの定数行列である.

オブザーバを構成するうえで、つぎの仮定をおく.

仮定 5.2.  $B_{\mathcal{R}^{n_p}}$  を  $\mathcal{R}^{n_p}$  全体あるいはつぎで定義される必ずしも有界とは限らない凸多面体とする.

$$B_{\mathcal{R}^{n_p}} = \{x_p \in \mathcal{R}^{n_p} : S_1 x_p \leq S_2\}, \quad S_1 \in \mathcal{R}^{f \times n_p}, \quad S_2 \in \mathcal{R}^f \quad (5.9)$$

このとき制御対象の初期状態は  $x_p(0) \in B_{\mathcal{R}^{n_p}}$  であるとする.

仮定 5.3.  $B_{\mathcal{R}^{n_d}}, B_{\mathcal{R}^{n_n}}$  をつぎで定義される有界な凸多面体とする.

$$B_{\mathcal{R}^{n_d}} = \{d \in \mathcal{R}^{n_d} : D_1 d \leq D_2\}, \quad D_1 \in \mathcal{R}^{g \times n_d}, \quad D_2 \in \mathcal{R}^g \quad (5.10)$$

$$B_{\mathcal{R}^{n_n}} = \{n \in \mathcal{R}^{n_n} : N_1 n \leq N_2\}, \quad N_1 \in \mathcal{R}^{h \times n_n}, \quad N_2 \in \mathcal{R}^h \quad (5.11)$$

このとき制御対象に対する外乱  $d(k)$ , 観測雑音  $n(k)$  は各々,  $d(k) \in B_{\mathcal{R}^{n_d}}, n(k) \in B_{\mathcal{R}^{n_n}}$  であるとする.

注意 5.3. 仮定より、考慮する外乱および観測雑音の正確な値は未知ではあるが、存在する範囲 (大きさの幅) は知ることができるものとする.

SVO は, ある時点  $k$  で考えられうるすべての状態を推定するオブザーバである. そこでその状態全体の集合を,  $X_p(k)$  と定義する. するとこのとき  $X_p(k)$  はつぎの二種類の集合の共通集合としてあらわすことができる.

まず, 時点  $k-1$  までの情報に基づいて, 考えられうる状態  $x_p$  全体の集合をつぎのようにあらわす.

$$\begin{aligned} X_{p_{pre}}(k) &= \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(k-1) + B_{p1} u(k-1) + B_{p2} d, \\ &\quad \exists x_p(k-1) \in X_p(k-1), d \in B_{\mathcal{R}^{n_d}}\} \end{aligned} \quad (5.12)$$

つぎに, 時点  $k$  の観測出力  $y(k)$  の情報のみに基づいて, 考えられうる状態  $x_p$  全体の集合をつぎのようにあらわす.

$$\tilde{X}_p(y(k)) = \{x_p \in \mathcal{R}^{n_p} : y(k) = C_{p2} x_p + n, n \in B_{\mathcal{R}^{n_n}}\}. \quad (5.13)$$

(5.12), (5.13) の条件を同時にみたす  $x_p$  の集合が時点  $k$  で考えられ得るすべての状態の集合  $X_p(k)$  である.

以上をふまえ, 初期状態から順に状態の存在可能な範囲を考えていく.

## 5.4.2 推定アルゴリズム

- $k=0$ , ( $x_p(0)$  の推定)

$x_p(0)$  の範囲は (5.9) と  $y(0)$  から, つぎであらわされる.

$$\begin{aligned} X_p(0) &= B_{\mathcal{R}^{n_p}} \cap \{x_p \in \mathcal{R}^{n_p} : y(0) = C_{p2} x_p + n, n \in B_{\mathcal{R}^{n_n}}\} \\ &= B_{\mathcal{R}^{n_p}} \cap \tilde{X}_p(y(0)) \end{aligned} \quad (5.14)$$

ここで,  $\tilde{X}_p(y(0))$  は,

$$\begin{aligned} \tilde{X}_p(y(0)) &= \{x_p \in \mathcal{R}^{n_p} : y(0) = C_{p2} x_p + n, n \in B_{\mathcal{R}^{n_n}}\} \\ &= \{x_p \in \mathcal{R}^{n_p} : y(0) - C_{p2} x_p = n \in B_{\mathcal{R}^{n_n}}\} \\ &= \{x_p \in \mathcal{R}^{n_p} : -N_1 C_{p2} x_p \leq N_2 - N_1 y(0)\} \end{aligned}$$

より,

$$\tilde{X}_p(y(0)) = \text{Set}(\tilde{M}_c, \tilde{m}(0)), \quad \tilde{M}_c = -N_1 C_{p2}, \quad \tilde{m}(0) = N_2 - N_1 y(0).$$

と変形できるので, 結果的に  $X_p(0)$  は,

$$X_p(0) = \text{Set}(M(0), m(0)), \quad M(0) = \begin{bmatrix} S_1 \\ \tilde{M}_c \end{bmatrix}, \quad m(0) = \begin{bmatrix} S_2 \\ \tilde{m}(0) \end{bmatrix}.$$

の形であらわすことができる. これにより,  $x_p(0)$  の存在範囲が規定される.

- $k=1$ , ( $x_p(1)$  の推定)

$x_p(1)$  の範囲はつぎであらわされる.

$$\begin{aligned} X_p(1) &= \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(0) + B_{p1}u(0) + B_{p2}d, \exists x_p(0) \in X_p(0), d \in B_{\mathcal{R}^{n_d}}\} \\ &\cap \{x_p \in \mathcal{R}^{n_p} : y(1) = C_{p2}x_p + n, n \in B_{\mathcal{R}^{n_n}}\} \\ &= X_{p_{pre}}(1) \cap \tilde{X}_p(y(1)) \end{aligned} \quad (5.15)$$

ここで,  $X_{p_{pre}}(1)$  は,  $x_p(0) \in X_p(0)$  を満足する  $x_p(0)$  と,  $u(0)$  および考慮する外乱  $d$  の大きさにより予想され得る状態全体の集合を意味する.

$A_p$  が正則であれば,

$$X_{p_{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : A_p^{-1}x_p - A_p^{-1}B_{p1}u(0) - A_p^{-1}B_{p2}d = x_p(0) \in X_p(0), d \in B_{\mathcal{R}^{n_d}}\}$$

と変形できるので, 結果的に,

$$X_{p_{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : M(0)A_p^{-1}x_p - M(0)A_p^{-1}B_{p2}d \leq m(0) + M(0)A_p^{-1}B_{p1}u(0), d \in B_{\mathcal{R}^{n_d}}\}$$

とあらわせる.

$\tilde{X}_p(y(1))$  に関しては,

$$\tilde{X}_p(y(1)) = \text{Set}(\tilde{M}_c, \tilde{m}(1)), \quad \tilde{M}_c = -N_1C_{p2}, \quad \tilde{m}(1) = N_2 - N_1y(1).$$

と変形できるので, 結果的に  $X_p(1)$  は,

$$X_p(1) = \{x_p \in \mathcal{R}^{n_p} : M_1(1)x_p + M_2(1)d \leq m_1(1)\}$$

$$M_1(1) = \begin{bmatrix} M(0)A_p^{-1} \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(1) = \begin{bmatrix} -M(0)A_p^{-1}B_{p2} \\ D_1 \\ \mathbf{0} \end{bmatrix}, \quad m_1(1) = \begin{bmatrix} m(0) + M(0)A_p^{-1}B_{p1}u(0) \\ D_2 \\ \tilde{m}(1) \end{bmatrix}.$$

となる. この場合は, 変数  $d$  を消去するために,  $\text{Rack}^{n_d}[(M_1(1) \ M_2(1)), m_1(1)]$  を求めれば,  $X_p(1) = \text{Set}(M(1), m(1))$  の形に変形できる.

$A_p$  が非正則の場合は,

$$X_{p_{pre}}(1) = \{x_p \in \mathcal{R}^{n_p} : x_p - B_{p1}u(0) - B_{p2}d = A_p x_p(0) \in X_p(0), d \in B_{\mathcal{R}^{n_d}}\}$$

より, 結果的に  $X_p(1)$  は,

$$X_p(1) = \{x_p \in \mathcal{R}^{n_p} : M_1(1)x_p + M_2(1)x_p(0) + M_3(1)d \leq m_1(1)\}$$

$$M_1(1) = \begin{bmatrix} I \\ -I \\ \mathbf{0} \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(1) = \begin{bmatrix} -A_p \\ A_p \\ \mathbf{0} \\ M(0) \\ \mathbf{0} \end{bmatrix}, \quad M_3(1) = \begin{bmatrix} -B_{p2} \\ B_{p2} \\ D_1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad m_1(1) = \begin{bmatrix} B_{p1}u(0) \\ -B_{p1}u(0) \\ D_2 \\ m(0) \\ \tilde{m}(1) \end{bmatrix}.$$



となる. この場合は, 変数  $x_p(0)$ ,  $d$  を消去するために,  $Rack^{n_p+n_d}[(M_1(1) \ M_2(1) \ M_3(t)), m_1(1)]$  を求めれば,  $X_p(1) = \text{Set}(M(1), m(1))$  の形に変形できる.

