

2014/06/14
OpenCAE勉強会@岐阜

振動周波数応答計算機能の 各種オープンソースCAEソフト の調査状況その1

SH

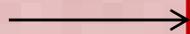
発表内容

- 動解析について
- 固有値解析とその他の線形振動・ 過渡解析
- 周波数応答解析について
- オープンソースCAE周波数応答解析手法
- 補足) Calculix linux install 方法
- まとめ

動的解析について

- a. 動的解析について：動的解析と静的解析の違いは、静的解析が慣性力を無視するのに対して、動的解析では慣性力項を考慮することである。ニュートン運動方程式を見れば違いは明瞭。
- b. 慣性力項を含まず、時間とともに物性値が変化する現象（応力緩和、粘弾性、クリープ）は動的解析とは区別して準静的問題という。

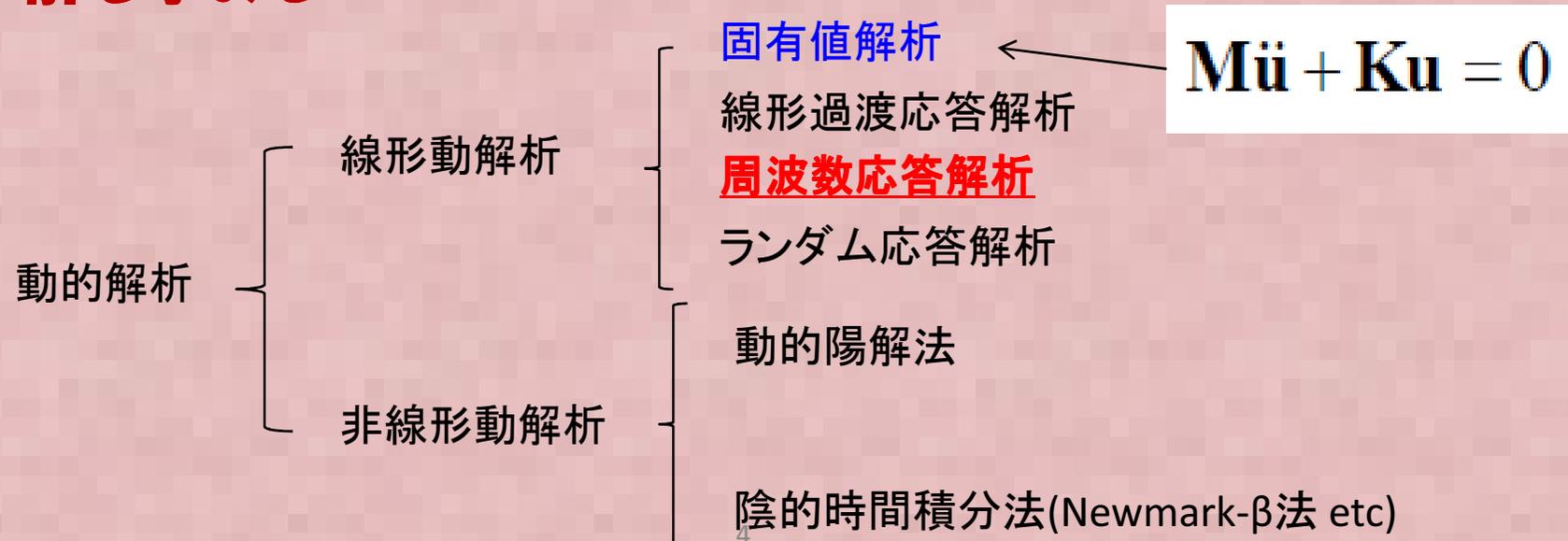
慣性力



$$M \frac{d^2 x}{dt^2} + Kx = F$$

動的解析について

- 動的解析の分類：動的解析は大きく非線形性（物性（速度依存etc）、接触など境界非線形）を考慮するか、しないかで大きく2種類に分類できる。
- 線形解析の場合は通常固有値計算を行い、この結果をベースに周波数領域で計算を行う。これに対して非線形解析の場合は直接時間積分を行い時間領域で解を求める



固有値解析とその他の線形振動・ 過渡解析の関係①

• 固有値計算とは? $M \frac{d^2 u}{dt^2} + Ku = F$

荷重 F が周期的三角関数で作用する場合

$$F(t) = F_0 e^{i\omega t} = F_0 (\cos \omega t + i \sin \omega t)$$

この場合、変位も同様に周期関数となることが想定される

$$u(t) = u_0 e^{i\omega t} = u_0 (\cos \omega t + i \sin \omega t)$$

加速度はこの場合

$$\frac{d^2 u(t)}{dt^2} = u_0 \frac{d^2 e^{i\omega t}}{dt^2} = -u_0 \omega^2 e^{i\omega t}$$

固有値解析とその他の線形振動・ 過渡解析の関係②

• 固有値計算とは?

$$M \frac{d^2 u}{dt^2} + Ku = F$$

運動方程式に代入し、両辺を $e^{i\omega t}$ で割る

$$(-\omega^2 M + K)u_0 = F_0$$

行列式 $\det(-\omega^2 M + K) \neq 0$ の場合は u_0 は自明解を持つ。
 $\det(-\omega^2 M + K) = 0$ の場合も解を持ち、この時の解が固有モード

