

博士論文

化学プラントにおけるプロセス制御の性能向上と  
新たなモデル予測制御に関する研究

東京農工大学大学院工学府応用化学専攻

山下研究室

昆 潤一郎

# 目次

第 1 章	緒論	3
1.1	本研究の目的	3
1.2	本論文の概要	4
第 2 章	参照軌道に基づく PID コントローラの最適チューニング	7
2.1	緒言	7
2.2	PID チューニングパラメータの最適設計アルゴリズム	9
2.2.1	PID コントローラアルゴリズム	9
2.2.2	参照軌道に基づく PID チューニングパラメータの設定	10
2.3	デザイン例	11
2.3.1	1 次遅れ系システム	11
2.3.2	他の手法との比較	12
2.4	提案手法の応用	18
2.4.1	積分系システム	18
2.4.2	I-PD コントローラ	19
2.4.3	制約条件を考慮した PID パラメータチューニング	22
2.5	ダイナミックシミュレーションと DCS による PID チューニング	26
2.5.1	シミュレーションモデル	28
2.5.2	モデル同定	29
2.5.3	DCS における PID パラメータデザイン	30
2.6	結言	36
第 3 章	伝達関数を応用した ARX モデルによるモデル同定	41
3.1	緒言	41
3.2	モデル同定手法	42
3.2.1	同定に用いるモデル構造	42
3.2.2	モデル同定	47
3.3	提案手法によるモデル同定例	47
3.3.1	2 次遅れ系モデル (Time Constant)	48
3.3.2	積分系モデル	49
3.4	既往手法との比較	51
3.4.1	既往の伝達関数を用いたモデル同定手法	51

3.4.2	比較結果	52
3.5	閉ループモデル同定による実プロセスの PID 制御チューニング	52
3.6	多変数への拡張	61
3.6.1	アルゴリズム	61
3.6.2	モデル予測制御 (MPC) の閉ループモデル同定	62
3.6.3	更新したモデルによる制御性能	64
3.7	実プロセスへの応用: 製油所の蒸留塔	67
3.7.1	適用プロセス	67
3.7.2	現行モデルの評価	69
3.7.3	再同定結果	69
3.7.4	試運転	73
3.8	結言	74
<b>第 4 章</b>	<b>ARX モデルを使用したモデル予測制御</b>	<b>77</b>
4.1	緒言	77
4.2	ARX モデル予測制御アルゴリズム	79
4.2.1	ARX モデル	79
4.2.2	制御アルゴリズム	80
4.3	PCR と ARX モデル予測制御の制御性比較	81
4.3.1	PCR アルゴリズム	81
4.3.2	制御性比較	82
4.4	未測定外乱に対するチューニング	83
4.5	様々な応答プロセスへの適用	88
4.5.1	逆応答を持つプロセス	88
4.5.2	積分系プロセス	89
4.5.3	特殊な応答プロセス	92
4.6	非線形プロセスへの適用	92
4.6.1	制御対象プロセス	94
4.6.2	ARX モデル予測制御の CSTR への適用	96
4.6.3	非線形モデル予測制御	97
4.6.4	制御シミュレーション結果	97
4.7	結言	99
<b>第 5 章</b>	<b>結論</b>	<b>103</b>
<b>付 録 A</b>	<b>ARX モデルの AR 項から MA 項を求める方法</b>	<b>105</b>

# 第1章 緒論

## 1.1 本研究の目的

20 世紀、めざましい科学技術の発展をとげ、その技術は様々な産業の分野に様々な変革をもたらすようになり、現在もさらなる技術および産業が発展を続けている。これらの発展により、我々の生活環境が改善され、さらなる生活環境の改善を試みるようになってきた。そして、この生活環境の改善により物資の生産意欲が促進され、交通機関や通信メディアが発展を遂げ、これらのスパイラルにより、様々な面でさらなる進化を続けている。その一方で、よい面とは反対に、見落とされがちになっていた環境問題がこれらの発達により深刻化してきている。20 世紀後半の 1980 年代には、山岳の氷河の後退、北極海の海水が薄くなったりなどの地球規模の現象で環境問題が地球温暖化としてはっきり現れるようになってきた。この地球温暖化の原因は、温室効果ガスとされる CO<sub>2</sub> の人為的な排出量が増えきたことによることが大きいとされている。発展のためには、莫大なエネルギーを必要とし、そのエネルギーが消費されることで、温室効果ガスが増加する。地球温暖化の原因はほとんどの原因が人間活動自体のエネルギー消費にあるため、地球温暖化は環境問題の中でもっとも解決が難しい問題である。そこで、すべての人々が地球温暖化対策を行っていく必要がある。そして、地球温暖化対策としての省エネルギーは各企業が行うべき活動の一つとなってきた。特に、莫大なエネルギーを消費するプロセス産業では、必須な活動となってきた。この活動で注目されているのがプロセス制御であり、これはハード面での改造を最小限に抑え、ソフト面での改造ですむというメリットがあり、省エネルギーでの地球温暖化対策としてこの技術が期待されている。

日本のプロセス産業 (石油精製、石油化学、化学系等) において、原料は円高にある現状では海外で調達する方向にあり、さらに、製品の価格が下り、日本の人件費が高いという状況で、国内での生産を続けるのはメリットが少ないということで、海外にプラントをたて、現地で製品の生産を行い輸入するという方法をとるようになってきている。そのために、国内の企業では、プラントを閉鎖したり、会社の合併や統合したりと生き残る努力を続けている。こういう状況の中においても、プラント内では原単位を下げるための省エネルギーの試算を行い続けている。よって、さらなる省エネルギーを実現する方法として、プロセス制御が非常に重要な技術となってきた。

そこで本研究では、プロセス産業におけるプロセス制御の制御性能を容易に向上させるための提案を行う。

## 1.2 本論文の概要

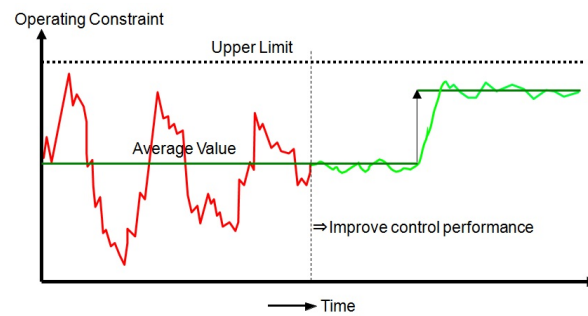
本論文の概要を述べる前に、プロセス制御の役割を考えると、第一にプロセス運転の安定性向上があげられる。この安定性が向上することで、図 1.1 に示すように製品仕様 (Upper Limit) から大きなマージンを取って運転していたところから、製品仕様に近づけての運転が可能になる。製品仕様に近づけて運転することでメリットが得られることになるが、そのイメージを図 1.2 に示す。まず、制御性を改善することで、設定値の周りの出現頻度は広いもの (青) から狭いもの (緑) になる。このことにより製品仕様に近づけてもオフスペックを回避しての運転が可能になる。運転コストは製品仕様に近づけると減らすことができる。この例としては、蒸留塔では製品仕様を守るために大きくマージンをとってリボイラースチームの炊きあげ過剰運転を行っているが、製品仕様に近づけることで過剰炊き上げを防ぎ、省エネ運転が実現可能となる。これがプロセス制御適用およびプロセス制御性能向上によるコストメリットである。さて、プロセス制御性能の向上して、安定性を改善し、コストメリットを得られるためには図 1.3 に示す 3 つの柱が必要であると考え。制御性能の向上のためには、制御対象に対して精度のよいプロセスモデリングを行い、それをもとに制御のチューニングパラメータ (コントローラチューニング) を決定すればよい。もしくは、制御性能が上がるようなコントローラアルゴリズムの開発が必要である。開発されたアルゴリズムを適用する際にも、精度のよいモデリングによりチューニングパラメータを決める必要がある。そこで、本研究は図 1.3 に示すプロセス制御に対する 3 つの柱についての研究を行うことにした。具体的な研究内容にふれる前に、プロセスプラントでのオペレーション階層とその周期を図 1.4 に示す。会社のマネジメントサイドで製品の生産計画を立案して、それをプラント側でプラント運転最適化に落とし込む。この場合、製品仕様に近づけるのはこの最適化の役目になる。最適化の下層にモデル予測制御があり、その下にさらに DCS 制御があり、DCS 制御では大部分が PID コントローラをベースとしたプロセス制御となっている。モデル予測制御は DCS の PID コントローラに設定値を落とす形が一般的であるので、PID コントローラの制御性能がモデル予測制御の性能を左右していることはこのことから明らかである。そこで、本研究として、第 2 章にコントローラのチューニングに関して PID コントローラチューニング手法を提案する。それから、その上層にモデル予測制御があるので、第 4 章にコントローラアルゴリズムに関して新たなモデル予測制御アルゴリズムを提案する。PID コントローラやモデル予測制御のデザインにはプロセスモデルが必要であるので、第 3 章にプロセスモデリングに関して新たなモデル同定手法を提案する。本論文の構成をまとめると以下ようになる。

第 2 章 参照軌道に基づく PID コントローラの最適チューニング

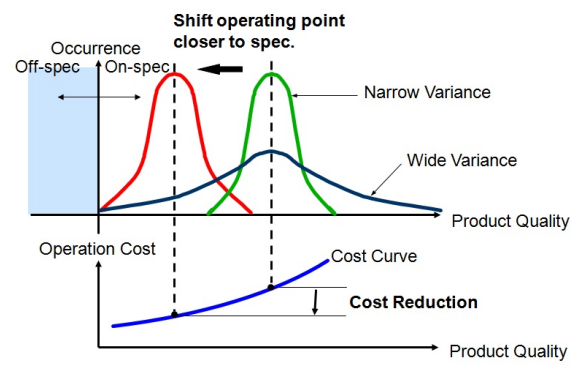
第 3 章 伝達関数を応用した ARX モデルによるモデル同定

第 4 章 ARX モデルを使用したモデル予測制御

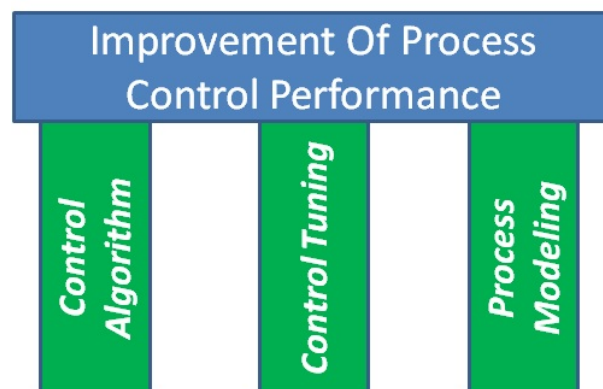
第 5 章 結論



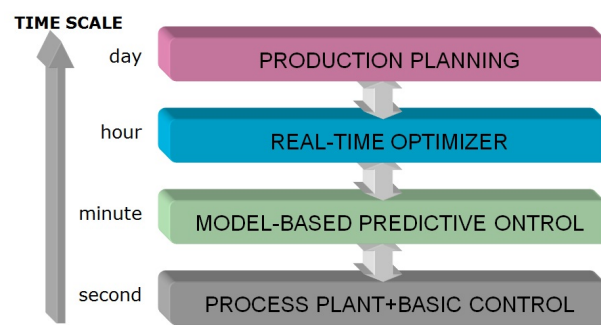
☒ 1.1: Increased Stability



☒ 1.2: Benefit of Increased Stability



☒ 1.3: Necessities for Control Performance Improvement



☒ 1.4: Typical Operation Layer & Cycle

## 第2章 参照軌道に基づくPIDコントローラの最適チューニング

### 2.1 緒言

PIDコントローラは設定値と測定値との偏差による比例 (Proportional)、積分 (Integral)、微分 (Derivative) をもとにしたシンプルな SISO (Single Input/Single Output) 構造のフィードバックシステムである。PIDコントローラは当初制御工学のフィードバックコントロール技術として開発された。制御工学の起源は 1788 年に比例制御アルゴリズムをベースにしたワットのスチームエンジンの回転スピード制御 [1] である。これについてマックスウェルが 1868 年に論文 “On Governors” を発表した [2]。これが制御理論の始まりである。マイノフスキーは 1922 年に PID コントローラの原型 [3] を提案した。制御工学は伝達関数と PID コントローラをベースとした古典制御理論から始まり現代制御理論に発展した。制御技術は今も発展を続けているが、制御工学の PID コントローラは未だに数多く適用され、役に立つ技術である。実際に、プロセス産業においてカスケードループを含む流量制御、温度制御、圧力制御、液面制御のような基本となるレギュラトリコントロールの 90% 以上が PID コントローラアルゴリズムをベースとした制御ループである [4]。PID コントローラは古典制御理論の代表的なプロダクトであるとともに未だに数多くの制御アプリケーションとして使われている。これは PID コントローラアルゴリズムがシンプル、効果的、ほとんどすべてのプロセスを制御するのに十分であるからで、さらに、DCS や PLC のような制御システムでは標準的な機能として装備されていて、簡単に適用・チューニングが行えるからである。式 (2.1) に比例帯を使用した PID コントローラを示す。

$$u_n(t) = \frac{100}{PB} \left( e_n(t) + \frac{1}{T_I} \int e_n(t) dt + T_D \frac{de_n(t)}{dt} \right) \quad (2.1)$$

ここで、 $u_n(t)$ 、 $e_n(t)$ 、 $PB$ 、 $T_I$ 、 $T_D$  はそれぞれ、正規化されたプロセス入力、設定値と測定値との正規化された偏差、比例帯、積分時間、微分時間である。

日揮株式会社のプラント試運転の経験より表 2.1 に典型的なチューニングパラメータをまとめた。

プロセスプラントではメカニカルな完成の後、プラント試運転が始まる。表 2.1 に示す PID チューニングパラメータを PID コントローラの初期値としてこの試運転が開始される。そして、この試運転中に PID チューニングパラメータは、ファインチューニングされる。試運転の後、PID コントローラの制御性能が悪くなった場合、その制



表 2.1: Typical PID Tuning Parameters

Controlled Variable	$PB$ [%]	$T_I$ [min]	$T_D$ [min]
Flow	$\approx 100$	$\approx 1$	-
Level	$\approx 150-200$	$\approx 2-3$	-
Pressure (Liquid)	$\approx 100$	$\approx 0.5$	-
Pressure (Vapor)* <sup>1</sup>	$< 50-100$	* <sup>2</sup>	-
Temperature	* <sup>3</sup> ( $\approx 100$ at initial)	* <sup>3</sup>	* <sup>4</sup>

Note

- \*1:一般的に、速いチューニングパラメータが適用される
- \*2:制御目的による
- \*3:システムによる
- \*4:多くのケースで微分時間を使う

御ループは再チューニングされる。しかし、コントローラの制御性能がよほど劣化しない限り、PID チューニングパラメータが再チューニングされることはほとんどない。この理由は以下の通りである。

- (1) 典型的な PID チューニングパラメータである程度の制御性能が確保できる
- (2) PID コントローラのチューニングパラメータが 3 つあり、組み合わせで性能が決まるために複雑である

しかしながら、プロセス産業で使用されているコントロールアルゴリズムのほとんどが PID コントローラであり、このことは PID コントローラのチューニングによりプラントの性能が決まってくるという意味になる。PID コントローラの制御性能が向上すれば、プラントのスループット、製品品質、製品収率、エネルギー消費量、安全性、安定性、操作性、生産性を改善できる。従って、PID コントローラのチューニングに関して、多くの研究、特許、パッケージが出回っている [4]。

PID コントローラのチューニング状況であるが、PID コントローラが期待通りの制御性能を発揮していない割合が 66%[20]、80%[21] と報告されている。PID コントローラの制御性能を発揮させるための改善策が必要であり、この PID コントローラの制御性能が改善された場合、どんなところにどの程度の効果があるかが、以下のように Matrikon 社の資料に記載されている。

- プラントの処理量向上：2-5%
- 製品品質の分散の減少：30%

- エネルギー消費の減少：5–10%
- 製品収率の向上：2–5%
- プロセスの安定性向上、運転の容易さの向上

本研究では、新たな参照軌道を使用した最適な PID チューニングパラメータ決定方法を提案する。プロセスモデルパラメータをベースにして PID チューニングパラメータを求める既往の研究では、マクローリン展開などの近似を使っている。しかしながら、本提案手法では、近似なしに参照軌道をベースとして最適化技術を使い、直接 PID チューニングパラメータを求めることができる。参照軌道と最適化を組み合わせることで、様々な形の参照軌道により PID チューニングパラメータ決定が行える新しい手法である。さらに、制約を考慮した PID チューニングパラメータ選定が可能であるという大きな特徴を持つ (詳細は、2.4.3 節参照)

## 2.2 PID チューニングパラメータの最適設計アルゴリズム

### 2.2.1 PID コントローラアルゴリズム

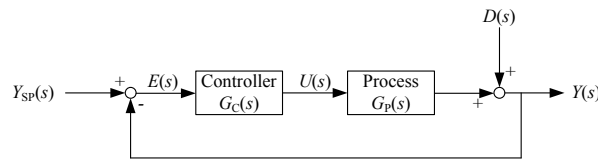


図 2.1: Block Diagram for Feedback Control

PID コントローラは図 2.1 に示すフィードバック制御アルゴリズムであり、3つの項 (比例動作、積分動作、微分動作) から構成されている。一般的に、ラプラス変換領域において PID コントローラのアルゴリズムは以下の式で与えられる:

$$\begin{aligned} U(s) &= G_C(s)E(s) \\ &= K_P \left( 1 + \frac{1}{T_I s} + T_D s \right) E(s) \end{aligned} \quad (2.2)$$

ここで、 $K_P$  はコントローラの比例ゲインである。

比例項は設定値と測定値の偏差に比例してプロセス入力进行操作する。積分項は比例アクションだけでは設定値と測定値の間に残るオフセットを補償する。無駄時間を持つ 2 次遅れシステムを含む大熱容量や鈍い応答を持つようなプロセスの場合、比例と

積分アクションだけではプロセスを制御するには十分な制御性能が得られない。なぜなら、安定するのに時間がかかるからである。このような場合、微分アクションを適用して、測定値の変化スピードにより制御性能を改善する。

### 2.2.2 参照軌道に基づく PID チューニングパラメータの設定

提案する PID チューニングパラメータの最適デザイン手法を説明するために、次の無駄時間を含む 1 次遅れシステムを考える：

$$Y(s) = \frac{K e^{-Ls}}{\tau s + 1} U(s) \quad (2.3)$$

ここで、 $K$ 、 $\tau$ 、 $L$  はそれぞれプロセスゲイン、時定数、無駄時間である。式 (2.3) を離散時間  $\Delta t$  により離散化すると以下の式が得られる：

$$\begin{aligned} y(k+1) = & e^{-\frac{\Delta t}{\tau}} y(k) + K(1 - e^{-\frac{R_1}{\tau}} e^{-\frac{\Delta t}{\tau}}) u(k-R) \\ & + K e^{-\frac{\Delta t}{\tau}} (e^{\frac{R_1}{\tau}} - 1) u(k-R-1) \end{aligned} \quad (2.4)$$

ここで、 $R$  と  $R_1$  はそれぞれ  $L/\Delta t$  の整数部分とその余りの部分である。

本提案手法では、希望する PID コントローラの閉ループ制御応答を定義する参照軌道を PID チューニングパラメータのデザインに使用する。参照軌道  $y_r(t)$  はラプラス領域で次のローパスフィルタで定義する：

$$Y_r(s) = \frac{e^{-Ls}}{\tau_r s + 1} Y_{SP}(s) \quad (2.5)$$

ここで、 $\tau_r$  は参照軌道の時定数、 $L$  は 1 次遅れシステム式 (2.3) の無駄時間である。設定値変更は次の単位ステップ入力と与えられる：

$$Y_{SP}(s) = \frac{1}{s} \quad (2.6)$$

次式の離散化の式は式 (2.5) と (2.6) から与えられる：

$$y_r(k) = \begin{cases} 0 & (k \leq L) \\ 1 - e^{-\frac{(k-R)\Delta t - R_1}{\tau_r}} & (k > L). \end{cases} \quad (2.7)$$

PID コントローラの式 (2.2) は離散時間領域では次式になる：

$$\begin{aligned} u(k+1) = & u(k) + K_P \left( (e(k) - e(k-1)) + \frac{1}{T_I} e(k) \Delta t \right. \\ & \left. + T_D \frac{e(k) - 2e(k-1) + e(k-2)}{\Delta t} \right) \end{aligned} \quad (2.8)$$

PID チューニングパラメータの最適デザインアルゴリズムは式 (2.4)、(2.7)、(2.8) による以下の評価関数最小化による手法である:

$$\min_{K_P, T_I, T_D} J = \sum_{k=0}^m |y_r(k) - y(k)| \quad (2.9)$$

ここで、 $m$  は最適化する範囲を表すサンプル数である。

式 (2.9) は準ニュートン法により解く。PID チューニングパラメータはたった一つのパラメータ  $\tau_r$  により計算・最適化される。この最適化チューニング手法は式 (2.4)、(2.7)、(2.8) を変更することで、どんなプロセス、参照軌道、PID コントローラアルゴリズム (例えば、I-PD) にも対応可能である。既往の研究では、近似なしに最適化と参照軌道を組み合わせて PID チューニングパラメータをデザインしたものはない。なぜなら、既往の研究ではプロセスモデルと PID チューニングパラメータとの関係式を得るようなものばかりである。

## 2.3 デザイン例

### 2.3.1 1 次遅れ系システム

この節では、無駄時間を含む 1 次遅れシステム (First Order plus Dead Time, FOPDT) システムに対する PID チューニングパラメータ最適デザインアルゴリズムによるデザイン例を示す。この例に使用する FOPDT システムのモデルパラメータを表 2.2 に示す。

表 2.2: Model Parameters of the FOPDT system

Parameters	Value	unit
$K$	1.5	-
$\tau$	50	s
$L$	10	s

本提案手法により PID チューニングパラメータのデザインパラメータを  $\tau_r=20$  秒とした。2.2.2 節に記述したプロセッサに従い FOPDT システムに対する最適 PID チューニングパラメータを求めた。PI コントローラと PID コントローラに分けて計算を行い、それらの最適化結果をそれぞれ図 2.2 と 2.3 に示す。また、最適化された PID チューニングパラメータを表 2.3 にまとめた。

これらの図から、PI コントローラより PID コントローラの方が参照軌道に対して忠実な軌道を描いて制御をしていることが分かる。PI と PID コントローラでは比例ゲインは同じであるが、積分時間を微分時間にある程度振り替えることでより参照軌

道に近づけている。PI と PID コントローラでは参照軌道からの誤差はさほど見られないのにも関わらずプロセス入力の変動はPI コントローラの方がPID コントローラよりかなり速く、大きくなっている。

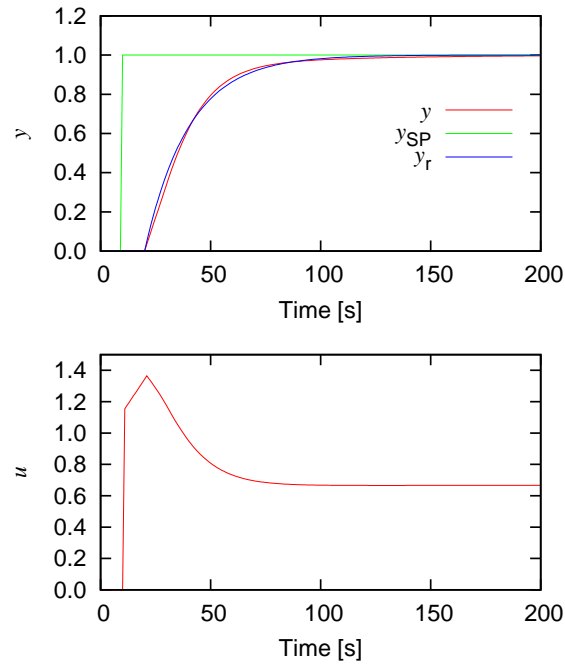


図 2.2: PI Controller Performance by Optimized Parameters

表 2.3: Optimized PID Tuning Parameters

Parameters	PI	PID	Unit
$K_P$	1.13	1.13	-
$T_I$	53.78	51.75	s
$T_D$	-	1.61	s

### 2.3.2 他の手法との比較

#### 既往のチューニング手法との比較

この節では本提案手法と以下にあげる既往の PID パラメータチューニング手法を表 2.2 の FOPDT システムを用いて比較を行う。

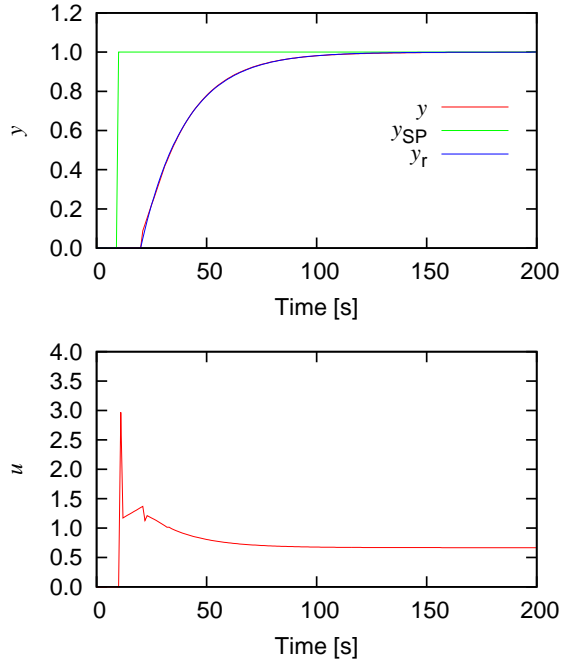


図 2.3: PID Controller Performance by Optimized Parameters

- Lee *et al.* method [12]

$$K_P = \frac{T_I}{K(\tau_r + L)}, T_I = \tau + \frac{L^2}{2(\tau_r + L)}, T_D = \frac{L^2}{6(\tau_r + L)} \left( 3 - \frac{L}{T_I} \right)$$

- CHR method [11]

$$K_P = 0.6 \frac{\tau}{KL}, T_I = \tau, T_D = 0.5L \quad (2.10)$$

CHR 法で得られた PID チューニングパラメータを表 2.4 に示す。式 (2.10) に示すように、CHR 法は  $\tau_r$  による調整パラメータを持っておらず、与えられたプロセス伝達関数により設定値変更追従に対する評価関数を最小化する形で PID チューニングパラメータが求まる。そこで、CHR 法の軌跡を基準として、その軌跡に一番近くなるような参照軌道の時定数  $\tau_r$  を求めた。その値は  $\tau_r=6$  秒である。この時定数をもとに比較を行った。デザインされた PID チューニングパラメータの比較を表 2.4 に、比較図を図 2.4 に示す。

この 3 つの手法の制御応答はほとんど同じであり、よい性能を示している。そこで、結果を比較するために、それぞれの結果に対して参照軌道とプロセス出力との間の積分 2 乗誤差 (ISE) を計算して、表 2.5 に示した。本提案手法が一番参照軌道にフィッティングされている。

次に、モデル誤差の効果を 3 つの手法で比較を行った。なぜなら、実プロセスは時変の非線形システムであるが、時不変の線形システムと仮定しても差し支えない誤差

であると言うときにその線形システム使って設計を行っているだけで、実際は設計に使ったモデルと実プロセスとはモデル誤差が存在し、制御性能はその設計に使ったモデルの精度に依存するからである。モデル誤差を表 2.2 に示した FOPDT システムのモデルパラメータすべてに  $\pm 10\%$  乗せたものを比較に使用した。それらのモデルパラメータを表 2.6 と 2.7 に示す。PID チューニングパラメータはすでに示した表 2.4 を使用した。モデル誤差による制御性能の比較を図 2.5 と 2.6 に示す。また、ISE の比較結果を表 2.8 を示す。それぞれの手法においてほとんど同じ制御性能が得られている。なぜなら、制御のアルゴリズムが PID コントローラであり、PID チューニングパラメータだけの違いでしかないからだと考える。しかしながら、本提案最適化手法の平均 ISE が 1 番よい結果となっている。もちろん、これは近似を何もせず参照軌道に最適に近づけるように最小化計算が行われているからである。近似を用いることはモデル誤差を持つことと同じで参照軌道に対する制御性が落ちることになる。

FOPDT システムにおける無駄時間と時定数の比  $L/\tau$  に対して Lee *et al.* の方法と提案手法の ISE により比較を行った結果を図 2.7 に示す。この図より、提案手法の方が Lee *et al.* の方法より  $L/\tau$  の全範囲で ISE が小さいことを示している。

表 2.4: Designed PID Tuning Parameters by each Approach

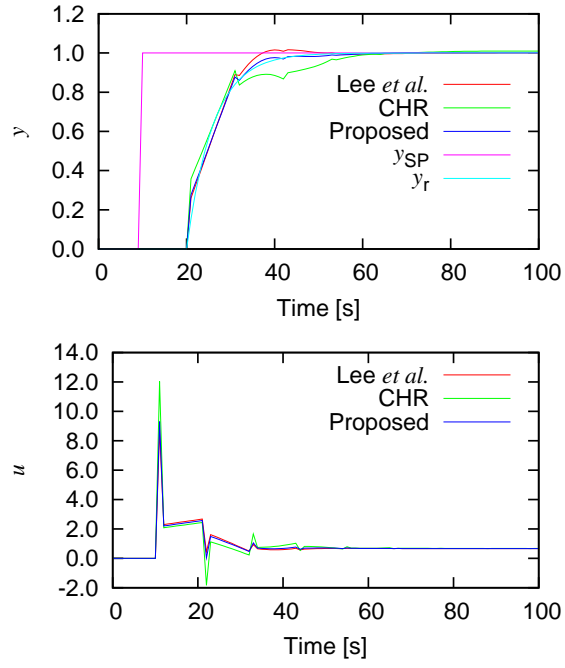
Parameters	Lee <i>et al.</i>	CHR	Proposed	Unit
$K_P$	2.21	2.00	2.12	-
$T_I$	53.13	50.00	52.24	s
$T_D$	2.93	5.00	3.36	s

表 2.5: ISE between reference trajectory and process output

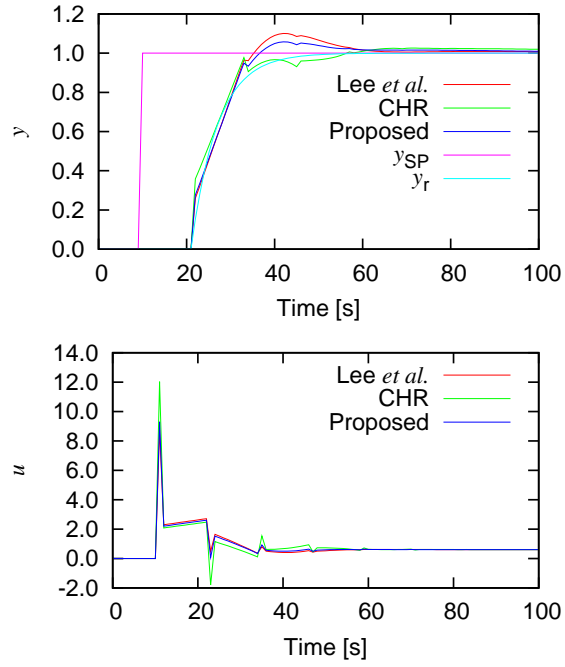
Lee <i>et al.</i>	CHR	Proposed
0.055	0.168	0.033

表 2.6: Model Parameters of the FOPDT system (+10%)

Parameters	Value	Unit
$K$	1.65	-
$\tau$	55	s
$L$	11	s



⊠ 2.4: Control Result by each Approach



⊠ 2.5: Control Performance with Model Mismatch (+10%)



表 2.7: Model Parameters of the FOPDT system(-10%)

Parameters	Value	Unit
$K$	1.35	-
$\tau$	45	s
$L$	9	s

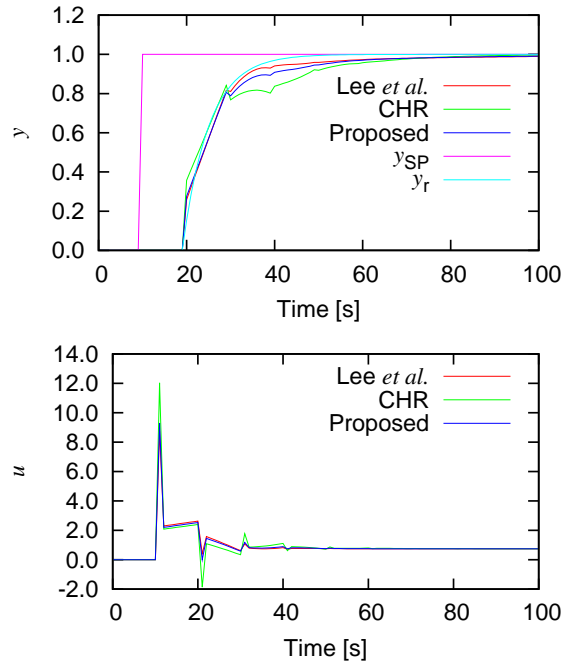


图 2.6: Control Performance with Model Mismatch (-10%)

表 2.8: ISE between reference trajectory and process output with model mismatch

	Lee <i>et al.</i>	CHR	Proposed
+10%	0.260	0.130	0.143
-10%	0.069	0.372	0.123
Average	0.165	0.251	0.133

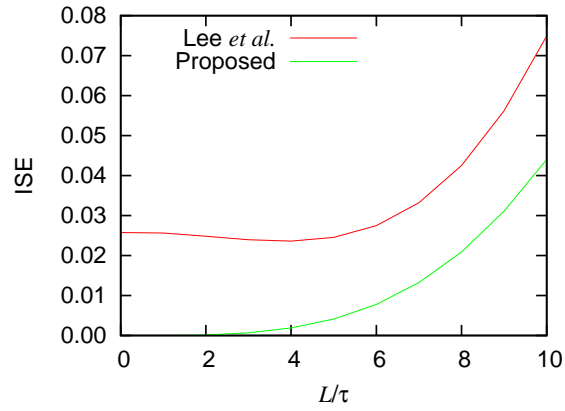


図 2.7: Comparison of the ISE generated by varying process dead time to time constant ratios.

