

FrontISTRの梁要素／シェル要素の解説 ～プログラム解説を中心に～

東京大学
新領域創成科学研究科
人間環境学専攻
橋本 学

2016年4月22日

第27回 FrontISTR研究会

＜ユーザーとの情報交換・プログラム更新履歴のまとめ／
FrontISTRの構造要素 (シェル要素／梁要素)の解説＞

この講演では

- 『FrontISTRに実装されている定式化を十分に理解し、解きたい問題に対してソースコードを自由にカスタマイズ(要素タイプを追加, 材料の種類を追加, ユーザサブルーチンを追加)できるようにすること』を最終目標とします
- 第25回 (2月2日開催) では, FrontISTRに実装されている構造要素 (Bernoulli-Euler梁要素／MITCシェル要素) を解説しましたが, 予定していた構造要素プログラムの解説を終えることができませんでした
- 今回は, 構造要素のプログラムを中心に解説します

FrontISTRの梁要素／シェル要素の解説 ～プログラム解説を中心に～

目次

1. 準備

- FrontISTRのソースプログラムとインストール手順
- FrontISTRの入力ファイル
- 構造要素の要素タイプ番号
- 構造要素が使用できるFrontISTRの解析機能
- 構造要素が使用できるFrontISTRの材料

2. プログラム解説

- ディレクトリfistr1内のプログラム
- 構造要素を用いた場合のプログラムの流れ
- 構造要素に関連するプログラム
- Bernoulli-Euler梁要素の剛性マトリックス計算プログラム
- MITCシェル要素の剛性マトリックス計算プログラム

FrontISTRの梁要素／シェル要素の解説 ～プログラム解説を中心に～

目次

1. 準備

- FrontISTRのソースプログラムとインストール手順
- FrontISTRの入力ファイル
- 構造要素の要素タイプ番号
- 構造要素が使用できるFrontISTRの解析機能
- 構造要素が使用できるFrontISTRの材料

2. プログラム解説

- ディレクトリfistr1内のプログラム
- 構造要素を用いた場合のプログラムの流れ
- 構造要素に関連するプログラム
- Bernoulli-Euler梁要素の剛性マトリックス計算プログラム
- MITCシェル要素の剛性マトリックス計算プログラム

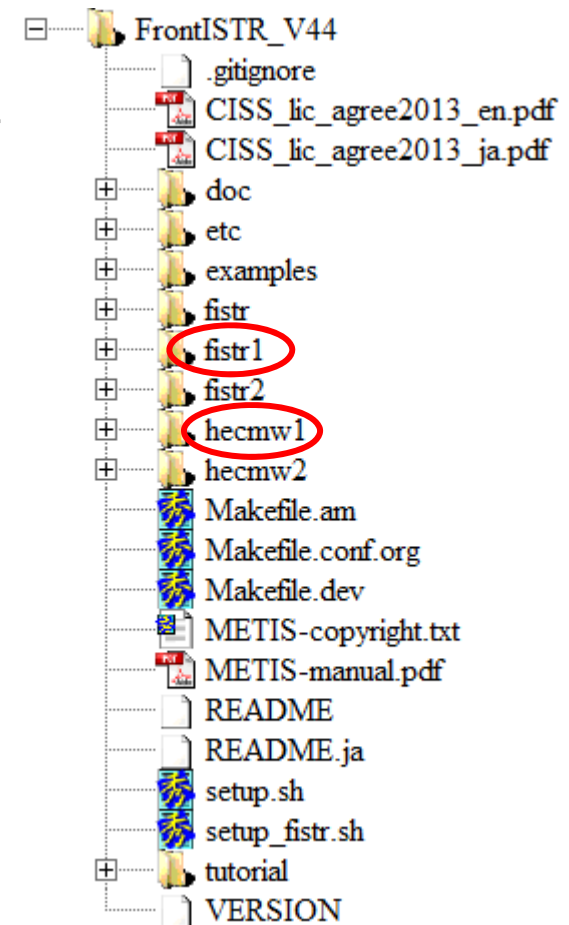
FrontISTRのソースプログラム

ソースプログラムFrontISTR_V44.tar.gzは、
FrontISTR研究会のホームページ (<http://www.multi.k.u-tokyo.ac.jp/FrontISTR/>) からダウンロードできます

ファイル
FrontISTR_V44.tar.gz

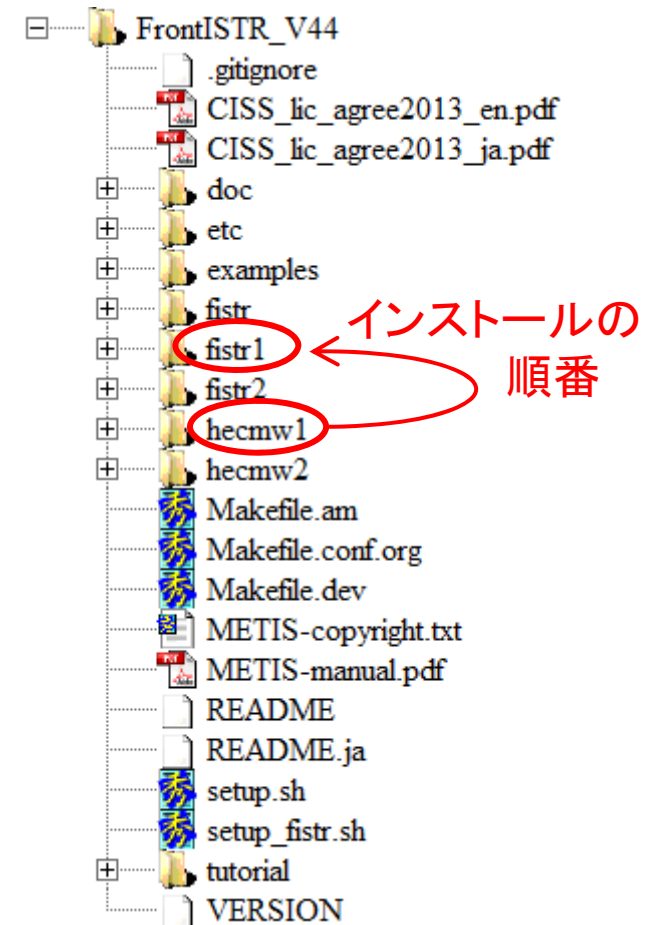


ディレクトリ
FrontISTR_V44



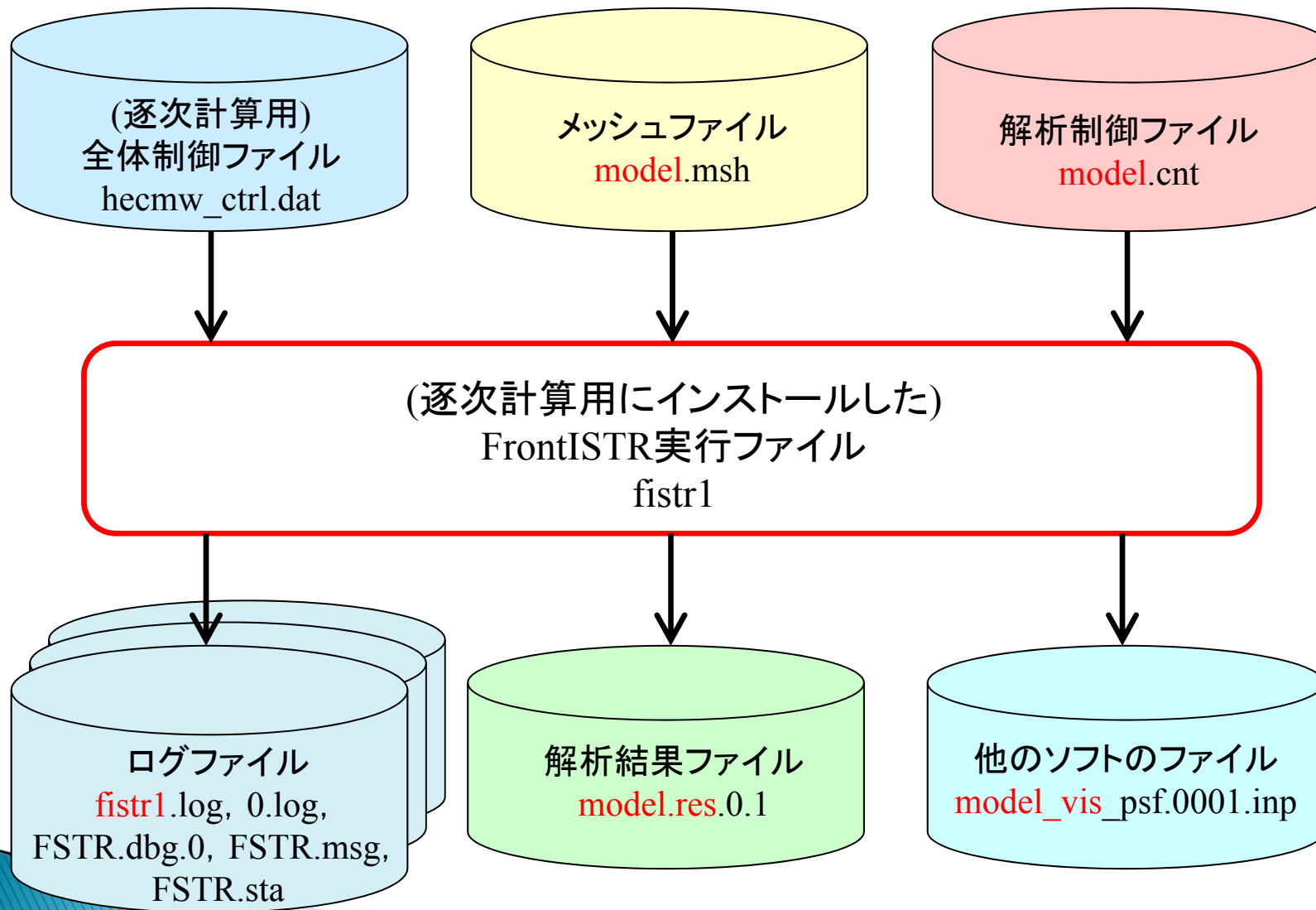
FrontISTRのインストール手順

- (1) ディレクトリhecmw1内のソースプログラムをコンパイルする
- (2) HEC-MWライブラリ(libfhecmw, libhecmw) が生成される
- (3) 実行ファイルhec2rcap, **hecmw_part1**, hecmw_vis1, ronv, rmergeが生成される
- (4) ディレクトリfistr1内のソースプログラムをコンパイルする
- (5) FrontISTRライブラリ (libffistr, libfistr) が生成される
- (6) 実行ファイル**fistr1**, neu2fstrが生成される



- ※ インストールオプションで--with-toolsを指定しない場合, hec2rcap, hecmw_part1, hecmw_vis1, ronv, rmergeは生成されません
- ※ fistr1がFrontISTRの実行ファイル, hecmw_part1がパーティショナの実行ファイルです

FrontISTRの逐次計算



※ 赤字の名前は自由に変更可能

全体制御ファイル (逐次計算)

(逐次計算用)
全体制御ファイル
hecmw_ctrl.dat

```
1 #↓
2 # for solver↓
3 #↓
4 !MESH, NAME=fstrMSH, TYPE=HECMW-ENTIRE↓
5   conrod.msh↓
6 !CONTROL, NAME=fstrCNT↓
7   conrod.cnt↓
8 !RESULT, NAME=fstrRES, IO=OUT↓
9   conrod.res↓
10 #↓
11 # for visualizer↓
12 #↓
13 !MESH, NAME=mesh, TYPE=HECMW-ENTIRE↓
14   conrod.msh↓
15 !RESULT, NAME=result, IO=IN↓
16   conrod.res↓
17 !RESULT, NAME=vis_out, IO=OUT↓
18   conrod vis↓
```

← メッシュファイル名の設定

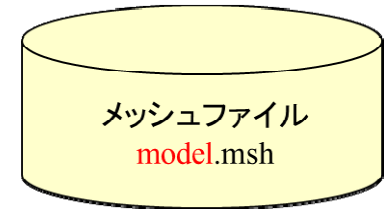
← 解析制御ファイル名の設定

← 解析結果ファイル名の設定

← その他のソフトのファイル作成の
ために入力するメッシュファイル
名と解析結果ファイル名の設定

← その他のソフトのファイル名の
設定

FrontISTRのメッシュファイル



```
1 !HEADER↓
2      conrod (2015/10/20) ↓
3 !NODE, NGRP=NODE_ALL↓
4 1, 1.5947469e-2, 1.5194545e-1, 0.0000000↓
5 2, 1.6402800e-2, 1.5406281e-1, -4.0000000e-3↓
6 3, 1.4900877e-2, 1.5004414e-1, -4.0000000e-3↓
7 4, 1.3912825e-2, 1.4809515e-1, -8.0000000e-3↓
8 5, 1.3486000e-2, 1.4375201e-1, 0.0000000↓
```

ヘッダー

節点の番号 I と
節点の座標値
(x^I, y^I, z^I)

```
96331 96328, -2.7018885e-2, 2.8273724e-2, 4.3311714e-3↓
96332 !ELEMENT, TYPE=342, EGRP=ELEMENT_ALL↓
96333 1, 11689, 11690, 11691, 11692, 69217, 69204, 69203, 69205, 69218, 69228↓
96334 2, 11689, 11693, 11694, 11691, 69250, 69207, 69206, 69204, 69229, 69230↓
96335 3, 11693, 11695, 11696, 11689, 69276, 69252, 69251, 69206, 69208, 69209↓
96336 4, 11697, 11693, 11695, 11696, 69251, 69277, 69253, 69288, 69252, 69276↓
96337 5, 11698, 1924, 11699, 11251, 28041, 69309, 28040, 67990, 28039, 67991↓
```

要素の番号 e と
要素内の
節点同士の
つながり
(コネクティビティ)

```
152447 56115, 8763, 11467, 3708, 2887, 36677, 36675, 57103, 33602, 33605, 33601↓
152448 !NGROUP, NGRP=U_FIXED↓
152449 1073↓
152450 1074↓
152451 1075↓
152452 1076↓
152453 1077↓
```

節点グループ

境界条件や集中荷重(外力)を与える

梁要素: TYPE=611
シェル要素: TYPE=741
TYPE=743

```
161751 68380 ↓
161752 !SECTION, TYPE=SOLID, EGRP=ELEMENT_ALL, MATERIAL=ALUMINUM↓
161753 !MATERIAL, NAME=ALUMINUM, ITEM=3↓
161754 !ITEM=1, SUBITEM=2↓
161755 0.0, 0.0↓
161756 !ITEM=2↓
161757 0.0↓
161758 !ITEM=3↓
161759 0.0↓
161760 !END↓
```

セクションデータ

材料データ

セクションの設定によって
要素グループに材料データが
与えられる

梁要素: TYPE=BEAM
シェル要素: TYPE=SHELL

ファイルの終わり

FrontISTRで使用可能な要素タイプ (1/2)

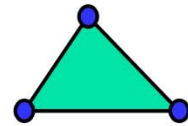
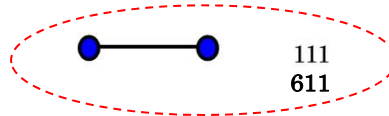
要素種類	要素タイプ番号	節点数	節点自由度数	説明
線要素	111	2	3	2節点リンク要素
	112	3	3	3節点リンク要素
平面要素	231	3	3	三角形1次要素
	232	6	3	三角形2次要素
	241	4	3	四角形1次要素
	242	8	3	四角形2次要素 (Serendipity族)
ソリッド要素	301	2	3	2節点トラス要素
	341	4	3	四面体1次要素
	342	10	3	四面体2次要素
	351	6	3	プリズム1次要素
	352	15	3	プリズム2次要素
	361	8	3	六面体1次要素
	362	20	3	六面体2次要素 (Serendipity族)
インターフェース要素	541	4×2	3	四角形面1次要素
	542	8×2	3	四角形面2次要素
梁要素	611*	2	6	2節点梁要素 (Bernoulli-Euler梁)
	641**	2×2	3	2節点梁要素 (Bernoulli-Euler梁)
シェル要素	731*	3	6	三角形1次要素 (MITC3シェル)
	761**	3×2	3	三角形1次要素 (MITC3シェル)
	741*	4	6	四角形1次要素 (MITC4シェル)
	781**	4×2	3	四角形1次要素 (MITC4シェル)
	743	9	6	四角形2次要素 (MITC9シェル)

※ 解析メッシュにソリッド要素, 梁要素, シェル要素が混在する場合,
節点自由度数を3に揃えるため, *ではなく**の要素タイプ番号を使用してください
(**の要素タイプ番号は後述します)

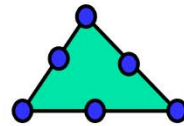
FrontISTRで使用可能な要素タイプ (2/2)

