

Compensator

コース概要

このコースでは、バージョン 2014.1 より TD Professional の標準機能となったアドオン Compensator の機能(変形領域の表示、変形領域の認識、点の取り出し など)を学習します。

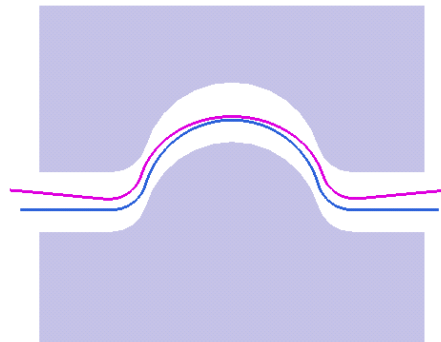
使用するファイル	S-rail.e3
	after_blank.nas
	before_blank.nas
	Rail_Mirror.e3
	Rail_Mirror_after.pt
	Rail_Mirror_before.pt

目次

Step 1: Compensator とは.....	3
Step 2: Compensator の動作.....	4
Step 3: 変形領域の認識.....	5
Step 4: 境界の条件と最適化.....	11
Step 5: 変形領域のクリーニングと対称拘束.....	15
Step 6: その他のコマンド.....	20

Step 1: Compensator とは

Compensator とは、スプリングバックなどの FEM 解析結果をもとに、見込み形状を作成するツールです。



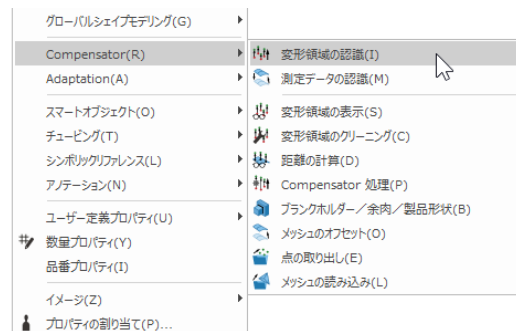
Compensator には次の機能が含まれます。

- Compensator ツール

変形領域の認識、変形領域の表示、変形領域のクリーニング、距離の計算 等

- メッシュの編集ツールの一部

メッシュの整列、対称性チェック、メッシュアナライザー 等

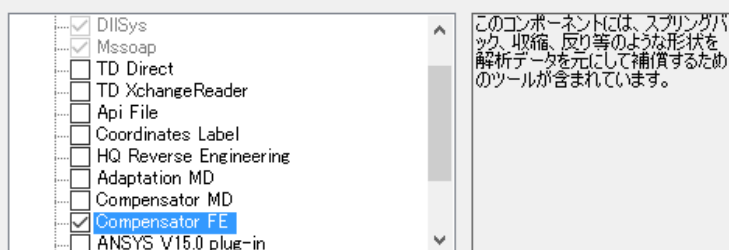


Compensator の各機能は、バージョン 2014.1 よりライセンス TD Professional で動作する機能となりました。

また、Compensator の各機能は、ThinkDeisgn のインストール時に「カスタムインストール」を選択してインストールします。

「Compensator FE」を選択してインストールしてください

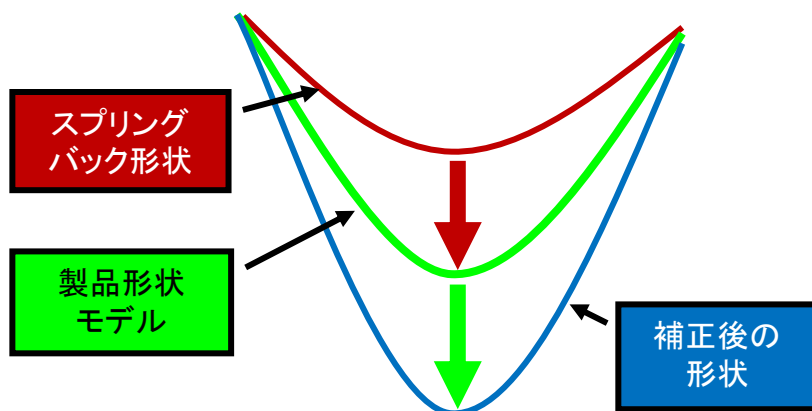
下のツリーのアイコンをクリックして、機能をインストールする方法を変更してください。



[リセット\(S\)](#)

Step 2: Compensator の動作

Compensator では、製品形状モデル(下図緑)と解析後のスプリングバック形状(赤)の 2 つを用いて 赤→緑 方向の変形量を計算し、それを製品形状モデル(緑)に適用することで、スプリングバックを補正した見込み形状(青)の CAD モデルを得ます。



赤と緑の形状は、解析ソフトのファイル(メッシュ)を直接使用します。以下のフォーマットがサポートされています。

Nastran format (.nas)

Autoform format (.af)

Forming Technologies format (.fpi)

STL

通常は 赤→緑 の変形量を計算しますが、緑→赤 方向の変形量を計算することでスプリングバック形状(赤)の CAD モデルを得ることもできます。

また、解析ソフトがスプリングフォワード形状(青)を直接算出可能な場合、緑→青 の変形量を計算することで、補正後の形状の CAD モデル(青)を得ることもできます。

変形量の計算は、解析ソフトのファイルから点を抽出して行います。

また使用する点は、その数や順序が揃っている必要があります。

Compensator は主として板金プレス製品に対して使用することが想定されていますが、解析前後のデータがあれば、その他の製品(樹脂製品の反りなど)に対しても適用することができます。



Step 3: 変形領域の認識

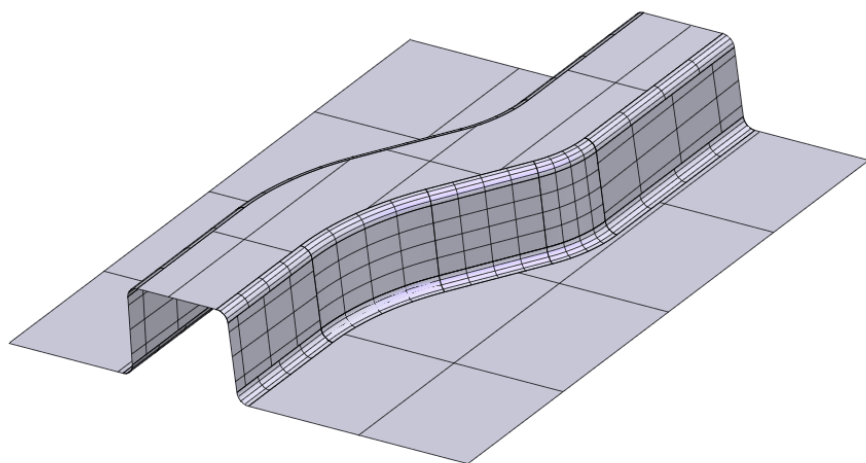
Compensator では、解析データからの点の取り出し、変形量の計算、モデルへの適用、の3段階で処理を行います。始めに簡単なデータを用いて実際に処理を行ってみます。ダウンロードしたファイルの中の、次のファイルを使用します。



S-rail.e3 (製品形状モデル)

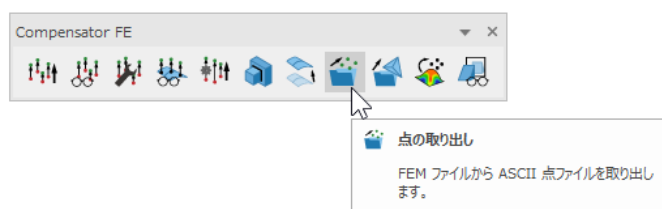
before_blank.nas (解析前の製品形状)


after_blank.nas (解析後のスプリングバック形状)

