

係数図法  
制御系設計の理論と応用

真鍋舜治

August 6, 2009

# Chapter 1

## 序論

本章では係数図法の概要を示す。基本的な考え方では、本書の執筆に当たった著者の基本的な考え方を示す。簡単な設計問題では、簡単な位置制御系の設計問題を解くことにより、係数図法の概略を示している。システムの表現では係数図法で用いる多項式表現と伝達関数表現・状態空間表現の関係を説明している。設計の概要では、係数図法による設計の具体的なステップを示す。制御構造では、色々な形の制御器も、係数図法の制御器で等価的に表されることを示す。歴史的背景では、係数図法に到達した略50年の過程を振り返る。最後にまとめでは、本章のまとめを行い、また係数図法の特徴を6個にまとめている。

### 1.1 基本的な考え方

最初に著者の基本的な考え方を、制御、フィードバック制御、代数的設計法の項目に分けて述べる。制御は妥協の産物であるということが、本書全体を通じる基本的な考え方であって、この考え方の背景をまず説明する。次に広い意味での制御の構造を示し、フィードバック制御は、制御の小さな部分しか占めないことを示す。最後に代数的設計法について簡単に説明し、古典制御・現代制御と比較する。そして代数的設計法の一つである係数図法が、古典制御・現代制御に内在する種々の問題点を解決していることを示す。

制御は妥協の産物であるということが、本書の基本的な立場である。制御の漢字は「制」と「御」であるが、制の偏は木が繁茂する様子の意味し、つくりは刀を意味していて、両方で余分の枝をそぎ落とすことを意味する。「御」は「馭」に通じ、馬を巧みに扱うことを意味する。制御という言葉は「相手を押さえて自分の思うように支配する」ことを意味するが、漢字の原意は少し異なっているように思える。木の枝をそぎ落とすことはできるが、枝を伸ばすことはできない。馬を走らせるとき、時速100 km/hで走らせる

ことも、時速 1 km/h で走らせることもできない。馬には自分の好きな速度があって、それからかけ離れた速度で走らせることはできないから、馬に乗る人は自分の走りたい速度と馬が走りたい速度の妥協点を見出さざるを得ないのである。

「制御」という言葉は略 2000 年前、前漢の時代に、司馬遷が表した史記の中にでている。秦王朝の歴史を記述した始皇本紀の中で、秦の二世皇帝は、国内の平和を保つため税を軽減を求めた大臣の進言に対して、次のように答えている。

「凡そ天下をたもつに貴しと為す所は、我意をほしいままにし、欲を極め、上は明法を重んじ、下は敢えて非違を為さずして、海内を制御することを得るに在り。」

ここで二世皇帝は「制御」という言葉を、臣民を自分の思いどおりに制御するという意味で用いている。このような態度は、良い政府として通例考えられているものとは、全く異なっており、この王朝が三世皇帝で終わったことも、無理からぬことと思われる。

良い制御は、制御する側の望みと、制御される側の望みとの妥協の産物である。あるいは「すべきこと」と「できること」の間の妥協ともいえる。現在、制御系の設計/シンセシス理論の基礎となっている考え方には色々あるが、それらはこの妥協の巾のどこかの点にある。「古典制御」は「できること」の側に寄っており、現代制御は「すべきこと」の側に寄っているように見える。

現代制御では、最適化に重きを置き過ぎているように思われる。妥協が重要な世界では、最適化を求めることは決して良い考え方とは言えない。第 1 には、最適化のため何らかの評価基準が必要になるが、最適化を求める努力の結果、設計結果は与えられた評価基準に対しては最適であるが、他の評価基準に対しては大変劣るものになることが挙げられる。常識的には、最適化の努力を、最終的な最適点に到達する以前に止めると、他の評価基準に対しても、かなり良い値になることが知られている。すなわちセカンドベストは他の評価基準でもセカンドベストになるのが通例である。第 2 に、最適点は必ずしもピークではなく、山脈や高原になっていることが多いことが挙げられる。その結果最適点を数学的に求めることが難しく、最終的には設計者の常識に委ねられることになる。これらの理由により、本書では最適化は出来るだけ避け、妥協をより重視している。

制御系設計では、このように妥協が大切であるから、妥協を容易にするような工夫が、設計過程の中に組み込まれていなければならない。問題が絵画的に表現されている場合、人間は妥協点を見つけるのに、優れた能力を発揮する。絵画的表現こそ妥協のための鍵と考えられ、この点について、十分の配慮が必要である。「数式よりグラフ」が本書の一つの特徴である。

フィードバック制御は、制御の小さな部分しか占めない。本書は動的システムのフィードバック制御の設計を目的としている。とくに不時変系 (LTI, Linear Time Invariant) が主題である。フィードバック制御は非常に重要であるが、これは制御全体に対しては、小さな部分しか占めないもので、これを極端に重要視することは避けなければならない。制御系全体は Fig. 1.1 に示される。これは制御対象、フィードバック制御器、知能形制御器で構成されている。

制御対象への入力はアクチュエータで、制御対象からの出力はセンサである。制御対象は動的なもので、微分方程式で表されている。ここに示す制御対象はエネルギー機器で、その特性はエネルギーによって制限されている。フィードバック制御器は情報機器であって、センサ出力は情報としてフィードバック制御器に入力され、その出力は情報の形でアクチュエータの入力に伝えられる。情報は一般に電子的な形態になっている。フィードバック制御器には、「コマンド生成」「制御アルゴリズム」「センサ情報処理」の三つの機能がある。

フィードバック制御器は、知能形制御器に接続されている。これも情報機器であって、大きな構造化された記憶でできていて、その中に多数の処理手続きと、大きな過去の経験が蓄積されている。知能形制御器の主要な目的は、特定な時刻にどのような制御をすべきかについて指令を与えることである。このような指令は、コマンド生成と制御アルゴリズムの修正という二つのもので構成されている。知能形制御器への入力は、制御対象の情報で、これらはアクチュエータとセンサの情報から得られる。また全体の制御の基本的なコマンドになる外部からの情報も与えられている。一般には多数のフィードバック制御器が、一つの知能形制御器に制御される形をとっている。知能形制御器は階層上では、フィードバック制御器の上位にある。

フィードバック制御器と知能形制御器をまとめて、広義の制御器となっている。一般にこれらは、一つの制御用計算機の中にある。制御用計算機が導入される以前の古い時代では、フィードバック制御器はアナログ形で、知能形制御器の機能は、リレーシーケンス回路が受け持っていた。注意すべきことは、大部分の制御機能は知能形制御に用いられていて、5~10%位しかフィードバック制御に用いられていないことである。すなわちフィードバック制御は、広義の制御の一部でしかない。したがってフィードバック制御を極端に精緻にすることは、無意味と考えられる。

フィードバック制御器は、階層的に知能形制御器の下位にあるので、制御対象の状態に適応することができ、その非線形性に対処することができる。したがって不時変系に対処するように設計されたフィードバック制御器は、できるだけ簡単なものがよい。これを複雑にすると、知能形制御器の適応機能や非線形機能を加えることが難しくなる。

昔は制御器も制御対象もエネルギー機器であった。ワットの遠心力ガバナーとそれに制御される蒸気機関に見られるとおりである。最近では制御対象が情報機器となる場合が多く、そのときは、制御器・制御対象ともに情報

機器となるので、エネルギー制約問題は現れてこない。本書は制御器が情報機器で、制御対象がエネルギー機器という力学的なシステムを扱っている。この場合では一般にアクチュエータが、最も厳しいパワーと応答の制限を受けていて、この制限が閉ループの特性に対して、最も大きな影響を与えている。このようなアクチュエータは高価であって、ディメンション  $p$  として表されているアクチュエータの数は、ディメンション  $m$  で表されているセンサの数より遥かに小さい。この場合には、システムは網形ではなく、ループ形となり、アクチュエータが多数のループの、共通の節となる形になる。

このような制御系全体は、人間の体の運動に対比できる。人間の体が制御対象で、手・足がアクチュエータである。センサは、眼・耳・皮膚・筋力の感覚・釣り合い感覚など色々ある。フィードバック制御器は、脳幹と小脳と考えられ、知能形制御器は大腦である。人間の体はエネルギー機器であり、その特性はエネルギー制限によって強く制約されている。アクチュエータの数はセンサの数より圧倒的に少ない。過去の教育と経験により蓄積された多量の記憶は、知能形制御器として働き、融通性が高くて複雑な制御が実現されている。

代数的設計法は、Fig. 1.2 に示すように、古典制御と現代制御の中間に位置している、第3の制御理論である。これはシステム表現に多項式が用いられていることより、多項式設計法とも呼ばれている。古典制御では、周波数応答法と根軌跡法が使用されている。いずれの方法でも、システム表現には伝達関数が用いられている。制御器パラメータと閉ループ特性の関係は、図式表現を用いて表されている。周波数応答法ではボーデ/ナイキスト線図が、根軌跡法では根軌跡図が用いられている。設計にあたっては、パラメータは未知のままに制御器の基本的構造を仮定し、その後閉ループ特性が設計仕様を満たすようにパラメータを調整する。このような設計のすすめ方を「外向き設計法」と呼ぶ (Chen, 1987)。

現代制御としては、極配置法、最適制御 (LQR, LQG)、H無限大制御が現在行われている。これらの方法はすべて、システム表現を状態空間法で行っている。制御器パラメータと閉ループ特性との関係は方程式で表されている。極配置法では、閉ループ特性を示す極位置は、Ackermann の定理 (Franklin, 1994) のような代数的マトリックスの関係で、制御器パラメータと関係付けられている。最適制御とH無限大制御では、閉ループ特性はRiccati方程式を通して、制御器パラメータに関係付けられている。設計にあたっては、設計仕様を満足する閉ループシステム全体が最初に選ばれ、制御器はそれから色々な方程式の解として求められる。このような設計法は「内向き設計法」 (Chen, 1987) と呼ばれる。

古典制御や現代制御と異なり、代数的設計法あるいは多項式設計法は、まだ発展途上にあるので、明確に定義されていない。広い意味ではラウス安定条件を用いる設計法も含まれている。ここではシステム表現に多項式が用いられ、制御器パラメータは、ラウス表を通して関係付けられている。この設

計法は明らかに外向き設計法である。狭い意味では、代数的設計法は、直接的設計法 (Franklin, 1994, p. 564) と呼ばれるものである。システム表現は多項式である。閉ループ特性は特性多項式の形で表現されている。特性多項式は、Diophantine 方程式を通して制御器パラメータと関係付けられている。通例の設計法では、特性多項式が最初に選定され、制御器パラメータは Diophantine 方程式の解として求められる。これは明らかに内向き設計法である。この方法は多項式を多項式マトリックスに置き換えることにより、MIMO (多入力多出力) 系にも容易に拡張できる。この方法については、多数の研究者が熱心に研究しているが (Kucera, 1979)(Kailath, 1980, p.306)(Chen, 1987)(Franklin, 1994, p.564)、特性多項式をどのように選ぶかについては、研究は殆どされていない。特性多項式は、極配置によって定めるのが通例である。

伝達関数表現は分かり易いけれども、極・零キャンセルを含む場合には不正確になる。一方状態空間表現は正確であるが分かりにくい。多項式表現は伝達関数表現のように分かり易く、状態空間表現と同じように正確である。このことが代数的設計法が、古典制御や現代制御より優れている点の一つである。他の利点として、Diophantine 方程式は線形であるので、二乗形である Riccati 方程式より解を求め易いことがある。

係数図法 (CDM, Coefficient Diagram Method) は、直接設計法と比べると、制御器パラメータと特性多項式が、Diophantine 方程式と係数図 (Coefficient Diagram) とで関係付けられていることが異なっている。この関係は、数式と同時に、図式で分かり易く表現されているので、制御器と特性多項式を同時に設計する「同時設計法」が可能になる。外向き設計法での問題点は、最初に仮定した制御器の構造によって、到達できる閉ループ特性が制限されてしまうことである。もし最初の仮定が適切でなかったならば、目的とする閉ループ特性は得られない。また設計が完成したとしても、これより良い制御器が存在するのではないかという疑点が残る。内向き設計法では閉ループ特性は保障されているが、設計された制御器が、形が簡単でかつロバスト性が高いという、基本的な特質を備えているかどうか分からない。外向き設計法では、このような性質は制御器の構造を仮定する段階で埋め込まれているので、古典制御で設計された制御器は、ロバストであるのが通例である。ロバスト性の問題は、現代制御とそれに伴う内向き設計法の導入により表面化したものと考えられる。

係数図法の同時設計法により、与えられた閉ループ仕様を満足する、簡単でロバストな制御器を容易に設計することができる。係数図は図式表現であるので、ボーデ/ナイキスト線図を用いる古典制御と同じような直感的な設計が可能になる。制御系設計では、制御器の簡単さと閉ループ特性がトレードオフの関係になる。制御器は「できること」を示し、閉ループ特性は「すべきこと」を示している。これらの間の良い妥協点を見出すことが良い制御への鍵である。妥協ということが制御系設計で最も重要であり、係数図法はそれに対して、ある解答を与えるものである。

