

非構造格子系熱流体解析システム

SCRYU/Tetra[®]
Version 12

**ユーザーズガイド
最適化編(オプション)**

株式会社ソフトウェアクリエイドル

2015年6月

SCRYU/Tetra®は、株式会社ソフトウェアクリエイドルの商品名です。本書の一部または全部を無断で複製・転載・改編することを禁じます。

CRADLE 株式会社ソフトウェアクリエイドル
Software Cradle Co., Ltd.

本 社 : 大阪市北区梅田3丁目4番5号
毎日インテシオ
TEL : 06-6343-5641 FAX : 06-6343-5580

東京支社 : 東京都品川区大崎1-11-1
ゲートシティ大崎ウエストタワー
TEL : 03-5435-5641 FAX : 03-5435-5645

SCRYU/Tetra©2015 Software Cradle

本文で使用するシステム名・製品名は、それぞれの各社の商標、または登録商標です。
本製品で使用されているMOGAおよびKriging法は、
東北大學 流体科学研究所 大林・鄭 研究室の研究成果を利用しています。
本アプリケーションを利用するためには、別途ネットワークライセンス(オプション)が必要です。
ライセンスの購入については担当営業までご相談ください。

目 次

第1章 最適化の概要

1.1 最適化問題とは	1-3
1.2 最適化の進め方	1-5
1.3 サンプリングと実験計画	1-6
1.4 代理モデル(Kriging法)	1-8
1.5 最適解の計算(多目的遺伝的アルゴリズム:MOGA)	1-10
1.6 最適化結果の見方と役立て方	1-12
1.7 Efficient Global Optimization	1-16

第2章 Extension Option(Optimization)の概要

2.1 Extension Option(Optimization)関連ファイル	2-2
2.2 Extension Option(Optimization)の操作手順	2-3
2.3 操作手順の具体例	2-4
(1) Extension Option(Optimization)の起動	2-4
(2) 初期設定	2-8
(3) 条件設定	2-10
(4) MOGAの実行	2-14
(5) 結果の確認	2-16

第3章 例題

3.1 例題の紹介	3-2
(1) 最適化の対象	3-2
(2) 想定する状況	3-3
3.2 流体解析条件	3-4
解くべき方程式	3-4
解析選択	3-4
物性値	3-4
境界条件	3-4
初期条件	3-4
その他	3-5
圧力と粘性応力の出力設定	3-5
解析メッシュ	3-6
3.3 Extension Option(Optimization)による最適化	3-8
(1) 最適化計算条件	3-8
設計変数	3-8
目的関数	3-8
(2) Extension Option(Optimization)の操作手順	3-9
3.4 結果の評価	3-15
(1) 応答曲面	3-15
(2) 寄与度分布	3-16
(3) 相関関係(目的関数/全体)	3-17

(4) 最適解分布	3-18
(5) 選定した解の検証	3-19
3.5 VBインターフェースによる最適化プロセスの自動化	3-21
(1) 起動方法	3-21
(2) ソースコードの説明	3-22
(3) ExecuteEGOについて	3-23

第4章 メニューリファレンス

[ファイル] - [新規作成]	4-2
[ファイル] - [開く]	4-3
[ファイル] - [閉じる]	4-4
[ファイル] - [Extension Option(Optimization)の終了]	4-5
[選択] - [すべて選択]	4-6
[選択] - [選択解除]	4-7
[表示] - [選択データのみ表示]	4-8
[表示] - [選択データのみ非表示]	4-9
[表示] - [すべて表示]	4-10
[表示] - [表示/非表示反転]	4-11
[表示] - [視点をリセット]	4-12
[表示] - [グラフツリーの表示]	4-13
[表示] - [メッセージウィンドウの表示]	4-14
[表示] - [ラバーボックス拡大表示]	4-15
[条件設定] - [条件設定ウィザード...]	4-16
[操作] - [1ボタンモード]	4-17
[操作] - [2ボタンモード]	4-18
[操作] - [3ボタンモード(CTRL)]	4-19
[操作] - [3ボタンモード]	4-20
[操作] - [3D回転]	4-21
[操作] - [2D回転]	4-22
[操作] - [移動]	4-23
[操作] - [拡大・縮小]	4-24
[操作] - [オペレーション]	4-25
[オプション] - [プログラムの詳細設定]	4-26
[ウィンドウ] - [新しいウィンドウを開く]	4-33
[ヘルプ] - [ユーザーズガイド]	4-34
[ヘルプ] - [バージョン情報Extension Option(Optimization)]	4-35
[ポップアップメニュー]	4-36
[ポップアップメニュー(グラフツリー)] - [視点を同期させる]	4-37
[ポップアップメニュー(ドローウィンドウ)] - [目的関数・設計変数一覧]	4-38
[ポップアップメニュー(ドローウィンドウ)] - [応答曲面の表示設定]	4-40
[ポップアップメニュー(ドローウィンドウ)] - [グラフの表示設定]	4-41

第5章 ファイル

5.1 ファイルの内容	5-2
5.2 条件ファイルの内容	5-4
5.3 Krigingファイルの内容	5-5
5.4 計算結果ファイルの内容	5-6

マニュアルの構成について

SCRYU/Tetraのマニュアルは、下記の13分冊構成となっております。

- **基礎編**

熱流体解析の基本的な考え方だけでなく、**SCRYU/Tetra**の各機能の詳細説明を含んだ総合解説書です。熱流体に初めて触れられる方から、各機能の理論的背景を確認されたい方まで、**SCRYU/Tetra**を使用される全ての方々を対象としています。

- **操作編**

SCRYU/Tetraの基本的な操作を基本例題を通して学ぶことができるチュートリアルです。実際に**SCRYU/Tetra**の操作を始める際には、まずこのガイドを紐解いてください。基本例題で基礎体力が付いたら、例題編もお試しください。

- **リファレンス(プリ)編**

SCRYU/Tetraのプリプロセッサ(プリ)の詳細解説書です。

- **リファレンス(ソルバー)編**

SCRYU/Tetraのソルバーの詳細解説書です。ソルバーコマンドとユーザー関数のリファレンスを含みます。

- **リファレンス(ポスト)編**

SCRYU/Tetraのポストプロセッサ(ポスト)の詳細解説書です。

- **リファレンス(VBインターフェース)編**

SCRYU/Tetraのプリプロセッサ、ソルバー、ポストプロセッサに用意されているVBインターフェースのメソッドリファレンスです。

- **リファレンス(ツール)編**

SCRYU/Tetraに付随した各種ツールについての操作説明書です。

- **例題編**

SCRYU/Tetraの解析機能とその利用法を学ぶための例題編です。解析機能特有の考え方を学んだり条件設定のしかたを調べたりと、解析機能を使いこなす際の足がかりとして最適です。実際的な工業製品を模した解析事例も紹介しています。

- **構造解析編(オプション)**

SCRYU/Tetraのオプションである構造解析機能の詳細解説書です。プリプロセッサ(プリ)とソルバーのリファレンス、また、操作を学ぶための例題を含みます。

- **流体構造連成(Abaqus[®])編(オプション)**

SCRYU/TetraのオプションであるSCRYU/Tetra I/F Option for Abaqus[®]の詳細解説書です。プリプロセッサ(プリ)とソルバーのリファレンス、また、操作を学ぶための例題を含みます。

- **最適化編(オプション)(本書)**

SCRYU/Tetraのオプションである最適化機能の詳細解説書です。最適化についての概要、操作説明、また、**SCRYU/Tetra**と連携した例題を含みます。

- **1D/3Dカップリング(GT-SUITE)編 (オプション)**

SCRYU/TetraのオプションであるSCRYU/Tetra I/F Option for GT-SUITEの詳細解説書です。

- **ファンモデリング・解析ツール(SmartBlades[®])編(オプション)**

SmartBlades[®]についての操作説明書です。

本書をお読みになるまえに

1. 本書が対応しているソフトウェアのバージョン

- **SCRYU/Tetra V12**

なお、本書は開発中のリリースを用いていますので、リリースにより計算時間、結果などが異なることがあります。

2. 本書で使用しているドライブ名やフォルダ名は、実際と異なることがあります。
3. [ファイル] - [開く]等の記述は、メニューバーの[ファイル]を選択し、さらに[開く]を選択することを意味します。

UNICODE化について

本製品では、V12より、多言語対応を目的として、UNICODE化がなされております。その一環として、全てのファイル入出力をUTF-8にて行う形式に、動作仕様が変更されました。以下の点にご注意ください。

- V12では、V11のプログラムが output した全てのファイル群の入力に対応しております。
- V12のプログラムで output されたファイル群は、原則として、V11以前の製品では使用できません。
- 本書における「文字数」もしくは文字列の「バイト数」という記述は、UTF-8ではASCII文字（半角英数記号）は1文字=1バイト、それ以外は1文字=2～4バイトを意味しております。

第1章 最適化の概要

本最適化オプション(Extension Option(Optimization)、以下EOoptiと記します。)は数値最適化を行うソフトウェアです。用いる数理的な手法を絞ることで、操作を簡単なウィザード操作とし、手軽にお使い頂くことができる最適化ソフトウェアとなっています。より幅広い熱流体解析結果の御利用にお役立てください。本章ではEOoptiの概要を述べます。方針として数式等はなるべく排して関連手法の特徴に焦点を当てて述べたいと思いますが、数理上の仕組みに御興味のある方は是非巻末の参考文献も御参照ください。

1.1 最適化問題とは

最適化問題とは、簡単に言おうとすれば、ある目標を目指すとき、その目標を達成する最適な指針を探す問題と言えます。しかしながら一言で目標を達成すると言っても、その実現方法にはいろいろな方針があり、またその方針に沿った行動の仕方(方法)もさまざまにあると思います。最適化問題を解く、ということは、想定できる、つまり現実的な方針や方法を決め、その範囲で最大限目標に近付くには具体的にどの方針でどの方法で行動すればよいのかを決めるここと、と言ってもよいかもしれません。しかしながら目標を掲げるにしても、一つだけではなく、2つ、3つと目標を掲げることもできます。そして、場合によってはある目標を達成するためには別の目標を諦めざるを得ない場合もあります。抽象的な話が続きますが、往々にして世の中にはさまざまな希望とでも言うべき目標があり、またその実現手段もさまざまで、かつ諦めざるを得ない事柄もあり、いわく「人生設計」という名の最適化問題を解くが如くまさに日々試行錯誤の連続であって、それこそが、ある意味では生きる愉しみになっているのかもしれません。

少し具体的な例を考えてみましょう。EOoptiは熱流体解析のオプションでありますし、図1のような簡単な熱流れの問題を取り上げて最適化問題を考えてみます。

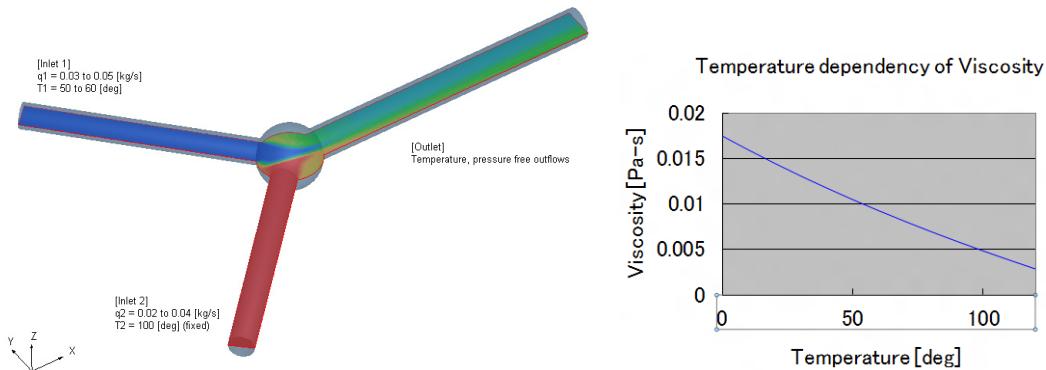


図1 热流れ問題の最適化例

図1左は簡単な集合管(マニホールド)です。流入口が2つあり、それぞれの流入口で想定される流体の流量 q と温度 T の範囲が決まっています。また、流体の粘性は右図のように温度によって変化します。このとき、流出口の温度が最も低くなり、かつ流れの損失が最も小さくなるためには流入境界の温度と流量をいくつにすればよいでしょうか。

定性的に考えると q_1, q_2 が大きくなれば損失は大きくなる傾向があります。また T_2 は変わらないので、流出口の温度を下げるには q_1 を多くし、 T_1 を下げるのが効果的です。しかしながら、流体は温度が下がると粘性が上がる所以、あまり温度を下げ過ぎると摩擦損失が増える可能性もあります。これは最適化問題として捉えると、目的が損失の低下、流出口温度の低下という2つ、それを変化させる手段が q_1, q_2, T_1 の3つの最適化問題となります。最適化ではほとんどの場合、目的のことを目的関数、手段のことを設計変数と呼びます。つまり、これは2目的関数、3設計変数の最適化問題になります。

もしも、今3つの設計変数が2つの目的関数にどのように影響するのかが分かっていれば何も悩むことはありません。影響を見て最良な q_1, q_2, T_1 を選ぶことができます。しかしながら厄介なのは、通常その影響度合が不明であることです。その影響度合を知るとは、系の損失、流出口温度という2つの目的関数を $\Delta p, T_{out}$ と記述すると、

$$\Delta p = \Delta p(q_1, q_2, T_1)$$

$$T_{out} = T_{out}(q_1, q_2, T_1)$$

という2つの関数の形を定めることとも見なせます。これらの関数が決まれば、それぞれの最小値はそれぞれの関数から分かります。また、2つの関数の最小値は必ず同じ q_1, q_2, T_1 で決まるとは限らず、

目的関数間の関係を知ることも大切です。3つの設計変数に対して変化する目的関数が示すのは3次の曲面で表現される空間になります。これを最適化問題で定義された設計空間と呼びます。目的関数の形、つまり設計空間を知ることが影響度合を測る大きな足がかりになります。まとめると、最適化とは設計空間の構造を調べ、その空間から目的に適する設計変数の組み合わせを見出すこと、とも言えます。

さて図1の問題については、今は提示だけに留めておき1.6節では実際にEOoptiで解いて設計空間をもう少し詳しく見てみましょう。その前に、EOoptiで行われる最適化の手順、また各手順で行われる計算の考え方を概観してみることにします。

1.2 最適化の進め方

最適化はそれそのものを計画した瞬間から始まります。そしてその計画が最も重要です。まず目的関数を定義することはもちろんですが、設計変数としてどのような量を定義するか、またその設計変数をどのように変更するのか、変更は現実的に行うことができるものか、目的を達成できない場合はどのように設計変数を選び直すか、あるいは設計変数を追加するのか、そもそも最適化に許される時間とコストはどのくらいか、といった大きな道筋を描いておくことが重要です。この道筋が曖昧であると、極端な場合はあれこれ試行する中で目標そのものを見失い、結局、最適化そのものが破綻することも考えられます。まれに混同した理解の例も見かけますが、最適化は設計製造そのものではありません。設計製造の指針を見出す作業です。最終的に「このようなものをこのような条件の下で造りたい」という確固とした目標がなければいくら指針だけを見ても仕方ありません。計画が立てられれば、あとは熱流体解析ソフトウェアやEOoptiまたは実測データなどを使って設計空間を再現し、目標に適う設計変数を見出すだけです。大事なのはソフトウェアが出す結果を選ぶのは最適化問題を解いている私たち自身であり、その選び方はいかに問題が明確に計画されているかに左右されるということです。

目的関数と設計変数が定まった、つまり問題が定義されたらいよいよ具体的な最適化計算のプロセスに入ります。1.1節で述べたように、目的関数は未知の関数です。まずはこれをできるだけ効率よく推定しなければなりません。そのために、サンプリングということを行います。これはいくつかの設計変数の組み合わせに対して実際に目的関数の値を得てみることです。図1の集合管の例では q_1, q_2, T_1 を変えて Δp と T_{out} を数ケース調べることになります。これは実測しても構いませんし、熱流体解析を行った結果を用いても構いません。もちろんサンプル数が多ければ多いほど目的関数の推定はしやすくなりますが、その分時間等コストが嵩みます。いかに効率良いサンプリングを行うかが最適化プロセスのコストに大きく関わります。(1.3節)

次に、目的関数の形を具体的に作ってみます。サンプル数が十分でない場合にはサンプル間を何らかの手段で補間しながら関数の形を作ります。これで具体的に目的関数と設計変数の関係、つまり設計空間を定量化できます。(1.4節)

設計空間が定まつたらその空間を探索し、例えば最大値など、目標とする目的関数值を実現する設計変数の組み合わせを探します。この組み合わせが最適解です。(1.5節)

目的関数が複数あり、それらにいわゆるトレードオフ(相反)の関係があると、最適解の計算結果はたいていの場合1つではなく複数の候補解が提示されます。この場合、設計空間等を参考にしながら1つの解を選択し、最適解を求める一連のプロセスが終了します。しかしながら、この結果はあくまで最適化計算の結果であり、実際にこのままの設計変数で製造することはまずありません。この結果を元に実測や解析を行って実際に目的が達成されるのかを十分評価して評価上問題がなければこの最適化結果を参考にして最終的な設計変数を決定します。(1.6節)

まとめると、以下のような流れになります。

1. 最適化の計画(目的関数、設計変数の設定)
2. サンプリング
3. 代理モデルの作成
4. 最適解(候補解)の計算
5. 最適解の選択
6. 選択解の評価(問題点があれば1または2に戻ります)

なお、EOoptiの操作上の手順は2章であらためて述べることにして、以下の節では各段階の手法を少し詳しく見てみたいと思います。

1.3 サンプリングと実験計画

既述のように目的関数が決まり、設計変数との詳細な関係が分かれば、設計空間を知ることができますが、その関係を調べるためにには実際に設計変数を変化させた際の結果を見なければなりません。設計変数はある範囲を持ちますし、また1つとは限りません。できるだけ正確に目的関数と設計変数の関係を知るには、実験や解析など手段は異なれど、いずれにせよ多くの設計変数での結果を見る必要があります。

このように、目的関数と設計変数の関係を調べるためにいくつか設計変数を変化させた結果を得ることをサンプリングと呼びます。コストの問題を考えるとサンプリングはできるだけ数が少ない方が望ましく、また想定される設計変数の変化の範囲をできるだけ均等に広く、しかし着目したい点があればその点を多くサンプルできるようにすることが望ましくなります。そのために、どのような設計変数でいくつのサンプリングをすればよいのか、ということを考えることが必要になります。これを実験計画と呼びます。実験計画を行う手法は実験計画法(Design of experiment: DOE)と呼ばれ様々な方法があります。EOoptiではラテン超方格法[文献 1]という手法を用います。

ラテン超方格法は複数の設計変数があるときに所定の回数のサンプリングに対して、各設計変数がそれらの範囲内で万遍なく現れ、また各設計変数の組み合わせが必ず重複しないように各サンプルでの設計変数を決定します。各設計変数そのものをサンプリング数に応じて均等に割り振るのではなく、設計変数の累積確率密度をサンプル数で割り、その密度の重みに応じてランダムに設計変数を選ぶことが特徴です。設計変数が2つの場合を特にラテン方格法と呼びます。名前はかつて數学者オイラー(Leonhard Euler)が方陣内に重複しないように記号を並べる問題を考察した際に記号としてラテン文字が使われたことに由来します。これは現代の「数独」(ナンプレ(Number place))というパズルと本質的には同じ問題です。例えば0から1の間で定義される3つの設計変数があり、10回のサンプリングを行う場合、ラテン超方格法の結果は表1のようになります。

表1 3つの設計変数のラテン超方格法によるサンプリング例

サンプル	設計変数1	設計変数2	設計変数3
1	0.8	0.2	0.9
2	0.9	0	0
3	0.3	0.6	0.5
4	0.1	0.7	0.8
5	0.4	0.3	0.1
6	0.5	0.5	0.4
7	0.6	0.9	0.3
8	0.2	0.4	0.7
9	0	0.1	0.2
10	0.7	0.8	0.6

サンプル間で設計変数の組み合わせの重複がなく、かつ各設計変数がそれらの定義域で万遍なく選ばれています。

ところで、サンプリング数はどの程度の数にすればよいでしょうか。もちろん、数を増やすことができればそれに越したことはありませんが、最低限の数については、ラテン超方格法のように設計変数が万遍なく選ばれる場合に目的関数が各設計変数の2次多項式で近似できると仮定すると、設計変数の数をNとして

$$\text{サンプル数の下限} = (N+1)(N+2)/2$$

とされています。つまり、設計変数が3つの場合、10のサンプル数は必要です。しかしながら、これはあくまで2次式の近似が仮定になっていますから、さらに複雑な応答を持つ目的関数であると、さらに

サンプル数が要ります。多くの場合上式で求められた最低サンプル数の2倍程度の数のサンプルが行われるようですが、複雑な目的関数ですと10倍程度のサンプル数が必要になることもあります。

1.4 代理モデル(Kriging法)

十分多くの数のサンプリングが行える場合は別にして、たいていの場合サンプリング数は限られ、目的関数の大雑把な形は分かっても、結局はサンプリング点間の関数の形は不明であり、そのままでは目的関数の最大値や最小値を求ることは困難な場合があります。そのため、サンプリング点間の値を内挿等で作り、目的関数を一つの線形関数として完成させることを考えます。サンプリングされたデータから目的関数を作る関数モデルを代理モデルと呼びます。また目的関数は設計変数の数の次元をもつ曲面となり、これは応答曲面と呼ばれます。なお、以下目的関数と応答曲面という語が混在することがありますが、同義とお考えください。代理モデルはサンプリングの結果を用いて近似的に応答曲面を作る方法と言えます。代理モデルはいくつも考えられますが、EOoptiではKriging法[文献 2, 3]を採用しています。

Kriging法そのものはデータ内挿法の一つです。最も一般的な内挿法は、恐らくデータ間の距離の重み付き平均だと思います。Kriging法も重み付き平均の一つですが、重みにバリオグラムという量を用います。バリオグラムはデータ間の距離と値の分散との関係を示す統計量で、データ間に相関性が存在する範囲を表現します。バリオグラムにも多種ありますが、ここではセミバリオグラムモデルの一つであるGaussianモデルを用いています。バリオグラムの重み付けを行うことで、内挿されたデータが(サンプリングされた)既存のデータと比較してどの程度の誤差分散があるのかを評価することができます(サンプリング点のデータの誤差分散はゼロ)。Kriging法は誤差分散が最小になる内挿値を採用する仕組みになっており、結果として、統計的ではありますが内挿値の確からしさを知ることができます。この確からしさを見て、例えば確からしさが低い範囲にはさらにサンプリングデータを追加するなど、応答曲面の精度を上げていく指標を得ることができるのが大きな特徴で、一般には複雑な形の応答曲面の表現に優れているとされます。また、一部の補間法と異なり、サンプリングされた点は正解(これ以上誤差を小さくすることはできないデータ)として見なされ、応答曲面に必ず乗るという特徴もあります。

ここで、Kriging法による応答曲面の例を見てみましょう。例として、あらかじめ次のような目的関数と2つの設計変数を用意します。

目的関数 : $F = \sin(x_1) + \sin(x_2)$

設計変数 : $x_1 (0 \leq x_1 \leq 2\pi), x_2 (0 \leq x_2 \leq 4\pi)$

この関数Fは図2のような曲面になります。

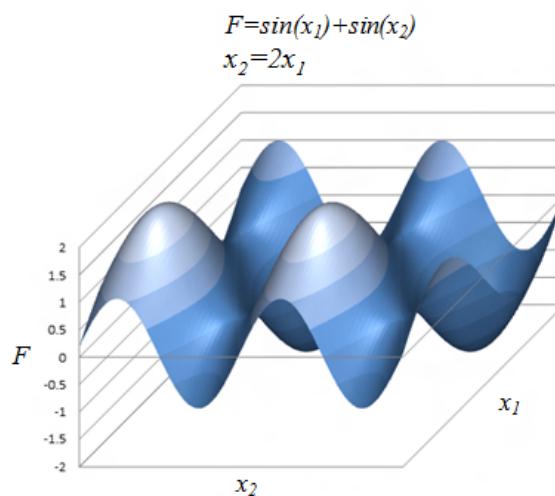


図2 $F = \sin(x_1) + \sin(x_2)$

最大値が2、最小値が-2の曲面です。この曲面をEOoptiのKriging法で50のサンプルと15のサンプルから近似してみます。すなわち、予め50または15の x_1, x_2 の組み合わせをランダム超方格法から作成して、それぞれの x_1, x_2 に対するFの値を作り、それらのデータから曲面を近似的に再現してみます。結果は次図のようになります。

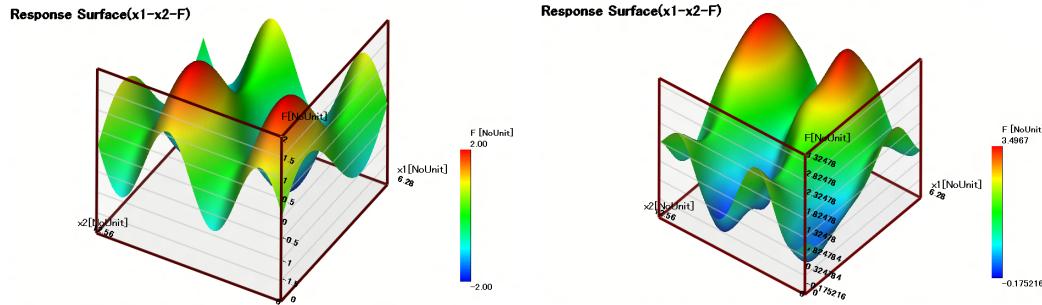


図3 Kriging法による応答曲面F 左:50サンプル、右:15サンプル

50サンプルの結果はかなりよくFを再現していますが、15サンプルの結果は大雑把な形は似ていても、定量的な誤差は大きく、例えばこのまま最大値を見積もると本来2である結果が3以上という結果になる可能性があります。先に最低限のサンプル数の例を述べましたが、それに従うと最低サンプル数は6つになります。しかしながらこの曲面は2次曲面ではなく、より複雑な凹凸を持つ曲面になっており、その場合は倍以上の15サンプルでも不十分のようです。この例では25サンプル程度で最大値・最小値共に誤差が1%以下となります。実際の最適化では応答曲面の正しい形は分かりません。先の最低限のサンプル数の指標で問題のない場合もあれば、明らかに大きな誤差を持つ場合もあります。従ってサンプリングができるだけ多くできればそれに越したことはありません。通常は数十程度のサンプルが必要になると考えておいて差し支えないと思います。

1.5 最適解の計算(多目的遺伝的アルゴリズム:MOGA)

最適化法とは応答曲面(設計変数の定義される範囲での目的関数の形)の最大(あるいは最小)値を求める方法です。目的関数が一つの場合はその最大値が最適解を示しますが、目的関数が多数ある場合は、一般には目的関数間のトレードオフ(相反)の関係から、一つ一つの目的関数の最大値が必ずしも最適解であると見なすことはできません(もしも全ての目的関数が互いに相反しない場合は目的関数の数に限らず、1つの最適解が定まります)。このような問題は多目的最適化問題と呼ばれます。このとき、通常は目的関数のトレードオフを考慮して、パレート最適解という概念を導入し、各目的関数で最大に近づく可能性のある解の集合を候補解として提示します。従って、多目的最適化の結果としては複数の解が生じることになり、最終的にはそれらの解から一つの解を選択する必要があります。

パレート最適解という概念は以下のように定義されています。

「2つの設計変数 x^1, x^2 について、 i 個の全ての目的関数 f で $f_i(x^1) \leq f_i(x^2)$ であり、かついくつかの目的関数で $f(x^1) < f(x^2)$ であるとき、 x^1 は x^2 に優越する」という。また目的関数の集合 F において n 個の設計変数が定義されるとき x^1 に優越する $x^n \in F$ が存在しない場合、 x^1 はパレート最適解という。」