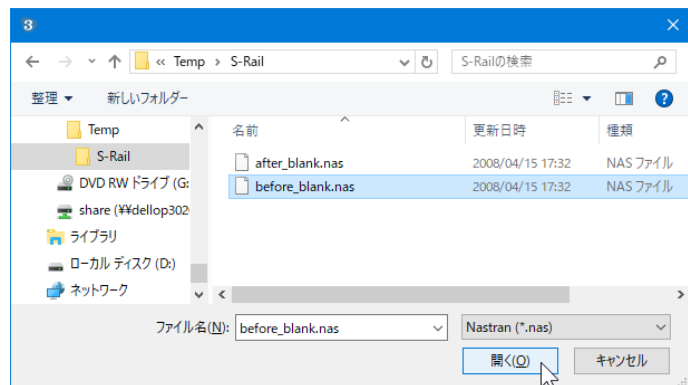
ダウンロードしたファイルから、**S-rail.e3** を開きます。




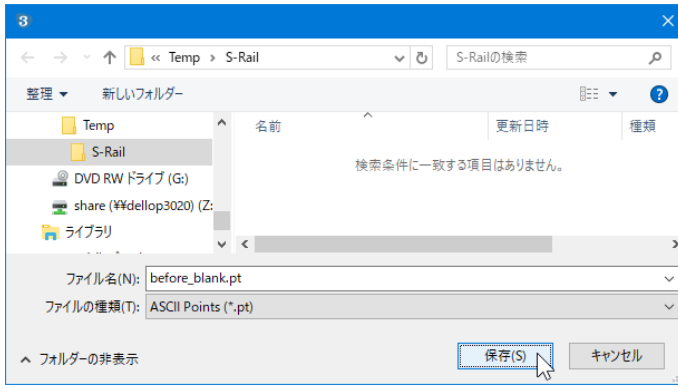
- 修正  **Compensator**  **点の取り出し** コマンドを選択します。
コマンドは、Compensator FE ツールバーから選択することができます。



-  **FEM ファイルの参照** を選択し、ファイルの種類として「Nastran (*.nas)」を選択します。
- before_blank.nas を選択し、「開く」をクリックします。



-  **適用** をクリックするとファイル名を入力するダイアログが表示されます。







- ファイル名には、*.nas ファイルの拡張子を pt に変更したものが表示されます。
- そのまま「保存」ボタンをクリックして保存します。
- 同じ手順で、after_blank.nas も pt ファイルとして保存します。

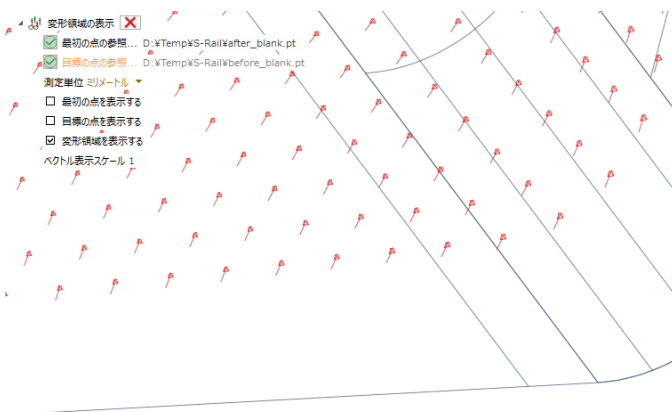
これで nastran 形式のファイルからノードの座標値を点として取り出すことができました。

この1組の点ファイルを使用して、変形量を計算します。

実際に変形量を計算する前に、変形量を画面に表示して確認することができます。





- **修正**  **Compensator**  **変形領域の表示** と選択します。
-  **最初の点の参照** で、after_blank.pt を選択します
-  **目標の点の参照** で、before_blank.pt を選択します。

両ファイルを選択すると、画面上に実際に変形する方向と量が矢印で表示されます。




- ☒ **最初の点／目標の点を表示する** にチェックすると、最初の点と目標の点も画面に表示します。
- **ベクトル表示スケール** の値を変更すると、変形量にスケールをかけて表示します。

変形の方法は、**最初の点→目標の点** の方向に表示されます。この例では、最初の点に after_blank.pt、目標の点に before_blank.pt を入力しているので、**スプリングバック形状→製品モデル形状** の方向の変形量を表示しています。

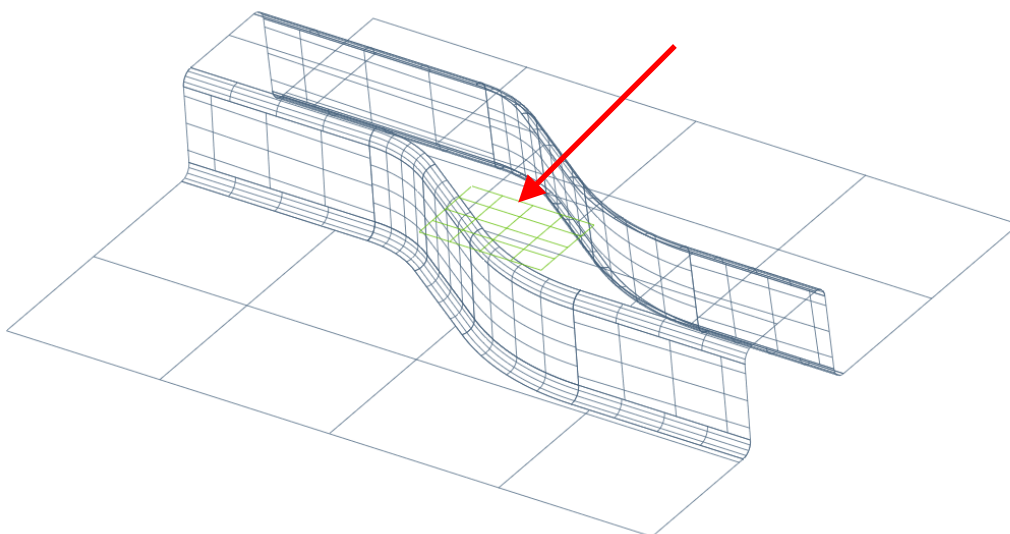
- 修正  **Compensator**  **変形領域の認識** と選択します。
-  **最初の点の参照** で、after_blank.pt を選択します
-  **目標の点の参照** で、before_blank.pt を選択します。



- その他のパラメーターはすべてデフォルトで、 **適用** をクリックします。
- 計算が実行され、下記のダイアログが表示されます。

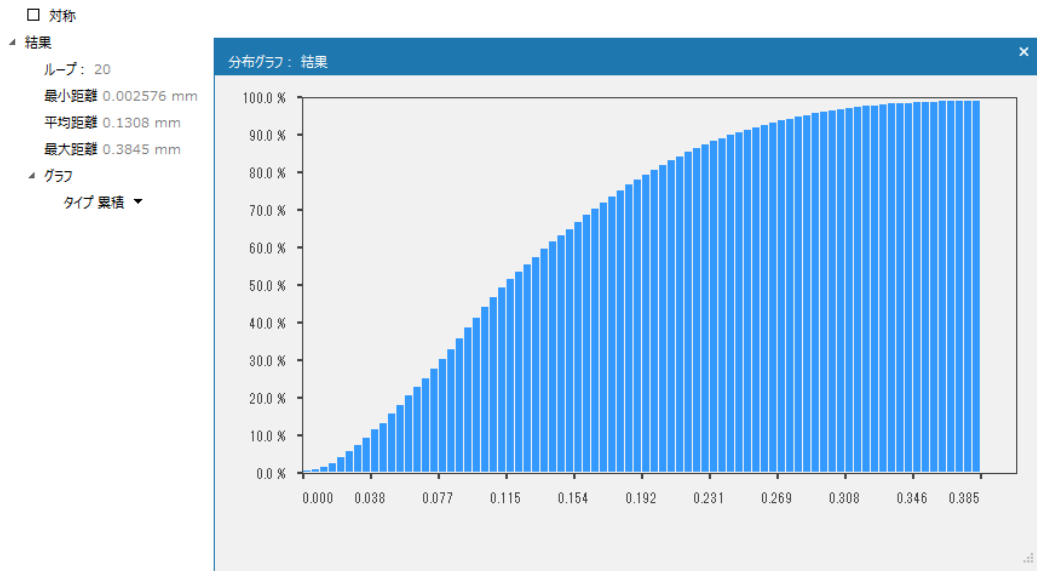


- 計算結果は、変形情報を含んだアドバンス GSM 面として作成されます。(緑色の面)



- **最大距離** オプションで指定した以上離れている点は計算に使用されません。
また、計算結果の最大値(最大距離)が **トレランス** で指定した値を下回れば計算を終了します。