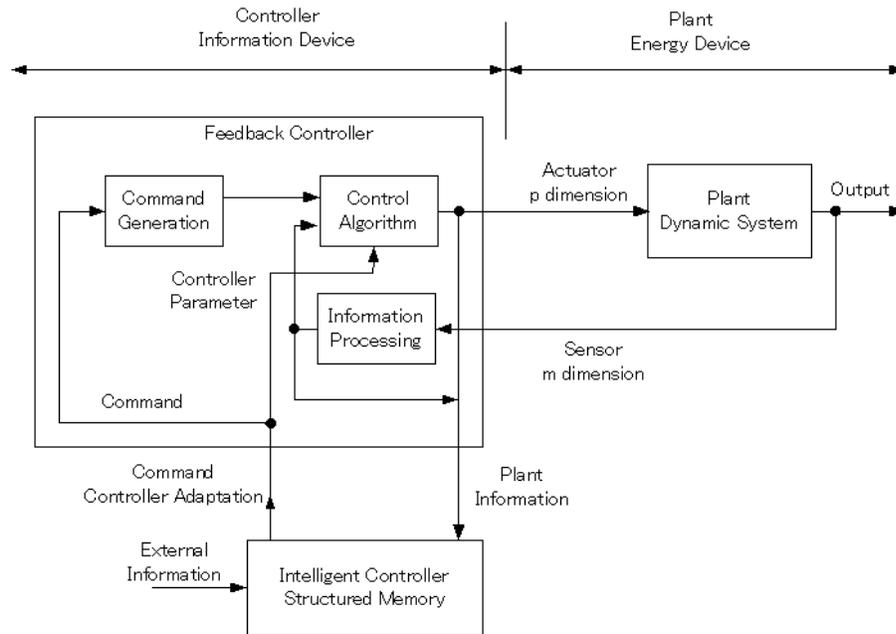


Figure 1.1: Control system

Classification	Design method	System representation	Controller/closed-loop relation	Approach
Classical control	Frequency response design	Transfer function	Bode/Nyquist diagram	Outward approach
	Root-locus design	Transfer function	Root-locus diagram	
Algebraic approach	Routh stability criterion	Polynomial	Routh table	Simultaneous design
	Coefficient diagram method	Polynomial	Diophantine equation Coefficient diagram	
	Direct method	Polynomial	Diophantine equation	
Modern control	Pole assignment	State space	Matrix relation (Ackermann's formula)	Inward approach
	Optimal control (LQR, LQG)	State space	Riccati equation	
	H-inf control			

Figure 1.2: Comparison of control theories

## 1.2 簡単な設計問題

本節では、簡単な位置制御系を設計することにより、係数図法の概略を示す。問題の設定、安定度指標・等価時定数の定義、設計の手続き、制御系の特性、係数成形法の5項目について、順次説明する。

まず問題の設定を行う。問題は Fig. 1.3 に示されるような一般的な位置制御系である。制御対象は電力増幅器とモータで構成されている。これは次のような微分方程式で表される。

$$(0.25s + 1)(s + 1)sy = u, \quad (1.1)$$

$$v = sy. \quad (1.2)$$

ここで、 $u$  は電力増幅器への入力、 $v$  は速度、 $y$  は位置である。記号  $s$  は微分オペレータ  $d/dt$  を表す。制御器は PD (Proportional and Differential) 制御器で、速度センサと位置のセンサが、必要なフィードバック信号を発生している。制御器は次の微分方程式で表される。

$$u = k_0(y_r - y) - k_1v. \quad (1.3)$$

ここで  $y_r$  は目標位置信号、 $k_0$  は位置ゲイン、 $k_1$  は速度ゲインである。これらの式から  $u$  を消去すると  $y_r$  と  $y$  の関係に関して、次の方程式が得られる。

$$(0.25s^3 + 1.25s^2 + s + k_1s + k_0)y = k_0y_r. \quad (1.4)$$

上式の左側  $y$  の前の項は特性多項式  $P(s)$  である。

$$P(s) = 0.25s^3 + 1.25s^2 + (1 + k_1)s + k_0. \quad (1.5)$$

問題は  $k_1$  と  $k_0$  を適切に選ぶことにより、安定性・応答性・ロバスト性のバランスのとれた、良い制御系を設計することである。

次に設計の準備として、安定度指標と等価時定数の定義を行う。特性多項式は一般に次の形で表される。

$$P(s) = a_n s + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 = \sum_{i=0}^n a_i s^i. \quad (1.6)$$

安定度指標  $\gamma_i$  と等価時定数  $\tau$  は次のように定義される。

$$\gamma_i = a_i^2 / (a_{i+1} a_{i-1}), \quad i = 1, \cdots, n-1. \quad (1.7)$$

$$\tau = a_1 / a_0. \quad (1.8)$$

安定度指標 (stability index) はシステムの安定度を示す重要な尺度である。これが重要なことは旧くから知られていて、様々の研究者によって、色々な名前が付けられてきた。Kessler (1960) はこれを「減衰度」(damping factor) と、Naslin (1968) は「特性比」(characteristic ratio) と、Brandenburg (1996) は「二重比」(double ratio) と呼んだ。「減衰度」はこの項の性質を正確に表現しているが、この言葉は現在別の意味で使われているので誤解が起こる。「特性比」はこの項の性質を適切に表していない。「二重比」はこれが  $(a_i/a_{i-1})/(a_{i+1}/a_i)$  のように、比の比で作られていることを示すが、その項の意味を示すものではない。このような理由から、「安定度指標」という新しい名前を与えることにした。安定度指標の標準値として、係数図法で推奨している値は次のとおりである。

$$\gamma_{n-1} = \cdots = \gamma_3 = \gamma_2 = 2, \quad \gamma_1 = 2.5. \quad (1.9)$$

このような推奨値が選ばれた理由は、係数図法の中心課題であり、後々の章で詳しく述べることにする。等価時定数 (equivalent time constant) は、応答速度の尺度である。安定度指標  $\gamma_i$  が標準値の場合、過渡応答は等価時定数  $\tau$  の 2.5 ~ 3 倍の間で収まる。係数図法では、安定度指標と等価時定数を設計の基礎としている。

つづいて設計の手続きを説明する。この問題は簡単なので、安定度指標  $\gamma_i$  を係数図法標準値 ( $\gamma_2 = 2, \gamma_1 = 2.5$ ) と選ぶことができる。安定度指標の定義をもとにして、次のような結果がえられる。

$$a_1 = 1 + k_1 = a_2^2/(a_3\gamma_2) = 1.25^2/(0.25 \times 2) = 3.125, \quad (1.10)$$

$$a_0 = k_0 = a_1^2/(a_2\gamma_1) = 3.125^2/(1.25 \times 2.5) = 3.125.$$

設計結果は次のようになる。

$$k_1 = a_1 - 1 = 2.125, \quad k_0 = 3.125, \quad \tau = 1. \quad (1.11)$$

設計結果は、オーバーシュートがない、応答は速い、ロバスト性が高いというバランスのとれた良い特性を示している。整定時間は  $\tau = 1$  に対応して 2.5 ~ 3 sec になっている。この設計では、簡単な算数だけで十分に高級な数学が必要ないことことに注目したい。

場合によっては、応答が遅くても良い場合がある。この時は、 $\gamma_2 > 2$ 、 $\gamma_1 = 2.5$  とし、 $k_1 = 2.125 \sim 0$  を自由パラメータとして設計を進める。

$$k_0 = a_0 = a_1^2/(a_2\gamma_1) = (1 + k_1)^2/(1.25 \times 2.5) = 0.32(1 + k_1)^2, \quad (1.12)$$

$$\gamma_2 = a_2^2/(a_3a_1) = 1.25^2/[0.25(1 + k_1)] = 6.25/(1 + k_1) > 2,$$

$$\tau = a_1/a_0 = 3.125/(1 + k_1).$$

ここで  $k_1$  が 0 となると応答は最も遅くなり、 $k_0 = 0.32$ 、 $\gamma_2 = 6.25$ 、 $\tau = 3.125$  となる。応答がさらに遅くてもよい場合は、設計は  $k_1 = 0$ 、 $\gamma_2 = 6.25$ 、 $\gamma_1 > 2.5$  の条件で、 $k_0 = 0.32 \sim 0$  を自由パラメータとして進められる。

$$\gamma_1 = a_1^2 / (a_2 a_0) = 1 / (1.25 k_0) > 2.5, \quad (1.13)$$

$$\tau = a_1 / a_0 = 1 / k_0 > 3.125.$$

このように、設計は簡単で融通性に富み、特別の要求にも容易に対処できる。係数図法がこのように有力なのは、安定度指標の標準値に、過去の設計経験が凝縮されている結果と考えられる。

次にこのようにして設計された制御系の特性について述べる。ここで Table 1.1 は、応答を遅くしてもよい場合で、 $\tau$  が設計値 1 から、2、3.125、5 と増加させている。このとき、応答波形を適切に保つため、安定度指標はシステムティックに増加させている。第 1 のステップは、 $\gamma_1 = 2.5$  と固定のまま、 $k_1$  が 0 になるまで、 $\gamma_2$  を増加させる。第 2 ステップは、 $\gamma_2 = 6.25$ 、 $k_1 = 0$  と固定したまま、 $k_0$  を減少させて、 $\gamma_1$  を大きくする。この係数図を Fig. 1.4 に示す。係数図では、特性多項式の係数  $a_i$  は対数目盛りで縦軸に、次数  $i$  は横軸に降幕の順に示されている。設計値 case#1 では、次のとおりである。

$$a_i = [a_3 \ a_2 \ a_1 \ a_0] = [0.25 \ 1.25 \ 3.125 \ 3.125]. \quad (1.14)$$

異なった  $\tau$  に対するステップ応答は、Fig. 1.5 に示されているが、すべて滑らかでオーバーシュートはない。

Table 1.2 は、 $k_0$  を 2 倍、半分にするにより、 $\gamma_1$  を半分、2 倍にした場合である。係数図を Fig. 1.6 に、ステップ応答を Fig. 1.7 に示す。安定度指標  $\gamma_1$  が小さなときは、安定性が劣化し、顕著なオーバーシュートが見られる。

これらより、応答速度は  $\tau$  に支配されており、ステップ応答の波形は、 $\gamma_1$  の強い影響を受けていることが分かる。この二つのパラメータが、ステップ応答を規定している。係数図上では、安定度指標  $\gamma_i$  が、特定の次数  $i$  での曲率になっていることは、その定義より明らかである。等価時定数は係数図の右端点での傾斜になっている。係数図とステップ応答とは、安定度指標と等価時定数を通して密接な関係をもっている。係数図が与えられれば、ステップ応答の概略を推定することができ、その逆も成り立つ。この事については、後の章で詳しく述べる。

上記の結果は、 $k_0$  と  $k_1$  を用いたパラメータ空間上で、Fig. 1.8 のように示される。ラウスの安定条件より、 $a_0 > 0$ 、 $a_2 a_1 > a_3 a_0$ 、となるので、安定領域は次のようになる。

$$0 < k_0 < 0.2 / (1 + k_1). \quad (1.15)$$

この安定領域は広すぎるので、パラメータの選択を適切に行うことができない。ここで  $\gamma_1 = a_1^2/(a_2a_0)$  を 2.5 と固定し、 $\gamma_2$  と  $\tau$  を変化させると、パラメータは次の方程式を満足する。

$$k_0 = 0.32(1 + k_1)^2. \quad (1.16)$$

この曲線の上では、どの点でも応答は滑らかであるが、この右側では  $\gamma_1$  が小となるのに対応して振動的になり、左側では  $\gamma_1$  が大になるのに対応して必要以上に緩慢になっている。設計パラメータは、この曲線上何処を選んでもよいが、点#1 では  $\tau$  が最小で応答が最も速いので、最適の選択と考えられる。

係数成形法は、係数図とステップ応答が密接な関係にあることを利用した図式設計法である。これにより、特性多項式と制御器を同時に設計することができる。この同時設計は係数図法の大きな特徴で、その意味については後に述べることにして、具体的な設計法を次に述べる。この例題では、特性多項式は次の二つの多項式の和になっている。

$$P(s) = P_0(s) + P_k(s), \quad (1.17)$$

$$P_0(s) = 0.25s^3 + 1.25s^2 + s,$$

$$P_k(s) = k_1s + k_0.$$

ここで  $P_0(s)$  は無制御の場合の特性多項式で、 $P_k(s)$  は制御器による特性多項式の変更部である。この係数図を Fig. 1.9 に示す。ここで  $P_0(s)$  は 1 点鎖線と印で、 $P_k(s)$  は破線と印で示す。係数図をこのように分解して描けば、制御器パラメータの変化で、特性多項式全体が、係数図上でどのように変化するか分かる。 $P(s)$  の係数図とステップ応答とは密接な関係があるので、ステップ応答が、どのように変化するかを容易に予測することができる。

設計に当たっては、設計者はまず  $P_0(s)$  の係数図を描く。また目標とするステップ応答より、目標となる  $P(s)$  の概略を係数図を描く。そしてこの間を埋めるように  $P_k(s)$  を定め、最終的な  $P(s)$  は、 $P_0(s)$  と  $P_k(s)$  の和として求める。このようにして、 $P(s)$  と  $P_k(s)$  の同時設計が可能になる。この方法は制御器の構造を定めるのに非常に有力であり、また目的に最も適した安定度指標と等価時定数を推定することも容易である。この方法は図式なので、設計者の直感を十分生かせることができ、設計を効率的に進めることができる。

Table 1.1:  $\tau$  variation

case	$k_1$	$k_0$	$\tau$	$\gamma_2$	$\gamma_1$
#1	2.125	3.125	1	2	2.5
#2	0.5625	0.78125	2	4	2.5
#3	0	0.32	3.125	6.25	2.5
#4	0	0.2	5	6.25	4

Table 1.2:  $\gamma_1$  variation

case	$k_1$	$k_0$	$\tau$	$\gamma_2$	$\gamma_1$
#1	2.125	3.125	1	2	2.5
#5	2.125	6.25	0.5	2	1.25
#6	2.125	1.5625	2	2	5

