

超最適化による調和設計

—未来時制工学のための超最適化HOPE—

Harmonious Design by Hyper Optimization

—HOPE : Hyper Optimization for Prospective Engineering—

高橋 武則
(Takenori TAKAHASHI)

【要 約】

製品は顧客にとっては必要な出力を生成する手段（装置あるいは道具）である。設計の観点から見ると製品には2つの回帰が存在する。一つはユーザーが製品を使用するための入出力回帰であり、もう一つはエンジニアが製品を設計するための超回帰である。超回帰とは下位の回帰のパラメータに対する上位の回帰のことである。前者は使用のための回帰で、後者は設計のための回帰である。

本研究では通常の設計、頑健設計、柔軟設計そして超最適化のための超回帰および数理計画法に基づくHOPEについて論じる。この方法は多数の設計因子がありかつコスト、デリバリー、安全性そして環境などの考慮すべき項目が多数ある場合に有効な解を合理的に与えるものである。この設計アプローチが調和設計である。

キーワード : 超最適化, 調和設計, 階層構造, 超回帰, 柔軟設計, 数理計画法, データ・マネジメント

【Abstract】

The product is a means (equipment or device) to generate the output requested by user. There are two kinds of regression from the viewpoint of design in it. One is input-output regression for user (customer) to use product and the other is hyper regression for engineer to design product. Hyperregression means the superior regression (upper level regression) of parameters in subordinate regression (lower level regression). The former is the regression for designing and the latter is the regression for using.

This paper discusses HOPE (hyper optimization for prospective engineering) which is the method based on Hyperregression and mathematical programming for ordinary design, robust parameter design, flexible design and hyper design. It enables us to get useful and effective solutions rationally when there are many design factors and many items to be considered such as cost, delivery, safety and environment. This kind of design approach is harmonious design

Keyword : hyper optimization, harmonious design, hierarchical structure, hyper regression, flexible design, mathematical programming, data management

1. はじめに

設計においては、多様なトレード・オフが生じるために、関係者全員の満足度を矛盾なく高める「絶対的な完全最適解」を得ることは困難である。そこで、全員が高い満足を得ることはできなくても納得して合意できる解を求める設計を行うことが現実的である。本研究ではこの設計のことを調和設計と呼ぶ。以下に調和設計を正式に定義するとともに、その定義に必要な定義要素も明らかにする。

「調和設計」とは、超最適化を用いて多種のトレード・オフの調整を通じて関係者の合意を形成することにより設計諸元を決定することである。そして、「超最適化」とは、超回帰（回帰係数の回帰）と超機構（複数の下位機構の上に位置する上位機構）といった超構造（2階層の階層構造）に基づく対話型逐次最適化のことである。

納得のもとでの合意 (agreement) は諦観のもとでの妥協 (compromise) とは異なる。合意は能動的で積極的で前向きなものであり、妥協は受動的で消極的で後向きなものである。調和設計は合意形成であり、そのための概念的基盤がHOPE (hyper optimization for prospective engineering: 未来時制工学のための超最適化) である。その目指すところは階層構造の設計である。^{[27], [45], [46]} これは未来時制工学 (未来時制の製品作りに関する工学: prospective engineering) のための超最適化 (2階の階層構造である超構造に基づいた最適化) のことであり、本研究ではこれを頭文字による短縮表現であるHOPEという略称を用いる。

数理計画法を用いて設計を行う場合において、複雑な定式化のもとでは厳密解を得ることが難しくなる。近年はコンピュータの発達により、短時間に得られる実用的な解としてヒューリスティック解が重要な役割を果たしている。HOPEにおいても厳密解が得られれば望ましいが、短時間でヒューリスティック解を求めて何度もPDCAを回す方が合理的と考えている。さらには、正式な解 (実行可能解の中から選択された解) とともに、準解 (quasi solution) も対象とする。本来数理計画法においてはすべての制約条件を満たす実行可能解の集合の中から

目的関数を最適化するものが最適解となる。しかし、多種の制約条件が用意された場合には、すべての制約条件を満たす実行可能解が存在しないことがしばしば発生する。その場合に、数学的には解なしであるが、実行可能解領域の近くに存在する条件を設計の検討対象にするのは現実的である。これを本研究では準解 (解に準ずるもの) と呼ぶ。もし制約条件を譲歩すれば、そのことによりそれは実行可能解となるからである。設計の話し合いの場では、数学的に厳密な解でも合意を得られないことはある。準解でも合意が得られる可能性はある。合意が得られたら、その段階で制約条件を緩和すれば準解は正式な解にすることができる。

TQM (total quality management: 総合的質経営) の目指すところは質を中核とした総合的な経営であり、その中身は質の創造と保証の2つから構成されている。^[53] 質の創造には企画 (何を作るかの概要決定) と設計 (作るものの諸元の詳細決定) があり、本研究は後者に焦点を絞り総合的質設計 (TQD: total quality design) のための概念的な基盤であるHOPEを提唱する。これは考え方と進め方と数理の3つの要素から構成されている。考え方は目的 (合意形成) であり、進め方 (PDCAサイクル) は手続き的手段であり、そして数理 (超最適化) は道具的手段である。この3つのものが有機的に働くことで総合的質設計を実現する調和設計が可能となる。

TQMの要点を簡潔に表現すれば、「望ましい質を低コストで迅速に提供できるように創造し保証する」となる。口語で簡潔に表現すれば、「良い、安い、早い」ということになる。このために必要な設計は質を中核とした総合的な設計である。それは、多種の指標 (特性、項目) に目配せをした上で、顧客・従業員・関係会社・社会・地球環境などの多数の関係するもの (ステークホルダー、関係者) にとって望ましい設計であることが必要である。そのアプローチにおいては関係するものが多いためにトレード・オフの問題を解決しなければならない。なお、特性とは商品の出力のことでそれは商品の存在意義にかかわるものであり、項目とは経営の視点から重視すべき商品の特徴 (作るうえでの特

徴と使ううえでの特徴) のことであり、両者はいずれも設計因子の関数である。

HOPEが考えるTQDとは、「設計対象の諸元の条件決定に関して質を中核において総合的な観点から様々な指標(特性と項目)を考慮して関係者の合意を形成すること」である。このタイプの設計を行うための基盤となる考え方の一つがHOPEである。その最も中核である考えは「設計=関係者の合意」である。そして、HOPEは合意を形成するに当たり、恣意的ではなく手続きを踏んで、主観的にではなく客観的に進めるための考え方と進め方と数理を用意している。

本研究は、最初に設計の基本的な考え方を明らかにしたうえで、HOPEを構成する重要なテーマである柔軟設計、多入力、連合設計などについて議論する。なお、多段設計や差分法^[43]や多頭法^[44]も重要なテーマであるが、これらについては別の機会に議論する。

2. 認識・創造・運営と演繹・帰納・仮説推論

椿^[56]は設計科学の立場から設計の抽象的なプロセスとして価値選択、変換、最適化、価値注入の4つの段階を示している。これは設計に企画や開発も含めた全体的なプロセスを俯瞰した構造化である。本研究はこの中の主に最適化およびその前後の部分について議論するものである。

設計のためにはデータを通して設計対象を客観的なモデル(モデル)として認識し、それに基づいて創造し、そのうえでうまく運営して採算の取れるものしなければならない。それを科学的に進めるためにはロジカルなアプローチとして演繹と帰納と仮説推論を効果的に用いる必要がある。

2.1 データ・マネジメント

かつて、リンカーン(Abraham Lincoln, 1809~1865, 第16代米国大統領)が米国大統領選挙においてゲティスバーグで行った演説の中に、彼が標榜した民主主義政治を端的に表している以下の表現がある。

government of the people, by the people, for the people (邦訳: 人民の人民による人民のための政治)

この表現自体はリンカーン本人のオリジナルではないが、そのことに関する議論については本研究では割愛する。

この演説で述べられている重要な3つの内容
government of the people
government by the people
government for the people
を短縮して表現する場合、英語では2つの名詞の間に入った前置詞を削除して前後の名詞の順番を逆転させることになる。するとこの場合3つの言葉はいずれも以下のものとなる。

people government

そして邦訳においても短縮表現する場合は同じで、やはり3つはいずれも人民政治という表現になるであろう。したがって短縮表現を用いる場合には、その真意は前述の3つの意味のいずれであるのかを明らかにしなければ誤解を招くことになる。

本研究ではデータ・マネジメントという言葉を用いる。これは短縮表現なので、最初にその意味を以下に明らかにしておきたい。

management based on data

本研究のデータ・マネジメントはデータに基づくマネジメント(management based on data)であって、データのマネジメン(management of data)ではない。managementに続く部分をbased on dataと表現するのは、dataを十分に使いこなして総合的な判断(意思決定)によって経営がなされることを強調したいためである。人々がデータをマネジメントするわけではないし、ましてデータ自体がマネジメントをするわけではない。

関連する言葉に以下のものがある。

①データ・サイエンス(data science)

意とするところはscience based on data

②データ・エンジニアリング(data engineering) 意とするところはengineering based on data

これらの両者はいずれもデータそのものをどう扱うかというのではなく、データに基づいた取り組みを意味している。そもそもデータそのものの処理を扱うのはスタティスティクス(statistics: 統計)なのである。しかしながら、データそのものに対して科学するという

science of dataの意味でデータ・サイエンスが用いられることも少なくないので注意が必要である。

データ・サイエンスとデータ・エンジニアリングを識別するキーワードは以下の認識と創造である。

***認識**：対象が有する因果メカニズムをモデル化したうえで理解して説明をすること。

***創造**：モデル（因果メカニズム）に基づいて設計し製作（量産）すること。

認識を主体とする活動はデータ・サイエンスで、認識と創造の両方を含めた全体の活動がデータ・エンジニアリングである。しかし、本研究はこれら2つのもののさらに上に位置するものとしてデータ・マネジメントがあることを強調したい。

2.2 もの作りのビジネスと理学・工学・経営学

もの作りの基盤は工学（固有技術）で、その根幹はメカニズム（法則や原理：例えばフックの法則やテコの原理など）を窮める理学（科学）である。しかし、それらだけではもの作りのビジネスを行うことはできない。以下に、振り子の周期の式を求めることと要求された周期の振り子時計を作ることを例に取り上げて説明する。

***理学**：メカニズムの本質を窮める。

振り子の周期と糸の長さのメカニズム（関係の本質）を窮めるのが理学である。

***工学**：メカニズムを応用して創造（製造）する。

振り子の周期と糸の長さの関係を応用して、例えば振り子時計を作るのが工学である。

***経営学**：創造したものでビジネスをする。

例えば振り子時計を作る場合、これでビジネスをするのが経営学である。

もの作りがビジネスになるためには工学（固有技術）だけでは不十分で、管理技術との連携が必要になる。そして、TQMとは事実に基づいた科学的なアプローチによる質を中核とした総合的な経営管理技術である。それ故に、TQMは理学と工学と経営学の3つから成り立っている。

注目する現象（実際に起きている現象＝事実）に対しては以下の3つのステージがある。

(1)理学（認識）：そのメカニズムを理解する。

対象に手を加えずに知識を獲得する。

(2)工学（創造）：そのメカニズムを応用する。

対象に手を加えて役に立つものを作る。

(3)経営学（運営）：そのメカニズムで商売をする。対象をやりくりして採算がとれるものにする。

事実（＝実際に起きている現象）を観測し記述するとデータ（数値、言語、画像、映像、音声ほか）となる。このデータに基づいて対象のメカニズムを数式で記述し、これをモデル（模型）と呼ぶ。したがって、事実に基づく品質管理（quality control略してQC、近年はむしろquality management略してQMと表現されることが多い）には、データに対して認識し創造し運営するという3つのステージが存在する。特にSQC (statistical quality control) はこの3つのステージのすべてを必要とする。そして、すべてに先行して構想がある。

①構想：現象に対して問題発見・課題設定を行う。

②認識：対象のメカニズムをモデルとして理解する。

理論モデルが望ましいが、近似モデルでもよい。

③創造：モデルに基づく創造（製作、対策）を考える。

問題解決・課題達成のための創意工夫をする。

④運営：やりくり（算段）して採算をとる。創造したものを経営的に妥当な（合理的な）ものに整える。

2.3 HOPEにおける理論式と近似式の位置付け

理論式があることはたいへん望ましいが、理論式がなくても寄与率の高い近似式があれば設計も品質保証もともに可能である。したがって理論式を求める努力を放棄してはならないが、それが絶対であると拘って無理な深追いをすることはない。しかし、何故データが理論式に整合しないのかを原因追究し、そして対策を講じ

て不整合を解消する努力は必要である。

また、理論式が理論的に存在している場合でも、実際のデータはそれに整合しないことが多く、その場合は近似式で設計し品質保証するというアプローチが実務的に有効である。

具体的な説明のために「振り子の周期」を応用した振り子時計を例に取り上げ、図1を用いて議論する。

1) 理学的アプローチ

振り子の等時性(振り子の法則)はガリレオ・ガリレイ(Galileo Galilei 1564~1642)によって発見された。そして糸の長ささと周期に関する理論式は以下のとおりである。

$$T = 2\pi\sqrt{l/g}, \quad g \text{は重力加速度 (9.8m/sec}^2\text{)}$$

方程式を解いてこれを手に入れることはとても重要であるが、しかしこれが分からなくても実験データに基づいて近似式を求めることができる。近似式は単なる対応でしかないためにメカニズムの本質は表現されていないが、それを内挿の範囲で用いて振り子時計を設計することができ、そのうえできちんと製造するならば品質保証をすることもできるのである。

2) 工学的アプローチ

図1の中に示しているように、理論式が分からなくてもきちんとした実験で得られたデータによって近似式を得ることはできる。多くの場

合は1次近似で十分であり、特別な場合でも2次近似を用いれば十分である。求めた式に基づいて要求された周期の振り子を設計した上で量産して品質保証をすることは、近似式を用いても十分に可能である。ただし、近似式は本質的な考察(理学的考察)ができないことと外挿(実験の水準の範囲外での式の適用)はリスクが高いことに注意が必要である。

3) 経営学的アプローチ

振り子時計をビジネスとして成功するには更なる総合的な観点からの設計が不可欠である。

①市場は(顧客は)如何なる振り子を求めているか?

- *周期 *精度 *使いやすさ
- *大きさ *形 *重さ *色 *価格

②それは合理的に提供(作り、販売)できるのか?

- *コスト(費用) *材料・部品の調達
- *設備・機械 *工具・治具
- *ライン編成 *製造工程のレイアウト

市場で求められているものを提供するのがマーケット・インの考えである。このためには①に示した市場のニーズを把握し、それを実現する設計であることが必要である。そして同時に、それを合理的に提供するために②に示した現場の要求や実態を反映する設計でなければならない。①を無視すればプロダクトアウトとなって買ってもらえないし、②を無視すれば経営的合

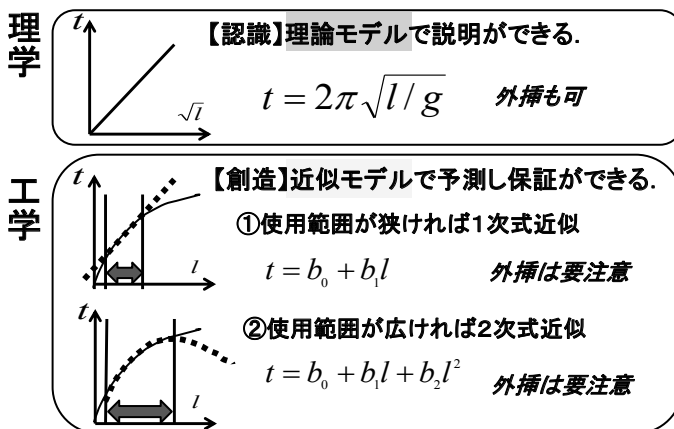


図1 理論モデルと近似モデル(1次近似と2次近似)

理性を欠くために採算がとれないことになる。

2.4 演繹と帰納と仮説推論

かつて論理学において論理的推論に関しては演繹 (induction) と帰納 (deduction) の2種類だけが存在していた。しかし近年になって第三のものとして仮説推論 (abduction) が登場してきた。^[62]

* 演繹

これは前提条件に基づいて論理的な推論で結論を導くことであり、理学が主として採用するアプローチである。推論のシナリオ (推論の進め方) の作成は発想であるが、推論そのものは論を進めることであって発想ではない。

* 帰納

これは、同じ条件を満足するいくつかの事実から普遍的で妥当な知見を導くことである。すなわち、事例から仮説を立てたうえで検証して結論を導くことであり、工学が主として採用するアプローチである。仮説は結論に関係する事例との出会いあるいはそれまでの経験・知見に基づいたアイデアの発想から生まれる。

仮説は発想であり、それ自体は全く自由なものである。それ故に検証が不可欠である。ただし、本研究では検証 (工学では再現性の確認や試作など) の手続きとして実験 (ときには調査) を基盤とするため図1の理論モデルと近似モデル (1次近似と2次近似) においては、研究室での高度に管理された実験以外は常に「共変量 (交絡因子)」も視野に入れなければならない。実務ではしばしばこれによる手痛いペナルティ (再現性の確認における失敗) を被る。

* 仮説推論^[62]

近年になって、帰納法と演繹法に並ぶ第三の推論法とし仮説推論がアメリカの哲学者パース (Chales S. Peirce, 1838～1914) により提唱された。これは起こった現象をうまく説明できる仮説を形成するための推論法のこと、仮説推論 (仮説的推論) あるいは仮説形成とも訳される。そして新たな科学的発見に有用なものであるとされている。

推論の例としては次のようなものがある。

[A] Aが起きた (現象)

[B] Bが起るとAが起きる (法則)

[C] 故にBが起きたであろう (仮説)

ここでは [A] の現象の原因を説明するため、最終的に [C] の仮説を立てている。しかし、AはB以外の原因でも起こるとしたら、[B] の法則が正しいとしても、それがこの場合に当てはめて良いかどうかは何とも言えない。しかし、可能性があるのであるから、試みる価値はある。

設計は創造力 (新しいものを造る能力) によるわけであるが、それは想像力 (現前にはないものを心に浮かべる能力) によるところが大きい。ただし科学的なアプローチで有用なものを創造するためにはモデル (式) が必要である。ひらめきだけの創造は職人や芸術の世界のものである。したがって計画に基づく実験を行って数式 (多くは近似式) を作成しそれに基づいて設計する。そして、近似式による設計は以下のようにアブダクションの一種であると考えられることができる。

[α] 実験という介入により a が起きた (現象)

実験を行ってデータ a を取得した。

[β] 近似式 β で a は説明 (予測) できる (法則)

近似式 β はデータ a を十分に説明している。

[γ] 故に β で a を設計できであろう (仮説)

