

## 描写因子による形態設計と極座標描写

### Morphology Design by Depictive Factor and Polar Coordinates Depiction

高橋 武則  
(Takenori TAKAHASHI)

#### 【要 約】

形態とは形状や状態を意味する言葉であり、形態設計とはある目標の形状あるいは状態を実現するために設計因子の水準を決定することである。これを行うためには設計対象の形状や状態が陽関数で描写されることが不可欠である。設計対象が2次元の形態の場合、横軸Wと縦軸yは原因と結果の関係を有していない。両者の本質は座標である。本研究は描写因子による形態設計について議論する。しかしながら、形態の関数は陽関数でないものが少なくない。本研究ではそれらに対して極座標描写により形態設計を可能にする方法を提案する。

キーワード：形態設計，描写因子，極座標描写

#### 【Abstract】

Morphology is the word to mean a shape and a state, morphology design is to decide the level of design factors to realize the aimed shape or aimed state. To describe the shape or the state that are targeted for the design by explicit function is indispensable to perform this. When a design object is in the two-dimensional form, cross axis W and vertical axis y do not have the relations of the cause and effect. The essence of both is coordinates. This paper discusses morphology design by depictive factor. However, many of morphology functions are not explicit function. The approach which enables morphology design by using polar coordinates depiction is proposed in this paper.

Keyword : Morphology design, Depictive factor, Polar coordinates depiction

#### 1. はじめに

本研究は「工学的設計とは数理模型を用いた自由な創造に基づく関係者の合意形成である」というパラダイム（思考規範）のもとで設計を議論する。しかしながら工学的設計の場合に用いる模型（数式）は、理学の場合の模型と異なり真の関数（本質的関数）である必要はない。工学的設計は設計目的に対して必要な区間にお

いて十分な適合性を持つ近似式で構わないというデータドリブン（データ駆動）の立場に立つ。データドリブンとは、データに基づいて判断し行動することを意味する<sup>[9],[10]</sup>。そして、本研究は工学的設計に対して超因子という概念に基づいてデータドリブンの観点から工学的設計について論じる。なお、以後は工学的設計を単に設計と表現する。

本研究は超因子という特別な役割（意味）を持つ因子というものを提案し、それに基づく設計を超設計と呼び、これまでの多種多様なタイプの設計を超因子・超設計という新たな視点から眺めて整理する。そして、その延長線上に存在する新たな設計の可能性について議論する。

超設計において鍵を握る重要な因子である超因子は設計因子の中から選ばれる。そして近似式としてモデル化された多変数関数（以後は基盤関数と表現）を超因子に関する多項式（以後は超構造関数と表現）に構成した上で最適化という形で設計を行う。超因子には多種類の役割がありどの因子を超因子にするのか、超因子の数をいくつにするのか、そして超因子にどのような役割を与えるのかは設計者の自由である。事前に超因子とその役割を決めて設計することも多いが、事後に超因子とその役割を決めて設計することも可能であることが超設計の重要な特徴である。それどころか、超因子の役割に関して事前に予定していた役割があった場合でも、事後に自由に変更することも可能である<sup>[5]</sup>。

本研究は設計で扱う基盤関数が高次の積項を有するかなり複雑な場合を取り上げる。これは直積実験を用いれば構成することができる。あるいは同形式の実験を複数行って結合しても構成することができる。そして、そのような場合における幾つかの設計アプローチを整理する。その上で、とくに形態設計を取り上げて、これに関して新しい方法を提案する。

形態とは形状や状態を意味する言葉で、形態設計とはある目標の形状あるいは状態を実現するための設計因子とその水準を決定することである<sup>[8]</sup>。これを行うためには設計対象の形状や状態の描写が不可欠である。この描写に用いられる因子を描写因子といい、多くの場合それは時空間（時間、空間）の変数である。その代表的な例を以下に示す。

\* 熱源を持たない保温ポット内の温度の時間推移

横軸が時間で縦軸が温度

※時間が原因で温度が変わるわけではない。

\* 円形の穴をあける切削加工の穴径の形状

横軸が角度で縦軸が穴径

※角度が原因で穴径が変わるわけではない。

本研究では最初超設計というアプローチを明らかにする。次に複雑な関数形の場合の設計方法について議論するとともに、閉じた図形に関する設計方法について提案する。最後に超構造関数が高次の場合に多重の合成関数を用いた設計方法について議論する。

## 2. 超設計とは(基盤関数, 超因子, 超構造関数)

### 2.1 基盤関数

実験データで推定された基盤関数は超設計のおおむねのモデル（数式）である。この段階では全ての設計因子は独立変数（説明変数）として対等である。

$$y = f(\mathbf{v}), \quad \mathbf{v} = (v_1, \dots, v_{p+1}) \quad (1)$$

この関数をそのまま用いて、望む $y$ の値の実現可能な設計因子の水準の決定を行うというのが通常の設計である。本研究はその先の進化した設計法として超設計を提案する。

### 2.2 超因子と超構造関数

複数の設計因子の中の一つないしは複数を超因子 ( $H_A, H_B, \dots$ ) という特別な因子に指定した上で、これに注目して構成した多項式が超構造関数である。以下に最も基本的で易しい構造であるところの超因子が唯一 ( $H$ のみ) でかつ超因子 $H$ に関する超構造関数の次数が1次の場合を示す。

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \lambda_1(\mathbf{x})H, \quad \mathbf{x} = (x_1, \dots, x_p) \quad (2)$$

式(1)から式(2)への移行は以下の数理的な構造である。

$$\mathbf{v} = (v_1, \dots, v_{p+1}) \rightarrow (H, \mathbf{x}), \quad \mathbf{x} = (x_1, \dots, x_p)$$

これが0次（超因子がない）の場合のもとの設計が通常設計である。そして、次数を高次にすることや複数の超因子を指定して多変数の高次式にすることが可能である。しかしながら、本章では最も簡単な1次の構造の場合で超設計の本質を説明する。なお、超因子に指定されなかった他の設計因子は超構造関数においては係数関数 $\lambda_0(\mathbf{x})$ と $\lambda_1(\mathbf{x})$ となる。そして超設計

を定義すると以下ようになる。

「超設計とは超構造関数を用いて設計因子の水準を決定して設計目的を実現することである」

超設計の結果として設計因子の値（水準）が決まることで係数関数の値が決定し、そのことにより最終的に超構造関数が確定する。

### 2.3 超因子の4つのタイプと超設計の役割

上記の説明では超因子を一般形の記号であるHで示したが、超因子の代表的なタイプには以下の4つがある。数学的な構造は同じであるが工学的な意味（超因子の役割）は大きく異なるので、それぞれの特徴を表すために超因子を表現するための記号を使い分ける。

- ①入力因子M：これで $y$ を制御する（望む値にする）。
- ②攪乱因子Z：これがばらつく $y$ が大きくなる。
- ③組織因子U：この水準ごとに $y$ に関する式が異なる。
- ④描写因子W：これが $y$ と組みになることで対象の形態（形状や状態）を描写する。

※描写因子の場合にはWと $y$ の両者の間には因果関係はないので、あえてWを $W_1$ とし、 $y$ を $W_2$ とした方が描写因子の意味が分かり易い。そこで第4章においてはWを $W_1$ とし、 $y$ を $W_2$ とした表現を採用する。