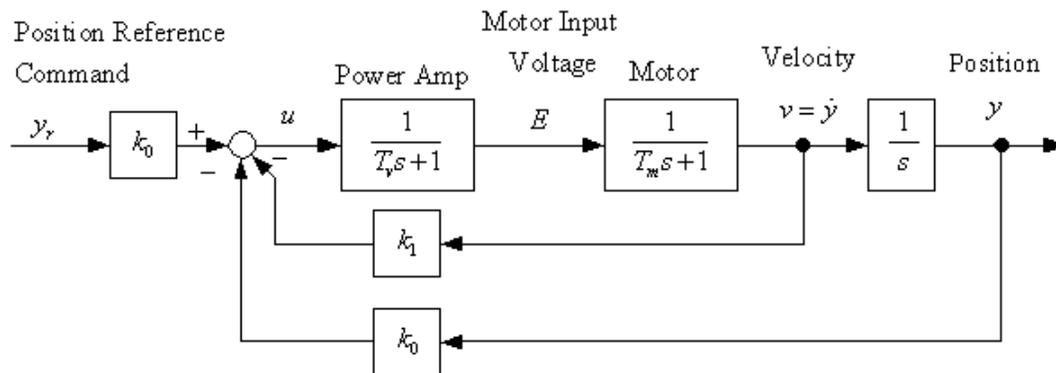
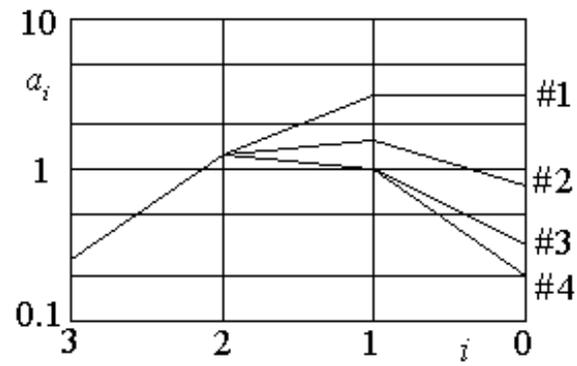
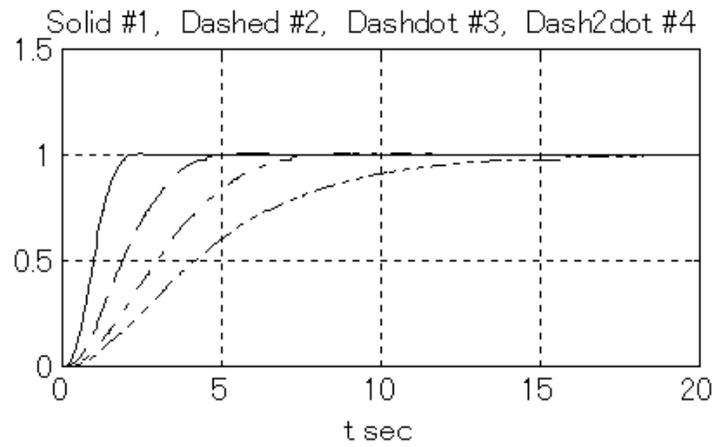
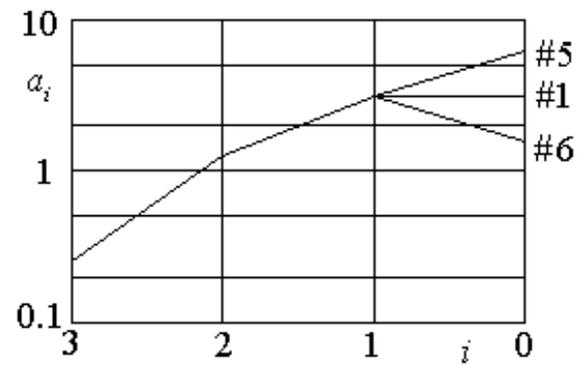


Figure 1.3: Position control system

Figure 1.4: Coefficient diagram  $\tau$  variationFigure 1.5: Step response,  $\tau$  variationFigure 1.6: Coefficient diagram  $\gamma_1$  variation

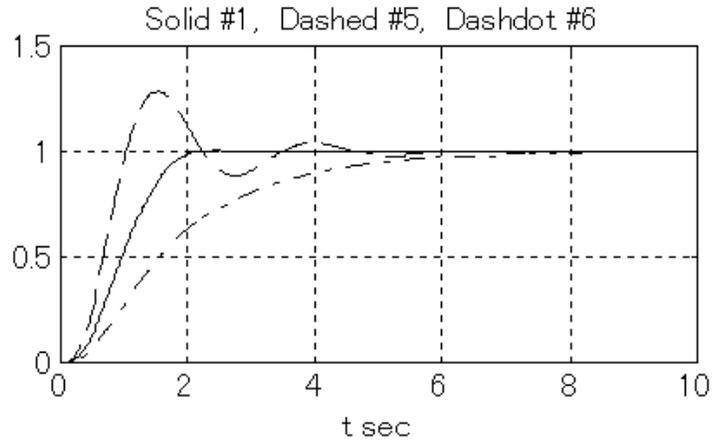


Figure 1.7: Step response,  $\gamma_1$  variation

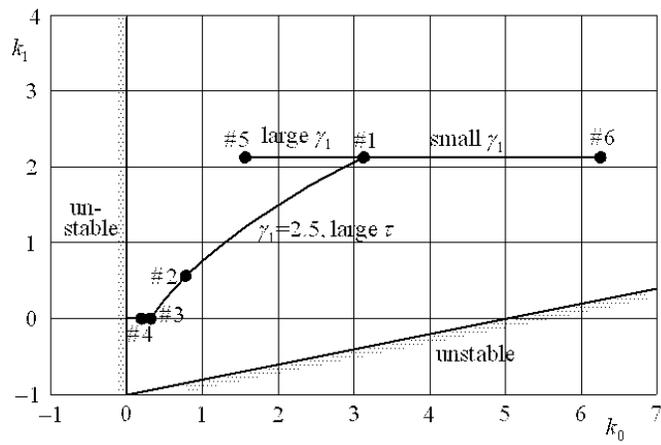


Figure 1.8: Parameter space

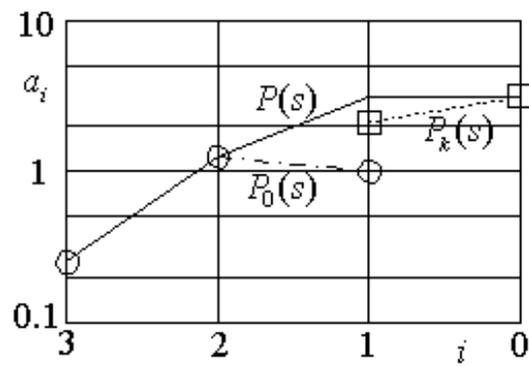


Figure 1.9: Coefficient diagram, coefficient shaping

### 1.3 システム表現

システムの表現には、古典制御では伝達関数表現を、現代制御では状態空間表現を用いるのが通例である。古典制御と現代制御では、解析・設計の手法が大きく異なるが、このような差異はシステム表現の差異から起こっているものと思われる (Franklin, 1994, p.471)。一つの国の文化がその国の言葉で特徴付けられることはよく知られているが、それと同じように対処できる問題の種類も、設計結果も、表現の差異に大きな影響を受けると考えられる。係数図法 (CDM) は、制御の非専門家が、自分の専門分野で実用的な制御器を設計するのに用いることを念頭においている。この理由で係数図法でのシステム表現には、多項式表現が用いている。多項式表現は動的システムの解析で通例使われている微分方程式表現と同じものだからである。

本節ではまず係数図法で用いる多項式表現について説明する。次にこれと伝達関数表現、さらに状態空間表現を比較する。そして多項式表現が、伝達関数表現の使い易さ、状態関数表現の正確さという両者の利点を兼ね備えていることを明らかにする。

係数図法で用いる多項式表現は、動的システムの微分方程式から直接導出される。システムは次のように表されている。

$$a_3 \frac{d^3 y}{dt^3} + a_2 \frac{d^2 y}{dt^2} + a_1 \frac{dy}{dt} + a_0 y = b_1 \frac{du}{dt} + b_0 u. \quad (1.18)$$

ここで  $u$  は入力、 $y$  は出力、 $t$  は時間である。ここで微分演算子  $d/dt$  を  $s$  で表すと方程式は次のようになる。

$$A(s)y = B(s)u, \quad (1.19)$$

$$A(s) = a_3 s^3 + a_2 s^2 + a_1 s + a_0, \quad (1.20)$$

$$B(s) = b_2 s^2 + b_1 s + b_0. \quad (1.21)$$

式 (1.19) で、 $y = B(s)x$  を満足するような変数  $x$  を新しく導入すると、別の式が得られる。

$$A(s)x = u, \quad y = B(s)x. \quad (1.22)$$

係数図法では、式 (1.22) 式 (1.19) とともに、標準形として用いられている。前者は右多項式分解 (Right Polynomial Fraction, RPF)、後者は左多項式分解 (Left Polynomial Fraction, LPF) と呼ばれている。本書ではラプラス変換は使用しないので、多項式  $A(s)$  及び  $B(s)$  は、必ず変数に前置される形で用いられる。また分母に用いられることはない。この点で、係数図法で用いている多項式表現は、一般に用いられている多項式表現とは異なっている。しかしブロック線図では、Fig. 1.10 や 1.11 のように、多項式が分母に置かれていること

がある。この時は、それらが式 (1.22) 及び式 (1.19) を表すものと解釈する。場合によっては、伝達関数  $G(s)$  が次のように表されている。

$$y = G(s)u = \frac{B(s)}{A(s)}u. \quad (1.23)$$

このとき式 (1.22) または式 (1.19) のいずれにでも解釈するので、伝達関数表現には曖昧さが残る。

多入力多出力 (Multi-Input-Multi-Output, MIMO) 系でも、1入力1出力 (Single-Input-Single-Output, SISO) 系と同じように扱うことができる。右多項式マトリックス分解 (Right Polynomial Matrix Fraction, RPF) では

$$A(s)x = u, \quad y = B(s)x, \quad (1.24)$$

となり、左多項式マトリックス分解 (Left Polynomial Matrix Fraction, LPMF) では次のようになる。

$$A^*(s)y = B^*(s)u. \quad (1.25)$$

ここで  $x$ 、 $y$ 、 $u$  は適切なディメンションのベクトルであり、 $A(s)$ 、 $B(s)$ 、 $A^*(s)$ 、 $B^*(s)$  などは、適切なディメンションのマトリックスである。ここで右多項式マトリックス分解と左多項式マトリックス分解は等価なので、次の関係が成立しなければならない。

$$A^*(s)B(s) = B^*(s)A(s) \quad (1.26)$$

マトリックスでは交換則は成立しないので、 $A^*(s) \cdot B^*(s)$  は  $A(s) \cdot B(s)$  とは異なったものになる。

多項式は加法・減法・乗法のみ可能であるので、環 (Ring) である。伝達関数は加法・減法・乗法・除法すべてが可能なので、体 (Field) である。ラプラス変換を使用していないので、微分方程式は非線形でも支障ない。したがって  $sy^2$  のような表現を用いてもよく、これは  $2ysy$  と解釈される。

また  $d^{1.4}y/dt^{1.4}$  のような非整数次 (分数次) の微分は、 $s^{1.4}y$  のように表す。非整数次微分方程式で表される動的システムは、現在分数次システム (Fractional Order System, FOS) と呼ばれている。このようなシステムへの拡張は、多項式に  $s^{1.4}$  のような  $s$  の非整数次項を入れるだけでよい。

伝達関数表現は、古典制御で広く用いられている。この表現は簡潔で、扱い易いが、次の3個の欠点がある。第1の欠点はこれがラプラス変換を前提としていることである。ラプラス変換は電気工学者にはよく知られているが、他の分野の専門家にはよく知られていないので、他の分野の専門家、例えば物理学者・化学者・生物学者・経済学者などが、自分の分野の動的システムに制御を導入しようとする場合には障害になる。またラプラス変換はシステムが線形であることを前提としている。ラプラス変換を非線形システムに適

用することは、可能ではあるが複雑になりすぎる。実世界のシステムは一般的には非線形で、これを線形システムと考えるのは近似に過ぎない。非線形を最初から除いて考えると、実システムの重要な要素を失ってしまうことになりかねない。

第2の欠点は、伝達関数表現が零初期条件を前提としていることである。このため初期条件が零でない時のシステムの解析が、不可能ではないとしても厄介になる。

第3の欠点は伝達関数表現に付き物の曖昧さである。今  $u$  から  $y$  への伝達関数が次のように与えられたとする。

$$Y(s)/U(s) = G(s) = s/s. \quad (1.27)$$

ここで  $Y(s)$  と  $U(s)$  は  $y$  と  $u$  のラプラス変換である。これを右多項式分解で表せば、

$$sx = u, \quad y = sx. \quad (1.28)$$

となり、さらに次のようになる。

$$y = u. \quad (1.29)$$

この結果が正しいことは、 $u$  の積分の微分が  $y$  となっていることより明らかである。左多項式分解では

$$sy = su. \quad (1.30)$$

であるから、 $s(y - u) = 0$  となり、さらに

$$y = u + constant. \quad (1.31)$$

となる。この結果が正しいことは、式 (1.30) が単に  $y$  の微分と  $u$  の微分とが等しいとしているだけであることより明らかである。この例は伝達関数の一般的な性質を示すもので、伝達関数の曖昧さを示している。伝達関数の分子項と分母項に共通項がある場合、その共通項は問題の解釈に如何によって、除去できる場合と残さねばならない場合とがある。この曖昧さは通例極零キャンセル問題として理解されている。

状態空間表現は現代制御で一般的に用いられている。状態空間表現は多項式表現と密接な関係がある。式 (1.22) に示す右多項式分解の状態空間表現を求めるには、次の状態変数を導入すればよい。

$$x_2 = \ddot{x} = s^2x, \quad x_1 = \dot{x} = sx, \quad x_0 = x. \quad (1.32)$$

状態空間表現は次のようになる。

$$[\dot{x}_2; \dot{x}_1; \dot{x}_0] = A[x_2; x_1; x_0] + Bu, \quad y = C[x_2; x_1; x_0] + Du, \quad (1.33)$$

$$A = \begin{bmatrix} -a_2/a_3 & -a_1/a_3 & -a_0/a_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1/a_3 \\ 0 \\ 0 \end{bmatrix},$$

$$C = [b_2 \quad b_1 \quad b_0], \quad D = [0].$$

ここで  $[\dot{x}_2; \dot{x}_1; \dot{x}_0]$  は列ベクトルで、 $[\dot{x}_2 \quad \dot{x}_1 \quad \dot{x}_0]^T$  に等しい。この表現は MATLAB のベクトル表現を流用したものである。右多項式分解から求められた状態空間表現は、可制御標準形と呼ばれる。ここで右多項式分解の中の  $x$  を基礎状態変数 (Basic State Variable) と呼ぶこととする。状態変数はすべて  $x$  とその微係数で表されるからである。

