非線形フィードバックによる

飛行制御系の再構成に関する研究

越 智 徳 昌

①

非線形フィードバックによる
飛行制御系の再構成に関する研究

１９９３年６月

越 智 徳 昌

目　次

１．序　論

１．１　はじめに

墜落に至るような航空機事故では，生死の境は故障発生後のパイロットの操作にあることが多い．故障直後，乗員がまがりなりにも機体の制御を取り戻すことができれば，生還の可能性は極めて高い．これに反して航空機が一度でも操縦不能に陥ってしまうと，その後，機体の制御を回復できる可能性は極めて低い．本論文ではこのような不測の事態に対し，航空機を自動的に回復させる方法の一案を示したものである（文献1-1～1-21）．このような故障の自動補償機能をもつ飛行制御系は，一般に「再構成可能な飛行制御系(Restructurable Flight Control System;以下ではRFCSと略記する)」と呼ばれる．ＲＦＣＳでは基本的に，故障が発生すると直ちにそれを発見し，オンラインで飛行制御系を修正することにより再構成が行われる．

本章では，まず，このような耐故障型の飛行制御システムが注目され，研究されるようになった背景を述べる．そこでは，航空機事故の防止において，耐故障型飛行制御システムが果たす役割を，事故統計やさまざまな事故例を通して明らかにする．次に，本研究で取り上げる再構成可能な飛行制御系の基本的な設計概念を示す．そして，これまで行われてきた主な研究例を紹介すると同時に，本研究の意義について述べる．最後に，本論文の概要を示す．

１．２　背景

民間航空は1958年にパンアメリカン航空がBoeing 707をニューヨークーパリ間に就航させて以来，ジェット旅客機の時代を迎えた．６０年代前半はコメット，B-707，DC-8，CV-880 などの第１世代のジェット機がつぎつぎと運航を開始する．その後，６０年代半ばには第２世代，７０年代には第３世代，そして，８０年代には第４世代の航空機がエアラインに投入されてきた．新型機が現れる度に初期故障や操縦に乗員が習熟してないことなどから事故率が増えるものの，1975年頃までは全体的に減少している．しかし，７５年以降は事故率が減少する様子は見られない．他方，航空機の事故原因としては，パイロット，航空機，空港／管制，天候，整備などが挙げられる．これらは人間的要因と機械的要因に分けられるが，ここでは人間的要因としてパイロット，機械的要因として航空機に注目する．本節では，これらの要因に対して従来取られてきた対策やその問題点を示す．そして，それらの問題を解決する上でＲＦＣＳがどのように有効であるのかを示す．

1.2.1　ＲＦＣＳによるパイロットミスの排除

まず人的要因による事故に対しＲＦＣＳがどのうように役立つか考える．事故原因の中で占める割合が圧倒的に大きいのはパイロットであり，全体の６割から７割に達する．従って，パイロットのミスを減らすことができれば事故率をかなり下げること

が期待できる．実際，パイロットのミスを減らす努力は行われてきており，この１０年間(1981-1990) の統計ではパイロットの要因の割合は減ってきている． 文献1-22は，その理由として，

①コクピット・プロシジャーの標準化促進

②計器着陸方式(Instrument Landing System;ILS)の普及

③対地接近警報(Ground Proximity Warning System;GPWS)の装着

④シミュレータによる訓練の進歩

が挙げられている．このうち②と③は操縦システムの自動化の一例であり，①についても離陸前のさまざまな手続きは飛行管理システムにより自動化されつつある（文献1-23,1-24,1-25,1-26,1-27）．このように自動化はパイロットのワークロードの低減とともにパイロットのミスを防ぐ上でも有効であると考えられる．勿論，④のようにパイロットの訓練・教育を含めた運航管理体制の整備も大きく貢献しているはずである．

それでは自動化を進めて行けば，事故率は減少し続けるだろうか．答は必ずしも肯定的ではない．自動化の促進には２つの問題点がある．一つは自動化と人間の関わりである．自動化が進めば，パイロットは自動システムを過度に信頼するようになり，飛行状態の監視を怠るなど，警戒心のゆるみが生じる．例えば，いくつかのインシデント（注1-1）や事故の例では， オートパイロット／オートスロットルの使用法を誤ったり，自動操縦装置に操縦を任せきりにしたりしていたことが明らかになっている．そして，それらのケースでは飛行計器を適切にモニターしていなかったことがインシデントや事故につながったと考えられている．例えば，表１－１に示された事故例１，２，３がこれに相当する．このような純粋に人為的な事故はＲＦＣＳでは防ぐことができない．もう一つは，自動システムが異常事態に対して柔軟性や適応性がないこと，つまり融通がきかないことである．例えば，機材の故障や火災などの緊急事態，気象の変化，ＡＴＣ(Air Traffic Control) クリアランスの変更など飛行が計画通りに行かなかった場合に適切に対処することは現在の自動システムでは一般に難しい．自動化のこの欠点は「脆さ(brittleness)」と呼ばれている．

第１の問題の対策としては，自動システムに対するパイロットの訓練・教育の充実や警戒を怠らせないようなシステムの設計が求められる．一方，第２の問題点は自動化の最も難しい点で，通常人間オペレータに任せられる部分である．そこでは人間の柔軟性や適応性に期待がかけられる．しかし，自動化されたシステムは人間にとっても分かりにくいものとなっているので対処することは容易ではない．しかも，自動化によりパイロットの技倆は下がっていると言われている（文献1-28, p.358）．また，極度の緊張下では人間の能力は十分に発揮できなことが多い上に，個人差，体調，精

注1-1： インシデント(incident)とは，機体の故障などの異常が発生したとき，それが結果的に機体の大きな破損や搭乗者の死亡などの重大な事態に至らなかったケースを指す．これに対し，重大な事態に至ったものを事故またはアクシデント(accident)と言う．例えば，空中異常接近（ニアミス）はインシデントであるが空中衝突は多くの場合アクシデントになる．

神状態などにより対処能力にはばらつきがある．それ故，異常事態にも何かの自動システムによって対処されることが望まれる．しかし，これを行おうとする場合，上で挙げたようなさまざまな異常事態が考えられるのでそれらに一般的に対処することは難しい．ここでは，ＲＦＣＳ設計の立場から飛行制御に関係するものに限定する．即ち，機体や制御器（舵面，アクチュエータ，エンジンなど）に故障が生じ操縦性や固有の動特性が変化する場合，あるいはセンサーや制御用機上計算機などに故障が発生した場合を考える．具体的には，次のようなインシデントや事故が挙げられる．

・着氷，氷結による揚力の減少や舵面の固着(事故例4,5,6)

・エンジン故障(事故例7,8)

・エンジン故障に舵の異常が重なった場合(事故例9,10)

・機体や操縦系統の破壊(事故例11,12,13,14,15,16,17,18)

上に挙げたのはいずれも機体や制御器に異常が生じた場合である．これらの故障やインシデントの中には，故障があまりにも重大で物理的に回復不能であるケース（事故例14,15,17）も含まれている．そのような場合には，どんな方法を用いても回復は望めないであろう．一方，回復は可能であったにもかかわらず，パイロットが適切な回復操作をとることができなかったために事故に至ったケース（事故例8）や逆に， 非常に厳しい状況ではあったが，パイロットの巧みな操縦により無事帰還できたケース（事故例7,11,12,13,16,18）もある．

もし，このような異常事態において自動的に飛行の安全（最低，釣合飛行）を回復することができれば，パイロットに依存しないで多くの事故を防げる可能性がある．そして，これを行うための飛行制御系がＲＦＣＳである．

上述のように，ＲＦＣＳは従来の自動システムの欠点である「脆さ」を克服する一つの方法であると同時に，それは人間の弱点である能力のばらつきや不確定性も排除する．事故例からも分かるように故障からの自動的な回復が求められるのは，自動化の進んだ新鋭機ばかりではない．ＲＦＣＳが求められる理由は文献1-29 の次のコメントが端的に示している．

「操縦系統に故障が生じた航空機を飛ばすために最も重要なことは，故障が検知された後の初期の段階で，緊急制御(emergency control) を行うことである．パイロットとのインタビューによれば，制御故障からの回復の最も重要な点は，故障に対してすぐに，そして適切に反応することである．これは特に高速で，あるいは地面近くで飛行する場合に重要となる．調べた限りでは，回復可能な事故の全てのケースで直ちに緊急制御を行うことの必要性が明かとなった．回復操作のために利用できる時間は，一般に，４～５秒であった．しかし，いくつかのケースでは２５秒まで利用できた．航空機が特に高性能機ではなくても，パイロットは比較的なじみのあるタイプの故障さえ，同定できなかったようである．そして，彼らは制御に関わる故障の際には，完全自動の回復操作(full-authority automatic response)が行われることを望んだ．」

1.2.2 ＲＦＣＳによる信頼性の向上

次に機械的要因による事故を除く上でＲＦＣＳがどのように有効であるかを見る．航空機事故では，人的要因の他に航空機の機材や搭載機器に生じた故障や異常が大き

な原因となる．故障を防ぐには，まずそれらの信頼性を高めることが考えられる．しかし，信頼性をあまりに追求するとコストが高くなる上，絶対に故障しないという保証もない．従って，航空機あるいはその飛行制御系を，ある程度故障を許容するようなシステムにすることが必要になる．つまり，耐故障性のあるシステムにする必要がある．耐故障型システムは構成要素の一部が壊れても，システムとしての機能が保たれるので，壊れにくい，即ち信頼性が高いシステムと言える．

航空機において，耐故障性に対する最も基本的な要求はシステムの一部が故障しても，全体が破局的な状態に陥らないようにすることである．そして，次に望まれることは，故障による性能劣化をできるだけ小さくし任務達成を可能にすることである．前者は一般にフェイルセーフと呼ばれている．フェイルセーフを有する設計法としては，次の３つが考えられる．

①ロバスト設計：故障発生時に特に対策を取らなくても故障を最小限にとどめ，それがシステム全体の性能に著しい影響を与えないように設計する．

②冗長設計：バックアップのように，故障時のために特に用意された機構を持つ．要素冗長性(component redundancy)ともいわれ，通常は複数の同じあるいは同じ機能を持つ機器を備える．

③制御系再設計：残存しているシステムの運用法を変更することにより故障を補償する．

①には，舵面分割やスマート・アクチュエータ（注 1-2），あるいは機体の構造設計における１ペイ・フェイルセーフのようなハードウェア的なものと，ロバスト制御系のようなソフトウェア的なものとがある．この方法は，特に付加的な機構や要素を必要としないので，コストや重量などに与える負担は小さい．しかし，どちらかと言うと消極的な方法であり，故障によって低下した機能が少しでも回復されることはない．また，この方法が有効な故障の範囲は一般に限られており，これだけでは故障対策として十分とは言えない．

②は主に機材の故障に対処する方法である．簡単で実現しやすいが，通常，冗長性をもたせることができる機材は種類，数ともに物理的に限られている．また，バックアップの機材は正常時には全く使用されず，無駄になる．さらに，過剰な冗長性は，コストを上げ保守性を悪くする．そして，重量増にもつながる．

③は純粋にソフトウェア的な方法である．これは，残っている正常な機材のみを有効に用いて，できるだけノミナルの制御性能が回復されるように飛行制御系を自動的に設計し直す．それゆえ，再構成可能な飛行制御系と呼ばれる．この方法は①の方法に比べて積極的な方法であり，有効な故障の範囲が広い．また，コストの増加，保守性の悪化，重量増などの問題もない．ただし，コンピュータへの負担は増加する．

注1-2： 直接駆動型またはエレクトロメカニカル型のアクチュエータで，油圧パワーを必要としない．従って，操舵が油圧系統の故障の影響を受けない．また，各アクチュエータが制御用コンピュータをもち，動作指令は中央のコンピュータから電気信号で送られる．

1.2.3 ＲＦＣＳに期待される効果

ＲＦＣＳも含め，上の２節で述べた航空機事故防止の対策が事故率の減少に対してどのような効果をもつかを図１－１に図式的に示す．図の縦軸は事故の発生率，横軸は事故原因である．横軸の左端は純粋に機械的な（ソフトウェアも含む）故障や異常を指す．右端は純粋に人間によるミスを指し，航空機には全く異常がないことを意味する．中間は機械的な故障と人間のミスが重なった場合である．左に近づくほど人間による対処が難しく，右に行くほど対処は容易であるにもかかわらず人間のミスで事故になるケースであると考えてよい．図の点線は自動システムを備えていない場合，二点鎖線は機械的構成要素の信頼性，整備性，ロバスト性，品質管理を改善したり冗長性を付加したりした場合，一点鎖線は自動システムを備えた場合，そして実線はＲＦＣＳを備えた場合を指す．この図は次のことを意味する．まず，自動システムをもたないやや旧式の航空機を考える．その場合の事故率は図の点線に示され，それは人的要因が大きいことを示している．次に，機械的構成要素の信頼性等を向上すると機械的要因による事故は減少する（二点鎖線）．さらに自動システムを導入するとパイロットのミスによる事故は減る．しかし，自動システムは機械的故障に弱いので，そのような故障に対してはあまり役に立たないか，もしくは事故の危険性を大きくすることになる（一点鎖線）．また，パイロットによる自動システムの誤った使用や自動システムへの過信などから生じる新しいタイプの事故が発生する．このため自動化しても人的要因による事故も急激に減少するわけではない．これに対し，ＲＦＣＳを備えると自動システムの機械的故障に対する欠点を除くことができる（実線）．それと同時に故障が発生した場合におけるパイロットのミスも防ぐことができる．ただし，故障とは無関係な純粋に人間的要因による事故はＲＦＣＳでも防ぐことはできない．

このようにＲＦＣＳは機械的要因による事故から人的要因が関係するものまで多くの事故を防ぐことができると考えられる．

1.2.4 パイロット支援システム（文献1-29,1-30）

ＲＦＣＳは故障を補償する有望な方法である．しかし，それだけで完全に異常事態に対処できるわけではない．ＲＦＣＳの緊急制御によって危険な状態から回復した後は，パイロットが機体の状況や操縦性能を判断し，飛行計画の変更等を行わなければならない．また，ＲＦＣＳが扱うのは飛行制御に関する異常事態だけである．先に述べたように，異常事態としてはいろいろな状況がありうる．それら全てに対処できるような自動システムをつくることは非常に困難であろう．それ故，パイロットに頼らざるを得ない場合も当然出てくる．このような場合には，人間にその能力（柔軟性，適応性）を最大限に発揮させ，計画，問題の解決，意志決定などを適切に行えるように情報や助言を提供するシステムが望まれる．これはパイロット支援システムと呼ばれ，現在の新型機においても，操縦システムの自動化及びグラスコクピットの普及とともに備えられるようになってきている．

1.2.5 まとめ

航空機事故の原因の７０％以上がパイロットのミスであるとされている．パイロッ

トによるミスを防ぐために，主に次の２つの対策がとられてきた．一つは乗員の教育・訓練を充実することであり，もう一つは，操縦の自動化を進めることである．しかし，教育・訓練の効果には限界があり，たとえそれが十分に行われたとしても人間は体調や精神状態等により常に一定の性能を発揮できるとは限らない．一方，自動化されたシステムは故障や飛行計画の変更等の異常が発生した場合に，柔軟かつ適切に対処することが難しいという問題点をもつ．それ故，これまでは異常時の対処は依然としてパイロットに頼っていた．即ち，人間の柔軟性・適応性に依存していた．そして，その能力が十分に発揮できるように，パイロット支援システムが備えられてきた．しかし，ここでも先に挙げた常に一定の性能を発揮できないという人間の欠点が残ることに変わりはない．むしろ，異常時には極度の緊張のために対処能力が低下する上，自動化により複雑になった操縦システムはパイロットが状況を把握するのを妨げる可能性がある．以上のような自動化の問題点や人間が本来もっている欠点に対する一つの対策が緊急制御（emergency control）の自動化である． 即ち，迅速かつ的確な対処が要求される故障発生直後の回復操作を全て自動制御に任せることである．実際，パイロットらは緊急制御が完全自動制御（full-authority post failure automatic control）によってなされることを望んでいると言われている（文献1-29）． 本研究はこの緊急制御の自動化を飛行制御系の再構成により達成しようとするものである．一度，危険な状態から脱しトリム飛行が回復できれば，後はパイロット支援システムの助けを得ながら，パイロット自身が冷静に対処できるであろう．

航空機事故のもう一つの大きな要因は機械的要因である．これによる事故を防ぐには機器の信頼性や品質管理，整備性の向上，あるいはフェイルセーフ構造のようなロバスト性の付加，バックアップを備える冗長性の付加などの対策がとられてきた．しかし，このような方法には経済的，技術的限界があり，それだけに頼ることはできない．これに対する別の対策として，正常な機器の運用法を変更することにより故障した機器の機能を補償することが考えられる．ＲＦＣＳはこの方法を積極的に利用するものであり，機械的要因による事故に対してもシステムとしての信頼性を高めるものである．

本節で述べたことがらをまとめると，ＲＦＣＳが求められるようになった背景と航空機事故対策におけるＲＦＣＳの役割は図１－２のようになる．

アメリカでは1978年の規制緩和後，1980年代の中ごろから事故やインシデントが目立ち始めた．この点について文献1-31は次のように説明している．

規制緩和による増便
↓
パイロットの不足
↓
パイロットの経験度，即ち質の低下
↓
セーフティマージンの低下
↓
事故

我国でも航空輸送量の増加に加え，これから１９９６年にかけて大量のパイロットがリタイアすると言われており，アメリカと似たような状況になりつつある．セーフティマージンをパイロット自身に求めることはますます難しくなってきているようである．

航空機事故は操縦者あるいは操縦システムだけで防げるものではない．多くの事故例が示しているように，航空機の安全性は整備員などの地上作業員，メーカ，運航会社，監督当局など航空機の設計，製造から運航にわたる多くの人や組織によって確保されなければならない．しかしながら，航空機の安全を守るそのような努力の中で，本研究で扱う再構成可能な飛行制御系は重要な位置を占めるものである．

１．３　再構成可能な飛行制御系（ＲＦＣＳ）

前節で述べたことを背景として，1982年に航空機の事故対策を考えるワークショップがNASAにより開かれた．そこで大きく取り上げられたのが再構成可能な飛行制御系と呼ばれる耐故障型飛行制御システムである．これは，基本的には，何らかの方法で航空機に発生した故障を突き止め，その結果を用いて飛行制御系を修正するものである．このワークショップを契機としてＲＦＣＳが注目され研究がさかんに行われるようになった．そこで行われた文献1-32,1-33,1-34の講演では，ＲＦＣＳを設計する上で解決しなければならない問題が具体的に示され，そのためにはどのような制御理論，あるいは同定・推定理論が適用可能か，また，どのようなハードウェアが利用できるかなどについて幅広い議論がなされている．こうして，ＲＦＣＳに関する研究の方向が提示された．３年後の1985年に出されたEternoらのレポート（文献1-35）では，故障の検出・同定から制御系の再設計までＲＦＣＳの設計に関してより具体的かつ詳細な検討が行われている．

以下では，まずこのレポートなどをもとにＲＦＣＳの定義，典型的な構成法などを示し，本論文で扱うＲＦＣＳの輪郭をもう少し鮮明にする．次に，ＲＦＣＳの設計に適用できる制御及び同定・推定手法について検討し，本研究で採用する手法の特徴を他の設計例と比較することにより明らかにする．

1.3.1 ＲＦＣＳの定義

ＲＦＣＳを次のように定義する．

「機体の一部やその構成要素が故障したり損傷を受けた場合に，直ちにその検出・同定を行い，システムに残っている機能を最大限に利用して自動的に制御系を再構成することにより，航空機の安全の確保及び制御性能の回復を可能にする機構」

一般にＲＦＣＳでは，航空機の運動制御に関する以下ようなの故障を考える．

①航空機の固有動特性に影響する故障
　主翼，尾翼，胴体などの構造部材の破損や形状変化
②制御器に関する故障
　舵面，アクチュエータ，フラップ，スラット，（可動）水平安定板，スピードブレーキなどの破損や固着，エンジンの停止
③センサーに関する故障
　加速度計，ジャイロ，レートジャイロ，速度計，ピトー管などの異常

ＲＦＣＳと似た言葉で，再形成可能な飛行制御系(Reconfigurable FCS)，自己修復型飛行制御系(Self-Repairing FCS)，耐故障型飛行制御系(Fault Tolerant FCS)などがある．「再形成」は予想される故障に対処でき，「再構成」は予想されない故障にも対処できると区別されるが，同様な意味で使われることも多い．「自己修復型」は米国で進められているあるプロジェクトの中で使用されている（文献1-36,1-37）．これについては，後で紹介する．「耐故障型」はロバスト制御のみによる飛行制御系も含むと考えられる．

ＲＦＣＳが実際の使用に耐えるためには，次のようなかなり厳しい要求を満たす必要がある．

①故障の影響が無視できるものから重大なものまで，さまざまな故障によるシステムの変化に対処できること．
②非常に不確定で雑音の多いシステム環境の下で機能すること．
③再構成により機能が著しく低下しないこと．
④再構成の間，そしてそれ以後もシステムの性能を維持できること．
⑤故障の状況をオペレータ（パイロット等）に効率よく通報できること．

ただし，⑤はパイロット支援システムとして働くので本論文では考慮しない．

1.2.2 節で述べたようにＲＦＣＳの大きな特徴は「残存している正常な制御器の運用法を変更することにより故障した制御器の機能を補償する」ことである．これを機能的冗長性(functional redundancy)という（文献1-22,1-38）．これにより「システムに残っている機能を最大限に利用する」ことが可能となる．例えば，昇降舵は通常，縦の運動の制御にしか用いない．しかし，左右の昇降舵を差動させればロール運動を制御することも可能である．このとき，昇降舵は補助翼とともにロール運動に関して冗長な制御器になっている．

より広い範囲の故障に対して再構成ができるためには，航空機はより大きな機能的冗長性をもつことが望まれる．ところで，機体設計の初期の段階から制御を行うことを考慮してつくられた航空機をＣＣＶ(Control Configured Vehicle)という．このような航空機では，より高度な機能をもたせるために，従来の機体にはない新しい制御器（垂直及び水平カナード，スラスト・ベクトリングなど）を備えている．従って，ＣＣＶのように新しい舵面をつけたり，従来の舵面でもそれを分割し独立に作動できるようにすることにより，機能的冗長性を増し，再構成の潜在的能力を高めることができる．さらに，従来は制御器として用いていなかったスピードブレーキやランディングギア格納扉を利用することも考えられる．例えば，文献1-33,1-39,1-40,1-41,1-42などで扱われている航空機は再構成により適した機体になっている．

1.3.2 ＲＦＣＳの典型的構成

上で定義されたＲＦＣＳを実現する一つの典型的な構成法が図１－３に示されている（文献1-35）．これは３つのモジュールから構成される．まず，基本コントローラであるロバスト多変数制御系を設計する．次に故障の検出・同定モジュールでは故障発生の検出を行い，それがどのような故障かを同定する．その結果を用いて自動再設計モジュールで故障の影響を補償するようにノミナルの制御系を再設計する．小さな故障の影響は主にロバスト制御系のロバスト性により補償され，大きな故障の影響は主に故障の検出・同定及び再設計を行うことによって補償される．各モジュールをもう少し詳しく説明する．

１）ロバスト多変数制御系

故障がない場合のノミナルの制御系は雑音や外乱，小さな環境変化（故障を含む）に対して，性能を維持できるロバスト性を持っていなければならない．ロバスト性は故障の検出・同定，及び再構成の間，性能が著しく低下しないためにも重要である．また，ＲＦＣＳは多くの制御入出力を扱うので本質的に多変数系である．

２）ＦＤＩモジュール

故障の検出・同定はＦＤＩ(Failure Detection and Identification)と呼ばれる．再構成においては，ＦＤＩがいかに速く正確に行えるかが，決定的に重要である． 即ち，制御系を再構成するためには，迅速かつ正確に故障(Failure)が生じたことを検出(Detection)し， その箇所を同定(Identification)しなければならない．ＦＤＩはもともと冗長性をもったセンサーやコンピュータシステムにおいて用いられた言葉である．そこでは，故障の検出・同定の後，さらに故障した部分を分離すること(Fault Detection and Isolation) をＦＤＩは意味した．したがって，分離を要する機器の場合には，ＦＤＩのＩはIsolation を指す．いずれにせよ，同定までは必要であるから，以下ではIdentificationを指すとする．ＦＤＩにおいてもロバスト性が要求され，雑音や外乱がある環境下やモデルに不確かさがある場合にも正しく機能することが重要である．

ＦＤＩには大きく分けて，２通りの方法がある．一つは，多重仮説検定法(Multiple Hypothesis Testing Method)である．この方法は，まず故障に関する多くの仮説（発生箇所，種類，程度）を設け，それらの一つ一つが正しい確率を求める．そして，その中で最も正しいと思われる（確率が大きい）仮説を採用して，故障が発生したかどうか，また，どこにどのような故障が発生したのかを決定する．仮説検定には多くの方法が提案されている（文献1-42）．もう一つは，パラメータ同定による方法である．これはシステムの挙動を記述する状態方程式や伝達関数のパラメータを最小自乗法などの同定アルゴリズムを用いて推定する．ただし，この方法は故障の影響がパラメータ変化として現れない場合には使えない．また，具体的な故障箇所や故障の種類の特定はできない．例えば，センサーやコンピュータ，通信機器などのＦＤＩには使えない．このような故障のＦＤＩには仮説検定法によらなければならない．しかしながら，後で述べるようにパラメータ同定による方法は仮説検定と比べて多くの長所をもつ．

なお，本論文ではセンサーの故障は扱わないが，そのＦＤＩには解析的冗長性と呼ばれる重要な概念が用いられる（文献1-42,1-43）． これは機能的冗長性の考え方に似ており，１つのセンサーを何通りにも使って（例えば，加速度計の出力を積分すれば速度計として使える）過剰な要素冗長性を抑えると同時に，故障