梁要素
600

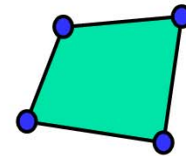
要素形状



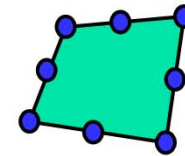
231



232



241

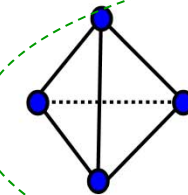


242

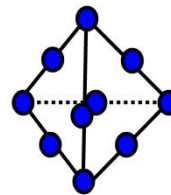
3次元ソリッド要素

300

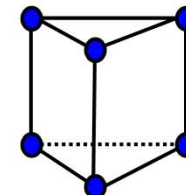
要素形状



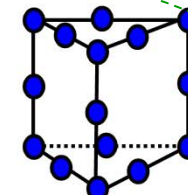
341



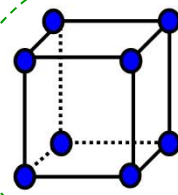
342



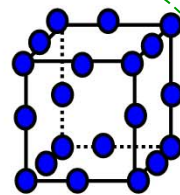
351



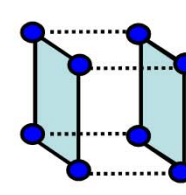
352



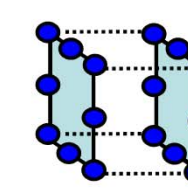
361



362



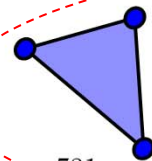
541



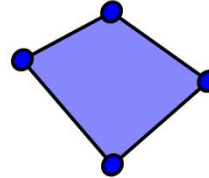
542

シェル要素
700

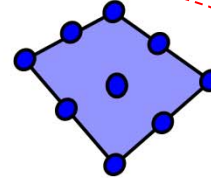
要素形状



731



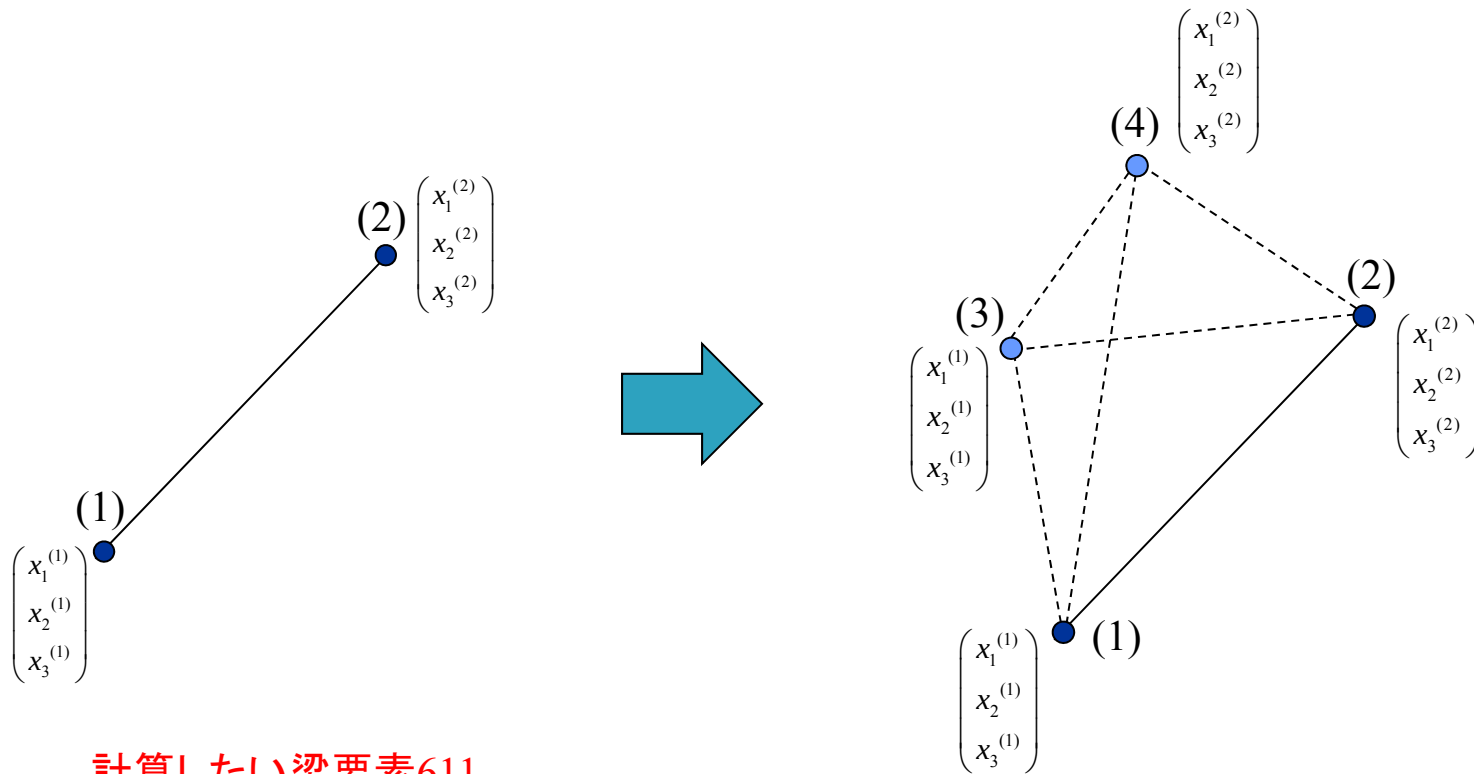
741



743

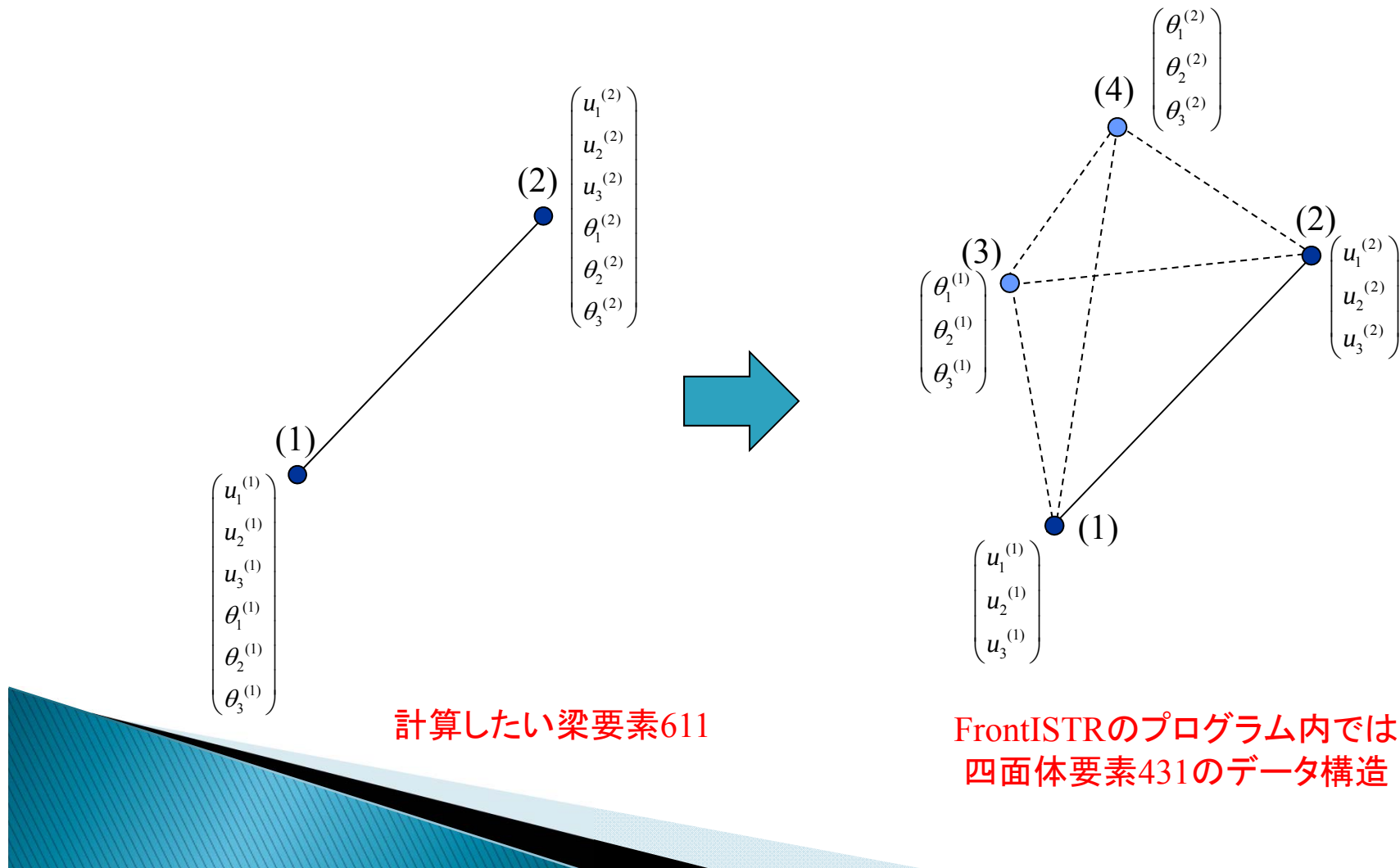
要素タイプ641 (1/3)

FrontISTRのメッシュファイルでは、節点座標を二つ用意する
(メッシュを表示する際に都合が良いためであるが、
計算には使用しないので (3) と (4) の節点座標はどのような値でも良い)



要素タイプ641 (2/3)

FrontISTRのプログラム内では, (3) と (4) の節点変位は梁の回転自由度となる



要素タイプ641 (3/3)

FrontISTRのプログラム内では、
梁要素の剛性マトリックス成分をテーブルに従って変更する

剛性マトリックス (梁要素611)

$$\mathbf{K} = \begin{pmatrix} K_{1,1} & K_{1,2} & \cdots & K_{1,11} & K_{1,12} \\ K_{2,1} & K_{2,2} & \cdots & K_{2,11} & K_{2,12} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ K_{11,1} & K_{11,2} & \cdots & K_{11,11} & K_{11,12} \\ K_{12,1} & K_{12,2} & \cdots & K_{12,11} & K_{12,12} \end{pmatrix}$$

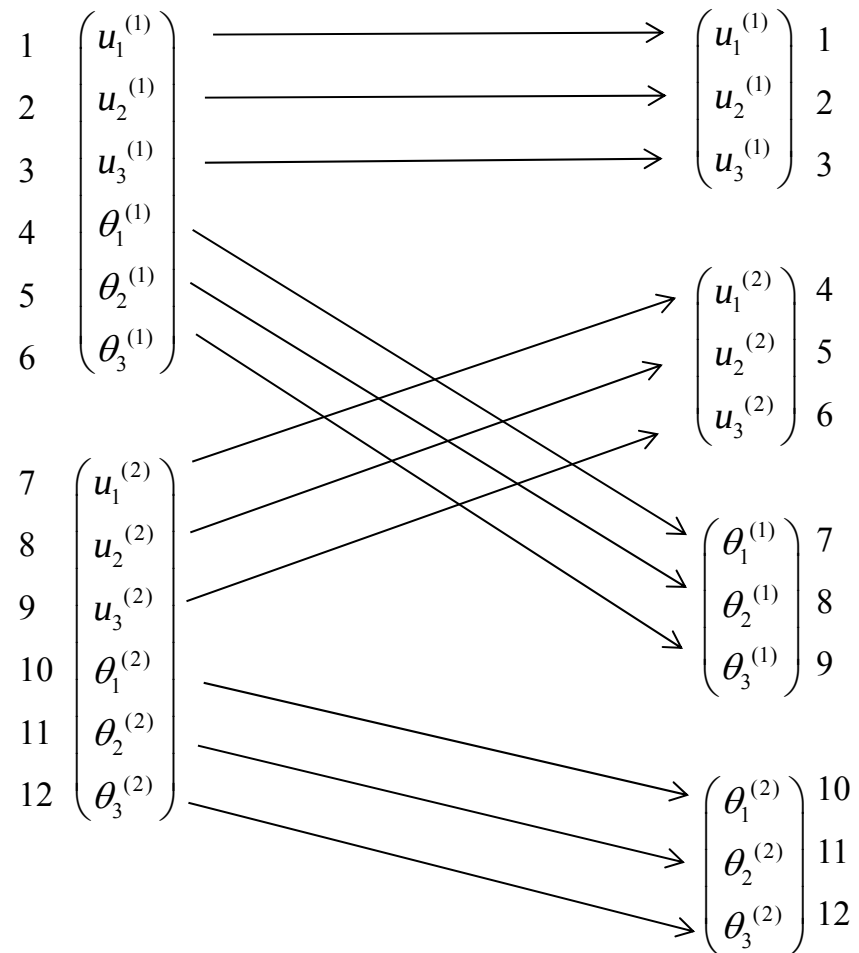


FrontISTRのプログラム内で計算する
剛性マトリックス

$$\mathbf{K}' = \begin{pmatrix} K'_{1,1} & K'_{1,2} & \cdots & K'_{1,11} & K'_{1,12} \\ K'_{2,1} & K'_{2,2} & \cdots & K'_{2,11} & K'_{2,12} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ K'_{11,1} & K'_{11,2} & \cdots & K'_{11,11} & K'_{11,12} \\ K'_{12,1} & K'_{12,2} & \cdots & K'_{12,11} & K'_{12,12} \end{pmatrix}$$

例えば, $K'_{4,8} = K_{7,5}$

テーブル



要素混在問題における3×3ブロックCSR形式での格納

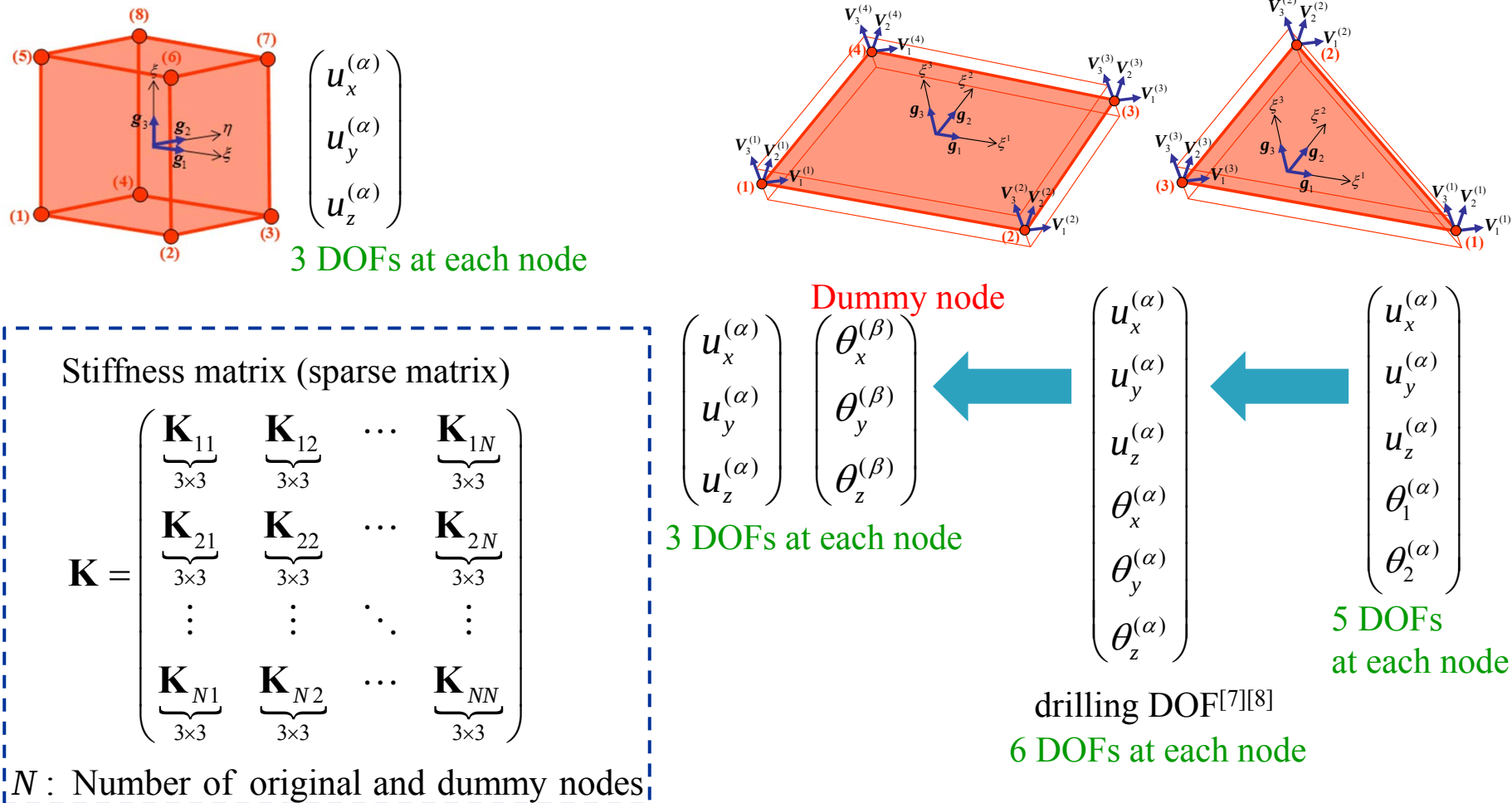


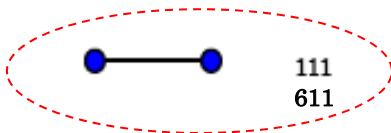
Fig. 3×3 BCSR format for solid and shell elements.

Hughes, T.J.R. and Brezzi, F., *Computer Methods in Applied Mechanics and Engineering*, Vol.72, pp.105-121, 1989.

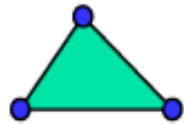
Nguyen-Van, Hieu and Mai-Duy, Nam and Tran-Cong, Thanh, *CMES: Computer Modeling in Engineering and Sciences*, Vol.49, No.2, pp.81-110, 2009.

梁要素の要素タイプ611と641

梁要素
600



111
611



231



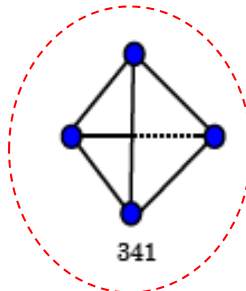
232



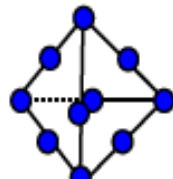
241



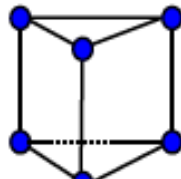
242



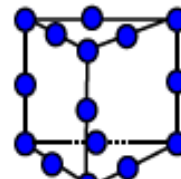
341



342

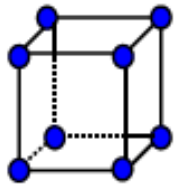


351

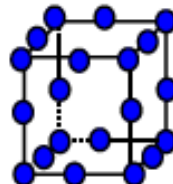


352

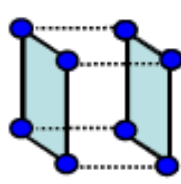
3次元ソリッド要素
300



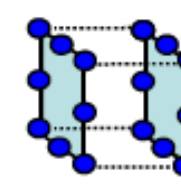
361



362

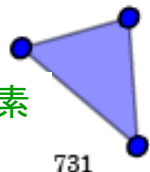


541

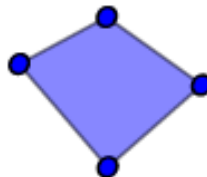


542

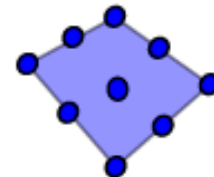
シェル要素
700



731



741



743

節点あたりの自由度

- ・ 3次元ソリッド要素:3
- ・ シェル要素:6
- ・ 梁要素:6

ソリッド要素と梁要素／シェル要素を混在させる場合、
節点あたり3自由度の
データ構造に統一して、
剛性マトリックスを作成し、
線形ソルバーに渡すようにする

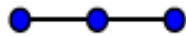
梁要素のデータ構造が
節点あたり3自由度を持つように、
四面体要素341のデータ構造を
利用した要素タイプ641を
導入する

シェル要素の要素タイプ361と761／要素タイプ351と751

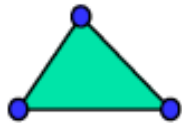
梁要素
600



111
611



112



231



232



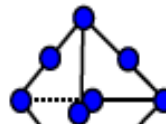
241



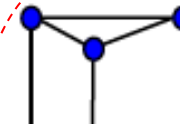
242



341



342

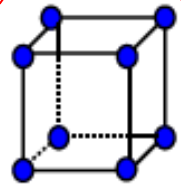


351

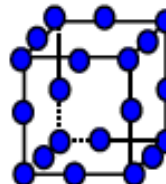


352

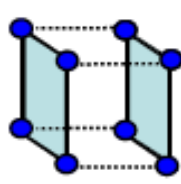
3次元ソリッド要素
300



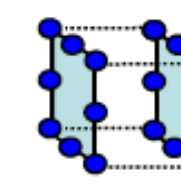
361



362

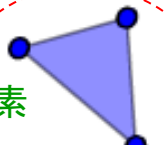


541

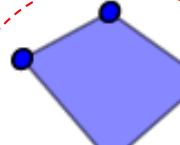


542

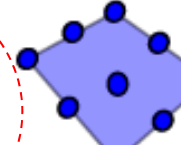
シェル要素
700



731



741



743

節点あたりの自由度

- ・ 3次元ソリッド要素:3
- ・ シェル要素:6
- ・ 梁要素:6

ソリッド要素と梁要素／シェル要素を混在させる場合、**節点あたり3自由度**のデータ構造に統一して、剛性マトリックスを作成し、線形ソルバーに渡すようにする

MITC4シェル要素のデータ構造が節点あたり3自由度を持つように、**六面体要素361 (351) のデータ構造を利用した要素タイプ761 (751) を導入する**

解析制御ファイル

解析制御ファイル
model.cnt

```
1 !VERSION↓
2 3↓
3 !SOLUTION, TYPE=STATIC↓
4 !WRITE, RESULT↓
5 !WRITE, VISUAL↓
6 !BOUNDARY↓
7 U_FIXED, 1, 3, 0.0↓
8 UY_GIVEN, 2, 2, 1.0e-3↓
9 !MATERIAL, NAME=ALUMINUM↓
10 !ELASTIC↓
11 7.0e10, 0.345↓
12 !SOLVER, METHOD=CG, PRECOND=3, ITERLOG=YES, TIMELOG=YES↓
13 5000000, 1↓
14 1.0e-8, 1.0, 0.0↓
15 !VISUAL, method=PSR↓
16 !surface_num = 1↓
17 !surface 1↓
18 !output_type = COMPLETE_MICROAVS↓
19 !END↓
```