- ▶ **結果** オプションを展開すると、計算結果の詳細とグラフを表示します。
- グラフは分布グラフと累積グラフに切り替えることができます。

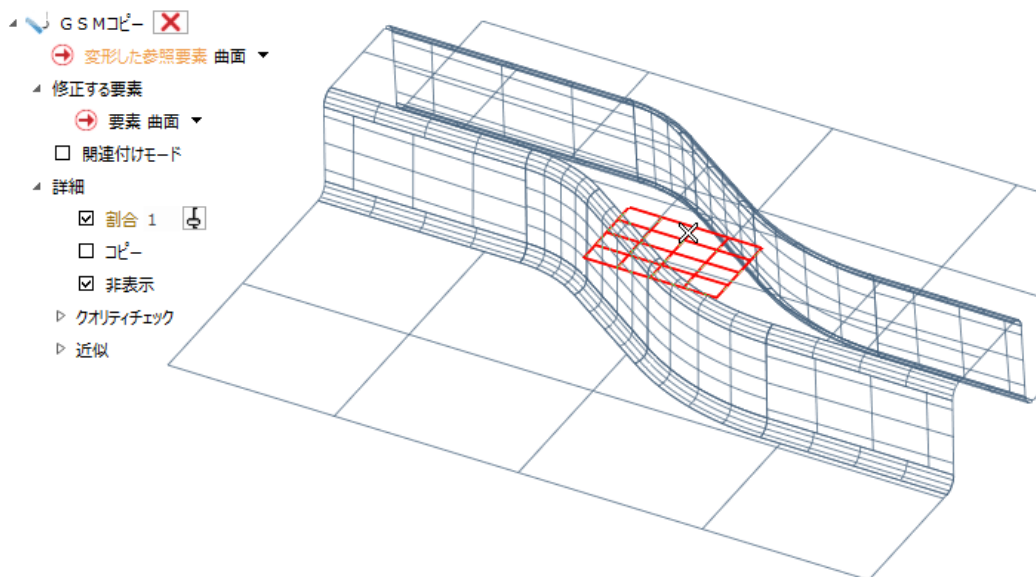




なお、コマンド実行時に **OK** を選択した場合、コマンドは変形情報を持つ面を作成してすぐに終了します。
結果のグラフ表示などを確認したい場合は、 **適用** を選択します。

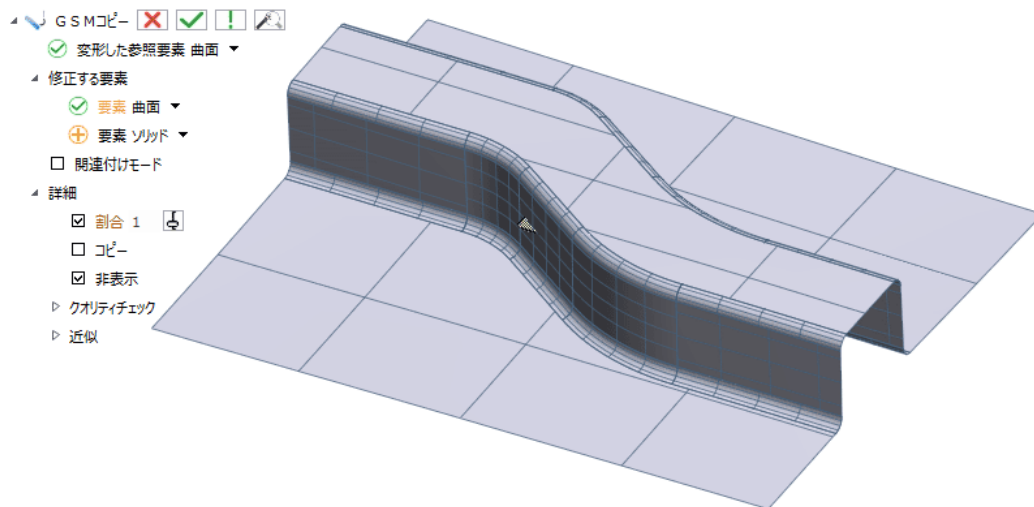
- 変形領域の認識** コマンドを終了します。


続いて、 **GSMコピー** コマンドで、緑の面の中に格納された変形情報を取り出して、他の面に適用します。

- 修正** **グローバルシェイプモデリング** **コピー** と選択します。
- 変形した参照要素** で **曲面** を選択し、変形領域の認識コマンドで作成した面(緑)を選択します。




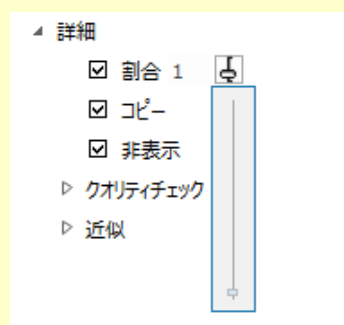
-  **要素** で **曲面** を選択し、その他の面をすべて選択します。
(一度すべての面を選択し、緑の面のみ選択を解除すると良いでしょう。)
-  **プレビュー** ボタンを押すとプレビューが表示されます。



- **詳細** オプション以下の ☒ **コピー** にチェックします。
-  **OK** をクリックしてコマンドを終了します。

割合 オプション:

 **GSMコピー** コマンドの詳細オプション以下には、**割合** オプションがあります。
このオプションでは、変形情報を適用する割合を指定します。1で 100% 適用です。



スライダーアイコンをクリックすると画面にスライダーが表示され、ドラッグすることで、適用率 0% ~ 100% までの変化の様子をリアルタイムに確認することができます。

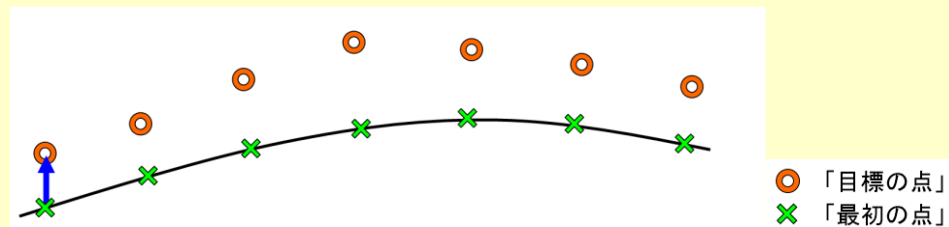
また 100% 以上、例えば、150% の割合で変形を適用したい場合は、このコマンドを2回に分けて使用し、1回目は 100%、2回目に 50% 等と適用すると良いでしょう。

トレランスとループ:

Compensator では、解析結果から取り出した点を利用して変形量を計算していますが、変形にすべての点を使用しているわけではありません。一方、結果(誤差)の評価はすべての点を使用しています。**Compensator** での変形量の計算とその評価は以下の考え方で行われています。

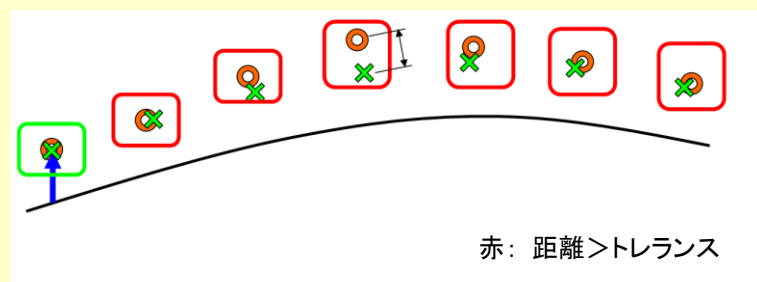
1. 最初の変形

ある点のペアを使用して変形量を計算し、すべての点をGSM変形(移動)します。



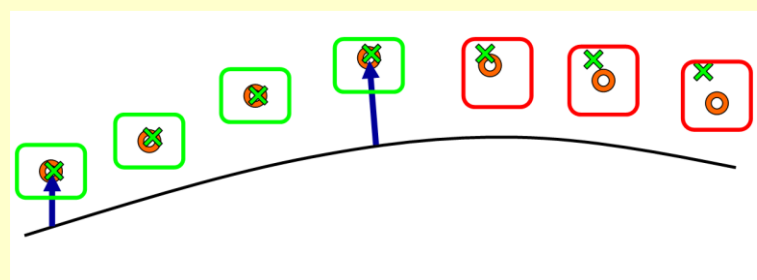
2. 誤差の評価

すべての点のペアについて距離を評価します。(ここまでの処理を「ループ」と呼んでいます。)




