

使用するソフトとバージョン

- ・ SALOME 7.6.0
- ・ FRAXST for Peridigm (ver3.00_170110)
- ・ ParaView 4.4.0 64-bit

始めに

FRAXST では多くのファイルを用いるため、作業用フォルダーを作成する。今回は Desktop に空のフォルダーを作成し、「FRAXST_test」と名前を付ける。空のフォルダーは、Desktop 上で右クリックし、出てきたメニューから「新しいフォルダー(F)」をクリックすることで作成できる。

1. Salome によるモデリング

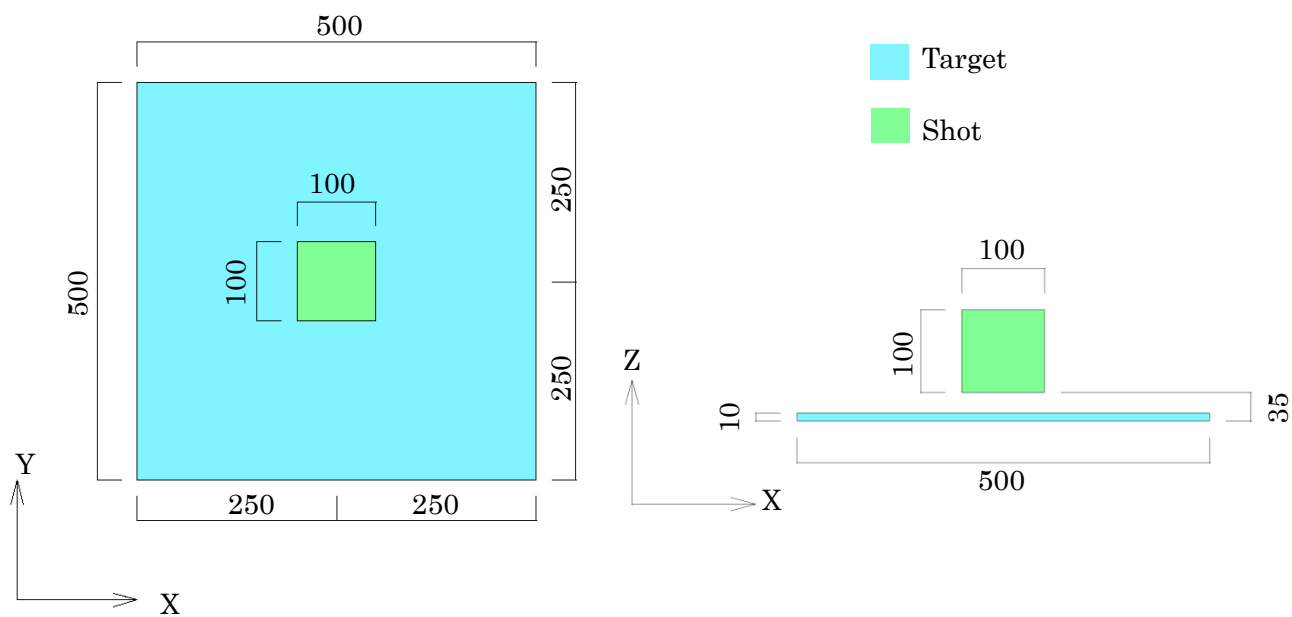


図 1 解析を行うモデル (mm)

1.1.モデリング

Launcher にある「SALOME」をクリックして Salome を起動したら、「SALOME」プルダウンメニューを「Geometry」に切り替え、新規作成する。ツールバーにある「ボックスを作成」をクリックするとウインドウ（図 2）が表示される。表示されたウインドウに表 1 のように名前と値を入力する。図 2.1 のように「Target」に対する寸法を入力したら、「適用(A)」をクリックする。次は、「Target」を「Shot」に書き換え、寸法を入力しなおす。図 2.2 のように「Shot」に対する寸法を入力したら、「適用して閉じる(p)」をクリックする。

表 1 モデリングを行う寸法 (m)

| 名前 | Target | Shot |
|----|--------|------|
| Dx | 0.5 | 0.1 |
| Dy | 0.5 | 0.1 |
| Dz | 0.01 | 0.1 |



図 2.1 Target のモデリング



図 2.2 Shot のモデリング

図 2 ボックスによるモデリング

Shot オブジェクトを初期の位置へ移動させる。ツールバーにある「移動」をクリックするとウインドウ（図 3）が表示される。オブジェクトは「Shot」をオブジェクトブラウザーから選択する。「Dx」「Dy」「Dz」に表 2 の値を入力し、「コピーを作成」のチェックを外したら、「適用して閉じる(p)」をクリックする。

表 2 「Shot」の移動方向と数値 (m)

| オブジェクト | Shot |
|--------|-------|
| Dx | 0.2 |
| Dy | 0.2 |
| Dz | 0.035 |

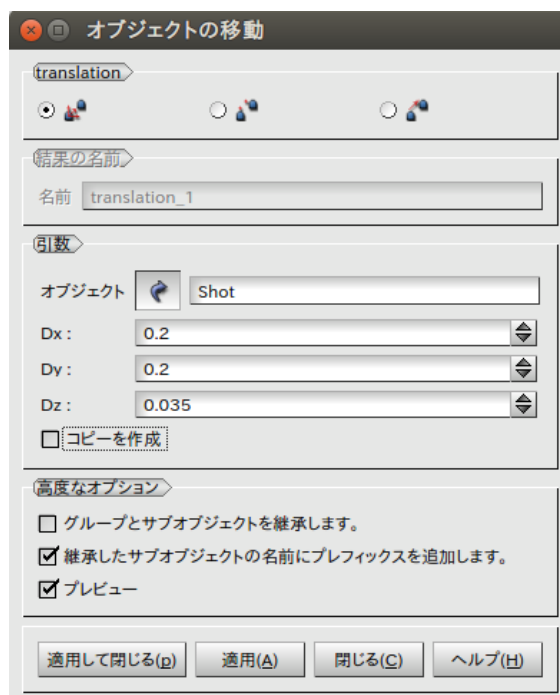



図 3 Shot オブジェクトの移動

1.2.境界条件の設定

1.2.1. 「Target」に対する条件の設定

オブジェクトブラウザーにある「Target」を右クリックし、「グループを作成」をクリックする。

「オブジェクトの種類」を「 (面)」に設定し、グループの名前を「Fix」とする。「Target」は側面の4辺を完全固定するため、「Target オブジェクト」の側面を選択して「追加」をクリックする。そうすることで空のウインドウに面の番号が追加される。また、Shift+クリックすることで4辺まとめて選択できる。正しく選択されると図 4.1 のように表示されるので、「適用して閉じる(p)」をクリックする。

ウインドウが閉じると、図 4.2 のように「Fix」オブジェクトである4辺と、「Shot」オブジェクトが表示される。正しく表示されない場合は、図 4.1 の選択がうまくいっていないので、ウインドウに4つの数字があるかを確認する。誤って拘束されない面を選択した場合は、その面を選択して「削除」をクリックする。

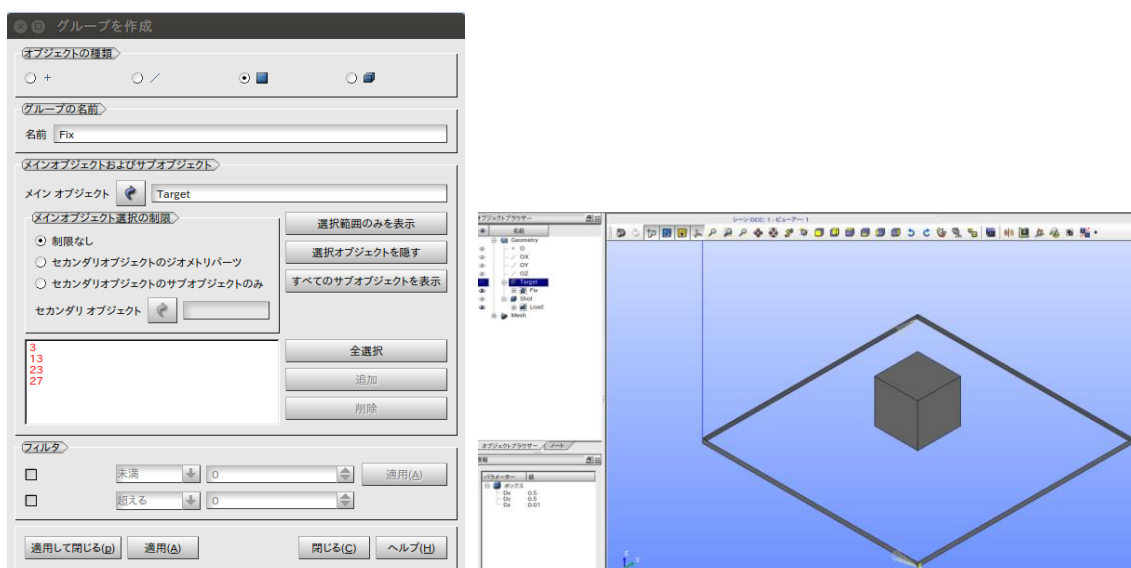



図 4.1 Fix グループを作成 図 4.2 正しく選択された「Fix」オブジェクト
図 4 「Target オブジェクト」に対する「Fix オブジェクト」の設定

1.2.2. 「Shot」に対する条件の設定

「Target」についてのグループを作成したら、次は「Shot」に対してグループを作成する。基本的には同じ手順であるため、オブジェクトブラウザーにある「Shot」を右クリックし、「グループを作成」をクリックする。

「オブジェクトの種類」を「 (立体)」に設定し、グループの名前を「Load」とする。「Shot」はオブジェクト全体をZ方向に運動させるので、「全選択」をクリックする。立体は一つしかないなので、ウインドウには「1」とだけ表示される。正しく選択されると図 5.1