つまり、例えばある1つの設計変数をAと決めたときに、他の全ての設計変数をどのように変えて、全ての目的関数で最大値に最も近い結果が得られる場合、Aはパレート最適解になります。また、ある特定の目的関数を最大(最小)にする解を弱パレート最適解と呼びます。図4は目的関数が2つの場合のパレート最適解を模式的に示したものです。このような図をパレート図とも呼びます。既述のようにパレート最適解は全ての目的に最大限適う設計変数の組み合せであり、複数存在します。パレート解を繋いだ曲線をパレートフロントと呼び、全ての目的関数を最適にする設計変数が存在する極限を示しています。例えば図4の場合ではパレートフロントより右上には候補解はありません。目的関数1に重きを置くなら解Aを、目的関数2に重きを置くなら解Bが選択できます。

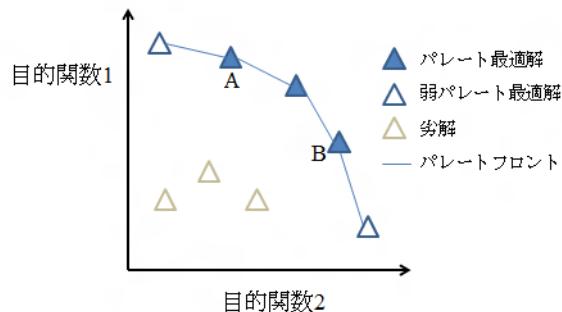


図4 目的関数1, 2の最大化が目的である場合のパレート最適解

さて、計算機で最適化問題を解く方法は多種多様にありますが、いずれにしても私たちは既に応答曲面の形は持っているのでその最大・最小値を探索すればよいことになります。その探索法もさまざまな手法があり、それぞれ一長一短があります。EOoptiでは多目的遺伝的アルゴリズム(Multi Objective Genetic Algorithm; MOGA)[文献 4など]という手法を用いています。

この方法は生物の進化のしかたを模倣するもので、まず、個体という解の候補を設計空間に万遍なく分布させます。それらの個体のうち、最も目的に適う順番に個体の優劣を決めます。ある水準を設け、劣る個体は削除し、逆に非常に優れた個体は解の候補となります。これをエリート選択と呼びます。残った個体、つまり劣っていないけれども驚くほど優れてもいない個体はペアとされ、交叉し、1つの新しい個体を生みます。これは、2つの親となる個体が存在する設計空間の位置を見て、より目的の適う位置に新しい1つの子の個体を置くことに相当します。削除された劣った個体数分の新しい個体が生まれたら1世代の進化が終了し、次世代で再び優劣の評価を行うところから手順を繰り返します。交叉の際には子は親の性質を引き継ぐため、設計空間内での親の位置によっては、局所的な設計空間の頂点や谷底に向かって子の位置が決まってしまうことがあります。このような局所的な頂点や谷底を局所解と呼び、個体はこれらをうまく避けて空間全体の最大、最小値を目指さなくてはなりません。

ん。局所解を避けるためには進化の過程で個体の突然変異を生じさせます。これは親の一方の性質だけ引き継ぐとか、あるいは親の性質を全く引き継がず唐突に設計空間に新しい個体を置く、という方法で行われます。進化を繰り返し、ある程度個体の位置が変わらなくなったら計算が終了し、そのときの個体の設計空間での分布が解となります。つまり、MOGAは設計変数を一つの生物個体として、それらが設計空間の最大値、最小値を目指して進化していく過程を模擬した計算を行うことで、最適な個体、つまり設計変数を求めようとする方法です。MOGAは他の方法と比較して局所解を避けやすく、また広く複雑な設計空間の探索に向いています。一方、他の方法と比べて計算コストが高いという欠点も指摘されていますが、アルゴリズムは並列処理に向いており、近年ではその欠点があまり目立つことはなくなりました。

MOGAの計算で重要なパラメータは1世代で存在する個体数です。多ければ多いほどより目的に適う個体が生まれる可能性が高まり、また候補解が多数得られます。一方、個体数が多いと計算時間は増加します。通常は500程度の個体数があれば十分なようです。もう一つ大事なパラメータとして進化する世代数があります。進化はやがて収束するのですが、世代数が十分でないと進化しきれずに計算が終了してしまうことがあります。設計空間の複雑さや世代あたりの個体数にもよりますが、100世代程度は進化させた方がよいようです。

1.6 最適化結果の見方と役立て方

前節までに最適化という計算の大きな流れと内部の仕組みを概観してきました。ここでは本章冒頭に例示した集合管の問題を例にとり、最適化の結果の見方を考えてみましょう。なお、この最適化の入力ファイルを同梱しますので、2章以降の操作法を読んだ後でも、よろしければEOoptiで実行し、さまざまな結果を実際に御覧ください。

1.1節図1の例は Δp , T_{out} という2つの目的関数、 q_1 , q_2 , T_1 という3つの設計変数がある問題です。この問題に対して、120のサンプリングを行い、最適化計算を行います。EOoptiでは目的関数と設計変数およびその範囲を決め、サンプル数を決めるとき、所定の数だけサンプリングする際の設計変数の組み合わせの表が作られます。これをサンプリング表と呼ぶことにします。サンプリング表にある設計変数の組み合わせについて実験や解析を行い、各設計変数の組み合わせに対する目的関数値が求められ、それをサンプリング表に記述したら、計算のための入力データの準備は終わりです。本例では境界条件の異なる120ケースの熱流体解析を行いサンプリングしました。引き続きKriging法で各目的関数に対する応答曲面を作り、その応答曲面の形を参考にMOGAで応答曲面の最大・最小値を探査します。結果としてサンプルのバラつき状態、応答曲面の形、目的関数に対する設計変数の寄与度、MOGAの結果としての最適解分布、各変数の相関図といったデータがグラフ化され、提示されます。以下、これらの結果を見ていくことにしましょう。

図5はサンプリング時にラテン超方格法により決められた、各サンプルでの設計変数 q_1 の分布を示しています。 q_1 は0.03から0.05という数値をとりますが、ある値に偏っていたりする場合はあまりよいサンプリングとは言えません。万一偏っている場合は再びラテン超方格法の計算を行い、サンプルを作り直した方がよいでしょう。

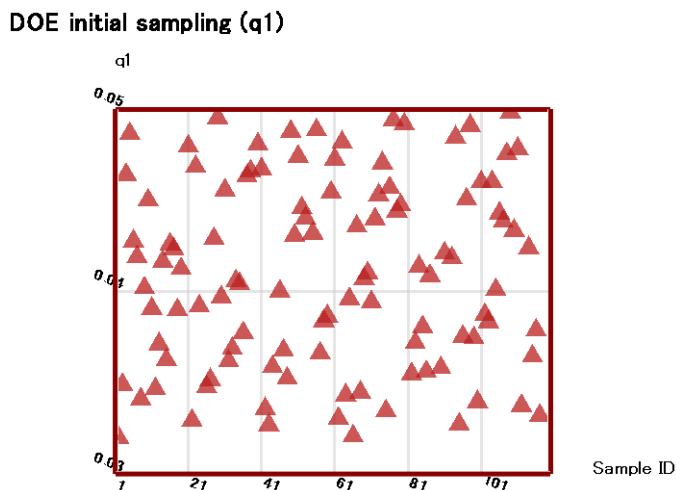
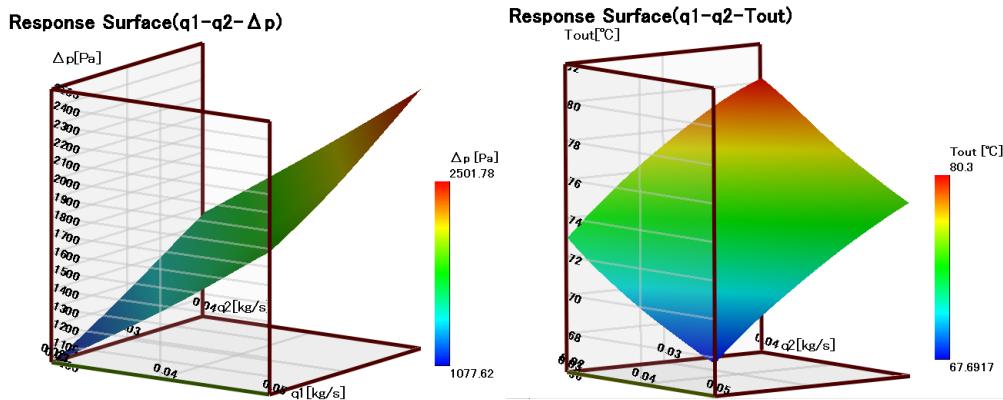


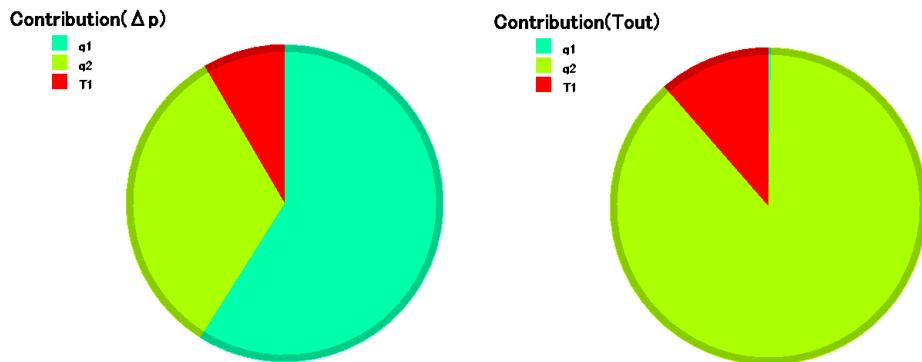
図5 サンプル数120のときの q_1 のサンプル値

次に、Kriging法で応答曲面が作られます。この問題では2つの応答曲面が作られますが、図6に応答曲面表示の例を示します。

図6 Kriging法による応答曲面(左 Δp 、右 T_{out})

ここでは設計変数 q_1, q_2 の変化に対する2つの応答曲面を示していますが、EOoptiでは全ての2つの設計変数の組み合わせで各応答曲面を図6のように図示します。図6左図を見ると、 q_1, q_2 の増加に伴い系の損失が上がっています。また比較をすると同じ変化幅では q_2 よりも q_1 の方が損失を大きく変化させることができます。右図はもう一つの目的関数 T_{out} の応答曲面です。 T_{out} については Δp と逆に q_2 の変化がかなり大きく T_{out} の変化に強く影響しそうであることが分かります。応答曲面の形が分かることで、このように具体的に設計変数の影響のしかたが定性的に非常に理解しやすくなります。

各設計変数の影響を割合(影響度合)として表示するのが、次図7の寄与度分布図です。

図7 寄与度分布図(左 Δp 、右 T_{out})

寄与度は分散分析(Analysis of variance; ANOVA)に基づいて計算されます[文献 4]。全ての設計変数が最小値から最大値まで変化するときに応答曲面の変化の分散をとり、これを全分散とします。また各設計変数のみが変化したときの応答曲面の変化の分散を個別分散とします。寄与度は各個別分散の全分散に対する割合を示します。例えば左図を見ると Δp を最も大きく変化させる可能性があるのは q_1 です。しかしながら右図を見ると q_1 は T_{out} の変化にはほとんど影響をしないようです。往々にして応答曲面間で支配的な寄与度を持つ設計変数が変化し、また寄与度に大きな乖離があると、目的関数間に強いトレードオフの関係が見られるようです。

さて、次にMOGAの計算が終了すると、最適解分布と相関関係の2つの結果図が得られます。本例では1世代512個体で100世代のMOGA計算を行った結果を見てみます。次に示す図8は目的関数の相関関係図です。引かれている線はMOGA計算の結果最終世代に残った個体数分の本数があります。縦軸は最大・最小値で0から1に規格化した目的関数の値を示しています。それぞれをクリックすると線ごとにハイライト表示することができます。図では、 Δp が最小に近いときと T_{out} が最大に近いときの線をハイライト表示しています。この線から2つの目的関数は基本的にはトレードオフの関係にあり、つまり、 Δp を小さくしようとすると、 T_{out} は大きくならざるを得ないということが分かります。線の集

合を見たときに線が密になっている場合、その線の付近にパレート最適解があることを示します。この例では Δp の最小値付近の線が密になっています。これは候補解のほとんどは Δp をできるだけ小さくする解になっていることを表します。しかしながら、Toutに注目すると最大最小値の中間値付近で線が密になっています。これはこの問題で設定された設計変数では Δp を下げることよりもToutを下げる方が難しかったとも見なせます。このように目的関数の相関関係図を見るとトレードオフの有無とその程度を視覚的につかみやすくなります。

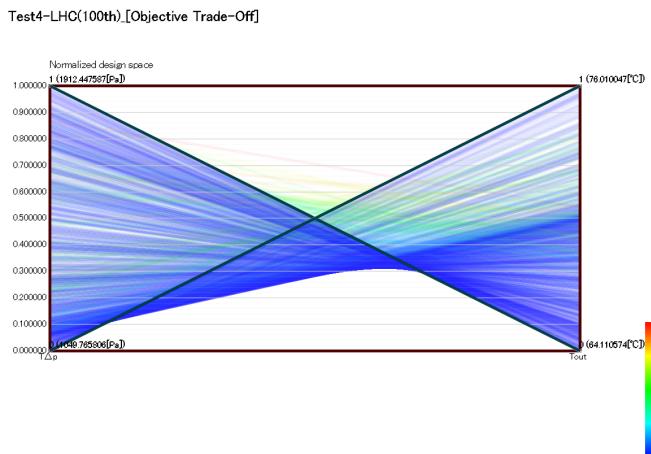


図8 目的関数の相関関係図

もう一つ、目的関数に設計変数も加えた全ての変数の相関関係図を見てみましょう。図9に相関関係図を示します。

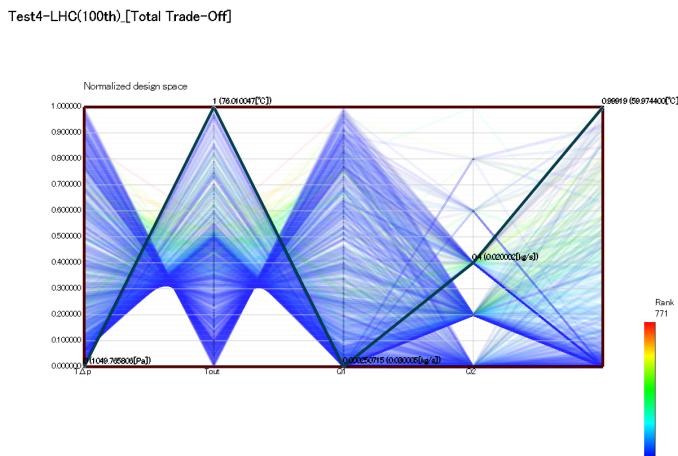


図9 全変数の相関関係図

図8同様、全ての変数は規格化されています。この図はおおまかな設計空間の構造を見るのに適しています。もちろん、応答曲面を見ても設計空間の推測はできますが、設計変数が多くなったりすると、各変数間の関係が見えにくくなります。基本的な意味と見方は図8の目的関数の相関関係図と同じです。図9では Δp が最小のときの線をハイライト表示しています。Toutを低くするという目的と相反しているのは図8と同じですが、具体的に設計変数をどのように組み合わせまた変更すれば損失を低下させるのに有効なのかが見えるようになります。この例ではq1はできるだけ小さくし、q2もあまり大きくせず、かつ流入温度を上げて流体の粘性を下げ摩擦損失を低下させれば系の損失が下がる、という傾向が見えます。言われてみれば当然かもしれません、このような傾向が視覚的に分かります。設計変数の濃淡の強さは寄与度の大きさとも関係します。例えばq2は他の設計変数に比べて線が濃い部分が多いですが、これは Δp もToutも比較的q2の変化に敏感である傾向を示します。また、全ての線を詳細に見ることで、全く想定しなかった設計変数の設定が見つかることがあります。

最後の結果として図10の最適解分布図を見てみます。最適解分布図という名前で呼びますが、多目的最適化の場合は候補解の分布になります。

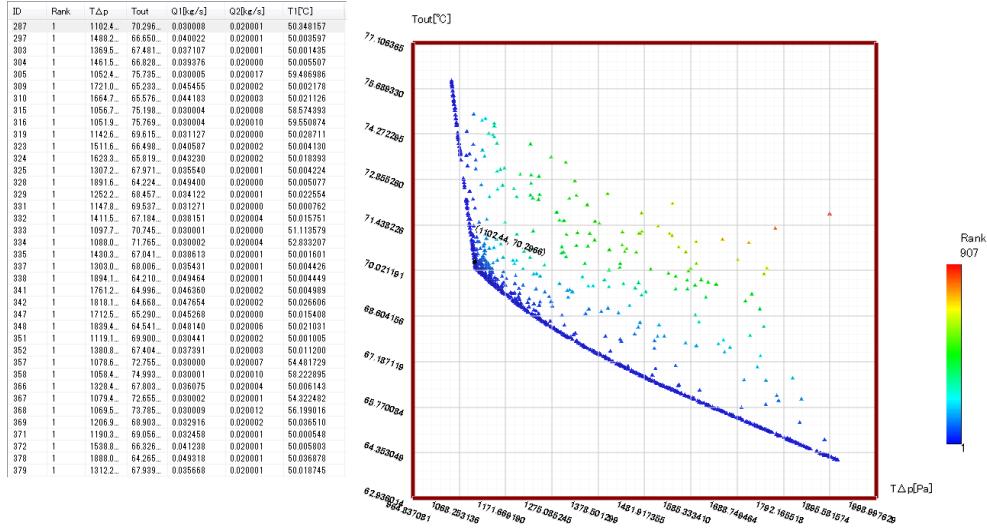


図10 最適解分布図

縦横軸に2つの目的関数をとったグラフに最終世代に残った個体が各点としてプロットされています。なお、EOoptiでは、目的関数が1つの場合は縦軸を目的関数、横軸を個体番号としたグラフ内に個体の分布が示され、目的関数が3つ以上の場合は任意の3つの目的関数をx, y, z軸とした3次元グラフに個体をプロットします。つまり、目的関数が4つ以上の問題では、最適解分布のグラフは複数作成されます。さて、この例では2つの目的関数がトレードオフの関係にあり、基本的にはToutを下げようとすれば Δp が上がります。しかしながら、Toutが70°C程度までは大きく Δp を上げなくてもToutを下げられそうです。点群は左上から右下に「く」の字を描くように分布しています。この「く」の字のラインがパレートフロントです。つまり、このラインよりも左下の領域では2つの目的関数を同時に最小化する解は見出せなかった、ということになります。MOGAで得られる解は目的関数にトレードオフの関係があるときは、このようなパレート最適解の分布です。最終的には私たちがこの点群から解を選択する必要があります。一例として、最も2つの目的を達成するには「く」の字の折れ曲がり点付近の解を選ぶのがよさそうです。EOoptiでは個体の点をクリックすると、個体のリストがポップアップし、選んだ個体の目的関数値、設計変数値が読み取れるようになっています。図10では、 $\Delta p = 1102.440$, $Tout=70.297$ となる解を選択しました。このとき、 $q_1=0.030$, $q_2=0.020$, $T1=50.348$ であり、例えばこの選択解からは、想定される3つの設計変数の最小値付近を狙うと2つの目的関数をバランスよく満たす結果が得られそうだ、ということが分かります。もちろん、この結論は一つの選択であり、またサンプリングデータの解析精度やサンプル数によって実現象と比較して誤差が含まれます。従って、実際に結果を定めるには本結論の検証も必要かと思いますが、しかしながら、設計空間に基づいて設計変数の目的関数に対する定性的な影響は評価でき、この知見は実設計の大きな助けとなると思います。

1.7 Efficient Global Optimization

最適化問題を実用的に解くためには、時間やコストによって取得するサンプル数に制限がある場合が多いため、より少ないサンプル数で最適解を求めることが望ましいことはいうまでもありません。特に、CFDのような非線形問題を扱う場合、1つのサンプル値を取得するまでに時間がかかるケースが多く、効率的にサンプリングすることが最適解を得るまでの時間コストを削減する上で非常に重要な点になります。

Efficient Global Optimization (EGO) はJonesら[5] が提案した大域的最適解を探索する手法で、応答曲面から最適解を取り得る期待値を推定します。この期待値が高い場所にサンプル点を追加することで、応答曲面の精度向上が望めるわけですが、EGOではこの期待値の指標としてExpected Improvement(EI) を用いています。EIは、現在得られているサンプル値の最小値・最大値に対してサンプル点を追加したときに、新たな最小値・最大値となり得る度合いを表しています。EIに基づいて新たなサンプル点を追加し、再び予測を行うという作業を繰り返すことで、限られたサンプル数で、できるだけ精度の高い応答曲面を生成することが可能になってきます。

EOoptiでは、応答曲面生成時にEIを算出しておらず、MOGAを用いることで、設定した設計変数の範囲内におけるEIの最大値を探索することができます。EIが最大値をとる設計変数値とそれに対する目的関数値を新たなサンプル値として追加し、応答曲面の生成を繰り返すことで、効率的に応答曲面の精度を高めることができます。

ここで、1.4 代理モデル(Kriging法)でも紹介した目的関数 $F=\sin(x)+\sin(y)$ を例にとり、EIを用いてサンプル点を追加しながら生成した応答曲面と、ラテン超方格法のみで生成した応答曲面との比較をしてみましょう。

図11は初期サンプル数を10とし、EIの最大値を取る設計変数値を新たなサンプル値として20回繰り返して追加したときの応答曲面、図12は初期サンプル数を30とし、ラテン超方格法を用いて得られた設計変数を用いた応答曲面です。

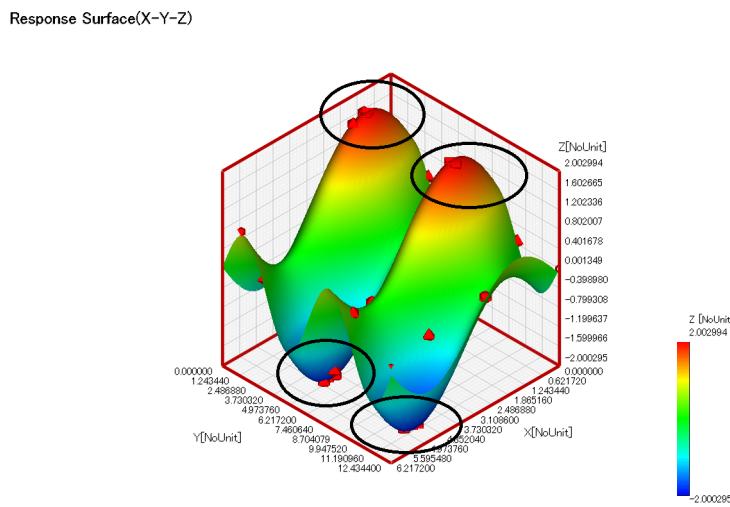


図11 EGOによる応答曲面
最小値・最大値付近にサンプル点が追加されている
(サンプル数 : 30 = 初期サンプル数 : 10+追加サンプル数 : 20)

Response Surface(X-Y-Z)

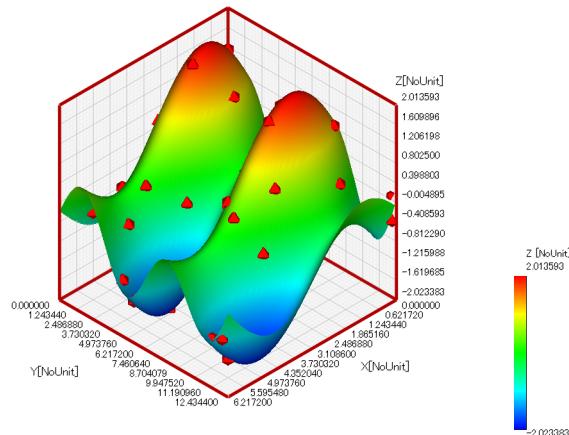


図12 ラテン超方格法を用いた応答曲面(サンプル数 : 30)
応答曲面全体は近似されているが、最小値・最大値の精度が図11に比べて落ちる

両者ともに総サンプル数は30と同じですが、EIを用いた場合、黒丸部分に見られるように、目的関数の最小値・最大値付近にサンプル点が追加され、応答曲面のピーク値をより良く捉えられているのが確認できます。一方、ラテン超方格法によるサンプリングでは、全体の形状は近似的に捉えられていますが、最小値・最大値付近の値はEGOに比べて精度が落ちているのがカラーバーの範囲を見ても分かります(-2.0 <= sin(x)+sin(y) <=2.0)。応答曲面の精度が最適化の結果に影響することを考えると、EIを指標としてサンプル点を追加することで、最適解の精度向上が期待できます。

EI(X-Y-Z)

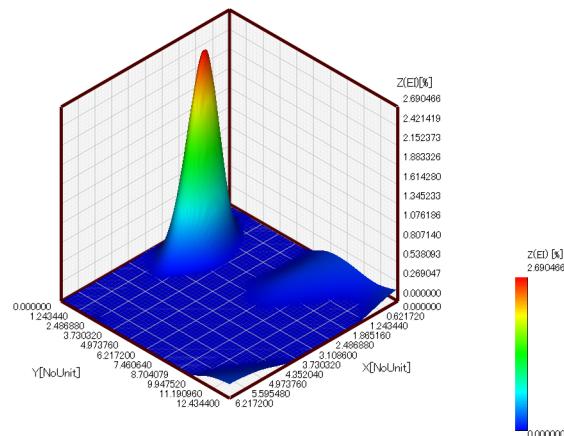


図13 EI値の分布図（サンプル数 : 16 応答曲面最大値探索時）
目的関数Fの最大値付近のEIが大きい

EOoptiでは、EIの他に、RMSE(Root Mean Squared Error : 二乗平均平方根誤差)を指標とし、MOGAを用いることで、この値が最大値をとる設計変数の組み合わせを、新たなサンプル値の候補として提示することもできます。EIは新たな最大値・最小値となり得る期待度を表すのに対し、RMSEは既存の個々のサンプル点での誤差を表すものです。応答曲面のピーク付近だけでなく、応答曲面全体の精度を上げたい場合、例えば寄与度分布の精度を上げたい場合は、RMSEを指標としてサンプル点を追加することが推奨されます。

RMSE(X-Y-Z)

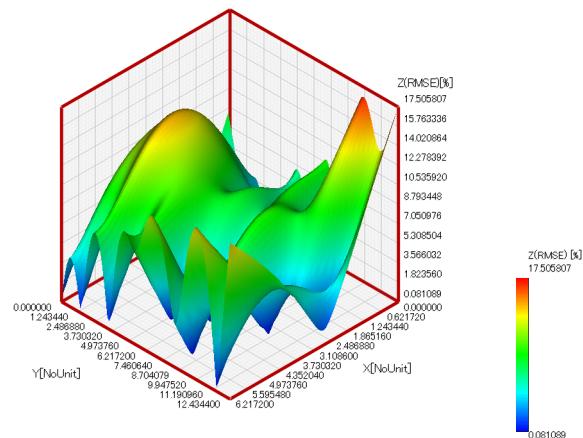


図14 RSME値の分布図(サンプル数：16)
サンプル点の値から離れた箇所ほど値が大きい

実際にサンプル値を追加するためには、提示された設計変数の値に対する目的関数値を取得する必要があるため、この値を自動で取得するためには、事前に値取得のための定義を行う必要があります。目的関数が熱流体解析結果である場合の実用例として、VBインターフェースを用いた事例を第3章 例題に紹介していますので、ご興味がある方は併せてご参照ください。

なお、本節で例示した比較の結果は、Chapter1\EGO Sample\フォルダ内のEGO_Sample.xlsxのVBAを実行することで確認することができます。

ここまで簡単ではありますがEOoptiで得られる結果を見てきました。ここで気付くのは、最適化という行為は設計空間を把握することであり、そこから想定した設計変数の振る舞いを調べ、その考察によって目標となるデザイン(目的関数)の実現を目指すこととも言えます。この設計空間を知るということがとても大切なことで、効率的にデザインを改善することはもちろん、場合によっては思いも寄らない改善のための設計変数を見つけることができるることも十分にあります。しかしながら、ソフトウェアが実際的に支援できるのは設計変数の振る舞いを図化して理解しやすい形で表示するところまでです。再掲になりますが重要なことなので今一度述べたいのですが、最適化計算だけでは、ある1つの答え、つまり最もよいデザインを得ることはできません。計算結果は設計空間に基づく設計変数の変更のしかたや選び方に示唆を与えるものであり、最終的な解つまりデザインを決めるのは私たちです。明確な設計目標を持ち、それを実現するさまざまなアイデアを持つエンジニアの皆さんとの思考、発見、決断のお手伝いをすることが、EOoptiの本質的な役割です。

長くなりましたが、これで最適化の概要については終わりたいと思います。以下、第2章ではEOoptiの操作に関する情報を記し、第3章では具体的な応用例として例題を掲げます。EOoptiを是非有効活用頂けることを祈って本章の稿を閉じたいと思います。

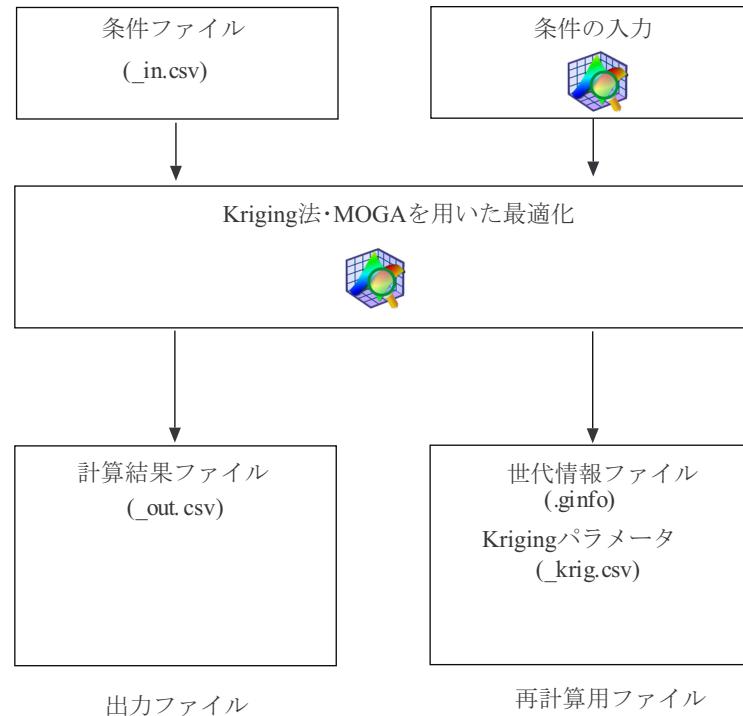
参考文献

1. R, L, Iman; W, J, Conover. A distribution-free approach to inducing rank correlation among input variables. *Communications in Statistics B*11, 1982, p.311-334.
2. S, Jeong; M, Murayama; K, Yamamoto. Efficient Optimization Design Method Using Kriging Model. *Journal of Aircraft* 42-2, 2005, p.413-420.
3. 高阪宏行. クリギングとその地理的応用. 日本大学文理学部自然科学研究所研究紀要 No.34, 1999, p.27-35.
4. S, Jeong; K, Chiba; S, Obayashi. Data Mining for Aerodynamic Design Space. *Journal of Aerospace Computing, Information and Communication* vol.2, 2005.
5. D. R. Jones; M. Schonlau; W. J. Welch. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization* 13, p455?492, 1998.