近似式 β で設計したものは再現するだろう。

設計の基盤となるモデル (数式) はデータに基づいて作成することになる。得られた結果 (観測値である) y に対して、これが出力 (生成) されるモデルは仮説である。これを作成する上で、変数選択が用いられるが、選択方式や選択指標や選択基準・削除基準に関しては決定的なものはない。したがって、関係者の知見を組込んで説明力の高いモデルを構築することが必要である。このモデルは一意に決定することはできず、選択指標や選択基準・削除基準の与え方で変わる。さらに加えて固有技術の知見やケチの原理 (同様に説明できるモデルが複数ある場合は最も単純なものを採用するという考え) も動員してモデルを構築する。このような形で行われる設計というものは予測すなわち仮説なので、その再現性の確認 (仮説検証) を行

うことは不可欠である。

ところで再現性の確認による検証の場合には十分に慎重にならなければならない。特に前提条件、共変量（交絡因子）、実施の順番、サンプリングなどに関する配慮・検討が重要になる。そして忘れてはならないことは、実験データに基づく重回帰の変数選択を用いたモデル化が実験領域の範囲内での想定した次数の多項式による近似でしかないということである。したがって得た模型（モデル）はとりあえず因果関係を便宜的に説明している仮説でしかない。それに基づいて数理計画法で行う最適化の解としての設計もやはり仮説（これでうまくいくかもしれないが、うまくいかないかもしれない）である。したがって、設計という名の仮説の再現性の確認は不可欠の重要な手続きとなる。

2.5 問題と課題の区別と設計

一見似ている2つの言葉である問題と課題の混乱は避けなければならない。両者は「現状を何とかしたい」という点では同じであるが、その本質はまったくタイプが異なり、それ故にアプローチも大きく異なる。

2.5.1 ありのままの姿・あるべき姿・ありたい姿

「ありのままの姿」は現状（あるいは、実態、結果など）と呼ばれる。それを確実に把握することからすべては始まる。これを把握する最初の段階では比較するものを持っていない。

しかし、現状（ありのままの姿）と比較する対象には「あるべき姿（規格、標準、規準）」と「ありたい姿（理想、目標）」の2つがある。これらと現状を比較することによってやるべきことが明らかになる。

- ・あるべき姿と現状の差をなくさなければならない。
- ・ありたい姿と現状の差を飛び越えなければならない。

2.5.2 問題と課題の定義

本項では、QM活動の核となる問題と課題を以下のように定義する。^{[11] . [21] . [23]}

問題：あるべき姿と現状の差

課題：ありたい姿と現状の差

また、問題解決と課題達成を以下のように定義する。

問題解決：あるべき姿と現状の差を埋める。

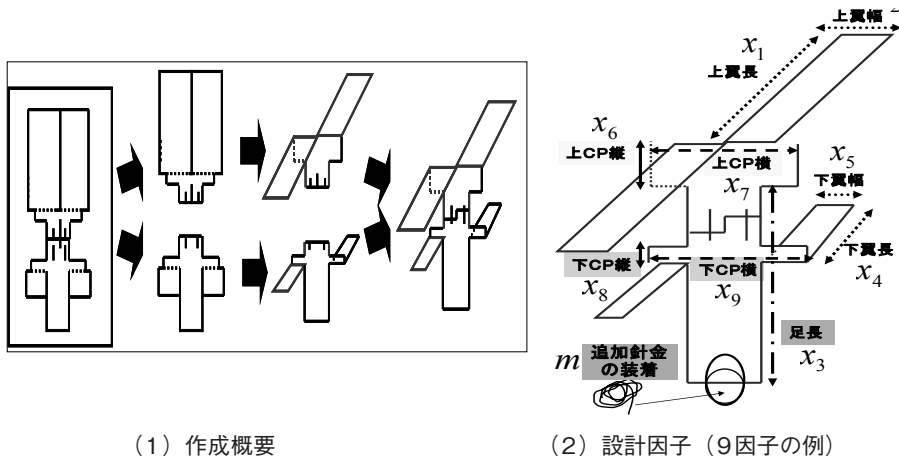
課題達成：ありたい姿と現状の差を飛び越える。

あるべき姿はそれがあべき姿であることをロジカルに示すことで必然のもの・当然のものとなるが、それは願望ではない。一方、ありたい姿はロジカルに示せるタイプのものではなく強く望むものである。したがって、ロジカルに示せない問題は課題（願望）であり、ロジカルに示せる課題は問題（必然）であるという逆説に注意が必要である。

問題には原因があり、それを追究して把握（解析）し、それに手を打って差を埋めることになる。課題には原因はなく、それに到達する道を創造（設計）することになる。前者が差を埋めるのに対して、後者は差を飛び越える（ステップアップする）という点でアプローチは全く異なる。問題と課題の混乱を避けなければアプローチを間違えるリスクが高い。危うい例を挙げると、「売り上げが少ない」という意識だけでは、それは問題なのか課題なのかは分からないのである。実務の取り組みでは問題解決と課題達成を峻別して実施しなければならない。問題解決の典型例は不良低減であり、課題達成の典型例は開発・開拓（新商品開発、新市場開拓）である。本研究が扱うのは課題達成の手段としての設計である。なお、頑健設計は一見すると問題解決に見えるが、原因に手を打つわけではないので課題達成である。

2.6 複葉型紙ヘリコプターとCRZ（顧客要求域）

議論を明確にするために例として図2に示す複葉型紙ヘリコプターを用いる。^[26] そして、マーケット・インの立場で設計する際に重要なCRZ（Customer Request Zone：顧客要求域）を図3に示している。^{[34] . [46]} 設計が顧客満足の中核とする限り顧客要求を無視することはできない。設計は関係者の合意形成ではあるが、そもそも品物は対価を支払ってくれる顧客がいて始めて存在することができるので、何をおい



(1) 作成概要

(2) 設計因子 (9 因子の例)

図2 複葉型紙ヘリコプター

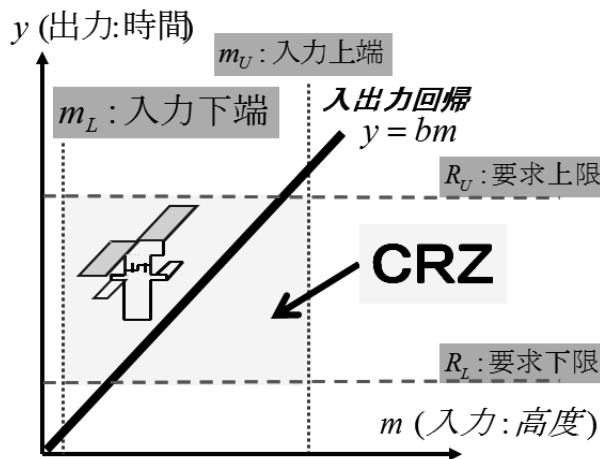


図3 CRZと入出力関数

でも顧客要求を最優先で配慮することが不可欠でこれを必須項目と呼ぶ。その後に関係者にとって重要な項目が配慮され、これらを考慮項目と呼ぶ。なお、入出力回帰は、顧客要求域CRZとともに入出力平面に明示して可視化する必要がある。

1) CRZ (顧客要求域)

良い製品は顧客要求を満たすものでなければならない。この重要な顧客要求を可視化したものが図3のCRZである。製品の本質が機能であり、それが入出力回帰であるとした場合に、顧客の要求は入出力平面に矩形(四角形)として表現することができる。

* 出力に関して

要求下限 R_L と要求上限 R_U

* 入力に関して

入力下端 m_L と入力上端 m_U

この矩形(四角形)が顧客要求であり、製品は必ずこれを満たさなければならない。

なお、ここでは顧客を強く意識してCRZとしているが、のちの5.2節では一般的な議論をするためにCを外したRZ (Request Zone: 要求域)と表現する。

2) 単純な最適化はプロダクト・アウト

もの作りのアプローチにおいてプロダクト・アウトとマーケット・インという2つのタイプ

が存在する。プロダクト・アウトとは「生産者志向で開発・製造・販売していく考え方」で、企業の一方的な立場から、例えば技術者の主義主張・夢・ドグマなどで開発・設計した製品を一方的に売りさばこうとする姿勢のことである。これに対してマーケット・インとは「消費者志向で開発・製造・販売していく考え方」で、消費者が望むものを実現するという姿勢のことである。両者を分ける鍵を握っているのはCRZの把握であり、設計におけるCRZの考慮である。

機能にのみ着目して最適化を行うと、その解（設計）は必ずしもマーケット・インの解ではない。何故ならばその場合の定式化に顧客要求が入っていないからである。顧客要求には様々な項目があるが、そのうちの最重要項目（それ故に設計における必須項目）は機能（入出力回帰）で、これはCRZ（顧客要求域）を必ず満たすということになる。他の要求は考慮項目として追加することにして、ここではCRZについて論じたい。

用いた指標が最適（例えば傾きが最大）であるというだけでは顧客要求を考慮していないので危険である。以下の準備が必要である

- (1) CRZを明示し、
- (2) その上に設計の条件に基づく入出力回帰の図を示した上で、
- (3) 両者の相互関係から設計の妥当性を吟味する。

このとき図4に示すように4種類のパターンが現れる。

- ① [不足] ではまったく話にならない。
- ② [不十分] はやがて不満を募らせる。

③ [過剰] はエンジニアは自己満足するが実は問題なのである。

* 大き過ぎる。 * 重過ぎる。 * コストが高過ぎる。

* 扱いにくい。

* 危ない（速過ぎる、出力が出過ぎる。）

④ [適合] に範囲がある場合には何らかの基準でよいものを選ぶ。

* 目的関数を用意して最適化を図る。

2.7 回帰における階層構造（回帰と超回帰）

数理の観点からみた設計とは設計因子（設計パラメータ）を用いた指標に関する最適化であり、本研究では最適解を数理計画法により求解することを意味する。このためにはモデル化（modeling）が不可欠で、これには数式の構造化としての超回帰（hyper regression）の決定と、設計の構造化としての超機構（hyper organization）の編成の二つがある。なお、超機構の編成とは複数の設計単位を下位の設計単位と上位の設計単位とで階層構造を形成することである。

これらのベースとして超構造（hyper structure）があり、さらには超構造の根源として一般形が多階層の階層構造がある。なお、2階層の場合を超構造と呼び、多くの場合は超構造で十分である。^{[46], [47]}

* 超構造（hyper structure）

階層構造（hierarchical structure）とはある構造の中に更に構造（多くは同種の構造）を内包した状態のことである。この最小形である2階層（2次）の階層構造のもとで、下位の存在

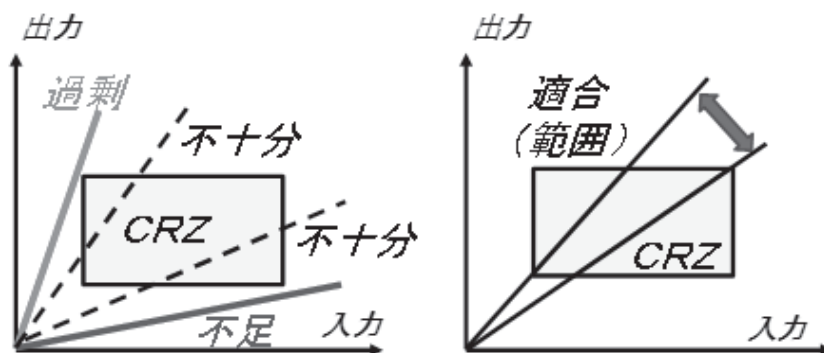


図4 CRZと入出力関数の相互関係

を司る（統率する）上位の存在を超存在と呼び、この構造のことを超構造と呼ぶ。理論的には多階層の構造に発展させることが可能であるが、本研究では2階層（下位と上位）構造に焦点を合わせて議論する。

***超回帰（hyper regression）**

商品とは商いの対象になるもので、対価が支払われることにより顧客に提供されるものである。これにはハードな存在である品物（作り手は製品と呼ぶ）とソフトな存在であるサービスとがある。本研究の考え方やアプローチはいずれにも適用可能であるが、説明を簡潔明快にするために本研究では品物の場合に焦点を合わせる。

①入出力回帰：使用の基盤

品物の使命とは望む（必要な）出力 y をきちんと生成することである。数理的には入力 m を出力 y に変換することであり、これは関数 $y = f(m)$ で表現され機能と呼ばれる。しかし実際には同じ入力に対して常に同じ出力となるわけではなくそこには誤差 ε の存在が避けられない。このために統計的には $y = f(m) + \varepsilon$ と表現される。真の $f(m)$ の正確な把握は困難であり時にはその関数形すらつかめない場合がある。その場合でも入力と出力の対のデータ (m, y) をとることができるので、現実的なアプローチと

してはこの関数を回帰分析に基づいて多項式に近似すればよい。この代表的な例を図5に示している。

②超回帰（入出力回帰の回帰母数の回帰）：設計の基盤

品物を巡って、それを使用する立場と設計する立場があり、両者の間に製造する立場が存在する。まずはクリアに本質を議論するために使用する立場と設計する立場に注目する。設計とは設計因子の集合

$$\mathbf{x} = (x_1, x_2, \dots, x_p)$$

を用いて品物（製品）の機能を決定することである。それは使用回帰の様々な推定回帰母数 t ($b_0, b_1, b_2, \text{RMSE}$ など) を目的変数として、設計因子を説明変数とした重回帰分析により変数選択を行なって設計用の回帰式（正確には重回帰式）を決定（モデリング）する必要がある。最も基本的な式はゼロ点比例式を例にとると式(1)のようになる。

$$y = f(\mathbf{x}, m) + \varepsilon = \beta_1(\mathbf{x})m + \varepsilon \tag{1}$$

以上のことから明らかなように、品物の設計の基盤は超回帰（入出力回帰の係数の回帰）であり、具体的な超回帰式を決定することをモデリング（模型化）と呼ぶ。

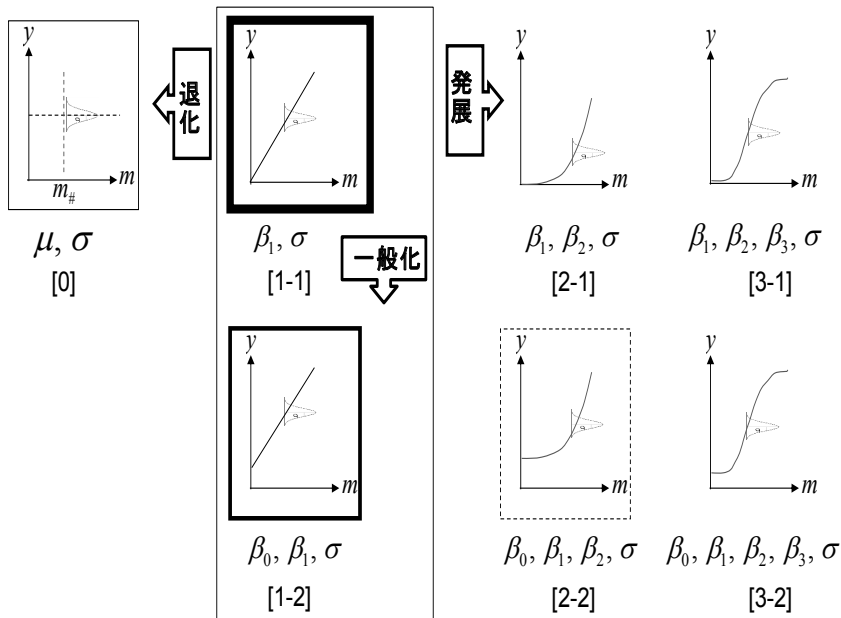


図5 様々な入出力回帰モデル

2.7.1 入出力回帰

図5における最も基本的な形は以下のものである。

$$f(\mathbf{x}, m) + \varepsilon = \beta_1(\mathbf{x})m + \varepsilon \quad (2)$$

これが退化したもの（入力がある固定値 $m_{\#}$ の場合）が以下のもの（ただしゼロ点比例式が退化した場合）である。

$$f(\mathbf{x}, m_{\#}) + \varepsilon = \mu(\mathbf{x}) + \varepsilon = \beta_1(\mathbf{x})m_{\#} + \varepsilon \quad (3)$$

一般形（1次式）では切片 β_0 （入力が0のときの出力）が存在する場合となるので以下のようになる。

$$f(\mathbf{x}, m_{\#}) + \varepsilon = \mu(\mathbf{x}) + \varepsilon = \beta_0(\mathbf{x}) + \beta_1(\mathbf{x})m_{\#} + \varepsilon \quad (4)$$

入出力関数が発展して高次になるといろいろな場合が登場する。図5では2次と3次の場合に関して切片がない場合とある場合を示している。実際には高々2次で十分で、例外的に3次の場合（変曲点のある場合）があり得る（例えば[5]の第7章）と考えている。もしさらに高次が必要な場合には多頭法^[39]を用いると良い。

これらのモデルについて、紙ヘリコプターの場合を取り上げて以下に具体的な例を示す。

①ゼロ点比例式

入力因子を高さ（解放高度）にすると、高さが0の場合の時間（滞空時間）は0で、高さが高くなればそれに比例して時間が長くなる。ただし、機体の出来栄えに問題があったり、飛行装置や飛行方法に問題があったり、部屋の気流の問題（空調設備、窓やドアの開閉など）により原点を通らない回帰になることが少なくないので注意する。そのような場合には、原点を通らない原因を調べて手を打つことで原点を通すか、さもなければ次に述べる1次式を用いる必要がある。ただし、原点を通るようになるまでは1次式を用いるが、原点を通すための努力（通らない原因を追究し、通すための対策を考えること）も並行して進めるのが現実的で望ましいアプローチである。

②1次式

入力因子を追加針金（クリップに針金を丸めて付ける）にするとある程度の重さまでは右下

がりの直線（原点を通らない）に近似することができる。しかし、重さがかなり重くなると線型近似（1次式近似、直線近似）は無理で次に述べる非線形近似（2次式）が必要になる。

③2次式

入力因子を追加針金にして広範囲で重くすると最初は②で述べたように直線的に右下がりの直線で近似できるが、やがてかなり重くなると自由落下の時間（高さのみで決まる時間）に漸近する。このために関数は単調減少の曲線を描くが、追加針金の長さ（重さ）をある範囲に限定するならばその範囲においてこの関数は2次式で近似することができる。

上記の各モデルにおける各々の回帰係数の値は、実験を行なってデータを取り回帰分析を行うことで統計的に把握ができる。その際に、誤差 ε に関してもその分散 σ^2 の値を統計的に把握（推定）することができるのである。しかしその後の使い方（例えば工程能力指数の把握など）として標準偏差 σ の形の方が便利なので本研究ではこちらを用いる。そうすると、機能の本質は回帰母数の組み（ β_0, β_1, σ ）ということになる。なお、データから推定した推定値を用いる場合には（ b_0, b_1, RMSE ）を、そしてこれら推定値の一般的表現の場合には t の記号を用いる。

【注】RMSE：root mean square error

④3次式

紙ヘリコプターでは具体例をあげることはできないが、もし関数に変曲点を持ったり極大値と極小値を持ったりする場合には3次式近似が必要になる。本研究で議論するHOPEのもとでは、2次および3次などの高次モデルは最小二乗法で多項式近似した場合の回帰母数の数が増えるという問題でしかない。