したセンサーの機能を補償するものである．

3）自動再設計モジュール

ＦＤＩにより得られた情報を用いて制御則の修正を行う．具体的には，フィードバックゲインなどノミナルの制御系のパラメータを修正したり，あるコントロール・ループを付加したり，あるいは除去したりする．また，舵面や機体の一部が破損したり，舵面が固着したりした場合には，トリムを修正する必要がある．そのとき，状態変数のトリム値は物理的に望ましい値（例えば，迎角の場合には失速迎角から十分離れている）にあり，制御変数のトリム値は，操縦のために十分な操舵可能範囲が残されるように修正されなければならない．

図１－３と同様の機能をもつ制御手法として適応制御がある．これはパラメータ同定によりＦＤＩを行う場合に相当する．特に，制御対象のパラメータを同定し，その結果を用いて新しい制御パラメータを求める方法は，上述の（ＦＤＩ＋コントローラの再設計）の方式に含めて考えることができる．これは間接法適応制御と呼ばれる．他方，適応制御には制御対象のパラメータではなく，直接，制御則のパラメータを調整する方式がある．これは直接法適応制御と呼ばれる．直接法はＦＤＩモジュールと再設計モジュールが一つになったような方式である．いずれの方式でも適応制御は環境条件によるシステムの特性変動に対しても故障と同様に対処できるという特徴を持つ．しかし，特に線形の適応制御系は，文献1-35，1-44が指摘しているように，外乱や非モデル化動特性の存在に弱く，実際問題へ適用するにはこのような点に関してロバスト性をもたせることが必要である．

1.3.3　ＲＦＣＳの設計

本節では，図１－３に示された各モジュール毎に，適用できる制御及び同定・推定手法について検討する．そして，本研究で採用する手法を他の設計例の手法と比較することによりその特徴を明らかにする．

1）ロバスト多変数制御系

従来そして現在でも航空宇宙の産業界では古典制御を用いるケースが多い．古典制御は１入出力系に対しては非常に有力な設計手法であるが多入出力系を扱うことは一般に難しい．一方，線形制御理論においては，近年LQG/LTRやＨ∞制御に代表されるロバスト多変数制御の研究がさかんである（文献1-45）．これらの方法の大きな特徴は従来，基本的に１入出力系に限られていた周波数領域での設計法（古典制御）を多入出力系に拡張した点である．航空宇宙の分野でもＨ∞制御などを適用した研究が増えてきている．例えば，Hubble Space TelescopeやX-29の姿勢制御などにＨ∞，H_2制御を適用した例がある（文献1-46,1-47）．

ところで，古典制御ではゲインを求めるのにボード線図を書いたり根軌跡を描いたりするなどかなり手間がかかる．一方，現代制御では古典制御よりも合理的に良い制御パラメータが得られる．しかし，ＬＱＧやＨ∞制御系の設計法を見て分かるように重みの選び方など試行錯誤的にならざるを得ないことが多い．さらに，ゲイン等の決

定には通常複雑な計算を要する．このためいずれの手法を用いるにしても，制御パラメータを決めるのは簡単ではなくオンラインでそれらを調整することは難しい．この対策としてゲインスケジュール方式が考えられるが，設計点が限られるという問題がある．これについては３）で述べる．このように，再設計のことを考えるとＬＱＧやＨ∞制御のような手法はＲＦＣＳの基本コントローラには向かないと言える．以上のことから「基本コントローラの設計法としてはできるだけ簡単なものがよい」という指針を得る．

ここで注意しておきたいのは「ＬＱＧやＨ∞制御のような手法が飛行制御系に向かない」と言っているのではないことである．この問題は基本コントローラをどのように考えるかということに関係する．つまり，基本コントローラをノミナルの飛行制御系とするか，それともＲＦＣＳのための特別な制御系とするかということに関係する．前者の場合は通常の飛行時にもＲＦＣＳの基本コントローラが航空機を制御しており，後者の場合は基本コントローラとは別の制御系が制御を行っていることになる．前者の方がＲＦＣＳも含めた飛行制御系全体の構成が簡単になる．そして，図１－３は前者の立場から描かれたと考えられる．しかし，上で述べたようなことを考えると基本コントローラをノミナルの飛行制御系とすることは必ずしも好ましいとは言えない．むしろ，両者を分離し，正常時には十分慎重に設計されたロバスト制御系で制御して，何かの異常が検出されればＲＦＣＳによる制御に切り替える方がよいのではないかと考えられる．これは本論文で扱うＲＦＣＳの設計の範囲を越えた問題であるが，実際にＲＦＣＳの搭載を考えた場合には重要な問題である．

ところで，上で述べた古典あるいは現代制御はいずれも線形制御理論である．つまり，制御対象が線形系であることを前提としている．しかし，多くの実用例から分かるように線形制御はかなり非線形な環境でも有効に働くことが知られている．航空機も通常，縦と横・方向運動の分離した線形系として扱われる．ところが，故障は時として重大な非線形性を引き起こす．それ故，線形制御では補償できない場合が出てくる．制御対象を線形と仮定した場合により問題となるのはＦＤＩである．仮説検定法でもパラメータ同定法でも一般にモデル誤差があると同定性能が大きく低下する．実際，線形の適応制御の実用化が難しいのもこの点によるところが大きい．線形制御にはもう一つ問題点がある．それはトリム点を求めなければならないことである．舵の固着，エンジンの停止，翼の一部破損などによってトリム点は変化する．そして，トリム点が分からなければ状態変数や制御変数を正しく定義できない．このように線形系の同定では微係数等のパラメータに加えてトリム点の同定も必要になる．以上のことから第２の指針として「制御則並びに同定モデルは航空機運動の非線形性を考慮できること」が挙げられる．

2）ＦＤＩ－仮説検定法とパラメータ同定法

ＲＦＣＳに関する研究の多くはＦＤＩとして仮説検定法を考えている．これに対し，本論文ではパラメータ同定法を用いる．以下では，ＲＦＣＳにおけるＦＤＩ手法として両者を比較しそれぞれの特徴を示す．そして，本研究でパラメータ同定法を用いた理由を述べる．

ＦＤＩは本来，ハードウェアの各構成要素の異常を検出・同定（分離）することである．それは基本的に各要素に対しあるタイプの異常が生じたか否かということを絶えずチェックしていくことと言ってよい．しかし，制御系の再構成を考えた場合，必要なのは故障後の航空機の動特性や制御性である．つまり，ハードウェア個々の故障そのものよりも，故障が動特性に与える影響を知ることの方が重要である．それは再構成が故障後の航空機モデルに基づいて行われるからである．勿論，そのような影響の一部は仮説検定型のＦＤＩによっても同定可能である．例えば，各舵面に対して破損や固着，フローティングなどの仮説を設けておき，適当なＦＤＩの手法を適用することはできる．その結果に基づいて制御微係数などの運動方程式を修正することもできる．しかし，動特性に重大な影響をもち，かつ仮説に含まれないような故障が生じる場合もある．また，複数の故障が同時に発生した場合も，仮説に基づく方法では考慮できるケースが数的に限られてくる．従って，再構成を行う場合には，個々の故障よりも航空機の動特性そのものを同定した方が都合がよい．故障の影響がパラメータに集約されており，しかも同定結果を直ちに再設計に利用できるからである．故障直後の緊急制御では故障箇所の特定は必ずしも要求されない．むしろ，熟練したパイロットのように機体応答から動特性の変化を知り，適応的に航空機の釣合を保つことの方が重要である．

ここで述べたことは，仮説に基づくＦＤＩの有効性を否定しているのではない．先に述べたようにセンサーやコンピュータなど，航空機動特性に直接影響しない機器の故障は仮説に基づくＦＤＩによらなければならない．また，分離を必要とする故障は，その箇所を特定しなければならないのでこれも仮説に基づくＦＤＩを用いる必要がある．一方，舵面やアクチュエータに関する故障でも，故障情報をパイロットに知らせることは飛行計画の変更など緊急事態から脱した後の操縦にとって重要である．さらに，地上での保守にとってもハードウェア各要素の故障情報は必要である．これらを行うためには仮説に基づくＦＤＩにより具体的な故障を明確にすることが要求される．

仮説検定法とパラメータ同定法によるＦＤＩの特徴をいくつかの点について比較すると表１－２のようになる．

これら２つの同定法はそれぞれ特徴をもっており，それが生かされるように使われるべきである．そして，故障後の緊急制御を行うという目的に限れば，パラメータ同定を用いて飛行制御系の再構成を行う方が適当ではないかと考えられる．

３）自動再設計

自動再設計には次の４通りの方法がある．

①故障後の入力の状態変化に対する影響が正常な場合と同じになるようにする．

②故障後の航空機に対し基本制御系の制御則を再度求める．

③制御則を何かの調整則に従って調整する（直接方式の適応制御）．

④人工知能（エキスパートシステム）を用いる．

以下では各方法について説明しその長所や短所を指摘する．またそれらの方法を用いた設計例を紹介する．

① これはコントロールミキサー法と呼ばれ，制御入力は正常な制御器がつくり出す

のと同じ力やモーメントを故障した制御器が発生するように修正される．具体的には入力を次のように修正する．

$$\delta_1 = E\delta_0 \tag{1.3-1}$$

ここで，$E = \{I + B_1^+(B_0 - B_1)\}$ であり，添字0,1はそれぞれ正常な場合及び故障した場合の値であることを示す．B_0，B_1は入力に関する係数行列である．また，B_1^+はB_1 の擬似逆行列である．この方法の特徴は次に述べる②の方法と異なり，フィードバックゲインなどの制御パラメータを計算し直す必要がないことである．もっとも，ノミナルの制御パラメータにＥ行列をかけたものを新しいパラメータとみれば，制御パラメータを修正していることになるが，基本的にノミナルの制御則がどんなものであっても単にＥをかけるだけで再設計は完了する．この方法は文献1-48で用いられている．なお，この方法を用いたSelf-Repairing FCSはモーションシミュレータによるシミュレーション（文献1-49），さらに実機による飛行試験（文献1-50，1-51）も行われている．一般入力と呼ばれる仮想的な入力を導入した場合も全く同様の考え方で入力を修正できる（文献1-52,1-53）．この場合は入力分配行列P_0を次のように修正する．

$$P_1 = B_1^+ B_0 P_0 \tag{1.3-2}$$

ここで，P_1 は故障後の入力分配行列である．本論文でも一般入力及び入力分配行列を用いている（第３章を参照）．ただし，提案する方法では入力分配行列は修正しない．

この方法は簡単で分かりやすいが，問題点としては故障が制御器に関するもので，しかもＢ行列に影響するものに限られることが挙げられる．例えば，舵面の部分的な破損には有効であるが，舵の固着は扱えない．また，翼の破損など固有の動特性が変化するような故障にも対応できない．

② これまで行われた研究の多くはこの分類に属する．この方法で再設計を行う場合，２通りの方式がある．一つはＦＤＩの結果に基づいてオンラインで制御則を修正する方式で，もう一つは予め計算された制御則を選び出すゲインスケジュール方式である．

・オンライン再設計方式

この方法は間接方式の適応制御と同じ考え方である．１）で指摘したように，基本制御系の設計が複雑な場合はこの方式をとることは好ましいとは言えない．これに属すると見られる研究例としては，例えば，文献1-54が挙げられる．この例では，制御器が故障した場合を考え，ＦＤＩにより同定された制御微係数を用いて最適レギュレータのフィードバックゲインを計算し直す．フィードバックゲインの計算において評価関数の重み行列は，一巡伝達関数のゲインができるだけ大きくなるように選ばれる．また，同文献では２次計画法を用いてトリム点を修正する方法も示している．別の例としては，モデル追従制御を用いた研究もいくつか行われている（文献1-55～58）．航空機に故障が発生しても出力が正常な場合と同じモデルに追従できれば制御性能は回復できたことになる．これらの設計例では，再設計においてＬＱ問題や２次計画問題を解かなければならないので再設計は必ずしも簡単ではない．

・ゲインスケジュール方式

通常，空力的な環境変化に対処するために使われる方法である．いろいろな飛行条件に対して，あらかじめコントローラのパラメータ（ゲインなど）を計算しておく．そして，実際の飛行条件の変化に合わせてそれらのゲインを変更していく．故障に対しても発生した故障や損傷が同定できれば，同様の考え方によりそれらを補償することができる．即ち，あらかじめ想定されたいろいろな故障に対して適切なパラメータを計算しておき，故障が発生した場合にはその故障に対応するパラメータを選び出してコントローラを修正すればよい．この方式を用いた例として，文献1-59がある．これはＬＱＧゲインの計算を，線形２次最適化に基づく同時安定化アルゴリズムを用いて行っている．スケジュールされたゲインはいろいろな故障モードの航空機を表現するプラントモデルの中のいくつかを同時に安定化するように決定される．これにより，同定誤差のために故障モードの選択が真のものから多少ずれていても安定性は保たれる．静的に不安定な高性能機に対して計算機シミュレーションを行いその有効性を示している．ゲインスケジュールによる設計は，概念的に簡単であり複雑な計算もオンラインでは必要ないので実際に適用しやすいが，設計点あるいは設計領域が限られてしまう点に問題がある．

②の２つの方法に共通な欠点は，まず同定が完了するまで再設計ができない点である．しかも，同定には適当な試験入力を与えることが望まれる．しかし，緊急制御を要する状況でそのような入力を与えることは事態を悪化する危険がある．また，同定されたパラメータを真値とみなして再設計を行うので，同定精度が制御性能に大きく影響する．つまり，同定が正確に行われなければならない．この問題については文献1-59のように制御則をロバスト化することによって，同定誤差の影響を軽減することはできるが，同定誤差が大きいと十分な性能を回復できないこともありうる．

③　これは直接方式の適応制御と呼ばれている．②がＦＤＩを行った後に自動再設計を行うという方法であったのに対し，③はＦＤＩと自動再設計が一体になったような方法である．つまり，航空機の故障自体は同定しないで，結果的に制御目的が達成されるように制御パラメータを直接調整するものである．この方法は故障直後から再設計が始まっているので，上で指摘した②の方法の問題がない．直接方式を用いた例としては，文献1-60がある．これは超安定理論を用いたモデル規範形適応制御系をＲＦＣＳとして設計している．突風，モデル誤差，舵面の飽和が存在してもシステムが安定に作動することをシミュレーションにより確かめている．直接方式の適応制御は主に線形系に対して発展したが，非線形系でも状態方程式が特殊な形（定数パラメータ及び制御入力に関して線形）をしている場合には適用できる．本研究で扱う航空機の非線形モデルはこの条件を満たすので，直接方式の適応制御系が構成できている．

熟練したパイロットは故障直後の機体応答から動特性の変化を知り，適応的に航空機の釣合を保つ．そのとき，パイロットは故障箇所が分かるかもしれないし分からないかもしれない．この方法はちょうどそれに似ており，自然な補償の仕方に思える．

④　人工知能による方法は数式に基づいた制御則を用いないという点で上で述べた方法とは全く異なる．人工知能は数式ではモデル化できないような状況も扱うことができ，強力なＲＦＣＳを構成できる可能性がある．例えば，文献1-29はまず，現在提案

されている多くの耐故障型(failure-robust)ＦＣＳまたはＲＦＣＳはそれぞれ限界があり十分な故障回復が期待できないことを指摘している．一例として，故障が引き起こす運用上及び性能上の拘束（速度と必要推力・利用可能推力の関係など）を陽に考慮できない点が挙げられている．そして，同文献はこのような問題を解決するためにエキスパートシステムを用いることを提案し，その具体的な設計法を示している．

他方，このような回復制御を行う機能とは別に，人工知能には基本コントローラ，ＦＤＩ，再設計を統合したシステムを効率的に運用する管理者としての機能をもたせることもできる．例えば，文献1-61はルールに基づいた耐故障型飛行制御系を提案している．この制御系の目的はセンサー，制御器，機体構造の故障を含む広い範囲の故障に対処することであり，その特徴は符号演算（推論，探索）と数値計算（状態推定，制御則の計算など）とをうまく統合している点である．

人工知能はＲＦＣＳの有望な設計手法になる可能性はあるが，文献1-29のように過渡応答を制御するコントローラとして用いることは問題があると思われる．その理由としてまず，仮説検定法のように考慮できる故障や異常の範囲が限られる点が挙げられる．さらに，制御の仕方は故障状態だけでなく初期状態や飛行条件によっても変わるので，それらを全てカバーすることは不可能であろう．むしろ，飛行制御は数式に基づくＲＦＣＳを主とし，故障状況に応じてそれを補助するような形で人工知能を用いるのが良いのではないかと考える．例えば，望ましい姿勢，迎角，速度，飛行高度などの設定，あるいは通常，飛行制御には用いられない制御器（フラップ，エンジン，水平安定板，降着ギアなど）の使用の決定などに使えばよい．

1.3.4　まとめ

ＲＦＣＳの基本コントローラの制御則は制御性能やロバスト性も考慮しなければならないが，制御系の再設計のことを考えると制御則の導出ができるだけ簡単なものがよい．そして，制御則並びに同定モデルは運動の非線形性を考慮できるものであることが望ましい．これは故障が引き起こす非線形性に対してロバスト性をもつことが必要だからである．また，非線形系として扱えばトリム点も特に考慮しなくてよい．ところが，上で紹介した例も含め，これまで行われたＲＦＣＳに関する研究は線形制御手法に基づいたものである．一方，ＦＤＩとしては，過去の研究例では制御器の故障に対しても仮説検定法を用いることを考えているものが大部分である（研究例の中にはＦＤＩについて特に触れてないものもある）．しかしながら，ＲＦＣＳ設計の立場からみれば仮説検定法よりもパラメータ同定法の方が適していると考えられる．

以上の経緯から，本研究では基本コントローラに設計の簡単な非線形制御手法を，そして，ＦＤＩにパラメータ同定法を用いている．しかも，ＲＦＣＳ全体が適応制御系になっており，回復制御が故障直後から始まるので緊急制御に有利である．このように，本研究のような手法は過去のＲＦＣＳの研究では用いられておらず，従来のものにはない優れた性能をもつことが期待できる．

1．4　非線形補償による再構成可能な飛行制御系

前節で述べたようにＲＦＣＳに関して多くの研究がこれまでなされてきた．筆者の知る限りでは，それらの研究は全て基本制御系の設計あるいは再構成において，線形制御理論に基づいている．線形制御系は実際には非線形な環境でもかなりうまく働くことが知られている．しかし，故障はしばしば非常に非線形性の強い運動を引き起こす．例えば，縦と横・方向が強く連成した運動やノミナルのトリム点から大きくはずれた運動を伴う．そのような場合には，線形モデルはもはや故障した航空機の動特性を十分には表現できない．その結果，線形制御に基づいたＲＦＣＳは故障を補償できなくなる．本研究はこの点を考慮して，非線形の運動方程式に基づき，非線形の制御法を用いてＲＦＣＳを設計する．非線形制御はフィードバック線形化法（文献1-62, 1-63）を用いる．この方法は非線形逆力学(Nonlinear Inverse Dynamics;NID)法としても知られ，航空機の飛行制御（文献1-64,1-65,1-66）やロボットのマニピュレータ制御（文献1-67）にも用いられている．一方，故障は非線形運動方程式のパラメータを逐次推定することにより同定される．推定アルゴリズムは繰り返し型最小自乗法(recursive least squares meth-od)が用いられている．同定結果は直ちに制御則の修正に用いられる．即ち，制御パラメータはプラントパラメータの推定値を用いて逐次修正される．この点で，本論文で提案する方法は適応制御の分類に属するものと言える．制御パラメータの更新においては複雑な計算はいらない．プラントパラメータと制御パラメータは１対１に対応しているので，単にノミナルのパラメータを推定値に置き換えればよい．その他にもいくつかの工夫がなされている．例えば，機能的冗長性を生かすための一般入力及び入力分配機構の導入，高周波の入力振動を抑えるための周波数依存型最適レギュレータの利用などがある．このＲＦＣＳにより，線形制御に基づいたものでは補償できないような大きな故障からも回復できることを計算機シミュレーションにより示す．

本研究で考えるＲＦＣＳは故障直後の緊急制御として自動的にトリム状態を回復することを目的とする．故障としては航空機運動に影響を及ぼす制御器の異常や機体の一部の破損などを考え，センサーや機上計算機は正常であると仮定する．また，全状態変数が入手可能であるとし，システムは確定系として扱う．即ち，ノイズの影響は考慮しない．

本論文の構成は以下の通りである．第２章では，機能的冗長性が利用される例を示す（文献1-5）．そこでは垂直尾翼に損傷を受けたBoeing 747型機に対して，両翼のエンジンの推力制御のみによる横・方向運動の安定化制御を試みる．航空機の運動は線形化運動方程式で表され，制御方法は最適レギュレータである．第３章では，本論文の主題であるフィードバック線形化法を用いたＲＦＣＳの設計法を示す（文献1-18）．航空機の運動は６自由度非線形運動方程式で記述し，提案したＲＦＣＳを小型高速機に適用した場合のシミュレーション結果を示す．舵面の一部が破損したり，固着したりした場合に水平飛行が回復できることを示す．次に，第４章では，実際の装置化を考慮して改良したものを示す（文献1-21）．このＲＦＣＳでは航空機運動の制御とアクチュエータ及びエンジン系の制御を分離することにより制御則が複雑になることを避けるとともに，制御入力の離散化を行っている．即ち，まずアクチュエータ及びエンジン動特性を無視した航空機モデルに対して，フィードバック線形化法による制御則を求める．そして，次にアクチュエータ，エンジン系に対して離散時間型のサーボコントローラを設計する．このとき，サーボ系の目標値はフィードバック線形化法によって求められた制御則で与えられる．このＲＦＣＳを主翼が破損した小型高速機に適用し，やはり水平飛行状態に回復できることをシミュレーションにより示す．第５章では，離散時間ＲＦＣＳを大型旅客機(Boeing 747)に適用する（文献1-20）．特に舵面が固着した場合に，通常，過渡的な姿勢制御には使われない水平安定板やエンジン推力をうまく用いることにより，安定な飛行が保たれることを示す．最後に，第６章で結論を述べる．

表1－1　インシデント及び事故例

No.	年月日	場　所	機　体	状　況	結果	原　因
1	1979.11.11	ﾙｸｾﾝﾌﾞﾙｸﾞ上空	ｴｱﾛ･ﾒｷｼｺ航空 DC-10	上昇中に失速し，約１分(8,500ft)落下 水平尾翼の一部破損.	帰還	パイロット ｵｰﾄﾊﾟｲﾛｯﾄ/ｵｰﾄｽﾛｯﾄﾙの誤使用，計器類の監視不十分
2	1985.2.19	ｻﾝﾌﾗﾝｼｽｺ上空	ﾁｬｲﾅ･ｴｱﾗｲﾝｽﾞ B-747	第4ｴﾝｼﾞﾝ停止後，通常の手順に従わず，操縦をｵｰﾄﾊﾟｲﾛｯﾄに任せきりにする. 操縦不能に陥り，約10000m落下.	帰還	パイロット 自動操縦への過度の依存，計器類の監視不十分
3	1992.1.20	ｽﾄﾗｽﾌﾞｰﾙ近郊 (ﾌﾗﾝｽ)	ｴｱ･ｱﾝﾃｰﾙ航空 A-320	ILSの設定されてない方向から進入中，急降下し，山に衝突.	墜落	パイロット ｵｰﾄﾊﾟｲﾛｯﾄの降下角と降下率を誤ってセット
4	1967.3.10	ｵﾚｺﾞﾝ州 ｷﾝｸﾞｽﾚｰ空港	ｳｪｽﾄｺｰｽﾄ航空 ﾌｫｯｶｰ F-27	離陸滑走中にｼｬｰﾍﾞｯﾄ状の雪が舵のﾋﾝｼﾞ部に入り氷結．舵が固着して操縦不能に陥る.	墜落	舵の固着
5	1978.2.12	ｳｪｽﾄﾊﾞｰｼﾞﾆｱ州 ﾍﾞﾝﾀﾞﾑ空港	ｱﾘｹﾞﾆｰ航空 ﾉｰﾙ262	離陸直後，積雪による翼端失速からﾛｰﾘﾝｸﾞを生じる.	墜落	積雪による揚力の変化 →動特性の変化
6	1968.12.27	ｱｲｵﾜ州 ｽｰｼﾃｨ空港	ｵｻﾞｰｸ航空 DC-9	主翼の着氷により，離陸後失速.	墜落	着氷による揚力の変化 →動特性の変化
7	1983.7.23	ｵﾝﾀﾘｵ州 ﾚｯﾄﾞ湖上空	ｴｱ･ｶﾅﾀﾞ航空 B-767	燃料切れにより両ｴﾝｼﾞﾝ停止．近くの滑走路に滑空して着陸.	帰還	地上係員 燃料の計算ミス
8	1985.9.6	ﾐﾙｳｫｰｷｰ州 ﾋﾞﾘｰﾐｯﾁｪﾙ空港	ﾐｯﾄﾞｳｪｽﾄ･ｴｷｽﾌﾟﾚｽ,DC-9	離陸上昇中に右ｴﾝｼﾞﾝに異常．ﾊﾟｲﾛｯﾄの誤った回復操作により失速.	墜落	パイロット 操縦ミス
9	1977.5.25	ｼｶｺﾞ ｵﾍｱ空港	ｱﾒﾘｶﾝ航空 DC-10	離陸直前に第1ｴﾝｼﾞﾝ脱落．左側ｽﾗｯﾄがﾘﾄﾗｸﾄ．左にﾛｰﾙし左翼が失速.	墜落	ｴﾝｼﾞﾝのとりつけミス． ﾊﾟｲﾛｯﾄは異常に気づかず.
10	1992.10.4	ｱﾑｽﾃﾙﾀﾞﾑ ｽｷﾎﾟｰﾙ空港	ｴﾙｱﾙ航空 B-747	離陸後，第3ｴﾝｼﾞﾝ停止，第4ｴﾝｼﾞﾝ脱落．空港に帰還中，ﾌﾗｯﾌﾟに異常.	墜落	ｴﾝｼﾞﾝ取り付け部に欠陥？
11	1979.9.17	ﾎﾞｽﾄﾝ東海上	ｴｱ･ｶﾅﾀﾞ航空 DC-9	後部圧力隔壁及びﾃｲﾙｺｰﾝが破損．操縦系統の一部も壊れるが操縦は可能.	帰還	圧力隔壁ｳｪﾌﾞに金属疲労

No.	年月日	場　所	機　体	状　況	結果	原　因
12	1986.10.26	四国上空	ﾀｲ航空 A-300	ﾄｲﾚに手投げ弾が持ち込まれ爆発．後部圧力隔壁が破壊．３つの油圧系統のうち２つを失う.	帰還	人為的な破壊
13	1988.4.28	ﾊﾜｲ ﾏｳｲ島上空	ｱﾛﾊ航空 B-737	前部客室の床から上の外板を突如失う.	帰還	外板の継ぎ目に無数のｸﾗｯｸ
14	1971.10.2	ﾍﾞﾙｷﾞｰ上空	BeA ｳﾞｨｯｶｰｽﾞ･ﾊﾞﾝｶﾞｰﾄﾞ951G	後部圧力隔壁が破壊．噴出した気流が水平尾翼中央部を直撃し破壊．左右の水平尾翼が分離.	墜落	腐食から隔壁に亀裂が発生
15	1985.8.12	御巣鷹山	日本航空 B-747	後部圧力隔壁及びﾃｲﾙｺｰﾝが破損．さらに垂直尾翼の前部，方向舵及び全油圧系統が破壊．操縦不能に陥る.	墜落	隔壁の修理ミス
16	1972.6.12	ｵﾝﾀﾘｵ州 ｳｨﾝｻﾞｰ上空	ｱﾒﾘｶﾝ航空 DC-10	後部貨物室ﾄﾞｱが吹き飛ぶ．貨物室の急減圧により床が落ち込み，第2ｴﾝｼﾞﾝを操作するﾜｲﾔが切られる．方向舵が固着，昇降舵も十分に動かない.	帰還	貨物室ﾄﾞｱのﾛｯｸ機構の欠陥
17	1974.3.3	ﾊﾟﾘ郊外	ﾄﾙｺ航空	16と同じ状況が発生．ただし，操縦系統のﾀﾞﾒｰｼﾞが大きく制御不能に陥る.	墜落	貨物室ﾄﾞｱのﾛｯｸ機構の欠陥 改善勧告に従わず，作業員の訓練が不十分.
18	1989.2.24	ﾎﾉﾙﾙ沖	ﾕﾅｲﾃｯﾄﾞ航空 B-747	右前方貨物室ﾄﾞｱとその上側が破壊．飛散したﾄﾞｱやｺﾝﾃﾅｰ，外板などが右主翼，第3，第4ｴﾝｼﾞﾝ，水平尾翼，垂直尾翼に損傷を与える.	帰還	貨物室ﾄﾞｱのﾛｯｸ機構の欠陥 類似のｲﾝｼﾃﾞﾝﾄの教訓が生かされず.