のように表示されるので、「適用して閉じる(p)」をクリックする。

オブジェクトとグループが同じであるためウインドウが閉じると図 5.2 のように「Load」オブジェクトと先ほど作成した「Fix」オブジェクトが表示される。

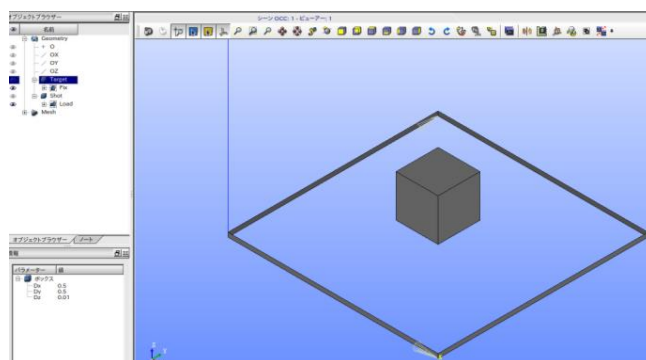
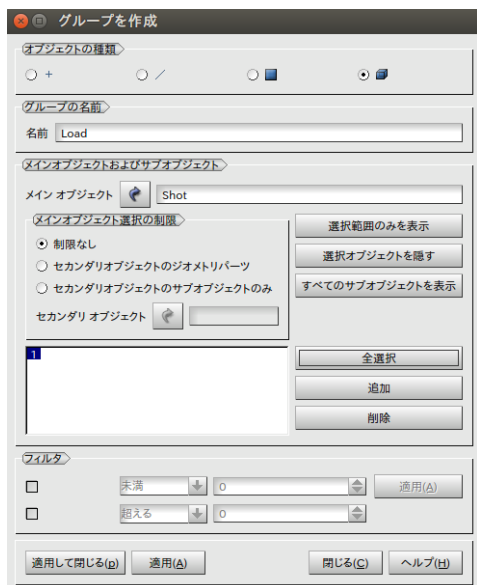


図 5.1 Load グループの作成 図 5.2 正しく選択された「Load」オブジェクト
 図 5 「Shot オブジェクト」に対する「Load オブジェクト」の設定

1.1 と 1.2 の作業が終了すると、図 6 のようにオブジェクトブラウザーが表示される。ツリーの表示が隠れている場合は各項目の先頭にある をクリックすることで、下の項目が表示される。また、 をクリックすることでモデルの表示、非表示を切り替えることが出来る。

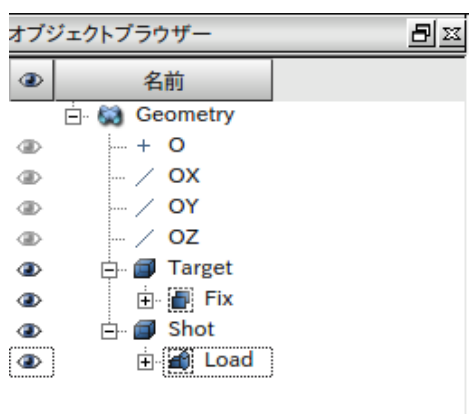



図 6 オブジェクトとグループが作成されたオブジェクトブラウザー

1.3. オブジェクトをメッシュ化

「Geometry」プルダウンメニューを「Mesh」に切り替える。「表示(V)」→「ツールバー(O)」にある「メッシュツールバー」「グループツールバー」にチェックがあるか確認をして、ない場合には項目をクリックしてチェックをつける。

1.3.1. 「Target」のメッシュ化

オブジェクトブラウザーにある「Target」を選択し、ツールバーにある「メッシュを作成」をクリックする。「メッシュを作成」というウインドウが表示されるので、名前を「Target_M」とする。ウインドウ内にある「詳細設定セットの割り当て」をクリックし表示されるメニューから、「3D : Automatic Hexahedralization」をクリックする。「仮定の構成」ウインドウが表示されるが、ここでは設定しないので、「キャンセル(C)」をクリックする。「メッシュを作成」ウインドウの「1D」タブを開いて、「詳細設定」の横にあるをクリックする。設定方法のメニューから、「Local Length」をクリックすると「仮定の構成」ウインドウが表示される。図 7.1 のように「長さ」を「Target」の板厚と同じ 0.01 と入力して、「OK」をクリックし、「適用して閉じる(p)」をクリックする。

「メッシュを作成」ウインドウを閉じたら、オブジェクトブラウザー内の「Target_M」を選択してツールバーにある「メッシュを作成」をクリックする。クリックすると計算が開始され、計算が成功すると、図 7.2 のように計算結果が表示される。メッシュが正しく表示されない場合はエラーログが表示されるので、1.3.1.の作業をもう一度行う。

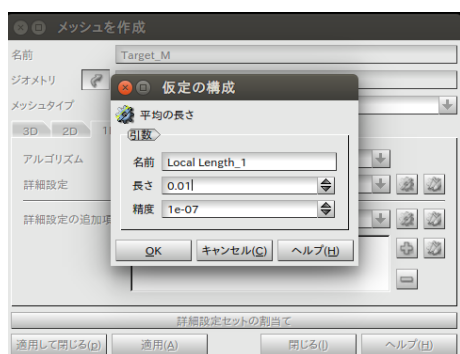


図 7.1 メッシュ間隔の平均長さの入力

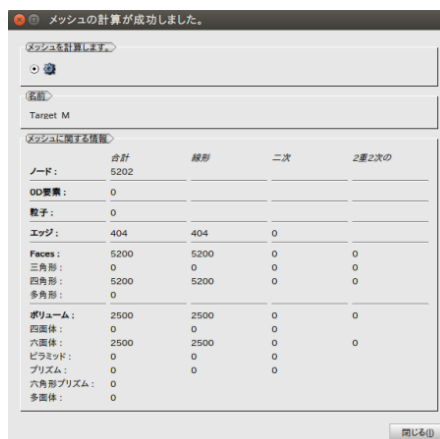


図 7.2 「Target_M」の計算結果

図 7 「Target」のメッシュ化

計算が終了すると、「Target」に対して $0.01 \times 0.01 \times 0.01$ (m)間隔で立方体メッシュが切られるため、縦 50×横 50×厚み 1 だけ要素が並び、2500 要素となる。

1.3.2. 「Shot」のメッシュ化

オブジェクトブラウザーにある「Shot」を選択し、ツールバーにある「メッシュを作成」

をクリックする。「メッシュを作成」というウインドウが表示されるので、名前を「Shot_M」とする。ウインドウ内にある「詳細設定セットの割り当て」をクリックし表示されるメニューから、「3D : Automatic Hexahedralization」をクリックする。「Shot」は立方体であるため、セグメントの数を設定するだけで立方体のメッシュが切られる。「仮定の構成」ウインドウが表示されたら、図 8.1 のように「セグメントの数」を 10 と入力し、「OK」をクリックし、「適用して閉じる(p)」をクリックする。

「メッシュを作成」ウインドウが閉じたら、オブジェクトブラウザー内の「Shot_M」を選択してツールバーにある「メッシュを作成」をクリックする。クリックすると計算が開始され、計算が成功すると、図 8.2 のように計算結果が表示される。メッシュが正しく表示されない場合はエラーログが表示されるので、1.3.2.の作業をもう一度行う。



図 8.1 「Shot」に対するセグメント数

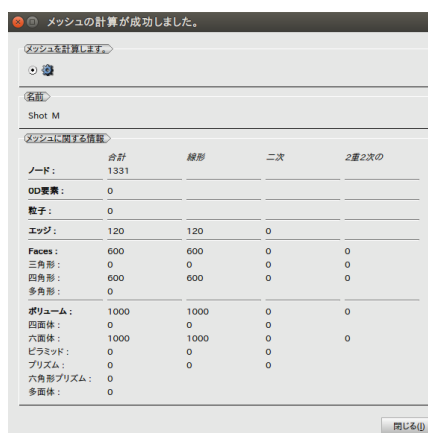


図 8.2 「Shot_M」の計算結果

図 8 「Shot」のメッシュ化

計算が終了すると図 9 のように表示される。「Shot_M」も「Target_M」と同じように $0.01 \times 0.01 \times 0.01$ (m) 間隔で立方体メッシュが切られるため、縦 $10 \times$ 横 $10 \times$ 厚み 10 だけ要素が並び、1000 要素となる。オブジェクトに対するメッシュ間隔は、FRAXST の設定に関係するので記憶しておくといよい。

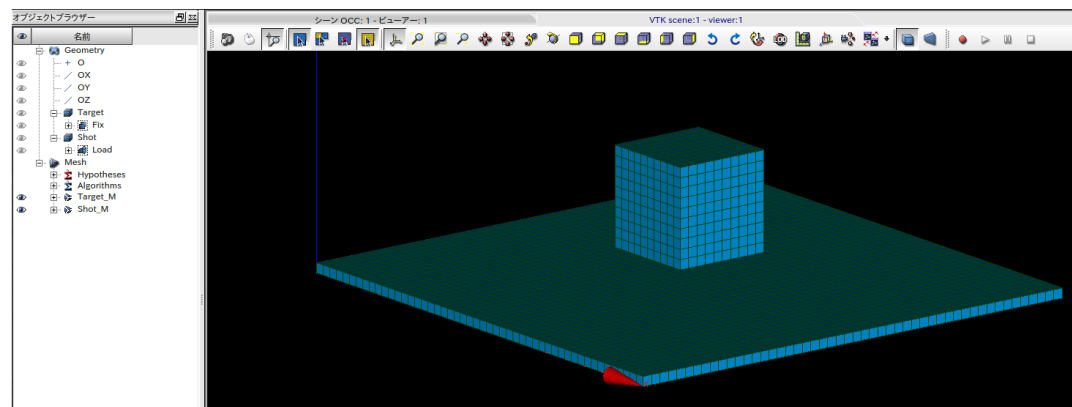


図 9 メッシュが切られた「Target_M」と「Shot_M」

1.4. メッシュに対するグループの作成

1.3.で作成したメッシュに対して、グループの「要素」と「ノード」を設定する。「要素」はメッシュを作成する際に元となったオブジェクトを設定して、「ノード」はオブジェクトに対する境界条件を設定する。