← バージョン情報

← 解析機能の種類

← 解析結果ファイルの出力

← その他のソフトのファイルの出力

← 境界条件の設定

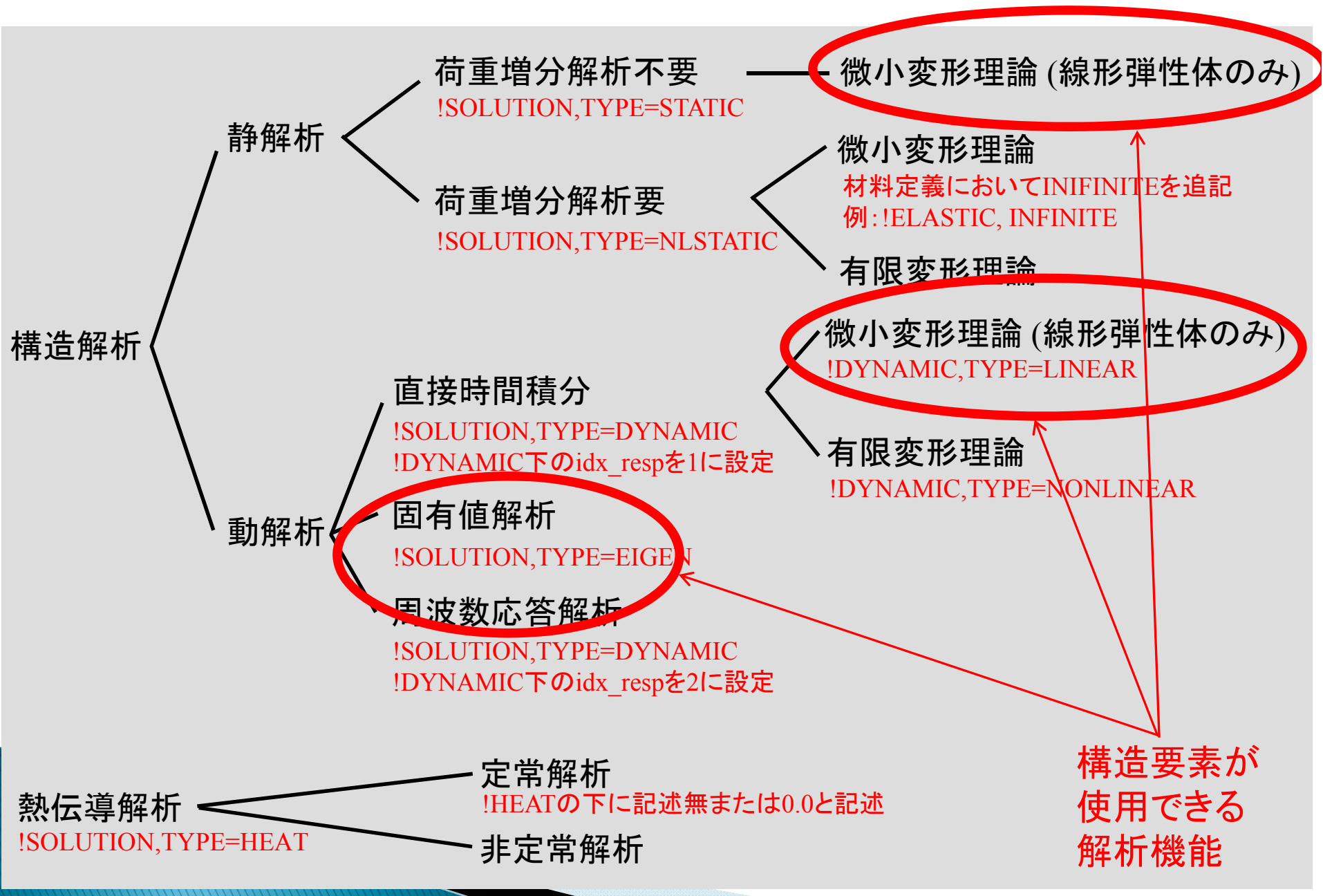
← 材料データの設定

← 線形ソルバーのパラメータの
設定

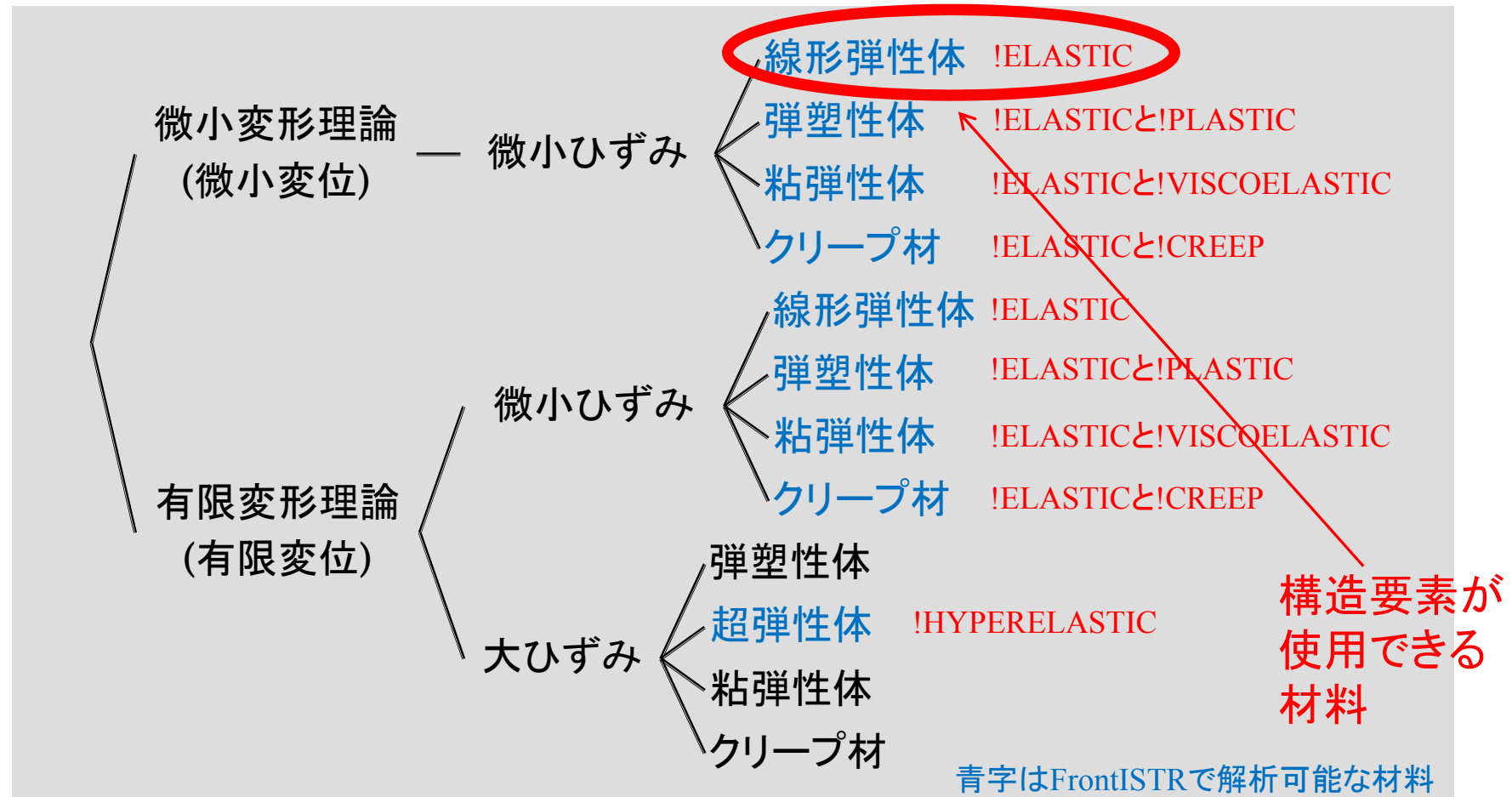
← その他のソフトのファイルの設定

← ファイルの終わり

解析機能の分類



材料の分類



有限変形理論

$${}^t_0\mathbf{E} = \frac{1}{2} \left\{ ({}^0\nabla \otimes {}^t\mathbf{u}) + ({}^0\nabla \otimes {}^t\mathbf{u})^T + ({}^0\nabla \otimes {}^t\mathbf{u}) \cdot ({}^0\nabla \otimes {}^t\mathbf{u})^T \right\}$$

ひずみ 変位こう配の2次項がある

大ひずみ

$${}^t_0\mathbf{S} = f({}^t_0\mathbf{E}, {}^t_0\mathbf{E} \cdot {}^t_0\mathbf{E}, \dots)$$

応力 ひずみの2次以上の項がある

記号の説明

t	時刻 [s]
N	次元 (3次元: $N = 3$)
Ω	有界領域 [m^N]
Γ	境界 [m^{N-1}]
$\mathbf{x}$	物質点の位置ベクトル [m]
∇	ナブラ [$1/m$]
$\mathbf{u}$	変位 [m]
$\mathbf{t}$	トラクション [Pa]
$\mathbf{b}$	単位質量当たりの体積力 [N/kg]
ρ	密度 [kg/m^3]
$\mathbf{n}$	外向き単位法線ベクトル [-]

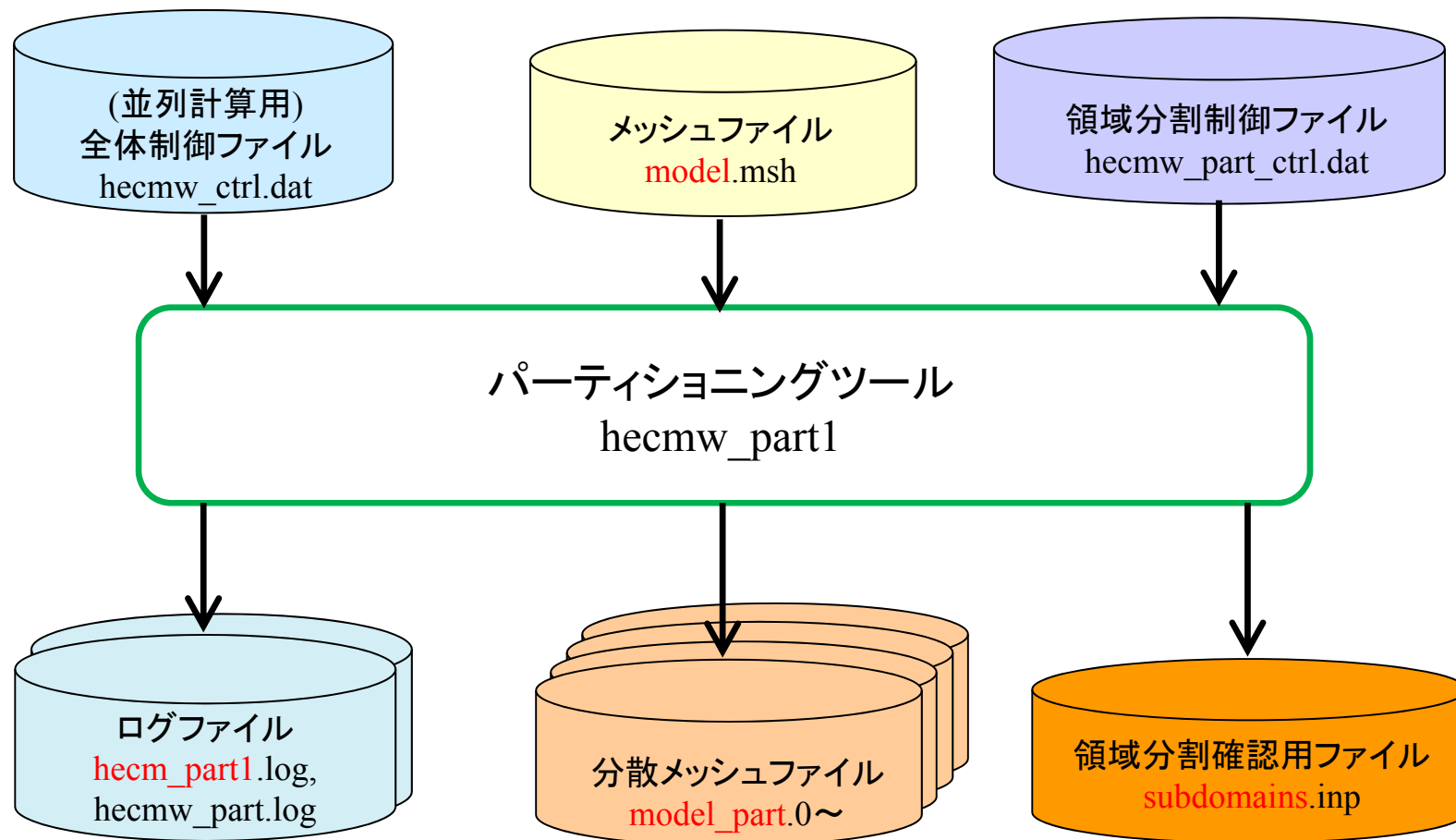
時刻 t の物理量

$${}^t_0\mathbf{E} = \frac{1}{2} \left\{ ({}^0\nabla \otimes {}^t\mathbf{u}) + ({}^0\nabla \otimes {}^t\mathbf{u})^T + ({}^0\nabla \otimes {}^t\mathbf{u}) \cdot ({}^0\nabla \otimes {}^t\mathbf{u})^T \right\}$$

基準となる時刻が時刻0の意味

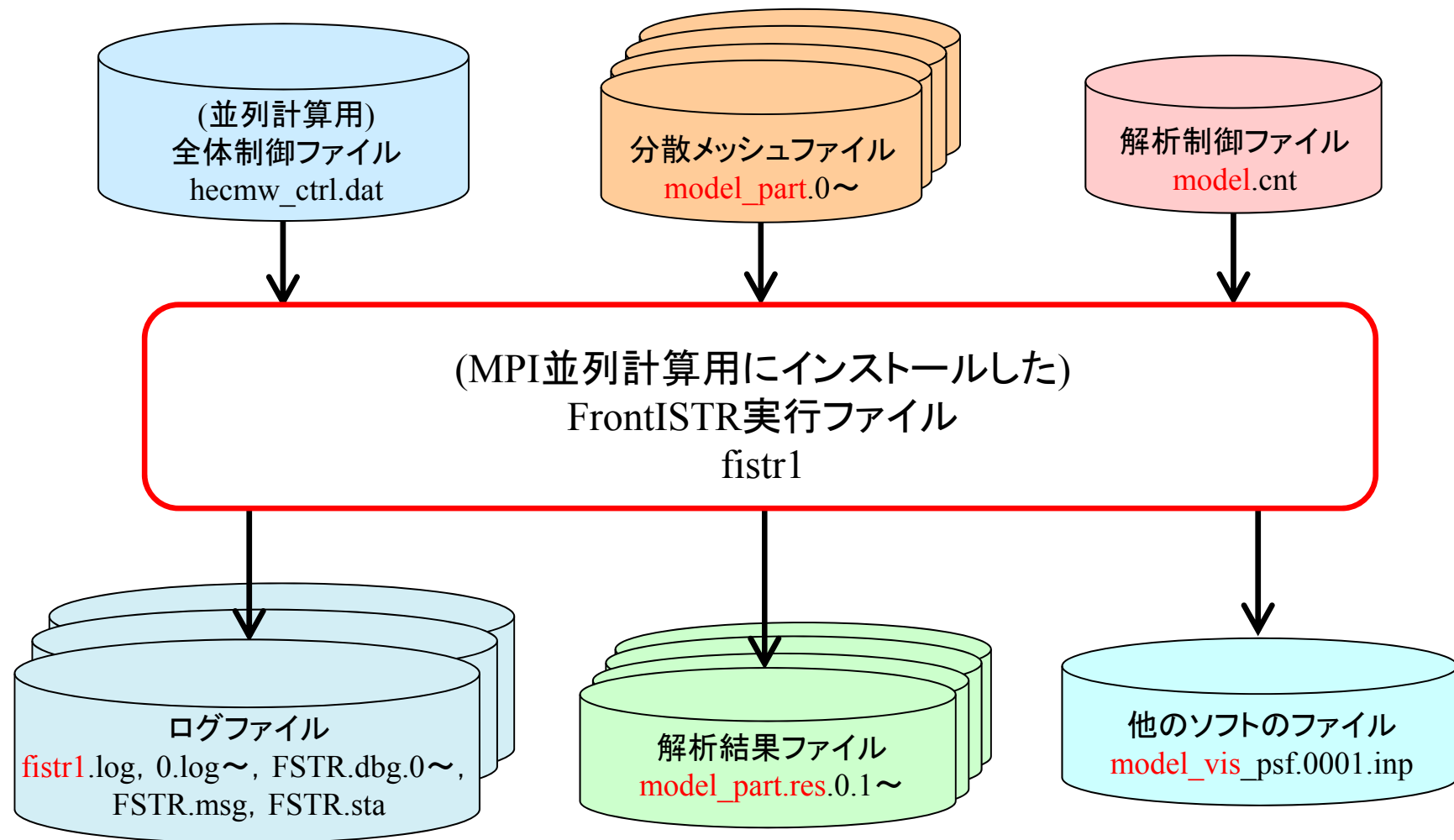
a, b	スカラー
$\mathbf{a}, \mathbf{b}$	ベクトル
$\mathbf{A}, \mathbf{B}, \mathbf{a} \otimes \mathbf{b}$	2階のテンソル
$\mathbf{a} \cdot \mathbf{b} = a_i b_i$	
$\mathbf{A} : \mathbf{B} = A_{ij} B_{ij}$	

FrontISTRの並列計算 (1/2)



※ 赤字の名前は自由に変更可能

FrontISTRの並列計算 (2/2)



※ 赤字の名前は自由に変更可能

全体制御ファイル (並列計算)

(並列計算用)
全体制御ファイル
hecmw_ctrl.dat

```
1 #↓
2 # for solver↓
3 #↓
4 !MESH, NAME=fstrMSH, TYPE=HECMW-DIST↓
5 conrod_part↓
6 !CONTROL, NAME=fstrCNT↓
7 ../PE1/conrod.cnt↓
8 !RESULT, NAME=fstrRES, IO=OUT↓
9 conrod_part.res↓
10 #↓
11 # for partitioner↓
12 #↓
13 !MESH, NAME=part_in, TYPE=HECMW-ENTIRE↓
14 ../PE1/conrod.msh↓
15 !MESH, NAME=part_out, TYPE=HECMW-DIST↓
16 conrod_part↓
17 #↓
18 # for visualizer↓
19 #↓
20 !MESH, NAME=mesh, TYPE=HECMW-DIST↓
21 conrod_part↓
22 !RESULT, NAME=result, IO=IN↓
23 conrod_part.res↓
24 !RESULT, NAME=vis_out, IO=OUT↓
25 conrod_vis↓
```

FrontISTR を使用するときの設定

- ← 分散メッシュファイル名の設定
- ← 解析制御ファイル名の設定
- ← 解析結果ファイル名の設定

パーティショニングツールを使用するときの設定

- ← メッシュファイル名の設定
- ← 分散メッシュファイル名の設定

- ← その他のソフトのファイル作成のために入力する分散メッシュファイル名と解析結果ファイル名の設定
- ← その他のソフトのファイル名の設定

領域分割制御ファイル

領域分割制御ファイル
hecmw_part_ctrl.dat

```
1 !PARTITION, TYPE=NODE-BASED, METHOD=PMETIS, DOMAIN=4, UCD=subdomains.inp↓
```

節点ベースで領域分割

領域分割数

グラフ分割ツール名

- 領域数が8より小さい場合
METHOD=PMETIS
Multilevel Recursive Bisectionによるグラフ分割
- 領域数が8より大きい場合
METHOD=KMETIS
k-way partitioningによるグラフ分割

領域分割確認用の
UCDフォーマット
ファイル名

FrontISTRの梁要素／シェル要素の解説 ～プログラム解説を中心に～

目次

1. 準備

- FrontISTRのソースプログラムとインストール手順
- FrontISTRの入力ファイル
- 構造要素の要素タイプ番号
- 構造要素が使用できるFrontISTRの解析機能
- 構造要素が使用できるFrontISTRの材料

2. プログラム解説

- ディレクトリfistr1内のプログラム
- 構造要素を用いた場合のプログラムの流れ
- 構造要素に関連するプログラム
- Bernoulli-Euler梁要素の剛性マトリックス計算プログラム
- MITCシェル要素の剛性マトリックス計算プログラム

FrontISTRの梁要素／シェル要素の解説 ～プログラム解説を中心に～

目次

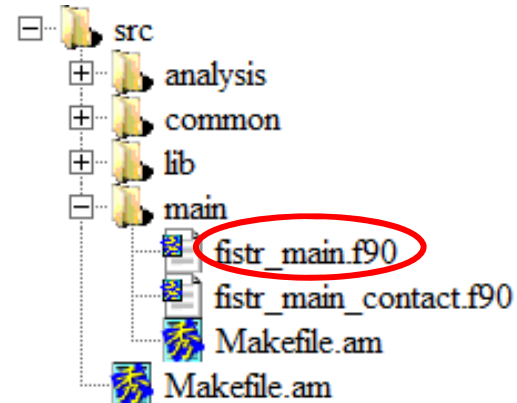
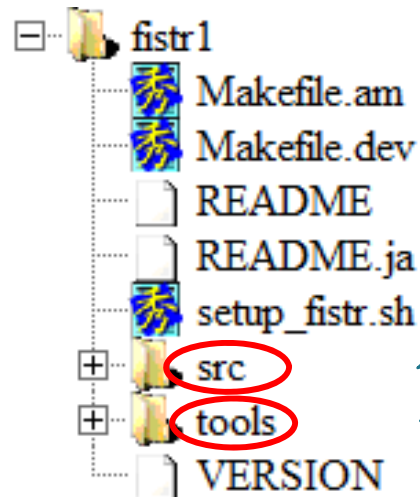
1. 準備

- FrontISTRのソースプログラムとインストール手順
- FrontISTRの入力ファイル
- 構造要素の要素タイプ番号
- 構造要素が使用できるFrontISTRの解析機能
- 構造要素が使用できるFrontISTRの材料

2. プログラム解説

- ディレクトリfistr1内のプログラム
- 構造要素を用いた場合のプログラムの流れ
- 構造要素に関連するプログラム
- Bernoulli-Euler梁要素の剛性マトリックス計算プログラム
- MITCシェル要素の剛性マトリックス計算プログラム

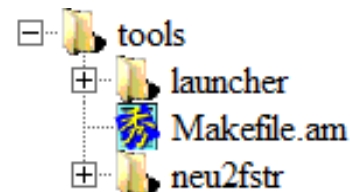
ディレクトリfistr1内のプログラム (1/2)



ディレクトリsrcの下が
FrontISTRライブラリ
(libffistr, libfistr) の
ソースプログラム

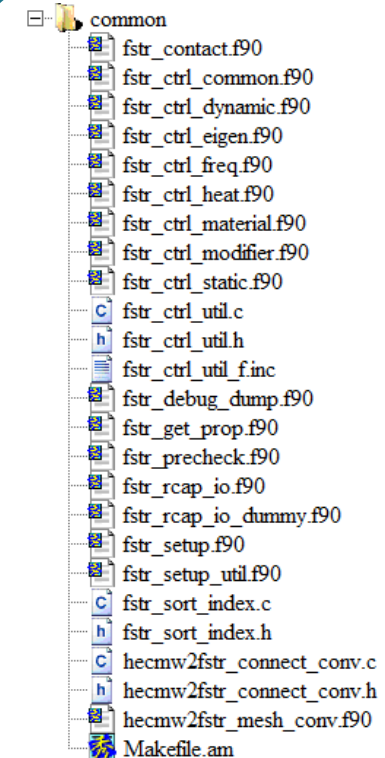
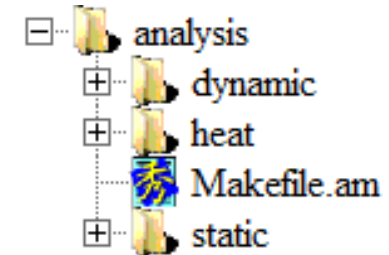
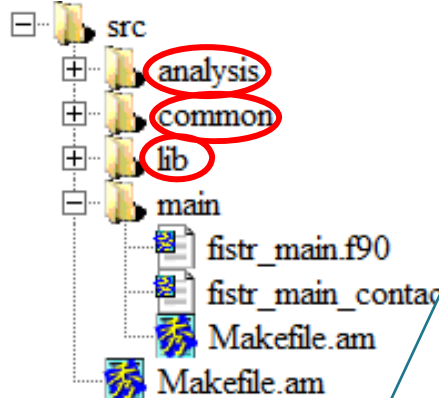
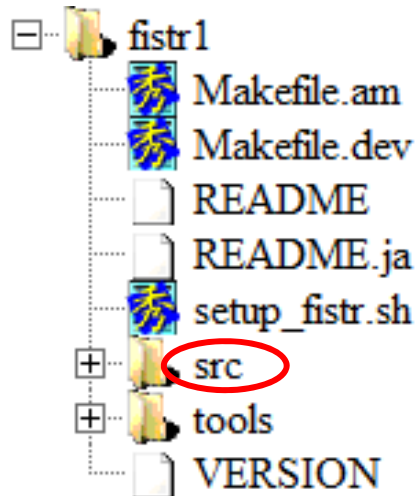
ディレクトリsrc内の
ディレクトリは
analysis, common,
lib, mainの四つ

