

- [1 FrontISTR 解析マニュアル](#)
  - [1.1 マニュアルリスト](#)
  - [1.2 本マニュアルの記載内容](#)
  - [1.3 解析の流れと入出力データの概略](#)
    - [1.3.1 解析の流れ](#)
    - [1.3.2 入力データの概略](#)
    - [1.3.3 出力ファイルの概略](#)
    - [1.3.4 全体制御データ](#)
    - [1.3.5 メッシュデータ](#)
    - [1.3.6 解析制御データ](#)
    - [1.3.7 出力ファイル](#)
    - [1.3.8 実行方法](#)
    - [1.3.9 実行時の制約](#)
  - [1.4 解析手順](#)
    - [1.4.1 逐次処理による解析](#)
    - [1.4.2 並列処理による解析](#)
  - [1.5 要素ライブラリおよび材料データ](#)
    - [1.5.1 要素ライブラリ](#)
    - [1.5.2 材料データ](#)
    - [1.5.3 非線形静解析](#)
  - [1.6 全体制御データ](#)
    - [1.6.1 全体制御データ概要](#)
    - [1.6.2 入力規則](#)
    - [1.6.3 ヘッダー一覧](#)
  - [1.7 単一領域メッシュデータ](#)
    - [1.7.1 単一メッシュデータ概要](#)
    - [1.7.2 入力規則](#)
    - [1.7.3 単一領域メッシュデータのヘッダー一覧](#)
  - [1.8 解析制御データ](#)
    - [1.8.1 解析制御データ概要](#)
    - [1.8.2 入力規則](#)
    - [1.8.3 解析制御データ](#)
    - [1.8.4 7.4.6 ソルバー制御データ](#)
    - [1.8.5 7.4.7 ポスト処理\(可視化\)制御データ](#)
  - [1.9 ユーザーサブルーチン](#)
    - [1.9.1 ユーザー定義材料の入力](#)
    - [1.9.2 弹塑性変形に関するサブルーチン \(uyield.f90\)](#)
    - [1.9.3 弹性変形に関するサブルーチン \(uelastic.f90\)](#)
    - [1.9.4 ユーザー定義材料に関するサブルーチン \(umat.f\)](#)
    - [1.9.5 ユーザー定義外部荷重の処理サブルーチン \(uload.f\)](#)
  - [1.10 ユーザーサブルーチン](#)
    - [1.10.1 ユーザー定義材料の入力](#)
    - [1.10.2 弹塑性変形に関するサブルーチン \(uyield.f90\)](#)
    - [1.10.3 弹性変形に関するサブルーチン \(uelastic.f90\)](#)
    - [1.10.4 ユーザー定義材料に関するサブルーチン \(umat.f\)](#)
    - [1.10.5 ユーザー定義外部荷重の処理サブルーチン \(uload.f\)](#)
  - [1.11 ステップ制御](#)
    - [1.11.1 解析上の時間について](#)
    - [1.11.2 静解析の制御](#)
    - [1.11.3 動解析の制御](#)

# 1 FrontISTR 解析マニュアル

本ソフトウェアは文部科学省次世代IT基盤構築のための研究開発「イノベーション基盤シミュレーションソフトウェアの研究開発」プロジェクトによる成果をシーズとして、継続的に開発されている並列有限要素解析プログラムです。本ソフトウェアを無償または営利目的でご使

用になる場合、「MITライセンス」をご了承頂くことが前提となります。



項目	説明
ソフトウェア名称	FrontISTR
バージョン	5.0
ライセンス形態	MIT License
問い合わせ先	一般社団法人 FrontISTR Commons 東京都文京区弥生二丁目11番16号 (東京大学大学院工学系研究科 総合研究機構内) E-mail : support@frontistr.com

## 1.1 マニュアルリスト

- [イントロダクション](#)
- [インストールマニュアル](#)
- [理論マニュアル](#)
- [解析マニュアル](#)
- [チュートリアル](#)

本マニュアルでは、大規模並列FEM非線形構造解析プログラムFrontISTRのデータ入力方法と実行方法を説明します。

## 1.2 本マニュアルの記載内容

- [解析の流れと入出力データの概略](#)
- [要素ライブラリおよび材料データ](#)
- [ステップ制御](#)
- [全体制御データ](#)
- [単一領域メッシュデータ](#)
- [解析制御データ](#)
- [ユーザーサブルーチン](#)

## 1.3 解析の流れと入出力データの概略

### 1.3.1 解析の流れ

ここでは、FrontISTRの解析の流れについて説明します。

1. FrontISTRならびに、周辺プログラムの実行ファイルのダウンロード、またはインストールを行います。
2. 指定のフォーマットに従って作成した入力ファイル (mesh.mshデータ、mesh.cntデータ、hecmw\_ctrl.datなど) を準備します。
3. 並列計算を行う場合は、領域分割（メッシュパーティショニング）を行うプログラム hecmw\_part1 を実行します。逐次計算の場合、この手順は必要ありません。
4. FrontISTRを実行します。
5. 並列計算を行った場合は、データマージプログラム rmerge を実行します。出力時に单一ファイルに出力する可視化フォーマットや、逐次計算の場合、この手順は必要ありません。

### **1.3.2 入力データの概略**

ここでは、FrontISTRの入力ファイルの概略について説明します。

FrontISTRは入力ファイルとして、全体制御データ、メッシュデータおよび解析制御データの3つのファイルが必要である。

モデル全体を単体のCPUで解析する場合には、単一領域メッシュファイルを使用する。複数のCPUで並列実行する場合には、HEC-MWの領域分割ツールである `hecmw_part` プログラムにより、予め単一領域メッシュデータを領域分割し、その結果としての分散領域メッシュデータを用いる。

`hecmw_part` の詳細は HEC-MW 領域分割マニュアルを参照すること。

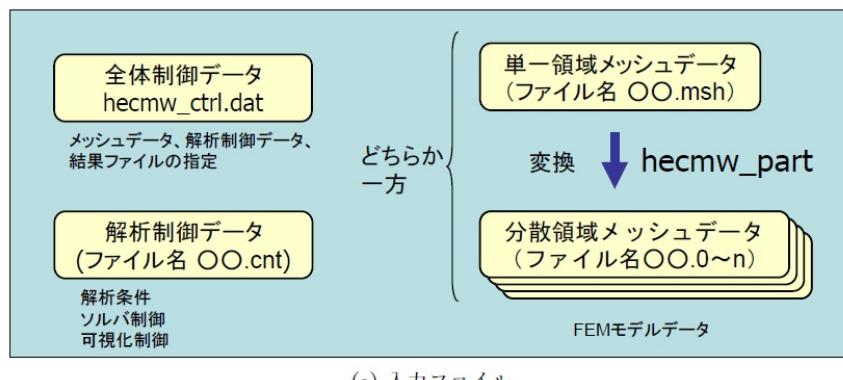
全体制御データ、解析制御データおよび単一領域メッシュデータはテキストデータであり、ユーザーはこのマニュアルの説明にしたがって、適当なエディタを用いて作成、編集することが可能であるが、FrontISTRの付属のツールとして、市販のプリポストプロセッサである Femap のサポートするニュートラルファイル (`*.neu`) を FrontISTR 入力データに変換する `neu2fstr` を用いて作成することもできる。

`neu2fstr` の詳細は `neu2fstr` マニュアルを参照すること。

### **1.3.3 出力ファイルの概略**

ここでは、FrontISTRの出力ファイルの概略について説明します。

FrontISTRの実行により、ログデータファイルと結果データファイルおよび可視化データを出力する。これらの出力の有無、内容は、解析制御ファイル中の記述および解析内容に依存する。可視化データは FrontISTR の実行後、作成された結果ファイルより、HEC-MW 付属のツールである `hecmw_vis` プログラムにより生成することも出来る。`hecmw_vis` の詳細は HEC-MW 可視化マニュアルを参照すること。



(a) 入力ファイル



(b) 出力ファイル

図 3.1.1 FrontISTR入出力ファイル

以下、上記入出力ファイルの概要について説明する。

### 1.3.4 全体制御データ

このファイルは、メッシュデータと解析制御データの入力ファイルおよび結果出力ファイルを指定する。

全体制御データの詳細は第5章に記載する。

```
#分散メッシュデータファイルのヘッダーの定義（領域分散モデルでは必須）
!MESH, NAME=fstrMSH, TYPE=HECMW-DIST
mesh.msh

#メッシュデータファイル名の定義（単一領域モデルでは必須）
!MESH, NAME=fstrMSH, TYPE=HECMW-ENTIRE
mesh.msh

#解析制御データファイル名の定義（必須）
!CONTROL, NAME=fstrCNT
mesh.cnt

#解析結果データファイル名の定義（任意）
!RESULT, NAME=fstrRES, IO=OUT
mesh.res

#可視化データファイル名の定義（任意）
!RESULT, NAME=vis_out, IO=OUT
mesh.vis
```

### 1.3.5 メッシュデータ

このファイルは有限要素メッシュを定義し、その材料データとセクションデータを定義する。また、解析制御データにて使用するグループデータを定義する。

メッシュデータの詳細は第6章に記載する。

```
#メッシュタイトルの設定
!HEADER
TEST MODEL A361

#節点座標の定義
!NODE
1, 0.0, 0.0, 0.0

#要素コネクティビティの定義
!ELEMENT, TYPE=361
1001, 1, 2, 3, 4, 5, 6, 7, 8

#セクションデータの定義
!SECTION, TYPE=SOLID, EGRP=ALL, MATERIAL=M1

#材料データの定義
!MATERIAL, NAME=M1, ITEM=1
!ITEM=1, SUBITEM=2
4000., 0.3

#節点グループの定義
!NGROUP, NGRP=FIX, GENERATE
1001, 1201, 50

#要素グループの定義
!EGROUP, EGRP=TOP, GENERATE
1001, 1201, 1

!END
```

### 1.3.6 解析制御データ

このファイルは解析の種別、変位境界条件、集中荷重など解析制御データを定義する。またソルバーの制御やビジュアライザーの制御データも、解析制御データに含まれる。

解析制御データの詳細は第7章に記載する。

```
#解析の種別の指定
!SOLUTION, TYPE=STATIC

#変位境界条件の定義
!BOUNDARY
FIX, 1, 3, 0.0

#集中荷重条件の定義
!CLOAD
CL1, 1, -1.0

#分布荷重条件の定義
!DLOAD
ALL, BX, 1.0

#参照温度の定義
!REFTEMP
20.0
```

```

#熱荷重（温度）条件の定義
!TEMPERATURE
ALL, 100.0

#ソルバーの制御
!SOLVER, METHOD=CG, PRECOND=1, TIMELOG=YES, ITERLOG=YES
10000,2
1.0e-8,1.0,0.0

#結果データ出力
!WRITE,RESULT

#メモリ渡しビジュアライザ制御
!WRITE, VISUAL

#以下、ビジュアライザーの制御データ
!visual
!surface_num =1
!surface_style =1

!END

```

### 1.3.7 出力ファイル

実行が終了すると、ログファイル(拡張子.log)が出力される。また、出力の指示により可視化用解析結果ファイル(拡張子.res)が出力される。ログファイルは、以下に示す内容が出力される。

- 変位、ひずみ、応力成分の最大・最小値
- 固有値
- 固有ベクトル値

### 1.3.8 実行方法

#### 1.3.8.1 FrontISTR の準備

FrontISTR の本体 (Linux 版は fistr1、Windows 版はfistr1.exe) をパスの通ったディレクトリまたは実行時のカレントディレクトリに格納する。

#### 1.3.8.2 入力ファイルの準備

3種類の入力ファイル hecmw\_ctrl.dat、解析制御データおよび（单一または分散領域）メッシュデータ用意し、hecmw\_ctrl.dat に解析制御データとメッシュデータのファイル名（パス名）を記述する。必要ならば、解析結果データファイルおよび可視化データファイルの指定も行っておくこと。

#### 1.3.8.3 単一領域の解析実行

Linux のターミナルもしくは Windows のコマンドプロンプトを立ち上げ、入力ファイルのあるディレクトリへカレントディレクトリを移動し、下記のように実行する（ただし'\$'はプロンプトを表す）

例) Linux の場合

```
$ ./fistr1
```

例) Window の場合

```
$ fistr1
```

#### 1.3.8.4 Linux 上での並列実行

Linux 版では予め MPIをインストールした環境で、並列実行用にコンパイルしなければならない。コンパイル方法の詳細はインストールマニュアルを参照のこと。実行は、MPIの実行環境の設定に依存する。以下に4領域での実行例を示す。

```
$ mpirun -np 4 ./fistr1
```

#### 1.3.8.5 Windows 上での並列実行

Windows版では、MPICH2 のライブラリを下記 URLよりダウンロードし、インストールする必要がある。並列実行の方法についてはMPICH2 のマニュアルを参照すること。

<http://www-unix.mcs.anl.gov/mpi/mpich/>

#### 1.3.8.6 並列接触解析の実行 (Ver.3.xからのユーザーへの注意)

Ver.3.xまでは、並列接触解析の場合のみ、分散メッシュデータではなく、単一領域メッシュデータを入力する必要があったが、Ver.5.xからは、並列接触解析の場合も、他の解析と同様に、分散メッシュデータを入力する使用方法に統一された。

#### 1.3.9 実行時の制約

FrontISTR Ver.3.5において、正常実行が確認できている機能と要素タイプを表 3.1.1に示す。

表 3.1.1 解析機能別対応要素一覧

要素番号	線形静解析	固有値解析	熱伝導解析	線形動解析	周波数応答解析	幾何学的非線形静／動解析	材料非線形	境界非線形静／動解析
111	×	×	○	×	×		×	×
112	×	×	×	×	×		×	×
231	○	○	○	○	×		×	×
232	○	○	○	○	×		×	×
241	○	○	○	○	×		×	×
242	○	○	○	○	×		×	×
301	○	×	×	○	×		○	○
341	○	○	○	○	○		○	○
342	○	○	○	○	○		○	×
351	○	○	○	○	○		○	○
352	○	○	○	○	○		○	×
361	○	○	○	○	○		○	○
362	○	○	○	○	○		○	×
541	×	×	○	×	×		×	×
542	×	×	×	×	×		×	×
611	○	×	×	○	×		×	×
641	○	×	×	○	×		×	×
731	○	○	○	○	×		×	×
732	×	×	×	×	×		×	×
741	○	○	○	○	×		×	×
743	○	○	×	○	×		×	×

761	○	×	×	×	×	×	×
781	○	×	×	×	×	×	×

注) ○: 対応 ×: 未対応

- 線形動解析では要素番号731、741、743で並列計算は未対応であるが、それ以外の要素番号での並列計算は可能である。
- 接触解析についての並列計算は直接法のみ対応している。
- 要素番号611は熱応力、重力、圧力、遠心力を考慮した解析には対応していない。
- 要素番号641は圧力、遠心力を考慮した解析には対応していない。
- 梁要素をソリッド要素と混在して解析を行う場合、利用可能な梁要素は要素番号641である。
- シェル要素をソリッド要素と混在して解析を行う場合、利用可能なシェル要素は要素番号761および781である。

## 1.4 解析手順

### 1.4.1 逐次処理による解析

#### 1.4.1.1 実行の流れ

FrontISTRを利用した、シングルプロセッサーの逐次処理による実行の流れを図3.1.2に示します。

図3.1.2 逐次処理による実行の流れ

#### 1.4.1.2 準備する入力ファイル

##### 1.4.1.2.1 (1) 全体制御データ（拡張子dat）

このファイルではメッシュデータと解析制御データの入力ファイルおよび解析結果出力ファイルを指定します。ファイル名は固定のhecmw\_ctrl.datです。

全体制御データの例を以下に示します。本例では、FrontSTRは単一領域メッシュデータmodel.mshと解析制御データmodel.cntを読み込み、解析結果データmodel.res.0.1を書き出します。また、hecmw\_visは単一領域メッシュデータmodel.mshと解析結果データmodel.res.0.1を読み込み、出力指定に対応するmodel\_vis\_psf.0000.(拡張子)を書き出します。詳細はユーザーマニュアル第5章を参照ください。

##### 1.4.1.2.2 (2) 単一領域メッシュデータ（拡張子msh）

このファイルでは、解析対象の全体メッシュ構成、材料データ、解析制御データで使用するグループデータなどを定義します。詳細はユーザーマニュアル第6章を参照ください。

##### 1.4.1.2.3 (3) 解析制御データ（拡張子cnt）

このファイルでは、解析種別、変位境界条件、荷重境界条件などを定義します。また、ソルバーの制御データやビジュアライザーの制御データも指定します。3章で解析制御データの例を示します。詳細はユーザーマニュアル第7章を参照ください。

##### 1.4.1.2.4 (4) 可視化制御データ（拡張子ini）

このファイルでは、hecmw\_visの制御データを指定します。ディフォルトのファイル名はhecmw\_vis.iniです。可視化制御データの例を以下に示します。本例では、MicroAVS用の非構造格子型データ（拡張子inp）を出力します。詳細はユーザーマニュアル7.3.3節および7.4.7節を参照ください。

#### 1.4.1.3 実行方法

FrontISTRは、入力ファイルがあるディレクトリにおいて以下のコマンドラインで実行します。

```
$ fistrl
```

可視化はふたつの方法で実行することができます。ひとつは、FrontISTRの後処理として実行する場合で、解析制御データに

```
!WRITE, VISUAL
```

を指定することにより、自動的に実行されます。この場合、可視化制御データを解析制御データに含めて記述することが必要です。

FrontISTRの実行終了後に可視化を行う場合は、まず解析制御データに

```
!WRITE, RESULT
```

を指定し、FrontISTRを実行します。

FrontISTRの実行終了後、入力ファイルと解析結果ファイルがあるディレクトリにおいて以下のコマンドラインでhecmw\_visを実行します。

```
$ hecmw_vis1
```

#### 1.4.1.4 出力ファイルの説明

##### 1.4.1.4.1 (1) 解析結果メッセージファイル（拡張子msg）

本ファイルには、FrontISTRの解析進行過程などのメッセージが output されます。1回の実行でひとつのファイルが作成され、ファイル名は固定のFSTR.msgです。

##### 1.4.1.4.2 (2) 解析結果ログファイル（拡張子log）

本ファイルには、FrontISTRの節点ごとおよび要素ごとの物理量の解析結果が output されます。また、物理量の最大/最小値や固有値解析結果も output されます。動解析の場合、すべてのステップの解析結果が本ファイルに出力されます。1回の実行でひとつのファイルが作成され、ファイル名は固定の0.logです。

##### 1.4.1.4.3 (3) 解析結果ファイル（拡張子なし）

本ファイルは、!WRITE, RESULTオプションを指定した場合に出力されます。

本ファイルには、FrontISTRの節点ごとおよび要素ごとの物理量の解析結果が output されます。ステップごとにファイルが生成され、ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー).0.(ステップ番号)

例：model.res.0.1

##### 1.4.1.4.4 (4) 解析結果ビットマップファイル（拡張子bmp）

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルには、可視化処理を行ったビットマップデータが出力されます。ファイル名は全体制御データで指定したファイルヘッダーを用いて命名されます。命名則の詳細は、hecmw1のドキュメント(0803\_001f\_hecmw\_PC\_cluster\_201\_vis.pdf)を参照ください。

#### 1.4.1.4.5 (5) 解析結果非構造格子型データファイル（拡張子`inp`）

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルを用いて、REVOCAP\_PrePostやMicroAVSなどでポスト処理が可能です。ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー)\_psf.(ステップ番号).inp

例：model\_vis\_psf.0000.inp

#### 1.4.1.4.6 (6) 解析結果ニュートラルファイル（拡張子`neu`）

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルを用いて、Femapでポスト処理が可能です。ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー)\_psf.(ステップ番号).neu

例：model\_vis\_psf.0000.neu

注：この他にFSTR.dbgファイルが出力されますが、デバッグ用ですので通常は参照不要です。

---

## 1.4.2 並列処理による解析

### 1.4.2.1 実行の流れ

FrontISTRを利用した、マルチプロセッサーの並列処理による実行の流れを図3.1.3に示します。

図3.1.3 並列処理による実行の流れ

### 1.4.2.2 準備する入力ファイル

#### 1.4.2.2.1 (1) 全体制御データ（拡張子`dat`）

このファイルではメッシュデータと解析制御データの入力ファイルおよび解析結果出力ファイルを指定します。ファイル名は固定のhecmw\_ctrl.datです。

全体制御データの例を以下に示します。本例ではまず、hecmw\_partが单一領域メッシュデータmodel.mshを読み込み、分散領域メッシュデータmodel\_8.0～nを書き出します。FrontSTRは分散領域メッシュデータmodel\_8.0～nと解析制御データmodel.cntを読み込み、解析結果データmodel.res.0～n.1を書き出します。また、hecmw\_visは分散領域メッシュデータmodel\_8.0～nと解析結果データmodel.res.0～n.1を読み込み、出力指定に対応するmodel\_vis\_psf.0000.(拡張子)を書き出します。詳細はユーザーマニュアル第5章を参照ください。

#### 1.4.2.2.2 (2) 単一領域メッシュデータ（拡張子`msh`）

このファイルでは、解析対象の全体メッシュ構成、材料データ、解析制御データで使用するグループデータなどを定義します。詳細はユーザーマニュアル第6章を参照ください。

#### 1.4.2.2.3 (3) 解析制御データ（拡張子cnt）

このファイルでは、解析種別、変位境界条件、荷重境界条件などを定義します。また、ソルバーの制御データやビジュアライザーの制御データも指定します。3章で解析制御データの例を示します。詳細はユーザーマニュアル第7章を参照ください。