「Target_M」に対して設定をする方法は、オブジェクトブラウザ内の「Target_M」を選択した状態で、ツールバーにある「ジオメトリに関連するグループを作成します」をクリックする。「ジオメトリからグループを作成」(図 10) ウィンドウが表示されたら、図 10.1 のようにジオメトリを設定する。ウィンドウが表示されたら、オブジェクトブラウザ内の「Target」を選択することで、「要素」のジオメトリに「Target」が追加される。次に「ノード」にある「」をクリックする。そのままでは「Target」が入っているので、オブジェクトブラウザ内の「Fix」(見えない場合は 1.2.の下段を参考)を選択すると、「Target」が「Fix」に変更されるので、「適用して閉じる(p)」をクリックする。

「Shot_M」に対しても同様の手順で設定する。「Shot_M」を選択して、ツールバーにある「ジオメトリに関連するグループを作成します」をクリックして、図 10.2 のように「ジオメトリ」には「Shot」、「ノード」には「Load」を設定する。



図 10.1 「Target_M」

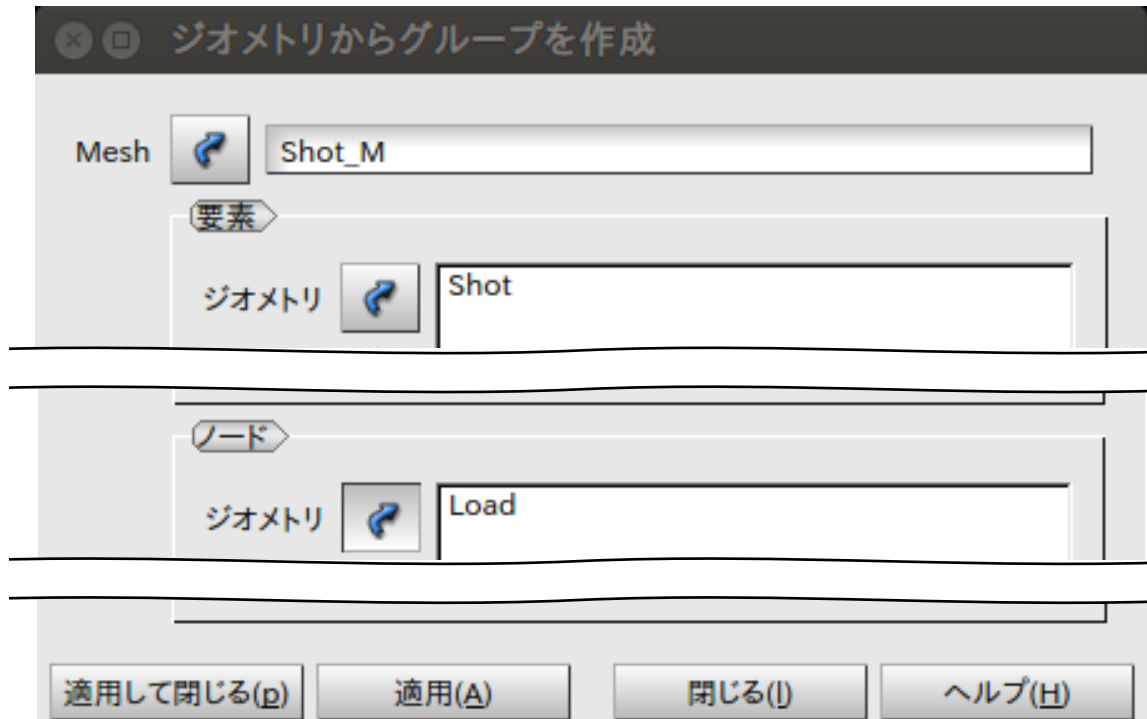


図 10.2 「Shot_M」

図 10 ジオメトリからグループの作成

1.5.メッシュファイルの結合

今回の例題のように物体が 2 以上ある場合は、コンパウンドを作成する必要がある。図 11.1 のようにコンパウンドしたいメッシュを全て選択 (Ctrl+クリック) したら、ツールバーにある「コンパウンドの作成」をクリックする。「コンパウンドの作成」ウインドウが表示されるので「初期メッシュの共通グループを作成します。」にチェックをつけて、図 11.2 のように設定されていることを確認して、「適用して閉じる(p)」をクリックします。「引数」内のメッシュが「2_objects」となっていない場合はメッシュが 1 つしか選択されていない状態なので、選択をし直して、1.5.の作業をもう一度行う。

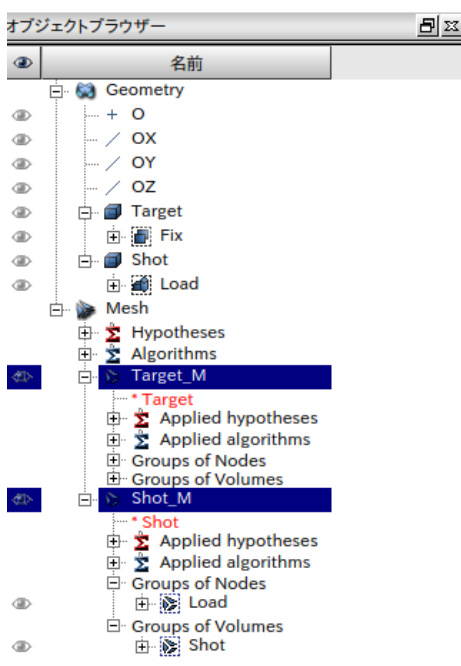


図 11.1 コンパウンドするファイル

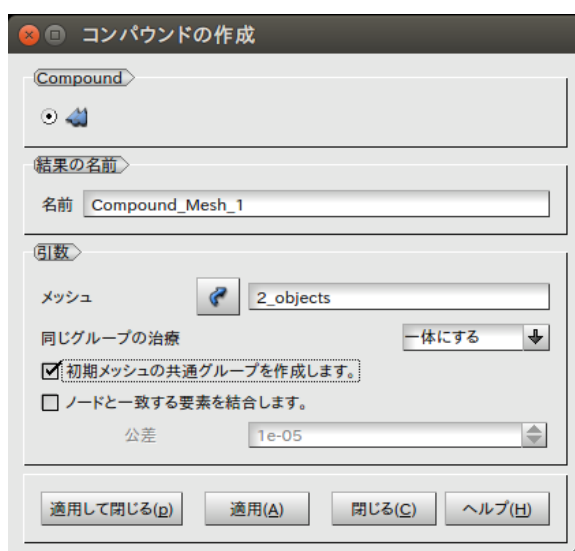


図 11.2 コンパウンドの作成

図 11 メッシュファイルの結合

コンパウンドが終了すると、オブジェクトブラウザー内に「Compound_Mesh_1」が作成される。ツリーを展開すると、図 12 のように表示される。

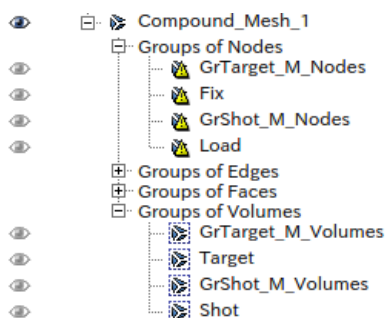


図 12 「Compound_Mesh_1」内のツリーデータ

1.6. 解析に用いるファイルの出力

FRAXST では「Groups of Nodes」と「Groups of Volumes」内にあるファイルを用いるため、ファイルとしてフォルダーに出力する。

1.6.1. 境界条件の出力

「Groups of Nodes」からは境界条件を抽出するため、「Fix」と「Load」を DAT ファイルとして出力する。「Gr～」と名前がついているものは解析で用いないので無視する。「Fix」を出力する場合は、「Groups of Nodes」内にある「Fix」を右クリックして、「エクスポート」→「DAT ファイル」をクリックする。「FRAXST_test」まで移動して「保存(S)」をクリックする。「Load」も同じように DAT ファイルに変換する。

1.6.2. メッシュファイルの出力

「Groups of Volumes」からはメッシュに対して物性値を設定するためのファイルを抽出するため、「Target」と「Shot」を UNV ファイルとして出力する。「Gr～」と名前がついているものは解析で用いないので無視する。「Target」を出力する場合は、「Groups of Volumes」内にある「Target」を右クリックして、「エクスポート」→「UNV ファイル」をクリックする。「FRAXST_test」まで移動して「保存(S)」をクリックする。「Shot」も同じように UNV ファイルに変換する。

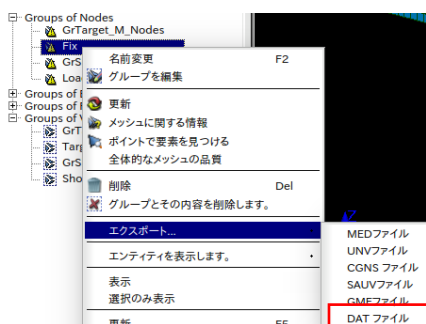


図 13.1 DAT ファイル

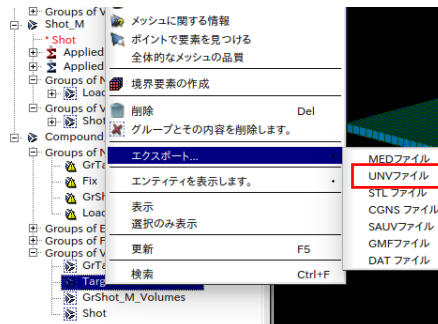


図 13.2 UNV ファイル

図 13 ファイルの変換

正しくファイルを出力すると、「FRAXST_test」フォルダーに図 14 のように 4 つのファイルが作成される。

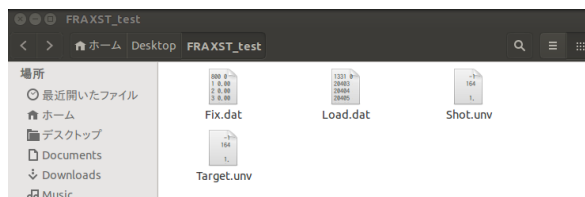


図 14 Salome で作成し、出力されたファイル

1.6.3. Salome の終了

1.6.2.までの一連の作業が終了したら、Salome で作成したモデリングのデータを hdf ファイルとして保存して、終了する。hdf ファイルは各層構造を持っているため、Salome で開くとオブジェクトブラウザ全ての情報が Salome 内に表示され、今までの作業のように編集ができる。