3. 次のループ

すべての点のペアの距離がトレランス値以下であれば終了。そうでない場合は点のペアを増やしてループを繰り返します。



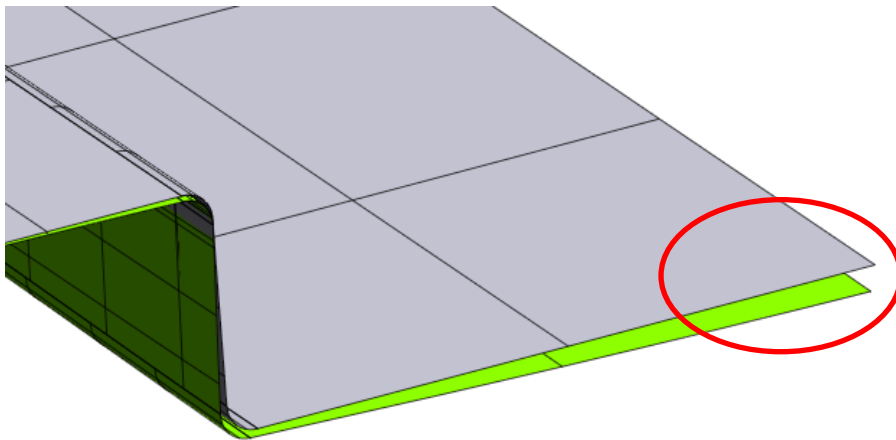
すべての点のペアの距離が目標のトレランス値以下になるようこのループを繰り返します。ただし、ループ数の上限(最大数)を指定している場合は、距離がトレランス値以下になっても上限に達した時点で終了します。


なお、ループの上限数はおよそ 300 程度を目処としてください。一般的にこれ以上ループを重ねても変形空間が荒れていくだけであまり良い結果は得られません。計算が収束しない場合は、目標としたトレランス値が小さすぎるか、計算に使用した点群が綺麗でないかの何れかです。後者の場合は、 **変形領域のクリーニング** を行うと結果が向上する可能性があります。

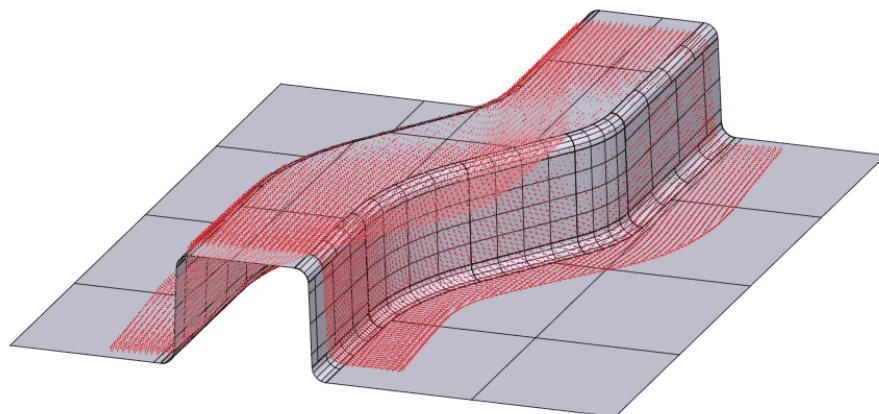
Step 4: 境界の条件と最適化


Step3 で得られた見込み形状(下図緑)は、形状周囲で元の形状(グレー)と少し離れています。


例えば下図は形状左奥の様子ですが、先端部で 2.5 mm 程度離れています。

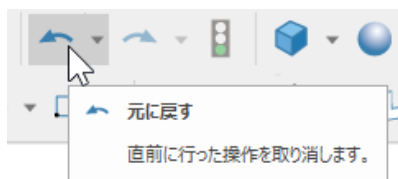







下図は  **変形領域の表示** コマンドで表示した計算に使用する点の位置です。形状の周囲が元の形状から離れるのは、赤く表示された凸部周辺の条件のみで形状全体を変形しており、形状周辺には何も条件が無いからです。



 **変形領域の認識** コマンドでは、維持する条件を設定し、動かしたくない部位を指定することができます。

- **元に戻す** コマンドで  **変形領域の認識** コマンド実行前に戻ります。
- または、**S-rail.e3** ファイルを開き直します。



- 修正  **Compensator**  変形領域の認識 と選択します。
-  **最初の点の参照** で、after_blank.pt を選択します
-  **目標の点の参照** で、before_blank.pt を選択します。
- ☒ **維持する曲線** にチェックすると、さらに下のノードが表示されます。
-  **曲線** ノードに維持したい境界線を入力します。

☑ 維持する曲線

グループ 1

✓ 曲線

拘束条件 位置を保つ ▼

位置のトレランス 0.01 mm

接線のトレランス 0.5 deg

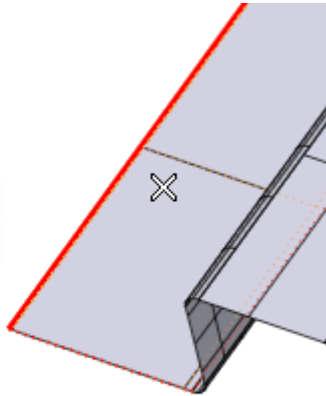
曲率のトレランス 0.05



丸み 0.1 mm

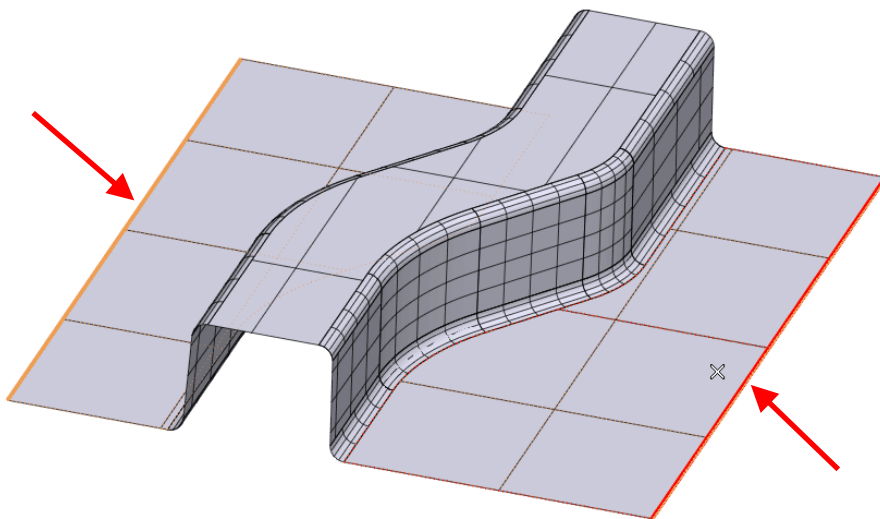
▷ 精度

☐ 対称

▷ 結果



- 拘束条件 は 位置を保つ (G0)、位置を保つ+接線を保つ (G1)、位置を保つ+接線を保つ+曲率を保つ (G2) の3種類を選択できますが、今回は 位置を保つ+接線を保つ を選択します。
- 曲線は  **曲線** ノードに複数入力できるほか、 ボタンで複数のグループに分けて入力することもできます。
- ここでは、下図のように、左右端の境界線を選択します。