$$(-\omega^2 M + K)u_0 = 0$$

または u_0 を x また ω^2 を λ とおくと

$$Kx = \omega^2 Mx = \lambda Mx \quad \rightarrow$$

上記の一般化固有値問題を解くのが固有値計算になる。

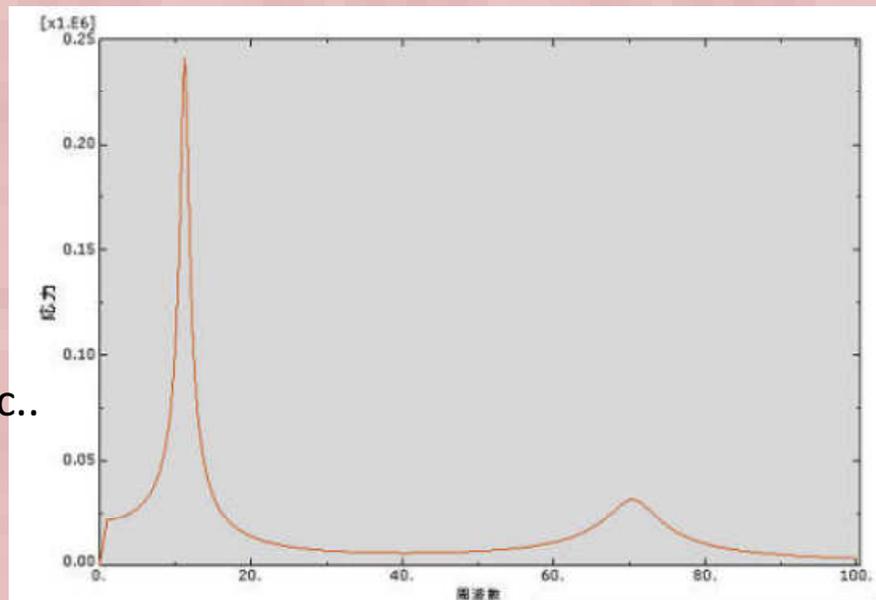
$\lambda = \omega^2$ が固有値 x は変位の固有ベクトルという。 ω は角速度
 $\omega(\text{rad/sec})$ は固有周波数 $f(\text{Hz})$ と $\omega = 2\pi f$ の関係がある

固有値の数値計算方法
-サブスペース法
-ランチョス法
-その他(べき乗法など)

周波数応答解析について

- 単純な正弦波($F=A\sin\omega t$)荷重 F に対する指定した周波数領域 ($\omega=2\pi f$ の関係で各速度 ω または周波数 f を変化させた場合)での応答(変位、速度、加速度、応力 など)を求める解析。以下のような周波数変化に対する、ある点の変位や加速度のカーブをアウトプットとすることが多い。

応力,
変位,
加速度 etc..



-通常は固有値解析の結果を利用するモード法が使われることが多い
-荷重の他に一定の強制加速度、強制変位で加振することができるソルバが多い(商用ソフトでは)

周波数(Hz)

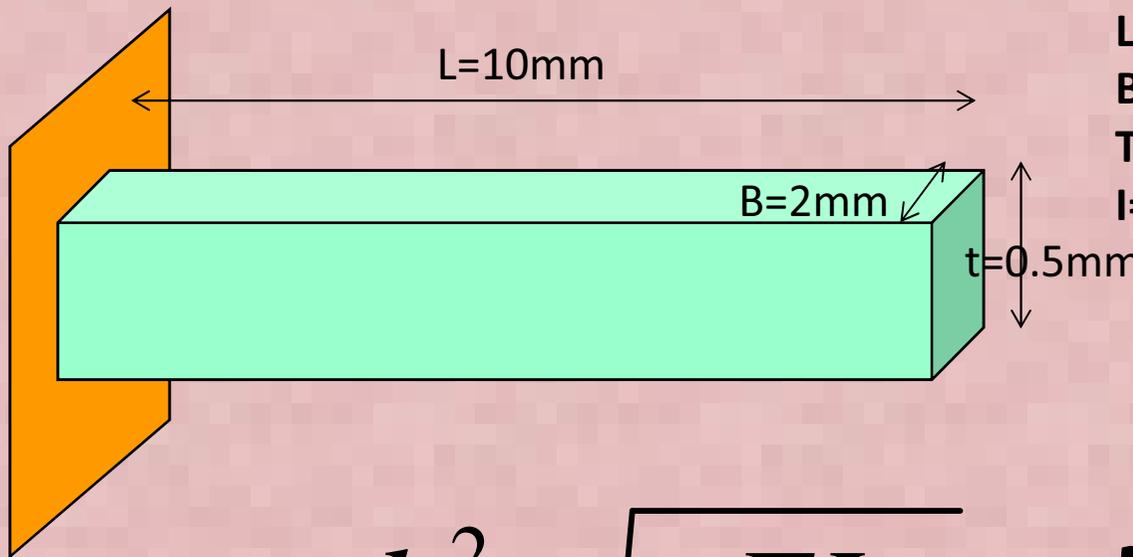
動的解析の可能なオープンソースCAEソフト

	線形動解析				非線形動解析		備考
	固有値	線形過渡応答解析	周波数応答解析	ランダム応答	動的陽解法	陰的時間積分法	
CodeAster	○	○	○	○?	△?	○	
Calculix	○	○	○	×	○	○	
Elmer	○	○	△	?	×	×	
FrontISTR	○	×	△	×	△	○	
Impact	×	×	×	×	○	×	
AdventureV2	○	—	○?	?	×	×	

固有値解析については、代表的なオープンソースCAEソフトにて解析が可能である。周波数応答解析も同様に計算できるようだが、今回はCalculix, FrontISTR, ABAQUS(Student Edition)にて周波数応答計算の比較を行い、それぞれのソフトでの計算結果・計算手順などをまとめた。