#### モデル予測制御との性能比較

参照軌道を用いた既往の方法では、PID チューニングパラメータを求めるためにマクローリン展開のような近似を使用しており、提案手法の既往の研究からの進歩性は前節で示した。本手法により PID コントローラの制御性能はモデル予測制御 (MPC) の制御性能に近づいたと考える。そこで、2.3.2 節のモデル誤差を用いてモデル予測制御と本手法で求めたチューニングパラメータを用いた PID コントローラとの比較を行った。この比較に用いたモデル予測制御は Predictive Function Control (PFC) [8] である。この PFC は 1 ステップ予測とローパスフィルタを用いた参照軌道によるモデル予測制御である。PFC でモデル誤差のない FOPDT システムを制御した場合、PFC はプロセス出力と参照軌道との誤差がゼロとなるアルゴリズムである。それ故、本ケーススタディではモデル誤差を適用した。結果を表 2.9 と図 2.8 に示す。この結果は本手法により求めた PID パラメータは PID コントローラとして PFC の制御性能とほぼ同等の性能を示している。

表 2.9: ISE between the reference trajectory and process output with model mismatch for PFC

	PFC	Proposed
+10%	0.104	0.143
-10%	0.103	0.123

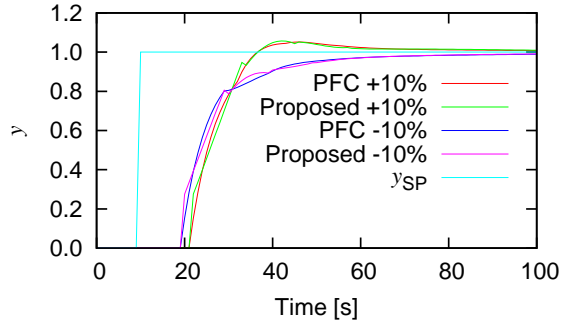


図 2.8: Comparison of control performance between PFC and the PID controller using the proposed tuning method.

## 2.4 提案手法の応用

### 2.4.1 積分系システム

多くのプロセス応答はFOPDTシステムとして近似することができる。しかしながら、式 (2.11) で表される積分システムはまったく違うプロセスなのでFOPDTシステムとして近似することは不可能である。そこで、この節では提案手法を改良して積分システムのPIDチューニングパラメータデザイン方法を提案する。

$$G_P(s) = \frac{Ke^{-Ls}}{s} \quad (2.11)$$

積分システムはドラムの液面に代表されるシステムであり、その主な制御目的は今まで取り扱ってきた設定値に対する追従性ではなく、外乱に対する補償動作である。積分システムでは以下のようにアルゴリズムの改良を行う。

まず、積分システムの外乱は以下のように表される：

$$D(s) = \frac{K_d}{s} \quad (2.12)$$

ここで、 $K_d$  は積分システムの傾きである。

次に、別なPIDチューニングパラメータデザイン手法が必要であるので、外乱補償に対して次の式 (2.13) の閉ループ応答を考える：

$$Y(s) = \frac{1}{1 + G_C(s)G_P(s)} D(s) \quad (2.13)$$

式 (2.11) の無駄時間を無視すると、式 (2.13) は式 (2.2)、(2.11)、(2.12) から次式に書き換えられる：

$$Y(s) = \frac{K_d T_I}{(1 + K_P K T_D) T_I s^2 + K_P K T_I s + K_P K} \quad (2.14)$$

式 (2.14) は 2 次遅れシステムである。従って、ローパスフィルター式 (2.5) の参照軌道に代え、次のような振動なしの 2 次遅れのインパルス応答として新しい参照軌道を定義して、外乱補償における積分システムに対する PID チューニングパラメータのデザイン方法に適用する:

$$Y_r(s) = \frac{K_d \tau_r^2}{(\tau_r s + 1)^2} \quad (2.15)$$

ここで、積分システムに対する PID コントローラの閉ループ応答は式 (2.14) で表され、その望まれる閉ループ応答は式 (2.15) である。これらの 2 式は  $K_d$  が共通して存在する。従って、PID チューニングパラメータは  $K_d$  には依存しないことが分かる。すなわち、積分システムは  $K_d$  なしの式 (2.11) と (2.12) から次式の離散化システムとなる:

$$\begin{aligned} y(k+1) &= y(k) + K(\Delta t - R_1)u(k-R) \\ &\quad + KR_1u(k-R-1) + \Delta t \end{aligned} \quad (2.16)$$

また、新しい参照軌道の式 (2.15) は離散時間領域では次式となる:

$$y_r(k) = (k\Delta t)e^{-\frac{k\Delta t}{\tau_r}} \quad (2.17)$$

積分システムの PID チューニングパラメータは 1 つのパラメータ  $\tau_r$  により式 (2.8)、(2.9)、(2.16)、(2.17) により定義された最適化問題を解くことで得られる。

このアルゴリズムを使ったデザイン例を示すために表 2.10 に示す積分システムのモデルパラメータを使用して、デザインパラメータ  $\tau_r=120$  秒で PID チューニングパラメータを求めた。

表 2.10: Model Parameters of Ramp System

Parameters	Value	Unit
$K$	0.1	1/s
$L$	10	s

最適化されたデザイン結果を表 2.11 と図 2.9 に示す。新しく定義した参照軌道に対して PI と PID コントローラの両方で積分システムを制御するのに十分な性能が得られた。

## 2.4.2 I-PD コントローラ

PID コントローラは設定値変更により急激なプロセス入力の変化を引き起こす場合がある。なぜなら設定値と測定値との偏差が直接プロセス入力に影響するからである。このプロセス入力の変化はプロセスプラントの運転において問題を起こす可能性がある

表 2.11: Optimized PID Tuning Parameters for the Ramp System

Parameters	PI	PID	Unit
$K_P$	0.17	0.17	-
$T_I$	259.2	249.0	s
$T_D$	-	10.27	s

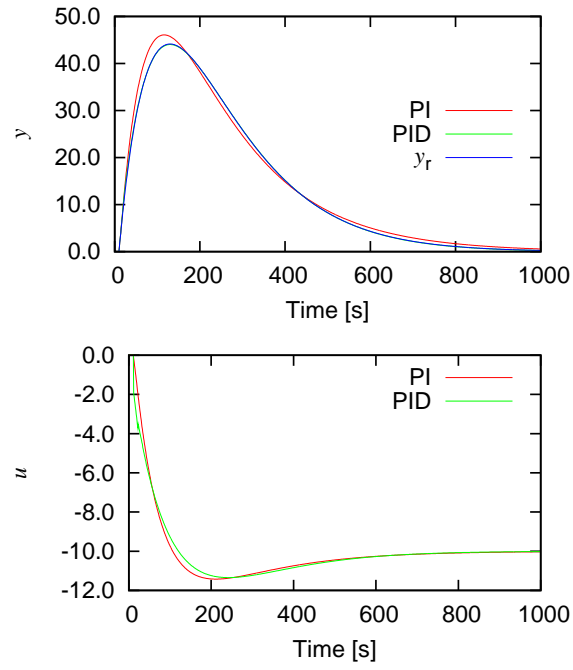


图 2.9: Control Performance with Ramp System

る。次式と図 2.10 で与えられる I-PD コントローラはそのような急激なプロセス入力の変化を防ぐことができる。

$$U(s) = K_P \left\{ -Y(s) + \frac{1}{T_I s} E(s) - T_D s Y(s) \right\} \quad (2.18)$$

式 (2.18) は離散時間領域では以下の式となる：

$$\begin{aligned} u(k+1) = & u(k) + K_P \left\{ -y(k) + y(k-1) \right. \\ & + \frac{1}{T_I} e(k) \Delta t \\ & \left. - T_D \frac{y(k) - 2y(k-1) + y(k-2)}{\Delta t} \right\} \end{aligned} \quad (2.19)$$

この場合、式 (2.19) は式 (2.8) に代え I-PD コントローラの最適な PID チューニングパラメータを得るために使用される。

表 2.2 の FOPDT モデルパラメータと  $\tau_r=20$  秒によって求めた I-PD コントローラの PID チューニングパラメータを表 2.12 にまとめ、制御性能を図 2.11 に示す。

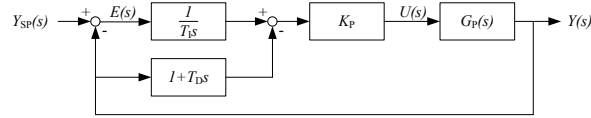


図 2.10: Block Diagram for I-PD Control

表 2.12: Optimized PID tuning Parameters for I-PD Controller

Parameters	PI	PID	Unit
$K_P$	2.97	4.29	-
$T_I$	28.28	27.16	s
$T_D$	-	4.22	s

これらの最適化計算結果は制御応答が振動しており、あまりよくない性能となっている。そこで、I-PD コントローラの PID チューニングパラメータをデザインするためにはこの振動を抑えるための新たな改良が必要である。従って、I-PD コントローラの PID チューニングパラメータを最適化するための次に示す新しい参照起動を考える：

$$Y_r(s) = \frac{e^{-Ls}}{(\tau_r s + 1)^n} Y_{SP}(s) \quad (2.20)$$

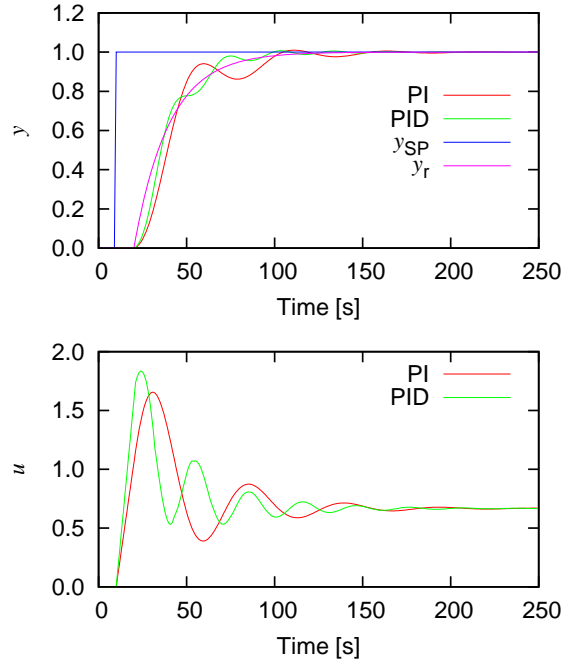


図 2.11: Control Performance for I-PD Controller

ここで、 $n$  はローパスフィルタの次数である。式 (2.20) は離散時間領域で次式のように表される:

$$y_r(k) = \begin{cases} 0 & (k \leq L) \\ 1 - e^{-\frac{(k-R)\Delta t - R_1}{\tau_r}} \sum_{i=1}^n \frac{((k-R)\Delta t - R_1)^{i-1}}{\tau_r^{i-1}(i-1)!} & (k > L) \end{cases} \quad (2.21)$$

PID チューニングパラメータの最適化を行うための参照軌道は式 (2.7) から新たな式 (2.21) に変更した。 $\tau_r=11$  秒と  $n=2$  の新たな参照軌道を用いた I-PD コントローラの PID チューニングパラメータデザイン結果を表 2.13 と図 2.12 に示す。このデザインに使われた参照軌道の  $\tau_r=11$  秒と  $n=2$  は  $\tau_r=20$  秒と  $n=1$  の参照軌道と同じような軌道となるように決定した。結果として、参照軌道の最初の部分の傾きが比較的ゆっくりになっているために振動が抑えられた。

#### 2.4.3 制約条件を考慮した PID パラメータチューニング

本提案手法では参照軌道に沿う PID チューニングパラメータを求める非線形最適化を実行する際にプロセス出力  $y$  とプロセス入力  $u$  とプロセス入力の变化速度  $\Delta u$  の制約条件を考慮することが可能である。本節では提案手法による制約条件を考慮したス

表 2.13: Optimized PID Tuning Parameters for I-PD Controller (n=2)

Parameters	PI	PID	Unit
$K_P$	2.48	3.40	-
$T_I$	27.45	27.23	s
$T_D$	-	4.40	s

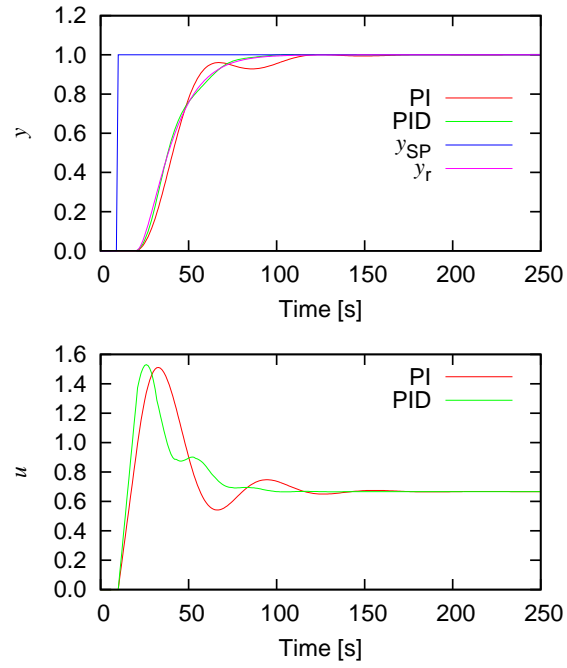


图 2.12: Control Performance for I-PD Controller by New Reference Trajectory (n=2)



タディを示す。最初のスタディでは、PID チューニングパラメータを求める式 (2.9) において以下の制約条件を考慮する:

$$\left. \begin{array}{l} u(k) < u_{\max} \\ |\Delta u(k)| < \Delta u_{\max} \end{array} \right\} (k = 1, 2, \dots, m) \quad (2.22)$$

ここで、 $u_{\max}$  と  $\Delta u_{\max}$  はそれぞれ  $u(k)$  と  $\Delta u(k)$  の最大値である。表 2.2 の FOPDT モデルパラメータと参照軌道  $\tau_r=6$  秒をもちいて制約条件を考慮した PID チューニングパラメータを求めるスタディを行った結果を表 2.14 と図 2.13 に示す。Case No. 1 はモデル誤差、制約条件なしでの 2.3.2 節と同じスタディ結果である。Case No. 2-4 は  $\Delta u(k)$  の制約条件の変化速度に制限を徐々に加えていった結果で、微分時間が減少していき、PI コントローラに近づいている。Case No. 1-3 では、 $u(k)$  と  $\Delta u(k)$  は制約条件として同じ値になっている。従って、最初の立ち上がり  $u(R+1) = 0 + \Delta u(R+1)$  となっていることがわかる。Case No. 4 では、微分時間がゼロになり、最初の立ち上がりの  $\Delta u(R+1)$  が大きくなることを防いでおり、 $u(R+2) > u(R+1)$  となっている。Case No. 5 では  $u(k)$  が制約にかかることで、再び微分時間が現れ、他のケースよりオーバーシュートを押さえることができている。本手法では、制約条件を考慮して PID チューニングパラメータを簡単に求めることができる。

表 2.14: PID control performance by tuning parameters with constraints

Case No.	1	2	3	4	5
Parameters					
$K_P$	2.12	2.09	2.06	1.97	1.76
$T_I$	52.24	58.39	61.49	60.46	57.63
$T_D$	3.36	1.38	0.44	0.00	0.12
$u_{\max}$	10.00	10.00	10.00	10.00	2.10
$u$	9.30	5.00	3.00	2.33	2.10
$\Delta u_{\max}$	10.00	5.00	3.00	2.00	2.00
$\Delta u$	9.30	5.00	3.00	2.00	2.00

さらに、制約を考慮して PID チューニングパラメータを求めるための詳細な検討を行うために図 2.14 に示すドラムの液面システムを用いたスタディを行った。このシステムは一般的な積分プロセスである。このシステムの制御目的は上流からの流量外乱を下流に対して緩和することである。通常は、PI コントローラがこのシステムには適用される。なぜなら、このドラムの役割は下流への流量外乱を最小化することであり、微分時間を使うと急激に下流への流量を操作することになるからである。この PI チューニングパラメータは以下のような設計要求がある:

- (1) ハードの制約として下流への流量の最大値/最小値

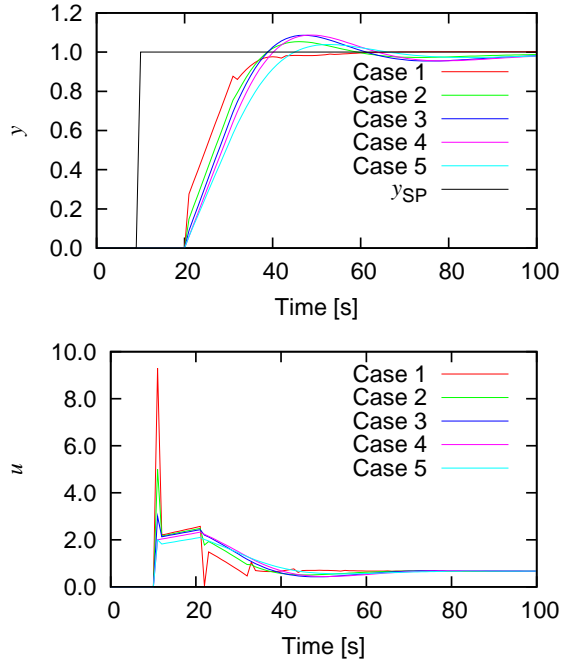


図 2.13: PID control performance by tuning parameters with constraints

- (2) ソフトの制約としてドラム内の液があふれ出すドラム液面の最大値/ドラム内の液がなくなりポンプの空引きによりポンプの破損を防ぐためのドラム液面の最小値
- (3) アイテム (1) と (2) を守った上で下流への流量外乱を緩和するための流量変化速度の最小化

ここで、ドラムの断面積を  $A=2 \text{ m}^2$ 、ドラムの使用できる液面を  $2 \text{ m}$ 、定常状態での液面を  $1 \text{ m}$  (50 %) および上流からの流量/下流への流量を  $0.02 \text{ m}^3/\text{s}$  とする。外乱としての上流からの最大流量を  $0.04 \text{ m}^3/\text{秒}$ 、下流への最大流量を  $0.042 \text{ m}^3/\text{秒}$ 、液面の最大制約条件を  $1.8 \text{ m}$  (90 %) と設定する。本スタディでは、下流への最大流量と液面の最大制約値を守り、下流への流量変化速度が最小になるように PI チューニングパラメータを求める。液面プロセスでは余り大きな無駄時間は現れてこない。無駄時間が存在したとしても液ホールドアップ時間の約 10% である [22]。そこで、このスタディでは無駄時間  $L = 20 \text{ 秒}$  を用いた。この積分プロセスでは、式 (2.11) の  $K$  と式 (2.12) の  $K_d$  はそれぞれ  $-\frac{1}{A}=-0.5 \text{ 秒}^{-1}$  と  $\frac{1}{A}=0.5 \text{ 秒}^{-1}$  に相当する。

この制約条件をハンドリングするスタディを行うために、式 (2.12) の外乱に対する単位ステップ応答に代えて次式の実ステップ応答を用いて式 (2.16) と (2.17) を修正する:

$$D(s) = \frac{K_d}{s} \frac{\Delta d_{\max}}{s}. \quad (2.23)$$

上式において、 $\Delta d_{\max}$  は外乱の最大変化量であり、このケーススタディでは  $0.02 \text{ m}^3/\text{秒}$  となる。よって、式 (2.16) と (2.17) は次式に書き換えられる：

$$y(k+1) = y(k) + K(\Delta t - R_1)u(k-R) + KR_1u(k-R-1) + K_d\Delta t\Delta d_{\max} \quad (2.24)$$

$$y_r(k) = K_d\Delta d_{\max}(k\Delta t)e^{-\frac{k\Delta t}{\tau_r}} \quad (2.25)$$

参照軌道の時定数  $\tau_r$  は前もってプロセス出力最大値  $y_{\max}$  から求めることができる。式 (2.25) の最大値が  $y_{\max}$  である。従って、 $\tau_r$  は次式を使うことで計算できる：

$$\tau_r = \frac{y_{\max}e}{K_d\Delta d_{\max}}. \quad (2.26)$$

このケーススタディでは、 $y_{\max}=1.8 \text{ m}$  (90 %) から  $\tau_r=217.5 \text{ 秒}$  を PI パラメータを最適化するために使用する。

本ケーススタディの結果を表 2.15 にまとめ、制御性能を図 2.15 に示す。それぞれのケーススタディを以下に説明する。最初、Case P1 は制約条件なしで本提案手法により PI チューニングパラメータを最適化した。このケースでは、液面の最大制約と下流への流量最大値を超えている。次の Case P2 では、液面の最大制約と下流への流量最大値を使い、PI チューニングパラメータを求めた。下流への流量が制約条件になっている。液面は最大制約内に収まっている。Case P2 の下流への流量変化速度は Case P1 より大きくなっている。安定的な運転のためには下流への流量に余裕が必要であるので、Case P3 では下流への流量の最大値を 0.041 とし、PI チューニングパラメータを計算した。下流への流量に余裕をとり 0.041 にすることができたが、Case P3 では下流への流量変化速度が Case P2 に比べ急激に大きくなっている。そこで、Case P4 では液面の最大値に余裕をとり、PI チューニングパラメータを求めた。結果として、Case P4 では Case P3 に比べて下流への流量変化速度を押さえ、下流への外乱を少なくすることができた。このスタディでは制約条件を考慮した PI チューニングパラメータを簡単に設計することができることを示した。

## 2.5 ダイナミックシミュレーションと DCS による PID チューニング

この節では、ダイナミックシミュレーションソフトウェアに常圧蒸留塔 (Crude Distillation Unit, CDU) のプレフラッシュドラムのモデルを構築して、プロセス産業で実績のある 2 つの DCS (Distributed Control System) の PID コントローラ機能を使い、提案手法の有用性を示す。DCS はインベンシス製の Foxboro I/A Series System と横河電機製の Centum VP である。このシミュレーションの目的は以下のとおりである：

- 第一原理を使ったダイナミックシミュレーションによる制御性確認

表 2.15: PI tuning parameters for Level Control with constraints

Parameters \ Case No.	P1	P2	P3	P4
$K_P$	-0.0192	-0.0236	-0.0344	-0.0248
$T_I$	473.0	630.6	1040.0	607.8
$y_{\max}$	-	1.800	1.800	1.700
$y$	1.839	1.730	1.548	1.700
$u_{\max}$	-	0.042	0.041	0.042
$u$	0.04286	0.042	0.041	0.042
$\Delta u$	0.000201	0.000244	0.000351	0.000256

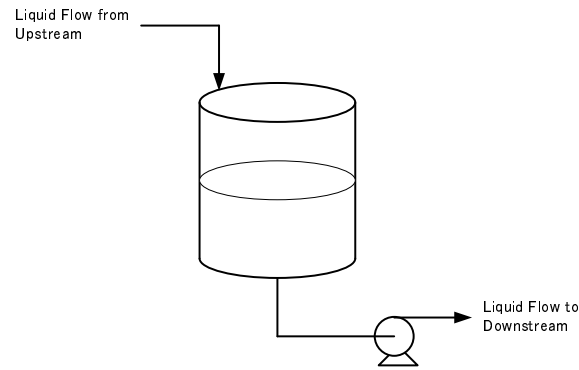


图 2.14: Process Scheme of a Drum Level Control System

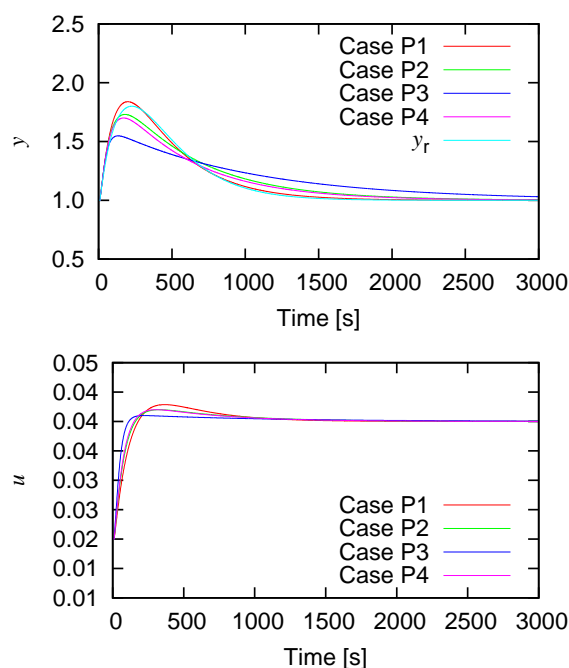


図 2.15: Practical Example of Level Control Tuning with constraints

- PID コントローラと I-PD コントローラのイラストレーション
- 実 DCS への適用事例

Foxboro I/A Series System は I-PD コントローラのオプションを持っていない、そこで、そのオプションを持っている Centum VP により I-PD コントローラのシミュレーションを行った。ダイナミックシミュレーションソフトウェアパッケージ:Invensys 製 Dynsim にシミュレータを構築した。Dynsim と Foxboro I/A Series System の通信は両方とも Invensys 製なので独自の通信プロトコルで行っている。このシステムは通常 Operator Training Simulator(OTS) としてプラント運転員の運転トレーニングに使用されている。Dynsim と Centum VP との通信は OPC(OLE for process control) インターフェースを用いて確立している。

### 2.5.1 シミュレーションモデル

この節で行われるシミュレーションは図 2.16 に示す CDU のプレフラッシュドラムを用いて実行される。CDU はその装置構成により 3 つのタイプがある。その中で、1 つの蒸留塔で低圧タイプの場合プレフラッシュドラムを持つ。このシミュレーションはそのプレフラッシュドラムで行う。CDU で、原油は脱塩ドラムにフィードされ、最

初に原油の塩分を取り除く。その後、脱塩原油は CDU 残渣油の熱交換器により加熱された後、プレフラッシュドラムにフィードされ、フラッシュ蒸留させ脱塩原油から軽い成分を取り除く。これは CDU 加熱炉の前に据え付けられた原油フィードコントロール弁での原油のフラッシュを防ぐためである。プレフラッシュドラムまわりの PID コントローラを図 2.16 に示す。プレフラッシュドラムのフィード流量は温度コントローラ (TC1)、圧力コントローラ (PC1) と液面コントローラ (LC1) に干渉している。コントロールバルブ特性は非線形を持たせるためにイコールパーセントを使用している。PID コントローラの設定値初期値、エンジニアリングレンジ、単位を表 2.16 に示す。

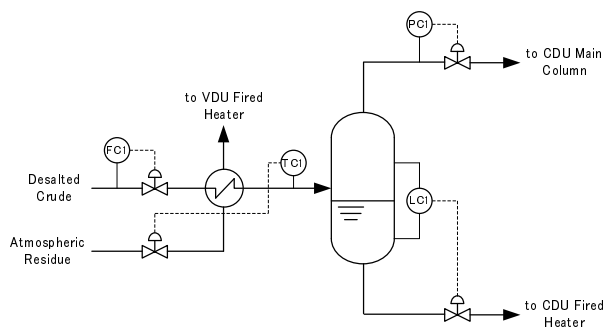


図 2.16: CDU Preflash Drum

表 2.16: Initial Conditions, Range and Unit of Controllers

Controllers	Initial Value	Range	Unit
FC1	700	0-1200	m <sup>3</sup> /h
TC1	200	0-300	°C
PC1	0.5	0-0.8	MPaG
LC1	50	0-100	%