詳しくは文献1-28,1-68,1-69,1-70,1-71を参照されたい.

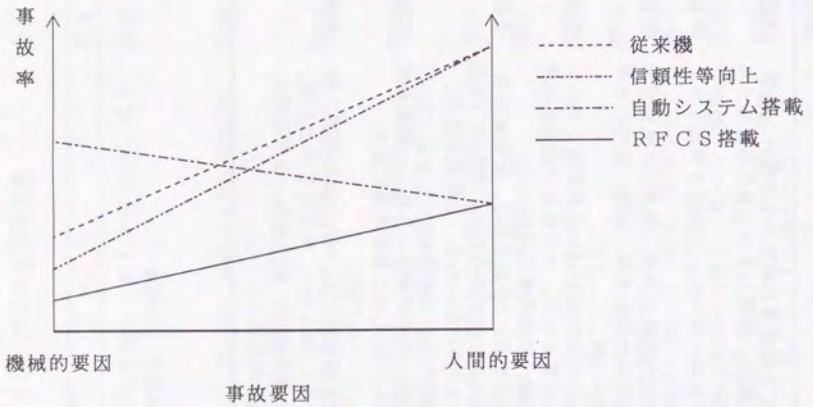

図１－１　事故対策と事故率

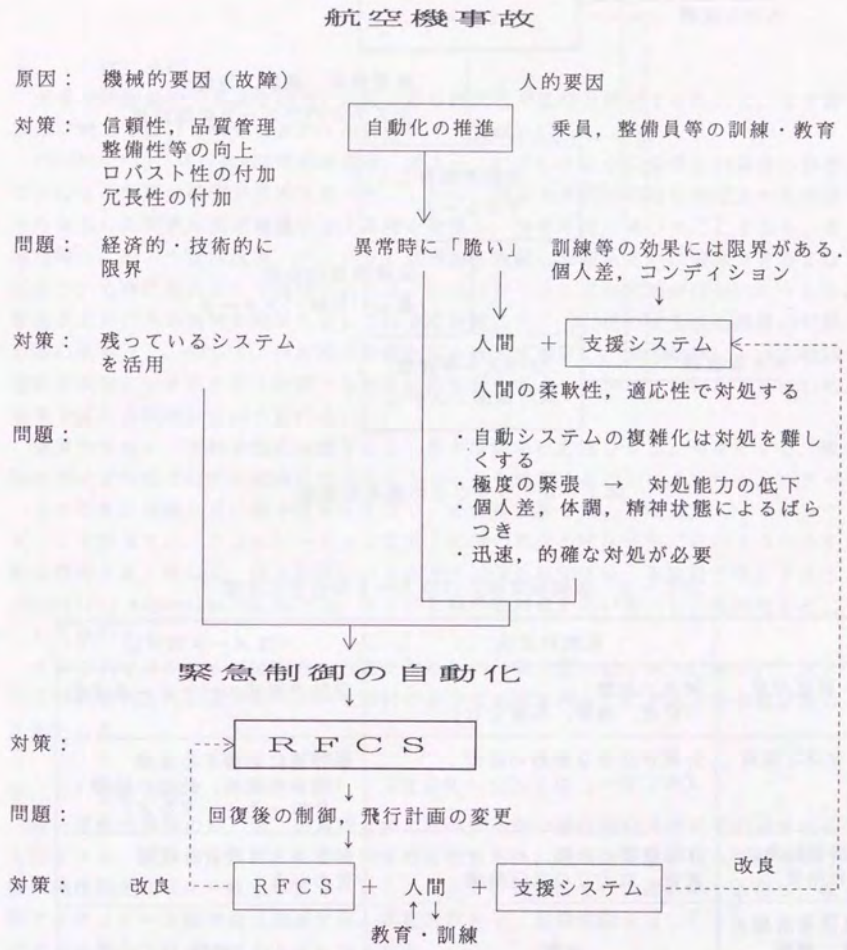

図１－２　航空機事故対策とＲＦＣＳ

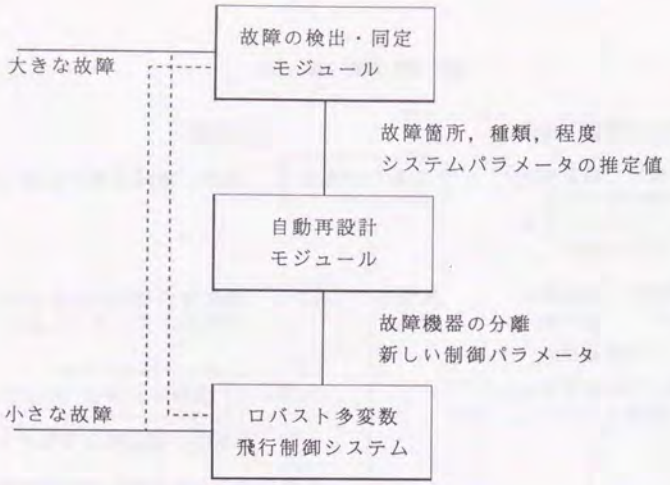

図１－３　ＲＦＣＳの基本的構造

表１－２　仮説検定法とパラメータ同定法の比較

	仮説検定法	パラメータ同定法
同定対象	個々の故障 （位置，種類，程度など）	運動方程式のパラメータ変化
有効な故障	分離が必要な機器の故障 （センサー，コンピュータなど）	動特性に影響する故障 （機体の破損，舵面の固着・フローティングなど）
同定結果の利用法	故障機器の分離，パイロットへの通報，地上での保守整備	制御系の再設計に直接使われる
扱える故障の数，種類，組み合わせ	有限	無限
False alarm/ Missed alarm	煩わされる	煩わされない
モデル依存性	フィルター法：大 分散法：比較的，小	大

２．線形系に対するＲＦＣＳの設計

２．１　はじめに

本章では機能的冗長性を利用したＲＦＣＳ設計の可能性を検討するために，まず線形系に対して設計されたＲＦＣＳの例を示す（文献1-5）．

1985年8月12日の日航123便の事故は，表１－１に示したように後部圧力隔壁の修理ミスによる隔壁の破壊が原因であった．しかし，墜落の直接の原因は与圧された客室から噴出した気流が垂直尾翼や油圧系統を破壊し，操舵不能に陥ったことである．事故当時，このような状況で，パイロットは機体を操縦し着陸あるいは着水できるかなどについて専門家によって議論されたり，シミュレータによる実験が行われたりした．筆者がＲＦＣＳの研究を始めた頃にこの事故が起こり，123便の安全性の回復の問題に関心をもった．そして，日航機の事故と似た状況を想定し，推力制御によって機体運動を安定化できるかどうか調べるために計算機シミュレーションを行うことにした．本章ではこの問題について述べる．

簡単のために，運動方程式は線形とし，基本制御系は最適レギュレータとする．故障の同定は運動方程式の微係数などのパラメータを推定することにより行う．パラメータの推定には繰り返し最小自乗法を用い，その推定値を用いて最適フィードバックゲインを計算する．シミュレーションでは日航機の事故の状況以外にもいくつかの状況を想定する．そして，推力制御による安全の回復の可能性や，事故発生後にＳＡＳ(Stability Augmentation System) などの自動操縦装置を用いることの危険性などについて検討する．

ここで示すＲＦＣＳは線形系を対象にするものの第１章の図１－３に描かれたＲＦＣＳの典型的な設計例であり，その設計手法や有効性を明らかにする上で意義があると思われる．

２．２　運動方程式

機体運動は通常の縦，横・方向が分離した線形化微小擾乱運動方程式で記述されるものとする．機体固定座標としては安定軸座標を用い，地面固定座標は水平定常飛行時に機体固定座標に一致するようにとる．エンジン動特性を一次遅れ系で近似し，舵面アクチュエータ動特性は無視する．状態変数をｘ，制御変数をu_cとすると，状態ベクトル微分方程式は次のように表される．

$$\dot{x} = A x + B u_c \tag{2.2-1}$$

ここで，縦の場合，

$$x = [u,\ \alpha,\ \theta,\ q,\ \delta_t]^T$$
$$u_c = [\delta_e,\ \delta_{tc}]^T$$

である．さらに，

u ＝ x 軸方向速度
α ＝迎角
θ ＝ピッチ角
q ＝ピッチ角速度
δ_t ＝エンジン出力
δ_e ＝エレベータ舵角
δ_{tc} ＝ δ_t に対するスロットル入力

である．また，A，Bはそれぞれ次の行列である．

$$A=\begin{bmatrix} X_u & X_\alpha & -g\cdot\cos\theta_0 & -W_0 & X_{\delta_t} \\ Z_u/U_0 & Z_\alpha/U_0 & -g\cdot\sin\theta_0/U_0 & 1+Z_q/U_0 & Z_{\delta_t}/U_0 \\ 0 & 0 & 0 & 1 & 0 \\ A(4,1) & A(4,2) & A(4,3) & A(4,4) & A(4,5) \\ 0 & 0 & 0 & 0 & -1/T \end{bmatrix}$$

$$B=\begin{bmatrix} X_{\delta_e} & 0 \\ Z_{\delta_e}/U_0 & 0 \\ 0 & 0 \\ M_{\delta_e}+M_{\dot\alpha}Z_{\delta_e}/U_0 & 0 \\ 0 & 1/T \end{bmatrix}$$

ここで，

$A(4,1)=M_u+M_{\dot\alpha}Z_u/U_0$，　$A(4,2)=M_\alpha+M_{\dot\alpha}Z_\alpha/U_0$，　$A(4,3)=-M_{\dot\alpha}g\cdot\sin\theta_0/U_0$

$A(4,4)=M_{\dot\alpha}(1+Z_q/U_0)+M_q$，　$A(4,5)=M_{\dot\alpha}Z_{\delta_t}/U_0+M_{\delta_t}$

Tはエンジン動特性の時定数である．

ここで，横・方向の場合，

$$x=[\beta,\ p,\ r,\ \phi,\ \Delta\delta_t]^T$$
$$u_c=[\delta_a,\ \delta_r,\ \Delta\delta_{tc}]^T$$

である．さらに，

β ＝横滑り角
p ＝ロール角速度
r ＝ヨー角速度
ϕ ＝ロール角
$\Delta\delta_t$ ＝左右エンジン出力差
δ_a ＝エルロン舵角
δ_r ＝ラダー舵角
$\Delta\delta_{tc}$ ＝ $\Delta\delta_t$ に対するスロットル入力

である．A，Bはそれぞれ次の行列である．

$$A=\begin{bmatrix} Y_\beta/U_0 & (W_0+Y_p)/U_0 & Y_r/U_0-1 & g\cdot\cos\theta_0/U_0 & 0 \\ L_\beta' & L_p' & L_r' & 0 & L_{\delta_t}' \\ N_\beta' & N_p' & N_r' & 0 & N_{\delta_t}' \\ 0 & 1 & \tan\theta_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1/T \end{bmatrix}$$

$$B=\begin{bmatrix} 0 & Y_{\delta_r}'/U_0 & 0 \\ L_{\delta_a}' & L_{\delta_r}' & 0 \\ N_{\delta_a}' & N_{\delta_r}' & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/T \end{bmatrix}$$

左翼側，右翼側のエンジン出力をそれぞれ δ_{tL}，δ_{tR} とすると，$\Delta\delta_t=\delta_{tL}-\delta_{tR}$ と定義される．$\delta_t=\delta_{tL}+\delta_{tR}$ である．

推力のみの制御では，縦の場合(2-1)式のBは第２列のみとなり，$u_c=\delta_{tc}$ となる．横・方向の場合も同様にBは第３列のみとなり，$u_c=\Delta\delta_{tc}$ となる．

垂直尾翼破損後の微係数の値は，無次元化微係数の推算式（文献2-1, pp.75-110）において垂直尾翼形状の関数になっているもの（$C_{y\beta}$, $C_{n\beta}$, C_{yr}, C_{lr}, C_{nr}, $C_{y\delta_r}$, $C_{l\delta_r}$, $C_{n\delta_r}$）のみを変更する．また，有次元微係数は破損後の質量，慣性モーメント，慣性乗積を変更する．

２．３　微係数の同定（文献2-2）

制御系を再構成するためには垂直尾翼破損後の微係数をオンラインで同定しなければならない．ここでは，繰り返し最小自乗法を用いる．

時刻ｋ－１以前の入出力情報から決まるベクトル $\phi(k-1)$ とパラメータベクトル θ^* により，あるサンプリング時刻ｋにおけるシステムの出力 $y(k)$ が

$$y(k-1)=\phi(k-1)^T\theta^* \tag{2.3-1}$$

と書けるものとする．このとき θ^* が未知である場合には，次の最小自乗推定アルゴリズムにより逐次的に θ^* の推定値 $\hat\theta(k)$ が得られる．

$$\hat y(k)=\phi(k-1)^T\hat\theta(k-1) \tag{2.3-2}$$

$$\hat\theta(k)=\hat\theta(k-1)+\frac{P(k-2)\phi(k-1)}{1+\phi(k-1)^TP(k-2)\phi(k-1)}[y(k)-\hat y(k)] \tag{2.3-3}$$

$$P(k-1)=P(k-2)-\frac{P(k-2)\phi(k-1)\phi(k-1)^T P(k-2)}{1+\phi(k-1)^T P(k-2)\phi(k-1)} \tag{2.3-4}$$

ここで，ϕ，$\theta^* \in R^{n+r}$（n：状態変数の数，r：制御変数の数）である．P(k)は(n+r)×(n+r) の共分散行列で，P(-1)は任意の正定行列とする．

文献2-2によると$\hat{\theta}(k)$は$k\to\infty$で一定値に収束することが保証されており，さらに$\phi(k)$が持続励振(persistent excitation)の条件を満たせば真値θ^*に収束する．後で示すシミュレーションでは同定のための試験入力として矩形波を用いており，この条件は満たされている．

状態変数x(k)が入手可能な場合には次のようにして(2.2-1)式のA，Bを同定することができる．サンプリング周期を適当に与えると(2.2-1)式は次の差分方程式に書き換えられる（付録２．３参照）．

$$x(k)=A_d x(k-1)+B_d u(k-1) \tag{2.3-5}$$

A_d，B_dを未知パラメータとし，時刻kにおける推定値をそれぞれ$A_d(k)$，$B_d(k)$とする．A_d，B_dをそれぞれ$A_d(k)$，$B_d(k)$に置き換えると

$$x(k)=A_d(k-1)x(k-1)+B_d(k-1)u(k-1) \tag{2.3-6}$$

となる．ここで，

$$x(k)=[x_1(k),\ldots,x_n(k)]^T$$
$$u(k)=[u_1(k),\ldots,u_r(k)]^T$$
$$A_d(k)=[a^1(k),\ldots,a_n(k)]^T$$
$$B_d(k)=[b_1(k),\ldots,b_n(k)]^T$$

（$a_i \in R^n$，$b_i \in R^r$）とおくと，(2.3-6)式より次の関係を得る．

$$x_i(k)=\begin{bmatrix} x(k-1) \\ u(k-1) \end{bmatrix}^T \begin{bmatrix} a_i(k-1) \\ b_i(k-1) \end{bmatrix} \tag{2.3-7}$$

$\phi(k)=[x(k)^T, u(k)^T]^T$, $\theta(k)=[a_i(k)^T, b_i(k)^T]^T$とおき，$y(k)=x_i(k)$ $(i=1,\ldots,n)$ とすれば(2.3-2)式から(2.3-4)式に示した最小自乗法が適用できる．これにより，行列A_d，B_dが同定でき，それらはそれぞれA，Bに逆変換ができる（付録２．３参照）．

２．４　機体の安定化（文献2-3）

機体の安定化制御は，離散時間最適レギュレータにより行う．最適制御則は以下のように求められる．離散時間システム

$$x(k+1)=A_d x(k)+B_d u(k) \tag{2.4-1}$$
$$y(k)=C x(k) \tag{2.4-2}$$

において，次の評価関数

$$J=\sum_{k=1}^{\infty}[y(k)^T Q y(k)+u(k)^T R u(k)] \tag{2.4-3}$$

（$Q \geqq 0$，$R>0$）を最小にする最適制御入力

$$u(k)=-K x(k) \tag{2.4-4}$$

を求める．最適フィードバックゲインは次のように与えられる．

$$K=(B_d^T S B_d+R)^{-1} B_d^T S A_d \tag{2.4-5}$$

ここで，Sは次の代数Riccati 方程式の解である．

$$S=A_d^T S A_d-A_d^T S B_d (B_d^T S B_d+R)^{-1} B_d^T S A_d+C^T Q C \tag{2.4-6}$$

２．５　同定と制御の手順

次のように同定及び制御を行い機体を安定化させる．

①操舵系，センサーのＦＤＩ（故障の検出・同定）
②推力制御に切り替えA_d，B_dを同定する．
③フィードバックゲインを求める．
④推力によるフィードバック制御を行い機体を安定化する．

２．６　計算機シミュレーション

2.6.1 機体諸元および飛行条件

シミュレーションでは制御対象としてBoeing 747型機を想定する．機体諸元，安定微係数等は文献2-4,2-5を参照した．また，破損後の垂直尾翼の揚力傾斜は煙風洞における実験から得た（付録Ａ２．１参照）．

おもな機体諸元

離陸全備重量（ＴＯＧＷ，４０％fuel）：$W=2.89\times10^5$kg
慣性モーメント　I_x：2.46×10^7kg・m^2
　　I_y：4.47×10^7kg・m^2
　　I_z：6.71×10^7kg・m^2
慣性乗積　I_{xz}：1.31×10^6kg・m^2
最大上昇推力：$7.55\times10^4\times4$kg・m／sec^2
最大巡航推力：$7.10\times10^4\times4$kg・m／sec^2
エンジン時定数：５sec

飛行条件

高度：６，０８０m，速度：．６５Mach（２０５m／sec），
釣合迎角：２．５deg，飛行経路角：０deg

以上の条件におけるA，Bは次のようになる．
縦の場合：

$$A=\begin{bmatrix} -.5343D-2 & 5.104 & -9.800 & .0 & .3448D-5 \\ -.5197D-3 & -.5365 & .0 & .9880 & -.1470D-8 \\ .0 & .0 & .0 & 1.0 & .0 \\ .7266D-4 & -1.226 & .0 & -.6381 & .7541D-7 \\ .0 & .0 & .0 & .0 & -.2 \end{bmatrix}$$

$$B=\begin{bmatrix} .1210 & .0 \\ -.3923D-1 & .0 \\ .0 & .0 \\ -1.686 & .0 \\ .0 & .2 \end{bmatrix}$$

横・方向の場合

$$A=\begin{bmatrix} -.1045 & .0 & -1.0 & .4783D-1 & .0 \\ -2.917 & -.7913 & .3431 & .0 & 5.177D-8 \\ 1.060 & -.2700D-1 & -.2057 & .0 & 2.385D-7 \\ .0 & 1.0 & .0 & .0 & .0 \\ .0 & .0 & .0 & .0 & -.2 \end{bmatrix}$$

$$B=\begin{bmatrix} .0 & .1420D-1 & .0 \\ .2107 & .1840 & .0 \\ .1073D-1 & -.6246 & .0 \\ .0 & .0 & .0 \\ .0 & .0 & .2 \end{bmatrix}$$

巡航時には最大巡航推力の７０％を使っているとする．したがって，使用可能な推力の調整量は次のようになる（文献2-5）．

$$|\delta_{tL}| = |\delta_t + \Delta\delta_t| / 2 \leqq 7.10\times104\times2\times0.3$$
$$= 4.26\times10^4 \quad N$$
$$|\delta_{tR}| = |\delta_t - \Delta\delta_t| / 2 \leqq 4.26\times10^4 \quad N$$

垂直尾翼の故障としてはつぎの４つの場合を考える（図２－１参照）．

a：完全破壊
b：垂直尾翼の前桁より前部が破損
c：垂直尾翼の後ろ桁より後部が破損
d：垂直尾翼の上半分が破損

破損による無次元微係数の変化は表２－１のようになる．ただし，表の数値は正常な機体状態の微係数に対する割合である．これらの微係数の推算法，推算値については付録Ａ２．２を参照されたい．

2.6.2 垂直尾翼破損の影響と推力制御の効果

機体動特性に対する垂直尾翼破損の影響と，最適レギュレータ理論による推力制御の効果を調べる．それぞれの破壊状況に対して機体の自由応答を調べる．さらに，操舵系が全て故障しているとして，推力制御のみで機体運動を整定させる．ただし，ｂの故障についてはヨーダンパが使えるとしたケースも調べている．それにより，故障後にＳＡＳのような自動操縦装置を使い続けることの問題点について検討する．この節のシミュレーションは連続時間系で行い，エンジン動特性は無視する．舵面はノミナルのトリム位置で固着しているとする．また，故障はすでに同定されているとする．

設定条件

i） 初期状態：$u(0)=10$ m/sec, $\alpha(0)=\beta(0)=0.05$ rad, 他はすべて０とする．

ii）最適レギュレータの評価関数

縦の運動についてはピッチレートを推力で整定させるためにつぎの評価関数を設定する．

$$J=\int_0^\infty (q(t)^2+10^{-12}\delta_t(t)^2)\,dt \qquad (2.6\text{-}1)$$

横・方向の運動については，ヨーレートを左右のエンジンの出力差で整定させるために次の評価関数を設定する．

$$J=\int_0^\infty (r(t)^2+10^{-10}\Delta\delta_t(t)^2)\,dt \qquad (2.6\text{-}2)$$

iii)故障ｂに関しては，横・方向の運動に対してヨーダンパの効果も調べる．ヨー角速度に関するレートジャイロの出力信号をr_Gとすると，ヨーダンパの入力δ_{rSAS}はつぎの伝達関数で発生される（文献2-4）．

$$\delta_{rSAS}(s)=\frac{5.05s}{(s+0.368)(s+3.68)}\,r_G(s) \qquad (2.6\text{-}3)$$

シミュレーション結果をまとめると表２－２のようになる．ここで，○は安定な応答を，×は不安定な応答を，△は安定だが推力の使用量が制限をこえる応答を示す．また，表中の－はラダー破損のためヨーダンパが使用できないことを示す．縦運動については故障により安定性は損なわれないが，横・方向運動についてはｃ以外は不安定になっている．故障ｂでは，方向舵が破損していないので，方向舵への油圧パワーが残っていればヨーダンパによって方向運動を安定化できる可能性がある．しかし，計算結果はヨーダンパを使用しても安定性は回復されないことを示している．これに対し，横・方向運動に対して最適レギュレータを構成し推力制御を行うと，いずれの