ディレクトリsrc/mainの
下が実行ファイルfistr1
のソースプログラム

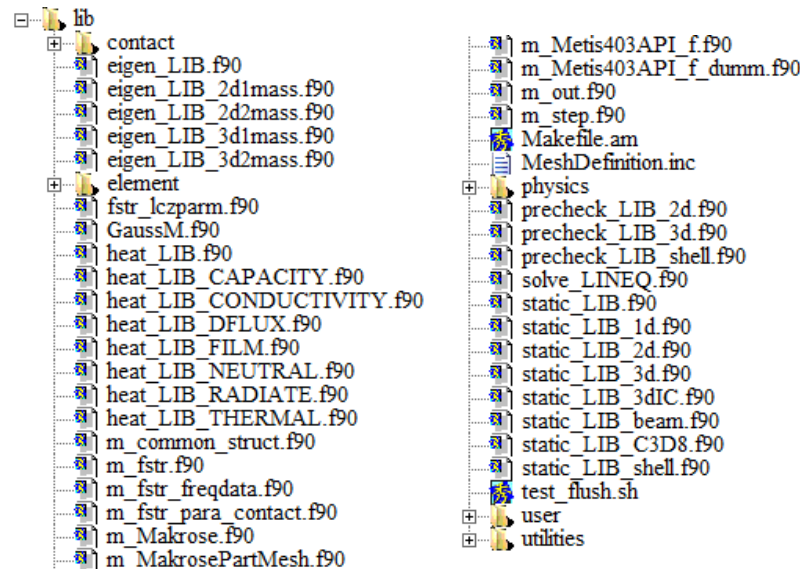


ディレクトリtoolsの下が
実行ファイルneu2fstr
のソースプログラム

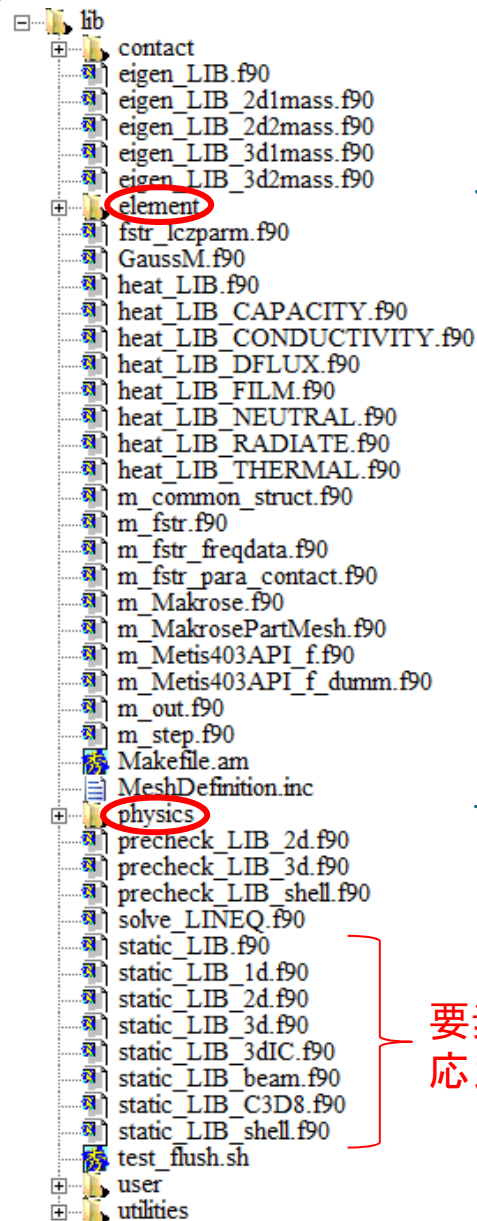
ディレクトリfistr1内のプログラム (2/2)



ディレクトリsrc内の
ディレクトリは
analysis, common,
lib, mainの四つ



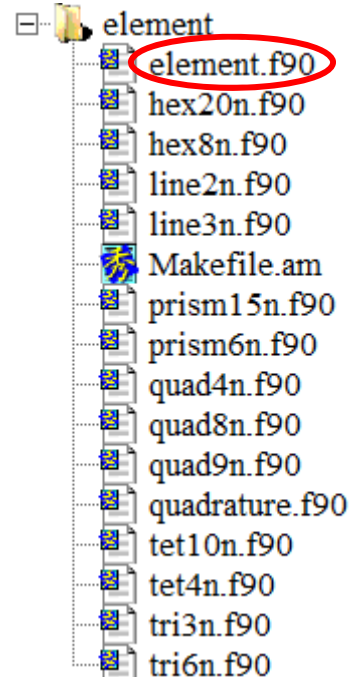
ディレクトリfistr1/src/lib/element内のプログラム



lib

- contact
- eigen_LIB.f90
- eigen_LIB_2d1mass.f90
- eigen_LIB_2d2mass.f90
- eigen_LIB_3d1mass.f90
- eigen_LIB_3d2mass.f90
- element**
- fstr_iczparm.f90
- GaussM.f90
- heat_LIB.f90
- heat_LIB_CAPACITY.f90
- heat_LIB_CONDUCTIVITY.f90
- heat_LIB_DFLUX.f90
- heat_LIB_FILM.f90
- heat_LIB_NEUTRAL.f90
- heat_LIB_RADIATE.f90
- heat_LIB_THERMAL.f90
- m_common_struct.f90
- m_fstr.f90
- m_fstr_freqdata.f90
- m_fstr_para_contact.f90
- m_Makrose.f90
- m_MakrosePartMesh.f90
- m_Metis403API.f.f90
- m_Metis403API_f_dumm.f90
- m_out.f90
- m_step.f90
- Makefile.am
- MeshDefinition.inc
- physics**
- precheck_LIB_2d.f90
- precheck_LIB_3d.f90
- precheck_LIB_shell.f90
- solve_LINEQ.f90
- static_LIB.f90
- static_LIB_1d.f90
- static_LIB_2d.f90
- static_LIB_3d.f90
- static_LIB_3dIC.f90
- static_LIB_beam.f90
- static_LIB_C3D8.f90
- static_LIB_shell.f90
- test_flush.sh
- user
- utilities

要素剛性マトリックスや
応力・内力の計算で使用



element

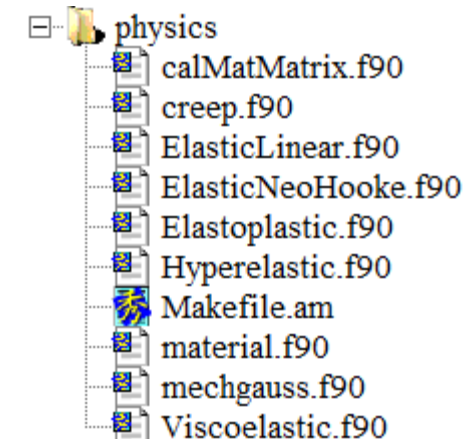
- element.f90**
- hex20n.f90
- hex8n.f90
- line2n.f90
- line3n.f90
- Makefile.am
- prism15n.f90
- prism6n.f90
- quad4n.f90
- quad8n.f90
- quad9n.f90
- quadrature.f90
- tet10n.f90
- tet4n.f90
- tri3n.f90
- tri6n.f90

有限要素の幾何情報を計算する
プログラム

Bマトリックスの計算で使用

材料情報を計算する
プログラム

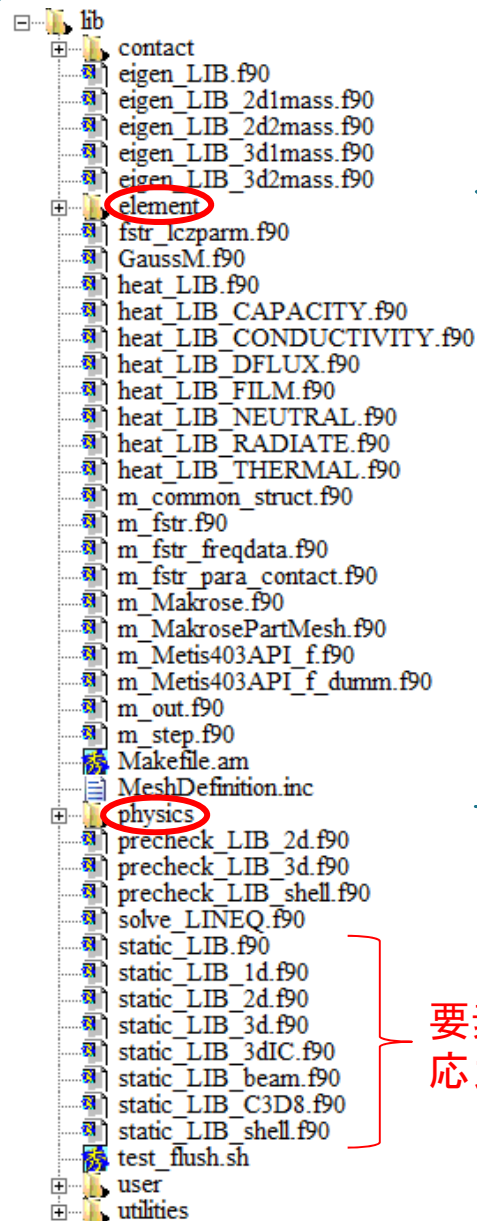
Dマトリックスの計算で使用



physics

- calMatMatrix.f90
- creep.f90
- ElasticLinear.f90
- ElasticNeoHooke.f90
- Elastoplastic.f90
- Hyperelastic.f90
- Makefile.am
- material.f90
- mechgauss.f90
- Viscoelastic.f90

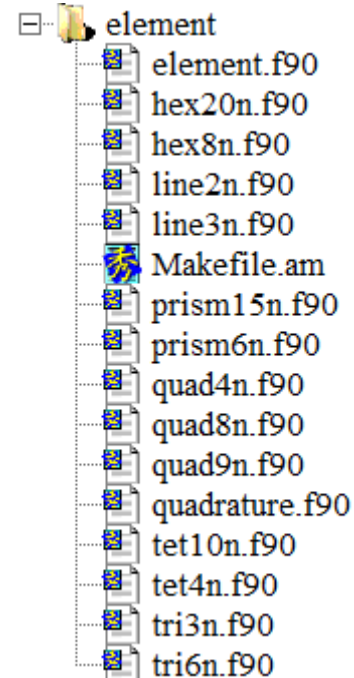
ディレクトリfistr1/src/lib/physics内のプログラム



lib

- contact
- eigen_LIB.f90
- eigen_LIB_2d1mass.f90
- eigen_LIB_2d2mass.f90
- eigen_LIB_3d1mass.f90
- eigen_LIB_3d2mass.f90
- element**
- fstr_iczparm.f90
- GaussM.f90
- heat_LIB.f90
- heat_LIB_CAPACITY.f90
- heat_LIB_CONDUCTIVITY.f90
- heat_LIB_DFLUX.f90
- heat_LIB_FILM.f90
- heat_LIB_NEUTRAL.f90
- heat_LIB_RADIATE.f90
- heat_LIB_THERMAL.f90
- m_common_struct.f90
- m_fstr.f90
- m_fstr_freqdata.f90
- m_fstr_para_contact.f90
- m_Makrose.f90
- m_MakrosePartMesh.f90
- m_Metis403API.f.f90
- m_Metis403API_f_dumm.f90
- m_out.f90
- m_step.f90
- Makefile.am
- MeshDefinition.inc
- physics**
- precheck_LIB_2d.f90
- precheck_LIB_3d.f90
- precheck_LIB_shell.f90
- solve_LINEQ.f90
- static_LIB.f90
- static_LIB_1d.f90
- static_LIB_2d.f90
- static_LIB_3d.f90
- static_LIB_3dIC.f90
- static_LIB_beam.f90
- static_LIB_C3D8.f90
- static_LIB_shell.f90
- test_flush.sh
- user
- utilities

要素剛性マトリックスや
応力・内力の計算で使用



element

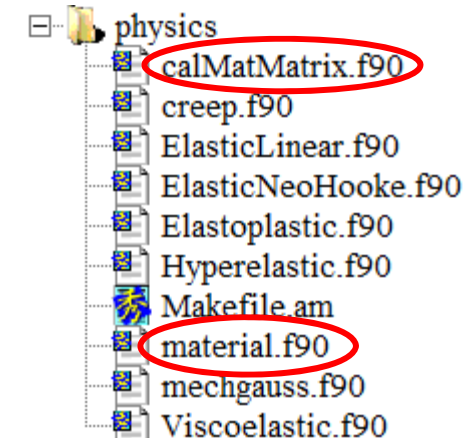
- element.f90
- hex20n.f90
- hex8n.f90
- line2n.f90
- line3n.f90
- Makefile.am
- prism15n.f90
- prism6n.f90
- quad4n.f90
- quad8n.f90
- quad9n.f90
- quadrature.f90
- tet10n.f90
- tet4n.f90
- tri3n.f90
- tri6n.f90

有限要素の幾何情報を計算する
プログラム

Bマトリックスの計算で使用

材料情報を計算する
プログラム

Dマトリックスの計算で使用



physics

- calMatMatrix.f90**
- creep.f90
- ElasticLinear.f90
- ElasticNeoHooke.f90
- Elastoplastic.f90
- Hyperelastic.f90
- Makefile.am
- material.f90**
- mechgauss.f90
- Viscoelastic.f90

微小変形静解析のプログラムの流れ

[main/fistr_main.f90] PROGRAM fstr_main ...メインプログラム

hecmw_init()

hecmw_get_mesh()

[main/fistr_main.f90] fstr_init() ...変数初期化・入力データ読み込み

hecmw_nullify_matrix()

hecmw_nullify_result_data()

[main/fistr_main.f90] fstr_init_file()

hecmw_mat_con()

[main/fistr_main.f90] fstr_condition()

hecmw_ctrl_get_control_file()

[main/fistr_main.f90] fstr_linear_static_analysis() ...線形静解析用のルーチンへ

[analysis/static/fstr_solve_LINEAR.f90] m_fstr_LINEAR::fstr_solve_LINEAR()

[analysis/static/static_mat_ass.f90] m_static_mat_ass::fstr_mat_ass() ...剛性マトリックスの作成

[analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()

hecmw_mat_clear()

[analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()

【ここで要素タイプに応じて分岐します】

[analysis/static/static_LIB_000.f90] m_static_LIB_000::STF_000() ...要素剛性マトリックスの計算

[lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix() ...Dマトリックスの計算

【ここで材料タイプに応じて分岐します】

[lib/physics/000.f90] m_000::000()

※ 梁要素の場合は使用しない

※ シェル要素の場合はMatlMatrix_Shell()

hecmw_mat_ass_elem() ...要素剛性マトリックスをassemble

[analysis/static/fstr_ass_load.f90] m_fstr_ass_load::fstr_ass_load() ...外力ベクトルの計算

[analysis/static/fstr_AddBC.f90] m_fstr_AddBC::fstr_AddBC() ...境界条件の処理

hecmw_allREDUCE_R1()

[lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ() ...線形ソルバーによる求解

hecmw_solve_000()

hecmw_update_O_R()

[analysis/static/fstr_Update.f90] m_fstr_Update::fstr_UpdateOD()

【ここで要素タイプに応じて分岐します】

[lib/static_LIB_000.f90] m_static_LIB_000::UpdateST_000() ...応力の計算

[lib/static_LIB_000.f90] m_static_LIB_000::STF_000() ...要素剛性マトリックスの計算

[lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix() ...Dマトリックスの計算

【ここで材料タイプに応じて分岐します】

[lib/physics/000.f90] m_000::000()

hecmw_update_O_R()

[analysis/static/static_output.f90] m_static_output::fstr_static_Output() ...結果の出力

[analysis/static/static_make_result.f90] m_static_make_result::fstr_write_static_result()

[main/fistr_main.f90] fstr_main::fstr_finalize() ...変数の削除

hecmw_finalize()

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=STATICと記述

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

梁要素を用いた微小変形静解析のプログラムの流れ

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=STATICと記述

```
[main/fistr_main.f90] PROGRAM fstr_main    ...メインプログラム
  hecmw_init()
  hecmw_get_mesh()
  [main/fistr_main.f90] fstr_init()    ... 変数初期化・入力データ読み込み
    hecmw_nullify_matrix()
    hecmw_nullify_result_data()
    [main/fistr_main.f90] fstr_init_file()
    hecmw_mat_con()
    [main/fistr_main.f90] fstr_condition()
    hecmw_ctrl_get_control_file()
  [main/fistr_main.f90] fstr_linear_static_analysis()    ... 線形静解析用のルーチンへ
    [analysis/static/fstr_solve_LINEAR.f90] m_fstr_LINEAR::fstr_solve_LINEAR()
    [analysis/static/static_mat_ass.f90] m_static_mat_ass::fstr_mat_ass()    ... 剛性マトリックスの作成
    [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()
    hecmw_mat_clear()
    [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()
    【ここで要素タイプに応じて分岐します】
    [analysis/static/static_LIB_beam.f90] m_static_LIB_beam::STF_Beam()    ... 梁要素の剛性マトリックスの計算
    hecmw_mat_ass_elem()    ... 要素剛性マトリックスをassemble
    [analysis/static/fstr_ass_load.f90] m_fstr_ass_load::fstr_ass_load()    ... 外力ベクトルの計算
    [analysis/static/fstr_AddBC.f90] m_fstr_AddBC::fstr_AddBC()    ... 境界条件の処理
    hecmw_allREDUCE_R1()
    [lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ()    ... 線形ソルバーによる求解
    hecmw_solve_66()
    hecmw_update_m_R()
    [analysis/static/fstr_Update.f90] m_fstr_Update::fstr_Update6D()
    [analysis/static/static_output.f90] m_static_output::fstr_static_Output()    ... 結果の出力
    [analysis/static/static_make_result.f90] m_static_make_result::fstr_write_static_result()
  [main/fistr_main.f90] fstr_main::fstr_finalize()    ... 変数の削除
  hecmw_finalize()
```

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

シェル要素を用いた微小変形静解析のプログラムの流れ

[main/fistr_main.f90] PROGRAM fstr_main ...メインプログラム

hecmw_init()

hecmw_get_mesh()

[main/fistr_main.f90] fstr_init() ... 変数初期化・入力データ読み込み

hecmw_nullify_matrix()

hecmw_nullify_result_data()

[main/fistr_main.f90] fstr_init_file()

hecmw_mat_con()

[main/fistr_main.f90] fstr_condition()

hecmw_ctrl_get_control_file()

[main/fistr_main.f90] fstr_linear_static_analysis() ... 線形静解析用のルーチンへ

[analysis/static/fstr_solve_LINEAR.f90] m_fstr_LINEAR::fstr_solve_LINEAR()

[analysis/static/static_mat_ass.f90] m_static_mat_ass::fstr_mat_ass() ... 剛性マトリックスの作成

[analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()

hecmw_mat_clear()

[analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()

【ここで要素タイプに応じて分岐します】

[analysis/static/static_LIB_shell.f90] m_static_LIB_shell::STF_Shell_MITC() ... シェル要素の剛性マトリックスの計算

[lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix_Shell() ... Dマトリックスの計算

【ここで材料タイプに応じて分岐します】

[lib/physics/ElasticLinear.f90] m_ElasticLinear::LinearElastic_Shell()

hecmw_mat_ass_elem() ... 要素剛性マトリックスをassemble

[analysis/static/fstr_ass_load.f90] m_fstr_ass_load::fstr_ass_load() ... 外力ベクトルの計算

[analysis/static/fstr_AddBC.f90] m_fstr_AddBC::fstr_AddBC() ... 境界条件の処理

hecmw_allREDUCE_R1()

[lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ() ... 線形ソルバーによる求解

hecmw_solve_66()

hecmw_update_m_R()

[analysis/static/fstr_Update.f90] m_fstr_Update::fstr_Update3D()

【ここで要素タイプに応じて分岐します】

[lib/static_LIB_shell.f90] m_static_LIB_shell::UpdateST_Shell_MITC() ... シェル要素の応力の計算

[lib/static_LIB_shell.f90] m_static_LIB_shell::STF_Shell_MITC() ... シェル要素の剛性マトリックスの計算

[lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix_Shell() ... Dマトリックスの計算

【ここで材料タイプに応じて分岐します】

[lib/physics/ElasticLinear.f90] m_ElasticLinear::LinearElastic_Shell()

hecmw_update_m_R()

[analysis/static/static_output.f90] m_static_output::fstr_static_Output() ... 結果の出力

[analysis/static/static_make_result.f90] m_static_make_result::fstr_write_static_result()

[main/fistr_main.f90] fstr_main::fstr_finalize() ... 変数の削除

hecmw_finalize()