#### 1.4.2.2.4 (4) 領域分割ユーティリティ制御データ（拡張子dat）

このファイルでは、hecmw\_partの制御データを指定します。ファイル名は固定のhecmw\_part\_ctrl.datです。領域分割ユーティリティ制御データの例を以下に示します。本例では、領域分割法PMETISにより、単一領域を8分割します。また、領域分割後のメッシュを表示するためのファイルmodel\_8.inpを出力します。詳細はhecmw1のドキュメント(0803\_001x\_hecmw\_part\_201\_users.pdf)を参照ください。

#### 1.4.2.2.5 (5) 可視化制御データ（拡張子ini）

このファイルでは、hecmw\_visの制御データを指定します。ディフォルトのファイル名はhecmw\_vis.iniです。可視化制御データの例を以下に示します。本例では、MicroAVS用の非構造格子型データ（拡張子inp）を出力します。詳細はユーザーマニュアル7.3.3節および7.4.7節を参照ください。

### 1.4.2.3 実行方法

hecmw\_partは、入力ファイルがあるディレクトリにおいて以下のコマンドラインで実行します。

```
$ hecmw_part1
```

FrontISTRは、入力ファイルがあるディレクトリにおいて以下のコマンドラインで実行します。なお、MPIプロセスの実行方法はそれぞれの環境に合わせて、修正が必要です。

```
$ mpirun -np 8 fistrl
```

可視化はふたつの方法で実行することができます。ひとつは、FrontISTRの後処理として実行する場合で、解析制御データに

```
!WRITE, VISUAL
```

を指定することにより、自動的に実行されます。この場合、可視化制御データを解析制御データに含めて記述することが必要です。

FrontISTRの実行終了後に可視化を行う場合は、まず解析制御データに

```
!WRITE, RESULT
```

を指定し、FrontISTRを実行します。

FrontISTRの実行終了後、入力ファイルと解析結果ファイルがあるディレクトリにおいて以下のコマンドラインでhecmw\_visを実行します。なお、MPIプロセスの実行方法はそれぞれの環境に合わせて、修正が必要です。

```
$ mpirun -np 8 hecmw_vis1
```

### 1.4.2.4 出力ファイルの説明

#### 1.4.2.4.1 (1) 領域分割ユーティリティログファイル（拡張子log）

本ファイルには、hecmw\_partの解析進行過程などのメッセージが出力されます。ファイル名は固定のhecmw\_part.logです。

#### **1.4.2.4.2 (2) 分散領域メッシュファイル（拡張子なし）**

本ファイルには、領域分割されたメッシュ構成、材料データ、解析制御データで使用するグループデータなどが出力されます。ファイルは分散領域ごとに作成され、ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!MESHで指定したファイルヘッダー).(分散領域番号)

例：model\_8.0 ~ model\_8.7

#### **1.4.2.4.3 (3) 領域分割メッシュ表示用ファイル（拡張子inp）**

本ファイルには、領域分割されたメッシュを表示するための非構造格子型データが出力されます。MicroAVSなどで表示できます。

#### **1.4.2.4.4 (4) 解析結果メッセージファイル（拡張子msg）**

本ファイルには、FrontISTRの解析進行過程などのメッセージが出力されます。1回の実行でひとつのファイルが作成され、ファイル名は固定のFSTR.msgです。

#### **1.4.2.4.5 (5) 解析結果ログファイル（拡張子log）**

本ファイルには、FrontISTRの節点ごとおよび要素ごとの物理量の解析結果が出力されます。また、物理量の最大/最小値や固有値解析結果も出力されます。動解析の場合、すべてのステップの解析結果が本ファイルに出力されます。ファイルは分散領域ごとに作成され、ファイル名は固定のn.log（nは分散領域番号）です。

#### **1.4.2.4.6 (6) 解析結果ファイル（拡張子なし）**

本ファイルは、!WRITE, RESULTオプションを指定した場合に出力されます。

本ファイルには、FrontISTRの節点ごとおよび要素ごとの物理量の解析結果が出力されます。分散領域ごとかつステップごとにファイルが生成され、ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー).(分散領域番号).(ステップ番号)

例：model\_8.res.0.1 ~ model\_8.res.7.1

#### **1.4.2.4.7 (7) 解析結果ビットマップファイル（拡張子bmp）**

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルには、可視化処理を行ったビットマップデータが出力されます。ファイル名は全体制御データで指定したファイルヘッダーを用いて命名されます。命名則の詳細は、hecmw1 のドキュメント (0803\_001f\_hecmw\_PC\_cluster\_201\_vis.pdf) を参照ください。

#### **1.4.2.4.8 (8) 解析結果非構造格子型データファイル（拡張子inp）**

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルを用いて、REVOCAP\_PrePostやMicroAVSなどでポスト処理が可能です。ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー)\_psf.(ステップ番号).inp

例：model\_vis\_psf.0000.inp

#### 1.4.2.4.9 (9) 解析結果ニュートラルファイル（拡張子neu）

本ファイルは、可視化制御データで指定した場合に出力されます。

本ファイルを用いて、Femapでポスト処理が可能です。ファイル名は全体制御データで指定したファイルヘッダーを用いて、以下のように命名されます。

命名則：(!RESULTで指定したファイルヘッダー)\_psf.(ステップ番号).neu

例：model\_vis\_psf.0000.neu

注：この他にFSTR.dbg.0～nファイルが出力されますが、デバッグ用ですので通常は参照不要です。

## 1.5 要素ライブラリおよび材料データ

### 1.5.1 要素ライブラリ

FrontISTRは、表 4.1.1に示す要素群を解析に使用することができる。FrontISTRはメッシュデータをHEC-MWを使用して入力するので、以下の要素ライブラリの記述はHEC-MWの説明に準じたものである。要素ライブラリを図 4.1.1に、要素コネクティビティおよび面番号の定義を 図 4.1.2に示す。

表 4.1.1 要素ライブラリ一覧

要素種類	要素番号	説明
線要素	111	2節点リンク要素
	112	3節点リンク要素
平面要素	231	3節点三角形要素
	232	6節点三角形二次要素
	241	4節点四角形要素
	242	8節点四角形二次要素
ソリッド要素	301	2節点ト拉斯要素
	341	4節点四面体要素
	342	10節点四面体二次要素
	351	6節点五面体要素
	352	15節点五面体二次要素
	361	8節点六面体要素
	362	20節点六面体二次要素
インターフェース要素	541	四角形断面一次要素
	542	四角形断面二次要素
梁要素	611	2節点梁要素
	641	2節点梁要素（3自由度節点4点による）
シェル要素	731	3節点三次元一次要素
	732	6節点三次元二次要素
	741	4節点三次元一次要素
	743	9節点三次元二次要素
	761	3節点三次元一次要素（3自由度節点6点による）
	781	4節点三次元一次要素（3自由度節点8点による）

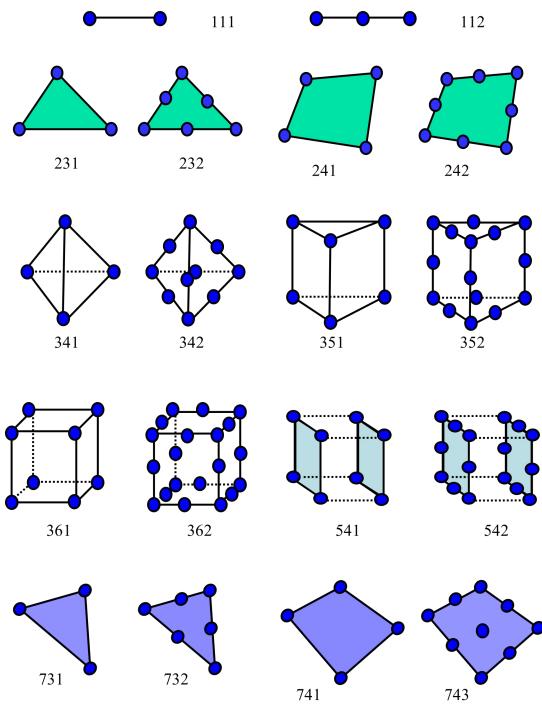
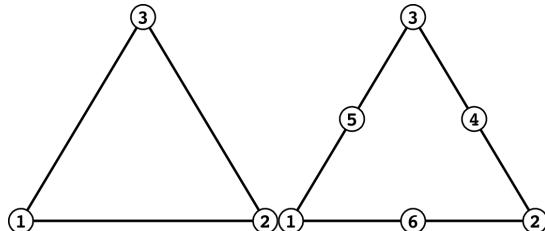


図 4.1.1 要素ライブラリ

### 1.5.1.1 (線要素)



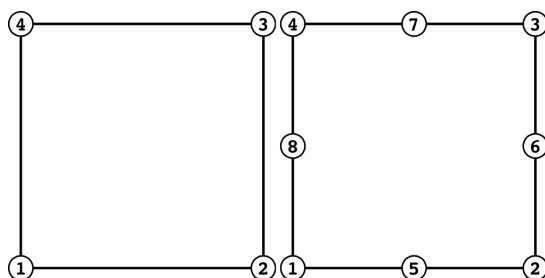
### 1.5.1.2 (三角形平面要素)



面番号 一次 二次

1	1 - 2	1 - 6 - 2
2	2 - 3	2 - 4 - 3
3	3 - 1	3 - 5 - 1

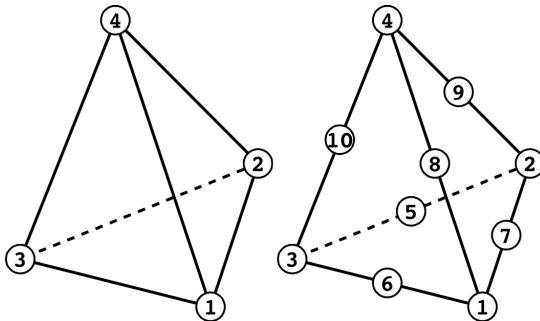
### 1.5.1.3 (四角形平面要素)



面番号 一次 二次

1	1 - 2 1 - 5 - 2
2	2 - 3 2 - 6 - 3
3	3 - 4 3 - 7 - 4
4	4 - 1 4 - 8 - 1

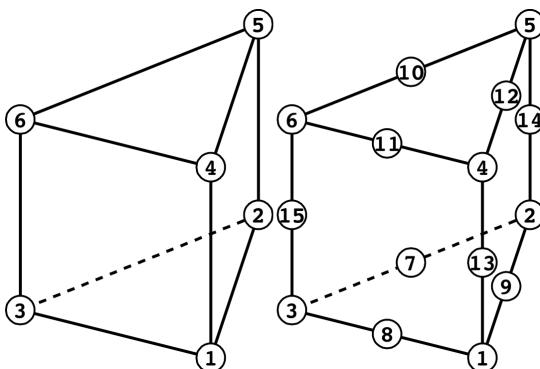
#### 1.5.1.4 (四面体要素)



面番号 一次 二次

1	1 - 2 - 3 1 - 7 - 2 - 5 - 3 - 6
2	1 - 2 - 4 1 - 7 - 2 - 9 - 4 - 8
3	2 - 3 - 4 2 - 5 - 3 - 10 - 4 - 9
4	3 - 1 - 4 3 - 6 - 1 - 10 - 4 - 8

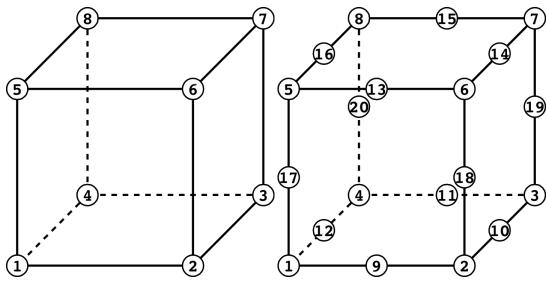
#### 1.5.1.5 (五面体要素)



面番号 一次 二次

1	1 - 2 - 3	1 - 9 - 2 - 7 - 3 - 8
2	4 - 5 - 6	4 - 12 - 5 - 10 - 6 - 11
3	1 - 2 - 5 - 4	1 - 9 - 2 - 14 - 5 - 12 - 4 - 13
4	2 - 3 - 6 - 5	2 - 7 - 3 - 15 - 6 - 10 - 5 - 14
5	3 - 1 - 4 - 6	3 - 8 - 1 - 13 - 4 - 11 - 6 - 15

#### 1.5.1.6 (六面体要素)



面番号	一次	二次
1	1 - 2 - 3 - 4	1 - 9 - 2 - 10 - 3 - 11 - 4 - 12
2	5 - 6 - 7 - 8	5 - 13 - 6 - 14 - 7 - 15 - 8 - 16
3	1 - 2 - 6 - 5	1 - 9 - 2 - 18 - 6 - 13 - 5 - 17
4	2 - 3 - 7 - 6	2 - 10 - 3 - 19 - 7 - 14 - 6 - 18
5	3 - 4 - 8 - 7	3 - 11 - 4 - 20 - 8 - 15 - 7 - 19
6	4 - 1 - 5 - 8	4 - 12 - 1 - 17 - 5 - 16 - 8 - 20

### 1.5.1.7 (梁要素)

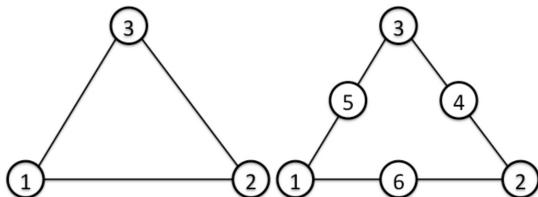


### 1.5.1.8 (3自由度節点による梁要素)



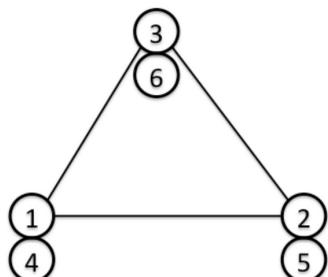
節点1, 2は並進自由度、節点3, 4は回転自由度

### 1.5.1.9 (三角形シェル要素)



Surface No.	Linear	Quadratic
1	1 - 2 - 3 [front]	1 - 6 - 2 - 4 - 3 - 5 [front]
2	3 - 2 - 1 [back]	3 - 4 - 2 - 6 - 1 - 5 [back]

### 1.5.1.10 (3自由度節点による三角形シェル要素)

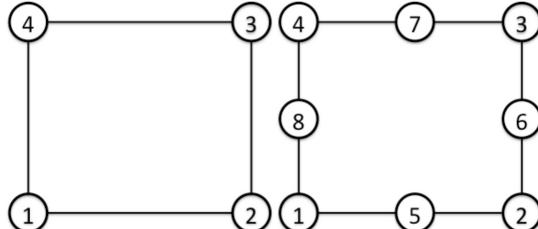


節点1, 2, 3は並進自由度、節点4, 5, 6は回転自由度

**Surface No. Linear**

- |   |                   |
|---|-------------------|
| 1 | 1 - 2 - 3 [front] |
| 2 | 3 - 2 - 1 [back]  |

**1.5.1.11 (四角形シェル要素)**

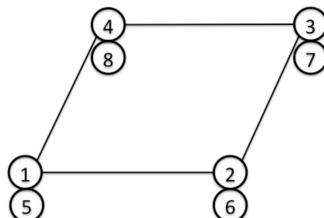


**Surface No. Linear**

- |   |                       |                                       |
|---|-----------------------|---------------------------------------|
| 1 | 1 - 2 - 3 - 4 [front] | 1 - 5 - 2 - 6 - 3 - 7 - 4 - 8 [front] |
| 2 | 4 - 3 - 2 - 1 [back]  | 4 - 7 - 3 - 6 - 2 - 5 - 1 - 8 [back]  |

**Quadratic**

**1.5.1.12 (3自由度節点による四角形シェル要素)**



節点1, 2, 3, 4は並進自由度、節点5, 6, 7, 8は回転自由度

**Surface No. Linear**

- |   |                       |
|---|-----------------------|
| 1 | 1 - 2 - 3 - 4 [front] |
| 2 | 4 - 3 - 2 - 1 [back]  |

図 4.1.2 コネクティビティおよび面番号

**1.5.2 材料データ**

**1.5.2.1 弹性静解析、線形動的解析および固有値解析**

FrontISTRの弾性静解析および固有値解析では、等方性弹性材料を使用し、要素ごとにヤング率、ポアソン比、密度、線膨張係数を定義する必要がある。これらの材料物性値はメッシュデータのヘッダー!SECTIONと!MATERIALにて定義する。以下にその例を示す。

```
#SECTIONの定義
!SECTION, TYPE=SOLID, EGRP=ALL, MATERIAL=M1
```

上記の!SECTIONでは、ソリッドタイプの要素で、グループ名=ALLに所属する要素の、材料データ名をM1とすることを意味する。

次に材料データの定義方法を示す。

```
#材料名M1の材料では3種の物値を定義の意
!MATERIAL, NAME=M1, ITEM=3
```

```

#!ITEM=1ではヤング率とポアソン比を定義（必須）
!ITEM=1, SUBITEM=2
 4000., 0.3

#!ITEM=2で質量密度を定義すること（ITEM=3のときには必須）
!ITEM=2
 8.0102E-10

#!ITEM=3で線膨張係数を定義すること
!ITEM=3
 1.0E-5

```

各ITEMの番号と物性種別が対応しており、ITEM番号さえ正しければ定義する順番は任意である。ただし、ITEM=1内ではヤング率、ポアソン比の順に定義しなければならない。

### 1.5.2.2 热伝導解析

FrontISTRの热伝導解析では、温度依存性を考慮した等方性材料が使用できる。定義する物性値はリンク、平面、ソリッドおよびシェル要素では、密度、比熱および热伝導率を、インターフェース要素ではギャップ热伝達係数とギャップ輻射係数である。これらの物性値の定義方法の例を以下に示す。

#### 1.5.2.2.1 リンク、平面およびソリッド要素の場合

!SECTIONと!MATERIALヘッダーにて定義する。

```

#セクションの定義
!SECTION, TYPE=SOLID, EGRP=ALL, MATERIAL=M1

```

上記の!SECTIONでは、ソリッドタイプの要素で、グループ名=ALLに所属する要素の、材料データ名をM1とすることを意味する。以下、その材料データの定義方法である。

```

#材料名M1の材料では3種の物値を定義の意
!MATERIAL, NAME=M1, ITEM=3

```

```

#!ITEM=1では密度と温度を定義（必須）
!ITEM=1, SUBITEM=1
 7850., 300.
 7790., 500.
 7700., 800.

```

```

#!ITEM=2では比熱と温度を定義（必須）
!ITEM=2
 0.465, 300.
 0.528, 500.
 0.622, 800.

```

```

#!ITEM=3では热伝導率と温度を定義（必須）
!ITEM=3
 43., 300.
 38.6, 500.
 27.7, 800.