以上より,  $x_p(1)$  の存在範囲が規定される.

同様の操作を繰り返せば, 時点  $t$  における状態の範囲はつぎのように与えることができる.

- $k = t$ ,  $(x_p(t))$  の推定)

$x_p(t)$  の範囲はつぎであらわされる.

$$\begin{aligned} X_p(t) &= \{x_p \in \mathcal{R}^{n_p} : x_p = A_p x_p(t-1) + B_{p1} u(t-1) + B_{p2} d, \exists x_p(t-1) \in X_p(t-1), \\ &\quad d \in B_{\mathcal{R}^{n_d}}\} \cap \{x_p \in \mathcal{R}^{n_p} : y(t) = C_{p2} x_p + n, n \in B_{\mathcal{R}^{n_n}}\} \\ &= X_{p_{pre}}(t) \cap \tilde{X}_p(y(t)) \end{aligned} \tag{5.16}$$

$\tilde{X}_p(y(t))$  に関しては,

$$\tilde{X}_p(y(t)) = \text{Set}(\tilde{M}_c, \tilde{m}(t)), \quad \tilde{M}_c = -N_1 C_{p2}, \quad \tilde{m}(t) = N_2 - N_1 y(t).$$

と変形できるので, 結果的に  $X_p(t)$  は,

$A_p$  が正則であれば,

$$\begin{aligned} X_p(t) &= \{x_p \in \mathcal{R}^{n_p} : M_1(t)x_p + M_2(t)d \leq m_1(t)\} \\ M_1(t) &= \begin{bmatrix} M(t-1)A_p^{-1} \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(t) = \begin{bmatrix} -M(t-1)A_p^{-1}B_{p2} \\ D_1 \\ \mathbf{0} \end{bmatrix}, \\ m_1(t) &= \begin{bmatrix} m(t-1) + M(t-1)A_p^{-1}B_{p1}u(t-1) \\ D_2 \\ \tilde{m}(t) \end{bmatrix}. \end{aligned}$$

となる. この場合は, 変数  $d$  を消去するために,  $Rack^{n_d}[(M_1(t) \ M_2(t)), m_1(t)]$  を求めれば,  $X_p(t) = \text{Set}(M(t), m(t))$  の形に変形できる.

$A_p$  が非正則の場合は,

$$X_{p_{pre}}(t) = \{x_p \in \mathcal{R}^{n_p} : x_p - B_{p1}u(t-1) - B_{p2}d = A_p x_p(t-1) \in X_p(t-1), d \in B_{\mathcal{R}^{n_d}}\}$$

より, 結果的に  $X_p(t)$  は,

$$\begin{aligned} X_p(t) &= \{x_p \in \mathcal{R}^{n_p} : M_1(t)x_p + M_2(t)x_p(t-1) + M_3(t)d \leq m_1(t)\} \\ M_1(t) &= \begin{bmatrix} I \\ -I \\ \mathbf{0} \\ \mathbf{0} \\ \tilde{M}_c \end{bmatrix}, \quad M_2(t) = \begin{bmatrix} -A_p \\ A_p \\ \mathbf{0} \\ M(t-1) \\ \mathbf{0} \end{bmatrix}, \quad M_3(t) = \begin{bmatrix} -B_{p2} \\ B_{p2} \\ D_1 \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad m_1(t) = \begin{bmatrix} B_{p1}u(t-1) \\ -B_{p1}u(t-1) \\ D_2 \\ m(t-1) \\ \tilde{m}(t) \end{bmatrix}. \end{aligned}$$

となる. この場合は, 変数  $x_p(t-1)$ ,  $d$  を消去するために,  $Rack[\cdot]$  の操作を  $n_p + n_d$  回おこな  
い,  $Rack^{n_p+n_d}[(M_1(t) \ M_2(t) \ M_3(t)), m_1(t)]$  を求めれば,  $X_p(t) = Set(M(t), m(t))$  の形に変  
形できる.

以上より,  $x_p(t)$  の存在範囲が規定される. したがって推定則はつぎのようにまとめられる.

### アルゴリズム 5.2. $X_p(k)$ の推定アルゴリズム

$k = 0$ :

$$\begin{aligned} X_p(0) &= B_{\mathcal{R}^{n_p}} \cap \tilde{X}_p(y(0)) \\ &= Set(M(0), m(0)) \\ (M(0), m(0)) &= \left[ \begin{pmatrix} S_1 \\ \tilde{M}_c \end{pmatrix}, \begin{pmatrix} S_2 \\ \tilde{m}(0) \end{pmatrix} \right], \tilde{M}_c = -N_1 C_{p2}, \tilde{m}(0) = N_2 - N_1 y(0). \end{aligned}$$

$k = t > 0$ :

$$\begin{aligned} X_p(t) &= X_{p_{pre}}(t) \cap \tilde{X}_p(y(t)) \\ &= Set(M(t), m(t)) \end{aligned}$$

〈 $A_p$  が正則

$$\begin{aligned} &(M(t), m(t)) \in \\ Rack^{n_d} &\left[ \begin{pmatrix} M(t-1)A_p^{-1} & -M(t-1)A_p^{-1}B_{p2} \\ \mathbf{0} & D_1 \\ \tilde{M}_c & \mathbf{0} \end{pmatrix}, \begin{pmatrix} m(t-1) + M(t-1)A_p^{-1}B_{p2}u(t-1) \\ D_2 \\ \tilde{m}(t) \end{pmatrix} \right], \\ &\tilde{M}_c = -N_1 C_{p2}, \tilde{m}(t) = N_2 - N_1 y(t). \end{aligned}$$

〈 $A_p$  が非正則

$$\begin{aligned} (M(t), m(t)) &\in Rack^{n_p} \left[ \begin{pmatrix} I & -A_p & -B_{p2} \\ -I & A_p & B_{p2} \\ \mathbf{0} & \mathbf{0} & D_1 \\ \mathbf{0} & M(t-1) & \mathbf{0} \\ \tilde{M}_c & \mathbf{0} & \mathbf{0} \end{pmatrix}, \begin{pmatrix} B_{p2}u(t-1) \\ -B_{p2}u(t-1) \\ D_2 \\ m(t-1) \\ \tilde{m}(t) \end{pmatrix} \right], \\ &\tilde{M}_c = -N_1 C_{p2}, \tilde{m}(t) = N_2 - N_1 y(t). \end{aligned}$$

注意 5.4. SVO の構成アルゴリズムは, 対象の安定性, 可観測性については全く言及しない.

## 5.4.3 例題

例題 5.2. 離散時間系の状態空間モデルが以下で与えられる 2 次の制御対象を考える.

$$\begin{aligned} \begin{bmatrix} x_{p1}(k+1) \\ x_{p2}(k+1) \end{bmatrix} &= \begin{bmatrix} 0.95123 & 0 \\ 0.04877 & 1 \end{bmatrix} \begin{bmatrix} x_{p1}(k) \\ x_{p2}(k) \end{bmatrix} + \begin{bmatrix} 0.04877 \\ 0.00123 \end{bmatrix} u(k) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} d(k) \\ z_1(k) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p1}(k) \\ x_{p2}(k) \end{bmatrix} \\ y(k) &= \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_{p1}(k) \\ x_{p2}(k) \end{bmatrix} + n(k) \end{aligned} \quad (5.17)$$

また, 補償器はつぎで与えられるものとする.

$$\begin{aligned} x_c(k+1) &= 0.60653 x_c(k) + 0.62955 r(k) - 0.62955 y(k) \\ u(k) &= -25 x_c(k) + 50 r(k) - 50 y(k) \end{aligned} \quad (5.18)$$

この制御対象と補償器からなる閉ループ系に, Set-Valued オブザーバを構成し, 制御対象の状態  $x_p = [x_{p1} \ x_{p2}]^T$  を推定した結果を Fig. 5.6 に示す. Fig. 5.6 において,  $x_{pi}$  は状態  $x_p$  の各要素の真値をあらわし, Max, Min は状態  $x_p$  の各要素の推定値  $\hat{x}_{pi}$  の最大値, 最小値をあらわす. また Reference は制御系に加える参照入力 (大きさは 1) をあらわす. 推定の初期条件は  $|x_{pi}| \leq 1$  とした. なお, 仮定 5.3 に対応して  $B_{\mathcal{R}^{n_d}}, B_{\mathcal{R}^{n_n}}$  としてつぎで定義される有界な凸多面体を考える.

$$\begin{aligned} B_{\mathcal{R}^{n_d}} &= \left\{ d \in \mathcal{R}^{n_d} : \begin{bmatrix} 1 \\ -1 \end{bmatrix}^T d \leq \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}^T \right\} \\ B_{\mathcal{R}^{n_n}} &= \left\{ n \in \mathcal{R}^{n_n} : \begin{bmatrix} 1 \\ -1 \end{bmatrix}^T n \leq \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix}^T \right\} \end{aligned}$$

これは外乱および観測雑音として,  $|d| \leq 0.1, |n| \leq 0.1$  を想定することを意味する. ただしここでは想定するのみであり, 実際に外乱および観測雑音は加えていない.