「ドキュメントを保存」をクリックすると、図 15 が表示される。「ファイル名」はデフォルトである「Study1」として、「検索する場所」は「FRAXST_test」のディレクトリとし、「保存(S)」をクリックする。



図 15 Salome で作成したファイルの保存

保存が終了したら、Salome のウインドウを×ボタンをクリックする。その際、本当に終了するかを問うウインドウが表示されるので、「OK(O)」をクリックすると Salome のウインドウが閉じられる。

2. FRAXST を用いた Peridigm による衝突解析

2.1. 設定条件

今回の解析では、図 16 の条件を用いる。「Shot」と連動している「Load」は 50m/s の速度で-Z 方向へ落下させ、「Target」の側面である「Fix」は X、Y、Z 方向完全拘束とする。材料は両方ともアルミニウムとし、衝突により破壊される「Target」のみ「破断ひずみ」を「0.005」と設定する。

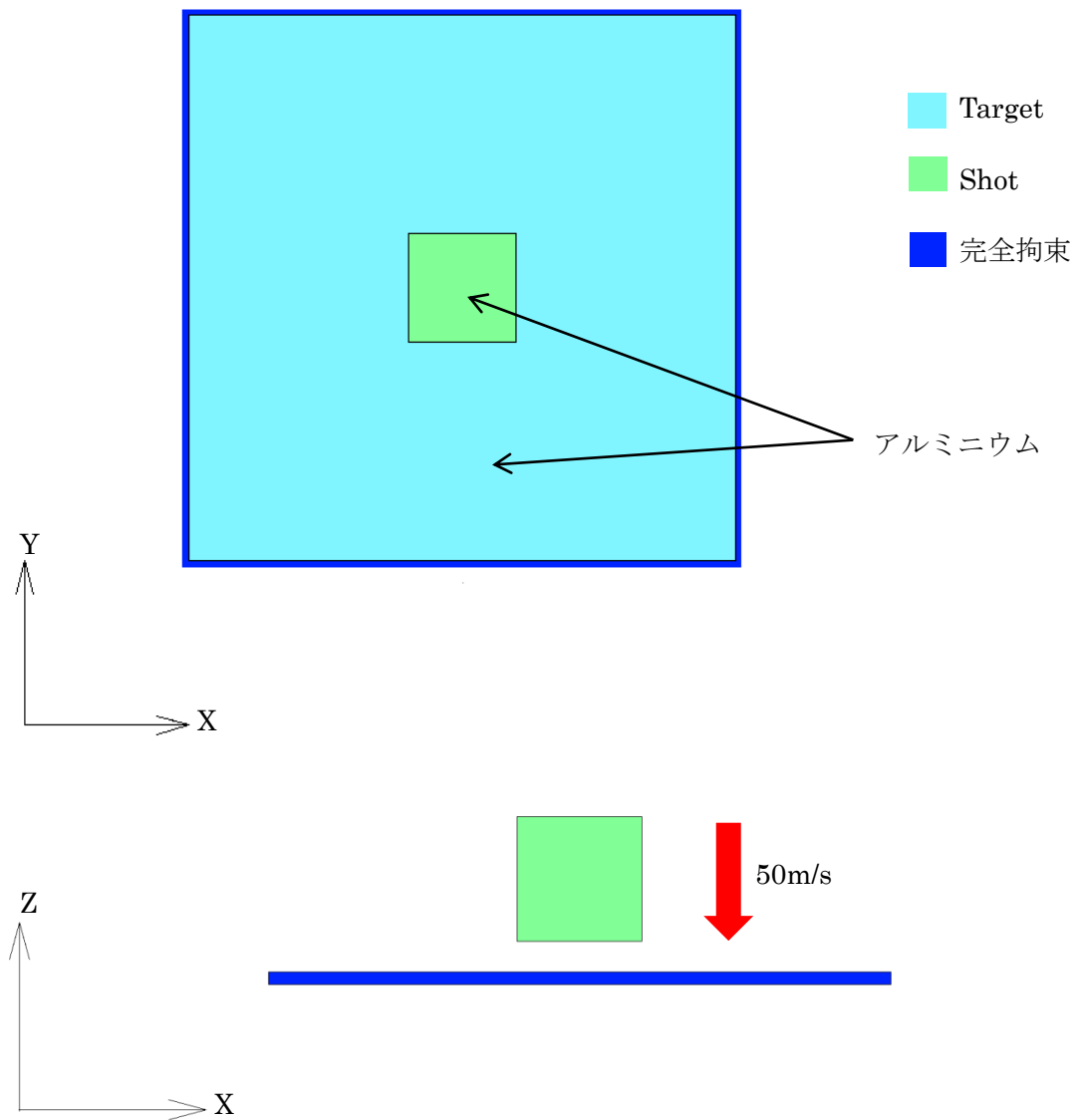


図 16 衝突速度と材料

2.2.FRAXST の起動と Genesis ファイルの作成

2.1.の作業が終わったら、Launcherにある「FRAXST」をクリックしてFRAXSTを起動する。起動すると、図 17 のようなウインドウが表示される。

2.2.1. 作業用 folder の選択

「作業用 folder 選択」にある「参照...」をクリックする。するとフォルダーを選択するウインドウが表示されるので、「デスクトップ」→「FRAXST_test」へ移動して「開く(O)」をクリックする。「作業用 folder 選択」にあるテキストボックス内に「FRAXST_test」のパスが表示されたことを確認する。

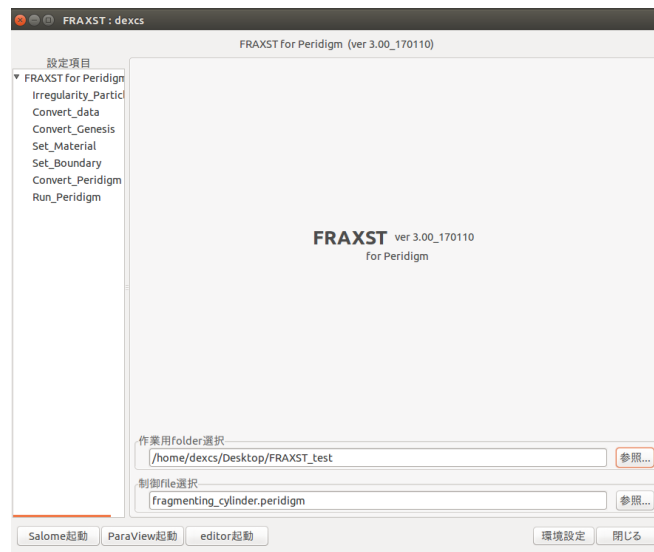


図 17 FRAXST 起動画面

2.2.2. txt ファイルへの変換

「設定項目」内のツリーから「Convert_data」をクリックすると図 18.1 が表示される。

「全 unvfile」には、「FRAXST_test」内にある全ての unv ファイルが表示されるため、このファイルを「HEX8file」や「TETRAfile」へ「追加>>」をクリックして移動させていく。「HEX8file」は Salome で切ったメッシュの形状が六面体であるモデルを追加するウインドウであり、「TETRAfile」は Salome で切ったメッシュの形状が四面体であるモデルを追加するウインドウである。本例題では、2つのモデルをどちらも六面体でメッシュを切ったため、「Target.unv」と「Shot.unv」を「HEX8file」へ追加する。

追加方法は、「Target.unv」をクリックして、「HEX8file」ウインドウの隣にある「追加>>」をクリックする。「Shot.unv」も同様に「HEX8file」へ追加する。

「全 datfile」には「Load.dat」と「Fix.dat」2つの境界条件が入っている。「datfile」はメッシュ形状に関わらず設定を行うため、2つとも「使用する datfile」へ追加する。追加方法は unvfile と同様で、「Load.dat」をクリックして「使用する datfile」の隣にある「追

加>>」をクリックする。「Fix.dat」も同様の手順で追加する。

ファイルの追加が完了し、図 18.2 のように設定できたことを確認したら、「ファイル変換」をクリックする。

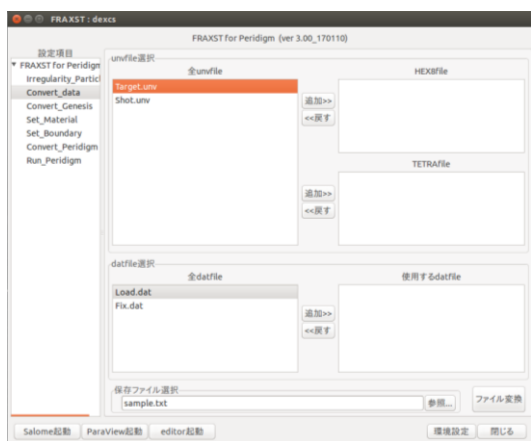


図 18.1 ウィンドウ

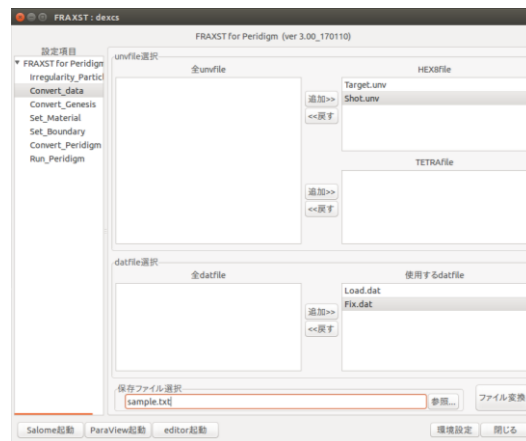


図 18.2 作業終了時

図 18 「Convert_data」

2.2.3. g ファイルへの変換

「設定項目」内のツリーから「Convert_Genesis」をクリックすると図 19.1 が表示される。「ファイル変換」をクリックすると、g ファイルに変換される。図 19.2 のように「file 情報」にメッシュから粒子モデルへ変換された g ファイルの情報が表示される。