故障においても少なくとも機体姿勢は安定化できる．

表２－２の代表的な場合の応答を図２－２から図２－９に示す．ここで，y_e，z_eはそれぞれ地面固定座標（初期状態で機体固定座標と一致）のｙ，ｚ軸方向の距離を表す．図２－２，２－３はそれぞれ正常な機体及び尾翼破損ａ～ｄの場合に，縦運動を推力制御で整定させようとしたときの，高度変化と推力の使用状況を示している．図２－４，２－５はそれぞれ正常な機体及び尾翼破損ａ，ｂの場合に，横・方向運動を整定させようとしたときのヨー角速度変化とy_e軸方向への変位を示している．図２－６，２－７は以上の縦，横・方向運動の制御を同時に行ったときに使われる左右エンジンの出力を，それぞれ示している．ａの場合は推力が使用可能な範囲を超えるので示していない．図２－８，２－９は尾翼破損ｂの場合に，横・方向運動を推力制御及びヨーダンパで整定させようとしたときのヨー角速度とy_e軸方向の変位をそれぞれ示している．ここで，ヨーダンパは(2.6-3)式で表されるノミナルのものを用いている．

以上の結果から分かるように垂直尾翼の破損は縦の運動には大きな影響を与えない（図２－２，２－３）が，横・方向の運動には大きな影響を与える（図２－４から２－７）．横・方向の場合でも壊れ方によって応答はかなり違い，特に垂直尾翼前縁が壊れると空力特性が大きく変わるので，運動への影響は大きい．また，図２－８，２－９は垂直尾翼は壊れたがヨーダンパが使えるとした場合（ｂ），これを使い続けると不安定になる可能性があることを示している．ノミナルの制御系は正常な機体に対して設計されているので，故障ｂのように航空機の動特性が大きく変化した場合にノミナルの制御系を用いると，単に制御性能が低下するだけでなく，閉ループ系が不安定化することもありうる．パイロットは通常は正常な航空機を操縦している．その意味でパイロットはノミナルの制御系に相当する．それ故，故障した航空機をパイロットが操縦するときもＰＩＯ(PilotInduced Oscillation)のようにヨーダンパと同様の不安定化を起こすことも考えられる．このように，ヨーダンパにしろパイロットにしろ正常な航空機に対して有効であっても，故障した航空機に対して有効に働くとは限らない．

2.6.3 制御系の再構成例

前節ではＦＤＩがすでに完了しているとして計算を行った．すなわち，故障した航空機の微係数は既知として，最適フィードバックゲインを求めた．しかし，実際にはＦＤＩを行っている間も航空機の状態は変化しているのであるから，ＦＤＩも含めてシミュレーションを行う必要がある．以下では運動方程式のパラメータを同定することによりＦＤＩを行う．そして，2.5 節で示した手順で制御系を再構成する．一般に離散時間型の同定法の方が収束性がよいので，ここでは繰り返し最小自乗法により離散時間の状態方程式のパラメータを推定する．したがって，最適レギュレータも離散時間型のものを用いる．前節の結果から，縦の運動は垂直尾翼が壊れても安定性が保たれるので，本節では横・方向の運動についてのみ検討する．また，エンジン動特性も考慮する．

設定条件

i)初期状態（同定開始時の状態）：$\beta(0)=0.02$ rad，他はすべて０とする．

ii)パラメータの初期推定値は正常な場合の値を使用する．(2.3-2)式から(2.3-4)式における共分散行列Ｐの初期値はdiag$\{10^{30}\}$とする．

iii)評価関数

$$J=\sum_{k=1}^{\infty}\left[r(k)^2+10^{-5}\Delta\delta_{tc}(k)^2\right] \tag{2.6-4}$$

iv)図２－１０のように同定のための試験入力$\Delta\delta_{tc}(k)$を故障発生直後に３秒間矩形波状に入れる．ただし，故障は発生後直ちに検出されるとする．図２－１０において，

$u_{max}=8.52\times10^4$ N

$u_{max}'=-0.98\times u_{max}$

である．

v)サンプリング周期：0.1 sec

vi)エンジン動特性を一次遅れ系で近似する．時定数を５秒とする．

図２－１１から図２－１５はパラメータの同定状況，各状態変数および同定器で得られた推定値の一部を示す．図２－１１は図２－１０の試験入力を入れたときの左右エンジンの出力差とその推定値を示している．図２－１２は微係数N_rの同定状況を，そして図２－１３は同定時間内（ｔ＝０～３ sec）のヨー角速度とその推定値を示している．状態変数の推定値とは推定されたパラメータから得られる状態変数の値（2.3-2)式の$\hat{y}(k)$）である．なお，パラメータは離散時間系のものが推定されるが図では連続時間のパラメータに変換している．変換については付録Ａ２．３を参照されたい．図２－１４，２－１５は同定終了後（ｔ≧３ sec）にパラメータの推定値を用いて最適レギュレータを構成し，フィードバック制御を行った場合の応答を示している．

これらの結果から分かるように，状態の推定値は真の状態によく一致し，パラメータは真値に収束している．同定は十分短い時間（約１秒）で完了し，機体運動が大きく発散する前に制御を開始できている．そして，推力制御により機体運動を整定できている．

２．７　まとめ

本章では垂直尾翼に重大な故障が発生し，かつ操舵系が完全にフェイルした航空機の安定性を回復する問題を考えた．Boeing 747のような多発のエンジンを持つ航空機では左右エンジンの推力差を用いてヨーイング運動の制御を行うことが可能である．即ち，方向舵と推力を機能的に冗長な制御器として利用できる．このことから，推力のみを用いて横・方向運動の安定化制御を試みた．まず，ＦＤＩでは試験入力として矩形波を用い，最小自乗法の共分散行列の初期値を十分大きくとることにより，十分短い時間で故障した航空機の動特性を同定できることが分かった．そして，その同定

結果を用いて推力を制御入力とした最適レギュレータを構成することにより，機体運動を整定できた．ただし，本章では制御対象を線形系とし，シミュレーションでも比較的小さな初期状態を選んだ．従って，ここで得られた結果から直ちに推力のみで機体制御ができるとは言えないが，少なくともその可能性があることは確認された．

垂直尾翼が壊れた場合に，人間が推力操作を行うことによりフゴイド運動とダッチロール運動を同時に整定させることは，熟練したパイロットにとっても難しいと言われている．それ故，ＲＦＣＳを構成することにより自動的に機体運動が整定できれば，航空機の安全性を回復する上で十分意義があると思われる．

最後に，本章で示したＲＦＣＳの問題点に触れておく．一つは，最適レギュレータの設計で常に問題となるのであるが，評価関数の重みの選び方が試行錯誤によらざるを得ないことである．実際の再構成では，そのようなことはできないので大きな問題である．２つ目は，サンプリング周期の選び方である．故障後の動特性のバンド幅などは分からないので，サンプリング周期は簡単には決められない．それ故，小さめに取っておいた方が安全であろう．これら以外に，このＲＦＣＳが線形制御系であるので故障が引き起こす運動の非線形性に対処できない点がある．この問題については３章以下で示されるＲＦＣＳで考慮される．

なお，Boeing 747において垂直尾翼に故障が生じた場合の問題は第５章でも扱う．そこでは，推力に加えて水平安定板を用いると効果的であることが示される．

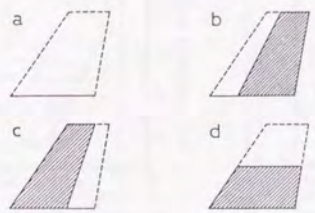

図２－１垂直尾翼の破損状況

表２－１　各破損状況に対する無次元微係数
（正常な場合に対する割合）

	Normal	a	b	c	d
$C_{y\beta}$	1	0	.41	.73	.53
$C_{n\beta}$	1	-.80	.54D-2	.48	.070
C_{lr}	1	.69	.83	.91	.78
C_{nr}	1	-.43	.26	.56	.19
$C_{y\delta_r}$	1	0	.67	0	.23
$C_{l\delta_r}$	1	0	.73	0	.21
$C_{n\delta_r}$	1	0	.67	0	.14

表２－２　各破損状況に対する自由応答，推力制御およびヨーダンパ制御による応答

	縦		横・方向		
	自由応答	推力制御	自由応答	ヨーダンパ制御	推力制御
正常	○	○	○	○	○
a	○	○	×	－	△
b	○	○	×	×	○
c	○	○	○	－	○
d	○	○	×	－	○

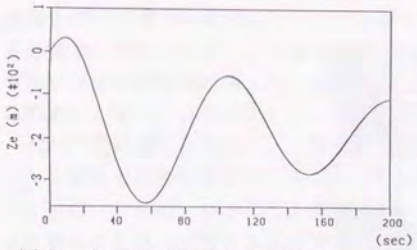

図２－２ 縦運動の推力制御時の高度変化
（正常な機体および尾翼破損 a～d）

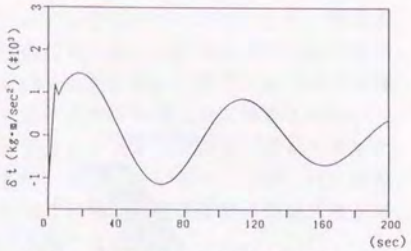

図２－３ 縦運動の推力制御時の推力調節
（正常な機体および尾翼破損 a～d）

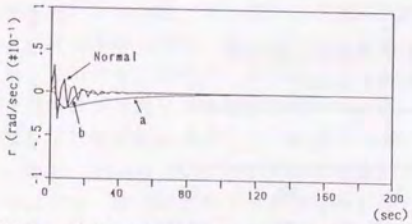

図２－４ 横・方向運動の推力制御時のヨー角速度
（正常な機体および尾翼破損 a，b）

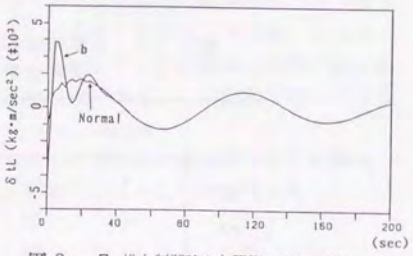

図２－７ 推力制御時の右翼側エンジン出力

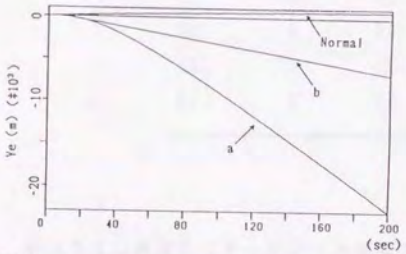

図２－５ 横・方向運動の推力制御時の y_e の変化
（正常な機体および尾翼破損 a，b）

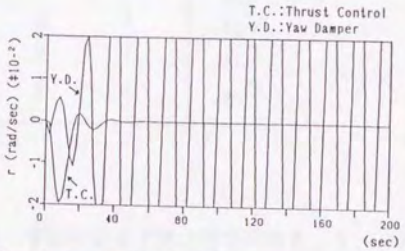

図２－８ 横・方向運動の推力制御時とヨーダンパ制御時の
ヨー角速度（尾翼破損 b）

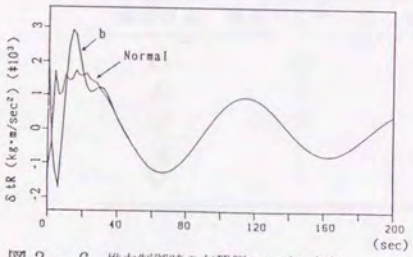

図２－６ 推力制御時の左翼側エンジン出力

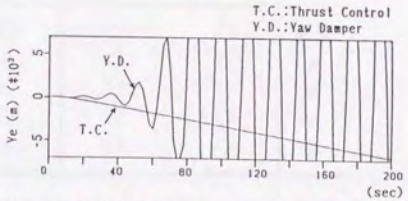

図２－９ 横・方向運動の推力制御時とヨーダンパ制御時の
y_e の変化（尾翼破損 b）

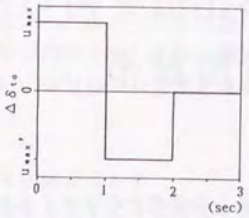

図２－１０ 同定のための試験入力 $\Delta\delta_{te}$

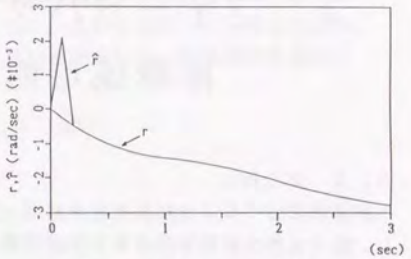

図２－１３ 同定時のヨー角速度 r とその推定値 $\hat{r}$（尾翼破損 b）

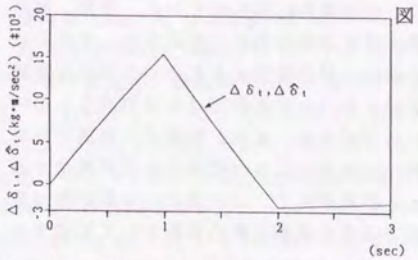

図２－１１ 試験入力 $\Delta\delta_{te}$ に対するエンジン出力 $\Delta\delta_t$

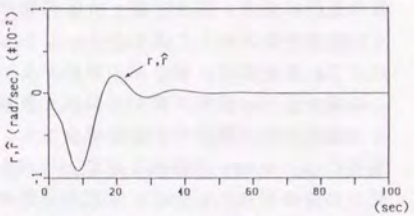

図２－１４ 制御時のヨー角速度 r とその推定値 $\hat{r}$（尾翼破損 b）

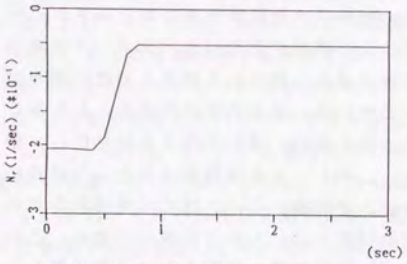

図２－１２ N_r の同定（尾翼破損 b）

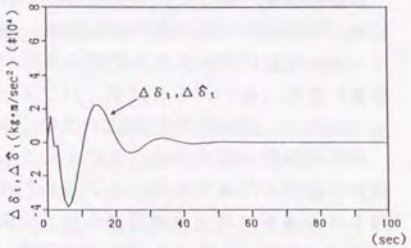

図２－１５ 制御時のエンジン出力 $\Delta\delta_t$ とその推定値 $\Delta\hat{\delta}_t$
（尾翼破損 b）

3．フィードバック線形化法を用いた再構成可能な飛行制御系

3．1　はじめに

第１章で述べたようにＲＦＣＳに関してこれまで多くの研究がなされてきた．しかし，その大部分は制御系を線形制御理論に基づいて設計している．線形制御系はある程度の非線形な環境においても有効に働くことが経験的に知られている．実際，航空機の運動は通常，縦及び横・方向の線形化微小擾乱運動方程式で記述され，それらは互いに非干渉系として扱われる．しかし，主翼の一部が破損するといった重要な故障が生じた場合には，縦と横の運動が互いに干渉しないとする仮定はもはや成立しない．この場合には非線形な運動方程式を直接扱う必要がある．また，故障が大規模ではなく方程式自体は線形で十分な場合でも，故障にともない大きな姿勢の変化が発生する場合には，やはり非線形な運動方程式を考える必要がある．この場合の非線形性は空気力の非線形性から生じ，安定微係数を定数ではなく状態変数の関数として記述する必要がある．

非線形性を考慮した正確な数学モデルを用いる別の理由としては，本システムが適応型の制御系である点が挙げられる．適応型の制御系では数学モデルの正確さがパラメータの推定精度に大きな影響をもつ．モデリング誤差が大きいとパラメータは適当な値に収束しない．その結果，パラメータの推定値から得られる制御入力は不適当なものになり，制御目的を達成できなくなる．ときには，系を不安定化することもある．

非線形運動方程式を扱う方法として，トリム点を求め，運動方程式を線形化し，制御則は線形の制御則を用いる方法も存在する．しかし，大きな故障が発生した場合にはトリム点がノミナルの値から大きく変化する．それ故，この方法には微係数などのパラメータだけでなくトリム点も同定しなければならないという問題点がある．これに対し，航空機を非線形プラントとして扱えば，状態変数や制御入力は平衡状態からの変動量ではなく実際の物理量になるので，同定や制御系設計でトリム点を特に意識することはない．これらの理由により，本論文では制御対象を非線形モデルで記述し，それに基づいて制御則やパラメータ調整則を導くという方法を採用する．

本論文では非線形プラントの制御法として，フィードバック線形化法を採用する．この方法は，まず，非線形運動方程式の中の非線形項を非線形な状態フィードバックにより相殺し，見かけ上線形系にする．そして，この線形系に対して線形制御の手法を適用する．本稿ではこの方法をＲＦＣＳの設計に適用する．航空機の非線形運動方程式の右辺が，通常は状態変数または制御入力の非線形関数と定数係数の線形結合であり，さらに入力に関しては線形であるという点に着目して，非線形モデルのパラメータを同定しながら制御を行う適応飛行制御系を設計する．

以下では，提案するＲＦＣＳの構成を概説した後，その設計法を説明する．即ち，3.2節でまず本方法の特徴を述べ，フィードバック線形化法を用いたＲＦＣＳの基本制御系の設計法を示す．3.3節ではそれを適応制御系に拡張する方法を示しＲＦＣＳの設計を完了する．本手法の有効性を実証するために，ある小型ジェット機の６自由度非線形運動方程式を用いて計算機シミュレーションを実施した．計算結果を最後に示すとともに，本制御系の特徴を考察する．

3．2　基本制御系の設計

本章で提案するＲＦＣＳの構造を図３．１に示す．基本制御系はフィードバック線形化法に基づいて設計される．制御入力は，出力が目標値Y^*に追従するように一般入力発生器(generic input generator) によってつくられる．そこで発生される入力は一般入力(generic input) と呼ばれ，実際の入力とは異なる仮想的な入力である．ＲＦＣＳでは通常，機能的冗長性を生かすために多くの制御器を扱うので入力数が増える．これに対し，一般入力は実際の入力の数よりも少なくなるように定義される．従って，この入力を導入することにより，入力数と出力数を等しくし入力が一意的に決まるようにできる．また，入力の数を減らすことによって同定すべきパラメータの数を減らすことができる．こうして決定された一般入力は，入力分配行列(control distributing matrix)Ｐをかけることにより実際の入力U_{CD}に変換される．

故障の同定は，非線形運動方程式のパラメータを推定することにより行われる．そして，制御則のパラメータの修正は基本的にパラメータの推定値をノミナルのものと置き換えればよい．即ち，このＲＦＣＳは直接方式の適応制御系となる．推定アルゴリズムとしては逐次型の最小自乗法を用いる．パラメータ同定でも実際の入力ではなく一般入力を用いている点に注意されたい．

これらの基本制御系とパラメータ同定器によりＲＦＣＳの基本的な構成はできあがる．しかし，実際にＲＦＣＳを使用することを考えると，入力飽和やアクチュエータ動特性も考慮しなければならない．これらの問題を次のように扱う．

まず，実際の入力が飽和した場合には一般入力U_Gと航空機に実際に作用する入力Ｕとは等価でなくなる．このときの一般入力をパラメータ同定で用いると，実際に航空機に作用したのとは異なる入力を入力とみなしてしまうことになるので，正しい同定ができない．この問題点を解決するために本論文では次のように一般入力の修正を行う．即ち，U_{CD}が入力制限を越えているかどうかを調べ，越えている場合にはある値M_rで全ての入力を割ることによって入力が制限内に入るようにする．M_rはU_{CD}と各入力に課せられた制限値とからオンラインで決定される．

一方，飛行制御系の設計においてアクチュエータ動特性を無視すると，ノミナルの設計で保証されていた性能が大きく低下することがある．ここで提案するＲＦＣＳではアクチュエータ動特性も考慮されている．しかし，パラメータ同定で一般入力を航空機に作用する入力とみなすためには，一般入力に対するアクチュエータが導入されなければならない．一般入力が仮想的な入力であるのと同様にこのアクチュエータも仮想的なものになる．本論文ではそれを「仮想アクチュエータ(imaginary actuator)」と呼ぶ．後で詳しく述べるが，このアクチュエータを実現するには，全てのアクチュエータの時定数が等しくなければならない．それらが異なる場合には，アクチュエー

タの出力をフィードバックして時定数を揃える．そして，その結果構成される閉ループ系を新たにアクチュエータとみなす．この閉ループ系への入力がU_{CD}である．ただし，時定数がもともと全て等しい場合には，U_{CD}はアクチュエータそのものへの入力となる．

以下では，図３－１に示されたＲＦＣＳの設計法について具体的に述べる．

3.2.1 フィードバック線形化制御則

状態方程式と出力方程式

航空機の運動方程式が(3.2-1)式で表されるとする．

$$\dot{X} = A(X) + B(X)U \tag{3.2-1}$$

制御される変数（出力）は(3.2-2)式で表されるとする．

$$Y = CX \tag{3.2-2}$$

ここで，$X \in R^n$，$U \in R^m$，$Y \in R^l$はそれぞれ状態，入力及び出力ベクトルである．$A(X) \in R^n$，$B(X) \in R^{n \times m}$は定数パラメータと状態変数の既知関数との線形結合の形をしているとする．Ｃは定数行列である．

アクチュエータの状態方程式

アクチュエータ動特性は次の１次遅れ系で表されるとする．

$$\dot{U} = \Lambda(-U + U_c) \tag{3.2-3}$$

ここで，$\Lambda = \mathrm{diag}(1/T_i)(i=1,\ldots,m)$であり，$T_i$は各アクチュエータの時定数である．また，$U_c$はアクチュエータへの入力ベクトルである．

入力分配機構及び一般入力

本論文で考えている航空機は多くの舵面をもっているので，$l < m$と仮定してよい．入力数を出力数に減らすために入力分配器と一般入力を導入する．一般入力ベクトルU_G（$\in R^{m'}$）は次式で定義される．

$$U = PU_G \tag{3.2-4}$$

ここで，P（$\in R^{m \times m'}$）は入力分配行列と呼ばれている．Ｐはフルランクをもつとし，ここでは$m' = l$とする．一方，アクチュエータ入力に対しても(3.2-4)式と同様に一般入力が定義される．アクチュエータに対する指令入力をU_{CD}とすると，U_{CD}に対する一般入力U_{CG}は次式で定義される．

$$U_{CD} = PU_{CG} \tag{3.2-5}$$

ここで，U_{CD}は一般にアクチュエータそのものへの入力ではない．それは，後で述べる仮想アクチュエータを構成するために，アクチュエータにフィードバックをかけてアクチュエータ時定数を変えているためである．U_{CD}はその閉ループ系に対する指令入力となる．U_{CD}とU_{CG}の関係については(3.3-9)式及び図３－１を参照されたい．

ＣＤは文献1-52,1-53でも用いられている．そこでは故障が発生した場合に，(1.3-2)式のように制御則は修正されずＣＤを修正することにより故障を補償している．これに対し，本方法ではＢではなくＢＰの形でパラメータが同定される（３．３節参照）ので，Ｐを修正することはできない．即ち，ＣＤは修正されず制御則が修正されることになる．第４章でも触れるが，一般入力を用いる場合には入力分配行列の選び方が制御性能に大きく影響する．それ故，Ｐは一般入力の物理的意味を考えて慎重に決定しなければならない．ところが，Ｐを(1.3-2)式のように変えてしまうとどのように入力分配されるのか分からなくなる．その結果，入力飽和などをおこしやすくなる恐れがある．本方法では上述のようにＰを変えることができないのであるが，できたとしてもそれは必ずしも好ましい方法とは思えない．

制御則

制御則は以下のようにフィードバック線形化法により導かれる．

(3.2-2)式の両辺を微分し，(3.2-1)および(3.2-4)式を用いると次式を得る．

$$\dot{Y} = A'(X) + B_G'(X)U_{CG} \tag{3.2-6}$$

ここで，$A'(X)$（$\in R^{m'}$）$= CA(X)$，$B_G'(X)$（$\in R^{m' \times m'}$）$= CB(X)P$である．

$Y' = \dot{Y}$と定義する．(3.2-6)式をさらに微分し，(3.2-3)式を用いると次式を得る．

$$\dot{Y}' = A_X'(X, U_G) + B_{XG}'(X)U_{CG} \tag{3.2-7a}$$

ここで，

$$A_X'(X, U_G) = C\left\{\left(\frac{\partial A(X)}{\partial X^T} + \sum_{i=1}^{m'} \frac{\partial B_{Gi}(X)}{\partial X^T} U_{Gi}\right) \times [A(X) + B_G(X)U_G] - B_G(X)\Lambda PU_G\right\} \tag{3.2-7b}$$

$$B_{XG}'(X) = CB_G(X)\Lambda P \tag{3.2-7c}$$

ここで，$U_G = [U_{G1},\ldots,U_{Gm'}]^T$，$B_G(X) = B(X)P = [B_{G1}(X),\ldots\ldots,B_{Gm'}(X)]$，$B_{Gi}(X) \in R^n$である．(3.2-7a)から(3.2-7c)式より，出力Ｙを目標出力Y^*に追従させる一般入力に対する制御則が次式で与えられる．

$$U_{CG} = B_{XG}'(X)^{-1}[-A_X'(X, U_G) + G'z + \ddot{Y}^*] \tag{3.2-8}$$

ここで，$z = [Y^T, y'^T]^T$，$y' = \dot{Y} - \dot{Y}^*$，$G' \in R^{m' \times m'}$である．また，$B_{XG}'(X)$は正則であるとする．

(3.2-8)式を(3.2-7a)式に代入しy'の定義を用いると閉ループ系は次式で表される．

$$\dot{z} = Ez \tag{3.2-9}$$

ここで，

$$E=\begin{pmatrix} 0_{m'm'} & I_{m'} \\ G' & \end{pmatrix} \in R^{2m'\times 2m'}$$

$I_{m'}$（$\in R^{m'\times m'}$）は単位行列，$0_{m'm'}$（$\in R^{m'\times m'}$）は零行列である．Eが安定行列（固有値の実数部が全て負である行列）となるようにG'が決められれば，出力Yに対する閉ループ系は漸近安定となる．即ち，$t\to\infty$で$y\to 0$，従って$Y\to Y^*$となる．$B'_{XG}(X)$が正則でない場合は，出力をさらに微分し$\dot{X}$に状態方程式(3.2-1)を代入する．U_Gの正則な係数行列が現れればそれが$B'_{XG}(X)$となる．現れなければ現れるまで微分を繰り返す（文献1-62,1-63参照）．しかし，制御対象によっては出力を微分しないで，近似的に出力を目標出力に追従できる場合がある．次にその方法について述べる．