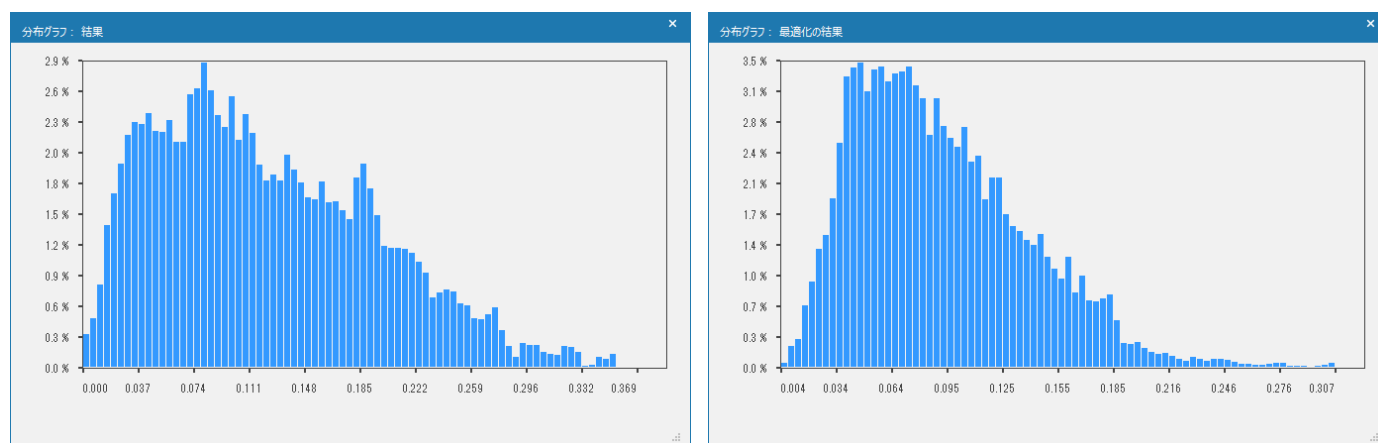


- また、☒ **最適化** にもチェックしてみます。
- ☒ **適用** をクリックすると計算が実行されます。

このデータの場合、以下のような結果が表示されます。




- ▶ **結果、最適化の結果** ノードを展開すると、最適化前と後のグラフを表示することができます。
(下図 左: 最適化前、右: 最適化後。グラフは両者を同時に表示することができます。)

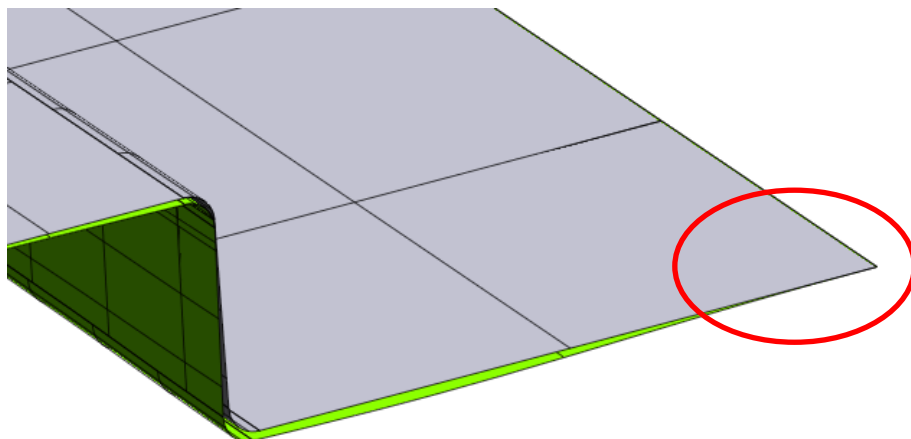


最適化前後で値を比較すると、平均距離、最大距離が小さくなっていますが、最小距離が微妙に大きくなっています。またグラフから、全体的に山のピークが少し鋭くなっていることがわかります。

通常、最適化処理では、形状がより滑らかになるよう微調整します。その結果、平均距離が小さくなり、そのピークが鋭くなる傾向となりますが、最小距離が少し増えることがあります。また、多くの場合最大距離も小さくなりますが、必ず小さくなるわけでもありません。

先ほどと同じ手順で、 **GSMコピー** コマンドで計算された変形情報他の面に適用します。

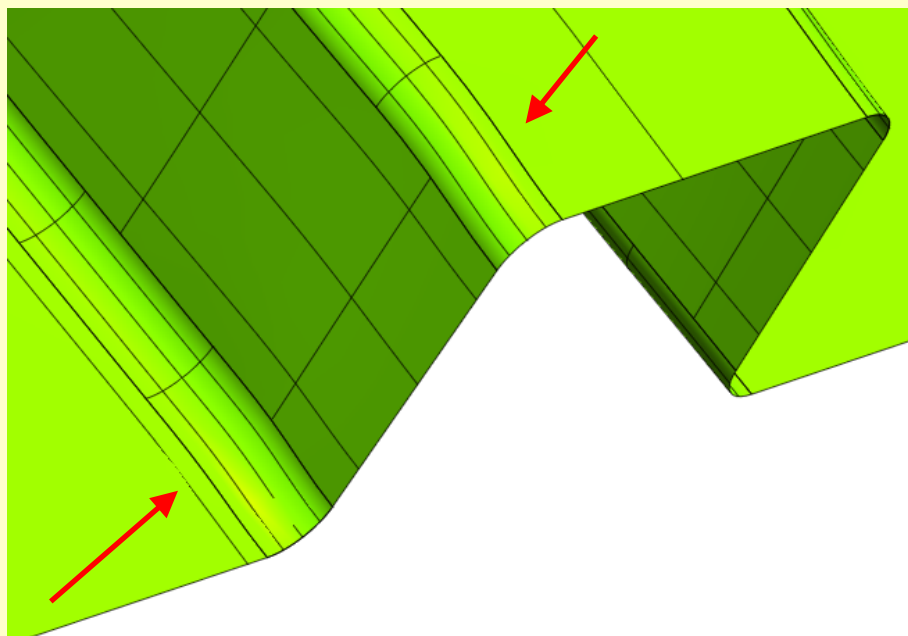
このステップの始めに示した位置を表示すると次のようになり、条件を追加した周辺部は元の形状と一致している様子がわかります。




拘束する位置:


このステップの例では、計算に使用する点から十分に離れた左右の境界線の位置と接線を拘束しました。

計算に使用する点からあまり離れていない場所を拘束すると、維持する場所と変形(形状が移動)する場所との繋がりがスムーズにならないことがあります。下図は同じデータで、上下の境界線を拘束した場合です。この例では変形する領域から維持する領域へ変わる部分の形状変化が急で、形状が少し折れているように見えます。

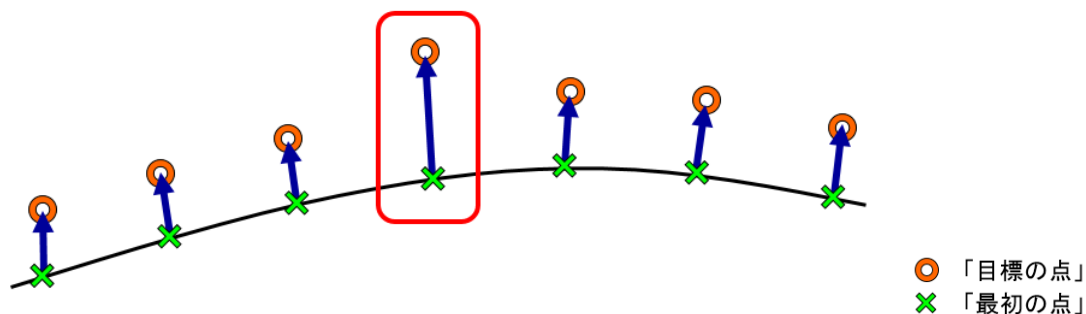


Step 5: 変形領域のクリーニングと対称拘束

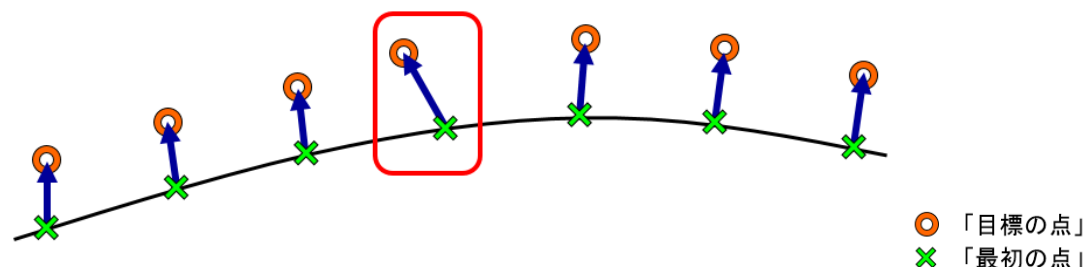
変形に使用する点群が荒れていると、何度ループ計算を繰り返しても目標値に近づかなかったり、出力された結果が波打っていたり、なかなか良い結果が得られないことがあります。そのような場合、 **変形領域のクリーニング** コマンドで荒れている部分を削除すると結果が向上する可能性があります。