式 (1.19)(1.20)(1.21) に相当する左多項式分解は次のように導出される。まずこれらの式を次のようにまとめる。

$$(a_3s^3 + a_2s^2 + a_1s + a_0)y = (b_2s^2 + b_1s + b_0)u. \quad (1.34)$$

これを次のように整理する。

$$\frac{a_0}{a_3}y - \frac{b_0}{a_3}u + s\left[\frac{a_1}{a_3}y - \frac{b_1}{a_3}u + s\left(\frac{a_2}{a_3}y - \frac{b_2}{a_3}u + sy\right)\right] = 0. \quad (1.35)$$

ここで  $y$  を状態  $z_2$  とする。また ( ) の内部の項を状態  $z_1$ 、[ ] 内部の項を状態  $z_0$  とする。このようにすると、次のような関係式が得られる。

$$y = z_2, \quad (1.36)$$

$$z_1 = sz_2 + (a_2/a_3)z_2 - (b_2/a_3)u, \quad (1.37)$$

$$z_0 = sz_1 + (a_1/a_3)z_2 - (b_1/a_3)u, \quad (1.38)$$

$$0 = sz_0 + (a_0/a_3)z_2 - (b_0/a_3)u. \quad (1.39)$$

この結果可観測標準形と呼ばれる状態空間表現がえられる。

$$[\dot{z}_2; \dot{z}_1; \dot{z}_0] = F[z_2; z_1; z_0] + Gu, \quad y = H[z_2; z_1; z_0] + Ju. \quad (1.40)$$

$$F = \begin{bmatrix} -a_2/a_3 & 1 & 0 \\ -a_1/a_3 & 0 & 1 \\ -a_0/a_3 & 0 & 0 \end{bmatrix}, \quad G = \begin{bmatrix} b_2/a_3 \\ b_1/a_3 \\ b_0/a_3 \end{bmatrix},$$

$$H = [1 \quad 0 \quad 0], \quad J = [0].$$

またこれらの新しい状態変数は、式 (1.36)(1.37)(1.38) において、 $u$  を  $A(s)x$  で置き換え、 $y = z_2$  を  $B(s)x$  で置き換えることにより、右多項式分解の  $x$  によって次のように表される。

$$\begin{bmatrix} z_2 \\ z_1 \\ z_0 \end{bmatrix} = \begin{bmatrix} b_2s^2 + b_1s + b_0 \\ (s + a_2/a_3)(b_1s + b_0) - \{(a_1/a_3)s + a_0/a_3\}b_2 \\ \{s^2 + (a_2/a_3)s + (a_1/a_3)\}b_0 - (a_0/a_3)(b_2s + b_1) \end{bmatrix} x. \quad (1.41)$$

これより次のような状態変数間の変換式が得られる。

$$\begin{bmatrix} z_2 \\ z_1 \\ z_0 \end{bmatrix} = \begin{bmatrix} b_2 & b_1 & b_0 \\ b_1 & b_0 + (a_2/a_3)b_1 - (a_1/a_3)b_2 & (a_2/a_3)b_0 - (a_0/a_3)b_2 \\ b_0 & (a_2/a_3)b_0 - (a_0/a_3)b_2 & (a_1/a_3)b_0 - (a_0/a_3)b_1 \end{bmatrix} \begin{bmatrix} x_2 \\ x_1 \\ x_0 \end{bmatrix} \quad (1.42)$$

このようにして、右・左多項式分解に対する状態空間表現が得られる。状態変数はすべて基礎状態変数  $x$  をもとにして定義される。状態変数間の変換関係式も導出された。多項式表現は状態空間表現と1対1の対応関係を持っている。右多項式分解は可制御標準形に、左多項式分解は可観測標準形に対応している。

このように状態空間表現は、曖昧さが無く正確である。非線形系への拡張も容易である。多入力多出力(MIMO)系も、1入力1出力(SISO)系と同じように扱える。非零の初期条件も容易に扱える。状態空間表現は、計算機による計算には良く適合している。しかしこれは設計に用いるには、次に示す3個の問題点より、適切なものとは考えられない。

第1の問題点は、状態空間表現では多様性が多過ぎることである。マトリックスの要素数は次数の二乗に比例し、多項式表現の係数が単に次数に比例するのに比べ遥かに多い。状態空間表現では、同じ入出力関係もつ場合でも、状態変数の選び方で、無数の表現法が可能になる。多項式表現では、右多項式分解と左多項式分解の2形式しか許されていない。制御器が設計されたとき、多項式表現ならば設計者はその制御器の意味を容易に理解できる。しかし状態空間表現では、要素の数が多く上、等価な表現が幾らでもあるので、制御器の意味を理解することは殆ど不可能である。

第2の問題点は、状態空間表現が不必要に硬直的なことである。状態空間表現では、次数が予め定められ、また表現はモニック、すなわち最高次項の係数が1としている。実際の設計では、次数も最高次項の係数も、設計の初期段階では分からないのが通例で、設計が進むにつれて変更されて行く。従ってこのような硬直性は設計では大変厄介なものになる。

第3の問題点は、システムを直列または並列に結合しようとするとき、状態空間表現では面倒なことである。多項式表現では、これらは多項式の乗算・加算に相当するので、容易に行うことができる。

多項式表現は、伝達関数表現・状態空間表現の長所を保ちながら、それらの問題点を上手に回避している。伝達関数表現と比較すると、差異は分母・分子を別々に扱うことだけなので、伝達関数表現の扱いやすさが受け継がれており、しかもラプラス変換の必要性・零初期条件の必要性・曖昧さなどの欠点は回避されている。状態空間表現と比較すると、主な差異は可制御標準形と可観測標準形のみが使用を許されていることであり、状態空間表現に見られる表現の正確さを受け継ぎながら、過剰な多様性・硬直性・直並列の困難さなどの欠点を回避している。

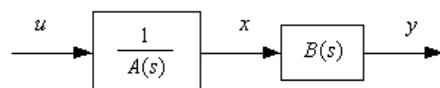


Figure 1.10: Block diagram of polynomial expression, RPF

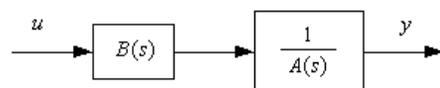


Figure 1.11: Block diagram of polynomial expression, LPF

## 1.4 設計の概要

係数図法による設計法の説明は、本書全体の課題である。設計の進め方は制御対象の性質と制御の要求仕様によって、大きく異なるものであるから、意味のある説明は特定の例題が与えられてのみ可能になる。それらについては後の章で詳しく説明することにし、本節では設計の進め方の概要のみを簡単に説明する。

まず係数図法で用いる数学的モデルについて説明する。次に設計の基礎とする係数図法入出力関係を求め、その意味について検討する。とくに極零キャンセルについては詳しく検討する。最後に係数図法設計ステップを一般的な形で説明する。

係数図法で用いる数学的モデルについて以下に説明する。係数図法設計に用いる標準ブロック線図は、1入力1出力 (Single-Input-Single-Output, SISO) の場合、Fig. 1.12 に示される。多入力多出力 (Multi-Input-Multi-Output, MIMO) への拡張も、適切な解釈の変更で可能であるが、説明を簡単にするためここでは取り上げない。制御対象の方程式は次のように与えられている。

$$A_p(s)x = u + d, \quad y = B_p(s)x. \quad (1.43)$$

ここで  $u, y, d$  は、入力・出力・外乱である。変数  $x$  は基礎状態変数と呼ばれている。また  $A_p(s) \cdot B_p(s)$  は制御対象伝達関数  $G_p(s)$  の分母・分子多項式と呼ばれる。この表現は右多項式分解 (RPF) で、状態空間表現では可制御標準形に対応する。この表現では状態変数はすべて基礎状態変数  $x$  とその高次微係数で表される。

制御器方程式は次のとおりである。

$$A_c(s)u = B_a(s)y_r - B_c(s)(y + n). \quad (1.44)$$

ここで  $y_r, n$  は目標値入力・出力雑音である。 $A_c(s)$  は制御器伝達関数の分母多項式であり、 $B_a(s)$  は目標値分子多項式、 $B_c(s)$  はフィードバック分子多項式と呼ばれる。制御器伝達関数は2個の分子多項式を持つので、2自由度系とよばれる。ここで  $G_c(s) = B_c(s)/A_c(s)$  はフィードバック信号に対する制御器伝達関数、 $G_a(s) = B_a(s)/A_c(s)$  は目標値信号に対する制御器伝達関数である。この表現は左多項式分解 (LPF) で可観測標準形に対応している。

ここで式 (1.44) より、式 (1.43) を用いて  $y$  と  $u$  を消去すると次の式が得られる。

$$P(s)x = B_a(s)y_r + A_c(s)d - B_c(s)n. \quad (1.45)$$

ここで  $P(s)$  は特性多項式で次式で与えられる。

$$P(s) = A_c(s)A_p(s) + B_c(s)B_p(s). \quad (1.46)$$

係数図法の数学的モデルでは、制御対象は右多項式分解 (RPF) で制御器は左多項式分解 (LPF) で表す。制御器は 2 自由度系である。このような選択で、後の章で示すように、設計は簡素化され一貫性を持つものになる。外乱  $d$  は制御対象入力のみ限定されている。外乱が制御対象の他の部分にも加えられているときは、制御対象入力での等価外乱に置き換えなければならない。同様に雑音  $n$  は制御対象出力に限定されているので、他の部分に加えられた雑音はすべて制御対象出力での等価雑音に置き換えなければならない。

係数図法入出力関係は、式 (1.45) より導出される。係数図法では基礎状態変数  $x$  の応答が最も重要と考えられている。制御対象出力  $y$  と制御対象入力  $u$  は、 $x$  を用いて次のように表される。

$$y = B_p(s)x, \quad u = A_p(s)x - d. \quad (1.47)$$

これより入出力関係は次のようにまとめられる。

$$P(s)x = B_a(s)y_r + A_c(s)d - B_c(s)n, \quad (1.48)$$

$$P(s)y = B_p(s)[B_a(s)y_r + A_c(s)d - B_c(s)n], \quad (1.49)$$

$$P(s)u = A_p(s)[B_a(s)y_r - B_c(s)n] - B_c(s)B_p(s)d. \quad (1.50)$$

この系は 3 入力 ( $y_r, d, n$ ) 3 出力 ( $x, y, u$ ) なので、9 個の伝達関数がある。しかし設計すべき多項式は 4 個 ( $P(s), A_c(s), B_c(s), B_a(s)$ ) しかないので、4 個の関係式だけで十分である。係数図法では次の 4 個の基本式を標準として用いる。

$$P(s)x = P(0)y_r, \quad (1.51)$$

$$P(s)y = B_p(s)B_a(s)y_r, \quad (1.52)$$

$$P(s)y = B_p(s)A_c(s)d, \quad (1.53)$$

$$P(s)(-y) = B_p(s)B_c(s)n. \quad (1.54)$$

式 (1.51) は  $B_a(s) = P(0)$  のときの、 $y_r$  に対する  $x$  の応答である。ここで  $P(0)$  は  $P(s)$  の 0 次の係数  $a_0$  である。この関係に対応する伝達関数  $T_0(s)$  を  $P(s)$  の 0 次正準伝達関数と呼ぶが、このことについてはまた後章で説明する。この伝達関数は特性多項式を規定するもので安定性のよい尺度である。式 (1.52) は目標値追従特性を示すもので、これに対応する伝達関数は  $W(s)$  である。式 (1.53) は外乱抑制特性を示すもので、これに対応する伝達関数は  $T_{dy}(s)$  である。式 (1.54) は雑音減衰特性を示すもので、これに対応する伝達関数は相補感度関数  $T(s)$  である。この関係式はロバスト性を調べるのに有効である。

係数図法の設計では、これらの 4 個の基本式を性能の仕様としている。まず式 (1.51, 1.53, 1.54) の仕様を満足するように  $P(s)$  を設計し、次に  $B_a(s)$

を調整して、式 (1.52) の仕様を満足させる。外乱抑制特性と目標値追従特性には、独立して別々に仕様を与えることができる。 $P(s)$  の設計はフィードバック制御の設計であり、 $B_a(s)$  の設計はフィードフォワード制御の設計である。

実際の設計では仕様が上記とは異なった形で与えられるのが通例であるので、係数図法の設計を始める前にこれらの仕様を上記 4 個の関係式に翻訳することが必要である。この結果与えられた仕様の曖昧さや矛盾が明らかになり、それらが自動的に修正されることになる。