また各時点においてオブザーバにより推定される凸多面体  $X_p(k)$  と実際の状態のトラジェクトリーを Fig. 5.7 に示す.

シミュレーションより, 凸多面体の推定値  $X_p(k)$  は常にその内部に真の状態を含みながら, 収束することが確かめられた. また線形拘束式の数に冗長な拘束式を消去した結果, 高々6本であった.

例題 5.3. 例題 5.2 と同一の条件で実際に外乱及び観測雑音を加える. 加える数値は各々,

$$\begin{aligned} d(k) &= 0.1 \cos 25\pi(0.05k) \\ n(k) &= 0.05 \cos 6\pi(0.05k) \end{aligned}$$

とした. 結果を Fig. 5.8 に示す.

シミュレーションより, 外乱および観測雑音を実際に加えた場合においても, 凸多面体の推定値  $X_p(k)$  は常にその内部に真の状態を含みながら, 収束することが確かめられた. また線形拘束式の数に冗長な拘束式を消去した結果, 高々7本であった.

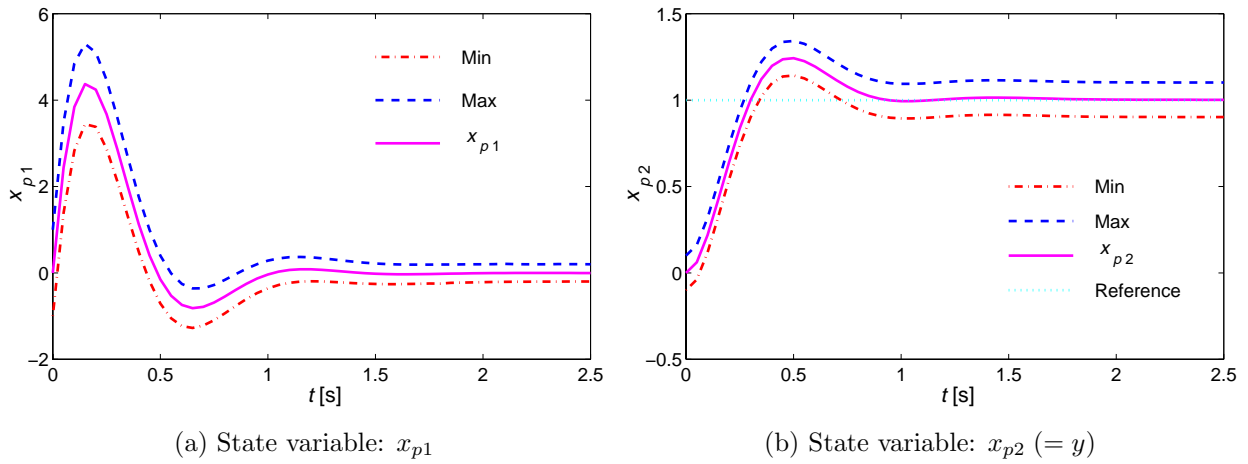


Fig. 5.6: State estimation results

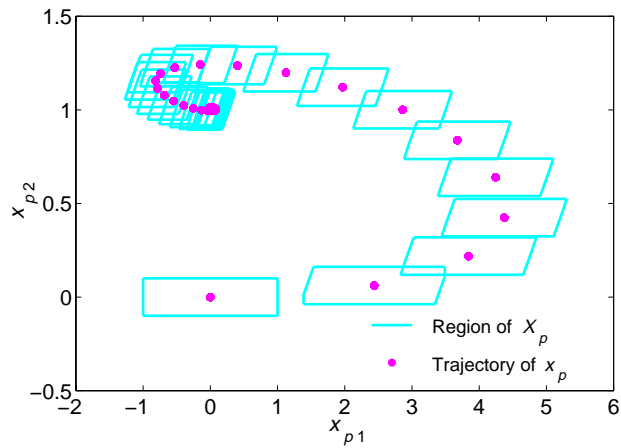


Fig. 5.7: Sets of possible states  $X_p(k)$  and trajectory

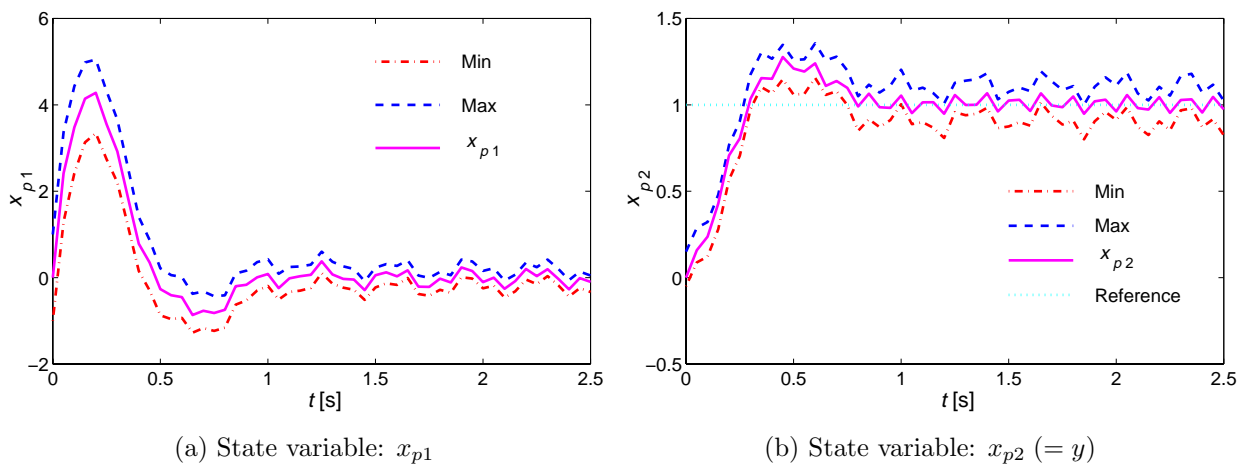


Fig. 5.8: State estimation results (with disturbance inputs)



# 第6章 出力フィードバックに基づく拘束システムのスイッチング制御

## 6.1 はじめに

拘束条件を有する系に対する制御手法として、4章ではスイッチング制御について述べた。スイッチング制御は、線形制御理論の枠組で複数の補償器を設計し、これらを切り換えることにより拘束条件による影響の軽減を目指すものである。ここでの特徴として、個々の補償器を構成する際に、その補償器を適用することが許される状態空間の領域‘最大CPI集合’を同時に構成することが挙げられる。補償器を切り換えるスーパーバイザは、最大CPI集合が与える情報と、制御対象の状態変数に基づき、適用可能な補償器のうち最も制御性能の優れているものを各時刻で選択する (Fig. 4.1)。これにより、拘束条件を有する制御系を広い状態空間の領域で安定化し、また同時に制御性能の劣化を抑制する。

しかしながら、これまでに提案されているスイッチング制御則は、補償器の切り換えの指針として状態空間上の領域を用いることから、状態変数が観測可能であるという仮定が必要不可欠である。しかし実システムを考えた場合、対象の状態変数すべてを観測することは困難であるため、前章で状態変数を推定する機構として Set-Valued オブザーバの構成法とその有効性について述べた。

本章では、前章で述べた Set-Valued オブザーバを状態推定機構として適用することにより、出力フィードバックに基づくスイッチング制御則の実現を考える (Fig. 6.1)。またその効果と問題点について考察する。

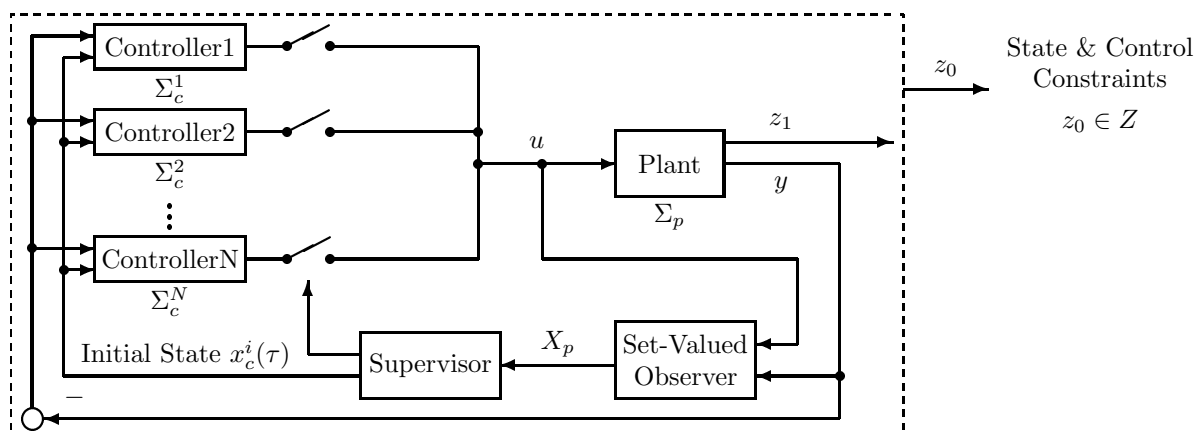


Fig. 6.1: Switching control

## 6.2 問題設定

本章では、制御対象  $\Sigma_p$  に対して、 $N$  個の補償器  $\Sigma_c^1, \Sigma_c^2, \dots, \Sigma_c^N$  を構成し、それらのスイッチングに基づいたレギュレータ問題を考える (Fig. 6.1).