第2章 Extension Option(Optimization)の 概要

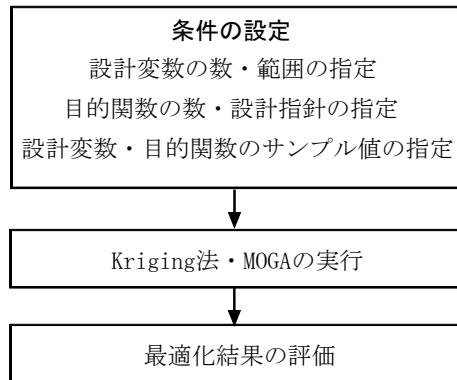
2.1 Extension Option(Optimization)関連ファイル

EOoptiでは、読み込んだ条件ファイル(_in.csv)、またはEOopti内で設定した条件を基に、Kriging法・MOGAを用いた最適化を行い、計算結果をcsv形式で出力します。



2.2 Extension Option(Optimization)の操作手順

EOoptiを用いて最適化を行う手順を以下に示します。



設計変数・目的関数のサンプル値の設定を行えば、後は自動的にKriging法・MOGAを用いた最適化を行うことができます。最適化結果の評価では、EOopti内で表示されるグラフを基に、ユーザーが最適化の結果を評価します。結果の評価方法については、[第1章 1.6 最適化結果の見方と役立て方](#)を参照してください。

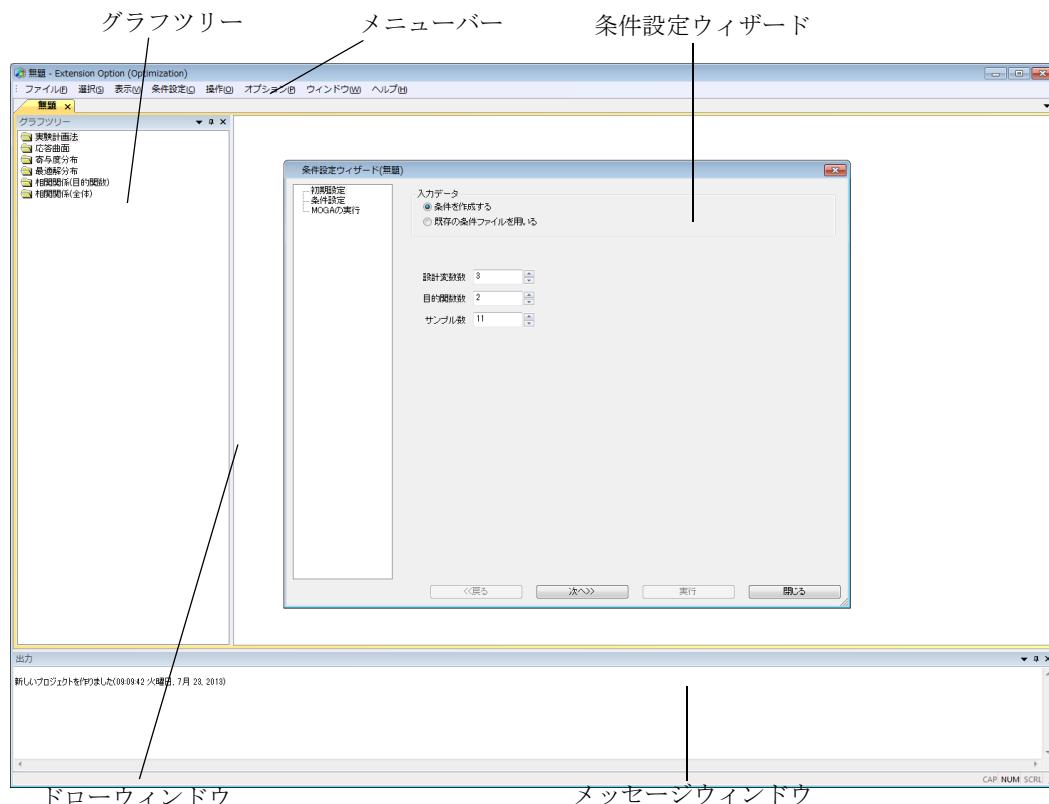
2.3 操作手順の具体例

(1) Extension Option(Optimization)の起動

スタートメニューから

[スタート] - [プログラム] - [Cradle]内にある起動ツールを起動させ、[Extension Option(Optimization)]を起動します。

起動後の画面構成を下図に示します。



- **メニューバー**
EOoptiのメニューが用意されています。
- **グラフツリー**
サンプル値の分布グラフや、Kriging法による応答曲面の生成、MOGAの計算結果のグラフを表示します。グラフを選択すると、ドローウィンドウにはそのグラフが表示されます。
- **ドローウィンドウ**
グラフツリーで選択されたグラフを表示します。ドローウィンドウでのマウスの各ボタンには下表の機能が付加されています。また、グラフの面・点をダブルクリックすることで、視点中心をその場所へ移動します。

操作	2ボタンマウス		3ボタンマウス	
	1ボタン モード	2ボタン モード	3ボタン モード(Ctrl)	3ボタン モード
選択				
ポップアップメニュー				
回転	メニューから 選択	Ctrl +	Ctrl +	
拡大・縮小	メニューから 選択	Ctrl +	Ctrl +	
移動	メニューから 選択	Ctrl +	Ctrl +	
2D回転	メニューから 選択	Ctrl + Shift +	Ctrl + Shift +	Shift +

注. 2次元グラフでは、回転、2D回転は行えません。

メニューの[操作]より[オペレーション]を選択すると、下のようなダイアログが表示され、上記のマウス操作系とはまったく別の操作系を利用することができます。



利用可能な操作系は、以下の通りです。

操作系の名称	移動	回転	ズーム
CRADLE 1ボタンモード	メニューから選択	メニューから選択	メニューから選択
CRADLE 2ボタンモード	Ctrl+マウス左ボタン+マウス右ボタン	Ctrl+マウス左ボタン	Ctrl+マウス右ボタン
CRADLE 3ボタンモード(CTRL)	Ctrl+マウス中ボタン+マウス右ボタン	Ctrl+マウス中ボタン	Ctrl+マウス右ボタン
CRADLE 3ボタンモード	マウス中ボタン+マウス右ボタン	マウス中ボタン	マウス右ボタン
TYPE A	Ctrl+Alt+マウス中ボタン	Ctrl+Alt+マウス左ボタン	Ctrl+Alt+マウス右ボタン
TYPE B	マウス中ボタン	マウス中ボタン+マウス右ボタン	回転操作後にマウス右ボタンだけを離す
	またはマウス中ボタン	またはマウス中ボタン+マウス左ボタン	または回転操作後にマウス左ボタンだけを離す
TYPE C	Ctrl+マウス中ボタン	マウス中ボタン	Shift+マウス中ボタン
TYPE D	Ctrl+マウス右ボタン	Ctrl+マウス左ボタン	Ctrl+Shift+マウス中ボタン あるいは Ctrl+マウスホイール
TYPE E	Shift+マウス中ボタン	マウス中ボタン	Ctrl+マウス中ボタン
TYPE F	マウス中ボタン+マウス右ボタン	マウス中ボタン	マウス中ボタン+マウス左ボタン
TYPE G	マウス中ボタン	Shift+マウス中ボタン	Ctrl+マウス中ボタン または ホイール

また、キーボードには下表の機能が付加されています。

キー	操作
X	Z Y Y-Z面に表示をセットします。
Y	Z X X-Z面に表示をセットします。
Z	Y X X-Y面に表示をセットします。
Shift+X	Z Y Y-Z面に表示をセットします。
Shift+Y	Z X X-Z面に表示をセットします。
Shift+Z	Y X X-Y面に表示をセットします。
A	現在の視点に一番近い軸に、表示をセットします。
Delete	表示された文字列の削除 グラフツリーに登録されているグラフの削除
C + 頂点をクリック	表示の中心の移動
Ctrl+C	表示の中心の移動

- メッセージウィンドウ
EOoptiの処理内容をテキスト表示します。
- 条件設定ウィザード
EOoptiの起動と同時に表示されます。このウィザード内で、最適化を行うための条件を設定します。

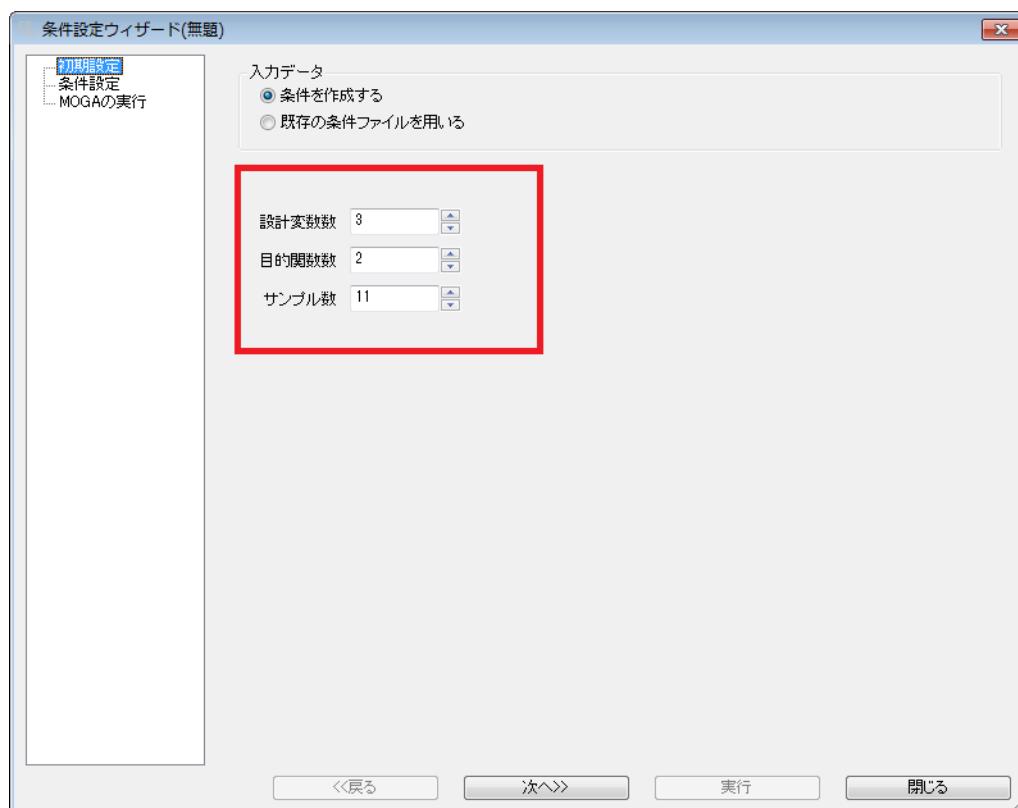
(2) 初期設定

EOoptiで最適化を行うためには、条件設定ウィザードで最適化を行うための条件を設定する必要があります。EOopti起動後、条件設定ウィザードが表示されるので、この中で最適化の条件を設定していきます。

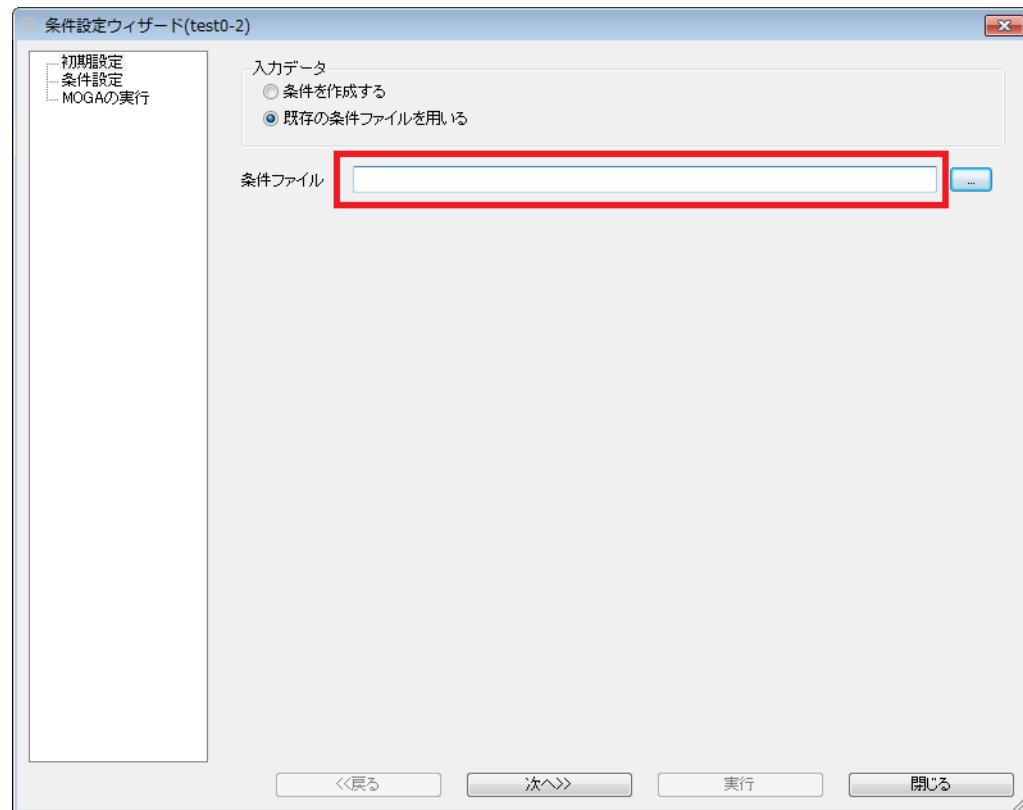
まず、初期設定では、

1. 最適化を実行するための入力データを作成するか
2. 既存の入力データを用いるか

を選択します。1の場合は[条件を作成する]を選択し、設計変数の数と、目的関数の数、およびサンプリング数を指定します。



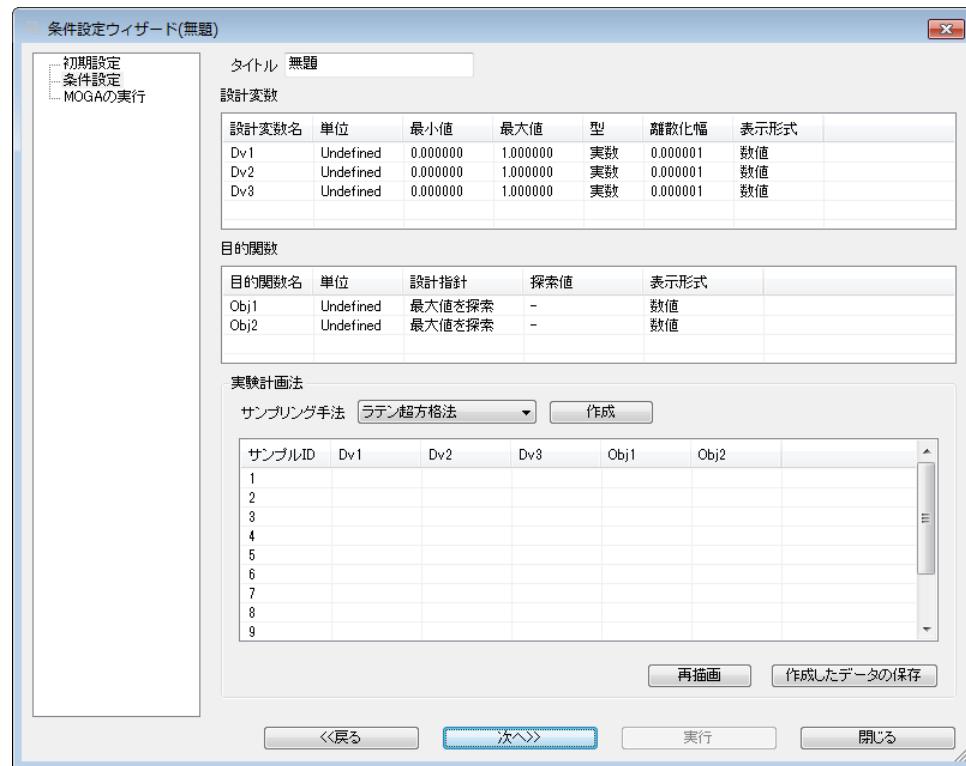
2の場合は、[既存の条件ファイルを用いる]を選択し、条件ファイル(_in.csv)のフルパスを指定します。



(3) 条件設定

次へ>>をクリックすると、[条件設定]ページに移動します。

[初期設定]で[条件を作成する]を選択した場合、デフォルトでは下図のような表示になります。



[条件設定]ページのキーボードには、下表の機能が付加されています。

Tab	リストのアイテムが編集状態のときに押すと、編集状態が右側に移動します。
Shift+Tab	リストのアイテムが編集状態のときに押すと、編集状態が左側に移動します。
Ctrl+Tab	リストのアイテムが編集状態のときに押すと、編集状態が下側に移動します。
Shift+Ctrl+Tab	リストのアイテムが編集状態のときに押すと、編集状態が上側に移動します。
Ctrl+C	リストのアイテムが選択状態のときに押すと、選択した行の文字列全てをコピーします。
Ctrl+X	リストのアイテムが選択状態のときに押すと、選択した行の文字列全てを切り取ります。
Ctrl+V	リストのアイテムが選択されている箇所に、コピー、または、切り取られた文字列を貼り付けます。
Ctrl+Z	Ctrl+X、Ctrl+Vによって変更されたリストの文字列を元に戻します。

注. 異なるリスト間のコピー、切り取り、貼り付けは行えません。例えば、設計変数のリストでコピーした内容を、目的関数のリストに貼り付けることはできません。

条件を入力するためには、リストのアイテムをテキスト編集状態にします。

アイテムをダブルクリックすると、下図のように、数値や文字列を入力することができます。

設計変数名	単位	最小値	最大値
フィン枚数	Undefined	0.0	1.0
Dv2	Undefined	0.0	1.0
Dv3	Undefined	0.0	1.0

各項目の設定内容は以下の通りです。

- [タイトル]

プロジェクトの名前を入力します。デフォルトでは[無題]です。

- [設計変数]リスト

[設計変数名]

設計変数の名前を入力します。デフォルトのままでも構いませんが、最適化結果の評価上、入力することをお勧めします。

[単位]

設計変数の単位を入力します。この単位は、流体解析で用いる単位とは異なり、結果を表示する上で用いるものです。デフォルトのままでも最適化の計算結果に影響は与えませんが、入力した方が、グラフ上での結果が評価しやすくなります。

[最小値]・[最大値]

ラテン超方格法によって設計変数のサンプル値を与えるときの、設計変数の範囲を指定します。

[型]

設計変数の数値型を指定することができます。ここで設定した設計変数の型が、実験計画法のサンプル値、及び最適解の値に反映されます。デフォルトは[実数]です。

[離散化幅]

離散化幅は、設計変数の最小値・最大値の範囲内の中で取りうる値の間隔に相当し、最適化の結果にもこの離散化幅が適用されます。例えば、ある設計変数の最小値を0.0、最大値を1.0、離散化幅を0.1とした場合、最適解の取りうる値は0.0, 0.1, 0.2…1.0と0.1刻みの値になります。用途としては、最適化の結果を評価する際、切りの良い設計変数の値だけを出力させたいような場合に用います。

[表示形式]

グラフに表示される設計変数値の表示形式を指定します。[指標]、または[数値]を選択できます。

デフォルトは[プログラムの詳細設定]-[画面の表示設定]-[数値の設定]で設定されている形式が適用されます。

- [目的関数]リスト

[目的関数名]

目的関数の名前を入力します。デフォルトのままでも構いませんが、最適化結果の評価上、入力することをお勧めします。

[単位]

目的変数の単位を入力します。設計変数リスト同様、最適化の計算結果に影響は与えませんが、入力した方が、グラフ上での結果が評価しやすくなります。

[設計指針]

最適解を探索する方向を指定します。選択項目には、[最大値を検索]、[最小値を検索]、[任意値を検索]があります。[最大値を検索]、[最小値を検索]を指定した場合は、各目的関数の最大値、または最小値に極力近い設計変数を求めるように計算が行われます。[任意値を検索]を指定した場合は、[検索値]で指定した目的関数値に極力近い設計変数を求めるように計算が行われます。

[探索値]

[設計指針]で[任意値を探索]を指定したときに入力が可能になります。ここで指定した目的関数値に極力近い設計変数を求めるように計算が行われます。

[表示形式]

グラフに表示される目的関数値の表示形式を指定します。指標、または数値を選択できます。

デフォルトは[プログラムの詳細設定]-[画面の表示設定]-[数値の設定]で設定されている形式が適用されます。

- [実験計画法]

サンプリング手法の作成をクリックすると、指定されたサンプリング手法を用いて、設計変数リストで指定した最小値、最大値の範囲内のサンプル値を自動生成します。生成された値は実験計画法のリスト内に出力され、サンプル値の散布図が下図のようにグラフツリーに追加されます。

[ラテン超方格法]

各設計変数内で値が重複しないサンプル値を生成します。値の間隔幅は、

$$(Max - Min) / (Sample - 1)$$

Max : 設定設計変数最大値

Min : 設定設計変数最小値

Sample : サンプル数

で求められます。設計変数が実数として扱えるなど、主に設計変数の値のとり方に制限がない場合に用います。

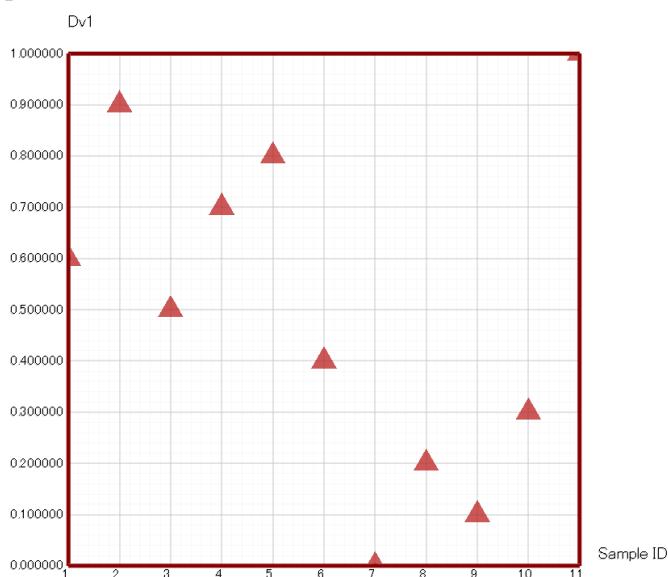
[ランダムサンプリング]

同じ設計変数の組み合わせが生成されないように、値を無作為に選んだサンプル値を生成します。

各サンプル値の間隔幅には設定された離散化幅を用います。設計変数の値のとり方に制限があり、離散化幅を適用したサンプル値を生成したい場合に用います。



DOE initial sampling (Dv1)



自動生成後のサンプル値も、リストのアイテムをテキスト編集状態にすることで値を変更することができます。再描画をクリックすると、変更されたサンプル値を反映した散布図が再描画されます。

設計変数のサンプル値が自動的に生成される一方で、目的関数の値は次図のように空欄のままでです。これは、目的関数の値は、流体解析等で得られた既知の値を入力する必要があるため、設計変数のサンプル値に対する目的関数の値を別途用意する必要があります。

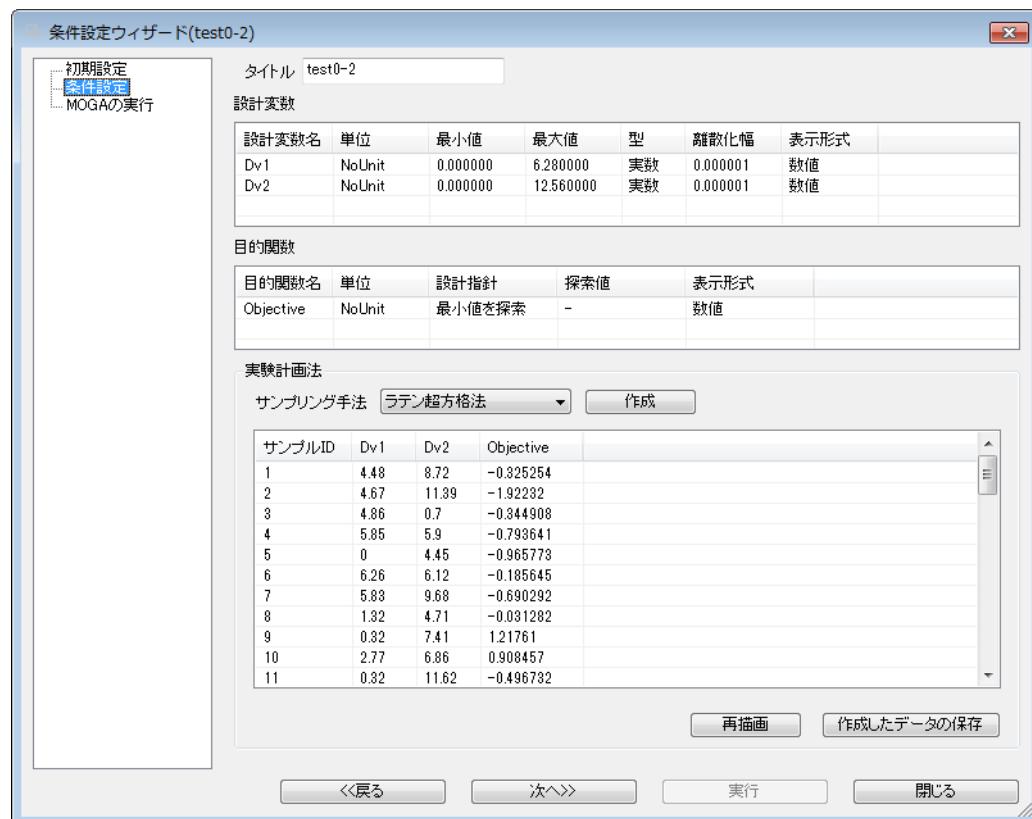
この時点でお一時作業を中断し、目的関数のサンプル値が得られた後に作業を再開したい、というような場合は、**作成したデータの保存**から、[条件設定]で設定した情報を条件ファイル(_in.csv)として保存することができます。

