

Peridigmを有効活用するための 統合支援ツールFRAXSTの開発

岐阜工業高等専門学校 柴田研究室

藤田耕平

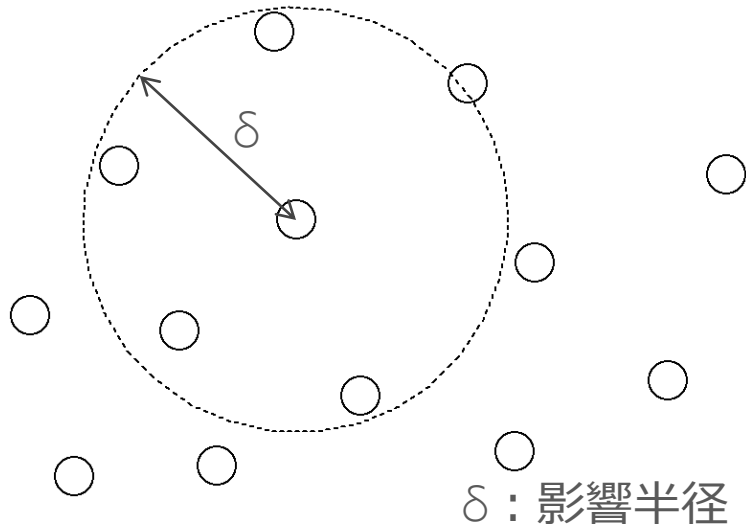
序論

- 災害などによる構造物の崩壊を再現するには粒子モデルが有効
- ▼
- 粒子モデルを用いた解析ソフトウェア Peridigmは入力データの作成が困難
- ▼
- Peridigmを効率よく活用できる統合支援ツールFRAXSTを開発

Peridigmとは

- Peridynamics理論を元とした破壊解析用ソフトウェア
- 粒子モデルに対し、初速度や変位を与えることで破壊現象を再現
- 任意形状の入力データの作成には有償ソフトウェアCubitが必要

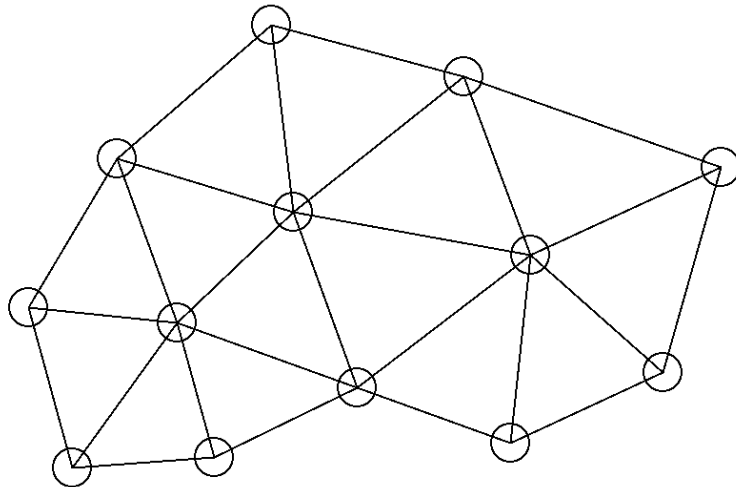
解析手法



Peridynamics理論

影響半径内の相互作用による
粒子の運動により、連続体を
表現する

破壊現象を容易に扱える



有限要素法

モデルを節点と要素により
分割することで解いていく

**要素が分離するほどの破壊
現象を解くことは難しい**

FRAXSTとは

- PythonとTrilinosを組み合わせ、
**オープンソースソフトウェアによる
モデル作成**
- プリ・ソルバー・ポストの連携
- GUIによる作業効率の向上