## 2.5.2 モデル同定

PID コントローラの最適なチューニングパラメータ決定のために、プロセスモデルの同定が必要である。実ステップ応答データを得るために図 2.17、2.18、2.19、2.20 に示すようにそれぞれのコントロールバルブに対してステップ変更を加え、モデル同定のためのステップ応答テストを行った。ステップ応答テストデータをもとに次式の評価関数により最小化計算を行うことで FOPDT または積分システムのモデルパラメー

タを求めた。詳細は第 3 章に示す。

$$\min_{\text{model parameters}} J = \sum_{k=0}^{nd} \{y_a(k) - y(k)\}^2 \quad (2.27)$$

ここで、 $nd$  は実運転データの数である。

FC1、TC1、PC1、LC1 のモデル同定結果をそれぞれ図 2.17、2.18、2.19、2.20 に示す。モデルパラメータを表 2.17 にまとめた。流量、温度、圧力、液面のプロセスを FOPDT と積分システムとしてモデル同定を行ったが、同定の結果からそれぞれのプロセスを制御するのに十分な精度が得られたと考える。積分システムに関しては、その傾きの類似性が重要であり、これを満たした精度がある。

表 2.17: Identified Model Parameters for CDU Preflash Drum

Parameters \ Loop Name	FC1 Eq. (2.3)	TC1 Eq. (2.3)	PC1 Eq. (2.3)	LC1 Eq. (2.11)	Unit
$K$	31.74 m <sup>3</sup> /h %	0.6619 °C/%	-0.0118 MPaG/%	-0.0446 %/%	-
	2.65 %/%	0.221 %/%	-1.48 %/%	-	-
$\tau$	2.35	82.18	24.57	-	s
$L$	3.15	13.18	2.54	4.49	s

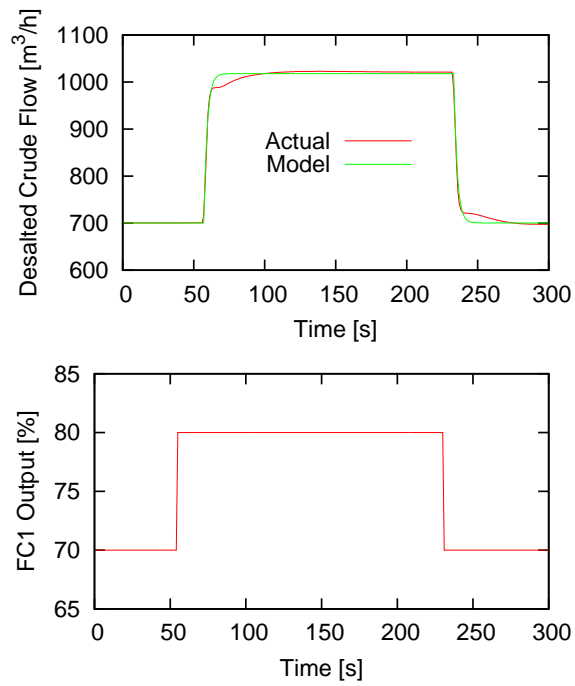
### 2.5.3 DCS における PID パラメータデザイン

2 つのプロセス産業で実績のある DCS(Foxboro I/A Series System と Centum VP) を使い、ダイナミックシミュレーションを用いて、より実運転に近い状態で PID コントローラの制御性能改善スタディを行った。これらの DCS の PID コントローラは比例帯を使っている。比例帯と比例ゲインとの関係は以下の式 (2.28) で与えられる:

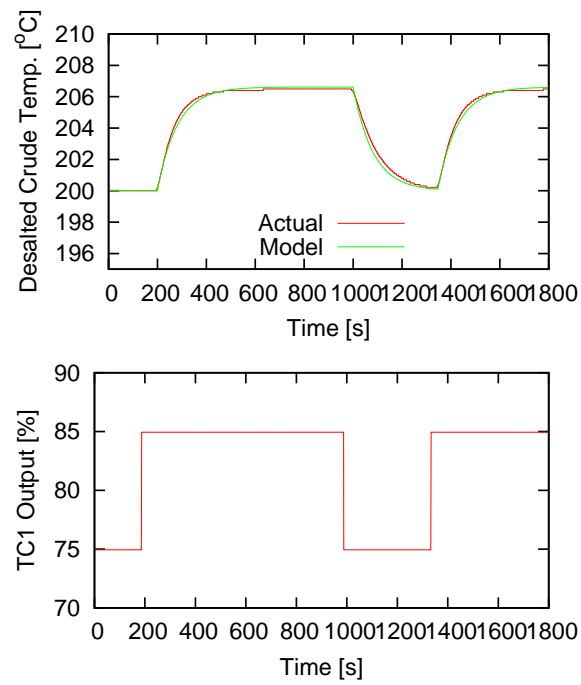
$$PB = \frac{100}{K_P} \quad (2.28)$$

標準的な PID と I-PD コントローラはそれぞれ Foxboro I/A Series System と Centum VP で実行する。2 つの DCS のスタディの最初のシミュレーション部分では表 2.1 をベースとした表 2.18 の初期 PID チューニングパラメータを使用している。

これらの初期パラメータはよりよい制御性能が得られるように典型的な PID チューニングパラメータからチューニングされる。流量と液面制御は通常、微分項を使用しないことから微分時間を 0 とした。

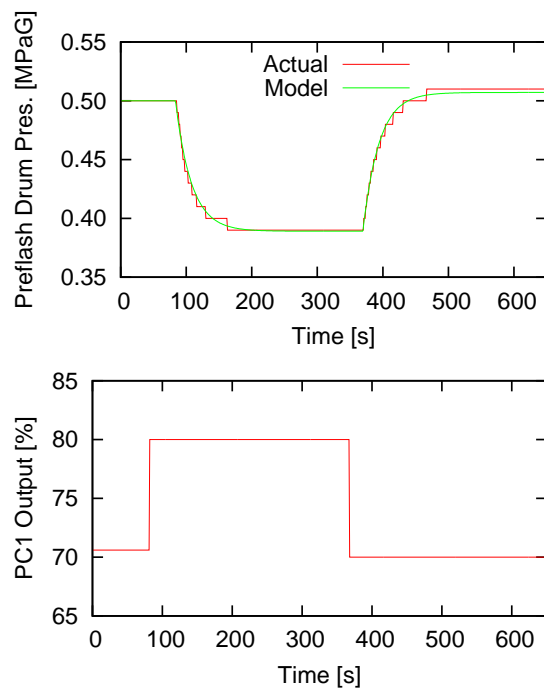


☒ 2.17: Identification Results of FC1

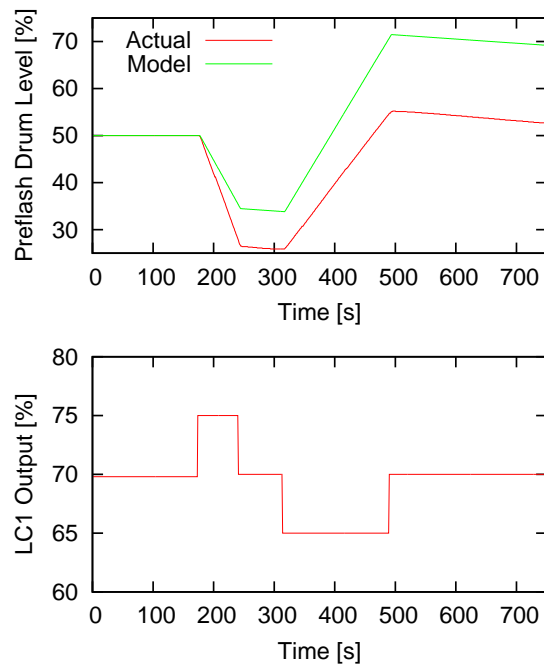


☒ 2.18: Identification Results of TC1





☒ 2.19: Identification Results of PC1



☒ 2.20: Identification Results of LC1

表 2.18: The initial PID Tuning Parameters for two Illustrations

Parameters	Loop Name	FC1	TC1	PC1	LC1	Unit
$PB$		300	100	50	200	%
$T_I$		60	82	60	120	s
$T_D$		-	-	-	-	s

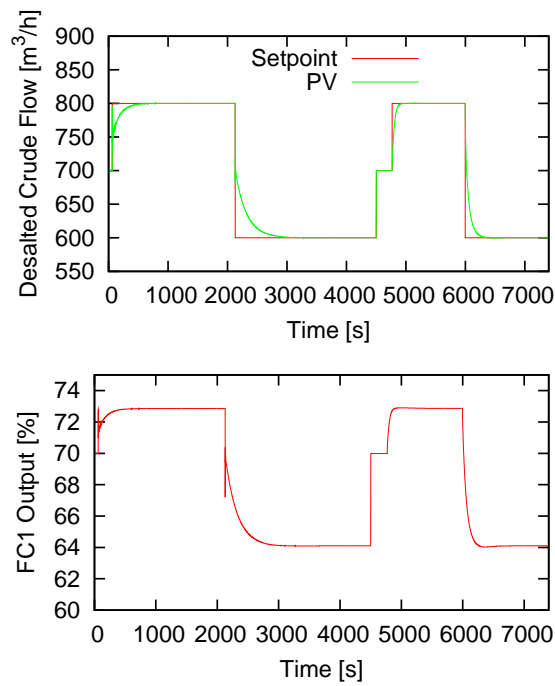
### Invensys Foxboro I/S Series System による PID コントローラデザイン

最初のスタディでは Foxboro I/A Series System の標準的な PID コントローラの式 (2.1) を使用した。表 2.17 のモデルパラメータをもとに PID コントローラの最適な PID チューニングパラメータを求めた。それらを表 2.19 にまとめた。

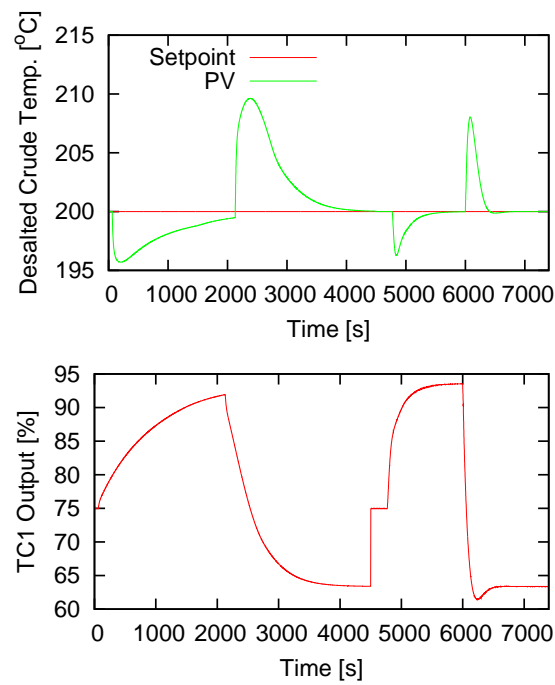
表 2.19: Optimized PID Tuning Parameters for PID Controller in Foxboro I/A Series System

Parameters	Loop Name	FC1	TC1	PC1	LC1	Unit
$\tau_r$		20	60	20	120	s
$PB$		2914	19.6	140	263	%
$T_I$		2.15	83.17	24.28	249.0	s
$T_D$		0	1.12	0.19	0	s

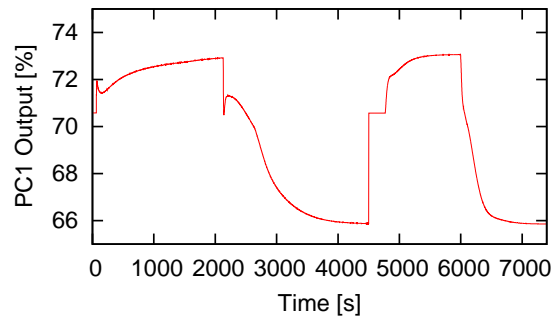
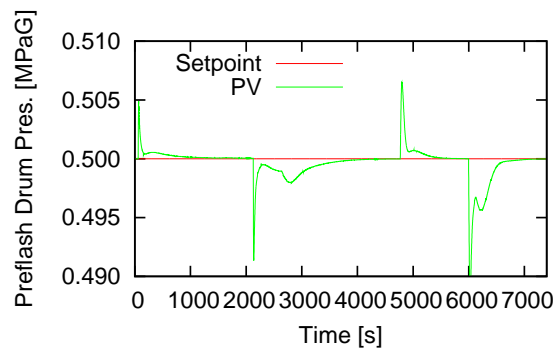
Foxboro I/A Series System の FC1、TC1、PC1、LC1 による制御シミュレーションの結果をそれぞれ図 2.21、2.22、2.23、2.24 に示す。これらの図のシミュレーション前半部分 0-4500 秒は表 2.18 のデフォルト PID チューニングパラメータを使用している。シミュレーション後半部分 4500-7400 秒は表 2.19 の最適化された PID チューニングパラメータを使用してシミュレーションを行っている。このシミュレーションでは FC1 の設定値をステップ変更することでプロセスに外乱を与えた。最適なデザインパラメータを適用した後、FC1 の設定値変更に対する急激なコントロール弁の動きが無くなったにもかかわらず、流量は速く、滑らかに設定値に達している。TC1 の外乱補償制御はよりよい制御性能が得られている。PC1 の制御応答はゆっくりになっている。これは比例帯が大きくなったことによる。しかしながら、PC1 の出力は滑らかになっている。LC1 は参照軌道にそって制御が実行できている。これは参照軌道の時定



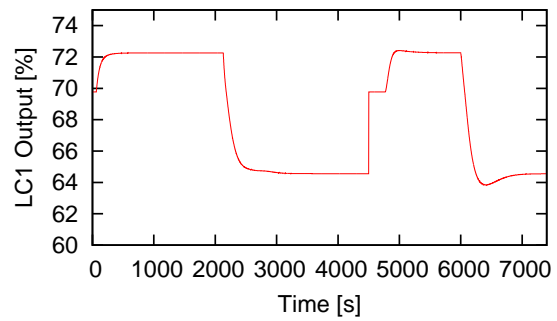
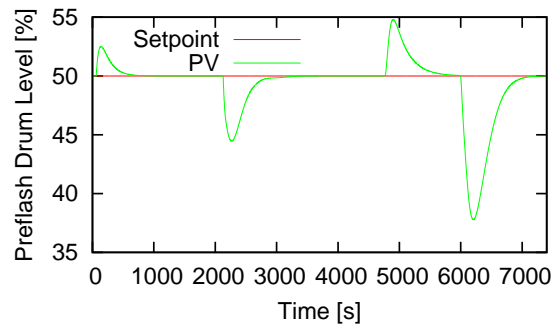
2.21: FC1 by Standard PID Controller



2.22: TC1 by Standard PID Controller



2.23: PC1 by Standard PID Controller



2.24: LC1 by Standard PID Controller

数をチューニングすることで下流への外乱が調整できることを意味している。すべての制御性能は参照軌道に従って制御できていることが確認できた。

## 横河 DCS(Centum VP) における I-PD コントローラデザイン

次のスタディでは Centum VP の I-PD コントローラの式 (2.18) を使用した。表 2.17 のモデルパラメータをもとに I-PD コントローラの最適な PID チューニングパラメータを求めた。それらを表 2.20 にまとめた。

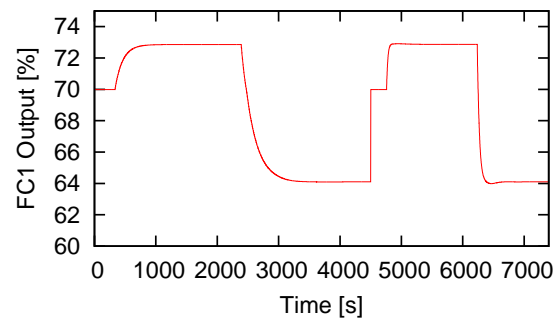
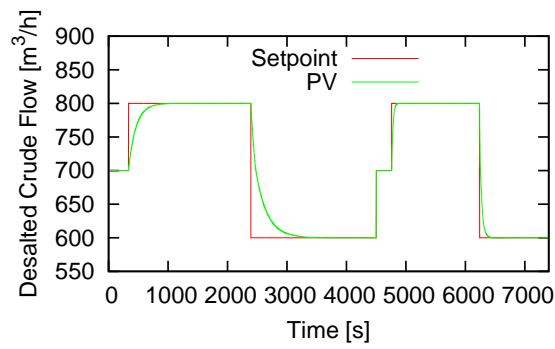
表 2.20: Optimized PID Tuning Parameters for I-PD Controller in Centum VP

Parameters	Loop Name					Unit
		FC1	TC1	PC1	LC1	
$n$		1	2	2	2	-
$\tau_r$		20	60	20	120	s
$PB$		496	14.2	126	262	%
$T_I$		8.28	81.34	23.17	249	s
$T_D$		0	4.75	0	0	s

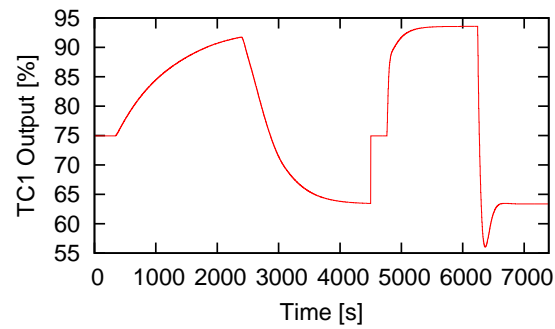
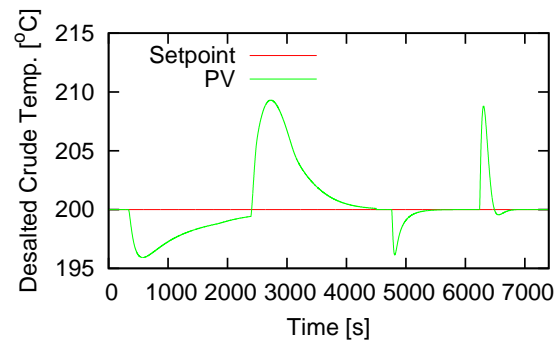
Centum VP の FC1、TC1、PC1、LC1 による制御シミュレーションの結果をそれぞれ図 2.25、2.26、2.27、2.28 に示す。これらの図のシミュレーション前半部分 0-4500 秒で、表 2.18 は I-PD コントローラのためのチューニングパラメータではないが、これらをデフォルト PID チューニングパラメータとして使用している。シミュレーション後半部分 4500-7400 秒は表 2.20 の最適化された PID チューニングパラメータを使用してシミュレーションを行っている。このシミュレーションでは FC1 の設定値をステップ変更することでプロセスに外乱を与えた。最適なデザインパラメータを適用した後、FC1 と TC1 はよりよい制御性能が得られた。PC1 のプロセス出力のピークは最適なパラメータよりデフォルトのパラメータの方がよい。しかしながら、外乱補償という観点からは最適なパラメータの方が速く補償できている。LC1 は参照軌道にそって制御が実行できている。すべての制御性能は参照軌道に従って標準的な PID コントローラの時と同様に制御できていることが確認できた。

## 2.6 結言

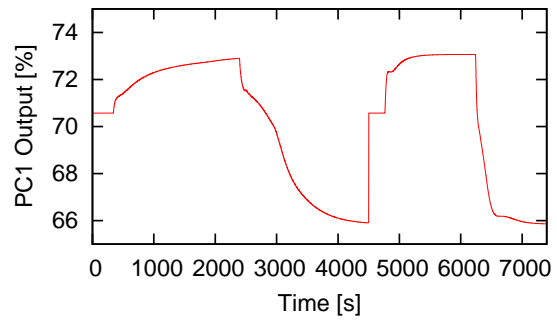
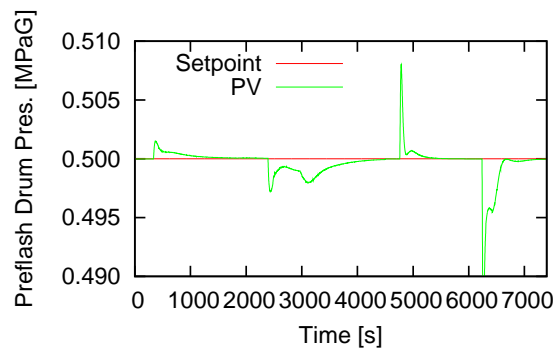
本研究では参照軌道に基づく最適な PID チューニングパラメータの決定方法を提案した。この提案手法では無駄時間を含む 1 次遅れシステムだけではなく様々なシステムに適用できることを示した。ひとたびモデルが同定されたなら、プロセスプラント



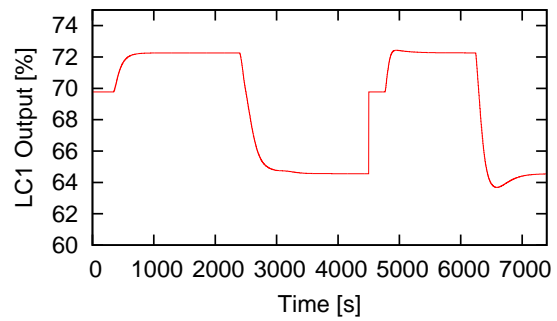
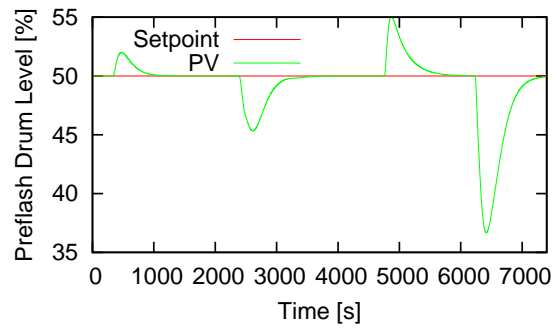
2.25: FC1 by I-PD Controller



2.26: TC1 by I-PD Controller



☒ 2.27: PC1 by I-PD Controller



☒ 2.28: LC1 by I-PD Controller

の操作性を改善するための PID パラメータをチューニングすることは提案手法により容易に実現できる。特に、制約条件を考慮し PID チューニングパラメータを求めることができる点が大きな進捗点であると考え。この手法はプラント運転の安全性、安定性、高い効率を実現するために有用で、役立つ手法である。



## 使用記号

$A$	= cross-section area	[m <sup>2</sup> ]
$E(s)$	= error between a measurement and a setpoint in the Laplace transform	[-]
$e(k)$	= error between setpoint and process output	[-]
$e_n(t)$	= normalized error based on engineering range	[%]
$D(s)$	= disturbance in the Laplace transform	[-]
$d(k)$	= disturbance	[-]
$G_C(s)$	= controller transfer function	[-]
$G_{DCLR}(s)$	= closed loop response transfer function for disturbance	[-]
$G_P(s)$	= process transfer function	[-]
$G_{PCLR}(s)$	= closed loop response transfer function for process	[-]
$J$	= performance index	[-]
$K$	= process gain	[-]
$k$	= discrete time by sampling time	[-]
$K_d$	= slope of ramp system	[-]
$K_P$	= proportional gain of PID controller	[-]
$L$	= dead time	[s]
$m$	= finite number of optimization	[-]
$n$	= order of low pass filter	[-]
$nd$	= number of actual data	[-]
$PB$	= proportional band of PID controller	[%]
$R$	= integer part of $L/\Delta t$	[-]
$R_1$	= remaining part of $L/\Delta t$	[s]
$s$	= complex variable in the Laplace transform	[-]
$T_D$	= derivative time of PID controller	[s]
$T_I$	= integral time of PID controller	[s]
$U(s)$	= process input in the Laplace transform	[-]
$u(k)$	= process input	[-]
$u_n(t)$	= normalized process input	[%]
$Y(s)$	= process output in the Laplace transform	[-]
$y(k)$	= process output	[-]
$y_a(k)$	= actual process output	[-]
$Y_{SP}(s)$	= setpoint of process output in the Laplace transform	[-]
$y_{SP}(k)$	= setpoint of process output	[-]
$Y_r(s)$	= reference trajectory in the Laplace transform	[-]
$y_r(k)$	= reference trajectory	[-]
$\Delta t$	= sampling time	[s]
$\tau$	= process time constant	[s]
$\tau_r$	= time constant for reference trajectory	[s]

## 第3章 伝達関数を応用した ARX モデルによるモデル同定

### 3.1 緒言

1970 年代後半に、フランスの ADERSA 社が IDCOM[6]、アメリカの Shell Oil 社が DMC[7] を最初の線形モデル予測制御としてプロセス産業界に発表してから、30 年あまりが過ぎた。この間、非線形モデル予測制御を含む様々なモデル予測制御技術が開発され、これらのモデル予測制御は、石油精製・石油化学に代表されるプロセス産業だけではなく、様々な分野への適用が進んできた [18]。プロセス産業におけるモデル予測制御技術は、プロセス制御の代表的な技術の一つとなった。

モデル予測制御の適用が進んでいく中、モデル予測制御のメンテナンスが大きな問題になってきている。なぜなら、プラントにおいては反応器の触媒の劣化が起こること、生産効率改善のための改造・増強工事が行われること、原料や運転条件の大幅な変更があることなどがあり、これらの理由によりモデル予測制御適用当初のダイナミックモデルの精度を維持することが難しく、モデル精度が落ちていくとともに制御性能が劣化していくからである。ダイナミックモデルの精度およびモデル予測制御の制御性能監視がモデル予測制御の性能を維持していくためには重要なことであり、思ふような制御性能が得られなくなった場合は、ダイナミックモデルを再同定する必要がある。この再同定を行うためには、ステップテストが必須であり、このテストはかなりの労力と時間を消費する。そこで、同定作業の省力化には、ダイナミックモデルのモデルミスマッチの検出、ステップテスト、モデル同定手法の開発が重要である。ダイナミックモデルのモデルミスマッチの検出には、Shigi ら [19] が閉ループの運転データを使用して検出する方法を提案している。閉ループでのステップテスト手法としては PRBS(Pseudo Random Binary Sequence)[13] という手法が提案されており、閉ループでのステップテスト手法としては Sotomayor ら [14] がステップテストの自動化を提案している。モデル同定手法としては、Ljung[15] が System Identification で様々な同定手法提案を行っており、Matlab System Identification Toolbox に様々な同定技術が取り入れられている。最近のモデル同定法としては、部分空間法による閉ループシステムの同定 [17] が提案されている。

モデル同定に費やす労力を軽減するために、閉ループでのステップテストデータによるモデル同定法が有用である。そこで、本研究では伝達関数モデルを ARX モデルに変換する方法を用いた新たなアプローチによるシンプルなモデル同定法を提案し、

実プロセスにおいて、ステップテストをモデル予測制御閉ループ状態で行い、モデル同定法に本同定法を使用した結果を示す。

## 3.2 モデル同定手法

### 3.2.1 同定に用いるモデル構造

本研究では、制御対象プロセスを線形モデルである ARX モデルで同定を行う。ARX モデルは次式で定義される。

$$y_m(k+1) = \sum_{i=1}^m \alpha_i y_m(k-i+1) + \sum_{i=1}^n b_i u(k-i+1) + d(k) \quad (3.1)$$

ここで、 $y_m$  はプロセスモデル出力、 $u$  はプロセス入力、 $\alpha_i$  は自己回帰係数 (AR)、 $b_i$  は外生入力係数、 $d$  は外乱、 $m$  と  $n$  は自己回帰係数と外生入力係数の打ち切り次数である。式 (3.1) はインクリメンタル形式で書き直し、 $d(k) - d(k-1)$  を無視しすると次式になる。

$$\begin{aligned} y_m(k+1) = y_m(k) &+ \sum_{i=1}^m \alpha_i \{y_m(k-i+1) - y_m(k-i)\} \\ &+ \sum_{i=1}^n b_i \{u(k-i+1) - u(k-i)\} \end{aligned} \quad (3.2)$$

式 (3.2) のモデルパラメータ ( $\alpha_i, b_i$ ) は、一般的には、運転データよりパラメータフィッティングで同定されるが、本研究では以下の方法をとる。プロセス産業界において、プロセスダイナミクスは通常無駄時間のある 1 次遅れ系 (First Order plus Dead Time, FOPDT) システムや 2 次遅れ系 (Second Order plus Dead Time, SOPDT) システムとして仮定され、表される。本研究で提案する方法では、式 (3.2) のパラメータを伝達関数のモデルパラメータで表すことで構築される。伝達関数  $G(s)$  は以下の式を使用する：

- 1 次遅れ系 (FOPDT) モデル

$$G(s) = \frac{K}{\tau s + 1} e^{-Ds} \quad (3.3)$$

- 2 次遅れ系 (SOPDT) モデル (Time Constant (TC))

$$G(s) = \frac{K(\tau_0 s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-Ds} \quad (3.4)$$

- 2 次遅れ系 (SOPDT) モデル (Damping Ratio (DR))

$$G(s) = \frac{K(\tau_0 s + 1)\omega^2}{s^2 + 2\zeta\omega s + \omega^2} e^{-Ds} \quad (3.5)$$

- 積分系 (Ramp) モデル

$$G(s) = \frac{K(\tau_0 s + 1)}{(\tau s + 1)s} e^{-Ds} \quad (3.6)$$

ここで、 $K$  はプロセスゲイン、 $\tau$  は時定数、 $D$  は無駄時間、 $\tau_0$  は進み時間、 $\tau_1$  and  $\tau_2$  は SOPDT システムの時定数、 $\zeta$  は減衰比、 $\omega$  は固定角周波数である。

上記のそれぞれの伝達関数は以下の方法で ARX モデルに変換できる。

まず最初に、AR パラメータは次のように伝達関数  $G(s)$  の分母だけから計算され、その分母は以下のような部分分数に表すことができる：

$$G(s) = \sum_{i=1}^M \frac{q_i}{s - s_i} \quad (3.7)$$

ここで、 $q_i$  は定数である。次数  $M$  は FOPDT で 1、SOPDT/Ramp で 2 である。パラメータ  $s_i$  は表 3.1 にまとめた。式 (3.7) に離散時間  $\Delta t$  でインパルス普遍法を適用すると、以下の式 (3.8) が得られる：

$$G(z) = \sum_{i=1}^M \frac{q_i}{1 - e^{s_i \Delta t} z^{-1}} \quad (3.8)$$

式 (3.8) は次の式 (3.9) に書き換えられる：

$$G(z) = \frac{\sum_{i=1}^M d_i z^{-i+1}}{1 + \sum_{i=1}^M r_i z^{-i}} \quad (3.9)$$

次の式は式 (3.9) の逆 Z 変換により得られる：

$$\begin{aligned} y(k) + r_1 y(k-1) + r_2 y(k-2) + \cdots + r_M y(k-M) \\ = d_1 u(k) + d_2 u(k-1) + \cdots + d_M u(k-M) \end{aligned} \quad (3.10)$$

表 3.3 の AR パラメータは式 (3.10) のパラメータにより次のように計算される：

$$\begin{aligned} \alpha_i &= -r_i \\ \begin{cases} r_1 = -e^{s_1 \Delta t} & (M=1) \\ r_1 = -(e^{s_1 \Delta t} + e^{s_2 \Delta t}), r_2 = e^{s_1 \Delta t} e^{s_2 \Delta t} & (M=2) \end{cases} \end{aligned} \quad (3.11)$$