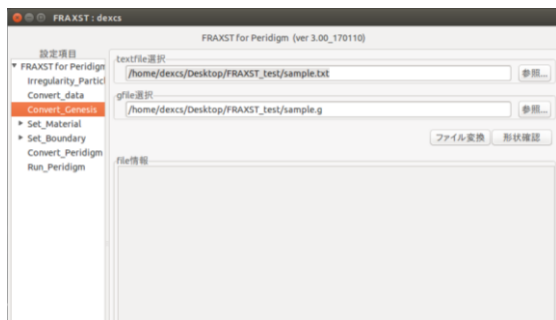


図 19.1 ウィンドウ

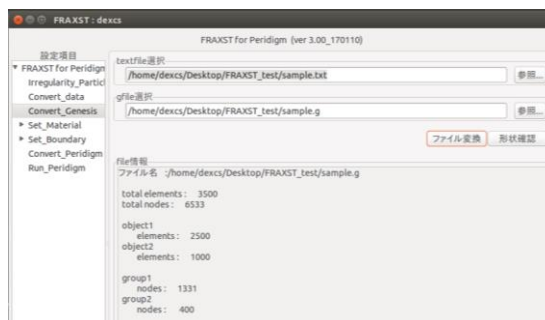


図 19.2 g ファイルへの変換後

図 19 「Convert_Genesis」

ファイル変換を行った後に、「形状確認」をクリックすると ParaView が起動する。ParaView の「Properties」内にある「Apply」をクリックすることで、図 20 のように g ファイルが正しく変換されたかを確認できる。ParaView を起動中は FRAKST を操作できないので、確認が終わったら ParaView のウィンドウを×ボタンをクリックして閉じる。

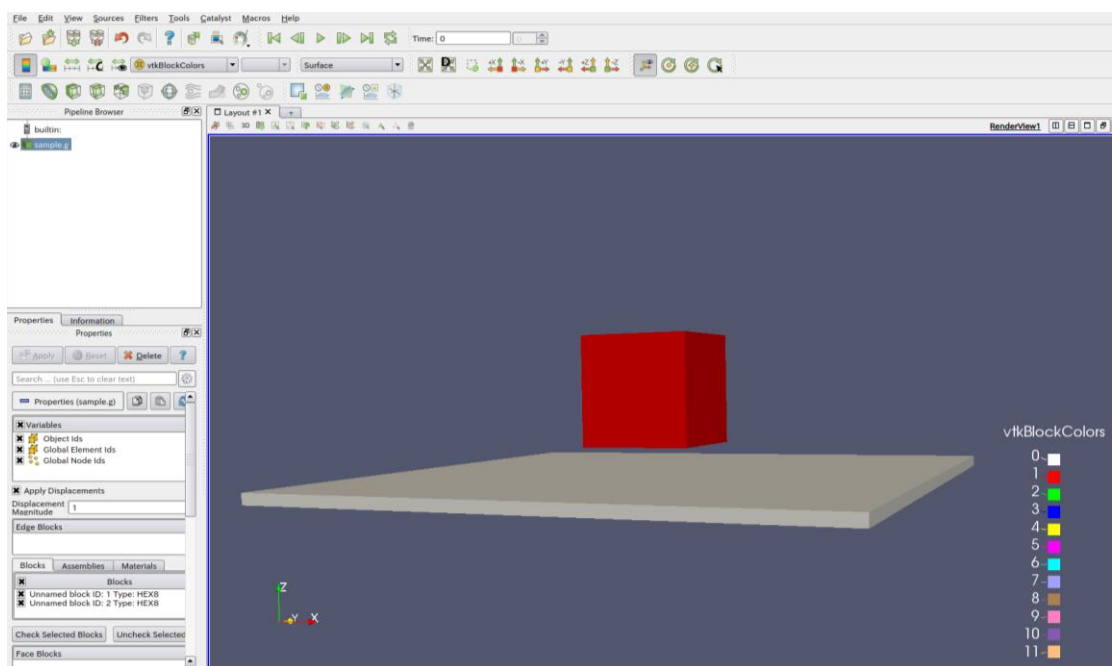


図 20 g ファイルに変換されたオブジェクト

2.3.unv ファイルに対する物性の設定

「設定項目」内のツリーから「Set_Material」をクリックすると図 21.1 が表示される。「材料DBの設定」は物性値の一覧が記されている csv ファイルのパスである。基本的には、操作する必要はないが、物性の値を変更したい場合や、新たな物質を追加したい場合は、このパスにあるデータを変更する。

「設定項目」ツリーの「Set_Material」先頭にある▼をクリックすることで、図 21.2 のように各オブジェクトのツリーが展開され、各 unv ファイルに対して物性を設定できる。ツリーに表示された順番通りに設定し、物性に関しては図 16 を参考にする。



図 21.1 ウィンドウ



図 21.2 ツリータブ展開後

図 21 「Set_Material」

2.3.1. Target.unv

「Target.unv」をクリックすると「Target」に対する物性への設定項目が表示される。

「材料モデル」は現在 4 種類の解析が可能であるが、本例題では最も設定が簡単な「塑性モデル」を選択する。

「物性値の設定」は材料を「プルダウンメニュー アルミニウム」を、クリックすると、材料の一覧が表示され、選択された物性値が入力される。本例題では、「アルミニウム」とする。

「物性値の確認」をクリックすると、現在選択している物質についての詳しい物性値を確認することができる。

「脆性破壊への切り替え」は「Target」に関しては破壊の様子を確認するため、脆性モデルとする。「脆性モデル」をクリックしてチェックを付け、「破断ひずみ」を 0.005 と入力する。「破断ひずみ」とは、物体に力が加わりひずみが発生し破断するまでの、ひずみの大きさを示す。ひずみの大きさが「破断ひずみ」の値を超えると、物体が破断していくため、本例題では、一般的なアルミニウムの破断ひずみよりも小さな 0.005 (0.5%) とすることで、破断させやすくしている。

「影響半径」はモデル化された粒子同士がお互いに影響を及ぼしあう範囲であり、この影響半径によって、粒子は連続体として解析が可能となる。影響半径は一般的に粒子間距離の 3 倍程度にすることで、解析精度が上がると言われている。粒子間距離は Salome で作成したメッシュの間隔と等しいため、本例題では 3 倍よりも少し大きい 0.032(m) とする。

全設定が完了すると、図 22 のようになる。このように設定したら、「設定」をクリックし、「Target.unv」に対する物性を確定させる。



図 22 「Target.unv」に対する物性

2.3.2. Shot.uuv

「Shot.unv」に対しても 2.3.1.と同様の手順で設定をする。材料モデルや材料、影響半径は全く同じであるが、「Shot」は破断しないため、「脆性モデル」にはチェックを入れないようにする。図 23 のように設定が出来たら、「設定」をクリックし、「Shot.unv」に対する物性を確定させる。

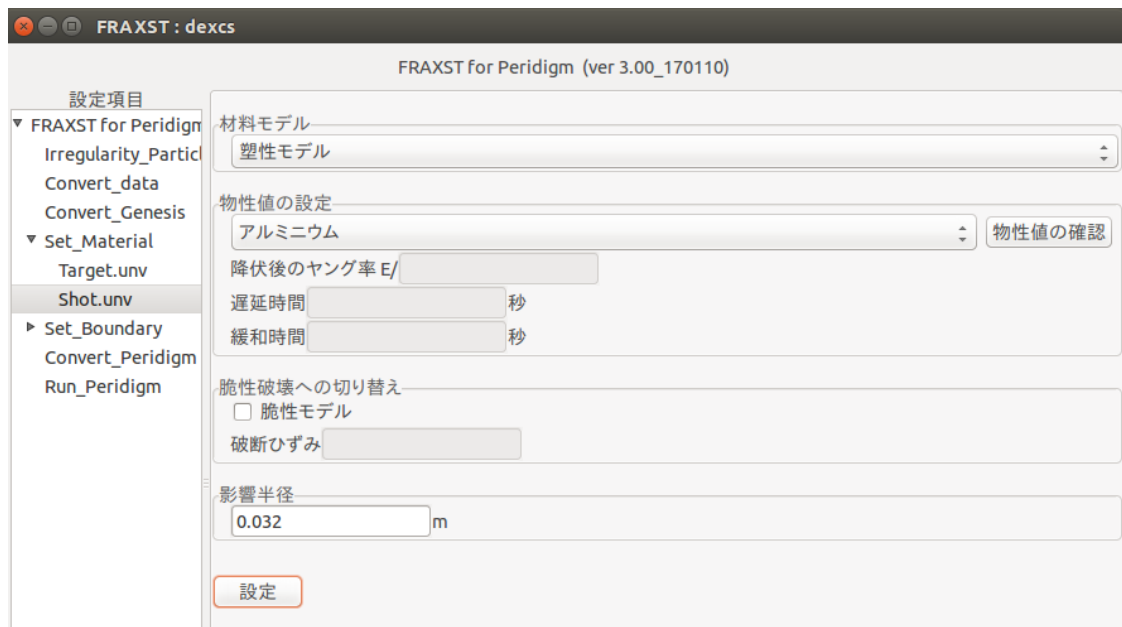


図 23 「Shot」に対する物性

2.4.dat ファイルに対する境界条件の設定

「設定項目」内のツリーから「Set_Boundary」をクリックすると図 24.1 が表示される。ここでは衝突（接触）解析であるかの有無を設定する。本例題では衝突解析を取り扱うので「衝突解析」にチェックがついていることを確認する。次に「接触条件」内の「探索半径」と「衝突半径」に「Target」の板厚である 0.01 を入力する。「探索頻度」と「バネ定数」はデフォルトの値をそのまま用いる。「バネ定数」に関してはまだ不明な点が多く、解析をするオブジェクトによって値を変える必要があり、大体は $1.0E+10 \sim 1.0E+12$ の範囲で設定する。バネ定数が小さすぎると、衝突せずに貫通し、大きくしすぎると衝突時に物体が激しく離散してしまう。図 24.2 のように設定が完了したら「設定」をクリックし、衝突解析に関する設定を確定する。