モデルの変換手順

モデリング

3D-CAD (Salome) により
解析モデル、境界条件の設定、メッシュの作成

座標抽出

UNVファイルをPythonにより読み込み
節点座標等の必要な数値を抽出

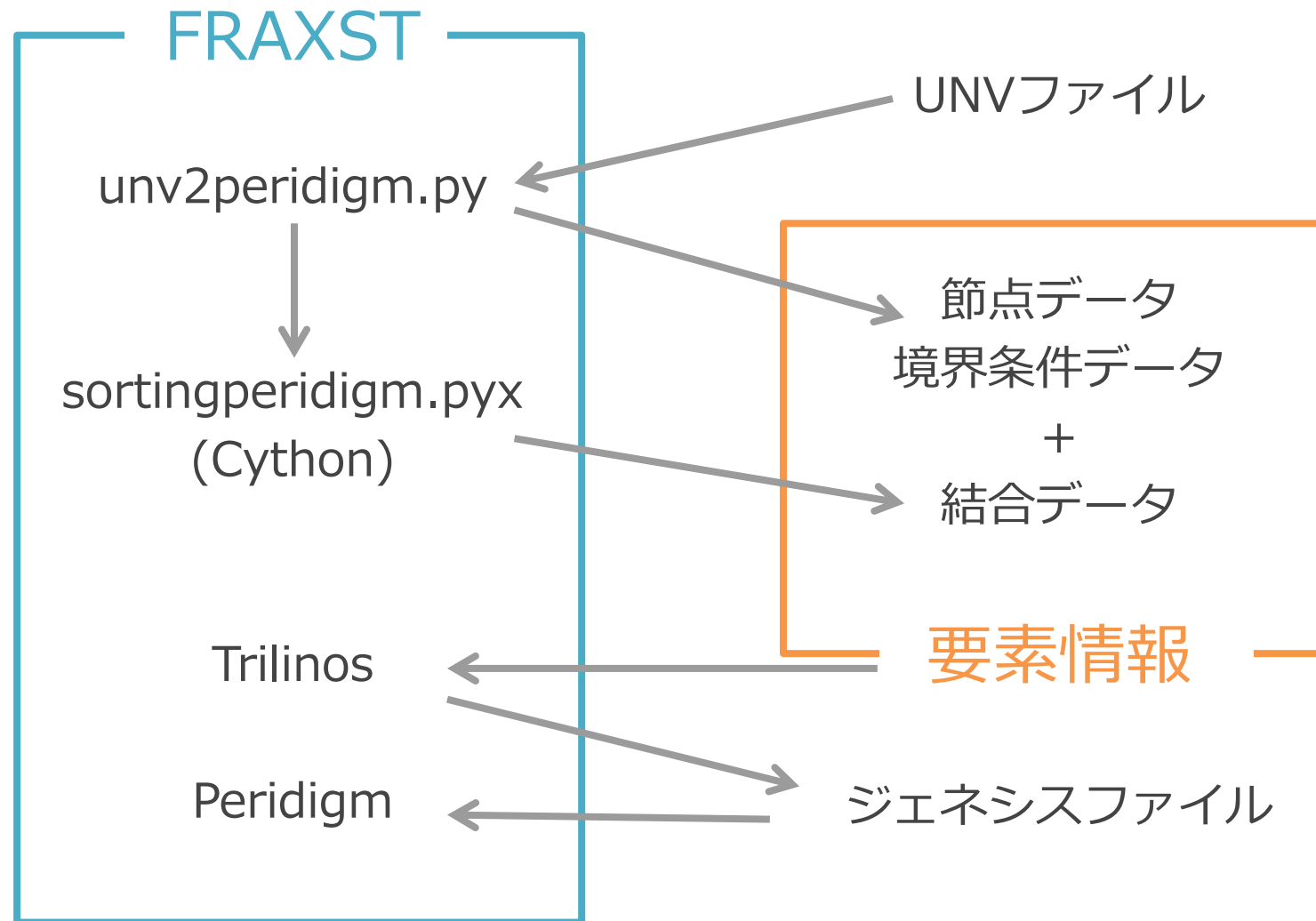
座標書出

Python + Cythonにより変換可能な
テキストファイルに書き換え

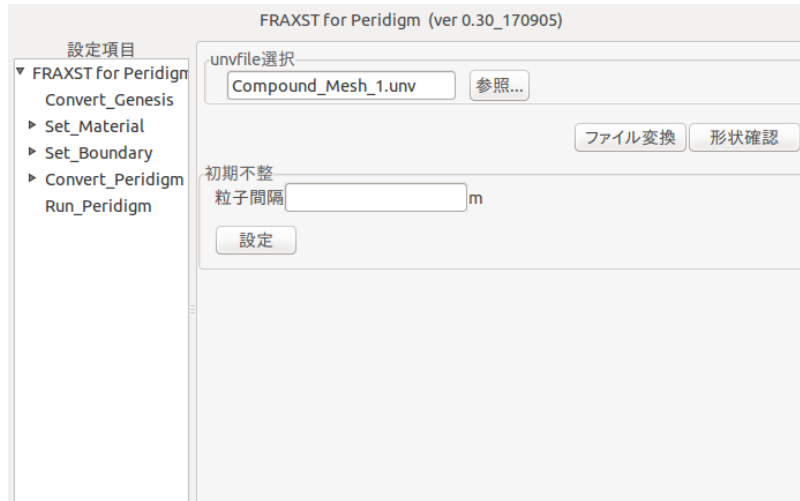
変換

Trilinosプログラムを使用し、テキストファイルを
ジェネシスファイルに再変換

モデルの変換手順



モデルの変換手順



GUI（マウス操作）

作業フォルダとUNVファイルを参照してファイル変換

```
! Database initial variables
Number of coordinates per node    =      3
Number of nodes                  =    6533
Number of elements               =    3500
Number of element blocks         =      2

Number of node sets              =      2
Length of node list              =    1731
Number of side sets              =      0

txtexo2 used 0.05 seconds of CPU time
ファイル名      :Peridigm_Model.g

total elements :    3500
total nodes   :    6533

object1
  elements :    2500
object2
  elements :    1000

group1
  nodes :    400
group2
  nodes :    1331
```

ログ表示

座標抽出～書出までを行い、
Trilinosのプログラムを起動
ジェネシスファイルを生成

Cythonとは

```
# -*- coding: utf-8 -*-  
from multiprocessing import Pool