上記より、AR パラメータの次数  $m$  は 1 か 2 である。

次に、パラメータ  $b_i$  は上述して求めた AR パラメータにより以下の手順で計算される。無駄時間を無視した伝達関数の単位ステップ応答  $f(t)$  を表 3.2 に示す。プロセスモデル出力  $y_m(k)$  をプロセスに単位ステップ入力を入れたとして式  $(f(t))$  を使うと以下のように表される：

$$y_m(k) = \begin{cases} 0 & (k\Delta t < D) \\ f(k\Delta t - D) & (k\Delta t \geq D) \end{cases} \quad (3.12)$$

無駄時間  $D$  は  $D = L\Delta t + R$  として表現できる。ここで、 $L$  は離散時間による無駄時間、 $R$  は  $D/\Delta t$  の余りである。式 (3.12) は  $L$  と  $R$  で以下のように表すことができる:

$$y_m(k) = \begin{cases} 0 & (k \leq L) \\ f((k - L)\Delta t - R) & (k > L) \end{cases} \quad (3.13)$$

ここで、 $b_i = 0$  ( $i = 1, 2, \dots, L$ ) と  $m = 2$  である。それから、式 (3.2) は次の式 (3.14) に書き換えられる。

$$\begin{aligned} y_m(k+1) = y_m(k) &+ \sum_{i=1}^2 \alpha_i \{y_m(k-i+1) - y_m(k-i)\} \\ &+ \sum_{i=1}^n b_i \{u(k-L-i+1) - u(k-L-i)\} \end{aligned} \quad (3.14)$$

式 (3.14) に単位ステップ入力を適用することで、次の式 (3.15) が得られる:

$$\left\{ \begin{array}{l} y_m(L+1) = b_1 \\ y_m(L+2) = y_m(L+1) + \alpha_1 y_m(L+1) + b_2 \\ y_m(L+3) = y_m(L+2) + \alpha_1 \{y_m(L+2) - y_m(L+1)\} + \alpha_2 y_m(L+1) + b_3 \\ y_m(L+4) = y_m(L+3) + \alpha_1 \{y_m(L+3) - y_m(L+2)\} \\ \quad + \alpha_2 \{y_m(L+2) - y_m(L+1)\} + b_4 \\ \vdots \\ y_m(L+i) = y_m(L+i-1) + \alpha_1 \{y_m(L+i-1) - y_m(L+i-2)\} \\ \quad + \alpha_2 \{y_m(L+i-2) - y_m(L+i-3)\} + b_i \end{array} \right. \quad (3.15)$$

式 (3.15) は単位ステップ応答  $f(t)$  により以下のように表される:

$$\left\{ \begin{array}{l} b_1 = f(\Delta t - R) \\ b_2 = f(2\Delta t - R) - f(\Delta t - R) - \alpha_1 f(\Delta t - R) \\ b_3 = f(3\Delta t - R) - f(2\Delta t - R) - \alpha_1 \{f(2\Delta t - R) - f(\Delta t - R)\} \\ \quad - \alpha_2 f(\Delta t - R) \\ b_i = 0 \quad (i = 4, 5, \dots) \end{array} \right. \quad (3.16)$$

ここで、 $b_i = 0$  ( $i = 4, 5, \dots$ ) は式 (3.15) に  $f(t)$  と  $\alpha_i$  を代入して得られる。

最終的に、 $\alpha_i$  と  $b_i$  は表 3.3 のようにまとめることができ、モデル同定に使用する

ARX モデルが以下の式のように構築できる:

$$y_m(k+1) = y_m(k) + \sum_{i=1}^2 \alpha_i \{y_m(k-i+1) - y_m(k-i)\} + \sum_{i=1}^3 b_i \{u(k-L-i+1) - u(k-L-i)\} \quad (3.17)$$

モデル同定に使用する式 (3.17) の特徴をあげると以下の通りである。

- 無駄時間に関する近似なしに伝達関数を ARX モデルに変換可能
- 無駄時間に関する連続性を保つことが可能
- 簡単に連続モデルを離散モデルに変換可能

表 3.1: Parameters  $s_i$

Process Model		$s_1$	$s_2$
FOPDT		$-\frac{1}{\tau}$	-
SOPDT (TC)	$\tau_1 \neq \tau_2$	$-\frac{1}{\tau_1}$	$-\frac{1}{\tau_2}$
	$\tau_1 = \tau_2$	$-\frac{1}{\tau_1}$	$-\frac{1}{\tau_1}$
SOPDT (DR)	$\zeta < 1$	$-\zeta\omega + \omega\sqrt{1-\zeta^2}i$	$-\zeta\omega - \omega\sqrt{1-\zeta^2}i$
	$\zeta = 1$	$-\omega$	$-\omega$
	$\zeta < 1$	$-\zeta\omega + \omega\sqrt{\zeta^2-1}$	$-\zeta\omega - \omega\sqrt{\zeta^2-1}$
Ramp		$-\frac{1}{\tau}$	0

表 3.2: Unit Step Response without Dead Time

Process Model		Unit Step Response without Dead Time $f(t)$
FOPDT		$K(1 - e^{-\frac{t}{\tau}})$
SOPDT (TC)	$\tau_1 \neq \tau_2$	$K\{\frac{\tau_0-\tau_1}{\tau_1-\tau_2}e^{-\frac{t}{\tau_1}} + \frac{\tau_0-\tau_2}{\tau_2-\tau_1}e^{-\frac{t}{\tau_2}} + 1\}$
	$\tau_1 = \tau_2$	$K\{-e^{-\frac{t}{\tau_1}} + \frac{\tau_0-\tau_1}{\tau_1^2}te^{-\frac{t}{\tau_1}} + 1\}$
SOPDT (DR)	$\zeta < 1$	$K\{-e^{-\zeta\omega t} \cos(\omega\sqrt{1-\zeta^2}t) + \frac{\tau_0\omega-\zeta}{\sqrt{1-\zeta^2}}e^{-\zeta\omega t} \sin(\omega\sqrt{1-\zeta^2}t) + 1\}$
	$\zeta = 1$	$K\{-e^{-\omega t} + (\tau_0\omega - 1)\omega te^{-\omega t} + 1\}$
	$\zeta > 1$	$K[\frac{\tau_0\omega-\zeta-\sqrt{\zeta^2-1}}{2\sqrt{\zeta^2-1}}e^{\omega(-\zeta+\sqrt{\zeta^2-1})t} - \frac{\tau_0\omega-\zeta+\sqrt{\zeta^2-1}}{2\sqrt{\zeta^2-1}}e^{\omega(-\zeta-\sqrt{\zeta^2-1})t} + 1]$
Ramp		$K\{-(\tau_0 - \tau_1)e^{-\frac{t}{\tau_1}} + (\tau_0 - \tau_1) + t\}$

表 3.3: Model Parameters of ARX Models based on Transfer Functions

Process Model	$\alpha_1$	$\alpha_2$	$b_1$	$b_2$	$b_3$	Remarks
$\frac{Ke^{-Ds}}{\tau s+1}$	$a$	0	$K(1-Ca)$	$Ka(C-1)$	0	$a=e^{-\frac{\Delta t}{\tau}}, C=e^{\frac{R}{\tau}}$
$\frac{K(\tau_0 s+1)e^{-Ds}}{(\tau_1 s+1)(\tau_2 s+1)}$	$a_1+a_2$	$-a_1 a_2$	$K(A_1 a_1+A_2 a_2+1)$	$-K\left[\{A_1(a_2+1)+1\}a_1+\{A_2(a_1+1)+1\}a_2\right]$	$K(A_1 C_1+A_2 R+1)a_1 a_2$	$a_1=e^{-\frac{\tau_1}{\tau_1}}, a_2=e^{-\frac{\tau_2}{\tau_2}},$ $A_1=\frac{\tau_0-\tau_1}{\tau_1-\tau_2}, A_2=\frac{\tau_0-\tau_2}{\tau_2-\tau_1}$
$\tau_1=\tau_2$	$2a$	$-a^2$	$K\{(A\Delta t-1)a+1\}$	$Ka\{a-(A\Delta t+1)\}$	$K(1-C-ACR)a^2$	$a=e^{-\frac{\Delta t}{\tau_1}}, A=\frac{\tau_0-\tau_1}{\tau_1}$
$\frac{K(\tau_0 s+1)\omega^2 e^{-Ds}}{s^2-2\zeta\omega s+\omega^2}$	$2a\cos(E\Delta t)$	$-a^2$	$K[-Ca\cos\{E(\Delta t-R)\}+ACa\sin\{E(\Delta t-R)\}+1]$	$Ka[C(B_1+AB_2)a+C\cos\{E(\Delta t-R)\}-AC\sin\{E(\Delta t-R)\}-2\cos(E\Delta t)]$	$Ka^2\{1-(B_1+AB_2)C\}$	$a=e^{-\zeta\omega\Delta t}, A=\frac{\tau_0\omega-\zeta}{\sqrt{1-\zeta^2}},$ $C=e^{\zeta\omega R}, E=\omega\sqrt{1-\zeta^2},$ $B_1=\cos ER, B_2=\sin ER$
$\zeta=1$	$2a$	$-a^2$	$K\{-Ca+AC(\Delta t-R)a+1\}$	$Ka[(1+AR)Ca+\{C-2-AC(\Delta t-R)\}]$	$K(1-C-ACR)a^2$	$a=e^{-\omega\Delta t}, A=(\tau_0\omega-1)\omega$ $C=e^{\omega R}$
$\zeta>1$	$a_1+a_2$	$-a_1 a_2$	$K(A_1 C_1 a_1+A_2 R a_2+1)$	$-K[\{A_1 C_1(a_2+1)+1\}a_1+\{A_2 R(a_1+1)+1\}a_2]$	$K(A_1 C_1+A_2 R+1)a_1 a_2$	$\beta_1=-\zeta\omega+\omega\sqrt{\zeta^2-1},$ $\beta_2=-\zeta\omega-\omega\sqrt{\zeta^2-1},$ $a_1=e^{\beta_1\Delta t}, a_2=e^{\beta_2\Delta t},$ $A_1=\frac{\tau_0\omega-\zeta-\sqrt{\zeta^2-1}}{2\sqrt{\zeta^2-1}},$ $A_2=-\frac{\tau_0\omega-\zeta+\sqrt{\zeta^2-1}}{2\sqrt{\zeta^2-1}},$ $C_1=e^{-\beta_1 R}, R=e^{\beta_2 R}$
$\frac{K(\tau_0 s+1)e^{-Ds}}{(\tau s+1)s}$	$a+1$	$-a$	$K\{A(1-Ca)+\Delta t-R\}$	$K\{a(2AC-\Delta t)+(a+1)(R-A)\}$	$Ka\{A(1-C)-R\}$	$a=e^{-\frac{\Delta t}{\tau}}, C=e^{\frac{R}{\tau}},$ $A=\tau_0-\tau$

### 3.2.2 モデル同定

提案するモデル同定手法は式 (3.17) による PEM (Prediction Error Minimization) 法を適用する。PEM 法とは同定パラメータを変数とする以下の式に示すようなプロセス出力の実測値とモデル出力の誤差を最小にする方法であり、本研究における同定パラメータは伝達関数モデルパラメータである。

$$\underset{\theta}{\text{minimize}} J = \sum_{i=1}^{n_d} \{y_m(i) - y(i)\}^2 \quad (3.18)$$

ここで、 $\theta$  は伝達関数モデルパラメータのベクトル (例えば、1 次遅れ系システムでは  $\theta=[K, \tau, D]^T$ )、 $y$  はプロセス出力の実測値、 $n_d$  はモデル同定のための実測データの数である。

式 (3.18) は準ニュートン法などを用いることによって、最適値を得ることができる。図 3.1 に提案するモデル同定スキームを示す。伝達関数の種類を選択した後、プロセスからの実測値を用いて最適化が実行される。

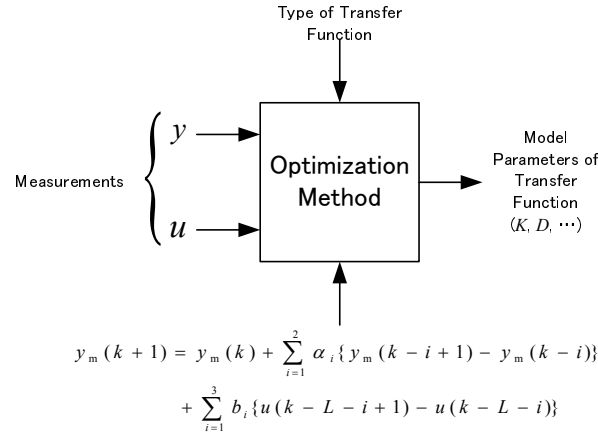


図 3.1: Proposed Identification of Linear Models

### 3.3 提案手法によるモデル同定例

提案手法によるモデル同定の例として、図 3.2 に示す蒸留塔である脱エタン塔の塔内温度制御と塔底液面制御のモデル同定に適用を行った。この脱エタン塔は混合フィードを塔頂製品エタンと塔底製品プロパンに分ける蒸留プロセスである。脱エタン塔のシミュレーションモデルは Invensys 製のダイナミックシミュレーションソフトウェア パッケージ Dynsim に構築した。この Dynsim は通常はプロセス設計と制御系設計に



使用される。塔内温度制御 (TC101) と塔底液面制御 (LC101) のプロセスモデルはそれぞれ SOPDT システムと Ramp システムとして同定する。

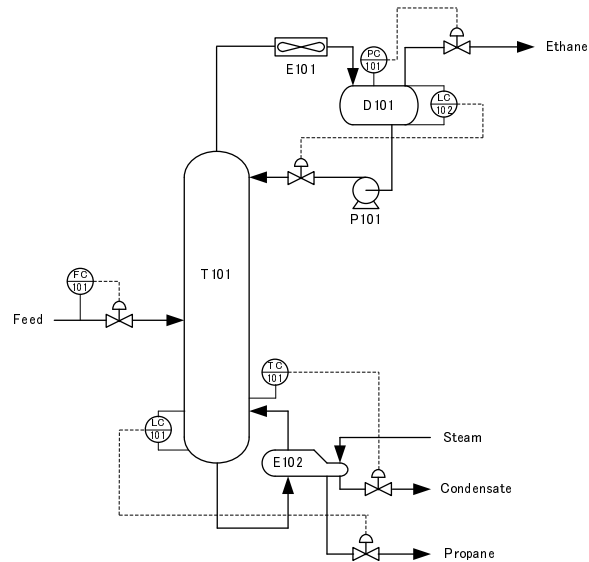


図 3.2: Scheme of the Deethanizer

### 3.3.1 2 次遅れ系モデル (Time Constant)

塔内温度制御 (TC101) は塔底付近の温度を制御するためにリボイラーのスチーム制御弁を操作している。この同定は図 3.3 に示す開ループステップ応答テストデータを使用して SOPDT システム (Time Constant) として行われる。同定されたモデルのパラメータを表 3.4 に示す。同定の結果を図 3.3 示すように、塔底温度の実測値に対して精度よく推定されている。

表 3.4: Model Parameters of the Column Temperature Process

Parameters	Value	Unit
$K$	4.17	K/%
$\tau_0$	109.17	s
$\tau_1$	209.72	s
$\tau_2$	218.87	s
$D$	3.31	s

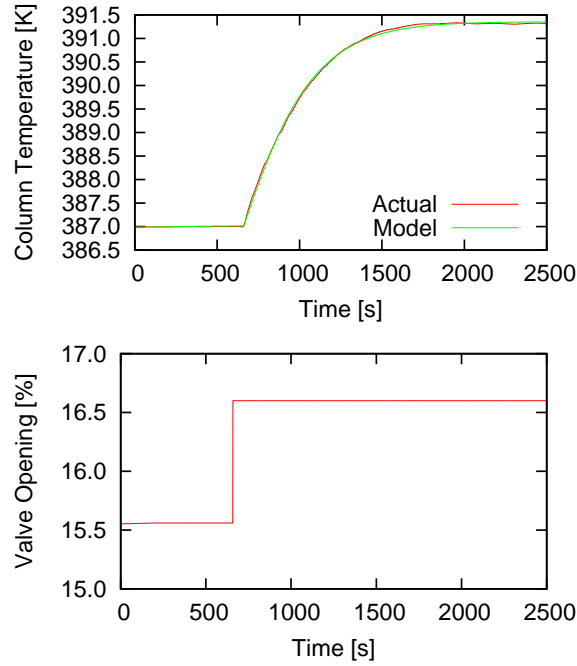


図 3.3: Model Identification Result for the Column Temperature

### 3.3.2 積分系モデル

この節では提案手法による積分系システムのモデル同定を脱エタン塔の塔底液面制御 (LC101) に対して適用した例を示す。積分システムを同定するためには、アルゴリズムの修正が必要である。なぜなら積分系システムでは、その傾きを同定することが FOPDT/SOPDT システムのプロセスゲインを同定するのと同様に重要であるからである。従って、式 (3.18) に代えて、提案手法による積分系システムの同定のための評価関数を次のような式に変更する：

$$\underset{\theta}{\text{minimize}} J = \sum_{i=1}^{n_d-1} \{\Delta y_m(i) - \Delta y(i)\}^2 \quad (3.19)$$

修正された提案手法を示すために、まず、図 3.4 に示す実運転データを収集する開ループステップ応答テストを実行した。次に、積分系システムのモデル同定を行い、表 3.5 に示すモデルパラメータを求めた。積分系システムのモデル同定結果を図 3.4 に示す。この結果から、同定モデルでは違う定常状態に落ち着いているが、積分系システムの傾きは精度よく同定されている。

表 3.5: Model Parameters of the Column Bottom Level Process

Parameters	Value	Unit
$K$	$-6.564 \times 10^{-5}$	(m/s)/%
$\tau_0$	0	s
$\tau$	0.905	s
$D$	0.246	s

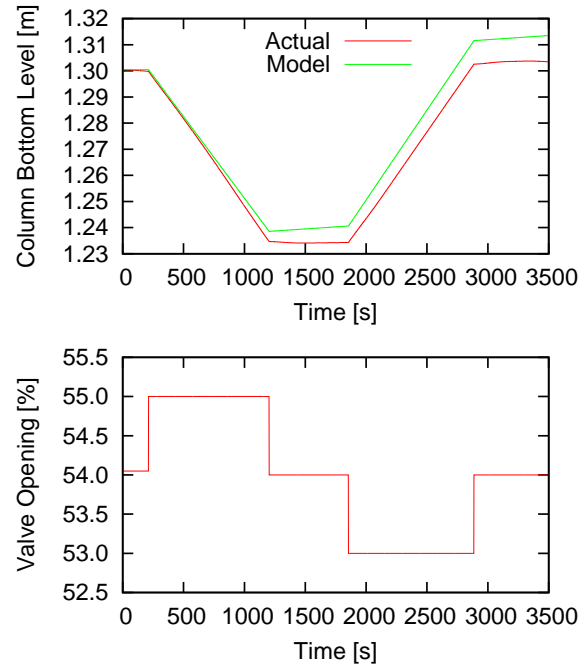


图 3.4: Model Identification Result for the Column Bottom Level

### 3.4 既往手法との比較

この節では、Ljung [15] により提案された既往手法 (Ljung Method, LM) と本提案手法 (ARX Method, ARXM) との比較を行う。この既往手法は Iterative Prediction-Error Minimization Method (PEM) と組み合わせて MatLab の System Identification Toolbox で実行することができる。これらの二つの手法は無駄時間の取り扱い方に違いがある。

#### 3.4.1 既往の伝達関数を用いたモデル同定手法

プロセスモデルのインパルス応答関数  $h(t)$  は次の  $N$  次ラゲール (Laguerre) 多項式として近似できる:

$$h(t) = \sum_{i=1}^N c_i l_i(t) \quad (3.20)$$
$$l_i(t) = \sqrt{2p} \frac{e^{pt}}{(i-1)!} \frac{d^{i-1}}{dt^{i-1}} (t^{i-1} e^{-2pt})$$

ここで、 $p$  は時間スケーリング因子である。 $p$  の最適な選択は Hirama *et al.* [16] により提示されている。

式 (3.20) のパラメータ  $c_i$  は次の式で定義される:

$$c_i = \int_0^\infty h(t) l_i(t) dt \quad (3.21)$$

パーセバルの定理 (Parseval's theorem) を適用することで、式 (3.21) は次のように書き直すことができる:

$$c_i = \frac{1}{2\pi} \int_{-\infty}^\infty G(-j\omega) \sqrt{2p} \frac{(j\omega - p)^{i-1}}{(j\omega - p)^i} d\omega \quad (3.22)$$

ここで、 $G(s) = \mathcal{L}[h(t)]$  である。パラメータ  $c_i$  は  $K$ 、 $\tau$ 、 $D$  のような伝達関数モデルパラメータの関数である。

式 (3.20) から、 $G(s)$  は直交関数展開級数として表すことができる:

$$G(s) = \sum_{i=1}^N c_i \sqrt{2p} \frac{(s-p)^{i-1}}{(s+p)^i} \quad (3.23)$$

式 (3.23) は容易に部分分数に展開することができるので、式 (3.23) の伝達関数モデルパラメータは Iterative Prediction-Error Minimization 法によりプラント運転データから同定することができる。

### 3.4.2 比較結果

既往手法と提案手法の比較を表 3.6 に示す FOPDT、SOPDT TC、SOPDT DR システム ( $G(s)$ ) で行った。ステップテストによるデータの収集において、表 3.6 の伝達関数を用いて以下のようなモデル同定用のデータを作成した:

- 図 3.5,3.7,3.9 に示すノイズなしの開ループステップ応答テストデータ
- 図 3.5,3.7,3.9 に示すノイズありの開ループステップ応答テストデータ
- 図 3.6,3.8,3.10 に示すノイズなしの閉ループステップ応答テストデータ
- 図 3.6,3.8,3.10 に示すノイズありの閉ループステップ応答テストデータ

ノイズは標準偏差 0.5 の白色ノイズであり、プロセス出力に追加した。閉ループステップテストは I-PD 制御を使用して、コントローラの設定値を変更することで行った。

LM と ARXM のモデル同定結果を表 3.6 に示す。表 3.6 に示したモデル同定結果のモデル誤差  $\epsilon$  は以下の式で計算した:

$$\epsilon = \frac{1}{n_d} \sum_{i=1}^{n_d} |SR_i - \hat{S}R_i| \quad (3.24)$$

ここで、 $SR$  は実伝達関数による単位ステップ応答、 $\hat{S}R$  は同定されたモデルによる単位ステップ応答である。

この結果より、ノイズフリーでの開ループと閉ループ両方のステップテストデータでは、LM と ARXM でほとんど同じような同定精度結果が得られた。しかしながら、測定ノイズを加えた状況下では、LM より ARXM の方が少しだけ精度の高いモデルが同定できた。この理由としては、LM では無駄時間に対して直交関数級数展開を行う仮定を用いているのに対して、ARXM では無駄時間に対して仮定なしに解析解を求めているからであると考えられる。

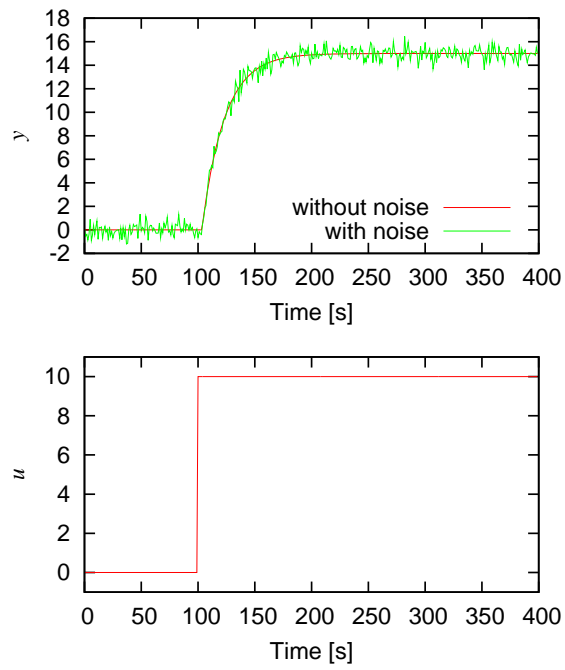
## 3.5 閉ループモデル同定による実プロセスの PID 制御チューニング

この節では実プロセスの SISO (Single Input Single Output) システムで閉ループステップ応答テストを行い、モデル同定を実行して、PID コントローラのチューニング結果を報告する。実プロセスの SISO システムを図 3.11 に示す。このシステムは 3 つのタンクで構成されており、中間のタンク液面は横河製 DCS Centum CS3000 の PID コントローラにより制御を行っている。

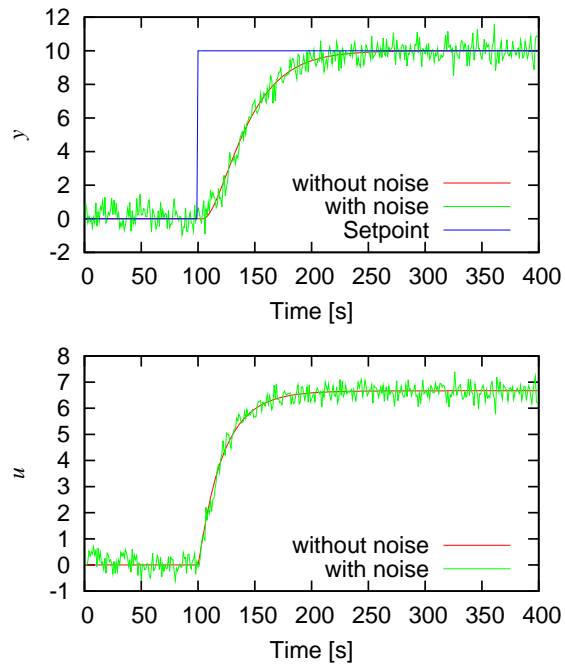
このシステムは財団法人 国際石油交流センター (Japan Cooperation Center, Petroleum (JCCP)) 殿が所有しているもので、JCCP 殿では定期的に産油国ダウンストリーム部

表 3.6: Comparison Results between LM and ARXM

$G(s)$			$\frac{1.5e^{-4.3s}}{20s+1}$	$\frac{1.5(20s+1)e^{-4.7s}}{(25s+1)(15s+1)}$	$\frac{1.5(15s+1)0.05^2e^{-4.8s}}{s^2+2\cdot0.05\cdot0.5s+1}$
without noise	Open Loop	LM	$\frac{1.500e^{-4.300s}}{20.000s+1}$	$\frac{1.500(20.000s+1)e^{-4.700s}}{(25.000s+1)(15.000s+1)}$	$\frac{1.500(15.000s+1)0.050^2e^{-4.800s}}{s^2+2\cdot0.050\cdot0.500s+1}$
			$\epsilon = 0.00000$	$\epsilon = 0.00000$	$\epsilon = 0.00000$
		ARXM	$\frac{1.500e^{-4.300s}}{20.000s+1}$	$\frac{1.500(21.091s+1)e^{-4.693s}}{(25.673s+1)(15.439s+1)}$	$\frac{1.500(15.002s+1)0.050^2e^{-4.800s}}{s^2+2\cdot0.050\cdot0.500s+1}$
			$\epsilon = 0.00000$	$\epsilon = 0.00012$	$\epsilon = 0.00001$
	Closed Loop	LM	$\frac{1.500e^{-4.300s}}{20.000s+1}$	$\frac{1.500(20.000s+1)e^{-4.700s}}{(25.000s+1)(15.000s+1)}$	$\frac{1.500(15.000s+1)0.050^2e^{-4.800s}}{s^2+2\cdot0.050\cdot0.500s+1}$
			$\epsilon = 0.00000$	$\epsilon = 0.00000$	$\epsilon = 0.00000$
		ARXM	$\frac{1.500e^{-4.300s}}{20.000s+1}$	$\frac{1.500(17.508s+1)e^{-4.749s}}{(23.799s+1)(13.640s+1)}$	$\frac{1.500(15.001s+1)0.050^2e^{-4.800s}}{s^2+2\cdot0.050\cdot0.500s+1}$
			$\epsilon = 0.00000$	$\epsilon = 0.00033$	$\epsilon = 0.00001$
with noise	Open Loop	LM	$\frac{1.494e^{-4.357s}}{19.830s+1}$	$\frac{1.482(47.037s+1)e^{-5.187s}}{(50.791s+1)(16.292s+1)}$	$\frac{1.503(16.407s+1)0.050^2e^{-5.050s}}{s^2+2\cdot0.050\cdot0.522s+1}$
			$\epsilon = 0.00373$	$\epsilon = 0.01929$	$\epsilon = 0.01299$
		ARXM	$\frac{1.499e^{-4.298s}}{19.821s+1}$	$\frac{1.507(58.386s+1)e^{-5.027s}}{(62.317s+1)(16.741s+1)}$	$\frac{1.504(16.015s+1)0.050^2e^{-5.000s}}{s^2+2\cdot0.050\cdot0.518s+1}$
			$\epsilon = 0.00152$	$\epsilon = 0.00876$	$\epsilon = 0.01134$
	Closed Loop	LM	$\frac{1.501e^{-4.153s}}{20.939s+1}$	$\frac{1.513(8.605s+1)e^{-3.008s}}{(19.586s+1)(10.668s+1)}$	$\frac{1.499(15.729s+1)0.048^2e^{-4.737s}}{s^2+2\cdot0.048\cdot0.501s+1}$
			$\epsilon = 0.00760$	$\epsilon = 0.01621$	$\epsilon = 0.01345$
		ARXM	$\frac{1.509e^{-4.000s}}{20.891s+1}$	$\frac{1.505(3.973s+1)e^{-3.000s}}{(19.728s+1)(6.002s+1)}$	$\frac{1.498(15.461s+1)0.048^2e^{-4.698s}}{s^2+2\cdot0.048\cdot0.499s+1}$
			$\epsilon = 0.00616$	$\epsilon = 0.01253$	$\epsilon = 0.01488$