「設定」をクリックしたら、「Set_Boundary」先頭にある▼をクリックすることで、境界条件に関する各オブジェクトツリーが展開される。2.3.と同様に、ツリーに表示された順番通りに設定を行っていく。

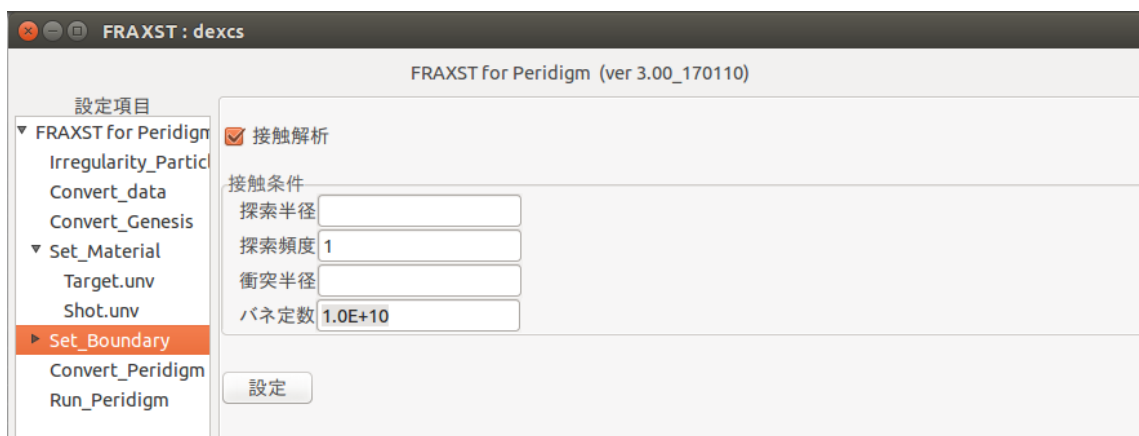


図 24.1 ウィンドウ

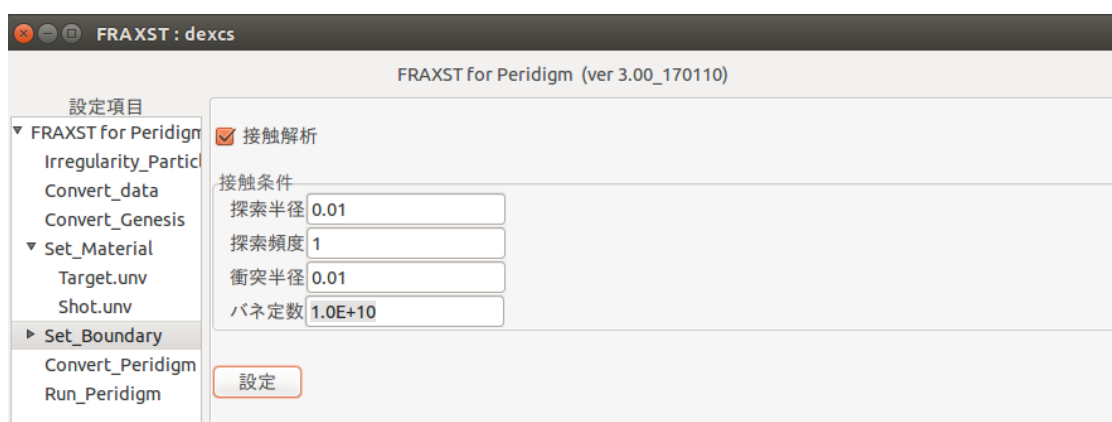


図 24.2 接触条件の設定後

図 24 「Set_Boundary」

2.4.1. Load.dat

「Load.dat」をクリックすると、「Load」すなわち「Shot」をどのように運動させるかを設定するウィンドウが表示される。

「解析の種類」は現在 4 種類が確認されているが、「温度制御」と「体積力」に関してはパラメータの設定方法が確立していないため、現段階では扱わない。

「Load」は図 16 のように、 $-Z$ 方向に 50m/s の速度を持っている。よって、「解析の種類」は「初速度(m/s)」とする。コンボボックスから「初速度(m/s)」を選択したら、「制御する値の設定」を X、Y、Z 方向に対して行う。鉛直下向きの運動のため X、Y 方向に関しては、0 を入力することで、運動を Z 方向に固定することが出来る。Z 方向のテキストボックスには -50 と入力する。

正しく設定すると図 25 のようになるので、「設定」をクリックすると、「Load.dat」に対する制御方法と方向を確定することが出来る。



図 25 「Load.dat」に対する境界条件

2.4.2. Fix.dat

「Fix.dat」も 2.4.1 と同様の手順で設定を行う。「Fix.dat」は「Target」の 4 辺を完全拘束するため、「変位制御(m)」として、X、Y、Z 方向を全て 0 とする。こうすることで、力が加わったとしても「Fix.dat」は移動されない。

正しく設定すると図 26 のようになるので、「設定」をクリックすると、「Fix.dat」に対する制御方法と方向を確定することが出来る。

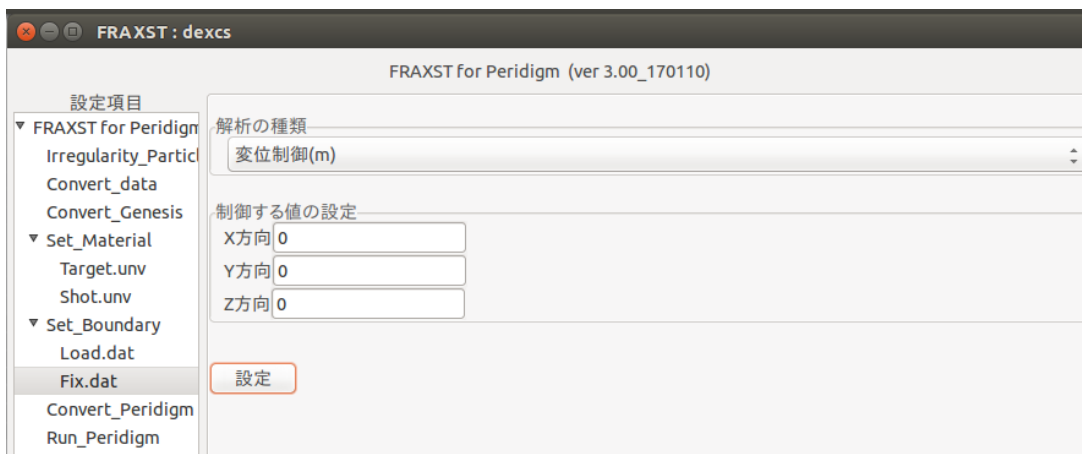


図 26 「Fix.dat」に対する境界条件

2.5.出力方法の設定と peridigm ファイルへの変換

「設定項目」内のツリーから「Convert_Peridigm」をクリックする。「衝突時間(FinalTime)の設定」には、何秒間の現象を解析するかを設定する。解析したい現象が何秒間で再現可能かを考えて時間を設定するとよい。本解析では、0.003 と入力する。「衝突時間の間隔」と「出力頻度」はデフォルトの値とする。衝突時間の間隔は、何秒ごとに解析をするかを定める値であり、この値が小さいほど解析時間が長くなる。それに加え、クーラン条件を考慮する必要がある、大きくしすぎると正しく解析が行えなくなる。大体 $1.0E-6 \sim 1.0E-8$ ほどにすると、安定した解析が可能となる。「Steps の確認」をクリックすると、端末内に「Step 数を 3000 に設定しました」と表示される。Step 数は「衝突時間」を「衝突時間の間隔」で除したものであり、時間刻みごとの解析の回数で表している。Step 数が大きいほど、解析に時間がかかる。Step を「出力頻度」で除することで、ParaView で表示できるコマ数が決定される。本例題の場合は、3000Step を 100 で除するため、30 コマの画像を表示することになる。

「出力の種類」では、ParaView で表示出来る結果を選択する。デフォルトでチェックされている 4 種類をよく使うため、そのままにする。「Displacement」は粒子が元の場所から、どれほど移動したか、「Velocity」は粒子がどれほどの速度を持っているか、「Damage」は粒子が受ける応力、「Force」は加わっている力が表示される。

「efile 名の設定」では、Peridigm で計算して出力される e ファイル名 (拡張子.e を除く) を設定する。本例題では、「sample」と入力する。