 **変形領域のクリーニング** コマンドでは、ある一定領域内の点の移動量、または移動角度が、その領域内の平均値から外れた点のペアを検出して削除します。

下図はある特定領域の点の移動量について、各移動量の平均から離れた点のペアを示しています。



下図は同様に、点の移動角度について、平均から離れた点のペアを示しています。



このような点のペアを検出し、検出されたペアを最初の点、目標の点、それぞれの点群から削除することができます。

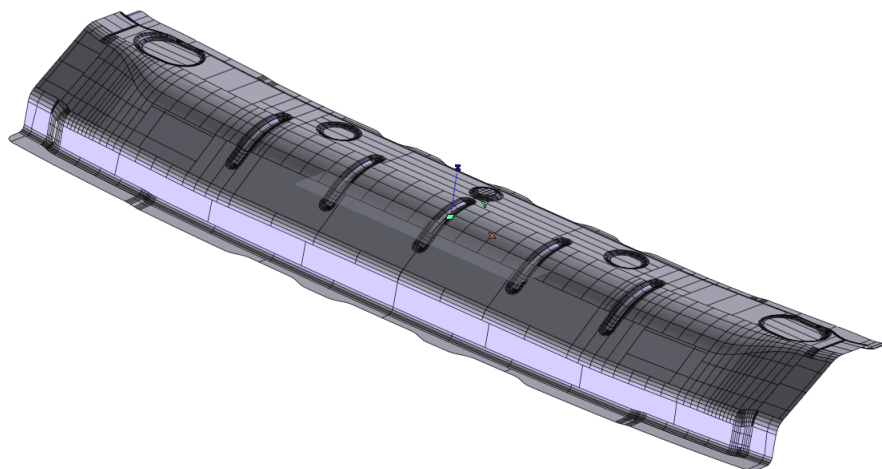
一度点を削除すると領域内の平均値が変わるため、何度かループ計算を行い、念入りに削除すべき点を調べるできるようになっています。

なお、もともとあまり荒れていない点群に対して、このコマンドを実行して点を削除してもあまり効果はありません。

ダウンロードしたファイルから、Rail_Mirror.e3 を開きます。

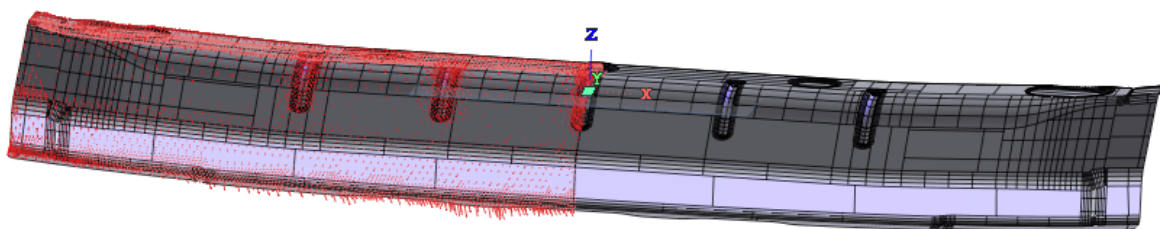
このファイルを、Rail_Mirror_after.pt、Rail_Mirror_before.pt の点群を使用して変形します。

はじめに、変形領域(点群)の位置を確認します。








- 修正 Compensator 変形領域の表示 コマンドを選択します
- 最初の点の参照 に Rail_Mirror_after.pt を選択します。
- 目標の点の参照 に Rail_Mirror_before.pt を選択します。
- ベクトル表示スケール に 5 程度を入力して、表示を大きくするとわかりやすいでしょう。

すると下図のように表示され、形状の左半分だけに点群があることがわかります。この形状は左右対称なので、YZ 平面で対称な変形条件を設定することになります。




変形領域の表示 コマンドを終了します。

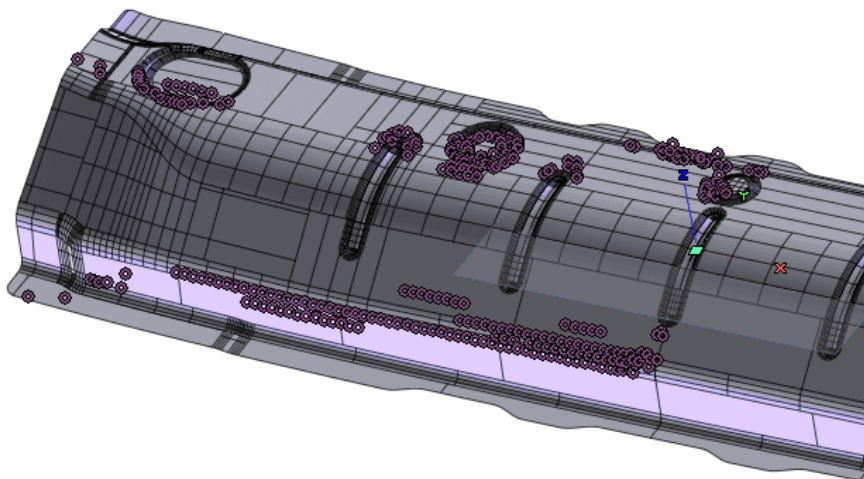
- 修正  **Compensator**  変形領域のクリーニング コマンドを選択します
-  最初の点の参照 に Rail_Mirror_after.pt を選択します。
-  目標の点の参照 に Rail_Mirror_before.pt を選択します。
- 隣 ノードと ▸ 矛盾点を表示する ノードを展開します。
-  プレビュー ボタンを押すと ▲ 範囲 で指定された領域内で点群間の計算が行われ、最小距離、平均距離などの結果が表示されます。




削除する点の基準は、▸ 矛盾点を表示する ノードで指定します。

計算された結果から、この例では、角度を基準に「矛盾点」を削除することになります。

- ☒ 矛盾点を表示する にチェックし、▲ 矛盾のない偏差の基準 に 角度 を指定します。
- しきい値 に 10 deg と入力します。
- 再度  プレビュー ボタンを押すと該当する点の位置が画面に表示されます。



- **矛盾点のクリーニング** で **最初の点と目標の点を削除する** を選択します。
- **ループ数** に **10** と入力します。
-  **適用** をクリックすると計算が実行されます。

矛盾点のクリーニング 最初の点と目標の点を削除する ▼

ループ数 10

☐ 削除する点を選択する

▷ 詳細

計算結果は、入力した pt ファイルと同じフォルダーに作成されます。

作成されるファイルは次の基準で名前が付けられます。

元のファイル名_bad_loop_ループ回数.pt

元のファイル名_clean_loop_ループ回数.pt

今回の例では、

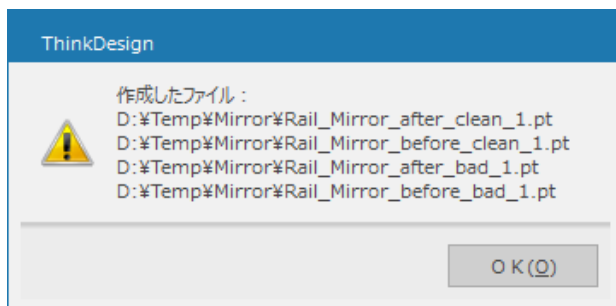
Rail_Mirror_after_bad_loop_1.pt と **Rail_Mirror_after_clean_loop_1.pt**、

Rail_Mirror_before_bad_loop_1.pt と **Rail_Mirror_before_clean_loop_1.pt**

のようなファイルが作成されます。

ファイル名に bad と追加されるものが削除された点群で、clean と追加されるものが矛盾点を削除済みの点群です。

ループ数を 1 と指定した場合は下のようなダイアログが表示されますが、2 以上を指定した場合は、ダイアログは表示されず、自動的にループを繰り返し、検出された点を削除していきます。(ループ 1 の場合、ファイル名に loop は入りません。)