したがって、本研究で主に取り上げるモデルは本質を簡明に議論するために以下に示すゼロ点比例式と1次式の2種類のモデルである。

$$f(\mathbf{x}, m) + \varepsilon = \mu(\mathbf{x}) + \varepsilon = \beta_1(\mathbf{x})m + \varepsilon \quad (5)$$

$$f(\mathbf{x}, m) + \varepsilon = \mu(\mathbf{x}) + \varepsilon = \beta_0(\mathbf{x}) + \beta_1(\mathbf{x})m + \varepsilon \quad (6)$$

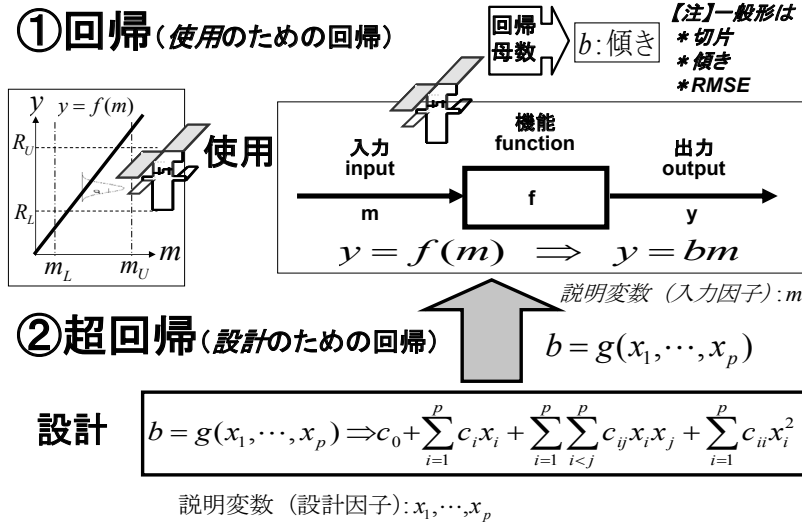


図6 階層構造の回帰：入出力回帰（使用の回帰）と超回帰（設計回帰）

2.7.2 超回帰（入出力回帰の回帰母数の回帰）

本研究ではhyper（超）という概念が基本となる。その一つが図6に示す階層構造である。

設計とは設計因子 $x = (x_1, \dots, x_p)$ を用いて品物（製品）の機能を決めることである。基本的な数式は入出力が1次式の場合以下のようになる。

$$y = f(\mathbf{x}, m) + \varepsilon = \beta_0(\mathbf{x}) + \beta_1(\mathbf{x})m + \varepsilon \tag{7}$$

しかし実際の設計のためには、使用回帰の様々な推定回帰母数 t （具体的には $b_0, b_1, RMSE$ ）を目的変数として、設計因子を説明変数とした重回帰分析により変数選択を行なって設計用の回帰式（正確には重回帰式）を決定（モデリング）する必要がある。以下では、複数の推定回帰母数を一般的に扱うために t と表現する。

$$t = g(x_1, \dots, x_p) = g(\mathbf{x}) \tag{8}$$

すでに述べたように、真の $g(x_1, \dots, x_p)$ の把握は容易ではないのでこれを多変数の多項式に近似するわけである。以下に本書が扱う1次モデルと積モデルと2次モデルを示す。このいずれを採用するかを決定し、そして各係数を決定することがモデリングである。

$$1 \text{ 次モデル} : t = c_0 + \sum_{i=1}^p c_i x_i \tag{9}$$

$$\text{積モデル} : t = c_0 + \sum_{i=1}^p c_i x_i + \sum_{i=1}^p \sum_{i < j} c_{ij} x_i x_j \tag{10}$$

$$2 \text{ 次モデル} : t = c_0 + \sum_{i=1}^p c_i x_i + \sum_{i=1}^p \sum_{i < j} c_{ij} x_i x_j + \sum_{i=1}^p c_{ii} x_i^2 \tag{11}$$

一般に設計因子の変域が狭ければ1次モデルで十分であり、変域が広がると積モデルが必要になり、変域がかなり広がると2次モデルが必要になる。これらの回帰のことを本研究では超回帰（hyper regression：入出力回帰母数の回帰）と呼ぶ。超回帰が把握できればこれを用いて望ましい品物（製品）、すなわち望ましい回帰母数をもつ品物を設計することができるのである。

厳密に数理統計学的な議論では母数 $(\beta_0, \beta_1, \sigma)$ と推定母数 $(b_0, b_1, RMSE)$ を区別して用いる必要があるが、本研究では原則として推定母数 $(b_0, b_1, b_2, RMSE)$ を用いて議論を進めて行く。なお、上記の入出力回帰は顧客にとっての使用の回帰であり、超回帰は設計者にとっての設計の回帰であるとも言えよう。本研究では、このようにして2種類の回帰を階層構造的にとらえて設計を行うところに本質的な特徴がある。

【注】ここでは推定回帰母数を各々ごとに分けて扱

っている。これは説明としてはやりやすいが、この考えは直積型の構造のデータにしか適用ができないという限界がある。この詳細については第6章で議論する。

超回帰においては状況に応じて3種類の多項式に近似する。

- * 1次モデル：1次項（主効果）のみのモデル
- * 積モデル：1次項と積項（交互作用）を合わせたモデル
- * 2次モデル：1次項と積項と2次項を合わせたモデル

一般に設計因子の水準幅が狭い場合には1次モデルで記述が十分な場合が多い。しかし、水準幅が広い場合には積項（交互作用）が必要となるために積項モデルでないと十分な記述が困難になる。さらに、水準幅がかなり広い場合には2次項が必要となるために2次モデルでないと十分な記述が困難になるのである。水準幅を広くとらなければならないのは、回帰母数が広範囲でないと様々な顧客要求を満たすことができないとか、回帰母数が広範囲でないと攪乱因子がパワフルな場合にその影響を十分に減衰することができないといった理由によるのである。したがって、実践的な設計においては1次モデルでは不十分な場合が多く、HOPEによる設計においては原則として2次モデルを見据

え、明らかに2次項がなければ積モデルを用い、明らかに2次項も積項もなければ1次モデルを用いることとする。ただし、高度な実験方法として拡張計画により追加実験で不足している項を追加する方法もある。それは拡張計画（追加実験の計画）を用いれば可能であり、近年のソフトはこの計画の作成を容易にしている。

図7に紙ヘリコプターの例を示している。紙ヘリコプターのモデル化の場合の超回帰では、設計因子の水準範囲によって以下のことが起きる。

- ・水準範囲が狭ければ1次項のみで十分である。
- ・水準範囲が広いと積項が必要となる。
- ・水準範囲がかなり広いと積項と2次項が必要となる。

図7には8つの因子が存在している。これらのうち X_1, X_2, X_3, X_4 の4つにおいて水準範囲をかなり広くした場合には2次項および積項（面積を構成している2組）を必要とする。また、 X_5, X_6, X_7, X_8 に関しても水準範囲を広めた場合には面積を構成している4組の因子の間では積項が必要となる。ただし、実際の変数選択でこれらの全ての項が選択されるわけではない。なお、超回帰においては3次モデルを必要とする場合はほとんどない。入出力の回帰においては、スイッチなどの場合に3次モデル

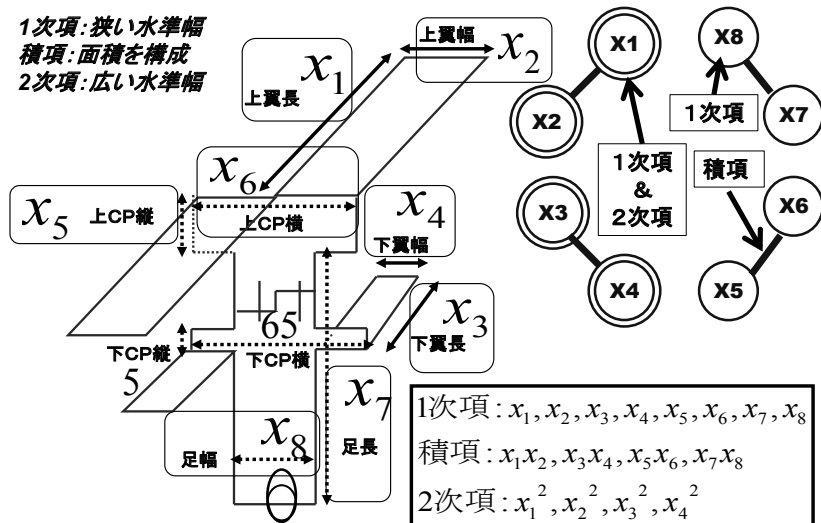


図7 積項と2次項がある場合の例

が登場する。^[5]

* 積項に関して

水準が大きい場合には以下の面積を構成する因子間では積項が必要になることがある。

- ・ 上翼長×上翼幅 ($X_1 * X_2$)
- ・ 下翼長×下翼幅 ($X_3 * X_4$)
- ・ 上CP縦×上CP横 ($X_5 * X_6$)
- ・ 足長×足幅 ($X_5 * X_6$)
- ・ 下CP縦×下CP横 (例では両者は固定している)

* 2次項に関して

サイズがかなり大きくなるにつれて飛行が困難になる因子である翼長と翼幅に関しては水準の範囲がかなり広がると2次項が必要になる。

- ・ 上翼長 (X_1^2), 下翼長 (X_2^2)
- ・ 上翼幅 (X_3^2), 下翼幅 (X_4^2)

2.7.3 直積型超回帰

攪乱因子(制御しない/制御できない因子)が存在する場合、例えば紙ヘリコプターの材料である紙質が何種類もあるという場合をとりあげる。紙質を指定することができたり、受入検査を行なってある条件を満たした紙質のみを受入れるならば問題はないが、もし複数の紙質を受入れる場合には紙質ごとに超回帰が異なる。その場合には2つのアプローチが考えられる。

- (1) 紙質ごとに別々のデザインとする。
- (2) 同じデザインで紙質の違いがあまり生じないものを採用する。

特に(2)のケースは頑健設計とよばれて近年大いに注目されているアプローチである。

さて、(2)の様な場合には実験計画法と回帰を合体した形の直積型超回帰を用いる。ここで紙質が2種類の場合をとりあげて説明する。この場合は超回帰が2つ(紙質ごとに)存在することになるが、それを別々に求めるのではなく、ダミー変数 z というものをを用いて1本の式にすることができる。すなわち2本の式を足して2で割ると式の平均(average)が求まりそれが式(12)の2行目である。いま求めた平均と各々の式との差である乖離(divergence)の式(12)の3行目である。平均の部分に乖離部分を足したり引いたりすると各々の式となる。

足すときは $z=1$ とし、引く時は $z=-1$ という規則で式(12)が作られ、この式1本で2本分の式の代わりになるのである。この構造は多水準になっても、また多攪乱因子になっても本質的には同じで、それらの拡張は容易である。^[5]

$$\begin{aligned}
 t &= A(x_1, \dots, x_p) + D(x_1, \dots, x_p)z \\
 &= a_0 + \sum_{i=1}^p a_i x_i + \sum_{i=1}^p \sum_{i < j}^p a_{ij} x_i x_j + \sum_{i=1}^p a_{ii} x_i^2 \\
 &\quad + \left(d_0 + \sum_{i=1}^p d_i x_i + \sum_{i=1}^p \sum_{i < j}^p d_{ij} x_i x_j + \sum_{i=1}^p d_{ii} x_i^2 \right) z \\
 &\quad z=1(n=1), z=-1(n=2)
 \end{aligned} \tag{12}$$

[注] 確率変数である個々の観測値と平均との差は偏差と呼ばれるが、ここで扱うものは確率変数ではなく母数であり、実験計画法の構造型では効果と呼ばれるものである。しかし、ここで扱うものは明らかに迷惑な効果でありそれは平均からの乖離であるために敢えて乖離と呼ぶ。

本研究ではこの式を直積型超回帰式と呼ぶ。なお、式(12)のダミー変数 z の前の部分(z の係数に当たる部分)である $D(x_1, \dots, x_p)$ は乖離の関数であるが、この絶対値が小さくなるような設計因子(x_1, \dots, x_p)の組合せは2種類の紙の違いを減衰することのできる有用な設計となるのである。水準の数が増えたり、攪乱因子の数が増えた場合には範囲(=最大-最小)を用いてこれを減衰する(範囲を狭める)ことになる。

3. 設計における柔軟性(自由)とHOPE概念

3.1 HOPE

既に述べたように、HOPEとは質経営に基づく設計・企画のための考え方の略称である。過去の質を扱う検査や現在の質を扱う製造(工程管理)である過去現在時制工学(retrospective engineering: 後方視型工学)に対して未来の品質を扱う設計・企画は未来時制工学(prospective engineering: 前方視型工学)となる。そして、多数の関係者や多種の特性・項目を扱う複雑な質の設計・企画では階層構造のもとでの最適化(階層最適化, 2階層の場合は超最適化)が避けられない。これらにどう対応したらよいかの考え方としてHOPEがある。

3.2 目的・目標・理由

HOPEは設計を本質的には柔軟な（自由な）創造活動であると考えている。しかし、思い付きでいろいろな案を作ってその中から選択するという設計はギャンブルとなる。それによる成功例もないわけではないが、多くの場合には失敗に終わることになる。したがってデータを取って式を作り、それに基づいて定式化して求解する工学的アプローチが重要である。

しかしながら、この工学的アプローチのステップを踏む場合に、ややもすると固定的な紋切り型のアプローチになる危険が高い。設計は作業標準を遵守することとは性質が異なり、手順を墨守することは創造と逆の方向に進むことになる。

「確実に」かつ「柔軟に」という、相反することを行うためには、目的と目標と理由の関係が重要になる。最も重要なものは目的でこれは原則として変更しない。もし目的を変更したとしたら、それは別の設計と考えなければならない。しかしながら、他のものは理由（意図を含む）が明らかであれば柔軟に対応すべきである。

1) 理由が明らかであれば目標を変えても良い。

目的が実現できるのであれば目標は変更しても構わない。ただし、恣意的に変えるのではなく、理由を明示しなければならない。

2) 意図的に因子の役割を変えても良い。

経営は顧客要求を合理的に満たしてビジネスを成り立たせることが使命で、工学はこのための手段であり、エンジニアの思いとは別に因子の役割を意図的に変更することは選択肢の一つである。ただし、目的は不変でも因子の役割の変更は必然的に目標の変更を伴うことになる。

3) 止むを得ない場合はシステム自体を変えても良い。

顧客要求の実現という目的に変わりがなければシステム変更も選択肢となる。この場合は因子自体が別のものになり、時には元のものとかかなり異なることもある。ただし、目的が同じであれば全く別の設計と解釈はしない。設計というものを、目的を実現する手段と考えれば様々な選択肢が存在するからである。例として、紙ヘリコプターをやめて紙パラシュートに変更す

る場合がある。目的に関しては空間のある地点から別の地点へ指定された時間で移動することで変更がないため、このシステム変更は受け入れてよい。ただし、システム変更に伴って設計因子も必然的に変更になる。

3.3 3レベルの計画（設計）：戦術・戦略・政略

マネジメントの要諦は計画（設計）で、
計画（設計）= 作戦

は条件設定（定式化の源泉）のレベルで異なる。設計は自由な創造活動であるが、HOPEはそれを3つのレベル（戦術、戦略、政略）に分類する。^[50] 経営の観点から見た設計の3レベルの担当者は以下ようになる。

〔戦術〕：現場の管理者が担う。

〔戦略〕：現場のトップが担う。

〔政略〕：経営陣が担う。

そして、3つのレベルの内容は以下ようになる。

戦術：制約条件と目的関数が所与のもとでの優れた解の獲得

【注】所与の条件下で求解するので一切の交渉は不要で創意工夫が決め手である。

戦略：制約条件を変更して得るかなり優れた解の獲得

【注】対内的な（所属組織内の）交渉（説得）を要するが対外的な交渉は原則として不要である。

政略：全体を大幅に変更して得る格段に優れた解の獲得

【注】対外的な（所属組織外の）交渉が必要となる。

なお、計画（設計）されたものを実現（実行）するのは以下に示す戦闘である。

・戦闘：所与の計画のもとでの行動基準に従った実行

3.4 特性要因図と因子役割図

図8に要因を整理する2種類の図である(1)特性要因図と(2)因子役割図を示している。^[41]
^[42]^[46]特性用要因図は単に特性と多数の因子との関連を見やすく体系的に分類整理しているも

ので、因果関係を示すもの（理学的表現）でありそれ故に設計と直結はしていない。これに対して因子役割図は、因子はどのような役割で特性と絡んでいるかを示すもの（工学的表現）であり、設計と直結する。そして、出力（特性）とともに項目を示すことで因子の経営項目（QCDSE）との絡みを示すもの（経営学的表現）となるので総合的な設計と直結した図になっている。ただし、役割の与え方（付与）は設計する立場の置かれている状況や設計の意図（根源的には経営の意図）に依存する。なお、設計ではプロダクト・アウトの立場とマーケット・インの立場があり、後者の場合にはCRZが不可欠で、これを満たすことは必須条件である。

設計は最終的には製品の諸元（設計因子）を決定することであるが、それらをどう決めるかは置かれている立場・状況で微妙に異なる。プ

ロダクト・アウトの立場で出力中心の設計を行う場合には、図9の（A）に示すようにすべての条件が所与のもとで好ましい解を求解する。しかし、より高いレベルの設計では図9の（B）に示すように、より柔軟なそしてより広範な視野で設計を行う。なお、実験因子とは実験時に計画に従って制御（介入）する因子のことである。以下に重要点を列挙する。

- * 共変量の影響は設備その他で防ぐことができる場合がある。
- * 前提条件は交渉や投資で変更することができる。
- * 実験因子は努力で変更することができる。
 - ・ 因子自体を変更する。
 - ・ 因子の水準を変更する。
- * そもそも因子の役割を変更する。
- * システム自体を変えてしまう。