以下に各々の超構造関数（数式）と意味の概要を紹介する。係数（定数項と傾き）には各々名前を付けている。また係数の添え字にはその意味するところの英語の頭文字を用いている。

- ①入力因子による制御設計：その都度自由に望む出力を実現する設計（構造：切片パートと傾きパート）

$$y = F(M; \mathbf{x}) = b_0(\mathbf{x}) + b_1(\mathbf{x})M \quad (3)$$

$b_0$  : inetercept,  $b_1$  : slope

- ②攪乱因子による頑健設計：攪乱の影響を減衰して出力 $y$ の安定を確保する設計（構造：安定パートと乖離パート）

$$y = F(Z; \mathbf{x}) = s(\mathbf{x}) + d(\mathbf{x})Z \quad (4)$$

$s$  : stable,  $d$  : divergent

- ③組織因子による連合設計：組織間で条件共有の合意を形成する設計（構造：平均パートと固有パート）

$$y = F(U; \mathbf{x}) = a(\mathbf{x}) + p(\mathbf{x})U \quad (5)$$

$a$  : average,  $p$  : particular

- ④描写因子による形態設計：目標の形態（形状、状態）を実現する設計（構造：独立パートと関連パート）

$$y = F(W; \mathbf{x}) = l_0(\mathbf{x}) + l_1(\mathbf{x})W \quad (6)$$

$l_0$  : unlinked,  $l_1$  : linked

上記の4つの数式は数学的にはまったく同じである。しかし、超因子の役割によって第1項と第2項に役割（意味）を与えると、実際の設計（工学的設計）においてそれらの意味は大きく異なる [5], [6], [7]。

### 2.4 直積の構造：パラメータに落とし込めば設計は可能

前節で議論したことをまとめると以下のようになる。

**【設計因子】 × 【パラメータ】 という直積の構造に落とし込めば超設計は可能である。**

図1に2種類の直積型の表を示している。左が直積型のデータ表で右が直積型のパラメータ表である。先ず実験を行うことにより左側の直積型のデータ表を作成する。次に実験No.ごとにパラメータ（偏回帰係数や回帰誤差）を推定する。そして、推定したパラメータの値で右側に示している直積型の表を作成する。

[注] 設計の対象には偏回帰係数とともに回帰誤差もある。偏回帰係数は略して係数と呼ぶが、回帰誤差は回帰係数ではなく回帰パラメータである。したがって、前者は単独で係数と呼ぶが、前者と後者を合わせた場合にはパラメータ（正確には回帰パラメータ）と呼ぶ。

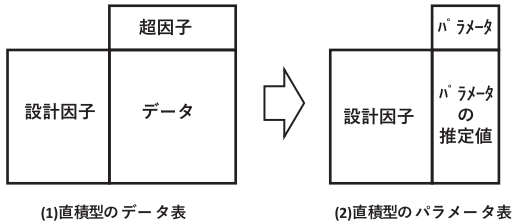


図1 2種類の直積型の表  
(データをパラメータに置き換える)

設計という観点から直積構造を考えると以下のことが重要になる。

\* 外側配置は設計されるもの：超因子⇒パラメータ

\* 内側配置は設計するもの：設計因子

外側配置をパラメータの組（多次元のパラメータ）に置き換えることができるのであればどのような複雑な関数でも設計を行うことができる。しかしながら、内側配置は何としても高々2次式におさえたい。何故ならば、外側の超因子に関する多次元のパラメータを設計するためのモデルを獲得するためには実験が必要で、そのための構造モデルは実験の規模と扱い易さおよび実実験の制約（主として実験数からの事情）から高々2次式の近似が現実的であるからである。ただし、シミュレーションの場合にはしばしばこの制約は緩和される。しかし、シミュレーションの場合でも複雑なものは計算にかなりの時間がかかるために、高々2次式におさえたいという場合も少なくない。そのような場合にはテイラー展開の数式構造が示すように、範囲を狭めれば高々2次式におさえることが可能になる。

### 3. 原形法・区分法・差分法・多頭法

ここで式(2)を再掲する。

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \lambda_1(\mathbf{x})H, \quad \mathbf{x} = (x_1, \dots, x_p) \quad (7)$$

これは超設計におけるもっとも基本的な1次の超構造関数である。前章ではこれを用いて超設計の基本的な考え方を述べた。しかしながら、現実には以下に示すような高次関数のタイプの超構造関数が少なくない。

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \sum_{i=1}^k \lambda_i(\mathbf{x})H^i \quad (8)$$

それどころか、ある程度高次の関数を用いてもうまく近似ができないタイプの超構造関数も多数存在する。そこで本研究はこれらのやっかいな超構造関数の扱い方を議論するために、図2に示すように関数の扱い方を大きく4つに分類する。そして、それぞれに対してアプローチする方法を以下に示す4種類の名称で呼ぶことにする。

- (1) 原形法：元データ  $y$  に対して多項式の表現をする。
- (2) 区分法：全体を区分して個々に原形法を適用する。
- (3) 差分法：差分 ( $d = y - \text{目標値}$ ) に原形法を適用する。
- (4) 多頭法：個々の水準ごとに関数を求める。

#### 3.1 原形法

対象の関数の形状が原型（そのまま）で多項式によって十分に描写（近似）が可能な場合には、そのままモデル化をすればよい。図2 [A]には典型的な例を3次まで示している。次数は4次以上であっても構わない。要は多項式で十分な近似描写できればその係数（パラメータ）を用いて設計することができるのである。

#### 3.2 区分法

##### 1) 区分による簡易化

全体としては複雑な形状でもそれを幾つかに区分すれば部分は簡単な形状になる場合が少なくない。多くの形状は分解すると以下のものに帰着できるので、高々3次で合成することができる。

\* 直線：1次式（線形式）

\* 単調な曲線：極値のない非線形式

\* 単調でない曲線：極値や変曲点のある非線形式

・ 極値（極小値, 極大値）：2次式あるいは3次式

・ 変曲点：3次式

なお、区分した場合は設計時においては区分したものをすべてを同時に取り上げて一緒に定式化を行うことが原則である。ただし、重要な一部だけを取り上げて設計するというアプローチも選択肢の一つである。この場合取り上げな