ループを繰り返す中で削除すべき点が無くなると、その時点で処理を終了します。

今回のデータでは、6回目のループで処理する点が無くなったため、**_bad_loop_6.pt** ファイルは、0 kb となっています。









したがって、ここでは点の処理が完了した次の2つのファイルを使用します。

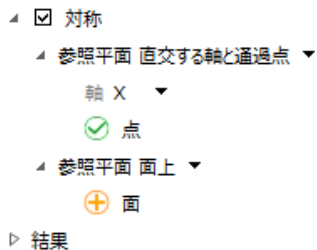
(計算途中の出力ファイルは削除してもかまいません。)


Rail_Mirror_after_clean_loop_6.pt

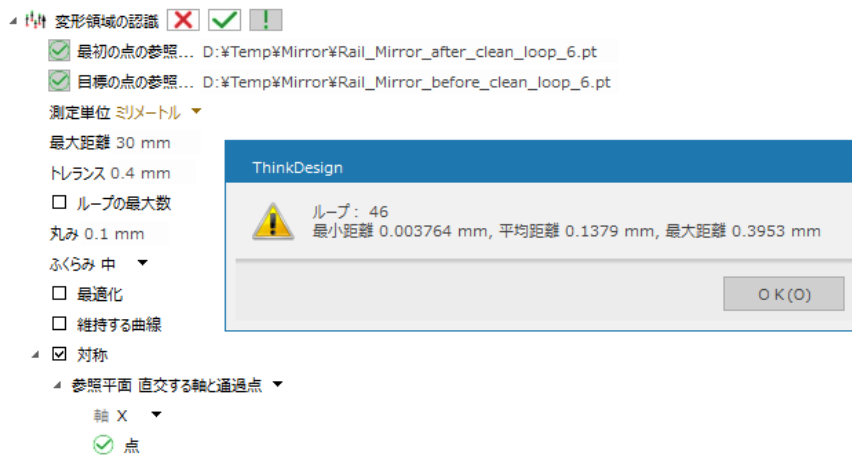
Rail_Mirror_before_clean_loop_6.pt


修正された点群を用いて計算を行います。

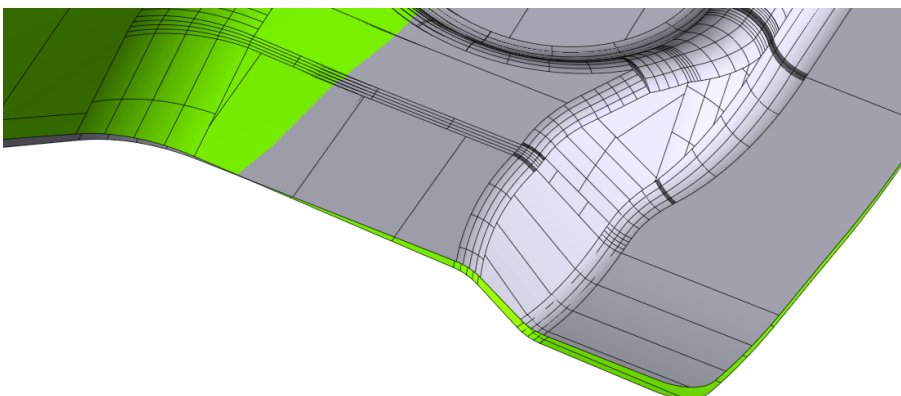
- 修正  **Compensator**  **変形領域の認識** と選択します。
-  **最初の点の参照** で、Rail_Mirror_after_clean_loop_6.pt を選択します
-  **目標の点の参照** で、Rail_Mirror_before_clean_loop_6.pt を選択します。
- ☒ **対称** にチェックし、 **参照平面** で **直交する軸と通過点** を選択します。
- **軸** で X を選択し、 **点** に座標系の原点を入力します。(ツール  **スナップ**  **原点** を使うと簡単です。)



- その他のパラメーターはデフォルトのままで、 **適用** をクリックします。



-  **GSMコピー** コマンドで変形を適用します。
- ☒ **コピー** オプションを使用すると、元のデータを残して結果をコピーとして作成します。



対称オプションを使用したので、変形領域のない右側の形状も左側と同様に変形されていることを確認してください。

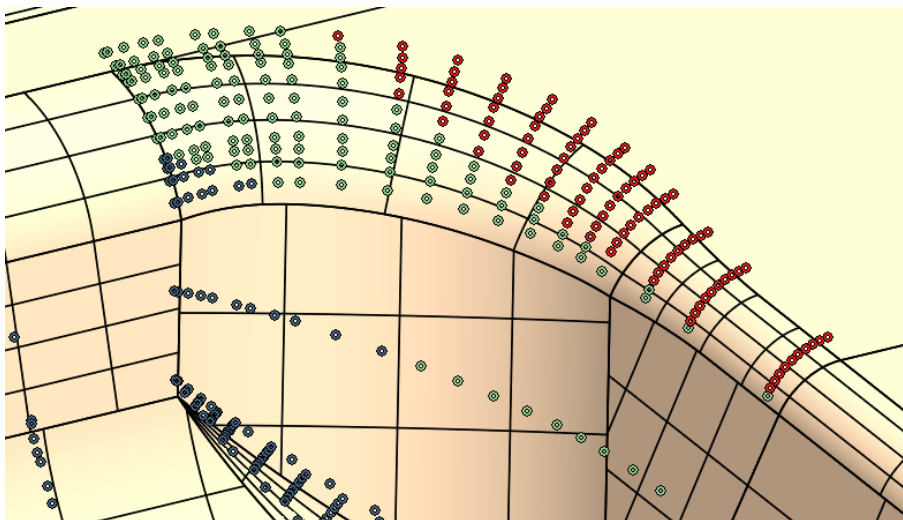
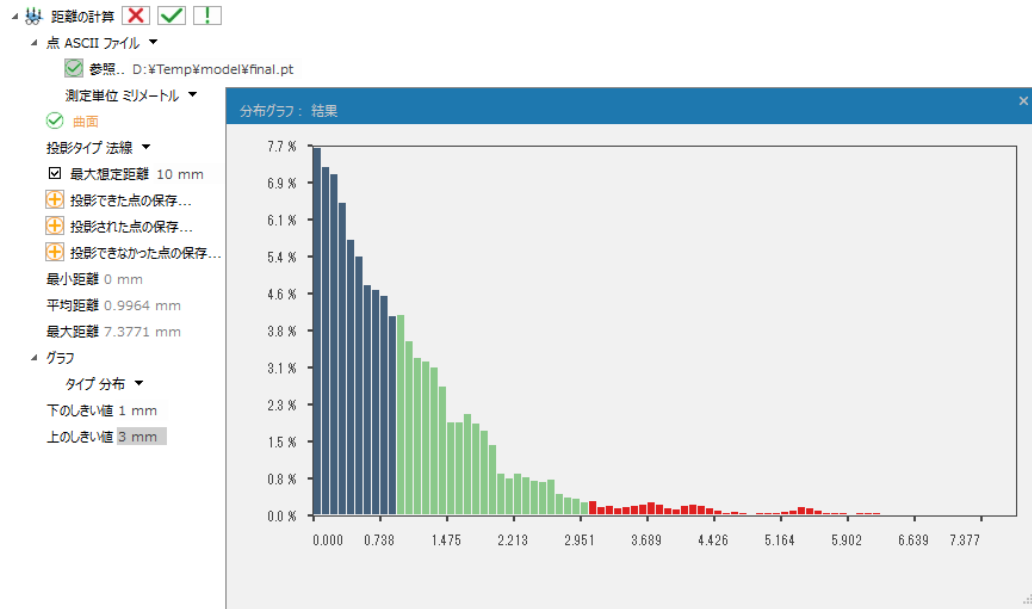
Step 6: その他のコマンド

次に、Compensator で利用できるその他のコマンドをいくつか簡単にご紹介します。

距離の計算 コマンド

点群と曲面との距離を測定し、画面上に黒、緑、赤で色分けして表示します。点群と曲面との関係を視覚的に確認することができます。(各色のしきい値は変更することができます。)