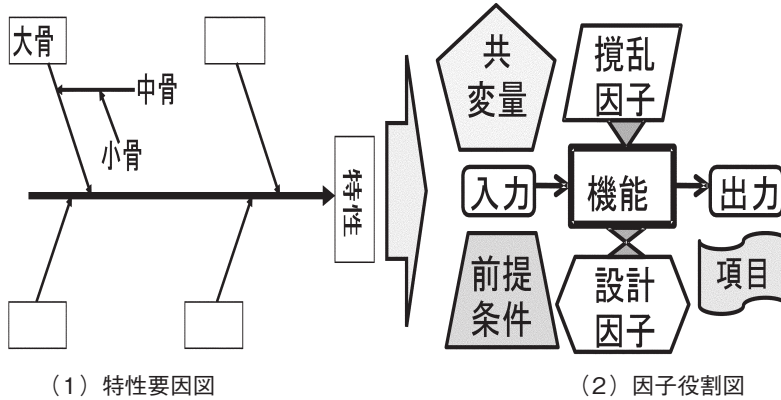


図8 特性要因図と因子役割図

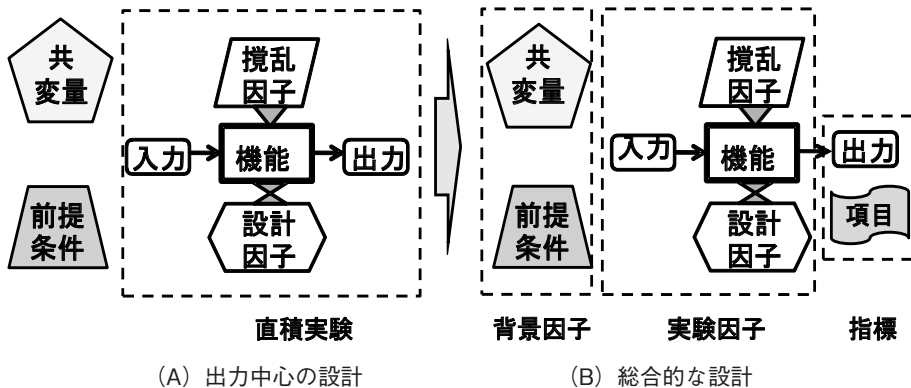


図9 出力中心の設計と総合的な設計

* 他社に作らせてしまう。

設計諸元（製品の諸元）は特性だけでなく QCDSE（品質、コスト、納期・数量、安全、環境）に影響を与える。これらは設計因子の関数で本研究では項目と呼ぶ。したがって、設計においては特性と項目の両方に配慮して求解しなければならない。

【注1】因子役割図は図9に示す実験時のものだけでなく操業時のものも作ると有効である。実験時には実験因子として攪乱因子も設計因子も制御されるが、操業時には攪乱因子は非制御因子となる。そして、設計因子の中には操業時に努力してもばらつく（分布する）ものがあることがある。その場合には、次のような因子役割図を作ると良い。攪乱因子は共変量の枠の中に書き、確実に制御できる設計因子は設計因子の枠の中に書き、ばらつく（分布する）設計因子は攪乱因子の枠の中に書く。この場合の対応は以下の2種類存在する。

操業時にばらつく（分布する）因子に対しては①変量シミュレーションと②柔軟設計（第5章で詳述）という2種類の対応がある。

①これをばらつく制御因子として扱い、そのばらつきのもとでなるべく有利な結果になるように変量シミュレーション（ばらつく制御因子を乱数で振る実験）を行って最適解を調整する。

②これを柔軟設計で攪乱因子にして求解する。

いずれもばらつく設計因子の対応であるが、①はあくまでも設計因子（制御因子）の扱いで、②は攪乱因子の扱いとなる。

【注2】感度分析は母数シミュレーションと考えることができる。平均値の変化でどの程度の影響がでるのかを検討する。これも最適解の調整であるが、ばらつきを想定していない点が特徴である。

3.5 実験因子と背景因子

設計では実験因子（実験時に計画通りの水準に制御する因子）に関心が集中する。しかし、共変量と前提条件は極めて重要なものである。共変量は実験を乱し、前提条件は適用限界を規定する。共変量に対してはその状態を記録しておくことが重要である。そのことで共変量の影響を把握でき、そして共変量の影響を考慮して

よりクリアな分析をすることができる。前提条件はそれを明示することで適用限界が明らかになり誤用（誤った適用）が避けられる。もし、前提条件が異なる設計単位の間で同一の設計を共有したい場合には、第8章で述べる連合設計を適用する。しかし、連合設計でも上手くいかない場合も存在するが、その場合には層別して別々に対応（個別対応）しなければならない。

3.6 設計における戦術・戦略・政略

マーケット・インの立場に立った設計を行う際に、顧客要求を実現するうえで以下に示す点（柔軟設計や連合設計の検討）が重要になる

* 入力因子は何が望ましいのか？

* 入力因子は一つだけで十分なのか？

* 本当に減衰したい攪乱因子は何か？

* 連合設計の必要はないのか？

- ・工場間（国や地域の相違）の減衰
- ・客層間の減衰

実際の設計のステージでは、因子と関数形と水準の3つの設定（条件の定め方・与え方）で事態は変化する。以下に低位のレベルから高位のレベルの設定を示す。

①水準の設定（主に戦術の範囲）

②因子の設定（主に戦略の範囲）

何を取り上げ、どの役割を与えるか。

③関数形の設定（主に政略の範囲）

どのタイプの関数を取り上げ、どの次数の式を用いるか。

【注】高位の設定の誤りや貧弱さは致命的である。

HOPE概念に基づく事後モデル化（柔軟設計）は戦略あるいは政略に該当する。そして、因子役割の設定（割振り、付与）は順列の数だけの可能性が存在し、事前に規定する役割というものはその中の一つでしかない。柔軟設計の詳細については第5章で議論する。

3.7 実験の計画段階での戦術・戦略・政略

実験の計画においても戦略・政略が存在する。それらは以下の状況によって計画を大きく左右する。

①固有技術でどこまで支えられるか？

- ・分かっているものについては実験の必要

がない。

②資金的制約や時間的制約でどこまでできるか？

・時間や資金の制約で回数の大枠が決まる。

③実験の目的・意義（価値）をどう評価するか？

・意義（価値）の評価で回数の大枠が決まる。

実験計画はその後の模型化と最適化に影響する。実験の計画段階でも戦術・戦略・政略が存在することを忘れてはならない。所与の条件下で良い計画を立てるという数学的な実験計画は戦術である。良い実験は計画の戦略・政略を必要とする。

実験の目的・意義と資金・時間・労力の制約から実験サイズの圧縮が必要となることが多い。そのような場合には、蓄積された固有技術の知見と既存データから得た知見を総合的に活用して最適計画（カスタム計画）や過飽和計画を用いて実験サイズの圧縮を検討すると良い。実験サイズの圧縮により計画が非直交となっても重回帰分析の枠組みを用いればモデル化も設計も可能である。計画が直交しているのは望ましい事ではあるが、モデル化や設計において直交性不可欠の条件ではない。

4. 3種類の調和

HOPE概念の設計における基本は調和であり、それには指標間調和と客体間調和と主体間調和の3種類がある。^[46] この議論のために主体、客体、指標を以下に定義する。

*主体：設計を行う者のことである。

*客体：設計で配慮がされる対象のことである。

*指標：指標は設計因子の関数で、特性（出力）と項目から構成される。

【注】特性とは対象のもつ特有の性質でその存在意義そのものに関わるものである。

個々別々に設計するのは特化設計である。様々なものを合わせて設計するのが調和設計である。その際に超最適化（超構造のもとでの最適化）が必要となるが、それには主体が複数存在する場合の連合最適化とそれ以外の総合最適

化と統合最適化連結最適化とが存在する。

4.1 指標間の調和（【総合最適化】）

指標間の調和とは、複数の指標間のトレード・オフを調整して設計を行うことである。指標間の調和においては、指標は以下に示す必須指標、考慮指標、把握指標の3つに分類され、最適化（多目的最適化）が実施される。

1) 必須指標

必須指標は設計の目的の根幹にかかわるものでその定式化の条件を満たすことが不可欠な指標である。したがってマーケット・インの立場から明らかなものは、入出力帰帰がCRZを必ず満たさなければならないということである。顧客の要求を満たさない製品は購入してもらえず、購入されない製品を製造しても経営は成り立たない。一方、製造の立場からは材料の種類間の差は頻繁な段取替えやQA（quality assurance）に影響するので、何としても減衰させなければならない。このように設計の根幹にかかわるものが必須指標となる。これは必ず定式化に組み込んで、設計に反映させなければならない。必須指標にはR（robustness：頑健性）とQ（quality：品質）の2つがある。

①R（頑健性）：攪乱因子の水準間の範囲（ばらつき）

②Q（品質）：攪乱因子の全水準間の平均

この両者は常に必須指標である。①を減衰する（小さくする）ことで頑健になり、②を顧客の要求に合わせることで市場に受け入れられる。

2) 考慮指標

必須指標に準じて重要な意味をもつ指標であり、必須指標を満たすために時には条件を譲歩することになる可能性のある指標である。考慮指標にはCDSE（費用、納期・数量、安全、環境）の4つがある。これらを定式化に際してできるだけ組み込み、設計に反映させることが肝要である。

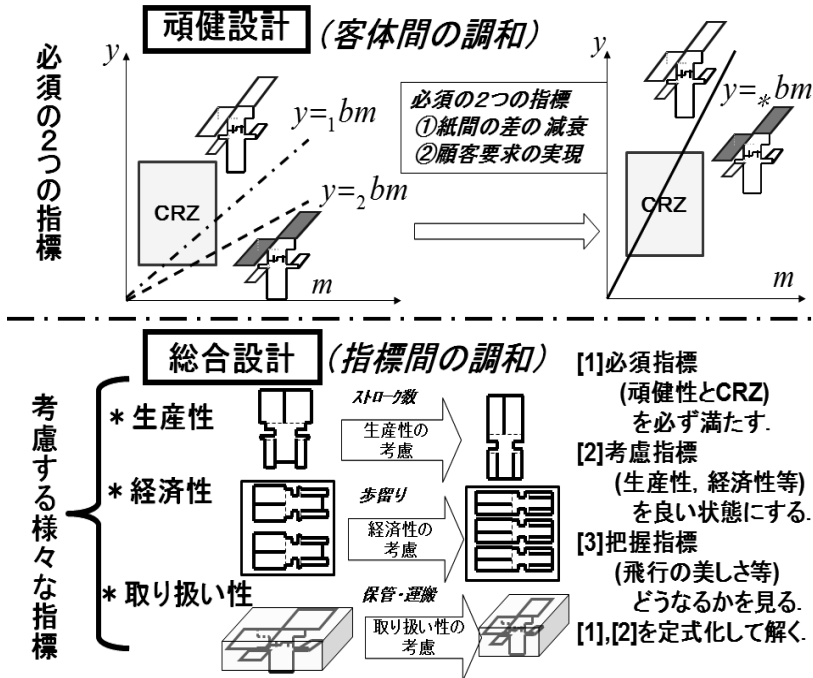


図10 指標間の調和と客体間の調和

3) 把握指標

把握指標は必須指標と考慮指標で設計された結果、どうなるのかを見る指標である。把握指標は考慮指標と同様にCDSEの4つである。考慮指標は定式化に加えられるが、把握指標は結果を見るのみで定式化には加えられない。

把握指標も解の水準の影響を受けるので、その状況を把握するために定式化の画面に登場させてその状態を注視する必要がある。そして、もしクリティカルな状態になった場合には考慮指標に格上げして定式化に加えなければならない。逆に、いろいろトライした結果、明らかにその挙動が問題にならない(十分受け入れられる範囲で挙動している)必須項目や考慮項目は形の上で把握項目にして定式化から外した方がよい。そのことによって計算が早くなるとともに、より有利な解が得られる可能性がある。明らかに問題のない条件を定式化に加えると、そのことが結果的にコンピュータによる求解では足枷となることに注意が必要である。

図10の下半分に考慮指標の例として紙ヘリコプターの設計における生産性、経済性、取り扱い性を示している。いずれの場合も左側の設

計より右側の設計の方が優れていることは図より明らかである。

4.2 客体間の調和 (【統合最適化】)

客体間の調和とは、複数の客体間のトレード・オフを調整して設計を行うことである。客体間の調和には以下のようなものがある。

1) 製造諸元：製造の場で制御できないあるいは敢えて制御しないものが設計における客体になる。

- ・ 素材：原料や材料のことで未加工のもの
- ・ 部品：素材に加工が加えられたもの
- ・ 製法：作業標準、機械・設備ほか
- ・ 環境：温度、湿度、水質、塵埃ほか

2) 使用状況：使用の場で制御できないあるいは敢えて制御しないものが設計における客体になる。

* 使用のTPO：いつ、どこで、どのような場合

* 使用環境：温度、湿度、天候、気圧ほか

上記の1)と2)を調和することは客体間調和設計で、客体の間の差を減衰することになり、これは頑健設計(ロバストデザイン)となる。実際には複数の客体を統合して客体間の差

を減衰することで一つの設計に統合する（共有する）という処理がなされる。

図10の上半分に頑健設計（客体調和）の例として紙ヘリコプターの設計における材料間の乖離の減衰を紹介している。2種類の紙の違いを減衰したうえでCRZを満たす設計が頑健設計である。そしてこれらの必須指標の定式化に加えてすでに述べた生産性、経済性、取扱い性などの考慮指標の定式化も加えたものが、指標間調和と客体間調和を合わせた調和设计である。

一方、関係者である顧客や従業員を客体として配慮した調和（受動的調和）で設計を行うこともできる。しかし関係者の場合にはできるだけ主体として話し合いの場に同席して自らの意思を示し、そのうえで次に述べる主体間の調和（能動的調和）で設計を行うことが望ましい。

4.3 主体間の調和（【連合最適化】）

設計において主体はステークホルダー（関係者）という形で複数になることが少なくない。主体が複数の場合の設計においては主体間の利害に関してトレード・オフが発生するので、主体間の調和が必要になる。

意思決定を行う主体には知識と判断力が必要であるが、主体となり得るものが知識と判断力において不十分であるという場合が少なくない。例えば、使用者と技術者の間には技術や知識の非対称性が存在する。この場合は、知識と判断力が不十分であっても主体となり得るものに対し、情報の理解、発信のための援助をすることで準主体（意思決定はするが対話のために支援が必要な主体）になれるようにし、調和设计に参加できるようにする工夫が必要である。

あるいは組織としての工場は法人格を有しているため、工場間で前提条件が異なる場合に、工場ごとに頑健設計（ロバスト・デザイン）を行うとともに工場間の差を減衰して全社として同一のデザインにするとすることも重要である。この場合には前提条件の異なる工場ごとのモデル（関数）を結合した上で、上位関数を定義して連合設計（複数の下位の設計単位を有する上位の設計単位という超構造の設計）を行う。

調和は対話による逐次最適化で実現することができる。このためにはP D C Aサイクルを回

すことが望ましい。その際、情緒的ではなく事実に基づいた客観的な最適化で合意を形成するにはデータに基づいて式（モデル）を作りそれを用いた最適化を繰り返すことが必要である。

この詳細については第8章で触れる。

5. 柔軟設計（多項式の自由な超構造化に基づく設計）の基礎概念

設計とは目的を実現する方策の自由な最適化である。方策がハードな存在の場合は製品と呼ばれ、商いの場では商品と呼ばれる。方策がソフトな場合はサービスと呼ばれ、これも商いの場では商品と呼ばれる。本研究は方策がハードな存在の場合を取り上げている。

「柔軟設計」とは、因子の役割を多様に交換して多面的に可能性を検討する自由な設計のことである。出力（特性）の実現においてはどの因子も出力に影響するため、どの因子にどの役割を与えるかは人間の都合であって、出力という目的変数から見ればすべての因子は説明変数として対等な存在である。

多変数の複合構造から成る多項式（基盤モデル）を因子の役割に応じて整理して超構造化すると設計のための基本が出来上がる。整理する上では以下の3種類の役割が重要になる。

- ①攪乱因子：入出力回帰を乱すもの
- ②入力因子：出力を制御するもの
- ③設計因子：入出力回帰の係数を決定するもの

まず、攪乱因子の含まれない平均パートと含まれる乖離パートとに整理する。もし攪乱因子がなければ平均パートのみとなる。もし攪乱因子が複数ある場合には、攪乱パートはさらに主効果のパートと交互作用のパートに分かれる。

次に各々のパートで、入力因子の含まれない切片パートと含まれる係数パートに分ける。もし入出力の構造が高次の場合には1次項パート、2次項パートというように必要な次数ごとのパートに分ける。これは入力因子に関する多項式の回帰モデルとしての記述である。本研究で扱う多くの場合は1次の場合である。そして、特殊な場合として高々3次（極大値、極小値、変曲点がある場合）までを意識しているが、理論的にはさらなる高次を扱うことも可能であ

る。なお、入力因子が退化した場合（入力因子の水準が固定された場合、あるいは入力因子自体がない場合）として入力因子がない場合は、定数パートだけになる。

HOPEの本質的な考え方は入出力回帰の回帰係数に設計因子の回帰（超回帰）を2階層の構造化（超構造化）した超回帰（2階層構造の回帰）を扱うことである。これが超構造化の回帰であり、それを攪乱因子（変化する前提条件）で整理して構造化しているのである。なお、超回帰においても原則として積項と2次項も有する多項式を視野に入れている。設計因子の水準が狭ければ1次模型を、少し広ければ積模型を、かなり広ければ2次模型を対象とする。

因子（変数）に付与した役割に応じて整理された式により以後の設計の基礎が固まる。故に役割が変われば超構造化が変わり、そのためそれに基づく設計も大きく変わることになる。したがって柔軟設計は設計における戦略あるいは政略である。

5.1 QCDの三拍子

特性（出力）にとって要因（因子）はいずれも説明変数でしかない。しかし、設計においては多数の要因にどの因子役割（入力因子、設計因子、攪乱因子、前提条件、共変量ほか）を設定する（割振る、付与する）かによって大きく

異なる。そして役割の設定（因子役割の与え方）は、関係者の置かれている状況（技術力、経済力、政治力、顧客の要求、市場の状態、業界の状態、国際関係ほか）に大きく依存する。しかも状況は流動的で時には時間の経過で変えなければならない場合すらも発生する。本稿では、ワンセットのデータから柔軟な設計を試みることのできる実験の計画と因子役割の設定に関して様々な可能性を吟味する方法を提案する。またこの考えに基づく多入力の新たな設計も紹介する。

QMの目指すところは簡潔に言えば、「望ましい質を低コストで迅速に提供し保証する」（QCDの三拍子：“良い、安い、速い”）ということに尽きる。この観点からすれば設計とは総合的質設計（質を中核とした総合的な設計）のことで、本研究はそのための考え方としてHOPEという概念を提案している。

5.2 柔軟設計における目的と目標と理由

すでに3.2で述べたように、最も重要なものは目的でこれは原則として変更してはならない。もし目的を変更したとしたら、それは別の設計と考えるなければならない。しかし、目的が実現できるのであれば役割（因子の役割）を変えても良い。このことはより上のレベル（戦略あるいは政略）で設計を扱うことになる。