```

```
def sortconnectivity(connectivity, grouplst, lines2):
```

```
    cdef :
```

```
        int i,n
```

```
        int group = len(grouplst)
```

```
        int line = len(lines2)
```

ループ処理に使用する変数 = 整数
int型と宣言することで高速化

```
    for i in range(group):
```

```
        for n in range(line):
```

```
            if grouplst[i] == lines2[n][:10].strip():
```

```
                connectivity.write(lines2[n][10:])
```

```
                break
```

- PythonにC言語を組み合わせた拡張モジュール
- データ型の宣言をすることでC言語の様に高速化
- 計算負荷が大きい箇所のみを変更

制御方法の設定

設定項目

- FRAXST for Peridigm
 - Convert_Genesis
 - Set_Material
 - target**
 - shot
 - Set_Boundary
 - fix
 - load
 - Convert_Peridigm
 - Run_Peridigm

材料モデル
弾性

物性値の設定
アルミニウム 物性値の確認

降伏後のヤング率
アワークラスモード

遅延時間 秒
緩和時間 秒

脆性破壊への切り替え
☒ 脆性モデル
破断ひずみ 0.005

影響半径
0.032 m

設定

材料

材料モデル

物性値（材料名から）

脆性・靱性の切り替え

影響半径

設定項目

- FRAXST for Peridigm
 - Convert_Genesis
 - Set_Material
 - target
 - shot
 - Set_Boundary
 - fix
 - load**
 - Convert_Peridigm
 - Run_Peridigm

解析の種類
初速度(m/s)