☒ 3.5: Open-Loop Testing Data for Model Identification of the FOPDT System



☒ 3.6: Closed-Loop Testing Data for Model Identification of the FOPDT System

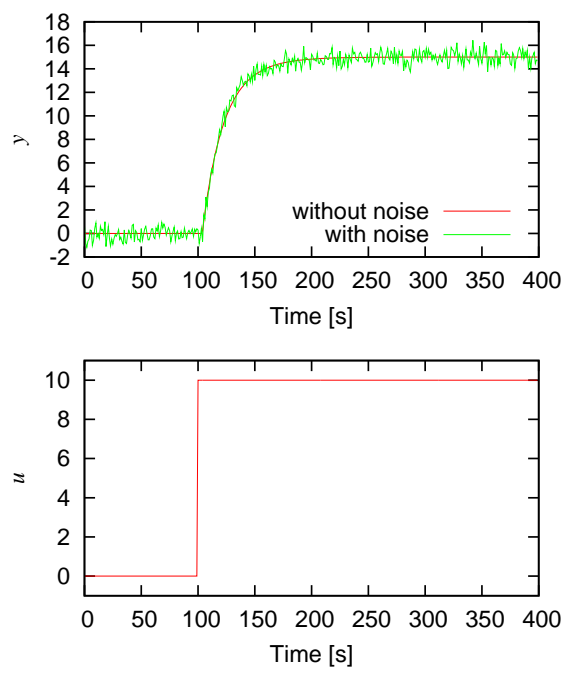
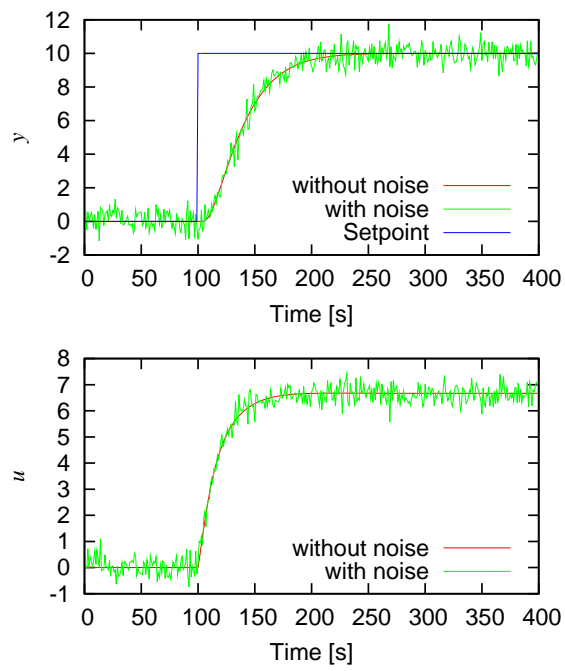


图 3.7: Open-Loop Testing Data for Model Identification of the SOPDT System (TC)





☒ 3.8: Closed-Loop Testing Data for Model Identification of the SOPDT System (TC)

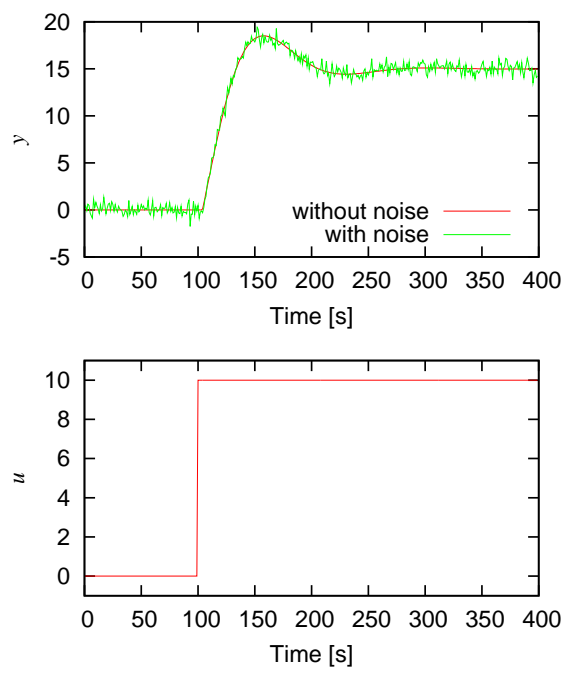


图 3.9: Open-Loop Testing Data for Model Identification of the SOPDT System (DR)

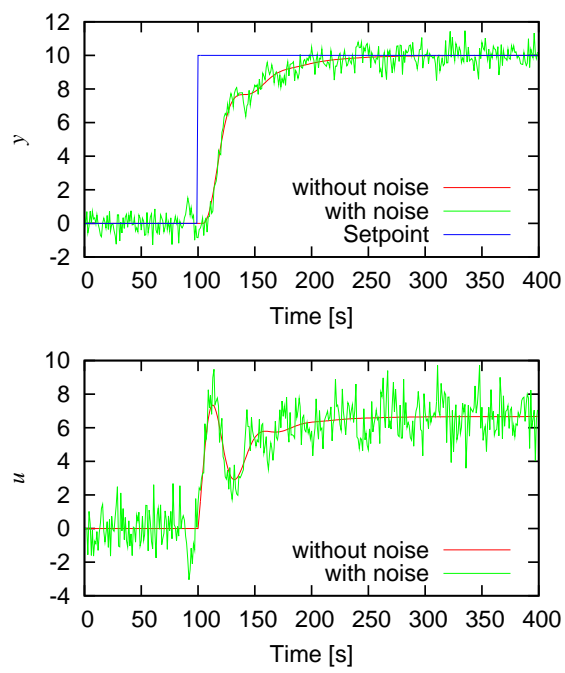


图 3.10: Closed-Loop Testing Data for Model Identification of the SOPDT System (DR)

門に関わる技術者、中間管理職、物流・販売関係者を日本に招いて日本の先進技術に関してその実用性も含めて研修を行っており、このシステムはPIDコントローラチューニングのトレーニングを行う際に使用される。

閉ループステップ応答テストは表 3.8 に示した PID チューニングの初期パラメータを使用して行った。このテストの結果を図 3.12 に示す。また、この図には SOPDT (Time Constant) モデルの同定結果も示してある。このシステムは中間タンクから最下層のタンクへの流量は中間タンクの液面の平方根に比例するという非線形を有するにもかかわらず、PID チューニングパラメータを求めるためには十分な同定結果を得ることができた。表 3.7 に同定された SOPDT モデルパラメータを示す。

PID コントローラのチューニングパラメータを伝達関数モデルから計算する方法は、本論文の 2 章で提案する手法を用いた。この手法は参照軌道を用いて、参照軌道を調節するただひとつのパラメータ・時定数により PID パラメータを求めるものである。時定数を長くすると PID コントローラはゆっくりな制御を行う。チューニングは時定数 100 秒、50 秒、20 秒で求めた表 3.8 に示すパラメータで行った。チューニング結果を図 3.13 に示す。図中の最初の設定値上下変更は表 3.8 の “Initial” を使っている。その次からの設定値上下変更は 100 秒、50 秒と 20 秒の PID パラメータをそれぞれ使用している。チューニングの結果、“Initial” パラメータより制御性がよくなっていることを確認できた。また、PID コントローラのチューニングスピードを望むように調整できている。通常、流量制御には微分時間を使用しないが、本プロセスはノイズがほとんどないので、微分時間の使用が可能であった。

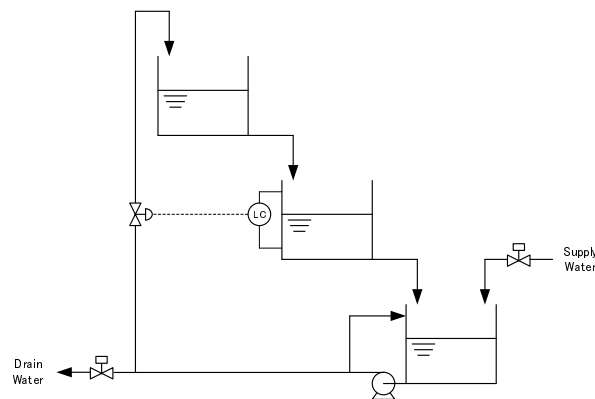


図 3.11: Scheme of the Tank System

表 3.7: Identified Second Order Model Parameters for the Tank System

2nd Order Parameter (Time Constant)	Value	Unit
$K$	8.701	mm/%
$\tau_0$	0.2954	s
$\tau_1$	27.93	s
$\tau_2$	27.93	s
$D$	8.659	s

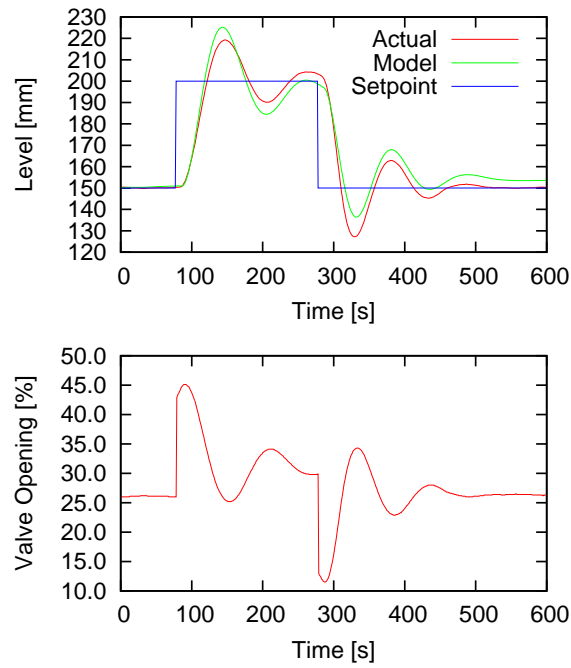


图 3.12: Closed Loop Identification Result for the Tank System

表 3.8: PID Tuning Parameters for the Tank System

	Initial	100s	50s	20s	Unit
Proportional Band	100	572.44	310.07	155.10	%
Integral Time	60	55.31	55.32	54.37	s
Derivative Time	0	15.20	14.56	15.66	s

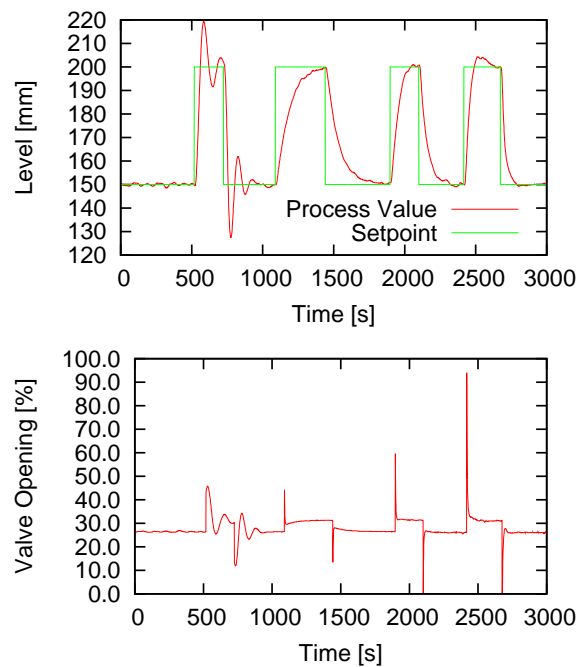


図 3.13: Performance of PID Controller for Tank System

## 3.6 多変数への拡張

この 30 年の間に、多くの線形モデル予測制御がプロセス産業に適用されてきている。このモデル予測制御の課題点はその制御性能を維持するためにはモデル予測制御のメンテナンスが必要だということである。モデル予測制御のプロセスモデルの精度監視と更新はこのメンテナンスにとって不可欠なことである。モデル予測制御は多変数を取り扱っているので、この節では、これまでの SISO システムのモデル同定手法を MIMO(Multi Input/Multi Output) システムのモデル同定に拡張する。

### 3.6.1 アルゴリズム

ある多入力多出力システム (MIMO システム) が線形システムであるという仮定の下で、MIMO システムは多入力 1 出力システム (MISO システム) の集合体であると定義できる。図 3.14 にモデル同定のための MISO システムを示す。MISO システムは、おのもののプロセス入力に対して SISO システムとなるのでそれらのプロセス出力を

合算する線形システムとして以下の式のように構築することができる。

$$y_m(k) = \sum_{i=1}^{n_u} y_{mi}(k) \quad (3.25)$$

$$\begin{aligned} y_{m1}(k+1) &= y_{m1}(k) + \sum_{i=1}^2 \alpha_{i1} \{y_{m1}(k-i+1) - y_{m1}(k-i)\} \\ &\quad + \sum_{i=1}^3 b_{i1} \{u_1(k-L_1-i+1) - u_1(k-L_1-i)\} \\ y_{m2}(k+1) &= y_{m2}(k) + \sum_{i=1}^2 \alpha_{i2} \{y_{m2}(k-i+1) - y_{m2}(k-i)\} \\ &\quad + \sum_{i=1}^3 b_{i2} \{u_2(k-L_2-i+1) - u_2(k-L_2-i)\} \\ &\vdots \\ y_{mn_u}(k+1) &= y_{mn_u}(k) + \sum_{i=1}^2 \alpha_{in_u} \{y_{mn_u}(k-i+1) - y_{mn_u}(k-i)\} \\ &\quad + \sum_{i=1}^3 b_{in_u} \{u_{n_u}(k-L_{n_u}-i+1) - u_{n_u}(k-L_{n_u}-i)\} \end{aligned} \quad (3.26)$$

ここで、 $n_u$  はプロセス入力の数を表す。

多変数モデル同定は、線形 MISO システムの式 (3.25) を式 (3.18) により最適化することで実行できる。

Matlab のシステム同定ツールボックスはまた線形 MISO システムのモデル同定にも使うことができる。このツールボックスを使ったモデル同定では、SOPDT システムの TC と DR は区別して同定する必要がある。しかし、提案手法の MISO 同定では、これらを区別する必要がなく、DR モデルとして TC モデルを同定することができる。これは  $\zeta$  の値で DR か TC かを区別することができるからである。

### 3.6.2 モデル予測制御 (MPC) の閉ループモデル同定

この節では、提案する多変数モデル同定手法によるモデル予測制御のモデルを閉ループ応答テストにより更新したケーススタディを示す。モデル予測制御の制御対象プロセスは定常状態で運転されている蒸留塔・C3/C4 スプリッター [14] である。そのプロセススキームを図 3.15 に示す。この図には同時に PID コントローラをベースとした通常の制御ループとモデル予測制御の制御スキームも含んでいる。この蒸留プロセスの目的は、LPG(liquefied petroleum gas) を塔頂製品プロパン/プロペンと塔底製品ブタン/ブテン/炭素分子 4 つ持つ他の炭化水素に分けることである。モデル予測制御は

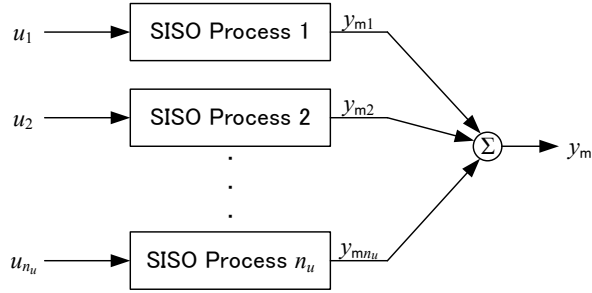


図 3.14: Representation of a Linear MISO System for Model Identification

このプロセスの運転安定化および塔底製品回収最大化を達成するために適用されている。モデル予測制御は2入力2出力システムを制御する。被制御変数 (CV) は蒸留塔の塔頂セクション1段目トレイ温度  $y_1$  と塔底製品の C3 モル分率  $y_2$  である。操作変数 (MV) は還流量  $u_1$  とリボイラーのホットオイル流量  $u_2$  である。実プロセスは次の伝達関数で表される:

$$\begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad (3.27)$$

$$G_{11}(s) = \frac{-1.135 \times 10^{-3}s - 1.235 \times 10^{-3}}{s^2 + 1.6380s + 0.09852} e^{-1s} \quad (3.28)$$

$$G_{12}(s) = \frac{-0.3 \times 10^{-3}s + 0.2 \times 10^{-2}}{s^2 + 2.4298s + 0.0651} e^{-1s} \quad (3.29)$$

$$G_{21}(s) = \frac{-2.218 \times 10^{-4}s + 5.656 \times 10^{-4}}{s^2 + 3.4948s + 0.5902} e^{-1s} \quad (3.30)$$

$$G_{22}(s) = \frac{0.7413 \times 10^{-4}s - 1.452 \times 10^{-3}}{s^2 + 2.6987s + 0.4023} e^{-1s} \quad (3.31)$$

最初に、モデル予測制御はオリジナルモデルとして次の伝達関数により制御を行った:

$$G_{11}(s) = \frac{-0.0873 \times 10^{-3}s - 0.1116 \times 10^{-3}}{s^2 + 0.1317s + 0.0073} e^{-1s} \quad (3.32)$$

$$G_{12}(s) = \frac{1.3 \times 10^{-3}s + 0.7 \times 10^{-2}}{s^2 + 2.2605s + 0.1366} e^{-1s} \quad (3.33)$$

$$G_{21}(s) = \frac{0.4227 \times 10^{-4}s + 0.1094 \times 10^{-4}}{s^2 + 0.1090s + 0.0243} e^{-1s} \quad (3.34)$$

$$G_{22}(s) = \frac{-1.2055 \times 10^{-4}s - 0.03824 \times 10^{-3}}{s^2 + 0.1342s + 0.0111} e^{-1s} \quad (3.35)$$



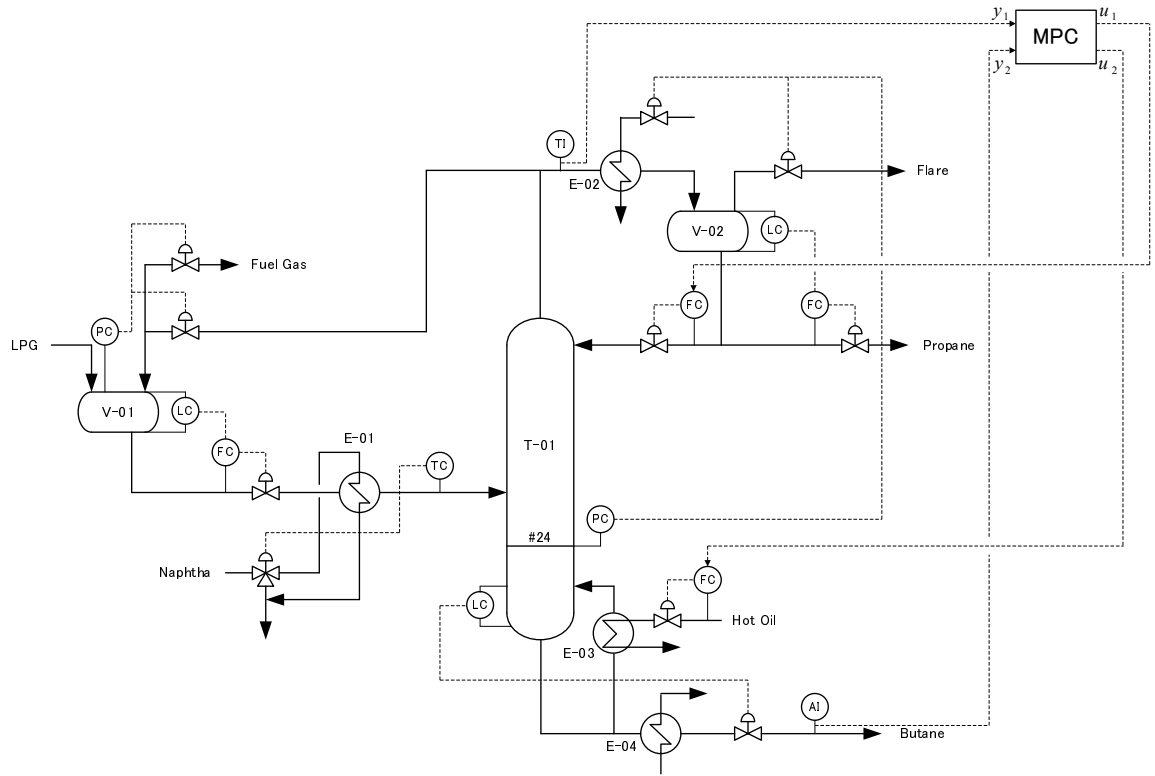
閉ループ応答モデル同定のための閉ループステップ応答テストを図 3.16 に示す。ノイズに関しては、PRBS ベースで還流量  $u_1$  に PRBS ベースの平均 0、大きさ  $150 \text{ m}^3/d$  のノイズを入れて実行した。モデル予測制御としてはプロセス産業で実績のあるインベンシス製 Connoisseur を使用した。プロセスモデルもこの Connoisseur に組み込んだ。Connoisseur では PRBS を使ったノイズをプロセスモデルのシミュレーションに追加できる。モデル同定をする際の最適化の初期値には、ステップテストに使用したモデル予測制御のオリジナルモデルを使用した。モデル同定によりオリジナルモデルの更新を行った結果を図 3.17 に示す。更新されたモデルパラメータを表 3.9 にまとめた。図 3.18 にオリジナルモデルと更新されたモデルによるそれぞれの推定値を示す。図 3.18 より、更新されたモデルでのプロセス出力推定値の精度がオリジナルモデルより改善されていることがわかる。図 3.17 より、更新されたモデルのプロセスゲインがオリジナルモデルより改善されている。塔頂温度のダイナミクスの部分は改善が行われているが、塔底製品 C3 ダイナミクスのダンピング部分はノイズがある状況で精度を改善することができなかった。しかしながら、提案手法により全体的にモデルの精度を改善することができた。

表 3.9: Comparison Table for each Model

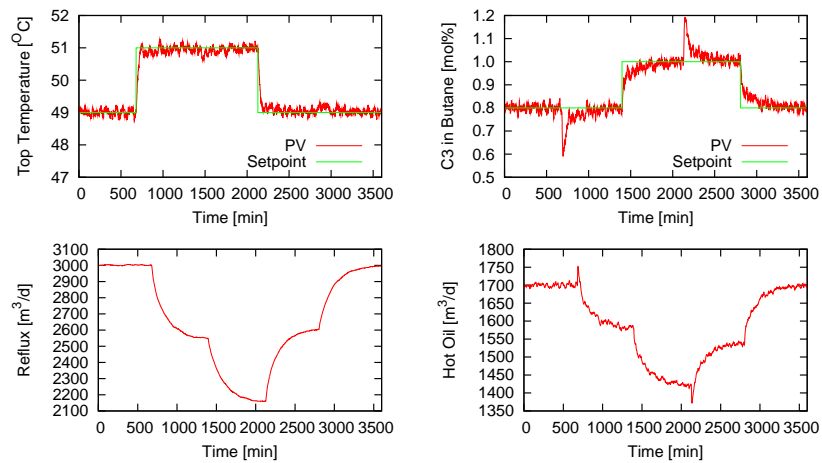
		Top Temperature			C3 in Bottom		
Reflux	$K$	Actual	Original	Updated	Actual	Original	Updated
	$\tau_0$	-0.01234	-0.01529	-0.01226	0.0009583	0.0004502	0.0008449
	$\zeta$	0.9190	0.7823	1.040	-0.3921	3.864	3.864
	$\omega$	2.609	0.7707	1.599	2.275	0.3496	0.3496
	$D$	0.3139	0.08544	0.1949	0.7632	0.1559	0.1559
	$D$	1	1	0.6764	1	1	0.9998
Hot Oil	$K$	Actual	Original	Updated	Actual	Original	Updated
	$\tau_0$	0.03072	0.05124	0.02996	-0.003609	-0.003445	-0.003312
	$\zeta$	-0.15	0.1857	-1.789	-0.05105	3.152	3.152
	$\omega$	4.762	3.058	3.444	2.127	0.6369	0.6369
	$D$	0.2551	0.3696	0.1895	0.6343	0.1054	0.1054
	$D$	1	1	1.083	1	1	0.9982

### 3.6.3 更新したモデルによる制御性能

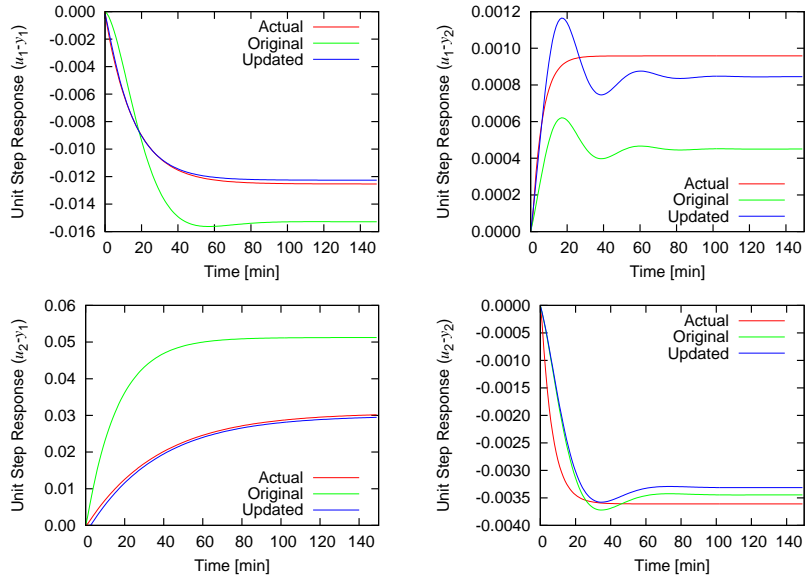
更新したモデルによる制御性能を図 3.19 に示す。3000 分までオリジナルモデルで制御を行い、3000 分で更新したモデルに切り替えた。モデル切り替えによるコントローラのチューニングパラメータの変更は行っていない。塔頂の温度制御の制御性はモデ



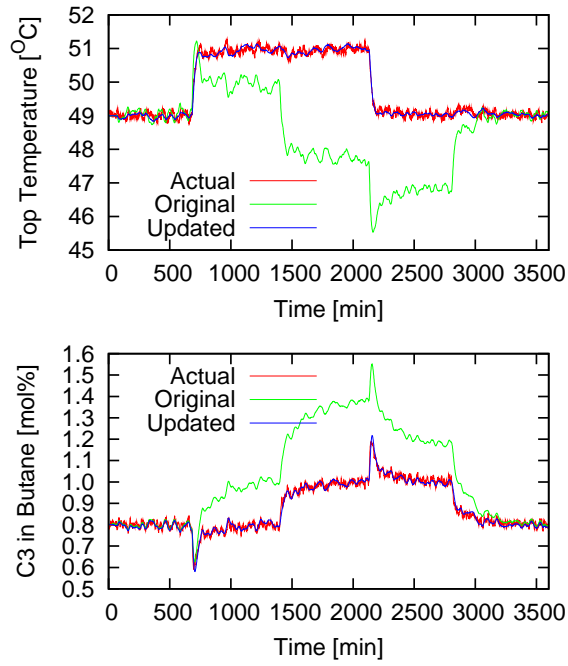
3.15: Scheme of the C3/C4 Splitter



3.16: Closed-loop Response of MPC for the C3/C4 Splitter



3.17: Comparison of Unit Step Responses for the C3/C4 Splitter



3.18: Closed-loop Response Model Identification Result for the C3/C4 Splitter

ルを切り替えてもあまり変わっていない。しかし、塔底製品 C3 の制御性ははっきりと改善されていることがわかり、操作変数の操作速度も上がっている。このスタディーにより、モデル予測制御の制御実行中に閉ループ応答テストを行うことで提案手法で多変数モデルのモデル更新を十分に行うことができることを確認した。

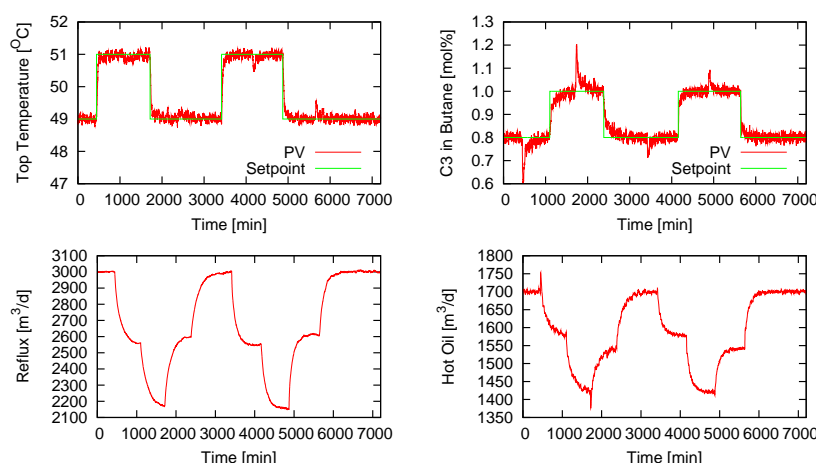


図 3.19: Comparison of Control Performance between Original Models and Updated Models

## 3.7 実プロセスへの応用:製油所の蒸留塔

本多変数モデル同定手法を JX 日鉱日石エネルギー水島製油所にある実プロセスのモデル予測制御のモデル更新に対して、閉ループモデル同定手法 [17] として適用を行ったので、その結果を報告する。

### 3.7.1 適用プロセス

モデル同定を行った石油精製における蒸留プロセスを図 3.20 に示す。製品としては、塔頂製品、中間段抜き出し製品、塔底製品の 3 種類であり、特に塔底製品が重要なため、その初留点 (Initial Boiling Point, IBP) で管理されている。

この蒸留塔へはモデル予測制御 (MPC) がすでに導入済みで、主な被制御変数 (Controlled Variables, CV)、操作変数 (Manipulated Variables, MV)、外乱変数 (Disturbance Variables, DV) を表 3.10 に示す。それから、MPC のスキームを図 3.20 に示す。 $y_1$  はソフトセンサーによる推定値、 $y_2$  と  $y_3$  は実測値からの計算値である。

表 3.10: Input/Output List for MPC

CV	$y_1$	Bottom Product IBP
	$y_2$	Middle Product to Feed Ratio
	$y_3$	Internal Reflux to Middle Product Ratio
MV	$u_1$	Reflux Flow Rate
	$u_2$	Steam Flow Rate
	$u_3$	Middle Product Flow Rate
DV	$d_1$	Feed Flow Rate

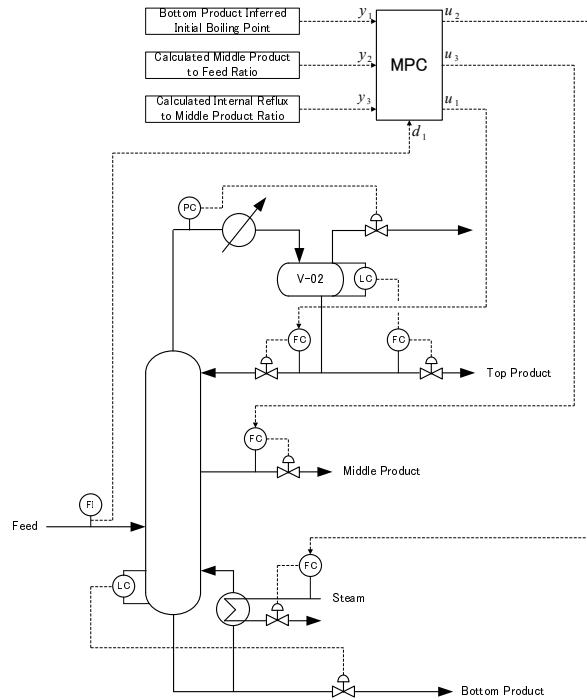


图 3.20: Scheme of The Distillation Column with MPC

### 3.7.2 現行モデルの評価

閉ループ・ステップ・テストで MPC に使われている現行のダイナミックモデルを更新する前に、現行のダイナミックモデル精度を通常運転のデータにより評価した。図 3.25 にこの評価に使用したデータセットを示す。このデータセットは運転モード変更操作を含んでいる。IBP、Middle Prod. to Feed Ratio、Int. Ref. to Mid. Prod. Ratio、Reflux、Steam、Middle Product と Feed はそれぞれ表 3.10 の  $y_1$ 、 $y_2$ 、 $y_3$ 、 $u_1$ 、 $u_2$ 、 $u_3$ 、 $d_1$  を表している。