固有値計算理論解

- 以下の片持ちはりの固有値振動数を計算する。



$$E = 70000\text{MPa}$$

$$L = 10\text{mm}$$

$$B = 2\text{mm}$$

$$T = 0.5\text{mm}$$

$$I = bt^3/12 = 0.020833\text{mm}^4$$

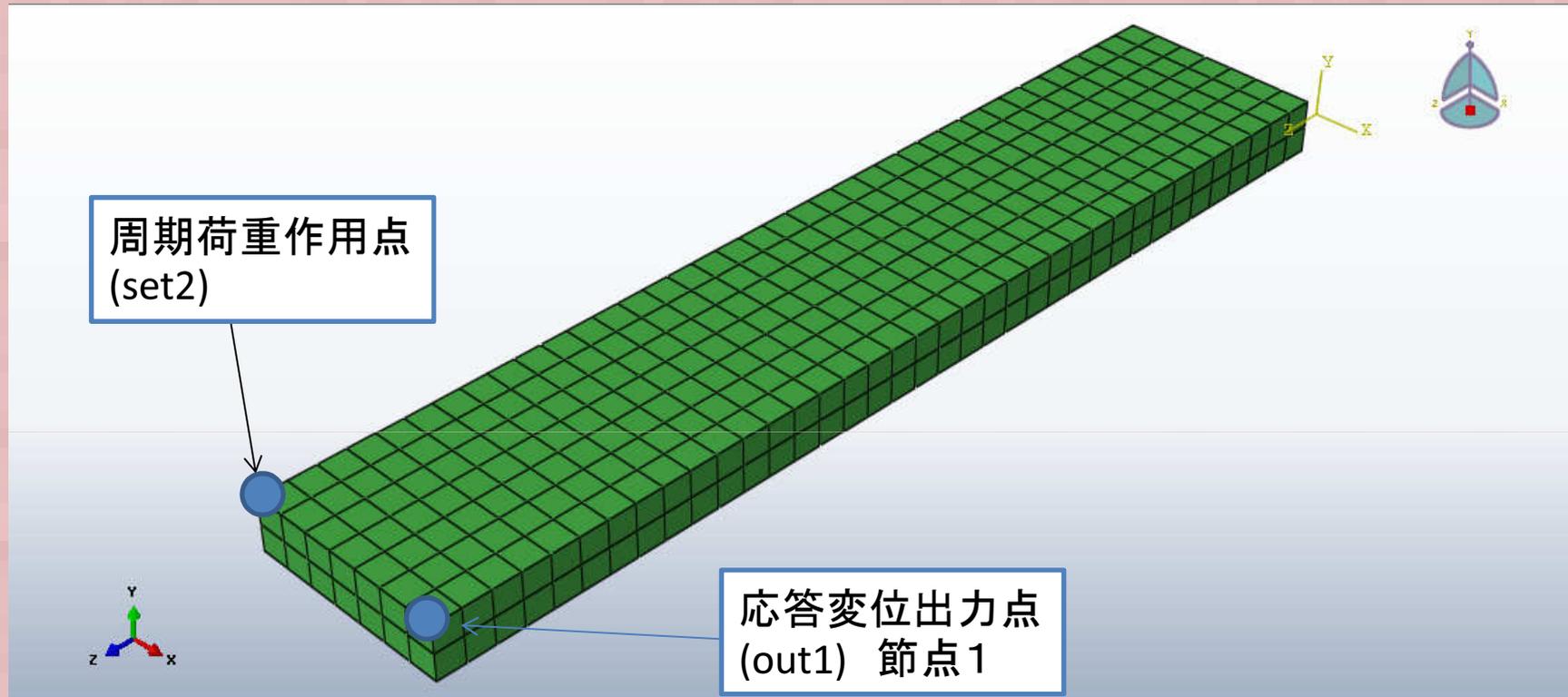
$$\text{密度} = 2.7 \times 10^{-9}$$

$$k = 1.875$$

$$f = \frac{k^2}{2\pi} \sqrt{\frac{EI}{\rho AL^4}} = 4114(\text{Hz})$$

1次固有振動数

周波数応答解析テストモデル概要



メッシュ概要

-節点数=912

-要素数=518

(要素:3D ソリッド)

-ABAQUS(商用ソフト)結果と比較するために、無料版のABAQUS

V6.12/studentedition でメッシュを作成, 計算した(計算できるのは1000節点まで)

固有値解析の梁のベンチマークモデルをそのまま流用

入力ファイル設定例① Step1 固有値解析部

- 以下はCalculixの解析ファイルの例です。

```
*Material, name=alumi
*Density
2.7e-09,
*Elastic
70000., 0.3
**
** BOUNDARY CONDITIONS
**
*Boundary
Set-1, 1,3, 0.0
**
**
** STEP: Step-1
*Step
*Frequency, STORAGE=YES
20, , , ,
*NODE PRINT, FREQ=9999, NSET=ALL
U
*NODE FILE, FREQ=9999, NSET=ALL
U
*End Step
```

密度

ヤング率

端点を固定

CalculixはAbaqusとほとんど同じ形式であり、1つの入力ファイル(input file)にテキスト形式で節点座標、要素コネクティビティ、材料物性、境界条件、解析条件を全て記述する

固有値解析を指示、Storage=Yes は Calculix 独特の指定(abaqus にはない) Step2 にて周波数応答解析する場合に固有値解析結果をDISKに保存するという指定 “入力ファイル名.eig” というファイルが同じフォルダに出力される 20は20モードまで計算して、出力する指定

結果出力指示

入力ファイル設定例② 周波数応答解析部

• 以下はCalculixの解析ファイルの例です。

```
** STEP: Step-2
*Step
*Steady State Dynamics
0., 50000., 20, 3.
*Modal Damping, Direct
1, 20, 0.1
** LOADS
** Name: Load-1 Type: Concentrated force
*Clload,
Set-2, 2, -1.
**
*NODE PRINT, FREQ=9999, NSET=out1
U
*NODE FILE, FREQ=9999, NSET=ALL
U
*End Step
```

周波数応答解析(Steady State Dynamics) を指定
0-50000Hz の範囲の応答を計算する、20 は各固有値間の
データ点数でデフォルト値はabaqusもCalculix でも20個。
データ点数を減らすと計算時間が早くなる。3はバイアスパ
ラメータだが、詳細は不明通常はデフォルトの3を使用

ダンピング係数、Direct は直接減衰率を指定
モードごとに減衰係数を変化させることもできる
ここでは1-20モード均一で0.1を指定

結果出力指示
Out1 点の変位(U) を出力

荷重点(set2: この例題では梁先端点(片側)を指定

各種オープンソースソルバでの周波数応答解析の実施方法が書いてある場所

- Calculix → <http://www.str.ce.akita-u.ac.jp/kako/j2011/sudo.html#i9>
または [この資料](#)
- Salome-meca/Code-Aster →
前田さんホームページ:
https://sites.google.com/site/codeastersalomemeca/home/code_aster-1/shuuhasuu