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=STATICと記述

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

微小変形動解析のプログラムの流れ

```
[main/fistr_main.f90] PROGRAM fistr_main    ...メインプログラム
  hecmw_init()
  hecmw_get_mesh()
  [main/fistr_main.f90] fistr_init()    ... 変数初期化・入力データ読み込み
    hecmw_nullify_matrix()
    hecmw_nullify_result_data()
    [main/fistr_main.f90] fistr_init_file()
    hecmw_mat_con()
    [main/fistr_main.f90] fistr_condition()
      hecmw_ctrl_get_control_file()
  [main/fistr_main.f90] fistr_linear_dynamic_analysis()    ... 動解析用のルーチンへ
    [analysis/dynamic/transit/fstr_solve_dynamic.f90] fstr_solver_dynamic::fstr_solve_dynamic()
    [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_alloc()    ... 動解析用の変数作成
    [analysis/static/fstr_Update.f90] m_fstr_Update::fstr_UpdateState()    ... 状態の更新
    【ここで動解析機能に応じて分岐します】
    [analysis/dynamic/transit/fstr_dynamic_implicit.f90] fstr_dynamic_implicit::fstr_solve_dynamic_implicit()    ... Newmark- $\beta$  法による時間積分
      hecMAT_init()
      [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()
      hecmw_mat_clear()
      [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()
      【ここで要素タイプに応じて分岐します】
      [analysis/static/static_LIB_000.f90] m_static_LIB_000::STF_000()    ... 要素剛性マトリックスの計算
      [lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix()    ... Dマトリックスの計算
      【ここで材料タイプに応じて分岐します】
      [lib/physics/000.f90] m_000::000()    ... 梁要素の場合は使用しない
      ... シェル要素の場合はMatlMatrix_Shell()
      hecmw_mat_ass_elem()    ... 要素剛性マトリックスをassemble
      [analysis/dynamic/mode/fstr_EIG_setMASS.f90] m_fstr_EIG_setMASS::setMASS()    ... 集中質量マトリックスの計算
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの始まり】  $n = 1, 2, 3, \dots$ 
      hecmw_matvec_00()
      [analysis/dynamic/transit/dynamic_mat_ass_load.f90] m_dynamic_mat_ass_load::dynamic_mat_ass_load()    ... 外力ベクトルの計算
      [analysis/dynamic/transit/dynamic_mat_ass_bc.f90] m_fstr_Update::dynamic_mat_ass_bc()    ... 変位の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_vl.f90] m_fstr_Update::dynamic_mat_ass_bc_vl()    ... 速度の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_ac.f90] m_fstr_Update::dynamic_mat_ass_bc_ac()    ... 加速度の境界条件処理
      [lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ()    ... 線形ソルバーによる求解
      hecmw_solve_000()
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの終わり】
      [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_finalize()    ... 動解析用の変数削除
      hecMAT_finalize()
  [main/fistr_main.f90] fistr_main::fstr_finalize()    ... 変数の削除
  hecmw_finalize()
```

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=DYNAMIC,

!DYNAMIC下の**idx_resp**を1,

!DYNAMIC,TYPE=LINEARと記述

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

梁要素を用いた微小変形動解析のプログラムの流れ

```
[main/fistr_main.f90] PROGRAM fistr_main    ...メインプログラム
  hecmw_init()
  hecmw_get_mesh()
  [main/fistr_main.f90] fstr_init()    ... 変数初期化・入力データ読み込み
    hecmw_nullify_matrix()
    hecmw_nullify_result_data()
    [main/fistr_main.f90] fstr_init_file()
    hecmw_mat_con()
    [main/fistr_main.f90] fstr_condition()
      hecmw_ctrl_get_control_file()
  [main/fistr_main.f90] fstr_linear_dynamic_analysis()    ... 動解析用のルーチンへ
    [analysis/dynamic/transit/fstr_solve_dynamic.f90] fstr_solver_dynamic::fstr_solve_dynamic()
    [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_alloc()    ... 動解析用の変数作成
    [analysis/static/fstr_Update.f90] m_fstr_Update::fstr_UpdateState()    ... 状態の更新
    【ここで動解析機能に応じて分岐します】
    [analysis/dynamic/transit/fstr_dynamic_implicit.f90] fstr_dynamic_implicit::fstr_solve_dynamic_implicit()    ... Newmark- $\beta$  法による時間積分
      hecMAT_init()
      [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()
        hecmw_mat_clear()
        [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()
        【ここで要素タイプに応じて分岐します】
        [analysis/static/static_LIB_beam.f90] m_static_LIB_beam::STF_Beam()    ... 梁要素の剛性マトリックスの計算
      hecmw_mat_ass_elem()    ... 要素剛性マトリックスをassemble
      [analysis/dynamic/mode/fstr_EIG_setMASS.f90] m_fstr_EIG_setMASS::setMASS()    ... 集中質量マトリックスの計算
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの始まり】  $n = 1, 2, 3, \dots$ 
      hecmw_matvec()
      [analysis/dynamic/transit/dynamic_mat_ass_load.f90] m_dynamic_mat_ass_load::dynamic_mat_ass_load()    ... 外力ベクトルの計算
      [analysis/dynamic/transit/dynamic_mat_ass_bc.f90] m_fstr_Update::dynamic_mat_ass_bc()    ... 変位の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_vl.f90] m_fstr_Update::dynamic_mat_ass_bc_vl()    ... 速度の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_ac.f90] m_fstr_Update::dynamic_mat_ass_bc_ac()    ... 加速度の境界条件処理
      [lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ()    ... 線形ソルバーによる求解
      hecmw_solve_66()
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの終わり】
      [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_finalize()    ... 動解析用の変数削除
      hecMAT_finalize()
  [main/fistr_main.f90] fstr_main::fstr_finalize()    ... 変数の削除
  hecmw_finalize()
```

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=DYNAMIC,

!DYNAMIC下の**idx_resp**を1,

!DYNAMIC,TYPE=LINEARと記述

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

シェル要素を用いた微小変形動解析のプログラムの流れ

```
[main/fistr_main.f90] PROGRAM fistr_main    ...メインプログラム
  hecmw_init()
  hecmw_get_mesh()
  [main/fistr_main.f90] fistr_init()    ... 変数初期化・入力データ読み込み
    hecmw_nullify_matrix()
    hecmw_nullify_result_data()
    [main/fistr_main.f90] fistr_init_file()
    hecmw_mat_con()
    [main/fistr_main.f90] fistr_condition()
      hecmw_ctrl_get_control_file()
  [main/fistr_main.f90] fistr_linear_dynamic_analysis()    ... 動解析用のルーチンへ
    [analysis/dynamic/transit/fstr_solve_dynamic.f90] fstr_solver_dynamic::fstr_solve_dynamic()
    [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_alloc()    ... 動解析用の変数作成
    [analysis/static/fstr_Update.f90] m_fstr_Update::fstr_UpdateState()    ... 状態の更新
    【ここで動解析機能に応じて分岐します】
    [analysis/dynamic/transit/fstr_dynamic_implicit.f90] fstr_dynamic_implicit::fstr_solve_dynamic_implicit()    ... Newmark- $\beta$  法による時間積分
      hecMAT_init()
      [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_mat_ass_main()
      hecmw_mat_clear()
      [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::fstr_local_stf_create()
      【ここで要素タイプに応じて分岐します】
      [analysis/static/static_LIB_shell.f90] m_static_LIB_shell::STF_Shell_MITC()    ... シェル要素の剛性マトリックスの計算
      [lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix_Shell()    ... Dマトリックスの計算
      【ここで材料タイプに応じて分岐します】
      [lib/physics/ElasticLinear.f90] m_ElasticLinear::ElasticLinear_Shell()
      hecmw_mat_ass_elem()    ... 要素剛性マトリックスをassemble
      [analysis/dynamic/mode/fstr_EIG_setMASS.f90] m_fstr_EIG_setMASS::setMASS()    ... 集中質量マトリックスの計算
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの始まり】  $n = 1, 2, 3, \dots$ 
      hecmw_matvec()
      [analysis/dynamic/transit/dynamic_mat_ass_load.f90] m_dynamic_mat_ass_load::dynamic_mat_ass_load()    ... 外力ベクトルの計算
      [analysis/dynamic/transit/dynamic_mat_ass_bc.f90] m_fstr_Update::dynamic_mat_ass_bc()    ... 変位の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_vl.f90] m_fstr_Update::dynamic_mat_ass_bc_vl()    ... 速度の境界条件の処理
      [analysis/dynamic/transit/dynamic_mat_ass_bc_ac.f90] m_fstr_Update::dynamic_mat_ass_bc_ac()    ... 加速度の境界条件処理
      [lib/solve_LINEQ.f90] m_solve_LINEQ::solve_LINEQ()    ... 線形ソルバーによる求解
      hecmw_solve_66()
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::fstr_dynamic_Output()    ... 結果の出力
      [analysis/dynamic/transit/dynamic_output.f90] m_dynamic_output::dynamic_output_monit()    ... 時刻歴の出力
      【時間ステップの更新のループの終わり】
      [common/fstr_setup.f90] m_fstr_setup::fstr_dynamic_finalize()    ... 動解析用の変数削除
      hecMAT_finalize()
  [main/fistr_main.f90] fistr_main::fstr_finalize()    ... 変数の削除
  hecmw_finalize()
```

FrontISTRの解析制御データ

(* .cntファイル) 内で

!SOLUTION,TYPE=DYNAMIC,

!DYNAMIC下の**idx_resp**を1,

!DYNAMIC,TYPE=LINEARと記述

[ディレクトリ/ファイル名]
モジュール名::サブルーチン名()を意味しています

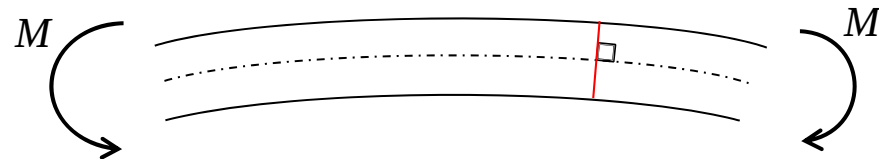
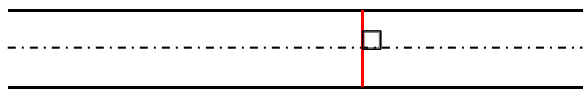
FrontISTRの梁要素の仮定

- (a) 時刻 $t = 0$ に梁の中立軸に垂直な直線は梁が変形する間も直線であるが、変形した中立軸に垂直である
(せん断変形は考慮されない)

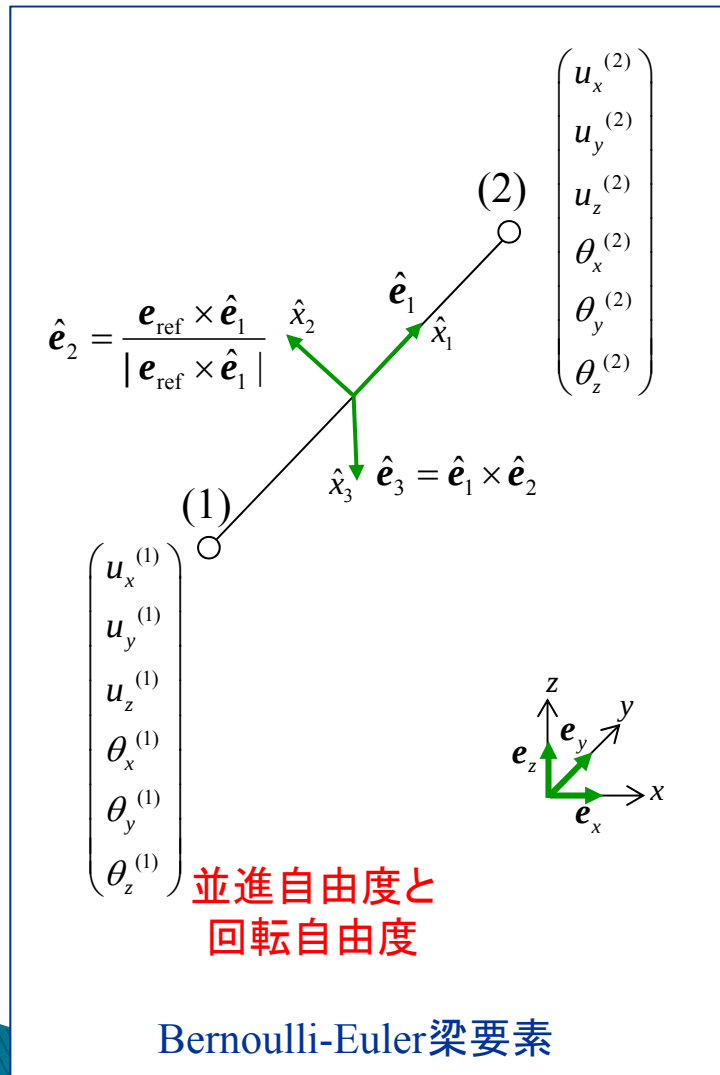
Bernoulli-Eulerの仮定

- (b) 梁の断面は変形しない
微小ひずみの仮定

$t = 0$



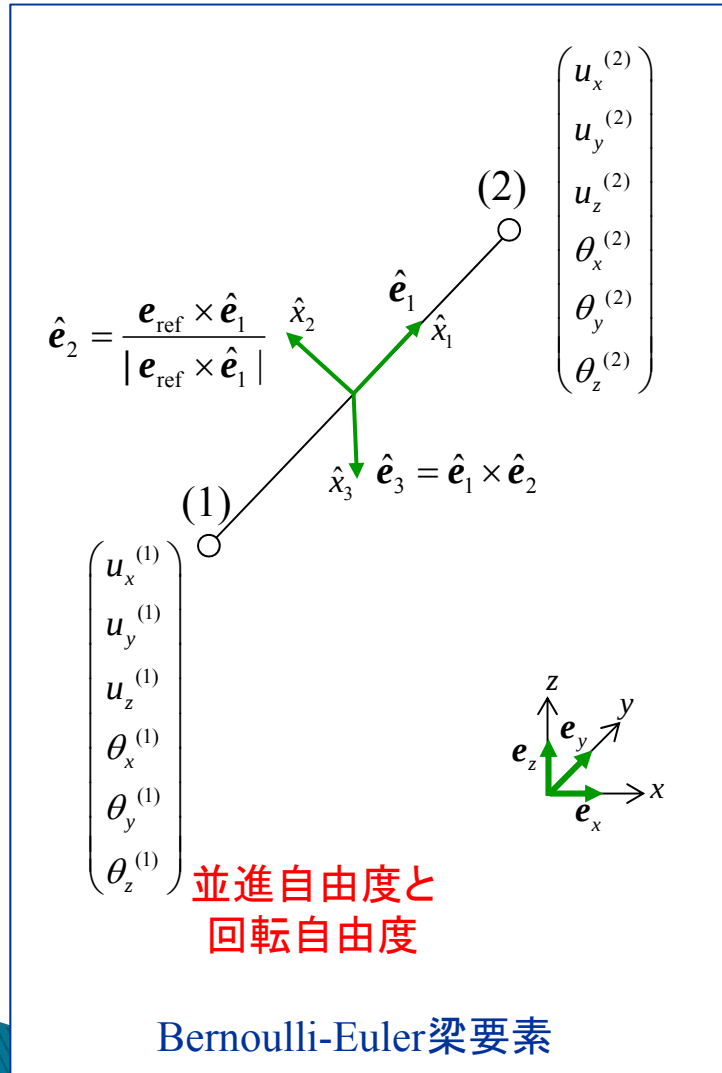
2節点Bernoulli-Euler梁要素 (1/4)



微小変形を仮定

- 中立軸方向の一様引張・一様圧縮
 $\hat{x}_1$ 方向変位: $\hat{x}_1$ の1次式を仮定
- 曲げ
 $\hat{x}_2$ 方向変位: $\hat{x}_1$ の3次式を仮定
 $\hat{x}_3$ 方向回転: $\hat{\theta}_3 = \frac{\partial \hat{u}_2}{\partial \hat{x}_1}$
 $\hat{x}_3$ 方向変位: $\hat{x}_1$ の3次式を仮定
 $\hat{x}_2$ 方向回転: $\hat{\theta}_2 = -\frac{\partial \hat{u}_3}{\partial \hat{x}_1}$
- ねじり
 $\hat{x}_1$ 方向回転: $\hat{x}_1$ の1次式を仮定

2節点Bernoulli-Euler梁要素 (2/4)



変位ベクトル

$$\mathbf{u} = u_i \mathbf{e}_i = \hat{u}_i \hat{\mathbf{e}}_i$$

$$\hat{\mathbf{e}}_i \cdot \mathbf{u} = (\hat{\mathbf{e}}_i \cdot \mathbf{e}_j) u_j = T_{ij}^e u_j$$

$$\begin{pmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \hat{u}_3 \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{e}}_1 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_1 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_1 \cdot \mathbf{e}_z \\ \hat{\mathbf{e}}_2 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_2 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_2 \cdot \mathbf{e}_z \\ \hat{\mathbf{e}}_3 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_3 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_3 \cdot \mathbf{e}_z \end{pmatrix} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}$$

局所座標成分

$\mathbf{T}^e$

全体座標成分

$$\begin{pmatrix} \hat{\theta}_1 \\ \hat{\theta}_2 \\ \hat{\theta}_3 \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{e}}_1 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_1 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_1 \cdot \mathbf{e}_z \\ \hat{\mathbf{e}}_2 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_2 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_2 \cdot \mathbf{e}_z \\ \hat{\mathbf{e}}_3 \cdot \mathbf{e}_x & \hat{\mathbf{e}}_3 \cdot \mathbf{e}_y & \hat{\mathbf{e}}_3 \cdot \mathbf{e}_z \end{pmatrix} \begin{pmatrix} \theta_x \\ \theta_y \\ \theta_z \end{pmatrix}$$

局所座標成分

$\mathbf{T}^e$

全体座標成分

応力テンソル

$$\boldsymbol{\sigma} = \sigma_{ij} \mathbf{e}_i \otimes \mathbf{e}_j = \hat{\sigma}_{ij} \hat{\mathbf{e}}_i \otimes \hat{\mathbf{e}}_j$$

$$\hat{\mathbf{e}}_i \cdot \boldsymbol{\sigma} \cdot \hat{\mathbf{e}}_j = (\hat{\mathbf{e}}_i \cdot \mathbf{e}_k) \sigma_{kl} (\mathbf{e}_l \cdot \hat{\mathbf{e}}_j) = T_{ik}^e \sigma_{kl} T_{jl}^e$$

2節点Bernoulli-Euler梁要素 (3/4)

局所座標系

$$\hat{\mathbf{K}}^e \hat{\mathbf{u}}^e = \hat{\mathbf{f}}^e$$

$$\hat{\mathbf{K}}^e = \begin{pmatrix} \frac{EA}{l} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{l} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_3}{l^3} & 0 & 0 & 0 & \frac{6EI_3}{l^2} & 0 & -\frac{12EI_3}{l^3} & 0 & 0 & 0 & \frac{6EI_3}{l^2} \\ 0 & 0 & \frac{12EI_2}{l^3} & 0 & -\frac{6EI_2}{l^2} & 0 & 0 & 0 & -\frac{12EI_2}{l^3} & 0 & -\frac{6EI_2}{l^2} & 0 \\ 0 & 0 & 0 & \frac{GJ_1}{l} & 0 & 0 & 0 & 0 & 0 & -\frac{GJ_1}{l} & 0 & 0 \\ 0 & 0 & -\frac{6EI_2}{l^2} & 0 & \frac{4EI_2}{l} & 0 & 0 & 0 & \frac{6EI_2}{l^2} & 0 & \frac{2EI_2}{l} & 0 \\ 0 & \frac{6EI_3}{l^2} & 0 & 0 & 0 & \frac{4EI_3}{l} & 0 & -\frac{6EI_3}{l^2} & 0 & 0 & 0 & \frac{2EI_3}{l} \\ -\frac{EA}{l} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{l} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_3}{l^3} & 0 & 0 & 0 & -\frac{6EI_3}{l^2} & 0 & \frac{12EI_3}{l^3} & 0 & 0 & 0 & -\frac{6EI_3}{l^2} \\ 0 & 0 & -\frac{12EI_2}{l^3} & 0 & \frac{6EI_2}{l^2} & 0 & 0 & 0 & \frac{12EI_2}{l^3} & 0 & \frac{6EI_2}{l^2} & 0 \\ 0 & 0 & 0 & -\frac{GJ_1}{l} & 0 & 0 & 0 & 0 & 0 & \frac{GJ_1}{l} & 0 & 0 \\ 0 & 0 & -\frac{6EI_2}{l} & 0 & \frac{2EI_2}{l} & 0 & 0 & 0 & \frac{6EI_2}{l^2} & 0 & \frac{4EI_2}{l} & 0 \\ 0 & \frac{6EI_3}{l} & 0 & 0 & 0 & \frac{2EI_3}{l} & 0 & -\frac{6EI_3}{l} & 0 & 0 & 0 & \frac{4EI_3}{l} \end{pmatrix}$$

$$\hat{\mathbf{u}}^e = \begin{pmatrix} \hat{u}_1^{(1)} \\ \hat{u}_2^{(1)} \\ \hat{u}_3^{(1)} \\ \hat{\theta}_1^{(1)} \\ \hat{\theta}_2^{(1)} \\ \hat{\theta}_3^{(1)} \\ \hat{u}_1^{(2)} \\ \hat{u}_2^{(2)} \\ \hat{u}_3^{(2)} \\ \hat{\theta}_1^{(2)} \\ \hat{\theta}_2^{(2)} \\ \hat{\theta}_3^{(2)} \end{pmatrix}$$

E : Young率

l : 梁の長さ

I_2, I_3 : 断面2次モーメント

J : ねじり定数

2節点Bernoulli-Euler梁要素 (4/4)

$$\underbrace{\begin{pmatrix} \mathbf{T}^e & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{T}^e & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{T}^e & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{T}^e \end{pmatrix}^T}_{\mathbf{K}^e} \hat{\mathbf{K}}^e \underbrace{\begin{pmatrix} \mathbf{T}^e & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{T}^e & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{T}^e & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{T}^e \end{pmatrix}}_{\mathbf{f}^e} \mathbf{u}^e = \underbrace{\begin{pmatrix} \mathbf{T}^e & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{T}^e & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{T}^e & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{T}^e \end{pmatrix}^T}_{\mathbf{f}^e} \hat{\mathbf{f}}^e$$