制御する値の設定
X方向 0
Y方向 0
Z方向 -50

設定

境界条件

初速度

変位制御（固定）

FRAXSTの展開

1 WindowsOSへの対応

2 多言語対応

FRAXSTの展開

1 WindowsOSへの対応

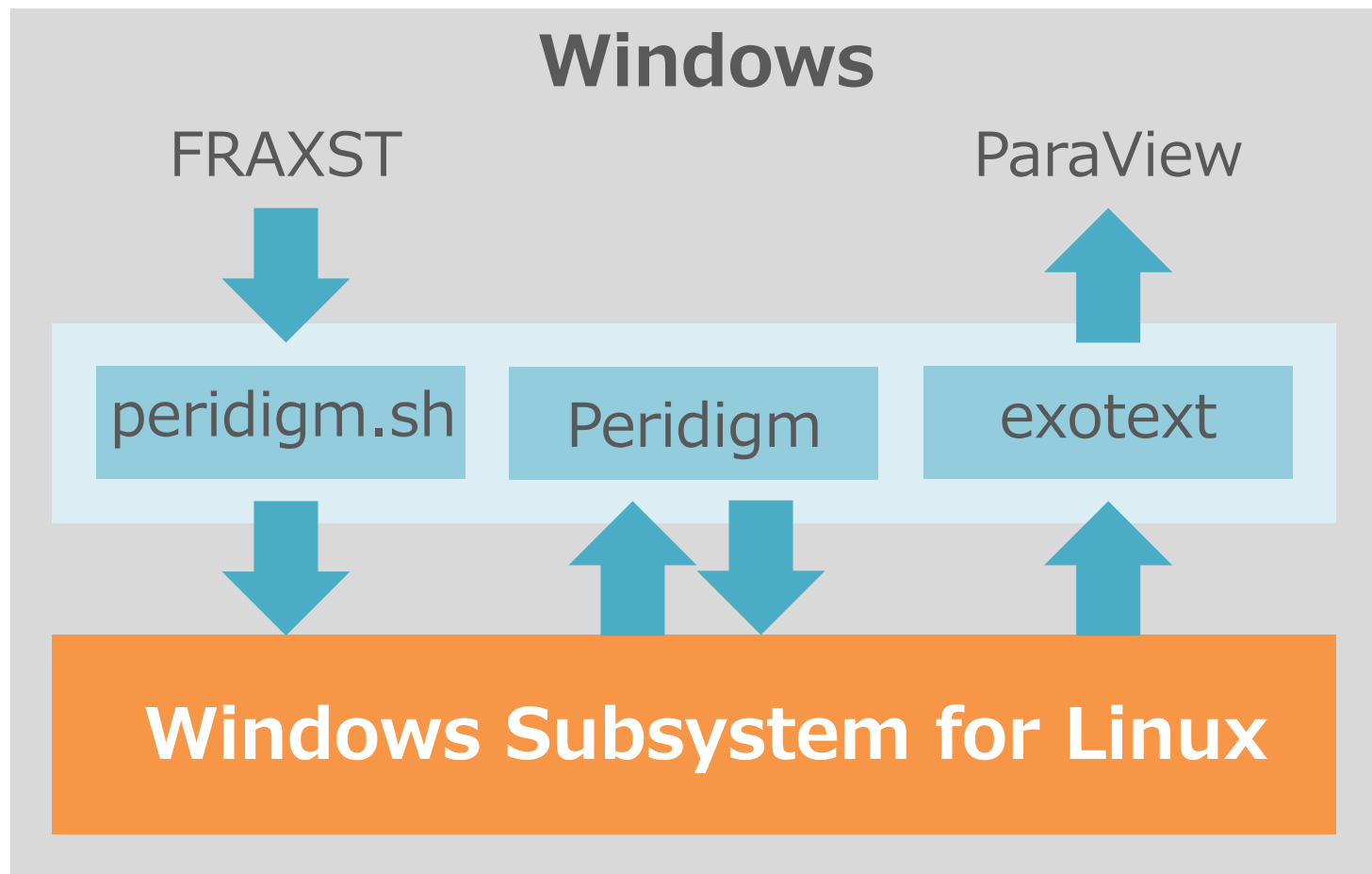
2 多言語対応

Windowsへの対応

- PeridigmはLinuxベースのOSでしか動かせない
- 間口を広くするにはWindowsOSへの対応が必要

Windows Subsystem for Linuxにより
Peridigmの計算を行う

Windowsによる計算の流れ



導入手順

- 1 WinXistrの構築
- 2 Cythonの導入
- 3 Peridigmの構築

Cythonの導入

Cython, a language for writing Python C extensions.

[Cython-0.27.3-cp27-cp27m-win32.whl](#)

[Cython-0.27.3-cp27-cp27m-win_amd64.whl](#)

[Cython-0.27.3-cp34-cp34m-win32.whl](#)

[Cython-0.27.3-cp34-cp34m-win_amd64.whl](#)

[Cython-0.27.3-cp35-cp35m-win32.whl](#)

[Cython-0.27.3-cp35-cp35m-win_amd64.whl](#)

[Cython-0.27.3-cp36-cp36m-win32.whl](#)

[Cython-0.27.3-cp36-cp36m-win_amd64.whl](#)

<http://www.lfd.uci.edu/~gohlke/pythonlibs/#cython>

Package	Class	Installed Version	Repository Version	Description
☐ mingw32-gcc-fortran	dev		5.3.0-3	The GNU FORTRAN Co
☐ mingw32-gcc-fortran	doc		5.3.0-3	The GNU FORTRAN Co