つぎの制御対象  $\Sigma_p$  を考える.

$$\begin{aligned} \Sigma_p \quad x_p(k+1) &= A_p x_p(k) + B_p u(k) \\ z_1(k) &= C_{p1} x_p(k) \\ y(k) &= C_{p2} x_p(k) \end{aligned} \quad (6.1)$$

ここで  $x_p \in \mathcal{R}^{n_p}$  は制御対象の状態,  $u \in \mathcal{R}^m$  は制御入力,  $z_1 \in \mathcal{R}$  は被制御量である. また  $y \in \mathcal{R}^{n_y}$  は観測出力であり,  $A_p, B_p, C_{p1}, C_{p2}$  は適当な大きさの定数行列である.

制御対象  $\Sigma_p$  に適用される補償器として、ここではつぎの線形フィードバック補償器  $\Sigma_c^i$  を考える.

$$\begin{aligned} \Sigma_c^i \quad x_c^i(k+1) &= A_c^i x_c^i(k) + B_c^i y(k) \\ u(k) &= C_c^i x_c^i(k) + D_c^i y(k), \quad i = 1, \dots, N \end{aligned} \quad (6.2)$$

ここで  $x_c^i \in \mathcal{R}^{n_{ci}}$  は補償器の状態,  $A_c^i, B_c^i, C_c^i, D_c^i$  は適当な大きさの定数行列である. 各補償器の次元  $n_{ci}$  は必ずしも等しくある必要はない. また静的な補償器に関しては, (6.2) において, 次数  $n_{ci} = 0$ ,  $u(k) = D_c^i y(k)$  と考えれば適用可能である.

**注意 6.1.** 各補償器はすでに構成されており、特にその系列  $\Sigma_c^i$ ,  $i = 1, \dots, N$  は順に制御性能が向上するように与えられているとする. したがって,  $\Sigma_c^N$  が最も望ましい制御性能を有する.

$\Sigma_p, \Sigma_c^i$  により構成される閉ループ系  $\Sigma^i$  はつぎのようになる.

$$\begin{aligned} \Sigma^i \quad x^i(k+1) &= A^i x^i(k) \\ z_1(k) &= C x^i(k) \end{aligned} \quad (6.3)$$

ただしここで  $x = \begin{bmatrix} x_p^T & (x_c^i)^T \end{bmatrix}^T \in \mathcal{R}^{n^i}$ ,  $n^i := n_p + n_{ci}$  であり、各行列はつぎで与えられる.

$$A^i = \begin{bmatrix} A_p + B_p D_c^i C_{p2} & B_p C_c^i \\ B_c^i C_{p2} & A_c^i \end{bmatrix} \in \mathcal{R}^{n^i \times n^i} \quad C = \begin{bmatrix} C_{p1} & \mathbf{0} \end{bmatrix} \in \mathcal{R}^{n_y \times n^i}$$

つぎに閉ループ系  $\Sigma^i$  に対する拘束条件を表現するために以下で与えられる変数  $z_0^i(k)$  を設ける.

$$\begin{aligned} z_0(k) &= C_{0p} x_p(k) + D_0 u(k) \\ &= C_0^i x^i(k) \in Z \subset \mathcal{R}^p \end{aligned} \quad (6.4)$$

集合  $Z$  としては、つぎのような凸多面体を考える.

$$Z = \{z_0 \in \mathcal{R}^p : F z_0 \leq f\}, \quad F \in \mathcal{R}^{r \times p}, f \in \mathcal{R}^r. \quad (6.5)$$

**仮定 6.1.** 閉ループ系  $\Sigma^i$  は漸近安定 ( $A^i$  の固有値が単位円内部),  $(C_0^i, A^i)$  は可観測であるとす. また,  $z_0(k)$  が十分小さい場合には拘束が破られないという観点から,  $0 \in \text{int} Z$  を仮定する. さらに  $Z$  は有界であるとする.

## 6.3 最大 CPI 集合

3章でふれたように, 制御系が拘束を破ることなく動作するための必要十分条件は, その初期状態が最大 CPI 集合に属していることである.

まず (6.3) の各閉ループ系  $\Sigma^i$  に対して最大 CPI 集合をつぎのように定義する.

定義 6.1. ( 最大 CPI 集合 )

$x_0^i = x^i(0)$  とする.  $\Sigma^i$  に対する最大 CPI 集合  $O_\infty^i$  をつぎのように定義する.

$$O_\infty^i = \{ x_0^i \in \mathcal{R}^{n^i} \mid z_0(k) = C_0^i (A^i)^k x_0^i \in Z, \forall k \in \mathcal{Z}^+ \}, \quad i = 1, \dots, N. \quad (6.6)$$

注意 6.2.  $O_\infty^i \neq \emptyset$ ,  $i = 1, \dots, N$  とする.  $O_\infty^i$  は  $\Sigma_c^i$  に対するすべての CPI 集合を内部に含む最大の CPI 集合である. すなわち  $O_\infty^i$  は, 各閉ループ系  $\Sigma^i$  が安全に動作できる最も大きな領域を与える. 任意の初期状態  $x^i(0) \in O_\infty^i$  に対して閉ループ系  $\Sigma^i$  の応答は常に  $x^i(k) \in O_\infty^i$ , かつ  $z_0(k) \in Z$  をみたす. 言い換えれば,  $\Sigma^i$  が常に拘束をみたすための必要十分条件は,  $x^i(0) \in O_\infty^i$  が成立することである.

注意 6.3. 最大 CPI 集合  $O_\infty^i$  は有界な凸多面体であり, 線形計画法を利用した数値計算により構成され, つぎのような形で表現される [6, 8].

$$O_\infty^i = \{ x_0^i \in \mathcal{R}^{n^i} : M_0^i x_0^i \leq m_0^i \}, \quad M_0^i \in \mathcal{R}^{g^i \times n^i}, \quad m_0^i \in \mathcal{R}^{g^i}. \quad (6.7)$$

ここで  $M_0^i$ ,  $m_0^i$  は, 凸多面体  $O_\infty^i$  を規定する線形拘束式を表現する行列であり,  $g^i$  は線形拘束式の数である. なおここでの不等式は, ベクトルの各要素ごとに成立しているものとする.

本節で述べるスイッチング制御は, この  $O_\infty^i$  と, オブザーバにより推定される状態の集合  $X_p$  の情報を指針として補償器の切り換えを達成するものである. 次節では補償器の切り換えについて考える.

## 6.4 補償器の切り換え

補償器を切り換えるスーパーバイザは, 最大 CPI 集合  $O_\infty^i$  が与える情報と SVO(アルゴリズム 5.1) が与える制御対象の状態の集合  $X_p(k)$  に基づき, 適用可能な補償器のうち最も制御性能の優れているものを各時刻で選択する. したがってスーパーバイザの具体的な役割は,

- 使用する補償器の選定
- 新たに使用する補償器の初期状態の決定

をオンラインで実行することである. これらについて考える.



## (補償器の選定)

補償器の選定は各時点における制御対象の状態の集合  $X_p(k)$  の値に依存する. 基本概念は, 現在の状態の集合  $X_p(k)$  に対しどの補償器を使用すれば, 安全でかつ最も制御性能の良い閉ループ系を構成できるかである.

これまで述べたように, 安全な制御の判断基準は最大 CPI 集合にある. 各  $O_\infty^i$  は制御対象と補償器の次数を合わせた  $n_p + n_{ci}$  次元の空間で定義されている. そこでまずこれを以下のように制御対象の状態空間へ射影する.

$$O_{\infty p}^i = \{ x_p \in \mathcal{R}^{n_p} \mid \exists x_c^i \in \mathcal{R}^{n_{ci}}, \begin{bmatrix} x_p \\ x_c^i \end{bmatrix} \in O_\infty^i \}, \quad i = 1, \dots, N \quad (6.8)$$

各  $O_\infty^i$  が凸多面体であるため, これを射影した  $O_{\infty p}^i$  もまた凸多面体である.  $O_{\infty p}^i$  の具体的な表現は, 2章で述べた Fourier-Motzklin の消去法に基づいてえることが可能である. この  $O_{\infty p}^i$  を用いると, 補償器  $\Sigma_c^i$  が適用可能となる条件が,  $X_p(k) \subset O_{\infty p}^i$  と表現される.

スイッチング制御を考える場合, 少なくとも一つは適用可能である補償器が存在しなければ制御則の実現は不可能であるため, つぎの仮定をおく.