サンプルID	Dv1	Dv2	Dv3	Obj1	Obj2
1	0.400000	0.100000	0.900000		
2	0.900000	0.300000	1.000000		
3	0.600000	0.000000	0.200000		
4	0.000000	0.800000	0.000000		
5	0.800000	0.400000	0.500000		
6	1.000000	0.500000	0.400000		
7	0.200000	0.200000	0.300000		
8	0.100000	0.900000	0.700000		
9	0.700000	0.700000	0.800000		

目的関数のサンプル値が得られた後、保存しておいた条件ファイルを読み込むことで、この[条件設定]ページから設定を続けることができます。また、条件ファイルはcsvファイルなので、他の編集ソフトを用いて目的関数のサンプル値を入力した後、条件設定ウィザードに読み込むことも可能です。

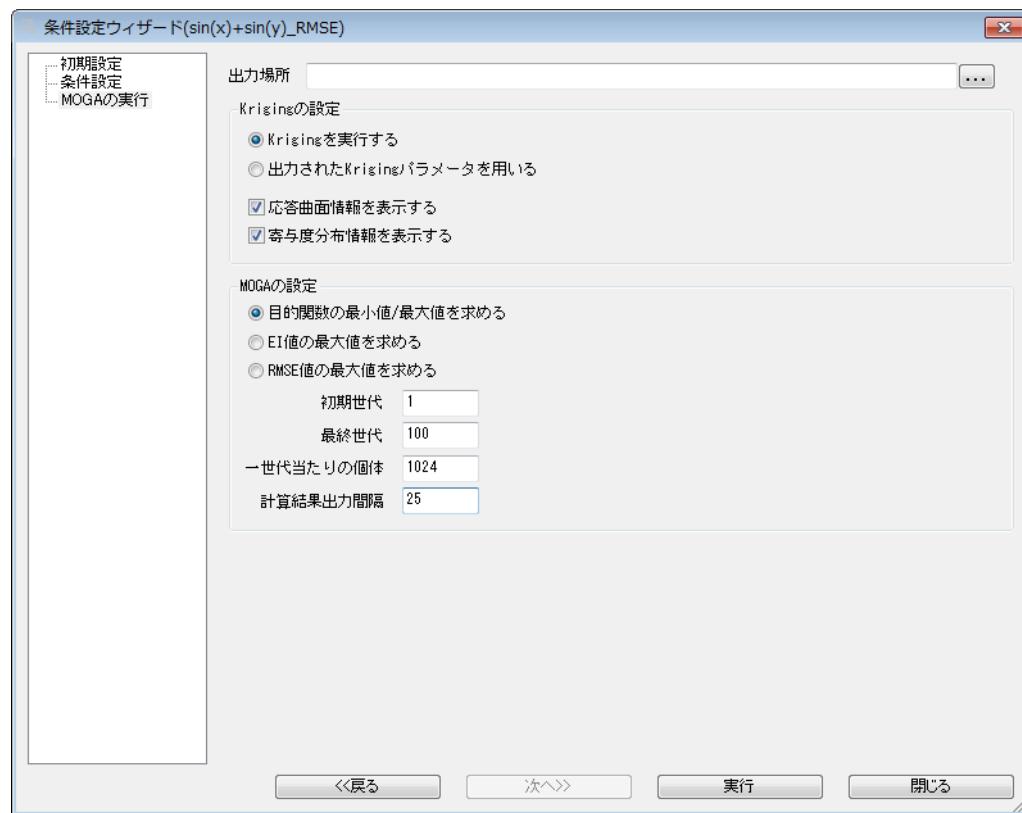
条件ファイル(_in.csv)の読み込みは、メニューの[ファイル]-[開く]からか、ドローウィンドウ、または条件設定ウィザードに条件ファイルをドラッグ&ドロップすることで行えます。

ユーザーフォルダ内のOptimization\Chapter2\test0-2_in.csvを読み込むと、次の図のような表示になります。



(4) MOGAの実行

次へ>>をクリックすると、[MOGAの実行]ページに移動します。



各項目の設定内容は以下の通りです。

- [出力場所]

Kriging法・MOGAによる最適化を行うと、計算結果がファイルとして出力されます。これらのファイルは、[出力場所]で指定したフォルダ内に出力されます。

注. 条件ファイルを読み込むと、このファイルが存在するフォルダ名が出力場所に指定されます。

- [Krigingの設定]

MOGAでは、最適化中に生成された設計変数の適合度を評価するために応答曲面を用います。

[Krigingを実行する]を選択すると、MOGAが開始される前に、[条件設定]ページで入力されたサンプル値を基に、Kriging法による応答曲面を生成します。[出力されたKrigingパラメータを用いる]を選択すると、応答曲面の生成は行わず、既に一度Kriging法を実行して出力されたKrigingパラメータを用いて最適化を行います。

[応答曲面情報を読み込む]、[寄与度分布の情報を読み込む]

チェックをONにすると、Kriging法による応答曲面生成後、応答曲面、寄与度分布のグラフがグラフツリーに追加され、ドローウィンドウ上に表示されます。OFFにすると、これらのグラフは実行後に追加されませんが、出力されたKrigingファイル(_krig.csv)、計算結果ファイル(_out.csv)を読み込むことで、表示させることができます。これらのファイルは、メニューの[ファイル] - [開く]から読み込むか、ドローウィンドウにファイルをドラッグ&ドロップすることによって表示されます。

- [MOGAの設定]

[目的関数の最小値/最大値を求める]

この項目が選択された状態で、**実行**をクリックすると、設定されている設計変数・目的関数のサンプル値を用いたMOGAが実行されます。

最小値、最大値どちらを求めるかは、目的関数の**設計指針**によって変わります。

[EI値の最大値を求める]

この項目が選択された状態で、**実行**をクリックすると、EI値の最大値を求めるMOGAが実行されます。

EI値の最大値をとる設計変数値と、それに対する目的関数の値を新たなサンプル値として追加することで、応答曲面精度向上が期待できます。

[RMSE値の最大値を求める]

この項目が選択された状態で、**実行**をクリックすると、RMSE値の最大値を求めるMOGAが実行されます。

RMSE値の最大値をとる設計変数値と、それに対する目的関数の値を新たなサンプル値として追加することで、応答曲面精度向上が期待できます。

EI値、RMSE値の詳細については、[1.7 Efficient Global Optimization](#)を参照してください。

[初期世代]

MOGAの計算開始サイクル数です。デフォルトは1です。2以上の値を指定すると、リスタート計算を行うことができます。(リスタート計算を行うためには、世代情報ファイル(ginfoファイル)が必要です。)

[最終世代]

MOGAの最終計算サイクル数です。デフォルトは100です。この値が大きくなるほど、メモリを多く消費し、計算時間が長くなります。

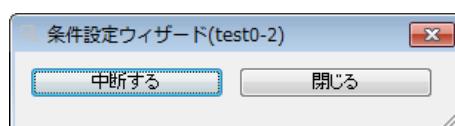
[一世代当たりの個体数]

計算中の各世代でランダムに生成する個体数で、最適解分布グラフの点数、相関関係グラフの線数に相当します。デフォルトは1024です。この値が大きくなるほど、メモリを多く消費し、計算時間が長くなります。

[計算結果出力間隔]

最適化の結果を出力する世代間隔数を指定します。ここで指定した世代間隔毎に、計算結果ファイル(_out.csv)が出力され、計算中のグラフの情報が更新されます。

条件設定で全ての条件が入力されていれば、実行を押すだけでKriging法・MOGAによる最適化が開始します。実行中、[条件設定ウィザード]は下図のように縮小表示されます。計算を途中で止めたい場合は、中断するをクリックしてください。[条件設定ウィザード]は、元の大きさに戻り、クリックしたときに計算していた世代の結果がグラフツリーに追加されます。



(5) 結果の確認

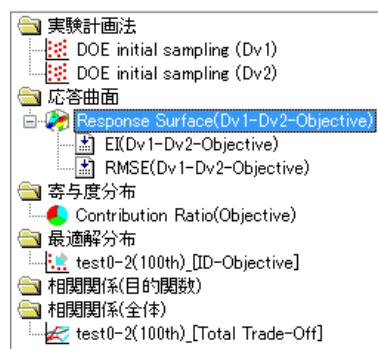
Kriging法・MOGAの実行が終了すると、計算結果がグラフツリーにグラフとして追加されます。

ここでは、追加されるグラフの種類について簡単に説明します。

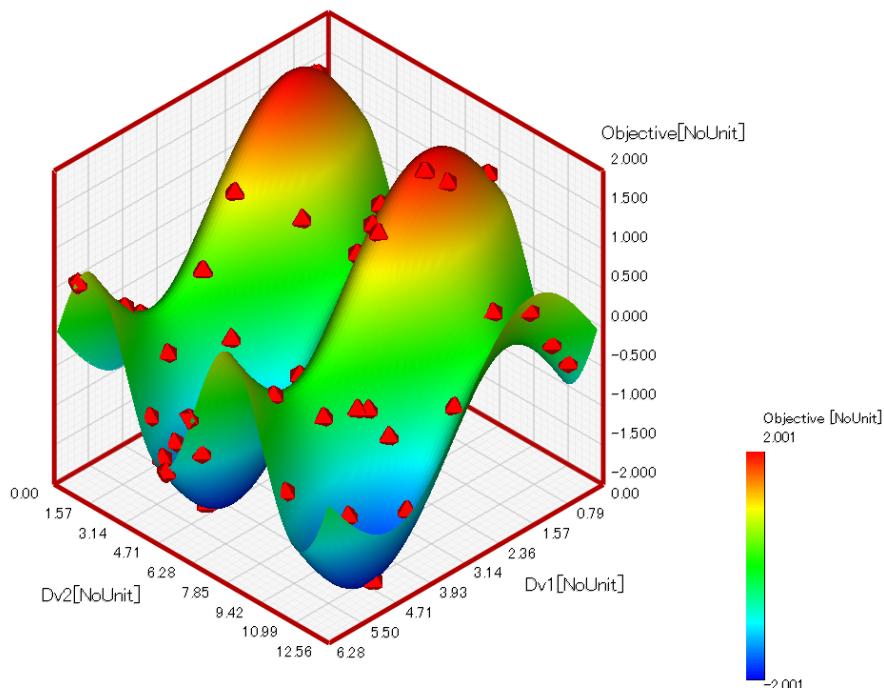
注. 以下に表示されるグラフは、Optimization\Chapter2\test0-2_in.csvを用いた計算結果のグラフです。

[応答曲面]

Kriging法によって生成された応答曲面が表示されます。曲面と同時に表示される赤点は[条件設定 ウィザード]の[条件設定]で指定した設計変数のサンプル値です。このサンプル点の表示のON/OFFは、ポップアップメニューの[応答曲面の表示設定]で変更できます。

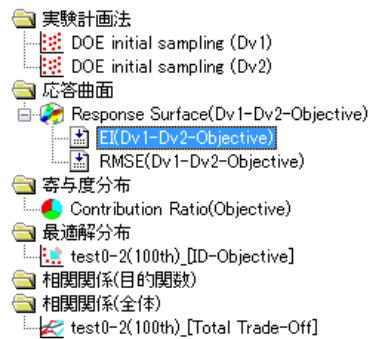


Response Surface(Dv1-Dv2-Objective)

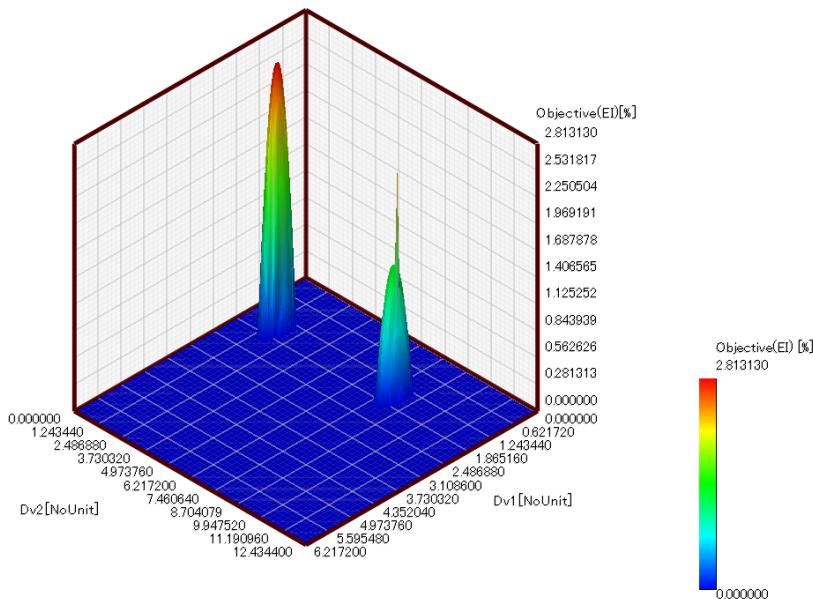


[EI]

[プログラムの詳細設定]-[Krigingの詳細設定]で[EI値の分布をグラフに追加する]のチェックがONになっているとき、グラフツリーの応答曲面の下に"EI"とタイトルが付いたグラフが追加されます。この項目を選択すると、算出されたEI値の分布を確認することができます。



EI(Dv1-Dv2-Objective)



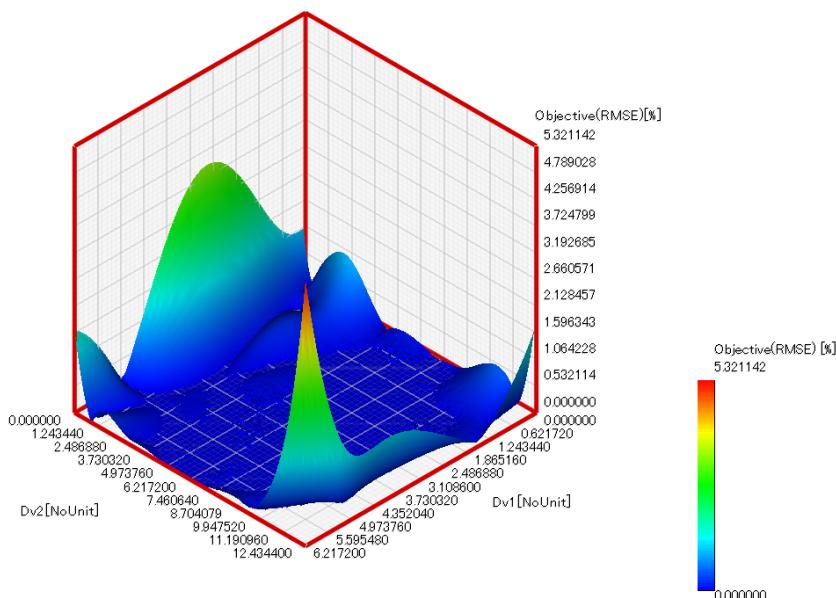
1.7 Efficient Global Optimizationでも紹介されているように、EIは目的関数の最小値・最大値が改善できる度合いを表しています。EI値が高い箇所の設計変数値を用いたサンプル点を追加することで、応答曲面の精度向上が期待できます。ここで表示される分布はEI値をサンプル点の最大値と最小値の差で割った比率(%)で表しています。

[RMSE]

[プログラムの詳細設定]-[Krigingの詳細設定]で[RMSE値の分布をグラフに追加する]のチェックがONになっているとき、グラフツリーの応答曲面の下に"RMSE"とタイトルが付いたグラフが追加されます。この項目を選択すると、算出されたRMSE値の分布を確認することができます。



RMSE(Dv1-Dv2-Objective)



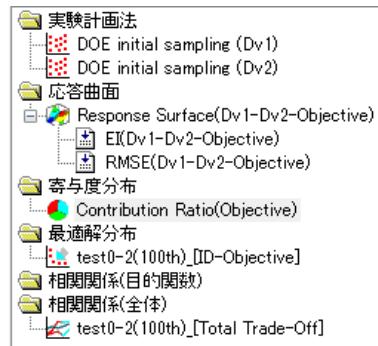
こちらの分布もEIと同様、サンプル点の最大値と最小値の差で割った比率(%)で表しています。

曲面上をダブルクリックすると、その曲面上の座標値が表示されます。

注. 設計変数が3つ以上の場合、グラフ表示に用いられていない設計変数の値には、デフォルトでは各設計変数の最小値・最大値の中間値を使用しています。中間値以外の設計変数の値を用いたい場合は、ドローウィンドウ上でポップアップメニュー[目的関数・設計変数一覧]で、設計変数の値を指定することで表示できます。詳しくは、第4章 [ポップアップメニュー(ドローウィンドウ)] - [目的関数・設計変数一覧]を参照してください。

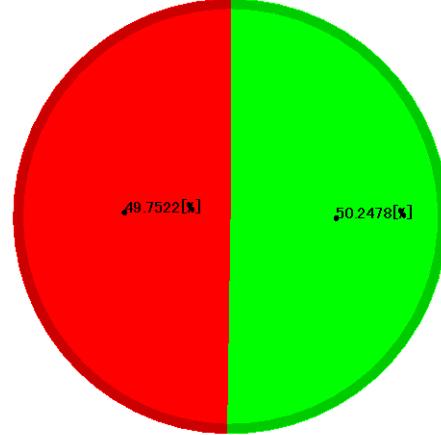
[寄与度分布]

各目的関数の寄与度を分散分析によって求めたものを、下図のような円グラフで表示します。各設計変数の扇形領域をクリックするとその扇形が選択状態になります。設計変数の数が多い場合は、必要に応じて、選択領域の表示・非表示をメニューの[表示]から行ってください。領域をダブルクリックすると、寄与度を%で表示します。



Contribution(Objective)

■ Dv1
■ Dv2

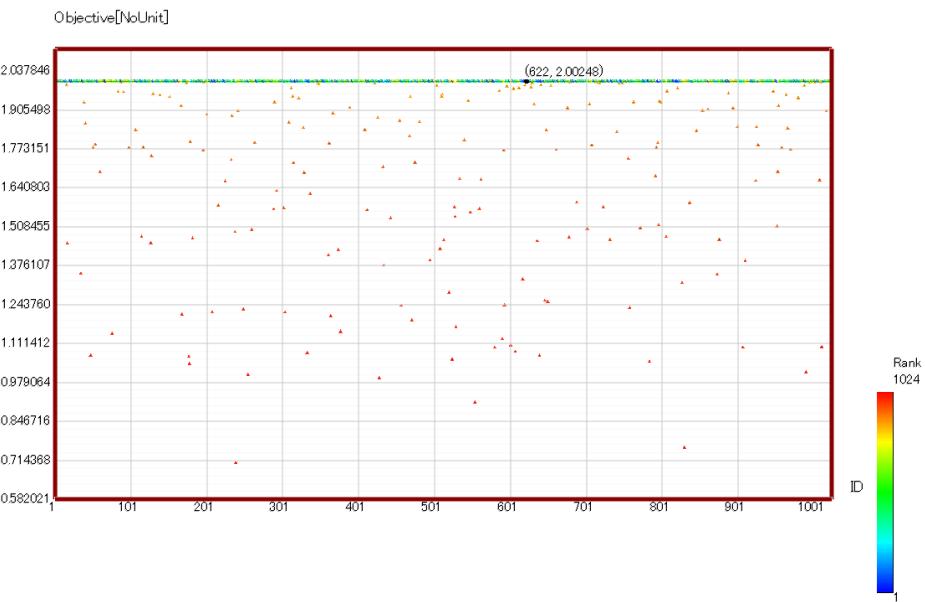


[最適解分布]

MOGAによる最終世代の最適解の分布を、各目的変数の組み合わせ分表示します。目的関数が1つの場合は、横軸に個体のID、縦軸に目的関数の値をとったグラフを表示します。目的関数が2つの場合は、2次元グラフ、3つの場合は3次元グラフで表示します。目的関数の数が4つ以上の場合は、各目的関数を3軸で組み合わせた分の3次元グラフを表示します。



test0-2(100th).[ID-Objective]



図中のカラーバーは、各IDのランクを表しています。ランク数が小さいほどパレート最適解に近いことを意味しており、最小ランク数の1がパレート最適解に相当します。

プロット点をダブルクリックすると、[目的関数・設計変数一覧]ダイアログが表示され、ダブルクリックした点の値を持つ個体IDが選択され、リストの一番上に表示が移動します。



[目的関数・設計変数一覧]の詳細な使い方は、[ポップアップメニュー(ドローウィンドウ)] - [目的関数・設計変数一覧]を参照してください。

[相関関係(目的関数)]

ここでは、各目的関数を、目的関数の最小値と最大値の幅で規格化した値を、個体数分折れ線グラフ化したものが追加されます。折れ線グラフを左クリックすると、各データの線が選択されます。ポップアップメニューから[目的関数・設計変数一覧]ダイアログを表示している場合、折れ線グラフを選択する度に、その値を持つ個体IDが選択され、リストの一番上に表示が移動します。Ctrl+マウス左ボタンを押した状態でドローウィンドウをドラッグすると、カーソルにヒットした折れ線グラフが選択され、その値がグラフ上に表示されます。

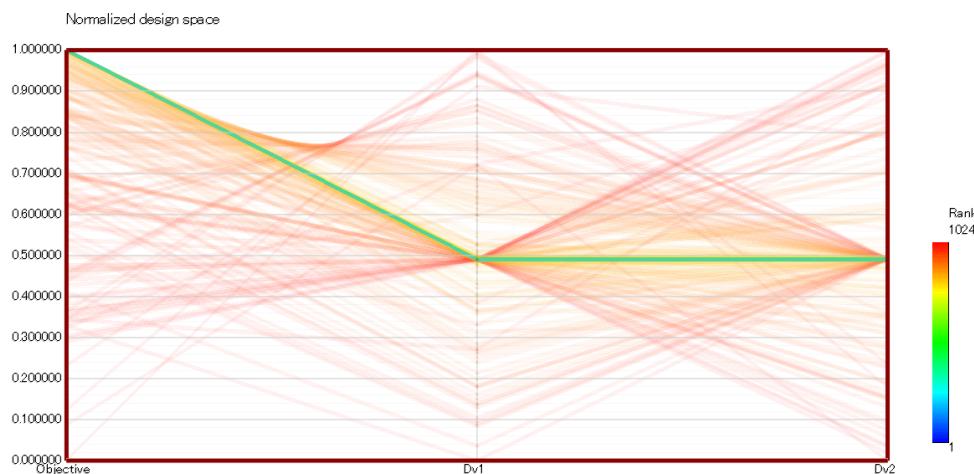
注. 目的関数が1つの場合は、このグラフは追加されません。

[相関関係(全体)]

相関関係(目的関数)は目的関数の折れ線グラフだけを表示しますが、ここでは設計変数の値も規格化し、目的関数の折れ線グラフとあわせて表示します。操作方法は、相関関係(目的関数)と同じです。



test0-2(100th)_[Total Trade-Off]



図中のカラーバーは、最適解分布グラフと同様、各IDのランクを表しています。

本章では、操作手順を中心とした説明のため、簡単な事例を用いましたが、実際に最適化を行う場合は、目的関数が2つ、あるいは3つといった複数の場合が多く、最終的には最適解分布グラフで形成されたパレートフロントや相関関係グラフから、目的に適った最適な解をトレードオフ選択する必要があります。この辺りの説明に関しては、[第1章 1.6 最適化結果の見方と役立て方](#)で紹介していますので、こちらをご参照ください。

第3章 例題

3.1 例題の紹介

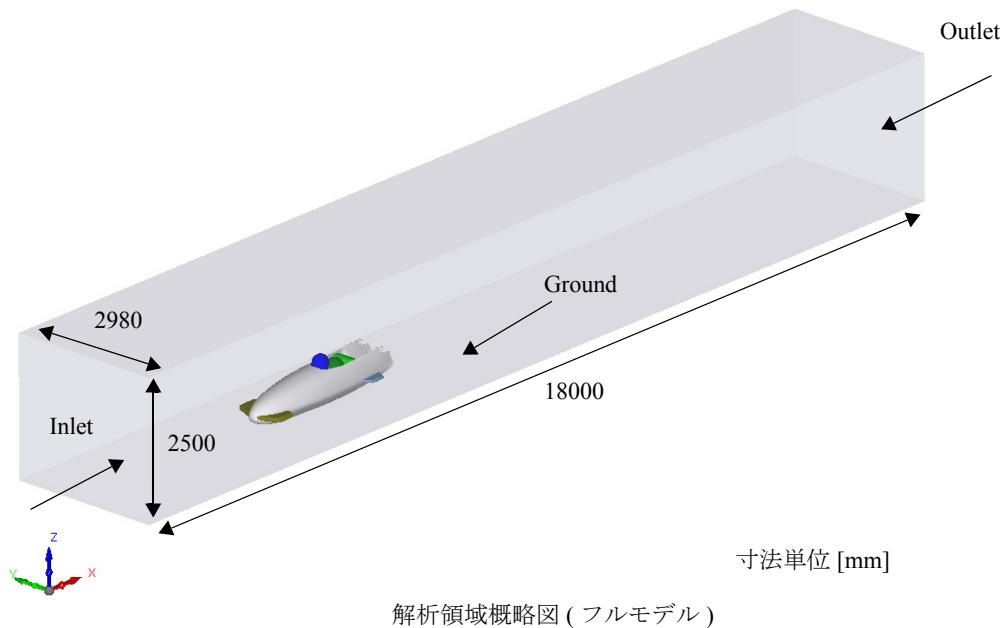
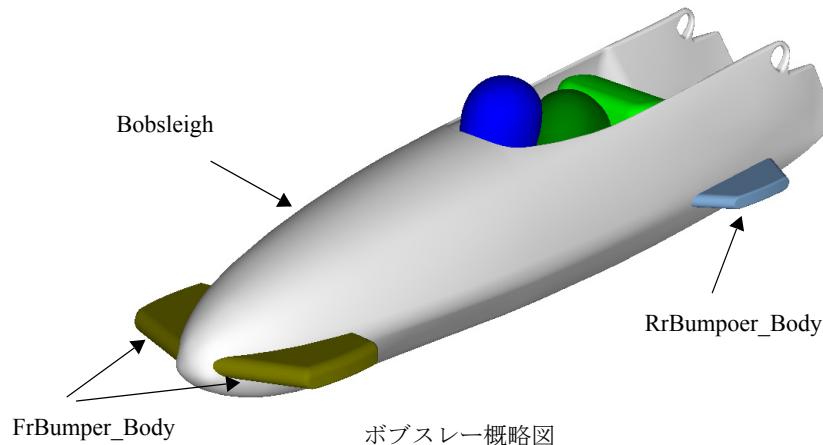
(1) 最適化の対象

本例題では、下図のようなボブスレーが時速140[km/h]で移動するとき、抗力、揚力が極力小さくなるようなボブスレーの前後バンパーの位置を、Extension Option (Optimization) (以下、EOoptiと記します。) を用いて求めます。