正しく設定すると、図 27 のようになるので「ファイル変換」をクリックする。

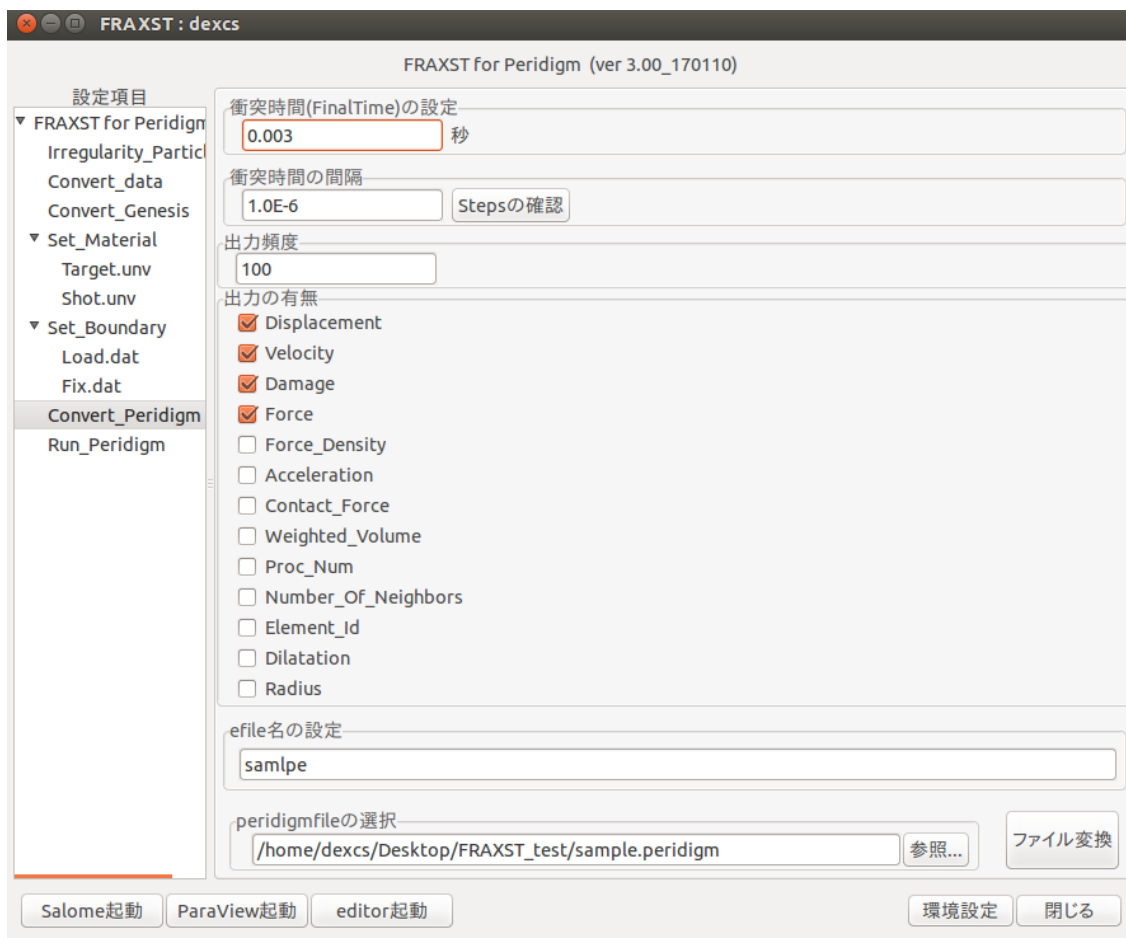


図 27 設定された「Convert_Peridigm」

※各設定をもう一度変更した場合も、最後に「ファイル変換」をクリックしないと、変更が保存されないため注意する。

2.6.Peridigm の実行

「設定項目」内のツリーから「Run_Peridigm」をクリックすると、図 28 が表示される。

「並列処理の設定」では、ファイルを分割して並列処理をするかを選択する。並列処理が可能な CPU ならば並列処理をした方が早く解析を行うことができるので、「並列処理をする」をクリックして、チェックをすることで、「CPU 数」に使用可能な最大の CPU コア数が入力される。最大よりも少ないコア数へ変更する場合は、「CPU 数」を書き換える。単コアのコンピュータを用いる場合は、「並列処理をする」のチェックが外れていることを確認する。

確認が終わったら、「Peridigm の実行」をクリックする。そうすると、端末に図 29.1 のような処理の進捗が表示され、「Explicit time integration」が 100%になると計算が完了し、のように処理時間が表示される。たとえば、本例題を Intel®Core™i7-3820 クロック数 3.60GHz×8 の CPU で 8 コア並列の計算をすると、1 分ほどで計算が終わる。単コアの場合でも 10 分以内に計算が終わる。また、「.e ファイルの作成に成功しました」とメッセージログも表示されるので「OK(O)」をクリックする。

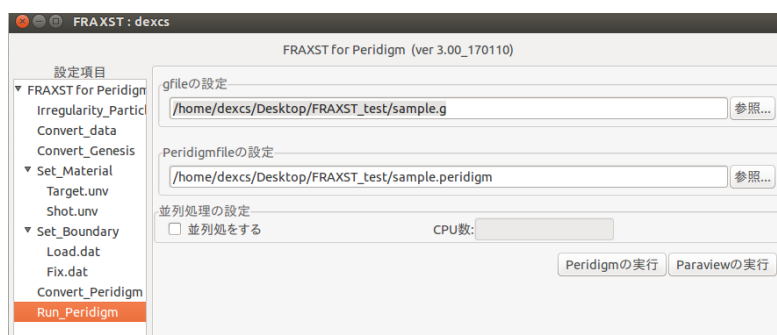


図 28 「Run_Peridigm」 ウィンドウ

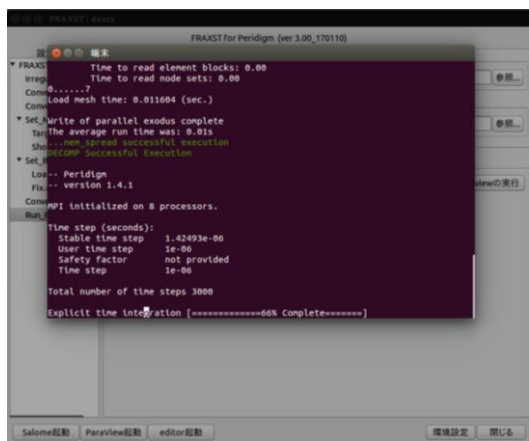


図 29.1 実行様子の様子

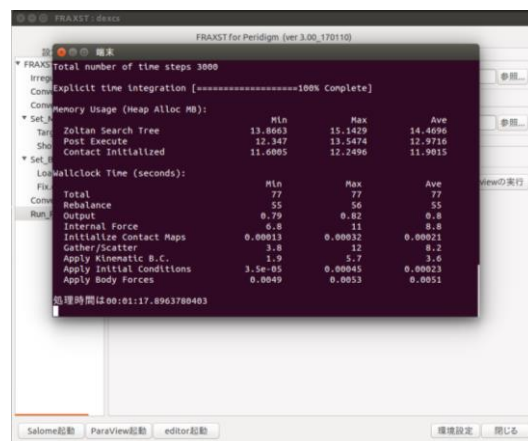


図 29.2 計算が終了した様子

図 29 Peridigm の実行

2.7.解析結果の表示

計算が終了したら、「ParaView の実行」をクリックすることで、「sample.e」がツリーに表示された状態の ParaView が起動する。「Properties」内にある「Apply」をクリックすることで、図 30 のように粒子モデルで構成されたオブジェクトが表示される。ツールバー内の▶をクリックすることで、Time が進んでいき、衝突の様子が確認できる。また、2.5. で設定した出力方法が「Properties」内にあるため、それをチェックし、「Apply」をクリックするとツールバー内のコンボボックス `vtkBlockColors` で、設定した出力の種類を選択できるようになる。