仮定 6.2. 制御対象の初期状態  $x_p(0)$  は,  $x_p(0) \in O_{\infty p}^i$  であるとする. ここでインデックス  $i$  は既知である. すなわち, 仮定 5.1 において,  $B_{\mathcal{R}^{n_p}} = O_{\infty p}^i$  とする.

つぎに,

$$O_{\infty p} = \bigcup_{i=1}^N O_{\infty p}^i$$

とする.  $O_{\infty p}$  は, 適用可能となる補償器が少なくとも一つは存在する制御対象  $\Sigma_p$  の状態すべてから成っている. ここで,  $X_p \subset O_{\infty p}$  に対して,  $I(X_p) = \{ i \mid X_p \subset O_{\infty p}^i \}$  を定義すれば, この集合は状態の集合  $X_p$  に対して適用可能な補償器のインデックスをすべて集めたものとなる. 補償器は順に制御性能が向上するように構成されているため,  $I(X_p(k))$  のなかから最も大きな値が選択されることが望ましい. したがってスーパーバイザは, 以下の方針で使用する補償器をオンラインで選択する.

$$i(k) = \max\{ i \mid X_p(k) \subset O_{\infty p}^i \}. \quad (6.9)$$

注意 6.4. ここでの補償器の選定は, 4章とは異なり, ‘状態の集合’ がどの最大 CPI 集合に属しているかを判定しなければならない. SVO により推定される状態の集合  $X_p$  は前章からわかるようにいくつかの線形拘束式で表現される凸多面体である. よってここでは以下の方法で判定を行う.

アルゴリズム 6.1. 閉ループ系  $\Sigma^i$  に対する最大 CPI 集合を制御対象の状態空間へ射影した  $O_{\infty p}^i$ , および SVO により推定された状態の集合  $X_p(k)$  が以下で表現される凸多面体であるとする.

$$\begin{aligned} O_{\infty p}^i &= \{ x_p \in \mathcal{R}^{n_p} : M_p^i x_p \leq m_p^i \}, \quad M_p^i \in \mathcal{R}^{g^i \times n_p}, \quad m_p^i \in \mathcal{R}^{g^i}. \\ X_p(k) &= \{ x_p \in \mathcal{R}^{n_p} : M(k) x_p \leq m(k) \}, \quad M(k) \in \mathcal{R}^{v(k) \times n_p}, \quad m(k) \in \mathcal{R}^{v(k)}. \end{aligned} \quad (6.10)$$

ここで  $M_p^i$ ,  $m_p^i$ ,  $M_X(k)$ ,  $m_X(k)$  は, 凸多面体  $O_{\infty p}^i$ ,  $X_p(k)$  を規定する線形拘束式を表現する行列であり,  $g^i$ ,  $v(k)$  は線形拘束式の数である. なおここでの不等式は, ベクトルの各要素ごとに成立しているものとする.

ここで,  $X_p(k) \subset O_{\infty p}^i$  の判定条件として線形計画問題

$$h = \text{maximize } (M_{p(j,:)}^i x_p), \quad \text{subject to } M(k) \leq m(k) \quad (6.11)$$

を解く. このとき, すべての行  $j = 1, \dots, g^i$  に対して  $h \leq m_{p(j)}^i$  であるならば,  $X_p(k) \subset O_{\infty p}^i$  が成立する.

注意 6.5. SVO による推定値は各時点で考えられうる状態全体の集合である. そのため, 制御対象の状態は観測可能という仮定をおいた 4 章と比較すれば やや切り換えのタイミングが遅れる可能性がある.

スーパーバイザは  $i(\tau) \neq i(\tau - 1)$  が成立した時刻  $\tau$  のみで補償器を切り換える. 切り換えが実行される際には, 新たに適用される補償器  $\Sigma_c^i$  の初期状態  $x_c^{i(\tau)}(\tau)$  を同時に決定することが必要となる.

(補償器の初期状態の決定)

新たに適用される補償器  $\Sigma_c^i$  の初期状態  $x_c^{i(\tau)}(\tau)$  の決定において,  $x_c^{i(\tau)}(\tau)$  がみたさなければならぬ最低限の条件は,  $x_p \in X_p(\tau)$  をみたく  $x_p$  に対して

$$\begin{bmatrix} x_p \\ x_c^{i(\tau)}(\tau) \end{bmatrix} \in O_{\infty}^{i(\tau)} \quad (6.12)$$

が成り立つことである.  $X_p(\tau) \subset O_{\infty p}^i$  であれば,  $O_{\infty p}^i$  の定義 (6.8) より, このような  $x_c^{i(\tau)}(\tau)$  は必ず存在し, またこの条件をみたくすれば全体の閉ループ系は拘束条件を破ることなく動作する. 本稿での制御目的は初期状態  $x_p(0)$  の影響を速やかに減衰させることであるため,  $\|x_c^i\|$  を最小化することを考え, 以下の二次計画問題を解き, 条件をみたく  $x_c^i$  を初期状態として与える.

$$\text{minimize } \begin{bmatrix} x_p^T & (x_c^i)^T \end{bmatrix}^T \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & I \end{bmatrix} \begin{bmatrix} x_p \\ x_c^i \end{bmatrix} \quad \text{subject to } \begin{bmatrix} M(\tau) & \mathbf{0} \\ M_1^i & M_2^i \end{bmatrix} \begin{bmatrix} x_p \\ x_c^i \end{bmatrix} \leq \begin{bmatrix} m(\tau) \\ m_0^i \end{bmatrix}. \quad (6.13)$$

ここで,  $I$  は  $n_{ci} \times n_{ci}$  の単位行列,  $M$ ,  $m$  は (6.10) で扱った変数である. また  $M_1^i$ ,  $M_2^i$ ,  $m_0^i$  は (6.7) で,

$$M_0^i = [M_1^i \ M_2^i] \in \mathcal{R}^{g^i \times n^i}, \quad M_1^i \in \mathcal{R}^{g^i \times n_p}, \quad M_2^i \in \mathcal{R}^{g^i \times n_{ci}}.$$

と定義したものである. ここでえられる  $x_c^i$  は条件 (6.12) をみたくすものである.

注意 6.6. 一般に, (6.12) をみたくす初期状態は複数存在し, またこの選択が制御入力の高さに直接関係することを考えれば, 全体の制御性能に大きな影響を与えることは明らかである. したがって全体の制御性能までを考慮した初期状態の有力な決定法の導出が必要な課題となっている.

注意 6.7. 新たに適用される補償器が静的なもの ( $n_{ci} = 0$ ) である場合は初期化の必要は生じない.

以上から補償器の切り換えに関してつぎの定理がえられる.

定理 6.1. 仮定 6.1, 6.2が成立しているとする. このとき, 制御対象  $\Sigma_p$ , 補償器  $\Sigma_c^i$ ,  $i = 1, \dots, N$  および (6.9), (6.12) のスイッチング制御則により構成される閉ループ系は漸近安定である. またこのとき, 拘束条件  $z_0(k) \in Z$  はすべての時刻において達成される.

(証明) 拘束条件が存在しない場合, 仮定 6.1 より各閉ループ系  $\Sigma^i$  は漸近安定である. また, 最大 CPI 集合の性質から,  $X_p(k) \subset O_{\infty p}^i \rightarrow X_p(k+1) \subset O_{\infty p}^i$  が成立する. いま時刻  $t$  において,  $x_p(t) \in O_{\infty p}^i$ ,  $X_p(t) \subset O_{\infty p}^i$  であったとすると, 定理 5.1 より  $x_p(\tau) \in O_{\infty p}^{i+1}$ ,  $X_p(\tau) \subset O_{\infty p}^{i+1}$  なる  $\tau > t$  が必ず存在し,  $\Sigma_c^i \rightarrow \Sigma_c^{i+1}$  の補償器の切り換えが生じる. したがって, スイッチング制御則 (6.9), (6.12) により, 任意の初期状態  $x_p(0) \in O_{\infty p}^i$  ( $i$  は既知) に対する応答は収束し, またこのとき適用される補償器  $\Sigma_c^i$  のインデックスの値  $i(k)$  は単調に増加するのみである. さらに, 拘束条件  $z_0(k) \in Z$  はすべての時刻において達成される.

## 6.5 例題

例題 6.1. 4章の例題 4.1を再び考える. 制御対象および適用する補償器は同一である. 例題 4.1と異なる点は, スーパーバイザが切り換えの指針として, SVOにより推定された状態の集合を用いる点である.

まず, 仮定 6.2に対応して, つぎの初期条件をおく.

制御対象の初期状態  $x_p(0)$  は Fig. 4.4の  $O_{\infty p}^1$  に属している. すなわち,  $B_{\mathcal{R}^{n_p}} = O_{\infty p}^1$  である. 初期状態の真値は,  $x_p(0) = [1.5 \ 0]^T \in O_{\infty p}^1$  とする.