極零キャンセルは伝達関数を用いる従来の設計法では一般的に用いられてきた。この問題をより丁寧に検討すると、性能仕様を与える 4 個の基本式の重要性が明らかになる。Fig. 1.12 に示す系で、開ループ伝達関数  $G(s)$  の分母多項式  $A_{cp}(s)$  と分子多項式  $B_{cp}(s)$  は次のように表される。

$$A_{cp}(s) = A_c(s)A_p(s), \quad B_{cp}(s) = B_c(s)B_p(s). \quad (1.55)$$

これより次の関係が得られる。

$$G(s) = B_{cp}(s)/A_{cp}(s), \quad (1.56)$$

$$P(s) = A_{cp}(s) + B_{cp}(s), \quad (1.57)$$

$$T(s) = B_{cp}(s)/P(s), \quad (1.58)$$

$$S(s) = A_{cp}(s)/P(s). \quad (1.59)$$

ここで  $S(s)$  は感度関数と呼ばれている。極零キャンセルでは  $B_c(s)$  は  $A_p(s)$  と共通因子を持ち、零極キャンセルでは  $A_c(s)$  は  $B_p(s)$  と共通因子を持つ。何れの場合でも  $A_{cp}(s)$  と  $B_{cp}(s)$  が共通因子を持つ。このとき  $G(s)$  を用いる古典制御でも、 $T(s)$  と  $S(s)$  を用いる  $H_\infty$  混合感度法でもこの共通因子は不可視である。しかしこの共通因子は  $P(s)$  には残っており、その影響は他の入出力関係や初期条件応答では現れてくる。共通因子が不安定ならば、システム全体が不安定になることはよく知られている。しかし振動的な極・零のように実部が小さいときは、安定な共通因子でもロバスト性が低下し、不安定になることは余り知られていない。このようなキャンセルには必ず小さな誤差が付き物で、これがロバスト性が低くなった系を不安定にするのである。

係数図法では設計は特性多項式  $P(s)$  が設計目標であるから、極零キャンセルを用いることなく設計を進める。また極零キャンセルが用いられるときでも、このような極零キャンセルが容認できるものか否かは  $P(s)$  を調べれば簡単に分かる。容認できないような極零キャンセルでは安定度指標が極端に小さくなるからである。

係数図法設計ステップについて次に簡単に説明する。設計を始める前に仕様の検討を行わなければならない。一般に仕様は曖昧で一貫性が無いからであ

る。検討の結果これらの仕様は4個の基本式 ( $T_0(s)$ ,  $W(s)$ ,  $T_{dy}(s)$ ,  $T(s)$ ) の仕様に翻訳される。これらはさらに4個の多項式 ( $P(s)$ ,  $A_c(s)$ ,  $B_c(s)$ ,  $B_a(s)$ ) の仕様に翻訳される。とくに特性多項式  $P(s)$  における安定度指標と等価時定数の許容範囲を定めることが最も重要である。このようにして仕様が明確になり一貫性のあるものになる。

第1のステップは制御対象を右多項式分解で表すことで、分母・分子多項式は次のようになる。

$$A_p(s) = d_{n_p} s^{n_p} + \cdots + d_1 s + d_0, \quad (1.60)$$

$$B_p(s) = n_{m_p} s^{m_p} + \cdots + n_1 s + n_0. \quad (1.61)$$

ここで  $d_i$  と  $n_i$  はこれらの多項式の係数であり、 $n_p$  と  $m_p$  はそれらの次数である。係数図法ではこのような制御対象の次数は  $m_p/n_p$  次というように表すものとする。

第2のステップは制御器を左多項式分解で表し、制御構造を定めることである。制御構造とは、制御器の次数と制御器パラメータがとりうる許容範囲を意味するものとする。制御器の分母・分子多項式は次のように表される。

$$A_c(s) = l_{n_c} s^{n_c} + \cdots + l_1 s + l_0, \quad (1.62)$$

$$B_c(s) = k_{m_c} s^{m_c} + \cdots + k_1 s + k_0, \quad (1.63)$$

$$B_a(s) = m_{m_a} s^{m_a} + \cdots + m_1 s + m_0. \quad (1.64)$$

ここで  $l_i$ ,  $k_i$ ,  $m_i$  はこれらの多項式の係数であり、 $n_c$ ,  $m_c$ ,  $m_a$  はこれらの次数である。係数図法では制御器の次数は  $m_c/n_c$  次と呼ぶ。目標値分子多項式  $B_a(s)$  は  $P(s)$  には何の影響も与えないので、その次数については言及しない。制御器の形が予め知られている場合には、制御構造は自動的に定まる。例えば1.2節「簡単な設計問題」で示したPD制御器は次のような制御構造を持っている。

$$n_c = 0, \quad m_c = 1, \quad m_a = 0, \quad l_0 = 1, \quad k_1 > 0, \quad k_0 = m_0 > 0. \quad (1.65)$$

このPD制御器は1/0次制御器である。またPID制御器は次のような制御構造を持っている。

$$n_c = 1, \quad m_c = 2, \quad m_a = 2, \quad (1.66)$$

$$l_1 = 1, \quad l_0 = 0, \quad k_2 = m_2 > 0, \quad k_1 = m_1 > 0, \quad k_0 = m_0 > 0.$$

PID制御器は2/1次制御器である。

一般に制御器の形は、過去の設計経験から予め知られている。しかし新しい設計では、制御構造を定めるには十分な注意が必要である。制御器の次数が制御対象の次数の影響を受けることはよく知られている。しかし実際には制御対象の性質に大きく影響される。ある制御対象は次数が高くて制御

し易く、簡単なPI制御器(1/1次)で十分である。ある種の制御対象では次数は低くても制御が難しく、制御器の次数は制御対象の次数に近くなる。さらに難しい制御対象では制御器は非最小位相形や不安定形になり、ロバスト性が極端に劣化する。このような場合は閉ループ系の安定性とロバスト性がトレードオフの形になり、どちらかを犠牲にしなければならない。制御器の次数はアクチュエータとセンサの性質にも影響される。アクチュエータとセンサは制御対象の一部と考えられ、これらを適切に選ぶことにより制御を容易にすることができる。

制御構造の設計では、外乱抑制  $T_{dy}(s)$  の低周波特性と、相補感度関数  $T(s)$  の高周波特性が最も重要である。制御構造の設計は複雑なので、詳細な扱いは後章に譲り、設計例により具体的な方法を明らかにする。

第3のステップは制御対象・制御器の分母・分子多項式と特性多項式の関係を求めることである。この関係は式(1.46)で示したが、次に再掲する。

$$A_c(s)A_p(s) + B_c(s)B_p(s) = P(s). \quad (1.67)$$

この方程式は一般に Diophantine 方程式と呼ばれている。これは式(1.62, 1.63)により、次のように形に展開される。

$$\left(\sum_{i=0}^{n_c} l_i s^i\right)A_p(s) + \left(\sum_{i=0}^{m_c} k_i s^i\right)B_p(s) = P(s) = \sum_{i=0}^n a_i s^i. \quad (1.68)$$

これはさらにマトリックス形式で表される。なお式の表現を分かり易くするため、制御対象は1/3次制御器はPIDとしている。

$$[l_1 \ l_0 \ k_2 \ k_1 \ k_0]M = [a_4 \ a_3 \ a_2 \ a_1 \ a_0], \quad (1.69)$$

$$M = \begin{bmatrix} d_3 & d_2 & d_1 & d_0 & 0 \\ 0 & d_3 & d_2 & d_1 & d_0 \\ 0 & n_1 & n_0 & 0 & 0 \\ 0 & 0 & n_1 & n_0 & 0 \\ 0 & 0 & 0 & n_1 & n_0 \end{bmatrix},$$

$$l_1 = 1, \quad l_0 = 0.$$

マトリックス  $M$  はシルベスタ (Sylvester) マトリックスと呼ばれる。マトリックス  $M$  は必ずしも正方ではないこと、既知のパラメータが方程式の両側にあることに注意が必要である。もし  $M$  が正方で左側のパラメータが総て未知ならば、この設計法は極配置設計法と等価になる。

第4のステップは特性多項式の係数が適切な形になるように、未知の制御器パラメータを決定することである。この段階で制御器と特性多項式が同時に定まることになる。この段階が係数図法で最も重要で、関連する数学についての準備の後に、後の章で詳しく述べることにする。

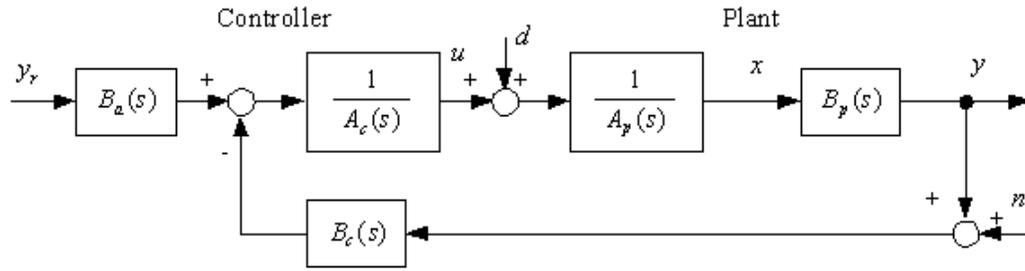


Figure 1.12: Standard block diagram of CDM

第5のステップは目標値追従特性を満足するように目標値分子多項式の係数  $m_i$  を定めることである。上記の5ステップが係数図法での一般的な設計ステップであるが、第4ステップが係数図法の核心である。

## 1.5 制御構造

制御系の設計法は種々あり、その結果同じ制御対象、同じ設計仕様に対して、異なった制御器が設計される。しかしこのような制御器に対しても、必ず等価な係数図法制御器が存在する。従ってこのような制御器を係数図法制御器に変換して、共通な形式のもとで比較すると、これらの設計法の特徴が明確になる。ここで一般の制御器の特徴を明確にするために、その制御器の制御構造という言葉を用いるが、これはその制御器と等価な係数図法制御器の次数と制御器パラメータの許容範囲と定義する。

本節では、各種の方法で設計された制御器の制御構造を検討する。先ず係数図法の制御構造について説明し、制御構造の意味を明確にする。次に古典制御について検討し、その制御構造と限界を明らかにする。続いてLQG制御について検討し、その制御構造を明らかにし、係数図法の制御構造と比較する。どのようなLQG制御器も、係数図法制御器で表すことができるが、その逆は成立しない。次に拡大LQRについて検討する。この構造は余り知られていないが、これにより低次の制御器が設計できる。一つの係数図法制御器に対して、同じ特性の拡大LQRが必ず存在するので、拡大LQRは係数図法の現代制御表現とも言える。最後にQパラメタリゼーション付きLQGについて検討する。この構造は $H_\infty$ 制御に関連して考えられてきたものであるが、この構造も係数図法表現で表すことができる。

係数図法の制御構造については1.4節で既に述べた。しかし他の設計法との比較を容易にするため、重要な点について更に説明を加える。係数図法の標準ブロック線図を Fig. 1.13 に再掲する。制御器の数学的モデルは次のとおりである。

$$G_c(s) = B_c(s)/A_c(s), \quad G_a(s) = B_a(s)/A_c(s), \quad (1.70)$$

$$A_c(s) = l_{n_c} + \cdots + l_1 s + l_0, \quad (1.71)$$

$$B_c(s) = k_{m_c} + \cdots + k_1 s + k_0, \quad (1.72)$$

$$B_a(s) = m_{m_a} + \cdots + m_1 s + m_0, \quad (1.73)$$

ここで $G_c(s)$ はフィードバック信号に対する制御器伝達関数、 $G_a(s)$ は目標値信号に対する制御器伝達関数である。この制御器の次数は、 $m_c/n_c$ 次と呼ぶことにする。係数図法では次数は任意であるが、一般には2/2次で十分である。3/3次あるいは更に高次の制御器が必要になるのは極めて稀である。制御器パラメータは一般には正であるが、零や負になることもあり得る。特に $l_1 = 1$ ,  $l_0 = 0$ になる場合、即ち

$$G_c(s) = \frac{k_2 s^2 + k_1 s + k_0}{l_2 s^2 + s}, \quad (1.74)$$

のような場合に、適用範囲が一番広い。この制御器は一般化PIDとも呼んでよいものである。これはPID制御器に似ているが、 $k_2$ と $k_1$ 必ずしも正ではなく、また $l_2$ が必ずしも小さい値ではなく設計によって定められると言う点で、PID制御器とは異なる。このような一般化により振動系の制御が大変容易になる。実際、振動系の制御はPIDやLQGでも難しいと考えられてきている。

古典制御では、二つの形の制御器が使われている。一つはPID制御器で次のようなものである。

$$G_c(s) = K\left(1 + \frac{1}{T_I s} + T_D s\right), \quad (1.75)$$

ここでパラメータはすべて正である。PID制御器は2/2次係数図法制御器とは、次のような制限のもとで等価である。

$$l_2 = 0, \quad l_1 = 1, \quad l_0 = 0, \quad (1.76)$$

$$k_2 = KT_D, \quad k_1 = K, \quad k_0 = K/T_I.$$

このような制限のため、 $k_1$ のみを零にしたり、弱い負値をいれたりすることが困難になる。このようなパラメータ選択は振動的な制御対象の制御で必要になることが多い。実際PID制御は、多数の時間遅れ要素を持つ制御対象に対して有効である。

今一つの形は進相遅相補償器で次のようなものである。

$$G_c(s) = K \frac{1 + \alpha_1 T_1 s}{1 + T_1 s} \frac{1 + \alpha_2 T_2 s}{1 + T_2 s}, \quad (1.77)$$

$$\alpha_1 > 1, \quad 0 < \alpha_2 < 1, \quad 0 < T_1 < T_2,$$

ここで前の部分が進相補償器で、後の部分が遅相補償器である。進相遅相補償器は2/2次係数図法制御器とは、次のような制限のもとで等価である。

$$l_2 = T_1 T_2, \quad l_1 = T_1 + T_2, \quad l_0 = 1, \quad (1.78)$$

$$k_2 = K(\alpha_1 \alpha_2 T_1 T_2), \quad k_1 = K(\alpha_1 T_1 + \alpha_2 T_2), \quad k_0 = K.$$

ここでは分母・分子多項式の根が負実で、複素根も正実根も除外されている点で、PID制御器よりも制限が厳しくなっている。進相遅相補償器は制御対象の極・零が負実の場合、即ち制御対象が折れ線近似ボデー線図で簡単に表現できるような場合に有効である。