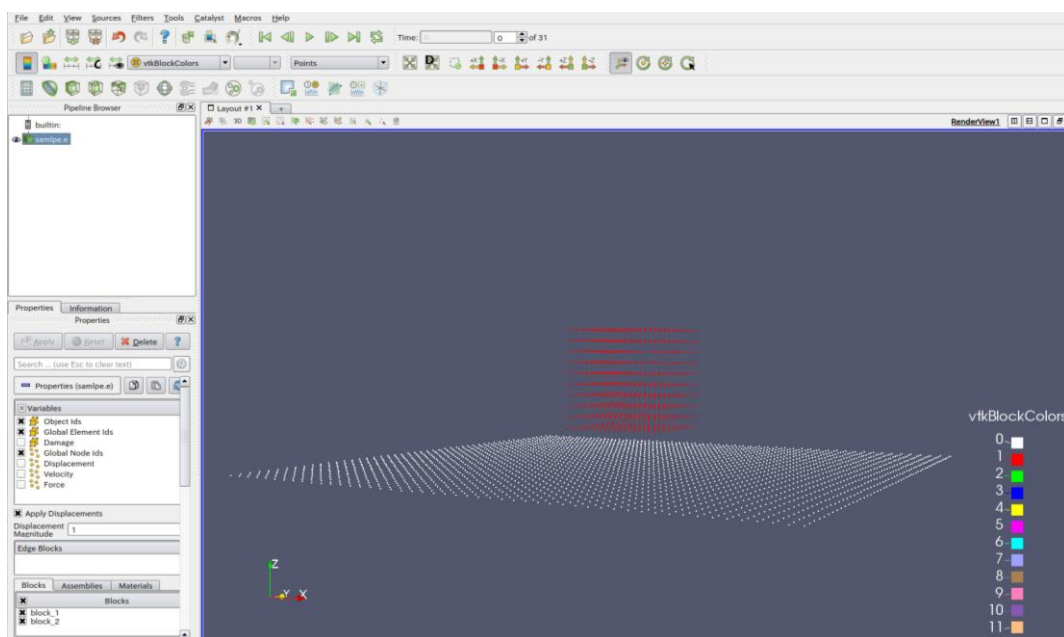


図 30 ParaView を起動して、「sample.e」を追加した様子

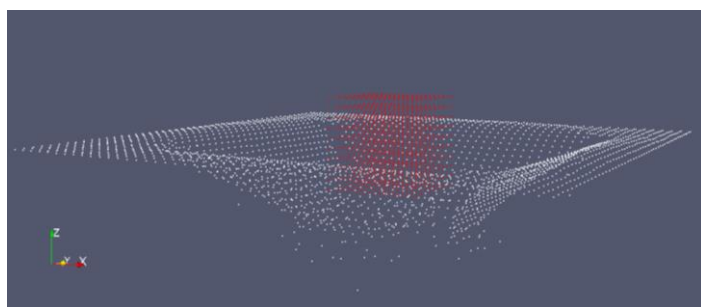


図 31.1. 全体図

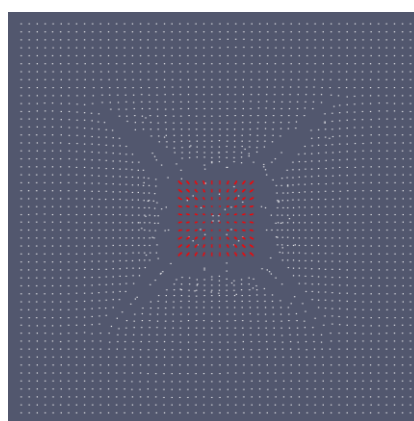


図 31.2. XY 平面図

図 31 「Target」が貫通した様子 (Time=0.003)

終わりに

FRAXST の基本的な粒子モデルの変換方法では、Salome でメッシュを切り、その中央に規則正しく粒子を配置しているため、計算結果も規則正しい壊れ方をしている。

そこで、応用として、FRAXST に乱数を用いて、粒子の配置をずらし、初期不整を発生させる方法がある。境界面が乱れてしまう欠点があるため、解析の精度に問題があるが、不規則な破壊形式となる。

1. 粒子モデルの不規則配置方法

2.2.1.が終わったら、「設定項目内」の「Irregularity_Paticle」をクリックする。。ここでは、メッシュの形状ごとに、初期不整を設けるかを設定できる。今回は本例題で使用する六面体メッシュに対して初期不整を設定する。

「HEX8file」にある「初期不整」をクリックし、「確率」と「粒子間距離を」入力する。「確率」では、粒子全体の何%を変化させるか設定できる。本例題では100(%)とする。「粒子間距離」にはメッシュを切った大きさ（本例題では 0.01）を入力する。粒子間距離によって、初期不整の上限値を決めるため、「粒子間距離」が大きすぎるとモデルの初期不整が大きくなりすぎて解析の際にエラーが発生してしまうので注意する。

図 32 のように入力したら、2.2.2.と 2.2.3.の作業を行う。

この時に「ファイル変換」をクリックするまで作業をしないと、入力した初期不整が適応されないため注意する。

peridigm ファイルは初期不整入力前と同じものを用いるため、2.3.~2.5.は行う必要はない。

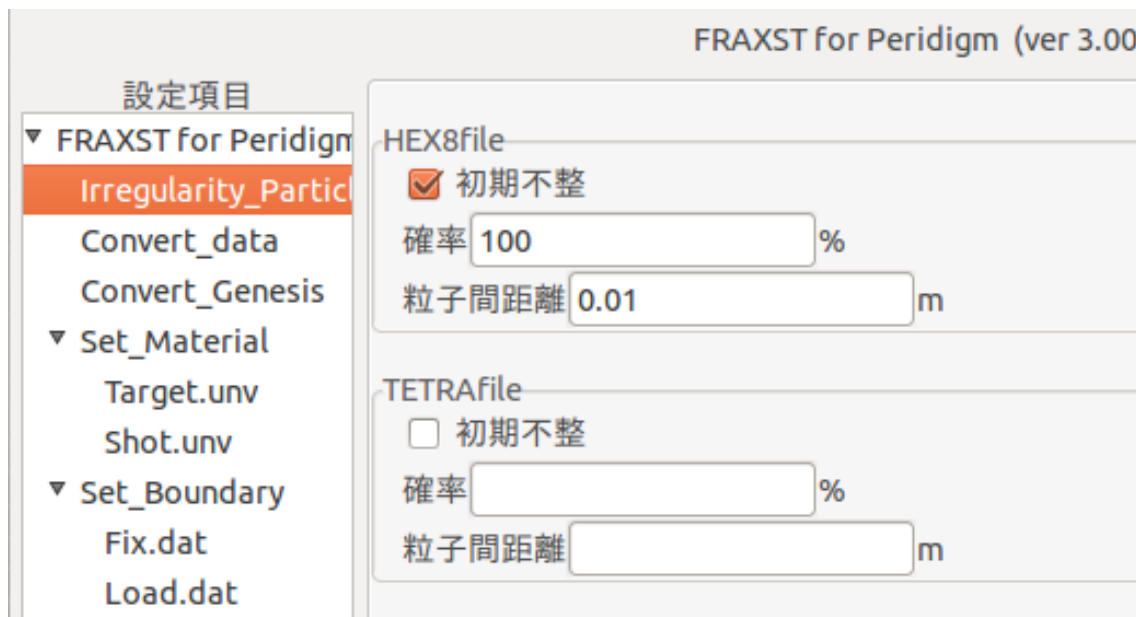


図 32 初期不整に関するデータを入力した「Irregularity_Particle」

2.6.のように「Peridigm の実行」を行うと、初期不整を考慮した.e ファイルが作成される。ParaView を確認すると、図 33 のように「Target」が破断する。図 31 の結果に比べて、不規則に破断や亀裂が生じる。初期不整は乱数により散らされているため、g ファイルへの「ファイル変換」を行うたびに違う結果が生じる。

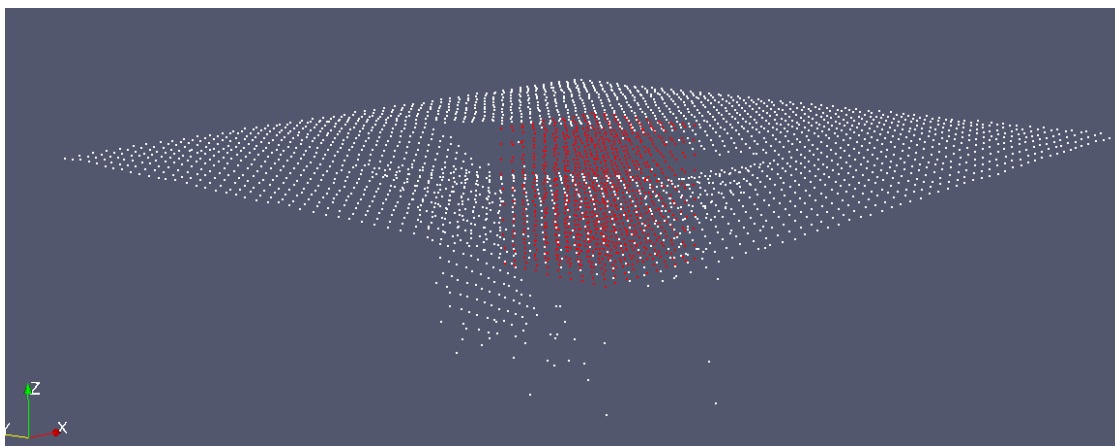


図 33.1 全体図

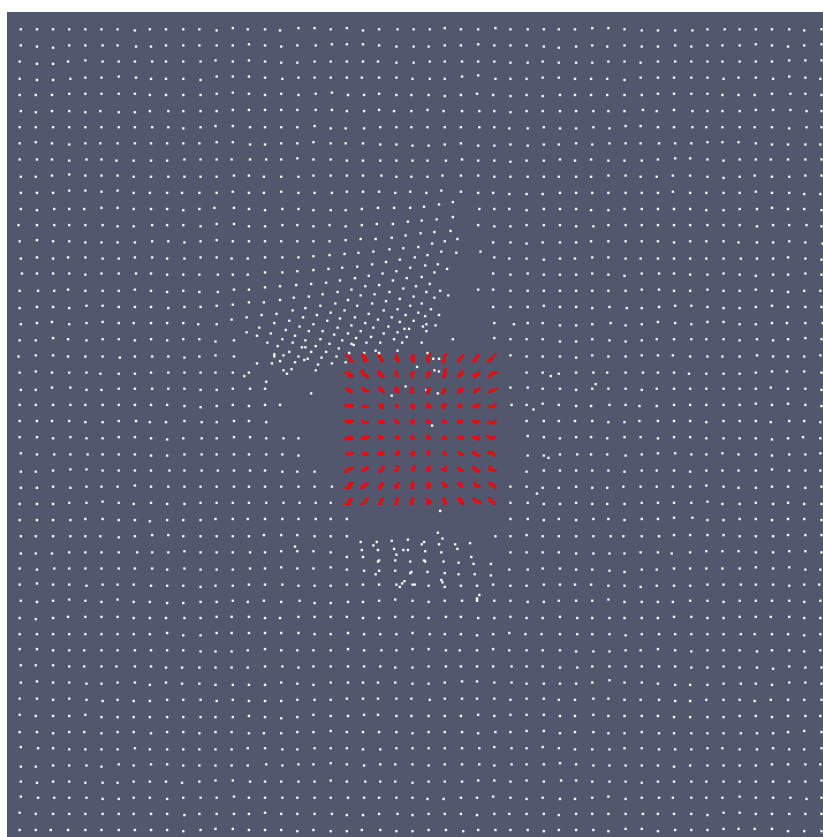


図 33.2 XY 平面図

図 33 初期不整を入力したモデルが破断された様子 (Time=0.003)