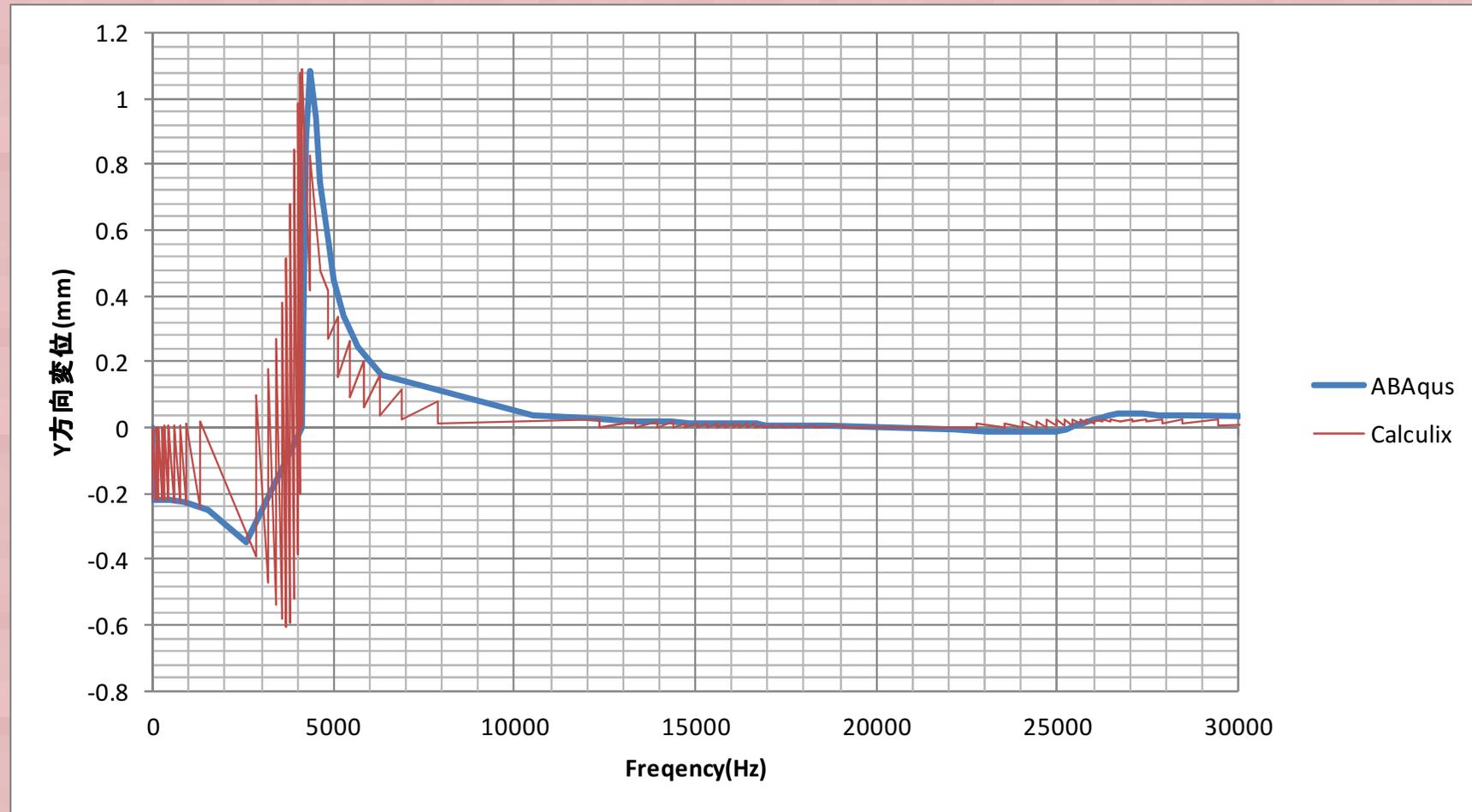
柴田先生のOpenCAE wiki にある藤井さん資料

<http://opencae.gifu-nct.ac.jp/pukiwiki/index.php?SALOME-Meca%A4%CE%BB%C8%CD%D1%CB%A1%B2%F2%C0%E2>

- FrontISTR → FrontISTRに同封されているチュートリアルガイドのP.46
- 参考: ABAQUS(student edition) での実施方法については下記参照 →
<http://jikosoft.com/cae/abaqus/Abaqus16.html>