全体座標系の要素剛性マトリックス $\mathbf{K}^e$ を作成後,
 $\mathbf{K}^e$ を全体剛性マトリックスに加える

$$\mathbf{u}^e = \begin{pmatrix} u_x^{(1)} \\ u_y^{(1)} \\ u_z^{(1)} \\ \theta_x^{(1)} \\ \theta_y^{(1)} \\ \theta_z^{(1)} \\ u_x^{(2)} \\ u_y^{(2)} \\ u_z^{(2)} \\ \theta_x^{(2)} \\ \theta_y^{(2)} \\ \theta_z^{(2)} \end{pmatrix}$$



[lib/static_LIB_beam.f90]の内容

モジュール名:m_static_LIB_beam

Bernoulli-Euler梁要素の要素剛性マトリックス, 外力ベクトル, 応力, ひずみを計算するモジュール

使用する他のモジュール

- ・[lib/element/element.f90] elementInfo
要素の情報を管理するモジュール
- ・[lib/physics/mechgauss.f90] mMechGauss
Gauss積分点の情報を管理するモジュール
- ・[lib/utilities/utilities.f90] m_utilities
補助的なサブルーチンや関数を集めたモジュール

メンバ変数

- ・整数型 ndof
節点での自由度数 (ndof = 6)

メンバ関数

- ・サブルーチン framtr()
座標変換マトリックスを計算するサブルーチン
- ・サブルーチン STF_Beam()
要素剛性マトリックスを計算するサブルーチン
- ・サブルーチン STF_Beam_641()
3×3ブロック梁要素の剛性マトリックスを計算するサブルーチン
- ・サブルーチン DL_Beam_641()
分布荷重から外力ベクトルを計算するサブルーチン
- ・サブルーチン TLOAD_Beam_641()
熱荷重から外力ベクトルを計算するサブルーチン
- ・サブルーチン NodalStress_Beam_641()
応力やひずみの節点平均値を計算するサブルーチン

[lib/static_LIB_beam.f90]

m_static_LIB_beam::STF_Beam()の内容

サブルーチン名: **STF_Beam()**

Bernoulli-Euler梁要素の要素剛性マトリックスを計算するサブルーチン

引数

- ・整数型 etype
要素タイプ番号
- ・整数型 nn
要素内の節点数
- ・実数型 ecoord(3, nn)
要素内の節点座標
- ・実数型 section(3)
参照ベクトルの成分
- ・実数型 e, p
Young率, Poisson比
- ・実数型 stiff(nn*6, nn*6)
要素剛性マトリックス

上位

- ・サブルーチン [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::FSTR_LOCAL_STF_CREATE()

下位

なし

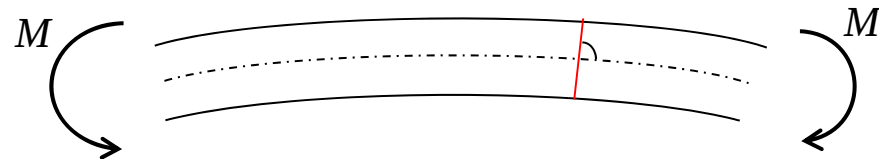
FrontISTRのシェル要素の仮定

- (a) 時刻 $t = 0$ にシェルの中立面に垂直な直線はシェルが変形する間も直線であるが、変形した中立面に垂直である必要はない
(せん断変形は考慮される)

Reissner-Mindlin板の仮定

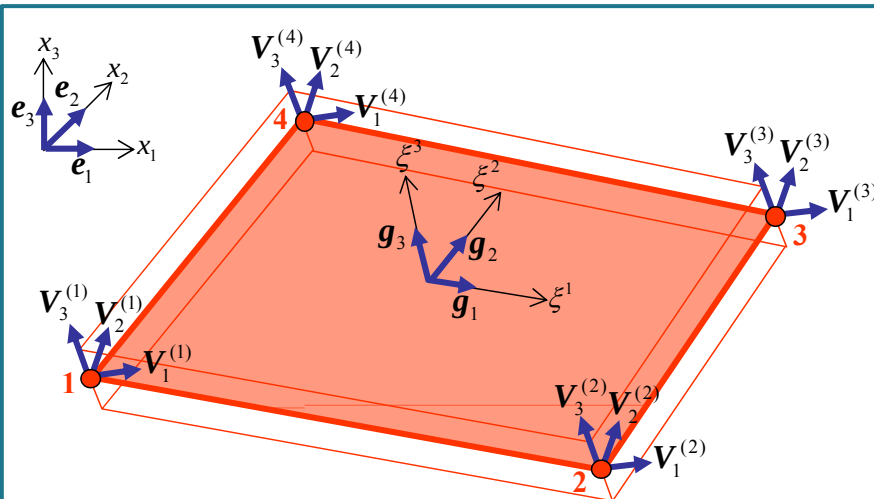
- (b) シェルのディレクタは変形しない
微小ひずみの仮定

$t = 0$

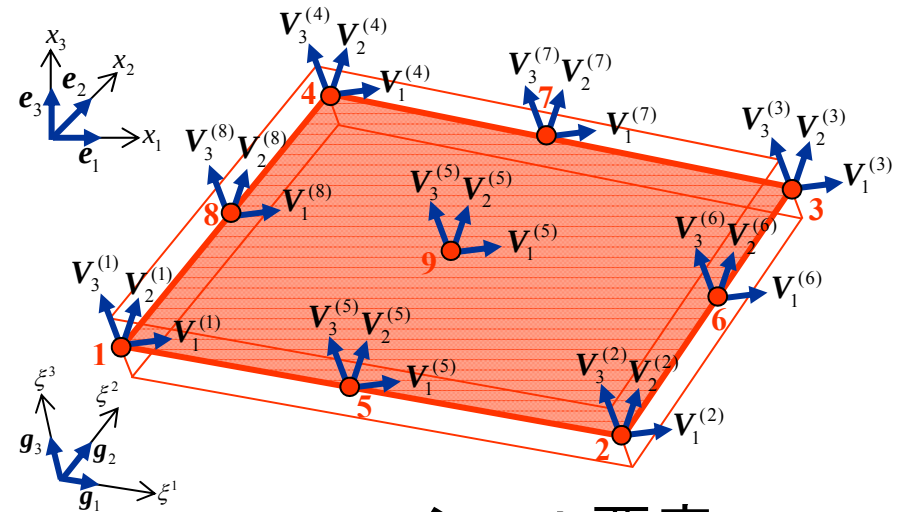


FrontISTRで利用できるシェル要素

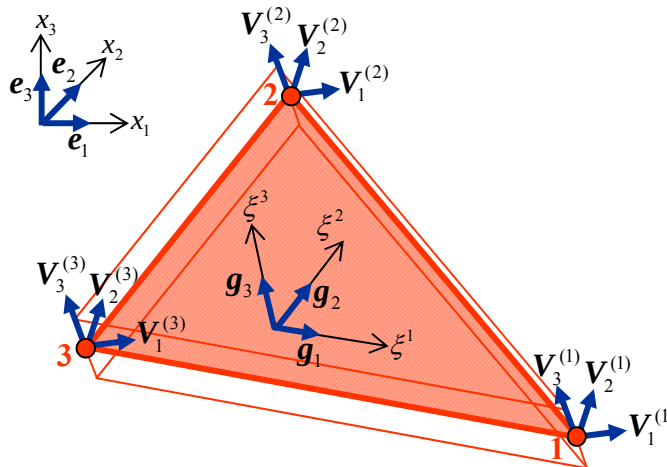
FrontISTRでは微小変形を仮定したMITCシェル要素が実装されている



MITC4シェル要素



MITC9シェル要素



MITC3シェル要素

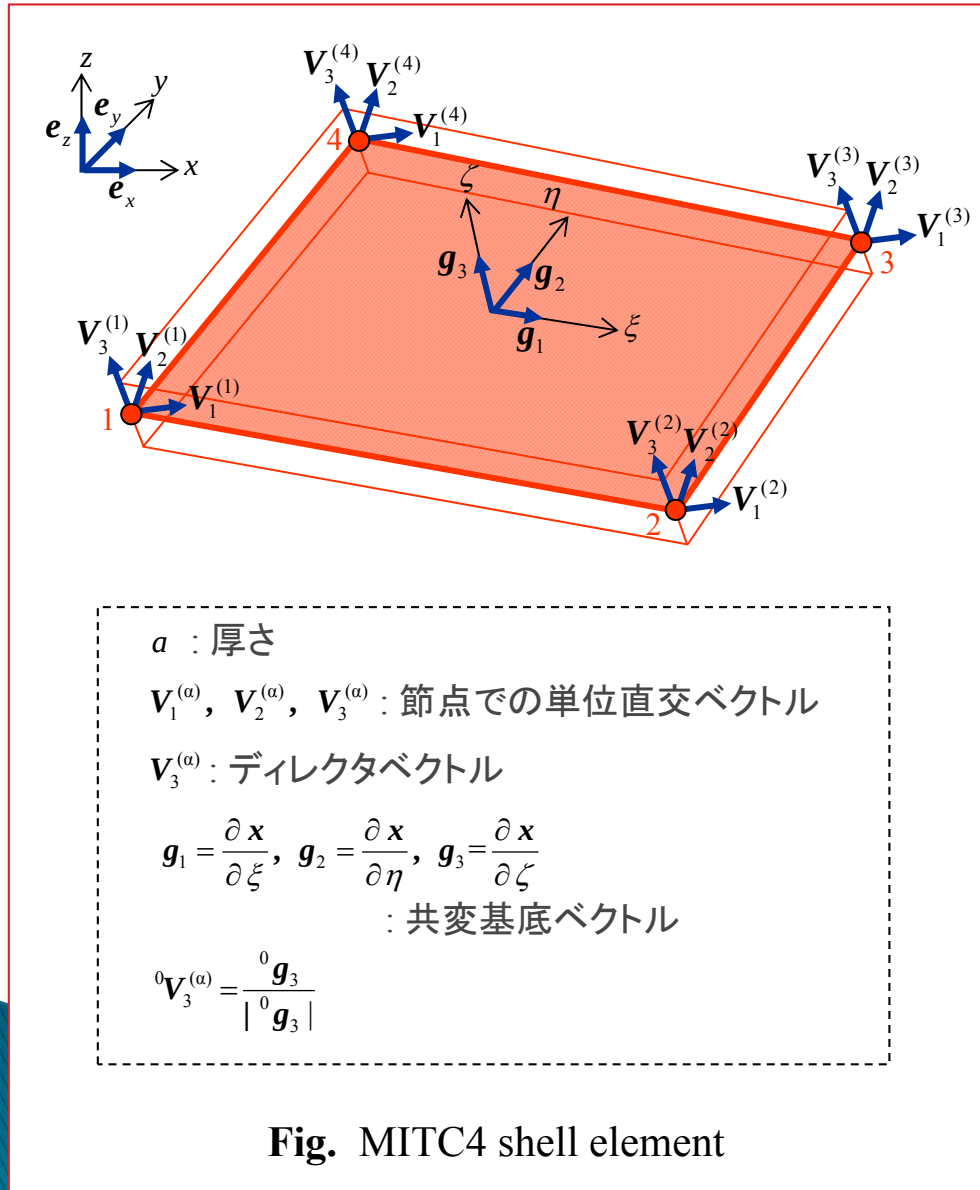
Bathe, K.J., *Finite Element Procedures*, Prentice Hall, (1995).

MITC (Mixed Interpolation of Tensorial Components) shell element

- (a) Isoparametric **degenerated** shell element
- (b) The **covariant components** measured in the convected system are used as the Green-Lagrange strain components.
- (c) The transverse shear strain components are evaluated using **the values interpolated at sampling points**.

MITC4シェル要素 (1/4)

Dvorkin, E.N. and Bathe, K.J., "A Continuum Mechanics Based Four-node Shell Element for General Non-linear Analysis," *Engineering Computations*, Vol.1, pp.77-88, (1984).



位置ベクトル

$$\mathbf{x} = \sum_{\alpha=1}^4 N^{(\alpha)}(\xi, \eta) \left(\mathbf{x}^{(\alpha)} + \frac{a}{2} \zeta \mathbf{V}_3^{(\alpha)} \right)$$

変位ベクトル

$$\begin{aligned} \mathbf{u} &= \mathbf{x} - {}^0\mathbf{x} \\ &= \sum_{\alpha=1}^4 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (\mathbf{V}_3^{(\alpha)} - {}^0\mathbf{V}_3^{(\alpha)}) \right\} \end{aligned}$$

(微小変形を仮定)

$$\begin{aligned} \mathbf{u} &= \sum_{\alpha=1}^4 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (\mathbf{V}_3^{(\alpha)} - {}^0\mathbf{V}_3^{(\alpha)}) \right\} \\ &\simeq \sum_{\alpha=1}^4 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (\boldsymbol{\theta}^{(\alpha)} \times {}^0\mathbf{V}_3^{(\alpha)}) \right\} \end{aligned}$$

$$\begin{cases} \mathbf{u}^{(\alpha)} = \underline{u_x^{(\alpha)}} \mathbf{e}_x + \underline{u_y^{(\alpha)}} \mathbf{e}_y + \underline{u_z^{(\alpha)}} \mathbf{e}_z \\ \boldsymbol{\theta}^{(\alpha)} = \underline{\psi_1^{(\alpha)}} \mathbf{V}_1^{(\alpha)} + \underline{\psi_2^{(\alpha)}} \mathbf{V}_2^{(\alpha)} \end{cases}$$

節点あたり5自由度

MITC4シェル要素 (2/4)

Dvorkin, E.N. and Bathe, K.J., "A Continuum Mechanics Based Four-node Shell Element for General Non-linear Analysis," *Engineering Computations*, Vol.1, pp.77-88, (1984).

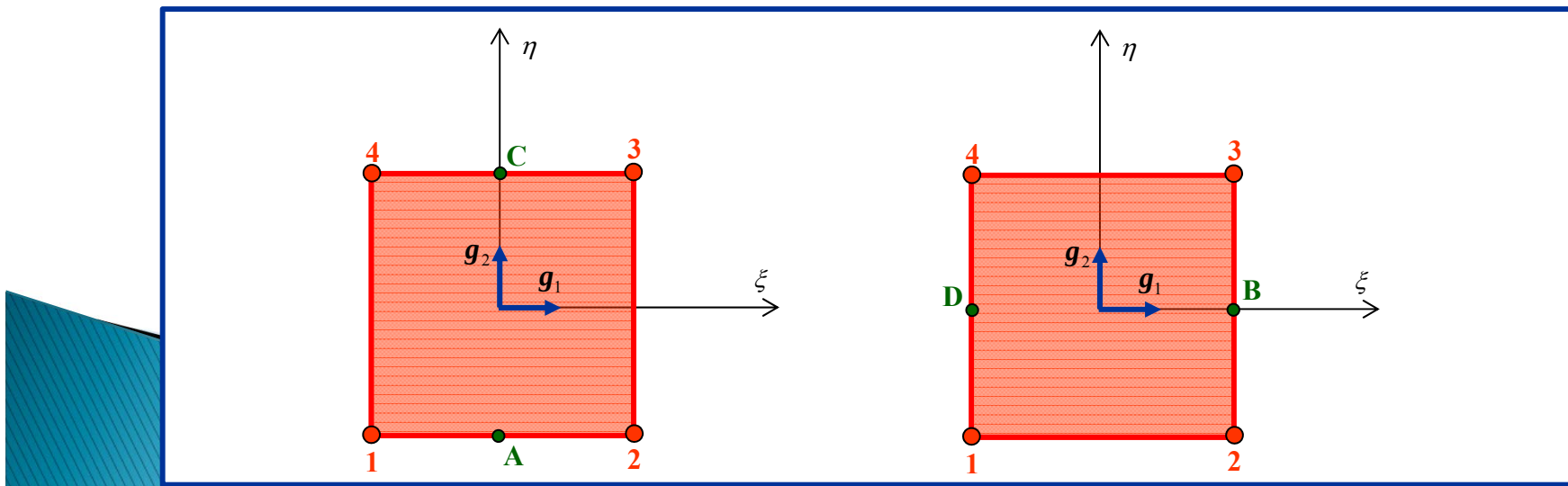
$$\Phi^* = \Phi + \int_V \lambda^{31} (\varepsilon_{31} - \varepsilon_{31}^{DI}) dV + \int_V \lambda^{23} (\varepsilon_{23} - \varepsilon_{23}^{DI}) dV$$

$$\Phi = \int_V \frac{1}{2} \boldsymbol{\sigma} : \boldsymbol{\varepsilon} dV - \int_{S_t} \mathbf{u} \cdot \underline{\mathbf{t}} dS - \int_V \mathbf{u} \cdot \rho \mathbf{b} dV$$

$$\boldsymbol{\sigma} = \mathbf{C} : \boldsymbol{\varepsilon} \quad \begin{cases} \boldsymbol{\sigma} = \sigma^{ij} \mathbf{g}_i \otimes \mathbf{g}_j \\ \boldsymbol{\varepsilon} = \varepsilon_{ij} \mathbf{g}^i \otimes \mathbf{g}^j \end{cases} \quad \text{Tensorial components}$$

$$\begin{cases} \varepsilon_{31} = \varepsilon_{31}^{AS} = \frac{1}{2}(1-\eta) \varepsilon_{31,A}^{AS} + \frac{1}{2}(1+\eta) \varepsilon_{31,C}^{AS} \\ \varepsilon_{23} = \varepsilon_{23}^{AS} = \frac{1}{2}(1-\xi) \varepsilon_{23,D}^{AS} + \frac{1}{2}(1+\xi) \varepsilon_{23,B}^{AS} \end{cases}$$

$$\begin{cases} \lambda^{31} = \delta(\xi) \delta(1+\eta) \lambda_A + \delta(\xi) \delta(1-\eta) \lambda_C \\ \lambda^{23} = \delta(1+\xi) \delta(\eta) \lambda_D + \delta(1-\xi) \delta(\eta) \lambda_B \end{cases}$$



MITC4シェル要素 (3/4)

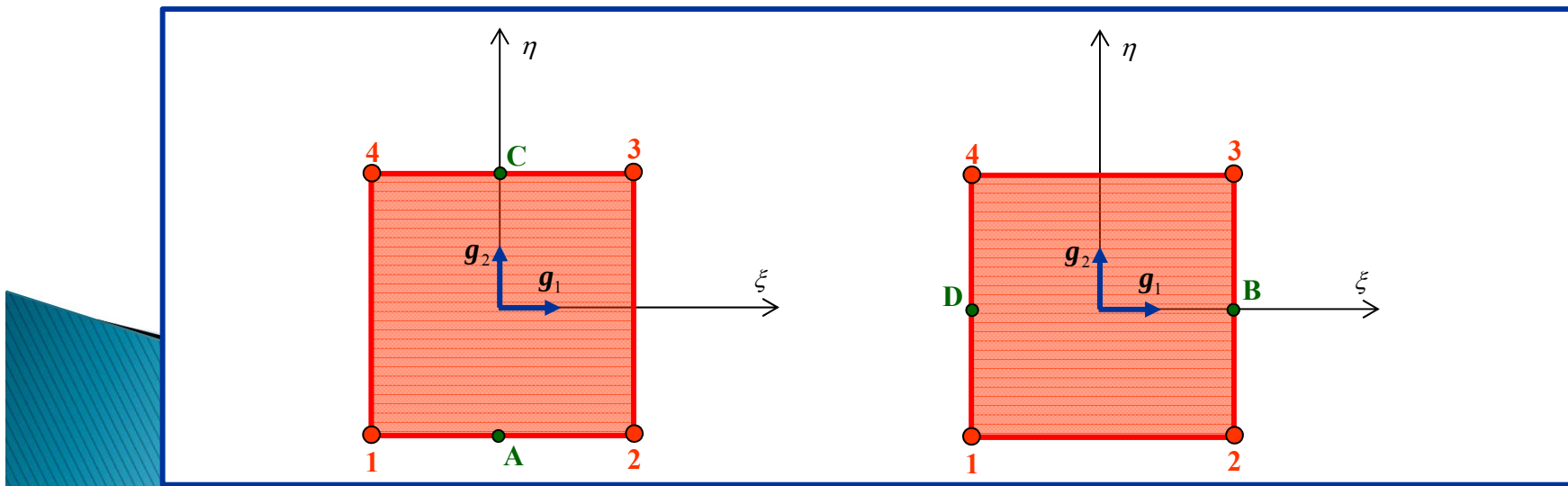
Dvorkin, E.N. and Bathe, K.J., "A Continuum Mechanics Based Four-node Shell Element for General Non-linear Analysis," *Engineering Computations*, Vol.1, pp.77-88, (1984).

$\delta \Phi^* = 0$ より, 仮想仕事の原理

$$\int_V \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon} \, dV = \int_{S_t} \delta \mathbf{u} \cdot \underline{\mathbf{t}} \, dS + \int_V \delta \mathbf{u} \cdot \rho \mathbf{b} \, dV$$

$$\begin{cases} \varepsilon_{31} = \varepsilon_{31}^{AS} = \frac{1}{2}(1-\eta) \varepsilon_{31}^{DI}(0, -1, 0) + \frac{1}{2}(1+\eta) \varepsilon_{31}^{DI}(0, 1, 0) \\ \varepsilon_{23} = \varepsilon_{23}^{AS} = \frac{1}{2}(1-\xi) \varepsilon_{23}^{DI}(-1, 0, 0) + \frac{1}{2}(1+\xi) \varepsilon_{23}^{DI}(1, 0, 0) \end{cases}$$

Mixed Interpolation of Tensorial Components



MITC4シェル要素 (4/4)

Dvorkin, E.N. and Bathe, K.J., "A Continuum Mechanics Based Four-node Shell Element for General Non-linear Analysis," *Engineering Computations*, Vol.1, pp.77-88, (1984).