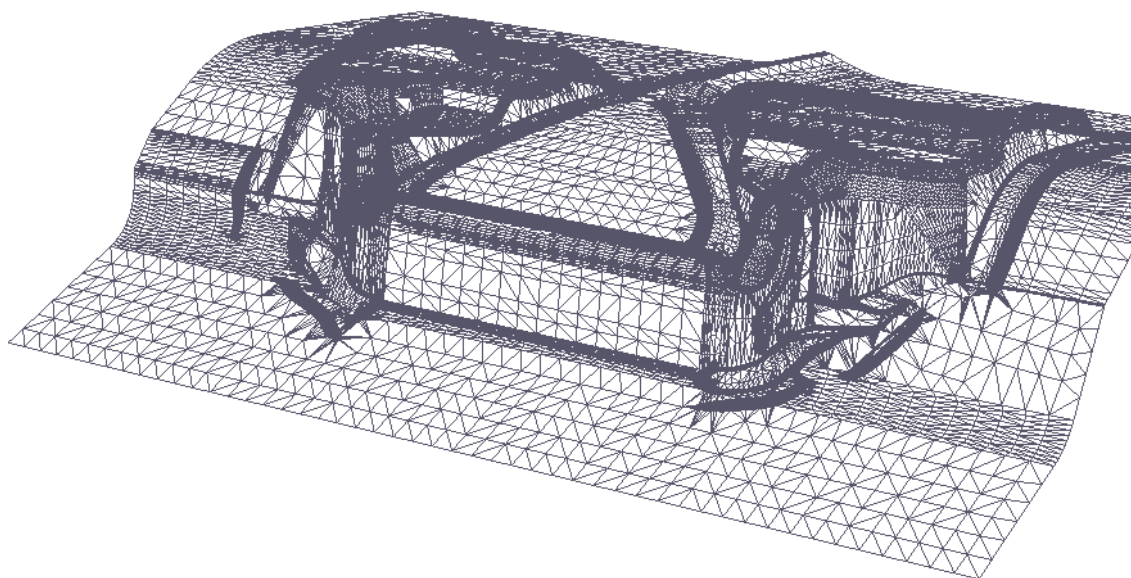
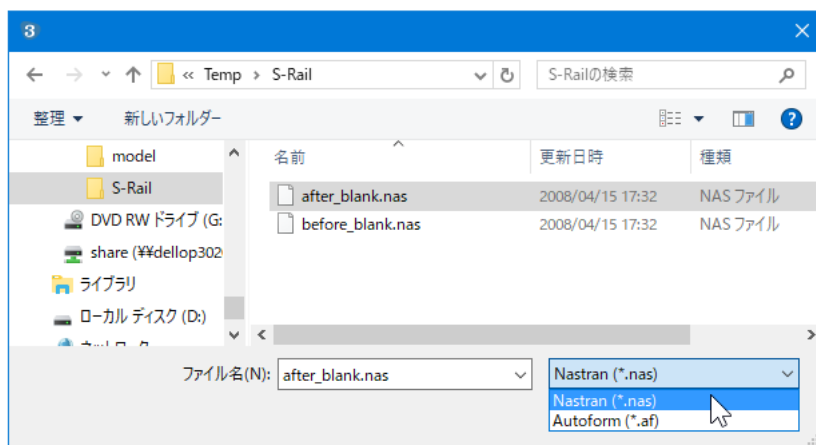
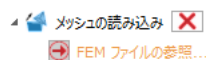
また、面に投影できた点やできなかった点などを保存することもできます。



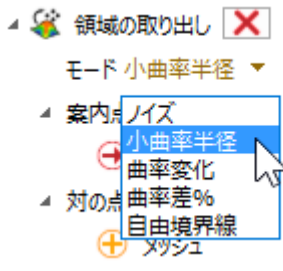
メッシュの読み込み コマンド

解析ソフトのファイルを直接メッシュとして読み込みます。

Nastran(*.nas)、Autoform(*.af) フォーマットに対応しています。



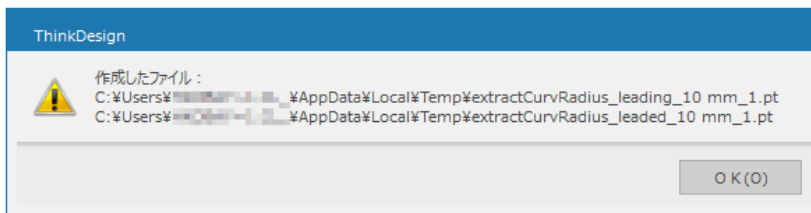
点データ(ASCII 点ファイル)や、メッシュなどからさまざまな条件に該当する点を取り出し、ファイルとして保存します。
条件には、**ノイズ**、**小曲率半径**、**曲率変化**、**曲率差%**、**自由境界線** を選択することができます。



対象のメッシュなどは「案内点」「対の点」としてペアで入力することができます。

処理は「案内点」を用いて行いますが、処理された「案内点」とペアとなる「対の点」上の点も出力され、Compensator での計算にそのまま用いることができます。

処理結果は、システムのテンポラリーフォルダー（デフォルトでは、C:\Users\ユーザー名\AppData\Local\Temp）に拡張子 pt のファイルで保存されます。ファイル名に leading と入るのが案内点、leaded と入るのが、対の点から出力されたファイルを表します。



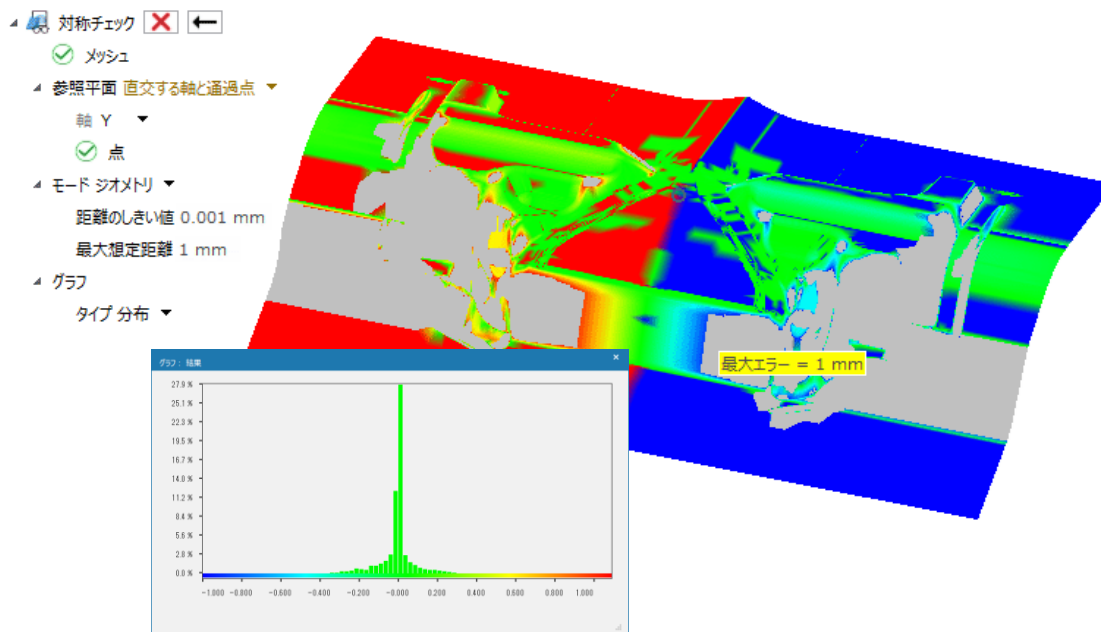
上の例は小曲率半径モードで、しきい値 10 mm を指定して処理したもので、以下のファイルが作成されています。

extractCurvRadius_leading_10 mm_1.pt

extractCurvRadius_leaded_10 mm_1.pt

対称性チェック コマンド

メッシュの対称性をチェックします。**変形領域の認識** コマンドで対称オプションを使用可能かどうか判断するためなどに使用することができます。



動作モードには「トポロジー」と「ジオメトリ」の2種類があります。

「トポロジー」モードでは、調べる形状は対称平面に対して、ノードの数や形、その位置までが対称であるかどうかを調べます。

「ジオメトリ」モードでは、ノードの構成は考慮せず、単純に形状の差違を調べます。