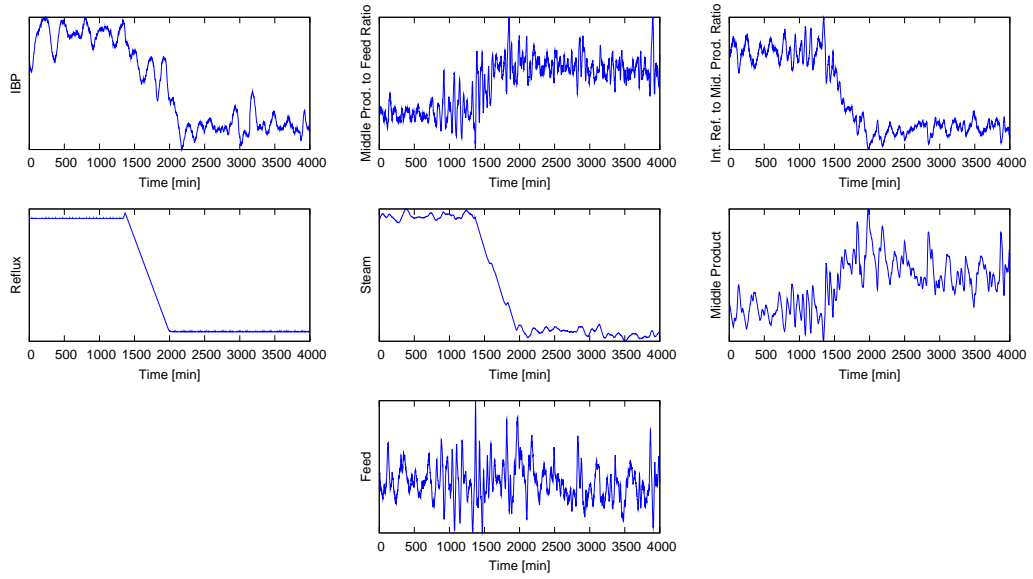
図 3.22 にモード変更データを使用して一次遅れモデルの同定を行った結果を示す。一次遅れモデル同定結果と比較のために、現行モデルも同じ図の中に示してある。通常運転データをもとにした現行モデルと同定した一次遅れモデルによるプロセス出力の比較結果を図 3.23 に示す。図 3.23 から、Middle Product to Feed Ratio と Internal Reflux to Middle Product Ratio は現行モデルと新しく同定されたモデルのモデル精度にそれほど違いがないことわかる。新しく同定されたモデルはこの同定によるモデル更新において精度があまり改善されていない。しかしながら、IBP のモデルの精度は図 3.23 からかなり改善されたことが確認できる。図 3.22 から IBP のモデルの精度の改善理由を推測すると、現行モデルでは Reflux と Middle Product のモデルが欠けていることが原因だと考えられる。特に、Reflux モデルの欠如は、同定に使用した運転データにおいて Reflux が大きく変更されていることからモデル精度に大きく影響していたと考えられる。Feed に対する Internal Reflux to Middle Product Ratio の同定されたモデルは現行のモデルとは方向が逆になっている。プロセスの観点から、現行モデルの方向が正しいと考えられ、通常の運転データでは、同定を行うためのデータの質（ステップサイズ、測定値にノイズが大きい）が足りなかったと思われる。

### 3.7.3 再同定結果

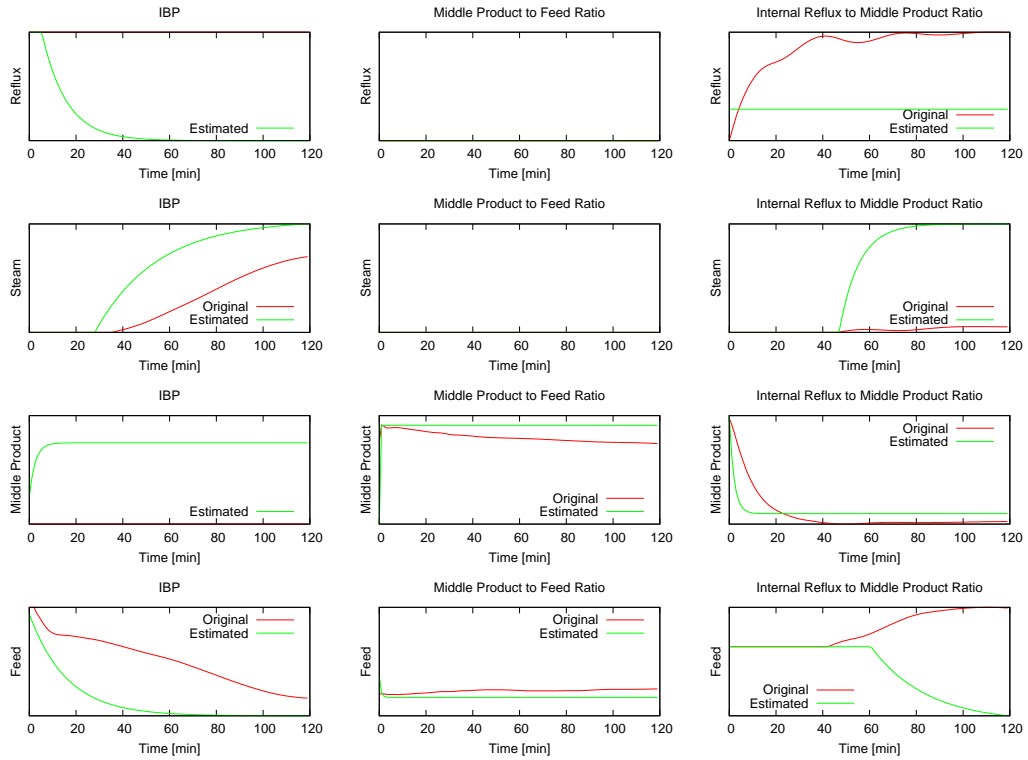
モデル予測制御による制御を実行した上で、再同定のためのステップテストを行った。そのステップテストにより得られた同定用の実運転データを図 3.25 に示す。採集したステップテストデータを用いて、本提案手法による閉ループモデル同定を行った。このステップテストデータを元に再同定を行ったモデルと閉ループステップテストを実行する際に使用したモデル予測制御のオリジナルモデルの比較結果を図 3.26 に示す。そして、図 3.25 に実運転データとモデルによるプロセス出力推定値の比較を示す。実運転データとのモデル精度  $\epsilon$  を以下の式で求めた。

$$\epsilon = \frac{1}{n_d} \sum_{i=1}^{n_d} \left( 1 - \frac{|y_i - \hat{y}_i|}{3\sigma} \right) \times 100 \quad (3.36)$$

ここで、 $n_d$  はデータのサンプル数、 $y$  は実運転データ、 $\hat{y}$  はダイナミックモデルによるプロセス出力推定値、 $\sigma$  は実運転データの標準偏差である。実運転データとのモデル精度を表 3.11 にまとめた。



3.21: Operation Data for Accuracy Check of the Original Models



3.22: Evaluation of the Original Models

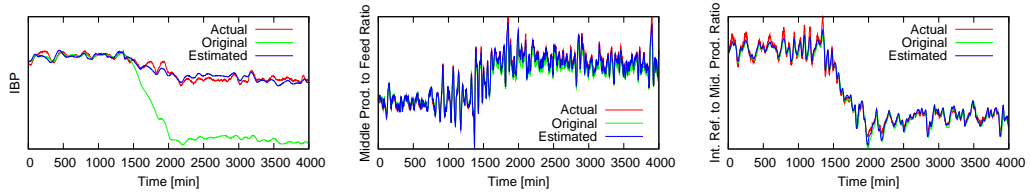


図 3.23: Estimation Comparison between the Original Models and the Estimated Models

Middle Product to Feed Ratio と Internal Reflux to Middle Product Ratio については、図 3.25 と表 3.11 から再同定により明らかにダイナミックモデルの精度の向上を確認することができた。IBP については、表 3.11 からはオリジナルのダイナミックモデルの方が精度が高いことになる。しかし、図 3.24 に各時刻における IBP のモデル精度の推移を示したところ、オリジナルモデルでは精度にばらつきがあるのに対して、同定モデルではモデル精度が徐々に減少していることがわかる。このモデル精度の減少は、ダイナミックモデルの同定に取り込まれていない未測定外乱の影響でバイアスが発生していると考えられ、このバイアスを考慮したモデル精度のばらつきを考えると、同定モデルのモデル精度が向上していると考えられる。これは、図 3.26 に示すように再同定により IBP の Reflux と Middle Product にモデルを追加して改善が行われたと考えられる。

表 3.11: Comparison of Prediction Fidelity  $\epsilon$

	$y_1$	$y_2$	$y_3$
Original	56.39%	92.75%	89.32%
New	40.20%	97.74%	97.76%

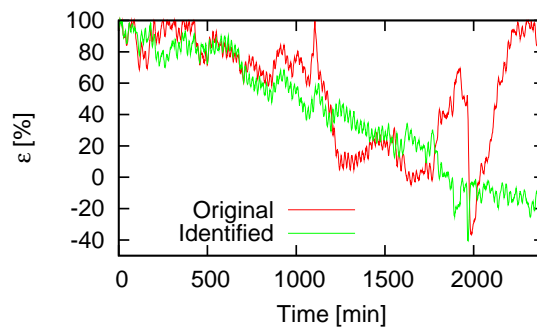


図 3.24: Comparison of IBP Prediction Fidelity



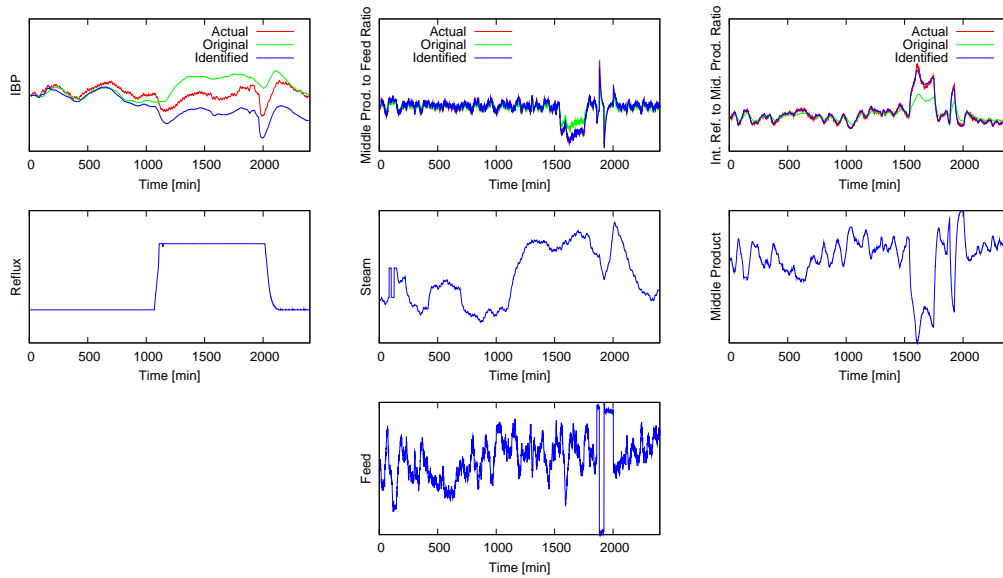


Figure 3.25: Operation Data for Identification by Closed-loop Step Testing and Comparison of Model Output between Original Models and Identified Models

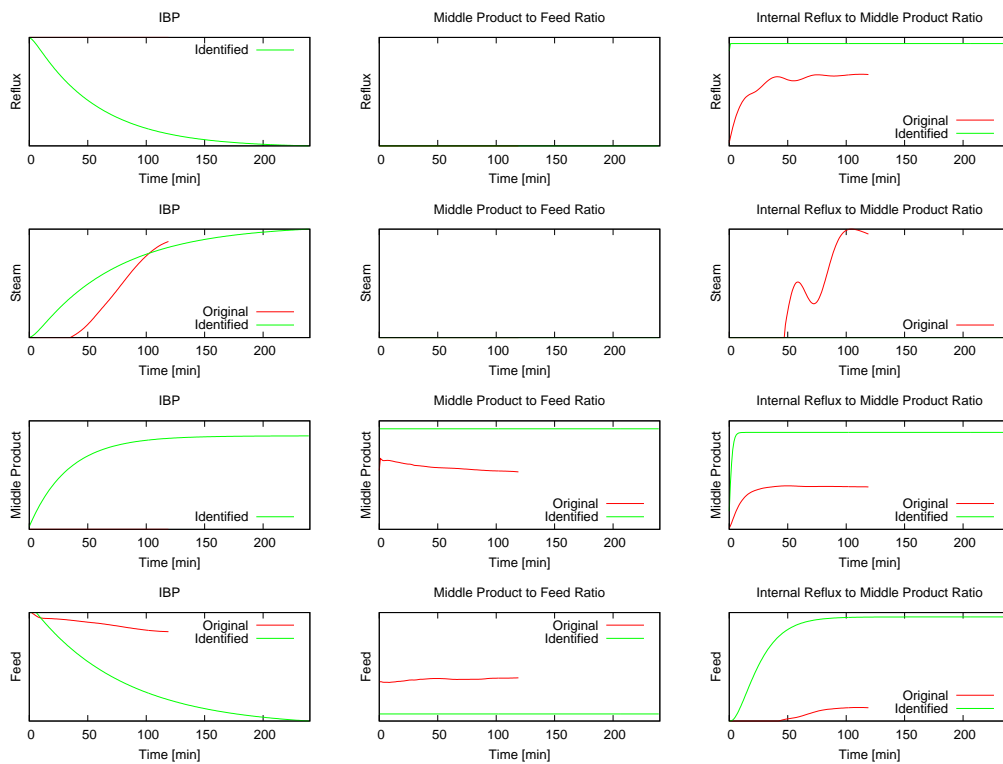


Figure 3.26: Comparison between Original Models and Identified Models

### 3.7.4 試運転

この節では、閉ループ運転データを用い新たに同定されたモデルを使用して、MPC の試運転を行った結果を示す。図 3.27 の 2200-4500 分の部分に同定されたモデルでの MPC の試運転結果を示す。図 3.27 の 0-2200 分の部分にステップテストで使ったオリジナルモデルでの試運転結果を比較のために載せた。IBP( $y_1$ ) と Middle Product to Feed Ratio ( $y_2$ ) は MPC の設定値タイプの被制御変数であり、Internal Reflux to Middle Product Ratio( $y_3$ ) は MPC の制約タイプの被制御変数である。そこで、表 3.12 に設定値タイプの変数である  $y_1$  と  $y_2$  について、同定されたモデルと以前に使われたモデルに対する設定値周りの標準偏差をまとめた。ステップテスト時と試運転時において Feed 流量にかなりの変更が見られるが、同定によりモデル更新を行ったことで MPC の制御性能が向上したと考えられる。それから、操作変数としての Steam 流量の動きが変わったことが確認できる。

表 3.12: Comparison of Standard Deviation around the Setpoint

	$y_1$	$y_2$
Original	1.058	0.0198
Identified	0.585	0.0171

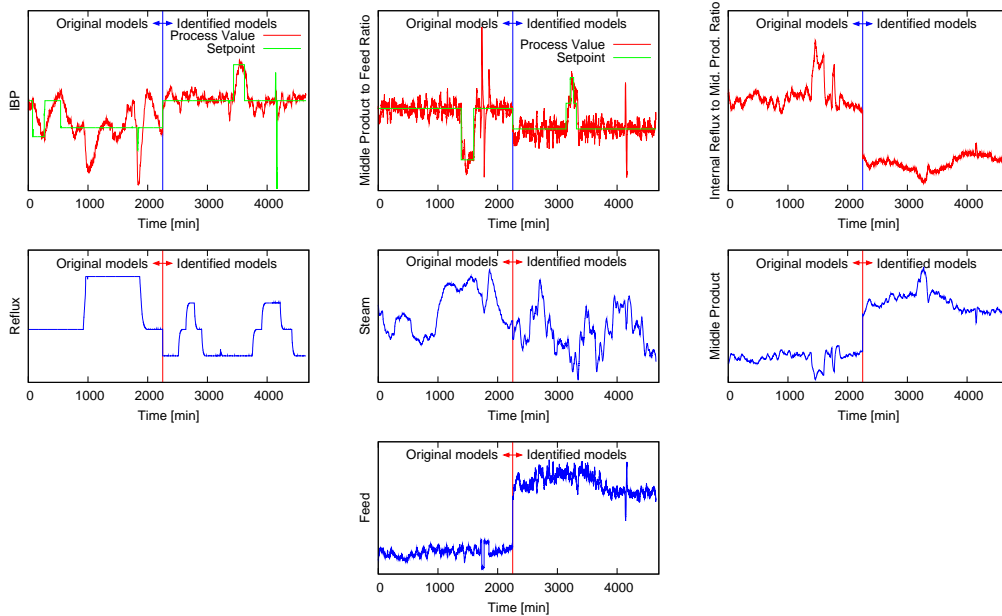


図 3.27: MPC Commissioning by the Identified Models

### 3.8 結言

プロセス制御において、PID コントローラとモデル予測制御の性能監視とメンテナンスはコントローラの性能を維持するためにはとても大切な問題である。そこで、本研究では、性能監視およびメンテナンスのベースとなるモデル同定手法に関して、無駄時間に対して仮定を使用することなしに伝達関数を変換した ARX モデルをベースとしたモデル同定手法により、直接伝達関数のモデルパラメータを求めることを提案した。いくつかのシミュレーションの結果を通して、本手法は以下のような特徴を有していることを確認できた。

- 解析解に基づいた簡単なアルゴリズムで簡単に使うことができる
- モデル同定のため収集されたデータのノイズフィルターの役目を果たす
- 同定されたモデルのスモーthingができる
- モデル同定のためのステップテストを短縮できる
- 閉ループステップテストにより精度のよいモデル同定が可能である
- モデル予測制御のモデル精度および制御性能改善ができる

さらに、この手法を商業運転中の製油所の蒸留塔に適用されているモデル予測制御のモデルメンテナンス問題に適用した。閉ループ応答を利用し本手法によるモデル同定の結果は以前のモデルに比べてモデル精度を改善することができた。そして、新しく同定されたモデルを使用してのモデル予測制御の試運転の結果もまたよりよい制御性を実現できた。本提案する手法は、モデル予測制御のモデルメンテナンス法としてかなり有効であることを確認した。

## 使用記号

$b_i$	= moving average coefficient	[-]
$c_i$	= coefficient of the Laguerre Polynomials	[-]
$d(k)$	= disturbance	[-]
$d_i$	= rearranged constant of numerator	[-]
$D$	= dead time	[s]
$f(t)$	= unit step response function without dead time	[-]
$G(s)$	= transfer function in the Laplace transform	[-]
$G(z)$	= transfer function in the Z transform	[-]
$h(t)$	= unit impulse response function	[-]
$J$	= performance index	[-]
$k$	= discrete time by sampling time	[-]
$K$	= process gain	[-]
$l_i$	= the Laguerre Polynomial	[-]
$L$	= discrete dead time by sampling time	[-]
$m$	= number of auto regression coefficient	[-]
$M$	= number of partial fraction	[-]
$n$	= number of moving average coefficient	[-]
$n_d$	= number of data	[-]
$n_u$	= number of process input	[-]
$N$	= order of the Laguerre Polynomials	[-]
$p$	= time scaling factor	[-]
$q_i$	= constant of numerator	[-]
$r_i$	= rearranged constant of denominator	[-]
$R$	= remainder of $D/\Delta t$	[s]
$s$	= complex variable in the Laplace transform	[-]
$s_i$	= constant of denominator	[-]
$SR_i$	= actual unit step response	[-]
$\hat{S}R_i$	= estimated unit step response	[-]
$u(k)$	= process input	[-]
$U(s)$	= process input in the Laplace transform	[-]
$y(k)$	= process output	[-]
$y_m(k)$	= output of process model	[-]
$Y(s)$	= process output in the Laplace transform	[-]
$z$	= complex variable in the Z transform	[-]
$\alpha_i$	= auto regression coefficient	[-]
$\Delta t$	= sampling time	[s]

$\epsilon$	= model fidelity	[-]
$\zeta$	= damping ratio	[-]
$\boldsymbol{\theta}$	= vector of transfer function parameters	[-]
$\tau$	= process time constant	[s]
$\tau_0$	= lead time	[s]
$\tau_1$	= time constant of 2nd order transfer function	[s]
$\tau_2$	= time constant of 2nd order transfer function	[s]
$\omega$	= fixed angular frequency	[rad/s]

## 第4章 ARX モデルを使用したモデル予測制御

### 4.1 緒言

多変数モデル予測制御の産業応用は石油精製、石油化学、化学分野のプロセス産業を中心として 1970 年代の後半、フランスの ADERSA 社の IDCOM [6] とアメリカの Shell Oil 社の DMC [7] が線形モデル予測制御として販売開始されて広く普及するようになった。現在では、様々なモデル予測制御パッケージが市販されており、石油精製、石油化学、化学分野だけではなく、医薬品、食品、ガスプロセス、用役プラント等に適用されるとともに、非線形を取り扱うモデル予測制御も開発され適用が進んでいる [18]。モデル予測制御は、その名の通り、実プロセスを模擬したプロセスモデルを規範として将来の被制御変数の予測を行い、被制御変数の設定値をもとに操作変数を決定している。プロセスモデルの表現方法としては、定常モデル (Steady State Model) と動特性モデル (Dynamic Model) に大別され、前者はプロセスプラント全体の運転最適化制御または多変数モデル予測制御の LP 最適化に使用される。しかし、定常モデルの主な用途はプラント設計である。プロセス制御はプロセスプラントの動的な動きの制御を行わなければならないので、プロセス制御の一つであるモデル予測制御は後者を採用している。動特性モデルとしては、さらに、ブラックボックスモデル (Black Box Model) と第 1 原理モデル (First Principle Model) に分けられる。前者は、モデルのパラメータを同定するために制御対象である実プラントの運転データを用いるものであり、モデルパラメータの一つ一つに物理的及び化学的意味を持たない。後者は、物理的及び化学的原理をベースとしたモデルであり、モデルパラメータの一つ一つに密度、粘度、反応速度定数等の意味がある。すべての物理的及び化学的現象を数式で表すことが難しいことから実プラントの運転データでモデルを同定するブラックボックスモデルがモデル予測制御で使用するモデルの主流となっている。ブラックボックスモデルの主なものを表 4.1 に示す。

モデル予測制御では上述したいずれかのモデルを用いて、未来を予測しながら制御を行っていくことになるが、未来の予測方法には、さらに 2 種類のアプローチがある。一つは独立モデルと呼ばれ、もう一方はリセットモデルと呼ばれる方法である [8]。1 次遅れ系伝達関数の ARX モデル (Auto Regressive model with eXogenous input) を例にすると、独立モデルは式 (4.1) で表され、リセットモデルは式 (4.2) で表される。

$$y_m(k+1) = y_0 + a(y_m(k) - y_0) + b(u(k) - u_0) \quad (4.1)$$

表 4.1: Various Black-Box Models

モデル名
インパルス応答モデル
ステップ応答モデル
ARX モデル
伝達関数モデル
状態空間モデル
ニューラルネットワークモデル

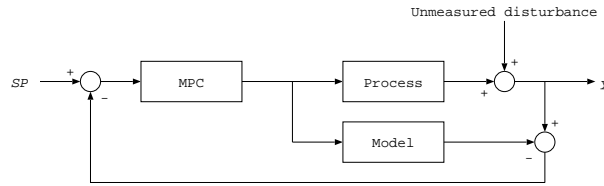


図 4.1: Control scheme of Independent (Non-aligned) Model

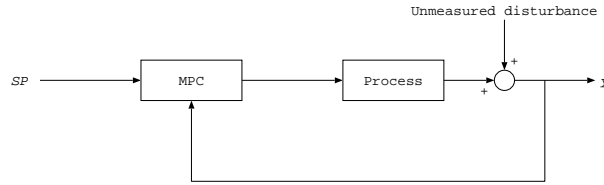


図 4.2: Control scheme of Re-aligned Model

$$y_m(k+1) = y(k) + a(y(k) - y(k-1)) + b(u(k) - u(k-1)) \quad (4.2)$$

$$a = e^{-\frac{\Delta t}{\tau}}, b = K_p(1 - a)$$

ここで、 $a$  は AR(自己回帰、Auto Regression) 項モデルパラメータ、 $b$  は MA(移動平均、Moving Average) 項モデルパラメータと呼ばれる。独立モデルを使用したモデル予測制御スキームを図 4.1 に、リセットモデルを使用したモデル予測制御スキームを図 4.2 に示す。

独立モデルを使用したモデル予測制御は、直接制御対象プロセスを制御せずに、独立モデルからの出力をフィードバックして、独立モデルを制御する。モデル誤差の修正としては、制御対象プロセスと独立モデルからの出力の偏差を計算して、その偏差により設定値を修正することで実現する。構造としては、IMC(Internal Model Control, [5])と同じである。一方、リセットモデルを使用したモデル予測制御は、制御対象プロセスの出力をフィードバックするだけのアルゴリズムである。参照軌道を使用したモデ

ル予測制御を例にとると、独立モデルの制御アルゴリズムは式 (4.3) で表され、リセットモデルの制御アルゴリズムは式 (4.4) で表される。

$$u(k) = u_0 + \frac{(1 - a_r)(SP - y(k)) + (1 - a)(y_m(k) - y_0)}{b} \quad (4.3)$$

$$\begin{aligned} u(k) &= u(k-1) \\ &+ \frac{(1 - a_r)(SP - y(k)) - a(y(k) - y(k-1))}{b} \end{aligned} \quad (4.4)$$

$$a_r = e^{-\frac{\Delta t}{\tau_r}}, b = K_p(1 - e^{-\frac{\Delta t}{\tau}})$$

式 (4.3) 及び (4.4) の詳細は、それぞれ、4.3.1 節と 4.2 節で説明する。本論文で提案するモデル予測制御は、式 (4.4) である。式 (4.3) は、式 (4.1) によりプロセス出力を逐次計算する必要がある。式 (4.4) は式 (4.1) とは違い実測値による AR 項が存在することで未測定外乱に対して応答の速い制御構造をしている。本論文では、未測定外乱に対して強いロバスト性を有するモデル予測制御を提案する。このモデル予測制御では、未来を予測する動特性モデルに ARX モデルとインクリメンタルモデルを組み合わせた線型モデルを適用して構築する。このモデル予測制御を ARX モデル予測制御と呼ぶことにする。

## 4.2 ARX モデル予測制御アルゴリズム

### 4.2.1 ARX モデル

ARX モデルは次の式 (4.5) で表される。

$$y(k+1) = \sum_{i=1}^m \alpha_i y(k+i-1) + \sum_{i=1}^n b_i u(k+i-1) + e(k) \quad (4.5)$$

本制御アルゴリズムを構築するために、 $e(k)$  を無視し、次式のような AR 項が 1 つの ARX モデルを適用する。次式は 1 次の ARX モデル (1st Order ARX Model) と呼ばれる。

$$\begin{aligned} y_m(k+1) &= y(k) + \alpha\{y(k) - y(k-1)\} \\ &+ \sum_{i=1}^n b_i\{u(k+i-1) - u(k+i-2)\} \end{aligned} \quad (4.6)$$

式 (4.6) の右辺は  $y$  も  $u$  も 1 サンプル前の値との偏差をとっているので、インクリメンタルモデルとも呼ばれる。このインクリメンタルモデルは、初期値を保持する必要



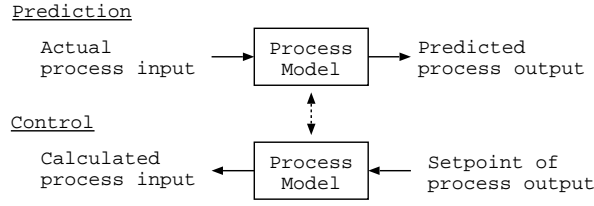


図 4.3: Relationship between prediction and control for process model

がなく、予測値  $y_m(k+1)$  がプロセス入出力の実測値を使用して推定できるので、モデル予測制御のリセットモデルとして使用できる。代表的な伝達関数の ARX モデル係数計算方法を付録 A に示す。1 次遅れ系伝達関数の ARX モデル係数に関して AR 項を  $\alpha = e^{-\frac{\Delta t}{\tau}}$  と決定すると MA 項は  $b_2$  以降 0 となり、式 (4.1) または (4.2) が得られる。ARX モデルは時系列モデルであり、他の時系列モデルには表 4.1 のステップ応答モデル、インパルス応答モデルがあり、これらも伝達関数からモデルパラメータを決定できる。

#### 4.2.2 制御アルゴリズム

一般的なモデル予測制御の基本的な制御アルゴリズムの概念図を図 4.3 に示す。将来予測に関して、プロセス入力を使用してプロセスモデルによりプロセス出力の将来を計算する。それに対して、制御に関してはその逆のプロセス出力の設定値を使用してプロセスモデルによりプロセス入力を計算する。これがモデル予測制御の基本的な制御アルゴリズムである。本論文で提案する ARX モデル予測制御はプロセスモデルとして式 (4.6) を用いるものである。サンプリング時間  $\Delta t$  に対して 1 ステップ先のプロセス出力値を予測しているので、1 ステップ先のプロセス出力値の設定値を決定すれば、ARX モデルにより制御のためのプロセス入力を計算できる。このような 1 ステップ先の予測を使用して制御を行う制御方法を 1 段予測制御アルゴリズムと呼ぶ。ARX モデル予測制御におけるプロセス出力の設定値決定方法を図 4.4 に示す。測定された現在値  $y(k)$  を起点にプロセス出力の設定値  $SP$  へ向けて 1 次遅れの参照軌道 (Reference trajectory) を引く。参照軌道において現在から 1 ステップ先の点を式 (4.6) との一致点 (Coincident point) とするとその点  $y_r$  は式 (4.7) で表される。

$$y_r = y(k) + \{SP - y(k)\}(1 - a_r) \quad (4.7)$$

よって、式 (4.6) と (4.7) より ARX モデル予測制御アルゴリズムとして次の式 (4.8) が得られる。

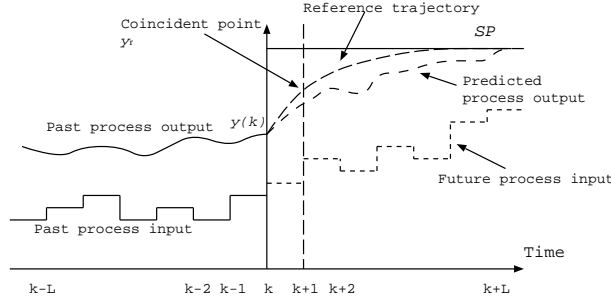


図 4.4: Conceptual scheme of MPC

$$\begin{aligned}
 u(k) &= u(k-1) \\
 &+ \frac{\{SP - y(k)\}(1 - a_r) - \alpha\{y(k) - y(k-1)\}}{b_1} \\
 &- \frac{\sum_{i=2}^n b_i\{u(k+i-1) - u(k+i-2)\}}{b_1}
 \end{aligned} \tag{4.8}$$