この初期条件に対して例題 4.1と同様のシミュレーションを行った結果を Fig. 6.2, Fig. 6.3に示す. Fig. 6.2は, 各時点での推定値  $X_p(k)$ の規定する領域および実際の状態  $x_p(k)$ の軌跡を2次元の状態空間上に示している. Fig. 6.3は各状態の推定値の最大, 最小値, 実際の状態の値, および制御入力の時応答を示している. 結果から推定値  $X_p(k)$ は時点  $k = 1(0.5[s])$ には真値に収束し, 以降常に真値と等しいことがわかる. これは定理 5.1に沿っている. 状態  $x_{p1}$ に関しては出力として観測できるために,  $k = 0(0.0[s])$ で真値に一致する. Fig. 6.2において,  $k = 0(0.0[s])$ における推定値が直線として与えられているのはこのためである. 以上よりSVOの推定はうまく機能しているといえる.

スイッチング制御の観点では, SVOの推定値の真値への収束が, 最初の切り換えの時刻より早いために, 応答は初期時点を除いてすべて例題 4.1と同一である. Fig. 6.3には, 比較のために単一の補償器  $\Sigma_c^1$ のみを用いた場合の結果も示してある. スwitching制御則により, 拘束を満足すると同時に性能の向上が達成されることが確認できる.

また一方で, SVOによる推定はいくつかの線形拘束式によって可能な状態の集合を表現するため, 推定自体のほかに補償器の選定, および補償器の初期状態の決定の際にやや時間を要するという問題が残る. この点の改善が課題の一つである.

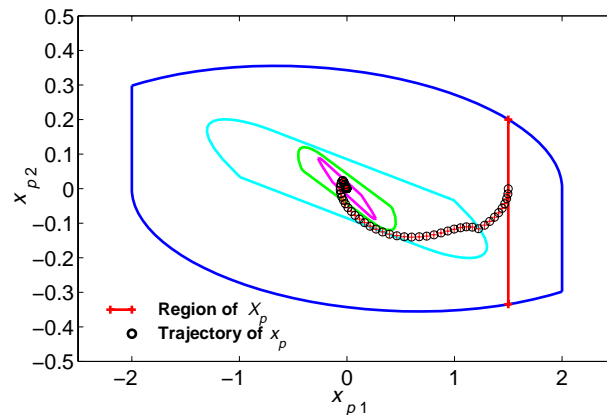
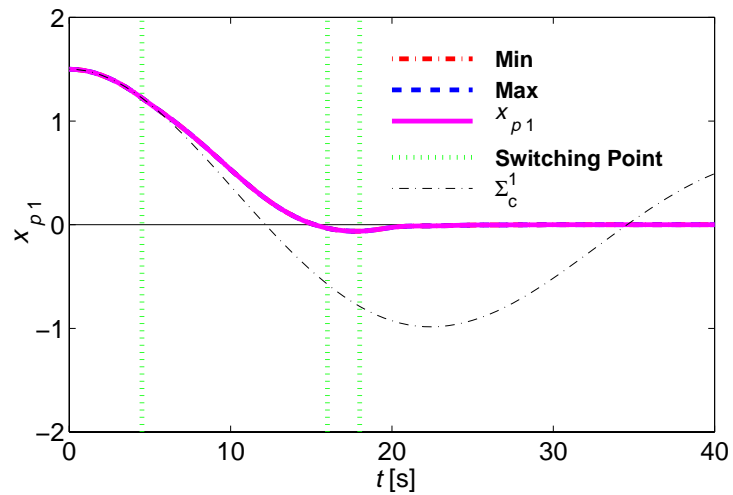
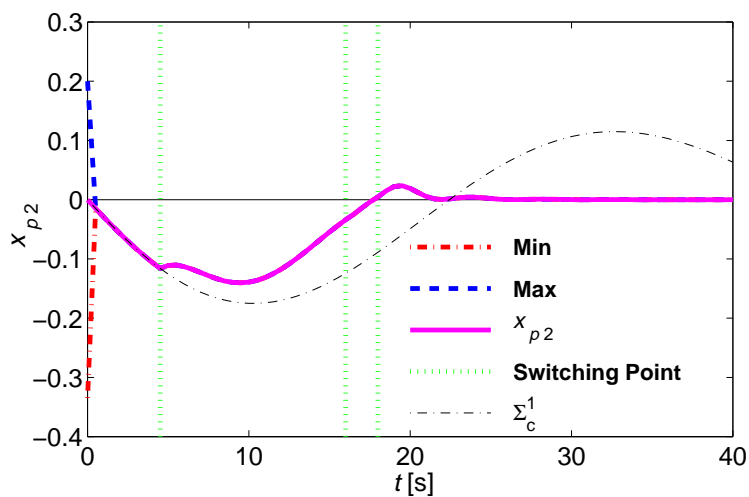


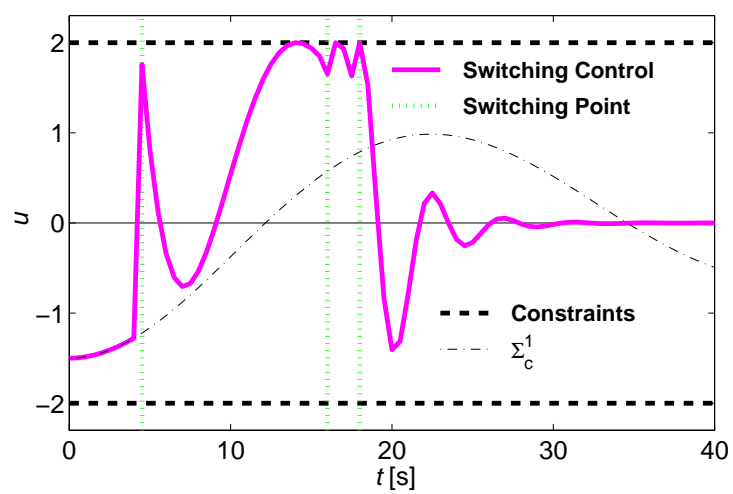
Fig. 6.2: State trajectories



(a) State variable  $x_{p1}(=z_1)$



(b) State variable  $x_{p2}$



(c) Control input  $u$

Fig. 6.3: Initial condition response (Switching control)

# 第7章 結論

## 7.1 本研究のまとめ

本論文は、制御入力および状態に拘束条件を有する系に対する制御系設計法として、スイッチング制御に注目し研究を行った。

本論文でえられた結果を要約すると以下ようになる。

- 制御系が課せられた拘束条件を破ることなく安全に動作する条件について述べた。安全な動作のための基盤は、最大 CPI 集合と呼ばれる状態空間上に規定される集合である。最大 CPI 集合は拘束条件の達成を保証する閉ループ系の初期状態の集合であり、すなわちこの集合に制御系の状態変数が属していれば、拘束は常に満足される。
- 拘束システムに対する制御手法として近年注目されている、スイッチング制御に関して、その有効性および問題点について述べた。スイッチング制御は、複数の補償器を線形制御理論の枠組で構成し、これらを付加的な補償機構であるスーパーバイザにより切り換えながら適用する制御手法である。安全な動作領域を規定する最大 CPI 集合は、制御対象と各補償器からなる閉ループ系に対し構成される。スーパーバイザは、各時点でえられる制御対象の状態と、最大 CPI 集合の情報から、安全な最も性能の優れた補償器を選択する。それにより拘束システムを広い動作範囲で安定化できるとともに、制御性能の劣化も抑制される。ただし、制御則の実現には、状態変数が観測可能であるという仮定が必要となる。
- 状態推定機構として、Set-Valued オブザーバの構成法について述べた。Set-Valued オブザーバの特徴は、システムのダイナミクスとえられる入出力データから、各時点で想定される状態変数すべてを集合として推定する点にある。推定される集合は、線形拘束式の組で規定される凸多面体として与えられる。したがって状態変数の真値は、推定された集合の内部に属することが保証されるため、正確な状態変数の情報が要求される制御には有効であるといえる。ここでは外乱を考慮しない場合と考慮する場合の二種類のオブザーバを構成し、シミュレーションによりその有効性を確認した。特に外乱を考慮しない場合は有限時間での真値への収束が保証された。
- 状態推定機構として、Set-Valued オブザーバを適用し、本論文の主題である、出力フィードバックに基づくスイッチング制御則を導出した。スーパーバイザは、各時点でえられる制御対象の状態変数の推定値（線形拘束式で規定される集合）と、最大 CPI 集合の情報から、安全な最も性能の優れた補償器を選択する。それにより拘束システムを広い動作範囲で安定化できるとともに、制御性能の劣化も抑制される。ただし切り換えの指針として用いる状態変数が集合として与えられるため、場合によっては、状態フィードバックに基づいた制御則に比べ、切り換えのタイミングが遅れる可能性がある。