古典制御では、制御器の次数は十分であるが、パラメータの選択巾が制限されているため、制御の有効性が十分ではない。また古典制御で、2自由

度系という名前が 1990s 年代に一般に使用される以前から、2 自由度系が実際に使用されていたことは注目に値する。制御器は次の形になっている。

$$u = G_a(s)y_r - G_c(s)y, \quad (1.79)$$

ここで  $G_a(s)$  は、前に述べたように、目標値信号に対する制御器伝達関数である。

LQG 制御 (Linear Quadratic Gaussian) の設計法は、Fig. 1.14 のブロック線図に基づいている。このブロック線図は Fig. 1.13 に示す係数図法標準ブロック線図に変換できる。制御対象の伝達関数は次のようになる。

$$y = C(sI - A)^{-1}Bu = \frac{C \operatorname{adj}(sI - A)B}{\det(sI - A)}u. \quad (1.80)$$

制御器の方程式は次のとおりである。

$$(sI - A)\hat{x} = L(y - My_r - C\hat{x}) + Bu, \quad (1.81)$$

$$u = Ny_r - K\hat{x}. \quad (1.82)$$

ここで式 (1.81) の  $u$  を式 (1.82) の  $u$  で置換する。その結果  $\hat{x}$  は  $y$  と  $y_r$  で表されることになる。この  $\hat{x}$  を式 (1.82) に用いると、制御器の伝達関数が次のように求められる。

$$u = Ny_r - K(sI - A + LC + BK)^{-1}[L(My_r - y) - BNy_r]. \quad (1.83)$$

この式を簡単化するため次の恒等式を用いる。

$$P_o(s) = \det[sI - A + LC] = \det \begin{bmatrix} sI - A + LC + BK & B \\ K & 1 \end{bmatrix} \quad (1.84)$$

$$= \det[sI - A + LC + BK][1 - K(sI - A + LC + BK)^{-1}B].$$

結果が次のようになる。

$$u = \frac{P_o(s)Ny_r + K \operatorname{adj}(sI - A + BK + LC)L(My_r - y)}{\det(sI - A + LC + BK)}. \quad (1.85)$$

制御対象・制御器多項式が次のように得られる。

$$A_p(s) = \det(sI - A), \quad (1.86)$$

$$B_p(s) = C \operatorname{adj}(sI - A)B, \quad (1.87)$$

$$A_c(s) = \det(sI - A + BK + LC), \quad (1.88)$$

$$B_c(s) = K \text{adj}(sI - A + BK + LC)L, \quad (1.89)$$

$$B_a(s) = P_o(s)N + B_c(s)M. \quad (1.90)$$

関連する特性多項式と Diophantine 方程式は次のようになる。

$$P_c(s) = \det(sI - A + BK), \quad (1.91)$$

$$P_o(s) = \det(sI - A + LC), \quad (1.92)$$

$$P(s) = P_o(s)P_c(s) = A_c(s)A_p(s) + B_c(s)B_p(s). \quad (1.93)$$

設計にあたっては、状態フィードバック特性多項式  $P_c(s)$  と、オブザーバ特性多項式  $P_o(s)$  が最初に定められる。次に  $K$  と  $L$  が Ackerman の公式などの方法を用いて計算される。これにより  $A_c(s)$  と  $B_c(s)$  が式 (1.88, 1.89) を用いて求められる。しかし更に易しい方法は式 (1.93) の Diophantine 方程式を解き、 $A_c(s)$  を  $B_c(s)$  直接求めることである。

LQG 制御構造の第 1 の欠点は、制御器の次数が  $(n_p - 1)/n_p$  次と固定されていて、低次の制御器は最初から除外されていることである。第 2 の欠点は制御器パラメータが状況によって正にも負にもなり得ることである。もし  $B_p(s)$  の性質が  $P_c(s)$  の選択に適切に反映されていないと、設計された制御器のパラメータが負になりロバスト性が悪化する。LQG 設計ではロバスト性の保障は全くない。以上の二つの欠点により、一般的な理解とは異なるが、LQG は良い設計法とは考えられない。また  $B_a(s)$  は、 $M$  と  $N$  で調整できるので LQG 制御は 2 自由度系である。

拡大 LQR (Augmented Linear Quadratic Regulator) 設計法は LOQ 設計法の改良である。記法を分かり易くするため、制御対象は  $2/3$  次とする。その分母・分子多項式は次のように与えられる。

$$A_p(s) = s^3 + d_2s^2 + d_1s + d_0, \quad (1.94)$$

$$B_p(s) = n_2s^2 + n_1s + n_0. \quad (1.95)$$

この系を 2 次拡大すると、拡大制御対象は次のようになる。

$$s^2u = v, \quad (1.96)$$

$$A_p(s)x = u, \quad (1.97)$$

$$y = B_p(s)x, \quad (1.98)$$

ここで  $v$  は拡大制御対象への入力である。新しく  $w = B_p(s)u$  なる変数を導入することにより、更に次のように書き換えられる。

$$s^2u = v, \quad (1.99)$$

$$w = B_p(s)u = (n_2s^2 + n_1s + n_0)u, \quad (1.100)$$

$$A_p(s)y = (s^3 + d_2s^2 + d_1s + d_0)y = w. \quad (1.101)$$

この拡大制御対象では、 $u_1 = su$ ,  $u_0 = u$ ,  $y_2 = s^2y$ ,  $y_1 = sy$ ,  $y_0 = y$  が状態変数となる。状態方程式は次のようになる。

$$[\dot{u}_1; \dot{u}_0; \dot{y}_2; \dot{y}_1; \dot{y}_0] = A[u_1; u_0; y_2; y_1; y_0] + Bv, \quad (1.102)$$

$$y = C[u_1; u_0; y_2; y_1; y_0],$$

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ n_1 & n_0 & -d_2 & -d_1 & -d_0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \\ n_2 \\ 0 \\ 0 \end{bmatrix},$$

$$C = [0 \ 0 \ 0 \ 0 \ 1].$$

ここで次のような状態フィードバック制御を考えれる。

$$v = (m_2s^2 + m_1s + m_0)y_r - (l_1u_1 + l_0u_0) - (k_2y_2 + k_1y_1 + k_0y_0). \quad (1.103)$$

ここで  $v = s^2u$  なることを考えると、この制御器は次のような係数図法制御器と同じものになることが分かる。

$$A_c(s)u = B_a(s)y_r - B_c(s)y, \quad (1.104)$$

$$A_c(s) = s^2 + l_1s + l_0,$$

$$B_c(s) = k_2s^2 + k_1s + k_0,$$

$$B_a(s) = m_2s^2 + m_1s + m_0.$$

この系のブロック線図は、外乱  $d$  と雑音  $n$  を加えて Fig. 1.15 に示される。これを Fig. 1.13 と比較すると、係数図法標準ブロック線図と等価であることが分かる。このように拡大 LQR は係数図法と全く同じの制御構造をもつことが分かる。係数図法制御器が設計されると、それと等価な拡大 LQR 制御器を見つけることができる。このような拡大 LQR 制御器は、係数図法制御器と同じものなので、低次でありまたロバスト性が保障されている。しかし逆に拡大 LQR 制御器を先に設計して、それから係数図法制御器を導くのは、非常に難しい。何故ならば現在の LQR 設計理論には、信頼できる重み関数選択法が与えられていないからである。

Q パラメタリゼーション付き LQG は LQG 制御器の変形である。この構造は  $H_\infty$  制御器設計で、総ての安定化制御器を求めるという過程の中で導かれたものである。このブロック線図は Fig. 1.16 に示されている。ここでは

$\eta$  から  $r$  への経路が標準 LQR に付け加わっている。 $Q$  パラメータは  $s$  の有理関数であるが、これを分子多項式  $Q_n(s)$  と分母多項式  $Q_d(s)$  の比で表す。多項式で表したブロック線図は Fig. 1.17 で表されるが、その導出の過程を次に示す。関連する式は次のとおりである。

$$\eta = y - My_r - C\hat{x}, \quad (1.105)$$

$$(sI - A)\hat{x} = L\eta + Bu, \quad (1.106)$$

$$u = Ny_r - K\hat{x} + r, \quad (1.107)$$

$$r = Q_d^{-1}(s)Q_n(s)\eta. \quad (1.108)$$

式 (1.105) の  $\hat{x}$  を式 (1.106) を用いて消去し、整理すると次のようになる。

$$[1 + C(sI - A)^{-1}L]\eta = y - My_r - C(sI - A)^{-1}Bu. \quad (1.109)$$

式 (1.84) に示すような、行列から行列式への変換を行うと次のようになる。

$$P_o(s)\eta = A_p(s)(y - My_r) - B_p(s)u. \quad (1.110)$$

式 (1.106) の  $\eta$  を式 (1.105) を用いて消去し、また  $u$  を式 (1.107) を用いて消去して、次の関係が得られる。

$$(sI - A + LC + BK)\hat{x} = L(y - My_r) + B(Ny_r + r). \quad (1.111)$$

式 (1.107) の  $\hat{x}$  を式 (1.111) を用いて消去して、次の関係が得られる。

$$u = [1 - K(sI - A + LC + BK)^{-1}B](Ny_r + r) \quad (1.112)$$

$$+ K(sI - A + LC + BK)^{-1}L(My_y - y)$$

式 (1.84) と同じような行列から行列式への変換を行うと次のようになる。

$$A_c(s)u = B_a(s)y_r - B_c(s)y + P_o(s)r. \quad (1.113)$$

式 (1.110, 1.113) が Fig. 1.17 に示す最終結果である。

ここで Fig. 1.17 を、Fig. 1.18 に示すような係数図法標準ブロック線図に変換する。新しい制御器の多項式  $A_c^*(s)$ 、 $B_c^*(s)$ 、および  $B_a^*(s)$  は次のように表される。

$$A_c^*(s) = Q_d(s)A_c(s) + Q_nB_p(s), \quad (1.114)$$

$$B_c^*(s) = Q_d(s)B_c(s) - Q_n(s)A_p(s),$$

$$B_a^*(s) = MB_c^*(s) + NQ_d(s)P_o(s).$$

ここで  $Q_d(s) = 1$ 、 $Q_n(s) = 0$  ならば、この制御器は LQG 制御器を係数図法表現で表したものになる。新しい特性多項式  $P^*(s)$  は次のよう与えられる。

$$P^*(s) = A_c^*(s)A_p(s) + B_c^*(s)B_p(s) \quad (1.115)$$

$$= Q_d(s)[A_c(s)A_p(s) + B_c(s)B_p(s)] = Q_d(s)P(s).$$

ここで  $P^*(s)$  の極は LOG の総ての極の上に、 $Q_d(s)$  による極が加わったものになる。従って安定性は保証されている。新しい感度関数  $S^*(s)$  と相補感度関数  $T^*(s)$  は次のように表される。

$$S^*(s) = S(s) + \delta S, \quad T^*(s) = T(s) - \delta S, \quad (1.116)$$

$$\delta S = \frac{Q_n(s)B_p(s)A_p(s)}{Q_d(s)P(s)}.$$

これより Q パラメータは安定性を変えることなく、感度関数と相補感度関数を変えるものであることが分かる。Q パラメータは LQG 設計の小改修に用いるべきである。

極位置の移動が必要な大改修を行いたいときは、 $Q_d(s)$  と  $Q_n(s)$  を次のようにすればよい。

$$Q_d(s) = P_1(s) = A_{c1}(s)A_p(s) + B_{c1}(s)B_p(s), \quad (1.117)$$

$$Q_n(s) = A_{c1}(s)B_c(s) - B_{c1}(s)A_c(s),$$

ここで  $A_{c1}(s)$  と  $B_{c1}(s)$  は目標とする制御器の分母・分子多項式であり、 $P_1(s)$  は目標とする特性多項式である。ここで制御器多項式と特性多項式は次のように与えられる。

$$A_c^*(s) = A_{c1}(s)P(s), \quad (1.118)$$

$$B_c^*(s) = B_{c1}(s)P(s),$$

$$P^*(s) = P_1(s)P(s).$$

分子分母の  $P(s)$  が消去されるので、この系は目標の系と同じような振る舞いをすることが分かる。このような大改修は制御器を必要以上に複雑化するので、望ましいものではない。制御器を新しく設計する方がよほど簡単である。

結論として、Q パラメタリゼーション付き LQG は、特性多項式はその極に  $Q_d(s)$  の極を加えただけの形に残し、感度関数と相補感度関数を改修する場合にのみ用いるのがよい。特性多項式の大改修は奨められない。何れにせよ Q パラメタリゼーション付き LQG は係数図法表現で示される。