1 次遅れ系伝達関数の ARX モデル係数に関して  $\alpha = e^{-\frac{\Delta t}{\tau}}$  と選択すると MA 項は  $b_2$  以降 0 となり、式 (4.8) から式 (4.4) が得られる。

## 4.3 PCR と ARX モデル予測制御の制御性比較

### 4.3.1 PCR アルゴリズム

PCR[9] は、独立モデルの式 (4.1) をプロセスモデルとして使用し、式 (4.7) の参照軌道を採用する 1 段予測制御である。式 (4.7) の  $y(k)$  は  $y_m(k)$  になり、式 (4.1) と (4.7) より式 (4.9) が得られる。

$$u(k) = u_0 + \frac{(1 - a_r)(SP - y_m(k)) + (1 - a)(y_m(k) - y_0)}{b} \tag{4.9}$$

式 (4.9) は、モデル誤差を全く含まない理想系プロセスを制御する場合のアルゴリズムである。実際は、プロセスモデルと実プロセスの間にはモデル誤差が存在する。それは式 (4.10) で計算される。

$$e_m(k) = y(k) - y_m(k) \tag{4.10}$$

よって、図 4.1 によりプロセス出力設定値  $SP$  をモデル誤差  $e_m(k)$  で修正すると式 (4.11) が求まる。式 (4.10) を式 (4.11) に代入することで、PCR アルゴリズム式 (4.3) が得られる。

$$u(k) = u_0 + \frac{(1 - a_r)(SP - e_m(k) - y_m(k)) + (1 - a)(y_m(k) - y_0)}{b} \quad (4.11)$$

### 4.3.2 制御性比較

PCR と ARX モデル予測制御を表 4.2 の 1 次遅れ系伝達関数モデルパラメータで制御性比較を行った。本シミュレーションにおいて、モデルパラメータに無駄時間  $D$  が存在するので、式 (4.6) に代えて無駄時間を含む次式を適用する。

$$y_m(k+1) = y(k) + \alpha\{y(k) - y(k-1)\} + \sum_{i=1}^n b_i\{u(k-L+i-1) - u(k-L+i-2)\} \quad (4.12)$$

ここで、 $L$  は  $D/\Delta t$  の整数部分である。シミュレーションの初期値及びモデル予測制御のチューニングパラメータをそれぞれ表 4.3 と表 4.4 に示す。ARX モデル予測制御の AR 項は  $\alpha = e^{-\frac{\Delta t}{\tau}}$  として MA 項の  $b_2$  以降の項を 0 にするように決定した。つまり、PCR の式 (4.3) と ARX モデル予測制御の式 (4.4) による比較である。

表 4.2: Model parameters of 1st order transfer function

1st Order parameter	Value	Unit
$K_p$	2	-
$\tau$	15	s
$D$	4	s

比較結果を図 4.5 に示す。シミュレーション時間 50 秒で設定値を +5 変更した。設定値変更に対して PCR と ARX モデル予測制御は全く同じ応答を示している。シミュレーション時間 450 秒でプロセス出力に +5 のステップ状未測定外乱を与えた。この場合には、PCR と ARX モデル予測制御の応答に明確な相違が確認できた。図より明らかに ARX モデル予測制御は PCR に比べて応答の速い制御結果が得られた。ARX モデル予測制御では、未測定外乱を与えたと同時にプロセス出力を操作してフィードフォワードアクションを行っている。それに対して、PCR ではシミュレーション結果からフィー

表 4.3: Initial conditions of simulation

Parameter	Value
$u$	100
$y$	200

表 4.4: Tuning parameters of MPC

Tuning parameter	Value	Unit
$\Delta t$	1	s
$\tau$	50	s
$\alpha$	$e^{-\frac{\Delta t}{\tau}}=0.9355$	-

ドバックのみのアクションであることがわかる。PCR アルゴリズムである式 (4.3) からも、未測定外乱に対しては  $y(k)$  が変化してからのフィードバック制御のみの対応になることがわかる。ARX モデル予測制御では式 (4.8) より実測値による AR 項が制御アルゴリズムの中に存在するので、未測定外乱が加わったと同時にフィードフォワードアクションが行われることがわかる。次の制御ステップからは、 $\alpha\{y(k) - y(k-1)\}$  は無駄時間のために  $y(k) = y(k-1)$  となるので、プロセス入力  $u$  はフィードバックとしての操作になる。このことより、未測定外乱の補償機能は、未測定外乱が入った時のプロセス入力 1 ステップの操作で決まっていることになる。AR 項による制御応答検討は 4.4 節で行う。

次に、 $0.1 \text{ 秒}^{-1}$  のランプ状未測定外乱を与えた比較結果を図 4.6 に示す。PCR と ARX MPC とともにオフセットがあるのがわかる。ARX MPC は PCR に比べて、未測定外乱に対して操作が速いのでオフセットが小さくなっている。ランプ状の外乱に対しては、オフセットは小さくすることができるもののオフセットが残るので、これ以降は、ステップ状の未測定外乱に限定して検討を行うものとする。

#### 4.4 未測定外乱に対するチューニング

図 4.5 に示すように ARX モデル予測制御は未測定外乱に対して応答の速い制御アルゴリズムであることが確認できた。しかし、ARX モデル予測制御では、反応熱のような未測定外乱だけでなく、常時入ってくるシステムノイズ及び観測ノイズも未測定外乱として取り扱うことになり、図 4.5 のようなプロセス入力の急激な操作は通常の制御性を悪くしてしまう可能性がある。そこで、未測定外乱に対してもチューニング

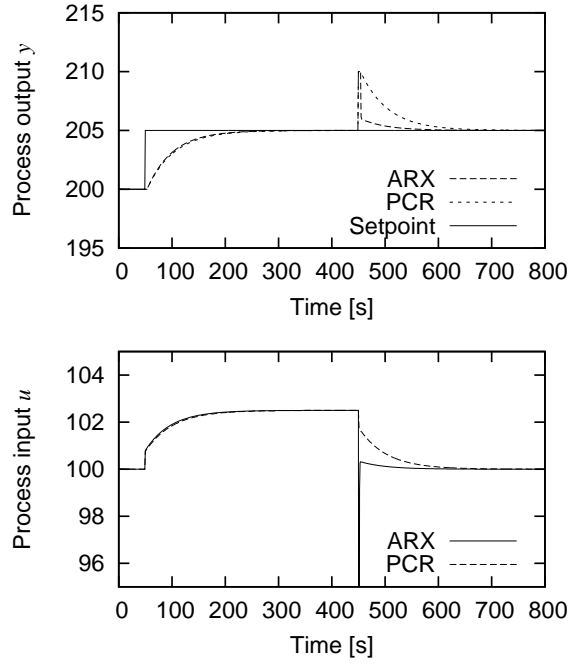


図 4.5: Comparison of control performance between PCR and ARX MPC

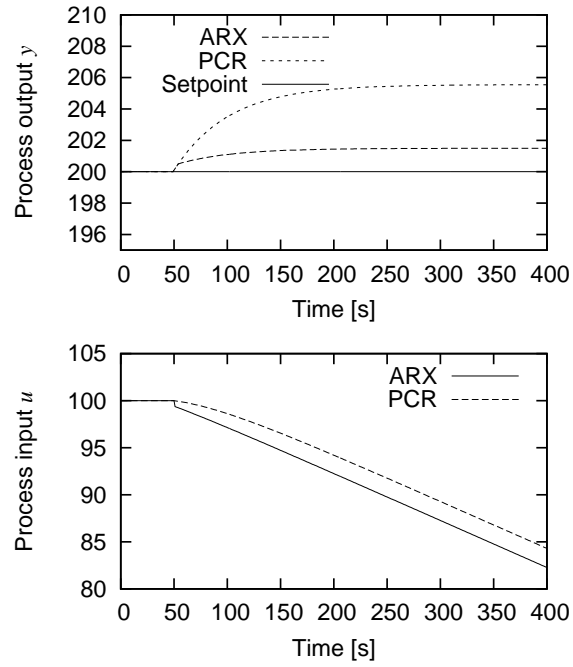
が必要である。ARX モデル予測制御では ARX モデルの AR 項が未測定外乱に対するチューニングパラメータとして使用可能である。4.3 節のシミュレーションで使用した表 4.2 のパラメータを使用して、図 4.7 に AR 項  $\alpha$  による MA 項  $b_i$  パラメータの影響を示す。打ち切り次数  $n$  は 100 とした。AR 項  $\alpha$  が小さくなるとともに、過去からの MA 項  $b_i$  の影響が大きくなっていくことがわかる。

ここで、AR 項  $\alpha$  変更に対するステップ応答への影響を確認した結果を図 4.8 に示す。AR 項  $\alpha$  によるプロセスモデルの精度は一定に保たれている。これは、打ち切り次数  $n$  をプロセス時定数に対してかなり大きめにとったために得られた結果である。図 4.7 より  $\alpha$  が小さくなればなるほど、打ち切り次数  $n$  の数を多く取らなければダイナミクスへ影響を及ぼしてしまう。打ち切り次数  $n$  に対する定常ゲインは次式で表される。

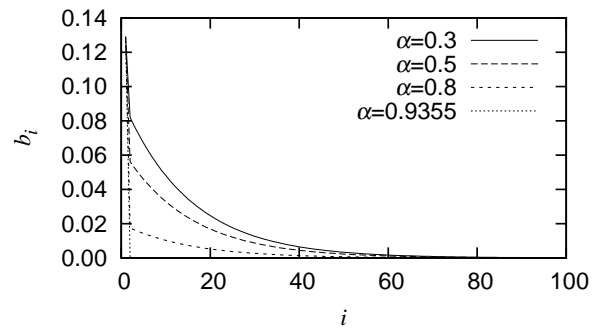
$$y_{ss}(n) = \frac{\sum_{i=1}^n b_i}{1 - \alpha} \quad (4.13)$$

打ち切り次数  $n$  は、式 (4.13) の値をプロセスゲインで割った比をとり 0.99 を目安とする。表 4.5 に AR 項  $\alpha$  に対して比率 0.99 を満足する打ち切り次数  $n$  をまとめた。

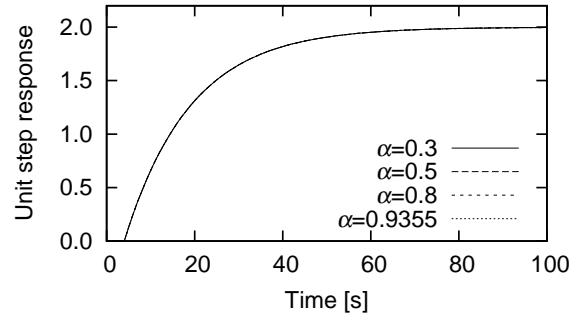
ここで、4.3 節と同様のシミュレーションを行い、AR 項  $\alpha$  変更に対して制御性への影響を確認した結果を図 4.9 に示す。



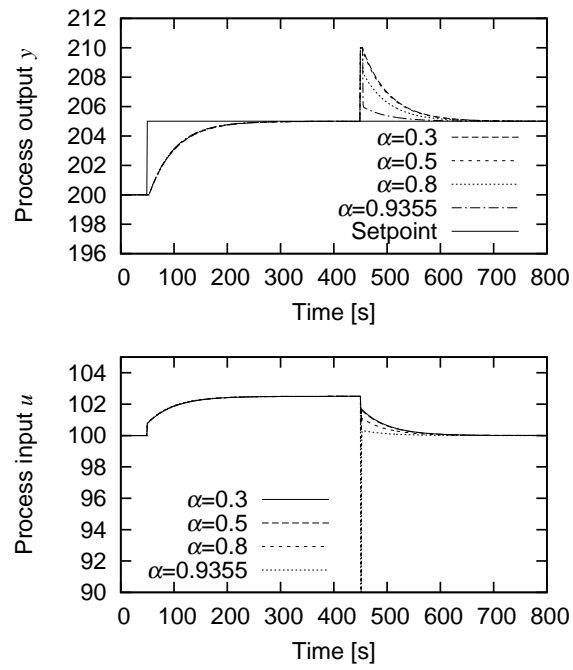
⊗ 4.6: Comparison of control performance between PCR and ARX MPC for ramp disturbance



⊗ 4.7: MA term parameters  $b_i$  for AR term  $\alpha$



4.8: Step response for AR term  $\alpha$



4.9: Tuning results for AR term  $\alpha$

表 4.5: Finite number of MA coefficient for AR term

$\alpha$	$n$
0.3	69
0.5	69
0.8	65
0.9355	1

設定値変更に対しては AR 項  $\alpha$  を変更しても影響は見られない。これは打ち切り次数  $n=100$  として AR 項  $\alpha$  に対してダイナミクスが変わらないようにしているためである。未測定外乱に対しては図より、AR 項  $\alpha$  が大きいほど制御応答が速いことがわかる。このことは、式 (4.8) より AR 項  $\alpha$  が大きいほどプロセス入力  $u$  が大きく操作されることから理解できる。AR 項  $\alpha$  が 0.3 と 0.5 ではほぼ同じ応答が得られている。AR 項  $\alpha$  の影響度合いを未測定外乱の大きさに対してどの程度設定値付近まで戻すことができるかという指標  $RR$  で表すと、式 (4.8) と (4.12) より以下の式を導くことができる。

$$RR = \begin{cases} 1 + a_r + \alpha & \text{at } D = 0 \\ \frac{\alpha^{D+1}-1}{\alpha-1}(1 - a_r) + \alpha^{D+1} & \text{at } D \neq 0. \end{cases} \quad (4.14)$$

式 (4.14) により AR 項  $\alpha$  による戻り率  $RR$  を表 4.6 に示す。

表 4.6: Return ratio for unmeasured disturbance

$\alpha$	$RR$
0.3	0.03
0.5	0.07
0.8	0.39
0.9355	0.80

この表 4.6 より 0.3 と 0.5 はほぼ同じ応答になることがわかる。AR 項  $\alpha$  は式 (4.14) により  $RR$  を決定することで求めることが可能である。

AR 項  $\alpha=0$  とした場合、MA 項  $b_i$  だけが残ることになりこれは表 4.1 に示したインパルス応答モデルになる。この場合の PCR と ARX モデル予測制御を比較した結果を図 4.10 に示す。

AR 項の効果がなくなった場合は、PCR と ARX モデル予測制御の制御性は全く同じになることがわかる。



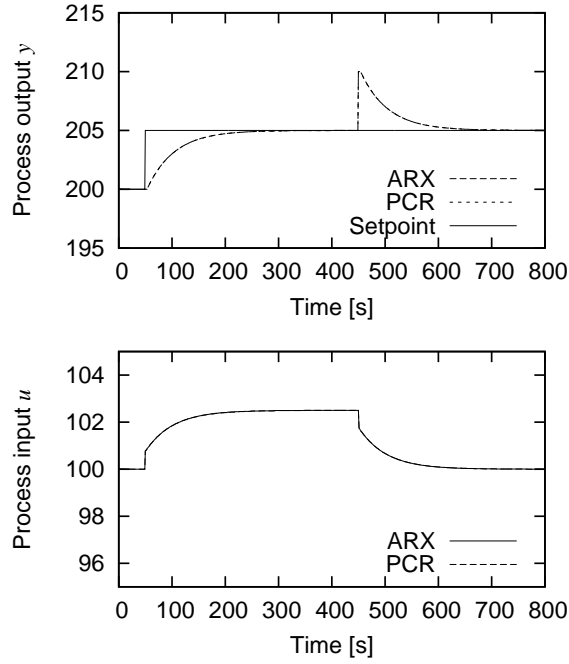


図 4.10: Comparison of control performance between PCR and ARX MPC at  $\alpha=0$

## 4.5 様々な応答プロセスへの適用

プロセス産業でのプロセス応答は 1 次遅れ+無駄時間系の伝達関数で近似できるものから、複雑な応答を示すものが存在する。ARX モデル予測制御では、様々な応答のプロセスを伝達関数近似することでそのプロセスを制御することが可能になる。本節では、様々な応答を持つプロセスに ARX モデル予測制御を適用した結果を示す。シミュレーションの初期値は表 4.3、ARX モデル予測制御の Tuning parameter は表 4.4 を使用する。伝達関数としては、付録 A に示すような 2 次遅れ系 (Time constant と Damping ratio) と積分系を使用し、プロセスを表現している。これらの伝達関数でプロセスの 90% は表現可能である。

### 4.5.1 逆応答を持つプロセス

蒸留塔の組成制御を行う場合、逆応答を伴うプロセスが存在する場合がある。組成制御以外にもまれに逆応答を伴うプロセスが存在するが、この逆応答プロセスは式 (4.15) の 2 次遅れ系 (Time constant) の伝達関数で表現することが可能である。

$$G(s) = \frac{K_p(\tau_0 s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad (4.15)$$

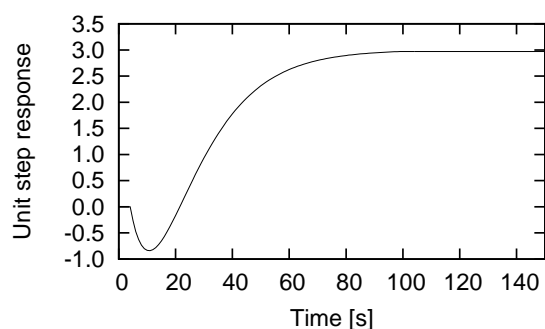


図 4.11: Inverse response process

表 4.7 に示す伝達関数パラメータを使用した場合の単位ステップ入力に対する応答を求めると図 4.11 のようになり、逆応答プロセスが表現できる。

表 4.7: Model parameters of inverse response process

2nd order parameter (Time constant)	Value	Unit
$K_p$	3	-
$\tau_0$	-15	s
$\tau_1$	10	s
$\tau_2$	15	s
$D$	4	s

ARX モデル予測制御によりこの逆応答プロセスを制御した結果を図 4.12 に示す。

シミュレーション開始 50 秒後に +5 の設定値変更を行い、550 秒後に +5 の未測定外乱をプロセス出力値に与えた。AR 項  $\alpha$  は、チューニングパラメータである参照軌道の時定数に変わって、チューニングパラメータの役目をしていることがわかる。逆応答の部分に影響して AR 項  $\alpha$  が大きい場合には、オーバーシュートが起こっていると考えられる。実プロセスで逆応答プロセスに ARX モデル予測制御を適用する場合、参照軌道の時定数と AR 項の両方を考慮したチューニングが必要だと考える。

#### 4.5.2 積分系プロセス

制御対象に欠かせないプロセスとしては、液面がある。これは積分系プロセスである。積分系プロセスは、液面だけではなく蒸留塔の温度にも表れる場合がある。この積分系プロセスは、式 (4.16) の伝達関数で表現することが可能である。

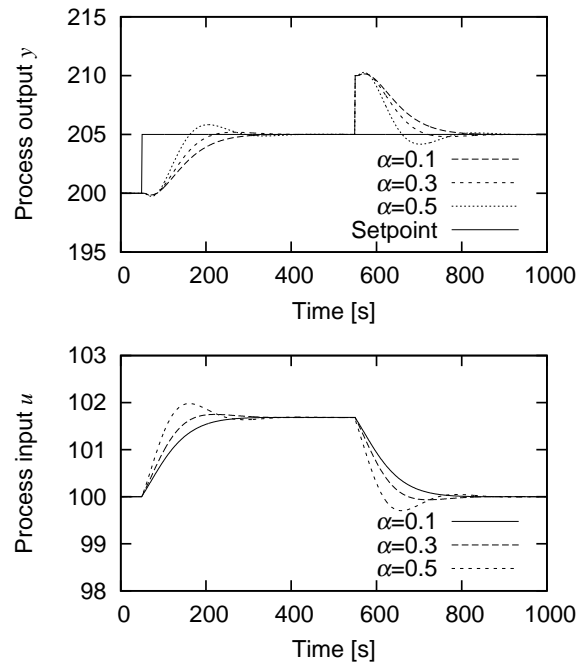


図 4.12: Control performance of inverse response process

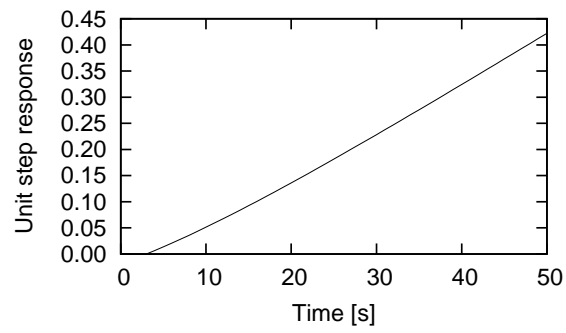
$$G(s) = \frac{K_p(\tau_0 s + 1)}{(\tau s + 1)s} \quad (4.16)$$

表 4.8 に示す伝達関数パラメータを使用すると図 4.13 の積分系プロセスが表現できる。

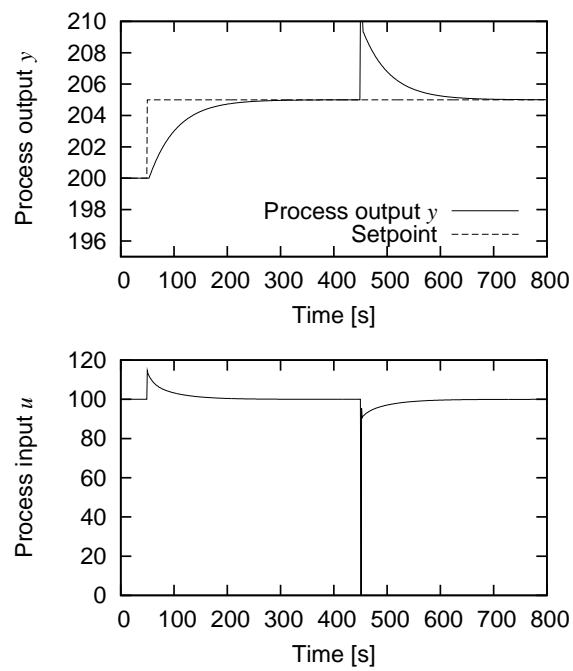
表 4.8: Model parameters of ramp process

Ramp parameter	Value	Unit
$K_p$	0.01	-
$\tau_0$	10	s
$\tau$	15	s
$D$	4	s

ARX モデル予測制御により積分系プロセスを制御した結果を図 4.14 に示す。この積分系プロセスモデルは AR 項  $\alpha=1$  とすることで ARX モデルで表現できる (付録 A を参照)。



☒ 4.13: Ramp process



☒ 4.14: Control performance of ramp process

シミュレーション開始 50 秒後に+5 の設定値変更を行い、550 秒後に+5 の未測定外乱をプロセス出力値に与えた。AR 項  $\alpha$  の値が大きいため未測定外乱に対してかなり急激なプロセス入力の変化が起こっている。このような操作は、プロセスに対する外乱になりかねないので、適切なプロセスによる検討が必要である。

### 4.5.3 特殊な応答プロセス

圧力が関係する変数、または、PID 制御の制御出力は、図 4.15 のような応答のプロセスになる場合がある。このような応答のプロセスは式 (4.17) の 2 次遅れ系 (Damping ratio) の伝達関数で表現することが可能である。

$$G(s) = \frac{K_p(\tau_0 s + 1)\omega^2}{s^2 + 2\zeta\omega s + \omega^2} \quad (4.17)$$

表 4.9 に示す伝達関数パラメータを使用すると図 4.15 の特殊な応答プロセスが表現できる。

表 4.9: Model parameters of special response process

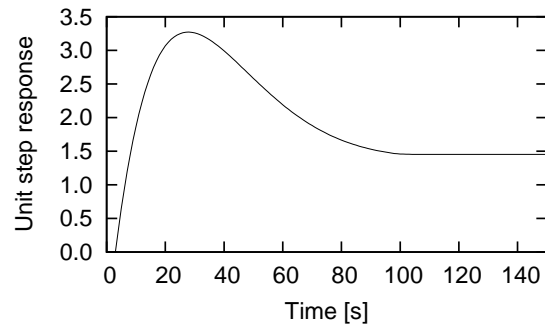
2nd order parameter (Damping ratio)	Value	unit
$K_p$	1.433	-
$\tau_0$	83.33	s
$\zeta$	0.8	-
$\omega$	0.054	rad/s
$D$	4	s

ARX モデル予測制御によりこの特殊な応答プロセスを制御した結果を図 4.16 に示す。

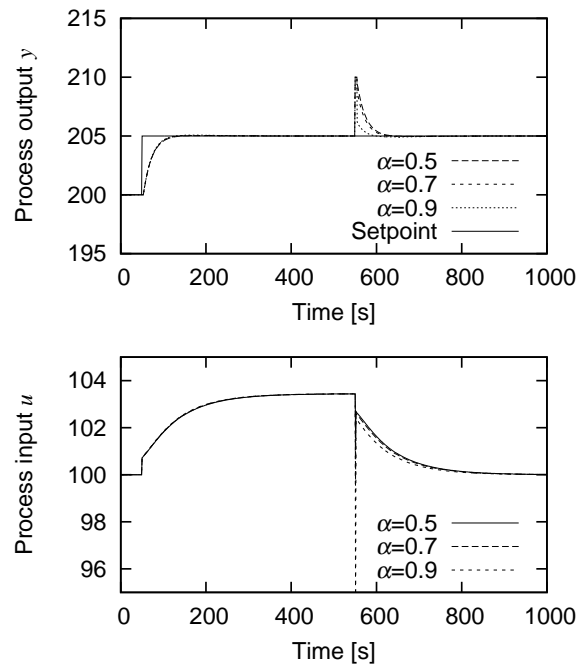
シミュレーション開始 50 秒後に+5 の設定値変更を行い、550 秒後に+5 の未測定外乱をプロセス出力値に与えた。AR 項  $\alpha=0.9$  では、未測定外乱に対して応答の速い制御を行っている。 $\alpha=0.5, 0.7$  では、ほぼ同じ制御応答を示している。

## 4.6 非線形プロセスへの適用

本節では、非線形プロセスへの適用の可能性を検討する。検討を行う制御対象プロセスは連続撹拌槽型反応器 (Continuous Stirred Tank Reactor, CSTR) を用い、この反応器の温度制御に対して ARX モデル予測制御を適用する。このプロセスは、製品の濃度及び反応器の温度の関係に対して強い非線形性を示すことが知られており、製品の品質を安定して一定に保つためには安定した反応器温度の制御が重要である。



☒ 4.15: Special response process



☒ 4.16: Control performance of special response process

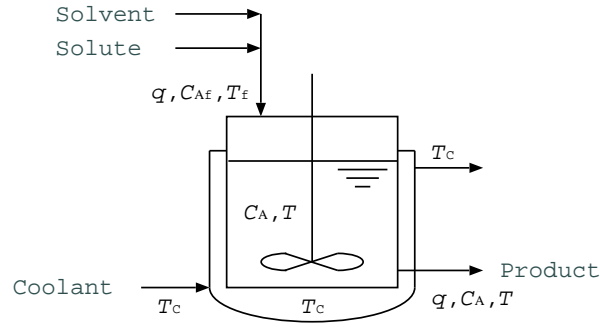


図 4.17: CSTR

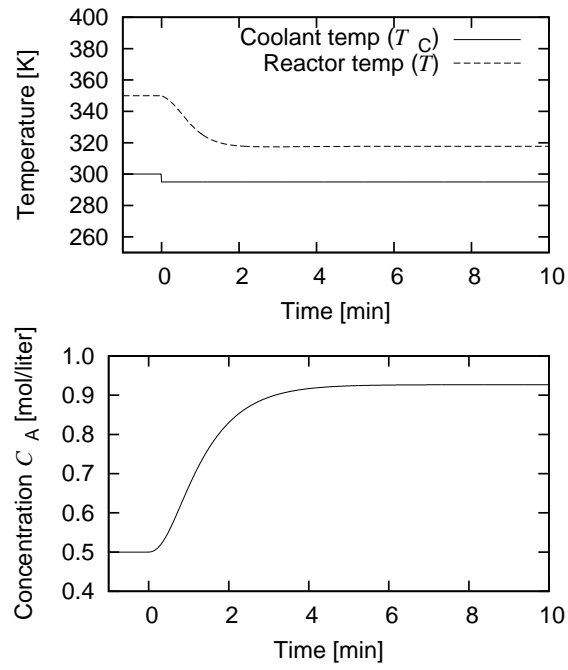
#### 4.6.1 制御対象プロセス

制御対象プロセスの概略図を図 4.17 に示す [10]。反応としては、原料 A を製品 B に反応させる 1 次可逆発熱反応を想定する。原料 A は溶媒 (製品 B) と混合され、反応器のフィードとして流量  $q$  を一定に保つようにし、濃度  $C_{Af}$  及び温度  $T_f$  で反応器に連続的に供給される。製品 B は、反応器の底部よりフィード流量と同じ流量  $q$  で抜き出される。この製品の品質は、原料 A の濃度で管理することとして、この濃度を  $C_A$  とする。ここで、反応に対して物性は変化せず、また、冷媒の流量は十分多いものとして反応器温度に対する冷媒の温度変化はないと仮定した。このプロセスの非線形モデルは、原料 A の濃度に対する物質収支式 (4.18) と反応器に対する熱収支式 (4.19) で表される。

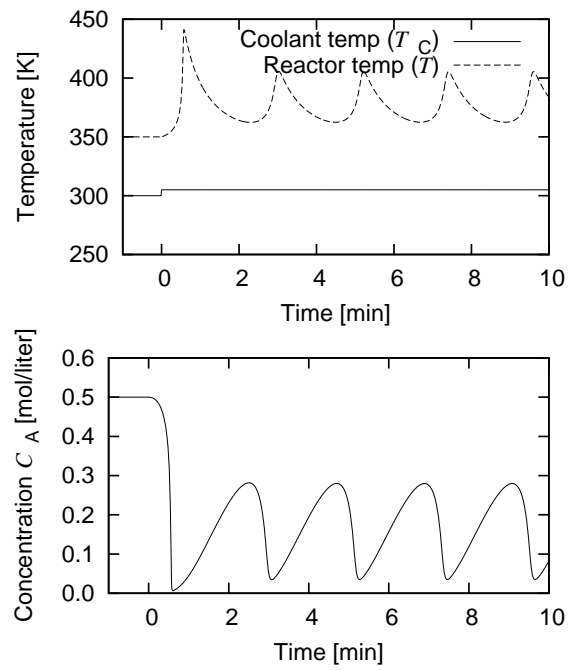
$$\frac{dC_A}{dt} = \frac{q}{V}(C_{Af} - C_A) - k_0 \exp\left(-\frac{E}{RT}\right) C_A \quad (4.18)$$

$$\begin{aligned} \frac{dT}{dt} = & \frac{q}{V}(T_f - T) + \frac{(-\Delta H)}{\rho C_p} k_0 \exp\left(-\frac{E}{RT}\right) C_A \\ & + \frac{UA}{V\rho C_p}(T_c - T) \end{aligned} \quad (4.19)$$