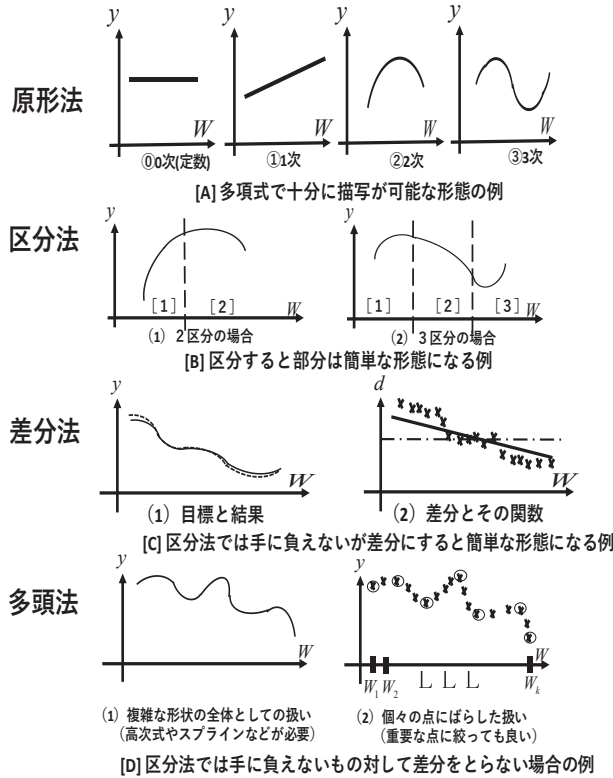


図2 様々な超構造関数

かった部分の様子も求解の結果の最適条件のもとで(最適解を代入して)どうなっているか確認をした方がよい。

図2 [B] の(1)は全範囲を2次で近似するのは精度が粗過ぎるので2つに区分してそれぞれに2次を当てはめればよいという場合である。図2 [B] の(2)は3次で近似するのは精度が粗過ぎるので3つに区分してそれぞれを2次、2次、2次で近似するのがよいという場合である。

ただし、区分した場合は隣接する2つの区間に同じ点(両者の端点)が境界点としてまたがる。その点は、前の区間では上端で、後の区間では下端となる。そして推定式で計算すると前の区間の上端の値と後の区間の下端の値は多くの場合に異なる。両者は本来同じ点なので、両者を等しいとする等式制約をおくか、あるいは両者の差の絶対値がある値以下であるという不等式制約をおく必要がある。

## 2) 事後の全体の合体による復元と模型化

解が得られたら全体を一つの関数として復元する。区分ごとに求められた関数ではあるが、すでに模型化はできているので重要な点(始点、終点、極大値、極小値、変曲点、変化の激しい部分の点など)について計算によりその推定値を求めることができる。区分の境界点(隣接する2つの区間にかかわる端点)は両方の点の値の平均値を用いる。このようにして推定した点の全体を近似する関数を以下の方法で求めて模型化すればよい。

\* 多項式で近似

\* スプライン関数で近似

これらの詳細については紙数の都合により割愛する。

## 3.3 差分法

### 1) 区分法でも手に負えない模型化

形態が複雑な場合には、区分数がかなりの数を必要とすることになるため区分法では手に負

えなくなる。しかし、もしも目標形態がある場合には、それとの差分をとる合理的なアプローチが可能になる。目標形態がかなり複雑なものであったとしても、差分は簡単な関数で描写できることが少なくないからである。そのような場合には差分を用いて設計を行う。そして、設計後は差分に目標形態を足し込んで本来の関数として復元すればよい。

運悪く、差分をとってもそれ自体が複雑な場合には差分に対して区分法の適用を検討する。あるいは差分に対して次に紹介する多頭法の適用を検討とするとよい。

## 2) 事後の足し込みによる復元

最適化はあくまでも差分に対して行われるので、差分に関するモデルが得られたら、それにより推定した差分の値に対応する位置の目標形態の値を足し込んで復元する。この場合には最適化された差分に目標形態を加えるだけで復元ができるので、区分法のように推定した値を用いてそれらを通る関数を求めるという更なるモデル化の必要はない。

## 3.4 多頭法

### 1) 差分法で扱えない模型化

すでに説明したように、原形法で対応することが困難な場合は、以下の2つの対応を用いることになる。

\* 目標関数があれば差分法を用いる。

\* 目標関数がない場合は区分法を用いる。

もし目標関数がない場合は差分をとることができないので、その場合には必然的に区分法となるわけである。しかし区分法でも困難な場合(多数の区分が必要な場合)には最後に多頭法を検討することになる。

目標関数がない場合には差分をとることができないので、対象の形態をそのまま描写しなければならない。その場合に形態が複雑なものであるならばそれらを点の集まりととらえて、個々の水準ごとに点としてばらして扱うことが考えられる。これは形態を点の集合として扱うわけなので、形態をベクトルととらえていることになる。そして点ごとに設計因子の関数を用意するので関数のベクトルということもでき

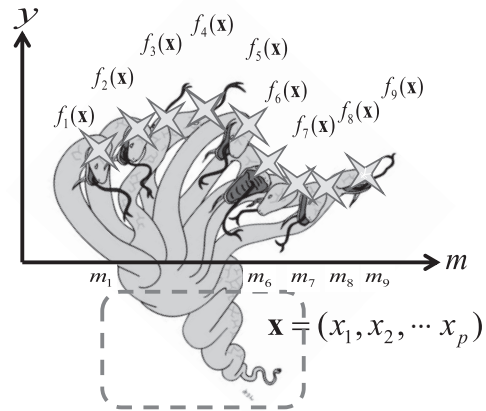


図3 ヒュドラと多頭法の関係  
(多数の点は全て同じ設計因子を共有)

る。

ここで扱う関数のベクトルは設計因子が同じであるという特徴を持っており、これは以下のように考えると多頭生物に似ている。

\* 全てが同じ設計因子という胴体を共有する。

\* 各々の点を頭と見れば多頭(多数の頭)である。

多頭法の多頭とは八岐大蛇(日本神話における多頭の蛇)やヒュドラ(Hydra:ギリシア神話における多頭の蛇)のように、胴(設計因子)は一つ(共通)で頭(点)が多数という意味からの命名である。図3にヒュドラと多頭関数の関係を示している。もしも関数形状が複雑であった場合には、必要な(重要な)個別の点を複数とり、各々を設計因子の関数として模型化する。図3には9点をとった場合を示しているが、これらはいずれも同じ設計因子を説明変数(独立変数)とした関数である。こうすることでかなり複雑な形状の関数も簡単に扱うことができる。ただし、これで扱えるものは陽関数に限られることに注意しなければならない。そうでない関数(陰関数ほか)を扱うことはできないので、このような場合のアプローチについては次章で議論する。

## 2) 事後の全体の模型化

多頭法の設計は設計因子を共有したもとの個別の点に関する同時定式化による多特性最適

化である。これで設計を行った後で、得られた解（各々別々の関数により推定されている）に対して全体を統合した形態（全体を一つの関数でモデル化したもの）をきちんと求める必要がある。したがって、得られたすべての点を用いて全体を一つの数式でモデル化するわけであるが、その方法は区分法で必要な点を推定した後のモデル化と同じように以下のアプローチを用いる。

- \* 多項式による近似
- \* スプライン関数による近似

### 3.5 要所区間・要所点をおさえたモデル化

実務における設計では全ての箇所を対象にせずとも要所の箇所をおさえればよい場合が少なくない。すなわち区分法の場合は要所の区分を、多頭法の場合は要所の点をおさえればよい。このアプローチはスプライン曲線（関数）と本質的に似ている。スプライン曲線の由来であるスプラインとは、バネ性のあるしなやかな金属製あるいは薄い木や竹製の細く長い定規のことである。スプラインは弾性変形を利用することにより、必ず滑らかな形になるという特長がある。指定された点（制御点）を通る実物のスプラインに対応する伝統的スプライン曲線とは異なり、B-スプライン曲線は制御点を必ずしも通らないスプライン曲線である。制御点を必ずしも通らないという特徴はある意味で最小二乗法に本質的に似ている。要所区間・要所点をおさえたモデル化は少ない区間・少ない点でモデル化が可能になる点で有用なアプローチである。ただし、要所区間・要所点の選択は固有技術情報に基づいたものでないと設計後の量産段階や使用段階で実務的なリスクが生じる可能性がある。