3.2.2 ピッチ角およびロール角の間接制御

一般的に，2種類の出力Y_1（$\in R$）及びY_2（$\in R$）を考える．Y_1を制御すべき出力とする．そして，Y_1，Y_2に関する微分方程式がそれぞれ次の形をしているとする．

$$\dot{Y}_1=A'_1(X) \tag{3.2-10}$$

$$\dot{Y}_2=A'_2(X)+B'_{G2}(X)\,U_G \tag{3.2-11}$$

ここで (3.2-6)式において$U_G\in R^{m'}$，$[A'_1(X),\ A'_2(X)]^T=A'(X)$，$[0,\ B'_{G2}(X)]^T=B'_G(X)$（$\in R^{1\times m'}$）である．(3.2-10)式には制御入力がないので，これは$B'_{XG}(X)$が特異な場合に相当する．いま，(3.2-10)式がY_2について解析的に解けるとする．即ち，

$$Y_2=a'_1(X,\ \dot{Y}_1) \tag{3.2-12}$$

Y_2に対する目標出力Y_2^*を次のように与える．

$$Y_2^*=a'_1(X,\ G_{Y_1}(Y_1-Y_1^*)+\dot{Y}_1^*) \tag{3.2-13}$$

ここで，G_{Y_1}は適当な負の定数である．$B'_{G2}(X)\neq 0$ならば(3.2-8)式と同じ制御則によりY_2をY_2^*に追従させることが可能である．$Y_2\to Y_2^*$が達成されると，(3.2-10)式のY_2は近似的に(3.2-13)式のY_2^*に置き換えられる．Y_2^*を(3.2-10)式のY_2に代入すると

$$\dot{Y}_1-\dot{Y}_1^*=G_{Y_1}(Y_1-Y_1^*) \tag{3.2-14}$$

となる．$G_{Y_1}<0$より，Y_1はY_1^*に収束する．

Y_1の制御にフィードバック線形化法を直接適用すると$A'_1(X)$の偏微分を計算しなければならなかった．しかし，上述の方法ではその必要がない．ただし，それが有効なのはY_2が(3.2-12)式のように解析的に得られる場合に限られる． アクチュエータ動特性が考慮される場合には，(3.2-7b)式のようにさらに偏微分の計算が要求される．このことは，上述の方法を用いなければY_1を制御するためには2階の偏微分が必要なことを意味する．その結果，制御系の設計はより複雑になる．後で示すシミュレーションではピッチ角，ロール角の制御において$Y_1=\theta$あるいはϕとしてこの方法を用いている（(3.4-10)式から(3.4-15)式を参照）．

3．3 適応制御系への拡張

3.3.1 適応型制御則

$A'_X(X,U_G)$及び$B'_{XG}(X)$のパラメータが故障によりノミナルの値から変化する場合には(3.2-8)式の制御則は次の適応型制御則に置き換えられる．

$$U_{CG}=\hat{B}'_{XG}(X)^{-1}\,[-\hat{A}'_X(X,U_G)+G'z+\ddot{Y}^*] \tag{3.3-1}$$

ここで，$\hat{A}'_X(X,U_G)$および$\hat{B}'_{XG}(X)$のパラメータはそれぞれ$A'_X(X,U_G)$および$B'_{XG}(X)$のパラメータの推定値である．前節で示した基本制御系にこれらのパラメータを推定する機能を付加すればＲＦＣＳが基本的に構成される．

3.3.2 パラメータ同定

(3.3-1)式のパラメータの推定値を求める方法について述べる． これには次のような特徴がある．

第1の特徴はパラメータ同定が離散時間的に行われることである． (3.3-1)式から分かるように，この制御則は連続時間の設計法に基づいている．従って，同定すべきパラメータは連続時間系のものである．しかし，一般に離散時間同定法の方が収束性にすぐれている．このため，本ＲＦＣＳは連続時間パラメータを離散時間的に推定するハイブリッド型適応制御系として構成されている．

第2の特徴は入出力の微分値を用いないで同定を行う点である．出力の微分値を用いることができれば，(3.2-6)式から直接A'（X），B_G'（X）のパラメータが同定できる．しかし，一般に微分値はノイズの影響を受けやすいので，それを用いるのは好ましくない．ここでは，微分値の使用を避けるために，制御対象の状態変数，入出力，及びそれらの関数の値をフィルタに通したものを同定で用いている．

ここで用いられているフィルタは1／（s＋λ）である．sはラプラスまたは微分演算子，λは適当な正の定数である．(3.2-6)式の両辺に1／（s＋λ）をかけると

$$Y-\lambda Y_f=A'_f(X)+B'_{Gf}(X,\ U_G) \tag{3.3-2}$$

となる．ここで，$Y_f=Y/(s+\lambda)$，$A'_f(X)=A'(X)/(s+\lambda)$，$B'_{Gf}(X,\ U_G)=\{B'_G(X)\,U_G\}/(s+\lambda)$である．$A'_f(X)$及び$B'_{Gf}(X,\ U_G)$もA'（X），$B'_G(X)\,U_G$と同様にX，$U_G$の関数と定数パラメータの線形結合の形をしている． このとき，(3.3-2)式の第i要素の方程式は次のようなパラメトリックな形で書ける．

$$\bar{Y}_i = \xi_i^T \eta_i \quad ; i = 1, \ldots, m' \tag{3.3-3}$$

ここで，$\bar{Y}_i = Y_i - \lambda Y_{fi}$である．また，$\xi_i$はフィルタリングされたX，Uの関数から構成される．η_iは定数のパラメータから成っているベクトルである．従って，$A'_f(X)$と$A'(X)$，$B'_{Gf}(X, U_G)$と$B_G(X)$のパラメータは同一であり，また(3.3-3)式において，Y_iとξ_iは入手可能なので繰り返し最小自乗法を用いれば逐次的にη_iが同定される．こうして$A'(X)$，$B'_G(X)$のパラメータが同定される．

第３の特徴は，(3.3-1)式の$A'_X(X, U_G)$および$B'_{XG}(X)$のパラメータを直接推定しないことである．勿論，(3.2-7)式に基づいてそれらのパラメータを推定することはできる．しかし，(3.2-6)式と(3.2-7)式を比べれば分かるようにアクチュエータ動特性を考慮すると，それを考慮しない場合に比べてパラメータ数が増える．このため，(3.2-6)式に基づいて$A'(X)$，$B'_G(X)$のパラメータを同定し，その結果を用いて$A'_X(X, U_G)$及び$B'_{XG}(X)$のパラメータを計算するという方法をとっている．

同定誤差と追従誤差の収束性

次に同定誤差εと追従誤差ｙの収束性を示そう．εは次式によって定義される．

$$\varepsilon = Y - \lambda Y_f - \hat{A}'_f(X) - \hat{B}'_{Gf}(X, U_G) \tag{3.3-4}$$

ここで，$\hat{A}'_f(X)$，$\hat{B}'_{Gf}(X, U_G)$はそれぞれ$A'_f(X)$，$B'_{Gf}(X, U_G)$の定数パラメータをその推定値に置き換えたものとして定義される．ε'を次のように定義する．

$$\varepsilon' = \ddot{Y} - \hat{A}'_X(X, U_G) - \hat{B}'_{XG}(X)\, U_{CG} \tag{3.3-5}$$

ε'を用いると，(3.2-8)，(3.2-9)式より次式を得る．

$$z = (s I_{2m'} - E)^{-1} [0_{m'}^T \ \varepsilon'^T]^T \tag{3.3-6}$$

ここで，$0_{m'} = [0, \ldots, 0]^T \in R^{m'}$である．状態方程式（(3.2-1)式，ただし，入力は一般入力）のパラメータが収束してくると，ε'はｓ（ｓ＋λ）εに近づく．即ち，

$$\varepsilon' \to s\,[\dot{Y} - \hat{A}'(X) - \hat{B}'_G(X)\, U_G] = s\,(s + \lambda)\,\varepsilon \tag{3.3-7}$$

(3.3-6)式および(3.3-7)式より，εが０に収束すれば，$z \to 0$，従って$y \to 0$となる．(3.3-7)式のε'とεの関係については付録Ａ３．２を参照されたい．なお，アクチュエータ動特性を無視すれば，ｙとεのより明確な関係が得られる．そして，εが０に収束しさえすればｙが０に収束することが示せる（付録Ａ３．３参照）．

パラメータ同定におけるＣＤの役割

ＣＤはパラメータ同定で重要な役割を果たす．ＣＤを用いることの一つの利点は$\hat{B}'_G(X)$のパラメータの数がＢ（Ｘ）のパラメータ数よりも小さいことである．実際，シミュレーションにおける航空機では，ＣＤを用いた場合にはパラメータ数が１３９であるのに対し，ＣＤを用いなければ１８７にもなる．このように，ＣＤを用いると

同定すべきパラメータの数が減少する．その結果，同定に要する計算量や時間が減少し，より簡単にそして迅速に再構成が達成されることになる．

ＣＤを用いるもう一つの利点として，縦と横・方向が分離した運動において，U_Gの方がＵより大きな独立性をもちうることが挙げられる．例えば内側昇降舵や外側昇降舵のように分割された舵面は，通常の飛行状態では同じような動きをする．しかも昇降舵は左右で同じ動きをする．従って，分割された各舵面を独立な入力と考えていると，同定信号としては同じような値が入るためそれぞれの舵面の効きを同定することが難しくなる．この場合は４枚の昇降舵を１つの入力（一般入力）として扱えばそのような問題はなくなる．結果的には，個々の舵面の効きは同定されないが，４つの昇降舵全体としての効きが同定される．航空機のピッチングを制御するためにはそれが分かれば十分である．従って，制御則は一般入力に対して求め，それによって決定される入力を４つの舵面に分配する．補助翼や方向舵等についても同様である．

仮想アクチュエータ

航空機への実際の入力Ｕはアクチュエータの出力であり，これは観測できる．しかし，Ｕが観測されても一般入力U_Gは(3.2-4)式からは決定できない．それは一般入力の数ｍ'が実際の入力の数ｍよりも小さいからである．一方，U_GはＡ'（Ｘ）およびB_G'（Ｘ）のパラメータを同定するために必要である．この問題を解決するために次の仮想アクチュエータを導入する．これは一般入力に対する仮想的なアクチュエータであり，その出力は常に(3.2-4)式を満たすように構成される．

まず，全てのアクチュエータの時定数が等しいとしよう．つまり，Λ＝diag（１／Ｔ）とする．Ｐはフルランクであるので(3.2-3)式は次のように書ける．

$$\dot{U}_G = (-U_G + U_{CG}) / T \tag{3.3-8}$$

この方程式が仮想アクチュエータの動特性を表す．

アクチュエータの時定数が異なる場合には，ＵをU_Cにフィードバックしてアクチュエータに関する閉ループ系の時定数が－１／Ｔとなるようにしなければならない．これは次のアクチュエータ入力を用いることにより達成される．

$$U_C = (I_m - \Lambda^{-1}/T)\, U + (\Lambda^{-1}/T)\, U_{CD} \tag{3.3-9}$$

ここで，$I_m \in R^{m \times m}$は単位行列である．U_{CD}を新しいアクチュエータ入力，そして閉ループ系を新しいアクチュエータとみなせばその動特性は(3.3-8)式で表される．Ｔは最大の時定数に等しいか，それより大きく与えられる．このとき，U_{CD}が舵面の可動範囲に関する拘束を破らなければ，Ｕがそれを破ることはない．従って，その拘束はU_{CD}に対してのみ課せばよい．

3.3.3 入力飽和と一般入力の修正

利用可能な入力の範囲は実際には限られている．舵面には操舵可能範囲があり，推力は最大のエンジン出力の範囲内でしか利用できない．それ故，制御系が発生した入力U_{CD}がその制限を超えた場合には，実際の入力Ｕは制限値で飽和する可能性がある．もし飽和すればコントローラが指令した入力とは違った入力が制御対象に与えられる

ことになる．一方，パラメータ同定では実際に制御対象に作用した入力を用いなければならない．本ＲＦＣＳでは同定入力として一般入力を用いているので，入力が飽和したときにはそれに対応する一般入力を求めなければならない．しかし，上で述べたように，飽和した実際の入力ＵからU_Gを求めることはできない．このため航空機に与えられた入力と等価な一般入力が得られない．この問題に対処するために，以下ではＵが飽和したときにＵとU_Gが矛盾しない入力の決定法を示す．簡単のために，ここではアクチュエータ動特性を無視して考える．

Ｕが飽和していなければ，$U=U_{CD}$である．しかし，飽和していれば一般に$U \neq U_{CD}$であり，その結果$U \neq PU_G$となる．このように一般入力は実際の飽和入力に必ずしも対応しないので，一般入力を用いて未知パラメータを同定するには，(3.2-4)式を満たすU_Gを求めなければならない．しかし，$m>m'$であるから一般に(3.2-4)式からＰの擬似逆行列によりU_Gを求めることはできない．この問題に対し次のように対処する．

Ｕの第ｉ要素U_iに次の不等式で表される制限が課せられているとする．

$$-U_{imax} \leqq U_i \leqq U_{imax}, \quad i=1,\ldots,m \tag{3.3-10}$$

ここで，次のようなM_{ri}とM_rを定義する．

$$M_{ri}=|U_i/U_{imax}| \tag{3.3-11}$$

$$M_r=\max(1, M_{r1}, \ldots, M_{rm}) \tag{3.3-12}$$

(3.3-12)式で与えられるM_rでU_Gを割ったものを新たに一般入力U_G'（$=U_G/M_r$）とする．このときU_G'により決定される実際の入力$U'=PU_G'$は必ず(3.3-10)式の制限内にある．この修正法をU_{CG}に対して行えば，アクチュエータ動特性が無視できない場合でも一般入力と実際の入力が矛盾することはない．この方法を簡単に述べると，制限値から最も大きく飛び出した入力がその制限値に一致するように全ての入力を一様に減少させるということになる．しかし，これは利用できるはずのコントロールパワーを，単に一般入力と実際の入力の矛盾を除くために無駄にしていることにもなる．この点についてはシミュレーションで定量的に検討する．

同様の問題は舵面が固着した場合にも生じる．これについては付録Ａ３．４で述べる．

3.3.4 設計手順のまとめ

最終的な設計手順は以下のようになる．

①制御したい出力Ｙを選び，その目標出力Ｙ*を与える．

②一般入力U_Gの数を決定し，入力分配行列Ｐを定義する．

③アクチュエータ時定数が異なる場合には，(3.3-9)式で与えられるアクチュエータ入力を用いる．その結果，閉ループ系の時定数はアクチュエータ時定数の中の最大値になる．

④(3.3-8)式により仮想アクチュエータを定義する．その時定数は③で構成した閉ループ系の時定数と同じにする．

⑤(3.3-2)式を同定の数学モデルとしてパラメータ同定器を構成する．同定信号としては，状態変数と一般入力を用いる．そして，Ａ（Ｘ），B_G（Ｘ），Ａ'（Ｘ）及びB_G'（Ｘ）のパラメータが逐次型の最小自乗アルゴリズムにより同定される．

⑥実際のアクチュエータへの制御入力は(3.3-9)式，(3.2-5)式，及び(3.3-1)式により与えられる．(3.3-1)式のパラメータはＡ（Ｘ），B_G（Ｘ），Ａ'（Ｘ）及びB_G'（Ｘ）のパラメータの推定値を用いて(3.2-7b)，(3.2-7c)式より計算される．

⑦もしU_{CD}のいずれかの要素が入力制限を超えれば，一般入力U_{CG}を$U_{CG}'=U_{CG}/M_r$に修正する．その結果，$U_{CD}'=PU_{CG}'$は入力制限内の値になる．M_rは(3.3-11)，(3.3-12)式によりオンラインで決定される．

３．４　シミュレーション

提案するＲＦＣＳの性能と特性を調べるために，７つの制御舵面（左右の昇降舵，補助翼，カナード，及び１つの方向舵）をもつ小型ジェット機の６自由度非線形運動方程式を用いて計算機シミュレーションを行った．故障がない場合も含め４ケースについて計算した．そして，同定誤差および追従誤差の収束性や一般入力を修正したことによる実際の入力への影響についても調べた．

3.4.1 航空機の数学モデル

航空機の剛体運動は以下の６自由度非線形運動方程式と２つの力学関係式で記述されるとする（文献3-1）．主な記号については図３－２を参照されたい．

$$\dot{u}=-g\sin\theta+vr-wq+(\rho V^2S/2M_a)(C_X+C_{X\delta}^T\delta)+T_h/M_a \tag{3.4-1}$$

$$\dot{w}=g\cos\theta\cos\phi+uq-vp+(\rho V^2S/2M_a)(C_Z+C_{Z\delta}^T\delta) \tag{3.4-2}$$

$$\dot{q}=[(I_Z-I_X)/I_Y]pr+(\rho V^2Sc/2I_Y)(C_m+C_{m\delta}^T\delta)+(\rho VSc^2/4I_Y)C_{mq}q \tag{3.4-3}$$

$$\dot{\theta}=q\cos\phi-r\sin\phi \tag{3.4-4}$$

$$\dot{v}=g\cos\theta\sin\phi+wp-ur+(\rho V^2S/2M_a)(C_Y+C_{Y\delta}^T\delta+C_{Y\delta_r}\delta_r)+(\rho VSb/4M_a)(C_{Y_p}p+C_{Y_r}r) \tag{3.4-5}$$

$$\dot{r}=[(I_X-I_Y)/I_Z]pq+(\rho V^2Sb/2I_Z)(C_n+C_{n\delta}^T\delta+C_{n\delta_r}\delta_r)+(\rho VSb^2/4I_Z)(C_{n_p}p+C_{n_r}r) \tag{3.4-6}$$

$$\dot{p} = [(I_Y - I_Z)/I_X]\, q\, r + (\rho V^2 S b / 2 I_X)\,(C_l + C_{l\delta}^T \delta + C_{l\delta_r}\delta_r) + (\rho V S b^2 / 4 I_X)\,(C_{l_p} p + C_{l_r} r) \qquad (3.4\text{-}7)$$

$$\dot{\phi} = p + (q \sin\phi + r \cos\phi) \tan\theta \qquad (3.4\text{-}8)$$

アクチュエータ動特性は次の１次遅れ系で表されるとする．

$$\dot{\delta} = \Lambda\,(-\delta + \delta_C) \qquad (3.4\text{-}9)$$

ここで，これらのベクトルは次のように定義される．

$\delta = [\bar{\delta},\ \delta_r]^T$

$\delta_C = [\bar{\delta}_C,\ \delta_{rC}]^T$

$\bar{\delta} = [\delta_{hL}, \delta_{hR}, \delta_{aL}, \delta_{aR}, \delta_{cL}, \delta_{cR}]^T$：舵面の操舵角ベクトル

$\bar{\delta}_C = [\delta_{hLC}, \delta_{hRC}, \delta_{aLC}, \delta_{aRC}, \delta_{cLC}, \delta_{cRC}]^T$：
各舵面のアクチュエータに対する指令入力ベクトル

$C_{i\delta} = [C_{i\delta_{hL}}, C_{i\delta_{hR}}, C_{i\delta_{aL}}, C_{i\delta_{aR}}, C_{i\delta_{cL}}, C_{i\delta_{cR}}]^T$：
無次元微係数（ｉ＝Ｘ，Ｚ，ｍ，Ｙ，ｎ，ｌ）

無次元微係数は迎角αと横滑り角βの関数である（付録Ａ３．５参照）．また，α，β，Ｖとｕ，ｖ，ｗの間には次の関係がある．

$V = (u^2 + v^2 + w^2)^{1/2}$

$w = u \tan\alpha$

$v = V \sin\beta$

添字の意味は以下の通りである．

Ｘ，Ｙ，Ｚ：それぞれ機体軸の前向き，横向き，下向きの座標軸に関する値
ｌ，ｍ，ｎ：それぞれロール，ピッチ，ヨーに関する値
ｈ：水平尾翼（全可動尾翼）
ａ：補助翼
ｃ：カナード
ｒ：方向舵
Ｌ：左側舵面，Ｒ：右側舵面

(3.4-1)式から(3.4-9)式において状態変数，制御変数等は次のように定義される．

状態変数：$X = [u,\ \alpha,\ q,\ \theta,\ v,\ r,\ p,\ \phi,\ \delta]^T$
制御変数：$U_C = [\delta_C^T,\ T_h]^T$

ｕ：ｘ軸方向速度（m/sec）
ｗ：Ｚ軸方向速度(m/sec)
ｑ：ピッチ角速度(rad/sec)
θ：ピッチ角(rad)
ｖ：Ｙ軸方向速度(m/sec)
ｒ：ヨー角速度(rad/sec)
ｐ：ロール角速度(rad/sec)
ϕ：ロール角(rad)
T_h：推力(N)

シミュレーションでは推力は一定であるとする．また，ＲＦＣＳの設計ではｗの代わりに迎角α(rad)が使われている．ここで，$\alpha = \tan^{-1}(w/u)$である．

機体形状に関するパラメータとその値は次の通りである．

Ｓ　：主翼面積＝４８．７７(m^2)
M_a：機体質量＝２２，６８５(kg)
ｃ　：平均空力翼弦長＝２．７６(m)
ｂ　：主翼幅＝１９．２(m)
I_X：Ｘ軸回りの慣性モーメント＝６７，７９０(kgm^2)
I_Y：Ｙ軸回りの慣性モーメント＝４２７，３４８(kgm^2)
I_Z：Ｚ軸回りの慣性モーメント＝４７６，５６４(kgm^2)
慣性乗積（I_{XZ}）＝０(kgm^2)

ノミナルの飛行条件に関するパラメータとその値は次の通りである．

ｇ：重力加速度＝９．８(m/sec^2)
ρ：大気密度＝０．５４９５(kg/m^3)
Ｖ：対気速度＝２２０(m/sec)＝．７１(Mach)
飛行高度＝７，６００(m)

文献3-1ではC_X，C_m等の無次元空気力，モーメントやC_{mq}，$C_{X\delta_{hL}}$等の無次元微係数はαおよびβの関数として表で与えられている．シミュレーションではその表の$-10 \leq \alpha \leq 30$(deg)，$-20 \leq \beta \leq 20$(deg)におけるデータをαまたはβの多項式で近似している．C_m，C_Y，C_nおよびC_lは，αに関しては８次，βに関しては４次の多項式で表されている．その他のパラメータはαに関する８次の多項式で表されている．同定モデルは(3.4-1)から(3.4-3)式及び(3.4-5)から(3.4-7)式と同じ形としている．その無次元空気力等もα，βの多項式としているが次数はともに２次である．詳しく付録Ａ３．５を参照されたい．

3.4.2 設計パラメータ

入力分配行列と一般入力を次のように与える．

$$P=\begin{bmatrix} P_{h1} & 0 & P_{a1} & 0 \\ P_{h1} & 0 & -P_{a1} & 0 \\ P_{h2} & 0 & P_{a2} & 0 \\ P_{h2} & 0 & -P_{a2} & 0 \\ 0 & P_{hc} & P_{ac} & 0 \\ 0 & P_{hc} & -P_{ac} & 0 \\ 0 & 0 & 0 & P_r \end{bmatrix}$$

$$U_G=[\delta_{lng1},\delta_{lng2},\delta_{lat},\delta_{dir}]^T$$

上の定義から分かるように，$\delta_r=\delta_{dir}$であり，δ_{Gr}は他の実際の入力から独立である．従って，一般入力の修正はδ_{Gr}に対しては行わない．Pの選び方については4．2節を参照されたい．

出力Yと目標出力Y*をそれぞれ$Y=[\alpha,\ \theta,\ v,\ \phi]^T$，$Y^*=[\alpha^*,\ \theta^*,\ v^*,\ \phi^*]^T$とする．

Yの中のθとϕに関する状態方程式（(3.4-4)式及び(3.4-8)式）は制御入力を含んでいないので$B_G'(X)$は特異になる．このためθとϕは(3.2-10)式から(3.2-14)式で記述された方法で制御されなければならない．それらの式においてそれぞれ$Y_1=\theta$またはϕ，$Y_2=q$またはpである．目標出力Y_2^*は次式で与えられる．

$$q^*=\{r\sin\phi+G_q(\theta-\theta^*)+\dot{\theta}^*\}/\cos\phi \qquad (3.4\text{-}10)$$

$$p^*=-(q\sin\phi+r\cos\phi)\tan\theta+G_p(\phi-\phi^*)+\dot{\phi}^* \qquad (3.4\text{-}11)$$

ここで，G_qおよびG_pは負の定数である．(3.4-4)式と(3.4-10)式より次式が得られる．

$$\theta-\theta^*=(q-q^*)\cos\phi/(s-G_q) \qquad (3.4\text{-}12)$$

同様に，(3.4-8)式と(3.4-11)式より次式が得られる．

$$\phi-\phi^*=(p-p^*)/(s-G_p) \qquad (3.4\text{-}13)$$

ここで，$s=d/dt$である．まず(3.3-1)式の制御則により$q\to q^*$（$p\to p^*$）は保証される．その結果，$G_q<0$（$G_p<0$）とすれば$\theta\to\theta^*$（$\phi\to\phi^*$）となる．