```

FrontISTRでは各ITEMの番号により物性の種類を識別しているので、両者の整合がとれていれば定義する順番は任意である。

#### 1.5.2.2.2 インターフェース要素の場合

!SECTIONヘッダーで定義する。（材料データは不要）

#セクションの定義

```
!SECTION, TYPE=INTERFACE, EGRP=GAP  
1.0, 20.15, 8.99835E-9, 8.99835E-9
```

上記の!SECTIONでは、インターフェース要素で、グループ名=GAPに所属する要素のギャップパラメータを定義している。

- 第1パラメータ: ギャップ幅
- 第2パラメータ: ギャップ熱伝達係数
- 第3パラメータ: ギャップ輻射係数1
- 第4パラメータ: ギャップ輻射係数2

#### 1.5.2.2.3 シェル要素の場合

!SECTIONとMATERIALヘッダーで定義する。

#セクションの定義

```
!SECTION, TYPE=SHELL, EGRP=SH, MATERIAL=M2  
10.0, 5
```

上記の!SECTIONでは、シェルタイプの要素で、グループ名=SHに所属する要素の、シェル特性を定義している。

- 第1パラメータ: シェル厚さ
- 第2パラメータ: 厚さ方向積分点数

また、同グループに所属する要素の材料物性を、材料データ名をM2とすることを意味する。

材料物性の定義方法は、等方性材料による単層シェルの場合は、ソリッド要素の場合と全く同様の指定方法が可能である。ソリッド要素の説明を参照のこと。

これに加え、異方性材料や、積層シェルのために、以下に示す定義方法が可能である。

\*\* (例) 等方性材料 単層の場合 \*\*

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1,SUBITEM=4  
0, 200000, 0.3, 2.0
```

- 第1パラメータ: 等方性0
- 第2パラメータ: ヤング率
- 第3パラメータ: ポアソン比
- 第4パラメータ: シェル厚さ

\*\* (例) 等方性材料 複層の場合(2層) \*\*

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1,SUBITEM=7  
0, 200000, 0.3, 2.0, 200000, 0.3, 2.0
```

- 第1パラメータ: 等方性0
- 第2パラメータ: ヤング率(第1層)
- 第3パラメータ: ポアソン比(第1層)
- 第4パラメータ: シェル厚さ(第1層)
- 第5パラメータ: ヤング率(第2層)
- 第6パラメータ: ポアソン比(第2層)
- 第7パラメータ: シェル厚さ(第2層)

積層数分の材料定数を入力することで、自動的に積層シェルを考慮した解析を行う。

シェルの総厚さは、自動的に各層の総和が入力される。積層順は、シェル法線方向、上から順に積層される。

\*\* (例) 異方性材料 単層の場合 \*\*

```
!MATERIAL, NAME=M1, ITEM=1
!ITEM=1, SUBITEM=9
1, 28600., 0.15, 32.3 , 28600., 12434., 12434., 12434., 0.0
```

- 第1パラメータ: 異方性1
- 第2パラメータ: ヤング率E1
- 第3パラメータ: ポアソン比12
- 第4パラメータ: シエル厚さ
- 第5パラメータ: ヤング率E2
- 第6パラメータ: せん断係数G12
- 第7パラメータ: せん断係数G23
- 第8パラメータ: せん断係数G13
- 第9パラメータ: 異方性角度θ

異方性角度はθ[degree]で指定する。

\*\* (例) 異方性材料 複層の場合(2層) \*\*

```
!MATERIAL, NAME=M1, ITEM=1
!ITEM=1, SUBITEM=17
1, 28600., 0.15, 32.3 , 28600., 12434., 12434., 12434., 0.0, 28600., 0.15, 32.3 ,
28600., 12434., 12434., 12434., 0.0
```

- 第1パラメータ: 異方性1
- 第2パラメータ: ヤング率E1(第1層)
- 第3パラメータ: ポアソン比12(第1層)
- 第4パラメータ: シエル厚さ(第1層)
- 第5パラメータ: ヤング率E2(第1層)
- 第6パラメータ: せん断係数G12(第1層)
- 第7パラメータ: せん断係数G23(第1層)
- 第8パラメータ: せん断係数G13(第1層)
- 第9パラメータ: 異方性角度θ(第1層)
- 第10パラメータ: ヤング率E1(第2層)
- 第11パラメータ: ポアソン比12(第2層)
- 第12パラメータ: シエル厚さ(第2層)
- 第13パラメータ: ヤング率E2(第2層)
- 第14パラメータ: せん断係数G12(第2層)
- 第15パラメータ: せん断係数G23(第2層)
- 第16パラメータ: せん断係数G13(第2層)
- 第17パラメータ: 異方性角度θ(第2層)

### 1.5.3 非線形静解析

FrontISTRの非線形静解析では、4.2.1に示した!SECTIONと!MATERIALにて定義する方法の他、解析制御データ中の!ELASTIC、!HYPERELASTIC、!PLASTICなども定義することができる。以下にその例を示す。

\*\* (例) 超弾性材料の定義 \*\*

```
!MATERIAL
#Neo Hooke超弾性材料の定義
!HYPERELASTIC, TYPE=NEOHOKE
#C 10とDを定義（必須）
1000.0, 0.00005
```

\*\* (例) 弹塑性材料の定義 \*\*

```
!MATERIAL
```

```
#等方性弾性材料の定義  
!ELASTIC, TYPE=ISOTROPIC  
  
#ヤング率とポアソン比を定義（必須）  
21000.0, 0.3  
  
#Drucker-Prager塑性材料の定義  
!PLASTIC, TYPE=DRUCKER-PRAGER  
  
#粘着力、摩擦角および硬化係数を定義（必須）  
500.0, 4.0, 10.0
```

## 1.6 全体制御データ

### 1.6.1 全体制御データ概要

全体制御データは、FrontISTRに対する入出力ファイルのファイル名を定義するものである。

全体制御データファイルの特徴は以下のとおりである。

- 自由書式に基づくASCII形式のファイルである。
- “!”で始まるヘッダーとそれに続くデータから構成されている。
- ヘッダーの記述の順番は基本的に自由である。
- データの区切り記号には“,”を使用する。

### 1.6.2 入力規則

全体制御データファイルは、ヘッダー行、データ行、コメント行から構成される。

ヘッダー行には必ず一つのヘッダーが含まれる。

#### ヘッダー

全体制御データファイル内で、データの意味とデータブロックを特定する。行頭が“!”で始まる場合、ヘッダーであるとみなされる。

#### ヘッダー行

ヘッダーとそれに伴うパラメータを記述する。ヘッダー行はヘッダーで始まっていなければならない。パラメータが必要な場合は、“,”を用いてその後に続けなければならぬ。パラメータが値をとる場合は、パラメータの後に“=”が続き、その後に値を記述する。ヘッダー行を複数行にわたって記述することはできない。

#### データ行

ヘッダー行の次の行から開始され、必要なデータを記述する。データ行は複数行にわたる可能性があるが、それは各ヘッダーで定義されるデータ記述の規則により決定される。データ行は必要ない場合もある。

#### 区切り文字

データの区切り文字にはカンマ“,”を用いる。

#### 空白の扱い

空白は無視される。

#### 名前

名前に使用可能な文字は、アンダースコア“\_”、ハイフン“-”、英数字“a-z A-Z 0-9”であるが、最初の一文字は“\_”または英字“a-z A-Z”で始まっていなければならない。大文字小文字の区別はなく、内部的にはすべて大文字として扱われる。また、名前の最大長は63文字である。

#### ファイル名

ファイル名に使用可能な文字は、アンダースコア“\_”、ハイフン“-”、ピリオド“.”、スラッシュ“/”、英数字“a-z A-Z 0-9”である。ファイル名は、特に記述がない限りパスを含んでもよい。相対パス、絶対パスのいずれも指定可能である。また、ファイル名の最大長は1023文字である。

### 浮動小数点データ

指数はあってもなくてもよい。指数の前には、“E”または“e”の記号をつけなければならない。“E”または“e”どちらを使用してもかまわない。“D”または“d”は使用不可。

### !!, # コメント行

行頭が“!!”または“#”で始まる行はコメント行とみなされ、無視される。コメント行はファイル中の任意の位置に挿入でき、その数に制限はない。

## 1.6.3 ヘッダー一覧

全体制御データは以下のヘッダーによって構成されている。

### ヘッダーネーム 内容

!CONTROL	解析制御データ定義
!MESH	メッシュデータ定義
!RESTART	リスタートデータ定義
!RESULT	解析結果データ定義
!SUBDIR	サブディレクトリ分割指示

各ヘッダーには、パラメータとそれぞれのヘッダーに対応したデータの項目がある。以下、上記各ヘッダーについてデータ作成例とともに説明する。

### 1.6.3.1 !CONTROL

解析制御データファイルを指定する。

```
!CONTROL, NAME=<name>
file

#使用例
!CONTROL, NAME=fstrCNT
myctrl.cnt
```

### パラメータ

NAME 識別子（必須）

### パラメータ名 パラメータ値 内容

NAME fstrCNT 解析制御データ

### 変数名 内容

file 解析制御データファイル名（相対パス、絶対パス共に指定可能。相対パスの場合はカレントディレクトリからのパスとなる）

### 1.6.3.2 !MESH

メッシュデータファイルを指定する。

```
!MESH, NAME=<name>, TYPE=<type> [,optional parameter]
fileheader
```

### #使用例

```
!MESH, NAME=fstrMSH, TYPE=HECMW-DIST, REFINE=1
Mesh.in
```

### パラメータ

NAME 識別子（必須）

TYPE メッシュタイプ（必須）

```

...
IO      入出力指定（省略可）
REFINE メッシュ細分化指定（任意）

パラメータ名 パラメータ値 内容
NAME    fstrMSH   Solver入力データ
        part_in    Partitioner入力データ
        part_out   Partitioner出力データ
TYPE    HECMW-DIST HEC-MW分散メッシュデータ
        HECMW-ENTIRE HEC-MW単一領域メッシュデータ
IO      IN        入力用（デフォルト）
        OUT       出力用
REFINE <integer> メッシュ細分化回数

変数名 内容
fileheader メッシュデータファイル名のヘッダー（相対パス、絶対パス共に指定可能。相対パスの場合はカレントディレクトリからのパスとなる）

```

#### 注意

IOパラメータの有無、パラメータ値は動作に何も影響を与えない。

TYPEがHECMW-DISTの場合、データ行に指定するfileheaderはファイル名末尾の「.<rank>」を除いたものである。

### 1.6.3.3 !RESTRAT

リスタートデータファイルを指定する。

```
!RESTART, NAME=<name>, IO=<io>
fileheader
```

```
#使用例
!RESTART, NAME=restart-in, IO=IN
restart.in
```

#### パラメータ 内容

```
NAME    識別子（必須）
IO      入出力指定（必須）
```

#### パラメータ名 パラメータ値 内容

```
NAME    <name>   ユーザー識別子
IO      IN        入力用
        OUT       出力用
        INOUT    入出力兼用
```

#### 変数名 内容

リスタートデータファイル名のヘッダー  
 fileheader(相対パス、絶対パス共に指定可能。相対パスの場合はカレントディレクトリからのパスとなる)

**\*\* 注意 \*\*** この定義によって生成されるファイル名は、“fileheader+.<rank>”となる。

### 1.6.3.4 !RESULT

解析結果データファイルを指定する。

```
!RESULT, NAME=<name> [,optional parameter]
```

```

fileheader

#使用例
!RESULT, NAME=fstrRES, IO=OUT, TYPE=BINARY
result.out

!RESULT, NAME=<name> [,optional parameter]

パラメータ 内容
NAME      識別子（必須）
IO        入出力指定（省略可）
TYPE      出力形式（省略可）

パラメータ名 パラメータ値           内容
NAME      fstrRES      Solver出力データ、Visualizer入力データ
          fstrTEMP     温度入力データ（熱伝導解析結果）
          vis_out       Visualizer出力データ
IO        IN           入力用
          OUT          出力用
TYPE      TEXT         テキスト形式（デフォルト）
          BINARY        バイナリー形式

変数名   内容
fileheader 解析結果データファイル名のヘッダー（相対パス、絶対パス共に指定可能。相対
            パスの場合はカレントディレクトリからのパスとなる）

** 注意 ** この定義によって生成されるファイル名は、“fileheader+.<rank>”となる。

```

### 1.6.3.5 !SUBDIR

ファイル格納のサブディレクトリ分割を指定する。

```

!SUBDIR, ON [,optional parameter]

#使用例
!SUBDIR, ON, LIMIT=8000

!SUBDIR, ON [,optional parameter]

```

**パラメータ 内容**

ON	有効（必須）
LIMIT	ファイル数（省略可）

**パラメータ名 パラメータ値 内容**

ON	なし
LIMIT	<integer> 1ディレクトリあたり最大ファイル数(デフォルト: 5000)

\*\* 注意 \*\* この定義によって、複数ファイルで構成される入出力データは自動的にサブディレクトリ配下となる。並列数がLIMITを超える場合、さらにTRUNK0、TRUNK1、……のサブディレクトリ配下となる。

## 1.7 単一領域メッシュデータ

### 1.7.1 単一メッシュデータ概要

FrontISTRのメッシュデータ取得には、単一領域メッシュデータファイルを入力する方法と並列処理するために分散メッシュデータファイルを入力する方法の二通りがある。

ここでは、単一領域メッシュデータについて記述する。

単一領域メッシュデータの特徴は以下のとおりである。

- 自由書式に基づくASCII形式のファイルである。
- “!”で始まるヘッダーとそれに続くデータから構成されている。
- ヘッダーの記述の順番は基本的に自由である。
- データの区切り記号には“,”を使用する。

### 1.7.2 入力規則

単一領域メッシュデータファイルは、ヘッダー行、データ行、コメント行から構成される。

ヘッダー行には必ず1つのヘッダーが含まれる。

#### ヘッダー

単一領域メッシュデータファイル内で、データの意味とデータブロックを特定する。  
行頭が“!”で始まる場合、ヘッダーであるとみなされる。

#### ヘッダー行

ヘッダーとそれに伴うパラメータの内容を記述する。  
ヘッダー行はヘッダーで始まっているなければならない。パラメータが必要な場合は、“,”  
を用いてその後に続けなければならない。  
パラメータが値をとる場合は、パラメータの後に“=”が続き、その後に値を記述する。  
ヘッダー行を複数行にわたって記述することはできない。

#### データ行

ヘッダー行の次の行から開始され、必要なデータを記述する。  
データ行は複数行にわたる可能性があるが、それは各ヘッダーで定義されるデータ記述  
の規則により決定される。  
データ行は必要ない場合もある。

#### 区切り文字

データの区切り文字にはカンマ“,”を用いる。

#### 空白の扱い

空白は無視される。

#### 名前

名前に使用可能な文字は、アンダースコア“\_”、ハイフン“-”、英数字“a-z A-Z 0-9”で  
あるが、最初の一文字は“\_”または英字“a-z A-Z”で始まっているなければならない。大  
文字小文字の区別はなく、内部的にはすべて大文字として扱われる。  
また、名前の最大長は63文字である。

#### ファイル名

ファイル名に使用可能な文字は、アンダースコア“\_”、ハイフン“-”、ピリオド“.”、ス  
ラッシュ“/”、英数字“a-z A-Z 0-9”である。  
ファイル名は、特に記述がない限りパスを含んでもよい。相対パス、絶対パスのいずれ  
も指定可能である。  
また、ファイル名の最大長は1023文字である。

#### 浮動小数点データ

指数はあってもなくてもよい。指数の前には、“E”または“e”の記号をつけなければな  
らない。

“E”または“e”どちらを使用してもかまわない。“D”または“d”は使用不可。

#### !!, # コメント行

行頭が“!!”または“#”で始まる行はコメント行とみなされ、無視される。  
コメント行はファイル中の任意の位置に挿入でき、その数に制限はない。

### 1.7.3 単一領域メッシュデータのヘッダー一覧

単一領域メッシュデータは以下のヘッダーにより構成されている。

#### ヘッダーネーム

!CONTROL 解析制御データ定義

!MESH	メッシュデータ定義
!RESTART	リスタートデータ定義
!RESULT	解析結果データ定義
!SUBDIR	サブディレクトリ分割指示
 ヘッダー名 内容	
!AMPLITUDE	非定常荷重
!EGROUP	要素グループ
!ELEMENT	要素情報
!EQUATION	拘束点情報
!HEADER	メッシュデータのタイトル
!MATERIAL	材料情報
!NGROUP	節点グループ
!NODE	節点情報
!SECTION	セクション情報
!SGROUP	面グループ
!ZERO	絶対零度
!CONTACT PAIR	接触面ペア
!END	読み込み終了

各ヘッダーには、パラメータとそれぞれのヘッダーに対応したデータの項目がある。

以下、上記各ヘッダーについてデータ作成例とともに簡単に説明する。データ作成例の右端に示している番号は上記表の説明番号である。

### 1.7.3.1 メッシュデータ例

!HEADER	M1-5
TEST MODEL CTLR10	
!NODE	M1-9
1, 0.00000E+00, 0.00000E+00, 0.00000E+00	
2, 0.50000E+01, 0.00000E+00, 0.00000E+00	
3, 0.10000E+02, 0.00000E+00, 0.00000E+00	
... ... ...	
!ELEMENT, TYPE=351	M1-3
1, 1, 2, 4, 34, 35, 37	
2, 2, 5, 4, 35, 38, 37	
3, 2, 3, 5, 35, 36, 38	
!SECTION, TYPE=SOLID, EGRP=ALL, MATERIAL=M1	M1-9
1.0	
!MATERIAL, NAME=M1, ITEM=2	M1-6
!ITEM=1, SUBITEM=2	
2.1E5, 0.3	
!ITEM=2, SUBITEM=1	
7.8e-6	
!NGROUP, NGRP=FIX, GENERATE	M1-7
2, 2, 1	
3, 3, 1	
1, 1, 1	
69, 69, 1	
67, 67, 1	
!NGROUP, NGRP=CL1	M1-8

### 1.7.3.2 !AMPLITUDE (M1-1)

ステップ内での荷重条件を与える変数の時間変化の指定

```
!AMPLITUDE, NAME=<name> [, optional parameter]
VAL1, T1, VAL2, T2, VAL3, T3
# (一行に四項まで)
```

#### パラメータ

NAME	名前 (必須)
DEFINITION	タイプ (省略可)
TIME	時間の種類 (省略可)
VALUE	値の種類 (省略可)
INPUT	外部ファイル名 (省略可)

#### パラメータ名 パラメータ値 内容

NAME	<name>	AMPLITUDE名
DEFINITION	TABULAR	デフォルト (現版ではデフォルトのみ)
TIME	STEP TIME	デフォルト (現版ではデフォルトのみ)
VALUE	RELATIVE	相対値 (デフォルト)
	ABSOLUTE	絶対値
INPUT	<filename> 外部ファイル名 (省略可)、2行目以降との併用も可能	

#### 変数名 属性 内容

VAL1	R	時間T1における値
T1	R	時間T1
VAL2	R	時間T2における値
T2	R	時間T2
VAL3	R	時T3における値
T3	R	時間T3

### 1.7.3.3 !EGROUP (M1-2)

要素グループの定義

1行目 {#行目-1 .affff5}

```
!EGROUP, EGRP=<egrp> [, optional parameter]
```

#### パラメータ

EGRP	要素グループ名 (必須)
GENERATE	要素グループに属する節点の自動生成 (省略可)
INPUT	外部ファイル名 (省略可)

#### パラメータ名 パラメータ値 内容

EGRP	<egrp>	要素グループ名
GENERATE	なし	要素グループに属する節点の自動生成
INPUT	<filename> 外部ファイル名 (省略可)、2行目以降との併用も可能	

2行目以降 (GENERATEを使用しない場合) {#行目以降generateを使用しない場合 .afff5}

(2行目) elem1, elem2, elem3 ...  
(以下同様)

#### 変数名 属性 内容

elemX I 要素グループに属する要素番号

2行目以降 (GENERATEを使用する場合) {#行目以降generateを使用する場合 .afff5}

(2行目) elem1, elem2, elem3  
(以下同様)

#### 変数名 属性 内容

elem1 I 要素グループ内の最初の要素番号

elem2 I 要素グループ内の最後の要素番号

elem3 I 要素番号増分 (省略可能、省略時はelem3=1となる)

注意 {#注意 .afff5}

- 1行に任意の数の要素を入れることができる。また次のオプションが始まるまで、任意の数の行を挿入することができます。
- 指定する要素は「!EGROUP」より前に定義されている必要がある。
- 「!ELEMENT」オプションで定義されていない要素は除外され、警告メッセージが表示される。
- 指定された要素が既に同じグループ内に存在する場合は無視され、警告メッセージが表示される。
- すべての要素は、「ALL」という名前の要素グループに属している（自動的に生成される）。
- ひとつのグループを複数回にわけて定義できる。

使用例 {#使用例 .afff5}

```
!EGROUP, EGRP=EA01
1, 2, 3, 4, 5, 6
101, 102
205
```

```
!EGROUP, EGRP=EA02
101, 102
```

```
!EGROUP, EGRP=EA01
501, 505
```

グループ「EA01」に「501, 505」が追加される。

```
!EGROUP, EGRP=EA04, GENERATE
301, 309, 2
```

グループ「EA04」に  
「301,303,305,307,309,311,312,313」が追加される。

```
311, 313
```

### 1.7.3.4 !ELEMENT (M1-3)

#### 要素の定義

1行目 {#行目-2 .afff5}

```
!ELEMENT, TYPE=<type> [, optional parameter]
```

#### パラメータ

TYPE 要素タイプ (必須)

**EGRP** 要素グループ名（省略可）  
**MATITEM** 材料物性を要素ごとに定義する場合の物性項目数  
 　　（セクションごとに物性を定義する場合は使用しない）  
**INPUT** 外部ファイル名（省略可）

**パラメータ名** **パラメータ値** **内容**

<b>TYPE</b>	111	ロッド、リンク要素（一次）
	231	三角形要素（一次）
	232	三角形要素（二次）
	241	四角形要素（一次）
	242	四角形要素（二次）
	301	ト拉斯要素（一次）
	341	四面体要素（一次）
	342	四面体要素（二次）
	351	三角柱要素（一次）
	352	三角柱要素（二次）
	361	六面体要素（一次）
	362	六面体要素（二次）
	541	インターフェース要素（四角形断面、一次）
	611	梁要素(一次)
	641	梁要素(一次, 3自由度節点4点による)
	731	三角形シェル要素（一次）
	741	四角形シェル要素（一次）
	743	四角形シェル要素（二次）
	761	三角形シェル要素（一次, 3自由度節点6点による）
	781	四角形シェル要素（一次, 3自由度節点8点による）
<b>EGRP</b>	<egrp>	要素グループ名（省略可）
<b>INPUT</b>	<filename>	外部ファイル名（省略可）、2行目以降との併用も可能

2行目以降 {#行目以降-1 .affff5}

(2行目) ELEM\_ID, nod1, nod2, nod3, ..., MAT1, MAT2, ...