## 4. 高度な形状（平面図形、立体図形）を扱う応用的な設計アプローチ

設計の重要な分野の一つは形態設計である。製品の形状の設計や製品の状態の設計である。この場合には様々な曲線が存在するので柔軟な対応が必要となる。前章では単一の陽関数を前提として基本的な議論を行った。しかし、単一の陽関数であっても複雑な関数の場合には工夫が必要となる。そのまま設計ができる原形法

が適用できない場合には区分法、差分法、多頭法あるいはこれらの組合せについて検討するとよい。本章ではそのアプローチでは対応することができない場合を議論する。

図形は曲線という形をとるので、これは開いた曲線と閉じた曲線に分類される。単一の陽関数とは開いた曲線であつ一つの関数で表現できるものである。そうではないものは工夫して単一の陽関数に落とし込むことが重要である。

[注] 最終的には明快な関数形とそのもとでのパラメータに落とし込めれば設計は可能である。しかし、閉じた曲線の多くはそのようなことが困難である。この点、開いた曲線は仮に複雑な形状であっても区分法、差分法、多頭法あるいはこれらの組合せで何とかすることができる。

このためのアプローチとして本研究は極座標描写を提案する。その場合には原点の決め方が重要で、これには最適化の方法として距離平方最小と距離偏差平方最小という2つの方法がある。これらを活用して単一の陽関数に落とし込んだら、前章で述べた4つのアプローチで超設計を行う。

複雑で高度な形状の場合の設計の本質は、「適格な関数形を定めてその係数（パラメータ）を把握（推定）して設計する」というものである。ただし、高度な図形の多くは陰関数であるが、その場合でも関数形と係数が定まれば問題はない。なお、陰関数とは多変数関数（しばしば多変数多項式） $f$ により $f(t_1, \dots, t_n) = 0$ の形の関数のことであり、設計の場合には描写因子の関数の係数が設計因子の関数となる。

### 4.1 陰関数表示の係数（パラメータ）による描写

描写因子が一つ（これを $W$ と表すが、正確には $y$ と合わせて二つの描写因子となっている）の場合の形態設計は以下の高次多項式構造で扱うことができる。

$$y = F(W; \mathbf{x}) = l_0(\mathbf{x}) + l_1(\mathbf{x})W + l_2(\mathbf{x})W^2 + \dots \quad (9)$$

しかし、よく用いられる形状であるところの円、楕円などは2次元の陰関数の描写となり、

球（球面）、楕円面、円柱などは3次元の陰関数の描写となる。このとき、描写因子 $\mathbf{W}$ は原因ではないということに注意しなければならない。 $\mathbf{W}$ は $y$ と組んで形態（形状、状態）を描写する座標なのである。なお、現実の設計が扱うのは平面図形と立体図形であるので、因果関係ではない形状を扱うこと強調するために、以後は2次元の $(y, \mathbf{W}_1)$ を $(\mathbf{W}_1, \mathbf{W}_2)$ に、3次元の $(y, \mathbf{W}_1, \mathbf{W}_2)$ を $(\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3)$ にというように表現することにする。

以下に示す陰関数の一般的記述では簡潔な表現のために、標準形を前提として係数の数は最小限にしている。なお、実際の場面では位置のずれ（平行移動）も問題になることがある。そのような場合には以下に示すように

$$\mathbf{W}_1 \rightarrow (\mathbf{W}_1 - c_1), \quad \mathbf{W}_2 \rightarrow (\mathbf{W}_2 - c_2)$$

位置のずれを記述するためのパラメータとして $c_1, c_2$ を増やせばよい。しかし、ここでは説明を簡単にするためにこれらのパラメータは用いない。

$$\begin{aligned} \text{2次元(平面)図形: } & F(\mathbf{W}_1, \mathbf{W}_2; \mathbf{x}) = 0 \\ & \rightarrow F(\mathbf{W}_1, \mathbf{W}_2; f_p(\mathbf{x}), f_q(\mathbf{x})) = 0 \\ \text{3次元(立体)図形: } & F(\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3; \mathbf{x}) = 0 \\ & \rightarrow F(\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3; f_p(\mathbf{x}), f_q(\mathbf{x}), f_r(\mathbf{x})) = 0 \end{aligned}$$

上記の式は各係数が設計因子の関数となっているために複数の超因子（描写因子）を有する超構造関数である。

#### 4.1.1 2次曲線（平面図形）

円錐曲線（円錐断面）とは、円錐面を任意の平面で切断したときの断面として得られる曲線群の総称である。代表的な図形を図4に、対応する数式を式（10）に示している。

なお、円は楕円の特殊形であり、放物線は多項式（陽関数）で表現することが可能である。したがって、純粋に陰関数のタイプで必要となる図形は楕円と双曲線の2つということになる。

①円:	$W_1^2 + W_2^2 = r^2$ $(pW_1^2 + pW_2^2 = 1)$
②楕円:	$pW_1^2 + qW_2^2 = 1$
③放物線:	$pW_1^2 + 2qW_2 = 1$ (2次多項式) $Y = c_0 + c_1X + c_2X^2$
④双曲線:	$pW_1^2 - qW_2^2 = 1$

(10)

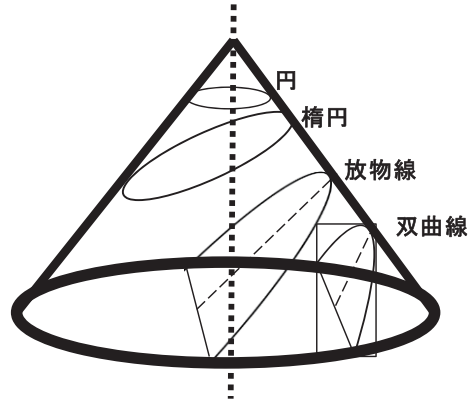


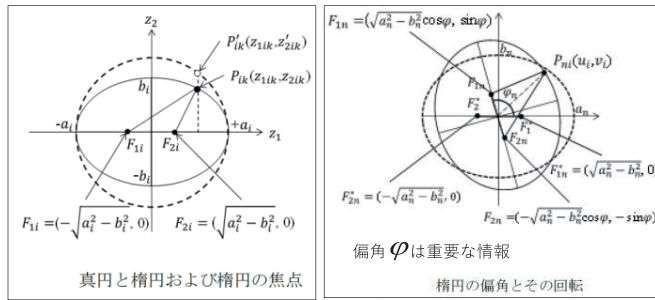
図4 円錐曲線（2次曲線）

円の場合にはその上のレベルの楕円で扱うとよい。実際に円を目指してものを作った場合には、多くの場合は歪みを生じて楕円となる。このため楕円を前提とした描写が必要である。もし、円を前提とした描写によって設計を行った場合には、楕円を描写することができないために誤った模型化のもとでの最適化となり、それ故に設計を失敗するリスクが高い。

設計のための模型（数式）を得るためには様々な条件の実験を行うことになるが、その結果は円を目指して作成しても出来上がるものはしばしば楕円になる。したがって、円は楕円の特種な場合として含まれているので、実際の設計においては楕円でアプローチするのが合理的である。

図5は円と楕円の構造を示しており、図6は楕円で扱うべきものを円で扱った場合に生じる問題を示している。実務では図6のような問題が少なくない。





**2パラメータ**  
 半径:  $r$ , 表面粗さ:  $\sigma$   
 \* LOF (Lack of Fit: 不適合) が表面粗さに含まれている. ※誤情報  
 \* 情報量が少なくかつ誤っている.

**5パラメータ**  
 長軸:  $a$ , 短軸:  $b$ , 偏角:  $\varphi$ , 距離和:  $L$ , 表面粗さ:  $\sigma$   
 \* 情報量が多く, 詳細に分かる.  
 \* 情報に誤りが無い.  
 \* 真円の場合を含んでいる.