(3.3-1)式において$\dot{q}^*$，$\ddot{q}^*$，$\dot{p}^*$，$\ddot{p}^*$が必要になる．ここでは次のように与える．

$$\dot{Y}_2^*=G_{Y2}(Y_2-\dot{Y}_1^*) \qquad (3.4\text{-}14)$$

$$\ddot{Y}_2^*=G_{Y2}(\dot{Y}_2-\ddot{Y}_1^*) \qquad (3.4\text{-}15)$$

上式で$G_{Y2}=G_q$またはG_pであり，Y_1，Y_2は上で定義した通りである．(3.4-10)式あるいは(3.4-11)式を微分すると厳密には(3.4-14)式あるいは(3.4-15)式のようにはならない．しかし，r，ϕ，θを微小とみなせばこれらの式が得られる．なお，$\ddot{\theta}^*$，$\ddot{\phi}^*$の項については，それらを加えると入力が飽和しやすくなるので除いている．

シミュレーションにおけるパラメータや条件は以下の通りである．

舵角制限：δ_{imax}（(3.3-10)式のU_{imax}）$=.4$(rad)(i=hL,hR)，
$.2618$(rad)(i=aL,aR)，$.3$(rad)(i=cL,cR)，$.5236$(rad)(i=r)

推力（一定）：$T_h=.4463\times10^5$(N)

パラメータ同定の周期：.05(sec)

制御パラメータ更新の周期：.05(sec)

初期条件：$X(0)=[230.,.05,.05,.05,5.,0,0,0]^T$

(3.2-8)式のG'：$G'=[G_1',\ G_2']$とおくと，
$G_1'=\mathrm{diag}\{-25,-100,-25,-100\}$，
$G_2'=\mathrm{diag}\{-10,-20,-10,-20\}$

(3.2-13)式のG_{Y1}：$G_q=-10$，$G_p=-30$

(3.3-2)式で用いられたフィルター：1/(s+30)

パラメータの初期推定値：つぎのトリム点におけるノミナルの値

トリム点：$X_{tr}=[220,.1,0,.1,0,0,0,0]^T$
$U_{tr}=[-.03046,-.03046,\ -.01994,-.01994,\ -.003592,-.003592,\ 0,\ .4436\times10^5]^T$

入力分配行列の非零要素：$P_{h1}=.4$，$P_{a1}=.4$，$P_{h2}=.2618$，$P_{a2}=.2618$，
$P_{hc}=-.3$，$P_{ac}=.3$，$P_r=.5236$

アクチュエータ時定数：$\Lambda^{-1}=\mathrm{diag}\{.05,.05,.04,.04,.033,.033,.05\}$(sec)

ノミナルのトリム点で線形化されたシステムの固有値
縦　　：$-.00547\pm j.04549$，$-.5828\pm j2.275$
横・方向：$-.2703\pm j2.166$，-1.033，$-.01393$

目標出力：$Y^*(t)=Y^*+(Y(0)-Y^*)\exp(at)$
ここで，$a=-2$，$Y^*=[.1,.1,.0,.0]^T$

3.4.3 シミュレーション結果

以上の条件の下で，次の4つの場合について入出力の時間応答を調べた．

ケース1）正常な場合

ケース2）左水平尾翼が-.2(rad)で固着し，同時に方向舵の効きが50％減少する（故障a）．

ケース3）左水平尾翼と方向舵が-.2(rad)で固着し，同時に左の補助翼と右のカナードの効きが50％減少する（故障b）．

ケース4）故障bで再構成を行わなかった場合．即ち，制御パラメータを修正しなかった場合

故障はt＝0で発生するとする．それぞれの場合に対するシミュレーション結果は以下の通りである．

ケース1）図3－3は出力の時間応答を示している．追従誤差は十分小さく定常状態では0になっている．この結果はノミナルの制御系の性能が望ましいものであることを示している．

ケース２）図３－４の結果から分かるように，出力の目標値への追従性は正常な場合に比べて多少悪くなっているが，約３秒で水平定常飛行に回復している．制御舵面の応答が通常の舵の動きと異なっている点に注意されたい．補助翼は通常は左右で逆向きに操舵されるが，ここではともに後縁下げにきられている．また，カナードは左右逆向きにきられているが，その大きさは左右で異なっている．これらのことは各舵が通常とは異なる動きをすることにより，故障によって失われた舵の機能を補償していることを意味する．つまり，機能的冗長性が生かされていることを示している．

ケース３）これはより厳しい故障の場合である．図３－５より，迎角とピッチ角の過渡応答はかなり良好であることが分かる．そして，定常状態での追従誤差はほとんど０になっている．これは航空機がほぼ水平飛行をしていることを示している．しかし，Ｙ軸方向速度ｖとロール角φの応答は減衰傾向にはあるものの，残っている舵面ではそれらの出力を目標値に追従させることはできない．その大きな原因はヨーコントロールの支配的な制御器である方向舵が固着してしまっていることである．このことは方向舵の固着が横・方向制御に重大な影響を及ぼし，他の舵面ではその機能を十分には補えないことを意味している．この故障を補償するためには，航空機は上下に分割された方向舵や２枚の垂直尾翼，あるいは垂直カナードをもつことが必要であろう．

ケース４）図３－６の出力の応答は航空機が激しいピッチングとローリングをしていることを示している．このように故障ｂが発生した場合に再構成を行わなければ航空機の運動は全く制御できない．この結果をケース３）の結果と比べると，再構成が航空機の安全に対して有効であることが分かる．

次に追従誤差ｙと同定誤差εの収束性を調べる．図３－７，図３－８にｙとεの時間履歴を示す．図３－７は故障ａの場合，図３－８は故障ｂの場合のものである．これらの図でεは５０倍に拡大されている．(3.3-6)式より$y=\varepsilon'/(s+5)^2$となる．εとε'の，式の上での関係は必ずしも明確ではないが，(3.3-7)式よりεが０に収束すればε'が０に収束することが分かる．その結果，上式よりｙが０に収束することになる．図３－７，図３－８はこのことを示している．つまり，ｙは遅れを伴いながらεと同じように変化し，εが収束した後は指数的に０に収束している（付録Ａ３．２参照）．ただし，故障ｂの場合のｖとｐについては，ケース３）で述べた理由により追従誤差は収束していない．

先に指摘したように，入力が飽和したときに一般入力を修正することは利用できるコントロールパワーを無駄にしていることになる．そこで，入力が飽和したときに一般入力を修正しなければ使えたであろうコントロールパワーがどれくらいになるか調べる．図３－９，図３－１０は，それぞれ故障ａ，故障ｂの場合の(3.3-12)式のM_rと次式で定義されるＪの時間履歴を示している．

$$J=\sum_{i}|\delta_i|/|\delta_{imax}| \qquad (3.4\text{-}16)$$

ここで，i=hR,aL,aR,cL,cRである．M_rとＪの定義から次のことが分かる．もしM_rが１であれば一般入力は修正されない．従って実際の入力も修正されない．一方，M_r

＞１であれば，入力は修正され１≦Ｊ（≦５）となる．M_rが大きくなるほど，入力は大きく減少される．最悪なのはM_rが非常に大きく，かつＪが１に近い場合である．これは，１つの実際の入力が他の入力に比べて制限値を大きく超えたために，他の入力が大きく減少され，利用できるはずのコントロールパワーが利用されなくなってしまうことを意味する．逆に，M_rが大きくてもＪが５に近ければコントロールパワーの損失は小さいと考えられる．故障ａでは，図３－９から分かるように入力はほとんど飽和していないので，この問題はない．これに対し，故障ｂでは図３－１０の結果を見ると，M_rは時刻ｔ＝0.1秒から0.6秒の間，大きな値になっている．一方，同じ時間帯にＪは２から３の値をとっている．このことは利用されていないコントロールパワーがかなり大きいことを示している．しかし，ｔ＝0.6秒以降はM_rは比較的小さく，ｔ＝1.8秒より後はM_r＝１となっている．つまり，入力は飽和していない．コントロールパワーが無駄になっているのは故障発生直後の短い時間ではあるが，そこでは通常大きな入力が要求され入力飽和が生じやすい．そして，故障直後の操舵が航空機を回復する上で重要であることは序論で述べた．従って，この時間帯にコントロールパワーが十分に利用できないのは問題である．しかし，本設計法では一般入力の修正を行わなければ正しい同定ができない．その結果，適切な制御則の修正ができなくなる．実際，一般入力の修正を行わないでシミュレーションを行うと，応答は発散してしまう．このため，多少のコントロールパワーを犠牲にしても，一般入力の修正を行わないわけには行かない．この点については何らかの改善が望まれる．

３．５　まとめ

本章ではフィードバック線形化法に基づいたＲＦＣＳの設計法を提案した．このＲＦＣＳでは舵面のアクチュエータ動特性も考慮されている．そして入力の数を出力の数に減らすために仮想的な入力である一般入力が用いられている．一般入力は入力分配器により実際の入力に変換される．一般入力の導入により同定すべきパラメータの数が減少し，計算機の負担も小さくなる．入力が飽和した際に一般入力と実際の入力の矛盾を除くために一般入力が修正される．この修正は利用できるコントロールパワーを無駄にすることにもなるが，それを行わなければパラメータ同定が正確にできない．その結果，故障によっては航空機の応答が発散することもある．

提案したＲＦＣＳの性能や特性を調べるために，２通りの故障を想定し計算機シミュレーションを行った．故障ｂで横・方向の制御性能が悪くなる点以外は良好な結果が得られた．横・方向の制御性能が劣化したのは主要なヨー制御器である方向舵が固着したためであると考えられる．このことは，ＲＦＣＳが制御器の機能的冗長性を利用しているとは言うものの，少なくともピッチ，ヨー，ロール，そして推力（パワー）に関する支配的な制御器はハードウェア的に冗長でなければならないことを示唆している．

提案したＲＦＣＳは航空機運動方程式のパラメータを推定することにより故障を同定する．それ故，運動方程式のパラメータに影響が現れる故障であれば，基本的にどんなものでも補償できる潜在的能力をもつ．例えば，単一の制御器の故障だけでなく

複数の箇所の同時故障や機体固有の動特性（安定性など）を変えるような故障にも対処できる．勿論，物理的に補償可能なだけの十分なコントロールパワーが残っていることが前提である．

最後に，このＲＦＣＳの問題点をまとめておく．まず，本ＲＦＣＳ特有の問題としてパラメータ同定機構を含めた制御系全体が大規模で複雑である点が挙げられる．このため，オンラインでの多くの計算が必要なことから計算遅れによる性能の低下が生じる可能性もある．これらの点を考慮すると，装置化のためには高速の計算機が必要になり，計算機等のハードウェアに対する要求が厳しくなる．第２に，上述の設計法から分かるように，本ＲＦＣＳではディジタル計算機の使用は不可欠であり，制御則を離散化しなければならない． 特に，3.3.3節で示した入力修正を行うには制御則は離散時間形式であることが必要である． ところが，本来の制御則は(3.3-1)式のように連続時間制御則である．勿論，それをTustin変換などにより離散化することはできるが，近似的なものになる．それ故，制御則は初期の設計から離散時間制御則として求められることが望ましい．第３の問題点として，一般入力を修正することによってコントロールパワーが十分に利用できないことがあるが，これについてはすでに述べた．第４に，提案したＲＦＣＳのような適応型の制御系は一般的にロバスト性に欠けるという問題をもつ．即ち，航空機やアクチュエータ動特性のモデル誤差，観測ノイズ（非定常）外乱などに対するロバスト性が小さく，これらの誤差やノイズ，外乱があると制御性能が大きく低下する可能性があると言われている（文献1-44）．最後に，パラメータ同定の周期，フィルタの時定数，閉ループ系の固有値，入力分配行列などをどのように決定するかという問題がある．同定の周期やフィルタの時定数は航空機の短周期モードなどからある程度の目安はつく．閉ループ系の極もあまり小さく選ぶと，大きな入力が要求されるので入力飽和が生じやすくなる．また，安定性に対するロバスト性も小さくなる．このようなことを考慮して，シミュレーションでは設計パラメータは試行錯誤的に決定された．従って，これらの選定に関して厳密な解析を行ってはいないし，明確な指針を得ているわけではない．しかし，パラメータの与え方により制御性能が大きく変わるので，実際の設計では十分慎重に決定されなければならない．

これらはいずれも重要な問題ではあるが，本論文ではそれを指摘するにとどめる．なお，制御則の離散化については第４章で扱う．また，入力分配行列の設定に対する一つの方針を同章で示す．