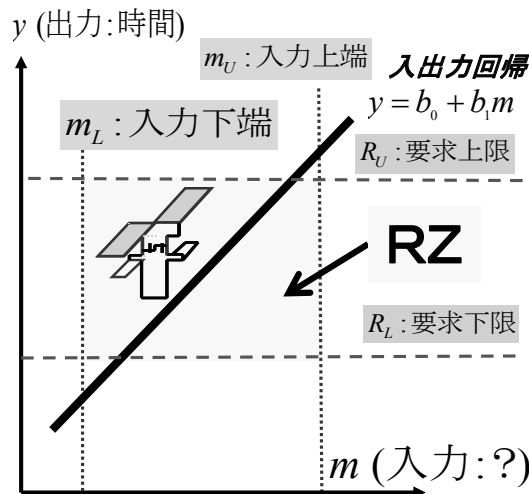


図11 要求域と入出力関数

図11は関係者の要求そのものである縦軸(出力、特性)は一切変更しないが、それを実現するための手段である横軸(入力因子)を変更することは可能であることを意味している。

5.3 実験因子と背景因子

出力(特性) y を生み出すための式(実際には製品が y を生み出すわけであるが、数的に機能とは関数)は関連するすべての因子を説明変数としたものである。その際にはすべての因子は特性要因図上に描かれ互いに対等である。ただし特性要因図は多数の因子を構造的に分かり易く表現するためにとりあえず分類されているが、それは整理や理解のための便宜的なもので y の説明変数としては対等である。しかし、設計では各因子は各々に役割が付与され、そのことにより設計での扱いが異なる。図12に示している役割とは以下のものである。

実験因子

- ①攪乱因子：入出力回帰を乱すもの
- ②入力因子：出力を制御するもの
- ③設計因子：入出力回帰の係数を決定するもの

背景因子

- ④共変量：実験の間状態が変化するもの
- ⑤前提条件：実験の間状態が変化しないもの

【注】もしその状態が変化したら y に大きな

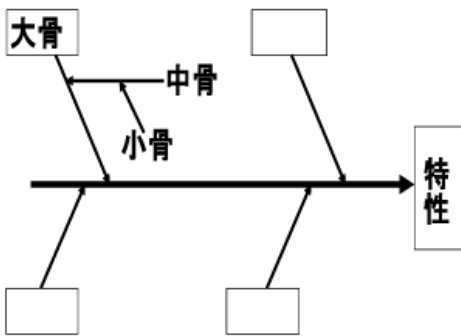
影響を与えるものが対象である。

柔軟設計では④と⑤の背景因子のもとでの①と②と③の実験因子に関してその役割を自由に付与したり、一旦付与した役割を交換することを意味する。

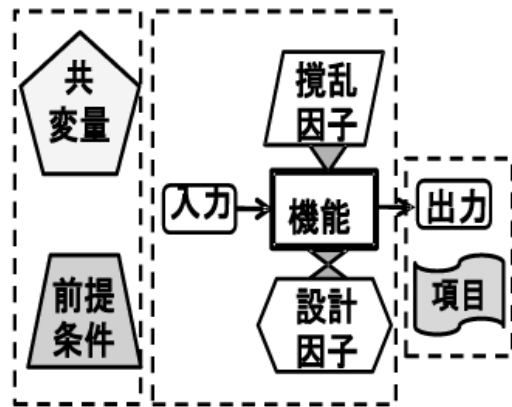
5.4 直接模型化と間接模型化

設計のためにはその基盤となる模型(式、モデル)が必要になる。模型化には出力 y に対して直接的に行う場合と間接的に行う場合がある。^[50] 従来は間接的な模型化が行われてきたが、本研究では直接的な模型化を提案する。この模型化のもとでは様々な形の超構造化を行うことで因子の役割を柔軟に変更していろいろな場合を想定した設計を行うことで柔軟な設計が可能になる。なお、図11では横軸の入力にはどの因子を用いてもよいので?マークを用いて表現している。

模型化とは式を構築することで、基本的には特性である出力とこれに関係する諸因子(入力因子、攪乱因子、設計因子)の間の関係を関数として表すことである。設計とは、これを用いて設計因子の水準を決定することである。この模型化には2つの方法がある。1つは両者を直接模型化する方法である。他の方法はこれに関してある構造を用意し、その構造のもとで統計処理を行い、それにより得られた推定値に対し



(1) 特性要因図



背景因子 実験因子 指標
(2) 柔軟設計のための因子役割図

図12 特性要因図と因子役割図

て推定値ごとに模型化を行なった上で、それらの模型を組合わせて特性の模型を構築（合成）する方法である。両者にはそれぞれ長所と短所があり、目的に応じて使い分ければよい。本章では両者の特徴（長所、短所）を明らかにする。

(1) **直接模型化**：目的変数を特性として全因子を用いて直接的に基盤模型を構築する。

⇒未定役割の設計（柔軟設計）ができる。

すべての因子は特性に対して対等である。

【注】小さな乖離を減衰することが困難になる場合がある。何故ならば、変数選択では小さな乖離は選択されない可能性があるからである。ただし、小さな乖離は減衰しなくても良い（不必要な減衰は有害である）という場合もある。

(2) **間接模型化**：データを一端何等かの統計処理をしたうえで統計量ごとに模型化しそれを組立てて間接的に合成模型として構築する。

⇒既定（指定）役割の設計となる。

間接模型化の場合の代表的な模型化としては以下の2つがある。

(A) **分離模型化**：平均と偏差（ y と平均の差）に分離した模型化後に合成して間接的に全体としての模型を構築する。

直積実験としての構造のもとで、run毎に平均と偏差を求めるために設計因子は既定の役割のもとでのアプローチとなる。

⇒既定（指定）役割の設計である。

平均と偏差ごとに個別に模型化されるので特性への本質的影響に注意する。

【注】平均の影響を絶つために偏差を用いるので小さな乖離を減衰する場合に有効である。ただし、過剰な減衰をする可能性もあり、またそのために平均の設計の自由さが犠牲になることもある。

(B) **係数模型化**：既定次数の回帰の係数ごとにそのrunごとの推定値を模型化しそれを用いて全体模型を構築する。

直積構造および入出力関数の次数が既定のもとで実験のrunごとに各項の回帰係数が求められるので、すべての役割（設計因子、入力因子、攪乱因子）は既定である。

⇒既定（完全指定）役割の設計である。

係数ごとに個別に模型化されるので係数間の特性への本質的影響に注意する。影響力がほとんど無くても採用されるし、影響力が極めて強くてもそのことは考慮されない。

6. 直接模型化と間接模型化

本研究の超構造化の目的は、特定の与えられた役割を担う各因子と他の因子との関係を明らかにするために注目する因子を共通因数として括り出して設計のベースを作ることである。

直接模型化で構築した式（基盤模型）は、その後超構造化で因子の役割を柔軟に（自由に）構成することができる。^[50] このため高度な設計が可能となる。これを本研究では柔軟設計と呼ぶ。柔軟設計においては、最初に直接模型化で構築した式（展開形）を基盤模型（意図的な構造を持たない中立的な模型）と呼び、その後の超構造化で意図的に構成した模型を構成模型（意図的な構造を有する模型）と呼ぶ。以後の議論において超構造化の例を一般形で示す。

因子役割分類の例： m は入力因子、 z は攪乱因子、 $\mathbf{x} = (x_1, x_2, \dots, x_p)$ は設計因子

6.1 直接模型化

6.1.1 基盤模型：直接模型化で構築（役割未定）

この式が得られれば、その後意図的な超構造化により様々な設計アプローチが可能になる。

$$\hat{y} = f(\mathbf{x}, m, z) \quad (13)$$

ただし、意図する超構造化ができるか、超構造化ができて意図する設計ができるかどうかはやって見なければわからない。しかし、可能性はあるのでいろいろと試みるべきである。そして、その試行錯誤の過程で豊富な固有技術的知見を獲得することができる。

6.1.2 構成模型：基盤模型を超構造化（役割付与）

①平均パートと乖離パートの構成

$$\begin{aligned} \hat{y} &= f(\mathbf{x}, m, z) \\ &= k_{(1)}(\mathbf{x}) + k_{(12)}(\mathbf{x}, m) + k_{(13)}(\mathbf{x}, z) + k_{(123)}(\mathbf{x}, m, z) \end{aligned}$$

$$\begin{aligned}
&= a_0(\mathbf{x}) + a_1(\mathbf{x})m + d_0(\mathbf{x})z + d_1(\mathbf{x})mz \\
&= \{a_0(\mathbf{x}) + a_1(\mathbf{x})m\} + \{d_0(\mathbf{x}) + d_1(\mathbf{x})m\}z \\
&= A(\mathbf{x}, m) + D(\mathbf{x}, m)z
\end{aligned} \tag{14}$$

この場合、切片（定数項）における攪乱因子 z の係数部分（乖離パート）および傾き（1次項）における攪乱因子の係数部分（乖離パート）の絶対値を小さくすることが攪乱因子の影響の減衰となる。そして切片および傾きにおいて攪乱因子 z の無い部分（平均パート）を最適化（最大化、最小化、目標化）するのが設計である。

総合的な設計の場合には設計因子で作った関数である経営諸項目（QCDSE：品質、費用、納期・数量、安全、環境）を定式化に加えて求解すればよい。

②切片パートと傾きパートの構成

$$\begin{aligned}
\hat{y} &= f(\mathbf{x}, m, z) \\
&= k_{(0)}(\mathbf{x}) + k_{(12)}(\mathbf{x}, m) + k_{(13)}(\mathbf{x}, z) + k_{(123)}(\mathbf{x}, m, z) \\
&= a_0(\mathbf{x}) + a_1(\mathbf{x})m + d_0(\mathbf{x})z + d_1(\mathbf{x})mz \\
&= \{a_0(\mathbf{x}) + d_0(\mathbf{x})z\} + \{a_1(\mathbf{x}) + d_1(\mathbf{x})z\}m \\
&= B_0(\mathbf{x}, z) + B_1(\mathbf{x}, z)m
\end{aligned} \tag{15}$$

これは式(13)を入力因子 m の有無で整理したものである。これにより切片および傾きの構成がどうなっているのかを把握することができる。

以上の2つの構成方法で明らかのように、どのような見方とアプローチを考えるかで超構造化の構成が変わる。そして超構造化の際のパートの括り方によって表現の違いは出るが、いずれの場合においても攪乱因子の影響の減衰とは以下のことを意味する。

$$|D(\mathbf{x}, m)z| = |d_0(\mathbf{x}) + d_1(\mathbf{x})m| \tag{16}$$

に示すところの乖離パート全体の絶対値を小さくすることであるが、それは以下の形で行うことになる。

$$|d_0(\mathbf{x})| \rightarrow 0, |d_1(\mathbf{x})| \rightarrow 0 \tag{17}$$

実際の定式化では、不等式制約（絶対値がある値以下という制約）にするとよい。減衰の理想は0にすることであるが、それは常に可能ではないからである。そして最適化とは、以下に

示す平均パートを最適化することである。

$$A(\mathbf{x}, m) = a_0(\mathbf{x}) + a_1(\mathbf{x})m \tag{18}$$

これには切片パートと傾きパートがあるので設計の目的に基づいて、総合的な意味で最大化、最小化、目標化（目標にできるだけ近づくこと）する。ただし、平均パートの最適化には以下の二つがあるので、最適化の仕方はいろいろ考えられ、目的にあった最適化をすればよい。

$$A(\mathbf{x}, m) = a_0(\mathbf{x}) + a_1(\mathbf{x})m \text{ の全体としての最適化 } \\ a_0(\mathbf{x}), a_1(\mathbf{x}) \text{ それぞれの個別の最適化}$$

以上で述べた2つの構成法①と②は直接模型化の後の超構造化の違いであって本質的には同じことを行っている。

6.2 2種類の間接模型化

間接模型化はその目的により分離模型化と係数模型化の2つの場合に分かれる。分離模型化の場合には特性を実験のrunごとに平均と偏差に分離してそれぞれに模型化を行なう。係数模型化とは最初に実験のrunごとに入出力の回帰式を推定し、次に各推定値（切片、傾きほか）に対して模型化（超回帰の推定）を行うことである。また、分離模型化の発展形に差分法（基準となるものとの差を模型化して設計し、事後に基準を加えて復元する方法^[43]）がある。そして係数模型化の延長に多頭法（複雑な関数に対しては水準ごとに別々に平均を模型化して設計し、事後に平均に対して関数近似を行って復元する方法^[44]）がある。それらの詳細については別の機会に論じる予定である。

6.2.1 分離接合模型化（平均偏差模型化）

平均と偏差のそれぞれで独立に変数選択をして式を求めた上で、特性値が求められるように接合した式を作成する。この場合に注意しなければならないのは、平均と偏差が別々に変数選択をされるということである。したがって選択結果は当然異なる。偏差に関する選択結果は平均の影響力に関係に行われるので、攪乱因子の変動が平均の変動に比べて相対的に小さい場合に、攪乱因子の変数選択がクリアになるということである。この場合の復元した式は以下の

ようになる。

- ① まずはデータを run (i) ごとに平均と偏差に分ける。

$$y_{(i)j} = \bar{y}_{(i)} + (y_{(i)j} - \bar{y}_{(i)}) = ave_{(i)} + dev_{(i)j} \quad (19)$$

- ② 次に平均と偏差の各々をモデル化する。

$$ave_{(i)} \Rightarrow \tilde{A}(\mathbf{x}, m) = \tilde{A}_0(\mathbf{x}) + \tilde{A}_1(\mathbf{x})m \quad (20)$$

$$dev_{(i)j} \Rightarrow \tilde{D}(\mathbf{x}, m, z) = \tilde{D}_0(\mathbf{x})z + \tilde{D}_1(\mathbf{x})mz \quad (21)$$

- ③ 最後に合成して復元する。

$$\begin{aligned} \tilde{y} &= \tilde{A}(\mathbf{x}, m) + \tilde{D}(\mathbf{x}, m, z) \\ &= \tilde{A}_0(\mathbf{x}) + \tilde{A}_1(\mathbf{x})m + \{\tilde{D}_0(\mathbf{x}) + \tilde{D}_1(\mathbf{x})m\}z \end{aligned} \quad (22)$$

式 (20) から式 (22) の各記号の上にカップ (U) マークがついていることに注意されたい。これは run ごとに平均と偏差に分離し、平均だけで求めた式と乖離だけで求めた式を意味している。平均データで求める場合には乖離パートがなく、偏差データで式を求める場合には平均パートがない。何故ならば偏差はその合計は必ず 0 になるからである。したがって、平均だけで求めたモデルと偏差だけで求めたモデルを合体して復元することで特性のモデルができて上がる。

6.2.2 推定組立模型化 (回帰係数模型化)

run ごとに独立に推定回帰係数を求めた上でそれらを目的変数とした超回帰 (回帰係数の回帰) を係数ごとに求める。得られた係数ごとの超回帰を組立てて特性値を求める式を作成する。以下に 1 次式の場合をとりあげる。

まずは run (i) ごとに回帰係数を推定する。

$$\hat{y}_{(i)j} = \hat{b}_{0(i)} + \hat{b}_{1(i)}m_j \quad (23)$$

次に得られた推定係数ごとに模型化する。

$$\hat{b}_{0(i)} \Rightarrow \tilde{\beta}_0(\mathbf{x}, z) = \tilde{A}_0(\mathbf{x}) + \tilde{D}_0(\mathbf{x})z \quad (24)$$

$$\hat{b}_{1(i)} \Rightarrow \tilde{\beta}_1(\mathbf{x}, z) = \tilde{A}_1(\mathbf{x}) + \tilde{D}_1(\mathbf{x})z \quad (25)$$

最後に組立てて復元する。

$$\begin{aligned} \tilde{y} &= \tilde{\beta}_0(\mathbf{x}, z) + \tilde{\beta}_1(\mathbf{x}, z)m \\ &= \tilde{A}_0(\mathbf{x}) + \tilde{A}_1(\mathbf{x})m \\ &\quad + \{\tilde{D}_0(\mathbf{x}) + \tilde{D}_1(\mathbf{x})m\}z \end{aligned} \quad (26)$$

各記号にチルダ (\sim) マークがついていることに注意されたい。これは run ごとに回帰の推定が行われて得られた推定回帰係数を目的変数とした回帰 (超回帰: 回帰係数の回帰) を意味している。そして、必要な係数 (切片も含む) を用いて m に関する多項式を構成 (合成) することにより特性の式とする。なお、式 (23) から式 (26) では各係数 (回帰母数) を明示しているが、2.7.3 における式 (9) から式 (12) においては一般の推定母数を表す記号である t を用いている。

このアプローチの利点は、それぞれの係数ごとの状況が把握できることと、入出力回帰が高次モデルの場合も扱いやすいことである。ただし、係数の間の相関の問題がある。1 次式の場合には中心化変換により中心化切片にすれば相関の問題については対応できる。

高次になった場合には係数の数が増えるので、係数間の相関の問題が発生する。一つの有効な方法として run ごとに直交多項式のアプローチを行うことが考えられる。これについては別の機会に報告したい。

6.2.3 直接模型化と間接模型化の比較

すでに述べたように、 y を推定 (予測) する方法に直接模型化と間接模型化の 2 種類のアプローチがある。なお、ここでの間接模型化は係数模型化を取り上げる。両者にはそれぞれ特徴があり、長所と短所がある。使い分けることがポイントである。

1) 直接模型化の特徴

まず y を目的変数として、必要な役割構造を記述する。つぎに、超構造化でくくられている階層構造を展開した形で重回帰分析の変数選択でモデルを作成する。これは 1 本の式 (基盤模型) として y を推定することができる。そして、超構造化で階層構造を構成すれば状態が良く理解できる。

柔軟設計のためには直接法が必要である。

* 計画に縛られずに解析・設計ができる。

* 構造が複雑で少し分かりにくく扱いにくい。

* y への寄与が少ない回帰母数は無視され

る。以下に2次式の場合を示す。

$$\begin{aligned}\hat{y} &= f(\mathbf{x}, m, z) \\ &= \hat{\beta}_0(\mathbf{x}, z) + \hat{\beta}_1(\mathbf{x}, z)m + \hat{\beta}_2(\mathbf{x}, z)m^2\end{aligned}\quad (27)$$

2) 間接模型化の特徴

runごとに回帰係数(回帰母数)の推定値を最小二乗法で求め、これを目的変数として各々別々に回帰を求める。この回帰式を超回帰式(回帰係数の回帰式)と呼ぶ。各々の回帰係数に関する超回帰式が求まったならそれらを用いてyの推定式を組立てる(構成する)。この方法ではyが直接推定されているのではなく、超回帰で推定された回帰係数で作成した組立式による推定である。以下に2次式の場合を示す。

$$\begin{aligned}\tilde{y} &= g(\mathbf{x}, m, z) \\ &= \tilde{\beta}_0(\mathbf{x}, z) + \tilde{\beta}_1(\mathbf{x}, z)m + \tilde{\beta}_2(\mathbf{x}, z)m^2\end{aligned}\quad (28)$$

なお、各係数ごとの超回帰は以下のように求められる。

$$\begin{aligned}\tilde{\beta}_0(\mathbf{x}, z) &= \tilde{A}_0(\mathbf{x}) + \tilde{D}_0(\mathbf{x})z \\ \tilde{\beta}_1(\mathbf{x}, z) &= \tilde{A}_1(\mathbf{x}) + \tilde{D}_1(\mathbf{x})z \\ \tilde{\beta}_2(\mathbf{x}, z) &= \tilde{A}_2(\mathbf{x}) + \tilde{D}_2(\mathbf{x})z\end{aligned}\quad (29)$$