せん断ひずみ成分

$$E_{IJ}^{AS}(\xi^1, \xi^2, \xi^3) = \sum_{k=1}^{n_{IJ}} h_{(k)}^{IJ}(\xi^1, \xi^2) E_{IJ}^{DI}(\xi_{(k)}^1, \xi_{(k)}^2, \xi^3)$$

$$E_{31}^{AS}(\xi^1, \xi^2, \xi^3) = \frac{1}{2} (1 - \xi^2) E_{31}^{DI}(0, -1, 0) + \frac{1}{2} (1 + \xi^2) E_{31}^{DI}(0, 1, 0)$$

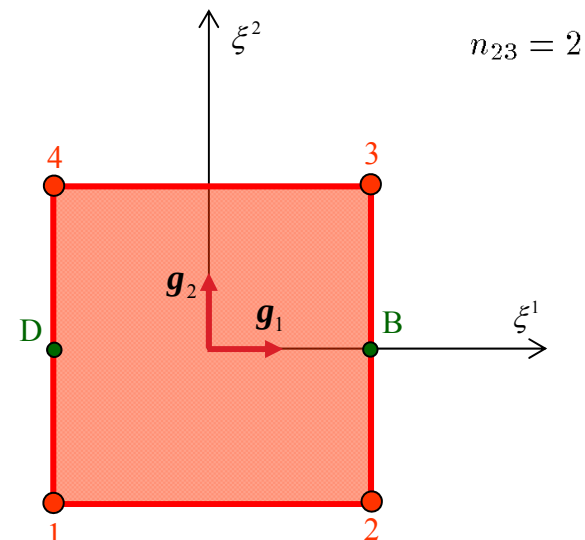
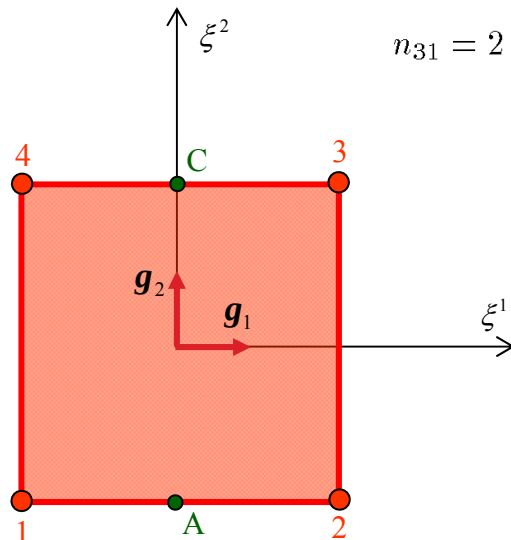
$$E_{23}^{AS}(\xi^1, \xi^2, \xi^3) = \frac{1}{2} (1 - \xi^1) E_{23}^{DI}(-1, 0, 0) + \frac{1}{2} (1 + \xi^1) E_{23}^{DI}(1, 0, 0)$$

$$h_{(k)}^{31}(\xi^1, \xi^2) = \left[\frac{1}{2} (1 + \xi_{(k)}^2 \xi^2) \right]$$

$$h_{(k)}^{23}(\xi^1, \xi^2) = \left[\frac{1}{2} (1 + \xi_{(k)}^1 \xi^1) \right]$$

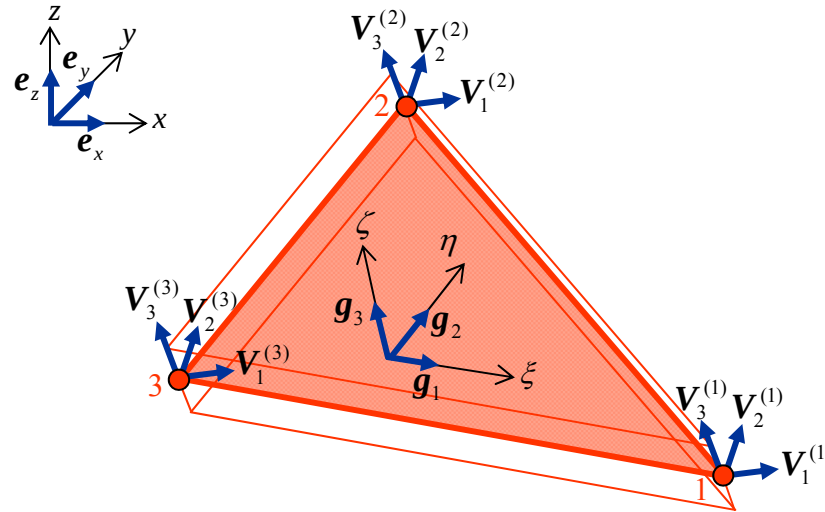
$$\begin{cases} \{\xi_{(k)}^1\} = \{0, 0\} \\ \{\xi_{(k)}^2\} = \{-1, 1\} \end{cases}$$

$$\begin{cases} \{\xi_{(k)}^1\} = \{-1, 1\} \\ \{\xi_{(k)}^2\} = \{0, 0\} \end{cases}$$



MITC3シェル要素 (1/2)

Lee, P.S. and Bathe, K.J., "Development of MITC Isotropic Triangular Shell Finite Elements, Computers & Structures, Vol.82, pp.945-962, (2004).



a : 厚さ

$V_1^{(\alpha)}, V_2^{(\alpha)}, V_3^{(\alpha)}$: 節点での単位直交ベクトル

$V_3^{(\alpha)}$: ディレクタベクトル

$$g_1 = \frac{\partial \mathbf{x}}{\partial \xi}, g_2 = \frac{\partial \mathbf{x}}{\partial \eta}, g_3 = \frac{\partial \mathbf{x}}{\partial \zeta}$$

: 共変基底ベクトル

$${}^0V_3^{(\alpha)} = \frac{{}^0g_3}{|{}^0g_3|}$$

Fig. MITC4 shell element

位置ベクトル

$$\mathbf{x} = \sum_{\alpha=1}^3 N^{(\alpha)}(\xi, \eta) \left(\mathbf{x}^{(\alpha)} + \frac{a}{2} \zeta V_3^{(\alpha)} \right)$$

変位ベクトル

$$\begin{aligned} \mathbf{u} &= \mathbf{x} - {}^0\mathbf{x} \\ &= \sum_{\alpha=1}^3 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (V_3^{(\alpha)} - {}^0V_3^{(\alpha)}) \right\} \end{aligned}$$

(微小変形を仮定)

$$\begin{aligned} \mathbf{u} &= \sum_{\alpha=1}^3 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (V_3^{(\alpha)} - {}^0V_3^{(\alpha)}) \right\} \\ &\simeq \sum_{\alpha=1}^3 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (\boldsymbol{\theta}^{(\alpha)} \times {}^0V_3^{(\alpha)}) \right\} \end{aligned}$$

$$\begin{cases} \mathbf{u}^{(\alpha)} = \underline{u_x^{(\alpha)}} \mathbf{e}_x + \underline{u_y^{(\alpha)}} \mathbf{e}_y + \underline{u_z^{(\alpha)}} \mathbf{e}_z \\ \boldsymbol{\theta}^{(\alpha)} = \underline{\psi_1^{(\alpha)}} V_1^{(\alpha)} + \underline{\psi_2^{(\alpha)}} V_2^{(\alpha)} \end{cases}$$

節点あたり5自由度

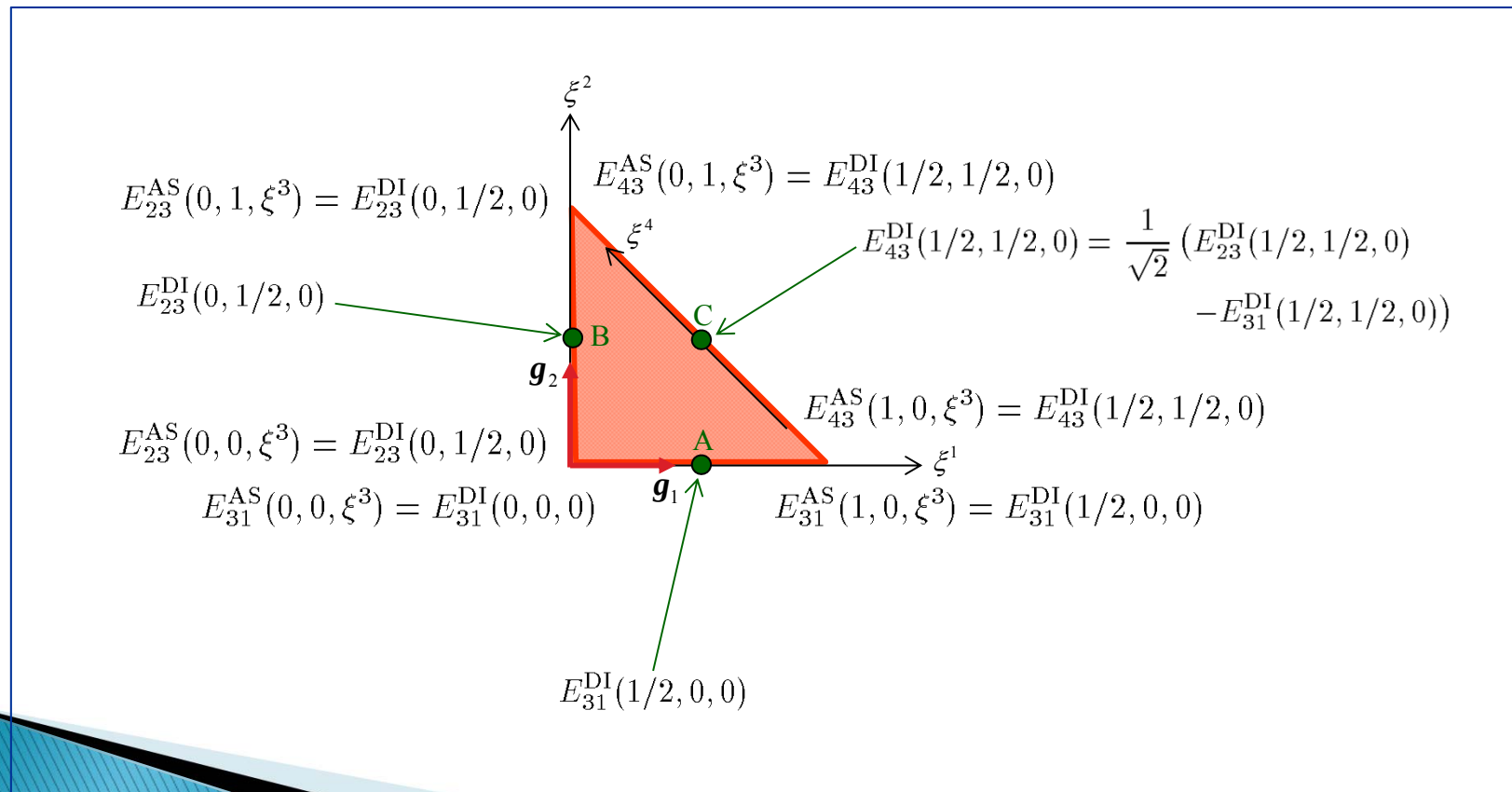
MITC3シェル要素 (2/2)

Lee, P.S. and Bathe, K.J., "Development of MITC Isotropic Triangular Shell Finite Elements, Computers & Structures, Vol.82, pp.945-962, (2004).

せん断ひずみ成分

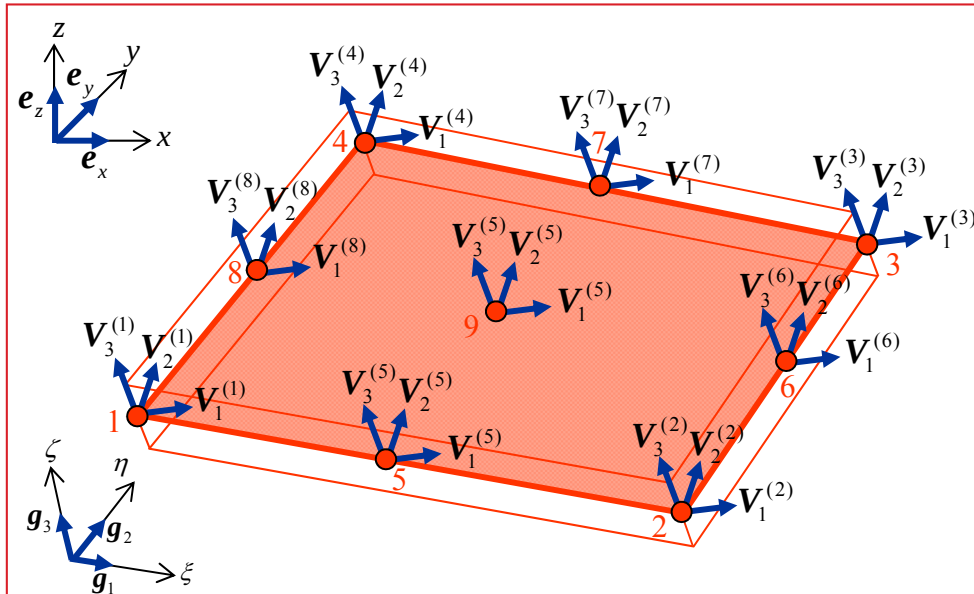
$$E_{23}^{AS}(\xi^1, \xi^2, \xi^3) = E_{23}^{DI}(0, 1/2, 0) + c_2 \xi^1$$

$$E_{31}^{AS}(\xi^1, \xi^2, \xi^3) = E_{31}^{DI}(1/2, 0, 0) + c_1 \xi^2$$



MITC9シェル要素 (1/2)

Bucalem, M.L. and Bathe, K.J., "Higher-order MITC general shell element," International Journal for Numerical Methods in Engineering, Vol.36, pp.3729-3754, (1993).



a : 厚さ

$V_1^{(\alpha)}, V_2^{(\alpha)}, V_3^{(\alpha)}$: 節点での単位直交ベクトル

$V_3^{(\alpha)}$: ディレクタベクトル

$$g_1 = \frac{\partial \mathbf{x}}{\partial \xi}, g_2 = \frac{\partial \mathbf{x}}{\partial \eta}, g_3 = \frac{\partial \mathbf{x}}{\partial \zeta}$$

: 共変基底ベクトル

$${}^0V_3^{(\alpha)} = \frac{{}^0g_3}{|{}^0g_3|}$$

Fig. MITC4 shell element

位置ベクトル

$$\mathbf{x} = \sum_{\alpha=1}^9 N^{(\alpha)}(\xi, \eta) \left(\mathbf{x}^{(\alpha)} + \frac{a}{2} \zeta V_3^{(\alpha)} \right)$$

変位ベクトル

$$\begin{aligned} \mathbf{u} &= \mathbf{x} - {}^0\mathbf{x} \\ &= \sum_{\alpha=1}^9 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (V_3^{(\alpha)} - {}^0V_3^{(\alpha)}) \right\} \end{aligned}$$

(微小変形を仮定)

$$\begin{aligned} \mathbf{u} &= \sum_{\alpha=1}^9 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (V_3^{(\alpha)} - {}^0V_3^{(\alpha)}) \right\} \\ &\approx \sum_{\alpha=1}^9 N^{(\alpha)}(\xi, \eta) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \zeta (\boldsymbol{\theta}^{(\alpha)} \times {}^0V_3^{(\alpha)}) \right\} \end{aligned}$$

$$\begin{cases} \mathbf{u}^{(\alpha)} = \underline{u_x^{(\alpha)}} \mathbf{e}_x + \underline{u_y^{(\alpha)}} \mathbf{e}_y + \underline{u_z^{(\alpha)}} \mathbf{e}_z \\ \boldsymbol{\theta}^{(\alpha)} = \underline{\psi_1^{(\alpha)}} V_1^{(\alpha)} + \underline{\psi_2^{(\alpha)}} V_2^{(\alpha)} \end{cases}$$

節点あたり5自由度

MITC9シェル要素 (2/2)

Bucalem, M.L. and Bathe, K.J., "Higher-order MITC general shell element," International Journal for Numerical Methods in Engineering, Vol.36, pp.3729-3754, (1993).

せん断ひずみ成分

$$E_{IJ}^{AS}(\xi^1, \xi^2, \xi^3) = \sum_{k=1}^{n_{IJ}} h_{(k)}^{IJ}(\Xi^1, \Xi^2) E_{IJ}^{DI}(\xi_{(k)}^1, \xi_{(k)}^2, \xi^3)$$

$$\begin{aligned} & h_{(k)}^{11}(\Xi^1, \Xi^2) \\ &= h_{(k)}^{31}(\Xi^1, \Xi^2) \\ &= \left[\frac{1}{2} (1 + \Xi_{(k)}^1 \Xi^1) \right] \\ & \quad \left[\frac{\Xi_{(k)}^2 \Xi^2}{2} (1 + \Xi_{(k)}^2 \Xi^2) + (1 - \Xi_{(k)}^2 \Xi_{(k)}^2)(1 - \Xi^2 \Xi^2) \right] \end{aligned}$$

$$\begin{aligned} & h_{(k)}^{22}(\Xi^1, \Xi^2) \\ &= h_{(k)}^{23}(\Xi^1, \Xi^2) \\ &= \left[\frac{\Xi_{(k)}^1 \Xi^1}{2} (1 + \Xi_{(k)}^1 \Xi^1) + (1 - \Xi_{(k)}^1 \Xi_{(k)}^1)(1 - \Xi^1 \Xi^1) \right] \\ & \quad \left[\frac{1}{2} (1 + \Xi_{(k)}^2 \Xi^2) \right] \end{aligned}$$

$$h_{(k)}^{12}(\Xi^1, \Xi^2) = \left[\frac{1}{2} (1 + \Xi_{(k)}^1 \Xi^1) \right] \left[\frac{1}{2} (1 + \Xi_{(k)}^2 \Xi^2) \right]$$

$$\begin{cases} \Xi^1 = \frac{\xi^1}{\sqrt{1/3}} \\ \Xi^2 = \frac{\xi^2}{\sqrt{3/5}} \end{cases}$$

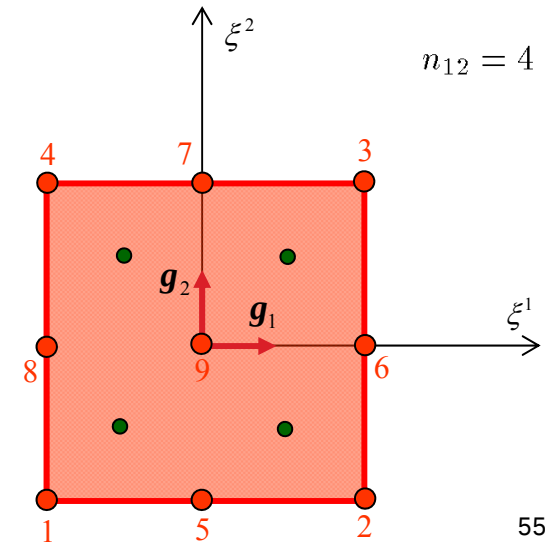
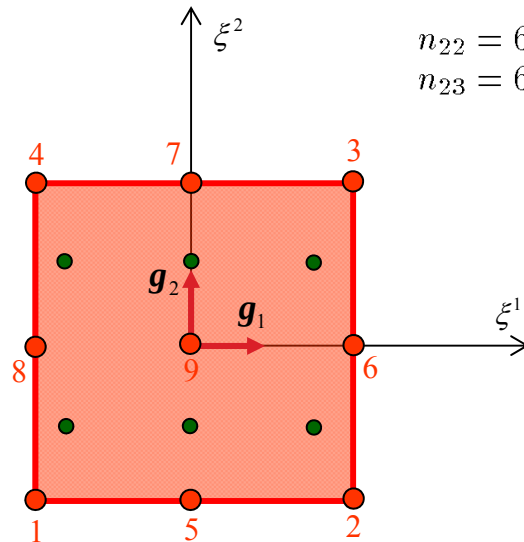
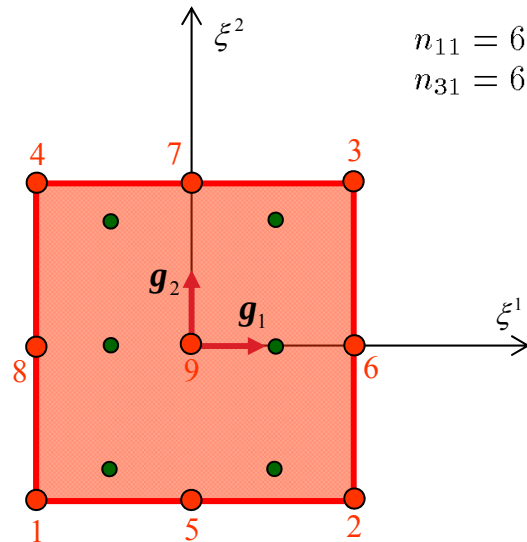
$$\begin{cases} \Xi^1 = \frac{\xi^1}{\sqrt{3/5}} \\ \Xi^2 = \frac{\xi^2}{\sqrt{1/3}} \end{cases}$$

$$\begin{cases} \Xi^1 = \frac{\xi^1}{\sqrt{1/3}} \\ \Xi^2 = \frac{\xi^2}{\sqrt{1/3}} \end{cases}$$