(以下同様)

**変数名** **属性** **内容**

<b>ELEM_ID</b>	I	要素番号
<b>nodX</b>	I	コネクティビティ
<b>MATy</b>	R	要素ごとの物性値

注意 {#注意-1 .affff5}

- 要素タイプ、コネクティビティの詳細は、「4章 要素ライブラリ」を参照のこと。
- コネクティビティで指定する節点は「!ELEMENT」より前に定義されている必要がある。
- 要素番号は連続している必要はない。
- 「!ELEMENT」オプションは何回でも定義できる。
- 要素番号は自然数でなければならない。省略は不可。
- 同じ要素番号を重複して使用する場合、最後に入力した値が使用される。この場合、警告メッセージが出力される。
- 定義されていない節点をコネクティビティに使用することはできない。
- ひとつの要素の定義を複数行にわたって記述してもよい。

使用例 {#使用例-1 .affff5}

```
!ELEMENT, TYPE=231
1, 1, 2, 3
2, 4, 8, 5
4, 6, 7, 8
!ELEMENT, TYPE=361, EGRP=A
101, 101, 102, 122, 121, 201, 202, 222, 221
102, 102, 103, 123, 122, 202, 203, 223, 222
103, 103, 104, 124, 123, 203, 204, 224, 223
```

### 1.7.3.5 !EQUATION (1-4)

拘束節点グループの定義

1行目 {#行目-3 .affff5}

```
!EQUATION [, optional parameter]
```

パラメータ

INPUT 外部ファイル名（省略可）

パラメータ名 パラメータ値 内容

INPUT <filename> 外部ファイル名（省略可）、2行目以降との併用も可能

2行目以降 {#行目以降-2 .affff5}

(2行目) NEQ, CONST

(3行目以降) nod1, DOF1, A1, nod2, DOF2, A2 ... (一行に七項まで)  
以下繰り返し

変数名 属性 内容

NEQ I 方程式の項数

CONST R 方程式の定数項（右辺値）

nod1 I/C 第1節点または節点グループ

DOF1 I 第1節点または節点グループの拘束自由度

A1 R 第1節点または節点グループの係数

nod2 I/C 第2節点または節点グループ

DOF2 I 第2節点または節点グループの拘束自由度

A2 R 第2節点または節点グループの係数

注意 {#注意-2 .affff5}

- 「!NODE」で定義されていない節点、節点グループが指定された場合は無視され、警告メッセージが表示される。
- 「nod1=nod2」の場合は無視され、警告メッセージが表示される。
- 節点グループを指定した場合、節点数の整合性が取れない場合はエラーとなる。
- 自由度番号は解析のタイプ、要素によって異なる。整合がとれない自由度については無視され、警告メッセージが表示される。

使用例 {#使用例-2 .affff5}

```
!EQUATION
3
101, 1, 1.0, 102, 1, -1.0, 103, 1, -1.0
2
NG1, 2, 1.0, NG5, 2, -1.0
```

### 1.7.3.6 !HEADER (M1-5)

メッシュデータのタイトル

1行目 {#行目-4 .afff5}

!HEADER

パラメータ

なし

2行目以降 {#行目以降-3 .afff5}

(2行目) TITLE

変数名 属性 内容

TITLE C ヘッダータイトル

使用例 {#使用例-3 .afff5}

!HEADER

Mesh for CFD Analysis

注意 {#注意-3 .afff5}

- 省略可能。
- ヘッダーは複数行にわたってもよいが、ヘッダーとして認識されるのは最初の行の127カラム目までである。
- 「!HEADER」を複数回定義すると、内容が更新され、警告メッセージが表示される。

### 1.7.3.7 !MATERIAL (M1-6)

材料物性の定義

物性が温度依存している場合は対応する温度ごとにテーブル入力が可能である。

1行目 {#行目-6 .afff5}

!MATERIAL, NAME=<name> [, optional parameter]

パラメータ

NAME 材料名（必須）

ITEM 物性項目数（省略可、省略した場合は「1」となる）

INPUT 外部ファイル名（省略可）

パラメータ名 パラメータ値 内容

NAME <name> 材料名

ITEM <ITEMnum> ユーザー定義による物性項目数

INPUT <filename> 外部ファイル名（省略可）、2行目以降との併用も可能

2行目以降 {#行目以降-5 .afff5}

(2行目) !ITEM=1, SUBITEM=<k>

(3行目) VAL1-1-1, VAL1-1-2, ... VAL1-1-k, TEMP1-1

(4行目) VAL1-2-1, VAL1-2-2, ... VAL1-2-k, TEMP1-2

..

(L+2行目) VAL1-L-1, VAL1-L-2, ... VAL1-L-k, TEMP1-L

以下「!ITEM=<ITEMnum>」まで繰り返し定義する

**サブパラメータ(「!ITEM」に対するもの)**

SUBITEM 各物性項目で定義されるサブ物性項目数  
(省略可、省略した場合は「1」となる。)

サブパラメータ名	パラメータ値	内容
SUBITEM	<subITEMnum>	ユーザー定義によるサブ物性項目数

【m番目の物性が温度依存している場合】 {#m番目の物性が温度依存している場合 .afff5}

温度依存のテーブルの項目数がNの場合、以下のように入力する：

```
!ITEM=m, SUBITEM=k
VALm1-1, ..., VALm1-k, TEMPm1
VALm2-1, ..., VALm2-k, TEMPm2
...
VALmN-1, ..., VALmN-k, TEMPm-N
```

**変数名 属性 内容**

VALmn-k R 物性値（温度依存）  
TEMPmn R 対応する温度

TEMPm1 < TEMPm2 < ... < TEMPmN でなければならない。

温度がTEMPm1以下のはVALm1、TEMPmN以上のはVALmNが使用される。

【m番目の物性が温度依存していない場合】 {#m番目の物性が温度依存していない場合 .afff5}

```
!ITEM=m, SUBITEM=k
VALm1-1, ..., VALm1-k
VALm2-1, ..., VALm2-k
...
VALmN-1, ..., VALmN-k
```

**変数名 属性 内容**

VALmn-k R 物性値（温度依存なし）

注意 {#注意-5 .afff5}

- 材料名が重複した場合はエラーとなる。
- 「!SECTION」オプションで参照されているMATERIALが定義されていない場合はエラーとなる。
- 「!ELEMENT」オプションで、パラメータ「MATITEM」を使用して要素ごとに物性を入力した場合の値が優先して使用される。この場合、「!MATERIAL」オプションを使用して入力した物性値は使用されない。
- 「!ITEM=m」サブオプションの数と、パラメータ「ITEM」の数が整合しない場合、定義されていないサブオプションがある場合はエラーとなる。
- 「!ITEM=m」サブオプションは、mの小さい順番に並んでいなくてもよい。
- 「!SUBITEM=k」サブオプション、温度依存性を使用する場合、省略した値は「0.0」となる。
- 温度依存性を使用する場合、温度の低い順に定義しなければならない。
- 温度依存性を使用する場合、同じ温度を2回以上使用した場合はエラーとなる。

使用例 {#使用例-5 .afff5}

```
!MATERIAL, NAME= STEEL, ITEM= 2
!ITEM=1 温度依存性なし
35.0
```

```

!ITEM=2
40.0, 0.0
45.0, 100.0
50.0, 200.0

!MATERIAL, NAME= CUPPER      項目数=1 (デフォルト値)
!ITEM=1 溫度依存性なし
80.0

```

誤った使用例 {#誤った使用例 .afff5}

例1【パラメータ「ITEM」と「!ITEM=m」サブオプションの数が整合していない-1】

```

!MATERIAL, NAME= STEEL, ITEM= 2
!ITEM=3
20.0
!ITEM=1
35.0
!ITEM= 2
40.0

```

例2【パラメータ「ITEM」と「!ITEM=m」サブオプションの数が整合していない-2】

```

!MATERIAL, NAME= STEEL, ITEM= 3
!ITEM=3
20.0
!ITEM= 2
40.0
!MATERIAL, NAME= CUPPER
...

```

弾性静解析および固有値解析 {#弾性静解析および固有値解析 .afff5}

```

!MATERIAL, NAME=<name>, ITEM=<ITEMnum>
!ITEM=1, SUBITEM=2
<Young_modulus>, <Poisson_ratio>
!ITEM=2
<Density>
!ITEM=3
<Expansion_coeff>

```

パラメータ 名	パラメータ値	内容
NAME	<name>	材料名、!SECTIONのMATELIALと対応 ユーザー定義による物性項目数（1以上） <Young_modulus>---ヤング率（必須） <Density>-----質量密度（ITEMnum=2のとき必須） <Expansion_coeff>-線膨張係数（ITEMnum=3のとき）
ITEM	<ITEMnum>	<Poisson_ratio>---ポアソン比（必須）

(例)

```

!MATERIAL, NAME=M1,
ITEM=3           --- 材料名M1の材料では3種の物値を定義の意
!ITEM=1, SUBITEM=2 --- !ITEM=1ではヤング率とポアソン比を定義（必須）
4000.0, 0.3
!ITEM=2           --- !ITEM=2で質量密度を定義すること（ITEM=2のときには必須）
8.0102E-0
!ITEM=3           --- !ITEM=3で線膨張係数を定義すること
1.0E-5

```

熱伝導解析 {#熱伝導解析-2 .afff5}

リンク、平面、ソリッド、3次元板要素の場合

```
!MATERIAL, NAME=<name>, ITEM=3
!ITEM=1, SUBITEM=2
<Density>, <Temperature>

!ITEM=2, SUBITEM=2
<Specific_heat>, <Temperature>

!ITEM=3, SUBITEM=2
<Conductivity>, <Temperature>
```

パラメータ名	パラメータ値	内容
NAME	<name>	材料名、!SECTIONの MATELIALと対応 ユーザー定義による物性項目数 (常に3)
ITEM	<ITEMnum>	<Density>——密度 <Specific_heat>—比熱 <Conductivity>—熱伝導率 <Temperature>—温度

(例)

```
!MATERIAL, NAME=M1,
ITEM=3           --- 材料名M1の材料では3種の物値を定義の意

!ITEM=1, SUBITEM=1      --- !ITEM=1では密度と温度を定義（必須）
7850., 300.
7790., 500.
7700., 800.

!ITEM=2, SUBITEM=1      --- !ITEM=2では比熱と温度を定義（必須）
0.465, 300.
0.528, 500.
0.622, 800.

!ITEM=3           --- !ITEM=3では熱伝導率と温度を定義（必須）
43., 300.
38.6, 500.
27.7, 800.
```

インターフェース要素の場合

!SECTIONヘッダーで定義する。(材料データは不要)

(例)

```
!SECTION, TYPE=INTERFACE, EGRP=GAP   --- セクションの定義
1.0, 20.15, 8.99835E-9, 8.99835E-9
```

上記の!SECTIONでは、インターフェース要素で、グループ名=GAPに所属する要素のギャップパラメータを定義している。

- 第1パラメータ: ギャップ幅
- 第2パラメータ: ギャップ熱伝達係数
- 第3パラメータ: ギャップ輻射係数1
- 第4パラメータ: ギャップ輻射係数2

参考 {#参考 .affff5}

```
program TEST
```

```

use hecmw
implicit REAL*8 (A-H,O-Z)
type (hecmwT_local_mesh) :: hecMESH

!C
!C !MATERIAL, NAME=SUS304, ITEM=3
!C !ITEM=1, SUBITEM= 3
!C     100.0, 200.0, 300.0, 0.00
!C     101.0, 210.0, 301.0, 1.00
!C     102.0, 220.0, 302.0, 2.00
!C     103.0, 230.0, 303.0, 3.00
!C !ITEM=3, SUBITEM= 2
!C     1000.0, , 0.00
!C     1001.0, 1., 1.00
!C     1002.0, 2., 2.00
!C     1003.0, 3., 3.00
!C !ITEM=2
!C     5000.0
!C
!C !MATERIAL, NAME=FEC, ITEM=2
!C !ITEM=1, SUBITEM= 3
!C     2100.0, 2200.0, 2300.0, 0.00
!C     2101.0, 2210.0, 2301.0, 1.00
!C     2102.0, 2220.0, 2302.0, 2.00
!C     2103.0, 2230.0, 2303.0, 3.00
!C     3103.0, 3230.0, 2304.0, 4.00
!C !ITEM=2
!C     6000.0, 10.0
!C     6500.0, 30.0
!C

hecMESH%material%n_mat = 2

nn= hecMESH%material%n_mat
allocate (hecMESH%material%mat_name(nn))

hecMESH%material%mat_name(1)= 'SUS304'
hecMESH%material%mat_name(2)= 'FEC'

nn= hecMESH%material%n_mat
allocate (hecMESH%material%mat_ITEM_index(0:nn))
hecMESH%material%mat_ITEM_index(0)= 0
hecMESH%material%mat_ITEM_index(1)= 3
hecMESH%material%mat_ITEM_index(2)= hecMESH%material%mat_ITEM_index(1) + 2

hecMESH%material%n_mat_ITEM=
hecMESH%material%mat_ITEM_index(hecMESH%material%n_mat)

nn= hecMESH%material%n_mat_ITEM
allocate (hecMESH%material%mat_subITEM_index(0:nn))

hecMESH%material%mat_subITEM_index(0)= 0
hecMESH%material%mat_subITEM_index(1)= 3
hecMESH%material%mat_subITEM_index(2)= hecMESH%material%mat_subITEM_index(1) + 1
hecMESH%material%mat_subITEM_index(3)= hecMESH%material%mat_subITEM_index(2) + 2
hecMESH%material%mat_subITEM_index(4)= hecMESH%material%mat_subITEM_index(3) + 3
hecMESH%material%mat_subITEM_index(5)= hecMESH%material%mat_subITEM_index(4) + 1

hecMESH%material%n_mat_subITEM=
&      hecMESH%material%mat_subITEM_index(hecMESH%material%n_mat_ITEM)

nn= hecMESH%material%n_mat_subITEM
allocate (hecMESH%material%mat_TABLE_index(0:nn))
hecMESH%material%mat_TABLE_index( 0)= 0
hecMESH%material%mat_TABLE_index( 1)= 4

```

```

hecMESH%material%mat_TABLE_index( 2)= hecMESH%material%mat_TABLE_index( 1) + 4
hecMESH%material%mat_TABLE_index( 3)= hecMESH%material%mat_TABLE_index( 2) + 4
hecMESH%material%mat_TABLE_index( 4)= hecMESH%material%mat_TABLE_index( 3) + 1
hecMESH%material%mat_TABLE_index( 5)= hecMESH%material%mat_TABLE_index( 4) + 4
hecMESH%material%mat_TABLE_index( 6)= hecMESH%material%mat_TABLE_index( 5) + 4
hecMESH%material%mat_TABLE_index( 7)= hecMESH%material%mat_TABLE_index( 6) + 5
hecMESH%material%mat_TABLE_index( 8)= hecMESH%material%mat_TABLE_index( 7) + 5
hecMESH%material%mat_TABLE_index( 9)= hecMESH%material%mat_TABLE_index( 8) + 5
hecMESH%material%mat_TABLE_index(10)= hecMESH%material%mat_TABLE_index( 9) + 2

hecMESH%material%n_mat_TABLE=
&           hecMESH%material%mat_TABLE_index(hecMESH%material%n_mat_subITEM)

nn= hecMESH%material%n_mat_TABLE
allocate (hecMESH%material%mat_VAL (nn))
allocate (hecMESH%material%mat_TEMP(nn))

hecMESH%material%mat_VAL = 0.d0
hecMESH%material%mat_TEMP= 0.d0

hecMESH%material%mat_VAL ( 1)= 100.0d0
hecMESH%material%mat_TEMP( 1)= 0.0d0
hecMESH%material%mat_VAL ( 2)= 101.0d0
hecMESH%material%mat_TEMP( 2)= 1.0d0
hecMESH%material%mat_VAL ( 3)= 102.0d0
hecMESH%material%mat_TEMP( 3)= 2.0d0
hecMESH%material%mat_VAL ( 4)= 103.0d0
hecMESH%material%mat_TEMP( 4)= 3.0d0

hecMESH%material%mat_VAL ( 5)= 200.0d0
hecMESH%material%mat_TEMP( 5)= 0.0d0

hecMESH%material%mat_VAL (13)= 5000.0d0

hecMESH%material%mat_VAL (14)= 1000.0d0
hecMESH%material%mat_TEMP (14)= 0.0d0
hecMESH%material%mat_VAL (15)= 1001.0d0
hecMESH%material%mat_TEMP (15)= 1.0d0
hecMESH%material%mat_VAL (16)= 1002.0d0
hecMESH%material%mat_TEMP (16)= 2.0d0
hecMESH%material%mat_VAL (17)= 1003.0d0
hecMESH%material%mat_TEMP (17)= 3.0d0

hecMESH%material%mat_VAL (18)= 0.0d0
hecMESH%material%mat_TEMP (18)= 0.0d0
hecMESH%material%mat_VAL (19)= 1.0d0
hecMESH%material%mat_TEMP (19)= 1.0d0
hecMESH%material%mat_VAL (20)= 2.0d0
hecMESH%material%mat_TEMP (20)= 2.0d0
hecMESH%material%mat_VAL (21)= 3.0d0
hecMESH%material%mat_TEMP (21)= 3.0d0

hecMESH%material%mat_VAL (22)= 2100.0d0
hecMESH%material%mat_TEMP (22)= 0.0d0
hecMESH%material%mat_VAL (23)= 2101.0d0
hecMESH%material%mat_TEMP (23)= 1.0d0
hecMESH%material%mat_VAL (24)= 2102.0d0
hecMESH%material%mat_TEMP (24)= 2.0d0
hecMESH%material%mat_VAL (25)= 2103.0d0
hecMESH%material%mat_TEMP (25)= 3.0d0
hecMESH%material%mat_VAL (26)= 3103.0d0
hecMESH%material%mat_TEMP (26)= 4.0d0

write (*,'(a,i10)') '%n_mat_ITEM ', hecMESH%material%n_mat_ITEM
write (*,'(a,i10)') '%n_mat_subITEM', hecMESH%material%n_mat_subITEM

```

```

    write (*,'(a,i10)') '%n_mat_TABLE ', hecMESH%material%n_mat_TABLE
end program TEST

```

### 1.7.3.8 !NGROUP (M1-7)

節点グループの定義

1行目 {#行目-7 .afff5}

**!NGROUP, NGRP=<ngrp> [, optional parameter]**

#### パラメータ

パラメータ名	パラメータ値	内容
NGRP	<ngrp>	節点グループ名（必須）
GENERATE	なし	節点グループに属する節点の自動生成（省略可）
INPUT	<filename>	外部ファイル名（省略可）

2行目以降 (GENERATEを使用しない場合) {#行目以降generateを使用しない場合-1 .afff5}

(2行目) nod1, nod2, nod3  
(以下同様)

#### 変数名 属性 内容

nodX I 節点グループに属する節点番号

2行目以降 (GENERATEを使用する場合) {#行目以降generateを使用する場合-1 .afff5}

(2行目) nod1, nod2, nod3  
(以下同様)

#### 変数名 属性 内容

nod1 I 節点グループ内の最初の節点番号  
nod2 I 節点グループ内の最後の節点番号  
nod3 I 節点番号増分（省略可能、省略時はnod3=1となる）

注意 {#注意-6 .afff5}

- 1行に任意の数の節点を入れることができる。また次のオプションが始まるまで、任意の数の行を挿入することができる。
- 指定する節点は「!NGROUP」より前に定義されている必要がある。
- 「!NODE」オプションで定義されていない節点は除外され、警告メッセージが表示される。
- 指定された節点が既に同じグループ内に存在する場合は無視され、警告メッセージが表示される。
- 全ての節点は、「ALL」という名前の節点グループに属している（自動的に生成される）。
- ひとつのグループを複数回にわけて定義できる。

使用例 {#使用例-6 .afff5}

```

!NGROUP, NGRP= NA01
1, 2, 3, 4, 5, 6
101, 102

```

```

!NGROUP, NGRP= NA02
101, 102
!NGROUP, NGRP= NA01           --- グループ「NA01」に「501, 505」が追加される。
501, 505
!NGROUP, NGRP= NA02           --- グループ「NA02」に「501, 505」が追加される。
501, 505
!NGROUP, NGRP= NA04,GENERATE --- グループ「NA04」に
301, 309, 2                   --- 「301,303,305,307,309,311,312,313」が追加される。
311, 313

```

### 1.7.3.9 !NODE (M1-8)

節点座標の定義

1行目 {#行目-8 .afff5}

**!NODE [, optional parameter]**

#### パラメータ

SYSTEM	座標系 (省略可)
NGRP	節点グループ名 (省略可)
INPUT	外部ファイル名 (省略可)

パラメータ名	パラメータ値	内容
SYSTEM	R	デカルト座標系 (デフォルト値)
	C	円筒座標系
NGRP	<ngrp>	節点グループ名 (省略可)
INPUT	<filename>	外部ファイル名 (省略可)、2行目以降との併用も可能

2行目以降 {#行目以降-6 .afff5}

(2行目) NODE\_ID, Xcoord, Ycoord, Zcoord  
(以下同様)

変数名	属性	内容
NODE_ID	I	節点番号
Xcoord	R	X座標
Ycoord	R	Y座標
Zcoord	R	Z座標

注意 {#注意-7 .afff5}

- 区切り記号を含めて節点座標を省略した場合、値は「0.0」となる。
- 既に定義されてる節点を再定義した場合、内容が更新され、警告メッセージが表示される。
- 「!ELEMENT」で参照されない節点は除外される。
- 「!ELEMENT」で定義される節点は「!ELEMENT」より前に定義されていなければならぬ。

使用例 {#使用例-7 .afff5}

```

!NODE, NGRP=TEST
1, 0.0, 0.0, 0.5
2, 0.0, 0.0, 1.0
3, 0.0,,1.5          --- Y座標は「0.0」
4,                      --- X,Y,Z座標は「0.0」

```

### 1.7.3.10 !SECTION (M1-9)

セクションの定義

1行目 {#行目-9 .afff5}

!SECTION, TYPE=<type>, EGRP=<egrp> [, optional parameter]

**パラメータ**

TYPE セクションタイプ (必須)  
EGRP 要素グループ名 (必須)  
MATERIAL ユーザー定義材料名 (必須)  
SECOPT 要素タイプ用補助パラメータ (省略可、省略した場合は=0となる)  
INPUT 外部ファイル名 (省略可)

パラメータ名	パラメータ値	内容
TYPE	SOLID	ロッド、三角形、四角形、四面体、五面体、六面体要素
	SHELL	シェル要素
	BEAM	梁要素
	INTERFACE	インターフェース要素
EGRP	<egrp>	要素グループ名
MATERIAL	<material >	ユーザー定義による材料名
SECOPT	<secopt>	= 0 : 指定なし、平面応力 = 1 : 平面ひずみ = 2 : 軸対称 = 10 : 0+次数低減積分 = 11 : 1+次数低減積分 = 12 : 2+次数低減積分
INPUT	<filename>	外部ファイル名 (省略可)、 2行目以降との併用も可能

2行目以降 {#行目以降-7 .afff5}

【TYPE=SOLID】の場合 {#typesolidの場合 .afff5}

(2行目) `THICKNESS`

**変数名 属性 内容**

THICKNESS R ト拉斯要素の場合、断面積 (必須)