Calculix/ABAQUS解析結果比較

応答点 (OUT1) のY方向変位を比較する

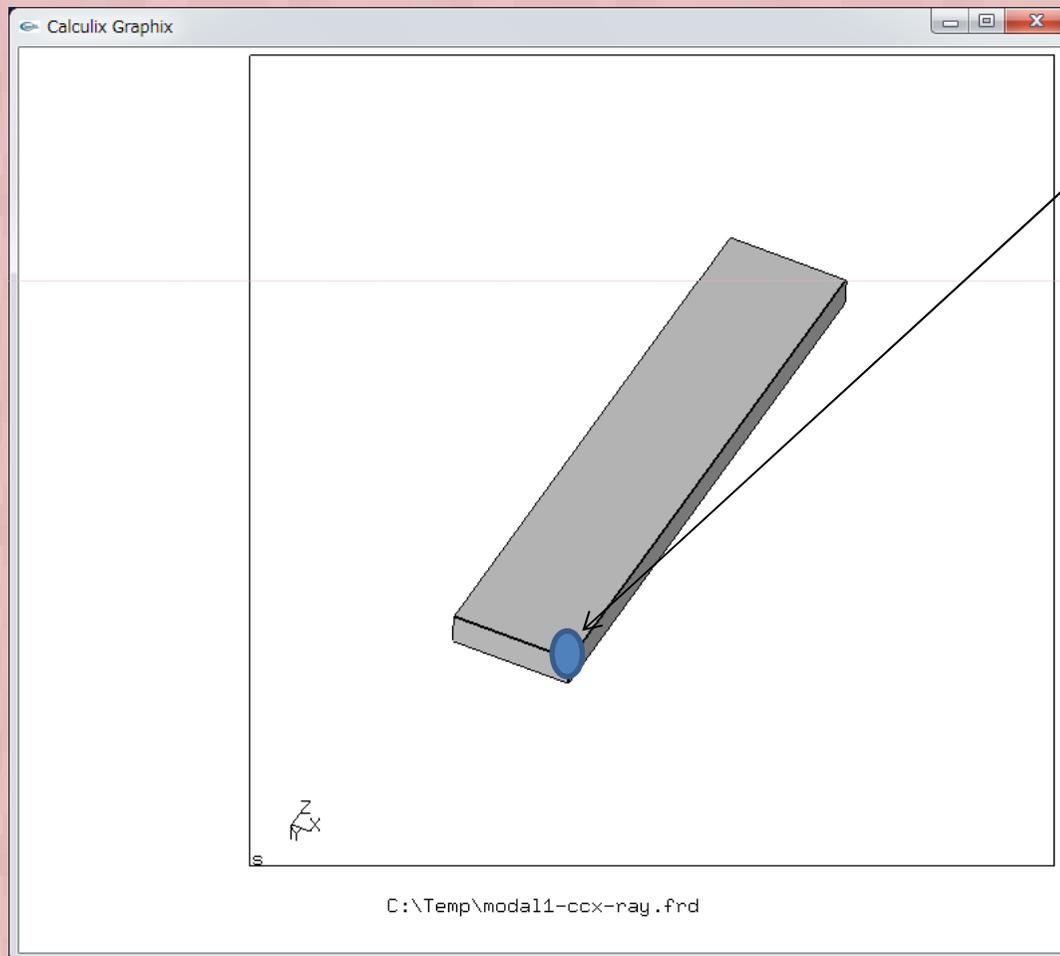


Calculix の変位が振動しており、おかしい。Calculix の変位の出力させ方 (ポスト処理?) の問題か？

(補足)Calculix解析結果 グラフデータ出力方法

応答点 (OUT1) の変位 (XYZ各方向、合成 (Magnitude) をCalculixのポストcgxから出力する方法を記載する

Calculix計算終了にできる “**”.frd という拡張子のファイルをダブルクリックすると以下のGUIが起動する(Windows 環境の場合)



- ① マウスカーソルをモデルが表示されているウインドウに移動させる
- ② 画面上で"qadd set" とコマンドを入力
- ③ マウスカーソルを応答を出力したい節点の上に移動させ、キーボード "n" を押す (節点=NODEをset グループに指定する)
- ④ cgx を起動したDOS画面にて指定した節点の番号、座標情報が出力されているので、間違いがないか確認する
- ⑤ キーボードから"q" を入力 (set への入力を完了)
- ⑥ コマンドを入力
"graph set t DISP ALL"
フォルダにgraph_set_DISP_ALL.out というテキストファイルが出力される (各成分を指定する場合はALL のかわりに D1, D2, D3などを指定する)

入力ファイル設定例① Step1 固有値解析部

- 以下はFrontISTRの解析条件ファイルの例です。

```
# Control File for FISTR
```

```
!VERSION
```

```
3
```

```
!WRITE,RESULT
```

```
!WRITE,VISUAL
```

```
!SOLUTION, TYPE=EIGEN
```

```
!EIGEN
```

```
20,
```

```
!! 1.0E-8, 60
```

```
!BOUNDARY
```

```
Set-1, 1, 3, 0.0
```

```
!SOLVER, METHOD=CG, PRECOND=1, ITERLOG=NO, TIMELOG=YES
```

```
10000, 2
```

```
1.0e-8, 1.0, 0.0
```

```
!VISUAL, method=PSR
```

```
!surface_num=1
```

```
!surface 1
```

```
!output_type = COMPLETE_REORDER_AVS
```

```
!END
```

FrontISTRもAbaqusとほとんど同じ形式であるが、3つの入力ファイル(msh, cnt, hecmw_cntl.dat)が必要。
テキスト形式でMSHファイルは節点座標、要素コネクティビティ、材料物性
CNT ファイルは境界条件、解析条件を記載

固有値解析を指示

20は20モードまで計算して、出力する指定

端点を固定

マトリックスソルバを指示

CGはCG法反復ソルバ

可視化結果出力指示

ParaViewで読めるAVS形式で出力

入力ファイル設定例② 周波数応答解析部

- 以下はFrontISTRの解析ファイルの例です。

```
!VERSION
3
!WRITE,RESULT
!WRITE,VISUAL
!SOLUTION, TYPE=DYNAMIC
!DYNAMIC
11, 2
1, 50000, 200, 15000.0
0.0, 6.6e-5
1, 1, 0.0, 1.0e-5
10, 2, 1
1, 1, 1, 1, 1, 1
!EIGENREAD
eigen_0.log
1, 20
!BOUNDARY
Set-1, 1, 3, 0.0
!FLOAD, LOAD CASE=1
Set-2, 2, 1.
!SOLVER,METHOD=CG,PRECOND=1,ITERLOG=NO,TIMELOG=YES
10000, 2
1.0e-8, 1.0, 0.0
!VISUAL, method=PSR
!surface_num = 1
!surface 1
!output_type = COMPLETE_REORDER_AVS
!END
```

周波数応答解析を指定 (DYNAMIC 11,2)