設計変数には、前部バンパーの高さ、後部バンパーの高さ、および、後部バンパー前後の移動量を、目的関数にはボブスレー車体にかかる抗力、揚力を用います。

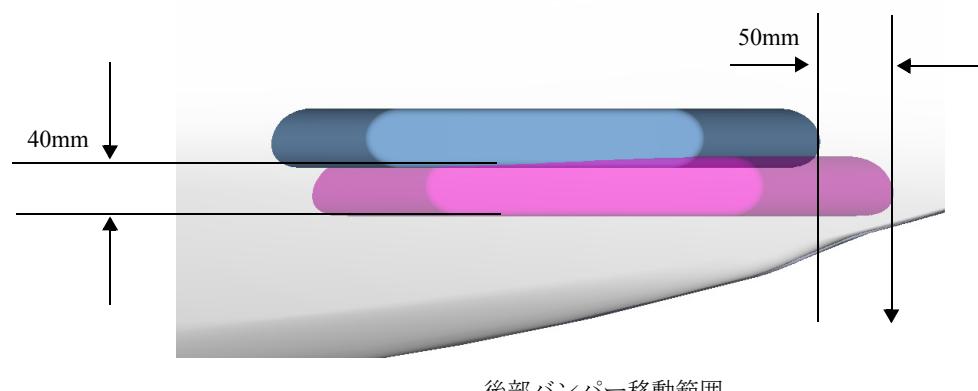
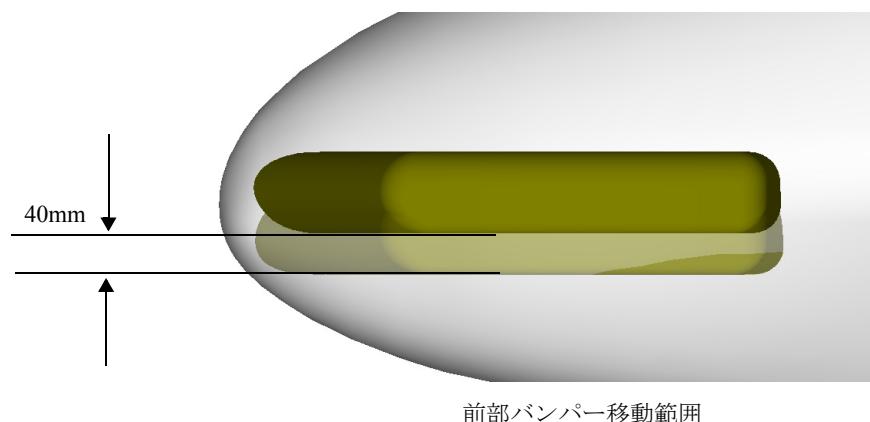
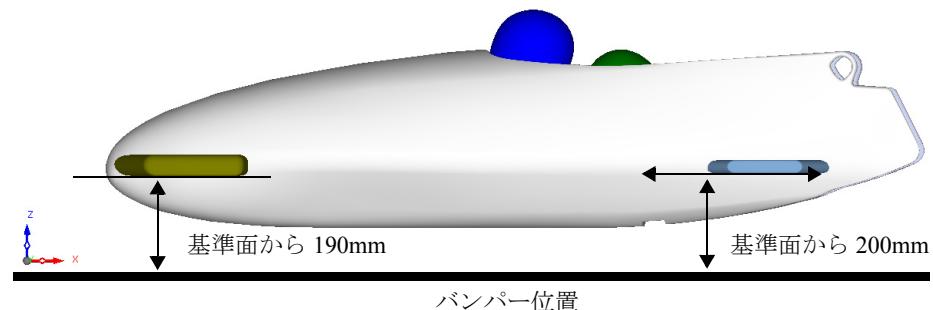
実験計画法による目的関数(抗力、揚力)のサンプル値取得には**SCRYU/Tetra**を用い、サンプリング数分の流体解析を行います。サンプル値収集後、Kriging法・MOGAによる最適化を実行し、最適化結果の評価を行います。また、最後に、これら一連の作業にVBインターフェースを用い、自動実行による最適化手法について紹介しています。

注. この例題で使用しているデータは、ユーザーフォルダのOptimization¥Chapter3¥SCTにあります。



(2) 想定する状況

バンパーの位置は、レギュレーションで定められた、以下の範囲以内で移動させます。



3.2 流体解析条件

EOoptiの例題ですから、最適化の設定から行いたいところですが、最適化を行う前に、応答曲面生成時に用いられる設計変数、目的関数のサンプル値が必要です。設計変数のサンプル値は、サンプリング手法により自動的に得ることができますが、この設計変数に対する目的関数の値(本例題では抗力、揚力の値)は別途用意する必要があります。このサンプル値には、冒頭でも述べましたように、**SCRYU/Tetra**による解析結果で得られた値を用いるため、最適化を行う前に流体解析を行う必要があります。今回は、ボブスレーの空力解析、ということで以下の解析条件で解析し、抗力、揚力の値を取得します。なお、今回は、計算時間短縮のため、解析領域にはハーフモデルを用い、バンパー位置の変更には、別途メッシュ生成したバンパー領域を解析領域にマージし、重合格子を用いて流れ場を解いています。

解くべき方程式

- 運動量保存式
- 質量保存式(圧力補正式)
- k-ε方程式

解析選択

- 流れ(乱流) : 乱流解析を行います。
- 重合格子 : 従属領域[FrBump_Slave], [RrBump_Slave]と独立領域[Detail_Region]のメッシュを重ねて計算を行います。

物性値

- 流体領域(MAT=1, 2, 3, 5) : 空気(20°C)
物性値ライブラリより[流体(非圧縮性)] - [空気(非圧縮20°C)]を使用します。
- バンパー形状領域(MAT=4, 6) : 銅(Cu)
物性値ライブラリより[純金属] - [銅(Cu)]を使用します。

境界条件

- 流入口 [Inlet] : 流速規定 38.9[m/s] ≈ 140[km/h]
- 流出口 [Outlet] : 表面圧力規定 0[Pa]
- 壁面
 [Bobsleigh], [FrBumper_Body], [RrBumper_Body] : 静止壁
 [Ground] : 平行移動壁 38.9[m/s] ≈ 140[km/h]

初期条件

- デフォルト(設定不要)

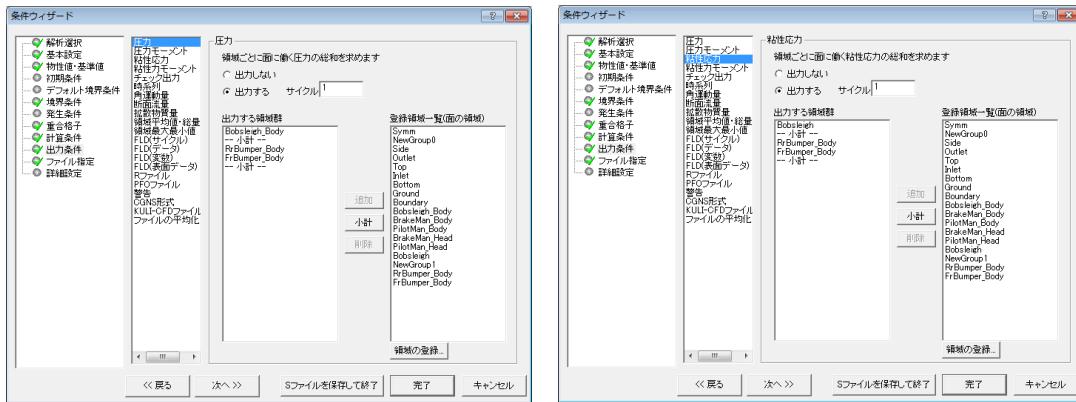
その他

- 流れモデル
3次元非圧縮性乱流
- 解析の種類
定常解析
- 計算サイクルおよび定常判定値
計算サイクル : 500[サイクル]
定常判定値 : $U, V, W, P, k, \epsilon 10^{-6}$
- 乱流モデル
Realizable k- ϵ モデル
- 図化ファイル
出力タイミング : 最終サイクルの結果を出力

圧力と粘性応力の出力設定

抗力、揚力取得のために、ボブスレーの壁面設定領域に働く圧力による力の総和と、粘性応力による力の総和をLFAファイルに出力します。

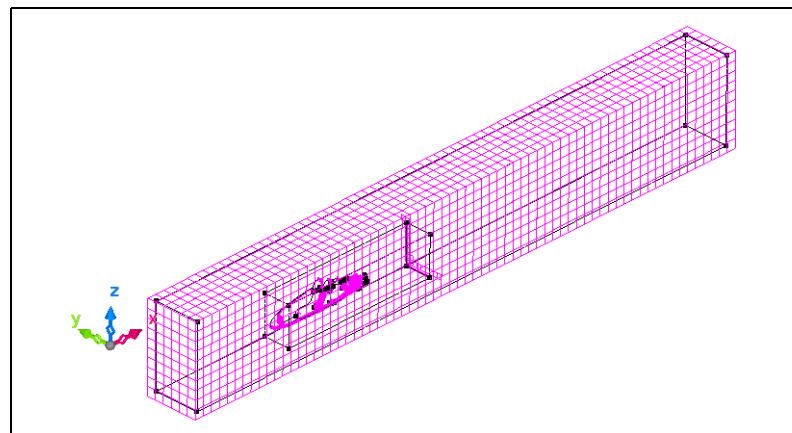
- [条件ウィザード] - [出力条件] - [圧力]で、[出力する]を選択します。上記の静止壁を設定した領域、[Bobsleigh], [FrBumper_Body], [RrBumper_Body]を選択して、追加をクリックします。
- [条件ウィザード] - [出力条件] - [粘性応力]で、[出力する]を選択します。圧力の設定と同じように、各領域を追加します。



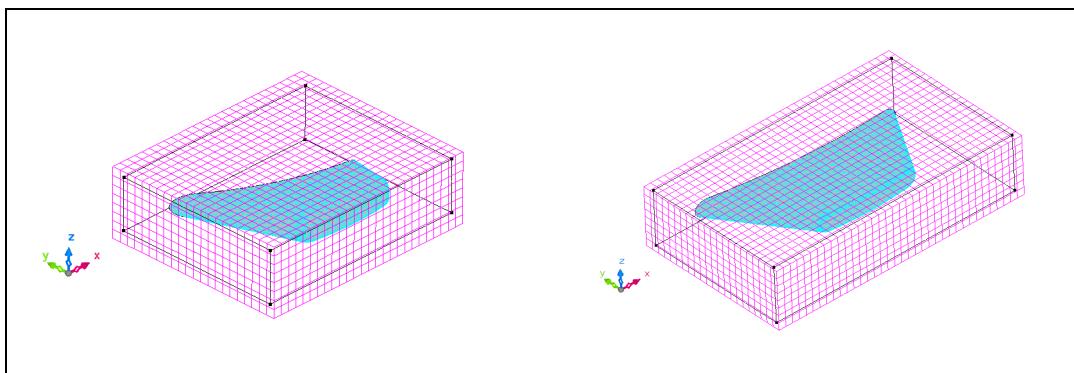
解析メッシュ

ハーフモデル、重合格子用いた解析を行うため、八分木、メッシュは以下のようになります。

- 八分木図



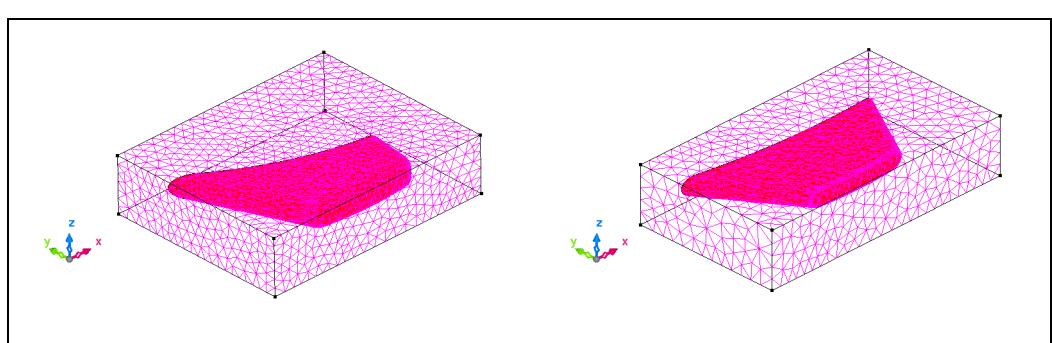
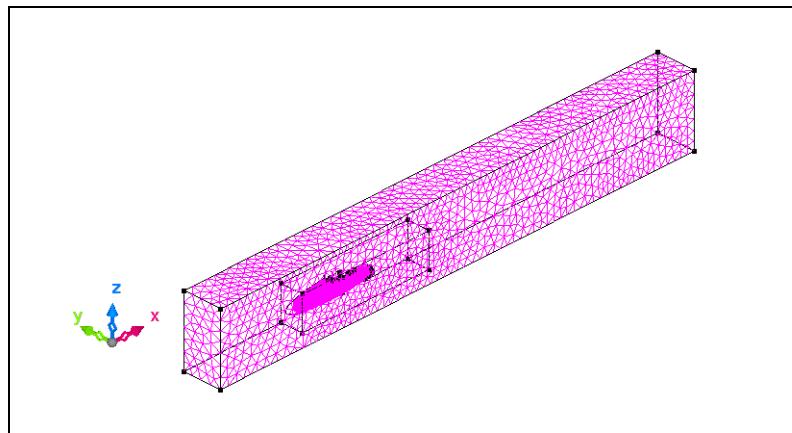
独立領域(Base.oct)



従属領域(FrBumper.oct)

従属領域(RrBumper.oct)

- メッシュ図



3.3 Extension Option(Optimization)による最適化

(1) 最適化計算条件

Extension Option (Optimization)による最適化の計算条件は以下の通りです。前部バンパー高さの移動量を[Fr Zpos]、後部バンパー前後移動量を[Fr Xpos]、高さ方向移動量を[FrZpos]とします。

設計変数

設計変数名	単位	最小値	最大値
Fr Zpos	mm	-40	0
Rr Xpos	mm	0	50
Rr Zpos	mm	-40	0

目的関数

目的関数名	単位	探索指針
抗力	N	最小値を探索
揚力	N	最小値を探索

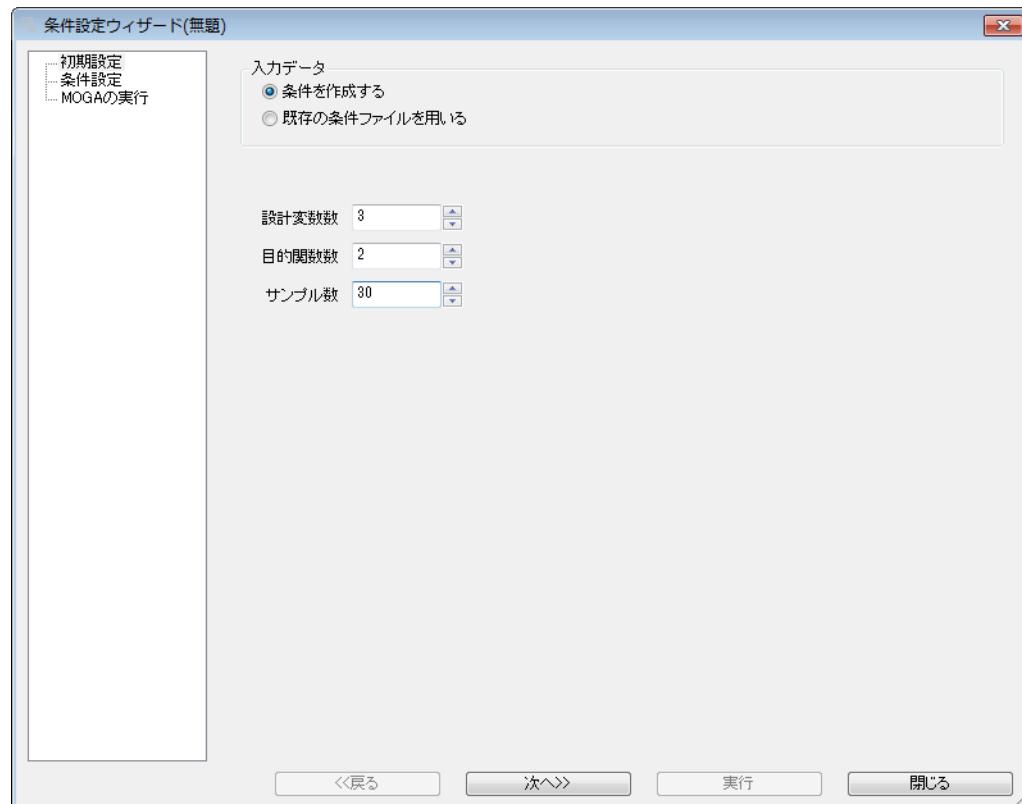
サンプリング数 : 30個
計算世代数 : 1000世代
個体数 : 1024個

(2) Extension Option(Optimization)の操作手順

ここでは、条件設定ウィザードの設定内容について説明します。

条件設定ウィザードの詳細な使い方については、[第2章 2.3 操作手順の具体例](#)を参照してください。

[初期設定]ページで、[設計変数数]に[3]、[目的関数数]に[2]、[サンプル数]に[30]を入力して次へ>>を押します。



[条件設定]ページで、[タイトル]に[Bobsleigh]、[設計変数]、[目的関数]リストに前ページに記述した条件を下図のように入力します。

Bobsleigh						
設計変数						
設計変数名	単位	最小値	最大値	型	離散化幅	表示形式
Fr Zpos	mm	-40	0	整数	1	数値
Rr Xpos	mm	0	50	整数	1	数値
Rr Zpos	mm	-40	0	整数	1	数値

目的関数						
目的関数名	単位	設計指針	探索値	表示形式		
Drag	N	最小値を探索	-	数値		
Lift	N	最小値を探索	-	数値		

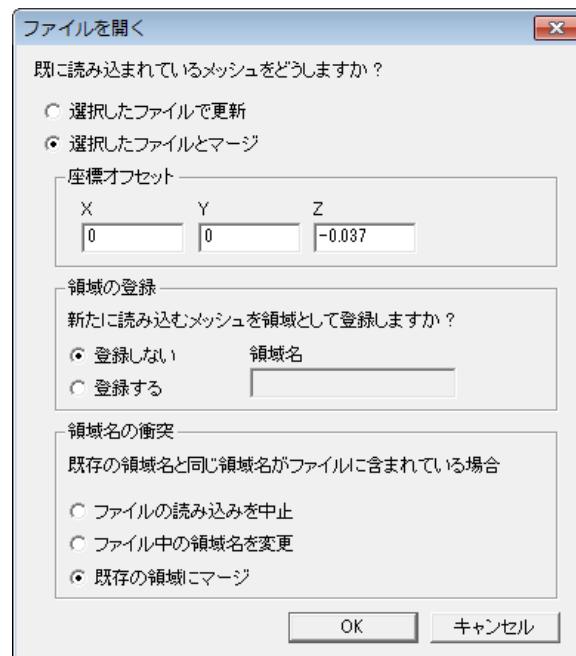
[実験計画法]リストの設計変数の値には、手作業での直接入力と、サンプリング手法による自動入力、どちらでも設定することができますが、第1章でも述べられているように、値が適度に分散したサンプル値を用いた方が応答曲面を広域にわたり表現する上で望ましいため、今回は値が重複しないサンプル値を自動生成するラテン超方格法を用います。作成をクリックすると、ラテン超方格法によって、設計変数のサンプル値が自動的に入力されます。

実験計画法						
サンプリング手法 ラテン超方格法 作成						
サンプルID	Fr Zp...	Rr X...	Rr Zp...	Drag	Lift	
1	-37	16	-36			
2	-23	12	-33			
3	-39	41	-7			
4	-21	5	-8			
5	-8	38	-6			
6	-11	28	-4			
7	-10	43	-23			
8	-28	10	-32			
9	-34	19	-11			

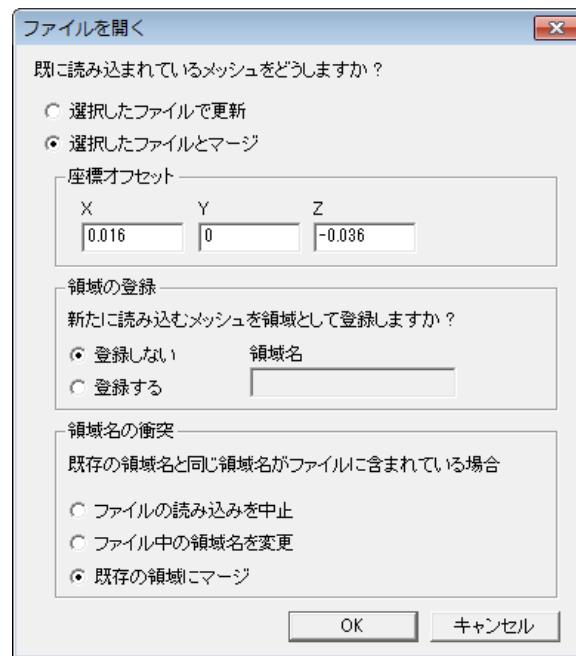
再描画 作成したデータの保存

目的関数値には、得られた設計変数(バンパー位置)のメッシュをSCRYU/Tetraで用意し、解析を行います。(以下で使用するメッシュ生成用データ(mdl, oct, jファイル)、解析条件(sファイル)は Optimization\Chapter3\SCTフォルダ内にあります。)

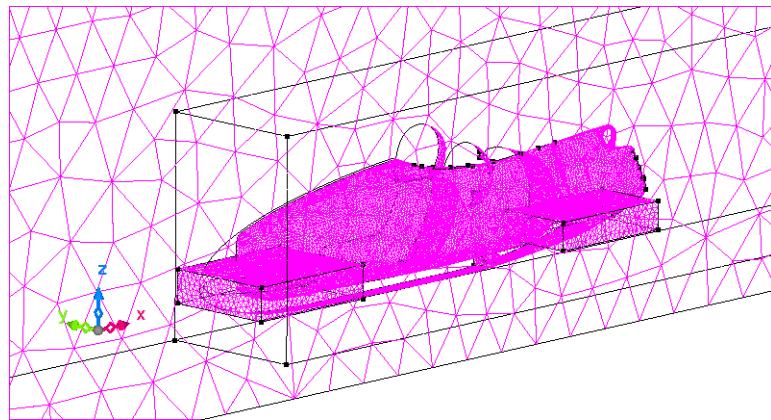
バンパー位置の変更には、重合格子を用いているので、前部バンパーのメッシュFrBumper.preと、後部バンパーのメッシュをRrBumper.preをBase.preにマージさせます。以下は、SCTpreにpreファイルを追加して読み込むときに表示されるダイアログですが、ここでラテン超方格法によって得られたバンパーの移動量を入力します。例えば、サンプルID=1のFr Zposは-37mm=-0.037mなので、Base.preが読み込まれているSCTpreにFrBumper.preを以下のように読み込ませます。



後部バンパーの移動量は、RrXpos = 16 [mm] = 0.016 [m]、
RrZpos = -36 [mm] = -0.036 [m]なので、Base.preとFrBumper.preをマージしたSCTpreに、RrBumper.pre
を以下の移動量を与えてマージさせます。



マージ後のメッシュは下図のようになります。



これらマージしたメッシュを解析用preファイルとして保存し、このメッシュを用いた流体解析を行います。抗力、揚力の値は、Lファイル中に出力される圧力、粘性応力による力の総和から求めます。(抗力には、X方向の力の総和を、揚力には、Z方向の力の総和を用います。)以下に、Lファイルから抗力、揚力の値を取得する方法を示します。

まず、**LFileView**を用いて、Lファイルを開きます。下図のようにX方向、Z方向の圧力、粘性応力による力の値を変数として登録します。

LFfileViewの詳細な使い方に関しては、[ユーザーズガイドツール編 第3部 第1章](#)をご参照ください。

Bobsleigh.i - LFileView

ファイル フォルダ 表示 属性 グラフ ヘルプ

戻る 前へ 次へ 後へ リセット 500 指定サイクルへ移動 Log

(YPLUS DISTRIBUTION)

0.0 - 5.0 :	212
5.0 - 11.6 :	2939
11.6 - 30.0 :	5712
30.0 - 100.0 :	15429
100.0 - 300.0 :	5052
300.0 - 1000.0 :	7285
1000.0 - Inf :	0

==== PRESSURE FORCE WORKING ON SURFACES ===-

REGION	AREA	FORCE-X	FORCE-Y	FORCE-Z
Bobsleigh	5.33733	33.3746	-159.280	-41.9164
SUBTOTAL:	5.33733	33.3746	-159.280	-41.9164
FrBumper_Body	0.226632	12.8812	-1.54242	-1.87045
RrBumper_Body	0.140242	2.71507	-3.15203	-7.56036
SUBTOTAL:	0.366874	15.5982	-4.69445	-9.43081
TOTAL:	5.70421	48.9709	-163.974	-51.3472

==== STRESS FORCE WORKING ON SURFACES ===-

REGION	AREA	FORCE-X	FORCE-Y	FORCE-Z
Bobsleigh	5.33733	7.13376	-0.268338	0.109129
SUBTOTAL:	5.33733	7.13376	-0.268338	0.109129
FrBumper_Body	0.140242	0.312197	0.0221969	-0.00923553
RrBumper_Body	0.226632	0.608399	-0.174386	0.000725619
SUBTOTAL:	0.366874	0.920596	-0.152169	-0.00850991
TOTAL:	5.70421	8.05435	-0.420556	0.100619

*** NCYC = 500 WRITTEN TO Bobsleigh_500.fld

*** NCYC = 500 WRITTEN TO Bobsleigh.r

レディ

変数登録 実数/グループ 実数/Cycle CYCLE 実数

[4] [4] [x] 実数/グループ

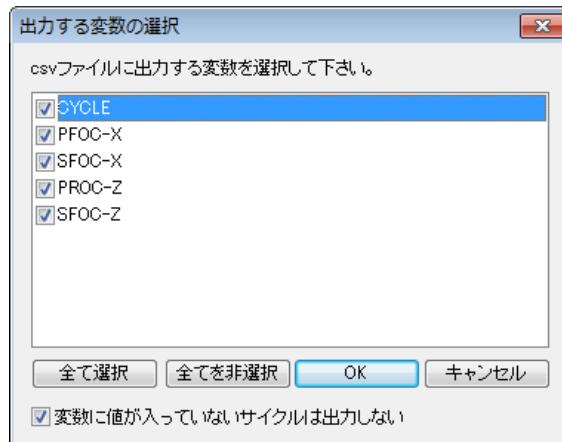
500サイクル 41804行 1文字

PFOC-X : 圧力によるX方向の力
 SFOC-X : 粘性応力によるX方向の力
 PFOC-Z : 圧力によるZ方向の力
 SFOC-Z : 粘性応力によるZ方向の力

これにより、
 抗力= $P_{FOC-X} + S_{FOC-X}$
 揚力= $P_{FOC-Z} + S_{FOC-Z}$
 となります。

今回は、これらの値を250～500サイクルの間で平均化した値を目的関数のサンプル値として用います。

LFileViewの[ファイル] - [変数をcsv形式で保存]で、下図のように登録した変数のチェックをONにしてcsvファイルを保存し、Excel等を用いて250～500サイクルの平均値を取得します。



#	xlview cs	v	PFOC-X	SFOC-X	PFOC-Z	SFOC-Z
Cycle						
490	488		50.5225	8.02936	-42.7223	0.131483
491	489		50.5688	8.03058	-43.458	0.129754
492	490		50.5788	8.03204	-44.2454	0.127765
493	491		50.5576	8.03374	-45.0593	0.125542
494	492		50.4988	8.03568	-45.9133	0.123104
495	493		50.4029	8.03779	-46.7632	0.12046
496	494		50.277	8.04013	-47.6032	0.117645
497	495		50.1197	8.04258	-48.4158	0.114708
498	496		49.9335	8.04503	-49.1724	0.111176
499	497		49.7195	8.04744	-49.8614	0.108844
500	498		49.4873	8.04982	-50.4651	0.105959
501	499		49.2364	8.05215	-50.9659	0.103208
502	500		48.9709	8.05435	-51.3472	0.100619
503			48.42207	8.041483	-44.7375	0.116019
504						