「TYPE=SOLID」の場合、「THICKNESS」は省略可。

【TYPE=SHELL】の場合 {#typeshellの場合 .afff5}

(2行目) THICKNESS, INTEGPOINTS

**変数名 属性 内容**

THICKNESS R シェル断面厚さ

INTEGPOINTS I シェル断面方向積分点

【TYPE=BEAM】の場合 {#typebeamの場合 .afff5}

(2行目) vx,vy,vz,area,Iyy,Iz,Jx

**変数名 属性 内容**

vx,vy,vz R 参考軸方向

area R 断面面積

Iyy, Izz R 断面二次モーメント

Jx R ねじり定数

【TYPE=INTERFACE】の場合 {#typeinterfaceの場合 .afff5}

(2行目) THICKNESS, GAPCON, GAPRAD1, GAPRAD2

**変数名 属性 内容**

THICKNESS R 断面厚さ

GAPCON R ギャップ熱伝達係数（省略時0）

GAPRAD1 R ギャップ輻射熱伝達係数-1（省略時0）

GAPRAD2 R ギャップ輻射熱伝達係数-2（省略時0）

注意 {#注意-8 .afff5}

- パラメータ「TYPE」が要素タイプと整合していない場合はエラーとなる。
- SECTION情報を持たない要素がある場合はエラーとなる。
- セクション名が重複した場合はエラーとなる。

使用例 {#使用例-8 .afff5}

```
!SECTION, EGRP=SOLID1, TYPE=SOLID, MATERIAL=STEEL  
!SECTION, EGRP=SHELL2, TYPE=SHELL, MATERIAL=STEEL  
1.0, 5
```

### 1.7.3.11 !SGROUP (M1-10)

面グループの定義

1行目 {#行目-10 .afff5}

```
!SGROUP, SGRP=<sgrp> [, optional parameter]
```

**パラメータ**

SGRP 面グループ名（必須）

INPUT 外部ファイル名（省略可）

**パラメータ名 パラメータ値 内容**

SGRP <sgrp> 面グループ名

INPUT <filename> 外部ファイル名（省略可）、2行目以降との併用も可能

2行目以降 {#行目以降-8 .afff5}

(2行目) elem1, lsuf1, elem2, lsuf2, elem3, lsuf3, ...  
(以下同様)

**変数名 属性 内容**

elemX I 面グループに属する要素番号

lsufX I 面グループに属する要素の局所面番号

注意 {#注意-9 .afff5}

- 要素タイプと面番号については、「4章 要素ライブラリ」を参照のこと。
- （要素、局所面番号）という組み合わせによって面を構成する。1行に任意の数の面を入れることができる。また次のオプションが始まるまで、任意の数の行を挿入することができる。（要素、局所面番号）という組み合わせは必ず同一の行になければならない。
- 指定する要素は「!SGROUP」より前に定義されている必要がある。
- 要素が「!ELEMENT」オプションで定義されていない場合は無視され、警告メッセージが表示される。

- 「!ELEMENT」オプションで定義されていない要素を含む面は除外され、警告メッセージが表示される。
- 要素タイプと面番号の整合性が取れない面は除外され、警告メッセージが表示される。
- ひとつのグループを複数回にわけて定義できる。

使用例 {#使用例-9 .afff5}

```
!SGROUP, SGRP= SUF01
101, 1, 102, 1, 103, 2, 104, 2
201, 1, 202, 1
501, 1
!SGROUP, SGRP= SUF02
101, 2, 102, 2
!SGROUP, SGRP= EA01
601, 1
602, 2
--- グループ「SUF01」に「(601,1), (602,2)」が追加。
```

誤った使用例 {#誤った使用例-1 .afff5}

例1【（要素、局所面番号）の組が複数行にわたっている】

```
!SGROUP, SGRP= SUF01
101, 1, 102, 1, 103
1, 104, 1
```

例2【局所面番号と要素タイプの整合性がとれない】

```
!ELEMENT, TYPE= 211, SECTION= A
101, 1, 2, 3
102, 2, 3, 4
...
!SGROUP, SGRP= SUF01
101, 1
101, 2
101, 4
--- 三角形要素に第4面は存在しないので、この組み合わせは無視
される
```

### 1.7.3.12 !ZERO (M1-11)

絶対零度

1行目 {#行目-11 .afff5}

!ZERO

パラメータ

なし

2行目以降 {#行目以降-9 .afff5}

(2行目) ZERO

**変数名 属性 内容**

ZERO R 絶対零度

注意 {#注意-10 .afff5}

- 省略可能。省略された場合は「絶対零度=0」となる。
- 「!ZERO」を複数回定義すると、内容が更新され、警告メッセージが表示される。

使用例 {#使用例-10 .afff5}

!ZERO  
-273.16

### 1.7.3.13 !CONTACT PAIR (M1-12)

接触解析に用いる接触面ペアの定義

1行目 {#行目-12 .afff5}

**!CONTACT PAIR, NAME=<name> [,optional parameter]**

#### パラメータ

NAME 接触ペア名（必須）

TYPE タイプ（省略可）

#### パラメータ名 パラメータ値 内容

NAME <name> 接触ペア名

TYPE NODE-SURF スレーブ面は節点グループ

マスター面は面グループ（デフォルト）

SURF-SURF スレーブ面、マスター面とも面グループ

2行目以降 {#行目以降-10 .afff5}

(2行目以降) SLAVE\_GRP, MASTER\_GRP  
(以下同様)

#### 変数名 属性 内容

SLAVE\_GRP C スレーブ面の節点/面グループ名

MASTER\_GRP C マスター面の面グループ名

### 1.7.3.14 !END (M1-13)

メッシュデータの終端

このヘッダーが表れると、メッシュデータの読み込みを終了する。

1行目 {#行目-13 .afff5}

**!END**

パラメータ

なし

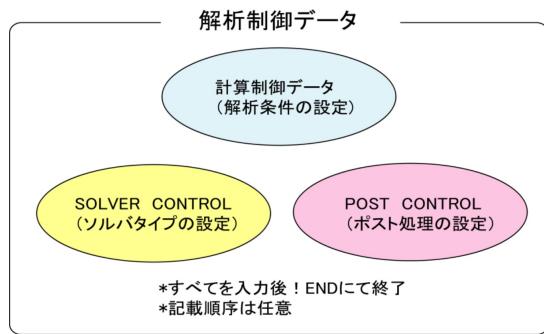
2行目以降 {#行目以降-11 .afff5}

なし

## 1.8 解析制御データ

### 1.8.1 解析制御データ概要

FrontISTRは、解析制御データファイルを入力して、下図に示す計算制御データ、ソルバー制御データおよびポスト処理（可視化）制御データを取得し、解析計算を実施する。



解析制御データファイルの特徴は以下のとおりである。

- 自由書式に基づくASCII形式のファイルである。
- “!”で始まるヘッダーとそれに続くデータから構成されている。
- ヘッダーの記述の順番は基本的に自由である。
- データの区切り記号には“,”を使用する。
- ファイル内は大きく分けて3つのゾーンに分かれている。
- ファイルの最後に“!END”を入力して終了とする。

#### 1.8.1.1 解析制御データ例

```
#####
# 計算制御データ部分 #####
### Control File for HEAT solver
!SOLUTION,TYPE=HEAT
!FIXTEMP
XMIN, 0.0
XMAX, 500.0

#####
# ソルバー制御データ部分 #####
### Solver Control
!SOLVER,METHOD=CG,PRECOND=1,ITERLOG=NO,TIMELOG=NO
100, 2
1.0e-8,1.0,0.0

#####
# ポスト制御(可視化)データ部分 #####
### Post Control
!WRITE,RESULT
!WRITE,VISUAL
!VISUAL, method=PSR
!surface_num = 1
!surface 1
!surface_style = 1
!display_method 1
!color_comp_name = TEMPERATURE
!color_subcomp = 1
!output_type = BMP
!x_resolution = 500
!y_resolution = 500
!num_of_lights = 1
!position_of_lights = -20.0, 5.8, 80.0
!viewpoint = -20.0 10.0 8.0
!up_direction = 0.0 0.0 1.0
!ambient_coef= 0.3
!diffuse_coef= 0.7
!specular_coef= 0.5
!color_mapping_style= 1
!!interval_mapping= -0.01, 0.02
!color_mapping_bar_on = 1
!scale_marking_on = 1
!num_of_scale = 5
```

```
!font_size = 1.5  
!font_color = 1.0 1.0 1.0
```

## 1.8.2 入力規則

解析制御データは、ヘッダー行、データ行、コメント行から構成される。

ヘッダー行には必ず一つのヘッダーが含まれる。

ヘッダー

解析制御データ内で、データの意味とデータブロックを特定する。  
行頭が “!” で始まる場合、ヘッダーであるとみなされる。

ヘッダー行

ヘッダーとそれに伴うパラメータを記述する。  
ヘッダー行はヘッダーで始まつていなければならない。パラメータが必要な場合は、“,”  
を用いてその後に続けなければならない。パラメータが値をとる場合は、パラメータの  
後に “=” が続き、その後に値を記述する。

ヘッダー行を複数行にわたって記述することはできない。

データ行

ヘッダー行の次の行から開始され、必要なデータを記述する。  
データ行は複数行にわたる可能性があるが、それは各ヘッダーで定義されるデータ記述  
の規則により決定される。

データ行は必要ない場合もある。

区切り文字

データの区切り文字にはカンマ “,” を用いる。

空白の扱い

空白は無視される。

名前

名前に使用可能な文字は、アンダースコア “\_”、ハイフン “-”、英数字 “a-z A-Z 0-9” で  
あるが、最初の一文字は “\_” または英字 “a-z A-Z” で始まつていなければならない。大  
文字小文字の区別はなく、内部的にはすべて大文字として扱われる。

また、名前の最大長は63文字である。

ファイル名

ファイル名に使用可能な文字は、アンダースコア “\_”、ハイフン “-”、ピリオド “.”、ス  
ラッシュ “/”、英数字 “a-z A-Z 0-9” である。  
ファイル名は、特に記述がない限りパスを含んでもよい。相対パス、絶対パスのいずれ  
も指定可能である。

また、ファイル名の最大長は1023文字である。

浮動小数点データ

指数はあってもなくてもよい。指数の前には、“E” または “e” の記号をつけなければな  
らない。

“E” または “e” どちらを使用してもかまわない。

コメント行

行頭が “!!” または “#” で始まる行はコメント行とみなされ、無視される。

コメント行はファイル中の任意の位置に挿入でき、その数に制限はない。

!END

メッセージデータの終端

このヘッダーが表れると、メッセージデータの読み込みを終了する。

## 1.8.3 解析制御データ

### 1.8.3.1 計算制御データのヘッダー一覧

FrontISTRでは、計算制御データに使用できる境界条件として以下のものがあげられる。

- 分布荷重条件 (物体力, 圧力荷重, 重力, 遠心力)
  - 集中荷重条件
  - 熱荷重

- 単点拘束条件 (SPC条件)
- ばね境界条件
- 接触
- 集中熱流束
- 分布熱流束
- 対流熱伝達境界
- 輻射熱伝達境界
- 規定温度境界

上記境界条件の定義方法は、メッシュデータ同様に “!” ヘッダーの形式で定義する。

以下、表 7.3.1 に全解析に共通な制御データのヘッダー一覧を示し、表 7.3.2 から解析種別別のヘッダー一覧を示す。

\*\* 表 7.3.2 静解析用制御データ \*\*

ヘッダー	意味	備考 説明番号
!STATIC	静解析の制御	2-1
!MATERIAL	材料名	2-2
!ELASTIC	弾性材料物性	2-2-1
!PLASTIC	塑性材料物性	2-2-2
!HYPERELASTIC	超弾性材料物性	2-2-3
!VISCOELASTIC	粘弾性材料物性	2-2-4
!CREEP	クリープ材料物性	2-2-5
!DENSITY	質量密度	2-2-6
!EXPANSION_COEFF	線膨張係数	2-2-7
!TRS	粘弹性温度依存性	2-2-8
!FUID	流体物性	2-2-9
!USE_MATERIAL	ユーザー定義材料	2-2-10
!BOUNDARY	変位境界条件	2-3
!SPRING	ばね境界条件	2-3-1
!CLOAD	集中荷重	2-4
!DLOAD	分布荷重	2-5
!ULOAD	ユーザー定義外部荷重	2-6
!CONTACT_ALGO	接触解析アルゴリズム	2-7
!CONTACT	接触	2-8
!TEMPERATURE	熱応力解析における節点温度	2-9
!REFTEMP	熱応力解析における参照温度	2-10
!STEP	解析ステップ制御	2-11
!AUTOINC_PARAM	自動増分制御	2-12
!TIME_POINTS	計算及び出力時刻の指定	2-13

\*\* 表 7.3.4 热伝導解析用制御データ \*\*

ヘッダー	意味	備考	説明番号
!HEAT	熱伝導解析の制御	熱伝導解析で必須	4-1
!FIXTEMP	節点温度		4-2
!CFLUX	節点に与える集中熱流束		4-3
!DFLUX	要素面に与える分布熱流束/内部発熱		4-4
!SFLUX	面グループによる分布熱流束		4-5
!FILM	境界面に与える熱伝達係数		4-6
!SFILM	面グループによる熱伝達係数		4-7

<b>!RADIADE</b>	境界面に与える輻射係数	4-8
<b>!SRADIATE</b>	面グループによる輻射係数	4-9
<b>!WELD_LINE</b>	溶接線	4-10

変数名	属性	内容
ng1,ng2, ...	C/I	節点グループ名/節点番号
t1, t2, ...	R	温度値

#### 1.8.3.1.0.0.1 TYPE= VELOCITY/ACCELERATIONの場合

(2行目) ng1, dof1, v1

(3行目) ng2, dof2, v2

...

変数名	属性	内容
ng1,ng2, ...	C/I	節点グループ名/節点番号
dof1, dof2, ...	I	自由度番号
v1, v2, ...	R	速度/加速度値

### 1.8.3.2 静解析用制御データ

#### 1.8.3.2.1 !STATIC (2-1)

静的解析を行う。(Default値、省略可)

##### 1.8.3.2.1.1 パラメータ

なし

#### 1.8.3.2.2 !MATERIAL (2-2)

##### 1.8.3.2.2.1 材料物性の定義

材料物性の定義は!MATERIALと以降に置く!ELASTICITY、!PLASTICITYなどとセットで使用する。!MATERIALの前に置く!ELASTICITY、!PLASTICTYなどは無視される。

注: 解析制御データで!MATERIALを定義すると、メッシュデータ内の!MATERIAL定義は無視される。解析制御データで!MATERIALを定義しない場合は、メッシュデータ内の!MATERAIL定義が用いられる。

##### 1.8.3.2.2.2 パラメータ

NAME = 材料名

#### 1.8.3.2.3 !ELASTIC (2-2-1)

弾性材料の定義

##### 1.8.3.2.3.1 パラメータ

TYPE = ISOTROPIC (Default値) / ORTHOTROPIC / USER  
DEPENDENCIES = 0 (Default値) / 1

\*\* 2行目以降 \*\*

#### 1.8.3.2.3.1.1 TYPE = ISOTROPICの場合

(2行目) YOUNGS, POISSON, Temperature

変数名	属性	内容
YOUNGS	R	ヤング率
POISSON	R	ポアソン比
Temperature	R	温度(DEPENDENCIES=1の時に必要)

#### 1.8.3.2.3.1.2 TYPE= ORTHOTROPICの場合

(2行目) E1, E2, E3, v12, v13, v23, G12, G13, G23, Tempreature

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \end{bmatrix} = \begin{bmatrix} 1/E_1 & -\nu_{12}/E_1 & -\nu_{13}/E_1 & 0 & 0 & 0 \\ & 1/E_2 & -\nu_{23}/E_2 & 0 & 0 & 0 \\ & & 1/E_3 & 0 & 0 & 0 \\ & & & 1/G_{12} & 0 & 0 \\ & & & & 1/G_{23} & 0 \\ & & & & & 1/G_{31} \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix}$$

対称

#### 1.8.3.2.3.1.3 TYPE = USERの場合

(2行目-10行目) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

#### 1.8.3.2.4 !PLASTIC (2-2-2)

塑性材料の定義

##### 1.8.3.2.4.1 パラメータ

YIELD = MISES (Default値)、Mohr-Coulomb、DRUCKER-PRAGER、USER  
HARDEN = BILINEAR (Default値)、MULTILINEAR、SWIFT、RAMBERG-OSGOOD、KINEMATIC、  
COMBINED  
DEPENDENCIES = 0 (Default値) / 1

\*\* 2行目以降 \*\*

##### 1.8.3.2.4.1.1 YIELD = MISESの場合 (Default値)

\*\* HARDEN = BILINEAR (Default値)の場合 \*\*

(2行目) YIELD0, H

\*\* HARDEN = MULTILINEARの場合 \*\*

(2行目) YIELD, PSTRAIN, Temperature  
(3行目) YIELD, PSTRAIN, Temperature

...続く

\*\* HARDEN = SWIFTの場合 \*\*

(2行目) ε0, K, n

\*\* HARDEN = RAMBERG-OSGOODの場合 \*\*

(2行目)  $\varepsilon_0$ , D, n

\*\* HARDEN = KINEMATICの場合 \*\*

(2行目) YIELD0, C

\*\* HARDEN = COMBINEDの場合 \*\*

(2行目) YIELD0, H, C

1.8.3.2.4.1.2 YIELD = Mohr-Coulomb または Drucker-Pragerの場合

\*\* HARDEN = BILINEAR(Default値)の場合 \*\*

(2行目) c, FAI, H

\*\* HARDEN = MULTILINEARの場合 \*\*

(2行目) FAI

(3行目) PSTRAIN, c

(4行目) PSTRAIN, c

...続く

HARDEN = 他は無視され、Default値(BILINEAR)になる。

変数名	属性	内容
YIELD0	R	初期降伏応力
H	R	硬化係数
PSTRAIN	R	塑性ひずみ
YIELD	R	降伏応力
$\varepsilon_0, K, n$	R	$\bar{\sigma} = k(\varepsilon_0 + \bar{\varepsilon})^n$
$\varepsilon_0, D, n$	R	$\varepsilon = \frac{\sigma}{E} + \varepsilon_0 \left( \frac{\sigma}{D} \right)^n$
FAI	R	内部摩擦角
c	R	粘着力
C	R	線形移動硬化係数
Tempearture	R	温度(DEPENDENCIES=1の時に必要)
v1, v2...v10	R	材料定数

\*\* YIELD= USERの場合 \*\*

(2行目以降) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

#### 1.8.3.2.4.2 使用例

```
!PLASTIC, YIELD=MISES, HARDEN=MULTILINEAR, DEPENDENCIES=1
276.0, 0.0, 20.
296.0, 0.0018, 20.
299.0, 0.0053, 20.
303.0, 0.008, 20.
338.0, 0.0173, 20.
372.0, 0.0271, 20.
400.0, 0.037, 20.
419.0, 0.0471, 20.
437.0, 0.0571, 20.
450.0, 0.0669, 20.
460.0, 0.0767, 20.
469.0, 0.0867, 20.
477.0, 0.0967, 20.
```

```

276.0, 0.0, 100.
276.0, 0.0018, 100.
282.0, 0.0053, 100.
295.0, 0.008, 100.
330.0, 0.0173, 100.
370.0, 0.0271, 100.
392.0, 0.037, 100.
410.0, 0.0471, 100.
425.0, 0.0571, 100.
445.0, 0.0669, 100.
450.0, 0.0767, 100.
460.0, 0.0867, 100.
471.0, 0.0967, 100.
128.0, 0.0, 400.
208.0, 0.0018, 400.
243.0, 0.0053, 400.
259.0, 0.008, 400.
309.0, 0.0173, 400.
340.0, 0.0271, 400.
366.0, 0.037, 400.
382.0, 0.0471, 400.
396.0, 0.0571, 400.
409.0, 0.0669, 400.
417.0, 0.0767, 400.
423.0, 0.0867, 400.
429.0, 0.0967, 400.