1-50000Hz の範囲の応答を計算する、200 は出力データ点数

ダンピング係数、FrontISTRではレーリー減衰しか指定できない

おおよそABAQUSなどの結果とオーダを合わせるため、 $\alpha=0$, $\beta=1e-5$ で設定してみた。

FrontISTRは固有値解析とは別解析として周波数応答解析を実施する必要がある。
2度解析を実施する手間がかかる

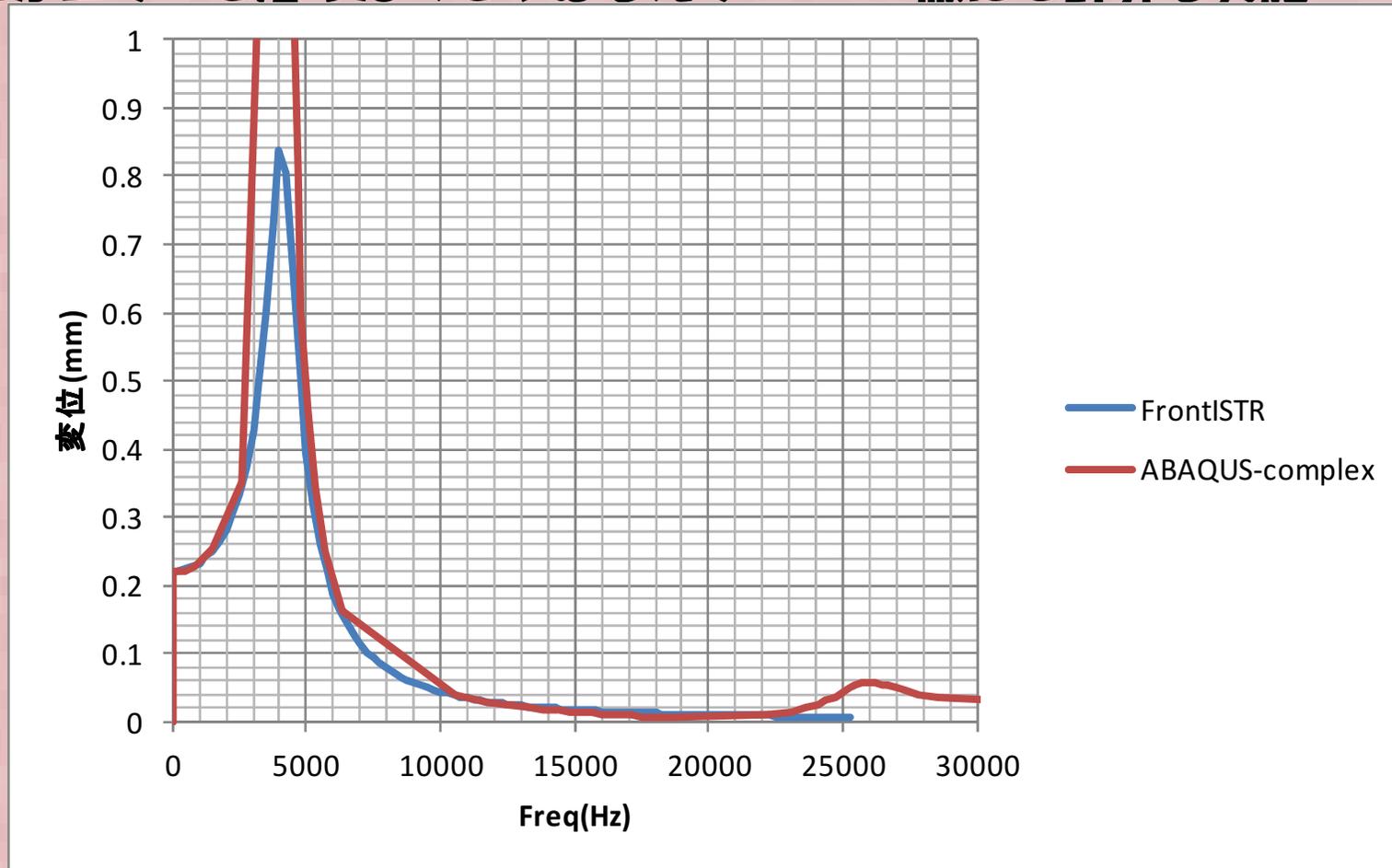
荷重点(set2: この例題では梁先 endpoint(片側)を指定
CASE=1は実部, CASE=2虚部

FrontISTRは荷重での加振以外はできない
(加速度での加振機能は無い)

加速度で与えたい時は大質量法を使えば良い?