上図より、

$$\text{抗力} = \text{PFOC-X} + \text{SFOC-X} = 48.42207 + 8.041483 = 56.463553[\text{N}]$$

$$\text{揚力} = \text{PFOC-Z} + \text{SFOC-Z} = -44.7375 + 0.116019 = -44.621481[\text{N}]$$

となり、これが目的関数のサンプル値となります。

この一連の作業を、サンプル数分行います。

しかし、この作業を繰り返すのは非常に手間がかかります。本例題ではこのようなルーチン化できる作業にVBインターフェースを用い、自動処理を行った事例を紹介していますので、あわせてご参考ください。

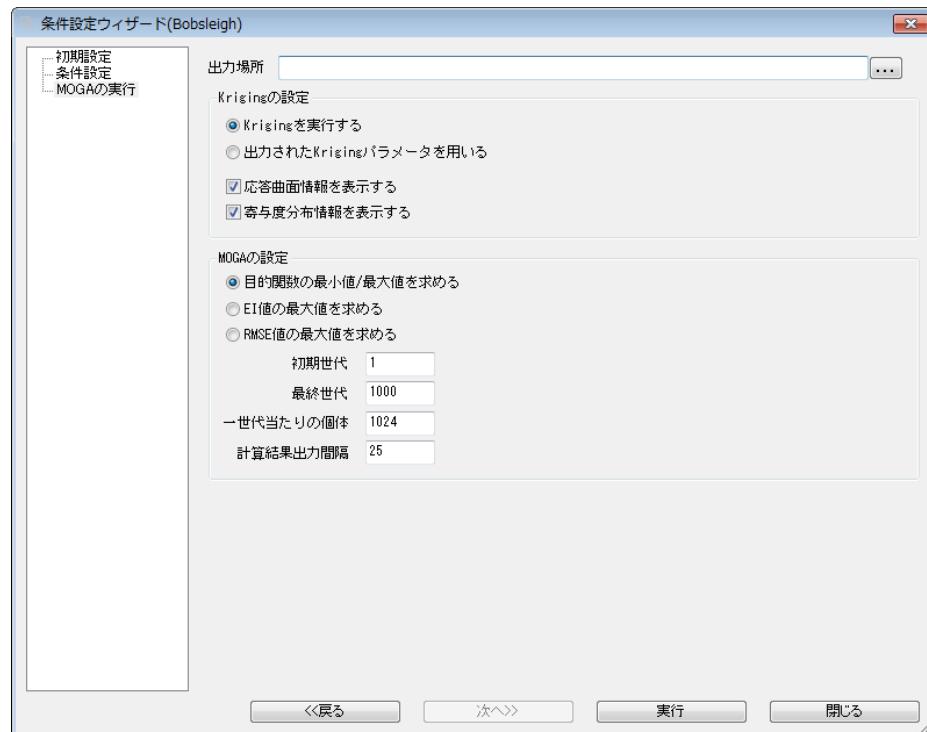
目的関数のサンプル値を全て入力することで、最適化を行うための全ての条件が揃います。入力作業中、途中で入力作業を中断したい場合は、**作成したデータの保存**から入力されている条件を条件ファイル(_in.csv)として保存できます。条件ファイルはcsv形式なので、他のソフトで抗力、揚力を入力した後、EOoptiに読み込むことができます。

今回の解析で得られた目的関数の値を入力すると以下のようになります。

実験計画法					
サンプリング手法 ラテン超方格法					
作成					
サンプルID	Fr Zp...	Rr X...	Rr Zp...	Drag	Lift
1	-37	16	-36	56.6894	-44.7999
2	-23	12	-33	56.0026	-45.5398
3	-39	41	-7	56.2102	-45.1942
4	-21	5	-8	56.0071	-42.5234
5	-8	38	-6	54.2364	-43.0503
6	-11	28	-4	56.3879	-43.0243
7	-10	43	-23	55.6756	-43.256
8	-28	10	-32	55.2802	-44.3906
9	-34	19	-11	56.8108	-43.2035

再描画 作成したデータの保存

これで全ての条件の入力が終了したので、**次へ>>**をクリックします。



今回は1000世代まで計算を行います。出力場所で、計算結果が出力されるパスを指定し、**実行**をクリックすると、Kriging法・MOGAによる最適化が開始します。

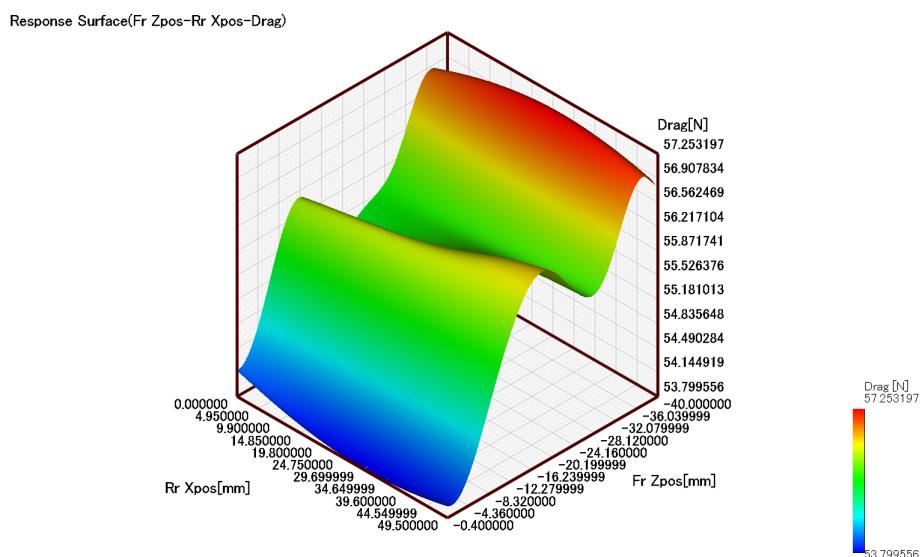
3.4 結果の評価

(1) 応答曲面

実行のクリック後、最初に行われるのが応答力面の生成と、寄与度分布の算出です。

応答曲面のデータから確認できる特徴的な内容としては、「設計変数ごとの変化に対する目的関数の挙動の"傾向"を、一目で確認できる」という点が挙げられます。

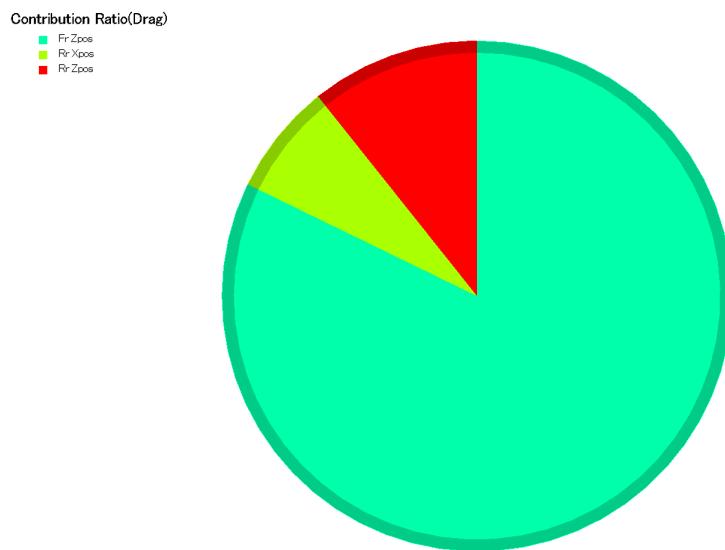
例えば、前部バンパー高さ-後部バンパー前後-抗力のグラフ(Response Surface(Fr Zpos-Rr Xpos-Drag))の応答曲面に着目すると、前部バンパーを上方に移動させることで、抗力は低下していくというのを視覚的に確認できます。



応答曲面は全ての設計変数/目的関数の組み合わせについて作成されますので目的関数・設計変数が多く存在する場合には、着目する目的関数・設計変数を2、3に絞り、その中で変化の傾向を確認するのが一般的です。

(2) 寄与度分布

Kriging実行後、各目的関数の寄与度を分散分析によって求めます。このデータからは目的関数ごとに、その変化を大きく左右する可能性、つまり影響度の高い設計変数を確認することができます。なお本例題の場合、Contribution(Drag)の寄与度分布から、抗力について特に影響度の高い設計変数は前部バンパー高さ(FrZpos)であることが分かります。この結果から、実際の設計を行う際には、これらの値の決定には特に注意を払わなければならないことが分かります。



ここまで見てきた「応答曲面」「寄与度分布」は、MOGAによる最適解探索のための参照用データです。ここからはMOGAによる最適解の探索結果を整理した「相関関係(目的関数/全体)」と「最適解分布」について順に見ていきます。

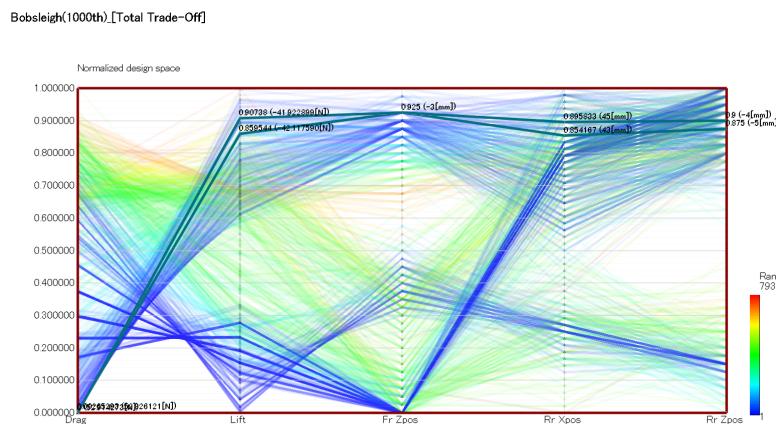
(3) 相関関係(目的関数/全体)

相関関係は、最終世代におけるサンプル毎の目的関数・設計変数の組み合わせをプロットしたグラフです。応答曲面のグラフでは複数の目的関数・設計変数についての変化の相互関係を一度に確認することができますが、この相関関係のグラフでは任意の目的関数・設計変数の変化に対する他のパラメータの挙動を、一枚のグラフで確認することができます。

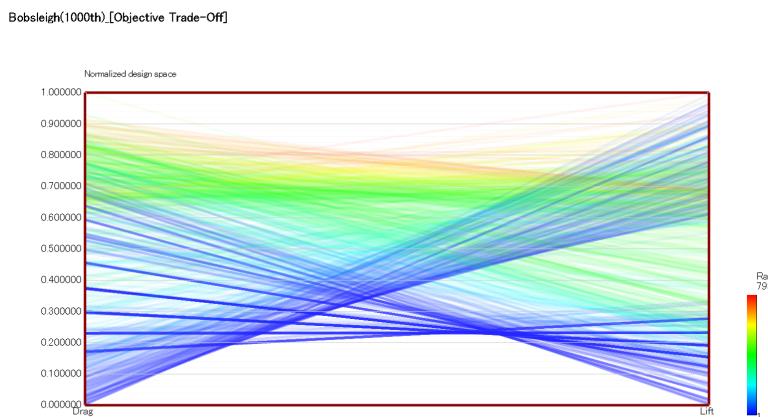
なお相関関係のグラフには、全目的関数・設計変数についてプロットした「相関関係(全体)」と、表示を目的関数のみ限定した「相関関係(目的関数)」の2種類があります。

まず「相関関係(全体)」は、最終世代まで残った各個体に設定されている設計変数値とその目的関数値、すなわち応答曲面上での値をプロットしたグラフです。折れ線グラフをクリックすると、目的関数・設計変数の組み合わせを確認することができます。例えば任意のパラメータの縦軸上にて、短い間隔で上方向へ数回クリックを行うと、そのパラメータの上昇が他のパラメータにどのような影響(変化)を与えるかを確認することができます。

また本例題の場合、例えば抗力の低下のみを重視する場合は、抗力最小側の数点をクリックすることで、抗力を低下させる際に求められる他のパラメータの数値の傾向を確認することができます。



また、もう一つの「相関関係(目的関数)」グラフは、「相関関係(全体)」のうち、目的関数の箇所の抜粋です。グラフの幅がより広く確保されていますので、目的関数の選択をよりスムーズに行うことができます。

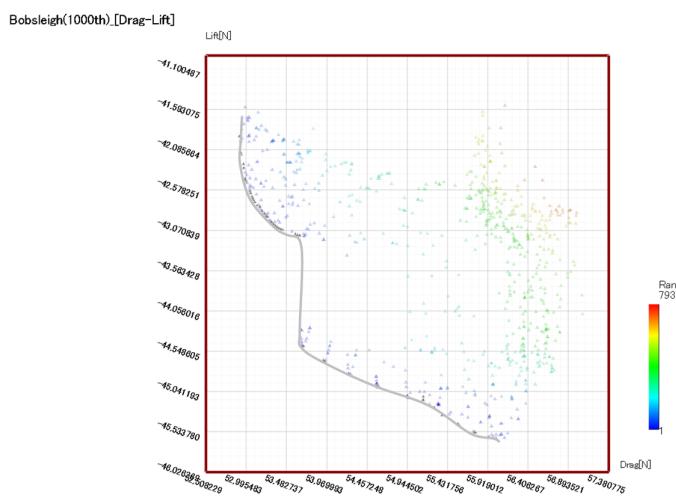


(4) 最適解分布

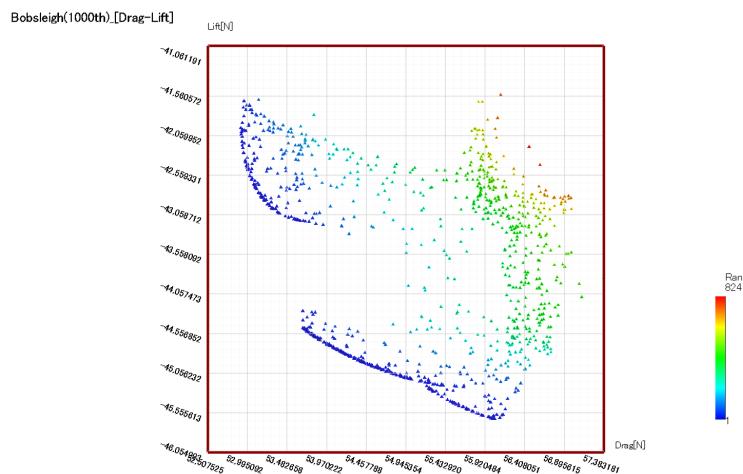
最適解分布は、最終世代まで残った個体のプロット、つまりこれらの全てがEOoptiの提示する最適解の候補といえます。その中でも特に、縦横軸のうち、一方の目的関数値を固定した場合に、他方の目的関数の目標を最大限満足する曲線状の解の集合をパレートフロントといい、本例題の場合は下図のように、選択点をつなげた曲線がそれに相当します。

補足. 選択点は、[目的関数・設計変数一覧]ダイアログで、ランクが1の個体を選択したもので、これら個体群がパレート最適解に相当します。

しかしEOoptiによる最適解の示唆はここまでです。この中から最終的に製品に採用する最適解を選定するのはユーザー様ご自身です。採用する設計数、短期的・長期的な運用コスト、その製品に最低限求められる性能等、最適化の背景・ご自身の環境等を考慮しながら、"ご自身に最適な解"を見つけてください。



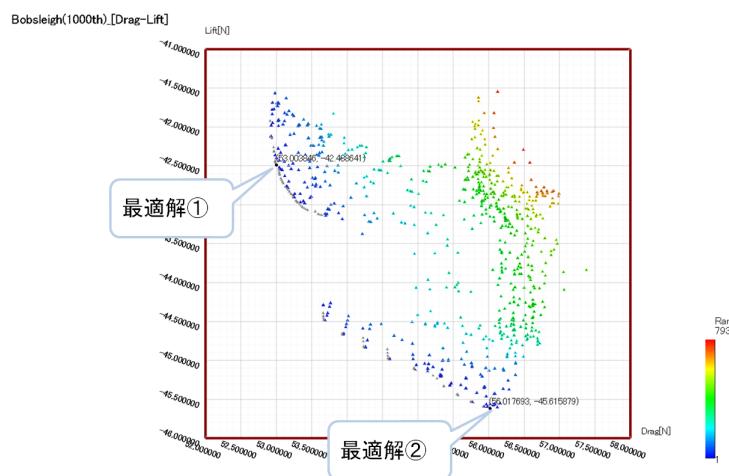
注. パレートフロント上の分布がまばらになってますが、これは、すべての設計変数を整数型に指定しており、算出される目的関数値が制限されているためです。設計変数の型・小数点以下桁数に制約がない場合は、実数値型を用い、離散化幅を小さくすることで、下図のようにパレートフロントはより明確に形成されます。



(5) 選定した解の検証

パレートフロント上からの最適解の選択後、忘れてはならないのが、その目的関数値の妥当性の確認です。パレートフロント上を含む全ての個体の持つ目的関数の値は、あくまで応答曲面上での値であり、代理モデルの予測した"推測値"です。そもそも代理モデルの精度が低ければ、いくら世代数を増やしても信頼できる最適解は得られません。応答曲面の精度の確認は一般的に、任意の設計変数の組み合わせでの応答曲面上での目的関数値と、サンプリング表を作成した際と同様の方法で求めた目的関数値とがきちんと一致するか、という比較により行います。なお、どんなに高い精度の応答曲面にも必ず誤差は発生しますが、許容誤差の目安としては10%程度を基準にお考えください。もしそれ以上となる場合はサンプル数を増やし、実験計画法の実施から最適化のプロセスをやり直す必要があります。

なお、ここでは一例として、下記の2個体を最適解として採用し、確認計算を行います。



その結果が次表になりますが、この表によると抗力、揚力に関してはそれぞれ 約5%以内の精度で結果が得られていることが分かります。このことから、これらの最適解は妥当であり、信頼できる解と評価することができます。

表3.1 最適化結果と確認計算結果との比較
ラテン超方格法 30点

	最適解①		最適解②	
	抗力[N]	揚力[N]	抗力[N]	揚力[N]
最適化結果(Kriging+MOGA)	53.00	-42.49	56.02	-45.62
確認計算結果(SCRYU/Tetra)	53.00	-40.61	54.75	-44.87
誤差 (最適化結果 - 確認計算結果)	0.00 (0.00%)	1.88 (4.63%)	-1.27 (2.32%)	0.75 (1.67%)

設計変数の値

	F Fr Zpos [mm]	Rr Xpos [mm]	Rr Zpos [mm]
最適解①	-4	41	-6
最適解②	-22	14	-34

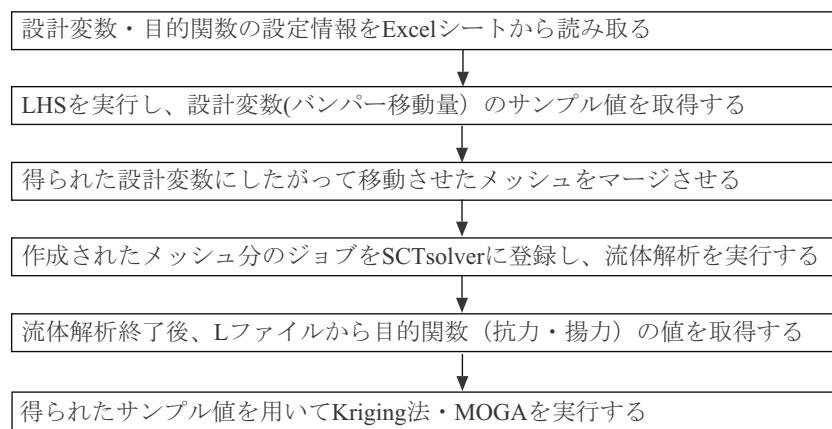
本例題では、一連の作業を確認しやすくするため、約160万要素と、少ないメッシュで流体解析を行いましたが、実際の製品設計で評価される際には、より多くの要素数のメッシュを用いて解析を行い、目的関数(抗力、揚力)の精度を向上させた方が最適化結果の信頼性も向上することは言うまでもありません。また、最適化の精度も、サンプル数に大きく影響します。こちらも結果取得までの時間を短縮するため、サンプル数を30としましたが、対象とする目的関数の分布によっては、より多くのサンプル数が必要となる場合があります。本例題においても、サンプル数を変えることによって、最適化の結果も異なる可能性がある点についてはご理解ください。

3.5 VBインターフェースによる最適化プロセスの自動化

ここでは、**SCRYU/Tetra**とEOoptiのVBインターフェースを組み合わせた、自動処理の例を紹介します。先ほどの、複数のメッシュをマージさせる設定方法では、サンプリング数分だけバンパー位置をずらさなければならず、非常に手間のかかる作業となっていました。また、流体解析によって得られた抗力、揚力もLファイルで出力される値を手作業で条件設定ページに入力する必要があり、煩雑な作業が発生します。

ここでは、こういった作業にVBインターフェースを用い、自動的にEOoptiを用いた最適化を行う手法について紹介しています。

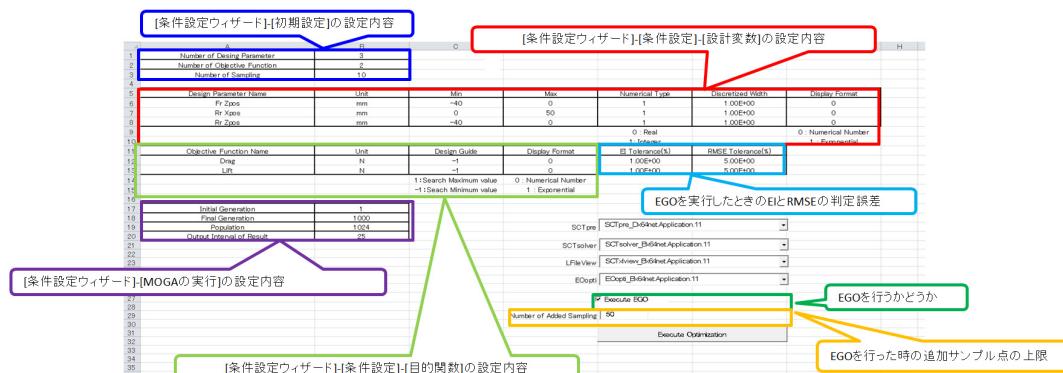
今回の自動処理の流れは以下のようになっています。



注. この自動処理を実行するには、10GB程のディスク容量を必要とします。実行する際は、Optimization\Chapter3\SCTフォルダを、十分なディスク容量がある場所にコピーしてから行ってください。

(1) 起動方法

Optimization\Chapter3\SCTフォルダを十分なディスク容量がある場所にコピーした後、コピーしたフォルダ内にあるExecuteOptimization_SCT.xlsを開きます。下図はこのファイルを開いたときの表示状態です。



シート内に入力されている値は最適化の条件で、EOoptiの[条件設定ウィザード]の[初期条件]、[条件設定]の[設計変数]、[目的関数]リストの入力値に相当します。

Execute Optimizationをクリックすると、上述の処理手順に従い最適化が実行されます。実行中に作成されるデータは、ExecuteOptimization_SCT.xlsファイルがあるフォルダ内に保存されていきます。

(2) ソースコードの説明

ソースコードについては、主なプログラムの流れのみについて、以下に表示します。コメントの直後で関数が呼ばれていますが、この関数が上述のプログラムの流れの各処理内容に相当します。各関数内で行われている詳細な処理内容については、ExecuteOptimization_SCT.xls内のソースコードを参照してください。

```

Sub ExecuteOptimization(typeModule As Module, typeEGO As EGO)
    Application.DisplayAlerts = False
    'Get path for analysis and optimization.
    Dim typeProjectPath As ProjectPath
    Call GetProjectPath(typeProjectPath)

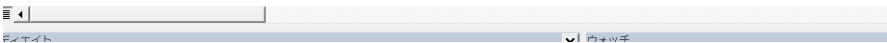
    'Get setting parameters of desing parameter and objective function from Excel sheet
    Dim typeBaseData As BaseData
    Dim typeDpCondition As DesignParameterCondition
    Dim typeObjCondition As ObjectiveFunctionCondition
    Call GetValuesFromSheet(typeBaseData, typeDpCondition, typeObjCondition)

    'Execute Latin Hypercube Sampling, and get sample values of desing parameter
    Dim typeDpSampleArray() As SampleValue
    If (GetDesignParameterValues( _
        typeModule, _
        typeProjectPath, _
        typeBaseData, _
        typeDpCondition, _
        typeObjCondition, _
        typeDpSampleArray() ) _ 
    ) = False Then
        Debug.Print ("GetDesignParameterValues failed.")
        Exit Sub
    End If

    'From sample values of design parameter, create mesh file for analysis to get sample value of drag and lift force.
    Call CreateMeshForAnalysis( _
        typeModule, _
        typeProjectPath, _
        typeBaseData, _
        typeDpSampleArray() )
    'Execute SCT solver.
    If (ExecuteSCTsolver( _
        typeModule, _
        typeBaseData, _
        typeProjectPath) = False) Then
        Debug.Print ("ExecuteSCTsolver failed.")
        Exit Sub
    End If

    'Get drag and force from L file.
    Dim typeObjSampleArray() As SampleValue
    Call GetObjectiveFunctionValues( _
        typeModule, _
        typeProjectPath, _
        typeBaseData, _
        typeObjSampleArray() )
    'Execute Kriging and MOGA.
    If (ExecuteMOGA( _
        typeEGO, _
        typeModule, _
        typeProjectPath, _
        typeBaseData, _
        typeObjCondition, _
        typeObjSampleArray() ) _ 
    ) = False Then
        Debug.Print ("ExecuteMOGA failed.")
    End If
End Sub

```



(3) ExecuteEGOについて

[Execute EGO]のチェックがOFFの場合は、上述例題のように、ラテン超方格法により設計変数の値を無作為に生成しますが、チェックをONにした場合は、初期サンプル値（ラテン超方格法）を用いた応答曲面生成後、MOGAによってEIとRMSEの最大値（目的関数の最大値と最小値の差で正規化された値）を求め、この結果からEI・RMSEを最大化する設計変数値が取得できます。そして、この値を用いて目的関数値である抗力・揚力をSCRYU/Tetraによる解析から求め、これを新たなサンプル値として追加します。このような一連の処理を繰り返すことで、応答曲面の精度を向上させていきます。EGO（Efficient Global Optimization）は一般的に、EIを指標としてサンプル点を追加していきますが、本サンプルVBA内の処理では、追加サンプル点が局所的な部分に集中しないよう、繰り返し処理の中でEIとRMSEを交互に評価しています。EI・RMSEの最大値が指定した閾値(Excel表内のToleranceの値で今回は1%としています。)以上の場合、取得した設計変数・目的関数値を新たなサンプル値として追加します。すべての目的関数のEIが閾値以下になったとき、サンプル値追加の繰り返し処理を終了し、目的関数(抗力・揚力)に対してMOGAを用い、最終的な最適解を求めます。

図3.1はラテン超方格法30点とEGO 30点の最適解分布を比較したものですが、両者のパレートフロントが大きく異なっているのが分かります。(パレートフロントの傾向を比較するため、設計変数の数値型は上記設定の整数ではなく、実数型を用いて比較しています。)