```

指定の温度また塑性ひずみに関する上記の入力データから内挿して、加工硬化係数を計算することになる。各温度値に対して、同じPSTRAIN配列を入力することが必要になる。

#### **1.8.3.2.5 !HYPERELASTIC (2-2-3)**

超弾性材料の定義

##### **1.8.3.2.5.1 パラメータ**

```

TYPE = NEOHOOKE (Default值)
      MOONEY-RIVLIN
      ARRUDA-BOYCE
      USER

```

\*\* 2行目以降 \*\*

##### **1.8.3.2.5.1.1 TYPE = NEOHOOKEの場合**

(2行目) C<sub>10</sub>, D

**変数名 属性 内容**

C <sub>10</sub>	R	材料定数
D	R	材料定数

##### **1.8.3.2.5.1.2 TYPE = MOONEY-RIVLINの場合**

(2行目) C<sub>10</sub>, C<sub>01</sub>, D

**変数名 属性 内容**

C <sub>10</sub>	R	材料定数
C <sub>01</sub>	R	材料定数
D	R	材料定数

-----

### 1.8.3.2.5.1.3 TYPE = ARRUDA-BOYCEの場合

(2行目) mu, lambda\_m, D

変数名	属性	内容
mu	R	材料定数
lambda_m	R	材料定数
D	R	材料定数

### 1.8.3.2.5.1.4 TYPE = USERの場合

(2行目-10行目) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

## 1.8.3.2.6 !VISCOELASTIC (2-2-4)

粘弹性材料の定義

### 1.8.3.2.6.1 パラメータ

DEPENDENCIES = 依存する変数の数(未実装)

\*\* 2行目以降 \*\*

(2行目) g, t

変数名	属性	内容
g	R	せん断緩和弾性率
t	R	緩和時間

## 1.8.3.2.7 !CREEP (2-2-5)

クリープ材料の定義

### 1.8.3.2.7.1 パラメータ

TYPE = NORTON (Default値)

DEPENDENCIES = 0 (Default値) / 1

\*\* 2行目以降 \*\*

(2行目) A, n, m, Tempearature

変数名	属性	内容
A	R	材料係数
n	R	材料係数
m	R	材料係数
Tempearture	R	温度(DEPENDENCIES=1の時に必要)

## 1.8.3.2.8 !DENSITY (2-2-6)

質量密度の定義

### 1.8.3.2.8.1 パラメータ

DEPENDENCIES = 依存する変数の数(未実装)

\*\* 2行目以降 \*\*

(2行目) density

変数名 属性 内容

density R 質量密度

#### 1.8.3.2.9 !EXPANSION\_COEFF(2-2-7)

線膨張係数の定義

##### 1.8.3.2.9.1 パラメータ

TYPE = 材質タイプ

ISOTROPIC(等方性:Default) / ORTHOTROPIC(直交異方性)

DEPENDENCIES = 0(Default値) / 1

\*\* 2行目以降 \*\*

##### 1.8.3.2.9.1.1 TYPE=ISOTROPICの場合

(2行目) expansion, Temperature

##### 1.8.3.2.9.1.2 TYPE=ORTHOTROPICの場合

(2行目) α11, α22, α33, Temperature

変数名 属性 内容

expansion R 線膨張係数

α11, α22, α33 R 線膨張係数

Tempearture R 温度(DEPENDENCIES=1の時に必要)

#### 1.8.3.2.10 !TRS (2-2-8)

熱レオロジー単純化(Thermorheological Simplicity)による粘弾性材料の温度依存性の定義

この定義は!VISCOELASTICの後ろに置かなければならない。前にある場合は、この定義が無視される。

##### 1.8.3.2.10.1 パラメータ

DEFINITION = WLF(Default値) / ARRHENUS

\*\* 2行目以降 \*\*

(2行目) θ₀, C₁, C₂

変数名 属性 内容

θ₀ R 参照温度

C₁, C₂ R 材料定数

#### 1.8.3.2.11 !FLUID (2-2-9)

流体の定義

##### 1.8.3.2.11.1 パラメータ

TYPE = INCOMP\_NEWTONIAN (Default値)

\*\* 2行目以降 \*\*

(2行目) mu

変数名 属性 内容

mu R 粘度

#### 1.8.3.2.12 !USER/MATERIAL (2-2-10)

ユーザー定義材料の入力

##### 1.8.3.2.12.1 パラメータ

NSTATUS = 材料の状態変数の数を指定する(デフォルト:1)

\*\* 2行目以降 \*\*

(2行目-10行目) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

#### 1.8.3.2.13 !BOUNDARY (2-3)

変位境界条件の定義

##### 1.8.3.2.13.1 パラメータ

GRPID = グループID

AMP = 時間関数名 (!AMPLITUDEで指定、動解析で有効)

ROT\_CENTER = 回転変位拘束の中心節点番号または節点集合名。

指定した場合、その !BOUNDARY は回転変位拘束であると認識される。

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_idS, DOF\_idE, Value

変数名 属性 内容

NODE\_ID I/C 節点番号または節点グループ名

DOF\_idS I 拘束自由度の開始番号

DOF\_idE I 拘束自由度の終了番号

Value R 拘束値(デフォルト:0)

##### 1.8.3.2.13.2 使用例

!BOUNDARY, GRPID=1

1, 1, 3, 0.0

ALL, 3, 3,

※ 拘束値は0.0

!BOUDARY, TORQUE\_CENTER=7, GRPID=1

ROT\_NODES, 1, 1, 3.141

ROT\_NODES, 2, 2, -4.188

ROT\_NODESに対して、節点7を中心とし、大きさ  $|(3.141, -4.188)| = 5.233[\text{rad}]$ 、回転軸  $(3/5, -4/5, 0)$  の回転を加える。

ROT\_CENTERによる回転は並進3自由度に対する変位拘束であり、シェル要素に対する 4, 5, 6

自由度拘束とは異なる。

ROT\_CENTERを指定した場合

- ROT\_CENTERに指定する節点集合は、1つだけの節点からなる集合とする。
  - 1つの!BOUNDARYデータブロックの中では、NODE\_IDは全て同一にする。
  - 指定されない自由度は全て0で拘束される。

#### 1.8.3.2.14 !SPRING (2-3-1)

ばね境界条件の定義

##### 1.8.3.2.14.1 パラメータ

GRPID = グループID

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_id, Value

変数名 属性 内容

NODE\_ID I/C 節点番号または節点グループ名

DOF\_id I 拘束自由度

Value R ばね定数

##### 1.8.3.2.14.2 使用例

```
!SPRING, GRPID=1
  1, 1, 0.5
```

#### 1.8.3.2.15 !CLOAD (2-4)

集中荷重の定義

##### 1.8.3.2.15.1 パラメータ

GRPID = グループID

AMP = 時間関数名 (!AMPLITUDEで指定、動解析で有効)

ROT\_CENTER = 回転中心節点番号または節点集合名。指定した場合、その !CLOAD はトルク荷重であると認識される。

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_id, Value

変数名 属性 内容

NODE\_ID I/C 節点番号または節点グループ名

DOF\_id I 自由度番号

Value R 荷重値

##### 1.8.3.2.15.2 使用例

```
!CLOAD, GRPID=1
  1, 1, 1.0e3
  ALL, 3, 10.0
!CLOAD, ROT_CENTER=7, GRPID=1
  TORQUE_NODES, 1, 3
```

TORQUE\_NODES, 3, -4

ROT\_NODESに対して、節点7を中心とし、大きさ $\|(3, 0, -4)\| = 5$ 、回転軸 $(3/5, 0, -4/5)$ のトルク荷重を加える。

ROT\_CENTERによるトルク荷重は、内部的には指定したトルク相当の節点荷重を与えるものであり、シェル要素に対する4, 5, 6自由度への節点荷重とは異なる。

ROT\_CENTERを指定した場合

- ROT\_CENTERに指定する節点集合は、1つだけの節点からなる集合とする。
  - 1つの!CLOADデータブロックの中では、NODE\_IDは全て同一にする。

#### 1.8.3.2.16 !DLOAD (2-5)

分布荷重の定義

##### 1.8.3.2.16.1 パラメータ

GRPID = グループID

AMP = 時間関数名 (!AMPLITUDEで指定、動解析で有効)

FOLLOW = YES(Default値) / NO

(圧力荷重の方向を変形に追随させるか否かの指定、有限変形解析で有効)

\*\* 2行目以降 \*\*

(2行目) ID\_NAME, LOAD\_type, param1, param2,...

変数名	属性	内 容
ID_NAME	I/C	面グループ名、要素グループ名または要素番号
LOAD_type	C	荷重タイプ番号
param*	R	荷重パラメータ(下記参照)

##### 1.8.3.2.16.1.1 荷重パラメータ

荷重タイプ 番号	種類	パラメー タ数	パラメータ並びとその意味
S	面グループで指定の面 への圧力	1	圧力値
P0	シェル要素への圧力	1	圧力値
PX	シェル要素X方向への 圧力	1	圧力値
PY	シェル要素Y方向への 圧力	1	圧力値
PZ	シェル要素Z方向への 圧力	1	圧力値
P1	第1面への圧力	1	圧力値
P2	第2面への圧力	1	圧力値
P3	第3面への圧力	1	圧力値
P4	第4面への圧力	1	圧力値
P5	第5面への圧力	1	圧力値
P6	第6面への圧力	1	圧力値
BX	X方向への体積力	1	体積力値
BY	Y方向への体積力	1	体積力値
BZ	Z方向への体積力	1	体積力値

GRAV	重力	4	重力加速度,重力の方向余弦
CENT	遠心力	7	角速度, 回転軸上の点の位置ベクトル、回転軸の方向ベクトル

### 1.8.3.2.16.1.2 使用例

```
!DLOAD, GRPID=1
  1, P1, 1.0
  ALL, BX, 1.0
  ALL, GRAV, 9.8, 0.0, 0.0, -1.0
  ALL, CENT, 188.495, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0
```

### 1.8.3.2.17 !ULOAD (2-6)

ユーザ一定義荷重の入力

#### 1.8.3.2.17.1 パラメータ

FILE = ファイル名(必須)

### 1.8.3.2.18 !CONTACT\_ALGO (2-7)

接触解析アルゴリズムの指定

#### 1.8.3.2.18.1 パラメータ

TYPE = SLAGRANGE (Lagrange乗数法)  
ALAGRANGE (拡張Lagrange乗数法)

### 1.8.3.2.19 !CONTACT (2-8)

接触条件の定義

#### 1.8.3.2.19.1 パラメータ

GRPID = 境界条件グループID  
INTERACTION = SSLID(Default値) / FSLID  
NTOL = 接触法線方向収束閾値(デフォルト: 1.e-5)  
TTOL = 接触切線方向収束閾値(デフォルト: 1.e-3)  
NPENALTY = 接触法線方向Penalty(デフォルト: 剛性マトリクス×1.e3)  
TPENALTY = 接触切線方向Penalty(デフォルト: 1.e3)

\*\* 2行目以降 \*\*

(2行目) PAIR\_NAME, fcoef, factor

変数名	属性	内容
PAIR_NAME	C	接触ペア名 (!CONTACT_PAIRにて定義)
fcoef	R	摩擦係数 (デフォルト: 0.0)
factor	R	摩擦のペナルティ剛性

#### 1.8.3.2.19.2 使用例

```
!CONTACT_ALGO, TYPE=SLAGRANGE
!CONTACT, GRPID=1, INTERACTION=FSLID
  CP1, 0.1, 1.0e+5
```

### 1.8.3.2.20 !TEMPERATURE (2-9)

熱応力解析に用いる節点温度の指定

#### 1.8.3.2.20.1 パラメータ

READRESULT = 热伝導解析の結果ステップ数。  
指定された場合、热伝導解析の結果ファイルから順次に温度を入力し、2行目以降は無視される。  
SSTEP = 热伝導解析結果の読み込むを行う最初のステップ番号（デフォルト：1）  
INTERVAL = 热伝導解析結果の読み込むを行うステップ間隔（デフォルト：1）

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, Temp\_Value

変数名	属性	内容
NODE_ID	I/C	節点番号または節点グループ名
Temp_Value	R	温度(デフォルト: 0)

#### 1.8.3.2.20.2 使用例

```
!TEMPERATURE
 1, 10.0
 2, 120.0
 3, 330.0
!TEMPERATURE
 ALL, 20.0
!TEMPERATURE, READRESULT=1, SSTEP=1
```

### 1.8.3.2.21 !REFTEMP (2-10)

熱応力解析における参照温度の定義

#### 1.8.3.2.21.1 パラメータ

なし

\*\* 2行目以降 \*\*

(2行目) Value

変数名	属性	内容
Value	R	参照温度(デフォルト: 0)

### 1.8.3.2.22 !STEP (2-11)

解析ステップの設定

非線形静解析、非線形動解析では必須

上記以外の解析でこの定義を省略すると、すべての境界条件が有効になり、1ステップで計算  
材料特性が粘弾性およびクリープの場合、TYPE=VISCOを指定し、計算時間条件を設定

#### 1.8.3.2.22.1 パラメータ

TYPE = STATIC(default値) / VISCO (準静的解析)  
SUBSTEPS = 境界条件の分割ステップ数(デフォルト: 1)

```
CONVERG = 収束判定閾値(デフォルト: 1.0e-6)
MAXITER = 非線形解析における最大反復計算回数(デフォルト: 50)
AMP = 時間関数名(!AMPLITUDEで指定)
INC_TYPE = FIXED(固定増分・default値) / AUTO(自動増分)
MAXRES = 最大許容残差の設定(デフォルト: 1.0e+10)
TIMEPOINTS = 時刻リスト名(!TIME_POINTS, NAMEで指定)
AUTOINCPARAM = 自動接触パラメータセット名(!AUTOINC_PARAM, NAMEで指定)
MAXCONTITER = 接触解析における最大接触反復回数(デフォルト: 10)
```

\*\* 2行目 \*\*

INC\_TYPE=FIXEDの場合 (TYPE=STATICの場合は省略可)

(2行目) DTIME, ETIME

**変数名 属性 内容**

DTIME R 時間増分(デフォルト: 1/SUBSTEPS)  
ETIME R ステップ時間幅(デフォルト: 1)

INC\_TYPE=AUTOの場合 (TYPEによらず指定)

(2行目) DTIME\_INIT, ETIME, MINDT, MAXDT

**変数名 属性 内容**

DTIME\_INIT R 初期時間増分  
ETIME R ステップ時間幅  
MINDT R 時間増分下限  
MAXDT R 時間増分上限

\*\* 3行目以降 \*\*

BOUNDARY, id	id!=BOUNDARYで定義したGRPID
LOAD, id	id!=CLOAD, !DLOAD, !SPRING,
!TEMPERATUREで定義したGRPID	
CONTACT, id	id!=CONTACTで定義したGRPID

#### 1.8.3.2.22.2 使用例

##### 固定時間増分の使用例

```
!STEP, CONVERG=1.E-8
0.1, 1.0
BOUNDARY, 1
LOAD, 1
CONTACT, 1
```

自動増分調整を有効にし、初期時間増分0.01、ステップ時間幅2.5、時間増分下限1E-5、時間増分上限0.3、最大サブステップ数を200に設定する

```
!STEP, INC_TYPE=AUTO, SUBSTEPS=200
0.01, 2.5, 1E-5, 0.3
```

自動増分調整を有効にし、時刻リストTP1を計算・結果出力時刻として指定する

```
!STEP, INC_TYPE=AUTO, TIMEPOINTS=TP1
0.1, 2.0, 1E-3, 0.2
```

#### 1.8.3.2.22.3 備考

- 自動増分調整の場合、SUBSTEPSは最大サブステップ数として扱われる

- 時刻リスト名TIMEPOINTSおよび自動接触パラメータセット名AUTOINCPARAMの指定はINC\_TYPE=AUTOのときのみ有効
- TIMEPOINTSを指定する場合、指定先の!TIME\_POINTは!STEPカードより前に定義されていなければならない。
- AUTOINCPARAMを指定する場合、指定先の!AUTOINC\_PARAMは!STEPカードより前に定義されていなければならない。また、本パラメータ省略時はデフォルトの自動増分パラメータセットが使用される

#### **1.8.3.2.23 !AUTOINC\_PARAM (2-12)**

自動増分パラメータの指定。

##### **1.8.3.2.23.1 パラメータ**

###### **パラメータ名 属性 内容**

NAME C 自動増分パラメータ名 (必須)

\*\* 2行目 \*\*

減少条件およびその際の時間増分減少率を指定する。

(2行目) RS, NS\_MAX, NS\_SUM, NS\_CONT, N\_S

###### **変数名 属性 内容**

RS R 時間増分減少率 (デフォルト : 0.25)

NS\_MAX I Netwon法最大反復数の閾値 (デフォルト : 10)

NS\_SUM I Netwon法合計反復数の閾値 (デフォルト : 50)

NS\_CONT I 接触反復数の閾値 (デフォルト : 10)

N\_S I 減少条件成立までのサブステップ数 (デフォルト : 1)

\*\* 3行目 \*\*

増加条件およびその際の時間増分増加率を指定する。

(3行目) RL, NL\_MAX, NL\_SUM, NL\_CONT, N\_L

###### **変数名 属性 内容**

RL R 時間増分増加率 (デフォルト : 1.25)

NL\_MAX I Netwon法最大反復数の閾値 (デフォルト : 1)

NL\_SUM I Netwon法合計反復数の閾値 (デフォルト : 1)

NL\_CONT I 接触反復数の閾値 (デフォルト : 1)

N\_L I 増加条件成立までのサブステップ数 (デフォルト : 2)

\*\* 4行目 \*\*

カットバックの設定を行う。

(4行目) RC, N\_C

###### **変数名 属性 内容**

RC R カットバック時の時間増分減少率 (デフォルト : 0.25)

N\_C I 連続カットバック回数の許容上限 (デフォルト : 5)

#### **1.8.3.2.23.2 使用例**

デフォルト設定と同じ設定の場合

```
!AUTOINC_PARAM, NAME=AP1
0.25, 10, 50, 10, 1
1.25, 1, 1, 1, 2
0.25, 5
```

#### 1.8.3.2.24 !TIME\_POINTS (2-13)

##### 1.8.3.2.24.1 パラメータ

パラメータ名	属性	内容
NAME	C	時刻リスト名（必須）
TIME	C	STEP（ステップ開始時刻起点の時刻による入力・デフォルト値）／TOTAL（初期からの通算時刻による入力）
GENERATE	-	開始時間、終了時間、時間間隔による時刻点の自動生成

\*\* 2行目以降 \*\*

GENERATEを使用しない場合

(2行目以降) TIME

##### 変数名 属性 内容

TIME R 時刻

GENERATEを使用する場合

(2行目) STIME, ETIME, INTERVAL

##### 変数名 属性 内容

STIME R 開始時刻

ETIME R 終了時刻

INTERVAL R 時刻点の間隔

##### 1.8.3.2.24.2 使用例

時刻 1.5, 2.7, 3.9 を通算時刻として GENERATE を使用せず定義する

```
!TIME_POINTS, TIME=TOTAL, TIME=,NAME=TP1
1.5
2.7
3.9
```

時刻 1.5, 2.7, 3.9 をステップ時刻として GENERATE を使用して定義

```
!TIME_POINTS, TIME=STEP, GENERATE, NAME=TP1
1.5, 3.9, 1.2
```

##### 1.8.3.2.24.3 備考

- 時刻点の入力は昇順に行わなければならない

#### 1.8.3.3 7.4.3 固有値解析用制御データ

##### 1.8.3.3.1 !EIGEN (3-1)

## 固有値解析のパラメータ設定

### 1.8.3.3.1.1 パラメータ

なし

\*\* 2行目以降 \*\*

(2行目) NGET, LCZTOL, LCZMAX

変数名	属性	内容
NSET	I	固有値数
LCZTOL	R	許容差 (デフォルト: 1.0e-8)
LCZMAX	I	最大反復数 (デフォルト: 60)

### 1.8.3.3.1.2 使用例

```
!EIGEN
 3, 1.0e-10, 40
```

## 1.8.3.4 7.4.4 熱伝導解析用制御データ

### 1.8.3.4.1 !HEAT (4-1)

計算に関する制御データの定義

### 1.8.3.4.1.1 パラメータ

なし

\*\* 2行目以降 \*\*

(2行目) DT, ETIME, DTMIN, DELTMX, ITMAX, EPS

変数名	属性	内容
DT	R	初期時間増分 ≤ 0: 定常計算 > 0: 非定常計算
ETIME	R	非定常計算時間(非定常計算時必須) 最小時間増分
DTMIN	R	≤ 0: 固定時間増分 > 0: 自動時間増分
DELT MX	R	許容変化温度
ITMAX	I	非線形計算最大反復数(デフォルト: 20)
EPS	R	収束判定値 (デフォルト: 1.0e-6)

### 1.8.3.4.1.2 使用例

```
!HEAT
  (データなし) -----
!HEAT
  0.0 -----
!HEAT
  10.0, 3600.0 -----
!HEAT
```

```
10.0, 3600.0, 1.0 ----- 自動時間増分非定常計算  
!HEAT  
10.0, 3600.0, 1.0, 20.0 --- 自動時間増分非定常計算
```

#### 1.8.3.4.2 !FIXTEMP (4-2)

規定温度の定義

##### 1.8.3.4.2.1 パラメータ

AMP = 流束履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) NODE\_GRP\_NAME, Value

変数名	属性	内容
NODE_GRP_NAME	C/I	節点グループ名または節点番号
Value	R	温度(デフォルト: 0)

##### 1.8.3.4.2.2 使用例

```
!FIXTEMP  
ALL, 20.0  
!FIXTEMP, AMP=FTEMP  
ALL, 1.0
```

#### 1.8.3.4.3 !CFLUX (4-3)

節点にあたえる集中熱流束の定義

##### 1.8.3.4.3.1 パラメータ

AMP = 流束履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) NODE\_GRP\_NAME, Value

変数名	属性	内容
NODE_GRP_NAME	C/I	節点グループ名または節点番号
Value	R	熱流束値

##### 1.8.3.4.3.2 使用例

```
!CFLUX  
ALL, 1.0E-3  
!CFLUX, AMP=FUX1  
ALL, 1.0
```

#### 1.8.3.4.4 !DFLUX (4-4)

要素の面にあたえる分布熱流束と内部発熱の定義

##### 1.8.3.4.4.1 パラメータ

AMP = 流束履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) ELEMENT\_GRP\_NAME, LOAD\_type, Value

変数名	属性	内容
ELEMENT_GRP_NAME	C/I	要素グループ名または要素番号
LOAD_type	C	荷重タイプ番号
Value	R	熱流束値

#### 1.8.3.4.4.2 使用例

```
!DFLUX
  ALL, S1, 1.0
!DFLUX, AMP=FLUX2
  ALL, S0, 1.0
```

#### 1.8.3.4.4.2.1 荷重パラメータ

##### 荷重タイプ番号 作用面 パラメータ

BF	要素全体	発熱量
S1	第1面	熱流束値
S2	第2面	熱流束値
S3	第3面	熱流束値
S4	第4面	熱流束値
S5	第5面	熱流束値
S6	第6面	熱流束値
S0	シェル面	熱流束値

#### 1.8.3.4.5 !SFLUX (4-5)

面グループによる分布熱流束の定義

##### 1.8.3.4.5.1 パラメータ

AMP = 流束履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) SURFACE\_GRP\_NAME, Value

変数名	属性	内容
SURFACE_GRP_NAME	C	面グループ名
Value	R	熱流束値

#### 1.8.3.4.5.2 使用例

```
!SFLUX
  SURF, 1.0
!SFLUX, AMP=FLUX3
  SURF, 1.0
```

#### 1.8.3.4.6 !FILM (4-6)

境界面にあたえる熱伝達係数の定義

#### 1.8.3.4.6.1 パラメータ

AMP1 = 熱伝達係数履歴テーブル名 (!AMPLITUDE で指定)  
AMP2 = 霧囲気温度履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) ELEMENT\_GRP\_NAME, LOAD\_type, Value, Sink