図5 円と楕円

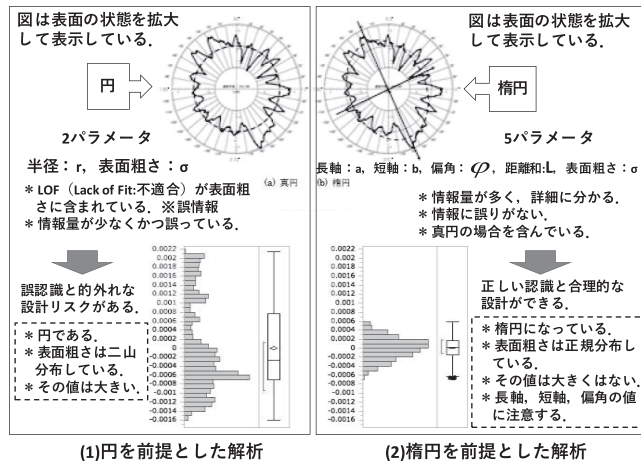


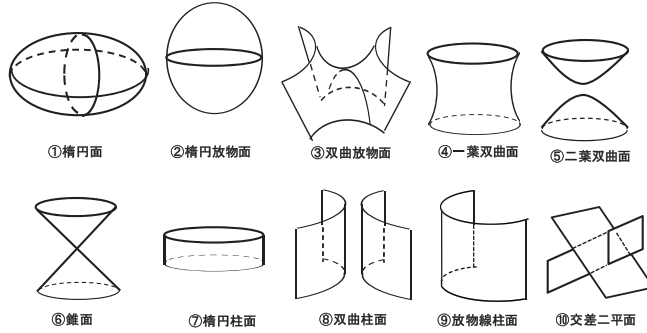
図6 楕円で扱うべきものを円で扱ったことによる問題

- ①楕円面:  $pW_1^2 + qW_2^2 + rW_3^2 = 1$
- ②楕円放物面:  $pW_1^2 + qW_2^2 + 2rW_3 = 1$
- ③双曲放物面:  $pW_1^2 + qW_2^2 + 2rW = 1$
- ④一葉双曲面:  $pW_1^2 + qW_2^2 + rW_3^2 = 1$
- ⑤二葉双曲面:  $pW_1^2 + qW_2^2 + rW_3^2 = 1$
- ⑥錐面:  $pW_1^2 + qW_2^2 + rW_3^2 = 0$
- ⑦楕円柱面:  $pW_1^2 + qW_2^2 = 1$
- ⑧双曲柱面:  $pW_1^2 + qW_2^2 = 1$
- ⑨放物柱面:  $pW_1^2 + 2qW_2 = 1$
- ⑩交差二平面:  $pW_1^2 - qW_2^2 = 0$

(11)

#### 4.1.2 2次曲面 (立体図形)

立体図形には表面が全て多角形である多面体もあるが, 多面体の設計は複数の多角形を組み合わせればよいので, 本研究は2次曲面を対象とする. 代表的な2次曲面を図7に示し, この図に対応する数式を式(11)に示している. 純粋に3次元の(3変数の)陰関数と言えるのは最初の, 6つ(①, ②, ..., ⑥)である. 何故ならば, ⑦~⑩は $W_3$ (高さ方向)が退化しており, 図形としての本質は2次元(2変数)であるからである. そして, 実際の製品の多くに用いられる立体図形の主体は以下のものである.

図7 立体図形（2次曲面）<sup>[1]</sup>

- \* 球：楕円面の特殊形
- \* 円筒：楕円柱面の特殊形

$$\sum_{i=1}^n \{p_{1\#}(W_{1i} - W_{1\#})^2 + q_{2\#}(W_{2i} - W_{2\#})^2 - 1\}^2 \rightarrow \text{最小} \quad (12)$$

#### 4.2 係数（パラメータ）の実験のランごとの推定法

実験計画は、複数の因子を列にとり最後にデータの列が用意され、各因子の水準の組合せを行にとるという表形式で示すことが多い。そして、行方向に関しては上から順に実験番号として数字が付けられる。各行は実験条件（水準の組合せ）を意味しているが、これをラン(run)と呼ぶ。なお、実験の実施はFisherの3原則のうちの無作為化の原則に基づいて無作為な（ランダムな）順番で実施することが望ましい。

単純な実験におけるデータ列は一つで、そこに入るデータは数理統計学的には母平均の推定値である。しかし、高度な実験におけるデータ列は複数となるため、これらのデータから必要な母数（パラメータ）の推定値を求めなければならない。

実験のランごとに求める係数の基本的な推定法は次の通りである。式(10)および式(11)のように標準形に持ち込んだうえで、推定した係数（パラメータ）を変数として扱ってn個のデータ（2次元データ、3次元データ）を用いて以下の最小化を行うことでモデル化ができる。

$$\sum_{i=1}^n (\text{左辺の計算値} - \text{右辺の定数})^2 \rightarrow \text{最小}$$

2次元の楕円の場合を例にとると以下のようになり、非線形最小二乗法で求解する。

この例の場合には楕円の中心の位置 ( $W_{1\#}$ ,  $W_{2\#}$ ) についても設計の対象にしている。また、より高度な設計の場合には、楕円の偏角  $\varphi$  (直交座標軸に対する楕円の長軸の傾き角) を扱う場合もある。設計因子の数が多ければ扱う係数の数が多くても然るべき高度な設計は可能である。

係数の数が多い場合に対する実際的な求解としては、数値計算で解くのが便利である。係数に初期値を与えてずれの平方和の最小化を行えばよい。初期値としては目標図形の係数の前後の値を与えるとよい。なお、図形の中心点（原点）や軸の回転角を係数（パラメータ）に加えるとより実践的な設計が可能になるが、それは応用的なことなので、紙数の都合によりその詳細は割愛する。

#### 4.3 係数関数の推定

超構造関数における各係数は設計因子の関数として最小二乗法でモデル化される。ここで注意しなければならないのは2段階の推定である。ラン毎に必要な母数の推定値が求められ、この推定値を用いて設計因子の関数として係数関数を推定する。本研究では係数関数に対して高々2次のフルモデルを想定している。したがって、最高次数が2次のフルモデルであるが、2次項がない（1次項と積項）場合や2次項も積項もない（1次項のみ）場合もある。事前の固有技術情報で必要なモデル構造を明らかにした上