☐ mingw32-gcc-fortran	info		5.3.0-3	The GNU FORTRAN Co
☐ mingw32-gcc-fortran	man		5.3.0-3	The GNU FORTRAN Co
☒ mingw32-gcc-g++	bin	5.3.0-3	5.3.0-3	The GNU C++ Compiler
☒ mingw32-gcc-g++	dev	5.3.0-3	5.3.0-3	The GNU C++ Compiler
☐ mingw32-gcc-g++	doc		5.3.0-3	The GNU C++ Compiler
☐ mingw32-gcc-g++	man		5.3.0-3	The GNU C++ Compiler
☐ mingw32-gcc-objc	bin		5.3.0-3	The GNU Objective-C C
☐ mingw32-gcc-objc	dev		5.3.0-3	The GNU Objective-C C
☐ mingw32-gcc-tools-epoc...	bin		2.64-1	special autoconf for gcc
☐ mingw32-gcc-tools-epoc...	doc		2.64-1	special autoconf for gcc
☐ mingw32-gcc-tools-epoc...	lic		2.64-1	special autoconf for gcc

General

Description

Dependencies

Installed Files

Versions

The GNU C++ Compiler

This package provides the MinGW implementation of the GNU C++ language compiler.

This is an optional component of the MinGW Compiler Suite; you require it only if you wish to compile programs wr

Cython

Webからダウンロード
WinXistrのPythonが32bit
のためwin32を選択

C,C++コンパイラ

MinGWによりダウンロード
(mingw32-gcc-g++)