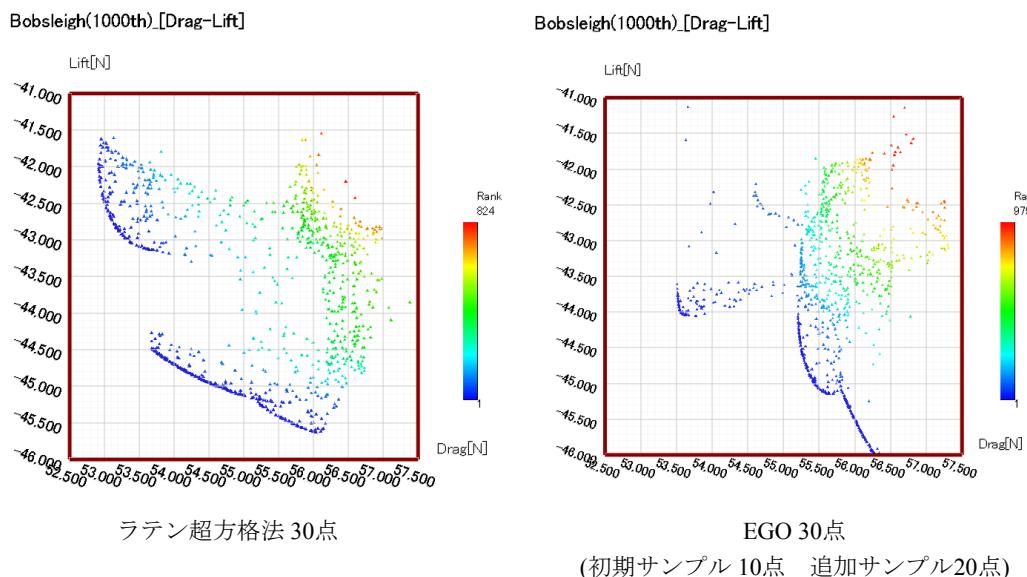


図 3.1 最適解分布比較図

これは、図3.2のように、EGO30点では、EI・RMSEを指標としてサンプル点を追加していく過程で、本来の応答曲面の分布特性である多峰性領域現れ、その結果、パレートフロントの分布も、ラテン超方格法30点とは大きく異なるものになった結果であると推測できます。ただし、この時点の追加サンプル数20点では、まだEIの最大値は指定した閾値以下の値にはなっておらず、さらにサンプル点を追加していくことで、パレートフロントの分布も、また異なったものになっていくことが予想されます。

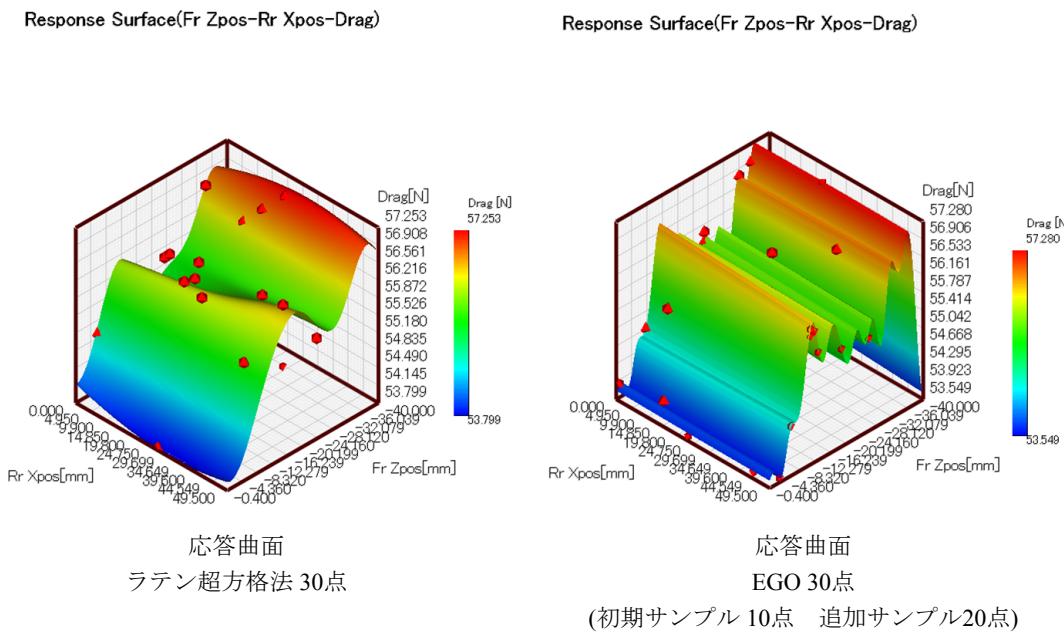


図 3.2 応答曲面比較図

このように、EGOを用いた場合は、目的関数の特性によっては、定量的な解を得るまでには多くのサンプル数が必要になるケースがあります。以上の点を踏まえ、逆に、定性的な最適解の傾向を見ることがのみを目的とした場合は、ラテン超方格法といったサンプリング手法のみを用い、応答曲面の特性によっては、サンプル数を減らし、応答曲面の分布を意図的にならすことで、最適解の傾向を把握するのも1つの手法といえます。

第4章 メニューリファレンス

[ファイル] - [新規作成]

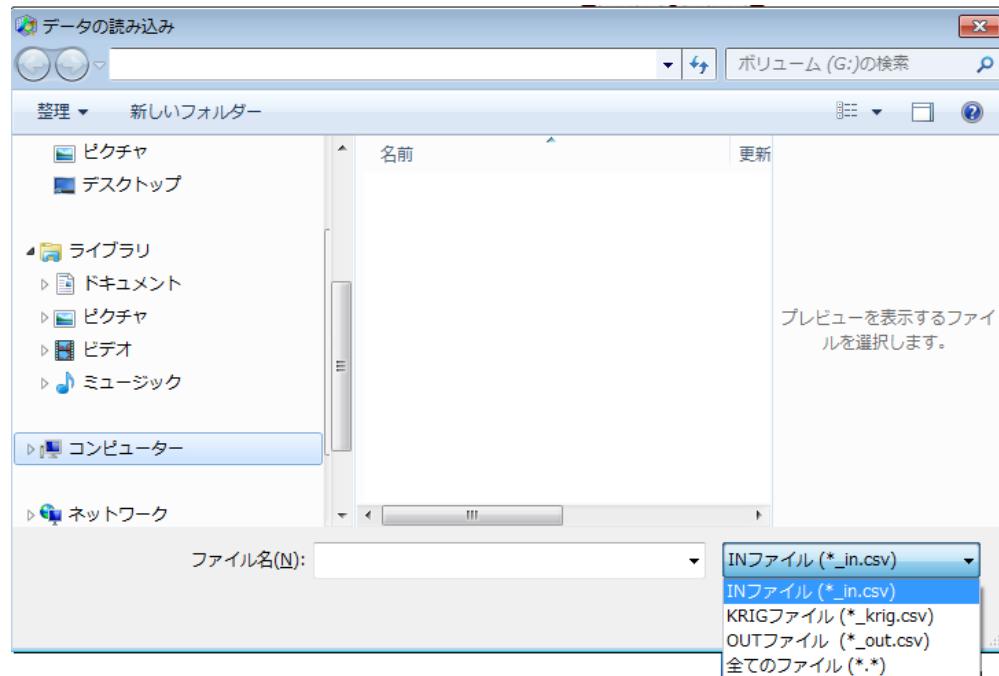
機能 新しくプロジェクトを作成します。

注. Kriging法・MOGAを実行しているときは、プロジェクトを作成することができません。

[ファイル] - [開く]

機能 現在のプロジェクトに、読み込んだファイル情報を追加します。

操作 ファイルの種類とファイル名を指定します。ファイルの種類には、条件ファイル(_in.csv)、Krigingパラメータファイル(_krig.csv)、計算結果ファイル(_out.csv)が選択できます。



[ファイル] - [閉じる]

機能 現在のプロジェクトを終了します。

操作 このメニューを選ぶだけで閉じます。

注. Kriging法・MOGAが実行中のドキュメントを閉じると、計算が強制的に終了されます。

[ファイル] - [Extension Option(Optimization)の終了]

機能 Extension Option(Optimization)を終了します。

操作 このメニューを選ぶだけで終了します。

注. Kriging法・MOGAが実行中のドキュメントがある場合、計算は強制的に終了されます。

[選択] - [すべて選択]

機能 表示されているすべてのプロット点・線・円グラフの面を選択します。

操作 このメニューを選ぶだけで選択します。

[選択] - [選択解除]

機能 表示されているすべてのプロット点・線・円グラフの面の選択状態を解除します。

操作 このメニューを選ぶだけで解除します。

[表示] - [選択データのみ表示]

機能 選択されているプロット点・線・円グラフの面のみを表示します。

操作 このメニューを選ぶだけで表示します。

[表示] - [選択データのみ非表示]

機能 選択されているプロット点・線・円グラフの面のみを非表示します。

操作 このメニューを選ぶだけで非表示になります。

[表示] - [すべて表示]

機能 すべてのプロット点・線・円グラフの面を表示します。

操作 このメニューを選ぶだけで表示が反転します。

[表示] - [表示/非表示反転]

機能 プロット点・線・円グラフの面の表示状態を反転させます。

操作 このメニューを選ぶだけで表示が反転します。

[表示] - [視点をリセット]

機能 グラフを画面の中央に表示します。

操作 このメニューを選ぶだけで表示されます。

[表示] - [グラフツリーの表示]

機能 グラフツリーを表示させます。

操作 グラフツリーを閉じている場合、このメニューを選択することで、グラフツリーを再び表示させることができます。

[表示] - [メッセージウィンドウの表示]

機能 メッセージウィンドウを表示させます。

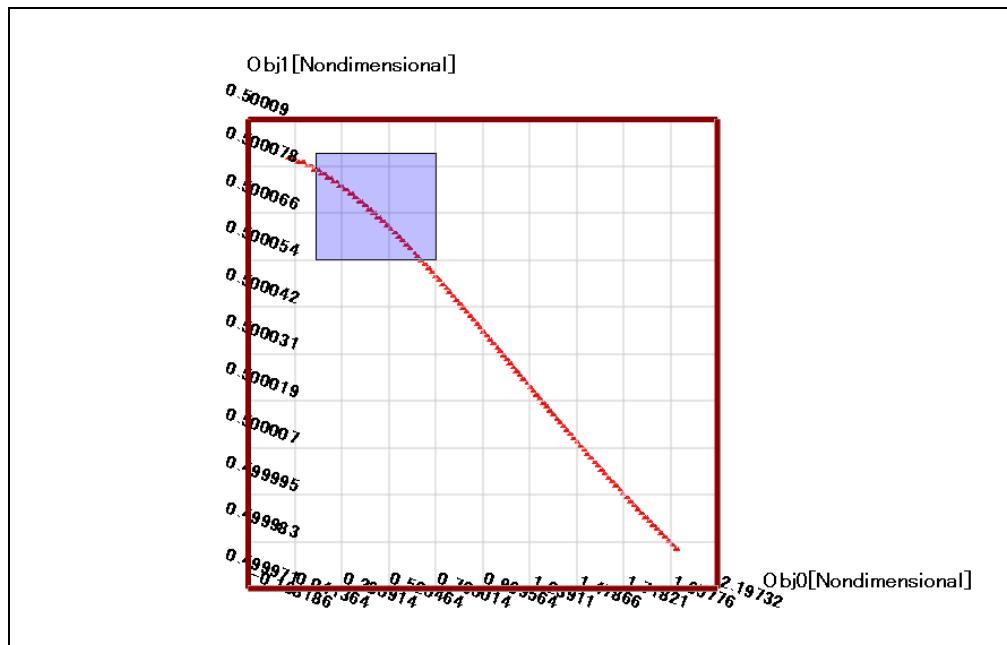
操作 メッセージウィンドウを閉じている場合、このメニューを選択することで、グラフツリーを再び表示させることができます。

[表示] - [ラバーボックス拡大表示]

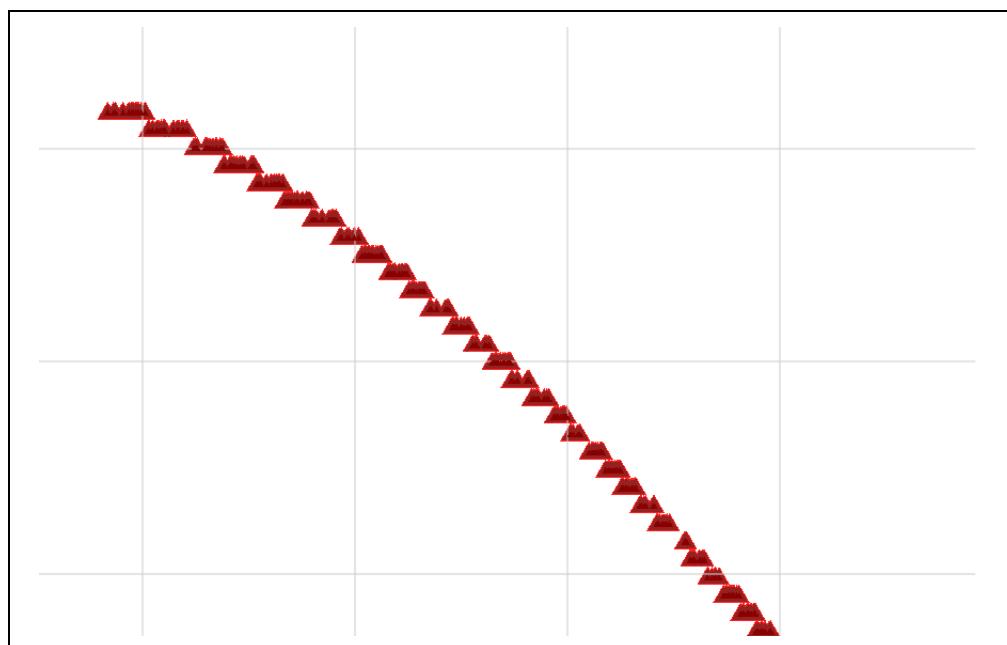
機能 ラバーボックスで囲った部分を拡大表示します。

操作 メニュー選択後、ドローウィンドウ上でマウス左ボタンをクリックした状態でドラッグし、拡大表示させる範囲を指定します。

1. 拡大表示する範囲を指定する。



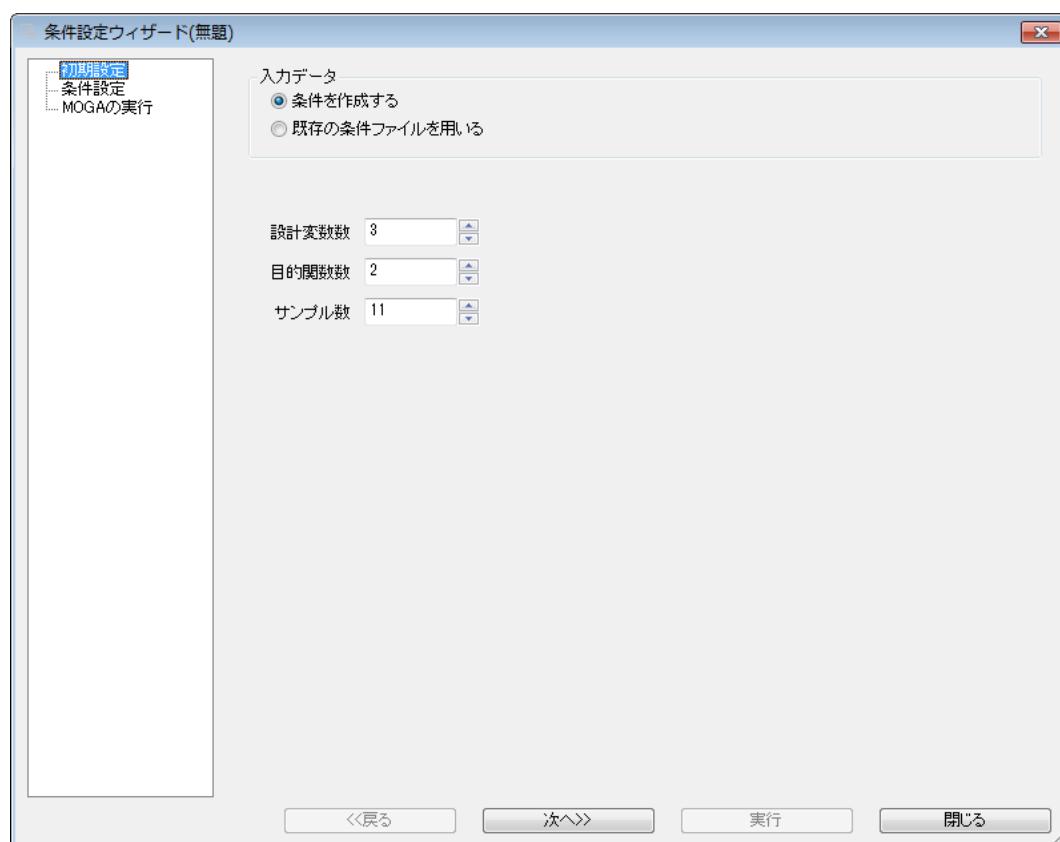
2. マウス左ボタンを離すと指定範囲を拡大表示する。



[条件設定] - [条件設定ウィザード...]

機能 [条件設定ウィザード]を表示します。

操作 このメニューを選ぶだけで表示されます。



参照 条件設定ウィザードの詳細な設定内容は

第2章 2.3 操作手順の具体例の(2) 初期設定

第2章 2.3 操作手順の具体例の(3) 条件設定

第2章 2.3 操作手順の具体例の(4) MOGAの実行

を参照してください。

[操作] - [1ボタンモード]

機能 マウス操作を1ボタンモードに切り替えます。

操作 このメニューを選択すると、マウス操作が1ボタンモードに切り替わります。

[操作] - [2ボタンモード]

機能 マウス操作を2ボタンモードに切り替えます。

操作 このメニューを選択すると、マウス操作が2ボタンモードに切り替わります。

[操作] - [3ボタンモード(CTRL)]

機能 マウス操作を3ボタンモード(CTRL)に切り替えます。

操作 このメニューを選択すると、マウス操作が3ボタンモード(CTRL)に切り替わります。

[操作] - [3ボタンモード]

機能 マウス操作を3ボタンモードに切り替えます。

操作 このメニューを選択すると、マウス操作が3ボタンモードに切り替わります。

[操作] - [3D回転]

機能 1ボタンモードのとき、画面の表示を回転させます。

操作 画面上で、マウスをドラッグすると、表示が回転します。

[操作] - [2D回転]

機能 1ボタンモードのとき、画面の表示を画面法線を軸に回転させます。

操作 画面上で、マウスを左右にドラッグすると、表示が画面法線を軸に回転します。

[操作] - [移動]

機能 1ボタンモードのとき、画面の表示を移動させます。

操作 画面上で、マウスをドラッグすると、表示が移動します。

[操作] - [拡大・縮小]

機能 1ボタンモードのとき、画面の表示を拡大縮小させます。

操作 画面上で、マウスを右にドラッグすると、表示が拡大し、左にドラッグすると表示が縮小します。

[操作] - [オペレーション]

機能 マウス操作系を変更します。

操作 このメニューを選ぶと、[マウス操作の変更]ダイアログが表示されます。

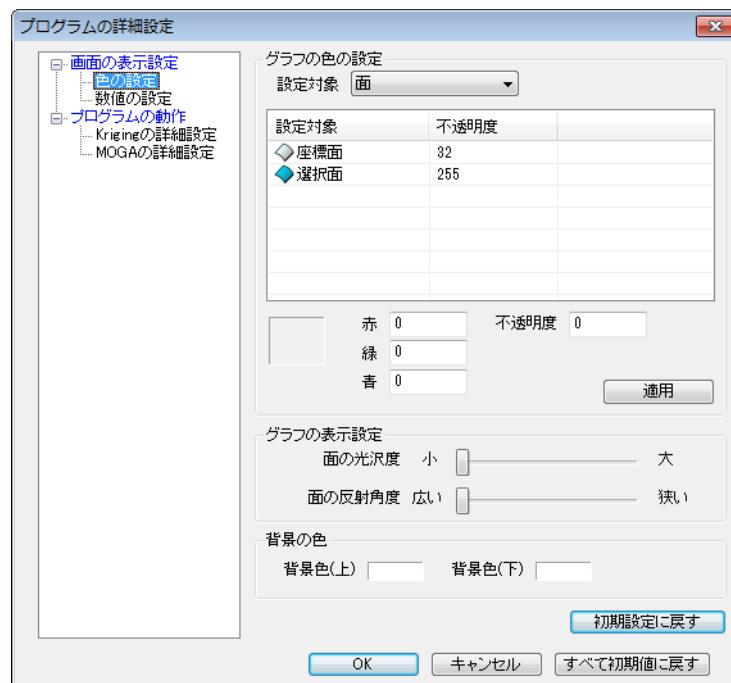


詳細は第2章 2.3 操作手順の具体例のドローウィンドウを参照してください。

[オプション] - [プログラムの詳細設定]

機能 EOopti内部の詳細なパラメータに対して設定を行うことができます。
操作 このメニューを選ぶことで、[プログラムの詳細設定]ダイアログが表示されます。

- [画面の表示設定]
 - [色の設定]



[グラフの色の設定]

[設定対象]は、ドローウィンドウ上に表示されるグラフの面・辺・点・文字列の色やサイズを変更することができます。

[グラフの表示設定]

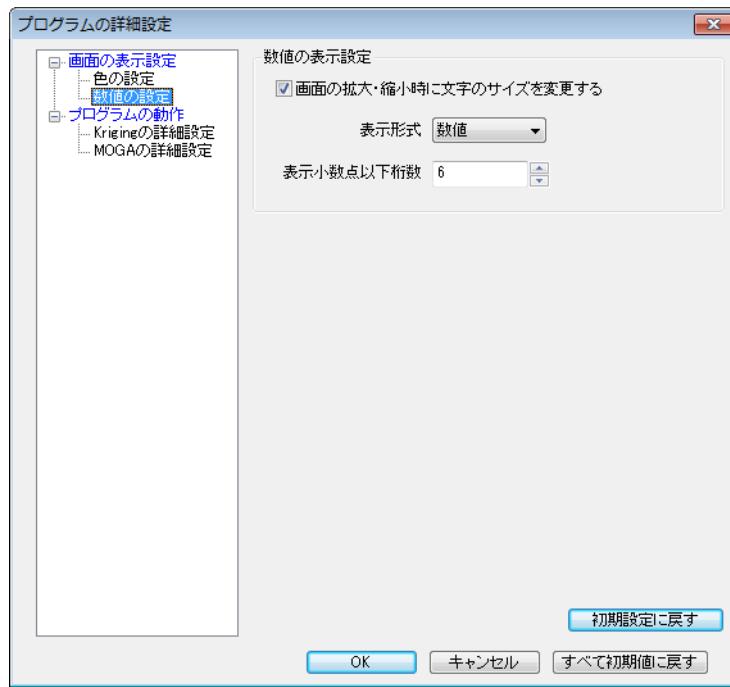
[面の光沢度]と[面の反射する角度]は面にハイライトをどのように入れるかを設定します。光沢度が高いと面が磨かれているように表示され、反射する角度を狭くすることで、細い光があたっているような効果が得られます。

[背景の色]

[背景色 (上)], [背景色 (下)]

これらはドローウィンドウの画面の背景の色を設定します。上下で異なる色に設定するとグラデーション効果が得られます。

- [数値の設定]



- [数値の表示設定]

[画面の拡大・縮小時に文字のサイズを変更する]

チェックをONにすると、ドローウィンドウ上でグラフを拡大・縮小したときに、グラフ上に表示されている数値や軸名称のサイズが同時に拡大・縮小されます。デフォルトはONです。

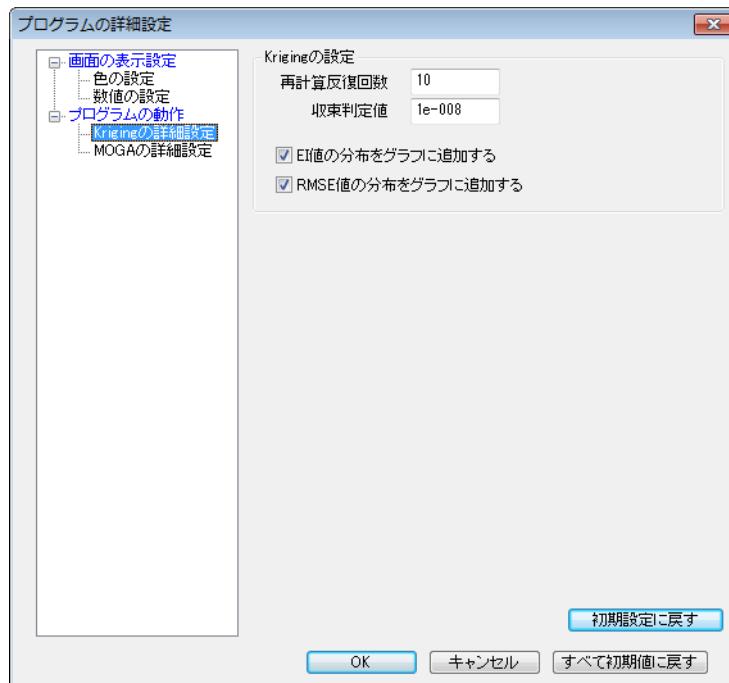
[表示形式]

[条件設定ウィザード]-[条件設定]の設計変数・目的関数の[表示形式]に適用されるデフォルト値です。デフォルトは[数値]です。

[表示小数点以下桁数]

ドローウィンドウのグラフ上に表示される数値桁数のデフォルト値です。デフォルトは6です。

- [Krigingの詳細設定]



Kriging法で使われる内部のパラメータを設定・変更できます。

- [Krigingの設定]

[再計算反復回数]

応答曲面生成時に算出されるKrigingパラメータは、尤度関数が最大となるような値を、遺伝的アルゴリズムを用い、収束判定を伴った反復計算を行うことで求めています[1]。

[再計算反復回数]は、反復計算によって尤度関数の最大値の誤差が収束判定値に達した後の再計算回数に相当します。再計算を行うことによって、より確実に尤度関数の最大値を求めることができます。デフォルトは10です。

[収束判定値]

尤度関数の反復計算の収束判定値です。尤度関数の反復計算による誤差が収束判定値以下になった後、[再計算反復回数分]の計算を行います。再計算回数分の誤差が全て収束判定値以下になった場合にKrigingの計算が終了します。デフォルトは1.0e-8です。

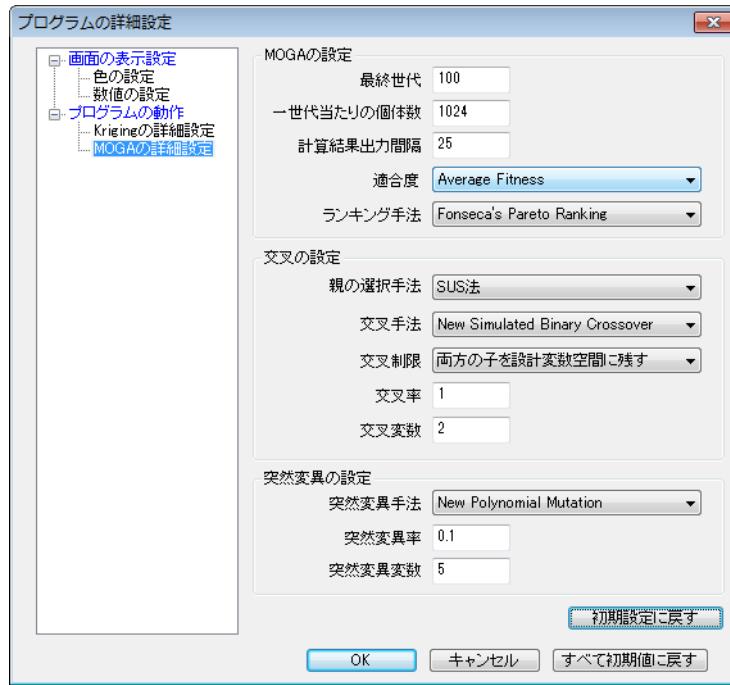
[EI値の分布をグラフに追加する]

チェックをONにすることで、応答曲面生成時に算出されるEI値の分布がグラフツリーに追加されます。デフォルトは、OFFです。

[RMSE値の分布をグラフに追加する]

チェックをONにすることで、応答曲面生成時に算出されるRMSE値の分布がグラフツリーに追加されます。デフォルトは、OFFです。

- [MOGAの詳細設定]



MOGAで使われる内部パラメータの設定・変更ができます。

- [MOGAの設定]

[最終世代]

[条件設定ウィザード]-[MOGAの実行]の[最終世代]のデフォルト値に相当します。デフォルトは100です。

[一世代当たりの個体数]

[条件設定ウィザード]-[MOGAの実行]の[一世代あたりの個体数]のデフォルト値に相当します。デフォルトは1024です。

[計算結果出力間隔]