ここで、この反応器モデルのモデルパラメータを表 4.10、初期値を表 4.11 に示す。式 (4.18)、(4.19) より反応器温度  $T$  に対して強い非線形性があることがわかる。冷媒の温度を  $-5^\circ\text{C}$ 、 $+5^\circ\text{C}$  ステップ変化させたときの反応器温度及び原料 A の濃度の経時変化を図 4.18 と図 4.19 にそれぞれ示す。冷媒の温度を  $-5^\circ\text{C}$  変化させたときは、製品 B が原料 A に戻る反応が起こり、原料 A の濃度が高くなって、発熱反応による発熱がなくなる方向になるので、反応器温度がスムーズに定常状態になっている。逆に、冷媒の温度を  $+5^\circ\text{C}$  変化させたときは、発熱反応は急激に起こり、原料 A の濃度が急激に減少する。原料 A の濃度の減少に伴い発熱反応が減少するため、製品 B が再び原料 A に戻る反応が起こる。この図より非線形性が強く、反応器温度を一定に制御しない限り安定した製品が得られないことがわかる。



☒ 4.18: Effect of coolant temperature( $T_c:-5$ )



☒ 4.19: Effect of coolant temperature( $T_c:+5$ )



表 4.10: Model parameters of CSTR

Parameter	Value	Unit
$q$	100	$\ell/\text{min}$
$C_{\text{Af}}$	1	$\text{mol}/\ell$
$T_{\text{f}}$	350	K
$V$	100	$\ell$
$\rho$	1000	$\text{g}/\ell$
$C_{\text{p}}$	0.239	$\text{J}/(\text{g}\cdot\text{K})$
$(-\Delta H)$	$5 \times 10^4$	$\text{J}/\text{mol}$
$\frac{E}{R}$	8750	K
$k_0$	$7.2 \times 10^{10}$	$1/\text{min}$
$UA$	$5 \times 10^4$	$\text{J}/(\text{min}\cdot\text{K})$

表 4.11: Initial conditions of CSTR simulation

Parameter	Value	Unit
$T_{\text{c}}$	300	K
$C_{\text{A}}$	0.5	$\text{mol}/\ell$
$T$	350	K

#### 4.6.2 ARX モデル予測制御の CSTR への適用

式 (4.18)、(4.19) で表される CSTR モデルに対して ARX モデル予測制御では反応熱を未測定外乱として取り扱うために式 (4.19) を初期値  $T_{\text{i}}, T_{\text{ci}}, C_{\text{Ai}}$  に対する式 (4.20) に書き直す。

$$\begin{aligned} \frac{d(T - T_{\text{i}})}{dt} &= -\left(\frac{q}{V} + \frac{UA}{V\rho C_{\text{p}}}\right)(T - T_{\text{i}}) \\ &\quad + \frac{UA}{V\rho C_{\text{p}}}(T_{\text{c}} - T_{\text{ci}}) + d(t) \end{aligned} \quad (4.20)$$

$$d(t) = \frac{(-\Delta H)}{\rho C_{\text{p}}} k_0 \left\{ \exp\left(-\frac{E}{RT}\right) C_{\text{A}} - \exp\left(-\frac{E}{RT_{\text{i}}}\right) C_{\text{Ai}} \right\} \quad (4.21)$$

式 (4.21) は未測定外乱項として取り扱うことができ、式 (4.18) は未測定外乱項の一部となる。式 (4.20) は 1 次遅れ系の伝達関数として取り扱うことができ、ARX モデル予測制御の適用が可能となる。CSTR の 1 次遅れ系伝達関数モデルパラメータである

プロセスゲイン  $K_p$  及び時定数  $\tau$  は式 (4.22) で表される。

$$K_p = \frac{UA}{\rho C_p q + UA}, \tau = \frac{V \rho C_p}{\rho C_p q + UA} \quad (4.22)$$

#### 4.6.3 非線形モデル予測制御

本論文が提案する ARX モデル予測制御と比較するために、上述のプロセスについて非線形モデル予測制御の適用も試みた [10]。SISO システムの非線形モデル予測制御を含むモデル予測制御は以下の式 (4.23) , (4.24) , (4.25) , (4.26) で定式化され、式 (4.24) を制約条件下で準ニュートン法、共役降下法、逐次 2 次計画法の最適化手法を用いて数値計算によって、プロセスへの制御入力を決定すればよい。

非線形モデル

$$\frac{dy(t)}{dt} = f(y(t), u(t), t) \quad (4.23)$$

制御演算用評価関数

$$\min_{u(t)} J = \int_0^\infty (y_r(t) - y(t))^2 dt \quad (4.24)$$

参照軌道

$$y_r(t) = y(0) + (SP - y(0)) \left(1 - e^{-\frac{t}{\tau}}\right) \quad (4.25)$$

制約条件

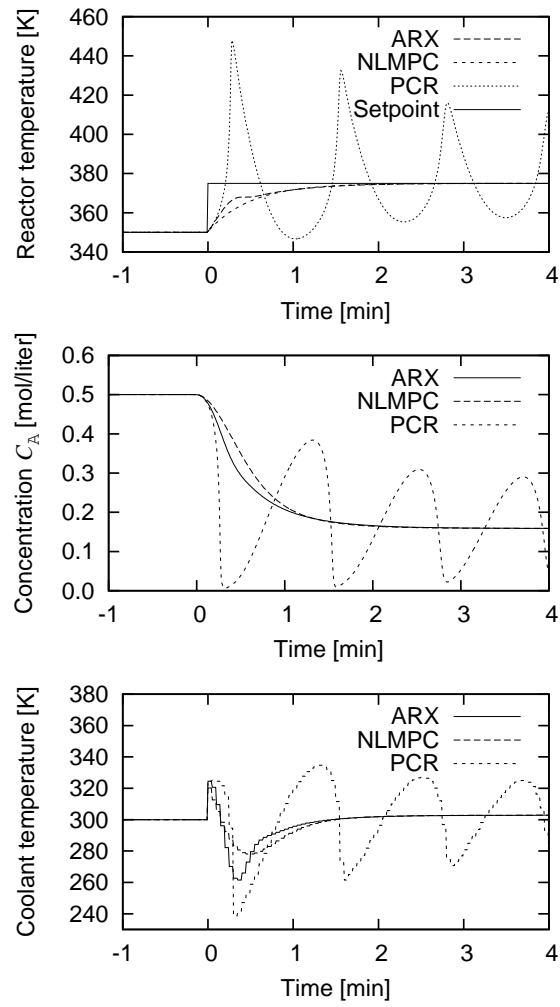
$$u_{\min} < u(t) < u_{\max}, y_{\min} < y(t) < y_{\max} \quad (4.26)$$

実際の最適化演算は時間を無限大とすることができないので、打ち切る時間が必要である。これはサンプリング時間の整数倍  $H$  で定義され、Control horizon と呼ばれるものである。

#### 4.6.4 制御シミュレーション結果

シミュレーションに使用した 1 次遅れ系伝達関数モデルパラメータ及びチューニングパラメータを、それぞれ、表 4.12 と表 4.13 にまとめた。非線形モデル予測制御に関して Control horizon は  $H=20$  とした。ARX モデル予測制御の制御性は、前述の非線形モデル予測制御と PCR を適用した場合の制御結果と比較を行った。

図 4.20 に非線形モデル予測制御 (NL MPC) , PCR , ARX モデル予測制御 (ARX) を CSTR の温度制御に適用した場合のシミュレーション結果を示す。0 分に +25K の設定値変更を行い、この際に ARX モデル予測制御に対して発熱反応による未測定外乱が加わるシミュレーションになっている。



✎ 4.20: Comparison of control performance for nonlinear process CSTR

表 4.12: Model parameters of CSTR 1st order transfer function

1st Order parameter	Value	Unit
$K_p$	0.6766	K/K
$\tau$	19.405	s

表 4.13: Tuning parameters of ARX MPC

Tuning parameter	Value	Unit
$\Delta t$	3	s
$\tau_r$	30	s
$\alpha$	$e^{-\frac{\Delta t}{\tau}}=0.8568$	-

PCR では振動が収まらずに不安定になっている。非線形モデル予測制御は 1 次遅れの参照軌道に従って制御が行われていることがわかる。ARX モデル予測制御は最初の立ち上がりは 1 次遅れの参照軌道から少しずれて制御が行われているが、設定値に近づくに従って、非線形モデル予測制御と同じ軌道をとって制御が行われている。ARX モデル予測制御の制御性は、線形のモデルを使用しているにもかかわらず非線形モデル予測制御と同等の制御性能が得られている。これは、各制御時間で徐々に外乱補償を行っているので振動的になるのを抑制できたものと考ええる。

## 4.7 結言

本論文では、ARX モデルを用いた 1 段予測型のモデル予測制御 (ARX モデル予測制御) を提案した。この ARX モデル予測制御は、未測定外乱に対してロバスト性の高い制御性が得られることが確認でき、未測定外乱に対しても AR 項  $\alpha$  をチューニングパラメータとすることで制御性をチューニング可能であることが提案できた。プロセスを伝達関数で表現できれば ARX モデルへの変換が可能であることから、1 次遅れ系、2 次遅れ系 (Time constant, Damping ratio)、積分系の伝達関数を使用すればほぼすべてのプロセスを近似できるので ARX モデル予測制御の適用範囲はかなり広いと考えられる。さらに、非線形なプロセスへの適用を行った結果、十分実用的であることが確認できた。ほぼすべてのプロセスプラントで、プラントオペレーションを PLC(Programmable logic controller) や DCS(Distributed control system) で行っている。PLC や DCS は収束計算が得意ではないこと及び収束計算による計算負荷をあま

りかけられないことから、収束計算の必要のない1 段予測予測制御である ARX モデル予測制御は PLC または DCS へ組み込み、プロセス制御を実行するのに適していると考えられる。

## 使用記号

$A$	=heat transfer area	[m <sup>2</sup> ]
$b_i$	=moving average coefficient	[-]
$C_A$	=concentration of reactant A	[mol/ℓ]
$C_p$	=specific heat of reactor solution	[J/(g·K)]
$D$	=dead time	[s]
$E$	=activation energy	[J/mol]
$e$	=measurement noise	[-]
$e_m$	=error of process output and model output	[-]
$G(s)$	=transfer function	[-]
$H$	=Control Horizon	[-]
$J$	=quadratic performance index	[-]
$k$	=discrete time by sampling time	[-]
$K_p$	=process gain	[-]
$k_0$	=frequency factor	[1/min]
$L$	=integer part of $D/\Delta t$	[-]
$m$	=finite number of auto reguression coefficient	[-]
$n$	=finite number of moving average coefficient	[-]
$q$	=inlet flow rate of reactant A	[ℓ/min]
$R$	=gas constant	[J/(mol·K)]
$RR$	=return ratio for unmeasured disturbance	[-]
$SP$	=setpoint of process output	[-]
$T$	=reactor temperature	[K]
$U$	=overall heat transfer coefficient	[J/(min·K·m <sup>2</sup> )]
$u$	=process input	[-]
$u_{\max}$	=maximum constraint of process input	[-]
$u_{\min}$	=minimum constraint of process input	[-]
$u_0$	=initial value of process input	[-]
$V$	=volume of the reactor	[ℓ]
$y$	=process output	[-]
$y_m$	=predicted process output	[-]
$y_{\max}$	=maximum constraint of process output	[-]
$y_{\min}$	=minimum constraint of process output	[-]

$y_r$	=reference trajectory of process output	[-]
$y_{ss}$	=steady state gain	[-]
$y_0$	=initial value of process output	[-]
$\alpha$	=auto regression coefficient for 1st order ARX model	[-]
$\alpha_i$	=auto regression coefficient for ARX model	[-]
$\tau$	=process time constant	[s]
$\tau_r$	=time constant of reference trajectory	[-]
$\tau_0$	=lead time	[s]
$\tau_1$	=time constant of 2nd order transfer function	[s]
$\tau_2$	=time constant of 2nd order transfer function	[s]
$\Delta t$	=sampling time	[s]
$(-\Delta H)$	=heat of reaction	[J/mol]
$\zeta$	=damping ratio	[-]
$\rho$	=density of reactor solution	[g/ $\ell$ ]
$\omega$	=fixed angular frequency	[rad/s]
$\langle \text{Subscript} \rangle$		
c	=coolant	
f	=inlet feed	
i	=initial value	

## 第5章 結論

化学プロセスプラントにおいて、機器と製品仕様の制約を考えた上でプラントを運転することはスループット、製品品質、製品収率、エネルギー消費量、安全性、安定性、操作性、生産性を向上させるためにとても重要なことである。その役目を担っているのはプロセス制御である。そして、このプロセス制御技術を支えているのは、以下のことである。

- プロセスモデリング
- コントローラのアルゴリズム
- コントローラのチューニング

これらのそれぞれに対して研究を行い、線形のプロセス制御をベースに新たな制御性能向上のための技術提案を行ってきた。研究の姿勢として、アルゴリズムは近似をあまり使わず容易に理解でき、プロセスプラントに従事するオペレータからエンジニアまでが容易に使用でき、メンテナンスできる技術開発である。プロセスプラントの制御構造としては、一番下層には DCS がありその中で構築される制御ループは PID コントローラをベースとするループが 90%以上を占めており、まず、最下層での制御性能を確保しなければ、その上位にどんなプロセス制御を構築したとしても、それらの性能を十分に発揮することはできない。そのため、PID コントローラの比例・積分・微分という 3 つのチューニングパラメータを適切なバランスで選んでチューニングする必要がある、そのバランスが崩れると、PID コントローラの性能を損なう可能性がある。そこで、コントローラのチューニング手法として、第 2 章で PID コントローラの PID チューニングパラメータデザイン手法を提案して、まず、最下層の制御性能向上の実現できることを確認した。多くの場合、DCS の上位にモデル予測制御を構築することになるが、このモデル予測制御はランプ上の反応熱外乱や外気温の急激な変化のような未測定外乱に対して制御性能が確保できない問題がある。そこで、制御アルゴリズムとして、第 4 章で未測定外乱に対して速い補償動作を行うことができる新たなモデル予測制御アルゴリズムを提案し、その補償動作スピードに対しても調整を行えることを確認し、そのアルゴリズムの有用性を証明した。それから、PID コントローラのチューニングやモデル予測制御適用においてだけではなく、プロセス制御を構築する上で、最も重要となるのは、プロセスモデリング技術であり、プロセスダイナミクスに対するモデル精度が維持・改善できれば、制御性能を維持し改善することが可



能である。そして、このプロセスモデル精度の維持・改善は一番時間と労力を要する作業である。そこで、この改善・維持のためのプロセスモデリング技術として、線形のプロセスモデルを容易に同定する新たな手法について、第3章で提案した。実プロセスに対して、モデル予測制御をオンラインにしながら設定値変更を行いモデル同定のためのデータ収集を行う閉ループステップ応答テストを行い、本提案同定手法により、モデル予測制御のモデルメンテナンスを行い、モデル精度向上およびモデル予測制御の性能改善が実現可能であることを確認した。本論文にて提案された手法を組み合わせることで、プロセスプラントの制御性能の向上を実現できることを提案できた。

## 付 録 A    ARX モデルの AR 項から MA 項 を求める方法

この付録では、以下の式で表現されるインクリメンタル形式の ARX モデルの AR 項  $\alpha$  を任意に与えることで、MA 項  $b_i$  を求める方法を示す。

$$\begin{aligned} y(k+1) &= y(k) + \alpha(y(k) - y(k-1)) \\ &\quad + b_1(u(k) - u(k-1)) + b_2(u(k-1) - u(k-2)) \\ &\quad + \cdots + b_n(u(k-n+1) - u(k-n)) \end{aligned} \quad (\text{A.1})$$

ここで、1 入力 1 出力のシステムに対する単位ステップ応答を以下のように表現することとする。

$$y(k) = f(k\Delta t) \quad (\text{A.2})$$

初期値は、 $u(0) = 0, y(0) = 0$  とする。次に時刻  $t = 0$  に単位ステップ入力が入ったとする。つまり、 $u(0) = 1$  となる。ARX モデルで 1 ステップ先の出力  $y(1)$  は以下の式になる。

$$y(1) = b_1 \quad (\text{A.3})$$

$y(1) = f(\Delta t)$  であることから  $b_1$  は以下の式になる。

$$b_1 = f(\Delta t) \quad (\text{A.4})$$

もう 1 ステップ進めると以下の式が得られる。

$$\begin{aligned} y(2) &= y(1) + \alpha y(1) + b_2 \\ &= f(\Delta t) + \alpha f(\Delta t) + b_2 \end{aligned} \quad (\text{A.5})$$

上式より  $b_2$  は以下の式になる。

$$b_2 = f(2\Delta t) - (1 + \alpha)f(\Delta t) \quad (\text{A.6})$$

さらに、もう 1 ステップ進めると以下の式になる。

$$\begin{aligned} y(3) &= y(2) + \alpha(y(2) - y(1)) + b_3 \\ &= f(2\Delta t) + \alpha(f(2\Delta t) - f(\Delta t)) + b_3 \end{aligned} \quad (\text{A.7})$$

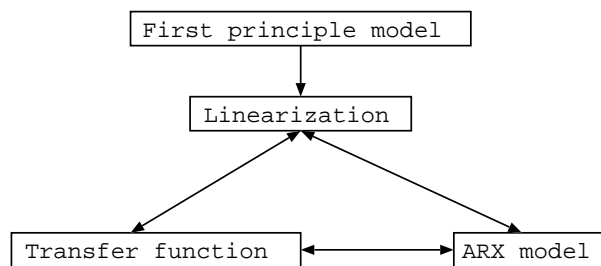


図 A.1: Scheme of ARX model transformation

上式より  $b_3$  は以下の式になる。

$$b_3 = f(3\Delta t) - (1 + \alpha)f(2\Delta t) + \alpha f(\Delta t) \quad (\text{A.8})$$

$b_3$  以降の MA 項モデルパラメータは以下のように一般化される。

$$\begin{aligned} b_i &= f(i\Delta t) - (1 + \alpha)f((i-1)\Delta t) \\ &+ \alpha f((i-2)\Delta t), i = 3, 4, \dots \end{aligned} \quad (\text{A.9})$$

上記より単位ステップ応答 ( $f(k\Delta t)$ ) が求まればすべての伝達関数モデルは ARX モデルとして表現できる。微分方程式で表現できる物理モデル (First Principle Model) をたてることが可能であれば、そのモデルを線形化することで ARX モデルに変換することが可能である。この変換スキームを図 A.1 に示し、以下に代表的な伝達関数に対応する ARX モデルパラメータをまとめた。

#### 1 次遅れ系

- 伝達関数

$$\frac{K_p}{\tau s + 1}$$

- 単位ステップ応答

$$f(k\Delta t) = K_p \left(1 - e^{-\frac{k\Delta t}{\tau}}\right)$$

- ARX モデルパラメータ

$$0 \leq \alpha < 1$$

$$b_1 = K_p (1 - a)$$

$$b_i = K_p (1 - a) (a - \alpha) a^{i-2}, i = 2, 3, 4, \dots$$

$$a = e^{-\frac{\Delta t}{\tau}}$$

#### 積分系

- 伝達関数

$$\frac{K_p(\tau_0 s + 1)}{(\tau s + 1)s}$$

- 単位ステップ応答

$$f(k\Delta t) = K_p \{(\tau_0 - \tau)(1 - e^{-\frac{k\Delta t}{\tau}}) + k\Delta t\}$$

- ARX モデルパラメータ

$$\alpha = 1$$

$$b_1 = K_p \{ (\tau_0 - \tau) (1 - a) + \Delta t \}$$

$$b_i = K_p (\tau_0 - \tau) (1 - a)^2 a^{i-2}, i = 2, 3, 4, \dots$$

$$a = e^{-\frac{\Delta t}{\tau}}$$

## 2 次遅れ系 (Time constant)

- 伝達関数

$$\frac{K_p(\tau_0 s + 1)}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

- 単位ステップ応答

$$f(k\Delta t) = K_p \left( A_1 e^{-\frac{k\Delta t}{\tau_1}} + A_2 e^{-\frac{k\Delta t}{\tau_2}} + 1 \right)$$

$$A_1 = \frac{\tau_0 - \tau_1}{\tau_1 - \tau_2}, A_2 = \frac{\tau_0 - \tau_2}{\tau_2 - \tau_1}$$

- ARX モデルパラメータ

$$0 \leq \alpha < 1$$

$$b_1: \text{式 (A.4)}$$

$$b_2: \text{式 (A.6)}$$

$$b_i: \text{式 (A.9)}, i = 3, 4, \dots$$

## 2 次遅れ系 (Damping ratio)

- 伝達関数

$$\frac{K_p(\tau_0 s + 1)\omega^2}{s^2 + 2\zeta\omega s + \omega^2}$$

- 単位ステップ応答

$$f(k\Delta t) = K_p \{ -e^{-\zeta\omega k\Delta t} \cos(Ek\Delta t) + Ae^{-\zeta\omega k\Delta t} \sin(Ek\Delta t) + 1 \} \text{ at } \zeta < 1$$

$$f(k\Delta t) = K_p \{ -e^{-\omega k\Delta t} + (\tau\omega - 1)\omega k\Delta t e^{-\omega k\Delta t} + 1 \} \text{ at } \zeta = 1$$

$$f(k\Delta t) = K_p \{ B_1 e^{\beta_1 k\Delta t} + B_2 e^{\beta_2 k\Delta t} + 1 \} \text{ at } \zeta > 1$$

$$A = \frac{\tau_0\omega - \zeta}{\sqrt{1 - \zeta^2}}, E = \omega\sqrt{1 - \zeta^2},$$

$$\beta_1 = -\zeta\omega + \omega\sqrt{\zeta^2 - 1}, \beta_2 = -\zeta\omega - \omega\sqrt{\zeta^2 - 1},$$

$$B_1 = \frac{\tau_0\omega - \zeta - \sqrt{\zeta^2 - 1}}{2\sqrt{\zeta^2 - 1}}, B_2 = -\frac{\tau_0\omega - \zeta + \sqrt{\zeta^2 - 1}}{2\sqrt{\zeta^2 - 1}}$$

- ARX モデルパラメータ

$$0 \leq \alpha < 1$$

$$b_1: \text{式 (A.4)}$$

$$b_2: \text{式 (A.6)}$$

$$b_i: \text{式 (A.9)}, i = 3, 4, \dots$$

## 参考文献

- [1] R. Brown, Society and Economy in Modern Britain 1700-1850, London:Routledge p.60 UK (1991)
- [2] J.C. Maxwell, On Governors, the Proceedings of the Royal Society **16** (1868) 220–283
- [3] N. Minorsky, Directional stability of automatically steered bodies, Journal of the American Society of Naval Engineering **34** (1922) 284–309
- [4] K.H. Ang, G. Chong and Y. Li, PID Control System Analysis, Design and Technology, IEEE trans. Control Systems Technology **13** (2005) 559–576
- [5] C.E. Garcia and M. Morari; Internal Model Control - A unified review and some new results -, Ind. Eng. Chem. Process Des. Dev. **21** (1982) 308–323
- [6] J. Richalet, A. Rault, J.L. Testud and J. Papon, Model Predictive Heuristic Control : Applications to Industrial Processes, Automatica **14** (1978) 413–428
- [7] C.R. Cutler and B.L. Ramaker, Dynamic Matrix Control : A Computer Control Algorithm, Proc. Joint Automatic Control Conference WP5-B San Francisco USA (1980)
- [8] ジャック・リシャレ, 江口 元, モデル予測制御 - PFC(Predictive Functional Control) の原理と応用 -, 日本工業出版 (2007)
- [9] Y. Noguchi and M. Kobari, Model Predictive Control with Single Heat Transfer Fluid for Batch Reactor Temperature Control, SICE Annual Conference 2005 MA2-17-5 Okayama Japan
- [10] M.A. Henson and D.E. Seborg, Nonlinear Process Control, Prentice Hall PTR, (1997) 5–8
- [11] K.L. Chien, J.A. Hrones and J.B. Reswick, On the Automatic Control of Generalized Passive Systems, Transaction of the American Society of Mechanical Engineering **74** (1952) 175–185.

- [12] Y. Lee, S. Park, M. Lee and C. Brosilow, PID Controller Tuning for Desired Closed-Loop Responses for SI/SO Systems. *AIChE Journal* **29** (1998) 106–115.
- [13] S.W. Sung and J.H. Lee, Pseudo-random binary sequence design for finite impulse response identification, *Control Engineering Practice* **11** (2003) 935–947.
- [14] OAZ. Sotomayor, D. Odloak and LFL. Moro, Closed-loop model re-identification of processes under MPC with zone control, *Control Engineering Practice* **17** (2009) 551–563.
- [15] L. Ljung, *System Identification : Theory for the User*, Prentice Hall, Englewood Cliffs: NJ, 1987.
- [16] Y. Hirama, H. Hamane and F. Hiroki, Closed Loop Modelling Method for Non-linear System Using Laguerre Polynomials, *ICCAS 2010: Gyeonggi-do, Korea*, 2010.
- [17] H. Ase and T. Katayama, A Realization-based Identification Method of Closed-loop Systems using Measurable Disturbance. *SICE J. Control, Measurement & System Integration* **2** (1) (2009) 10–19.
- [18] S.J. Qin and T.A. Badgwell, A Survey of Industrial Model Predictive Control Technology, *Control Engineering Practice* **11** (2003) 733–764.
- [19] Y. Shigi, M. Kano, and S. Hasebe, (2009). Detection of Significant Model-Plant Mismatch Using Stepwise Method for Model-Based Control, *AIChE Annual Meeting (251c)*, (2009) Nashville, USA.
- [20] Honeywell’s Experience, *CPC VI Meeting* 2001
- [21] Texas A&M Instrumentation Symposium for the Process Industries, January, 1996
- [22] K. Wu, C. Yu and Y. Cheung, A two degree of freedom level control, *Journal of Process Control*, 11, 311–319 (2001)

## 謝辞

本研究を行うに当たり、絶えず熱心なご指導を賜りました山下 善之教授に深く感謝いたします。東北大学時代からのお付き合いとなり、その当時、私は松本研究室で学士・修士の学生として、山下教授はお隣の研究室・鈴木研究室で助手としてお互い研究に従事していました。私が就職したあとも、学会で時々お会いして、私の博士号取得に対して前向きに考えいただき、暖かく研究室に迎えていただき、研究活動を見守っていただきありがとうございました。

それから、山下研究室でお世話になった北島 禎二助教、米田 健一助教、峰島 知芳助教に感謝いたします。あと、研究室で一緒に研究に勤しんだ蛸原 雄太君、タン キーシェン君、田中 宏幸君、佐々木 愛桂さん、橋本 寛喜君、笹川 薫君、山田 直樹君、高見 俊介君、菰田 充華さん、田中 麻穂さんに感謝いたします。

日揮株式会社の小崎 恭寿男さん、細谷 敬三さん、野口 芳和さんには、日揮株式会社の業務に従事しながら社会人ドクターとして博士号取得をとるためのサポートをしていただき感謝しています。

JX 日鉱日石エネルギー株式会社の田中 太郎様、田代 敦様、大宮司 理晴様には、第3章の新たな同定手法を実プロセスに適用する際にご協力いただき深く感謝いたします。

財団法人 国際石油交流センター (JCCP) の佐々木 照彦様にも、第3章の新たな同定手法の閉ループモデル同定において JCCP 殿の装置を使用させていただいたことに感謝いたします。

最後に、社会人ドクターとして博士論文を書くために関わっていただいたすべての皆様に感謝したいと思います。

2012 年 9 月

## 投稿論文

- 昆潤一郎、山下善之「ARX モデルを使用したモデル予測制御」、化学工学会化学工学論文集、第 36 巻、第 4 号、pp. 394-404、2010
- Junichiro Kon, Yoshiyuki Yamashita “ Optimal Tuning Parameters of a PID controller based on Reference Trajectory ”, Journal of Chemical Engineering of Japan, accepted, on-line available
- Junichiro Kon, Yoshiyuki Yamashita, Taro Tanaka, Atsushi Tashiro, Masaharu Daiguji “ Practical Application of Model Identification based on ARX Models with Transfer Functions ”, Control Engineering Practice, accepted



## 口頭発表

### 【国際学会における発表】

- Junichiro Kon, Yamashita Yoshiyuki 「Optimal Tuning Parameters of PID Control based on Reference Trajectory」, 13th Asia Pacific Confederation of Chemical Engineering Congress (APCChE 2010)、台湾、2010年10月、Paper No. 10832
- Junichiro Kon, Yamashita Yoshiyuki 「Model Predictive Control based on ARX Models」, International Conference on Control, Automation and Systems 2010 (ICCAS 2010)、韓国、2010年10月、Paper No. 100445
- Junichiro Kon, Yamashita Yoshiyuki 「Model Identification based on ARX Models with Transfer Functions」, 14th Asia Pacific Confederation of Chemical Engineering Congress (APCChE 2012)、シンガポール、2012年2月、Paper No. 32
- Junichiro Kon, Yamashita Yoshiyuki, Taro Tanaka, Atsushi Tashiro, Masaharu Daiguji 「Practical Application of Model Identification based on ARX Models with Transfer Functions」, 6th International Conference on Process Systems Engineering (PSE ASIA 2013)、マレーシア、2013年6月、Submitted

### 【国内学会・シンポジウム等における発表】

- 昆潤一郎、山下善之 「ARX モデルを使用したモデル予測制御」、『SICE』、第10回制御部門大会、熊本、2010年3月
- 昆潤一郎、山下善之 「参照軌道にもとづくPIDコントローラの最適チューニング」、『化学工学会』、第42回秋季大会、京都、2010年9月
- 昆潤一郎、山下善之 「伝達関数を応用したARXモデルによるモデル同定」、『化学工学会』、第76年会、東京、2011年3月
- 昆潤一郎、山下善之、田中太郎、田代敦、大宮司理晴 「伝達関数を用いたARXモデルによるモデル同定法の実プロセスへの適用」、第54回自動制御連合講演会、愛知、2011年11月