Peridigmの構築

Peridigmのバイナリを準備

Ubuntu版のバイナリをC:¥DEXCSにコピー

足りないライブラリの導入

liblapack3 , libopenmpi-devを
Windows Subsystem for Linuxに導入

bash.exeを別ファイルへコピー

C:¥Windows¥System32 → C:¥DEXCS

FRAXSTの展開

1 WindowsOSへの対応

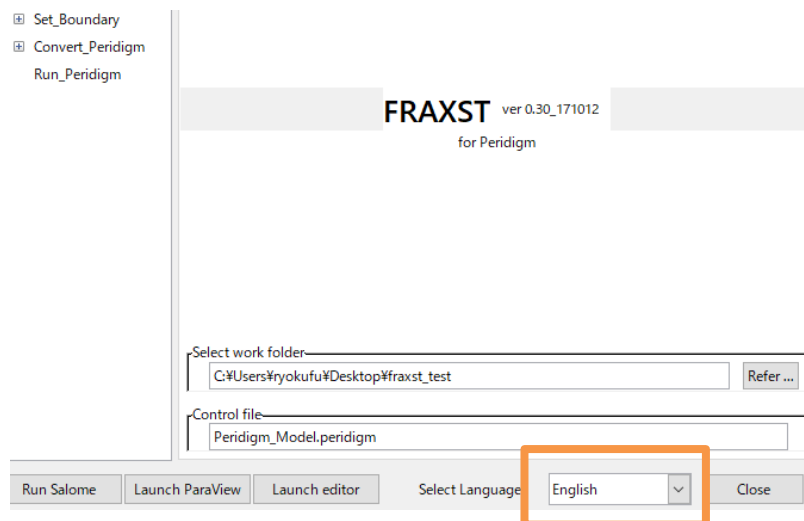
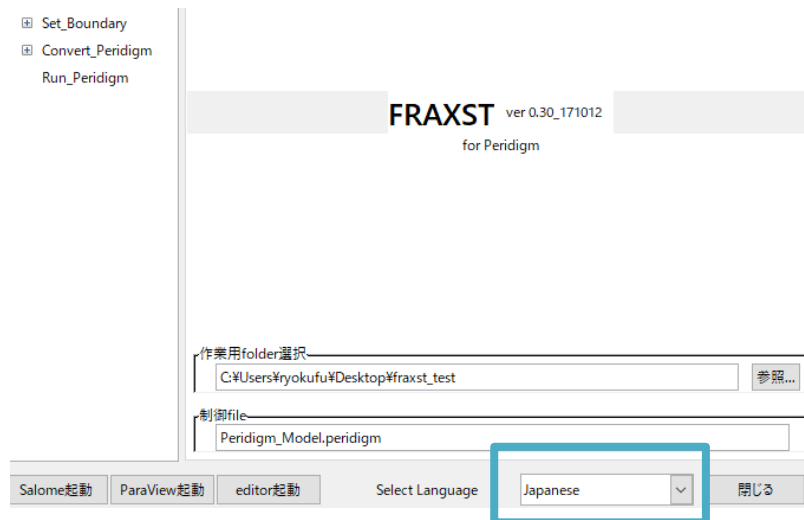
2 多言語対応