で、それに対応する実験計画をD-最適計画を用いて立てる。近年の進んだ統計ソフトを用いれば実験計画を容易に作成することができる。

設計のための最適化に当たっては各係数の関数（係数関数）を用いて定式化することができる。あるいは、係数関数だけでなくそれらを用いて作成した合成関数（関数の関数なので構造的には上位の関数となる）を用いて定式化してもよい。

## 5. 立体図形を平面図形の組合せで描写するアプローチ

立体図形の場合には重要な部分の平面図形（正面図、平面図、側面図などのように投影した図形）を組み合わせて描写する方法（5.1節で議論）がある。これは立体図形を幾つかに区分するのは異なり、多頭法に近いアプローチである。区分法は区分法として存在するが、設計対象が立体図形の場合には複数の平面図形の組合せで描写するアプローチが存在する。

### 5.1 投影法（第三角法）

製図のように第三角法（正面図、平面図、側面図で組み合わせた表現）で対象を描写する。3次元の対象物は原則として2つないしは3つの投影面に投影させなければ形状を描写できない。ときには正六面体のサイコロのように、6つの投影（平面図、正面図、左側面図、右側面図、下面図、背面図）を用いることもある。さらには断面図や展開図を用いることも有効である。

製図において楕円柱面のような場合には、側面図と正面図は同じになるので側面図は省略され正面図と平面図で描写する。

### 5.2 抜き出し法

形態のあれこれ多くの部分を取り上げたりあるいは形態の全体を取り上げたりするのではなく、必要な（重要な）部分を抜き出して用いるという選択肢がある。例えば、一葉双曲面の場合には高さ方向の先端部、中央部、後端部を取り出して3枚の平面図形で扱うというアプローチである。その際に重要なことは、設計目的に合った要素を抜き出すということである。

### 5.3 断面図法

断面図とは製図において物体を一つの平面で切断したと仮定して、その内部構造を描き表した図のことである。設計の対象によっては断面図を用いる必要のあるものも存在する。前節（5.2節）までは外形についての議論をしてきたが、設計対象によっては内部構造も重要な場合がある。そのような場合には断面図を加えて設計を行えばよい。

## 6. 極座標描写による形態設計

### 6.1 2次元と3次元の極座標描写

平面図形と立体図形は、それらが図4や図7に示すような典型的な形状の場合には陰関数のパラメータ（係数）を用いての設計が可能である。しかし、実際にはそれらのような典型的な形状ではない場合も多く、そのような場合には図4や図7に示すような定型的なパラメータに落とし込むことができない。例として、図8に一筋縄ではいかない曲線を3つ示している。

これらのような曲線の場合には「極座標変換による陽関数化」（以後は極座標描写と呼ぶ）を用いることを提案する。平面図形（曲線）の場合には角度 $\theta$ と距離 $r$ による2パラメータ

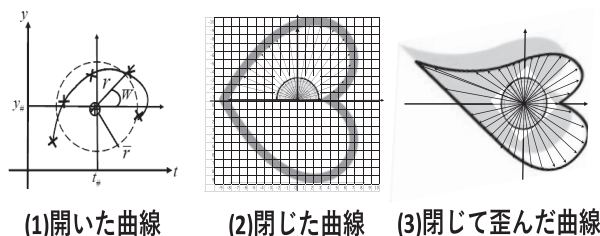


図8 容易にパラメータに落とし込めない図形

$(\theta, r)$  で描写し、立体図形 (曲面) の場合には 2 つの角度  $\theta, \lambda$  と一つの距離  $r$  の 3 パラメータ  $(\theta, \lambda, r)$  で描写する。

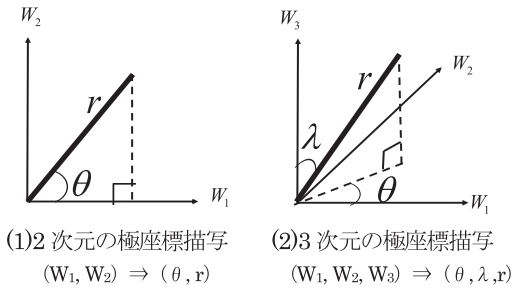


図9 極座標描写

### 6.2 極座標の原点の決め方

極座標描写とは極座標 ( $n$ 次元ユークリッド座標を 1 個の動径と  $n-1$  個の偏角で表現した座標) を用いて対象を表現することである。この場合に重要なのは極座標の原点の決定である。これには代表的なアプローチとして以下の 2 つの方法が考えられる。以後は説明を簡単にするために 2 次元の場合を例として取り上げて解説する。

\* A 法: その点からの距離の平方和を最小にする点

※なるべく大きな距離が出ないようにする。

\* B 法: その点からの距離の偏差平方和を最小にする点

※なるべく距離を同じ長さに揃えたい。

次に 2 つのアプローチについて具体的に解説する。

#### 1) A 法 (距離平方和最小)

$n$  個の点の距離の平方和は以下ようになる。

$$r_i = \sqrt{(W_{1i} - W_{1*})^2 + (W_{2i} - W_{2*})^2} \quad (13)$$

$$S(W_{1*}, W_{2*}) = \sum_{i=1}^n r_i^2 = \sum_{i=1}^n \{(W_{1i} - W_{1*})^2 + (W_{2i} - W_{2*})^2\} \rightarrow \text{最小} \quad (14)$$

式 (14) の最適化の解は次の式を解くことで得られる。

$$\begin{cases} \frac{\partial S(W_{1*}, W_{2*})}{\partial W_{1*}} = 0 \\ \frac{\partial S(W_{1*}, W_{2*})}{\partial W_{2*}} = 0 \end{cases} \quad (15)$$

この解はよく知られているように以下のようにになる。

$$W_{1*} = \sum_{i=1}^n W_{1i} / n = \bar{W}_1, \quad W_{2*} = \sum_{i=1}^n W_{2i} / n = \bar{W}_2 \quad (16)$$

これはそれぞれの平均を求めればよいので極めて簡単な計算で求めることができる。

#### 2) B 法 (距離の偏差平方和最小)

ここで求めたいものは以下の最適化 (偏差平方和の最小化) の解である。

$$\begin{aligned} f(W_{1\#}, W_{2\#}) &= \sum_{i=1}^n (r_i - \bar{r})^2 \rightarrow \text{最小} \\ r_i &= \sqrt{(W_{1i} - W_{1\#})^2 + (W_{2i} - W_{2\#})^2} \\ \bar{r} &= \sum_{i=1}^n r_i / n \end{aligned} \quad (17)$$

上記のものを書き直すと以下ようになる。

$$f(W_{1\#}, W_{2\#}) = \sum_{i=1}^n \{ \sqrt{(W_{1i} - W_{1\#})^2 + (W_{2i} - W_{2\#})^2} - (\sum_{i=1}^n \sqrt{(W_{1i} - W_{1\#})^2 + (W_{2i} - W_{2\#})^2}) / n \}^2 \rightarrow \text{最小} \quad (18)$$