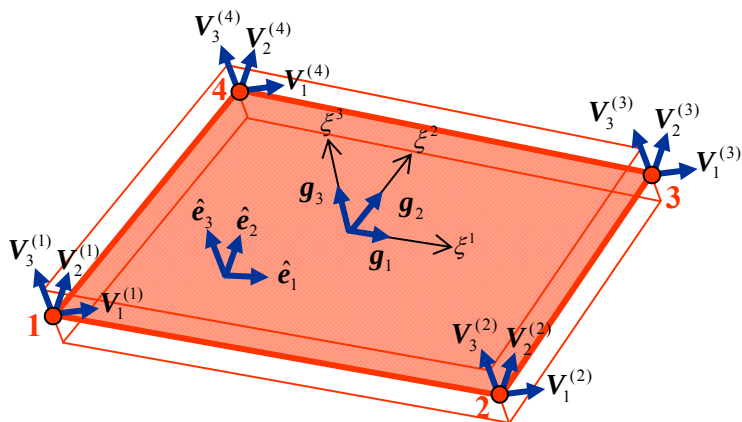
$$\begin{cases} \{\Xi_{(k)}^1\} = \{-1, 1, 1, -1, 1, -1\} \\ \{\Xi_{(k)}^2\} = \{-1, -1, 1, 1, 0, 0\} \end{cases}$$

$$\begin{cases} \{\Xi_{(k)}^1\} = \{-1, 0, 1, 1, 0, -1\} \\ \{\Xi_{(k)}^2\} = \{-1, -1, -1, 1, 1, 1\} \end{cases}$$

$$\begin{cases} \{\Xi_{(k)}^1\} = \{-1, 1, 1, -1\} \\ \{\Xi_{(k)}^2\} = \{-1, -1, 1, 1\} \end{cases}$$



弾性テンソル成分の変換



$V_1^{(\alpha)}, V_2^{(\alpha)}, V_3^{(\alpha)}$: 節点での単位直交ベクトル

$$g_1 = \frac{\partial \mathbf{x}}{\partial \xi}, \quad g_2 = \frac{\partial \mathbf{x}}{\partial \eta}, \quad g_3 = \frac{\partial \mathbf{x}}{\partial \zeta}$$

: 共変基底ベクトル

$\hat{e}_1, \hat{e}_2, \hat{e}_3$: 単位直交ベクトル

$$\hat{e}_3 = \frac{g_3}{|g_3|}, \quad \hat{e}_1 = \frac{g_2 \times \hat{e}_3}{|g_2 \times \hat{e}_3|}, \quad \hat{e}_2 = \hat{e}_3 \times \hat{e}_1$$

$$V_3^{(\alpha)} = \hat{e}_3, \quad V_2^{(\alpha)} = \frac{\hat{e}_3 \times \mathbf{e}_1}{|\hat{e}_3 \times \mathbf{e}_1|}, \quad V_1^{(\alpha)} = V_2^{(\alpha)} \times V_3^{(\alpha)}$$

各節点において

弾性テンソル

$$\begin{aligned} \mathbf{C} &= C^{ijkl} {}^0g_i \otimes {}^0g_j \otimes {}^0g_k \otimes {}^0g_l \\ &= \hat{C}^{ijkl} \hat{e}_i \otimes \hat{e}_j \otimes \hat{e}_k \otimes \hat{e}_l \end{aligned}$$

$$\begin{pmatrix} \hat{C}^{1111} & \hat{C}^{1122} & \hat{C}^{1112} & \hat{C}^{1123} & \hat{C}^{1131} \\ \hat{C}^{2211} & \hat{C}^{2222} & \hat{C}^{2212} & \hat{C}^{2223} & \hat{C}^{2231} \\ \hat{C}^{1211} & \hat{C}^{1222} & \hat{C}^{1212} & \hat{C}^{1223} & \hat{C}^{1231} \\ \hat{C}^{2311} & \hat{C}^{2322} & \hat{C}^{2312} & \hat{C}^{2323} & \hat{C}^{2331} \\ \hat{C}^{3111} & \hat{C}^{3122} & \hat{C}^{3112} & \hat{C}^{3123} & \hat{C}^{3131} \end{pmatrix} = \frac{E}{1-\nu^2} \begin{pmatrix} 1 & \nu & 0 & 0 & 0 \\ \nu & 1 & 0 & 0 & 0 \\ 0 & 0 & \frac{1-\nu}{2} & 0 & 0 \\ 0 & 0 & 0 & k \frac{1-\nu}{2} & 0 \\ 0 & 0 & 0 & 0 & k \frac{1-\nu}{2} \end{pmatrix}$$

E : Young率

ν : Poisson比

k : せん断補正係数

$$\begin{aligned} C^{ijkl} &= (\mathbf{C} : {}^0g^i \otimes {}^0g^j) : {}^0g^k \otimes {}^0g^l \\ &= \hat{C}^{mnop} ({}^0\hat{e}_m \cdot {}^0g^i) ({}^0\hat{e}_n \cdot {}^0g^j) ({}^0\hat{e}_o \cdot {}^0g^k) ({}^0\hat{e}_p \cdot {}^0g^l) \end{aligned}$$

シェル要素による定式化の注意

- (1) 5自由度として計算する場合
(ディレクタ回りの回転自由度を計算しない場合)
 - (2) 6自由度として計算する場合
(ディレクタ回りの回転自由度をdrilling DOFとして計算する場合)
-
- (A) 隣接要素の節点で同じディレクタベクトルを使用する場合
 - (B) 隣接要素の節点で別のディレクタベクトルを使用する場合

FrontISTRでは、(2) と (B) の定式化を使用しています



Drilling DOFの考慮 (1/2)

シェル要素の節点あたりの自由度数を6として, 六つ目の自由度 (**drilling DOF**と呼ばれる三つ目の回転自由度) を考慮 (Hughes and Brezzi 1989)

$$\begin{aligned} \mathbf{u} &= \sum_{\alpha=1}^n N^{(\alpha)}(\xi^1, \xi^2) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \xi^3 (\mathbf{V}_3^{(\alpha)} - {}^0\mathbf{V}_3^{(\alpha)}) \right\} & \begin{cases} \mathbf{u}^{(\alpha)} = u_x^{(\alpha)} \mathbf{e}_x + u_y^{(\alpha)} \mathbf{e}_y + u_z^{(\alpha)} \mathbf{e}_z \\ \boldsymbol{\theta}^{(\alpha)} = \theta_x^{(\alpha)} \mathbf{e}_x + \theta_y^{(\alpha)} \mathbf{e}_y + \theta_z^{(\alpha)} \mathbf{e}_z \end{cases} \\ &\simeq \sum_{\alpha=1}^n N^{(\alpha)}(\xi^1, \xi^2) \left\{ \mathbf{u}^{(\alpha)} + \frac{a}{2} \xi^3 (\boldsymbol{\theta}^{(\alpha)} \times {}^0\mathbf{V}_3^{(\alpha)}) \right\} \end{aligned}$$

drilling DOFを考慮した場合のエネルギー

$$\begin{aligned} \Phi^{**} &= \int_V \frac{1}{2} \boldsymbol{\sigma} : \boldsymbol{\varepsilon} dV - \int_{S_t} \mathbf{u} \cdot \underline{\mathbf{t}} dS - \int_V \mathbf{u} \cdot \rho \mathbf{b} dV + \int_V \lambda^{31} (\varepsilon_{31} - \varepsilon_{31}^{\text{DI}}) dV + \int_V \lambda^{23} (\varepsilon_{23} - \varepsilon_{23}^{\text{DI}}) dV \\ &\quad + \int_V \frac{\alpha}{2} \left\{ {}^0\mathbf{V}_3 \cdot \left(\boldsymbol{\theta} - \frac{1}{2} \mathbf{e} : \boldsymbol{\Theta} \right) \right\} \left\{ {}^0\mathbf{V}_3 \cdot \left(\boldsymbol{\theta} - \frac{1}{2} \mathbf{e} : \boldsymbol{\Theta} \right) \right\} dV \quad (\alpha / \mu = 0.0001 \sim 1.0) \end{aligned}$$

$$\boldsymbol{\Theta} = \frac{1}{2} \left\{ {}^0\nabla \otimes \mathbf{u} - ({}^0\nabla \otimes \mathbf{u})^T \right\} \quad \mathbf{e} = e_{ijk} \mathbf{e}_i \otimes \mathbf{e}_j \otimes \mathbf{e}_k$$

Hughes, T.J.R. and Brezzi, F., *Computer Methods in Applied Mechanics and Engineering*, Vol.72, pp.105-121, (1989).

Nguyen-Van, Hieu and Mai-Duy, Nam and Tran-Cong, Thanh, *CMES: Computer Modeling in Engineering and Sciences*, Vol.49, No.2, pp.81-110, (2009).

Drilling DOFの考慮 (2/2)

$\delta \Phi^{**} = 0$ より, 仮想仕事の原理

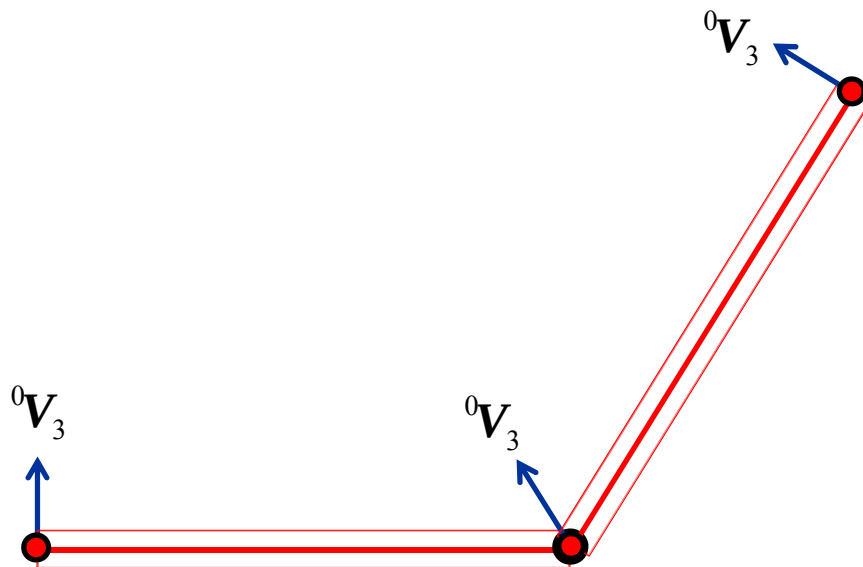
$$\begin{aligned} \int_V \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon} \, dV + \int_V \alpha \left\{ {}^0V_3 \cdot \left(\boldsymbol{\theta} - \frac{1}{2} \mathbf{e} : \boldsymbol{\Theta} \right) \right\} \left\{ {}^0V_3 \cdot \left(\delta \boldsymbol{\theta} - \frac{1}{2} \mathbf{e} : \delta \boldsymbol{\Theta} \right) \right\} dV \\ = \int_{S_t} \delta \mathbf{u} \cdot \underline{\mathbf{t}} \, dS + \int_V \delta \mathbf{u} \cdot \rho \mathbf{b} \, dV \end{aligned}$$

$$(\alpha / \mu = 0.0001 \sim 1.0)$$

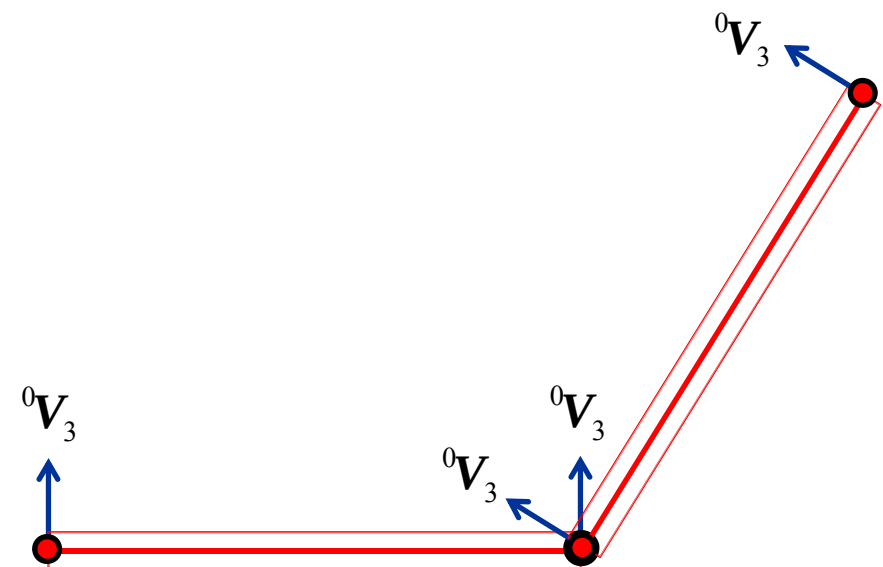
$$\begin{cases} \varepsilon_{31} = \varepsilon_{31}^{\text{AS}} = \frac{1}{2}(1 - \eta) \varepsilon_{31}^{\text{DI}}(0, -1, 0) + \frac{1}{2}(1 + \eta) \varepsilon_{31}^{\text{DI}}(0, 1, 0) \\ \varepsilon_{23} = \varepsilon_{23}^{\text{AS}} = \frac{1}{2}(1 - \xi) \varepsilon_{23}^{\text{DI}}(-1, 0, 0) + \frac{1}{2}(1 + \xi) \varepsilon_{23}^{\text{DI}}(1, 0, 0) \end{cases}$$

変形前のディレクタベクトル

(A) 隣接要素の節点で
同じディレクタベクトル
を使用する場合



(B) 隣接要素の節点で
別のディレクタベクトルを
を使用する場合



FrontISTRでは (B) で実装されて
います

[lib/physics/ElasticLinear.f90]の内容

モジュール名 : m_ElasticLinear
線形弾性体のDマトリックスを計算するモジュール

使用する他のモジュール

- ・ `lib/physics/material.f90` mMaterial
材料物性の情報を管理するモジュール

メンバ変数

- ・ 整数型 `kreal`
実数型の種別値

メンバ関数

- ・ サブルーチン `calElasticMatrix()`
3次元問題, 平面ひずみ問題, 平面応力問題, 軸対称問題のDマトリックスを計算するサブルーチン
- ・ サブルーチン `calElasticMatrix_ortho()`
直交異方性がある場合, 3次元問題のDマトリックスを計算するサブルーチン
- ・ サブルーチン `LinearElastic_Shell()`
シェル要素を使用する場合, 埋め込み座標系成分の弾性テンソルを計算するサブルーチン

[lib/physics/ElasticLinear.f90]

m_ElasticLinear::calLinearElastic_Shell()の内容

サブルーチン名: **LinearElastic_Shell()**

MITCシェル要素の弾性テンソルを計算するサブルーチン

引数

- ・構造体(tMaterial) matl
材料に関連するデータ
- ・整数型 sectType
問題の種類 (3次元問題/平面ひずみ/平面応力問題/軸対称問題)
- ・実数型 c(:, :, :, :)
弾性テンソルの成分
- ・実数型 e1_hat(3), e2_hat(3), e3_hat(3),
直交座標系の単位基底ベクトル
- ・実数型 cg1(3), cg2(3), cg3(3),
反変基底ベクトル
- ・実数型 alpha
drilling DOFのための係数
- ・整数型 n_layer
積層の数

上位

- ・サブルーチン [lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix_Shell()

下位

- ・サブルーチン [lib/physics/ElasticLinear.f90] m_ElasticLinear::LinearElastic_Shell()

[lib/physics/calMatMatrix.f90]の内容

モジュール名 : m_MatMatrix

各材料のDマトリックスを計算するサブルーチンと呼ぶモジュール

使用する他のモジュール

- `lib/physics/material.f90` `mMaterial`
材料物性の情報を管理するモジュール
- `lib/physics/mechgauss.f90` `mMechGauss`
Gauss積分点の情報を管理するモジュール
- `lib/physics/ElasticLinear.f90` `m_ElasticLinear`
線形弾性体のDマトリックスを計算するモジュール
- `lib/physics/Hyperelastic.f90` `mHyperElastic`
超弾性体の4階の弾性テンソルを計算するモジュール
- `lib/physics/Elastoplastic.f90` `m_ElastoPlastic`
弾塑性体のDマトリックスを計算するモジュール
- `lib/physics/Viscoelastic.f90` `mViscoElastic`
粘弾性体のDマトリックスを計算するモジュール
- `lib/physics/creep.f90` `mCreep`
クリープを考慮した剛性マトリックスを計算するためのモジュール
- `lib/user/uelastic.f90` `mUElastic`
ユーザ定義の弾性体のDマトリックスを計算するモジュール
- `lib/user/umat.f90` `mUmat`
ユーザ定義の材料のDマトリックスを計算するモジュール

メンバ変数

- 整数型 `kreal`
実数型の種別値

メンバ関数

- 関数 `getNlgeomFlag()`
未使用のサブルーチン
- サブルーチン `MatlMatrix()`
各材料のDマトリックスを計算するサブルーチンと呼ぶサブルーチン
- サブルーチン `StressUpdate()`
各材料の応力とひずみを計算するサブルーチンと呼ぶサブルーチン
- サブルーチン `mat_c2d()`
材料が超弾性体の場合、4階の弾性テンソルを問題の種類（3次元問題/平面ひずみ/平面応力問題/軸対称問題）に応じたDマトリックスに変換するサブルーチン
- サブルーチン `MatlMatrix_Shell()`
シェル要素を使用する場合、各材料（現バージョンでは、線形弾性体のみ）のDマトリックスを計算するサブルーチンと呼ぶサブルーチン
- サブルーチン `mat_c2d_Shell()`
シェル要素を使用する場合、4階の弾性テンソルをDマトリックスに変換するサブルーチン

[lib/physics/calMatMatrix.f90]

m_MatMatrix::MatlMatrix_Shell()の内容

サブルーチン名: **MatlMatrix_Shell()**
MITCシェル要素のDマトリックスを計算するサブルーチン

引数

- ・構造体(tGaussStatus) gauss
Gauss積分点に関連するデータ
- ・整数型 sectType
問題の種類 (3次元問題/平面ひずみ/平面応力問題/軸対称問題)
- ・実数型 D(:, :)
Dマトリックス
- ・実数型 e1_hat(3), e2_hat(3), e3_hat(3),
直交座標系の単位基底ベクトル
- ・実数型 cg1(3), cg2(3), cg3(3),
反変基底ベクトル
- ・実数型 alpha
drilling DOFのための係数
- ・整数型 n_layer
積層の数

上位

- ・サブルーチン [lib/static_LIB_shell.f90] m_static_LIB_shell::STF_Shell_MITC()
- ・サブルーチン [lib/static_LIB_shell.f90] m_static_LIB_shell::Update_Shell_MITC()
- ・サブルーチン [lib/static_LIB_shell.f90] m_static_LIB_shell::ElementStress_Shell_MITC()

下位

- ・サブルーチン [lib/physics/ElasticLinear.f90] m_ElasticLinear::LinearElastic_Shell()

[lib/static_LIB_shell.f90]の内容

モジュール名 : m_static_LIB_3d

MITCシェル要素の要素剛性マトリックス, 外力ベクトル, 応力, ひずみを計算するモジュール

使用する他のモジュール

- ・ [lib/element/element.f90] elementInfo
要素の情報を管理するモジュール
- ・ [lib/physics/mechgauss.f90] mMechGauss
Gauss積分点の情報を管理するモジュール
- gauss_integration
- ・ [lib/physics/calMatMatrix.f90] m_MatMatrix
各材料のDマトリックスを計算するサブルーチンと呼ぶモジュール
- ・ [lib/physics/material.f90] mMaterial
材料物性の情報を管理するモジュール
- ・ [lib/utilities/utilities.f90] m_utilities
補助的なサブルーチンや関数を集めたモジュール

メンバ変数

- ・ 整数型 kint
整数型の種別値
- ・ 実数型 kreal
実数型の種別値

メンバ関数

- ・ サブルーチン STF_Shell_MITC()
要素剛性マトリックスを計算するサブルーチン
- ・ サブルーチン Udata_Shell_MITC()
Gauss積分点での応力の反変成分を計算するサブルーチン
- ・ サブルーチン ElementStress_Shell_MITC()
Gauss積分点での応力の x, y, z 座標系成分を計算するサブルーチン

[lib/static_LIB_shell.f90]

m_static_LIB_shell::STF_Shell_MITC()の内容

サブルーチン名: **STF_Shell_MITC()**
MITCシェル要素の剛性マトリックスを計算するサブルーチン

引数

- ・整数型 etype
要素タイプ
- ・整数型 nn
要素内の節点数
- ・整数型 ndof
節点の自由度数 (ndof = 6)
- ・実数型 ecoord(3, nn)
要素内の節点座標
- ・構造体(tGaussStatus) gausses(:)
Gaussの積分点に関連するデータ
- ・実数型 stiff(:, :)
要素剛性マトリックス
- ・実数型 thick
シェル要素の厚さ
- ・整数型 nlayer
積層の数
- ・整数型 mixflag
3×3ブロックングシェル要素の判別に使用するフラッグ
- ・実数型 nddisp(3, nn) 省略可能
節点変位

上位

- ・サブルーチン [analysis/static/static_mat_ass_main.f90] m_static_mat_ass_main::FSTR_LOCAL_STF_CREATE()
- ・サブルーチン [analysis/static/fstr_Update.f90] m_fstr_Update::fstr_Update6D()

下位

- ・サブルーチン [lib/element/element.f90] elementInfo::getNodalNaturalCoord()
- ・サブルーチン [lib/element/element.f90] elementInfo::getShapeFunc()
- ・サブルーチン [lib/element/element.f90] elementInfo::getShapeDeriv()
- ・サブルーチン [lib/element/element.f90] elementInfo::getQuadPoint()
- ・サブルーチン [lib/physics/calMatMatrix.f90] m_MatMatrix::MatlMatrix_Shell()