## 7.2 今後の課題

本論文での提案手法を含めスイッチング制御系の構成において、以下の課題が残る。

- スwitching制御則を構成する個々の補償器の具体的な設計指針を与える。
- 切り換えの際に新しく使用する補償器の初期状態をどのように決定するか。本論文では、補償器の状態変数のノルムを最小化することで決定したが、それが最適であるという理論的裏付けはない。
- 線形拘束式のオンライン処理に要する時間の短縮。Set-Valued オブザーバの推定においては、線形拘束式を扱うために、その数に応じてオンラインでの計算負荷も変化する。またスイッチング制御則との結合を考えると、切り換えの判定や初期状態の決定の際のオンライン計算にも影響を与える。
- 外部入力を考慮した場合のスイッチング制御則の導出。本論文では外部入力を考慮しない系に対して制御則を導出した。外部入力を有する系に対する最大 CPI 集合の構成法は提案され [5]、種々の制御則に応用されているため [6, 7, 8]、拡張は可能であると考えられる。

## 参考文献

- [1] A. Bemporad and E. Mosca: Fulfilling Hard Constraints in Uncertain Linear Systems by Reference Managing, *Automatica*, **34**–4, 451/461, 1998.
- [2] E. G. Gilbert and K. T. Tan: Linear Systems with State and Control Constraints: The Theory and Application of Maximal Output Admissible Sets, *IEEE Transactions on Automatic Control*, **36**–9, 1008/1020, 1991.
- [3] E. G. Gilbert, I. Kolmanovsky and K. T. Tan: Discrete-time Reference Governors and the Nonlinear Control of Systems with State and Control Constraints, *International Journal of Robust and Nonlinear Control*, **5**, 487/504, 1995.
- [4] George. B. Dantzig and Mukund N. Thapa: *Linear Programming, 1: Introduction*, Springer Series in Operations Research, Springer, 1997.
- [5] 平田, 藤田: 外部入力を有する線形離散時間システムに対する拘束条件の解析, 電気学会論文誌 C, **118**–3, 384/390, 1998.
- [6] 平田, 藤田: 入力に制限を有する線形離散時間システムに対するスイッチング状態フィードバック制御則の構成法, システム制御情報学会論文誌, **11**–6, 316/323, 1998.
- [7] 平田, 藤田: 参照入力集合の連結にもとづく拘束システムの制御, 第 21 回 Dynamical System Theory シンポジウム, 239/242, 1998.
- [8] 平田, 藤田: 外部入力を有する拘束システムに対するスイッチング状態フィードバック制御則の構成法, システム制御情報学会論文誌, **12**–7, 1999. (掲載予定)
- [9] I. Kolmanovsky and E. G. Gilbert: Multimode Regulators for Systems with State & Control Constraints and Disturbance Inputs, *Control using logic-based switching*, Lecture Notes in Control and Information Sciences, **222**, 104/117, Springer, 1997.
- [10] Jeff S. Shamma and Edward Tu: Optimality of Set-Valued Observers for Linear Systems, *Proceedings of the 34th IEEE Conf. on Decision and Control*, 2081/2086, New Orleans, LA, December 1995.
- [11] Jeff S. Shamma: Optimization of the  $l^\infty$ -Induced norm Under Full State Feedback, *IEEE Transactions on Automatic Control*, **AC-41**(4): 533/544, 1996.
- [12] Jeff S. Shamma and Kuang-Yang Tu: Output Feedback Control for Systems with Constraints and Saturations: Scalar Control Case, *Systems & Control Letters*, **35**, 1/11, 1998.



- [13] S. S. Keerthi and E. G. Gilbert: Computation of Minimum-Time Feedback Control Laws for Discrete-Time Systems with State-Control Constraints, *IEEE Transactions on Automatic Control*, **AC-32**(5): 432/435, 1987.
- [14] 須田 他: PID 制御, 朝倉書店, 1992.
- [15] Vasek Chvatal: *Linear Programming*, FREEMAN, 1983.
- [16] 渡辺, 内田, 藤田: 出力可到達集合解析に基づいたアンチwindアップコントローラの解析, 第 26 回 制御理論シンポジウム, 15/20, 1997.
- [17] Y. Peng, D. Vrancic and R. Hanus: Anti-Windup, Bumpless, and Conditioned Transfer Techniques for PID Controllers, *IEEE Control Systems*, **16**-4, 48/57, 1996.

# 付録A

## A.1 定理5.1の証明

証明のために、まずつぎの補題を考える.

補題 A.1.  $\text{rank}(H) = n_p$  であるとき,  $\text{rank}(H^T H)$  の値もまた  $n_p$  である.

(証明)

$\text{rank}(H^T H) \neq n_p$ , すなわち  $H^T H x_p = 0, \exists x_p \neq 0$  であると仮定する. このとき

$$x_p^T H^T H x_p = 0 \Rightarrow \|H x_p\| = 0 \Rightarrow H x_p = 0$$

であるが, いま  $\text{rank}(H) = n_p$  であるから

$$H x_p = 0 \Leftrightarrow x_p = 0.$$

したがって,  $H^T H x_p = 0$  となるのは  $x_p = 0$  のみとなり仮定に矛盾する.

以上より,  $\text{rank}(H^T H) = n_p$ .

□

(証明) [定理 5.1]

〈 $A_p$  が正則のとき〉 アルゴリズム 5.1 より,

$$X_p(0) = \{x_p(0) \in \mathcal{R}^{n_p} : \begin{bmatrix} S_1 \\ -C_{p2} \\ C_{p2} \end{bmatrix} x_p(0) \leq \begin{bmatrix} S_2 \\ -y(0) \\ y(0) \end{bmatrix}\}$$

$$X_p(1) = \{x_p(1) \in \mathcal{R}^{n_p} : \begin{bmatrix} S_1 \\ -C_{p2} \\ C_{p2} \\ -C_{p2}A_p \\ C_{p2}A_p \end{bmatrix} A_p^{-1}x_p(1) \leq \begin{bmatrix} S_2 + S_1A_p^{-1}B_pu(0) \\ -y(0) - C_{p2}A_p^{-1}B_pu(0) \\ y(0) + C_{p2}A_p^{-1}B_pu(0) \\ -y(1) \\ y(1) \end{bmatrix}\}$$

$$X_p(2) = \{x_p(2) \in \mathcal{R}^{n_p} : \begin{bmatrix} S_1 \\ -C_{p2} \\ C_{p2} \\ -C_{p2}A_p \\ C_{p2}A_p \\ -C_{p2}A_p^2 \\ C_{p2}A_p^2 \end{bmatrix} A_p^{-2}x_p(2) \leq \begin{bmatrix} S_2 + S_1(A_p^{-1}B_pu(0) + A_p^{-2}B_pu(1)) \\ -y(0) - C_{p2}(A_p^{-1}B_pu(0) + A_p^{-2}B_pu(1)) \\ y(0) + C_{p2}(A_p^{-1}B_pu(0) + A_p^{-2}B_pu(1)) \\ -y(1) - C_{p2}A_p^{-1}B_pu(1) \\ y(1) + C_{p2}A_p^{-1}B_pu(1) \\ -y(2) \\ y(2) \end{bmatrix}\}$$

と順に  $X_p(k)$  を求めると,  $n_p - 1$  時点では,

$$X_p(n_p - 1) = \{x_p(n_p - 1) \in \mathcal{R}^{n_p} : G_1 A_p^{-(n_p-1)} x_p(n_p - 1) \leq G_2\} \quad (\text{A.1})$$

$$G_1 = \begin{bmatrix} S_1 \\ -C_{p2} \\ C_{p2} \\ -C_{p2}A_p \\ C_{p2}A_p \\ \vdots \\ -C_{p2}A_p^{n_p-2} \\ C_{p2}A_p^{n_p-2} \\ -C_{p2}A_p^{n_p-1} \\ C_{p2}A_p^{n_p-1} \end{bmatrix}, G_2 = \begin{bmatrix} S_2 + S_1(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ -y(0) - C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ y(0) + C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ -y(1) - C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p - 2)) \\ y(1) + C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p - 2)) \\ \vdots \\ -y(n_p - 2) - C_{p2}A_p^{-1}B_p u(n_p - 2) \\ y(n_p - 2) + C_{p2}A_p^{-1}B_p u(n_p - 2) \\ -y(n_p - 1) \\ y(n_p - 1) \end{bmatrix}$$

となる. (A.1) の  $x_p(n_p - 1)$  に対する条件はつぎの式を同時にみたすことと等価である.

$$S_1 A_p^{-(n_p-1)} x_p(n_p - 1) \leq S_2 + S_1(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \quad (\text{A.2})$$

$$\begin{bmatrix} C_{p2} \\ C_{p2}A_p \\ \vdots \\ C_{p2}A_p^{n_p-2} \\ C_{p2}A_p^{n_p-1} \end{bmatrix} A_p^{-(n_p-1)} x_p(n_p - 1) \leq \begin{bmatrix} y(0) + C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ y(1) + C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p - 2)) \\ \vdots \\ y(n_p - 2) + C_{p2}A_p^{-1}B_p u(n_p - 2) \\ y(n_p - 1) \end{bmatrix} \quad (\text{A.3})$$

$$- \begin{bmatrix} C_{p2} \\ C_{p2}A_p \\ \vdots \\ C_{p2}A_p^{n_p-2} \\ C_{p2}A_p^{n_p-1} \end{bmatrix} A_p^{-(n_p-1)} x_p(n_p - 1) \leq - \begin{bmatrix} y(0) + C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ y(1) + C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p - 2)) \\ \vdots \\ y(n_p - 2) + C_{p2}A_p^{-1}B_p u(n_p - 2) \\ y(n_p - 1) \end{bmatrix} \quad (\text{A.4})$$

(A.2) は変形すると結果的に

$$S_1 x_p(0) \leq S_2$$

となり, これは (5.2) で与えた初期条件に等しいので常に成り立つ.