変数名	属性	内容
ELEMENT_GRP_NAME	C/I	要素グループ名または要素番号
LOAD_type	C	荷重タイプ番号
Value	R	熱伝達係数
Sink	R	霧囲気温度

#### 1.8.3.4.6.2 使用例

```
!FILM
  FSURF, F1, 1.0, 800.0
!FILM, AMP1=TFILM
  FSURF, F1, 1.0, 1.0
```

#### 1.8.3.4.6.2.1 荷重パラメータ

荷重タイプ番号	作用面	パラメータ
F1	第1面	熱伝達係数と霧囲気温度
F2	第2面	熱伝達係数と霧囲気温度
F3	第3面	熱伝達係数と霧囲気温度
F4	第4面	熱伝達係数と霧囲気温度
F5	第5面	熱伝達係数と霧囲気温度
F6	第6面	熱伝達係数と霧囲気温度
F0	シェル面	熱伝達係数と霧囲気温度

#### 1.8.3.4.7 !SFILM (4-7)

面グループによる熱伝達係数の定義

#### 1.8.3.4.7.1 パラメータ

AMP1 = 熱伝達係数履歴テーブル名 (!AMPLITUDE で指定)  
AMP2 = 霧囲気温度履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) SURFACE\_GRP\_NAME, Value, Sink

変数名	属性	内容
SURFACE_GRP_NAME	C	面グループ名
Value	R	熱伝達率
Sink	R	霧囲気温度

#### 1.8.3.4.7.2 使用例

```
!SFILM
  SFSURF, 1.0, 800.0
```

```
!SFILM, AMP1=TSFILM, AMP2=TFILM  
SFSURF, 1.0, 1.0
```

#### 1.8.3.4.8 !RADIADE (4-8)

境界面にあたえる輻射係数の定義

##### 1.8.3.4.8.1 パラメータ

AMP1 = 輻射係数履歴テーブル名 (!AMPLITUDE で指定)  
AMP2 = 雰囲気温度履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) ELEMENT\_GRP\_NAME, LOAD\_type, Value, Sink

変数名	属性	内容
ELEMENT_GRP_NAME	C/I	要素グループ名または要素番号
LOAD_type	C	荷重タイプ番号
Value	R	輻射係数
Sink	R	雰囲気温度

##### 1.8.3.4.8.2 使用例

```
!RADIADE  
RSURF, R1, 1.0E-9, 800.0  
!RADIADE, AMP2=TRAD  
RSURF, R1, 1.0E-9, 1.0
```

##### 1.8.3.4.8.2.1 荷重パラメータ

荷重タイプ番号	作用面	パラメータ
R1	第1面	輻射係数と雰囲気温度
R2	第2面	輻射係数と雰囲気温度
R3	第3面	輻射係数と雰囲気温度
R4	第4面	輻射係数と雰囲気温度
R5	第5面	輻射係数と雰囲気温度
R6	第6面	輻射係数と雰囲気温度
R0	シェル面	輻射係数と雰囲気温度

#### 1.8.3.4.9 !SRADIATE (4-9)

面グループによる輻射係数の定義

##### 1.8.3.4.9.1 パラメータ

AMP1 = 輻射係数履歴テーブル名 (!AMPLITUDE で指定)  
AMP2 = 雰囲気温度履歴テーブル名 (!AMPLITUDE で指定)

\*\* 2行目以降 \*\*

(2行目) SURFACE\_GRP\_NAME, Value, Sink

変数名	属性	内容
SURFACE_GRP_NAME	C	面グループ名
Value	R	輻射係数

Sink

R 雰囲気温度

#### 1.8.3.4.9.2 使用例

```
!SRADIATE
  RSURF, 1.0E-9, 800.0
!SRADIATE, AMP2=TSRAD
  RSURF, 1.0E-9, 1.0
```

#### 1.8.3.4.10 !WELD\_LINE (4-10)

溶接線（直線）の定義

##### 1.8.3.4.10.1 パラメータ

なし

\*\* 2行目 \*\*

(2行目) I, U, Coef, V

変数名	属性	内容
-----	----	----

I	R	電流
U	R	電圧
Coef	R	入熱効率
V	R	溶接トーチの移動速度

\*\* 3行目 \*\*

(3行目) EGROUP, XYZ, C1, C2, H, tstart

変数名	属性	内容
-----	----	----

EGROUP	C	入熱する要素グループ名
XYZ	I	溶接トーチの移動方向(自由度番号)
C1	R	溶接トーチの始点座標
C2	R	溶接トーチの終点座標
H	R	溶接源の幅、この幅範囲内の要素は入熱を受ける
tstart	R	溶接開始時刻

#### 1.8.3.5 7.4.5 動解析用制御データ

##### 1.8.3.5.1 !DYNAMIC (5-1)

動解析の制御

!BOUNDARY、!CLOAD、!DLOADで指定された各!AMPLITUDEにおける時刻tは、0.0から始まっているなければならない。

##### 1.8.3.5.1.1 パラメータ

```
TYPE = LINEAR    : 線形動解析
      NONLINEAR : 非線形動解析
```

\*\* 2行目以降 \*\*

(2行目) idx\_eqa, idx\_resp

変数名	属性	内容
		運動方程式の解法(直接時間積分法)(デフォルト: 1)
idx_eqa	I	1: 陰解法(Newmark-β法) 11: 陽解法(中央差分法)
		解析の種類(デフォルト: 1)
idx_resp	I	1: 時刻歴応答解析 2: 周波数応答解析

1.8.3.5.1.1.1 idx\_resp=1の場合(時刻歴応答解析)

(3行目) t\_start, t\_end, n\_step, t\_delta

変数名	属性	内容
t_start	R	解析開始時間(デフォルト: 0.0)、未使用
t_end	R	解析終了時間(デフォルト: 1.0)、未使用
n_step	I	全STEP数(デフォルト: 1)
t_delta	R	時間増分(デフォルト: 1.0)

(4行目) gamma, beta

変数名	属性	内容
gamma	R	Newmark-β法のパラメータγ(デフォルト: 0.5)
beta	R	Newmark-β法のパラメータβ(デフォルト: 0.25)

(5行目) idx\_mas, idx\_dmp, ray\_m, ray\_k

変数名	属性	内容
		質量マトリックスの種類(デフォルト: 1)
idx_mas	I	1: 集中質量マトリックス 2: consistent質量マトリックス
idx_dmp	I	1: Rayleigh減衰(デフォルト: 1)
ray_m	R	Rayleigh減衰のパラメータRm(デフォルト: 0.0)
ray_k	R	Rayleigh減衰のパラメータRk(デフォルト: 0.0)

(6行目) nout, node\_monit\_1, nout\_monit

変数名	属性	内容
nout	I	未使用
node_monit_1	I	モニタリング節点番号(グローバル)または節点グループ名
nout_monit	I	モニタリングの結果出力間隔(デフォルト: 1)

注) 本行で指定したモニタリング節点の情報は、節点のグローバル番号をNIDとして、変位についてはファイル<dyna\_disp\_NID.txt>へ出力され、その並びは、step番号、当該時間、NID、u<sub>1</sub>、u<sub>2</sub>、u<sub>3</sub>である。速度および加速度についても、それぞれファイル<dyna\_velo\_NID.txt>、<dyna\_acce\_NID.txt>へ同様の並びで出力される。節点ひずみについてはファイル<dyna\_strain\_NID.txt>へ出力され、その並びは、step番号、当該時間、NID、e<sub>11</sub>、e<sub>22</sub>、e<sub>33</sub>、e<sub>12</sub>、e<sub>23</sub>、e<sub>13</sub>である。節点応力についてはファイル<dyna\_stress\_NID.txt>へ出力され、その並びは、step番号、当該時間、NID、s<sub>11</sub>、s<sub>22</sub>、s<sub>33</sub>、s<sub>12</sub>、s<sub>23</sub>、s<sub>13</sub>、smisesである。節点グループを用いて複数節点のモニタリングを指定した場合、上記の各ファイルは節点ごとに別ファイルに出力される。また、この出力を指定した場合、解析モデル全体の運動エネルギー、変形エネルギーおよび全エネルギーが<dyna\_energy.txt>へ出力される。

(7行目) iout\_list(1), iout\_list(2), iout\_list(3), iout\_list(4), iout\_list(5),

iout_list(6)		
変数名	属性	内容
iout_list(1) I		変位の出力指定(デフォルト: 0) 0: 出力しない 1: 出力する
iout_list(2) I		速度の出力指定(デフォルト: 0) 0: 出力しない 1: 出力する
iout_list(3) I		加速度の出力指定(デフォルト: 0) 0: 出力しない 1: 出力する
iout_list(4) I		反力の出力指定(デフォルト: 0) 0: 出力しない 1: 出力する
iout_list(5) I		ひずみの出力指定(デフォルト: 0) 0: 出力しない(要素ベース及び節点ベース) 1: 出力する 2: 出力する(節点ベース) 3: 出力する(要素ベース)
iout_list(6) I		応力の出力指定(デフォルト: 0) 0: 出力しない(要素ベース及び節点ベース) 1: 出力する 2: 出力する(節点ベース) 3: 出力する(要素ベース)

#### 1.8.3.5.1.2 使用例

```
!DYNAMIC, TYPE=NONLINEAR
1 , 1
0.0, 1.0, 500, 1.0000e-5
0.5, 0.25
1, 1, 0.0, 0.0
100, 55, 1
0, 0, 0, 0, 0
```

#### 1.8.3.5.1.3 idx\_resp=2の場合(周波数応答解析)

(3行目) f\_start, f\_end, n\_freq, f\_disp

変数名	属性	内容
f_start	R	下限周波数
f_end	R	上限周波数
n_freq	I	応答計算点数
f_disp	R	変位を測定する周波数

(4行目) t\_start, t\_end

変数名	属性	内容
t_start	R	実時間での振動開始時間
t_end	R	実時間での振動終了時間

(5行目) idx\_mas, idx\_dmp, ray\_m, ray\_k

変数名	属性	内容
idx_mas	I	質量マトリクスの種類(1固定(集中質量マトリクス))

idx\_dmp I 1: Rayleigh減衰  
 ray\_m R Rayleigh減衰のパラメータRm(デフォルト: 0.0)  
 ray\_k R Rayleigh減衰のパラメータRk(デフォルト: 0.0)

(6行目) nout, vistype, nodeout

変数名	属性	内容
nout	I	時間空間でのサンプリング数
vistype	I	可視化データ出力指定(1: モード空間, 2: 物理空間)
nodeout	I	周波数空間モニタリング節点ID

(7行目) iout\_list(1), iout\_list(2), iout\_list(3), iout\_list(4), iout\_list(5),  
 iout\_list(6)

変数名	属性	内容
iout_list(1)	I	出力制御 変位 (1: 出力する、 0: 出力しない)
iout_list(2)	I	出力制御 速度 (1: 出力する、 0: 出力しない)
iout_list(3)	I	出力制御 加速度 (1: 出力する、 0: 出力しない)
iout_list(4)	I	無視
iout_list(5)	I	無視
iout_list(6)	I	無視

#### 1.8.3.5.2 使用例

```

!DYNAMIC
 11 , 2
 14000, 16000, 20, 15000.0
 0.0, 6.6e-5
 1, 1, 0.0, 7.2E-7
 10, 2, 1
 1, 1, 1, 1, 1, 1

```

#### 1.8.3.6 !VELOCITY (5-2)

速度境界条件の定義

##### 1.8.3.6.1 パラメータ

TYPE = INITIAL (初期速度境界条件)  
 = TRANSIT (!AMPLITUDEで指定した時間歴速度境界条件 ; デフォルト)  
 AMP = 時間関数名 (!AMPLITUDEで指定)  
 !AMPLITUDEで時間 t と係数 f(t) の関係を与える。  
 下記 Value に係数 f(t) を乗じた値がその時刻の拘束値になる  
 (指定しない場合: 時間と係数関係は f(t) = 1.0 となる)。

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_idS, DOF\_idE, Value

変数名	属性	内容
NODE_ID	I/C	節点番号または節点グループ名
DOF_idS	I	拘束自由度の開始番号
DOF_idE	I	拘束自由度の終了番号
Value	R	拘束値(デフォルト: 0)

#### 1.8.3.6.2 使用例

```

!VELOCITY, TYPE=TRANSIT, AMP=AMP1
 1, 1, 1, 0.0
 ALL, 3, 3
※ 拘束値は0.0
!VELOCITY, TYPE=INITIAL
 1, 3, 3, 1.0
 2, 3, 3, 1.0
 3, 3, 3, 1.0

```

- 速度境界条件の場合、変位境界条件の場合とは異なり、複数の自由度をまとめて定義できないため、DOF\_idSとDOF\_idEは同一番号でなければならない。
  - TYPEがINITIALの場合、AMPが無効になる。

### 1.8.3.7 !ACCELERATION (5-3)

加速度境界条件の定義

#### 1.8.3.7.1 パラメータ

```

TYPE = INITIAL (初期加速度境界条件)
      = TRANSIT (AMPLITUDEで指定した時間歴加速度境界条件: デフォルト)
AMP  = 時間関数名 (!AMPLITUDEで指定)
      !AMPLITUDEで時間tと係数f(t)の関係を与える。
      下記 Value に係数f(t)を乗じた値がその時刻の拘束値になる
      (指定しない場合: 時間と係数関係はf(t) = 1.0となる)。

```

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_idS, DOF\_idE, Value

変数名	属性	内容
NODE_ID	I/C	節点番号または節点グループ名
DOF_idS	I	拘束自由度の開始番号
DOF_idE	I	拘束自由度の終了番号
Value	R	拘束値(デフォルト: 0)

#### 1.8.3.7.2 使用例

```

!ACCELERATION, TYPE=TRANSIT, AMP=AMP1
 1, 1, 3, 0.0
 ALL, 3, 3
※ 拘束値は0.0
!ACCELERATION, TYPE=INITIAL
 1, 3, 3, 1.0
 2, 3, 3, 1.0
 3, 3, 3, 1.0

```

- 加速度境界条件の場合、変位境界条件の場合とは異なり、複数の自由度をまとめて定義できないため、DOF\_idSとDOF\_idEは同一番号でなければならない。
  - TYPEがINITIALの場合、AMPが無効になる。

### 1.8.3.8 !COUPLE (5-4)

連成面の定義(連成解析でのみ使用)

#### 1.8.3.8.1 パラメータ

TYPE =	1: 片方向連成	(FrontISTRはデータ受信から開始)
	2: 片方向連成	(FrontISTRはデータ送信から開始)

3: Staggered双方向連成 (FrontISTRはデータ受信から開始)  
 4: Staggered双方向連成 (FrontISTRはデータ送信から開始)  
 5: 分離反復双方向連成 (FrontISTRはデータ受信から開始)  
 6: 分離反復双方向連成 (FrontISTRはデータ送信から開始)  
**ISTEP =** ステップ数  
 解析初期からここで指定したステップ数まで流体力を0倍から1倍まで線形関数を乗じて増大させ、  
 それ以降は入力流体力をそのまま適用する  
**WINDOW => 0** : 流体力にウインドウ関数(\*)を乗じて適用

(\*)  $\frac{1}{2}(1 - \cos \frac{2\pi i}{N})$ , ( $i$  : 現ステップ、  $N$  : 現在の解析の総ステップ数)

\*\* 2行目以降 \*\*

(2行目) COUPLING\_SURFACE\_ID

変数名	属性	内容
SURFACE_ID	C	面グループ名

#### 1.8.3.8.2 使用例

```
!COUPLE , TYPE=1
SCOUPLE1
SCOUPLE2
```

#### 1.8.3.9 !EIGENREAD (5-5)

周波数応答解析に用いる固有値・固有モードの指定

##### 1.8.3.9.1 パラメータ

なし

\*\* 2行目以降 \*\*

(2行目) eigenlog\_filename

変数名	属性	内容
eigenlog_filename	C	固有値解析のログファイル名

(3行目) start\_mode, end\_mode

変数名	属性	内容
start_mode	I	固有値解析に使用する指定モード始点
end_mode	I	固有値解析に使用する指定モード終点

#### 1.8.3.9.2 使用例

```
!EIGENREAD
eigen_0.log
1, 5
```

#### 1.8.3.10 !FLOAD (5-6)

周波数応答解析用集中荷重の定義

##### 1.8.3.10.1 パラメータ

LOAD CASE = (実部の指定: 1, 虚部の指定: 2)

\*\* 2行目以降 \*\*

(2行目) NODE\_ID, DOF\_id, Value

変数名	属性	内容
NODE_ID	I/C	節点番号または節点グループ名または面グループ名
DOF_id	I	自由度番号
Value	R	荷重値

#### 1.8.3.10.2 使用例

```
!FLOAD, LOAD CASE=2
    _PickedSet5, 2, 1.0
```

### 1.8.4 7.4.6 ソルバー制御データ

#### 1.8.4.1 !SOLVER (6-1)

ソルバーの制御

必須の制御データ.

#### 1.8.4.2 パラメータ

METHOD = 解法 (CG、BiCGSTAB、GMRES、GPBiCG、DIRECT、DIRECTmkl、MUMPS)  
DIRECT : 接触解析以外での直接法(逐次処理のみ)  
DIRECTmkl : 接触解析におけるIntel MKLによる直接法(逐次処理のみ)  
MUMPS : 並列直接法パッケージMUMPSによる直接法  
直接法を選択したとき、データ行は無視される。  
1、2自由度問題では、CG、DIRECT、MUMPSのみ有効  
シェル要素は、DIRECT、MUMPSのみ有効  
3自由度用の反復法はOpenMPによるスレッド並列が利用可能

PRECOND = 反復法の前処理手法 (1、2、3、5、10、11、12)  
1, 2 : (Block) SSOR (3自由度用のみマルチカラーオーダリング付き)  
3 : (Block) Diagonal Scaling  
5 : マルチグリッド前処理パッケージMLによるAMG(試験的)  
10 : Block ILU(0)  
11 : Block ILU(1)  
12 : Block ILU(2)  
10, 11, 12は3自由度問題でのみ利用可能  
OpenMPによるスレッド並列時はSSORまたはDiagonal Scalingを推奨

ITERLOG = 反復法ソルバー収束履歴出力の有無 (YES/NO) (デフォルト: NO)

TIMELOG = ソルバー計算時間出力の有無 (YES/NO) (デフォルト: NO)

USEJAD = ベクトル機向けオーダリングの有無 (YES/NO) (デフォルト: NO)  
3自由度問題で反復法使用時のみ有効

SCALING = 行列の対角成分を1とするスケーリングの有無 (YES/NO) (デフォルト: NO)  
3自由度問題で反復法使用時のみ有効

DUMPTYPE = 行列ダンプ型式(NONE、MM、CSR、BSR) (主にデバッグ用)  
NONE : ダンプしない(デフォルト)  
MM : マトリックスマーケット型式  
CSR : Compressed Sparse Row (CSR) 型式  
BSR : Blocked CSR型式

DUMPEXIT = 行列ダンプ直後のプログラム終了 (YES/NO) (デフォルト: NO)

**MPCMETHOD** = 多点拘束条件の処理手法(1、2、3)  
 1: ベナルティ法 (直接法使用時のデフォルト)  
 2: MPC-CG法  
 3: 陽的自由度消去法 (反復法使用時のデフォルト)  
  
**ESTCOND** = 条件数推定の頻度 (試験的)  
 指定された反復ごと、および、反復終了時に条件数推定を実施  
 0の場合は推定を行わない  
  
**METHOD2** = 第2の解法 (BiCGSTAB、GMRES、GPBiCG)  
 METHODにCGを指定した場合のみ有効  
 CG法が発散した場合に自動的に切り替えて求解を行う  
 他のパラメータやデータ行の情報は同じものが利用される

\*\* 2行目以降 \*\*

(2行目) NIER, iterPREmax, NREST, NCOLOR\_IN

変数名	属性	内容
NIER	I	反復回数(デフォルト: 100) Additive Schwarzによる前処理の繰り返し数(デフォルト: 1)
iterPREmax	I	(推奨値は、逐次計算、前処理に対角スケーリングを用いる場合、 および、MPCを含むモデルの計算では1、その他の並列計算では2)
NREST	I	クリロフ部分空間数(デフォルト: 10) (解法としてGMRESを選択したときのみ有効)
NCOLOR_IN	I	マルチカラーオーダリングにおける目標色数(デフォルト: 10) (OpenMPのスレッド数が2以上の時のみ有効)

(3行目) RESID, SIGMA\_DIAG, SIGMA

変数名	属性	内容
RESID	R	打ち切り誤差(デフォルト値: 1.0e-8)
SIGMA_DIAG	R	前処理行列計算時に対角成分にかける倍率(デフォルト値: 1.0)
SIGMA	R	未使用(デフォルト値: 0.0)

#### 1.8.4.2.1 使用例

```
!SOLVER, METHOD=CG, PRECOND=1, ITERLOG=YES, TIMELOG=YES
 10000, 2
 1.0e-8, 1.0, 0.0
```

### 1.8.5 7.4.7 ポスト処理(可視化)制御データ

#### 1.8.5.1 !VISUAL (P1-0)

可視化手法を指定する。

**METHOD** = PSR : サーフエスレンダリング  
**visual\_start\_step** : 可視化処理を始めるタイムステップ番号の指定 (デフォルト: 1)  
**visual\_end\_step** : 可視化処理を終了するタイムステップ番号の指定 (デフォルト: すべて)  
**visual\_interval\_step** : 可視化処理を行うタイムステップ間隔の指定 (デフォルト: 1)

#### 1.8.5.2 !surface\_num, !surface, !surface\_style (P1-1~3)

#### 1.8.5.2.1 !surface\_num (P1-1)

1つのサーフェスレンダリング内のサーフェス数

例: 図7.4.1は4つのサーフェスがあり、2つは等値面でpressure=1000.0 と pressure=-1000.0、2つは平面の切り口で z= -1.0 と z= 1.0である。