多言語対応

- Peridynamicsは海外でも盛んに研究が行われている
- Peridynamicsに基づいている
Peridigmにも高い関心があると考えられる

英語と日本語を切り替えることで
利用者の拡大を狙う

多言語対応



Select Languageを
変更することで

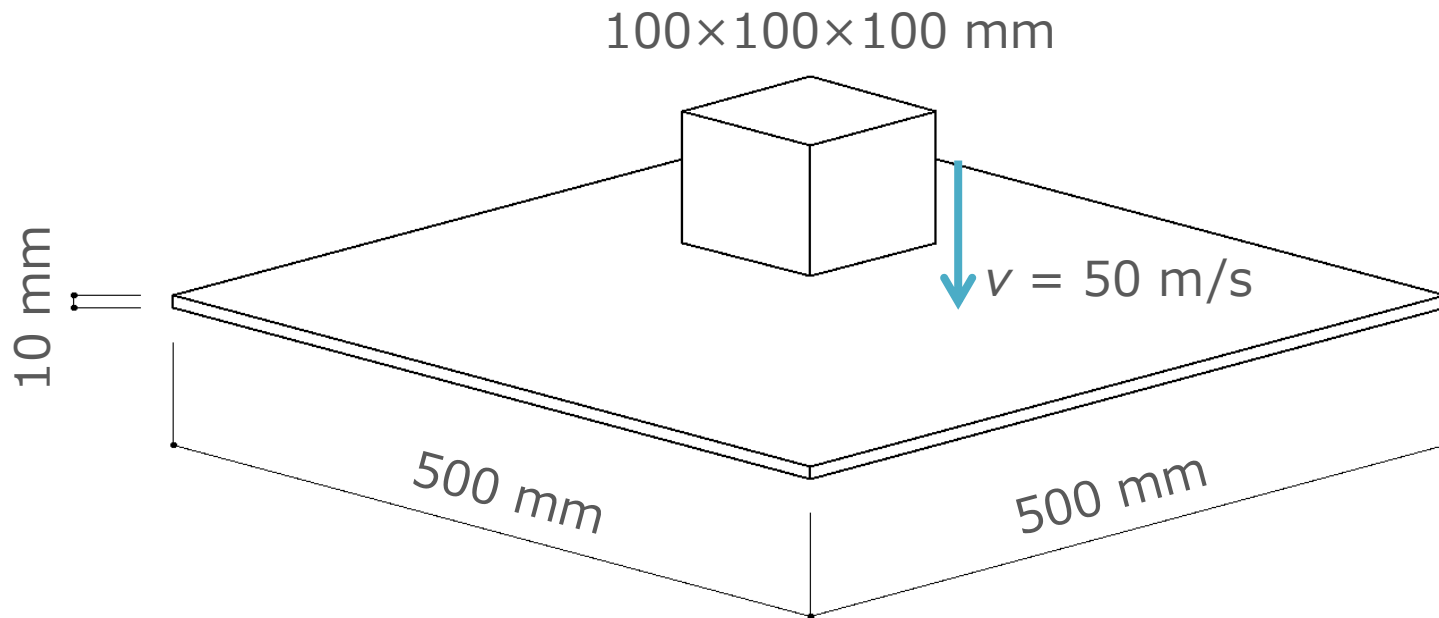
日本語と**英語**

の切り替えが可能



**海外ユーザーにも
FRAXSTが利用される**

平板の衝突破壊解析の例題



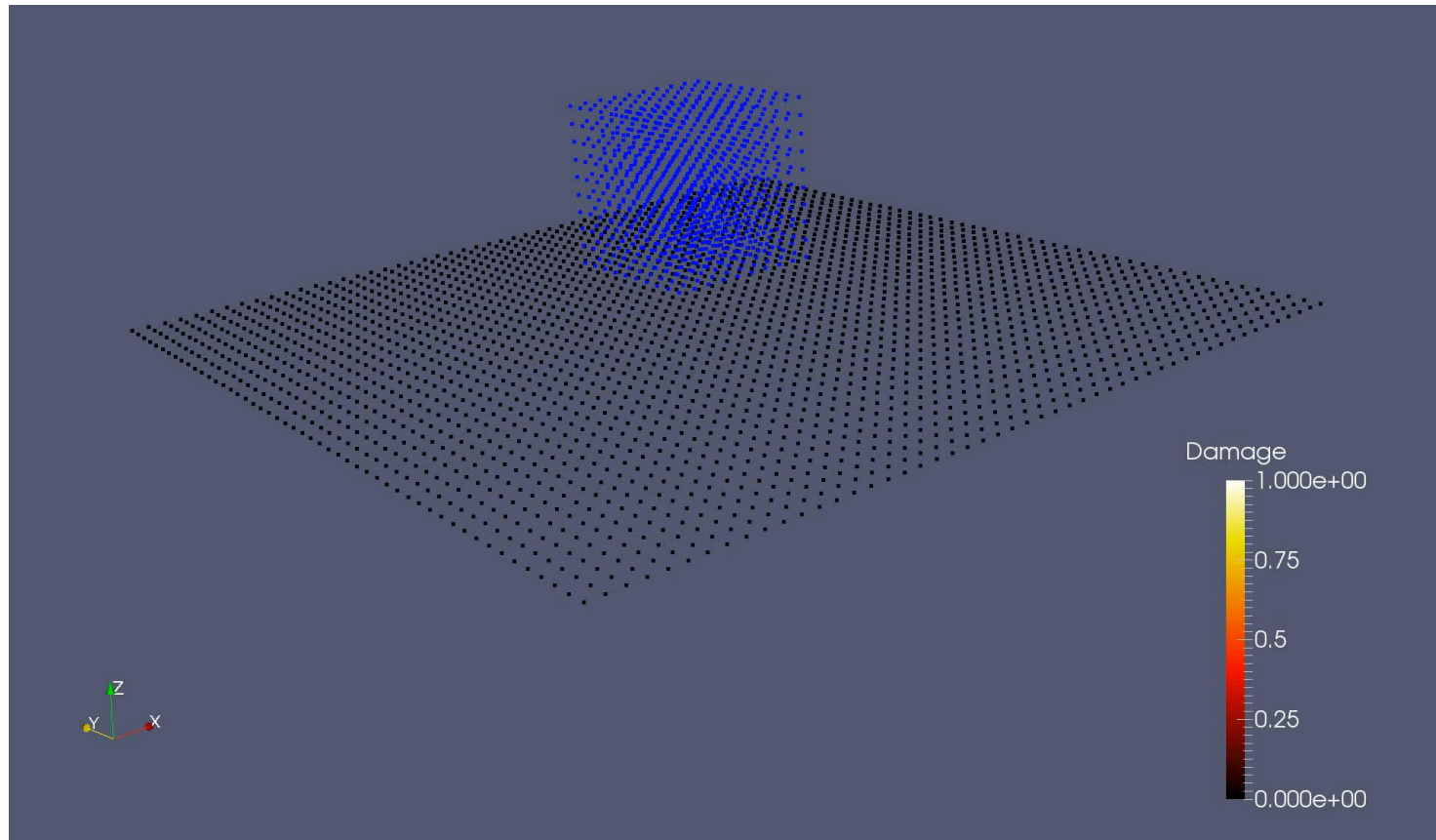
材料 : 2物体ともアルミニウム

粒子数 : 3600