これは少し高度な計算になるが、コンピュータを用いれば数値計算により容易に求めることができる。

図10に極座標変換による描写の例を示している。この場合の極座標の原点はA法で求めてもあるいはB法で求めても良い。図10では簡単なA法で原点を求めたもとでの結果を示している。この例の場合には、得られた陽関数は精度が粗くてもよければ 3 次関数で近似し、精度を上げたいならば 2 区分 (2 つの 2 次関数) で近似すればよい。さらに近似精度を上げたいならば近似に用いる多項式の次数を上げる必要がある。

### 6.3 閉じた曲線の場合の極描写

図11 (1) に示すような閉じた曲線は全体を

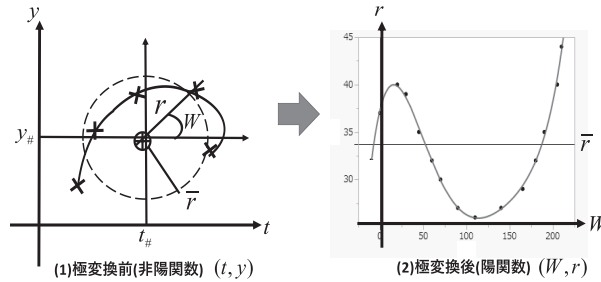


図10 極座標変換による陽関数化の例

そのまま極描写によって陽関数化することができる。しかし、図11(2)に示すような場合には閉じた曲線は全体をそのまま極描写によって陽関数化することができないので、幾つかに区分をしなければならない。この場合、区分した各部分は開いた曲線になる。なお、図形を区分した場合には、区分した図形ごとに極座標の原点を求める必要がある。

[注] 閉じた曲線を区分する必要があるかないかは、原点から360°回したときに動径が複数の点に交わらないという条件を満たす原点がとれるかどうかである。そのような原点が取れば区分の必要はないが、取れなければ区分しなければならない。

### 6.4 A法とB法の比較

A法とB法の比較の一例を図12に示している。これらはあくまでも一例として参考にされたい。

#### (1) 閉じた曲線で円に近い場合

閉じた真円(あるいは真楕円)の場合にはA法とB法の両者の解は必ず円や楕円の中心となる。

そして真円(あるいは真楕円)に近い場合も両者はほぼ同じになる。しかし、真円や楕円でない場合には両者の解は異なるが、歪み方のパターンによって両者の良し悪しはcase by caseとなるので明確な優劣はつけ難い。

#### (2) 開いた曲線で円に近い(円弧に近い)場合

B法は円弧である限り必ず円の中心となる。しかし、A法は欠けた部分の影響を受けて円の中心からずれる。この場合にはB法の方が明らかに優れている。けれどもA法が絶対に使えないというわけではない。

#### (3) 閉じた曲線で凹みが大

B法の原点は図形の外に出るために単一の陽関数化ができない。しかし、A法の原点は図形の中に存在するために単一の陽関数化ができる。この場合は、B法を用いるならば図形を区分するとよい。B法のための数値計算ができない場合にはA法で陽関数化すればよい。

#### (4) 閉じた曲線で凹みが大

いずれも原点は図形の外に出るために単一の

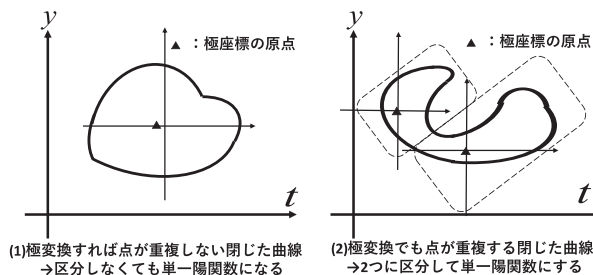


図11 極描写と区分による閉じた曲線の単一陽関数化

陽関数化ができない。しかし、A法は2区分にするとうまく行すが、B法は2区分してもうまく行かない。

上記の(1)から(4)により、2つの方法には特徴があるために使い分ける。可能であれば両方法で原点を求めて都合の良い方を選択するとよい。なお、図12(4)に示すように、区分が必要な場合も存在する。この場合には、区分後はA法でなければならないことに注意する。

## 6.5 極座標描写による形態設計の例としての円と球

極座標描写による形態設計の簡単な例として図13には目標図形が円と球の場合を示している。設計に関する説明はより明快に行うことの可能な円の場合を取り上げて行う。

実際の設計において重要なことは、表面には凹凸がありこれは残差(平均からの差)で把握すべきであるということである。図13における上下の変動は、表面の凹凸(円の場合は半径との差、楕円の場合は2つの焦点からの距離和との差)を意味するものであると考えられることが多いが、これは厳密にはLOF(lack of fit: 不適合)と誤差の合体したものである。もし、

図形が真円から根本的にずれた場合には残差には誤差だけでなくLOFも含まれることに注意が必要である。したがって製造に問題があって明らかにLOFの存在が予想される場合には円は楕円として扱う必要があり、球は楕円面として扱う必要がある。そうしないと第4章の図6の問題を抱えることになる。

## 7. 高次の超構造関数

### 7.1 高次の超構造関数の事例

超構造関数が高次の関数になった場合には高度な設計を行う必要がある。例えば以下の事例がある。

- (1) 2次の実例：射出成型の実例<sup>[2]</sup>
- (2) 3次の実例：水晶発振子実例<sup>[4]</sup>

このような高次の場合には超構造関数の係数や偏微分の活用が重要になる。本章では最初に超構造関数が2次の場合を取り上げ、その後に3次の場合を取り上げる。

### 7.2 超構造関数が2次の場合

#### 1) 頑健設計への応用

超構造関数は多変数関数を超因子Hに注目して構成した多項式なのでこれをHで偏微分することができる。

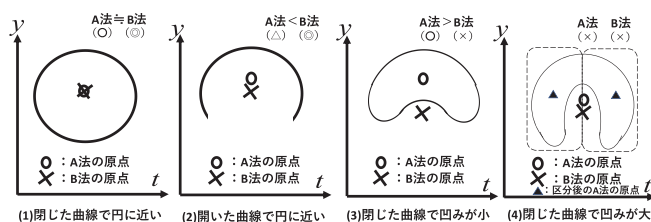
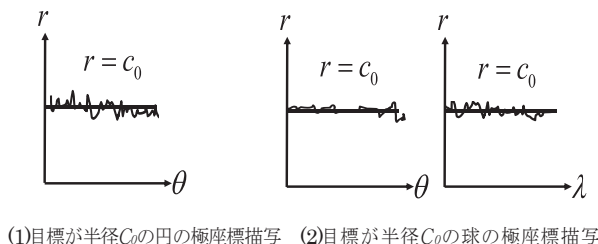


図12 A法とB法の比較



(1)目標が半径 $C_0$ の円の極座標描写 (2)目標が半径 $C_0$ の球の極座標描写

図13 極座標描写による形態設計の例

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \lambda_1(\mathbf{x})H + \lambda_2(\mathbf{x})H^2 \quad (19)$$

式 (19) を  $H$  で偏微分した結果は以下のようになる。

$$\partial y / \partial H = \partial F(H; \mathbf{x}) / \partial H = \lambda_1(\mathbf{x}) + 2\lambda_2(\mathbf{x})H \quad (20)$$