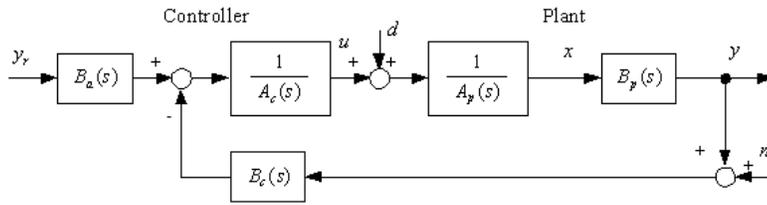


Figure 1.13: CDM standard block diagram

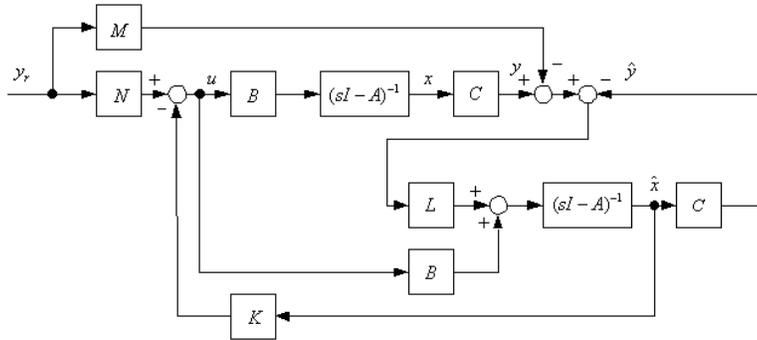


Figure 1.14: LQG block diagram

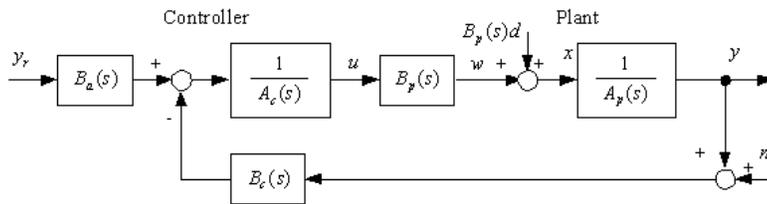


Figure 1.15: Augmented LQR block diagram

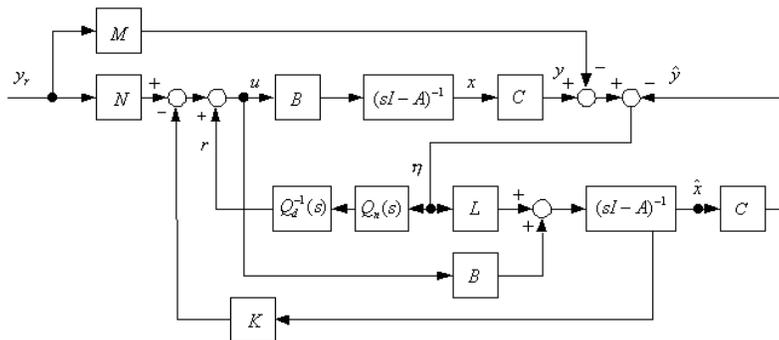


Figure 1.16: LQG with Q-parameterization

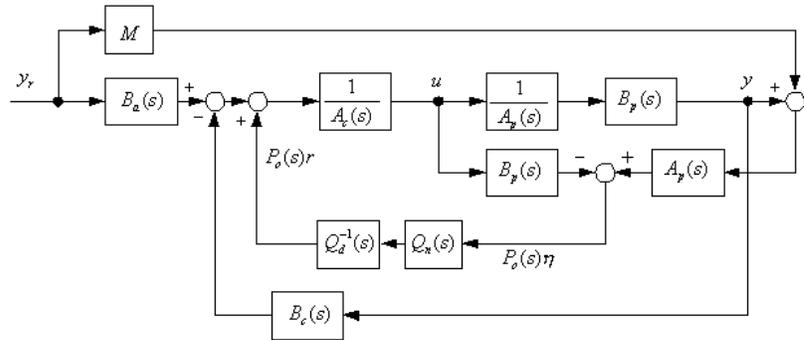


Figure 1.17: Polynomial block diagram for LQG with Q-parameterization

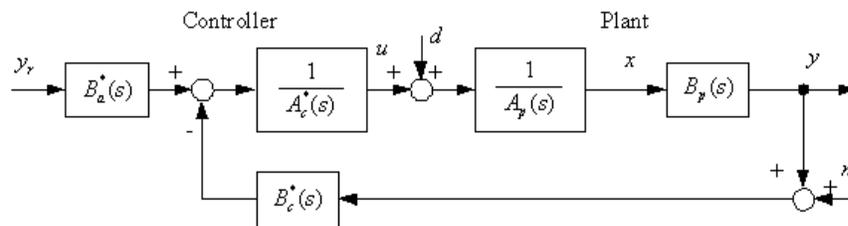


Figure 1.18: CDM standard block diagram for LQG with Q-parameterization

## 1.6 歴史的背景

係数図法は、制御系設計についての過去の色々な考え方や経験をもとにして発展してきたものである。しかし一言でいえば、係数図法は J. C. Maxwell の *On Governors* (Maxwell, 1868) での考え方の実用化である。その中の幾つかの重要な項目は開発小史に述べられている。係数図法には三つの特徴が過去の成果に付け加えられている。新しい3特徴ではこのような新しい特徴が加えられてきた過程を詳しく述べる。最後に係数図法の現状と将来展望について述べる。

係数図法に関連した開発小史について以下に述べる。最初に多項式設計法を扱ったのは J. C. Maxwell の *On Governors* (1868) と Routh の安定判別法 (1877) であって、特性多項式の係数を用いて安定性が検討されている (Franklin, 1944)。しかしこの安定判別法の原形は、その後も原形のまま伝えられて、これを有効な設計法に発展させるような特別な試みは、Lipatov による試みまで長い間なされなかった (Lipatov, 1978)。

1950年代には周波数応答法が制御系設計法に広く用いられていた。この時代には、良い設計のためには位相余裕・ゲイン余裕などの安定基準だけでは十分ではなく、開ループ伝達関数の周波数特性がかなり広い周波数領域にわたって適切な形状を持っていなければならないことが一般に知られていた (Tustin, 1958)。

Chestnut は有名な著書の中で、開ループ伝達関数の折れ線近似ボデー線図 (ゲイン線図) で、折れ点の位置とそこでの傾斜の変化の重要性を指摘し、それを基礎にした制御系設計法を提案した (Chestnut, 1951)。この提案は大変実用的なもので、1950年代のみならず現在でも産業界では広く用いられている。

折れ線近似ゲイン線図は 0 dB 線と -20 dB/decade の傾斜で交わるべきこと、折れ点での傾斜の変化は  $\pm 20$  dB/decade であるべきこと、二つの折れ点は2倍以上離れているべきこと、というような経験則が簡単な制御系の実用的設計には広く用いられていた。

このような簡単な制御系では、折れ点の分離の度合いは、ほぼ安定度指標に相当する。二つの折れ点が2倍以上離れているべきという経験則は、安定度指標  $\gamma_i$  が2以上ということに相当する。この経験則をより複雑な系に適用しようという努力の中から、折れ点よりも安定度指標を、ボデー線図よりも係数図を用いるようになり、ついに係数図法に到達したのである。

Graham は特性多項式の係数と過渡応答の関係を調べた結果、望ましい特性多項式の標準形を提案した (Graham, 1953)。これは一般には ITAE (Integral Time Absolute Error) 標準形と呼ばれている。この標準形の係数は、係数図法で提案している値と似ているが、より振動的である。

ITAE には、制御系設計に用いる場合、融通性が足りないことによる問題点がある。まず標準形はそれぞれの次数の特性多項式について、異なったもの

が与えられているので、設計の過程で次数が変化する場合は使いにくくなる。また標準形は一つしかなく、状況に応じてそれを変形する手段が与えられていないので、場合によっては不必要にロバスト性の低い制御器が設計される (Franklin, 1994, p. 534, Ex. 7.21)。このような問題点はあるけれども、この研究は実用的な設計の面では、大きな貢献を果たした。

同じ頃 Kessler は多ループ系の設計法を確立するため精力的な研究を行い、ケスラー正準多ループ構造 (*Kessler canonical multi-loop structure*) と呼ばれる標準形に到達した (Kessler, 1960)。これは Fig. 1.19 に示されるようなものである。提案された系は ITAE 標準形よりも安定で、ロバスト性が高く、多ループ制御系の設計に適している。このような理由で鉄鋼業の圧延ミル制御に広く用いられた。しかし Kessler 標準形は、すべての安定度指標  $\gamma_i$  を 2 とすることと等価なので、8 % のオーバーシュートがある。その後オーバーシュート無しの条件は  $\gamma_1$  を少し大きく、 $\gamma_1 = 2.5$ 、とすることで簡単に得られることが分かったので、係数図法ではこの修正を行っている。

ヨーロッパでは多数の研究者が同じような考え方を発展させてきた (Brandenburg, 1996)(Zaeh, 1987)(Naslin, 1968)。安定度指標には色々な名前が付けられ、Kessler は *damping factor*、Brandenburg は *double ratio*、Naslin は *characteristic ratio* と呼んだ。北森は Graham の方法の改良版として、特性多項式の係数は、低次のみ規定するという方法を提案したが、これにより設計の融通性が大きく改善された。

制御系の安定性は、特性多項式の係数を用いる Routh-Hurwitz の安定条件によって調べることができる。しかしこの方法では係数の変化が安定性に及ぼす影響が明確ではない。Lipatov と Sokolov は安定性と不安定性についての十分条件を提案した (Lipatov, 1978)。これは形が大変簡単で安定性・不安定性と係数との関係が非常に明確になるので、これらの条件は係数図法の設計手順の中に組み込まれた。

その後これらの条件は、係数図上で図式に易しく解釈できることが明らかになり、設計の強力な道具となった。係数図が始めて使われた頃から、係数図の曲率が安定度指標  $\gamma_i$ 、したがって安定性に深く関係していることは分かっていた。しかし Lipatov の安定条件を直接図式に解釈できるようになり、安定性に関するより正確な情報が容易に得られるようになった結果、より複雑な設計が可能になったのである。係数図の傾きは等価時定数  $\tau$ 、したがって応答速度に深く関係している。また経験を積み重ねた結果、時間応答・周波数応答を推定する図式的手法も開発されてきた。係数図を作成する過程で設計者は、特性多項式を構成する要素多項式 (*component polynomials*) を求めなければならない。そしてこのような要素多項式の係数と特性多項式の係数の大きさの比がロバスト性の鍵であることが明らかになった。係数図上で要素多項式が特性多項式の下にあれば、特性多項式の係数のパーセント変化は、関係するパラメータのパーセント変化に比べて小さく、この系は大変ロバストである。逆に要素多項式が特性多項式より上にあると、パラメータの僅かなパーセント変化でも、特性多項式の係数のパーセント変化は非常に大

きく、この系はロバスト性が大変低い。このように係数図は、制御系の重要3要素すなわち安定性・応答性・ロバスト性について十分な情報を備えている。経験を積むと設計者は係数図を一瞥するだけで制御系の全体像を頭に浮かべることができる。係数図は係数図法の中で、最も重要な構成要素であり、古典制御でボード線図が果たしている役割と同じ役割を果たしている。

制御系の設計には古典制御理論と現代制御理論が用いられている。古典制御理論ではシステム表現には伝達関数が用いられ、設計は通例ボード線図や根軌跡を用いて進められる。現代制御理論では状態空間表現が用いられ、設計は Riccati 方程式を解くことにより進められる。しかしこの他に第3の方法として、代数的設計法あるいは多項式設計法と呼ばれるものがある。この設計法ではシステム表現には多項式が用いられ、設計は Diophantine 方程式を解くことによって行われる。この方法の第1の長所としては、多項式表現が状態空間表現と同じ正確さを持ちながら、より分かり易いことが挙げられ、第2の長所としては Riccati 方程式が2次形であるのに対し、Diophantine 方程式は線形で解法が簡単であることが挙げられる。制御について過去多くの研究がなされてきたが、これらの研究は特性多項式が定まった後での問題を扱っていて、与えられた問題に最も適した特性多項式を選定する問題は、殆ど研究されてこなかった。特性多項式は極配置法などを用いて、どちらかと言えば試行錯誤的に求められるのが現状である。係数図法では問題に最も適した特性多項式を見付けることが最大の関心事である。特性多項式が求められた後に制御器を求めるのは、代数的設計法では非常に簡単で、係数図法でもこのような手順を採用している。

新しい3特徴が係数図法の発展過程で付け加えられた。係数図法に採用された色々な考え方や経験は開発小史の中で説明したが、係数図法の特徴を明らかにするため、係数図法の発展過程で付け加えられた新しい3特徴について以下に説明する。

第1に付け加えられたものは係数図の導入である。これにより制御系の重要3要素の安定性・応答性・ロバスト性が一つの図の上で図式に表現され、システム全体の理解が容易になった。第2に付け加えられたものは Kessler 標準形の改修で、その結果 Kessler 標準形の8%オーバーシュートがオーバーシュート無しになった。第3に付け加えられたものは、Lipatov の安定十分条件で、これは係数図法に適合するように変形されている。

このような特徴が加えられた背景を明らかにするため、著者(真鍋舜治)は読者にお許しを願って個人的な経験について説明させて頂きたいと思う。著者が自動制御と初めての出会ったのは、1952-1954年に米国のオハイオ州立大学に留学していたときであった。Warren 教授の講義で始めて Routh 安定判別法を学びその美しさに感動した。また一人の空軍大尉の方が修士論文を書いているのに出会ったが、その方の研究は3次系のステップ応答波形を辛抱強く計算し、図に描くことであった。計算尺が唯一の計算手段であった時代にはこれは大変な仕事であった。後にこの同じ図が Graham の論文に載

せられていることに気付き大変驚いた次第である。

日本に帰国して後、著者は三菱電機研究所でモータ制御の分野の仕事についた。最初は米国で習った Nyquist 安定条件を試みたが、計算の負荷が重くて不便であった。その後 Chestnut の教科書に出会ったが、ここでは設計にはボデー線図が用いられ、また折れ点同士が互いに分離していることの重要性が強調されている。その頃先輩技術者の方より安定性を保つ鍵は制御器・制御対象の中の時定数を 2 倍以上離すことであると教えられた。この経験則は Chestnut の考え方に一致し、またアナログ計算機シミュレーションの結果にも一致していた。