間接法は扱いやすいがそのために必要な条件と短所がある。

* 構造が明快でとても分かり易いし扱いやすい。

* 直積型の計画が必要である。

内外の計画自体は直交計画ではなくてよい。

* 各々独立で扱うので、yへの本質的な寄与の程度は無視される。

本稿では同じyに対して3つのアプローチで模型化をして予測をしているが、三者は等価ではないことに注意しなければならない。

$$\hat{y} \neq \tilde{y}, \quad \hat{y} \neq \tilde{y}, \quad \tilde{y} \neq \tilde{y}$$

6.2.4 モデルの不適合について

モデルが不適合の場合、すなわち寄与率が低い場合には以下の点を順に確認する。

①異常値(ミス)

⇒ありえない値やかけ離れた値はないか吟味する。

②因子(この場合は設計因子を意味する)

⇒重要な因子で抜けているものはないかどうか

かについて固有技術との整合を確認する。

③水準の幅

⇒狭いと変化が少なくて寄与率が低くなる。

広いと積項や2次項が必要でそれらを欠くと寄与率が低くなる。

④モデル構造(1次モデル、積モデル、2次モデル)

⇒不足の項(積項や2次項)を補うため追加実験(拡張計画)が必要かどうか確認する。

⑤前提条件

⇒重要な前提条件(因子と水準)を明示して確認する。

⑥共変量

⇒その影響に対応(差し引くなど)するために共変量の状態をできるだけいねいに記録し、それを加えた処理をする。

7. 因子役割の未定と既定

7.1 因子役割の未定・既定と模型化の間接・直接

超構造化に基づいて模型化する場合においては、役割が未定の場合と既定の場合に分かれる。未定の場合には自由に役割を付与することができるが、既定の場合には既に決められた役割を変更できない。ただし、すべての因子の役割が決められていない場合には、決められていない因子については役割を設計時に付与したり変更したりすることができる。

A) 直接模型化: 役割未定

全く自由にすべての因子で柔軟設計ができる。

B) 間接模型化

一端は統計処理をするために一部ないしは全部の役割を規定せざるを得ない。

①分離模型化: 内側役割既定(外側役割未定)

run単位での処理をするために内側は既定される。故に外側の因子のみ柔軟設計ができる。

②係数模型化: 内外役割既定(役割完全指定)

全ての役割を指定しないと回帰係数の推定ができない。したがって柔軟設計はできない。

因子の役割の割振りに絶対的なものはない。これは人間の側の都合と意図によるものである。そして設計のためのモデル化の数理的な本質は超構造化である。因子の役割が発揮できるような構造に超構造化すれば、それに基づいて設計をすることができる。ただし、自由自在な超構造化を行うためには高次の積項（交互作用）が必要である。

多様な積項の構造の場合には最適計画のソフト（たとえばSAS社のJMPではカスタム計画）を用いれば効率の良い実験（必要な構造が吟味できかつ実験数ができるだけ少なくおさえることのできる実験）を簡単に計画することができる。しかし、そのようなソフトが利用できない場合には直積計画を利用して工夫すれば比較的効率の良い実験の計画を立てることができる。この場合のポイントは

- * 内側配置と外側配置の因子（および交互作用や高次項）はすべての積について吟味できる。
- * 積に関して必要なものが吟味（有無の確認）できかつ実験数が少ないものを採用する。

【注】 不必要な内側配置の積項および外側配置の積項の数はできるだけ少ない方がよい。

直接モデル化は重回帰分析の基礎に立っているので変数間の相関の問題は多重共線性がある場合を除いてクリアされる。つまり直交性や直積実験であることが必要ないので適用範囲が広く、データ数の圧縮も可能である。もちろん直交性があれば便利であるし、直積実験のデータであればその後の処理がし易いことは言うまでもない。

しかし、実務的には以下のニーズは高い。

- * 既存のデータを活用する。
- * なるべく少数のデータをとることで設計したい。

これらに対して直接モデル化は応えることができる。

7.2 多入力因子のアプローチ

顧客要求の幅が広い場合には単一の入力因子では実現できないことが起こる。この場合、複数の入力因子を用いることで実現することが可

能になる。本節では多入力因子の基本的な議論を行うために入力因子が2つの場合を取り上げて本質について論じる。なお、これを3入力以上の多入力へ展開することは容易である。

7.2.1 攪乱因子がない場合の2入力 (mとz)

攪乱因子がない場合に、2つの因子mとzを入力因子に割振ることができる。残りは設計因子である。この場合、mとzは入力因子として対等である。これは式(7)の発展的な応用である。攪乱因子というものはその水準により出力のレベルを変えるのでそれ自体を入力として利用することができるのである。以下が同時2入力の式である。

$$\begin{aligned}\hat{y} &= f(\mathbf{x}, m, z) \\ &= k_{(0)}(\mathbf{x}) + k_{(12)}(\mathbf{x}, m) + k_{(13)}(\mathbf{x}, z) + k_{(123)}(\mathbf{x}, m, z) \\ &= a_0(\mathbf{x}) + a_1(\mathbf{x})m + d_0(\mathbf{x})z + d_1(\mathbf{x})mz \\ &= B_0(\mathbf{x}) + B_1(\mathbf{x})m + B_2(\mathbf{x})z + B_3(\mathbf{x})mz\end{aligned}\quad (30)$$

もし2つの入力因子を2段階入力にして第1入力と第2入力とするならば、因子の種類は3種類（第1入力因子、第2入力因子、設計因子）となり以下のようなになる。

$$\begin{aligned}\hat{y} &= f(\mathbf{x}, m, z) \\ &= B_0(\mathbf{x}) + B_1(\mathbf{x})m + B_2(\mathbf{x})z + B_3(\mathbf{x})mz \\ &= \{B_0(\mathbf{x}) + B_1(\mathbf{x})m\} + \{B_2(\mathbf{x}) + B_3(\mathbf{x})m\}z \\ &= A(\mathbf{x}, m) + D(\mathbf{x}, m)z\end{aligned}\quad (31)$$

【注】 2段階入力は自動車のギアとアクセルを用いた加速システムに似ている。第1入力を車のギアのように用いて、第2入力をアクセルのように用いる2段階の入力制御と考えることができるからである。ギアを入れることは最初にダミー変数であるzの水準を指定することにあたる。

今回の場合は少なくとも2因子交互作用が必要で、場合によったら3因子交互作用（式(31)における第4項の部分）が必要になる。より高度な2入力では2次項も必要となるが、その説明については割愛する。

7.2.2 攪乱因子 (v) がある場合の2入力 (mとz),

攪乱因子 (v) がある場合なので割振る因子の種類は2つの入力因子と攪乱因子と設計因子である。2つの入力因子 (mとz) を2段階にして一方をギア入力とすれば因子の種類は4種類 (設計因子, ギア (第1) 入力因子, アクセル (第2) 入力因子, 攪乱因子) となる。この場合の超構造化は以下のとおりである。なお, 構造を分かり易くするために, 基盤模型はvのない場合である式 (15) の構造を活かした表現を用いる。

$$\begin{aligned}\hat{y} &= f(\mathbf{x}, m, z, v) \\ &= k_{(1)}(\mathbf{x}, v) + k_{(12)}(\mathbf{x}, m, v) + k_{(13)}(\mathbf{x}, z, v) + k_{(123)}(\mathbf{x}, m, z, v) \\ &= B_0(\mathbf{x}, v) + B_1(\mathbf{x}, v)m + B_2(\mathbf{x}, v)z + B_{12}(\mathbf{x}, v)mz \\ &= a_0(\mathbf{x}) + a_1(\mathbf{x})m + a_2(\mathbf{x})z + a_{12}(\mathbf{x})mz \\ &\quad + \{d_0(\mathbf{x}) + d_1(\mathbf{x})m + d_2(\mathbf{x})z + d_{12}(\mathbf{x})mz\}v \\ &= A(\mathbf{x}, m, z) + D(\mathbf{x}, m, z)v\end{aligned}\quad (32)$$

$$\begin{aligned}B_0(\mathbf{x}, v) &= a_0(\mathbf{x}) + d_0(\mathbf{x})v, & B_1(\mathbf{x}, v) &= a_1(\mathbf{x}) + d_1(\mathbf{x})v \\ B_2(\mathbf{x}, v) &= a_2(\mathbf{x}) + d_2(\mathbf{x})v, & B_{12}(\mathbf{x}, v) &= a_{12}(\mathbf{x}) + d_{12}(\mathbf{x})v\end{aligned}\quad (33)$$

この場合は少なくとも3因子交互作用が必要で, 場合に拠ったら4因子交互作用も必要になる。なお, 応答曲面モデル (2次のフルモデル) の場合には, 式 (32) にmとzの2つの2次項を加えればよい。

7.2.3 攪乱因子が複数 (v,w) で2入力 (mとz)

以下に示す式 (34) は式 (32) の理論的な拡張を示すものである。ここまで複雑なものは数が少ないと思われる。なお, この場合も構造を分かり易くするために, 敢えてvとwのない場合の式 (14) の構造を活かした表現を用いる。また, 構造を分かり易くするために, 基盤模型はwのない場合である式 (32) の構造を活かした表現を用いる。さらに, 攪乱因子が2因子の場合なので, 攪乱因子間の交互作用も視野に入れる必要がある。

$$\begin{aligned}\hat{y} &= f(\mathbf{x}, m, z, v, w) \\ &= A(\mathbf{x}, m, z) \\ &\quad + D_v(\mathbf{x}, m, z)v + D_w(\mathbf{x}, m, z)w + D_{vw}(\mathbf{x}, m, z)vw\end{aligned}\quad (34)$$

8. 連合設計 (連合模型を用いた逐次対話型設計)

設計の本質は4つの側面からそれぞれ以下のようにとらえることができる。

- *工学的には
メカニズム (機構, 仕組み) とその諸元 (因子と水準) の決定
- *行動的には
制約条件を踏まえた上での自由な創造
- *数理的には
多目的最適化 (数理計画法)
- *運営的には
関係者の合意形成

本章では運営的な観点から設計をとらえ, 複雑な機構 (設計単位, 構成単位) のもとでの階層構造的な設計である連合設計についても論じる。

調和設計において設計単位を考える場合に以下の3種類の調和が存在する。

- ①特性：特性が複数存在する場合には特性間を調和する。(各特性に利害関係者が存在)
- ②ユニット：ユニットが複数存在する製品の場合にはユニット間を調和する。
- ③組織：組織が複数存在する場合には組織間を調和する。

一つの製品に関して重要な特性が複数存在することは珍しくない。これらを調和することが必要となり, その場合には複数の特性を束ねて設計しなければならない。この場合は多特性最適化 (同時設計) となる。特性が一つの製品の場合でも, 製品が複数のユニットから構成される場合には, 製品全体としては複数のユニットを束ねて設計しなければならない。そして, 設計に拘る組織が複数の場合には組織全体として複数の組織が連合して設計しなければならない。

組織の例として, 企業が複数の工場を有して同じ製品を製造する場合が挙げられる。特性とユニットは意思を持たないので束ねたうえでの多目的最適化 (同時設計) で扱うことができる。しかし組織の場合には意思を持っているので, 話し合いが必要になる。この場合の連結を敢えて特別に「連合」と呼ぶ。連合とは, 二つ以上のものが結びついて一つの組織体になることで

あり、その典型的な例としてUN（国際連合）、EU（欧州連合）、UK（英国：グレートブリテン及び北アイルランド連合王国）、USA（合衆国）などがある。本研究の設計の場合では、複数の設計単位が集まって一つの設計単位を構成して設計することが連合設計である。特性の場合もユニットの場合も組織の場合も、数理的には一つの定式化としてまとめたうえで求解（設計）を行う点で三者は同じであるが、組織の連結の場合に限り本研究では敢えて「連合」と呼ぶ。

8.1 階層構造とプラットフォーム

8.1.1 階層構造

HOPEは設計自体に対しても階層構造を適用する。階層構造的設計とは、組織や機構が階層構造を形成するもとの設計のことである。階層構造を無視して個別に設計を行うと局所最適（個別最適）に陥るが、全体を階層構造として扱って全体最適をはかることが重要である。代表的な応用には以下の3つの場合がある。

- ①組織：設計単位間での整合性のある設計
- ②市場：品揃えの低減（嗜好の乖離の減衰）
- ③機械：ユニット数の低減（プラットフォーム化）

このためには以下の考えが必要である。

設計組織・設計機構とは設計の主体と設計の客体を合わせたものである。

- * 主体：誰が設計するのか
⇒設計のPDCAの実施者
- * 客体：何について設計をするのか
- * 設計組織・設計機構：設計主体+設計客体

連合設計（設計単位が複合構造の場合の設計）

設計条件の共有（全工場で同一）の利点大

- ①全社的に生産性が高まる。
- ②全社的にコストが安くなる。
- ③全社的に扱う品数が少なくてすむ。
- ④原材料の調達や在庫管理が楽になる。

しかし、この場合の設計は個々ばらばらの個別設計に比べると構造的な最適化となり、これを本研究では連合設計と呼ぶ。連合設計は設計単位に構造があるとともに多数の項目を取り上げた高度な最適化である。なお設計単位間の差を減衰するように定式化した場合には結果として本質的に頑健設計と同じアプローチになる。

8.1.2 プラットホーム

プラットフォームとは自動車製造の分野ではシャーシー（車台）を意味するが、近年は合理的な経営の観点から、自動車の異なるモデルの間で共通して使われる車台を中心とした基本的な構造（デザイン）の意味で用いられることが多い。すなわち、自社の多車種間でプラットフォームを共有すると生産費用を圧縮することができる。個々の車種別の設計を統合して、多数の車種の間でデザインをシェアするという考えである。ただし、異なる車種間でデザインをシェアしても問題が起きず、多くのメリットがなければならない。連合設計は本質的にこの考えに似ている。

同様の考えがコンピュータの世界でも行われているが、こちらではオープン・プラットフォームという形（戦略）で他社製品との間の共通性・互換性を図っている。多数の会社の製品との共通性・互換性があればそのことにより優位に立つことができるからである。異なるものの間の乖離を減衰して共有化するという点では連合設計の考えと基本的に同じである。

ただし、プラットフォームは初めに広く共通する仕様を用意し、それに合わせるという点で連合設計とは異なる。共有のメリットを実現するという狙いは同じであるが、アプローチは異なっている。連合設計は設計時点で異なるもの間で共有できるものを見出すアプローチである点ではじめに共有すべきものを明示する共有プラットフォームとは逆のアプローチである。

8.2 設計単位と設計の主体・客体・指標

設計には主体と客体があり、誰が設計するのかが主体で、何に対して設計をするのかが客体である。設計の単位とは設計の主体のもとに客体が存在するワンセットの状態である。本研究では複数の設計単位に対して全体を一つの設計として扱う構造的設計のための方法として連合設計を提案している。設計の単位と構造には、たとえば

- * [上位単位] 全社→[下位単位] 複数の工場
- * [上位単位] 工場→[下位単位] 複数のライン

* [上位単位] 市場→ [下位単位] 複数の層といったものがある。これらを構造的に一つのものとして設計するのが連合設計である。

8.2.1 連合設計 (united design) : 全体最適化

設計因子の条件の共有ができるとメリットが多い。生産性が高まり、コストは安くなり、品揃えが少なくてすみ、原材料の調達や在庫管理が楽になる。しかし、この場合の設計は個々ばらばらの個別設計に比べると構造的な最適化となり、これを本研究では連合設計と呼ぶ。連合設計は設計単位に構造があるとともに多数の項目を取り上げた総合的な最適化である。

個々ばらばらに個別最適化 (部分最適化) を行った場合には設計因子の条件を共有することは困難である。もし、連合設計のもとで全体最適化を行うならば設計因子の条件の共有が可能となって合理的な設計が可能になるが、その実施はかなり難しそうな印象を受ける。しかし、近年の進化したソフトを用いれば結合機能 (複数のファイルを結合する機能) などによって簡単に連合設計を行うことができる。

8.2.2 超機構における設計意思

以下に設計単位の議論を行う。設計には設計主体 (設計するもの) と設計客体 (設計されるもの) があり、いずれも設計において構造を形成する。設計主体とは定式化を行うとともに求解の結果を評価し、解を受入れるかそれとも更なる定式化を行うかという設計意思を有する存在であり、複雑な設計では設計主体が構造を有している。これに対して、設計客体とは設計意思を有さないもので、こちらも複雑な設計では設計客体が構造を有している。ここでは設計主体に焦点を合わせて議論するが、設計客体の場合も同様に考えれば良い。両者は直積実験の場合には数理的な構造に関しては同じであるが、定式化と解の評価に意思を表すかどうかの相違であり、設計客体の場合も誰かがその意思を代理すれば設計主体として扱うことができるのである。

8.2.3 連結模型に基づく連合設計

①下位の設計単位

設計にも構造がある。品物の条件やそれを作る工程の条件を決めることが設計であるが、実際の設計には単位がある。例えば、設計する商品のターゲットはいかなる層か (顧客の層: 若者・年配者)、製造するのはどの組織か (製造の組織: 工場A・工場B, ライン1・ライン2) というように下位の設計単位がある。しかし、正確に言えば、次の②で示す上位の設計単位というものを考えなければ、そもそも下位の設計単位というものの自体は存在しないわけである。

下位の設計単位での個別設計は危険に満ちているので注意しないと大きなトラブルやアクシデントを招きやすい。同じ企業で同じものを作っているからといって工場やラインが異なると前提条件 (立地, 職場環境, 設備機器類, 人員の能力やキャリア, 調達する原材料ほか) が異なる。したがって、どこかの設計単位で成功した設計であるからといって、他の設計単位が安易にデッドコピーすることは危険である。

②上位の設計単位

複数の設計単位を司るものとして上位の設計単位が存在することがある。この場合には設計自体が構造を形成する。構造を有する設計の具体例には以下のようなものがある。

* 顧客の層の構造: 市場 (上位) ⇒ 若者層・年配者層 (下位)

* 製造の組織の構造: 企業 (上位) ⇒ 工場A・B (下位)

工場 (上位) ⇒ ライン1・2 (下位)

設計単位 (上位を意識した場合には下位の設計単位) が複数ある場合でも、もし各設計単位が独立に (別々に) 行なう場合であればこれは個別設計が複数あるという単純な状況である。しかし、もしそれらの全体を意識してできるだけ設計因子の条件を共有 (シェア) しようと考ええるならば、下位の各々の単位のことを視野に入れて考慮しながら上位の単位で全体最適を行う必要がある。

8.2.4 人格・法人格と意思表示・意思決定

高度な設計を行う場合には、

①主体 (設計を行うもの)