再現時間 : $0 \sim 0.003$ 秒間を 1.0×10^{-6} 刻みで出力

計算時間 : 約1分 (Intel®Core™i7-3820 8コア並列)

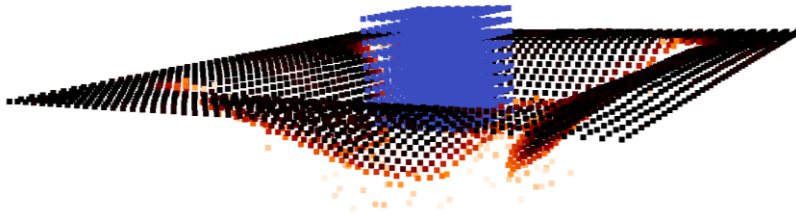
解析結果



解析結果

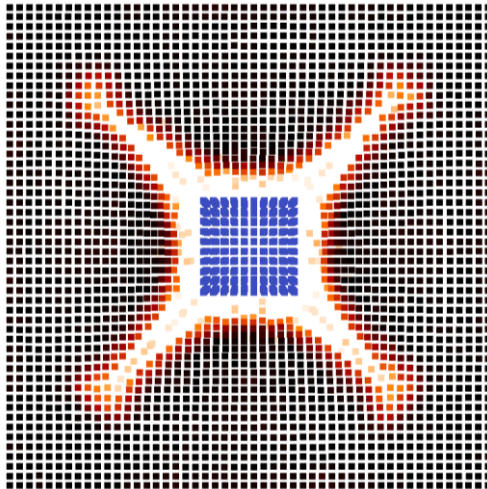
全体図

衝突により平板が破壊される



XY平面

対角線上に亀裂が入る



- 3D-CADから出力したUNVファイルから変換するため、モデリングが容易
- GUIによる操作性の向上
- 広く普及しているOSや言語で動作することでPeridigmを用いた研究の活性化