FrontISTR/ABAQUS解析結果比較

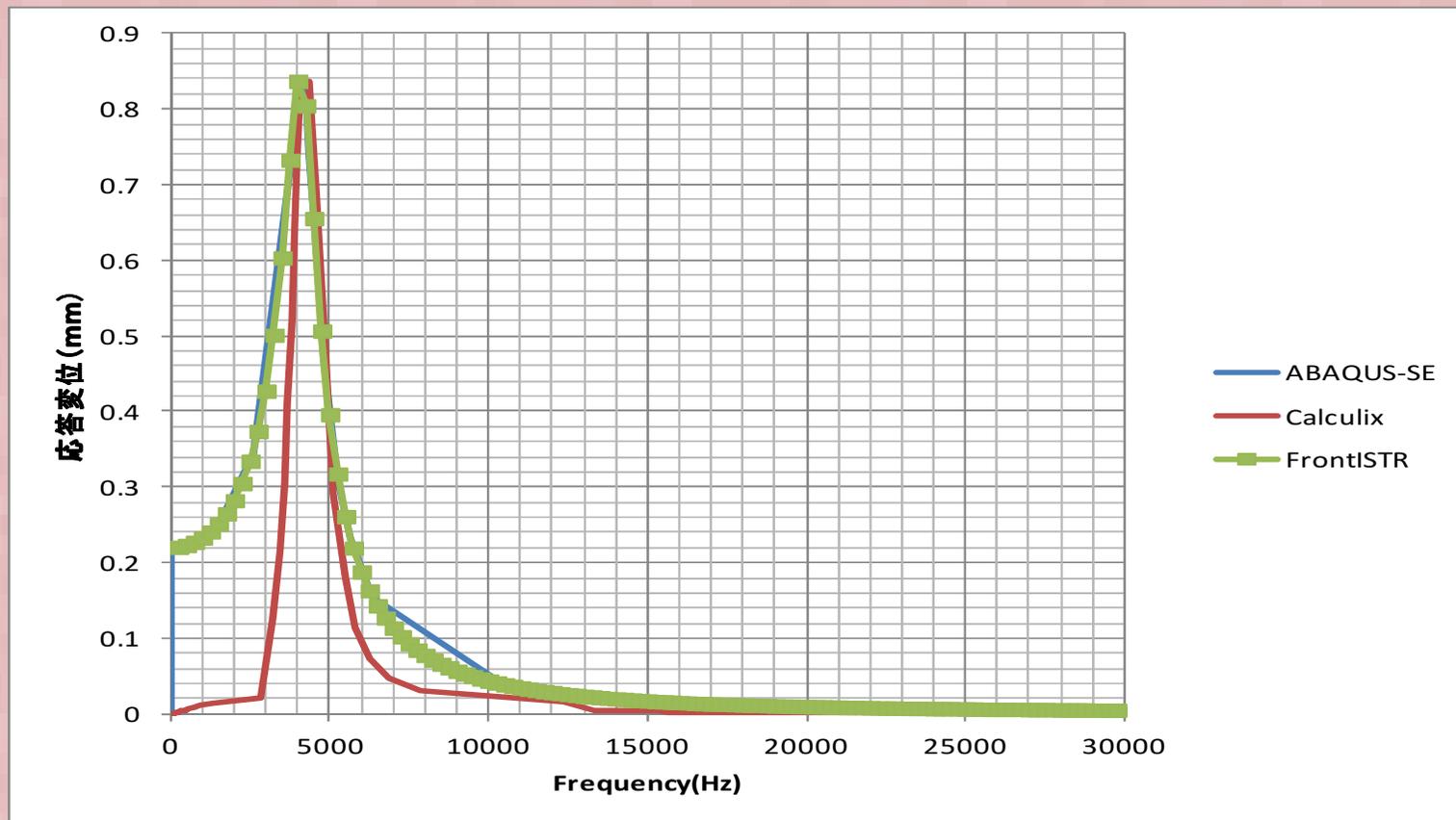
応答点 (OUT1) の変位 (大きさ) を比較する→ FrontISTRでは大きさしか出てこない模様、なおLinux版では固有値結果ファイルの読み込みでなぜかエラーで落ちたのでやむをえずWindows版にて計算を実施



おおよそ傾向は一致するが、値が違うのはFrontISTRとABAQUSで減衰係数の値が違うためと思われる。ABAQUSも同様にレーリー減衰を設定すれば同じになると思われる

FrontISTR/ABAQUS/Calculix解析結果比較

その後、FrontISTRのレーリー減衰に他のソルバも再度あわせ、応答点 (OUT1) の変位 (大きさ) を比較した。Calculixで振動していたように見えたのは同じ時間で実部と虚部が出力されていたためと判明、簡単なプログラムを作成して、大きさ ($\sqrt{\text{実部}^2 + \text{虚部}^2}$) を計算させた



ピークの応答は3ソルバで一致したが、FrontISTRとABAQUSでほぼ同じ上昇カーブを描くが、Calculixでは、途中の傾向が異なるが原因は不明です。

補足) Calculix linux install 方法①

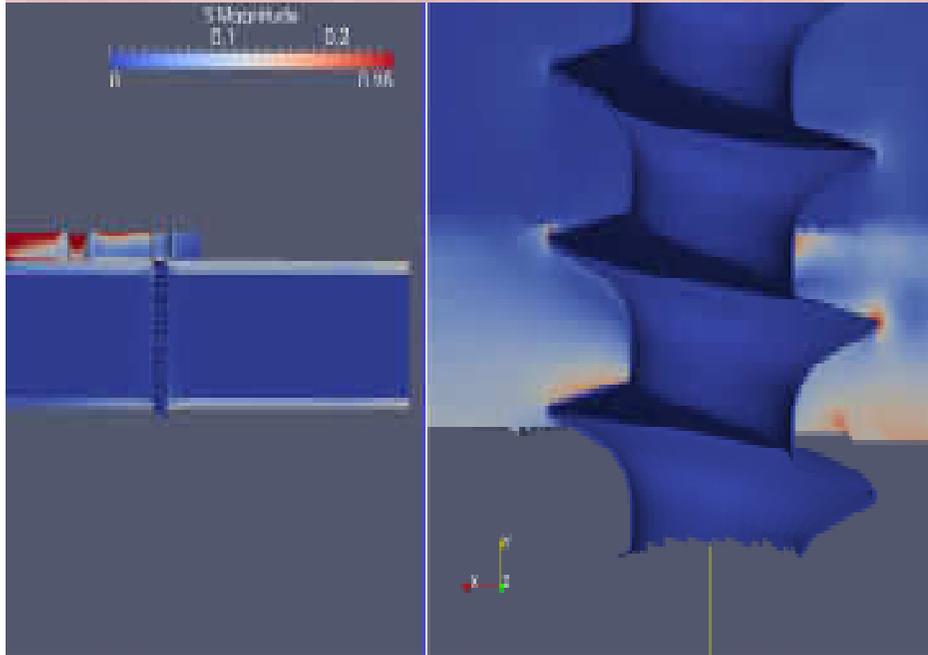
- CalculixはWindows版はインストールが簡単なため、もっぱらWindows版を利用していたが、行列ソルバの設定に問題があって、大規模な問題が解けない、また単一CPU(シングルコア)でしか計算できないという問題があり、Linux環境でのinstallを試してみた。