[条件設定ウィザード]-[MOGAの実行]の[計算結果出力間隔]のデフォルト値に相当します。デフォルトは25です。

[適合度]

個体の適合度の算出手法を設定します。デフォルトは[Average Fitness]です。

以下に各適合度の算出手法を示します。

F_i :個体*i*の適合度

N :個体数

μ :ランクに属する個体数

r_i :個体*i*のランク

[Fitness Base on Rank (1/rank)]

適合度を次式に基づいて求めます。

$$F_i = 1/r_i$$

[Fitness Base on Rank]

適合度を次式に基づいて算出します。

$$F_i = \alpha \cdot (1 - \alpha)^{r_i - 1}$$

$$\alpha = 0.08$$

[Average Fitness]

適合度を、次式に基づいて求めます。同ランクの個体の適合度の平均値(平均適合度)を求めていきます。

$$F_i = N - \sum_{k=1}^{r_i-1} \mu(k) - 0.5(\mu(r_i) - 1)$$

[ランキング法]

個体群に対して解の優劣を判定するためのランキング法を設定します。デフォルトは[Fonseca's Pareto Ranking]です。

[Goldberg's Pareto Ranking]

Goldberg[2]によって提案されたランキング法です。以下のようなステップによって各個体に対してランクを与えていきます。

Step0 :

ランクを表す指標を $i = 1$ とする。

Step1 :

現在の世代の個体群の中で、どの個体に対しても勝るとも劣らない個体に対してランク i を割り当てる。

Step2 :

ランク付けされた個体を取り除き、 $i = i+1$ とする。

Step3 :

全ての個体に対してランクを割り当てた場合終了する。そうでなければStep1へ戻る。

[Fonseca's Pareto Ranking]

Fonseca[3]らによって提案され、Goldbergのランキング法を拡張した手法です。すべての非劣解を持つ個体にランク1を割り当て、他の個体に対しては非劣解数(自分より優れた個体の数)に1を加えた数をランク数として割り当てています。

- [交叉の設定]

[親の選択手法]

交叉をする際の親の選択手法を設定します。デフォルトは[SUS]です。

[RSW]

RSW(Roulette Wheel Selection)は、個体群をルーレット盤に見立て、個体の適合度の度合いから選択確率に応じてルーレット盤上で占める割合を変更していきます。この作業を各個体に対して行っていき、ルーレット盤上での占有率を求めていくという手法です。適合度の高い個体ほど、占有率が大きくなり、選択される確率が高くなります。

[SUS]

SUS(Stochastic Universal Selection)は、RSWを拡張した手法で、ルーレット盤の外に選択個体数分のマーカーを付けてルーレットを回し、マーカーに位置する個体を選択する手法です。マーカーが等間隔のため、期待値分の個体数は必ず選択されるのが特徴です。

[交叉手法]

交叉は、2つの親個体から次の世代に残す子個体を生成する手法です。ここでは交叉の手法を選択できます。デフォルトは、[New Simulated Binary Crossover]です。

以下に各交叉手法を簡単に説明します。

o_i : 子個体の設計変数値

p_i : 親個体の設計変数値

α : 交叉変数

γ : 乱数($0 \leq \gamma \leq 1$)

[Arithmetic Crossover]

選択された2つの親個体の線形結合によって以下のように求められます。

$$\begin{aligned}o_1 &= \beta \cdot p_1 + (1 - \beta) \cdot p_2 \\o_2 &= \beta \cdot p_2 + (1 - \beta) \cdot p_1 \\\beta &= (2 \cdot \alpha + 1) \cdot \gamma - \alpha\end{aligned}$$

[Blend Crossover]

Blend Crossoverは2親個体を頂点とする範囲を下式のように若干外側に広げ、その中で子個体を一様分布によって求める手法です。

$$\begin{aligned}x_1 &= \min(p_1, p_2) - \alpha \cdot d \\x_2 &= \max(p_1, p_2) + \alpha \cdot d \\d &= |p_1 - p_2|\end{aligned}$$

交叉変数 α は文献[4]によると0.5,[5]によると0.36が用いられています。

[Simulated Binary Crossover (Fixed)]

Simulated Binary Crossover(SBX)は、下式の β を用いて子個体の設計変数値を求めます。SBXの大きな特徴は、乱数 γ 、交叉変数 α の値を一定にした場合でも、親個体が互いに接近していれば親個体に近い子個体が生成され、親個体が離れている場合には親個体から離れた子個体が生成される点です。最適解探索の初期段階では大域的探索を行うために設計変数空間に散布された個体が生成されますが、探索が進み個体群が収束してきたときには近傍の子個体が生成されるというのが特徴です。

$$\beta = \begin{cases} (2\gamma)^{\frac{1}{\alpha+1}} & \text{if } \gamma \leq 0.5 \\ \frac{1}{2(1-\gamma)^{\frac{1}{\alpha+1}}} & \text{otherwise} \end{cases}$$

$$o_1 = 0.5[(1 + \beta)p_1 + (1 - \beta)p_2]$$

$$o_2 = 0.5[(1 - \beta)p_1 + (1 + \beta)p_2]$$

[Simulated Binary Crossover (Fixed)]では、交叉時に全設計変数に対して交叉を行います。

[Simulated Binary Crossover]

交叉手法は[Simulated Binary Crossover (Fixed)]と同じですが、交叉確率は各設計変数に対して適用するため、より親個体に近い子個体が生成される可能性が高くなります。

[New Simulated Binary Crossover]

[Simulated Binary Crossover]の β 値の算出時に用いる乱数 γ 値を一様乱数ではなく、親の設計変数値に応じて値を変更するように拡張したものです。

[交叉制限]

交叉による設計変数値の制限を選択できます。デフォルトは、[両方の子を設計変数空間に残す]です。

[両方の子を設計空間に残す]

交叉によって新たに生成された個体の設計変数値が指定範囲を超えた場合、設計変数値を新たに求めなおします。

[個体の上限値を制限する]

交叉によって新たに生成された個体の設計変数値が指定範囲を超えた場合、設計変数値の上限・下限値を新たな設計変数値として使用します。

[交叉率]

選択した[交叉手法]に適用される交叉率を設定します。デフォルトは1.0です。

[交叉変数]

選択した[交叉手法]に適用される交叉変数を設定します。デフォルトは2.0です。

- [突然変異の設定]

[突然変異手法]

突然変異は、最適解が局所的な解に陥らないよう、交叉によって生成された新たな個体の設計変数値を、指定した突然変異率の確率に応じて変更させる手法です。ここでは、突然変異の手法を選択できます。デフォルトは[New Polynomial Mutation]です。

以下に各交叉手法を簡単に説明します。

α : 突然変異変数

γ : 乱数($0 \leq \gamma \leq 1$)

[Random Mutation]

設定されている突然変異率に従って、乱数によるランダムな設計変数値を与える手法です。

[Uniform Mutation]

子個体の設計変数値に、以下の式によって求められた値 μ を追加します。追加した設計変数値が指定範囲以内に収まるよう、 μ の値は調整されます。

$$\mu = (\gamma - 0.5)\alpha$$

[Polynomial Mutation]

μ の値を以下の式によって求め、子個体の設計変数値に追加します。

$$\mu = \begin{cases} (2\gamma)^{\frac{1}{\alpha+1}} & \text{if } \gamma \leq 0.5 \\ \frac{1}{2(1-\gamma)}^{\frac{1}{\alpha+1}} & \text{otherwise} \end{cases}$$

[New Polynomial Mutation]

μ 値の算出時に用いる γ 値を一様乱数ではなく、親の設計変数値に応じて値を変更するように[Polynomial Mutation]拡張したものです。

[突然変異変率]

選択した突然変異手法に適用される突然変異変率を設定します。デフォルトは0.1です。

[突然変異変数]

選択した突然変異手法に適用される突然変異変数 α を設定します。デフォルトは5.0です。

参考文献

1. S, Jeong; K, Chiba; S, Obayashi. Data Mining for Aerodynamic Design Space. Journal of Aerospace Computing, Information and Communication vol.2, 2005.
2. Goldberg, D; Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, Reading MA, 1989
3. Fonseca, C.M. and Fleming, P.J: J. Fleming, Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization, Proceedings of the 5th International Conference on Genetic Algorithms, pp416-423, 1993
4. Eshelman, L. J., Caruna, R.A. Schaffer, D. Real-coded Genetic Algorithms and Interval-schemata. Proceedings of Foundations of Genetic Algorithms 2(FOGA-2), pp. 187-202, 1993.
5. Eshelman, L. J., Mathias, K. E. and Schaffer, J.D. Crossover Operator Biases: Exploiting the Population Distribution. Proceedings of the Seventh International Conference on Genetic Algorithms, pp. 354-361, 1997.

[ウィンドウ]-[新しいウィンドウを開く]

機能 現在開いているプロジェクトを別のウィンドウで表示します。

操作 このメニューを選ぶだけで実行されます。

補足 このメニューで作られるウィンドウに表示されるのは同じプロジェクトですが、その中の別々のグラフを表示できます。同じプロジェクトのウィンドウが全て閉じられるとプロジェクトが閉じられます。

[ヘルプ] - [ユーザーズガイド]

機能 Extension Option(Optimization)のPDFマニュアルを開きます。

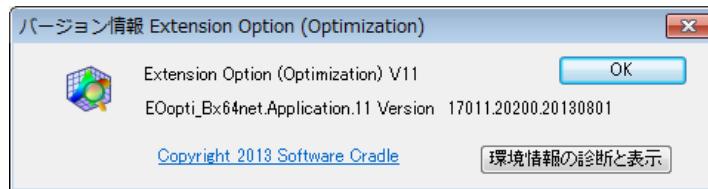
操作 このメニューを選ぶだけで表示されます。

補足 PDFを表示するソフトウェアがインストールされていないと表示されません。

[ヘルプ] - [バージョン情報Extension Option(Optimization)]

機能 Extension Option(Optimization)のバージョン情報を表示します。

操作 このメニューを選ぶだけで表示されます。



環境情報の診断と表示をクリックすると、Extension Option(Optimization)を起動しているPCとExtension Option(Optimization)の情報がテキストファイルとして表示されます。

何らかの不具合などが起こって弊社サポートに問い合わせる際には、このファイルを添付するようにしてください。調査が迅速に行えるようになります。

```
C:\$Users\$*****\$AppData\$Local\$Temp\$Extension Option (Optimization)_20130801_1640_環境情報.txt
=====
OS =====
Windows 7 Ultimate Edition, 64-bit Service Pack 1 (build 7601)
=====
起動中のプログラム =====
C:\$Program Files\$Cradle\$EOopti11\$Programs_x64\$EOopti_Bx64net.exe (EOopti Version 7011.20200.20130801)
=====
メモリ情報 =====
物理メモリ(マシン全体):24573(MB)
物理メモリ(利用可能):17812(MB)
物理メモリ(Extension Option (Optimization)):80(MB)
仮想メモリ(Extension Option (Optimization)):85(MB)

=====
システム共有ライブラリの情報 =====
mfc100u.dll (10.00.40219.332)
msvcr100.dll (10.00.40219.332)
=====
ホームフォルダ =====
C:\$Users\$*****\$AppData\$Roaming\$Cradle\$EOopti11\$EOopti_Bx64net\$
=====
環境設定ファイル =====
C:\$Users\$*****\$AppData\$Roaming\$Cradle\$EOopti11\$EOopti_Bx64net\$EOopti_Bx64net.env
```

[ポップアップメニュー]

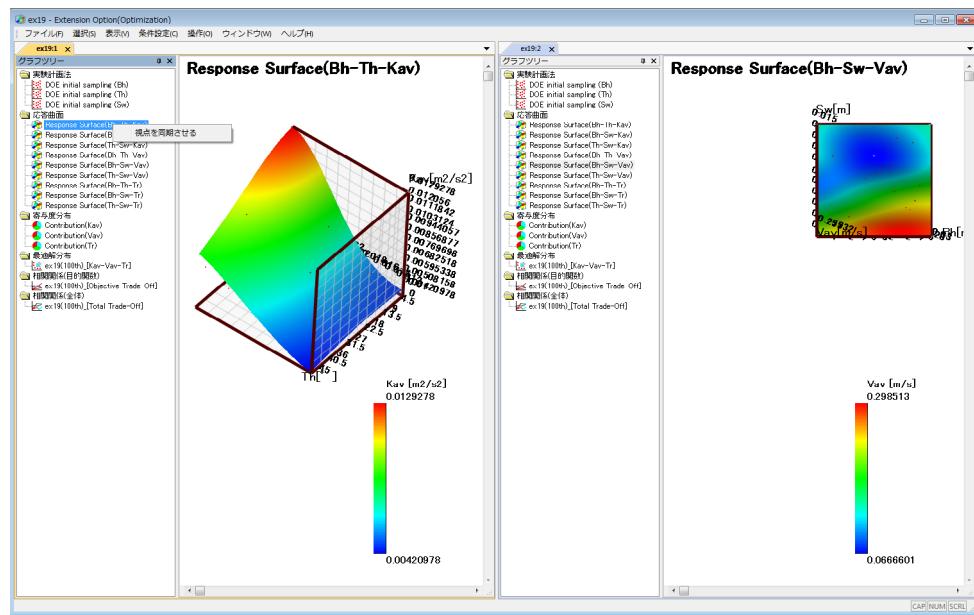
ポップアップメニューはグラフツリー、またはドローウィンドウ上でマウス右ボタンをクリックすると表示されます。

[ポップアップメニュー(グラフツリー)] - [視点を同期させる]

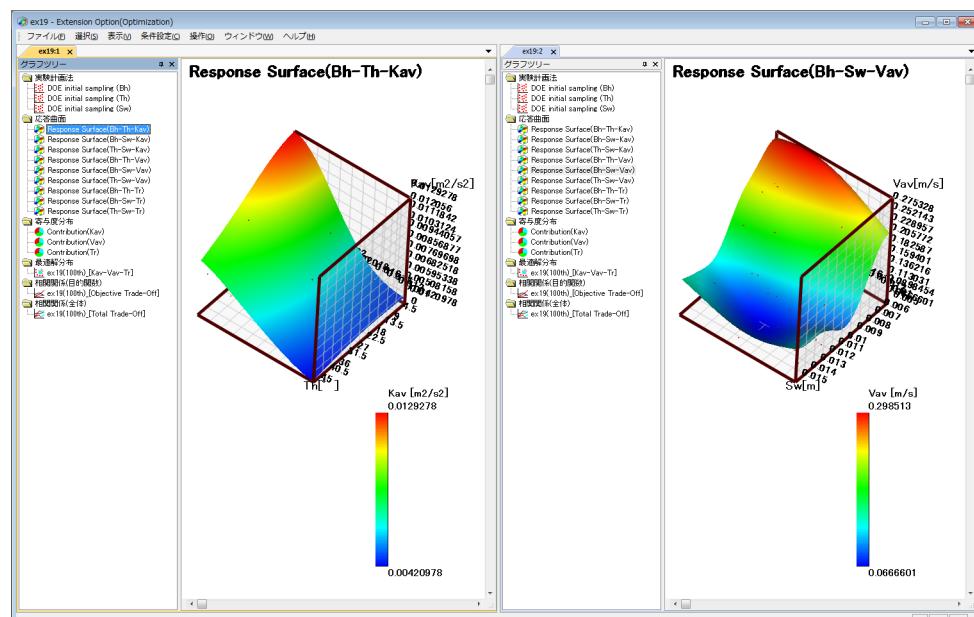
機能 [ウィンドウ]-[新しいウィンドウを開く]で1つのドキュメント内で複数のウィンドウが開いている場合、他のウィンドウのグラフの視点を同期させます。

操作 グラフツリーでポップアップメニューを開き、[視点を同期させる]を選択します。

視点同期前の表示状態



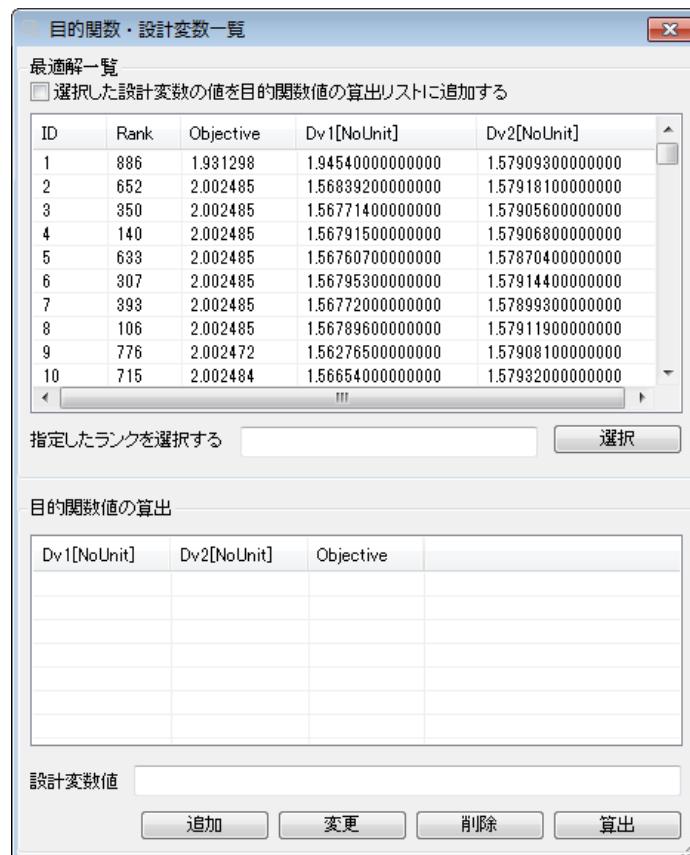
視点同期後の表示状態



注. 同期されるグラフは、選択されたグラフと同じ種類のグラフのみです。

[ポップアップメニュー(ドローウィンドウ)] - [目的関数・設計変数一覧]

- 機能** 最適化結果として出力された目的関数、設計変数の一覧リストを表示します。
- 操作** グラフツリーで応答曲面グラフ、最適解分布グラフ、相関関係グラフを選択しているときに、ドローウィンドウ上でポップアップメニューを開くと選択できます。



• [最適解一覧]

リストの行を選択すると、該当する最適解分布グラフの点、または相関関係グラフの折れ線グラフが選択され、値がグラフ上に表示されます。リストの各列には以下の値が表示されます。

1列目：出力された世代の個体番号(ID)

2列目：各個体のランク(ランク 1 の個体がパレート最適解に相当します。)

3列目以降：各個体における目的関数値、設計変数値

[選択した設計変数の値を目的関数値の算出リストに追加する]

チェックをONにすると、リストで選択した設計変数の値が、[目的関数値の算出]のリストに出力されます。

[指定したランクを選択する]

エディットボックスに値を入れることで、入力されたランクの個体のみを選択することができます。カンマ(,)区切りでランク値を入力することで、複数のランクを指定することも可能です。

- [目的関数値の算出]

[設計変数値]のエディットボックスに設計変数値を入力し、[追加]をクリックすると、指定した設計変数値がリストに追加されます。また、[選択した設計変数の値を目的関数値の算出リストに追加する]のチェックがONのとき、[最適解一覧]リストで選択された設計変数値がリストに追加されます。[算出]をクリックすると、入力された設計変数値に対する目的関数値がリストに表示されます。

[設計変数値]

[目的関数値の算出]リストに追加する設計変数値を入力します。設計変数数が複数の場合、カンマ(,)区切りで、各設計変数の値を入力します。

追加

クリックすると、[設計変数値]に入力されている設計変数の値がリストに追加されます。

変更

クリックすると、リストの選択行の設計変数値が[設計変数値]のエディットボックスに入力されている値に変更されます。

削除

クリックすると、リストの選択行が削除されます。

算出

クリックすると、リストの各行の設計変数値に対する目的関数値が算出され、リスト上に値が表示されます。

いずれのリストも、行を選択した状態でCtrl+Cキーを押すと、クリックボードに選択された値がコピーされ、他のエディターに貼り付けることができます。また、

- [ポップアップメニュー]

リスト上でマウス右ボタンをクリックすると、以下のポップアップメニューが表示されます。

[選択した設計変数値を通過する応答曲面を生成する]

このメニュー選択すると、リスト内で最後に選択された行の設計変数値を通過する応答曲面を生成し、応答曲面グラフが更新されます。

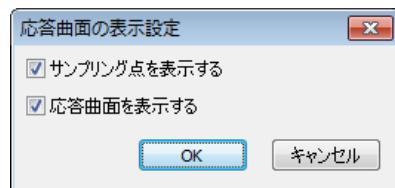
[選択した設計変数の値をサンプル点として追加する]

このメニュー選択すると、リスト内で最後に選択された行の設計変数値を新たなサンプル点として、[条件設定ウィザード]-[条件設定]-[実験計画法]のリストに設計変数値が追加されます。

[ポップアップメニュー(ドローウィンドウ)] - [応答曲面の表示設定]

機能 応答曲面のグラフの表示設定を行います。

操作 グラフツリーで応答曲面グラフを選択しているときに、ドローウィンドウ上でポップアップメニューを開くと選択できます。



- **[サンプリング点を表示する]**

チェックをON/OFFにすると、実験計画法で与えられたサンプル値のプロット点が、表示/非表示されます。デフォルトはONです。

- **[応答曲面を表示する]**

チェックをON/OFFにすることで、応答曲面の表示/非表示が行えます。デフォルトはONです。

[ポップアップメニュー(ドローウィンドウ)] - [グラフの表示設定]

機能 グラフの表示範囲の設定が行えます。

操作 グラフツリーで[実験計画法]、[応答曲面]、[最適解分布]、[相関関係(目的関数)]、[相関関係(全体)]グラフを選択しているときに、ドローウィンドウ上でポップアップメニューを開くと選択できます。



- [グラフの表示設定]

[最小値]

グラフの表示範囲の最小値を指定します。

[最大値]

グラフの表示範囲の最大値を指定します。

[目盛り間隔]

数値を表示させる目盛りの間隔を各軸方向に対して指定します。

[補助目盛り間隔]

目盛り間隔の間に表示させる補助目盛り間隔を指定します。

[表示桁数]

グラフに表示される桁数を指定します。デフォルトは、[オプション]-[プログラムの詳細設定]-[数値の設定]-[表示小数点以下桁数]で指定されている桁数が入力されます。

[表示形式]

グラフに表示される数値の表示形式を指定します。デフォルトは、[オプション]-[プログラムの詳細設定]-[数値の設定]-[表示形式]で指定されている形式が適用されます。

[頂点の大きさ]

グラフに表示される点の大きさを変更します。

- [表示/非表示設定]

[タイトルを表示する], [目盛り数値を表示する], [軸名称を表示する], [座標面を表示する]

チェックをON/OFFにすることで、グラフ上に表示されている数値やタイトル等を表示/非表示にす
ることができます。デフォルトはONです。

- [元に戻す]

クリックすると、ダイアログを開いたときの設定値に戻ります。

第5章 ファイル

5.1 ファイルの内容

EOopti Version12で扱う主なファイルは、条件ファイル(_in.csv)、Krigingファイル(_krig.csv)、計算結果ファイル(_out.csv)です。各ファイルには、いくつかのセクション毎に出力内容が分かれており、各セクションは

セクション名
出力内容
セクション名-END

という形式で出力されます。以下に、各セクションの出力形式について説明します。
なお、文中に使われる変数は次の通りです。

N_{dv} : 設計変数数
 N_{obj} : 目的関数数
 N_{sample} : サンプル数
 $N_{population}$: 個体数

- GENERATIONセクション
MOGAの計算終了時における世代数が出力されます。出力形式は以下の通りです。

GENERATION
世代数
END_GENERATION

- INPUTDATAセクション
Kriging法・MOGAを実行するための条件が出力され、[条件設定ウィザード]-[条件設定]の設定値に相当します。出力形式は以下の通りです。

INPUTDATA
プロジェクト名
設計変数の名前× N_{dv}
設計変数の単位名× N_{dv}
設計変数の最小値× N_{dv}
設計変数の最大値× N_{dv}
設計変数の数値型× N_{dv}
設計変数の離散化幅× N_{dv}
設計変数の表示形式× N_{dv}
目的関数の名前× N_{obj}
目的関数の単位名× N_{obj}
目的関数の設計指針× N_{obj}
目的関数の探索値× N_{obj}
目的関数の表示形式× N_{obj}
設計変数のサンプル値× N_{dv} , 目的関数のサンプル値× N_{obj} (N_{sample} 分繰り返す)
END_INPUTDATA

- KRIGPARAMセクション

Kriging法によって算出されたKrigingパラメータ(応答曲面のフィッティング値)が出力されます。このセクションは、[条件設定ウィザード]-[MOGAの実行]で、[Krigingを実行する]を選択した場合にOutputされます。出力形式は以下の通りです。

KRIGPARAM

Krigingパラメータ $\times N_{dv}$
END_KRIGPARAM

- ANOVARESULTセクション

寄与度分布の算出結果が出力されます。このセクションは、[条件設定ウィザード]-[MOGAの実行]で、[Krigingを実行する]を選択した場合にOutputされます。出力形式は以下の通りです。

ANOVARESULT

寄与度 $\times N_{dv}$ (N_{obj} 分繰り返す)
END_ANOVARESULT

- RSRESULTセクション

MOGAによる目的関数の最適解情報が出力されます。このセクションは、[条件設定ウィザード]-[MOGAの実行]で、[目的関数の最小値/最大値を求める]を選択した場合にOutputされます。出力形式は以下の通りです。

RSRESULT

個体のランク, 目的関数の値 $\times N_{obj}$, 設計変数の値 $\times N_{dv}$ ($N_{population}$ 分繰り返す)
END_RSRESULT

- EIRESULTセクション

MOGAによるEI値の最適解情報が出力されます。このセクションは、[条件設定ウィザード]-[MOGAの実行]で、[EI値の最大値を求める]を選択した場合にOutputされます。出力形式は以下の通りです。

EIRESULT

個体のランク, EI値 $\times N_{obj}$, 設計変数の値 $\times N_{dv}$ ($N_{population}$ 分繰り返す)
END_EIRESULT

- RMSERESULTセクション

MOGAによるRMSE値の最適解情報が出力されます。このセクションは、[条件設定ウィザード]-[MOGAの実行]で、[RMSE値の最大値を求める]を選択した場合にOutputされます。出力形式は以下の通りです。

RMSERESULT

個体のランク, RMSE値 $\times N_{obj}$, 設計変数の値 $\times N_{dv}$ ($N_{population}$ 分繰り返す)
END_RMSERESULT

5.2 条件ファイルの内容

条件ファイル(_in.csv)の出力内容は以下の通りです。

各セクションの詳細については、[5.1 ファイルの内容](#)を参照してください。

バージョン情報

INPUTDATAセクション

条件ファイルを読み込むことで、[条件設定ウィザード]-[条件設定]の設定値が適用されます。

5.3 Kriging ファイルの内容

Kriging ファイル(_krig.csv)の出力内容は以下の通りです。

各セクションの詳細については、[5.1 ファイルの内容](#)を参照してください。

バージョン情報

INPUTDATAセクション

KRIGPARAMセクション

ANOVARESULTセクション

Kriging ファイルを読み込むことで、[条件設定ウィザード]-[条件設定]の設定値、および応答曲面、寄与度分布が表示されます。また、Kriging ファイルが存在している場合、[条件設定ウィザード]-[MOGAの実行]で、[出力したKrigingパラメータを用いる]を選択することで、応答曲面生成時間を短縮し、即座にMOGAを実行させることができます。

5.4 計算結果ファイルの内容

計算結果ファイル(_out.csv)の出力内容は以下の通りです。

各セクションの詳細については、[5.1 ファイルの内容](#)を参照してください。

バージョン情報

INPUTDATAセクション

KRIGPARAMセクション

ANOVARESULTセクション

RSRESULTセクション(目的関数のMOGA実行時)

EIRRESULTセクション(EI値のMOGA実行時)

RMSERESULTセクション(RMSE値のMOGA実行時)

計算結果ファイルを読み込むことで、[\[条件設定ウィザード\]-\[条件設定\]](#)の設定値が適用され、すべてのグラフの表示が可能になります。