この関数の絶対値を 0 に近付ければ  $H$  の影響は減衰できる。ただし  $H$  自身もその水準が決定され、それは使用時・操業時の  $H$  の目標水準（この前後に  $H$  の値がばらつくという水準）となる。これは非線形を応用した頑健設計であるが、この場合の  $H$  は直積実験<sup>[11]</sup>において外側に割り付けることもできるし、内側に割り付けることもできる。実務的には、外側に割り付けたら外乱（攪乱因子）を意味し、内側に割り付けたら内乱（ばらつく設計因子）を意味する。なお、これを内側に割り付けた場合には、外側にはさらに別の外乱を起こす因子（攪乱因子）を割り付けることもできる。

なお、超構造関数が 1 次式の場合には設計因子がなければ  $H$  のばらつきの減衰はできないが、2 次式の場合には設計因子がなくても  $H$  のばらつきの減衰は可能となる。また、式 (20) の第 2 項には超因子  $H$  の前に係数関数（設計因子の関数）があるので、この絶対値を小さくすることも減衰に役立つことになる。ただし、設計因子は式 (20) の第 1 項の定数項にも設計因子の関数があるため、第 2 項にあって第 1 項にない設計因子を用いなければならないことに注意する。

## 2) 最大、最小、範囲への応用

2 次関数の場合は極値が一つなのでこれを求めた上でその合成関数を用いて設計する。例えば極値を含むある区間の中の最大や最小を求める場合には以下の合成関数（極値の横座標を  $p$  とする）を用いればよい。

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \lambda_1(\mathbf{x})H + \lambda_2(\mathbf{x})H^2 \quad (21)$$

$$p = -\lambda_1 / (2\lambda_2), \quad p = p(\mathbf{x}), \lambda_1 = \lambda_1(\mathbf{x}), \lambda_2 = \lambda_2(\mathbf{x}) \quad (22)$$

範囲  $m_L \sim m_U$  での最大と最小は以下の合成関数となる。

$$y_{Max}(\mathbf{x}) = \text{Max}\{F(p(\mathbf{x}); \mathbf{x}), F(m_L; \mathbf{x}), F(m_U; \mathbf{x})\} \\ y_{Min}(\mathbf{x}) = \text{Min}\{F(p(\mathbf{x}); \mathbf{x}), F(m_L; \mathbf{x}), F(m_U; \mathbf{x})\} \quad (23)$$

$$y_{Ran}(\mathbf{x}) = y_{Max}(\mathbf{x}) - y_{Min}(\mathbf{x}) \quad (24)$$

両者の差（最大から最小を引くので非負）は範囲（差の絶対値）でこれはばらつきを意味し、これを小さくすることは一つのタイプの頑健設計である。設計はこれらを用いて数理計画法<sup>[3]</sup>で求解すればよい。その定式化は、平均値を制約条件にして範囲を最小化するか、範囲のある値以下にするという不等式制約にした上で、平均値を最適化するかのいずれかになる。

## 7.3 超構造関数が 3 次の場合

超構造関数が 3 次関数の場合もある。その場合には極大値、極小値そして変曲点が設計の対象となる。

$$y = F(H; \mathbf{x}) = \lambda_0(\mathbf{x}) + \lambda_1(\mathbf{x})H + \lambda_2(\mathbf{x})H^2 + \lambda_3(\mathbf{x})H^3 \quad (25)$$

3 次関数で極大値と極小値がある場合には各々の横座標  $p_i$  は以下の式で与えられる。

$$p_U, p_L = (-\lambda_2 \pm \sqrt{\lambda_2^2 - 3\lambda_3\lambda_1}) / (3\lambda_3) \\ p_U = p_U(\mathbf{x}), p_L = p_L(\mathbf{x}), \lambda_1 = \lambda_1(\mathbf{x}), \lambda_2 = \lambda_2(\mathbf{x}), \lambda_3 = \lambda_3(\mathbf{x}) \quad (26)$$

添字  $U$  は大きい方の値（右上がりの 3 次関数の場合は右側の極値）を意味し、添字  $L$  は小さい方の値（右上がりの 3 次関数の場合は左側の極値）を意味する。そして、いずれも合成関数でその実体は設計因子の関数である。因みに変曲点は二種類の極値の midpoint なので省略する。

3 次関数の設計は 2 つの極値が存在する場合がある。その場合には代表的な設計対象として以下の 3 つのものがある。

\* 極値間左右差、\* 極値間上下差、\* 極値間勾配

これらは以下の合成関数として扱えばよい。

$$\text{極値間左右差} : p_U(\mathbf{x}) - p_L(\mathbf{x}) \\ \text{極値間上下差} : F(p_U; \mathbf{x}) - F(p_L; \mathbf{x}) \\ \text{極値間勾配} : (F(p_U; \mathbf{x}) - F(p_L; \mathbf{x})) / (p_U(\mathbf{x}) - p_L(\mathbf{x})) \quad (27)$$

本章のアプローチの対象となる事例は文献 [4] である。その内容については紙数の都合に

より割愛する.

## 8. おわりに

本研究は最初に超設計というアプローチを明らかにした. 次に複雑な関数形の場合の設計方法について議論するとともに, 閉じた図形に関する設計方法について提案した. そして, 最後に超構造関数が高次の場合に多重の合成関数を用いた設計方法について議論した.

加工・鍛造・鋳造・射出成型・半導体製造などの設計においては形状の設計は重要なテーマである. 本研究はそのための有用な設計方法を提案した. この方法を広く実務に適用することが今後の課題である.

### 【参考文献】

- [1] 平下幸男 (1992): 「数理科学のレッスン」, 産業図書.
- [2] 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, 122-136.
- [3] Myers R. H., Montgomery D.C., and Anderson Cook C. M., (2009): *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*, Wiley, New York.
- [4] 高橋星太, 濱口勝重, 高橋武則 (2015): "HOPE手法を用いた水晶発振子の設計パラメータの最適化", JSQC第109回研究発表会発表要旨集, pp. 39-42.
- [5] Takahashi, T. (2015): "Proposal of Flexible Design and its Application", *Proc. of the Asian Network for Quality Congress 2015 in Taipei, CD proceeding*, PP.1-10.
- [6] 高橋武則 (2017): "超構造関数による柔軟設計", 日本品質管理学会第113回研究発表会発表要旨集, pp.157-160.
- [7] Takahashi, T. (2017): "Hyper Design based on Hyper Factors", *Proc. of the Asian Network for Quality Congress 2017 in Kathmandu, CD proceeding*, PP.1-10.
- [8] 高橋武則 (2018): "描写因子による形態設計" JSQC第116回研究発表会発表要旨集, pp.139-142.
- [9] 椿広計 (2006): 「ビジネスへの統計モデルアプローチ」, 朝倉書店.
- [10] 椿広計 (2006): "統計科学の横断性と設計科学への寄与", 「横幹」, 1, [1], 22-28.
- [11] Wu, C. F. J. and Hamada, M. (2009): *Experiments: Planning, Analysis, and Optimization (2nd ed.)*, Wiley, New York.