1950 年代の終わり頃、著者は 45,000 HP の遷音速風洞プロジェクトに参画した。ここで制御系が正しく作動することを検証するため 1/100 のスケールモデルの試験が行われた。その制御対象の伝達関数を求める過程で 20 Hz の振動モードが観測されたが、モデルが小さいので、周波数が高く制御系との有害な干渉は起こらなかった。しかし実系では振動周波数は低くなるので、制御ループとの干渉が予測され、ボデー線図とアナログ計算機シミュレーションを用いて、詳しい検討が行われた。しかしこのボデー線図の形は通例の制御系のものとは大変異なった形のものであった。この制御対象の伝達関数  $G_p(s)$  は次のように表される。

$$G_p(s) = \frac{(s/\omega_1)^2 + 1}{s[(s/\omega_2)^2 + 1]}, \quad (1.119)$$

ここに  $\omega_1 \cdot \omega_2$  はそれぞれ反共振・共振周波数である。この系は二慣性共振系と呼ばれている。

この頃三菱電機の吉田太郎より、私的情報交換の中で「良い制御系を作るにはフィードバック信号を二つに分け飽和の前後にそれぞれを与えると良い」と教えられた。著者はちょうどそのとき Tustin の大形対象物に関する位置制御の論文を読んでいた (Tustin, 1958)。このようなモーションコントロールではアクチュエータの飽和が非常に優勢になり、飽和が起こるとゲインが非常に下がる。このため位相余裕は交差周波数の付近に留まらず、さらに低い周波数領域でも適切な値に保たれなければならない。そして Tustin は開ループ伝達関数として次のようなものを提案している。

$$G(s) = A_0(\omega_c/s)^k, \quad 1 < k < 2, \quad (1.120)$$

ここで飽和がないときは  $A_0 = 1$  で、飽和が起こると  $A_0$  は非常に小さくなる。このような系はその後 1990 年代に分数次系 (*Fractional Order System*) と名付けられた。著者は吉田の提案が分数次系の実用的な実現方法であることに気付き、その後研究を進めて、 $k$  の最適値は 1.4 近辺、即ち折れ線近似ボデー線図で -28 dB/decade の傾斜になっていることが分かった (Manabe, 1960a)。これは大変簡単で実用的な設計基準で、レーダアンテナ制御や他のモーションコントロールの設計に利用された。この頃は開ループ伝達関数の折れ線近

似ポーデ線図による設計が主要な設計手法であって、制御対象は通例では実極と実零で構成されていた。

1970年代後半著者は人工衛星の姿勢制御系の開発に従事した。ここには二つの振動問題がある。一つは太陽電池パドルのような可撓性付属物の衛星姿勢への干渉である。このとき制御対象の伝達関数は二慣性共振系の式(1.119)と同じような形になるので、その制御問題は二慣性共振系の制御と同じような方法で解決することができた。今一つの問題はモーメントムホイール (momentum wheel) をもつ人工衛星のニューテーション減衰制御 (nutation attenuation control) である。制御対象の伝達関数  $G_p(s)$  は次の形になる。

$$G_p(s) = \frac{1}{s^2 + 1}, \quad (1.121)$$

ここでニューテーション周波数は 1 rad/sec に正規化されている。入力は第2の小形ホイールのモーメントムで、出力は人工衛星のロール角である。一般的に言えば振動制御の最も良い手法は、ばね力を弱くし減衰力を強めることである。ばね力を弱くするという制御は、正帰還を加えることと等価であり、制御器は非最小位相形となる。当時設計された制御器は非最小位相形制御器であった (Terasaki, 1967)。設計手法は極配置法であったので、ロバスト性はまた別箇に検討されねばならなかった。

著者が設計した制御器は次のようなものである。

$$G_c(s) = \frac{k_2 s^2 + k_1 s + k_0}{l_2 s^2 + s}. \quad (1.122)$$

この制御器はPID制御器のように見えるが、 $k_1$  が零か弱い負値をとること、また  $l_2$  がかなり大きな値をとるという点で、PIDとは異なっている。この形の制御器を零PID制御器 (Zero-PID Controller) と名付けた (Manabe, 1981)。

1980年代を通して、著者はこのような3種の制御問題、すなわち式(1.119)の二慣性共振系、式(1.120)の分数次制御系、式(1.121)のニューテーション制御系が統一的に設計できる方法を模索した。このとき古典制御を開ループ設計から閉ループ設計に変更することも考えたが、成果は得られなかった (Manabe, 1983)。1985年7月著者はケンブリッジ大学を訪問し、MacFahlen教授のご尽力で J. C. Maxwell の論文を調べることができた。そのとき著者は Maxwell の方法を更に発展させることが制御系設計問題の解決法になるのではないかと考えた。著者は現代制御を余り信用していなかったが、その理由はPID制御器と等価な制御器を設計するために、適切な重み関数を選択することが困難であったからである。

1988年5月のある夕暮れ、事務所の建物に向かって歩いているとき、突然係数図のイメージが頭に浮かんだ。そして上記の3種の制御対象について、統一的な手法で制御器が設計できることに突然気が付いた。著者のノートブックに最初に描かれた係数図は1988年5月13日の日付が記されている。以上が係数図に到る長い旅路である。

著者は1990年から制御理論を東海大学で教え始めた。そしてKessler 標準形を設計の基礎としていた。1990年夏一人の修士課程の学生が安定度指標  $\gamma_1$  を大きくすると、オーバーシュート無しの条件が得られることを見つけた。その学生は制御系設計の宿題を解く過程で、この重要な結果を発見したのである。その後詳しく調べて見ると、オーバーシュート無しの条件は3次系では  $\gamma_1$  は2.7、4次系では2.5であることが分かった。安定度指標  $\gamma_1$  の標準値は簡単化と使い易さのため2.5と選ばれているが、逆数が  $1/2.5 = 0.4$  となるなど便利であるからである。係数図法が最初に或るシムposiumで紹介されたときは、 $\gamma_1 = 2.7$  であり (Manabe, 1990)、その後雑誌に掲載されたときは  $\gamma_1 = 2.5$  とした (Manabe, 1991)。その後安定度指標の選定法について色々改良が加えられ、現在はロバスト性を十分考慮し、より融通性の富む標準が用いられている。

余談であるが、著者はKesslerの研究については1970年代の終わり頃から伝え聞いていたが、実際にKesslerの論文を入手したのは、1992年9月18日、DüsseldorfのVDI (Association of German Engineers) 事務所でのことであった。

1989年頃著者はたまたまBoseの論文 (Bose, 1988) を読み、安定・不安定十分条件についてのLipatovとSokolov論文 (Lipatov, 1979) を知った。著者はこれが係数図法の数学的基礎となる重要な定理であることに、直ぐ気が付いた。また安定条件を図式に解釈できるように、限界値を導入した。安定性は安定度指標  $\gamma_i$  を限界値  $\gamma_i^*$  と比較することにより求められる。しかしこの関係は  $\gamma_i$  と  $\gamma_i^*$  の計算が必要なので必ずしも便利な関係ではなかった。それから数年後この条件は、図式解釈による係数間の関係に置き換えられることが分かった。この解釈により、完全に係数図を基礎とする設計が可能になったのである。

係数図法の現状と将来展望について以下に簡単に述べる。係数図法は未熟な幼少期から健全な設計理論へと成長して行くにつれ、好い設計結果が報告されるようになった。著者は係数図法の有効性を示すため、何か特殊で面白いものを作りたいと思った。そこでセンサ1個 (角度検出器) のみ、長さ20 cmの倒立振子を作り、おもちゃの模型自動車の上で立たせた。制御器は2次の不安定制御器で、4個のパワートランジスタと、3個のICに組み込まれた12個の演算増幅器で構成されている。この模型自動車のある国際会議で実演し、このような特殊な制御器を設計する上で、係数図法が有効であること報告した (Manabe, 1994a)。他の設計例や、過去の良い設計の係数図法による解釈についても報告されている (Manabe, 1994b)(Manabe, 1997b)(Manabe, 1998a)。田中は独自で同じような設計法提案したが、ここでは安定度指標の逆数である  $\alpha$  パラメータを用いている (Tanaka, 1992b)。これは実際にガスタービンの制御に適用された (Tanaka, 1992a)。堀は二慣性共振系の制御器設計に安定度指標を用いた (Hori, 1994)。係数図法とLQG設計の関係が明らかにされ、係数図法と等価なものとして、拡大LQRが導き出された。また

Lipatov の論文はさらに詳しく検討され (Manabe, 1999)、それにより係数図法の数学的基礎はより明確になった。これらの結果はまとめられ、係数図法的设计手法は一貫性のある形で IFAC (国際自動制御連盟) のシンポジウムでは発表された (Manabe, 1998b)。

係数図法に関する最初のオーガナイズドセッションは Y. C. Kim によって 1988 年 10 月韓国自動制御会議のために計画された。同様のオーガナイズドセッションは第 3・第 4 アジア制御会議 (Asian Control Conference) 2000 年・2002 年にも行われ (Manabe and Kim, 2000)(Manabe, 2002)、多数の研究者が係数図法の経験を紹介した。MIMO 系への応用も進められている。この過程で係数図法と多項式设计法は共通点が多いことが分かってきた。二つの方法の情報交換は両方にとって有益であろう。係数図法は過去二・三十年にわたる多項式设计の豊かな研究結果の恩恵を受けることになる。

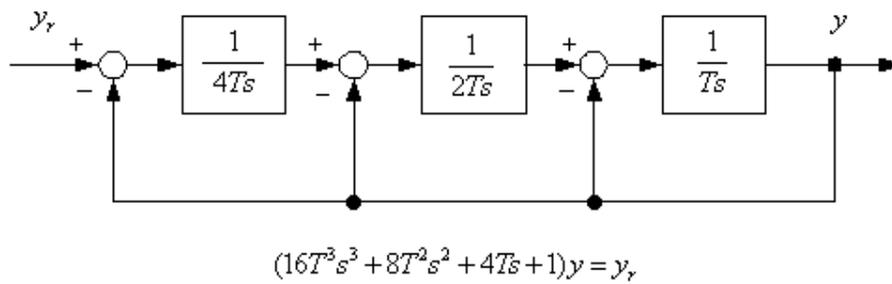


Figure 1.19: Kessler's canonical multi-loop structure

## 1.7 まとめ

本章で述べた重要な点を次にまとめる。基本的な考え方では、係数図法での3個の基本的な考え方について述べた。第1に「制御は妥協である」という考え方を述べた。第2に「フィードバック制御は、広い意味の制御の小さな部分でしかない」ので、これを極端に精緻にすることは、無意味と考えられることを述べた。最後に代数的設計法が、古典制御・現代制御に続く第3の設計法であることを述べた。係数図法は代数的設計法の1種であるが、古典制御が「外向き設計法」、現代制御が「内向き設計法」であるのと異なり、「同時設計法」であることを示した。

簡単な設計例では、典型的な位置制御系を例として、係数図法による制御系設計法を示した。安定度指標・等価時定数の定義が与えられ、これを用いて制御器がシステムティックに設計されていく過程が示されている。係数図が導入され、それとステップ応答が密接に関係していることが示されている。最後に係数図を媒体として、制御器と閉ループ特性(ステップ応答)を同時に設計する、係数成形法について述べている。

システム表現では、係数図法で用いている多項式表現が、色々な意味で伝達関数表現や状態空間表現より勝っていることを示している。多項式表現は微分方程式表現そのものであるから、非線形系や分数次系への拡張が容易である。伝達関数表現のように分かり易いが、曖昧さが無い。状態空間表現のように正確であるが、状態空間特有の使いづらさを避けることができる。

設計の進め方では、設計の進め方の概略を述べている。先ず設計の基礎としている数学的モデルを示している。次に係数図法設計の基礎となっている、4個の入出力関係について述べている。続いて極零キャンセル問題を検討し、係数図法では、不適切な極零キャンセルが、前もって予防されていることを示している。最後に具体的な設計ステップを示しているが、最も重要な制御器と特性多項式の同時設計の問題は、一般論が難しいので、後章に譲り例題による説明に委ねている。

制御構造では、色々な設計法で得られた制御器も、係数図法の標準制御器に等価変換できることを示している。この等価制御器の次数とパラメータの範囲を制御構造と呼んでいる。先ず係数図法制御器の制御構造を述べ、パラメータの範囲が非常に広く採られていることを示している。次に古典制御の制御構造を述べ、パラメータの範囲がかなり制限されていることを示している。続いてLQG設計の制御構造を述べ、低次の制御器が作れないこと、ロバスト性の保障がないことを示している。さらに拡大形LQR (Linear Quadratic Regulator) の制御構造について述べ、これが係数図法制御器と正確に対応していることを示している。最後にQパラメタリゼーション付きのLQRについて述べ、これと等価な係数図法の標準制御器を導出している。

歴史的背景では、係数図法が形成されていく過程を概観した。先ず係数図法を生み出す背景となった制御理論について、簡単な歴史を述べている。次に係数図法で新たに付け加えられた3個の特徴、すなわち「係数図」「ケス

ラー標準形の修正」「リパトフの安定十分条件の図式解釈」についての、著者の経験について述べている。最後に係数図法の現在と将来の展望について簡単に述べている。

最後に、「係数図法とは何か」という読者の問いに答えたいと思う。係数図法は多項式環の上での代数的設計法で、ボード線図の代わりに係数図が、ラウスの安定条件の代わりにリパトフの安定十分条件用いられている。係数図法は次の6個の特徴をもっている。

1. 係数図法は多項式環の上での代数的設計法である。これは古典制御・現代制御につづく第3の制御系設計法である。
2. 多項式または多項式マトリックスがシステム表現に用いられている。多項式表現は、古典制御での伝達関数表現と同じように分かり易く、また現代制御での状態空間表現と同じ正確さをもっている。
3. 制御器と特性多項式が同時に設計される。これによりロバスト性が安定性・応答性と同じように設計に組み込まれる。古典制御では、制御器が最初に仮定され、その後閉ループ特性を確認する、外向け設計法が用いられている。また現代制御では、閉ループ特性が最初に与えられ、これに見合った制御器が求められる、内向き設計法が用いられている。これらに反し、係数図法では制御器と閉ループ特性を同時に設計する同時設計法を用いている。
4. 係数図法では係数図が有効に利用されている。係数図では、安定性・応答性・ロバスト性についての十分な情報を、図の形で表現しているので、設計を図式に進めることができる。係数図は古典制御でのボード線図と同じ役割を果たしている。
5. リパトフの安定十分条件を理論的根拠としている。この条件は係数図上で図式に表現される。係数図法では、これをラウスの安定条件の代わりに用いている。
6. ケスラー標準形が修正され係数図法標準形として用いられている。主な修正点は、 $\gamma_1 = 2.5$  とすることであるが、ロバスト性が問題になるときには、このためにさらに修正が必要になる。