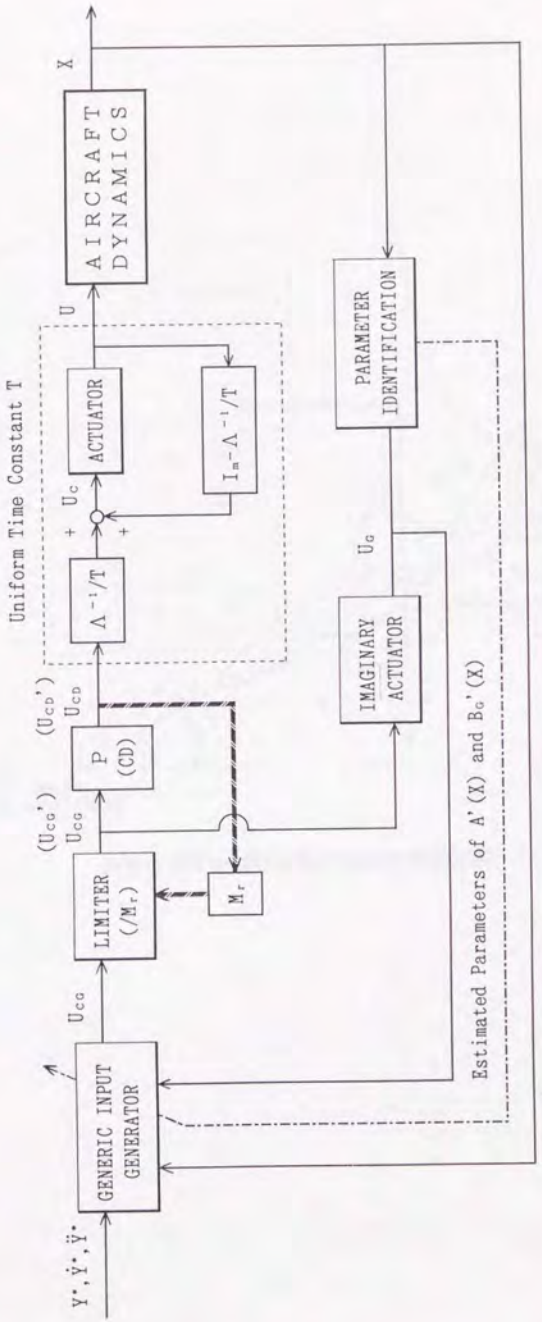

図３－１　ＲＦＣＳのブロック線図

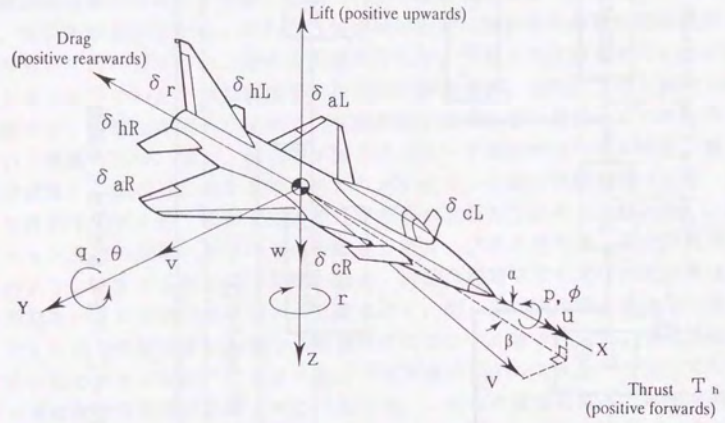

図３－２　機体固定座標と状態及び制御変数の定義

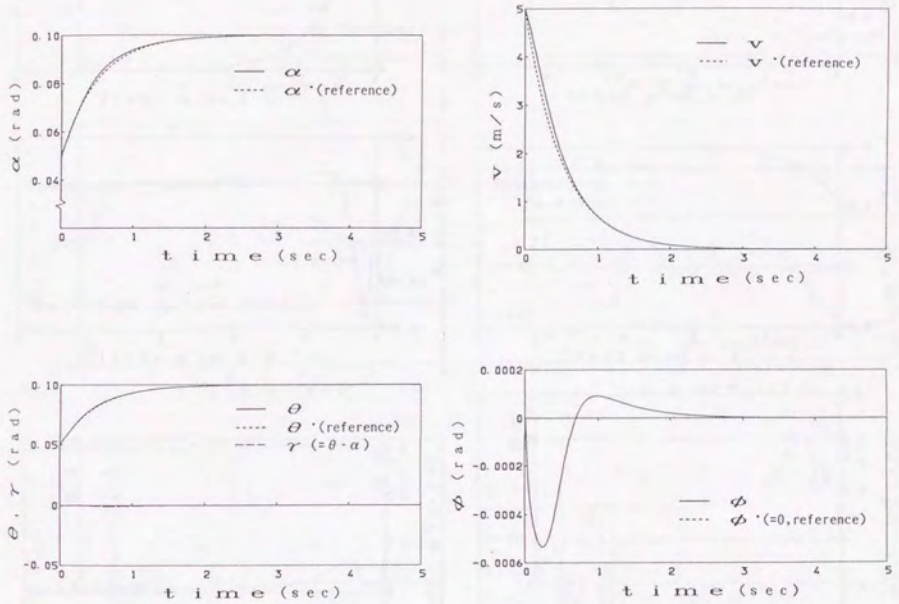

図３－３　正常な場合の出力の時間応答

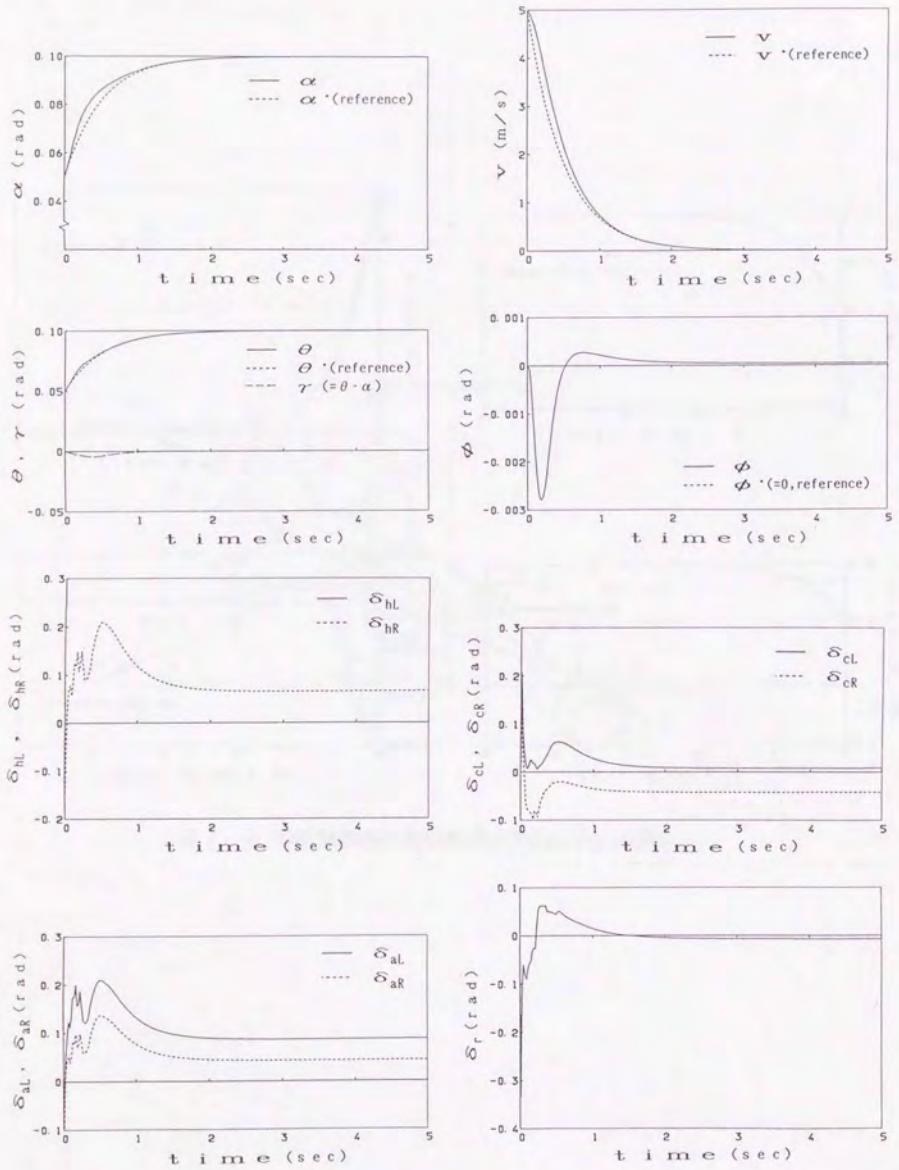

図３－４　故障ａの場合の入出力の時間応答

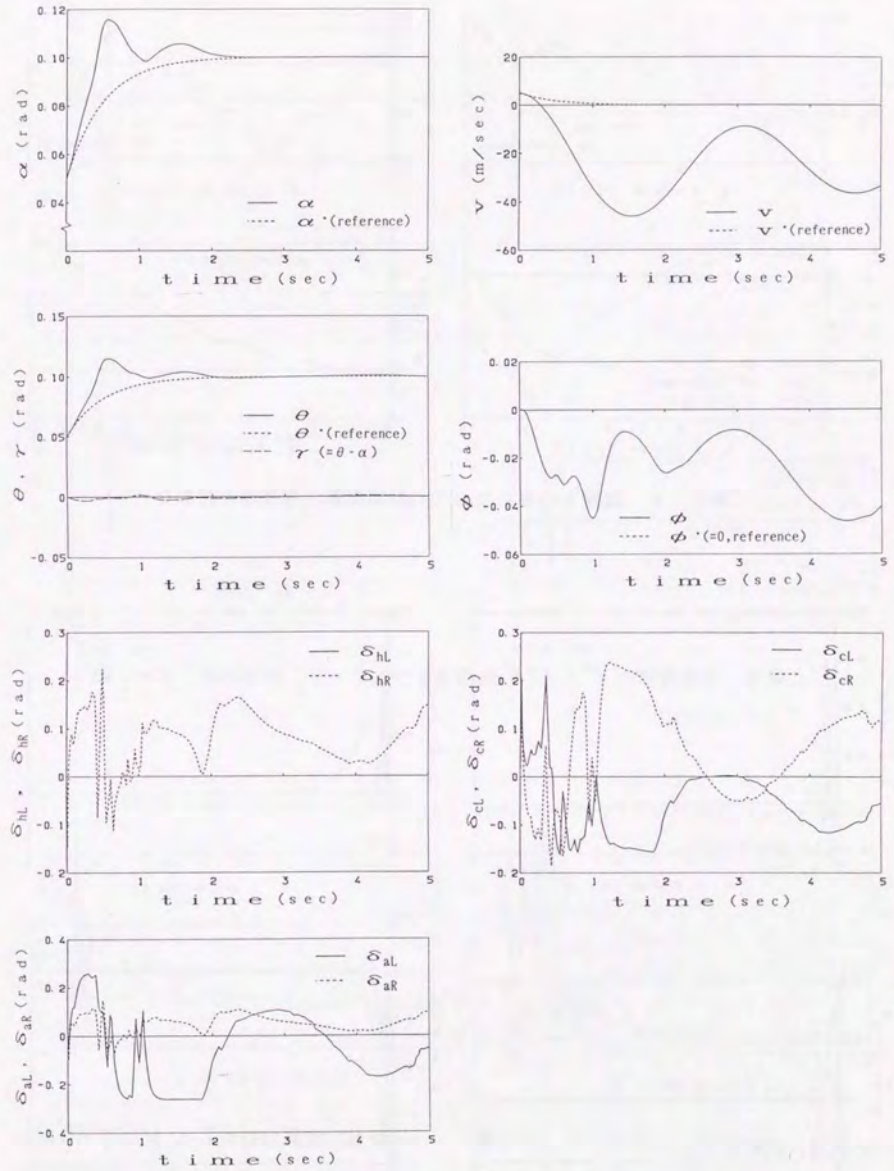

図３－５　故障ｂの場合の入出力の時間応答（再構成を行う）

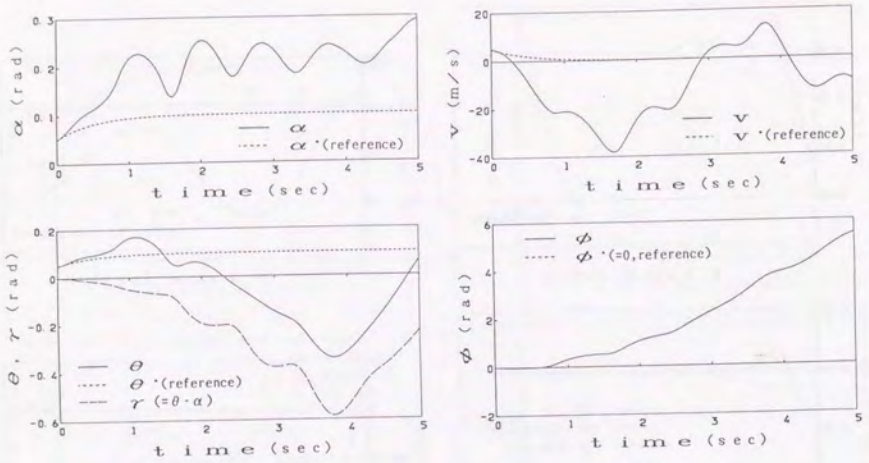

図３－６　故障ｂの場合の出力の時間応答（再構成を行わない）

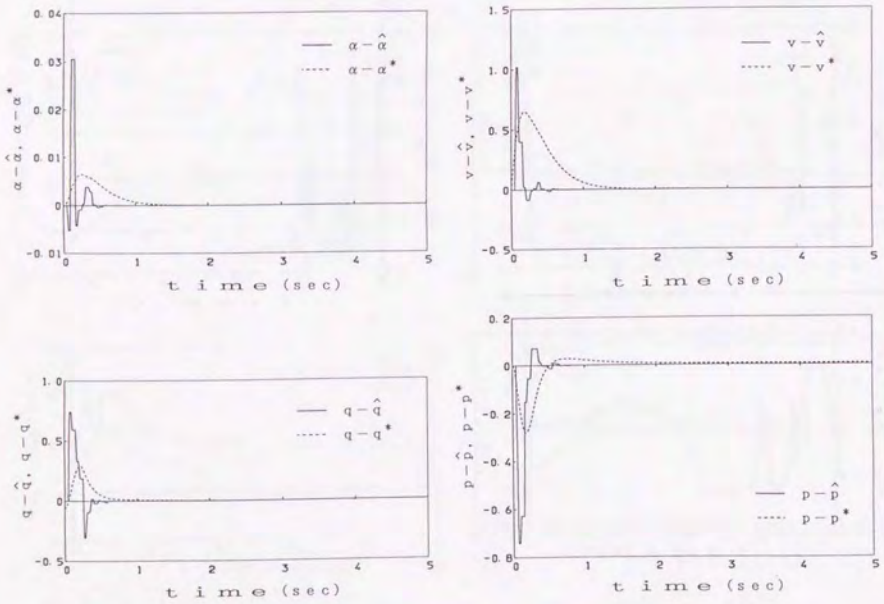

図３－７　同定誤差（Ｙ－$\hat{Y}$）と追従誤差（Ｙ－Ｙ*）の時間履歴（故障ａ）

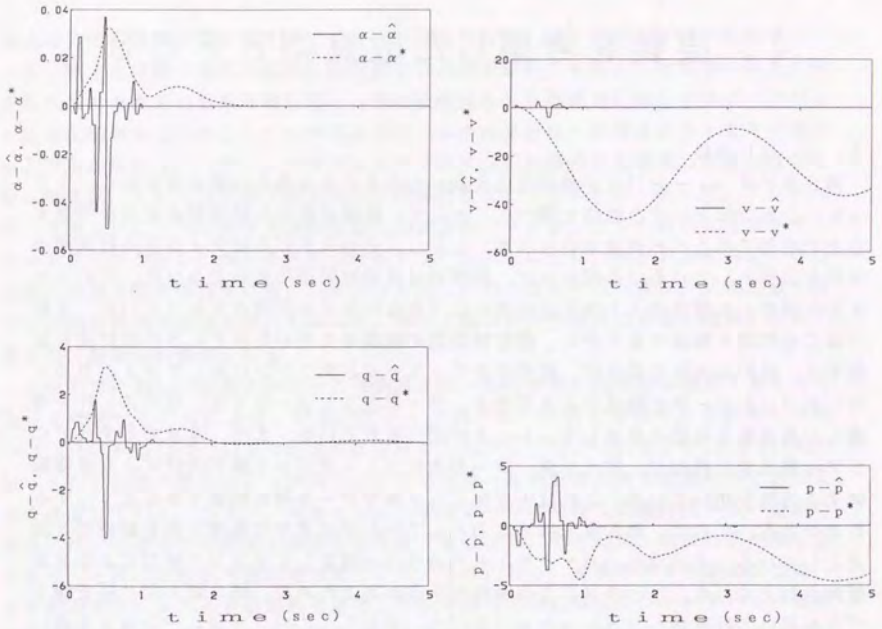

図３－８　同定誤差（Ｙ－$\hat{Y}$）と追従誤差（Ｙ－Ｙ*）の時間履歴（故障ｂ）

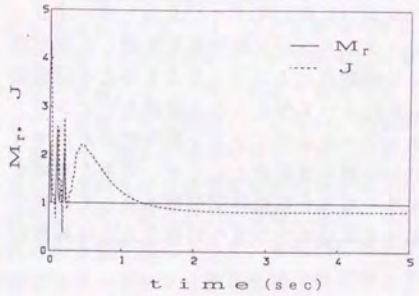

図３－９　M_rとＪの時間履歴（故障ａ）

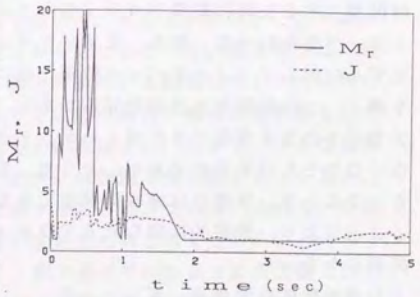

図３－１０　M_rとＪの時間履歴（故障ｂ）

4．実装を考慮した改良型ＲＦＣＳ

4．1　はじめに

第３章ではフィードバック線形化法を用いたＲＦＣＳの基本的構成法を示し，シミュレーションによりその性能を調べた．そして，故障が生じた航空機の安全を回復する上で有効であることが確かめられた．しかし，そのＲＦＣＳはディジタル計算機の使用を前提としているにも関わらず，制御則は連続時間型のものであった．そのため入力が飽和した場合の入力修正が現実にはできないなどの問題点をもっていた．本章ではこの問題を解決するために，離散時間型の制御則を用いたＲＦＣＳの設計法を提案する．設計の大きな変更は，舵面アクチュエータとエンジンに対してディジタル・サーボコントローラを構成する点である．サーボコントローラでは，ロバスト性を考慮して周波数依存型の最適レギュレータが用いられている．また，本章で示すＲＦＣＳでは第３章と異なり，ピッチ角，ロール角をフィードバック線形化法により直接制御する方法を用いている．これは大きなピッチ角やロール角も制御できるようにするためである．さらに，第３章では４出力としていたが本章では横滑り角を除いて３出力としている．その他に，バックサイドでの飛行を想定してエンジン推力による速度制御を加えている．このＲＦＣＳの有効性を確かめるために，第３章と同じ航空機モデルを用いて計算機シミュレーションを行う．そして，このシミュレーションを通してロバスト性の検証，一般入力を採用することの有効性，非線形制御則の必要性，推力制御の有効性などを明らかにする．

以下では本章で提案するＲＦＣＳのこれらの特徴を具体的に説明する．

１）ディジタル・サーボコントローラ

ＲＦＣＳの基本的な構成は第３章で示したものと同じであるが，その制御則を離散時間型にするために舵面アクチュエータとエンジンに対してディジタル・サーボコントローラを設計する．即ち，まずアクチュエータとエンジン動特性を無視した航空機モデルに対してフィードバック線形化法により操舵角とエンジン出力を与える制御則を導く．この制御則は連続時間型である．これによって与えられる操舵角とエンジン出力がそのまま実現できればよいが，アクチュエータ及びエンジン動特性が無視できないのでそれは不可能である．そこで，それらの値を目標値としてサーボ系を構成する．このとき，目標値は連続時間型であるがサーボ系を離散時間制御系として構成することにより，最終的に得られるアクチュエータ及びエンジンへの指令入力は離散時間型になる．

２）周波数依存型最適レギュレータ

サーボ系の設計では周波数依存型の最適レギュレータが用いられている．これにより入力の周波数が高くなるのを抑えることができる．高周波の入力は制御器に与える機械的負荷が大きく，さらに機体の高周波モードを励起し不安定化する恐れがある．

３）ピッチ角とロール角の直接制御

一般に，フィードバック線形化法を適用するには，制御したい出力に関する微分方

程式の右辺に制御入力が含まれていなければならない．ところが，ピッチ角及びロール角に関する状態方程式の右辺には制御入力の項がない．そのような場合には入力項が現れるまで出力の微分を繰り返し，その結果得られる高次系に対してフィードバック線形化制御則を求める．ピッチ角及びロール角の場合は１回微分すると舵角や推力の入力項が現れる．しかし，アクチュエータやエンジン動特性を考慮しなければならない場合は，それらは制御入力ではなく状態変数として扱われる．そして，アクチュエータやエンジンへの指令入力が制御入力となる．それ故，さらに微分してアクチュエータ及びエンジンに関する状態方程式を用いることにより，制御入力を右辺に含む３階の微分方程式が得られる．しかしながら，出力の微分を行う度に，状態変数の微分に状態方程式の右辺を代入するので，その３階の微分方程式は非常に複雑になる．従って，制御則も複雑になる．

この問題に対し，前章ではピッチ角速度とロール角速度に適当な目標値を与えてそれらを制御することにより，間接的にピッチ角とロール角を制御する方法をとった（(3.2-10)から(3.2-14)式で示された方法）．しかし，(3.4-14)および(3.4-15)式で示された目標値の微分はピッチ角やロール角，ロール角速度がある程度微小であることを仮定している．つまり，大きなピッチ角やロール角を伴う非線形性が強い運動の場合には，不適切な目標値の微分を与えることになる．その結果，制御性能が悪くなったり，ときには制御不能に陥ったりする．これに対し，本章で示すＲＦＣＳでは，まず舵面アクチュエータ動特性を無視した航空機モデルに対してフィードバック線形化制御則を求め，次にそれを目標値としてアクチュエータに対してディジタル・サーボコントローラを構成する．これによりフィードバック線形化制御則を求める段階ではピッチ角及びロール角の２階微分が分かればよい．そして，第３章で示した設計法のような線形近似は全く含まれていない．従って，より簡単で，非線形性に対してロバストなディジタル制御系になっている．

４）４（一般）入力３出力系

第３章のＲＦＣＳでは，入出力数はともに４であった．本章で示すＲＦＣＳでは一般入力の数（４）は変わっていないが出力数を４から３にしている．第３章では，迎角，ピッチ角，ロール角に加えて横滑り角（Ｙ軸方向速度）を制御していたのに対し，本章では横滑り角は制御していない．比較的小さな故障の場合や穏やかな運動をしている場合は，確かに横滑り角も制御した方が出力の応答はよい．しかし，大きな故障や激しい運動を伴う場合は制御性能が低下し，ときには制御不能に陥ることもある．これは恐らく，他の出力と同時に横滑り角も制御することはコントロールパワーが限られている航空機にとって，物理的に要求が厳しすぎるためと考えられる．ところで，一般にフィードバック線形化法では付録Ａ３．１節で述べているように入力数と出力数は等しいとする．本章で示すＲＦＣＳのように入力数の方が多い場合には，制御目的を達成する入力は無数に存在し，それを一意的に決めることはできない．しかし，擬似逆行列(pseudo inverse)を用いれば，そのような入力の一つを求めることができる．こうして決定された入力は制御目的を達成する入力の中で最も大きさが小さい．つまり，入力ベクトルのユークリッド・ノルムが最小であり，制御に要するエネルギーが最も小さいと見ることができる．一つの合理的な決定法であろう．

5）推力による速度制御

3章では推力は一定とし，飛行速度（X軸方向速度）は制御していなかった．ただし，水平定常飛行に回復することにより速度はある一定値に落ち着いていた．ここでは，推力による速度制御と舵面による迎角・姿勢制御が比較的干渉が少ない点に着目して，速度と迎角・姿勢を別々に制御する．これはオートスロットルとオートパイロットの関係と似ている．もう少し正確に言えば，まず飛行速度一定とした（uに関する状態方程式を除いた）システムに対し，迎角と姿勢を制御する舵角を決定し，次にその舵角をとった場合のuに関する状態方程式から飛行速度を制御する推力を求める．故障が生じたときに速度，高度ともに十分余裕があれば，必ずしも推力による速度制御を行わなくとも水平飛行は回復される．少なくともフロントサイドと呼ばれる飛行速度領域では，操舵のみで機体の姿勢および経路角を制御することができる．ところが，バックサイドと呼ばれる低速の飛行速度領域では，舵面のみで機体運動を制御することは極めて困難になる（文献2-1,pp.237-242 参照）．バックサイド領域での飛行が特に問題になるはアプローチおよび着陸時に推力を絞り，低速で飛行しているときである．例えば，バックサイド領域で飛行しているときに，機体姿勢を回復するために引き起こしの操舵を行った場合，あるいは昇降舵が機首上げの位置で固着した場合には，経路角安定がないため迎角が増し，高度が下がる．高度に余裕がなければ墜落に至ることもある．この場合，高度の低下を防ぐには推力を増さなければならない．つまり，推力による速度制御が必要になる．

本章で提案するＲＦＣＳを用いて，第３章で使用したのと同じ航空機モデルに対し，以下の６通りの計算機シミュレーションを行う．

①右主翼の半分が破損した場合

②故障発生後にピッチ角３０度，ロール角６０度という大きな姿勢変化を行う場合（再構成を行わなかった場合）

③ＲＦＣＳの設計において一般入力を用いた場合と実際の入力を用いた場合
出力誤差，同定誤差の収束性を比較

④バックサイドでの飛行時に推力制御を行った場合と行わなかった場合
フロントサイドでの飛行時の結果との比較

⑤線形の同定モデルと制御則を用いたＲＦＣＳで再構成を行った場合
非線形のものとの比較

⑥一般入力の個数が３の場合と４の場合の比較

最後に，本章のまとめを述べる．

４．２　航空機モデルの記述

航空機モデルそのものは前章で用いたものと同じであるが，推力による速度制御と舵面による姿勢制御を区別するために，運動方程式を２つに分けて記述する．以下では，エンジンおよびアクチュエータ動特性は１次遅れ系で近似できるとする．

状態方程式

$$\text{S1:}\quad \dot{u}=A_1(X)+B_{11}(X)\delta+B_{12}T_h \tag{4.2-1}$$

$$\dot{T}_h=(-T_h+T_{hc})/T_e \tag{4.2-2}$$

$$\text{S2:}\quad \dot{X}_2=A_2(X)+B_2(X)\delta \tag{4.2-3}$$

$$\dot{\delta}=\Lambda(-\delta+\delta_c) \tag{4.2-4}$$

これらの方程式において，記号は以下のように定義されている．

状態変数ベクトル：$X_a=[X^T,U^T]^T$,

制御変数ベクトル：$U_c=[\delta_c{}^T,T_{hc}]^T$.

ここで，$X=[u,X_2{}^T]^T$，$U=[\delta^T,T_h]^T$である．さらに，

$X_2=[w,q,\theta,v,r,p,\phi]^T$,

$\delta=[\delta_{hL},\delta_{hR},\delta_{aL},\delta_{aR},\delta_{cL},\delta_{cR},\delta_r]^T$,

$\delta_c=[\delta_{hLc},\delta_{hRc},\delta_{aLc},\delta_{aRc},\delta_{cLc},\delta_{cRc},\delta_{rc}]^T$

である．各状態変数および制御変数の定義は3.4.1節を参照されたい．

$A_1(X)\in R^1$，$B_{11}(X)\in R^{1\times7}$，および$A_2(X)\in R^7$，$B_2(X)\in R^{7\times7}$の各要素は定数パラメータと状態変数の既知関数との線形結合の形をしているとする．詳しくは付録Ａ４．１を参照されたい．また，$\Lambda=\mathrm{diag}\{1/T_i\}$である．ここで，$T_i$はｉ番目(i=1,....,7)の舵面に対するアクチュエータの時定数である．T_eはエンジンの時定数である．

(4.2-1)及び(4.2-3)式から分かるように，推力によって発生される力やモーメントはＸ軸方向の力以外は無視できると仮定している．一般に，速度uはX_2の状態変数に比べて変化が遅いのでuの変化が(4.2-3)式に与える影響は比較的小さい．従って，(4.2-3)および(4.2-4)式によって表されるシステム(S2)は，(4.2-1)および(4.2-2)式によって表されるシステム(S1)から近似的に分離していると見なせる．このとき，S2に対してδがS1とは独立に決定できることになる．そして，そのδを既知とすればS1は単入出力系として扱える．そこでは入力が推力であり，出力がＸ軸方向速度である．

一般入力と入力分配行列

3.4.1節と同様に一般入力を$\delta_G=[\delta_{ing1},\delta_{ing2},\delta_{iat},\delta_{dir}]^T\in R^4$ と定義する．実際の入力δとの関係は

$$\delta=P\delta_G \tag{4.2-5}$$

である．ここで，$P\in R^{7\times4}$は入力分配行列である．

数学的にはＰはフルランクであれば任意に選ぶことができる．しかし，ここでは航空機動特性を考慮して次の原則に従って選ぶことにする．

1)一般入力を実際の入力に対応づける．つまり，δ_{ing1}とδ_{ing2}が昇降舵などの縦の制御舵面に対応し，δ_{iat}が補助翼のような横運動の制御舵面に，そして，δ_{dir}が方向舵のような方向を制御する舵面に対応するようにＰを選ぶ．

2)Ｐの各要素の大きさは関係する実際の舵面の舵角限界にできるだけ比例するように与える．

縦の運動に関する一般入力が２つ（δ_{ing1}，δ_{ing2}）とられているのは，縦の２つの出力（迎角とピッチ角）を制御するためである．このようにＰを選べば，制御要求が

厳しすぎない限り実際の舵角に極端なばらつきが起きたり，入力が飽和したりすることはないだろうと期待できる．なお，文献1-32ではminimum effort control法に基づいて入力分配を行う方法を提案している．即ち，各状態を制御するのに要する各入力のcontrol effortを調べ，その中で最も効果的な入力をたくさん使うようにPを決定する．ただし，この方法では効果的な入力に対応する制御器が故障した場合には制御性能に悪影響を及ぼしやすい．

一般入力を用いた状態方程式

(4.2-5)式を(4.2-1)式及び(4.2-3)式に代入するとそれぞれ次のようになる．

$$\dot{u} = A_1(X) + B_{11G}(X)\delta_G + B_{12}T_h \tag{4.2-6}$$

$$\dot{X}_2 = A_2(X) + B_{2G}(X)\delta_G \tag{4.2-7}$$

ここで，$B_{11G}(X)=B_{11}(X)P\in R^{1\times 4}$，$B_{2G}(X)=B_2(X)P\in R^{7\times 4}$である．(4.2-6)式及び(4.2-7)式は次のように書くこともできる．

$$\dot{X}=A(X)+B_G(X)U_G \tag{4.2-8}$$

ここで，$A(X)=[A_1(X),A_2(X)^T]^T$，$B_G(X)=\begin{pmatrix} B_{11G}(X) & B_{12} \\ B_{2G}(X) & 0 \end{pmatrix}$，$U_G=[\delta_G{}^T,T_h]^T$である．

一般入力に対するアクチュエータ動特性は次式で定義される．

$$\dot{\delta}_G=\Lambda_G(-\delta_G+\delta_{Gc}) \tag{4.2-9}$$

ここで，$\delta_{Gc}\in R^4$は(4.2-10)式で定義される指令一般入力である．

$$\delta_c=P\delta_{Gc} \tag{4.2-10}$$

また，$\Lambda_G=I_4/T_{max}$(I_iはixiの単位行列)であり，T_{max}はアクチュエータ時定数の中で最大のものである．以下では，(4.2-8)及び(4.2-9)式を状態方程式とみなして制御系を設計する．

4．3　連続時間ＲＦＣＳ

4．1節で述べたように，本節で示す連続時間制御系はアクチュエータ及びエンジンに対して設計されるディジタルサーボ系の目標値発生器となる．まず，パラメータが既知の場合の基本制御系の構成法を示し，次にそれにパラメータ調整機構を付加した適応型のものを示す．

基本制御系

制御すべき出力を速度，迎角，ピッチ角，ロール角とする．即ち，Y＝［u，α，θ，φ］^{T}である．ここで，$Y_1=\alpha$，$Y_2=[\theta,\ \phi]^T$とおく．Yを微分すると次の方程式が得られる．

$$\dot{u} = A_1(X) + B_{11G}(X)\delta_G + B_{12}T_h \tag{4.3-1}$$

$$\dot{Y}_1 = A_{21}'(X) + B_{2G1}'(X)\delta_G \tag{4.3-2}$$

$$\dot{Y}_2 = A_2'(X_2) \tag{4.3-3}$$

ここで，$A_{2i}(X_2)$，$B_{2Gi}(X)$をそれぞれ$A_2(X)$の第ｉ要素，$B_{2G}(X)$の第ｉ行とおくと，(4.3-1)から(4.3-3)式において

$A_2'(X_2)=[A_{23}(X_2),A_{27}(X_2)]^T$

$A_{21}'(X)=(A_{21}(X)-A_1(X)\tan\alpha)\cos^2\alpha/u$

$B_{2G1}'(X)=(B_{2G1}(X)-B_{11G}(X)\tan\alpha)\cos^2\alpha/u$

と定義される．S2はS1から分離しているとみなすことができるので，まず(4.3-2)及び(4.3-3)式よりY_1,Y_2を制御するδ_Gを決定し，次に(4.3-1)式よりuを制御するT_hを求める．

(4.3-3)式は制御入力を含んでいないが，それを微分して(4.2-6)式，(4.2-7)式を用いると次式のように入力が現れる．

$$\ddot{Y}_2=A_2''(X)+B_{2G}'(X)\delta_G \tag{4.3-4}$$

ここで，$A_2''(X)=\{\partial A_2'(X_2)/\partial X_2{}^T\}A_2(X)$，$B_{2G}'(X)=\{\partial A_2'(X_2)/\partial X_2{}^T\}B_{2G}(X)$である．(4.3-2)及び(4.3-4)式にフィードバック線形化法を適用すると，出力$[Y_1,Y_2{}^T]^T$を目標出力$[Y_1{}^*,Y_2{}^{*T}]^T$に追従させる制御則が次のように得られる．

$$\delta_G=\begin{pmatrix} B_{2G1}'(X) \\ B_{2G}'(X) \end{pmatrix}^{+}\left\{-\begin{pmatrix} A_{21}'(X) \\ A_2''(X) \end{pmatrix}+\begin{pmatrix} G_1y_1 \\ G_2\eta \end{pmatrix}+\begin{pmatrix} \dot{Y}_1{}^* \\ \dot{Y}_2{}^* \end{pmatrix}\right\} \tag{4.3-5}$$

上式において，$^+$は擬似逆行列を表す．また，新しい変数y_1及びηは次のように定義されている．

$y_1=Y_1-Y_1{}^*$

$\eta=[y_2{}^T,y_2{}'^T]^T$

ここで，$y_2=Y_2-Y_2{}^*$，$y_2'=\dot{y}_2=\dot{Y}_2-\dot{Y}_2{}^*$である．

$[B_{2G1}'(X)^T,B_{2G}'(X)^T]^T$はフルランク(=3)をもつとする．(4.3-5)式を(4.3-2)，(4.3-4)式に代入するとそれぞれ

$$\dot{y}_1=G_1y_1 \tag{4.3-6}$$

$$\dot{y}_2'=G_2\eta \tag{4.3-7}$$

となる．G_1が負の値であれば，$t\to\infty$において$y_1\to 0$，即ち，$Y_1\to Y_1{}^*$となる．y_2については(4.3-2)式とy_2'の定義より，次式が得られる．

$$\dot{\eta}=E\eta \tag{4.3-8}$$

ここで，$E=\begin{pmatrix} 0 & I_2 \\ G_2 & \end{pmatrix}\in R^{4\times 4}$である．Eが安定行列になるように$G_2\in R^{2\times 4}$を選ぶと，$t\to\infty$において$\eta\to 0$，従って，$y_2\to 0$となる．

(4.3-5)式によってδ_Gが得られたので，(4.3-1)式の第１項と第２項が分かる．このとき，推力に対する制御則は次式で与えられる．

$$T_h = B_{12}^{-1}\{-A_1(X) - B_{11G}(X)\,\delta_G + G_3(u-u^*) + \dot{u}^*\} \tag{4.3-9}$$

ここで，G_3は負の定数であり，u^*はuの目標値である．(4.3-6)式と同様に，uがu^*に追従することを示すことができる．

このように(4.3-5)および(4.3-9)式の制御則により，制御目的($Y \to Y^*$)が達成される．

適応制御系

(4.2-1)および(4.2-3)式の運動方程式のパラメータには航空機動特性に対する故障の影響が反映される．従って，それらを推定すれば故障後の運動方程式を知ることができる．そして，それに基づいて得られた制御則は故障した航空機において制御目的を達成するものとなる．ここでは，パラメータは繰り返し最小自乗法によりオンラインで同定される．制御則のパラメータは$A_1(X)$，$B_{11G}(X)$等のパラメータの推定値を用いて一定時間毎に更新される．(4.3-5)及び(4.3-9)式より，適応型の制御則は次のようになる．

$$\delta_G = \begin{pmatrix} \hat{B}_{2G1}'(X) \\ \hat{B}_{2G}'(X) \end{pmatrix}^{+} \left\{ -\begin{pmatrix} \hat{A}_{21}'(X) \\ \hat{A}_2''(X) \end{pmatrix} + \begin{pmatrix} G_1 y_1 \\ G_2 \eta \end{pmatrix} + \begin{pmatrix} \dot{Y}_1^* \\ \dot{Y}_2^* \end{pmatrix} \right\} \tag{4.3-10}$$

$$T_h = \hat{B}_{12}^{-1}\{-\hat{A}_1(X) - \hat{B}_{11G}(X)\,\delta_G + G_3(u-u^*) + \dot{u}^*\} \tag{4.3-11}$$

ここで，$\wedge$はその関数の値がノミナルのパラメータを推定値に置き換えて得られたものであることを示している．同定誤差と出力誤差（$Y-Y^*$）の収束性は付録A３．３と同様に示される．

４．４　離散時間サーボコントローラ

現在のところ，非線形系に対する離散時間制御系の設計法は線形系に対するのもほど研究が進んでいない．このため，確立された設計法はほとんどないと言ってよい．勿論，微分方程式を差分方程式で近似することにより連続時間の非線形制御系の設計法を適用できる場合もある．フィードバック線形化法の場合はこれが可能である．しかし，こうした方法はサンプリング周期が十分に小さいことが要求されるため実用的ではない．そこで，本論文ではアクチュエータ及びエンジンに対して離散時間サーボコントローラを構成することによって，制御系の離散化を試みる．サーボコントローラの目標信号はアクチュエータとエンジンの動特性を無視した連続時間制御則により発生される．アクチュエータ及びエンジンは線形系として扱われているので，それらに対し線形の離散時間制御系の設計法を用いてディジタル・サーボコントローラを設計することができる．そこでは，ロバスト安定性を考慮して，コントローラは周波数依存型最適レギュレータ(Frequency Dependent Optimal Regulator:FDOR)（文献4-1, 4-2）により補償される．ＦＤＯＲは入出力の周波数特性に重みをかけることができる．例えば，高周波入力に大きな重みをかけると，ＲＦＣＳは搭載機器に悪影響を与えたり柔軟モードのようなモデル化されていない高周波モードを励起したりする振動