②客体（設計の対象となるもの）

③指標（設計の手がかりとなるもの）

を体系的に整理しなければならない。本稿は主として主体と客体の構造に焦点を合わせて議論する。客体が人格・法人格を有するもの（意思表示と意思決定ができる存在）の場合には、客体を主体にすることができる。客体が複数存在する場合は、客体間の乖離を減衰するためにこれまで攪乱因子（誤差因子）として扱われることがほとんどであった。

人格を持たない材料や環境などについては仕方がないとしても、人格・法人格を有するもの場合にはその意思を設計に反映すべきである。このためにはそれらを客体として外側に配置するのではなくて主体として扱い設計単位を超構造に構成すればよい。すなわち、客体ごとに独立の設計を準備し、それらを連合して全体の設計を準備するわけである。この場合、上位の設計の中に個々の下位の設計が含まれるために設計構造が超構造となる。本稿では2階層の構造を超構造と呼び、主体が2階層の構造である設計（超構造の設計）を連合設計と呼ぶ。そして、本稿は超構造のもとの最適化である超最適化法HOPEと調和設計におけるその意義及びアプローチに関して、直積配置に上側配置の概念を加えた直方配置を提案して議論する。

前述のように、設計には主体（誰が設計するのか）と客体（何に対して設計をするのか）がある。設計の単位とは設計の主体のもとに客体が存在する一組（ワンセット）の状態である。本稿は設計の単位が複数の場合に、全体を一つの設計として扱う構造的設計のための方法として連合設計を議論する。設計の単位と構造には、たとえば全社→複数の工場、工場→複数のラインや市場→複数の層といったものがある。これらを構造的に一つのものとして設計するのが連合設計である。本稿は調和設計における連合設計を議論する。

8.3 調和設計のための連合型最適化

調和とは、うまくつり合い全体が整っていることを意味する。言い換えれば、いくつかのものが矛盾なく互いに程よい事である。これを目指した設計が調和設計である。設計単位が一つ

の場合であっても、そのもとで複数の客体や複数の指標（特性と項目）が存在することが多い。したがって、主体（意思決定者）が一人であっても、客体間や指標間の調和は不可欠である。

もし、設計単位が複数存在する場合には、更に一段上のレベルでの調和である主体間の調和が必要となる。すなわち、複数の設計単位が連合して一つになり、もとの設計単位が下位の設計単位という超構造（2階層の構造）が形成される。そのもとでうまくつり合い全体が整った形の設計が調和設計における連合型最適化である。これは主体間の合意形成に他ならず、それをデータに基づいた科学的なアプローチで実現しようとするのが調和設計の目的である。したがって、式（モデル、模型）が得られたという前提で以後の議論を進める。

8.4 従属内包型の最適化と独立連合型の最適化^[49]

8.4.1 従属内包型の最適化

人格・法人格を有するものが客体となってしまった場合には意思決定に関与することはできない。あくまでも考慮・配慮される対象であって意思決定をする立場にはない。時には意見や希望をていねいに聞かれることがあっても、それは原則として参考情報（考慮するための情報の一つ）でしかない。実際の設計においては客体が複数ある場合にはそれは外側に配置され、攪乱因子として扱われることになる。客体間の乖離の減衰や全体の平均の調整という点では考慮されるが、自らの意思を定式化に盛り込んで求解するということはできない。結果として決定されたものを受け入れるという受動的立場である。

設計単位が複数存在しても、それらの共通性が高い（立場、歴史、文化、習慣、風土、環境ほかが似ている）場合には、全体を一つにして各々の設計単位をその中に内包された形（攪乱因子＝客体としての扱い）でモデリングを行ったうえで定式化をして求解を行う。これは通常のロバストパラメータデザインとなる。すなわち、攪乱因子が一つ増えた形が多攪乱因子の直積配置となる。

以上を要約すると以下ようになる。

*対象を「客体」として攪乱因子とする。

⇒考慮はされるが能動的意思を発動できない。

*配置は外側配置：攪乱因子として割り付ける。

*外側配置に組込まれ従属した形で模型化され、設計段階では内包のままで設計する。

従属内包型の設計においては、各々の存在は従属した存在であり、あくまでも内包された形のままでことが進められるという形の設計（受動的な設計）となる。

【注】従属内包とは、設計主体であるべきものが上位の単位に従属しその中に内包され、自らの意思を主張できないことを意味する。宗主国に隷属する植民地のような状況を意味する。

8.4.2 独立連合型の最適化

人格・法人格を有するものが主体となった場合には意思決定に関与することできる。この場合には自らの立場に立って意見を述べかつ定式化に加わり、求解の結果に対してPDCAサイクルを回すことができる。

この状況で最適化を行なうと主体が複数存在するならば設計単位が超構造を持つことになる。したがって最適化（設計）の構造も超構造となるが、これは多目的最適化とは異なる。多目的最適化とは、設計単位は一つであるが扱う指標（特性と項目）が複数の場合である。設計単位が複数あってもそれらは単に並列で定式化がなされるのであれば実態としては多目的最適化と変わらない。複数の当事者間の調整は重要であるが連合設計ではない。独立な複数のものがある目的のもとに組織化されたものが連合で、この組織のもとで行う設計が連合設計である。複数の設計単位の上位の設計単位が存在する場合は連合設計（超構造最適化）である。

設計単位の独立性が高い（立場、歴史、文化、習慣、風土、環境ほかが異なる）場合には独立してモデリングを行い連合した形で定式化をして求解を行う。

以上を要約すると以下ようになる。

*対象を「主体」として設計単位とする。

⇒意思を持って能動的に設計に参加する。

*配置は上側配置：攪乱因子（外側配置）としては扱わない。

*設計単位ごとに独立に模型化したうえで設計段階で全体を連合する。

独立連合型の設計においては、各々の存在（設計単位）は独立した存在であり、あくまでも連合してことを進めるという形の設計（能動的な設計）となる。

【注】独立連合とは、設計主体であるべきものが上位の単位に参加はするが、全体としての決定に際して自らの意思を主張できる機構を意味する。EU（European Union：欧州連合）やUK（United Kingdom：連合王国）およびUSA（United States of America：アメリカ合衆国）の状態である。

8.5 直方配置と超構造の設計

従来のロバストパラメータ設計では直積配置が用いられている。これは以下の構造の配置である。

内側配置 × 外側配置
（設計因子） （攪乱因子）

しかし、何らかの事情で連合する必要がある場合には構造的な設計アプローチが必要になる。上位の設計の中に下位の設計が独立性を保持して存在する構造の設計は超構造の設計（連合設計）となる。下位の個々の設計を重視しながらも上位の設計は全体の整合を図る必要がある。

ここで複数の「直積配置」を重層化した構造を考える。本研究はこれを「直方配置」と呼ぶ。直積配置×上側配置（設計主体）

直積配置は2次元構造であるのに対して直方配置は3次元構造となっている。すなわち、直方型データは

内側配置 × 外側配置 × 上側配置
（設計因子） （攪乱因子） （設計単位）

という構造を持っている。なお、この議論において入力因子を外側因子としては扱わない。一連の設計は入出力関数を設計するわけであるから、入力因子は設計される対象として扱う。

図13は4階層の構造をデータ表の形で示している。ここでは以下の構造のもとで議論する。

*顧客は入力因子 m で出力を制御する。

*入出力関数は原点を通り傾きが β である。

- * 傾き β は設計因子の関数である。
- * 設計因子の関数は攪乱因子の水準で異なる。
- * 以上のことが設計単位で異なる。

そして、それぞれの表の上に対応する関数を明示している。ここでの関数表示は内容を簡明

$$y = \mu_{\#} + \varepsilon = f(m = m_{\#}) + \varepsilon$$

No.	翼長	翼幅	足長	#
1	1	1	1	1

(0) 無階層の回帰 (入力が固定された回帰)

$$\textcircled{1} y = f(m|\beta) + \varepsilon$$

No.	翼長	翼幅	足長	100	150	200
1	1	1	1			

(1) 1階層の回帰

$$\textcircled{2} y = f(m|\beta(\mathbf{x})) + \varepsilon$$

内

No.	翼長	翼幅	足長	100	150	200
1	1	1	1			
2	1	1	2			
3	1	2	1			
4	1	2	2			
5	2	1	1			
6	2	1	2			
7	2	2	1			
8	2	2	2			

(2) 2階層の回帰

$$\textcircled{3} y = f(m|\beta(\mathbf{x}; \mathbf{z})) + \varepsilon$$

内

外

No.	翼長	翼幅	足長	薄紙			厚紙		
				100	150	200	100	150	200
1	1	1	1						
2	1	1	2						
3	1	2	1						
4	1	2	2						
5	2	1	1						
6	2	1	2						
7	2	2	1						
8	2	2	2						

(3) 3階層の回帰

$$\textcircled{4} y = f(m|\beta(\mathbf{x}; \mathbf{z}; \mathbf{u})) + \varepsilon$$

上

No.	翼長	翼幅	足長	第1紙						第2紙						
				薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	薄紙	
1	1	1	1													
2	1	1	2													
3	1	2	1													
4	1	2	2													
5	2	1	1													
6	2	1	2													
7	2	2	1													
8	2	2	2													

(4) 4階層の回帰

図13 データ表に表現した4階層の回帰 (内側L8で外側と上側はいずれも1因子2水準)

に議論するために因子役割が既定のゼロ点比例式の場合で示している。なお、出力が特定の値に固定している場合(静特性)は回帰の退化した特殊形として位置づけている。

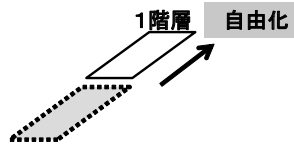
一方、図14の方は同じ内容に関して空間的な意味合いを表現している。この図から、

(内側L8で外側と上側はいずれも1因子2水準)



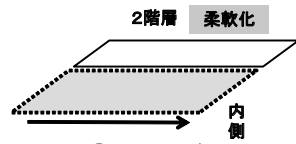
$$y = \mu_{\#} + \varepsilon = f(m = m_{\#}) + \varepsilon$$

(0) 無階層の回帰 (入力が固定された回帰)



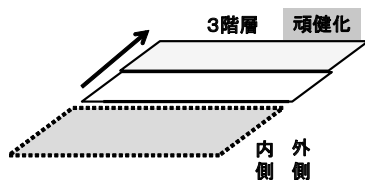
$$\textcircled{1} y = f(m|\beta) + \varepsilon$$

(1) 1階層の回帰



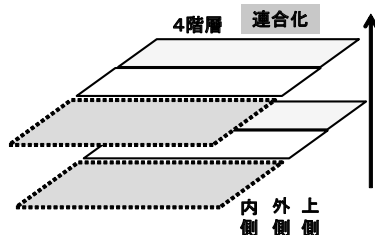
$$\textcircled{2} y = f(m|\beta(\mathbf{x})) + \varepsilon$$

(2) 2階層の回帰



$$\textcircled{3} y = f(m|\beta(\mathbf{x}; \mathbf{z})) + \varepsilon$$

(3) 3階層の回帰



$$\textcircled{4} y = f(m|\beta(\mathbf{x}; \mathbf{z}; \mathbf{u})) + \varepsilon$$

(4) 4階層の回帰

図14 様々な階層の回帰

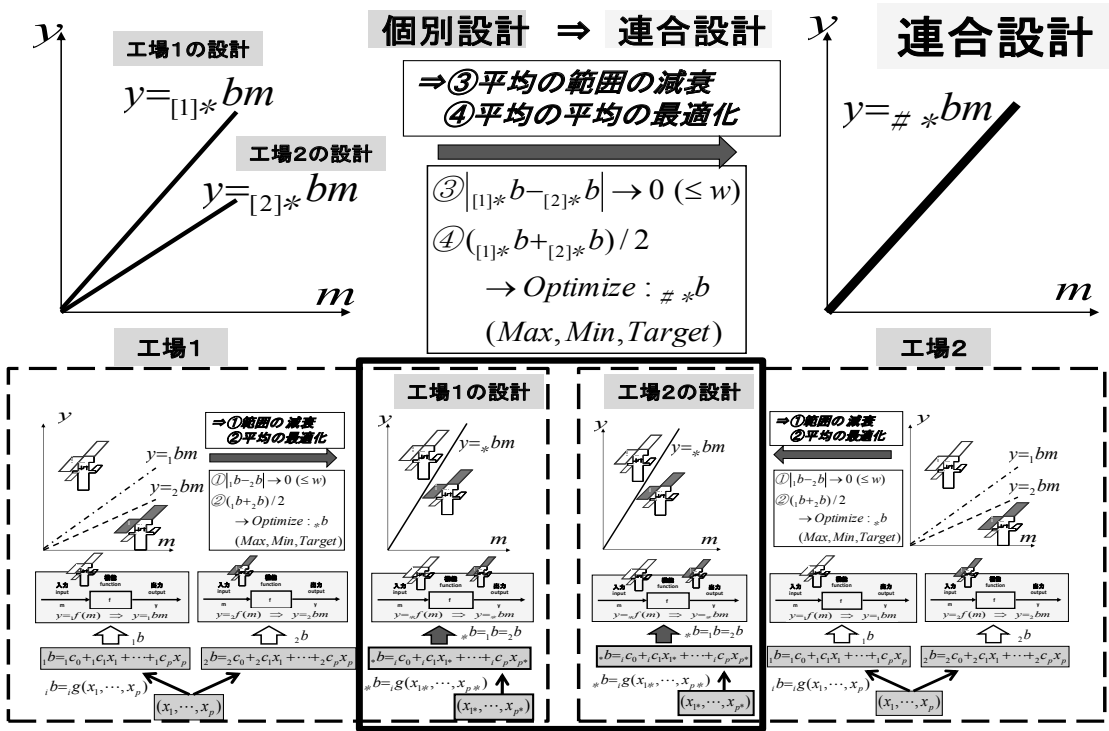


図15 連合設計の概要

固定化→自由化→柔軟化→頑健化→連合化
 という4階層の回帰への発展が視覚的に確認できる。

内側配置 × 外側配置
 (設計因子) (客体：攪乱因子)

この配置は前提として主体が一つの場合である。もし、主体が複数になったとすると、直積配置が主体の数だけ存在することになる。主体が独立であるならば、各々が独自にロバストパラメータ設計(頑健設計)を行えばよい。

8.6 メタアナリシスと連合型最適化(調和設計)

医療および医薬の分野で用いられるメタアナリシス(meta-analysis)とは、過去に独立して行われた複数の臨床研究データを集積して、一段と高いレベルでの解析を行うアプローチである。もともとメタという接頭辞は「より高いレベルの」、「より広い視点からの」という意味がある。この場合は相互に補うことが主目的である。個々では不十分であるが、結合(統合)することで十分なものになるという構造である。しかしながら連合は、各々が個としてはそれな

りに十分であるが上位の目的を実現するために設計単位が参集して合意を形成するものである。

メタアナリシスの類比からメタデザインという概念が考えられる。それは、独立して行われた複数のデザインを集積して、一段と高いレベルでの設計を行うアプローチと言うことになる。しかし、解析(アナリシス)では補うことはあり得るが、デザインの場合には補うのではなく相談して足並みを揃えるのである。したがって本研究ではメタデザインという表現は用いない。しかし、メタモデリングの場合には意義がある。平均パートと乖離パートを有するモデリングはメタモデリングという見方ができる。何故ならば一つの式で構造的にすべての水準の場合の式を表現できるからである。この詳細な理論的考察は別の機会に議論する予定である。

複数の設計単位がある場合にこれらを連合して、上位の単位の設計が下位の単位の設計を包含し、全体を調和させる形で設計が行われる。この点は複数のデータを統合してより質の高い解析を行うメタアナリシスとは本質的に異なる

ものである。すなわち多くの設計を合わせてより確かな設計を行うのではなく、上位の設計の中に下位の設計が存在（内在）するという超構造の設計なのである。したがって、下位の単位の設計を配慮しながら上位の単位の設計を行うという特徴からメタデザインとは呼べない。ただし設計前のモデリング（解析）においてはメタモデリングとして扱うことも十分に考えられる。しかし、これについては別の機会に議論する予定である。

連合設計は多特性最適化として扱うことができそうに見えるが、複数の特性をウエイトをつけて結合して定式化したうえで解くのはあくまでも多特性同時最適化（同時設計）であって連合設計ではない。何故ならば設計単位は一つで特性のレベルにおいて整理しているだけだからである。複数の項目を取り上げる場合も同様に多特性同時最適化（同時設計）である。連合設計とは複数の設計単位が存在しかつそれより上位の設計単位が存在する場合である。横並びの複数のものに対する設計は同時最適化（同時設計）であって連合設計ではない。つまり、一方の単位の水準決定が他の単位の水準決定に影響を及ぼす場合に相互に影響を配慮する設計が同時最適化なのである。しかしこれとは別に、上位の単位がそれ自体の立場からの目的関数と制約条件とともに下位の単位の指標に基づく目的関数や制約条件を有しているという場合が連合設計なのである。

8.7 例を用いた概要説明

超回帰と連合設計について複葉型紙ヘリコプター^{[29], [52]}の例を用いて説明する。実技演習において、2つの班（工場と考える）という下位の設計単位とクラス全体（全社と考える）という上位の設計単位が存在する。各班（各工場）がそれぞれに頑健設計に成功しても両者の機体デザインは同じではない。しかし、クラス全体（全社）として同一のデザインで製作したいという場合を考える。これに対してどのようにアプローチしたらよいかについて本節で議論する。

8.7.1 複葉型紙ヘリコプターの場合の連合設計

図15は連合設計の概要を複葉型紙ヘリコプターの場合で示している。2つの工場がある場合に、各々が2種類の紙質に対して頑健設計を成功しても、2つの工場の機体デザインは異なる。しかし、全社としては市場に同じ機体デザインで供給したい。このような場合には、2つの設計単位で同じ機体デザインにする必要がある。これは設計が2階層の階層構造を成した場合であり、連合設計と呼ぶ。理論的には多階層も可能であるが、本研究では2階層の場合である連合設計を取り上げる。

8.7.2 連合設計における調和設計と特化設計

本来調和設計と特化設計は対極に存在する設計法である。しかし、関係者の合意のもとに、ある特別のものを最優先することで求解し、それが受け入れられた場合には特化設計が結果として調和設計になるという場合もあり得る。

1) 全体最適化

連合設計は個々の設計単位とともに全体を考慮した構造的設計である。図16（定式化と解の代数的表現）および図17（解の幾何的表現）では工場が2つ（教育では実習の班が2つ）ある場合を取り上げている。すなわち設計単位が2つの場合なので3つの設計アプローチが考えられる。

- ①上位最適：2つの設計単位および全体にとって納得のいく望ましい設計にする。
- ②下位最適1：設計単位1を優先した設計を設計単位2も使用する。
- ③下位最適2：設計単位2を優先した設計を設計単位1も使用する。