(A.3), (A.4) は, つぎの等式条件と等価である.

$$\begin{bmatrix} C_{p2} \\ C_{p2}A_p \\ \vdots \\ C_{p2}A_p^{n_p-2} \\ C_{p2}A_p^{n_p-1} \end{bmatrix} A_p^{-(n_p-1)} x_p(n_p - 1) = \begin{bmatrix} y(0) + C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p - 2)) \\ y(1) + C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p - 2)) \\ \vdots \\ y(n_p - 2) + C_{p2}A_p^{-1}B_p u(n_p - 2) \\ y(n_p - 1) \end{bmatrix} \quad (\text{A.5})$$

ここで, (A.5) をつぎのように定義する.

$$HA_p^{-(n_p-1)}x_p(n_p-1) = \eta \quad (\text{A.6})$$

ただし,

$$H = \begin{bmatrix} C_{p2} \\ C_{p2}A_p \\ \vdots \\ C_{p2}A_p^{n_p-2} \\ C_{p2}A_p^{n_p-1} \end{bmatrix}, \quad \eta = \begin{bmatrix} y(0) + C_{p2}(A_p^{-1}B_p u(0) + \cdots + A_p^{-(n_p-1)}B_p u(n_p-2)) \\ y(1) + C_{p2}(A_p^{-1}B_p u(1) + \cdots + A_p^{-(n_p-2)}B_p u(n_p-2)) \\ \vdots \\ y(n_p-2) + C_{p2}A_p^{-1}B_p u(n_p-2) \\ y(n_p-1) \end{bmatrix}$$

である.

いま仮定より,  $(C_{p2}, A_p)$  は可観測であるから

$$\text{rank}(H) = n_p \quad (\text{A.7})$$

補題 A.1 より,  $\text{rank}(H^T H) = n_p$  であるから  $H^T H$  は正則.

(A.6) の両辺に左から  $(H^T H)^{-1}H^T$  をかけて

$$(H^T H)^{-1}H^T(HA_p^{-(n_p-1)}x_p(n_p-1)) = (H^T H)^{-1}H^T\eta.$$

したがって,

$$A_p^{-(n_p-1)}x_p(n_p-1) = (H^T H)^{-1}H^T\eta$$

より,

$$x_p(n_p-1) = A_p^{n_p-1}(H^T H)^{-1}H^T\eta \quad (\text{A.8})$$

となる.

以上より, アルゴリズム 5.1 による  $n_p - 1$  時点での推定値は一意に決定される.  $\square$

〈 $A_p$  が非正則のとき〉 アルゴリズム 5.1 より,

$$X_p(0) = \{x_p(0) \in \mathcal{R}^{n_p} : \begin{bmatrix} S_1 \\ -C_{p2} \\ C_{p2} \end{bmatrix} x_p(0) \leq \begin{bmatrix} S_2 \\ -y(0) \\ y(0) \end{bmatrix} \}$$

$$X_p(1) = \{x_p(1) \in \mathcal{R}^{n_p} : \begin{bmatrix} I & -A_p \\ -I & A_p \\ \mathbf{0} & S_1 \\ \mathbf{0} & -C_{p2} \\ \mathbf{0} & C_{p2} \\ -C_{p2} & \mathbf{0} \\ C_{p2} & \mathbf{0} \end{bmatrix} \begin{bmatrix} x_p(1) \\ x_p(0) \end{bmatrix} \leq \begin{bmatrix} B_p u(0) \\ -B_p u(0) \\ S_2 \\ -y(0) \\ y(0) \\ -y(1) \\ y(1) \end{bmatrix} \}$$



$$S_1 x_p(0) \leq S_2 \quad (\text{A.12})$$

(A.12) は初期条件より常に成り立つ。

(A.10) より,

$$\begin{aligned} x_p(1) &= A_p x_p(0) + B_p u(0) \\ x_p(2) &= A_p x_p(1) + B_p u(1) = A_p^2 x_p(0) + A_p B_p u(0) + B_p u(1) \\ x_p(3) &= A_p x_p(2) + B_p u(2) = A_p^3 x_p(0) + A_p^2 B_p u(0) + A_p B_p u(1) + B_p u(2) \\ &\vdots \\ x_p(n_p - 1) &= A_p x_p(n_p - 2) + B_p u(n_p - 2) \\ &= A_p^{n_p - 1} x_p(0) + A_p^{n_p - 2} B_p u(0) + \cdots + A_p B_p u(n_p - 3) + B_p u(n_p - 2) \end{aligned} \quad (\text{A.13})$$

とすべて  $x_p(0)$  を含む式であらわせる。各式を (A.11) に代入すると, (A.11) はつぎのように変形できる。

$$\begin{bmatrix} C_{p2} A_p^{n_p - 1} \\ C_{p2} A_p^{n_p - 2} \\ \vdots \\ C_{p2} A_p \\ C_{p2} \end{bmatrix} x_p(0) = \begin{bmatrix} y(n_p - 1) - C_{p2} (A_p^{n_p - 2} B_p u(0) + \cdots + A_p B_p u(n_p - 3) + B_p u(n_p - 2)) \\ y(n_p - 2) - C_{p2} (A_p^{n_p - 3} B_p u(0) + \cdots + A_p B_p u(n_p - 4) + B_p u(n_p - 3)) \\ \vdots \\ y(1) - C_{p2} B_p u(0) \\ y(0) \end{bmatrix} \quad (\text{A.14})$$

ここで, (A.14) をつぎのように定義する。

$$H_2 x_p(0) = \eta_2 \quad (\text{A.15})$$

ただし,

$$\begin{aligned} H_2 &= \begin{bmatrix} C_{p2} A_p^{n_p - 1} \\ C_{p2} A_p^{n_p - 2} \\ \vdots \\ C_{p2} A_p \\ C_{p2} \end{bmatrix} \\ \eta_2 &= \begin{bmatrix} y(n_p - 1) - C_{p2} (A_p^{n_p - 2} B_p u(0) + \cdots + A_p B_p u(n_p - 3) + B_p u(n_p - 2)) \\ y(n_p - 2) - C_{p2} (A_p^{n_p - 3} B_p u(0) + \cdots + A_p B_p u(n_p - 4) + B_p u(n_p - 3)) \\ \vdots \\ y(1) - C_{p2} B_p u(0) \\ y(0) \end{bmatrix} \end{aligned}$$

である。

いま仮定より,  $(C_{p2} \ A_p)$  は可観測であるから

$$\text{rank}(H_2) = n_p \quad (\text{A.16})$$

補題 A.1 より,  $\text{rank}(H_2^T H_2) = n_p$  であるから  $H_2^T H_2$  は正則。

(A.15) の両辺に左から  $(H_2^T H_2)^{-1} H_2^T$  をかけて

$$(H_2^T H_2)^{-1} H_2^T (H_2 x_p(0)) = (H_2^T H_2)^{-1} H_2^T \eta_2.$$

したがって,

$$x_p(0) = (H_2^T H_2)^{-1} H_2^T \eta_2 \quad (\text{A.17})$$

となる. (A.17) を (A.13) に代入すれば

$$x_p(n_p - 1) = A_p^{n_p-1} (H_2^T H_2)^{-1} H_2^T \eta_2 + A_p^{n_p-2} B_p u(0) + \cdots + A_p B_p u(n_p - 3) + B_p u(n_p - 2) \quad (\text{A.18})$$

となる.

以上より, アルゴリズム 5.1 による  $n_p - 1$  時点での推定値は一意に決定される.  $\square$

# 謝辞

本研究を進めるにあたり、終始暖かい御指導をいただいた、北陸先端科学技術大学院大学情報科学研究科 藤田政之 助教授に心より感謝の意を表します。

また、多くの御支援と御教示をいただきました、北陸先端科学技術大学院大学 示村悦二郎 先生に深く感謝いたします。

また、多くの御指導をいただきました、北陸先端科学技術大学院大学情報科学研究科 望山洋 助手に心より感謝いたします。

そして、研究における懇切丁寧な御指導だけでなく、日常生活においても暖かい御指導、御助言をいただいた、本講座博士後期課程 平田研二氏をはじめ 先輩の皆様にも心より感謝いたします。

また、良き相談相手となっていたいただいた、研究室の同輩の皆様にも心より感謝いたします。

最後に、これまでの研究生活を可能にし、暖かく見守り、また励ましてくださった両親に深く感謝いたします。