的な入力を除くことが期待できる．以下では，まず，ＦＤＯＲについて簡単に説明し，次にそれを組み込んだディジタル・サーボ系の構成法を示す．

周波数依存型最適レギュレータ

次式で表される線形系を考える．

$$\dot{\bar{x}} = A\,\bar{x} + B\,\bar{u} \tag{4.4-1}$$

$$\bar{y} = C\,\bar{x} \tag{4.4-2}$$

ここで，A，B，Cは適当な大きさの定数行列である．
このシステムに対して周波数依存型の評価関数を次のようにおく．

$$J = (1/2\pi)\int_{-\infty}^{+\infty} \{\bar{y}^* Q\,\bar{y} + \bar{u}^* R(j\omega)^* R(j\omega)\,\bar{u}\}\,d\omega \tag{4.4-3}$$

ここで，＊は共役転置を表す．また，ωは角周波数，jは虚数単位（$=(-1)^{1/2}$）である．Qは定数の準正定行列であり，$R(j\omega)^* R(j\omega)$（＞０）は周波数依存型の重み行列である．R(s)の各要素はラプラス演算子sの有理関数であるとする．いま，新しい入力$\bar{v}$を導入しよう．$\bar{v}$は次式で定義される．

$$\bar{v} = R(s)\,\bar{u} \tag{4.4-4}$$

(4.4-4)式を(4.4-3)式に代入し，パーセバルの定理を用いると次式を得る．

$$J = \int_{0}^{+\infty} (\bar{y}^T Q\,\bar{y} + \bar{v}^T \bar{v})\,dt \tag{4.4-5}$$

最適レギュレータ理論により，最適制御則，$\bar{v}_{opt}$，が(4.4-5)式の評価関数に対して得られる．そして，(4.4-4)式より，元の入力$\bar{u}$が次のように得られる．

$$\bar{u} = R(s)^{-1}\,\bar{v}_{opt} \tag{4.4-6}$$

サーボコントローラの構成

(4.4-1)及び(4.4-2)式で表されるシステムは具体的にはアクチュエータあるいはエンジンである．その伝達関数をG(s)とおく．ここで，(4.4-5)式を評価関数とする最適レギュレータ問題は伝達関数が$G(s)R(s)^{-1}$である拡大系に対して解かれる点に注意されたい．ＦＤＯＲは本来連続時間系に対して設計されるべきものであるが，ここでは離散時間系に適用している．つまり，拡大系$G(s)R(s)^{-1}$を離散時間形に変換し，離散時間の最適レギュレータ理論を用いて制御則を決定する．厳密な理論的検討はなされていないが，シミュレーションではその有効性が確かめられている．

一般入力に対する仮想アクチュエータの動特性は次式で表される．

$$\delta_G(k) = \{p_0/(z-e_0)\}\,\delta_{Gc}(k) \tag{4.4-7}$$

ここで，$e_0=\exp(-\Delta t/T_{max})$，$p_0=1-e_0$である．zはシフト演算子，$\Delta t$はサンプリング周期である．また，$k=t/\Delta t$である．

(4.4-6)式から分かるように$R(s)^{-1}$が入力補償器になる．これを離散時間形にしたものを次のように選ぶ．

$$R(z)^{-1}=p_1/(z-e_1) \tag{4.4-8}$$

ここで，e_1は適当な定数($0<e_1<1$)であり，$p_1=1-e_1$とする．

離散時間の最適レギュレータ理論を(4.4-7)式と(4.4-8)式から成る拡大系に適用すると，最適制御則が次のように得られる．

$$v'(k)=K_{1a}\delta_G(k)+K_{2a}\delta_{pr}(k)+v_c(k) \tag{4.4-9}$$

ここで，K_{1a}及びK_{2a}は最適フィードバックゲインである．$v_c(k)$は外部入力ベクトル，$\delta_{pr}(k)$は補償器の出力ベクトルである．ＲＦＣＳのブロック線図を図４－１に示す．この図に示されたようにＦＤＯＲは比例制御系の中に置かれている．そして，サーボ系の定常誤差を除くためにフィードフォワード補償器が付加されている．比例ゲインをK_{pa}，フィードフォワード補償器を$p_2/(z-e_2)$，そのゲインをK_{3a}とすると仮想アクチュエータへの一般指令入力は次式で与えられる．

$$\delta_{Gc}(k)=\{p_1/(z-e_1)\}v'(k)+\{K_{3a}p_2/(z-e_2)\}\delta_G^*(k) \tag{4.4-10}$$

(4.4-10)式において，$\delta_G^*(k)$は$\delta_G(k)$に対する参照入力であり，(4.3-10)式で与えられる．そして，$v'(k)$とK_{3a}はそれぞれ次式で与えられる．

$$v'(k)=(K_{1a}-K_{pa})\delta_G(k)+K_{2a}\delta_{pr}(k)+K_{pa}\delta_G^*(k) \tag{4.4-11}$$

$$K_{3a}=1-K_{1a}/(1-K_{2a}) \tag{4.4-12}$$

フィードフォワード補償器の動特性は，その出力（(4.4-10)式の右辺第２項）がフィードバック入力（(4.4-10)式の右辺第１項）に過渡状態で影響しないように十分遅く与えられている．最終的に，実際のアクチュエータへの指令入力δ_cは(4.4-10)式の$\delta_{Gc}(k)$を(4.2-10)式に代入することにより得られる．

エンジンに対するサーボコントローラも同様に構成される．即ち，制御則は

$$T_{hc}(k)=\{p_3/(z-e_3)\}v_e'(k)+\{K_{3e}p_4/(z-e_4)\}T_h^*(k) \tag{4.4-13}$$

となる．ここで，

$$v_e'(k)=(K_{1e}-K_{pe})T_h(k)+K_{2e}T_{hpr}(k)+K_{pe}T_h^*(k) \tag{4.4-14}$$

$$K_{3e}=1-K_{1e}/(1-K_{2e}) \tag{4.4-15}$$

T_{hpr}は補償器の出力ベクトルである．$T_h^*(k)$は$T_h(k)$に対する参照入力であり，(4.3-11)式で与えられる．入力分配は不要である．

４．５　シミュレーション

６自由度非線形の航空機モデルを用いて計算機シミュレーションを行い，本章で提案したＲＦＣＳの性能を示す．航空機モデルは第３章で用いたものと同じである．詳しくは３．４節または付録Ａ４．１を参照されたい．ただし，付録Ａ４．１では主翼の破損のような大きな故障が発生した場合を考慮して運動方程式の表現を若干前章とは変えている．

状態方程式は(4.2-1)から(4.2-4)式で与えられている．４．３節で述べたように出力と参照出力をそれぞれ$Y=[u,\alpha,\theta,\phi]^T$，$Y^*=[u^*,\alpha^*,\theta^*,\phi^*]^T$とする．ここで，$Y^*$は次のように$Y(0)$を初期値とし$Y^*(\infty)$に指数収束する時間関数で与える．

$$Y^*=Y^*(\infty)+\mathrm{diag}\{\exp(-.5t),\exp(-3t),\exp(-2t),\exp(-3t)\}(Y(0)-Y^*(\infty))$$

定常状態として４つの場合を考える．

$$Y^*(\infty)=\begin{cases}[220,\ .1,\ .1,\ 0]^T & \text{(T.P.1)}\\ [220,\ .1,\ .5236,\ 1.047]^T & \text{(T.P.2)}\\ [100,\ .1524,\ .1,\ 0]^T & \text{(T.P.3)}\\ [150,\ .1524,\ .1,\ 0]^T & \text{(T.P.4)}\end{cases}$$

(T.P.1)は水平直線飛行，(T.P.2)はピッチ角３０度，ロール角６０度のマニューバ，(T.P.3)はバックサイドで経路角－３度の直線飛行，(T.P.4)はフロントサイドで経路角－３度の直線飛行である．ここで，飛行機効率を.7，有害抵抗係数を.02とすると，最小抵抗速度は本機の場合高度300(m)で115(m/sec)である（文献2-1,pp.237-242参照）．

航空機の諸元，アクチュエータ時定数は３．４節に示してある．エンジン時定数は$T_e=1$(sec)とする．

ノミナルの飛行条件としては次の３つを考える．

ＦＣ１：高度h=2100(m)，対気速度V=221(m/sec)(Mach .66)，迎角α=.1(rad)
ＦＣ２：高度h=300(m)，対気速度V=100(m/sec)(Mach .29)，迎角α=.1(rad)
ＦＣ３：高度h=300(m)，対気速度V=150(m/sec)(Mach .44)，迎角α=.1(rad)

入力分配行列P_4は次のように与える．

$$P_4=\begin{pmatrix} .4 & 1.0 & .4 & .0\\ .4 & 1.0 & -.4 & .0\\ .2618 & .5 & .2618 & .0\\ .2618 & .5 & -.2618 & .0\\ -.2 & -.3 & .3 & .0\\ -.2 & -.3 & -.3 & .0\\ .0 & .0 & .0 & .5236\end{pmatrix}$$

4．3節で示した連続時間ＲＦＣＳの固定パラメータ：

$G_1=-10,\ G_2=[-225I_2,\ -30I_2]$.

離散時間サーボコントローラの固定パラメータ：

アクチュエータ系

$e_0=.819,\ p_0=.181,\ e_1=.8,\ p_1=.2,\ e_2=.998,\ p_2=.002,$

$K_{pa}=3.,\ K_{1a}=-1.208,\ K_{2a}=-3.597$

エンジン系

$e_0=.819,\ p_0=.181,\ e_3=.9,\ p_3=.1,\ e_4=.998,\ p_4=.002,$

$K_{pe}=5.,\ K_{1e}=-1.308,\ K_{2e}=-4.795$

各舵面の舵角制限は前章の通りである．利用可能なエンジン推力の範囲は$0\leqq T_h\leqq 1.78\times10^5$(N)とする．

各サンプリング周期は以下のように与える．

アクチュエータへの指令入力更新の周期　：.01(sec)
エンジンへのスロットル入力更新の周期　：.20(sec)
航空機の推定パラメータ更新（同定）の周期：.05(sec)
制御則のパラメータ更新の周期　：.05(sec)

初期条件としては次の６つを考える．

$$X(0)=\begin{cases}[230,.2,0,.1,5,0,0,.2]^T & (I.C.1)\\ [220,.1,0,.1,0,0,0,\ 0]^T & (I.C.2)\\ [88,\ .2,0,.1,5,0,0,.2]^T & (I.C.3)\\ [130,.2,0,.1,5,0,0,.2]^T & (I.C.4)\\ [230,.15,0,.1,0,0,0,.1]^T & (I.C.5)\\ [230,.2,0,.1,0,0,0,.2]^T & (I.C.6)\end{cases}$$

ここでは$w(0)$の代わりに$\alpha(0)$が示されている．

航空機パラメータの初期値は次のトリム点における正常な場合の値とする．X_{tr}ではw_{tr}の代わりにα_{tr}が示されている．

Ｆ／Ｃ１：$X_{tr}=[220,.1,0,.1,0,0,0,0]^T$,
$U_{tr}=[-.1560,-.1560,.008112,.008112,-.06466,-.06466,.0,.1118\times10^6]^T$

Ｆ／Ｃ２：$X_{tr}=[100,.1524,0,.1,0,0,0,0]^T$,
$U_{tr}=[.08577,.08577,-.02840,-.02840,.06655,.06655,.0,.1349\times10^5]^T$

Ｆ／Ｃ３：$X_{tr}=[150,.1524,0,.1,0,0,0,0]^T$,
$U_{tr}=[-.2559,-.2559,.01015,-.01015,-.1020,-.1020,.0,.6603\times10^5]^T$

次の４通りの故障を考える．

故障Ａ：右主翼の半分（$b/4\leqq y\leqq b/2$ に対応する部分）が破損し，右水平尾翼の効きが正常な場合の５５％になる．ここで，右主翼の破損に伴い右補助翼の効きが正常な場合の５０％になるとする．

故障Ｂ：左水平尾翼が-.1(rad)で固着し方向舵の効きが５０％になる．

故障Ｃ：右水平尾翼の効きが７７％になり，右補助翼の効きが０％になる．

故障Ｄ：左水平尾翼が-.4(rad)で固着する．

右主翼の半分が破損した場合の航空機モデルについては付録Ａ４．２を参照されたい．

最小自乗法におけるゲイン行列の修正：繰り返し最小自乗法では推定パラメータが更新される度に，ゲイン行列が小さくなる（文献2-2,pp.58-61）．これがあまり小さくなると推定パラメータの修正量が小さくなるので同定が進まなくなる．そこで，ゲイン行列のトレースがある値より小さくなった場合は，それが設定された値になるようにゲイン行列をリセットしている．詳しくは付録Ａ４．３を参照されたい．

このＲＦＣＳは多くの推定パラメータ（１３９個）を扱い，かなりの量の計算がオンラインで要求される．従って，実機に搭載する場合は高性能の計算機が必要になる．さらに，計算機と他の装置とのインターフェイス（Ａ／Ｄ，Ｄ／Ａ変換など）に要する時間も無視できない．計算とインターフェイス操作に起因する遅れはＲＦＣＳの性能に好ましくない影響を与える可能性がある．その遅れに対するロバスト性を調べるために，シミュレーションでは通常のディジタル制御則より１サンプリング周期だけ遅らせて制御入力を航空機に加えている（図４－２参照）．パラメータ推定においても同様に１サンプリング周期だけ遅らせて値を更新している．従って，制御則のパラメータの更新も遅れることになる．

シミュレーション結果

表４－１にまとめられている６ケースについてシミュレーションを行った．この表で「制御？」の欄の「行わない」は制御舵面をトリム舵角に固定した自由応答を意味する．また，「再構成？」の欄の「行わない」は制御則のパラメータが更新されず，ノミナルのパラメータが用いられることを示す．

以下で示す時間応答の図において$\gamma(=\theta-\alpha)$は飛行経路角を表し，添え字'tr'は目標のトリム点における値であることを示す．

ケース　１（正常／故障Ａの場合の自由応答）

図４－３から分かるように，正常な航空機の時間応答は典型的な縦のモード（長周期及び短周期モード）を示している．横・方向運動についても典型的な応答を示す．ここで，正常な場合のロール角の応答は時間軸と重なってよく見えない．これに対し，右主翼が破損した航空機はロール運動に関する安定性を失っている．実際，航空機は約160(deg/sec)でローリングしながら降下し，故障発生後約17.5秒で墜落している．

ケース　2（故障Aで再構成を行った場合）

図４－４ａ及びｂの結果より，制御系を再構成することで航空機の安全が回復できることが分かる．故障Aの場合，再構成を行わなければ制御不能に陥ることが確かめられている．再構成を行った場合には，航空機は過渡応答において約150度もローリングするものの，速度ｕは約15秒で，他の出力は約10秒で目標のトリム点に達し，ほぼ整定している．横滑り角β（あるいは横速度ｖ）は制御されていないので，０には収束していない．高度の図より高度の損失は約160ｍ であり比較的小さな高度の損失で水平飛行に回復できていることが分かる．サーボコントローラの出力δ_Gの目標値δ_G^*への追従性を見るために，δ_Gとδ_G^*の時間応答を示す．過渡的には多少遅れがあるが，δ_Gがδ_G^*に追従している様子が分かる．T_hとT_h^*についても同様である．

ケース　3（故障Bで再構成を行わなかった場合）

ここでは，故障Bが発生した航空機に対し３０度のピッチアップと６０度の右ローリングを同時に行わせる．初期状態は水平定常飛行である．再構成を行わなかった場合の応答が図４－５ａ，ｂに示されている．かなり大きな運動が要求されているにもかかわらず，再構成を行わなくても制御目的が達成されていることが分かる．勿論，再構成を行っても制御目的は十分に達成される．このことは，基本制御系が相当ロバストであることを意味している．速度ｕについては利用できる推力の大きさが限られているので，航空機は２０秒以内に目標速度220(m/sec)に達することができない．

ケース　4（一般入力の有効性の検討：故障Ｂ／Ｃの場合）

このケースではパラメータ同定において一般入力を用いることの有効性を調べる．同定誤差εを次式で定義する．

$$\varepsilon=\{\dot{X}-A(X)-B_G(X)U_G\}/(s+30) \qquad (4.5\text{-}1)$$

図４－６，７にはそれぞれ一般入力を用いた場合と用いなかった場合の各入出力の時間応答が示されている．図４－８には$(\sum_{i=1}^{6}\varepsilon_i^2)^{1/2}$の値の時間履歴が描かれている．ただし，この値が１より大きい部分は描かれていない．ここでは，故障Ｂを想定し，初期状態は水平定常飛行とする．そして，目標のトリム状態は初期状態と同じ水平定常飛行(T.P.1)である．図から分かるように実際の入力を用いた場合は応答が収束せず，同定誤差も０にならない．一方，一般入力を用いた場合はεが約６秒で０に収束し，出力誤差も約１２秒で目標値にほぼ収束している．図４－９，１０，１１は故障Ｃの場合の結果である．この場合には出力の時間応答は一般入力を用いない方がよく，同定誤差の収束速度は大体同じである．しかし，いろいろな故障に対して両者を比べると，一般入力を用いた方がトリム状態に回復できるケースが多い．さらに，一般入力を用いないＲＦＣＳでは故障Ａを補償できないことが確かめられている．どちらが優れているか一概には言えないが，一般入力を用いた方が同定すべきパラメータの数

が少ないので計算量が減り，計算機への負担が小さいのは事実である．実際，(4.2-6)及び(4.2-7)式のパラメータの数(139)は(4.2-1)及び(4.2-3)式のパラメータの数(187)より２６％小さい．また，シミュレーションで使用した大型計算機におけるＣＰＵ時間を比べると，ある計算例では一般入力を用いた方が29.97秒，実際の入力を用いた方が33.37秒であった．これらのことから，筆者は一般入力を用いたＲＦＣＳを使用した方がよいと考える．

ケース　5（バックサイド飛行時の推力制御の効果）

図４－１２の結果は推力制御を行った場合のものである．これを見ると，速度ｕ以外の出力は目標値に収束していないが，機体運動はほぼ安定な状態で整定していることが分かる．飛行経路角は大体目標の－３度になっている．しかし，αとθが目標値に収束していないので，$\gamma=-3$(deg) となったのは偶然である．高度の図から，飛行経路は目標より約７０ｍ下になっているものの航空機は直線飛行を行っていることが分かる．これに対し，図４－１３に示されているように，バックサイド飛行時に推力制御を行わなければ，約15秒で墜落する．操舵は機首上げにとられているにもかかわらず，速度が増加し経路角が減少するという典型的なバックサイドでの飛行形態（経路角不安定）を示してる．一方，図４－１４及び図４－１５はそれぞれフロントサイドで推力制御を行った場合と行わなかった場合の結果を示している．推力制御を行わない場合は速度ｕが制御されないので，飛行経路が目標値より若干下がり気味であるが，他の出力は推力制御を行った場合と同様に良好に制御されている．このように，フロントサイドでは推力制御は必ずしも必要でない．推力制御を行った場合でバックサイドとフロントサイドの結果（図４－１２と図４－１４）を比べると，後者の方がよいことが分かる．これはバックサイドでの飛行制御の難しさを示している．

ケース　6（線形制御則との比較）

図４－１６と図４－１７は線形の同定モデルと制御則を用いたＲＦＣＳによって制御された場合の結果を示している．ここで，同定モデルと制御則が線形の運動方程式に基づいているという点以外は，線形のＲＦＣＳは非線形のものと全く同様の設計法でつくられている．図４－１６は初期状態が比較的小さい場合(I.C.5)の応答である．出力は約１０秒で目標値に収束している．一方，図４－１７は初期状態が多少大きくなった場合(I.C.6)の結果である．航空機は２回転以上ローリングし，経路角が－４０度前後に達している．航空機は高度を下げ続け，水平飛行は回復されていない．ロール角の図で目標値ϕ^*が変化しているのは，常に$|\phi-\phi^*|<180$(deg)となるようにϕ^*を修正しているためである．このように，線形のＲＦＣＳは小さな擾乱に対しては有効に働くが，擾乱が大きくなって非線形性が強くなると全く同定・制御ができなくなる．これに対し，図４－１８に示されているように，非線形のＲＦＣＳを用いると初期状態が大きい場合(I.C.6)でも出力は良好に制御されている．なお，故障Ａでは線形のＲＦＣＳは航空機運動を整定できないことが確かめられている．

最後に，３個の一般入力を用いたＲＦＣＳ（3GI-typeと呼ぶ）について触れておく．出力$[Y_1,Y_2^T]^T$の数は３なので，(4.3-5)式の制御則を３個の一般入力，例えば$\delta_G=$

$[\delta_{ing1}, \delta_{ing2}, \delta_{roll}]^T$，に対して決定することも可能である．ただし，3GI-typeは４個の一般入力を用いたＲＦＣＳ（4GI-typeと呼ぶ）より入力数が少ないので，同定すべきパラメータの数が小さくなる．両者の性能を比較するために，数通りの故障と初期状態に対してシミュレーションを行った． 入力分配行列P_3を次のように選ぶことにより，δ_{ing1}とδ_{ing2}を昇降舵のような縦運動の制御器，δ_{roll}を補助翼のような横運動の制御器に対応させることができる．

$$P_3=\begin{pmatrix} .4 & 1.0 & .4 \\ .4 & 1.0 & -.4 \\ .2618 & .5 & .2618 \\ .2618 & .5 & -.2618 \\ -.2 & -.3 & .3 \\ -.2 & -.3 & -.3 \\ .0 & .0 & .5236 \end{pmatrix}$$

$P_3(7,3)$（$=P_4(7,4)$）以外のP_3の要素はP_4と同じである．ここで，$P_3(7,3)$は旋回する場合の操舵を考えると正の値でなければならない．大抵の故障において，結果は４個の一般入力を用いた場合と大きな違いはない．しかし，故障Ａや制御舵面が固着したいくつかの故障では，３個の一般入力を用いたＲＦＣＳは故障を補償できなかった．この点について筆者は次のように考える．方向舵はヨーイングに対しては支配的な影響をもつが，ローリングに対しては比較的小さな影響しかもたない．従って，方向舵でロール運動を制御しようとすると，好ましくないヨーイングを引き起こす．そして，そのヨーイングによってローリングが引き起こされてしまう．このローリングの方向は当初，方向舵によって意図されていた方向と逆である点に注意されたい．例えば，右バンクを抑えるために負の操舵（$\delta_{roll}<0$）をしたとしよう．このとき，方向舵は右に切られ，補助翼とともに負のローリングモーメントを発生する．しかし，方向舵は同時に正のヨーイングを引き起こし，それは正のローリングを発生させることになる．ただし，$P_3(7,3)$の値の選び方により応答特性は変化する．$P_3(7,3)$を大きくとるとヨーイングから生じるローリングの影響が大きくなり応答は悪くなる．しかし，その値をどのように選ぼうとも，3GI-typeでは方向舵はロール制御舵面とみなされ，その舵角は常に補助翼の舵角に比例する．これに対し，4GI-typeでは，方向舵の操舵はそのような束縛を受けない．そのことがロール制御に好ましい影響を与えているのではないかと考えられる．

以上の計算例では，うまく回復できた場合を示したが，勿論故障の程度や初期条件などによっては，回復できない場合も確認されている．例えば，ケース２では右水平尾翼の効きを５０％まで落とすと，同じ初期条件でも水平飛行に戻ることはできない．

４．６　まとめ

本章では，アクチュエータとエンジンに対しディジタル・サーボコントローラを構成することにより制御入力が離散時間形式である再構成可能な飛行制御系を設計した．

サーボコントローラへの参照入力はアクチュエータとエンジン動特性を無視した航空機運動方程式に対して得られるフィードバック線形化制御則である．そして，実際のアクチュエータあるいはエンジン出力をこの制御則に追従させることにより制御目的を達成する．ＲＦＣＳの性能は計算機シミュレーションにより調べられた．その結果，以下のような結論を得た．

1）このＲＦＣＳは制御器の故障だけでなく安定性のような固有の動特性を変える機体の故障をも補償できる．そして，主翼破損のような大きな故障の場合には過渡的に非常に大きな運動を行うものの，制御系は航空機の安全を回復することに成功している．このことから，本ＲＦＣＳは非線形性の強い状況でも有効に働くことが分かる．

2）パラメータ同定で一般入力を用いると同定すべきパラメータの数が減り，同定に要する計算量を減らすことができる．このことは，ＲＦＣＳが実際の入力の機能的冗長性を利用しながら，故障の同定を効率的に行うことができることを意味する．全てのシミュレーションにおいて，一般入力を用いることの優位性が現れるわけではないが，一般入力を用いた方が計算機への負荷が小さいのは事実である．

3）一般入力の数や入力分配行列をどのように与えるかは物理的意味を考えて決めなければならない．これの与え方がＲＦＣＳの性能に大きく影響する．航空機では少なくとも縦，横，方向の運動を制御する制御器，即ち，昇降舵，補助翼，方向舵に対応する一般入力を定義することが望ましい．

4）本章で提案したＲＦＣＳは推力による速度制御の機能をもつ．これはフロントサイドの飛行では必要でない場合が多いが，バックサイドの飛行では重要である．特に，着陸アプローチのような低高度，低速時に故障が起きた場合には推力による速度制御が不可欠である．

5）ノミナルの制御系は故障によってはかなりのロバスト性を示す．即ち，再構成を行わなくても，制御目的を達成することが可能であることがシミュレーションから分かった．しかし，再構成を行うことにより一層航空機の安全性を高めることができる．これは，特に主翼破損のような大きな故障が発生した場合に言える．