いずれの場合もHOPE理論を用いているのでヘリコプターには段差（図形における凹凸）がないために作り易い機体形状になっている。ただし②と③では、それぞれ特化して設計した設計単位においては良好であるが、他の設計単位では入出力回帰が大きくずれている。このことから、②と③は個別設計としては良くても、連合設計としては問題であることが分かる。も

①調和設計
上位最適(全体)

②特化設計1
下位最適1(工場1)

③特化設計2
下位最適2(工場2)

条件設定					条件設定					条件設定				
予測変数	目標	以上	以下	推定値	目標	以上	以下	推定値	目標	以上	以下	推定値		
T(範囲, 傾き)	制約	0		0.00029 #	T(範囲, 傾き)	制約	0	0	0.00004 #	T(範囲, 傾き)	なし		0.04217	
T(平均, 傾き)	なし			0.72025	T(平均, 傾き)	制約	0.7	0.875	0.73036	T(平均, 傾き)	なし		0.75131	
T(厚紙, 傾き)	なし			0.72039	T(厚紙, 傾き)	なし			0.73038	T(厚紙, 傾き)	なし		0.73022	
T(薄紙, 傾き)	なし			0.72010	T(薄紙, 傾き)	なし			0.73034	T(薄紙, 傾き)	なし		0.72239	
T.2(範囲, 傾き)	制約	0	0	0.00882 #	T.2(範囲, 傾き)	なし			0.05340	T.2(範囲, 傾き)	制約	0	0	0.00000
T.2(平均, 傾き)	なし			0.70511	T.2(平均, 傾き)	なし			0.74193	T.2(平均, 傾き)	制約	0.7	0.875	0.72091
T.2(厚紙, 傾き)	なし			0.70071	T.2(厚紙, 傾き)	なし			0.71523	T.2(厚紙, 傾き)	なし			0.72091
T.2(薄紙, 傾き)	なし			0.70552	T.2(薄紙, 傾き)	なし			0.76863	T.2(薄紙, 傾き)	なし			0.72091
T(段差①(上翼幅と上CP横)) 制約	0	0		-0.00006 #	T(段差①(上翼幅と上CP横)) 制約	0	0		0.00000	T(段差①(上翼幅と上CP横)) 制約	0	0		0.00000
T(段差②(下翼幅と下CP横)) 制約	0	0		0.00027 #	T(段差②(下翼幅と下CP横)) 制約	0	0		0.00000	T(段差②(下翼幅と下CP横)) 制約	0	0		0.00000
T(段差③(下翼長と足長)) 制約	0	0		0.00004 #	T(段差③(下翼長と足長)) 制約	0	0		-0.00000	T(段差③(下翼長と足長)) 制約	0	0		-0.00000
T(面積)	なし			5851.46885	T(面積)	最小化			5879.05050	T(面積)	最小化			5761.88991
★全最小(傾き)	制約	0.7		0.70071	★全最小(傾き)	なし			0.05340	★全最小(傾き)	最小化			0.05148
★平均間の範囲	最小化			0.01513	★全平均(傾き)	なし			0.73615	★全平均(傾き)	なし			0.73811

設計/制御因子					設計/制御因子					設計/制御因子							
設計/制御因子	設計値	ロック	最小	最大	間隔	設計/制御因子	設計値	ロック	最小	最大	間隔	設計/制御因子	設計値	ロック	最小	最大	間隔
上翼長	* 41.24485		40	70		上翼長	* 41.40636		40	70		上翼長	* 49.83948		40	70	
上翼幅	* 20.04847		20	30		上翼幅	* 21.32249		20	30		上翼幅	* 20.29438		20	30	
足長	* 72.86999		70	80		足長	* 72.40883		70	80		足長	* 74.16115		70	80	
下翼長	* 37.87003		20	40		下翼長	* 37.40883		20	40		下翼長	* 39.16115		20	40	
下翼幅	* 19.58644		15	30		下翼幅	* 19.60405		15	30		下翼幅	* 15.10214		15	30	
上CP横	* 7.5		5	10		上CP横	* 5.351751		5	10		上CP横	* 5.193873		5	10	
上CP横	* 40.097		30	50		上CP横	* 42.64499		30	50		上CP横	* 40.58875		30	50	
下CP横	* 10		5	10		下CP横	* 5.307614		5	10		下CP横	* 9.515036		5	10	
下CP横	* 64.17252		55	65		下CP横	* 64.20809		55	65		下CP横	* 55.20428		55	65	

図16 HOPEにおける連合設計の代数的表現 (調和设计と特化設計)

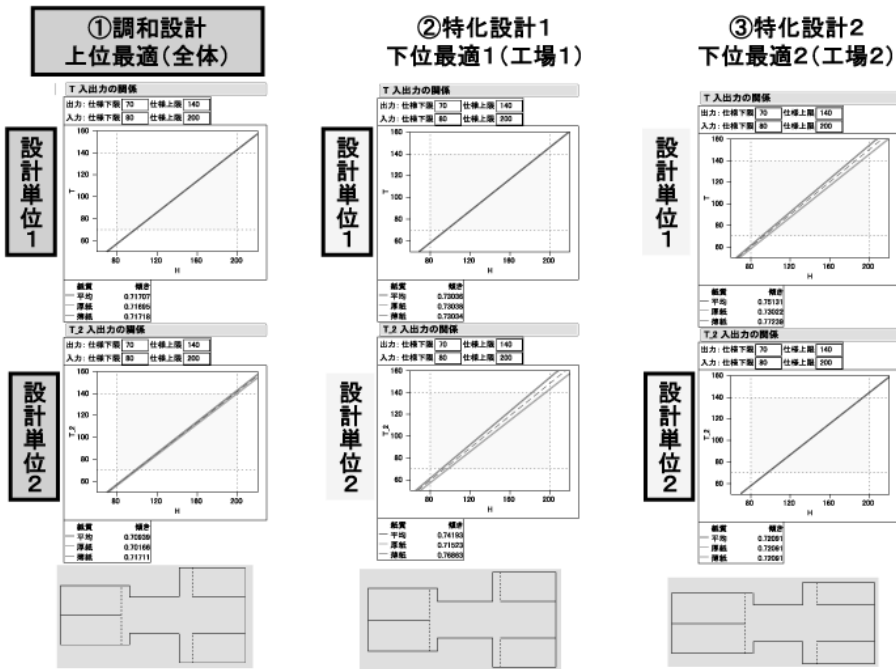


図17 HOPEにおける連合型最適化の幾何的表現 (調和设计と特化設計)

もちろん、2つの設計単位の前提条件をすべて揃えれば良いわけであるが、それは困難であることが多い。①の上位最適とは、2つの下位単位の前提条件を各々ごとにそのままにしたもとの、2つの設計単位における設計因子の条件の共有化を実現している。

以上が連合設計の一つの例である。連合設計は、階層構造をもった設計であるということが

本質であって、定式化そのものには様々な場合が存在する。

2) 総合的かつ連合的最適化 (総合的連合最適化)

図16と図17が示す設計では、より正確に言えば

★連合最適化: 複数の設計単位の全体に対し

て複数の設計単位に対する構造的最適化と同時に

*総合最適化：多種の項目に配慮して多種の項目に対する同時要因最適化を行なっているのである。

この例のように、HOPEは総合的でかつ連合的な最適化を目指している。

8.7.3 解の適用限界

最適解も頑健設計もその適用限界に注意しなければならない。唯一絶対の最適解や頑健設計などは存在しない。それらは以下の2つに依存している。

(1) 前提条件は何か

(2) 定式化はどうしたのか

前提条件という基盤の上での最適化であるから、前提条件が変わったらその後のすべては御破算となる。そして、前提条件が同じであっても、定式化が変われば最適解は変化する。したがって、可能な限り最適解（設計）は用いたデータだけでなく前提条件と定式化をセットでドキュメントすべきである。

連合設計の立場からすると設計単位が複数あれば原則として互いの前提条件は異なっているはずである。したがってある設計単位での最適解を他の設計単位へそのまま適用するのは危険あり、もし適用するのであれば前提条件も同じにすることが不可欠である。安易な適用の危険は特化設計の設計条件を別の設計単位のモデルに入力して計算をした結果図の16と図17を見れば明らかに把握することができる。

この場合の合理的な戦略は各々の前提条件を尊重した（差異を踏まえた）連合設計であるが、その場合でも定式化により解は異なる。何のためにどういう定式化をしたのかが決定的に重要なのである。

9. おわりに

本研究では通常の設計、頑健設計、柔軟設計そして連合設計のための超回帰および数理計画法に基づくHOPEについて論じた。これにより調和を目指した設計が可能であることを示した。

設計とは本来自由な創造活動であり、多数の

因子の役割を自由に検討できることを多くの設計者は望んでいる。本研究はこのための方法として柔軟設計を提案した。そして、昨今は複数の設計単位を視野に入れた設計が登場し、そこでは設計に階層構造が生じるために構造的な設計のアプローチが必要とされている。本研究はそのための連合設計を提案し、例を用いてその有効性を示した。また、顧客の要求域はどんどん広がる傾向にあり、広い出力範囲に対応が可能な多入力も扱えるアプローチも提案した。

本研究の議論を踏まえると、設計（計画）の構造は以下のように整理することができる。

*戦術：所与の条件のもとでの求解

*戦略：部分構造に手を加えたもとでの求解

*政略：全体構造に手を加えたもとでの求解

このもとでは、通常の設計（所与の条件のもとでの求解）は戦術、柔軟設計（役割に手を加えたもとでの求解）は戦略（一部政略）、連合設計（全体構造に手を加えたもとでの求解）は政略と言えよう。

ところで、本研究での提案は質問紙調査（アンケート調査）に対しても応用が可能である。特に市場実験型のアンケートであるコンジョイント分析に有効と考えられる。多くのアンケートではセグメント（層）が問題となる。セグメントごとに分析して最適解を求解するのは品揃えがたいへんになる。もしセグメントを攪乱因子と考えると頑健設計を行なえば合理的な頑健解が得られる可能性がある。仮に全体としての良い頑健解が得られない場合でも、コアとして共通する部分に関してプラットフォーム的に頑健設計を適用することも考えられる。

本研究で提案した様々な方法を実務に応用することが今後の課題である。

【参考文献】

- [1] Joseph, V. R. and Wu, C. F. J. (2002): "Robust Parameter Design of Multiple-Target System", *Technometrics*, 44, [4], 338-346.
- [2] Joseph, V. R. (2003): "Robust Parameter Design With Feed-Forward" *Technometrics*, 45, [4], 284-292.
- [3] 河村敏彦(2011):「ロバストパラメータ設計」,

- 日科技連出版社.
- [4] 河村敏彦, 高橋武則(2013): “動特性のパラメータ設計に対する統計的モデリングと最適化”, 日本品質管理学会誌品質第43巻第3号, pp.102-109.
- [5] 河村敏彦, 高橋武則(2013): 「統計モデルによるロバストパラメータデザイン」, 日科技連出版社.
- [6] 河村敏彦, 高橋武則(2012): 誤差因子に繰り返しがある場合の望目特性に対するパラメータ設計, Research Memorandum No.1157 (統計数理研究所), pp.1-12.
- [7] 木暮正夫(1988): 「日本のTQC」, 日科技連出版社.
- [8] 久保幹雄, 田村明久, 松井知己(2005): 「応用数理計画ハンドブック」, 朝倉書店.
- [9] Kume, H., Takahashi T., et. al (1985): Statistical Methods for Quality Improvement, AOTS.
- [10] Miller, A. and Wu, C. F. J. (1996): “Parameter-design for signal-response systems: a different look at Taguchi’s dynamic parameter design”, Statistical Science, 11, [2], 122-136.
- [11] 宮川雅巳(2000): 「品質を獲得する技術」, 日科技連出版社.
- [12] 宮川雅巳(2006): 「実験計画法特論」, 日科技連出版社.
- [13] Montgomery, D. C. (2001): Design and Analysis of Experiments, John Wiley & Sons.
- [14] Mori, T. (2011): Taguchi methods: Benefits, Impacts, Mathematics, Statistics, and Applications, ASME PRESS, New York.
- [15] Myers, R.H., Montgomery, D.C. and Anderson Cook, C. M. (2009): Response Surface Methodology: Process and Product Optimization Using Designed Experiments (3rd ed.), Wiley, New York.
- [16] Phadke, M. S. (1989): Quality Engineering using Robust Design, Prentice Hall, Englewood cliffs, New Jersey.
- [17] Shoemaker, A. C., Tsui, K. L., and Wu, C. F. J. (1991): “Economical experimentation method for robust design”, Technometrics, 33, [4], 415-427.
- [18] 田口玄一(1976, 1977): 「第3版実験計画法(上)(下)」, 丸善.
- [19] 高橋武則監修(1980): 「応用スキー解析」, WISE PRESS.
- [20] 高橋武則(1986): 「統計的推測の基礎」, 文化出版局.
- [21] 高橋武則(1991): 問題を構成する要素と構造, 品質, 第21巻第2号 pp.23-33.
- [22] 高橋武則(1992): 「統計的方法と管理・改善」, 品質月間委員会
- [23] 高橋武則(1993): 「統計モデルとQCの問題解決法」, 日本規格協会
- [24] 高橋武則(1997): 模擬生産・模擬実験を用いた使命遂行型教育のもとでの回帰診断, 品質管理誌第48巻第1号, pp.86-95.
- [25] 高橋武則(1998): 「模擬生産・模擬実験と統計的品質管理」, 品質月間委員会
- [26] 高橋武則, 田原 慎一郎(2003): “因子数の多い実験のための複葉型紙ヘリコプターの提案”, JSQC第71回研究発表会発表要旨集, pp.219-222.
- [27] 高橋武則(2008): “超回帰最適化に基づく総合的頑健設計—統計的質経営と頑健設計の融合—”, JSQC第86回研究発表会要旨集, pp.95-98.
- [28] Takahashi, T. (2003): “Robust design for mass Production”, Journal of Materials Processing Technologies, 143-144, pp.68-73.
- [29] Takahashi, T. and Saito A. (2005): “Education of Robust Parameter Design by Twin Rotor Paper Helicopter”, Proc. of International Conference on Quality '05 Tokyo, CD proceeding.
- [30] Takahashi, T. (2009): “Robust Design by Hyper-regression Optimization”, Proc. of the 7th Asian Network for Quality Congress, Tokyo, pp.668-677.
- [31] Takahashi, T. (2009): “Quality Design and Evaluation Based on Hyper Structure for Quality Management”, Proc. of the 7th Asian Network for Quality Congress, Tokyo, pp.1502-1511.
- [32] Takahashi, T. (2009): “Robust Design by Hyper-regression Optimization ~ Fixed Output and Free Output ~” Proc. of the 7th Asian Network for Quality Congress, Tokyo, pp.1512-1521.
- [33] Takahashi, T. (2010): “HOPE Theory and JMP Software for Robust Design”, Discovery Summit 2010, pp.1-20.
- [34] 高橋武則(2010): “顧客要求域とPDCA型設計”, JSQC第93回研究発表会発表要旨集, pp.103-106.
- [35] 高橋武則(2010): “複数の特性の結合設計”, JSQC第93回研究発表会発表要旨集, pp.53-56.
- [36] 高橋武則(2010): “HOPE(階層構造最適化)を用いた設計における最適解の摂動的変更による上位の最適化”, JSQC第40回年次大会, pp.109

- 112.
- [37] 高橋武則 (2011): “解像度の高い階層解析”, JSQC第95回研究発表会発表要旨集, pp.233-236.
- [38] 高橋武則 (2011): “調和設計”, JSQC第95回研究発表会発表要旨, pp.237-240.
- [39] 高橋武則 (2011): “頑健設計のための分離型設計”, JSQC第95回研究発表会発表要旨集, pp.241-244.
- [40] 高橋武則 (2011): “HOPEにおける超設計～複数の設計単位に対する構造的設計～”, JSQC第96回研究発表会発表要旨集, pp. 57-60.
- [41] 高橋武則 (2012): “因子役割未定型設計の理論”, JSQC第98回研究発表会発表要旨集, pp. 233-236.
- [42] 高橋武則 (2012): “因子役割未定型設計の適用”, JSQC第98回研究発表会発表要旨集, pp.237-240.
- [43] 高橋武則 (2012): “入出力データの差分への分解と復元”, JSQC第42回年次大会研究発表会発表要旨集, pp. 141-144.
- [44] 高橋武則 (2012): “入出力データに対するベクトル型の超回帰”, JSQC第42回年次大会研究発表会発表要旨集, pp. 145-148.
- [45] 高橋武則 (2012): “調和設計の構造”, JSQC第100回研究発表会発表要旨集, pp.25-28.
- [46] 高橋武則, 鈴木圭介 (2012): “調和設計の企画化とモデル化と最適化”, 目白大学経営学研, 11, 17-43.
- [47] 高橋武則 (2012): “非線形入出力の解析と設計”, JSQC第100回研究発表会発表要旨集, pp.29-32.
- [48] 高橋武則, 河村敏彦 (2012): “多水準をもつ誤差因間の交互作用を考慮した望目特性のロバストパラメータ設計”, Research Memorandum No.1160 (統計数理研究所)pp.1-11.
- [49] 高橋武則 (2013): “調和設計における超設計”, JSQC第101回研究発表会発表要旨集, pp.135-138.
- [50] 高橋武則 (2013): “調和設計における事前モデル化と事後モデル化”, JSQC第102回研究発表会発表要旨集, pp.37-40.
- [51] 高橋武則, 河村敏彦 (2013): “非線型システムに対するロバストパラメータ設計”, 日本品質管理学会誌品質第43巻第2号, pp.85-92.
- [52] 高橋武則 (2014): “模擬試験における認識・創造・経営の教育一質を中核としたデータ・マネジメントの体験教育”, 日本品質管理学会誌品質第44巻第1号, pp.49-57.
- [53] TQM委員会 (2008): 「TQM-21 世紀の総合的質経営」, 日科技連出版社.
- [54] 筑波大学ビジネス科学研究科編 (2003): 「ビジネス数理への誘い」, 朝倉書店.
- [55] 椿広計 (2006): 「ビジネスへの統計モデルアプローチ」, 朝倉書店.
- [56] 椿広計 (2006): “統計科学の横断性と設計科学への寄与”, 横幹第1巻第1号, pp.22-28.
- [57] H.Tsubaki, K. Nishina, S.Yamada (Eds.) (2007): The Grammar of Technology Development, Springer.
- [58] 椿広計, 河村敏彦 (2008): 「設計科学におけるタグチメソッド」, 日科技連出版社
- [59] 東京理科大学工学部経営工学科編 (2005): 「マネジメントサイエンス」, 培風館
- [60] (社)日本品質管理学会編 (2009): 新版品質保証ガイドブック (2009), 日科技連出版社.
- [61] Wu, C. F. J. and Hamada, M. (2009): Experiments: Planning, Analysis, and Optimization (2nd ed.), New York.
- [62] 米盛裕二 (2007): 「アブダクション-仮説と発見の論理」, 勁草書房.