Calculixのダウンロードページ

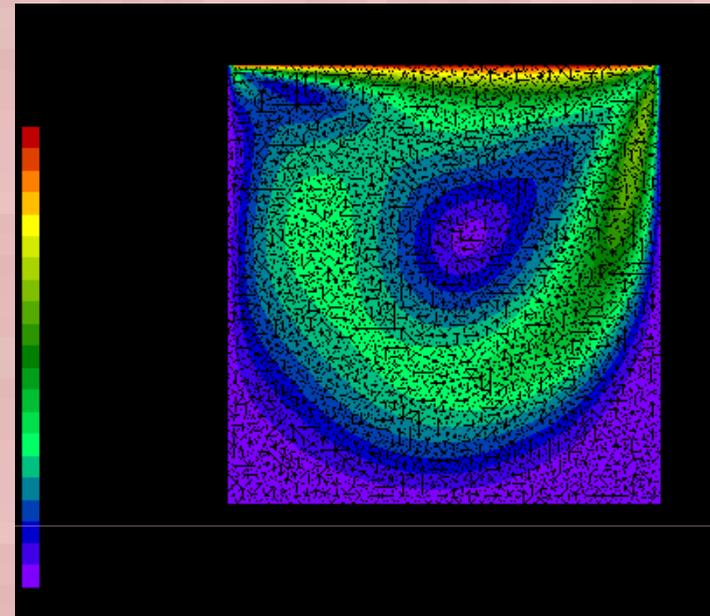
<http://www.dhondt.de/>

のccxを見て a short installation guideをクリックすると、インストールの簡単なマニュアルが出てくる(英語だが)。基本的にはこれにそって行えばよいが、実はうまくいかない。

Calculix



Calculix Extras
project 解析事例



Cavity FLOW in Calculix

- 商用ソフトABAQUSと同様の入力書式をもつオープンソース ABAQUSを仕事で使っている人は文法を勉強しないでそのまま使える。知らない人もABAQUSのマニュアルを見れば大体使い方が分かる。(テキスト入力ベースのモデラー、メッシャー、ソルバ、POSTを包含した非線形構造解析ソフト、一部流体解析も可能)
- <http://www.bconverged.com/calculix/> Windowsの実行バイナリを公開
- 非線形(大変形、接触解析、材料非線形(塑性、クリープ、温度依存etc)が可能
- 課題; 標準設定ではあまり大規模な計算(100万メッシュ以上?)には対応していない。
使用している行列ソルバSPOOLS がシングルコアで実行するように設定されているため

補足) Calculix linux install 方法②

- SPOOLES.2.2 と ARPACK のライブラリが必要でこれらのコンパイルが面倒

- <http://www.libremechanics.com/?q=node/9>

の方法でコンパイルするとうまくいく(英語)↓

Then Download this files:

CCX
CGX (executable)
Spooles 2.2
ARPACK
ARPACK PATCH

Unpack all these files in /usr/local folder making overwriting the content of ARPACK folder with PATCH.

Gcc と Gfortran の事前インストールが必要

3. Now compile SPOOLES, we should make some changes before we use MAKE, the changes are:

In /usr/local/spooles.2.2/Tree/src/makeGlobalLib change:

drawTree.c to draw.c

In /usr/local/spooles.2.2/Make.inc change:

CC = /usr/lang-4.0/bin/cc to CC = /usr/bin/gcc

On spooles.2.2 folder make:

```
$ sudo make lib  
$ cd MT/src/  
$ sudo make
```

Spoolsをコンパイルする時に、<http://www.dhondt.de/>に大規模問題用にSpoolsの修正部分のソースファイルが上がっているのでこれに差し替えると大規模問題でも計算できる

3. Now compile ARPACK, we should make some changes before we use MAKE, the changes are:

In /usr/local/ARPACK/ARmake.inc change:

home = \$(HOME)/ARPACK to home = /usr/local/ARPACK

PLAT = SUN4 to PLAT = linux

FC = f77 to FC = gfortran

FFLAGS = -O -cg89 to FFLAGS = -O2

MAKE = /bin/make to MAKE = /usr/bin/make

In /usr/local/ARPACK/UTIL/second.f change:

EXTERNAL ETIME to *EXTERNAL ETIME

On ARPAK folder make:

\$ sudo make lib

3. Finally compiling CalculiX:.

In /usr/local/CalculiX/ccx_2.6/src lets make sure that Makefile ist exatly like this:

```
CFLAGS = -Wall -O3 -I ../.././spooles.2.2 -DARCH="Linux" -DSPOOLES -DARPACK -  
DMATRIXSTORAGE -DUSE_MT=1
```

```
FFLAGS = -Wall -O3
```

```
CC=cc
```

```
FC=gfortran
```

~ 中略 ~ → このホームページはV2.6をベースに書かれているが
V2.7以降は MakefileMT(マルチスレッド版)のMakefileがあるのでこれをそのまま
使えば問題ないはず。これを使うとスレッド並列版モジュールができる

```
ccx_2.6_MT: $(OCCXMAIN) ccx_2.6_MT.a $(LIBS)
```

```
./date.pl; $(CC) $(CFLAGS) -c ccx_2.6.c; $(FC) -Wall -O3 -o $@ $(OCCXMAIN)
```

```
ccx_2.6_MT.a $(LIBS)
```

```
ccx_2.6_MT.a: $(OCCXF) $(OCCXC)
```

```
ar vr $@ $?
```

Finishing with:

```
$ sudo make
```

補足) Calculix linux install 方法③

- 並列計算使用時は以下の環境変数指定をおこなう
\$ export OMP_NUM_THREADS=#
(#の部分に2と指定すると2 並列で計算)
\$ ccx_2.7MT -i jobname
(入力ファイルを指定して計算を実施)
- Pump1次の固有値計算では1CPU計算時と比較して10秒程度計算が早くなった(元々40秒程度で終わる問題なので、あまり意味は無い?)
- 日本語でのInstall方法についての説明は少し古いが
<http://freecaetester.blog62.fc2.com/blog-entry-237.html> を参考にすると良い

報告まとめ

- 周波数応答解析についてABAQUS / *student*, *Calculix*, *FrontISTR*各ソルバについてベンチマークを行い、計算結果を比較した。
- おおよそ傾向が一致する結果となったが答えをチャントあわせるためには更なる修行が必要なようだ(引き続き調査予定、今回未実施のSalome-meca(*CodeAster*)とElmerでも実施してみます)。