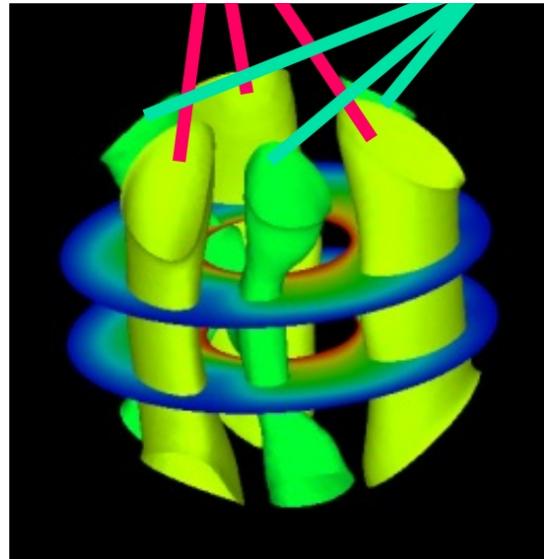


図7.4.1 surface\_numの設定例

#### 1.8.5.2.2 !surface (P1-2)

サーフェスの内容を設定する。

例： 図7.4.2は4つのサーフェスがありその内容は以下の通りである。

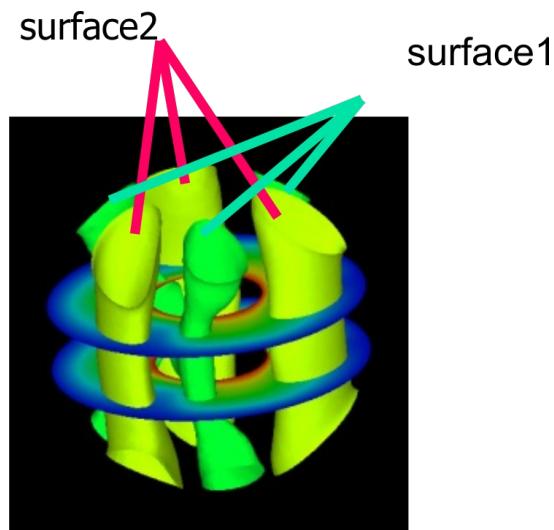


図7.4.2 surfaceの設定例

```
!surface_num = 2
!SURFACE
!surface_style = 2
!data_comp_name = press
```

```

!iso_value = 1000.0
!display_method = 4
!specified_color = 0.45
!output_type = BMP
!SURFACE
!surface_style = 2
!data_comp_name = press
!iso_value = -1000.0
!display_method = 4
!specified_color = 0.67

```

#### 1.8.5.2.3 !surface\_style (P1-3)

サーフェスのスタイルを指定する。

1. 境界面
2. 等値面
3. 任意の2次曲面

$$\text{coef[1]}x^2 + \text{coef[2]}y^2 + \text{coef[3]}z^2 + \text{coef[4]}xy + \text{coef[5]}xz \\ + \text{coef[6]}yz + \text{coef[7]}x + \text{coef[8]}y + \text{coef[9]}z + \text{coef[10]}=0$$



図7.4.3 surface/styleの設定例

#### 1.8.5.2.4 !display\_method (P1-4)

表示方法 (省略値: 1)

1. 色コードの表示
2. 境界線表示
3. 色コード及び境界線表示
4. 指定色一色の表示
5. 色分けによる等値線表示

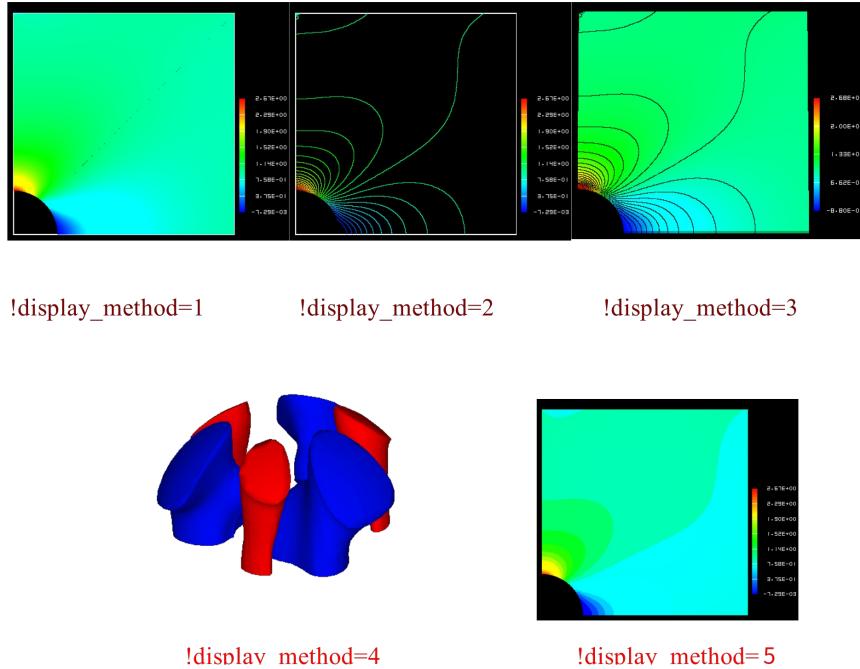


図7.4.4 display\_methodの設定例

### 1.8.5.3 !color\_comp\_name, !color\_comp, !color\_subcomp (P1-5, P1-7, P1-8)

物理量からマラマップへの対応を指定する。必要な物理量やその自由度番号に名前をつける。これにより結果データの構造体node\_label(:)やnn\_dof(:)に名前がはいる。

Then you can define which one you hope to map into color by

#### 1.8.5.3.1 !color\_comp\_name (文字列、省略値: 初めの変数)

例:

```
!color_comp_name = pressure
    静解析では = DISPLACEMENT : 結果変位データの指定
    = STRAIN : ひずみデータの指定
    = STRESS : 応力データの指定
    伝熱解析では = TEMPERATURE : 結果温度データの指定
```

#### 1.8.5.3.1.1 !color\_comp (整数、省略値: 0)

物理量の識別番号 (0以上の整数)

例:

```
!color_comp = 2
```

結果データ種別の識別番号指定と成分名だが、未実装。

#### 1.8.5.3.1.2 !color\_subcomp (整数、省略値: 1)

物理量がベクトル量のような自由度数1以上の時、その自由度番号

例:

```
!color_subcomp = 0
```

```

!color_comp_name=DISPLACEMENT指定の場合
 1: X成分, 2: Y成分, 3: Z成分

!color_comp_name=STRAIN指定の場合
 1:  $\epsilon_x$ , 2:  $\epsilon_y$ , 3:  $\epsilon_z$ 
 4:  $\epsilon_{xy}$ , 5:  $\epsilon_{yz}$ , 6:  $\epsilon_{zx}$ 

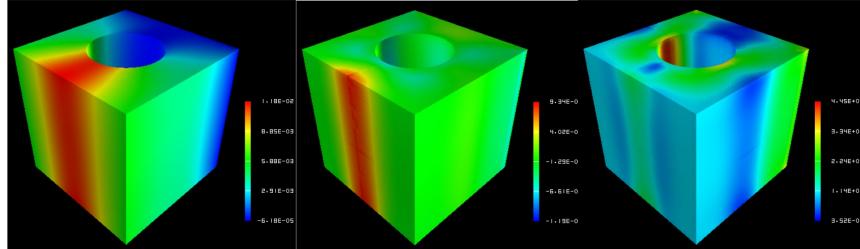
!color_comp_name=STRESS指定の場合
 1:  $\sigma_x$ , 2:  $\sigma_y$ , 3:  $\sigma_z$ 
 4:  $\tau_{xy}$ , 5:  $\tau_{yz}$ , 6:  $\tau_{zx}$ 

!color/comp_name=TEMPERATURE指定の場合
 1: 温度

```

構造解析において例えば

**物理量 変位 ひずみ 応力**  
自由度数 3 6 7



```

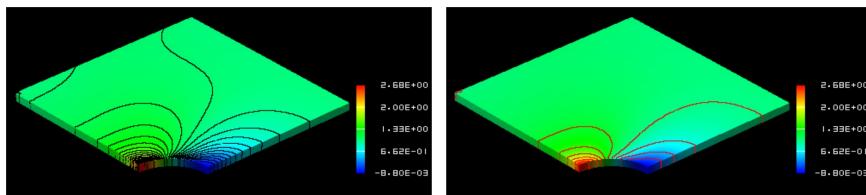
!color_comp_name=displacement !color_comp_name=strain !color_comp = 3
!color_subcomp = 1           !color_subcomp_name = 1   !color_subcomp = 7

```

図7.4.5 color\_comp, color\_subcompおよびcolor\_comp\_nameの設定例

#### 1.8.5.3.2 !isoline\_number, !isoline\_color (P1-9, P2-22)

display\_method=2,3または5の時



```

!isoline_number = 30          !isoline_number = 10
!isoline_color = 0.0, 0.0, 0.0 !isoline_color = 1.0, 0.0, 0.0

```

図7.4.6 isoline\_numberとisoline\_colorの設定例

#### 1.8.5.3.3 !initial\_style, !deform\_style (P1-15, P1-16)

初期の形状、変形後の形状の表示スタイルを指定する。

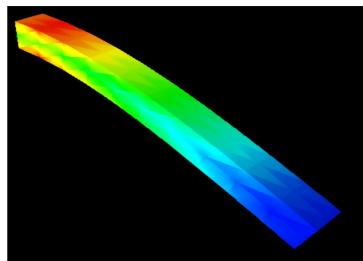
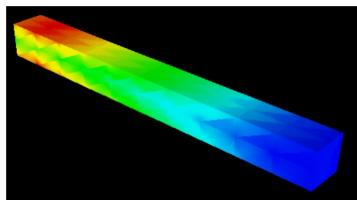
0. 無
1. 実線メッシュ(指定がなければ青で表示)
2. グレー塗りつぶし
3. シェーディング(物理属性をカラー対応させる)
4. 点線メッシュ(指定がなければ青で表示)

#### 1.8.5.3.3.1 !deform\_scale (P1-14)

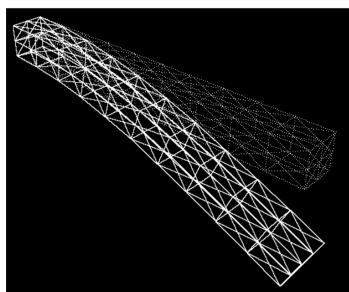
変形を表示する際の変位スケールを指定する。

Default: 自動

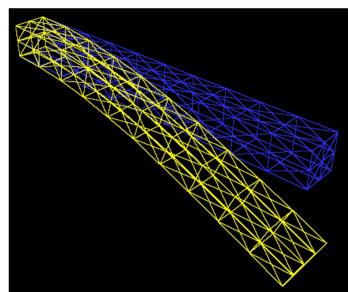
```
standard_scale = 0.1 * sqrt(x_range2 + y_range2 + z_range2) / max_deform
```



```
!initial_style=2  
!deform_style = 0
```

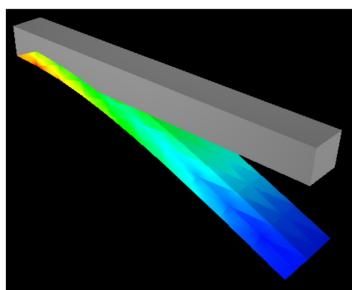
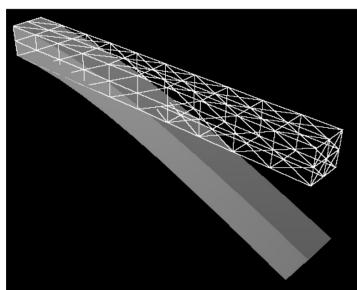


```
!initial_style=0  
!deform_style = 2
```



```
!initial_style=4  
!deform_style = 1  
!initial_line_color = 1.0, 1.0, 1.0
```

```
!initial_style=1  
!deform_style = 1 NASTRAN style  
!initial_line_color = default
```



```
!initial_style=1  
!deform_style = 2
```

```
!initial_style=2  
!deform_style = 3
```

図7.4.7 display\_stylesの設定例

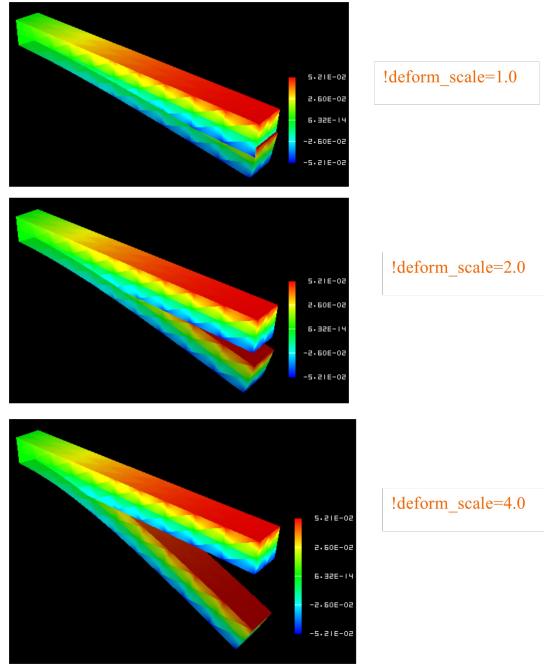


図7.4.8 deform\_scaleの設定例

#### 1.8.5.3.3.2 !output\_type (P1-19)

出力ファイルの型を指定する。 (省略値: AVS)

AVS	: AVS用UCDデータ (物体表面上のみ)
BMP	: イメージデータ (BMPフォーマット)
COMPLETE_AVS	: AVS用UCDデータ
COMPLETE_REORDER_AVS	: AVS用UCDデータで 節点・要素番号を並び替える
SEPARATE_COMPLETE_AVS	: 分割領域ごとのAVS用UCDデータ
COMPLETE_MICROAVS	: AVS用UCDデータで物理量をスカラーで出力する
BIN_COMPLETE_AVS	: COMPLETE_AVSをバイナリー形式で出力する
FSTR_FEMAP_NEUTRAL	: FEMAP用ニュートラルファイル

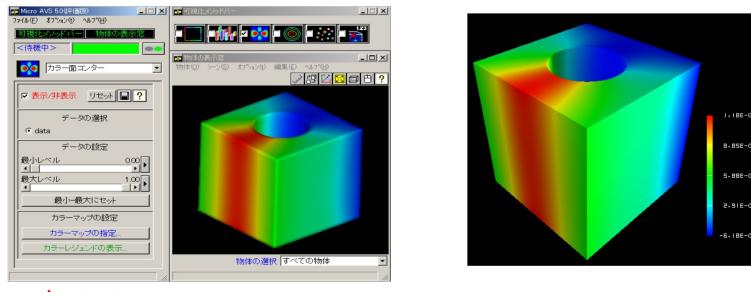


図7.4.9 output\_typeの例

#### 1.8.5.3.4 !x\_resolution, !y\_resolution (P2-1, P2-2)

output\_type=BMPの時、解像度を指定する。

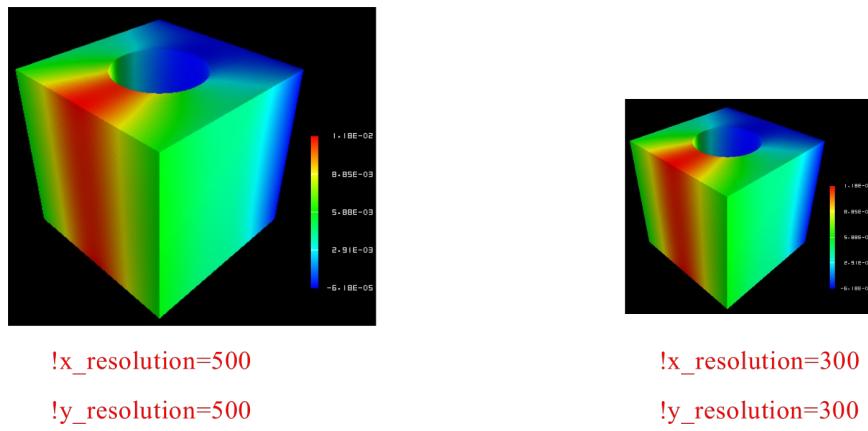


図7.4.10 `x_resolution`と`y_resolution`の設定例

#### 1.8.5.3.5 `!viewpoint`, `!look_at_point`, `!up_direction` (P2-5, P2-6, P2-7)

**1.8.5.3.5.1 `viewpoint`:** 視点の位置を座標で指定する。

省略値

```
x = (xmin + xmax)/2.0,
y = ymin + 1.5 * (ymax - ymin),
z = zmin + 1.5 * (zmax - zmin)
```

**1.8.5.3.5.2 `look_at_point`:** 視線の位置を指定する。

(省略値: データの中心)

`up_direction`: Viewpoint, `look_at_point` と`up_direction` にてビューフレームを指定する。  
 default : 0.0 0.0 1.0

View coordinate frame:

原点	:	<code>look_at_point</code>
<code>z</code> 軸	:	<code>viewpoint</code> - <code>look_at_point</code>
<code>x</code> 軸	:	<code>up</code> × <code>z</code> axis
<code>y</code> 軸	:	<code>z</code> axis × <code>x</code> axis

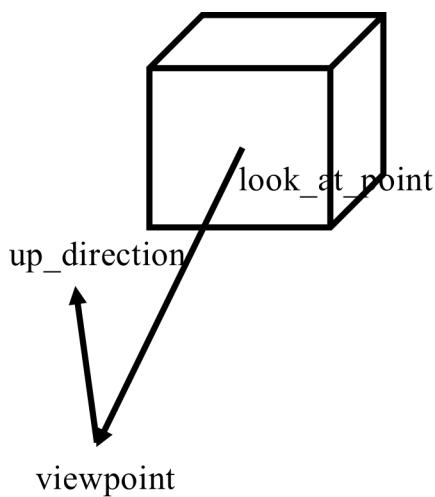


図7.4.11 ビューフレームの決定法

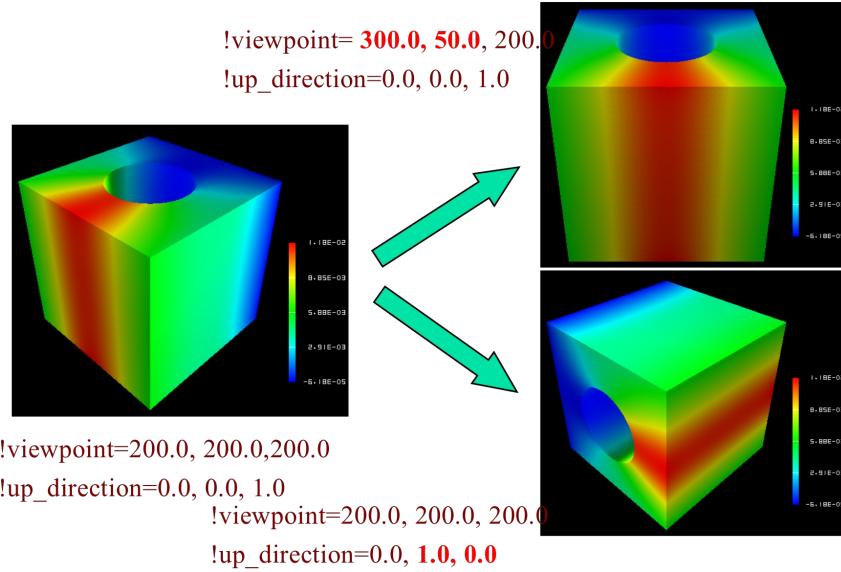


図7.4.12 !viewpoint, !look\_at\_pointとup\_directionの設定例

#### 1.8.5.3.6 !ambient\_coef, !diffuse\_coef, !specular\_coef (P2-8, P2-9, P2-10)

照明モデルの係数設定

ambient\_coefを増加すると3次元の奥行き方向の情報が損なわれる。

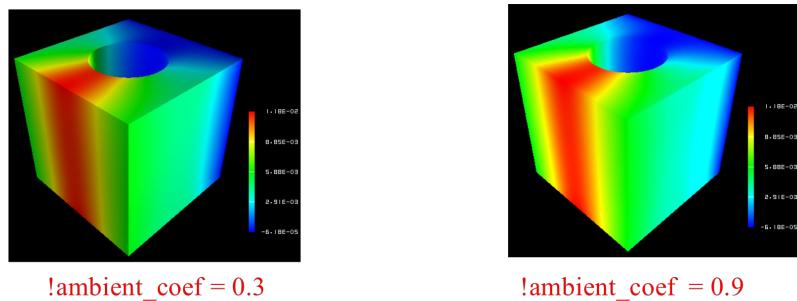


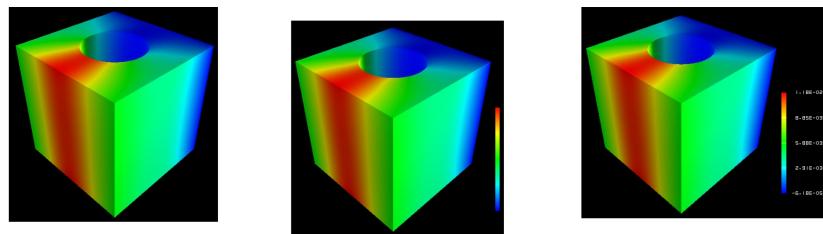
図7.4.13 照明モデルパラメータの設定例

#### 1.8.5.3.7 !color\_mapping\_bar\_on, !scale\_marking\_on, !num\_of\_scales (P2-16, P2-17, P2-18)

**!color\_mapping\_bar\_on** : color mapping barの表示有無を指定する。  
 0: off 1: on (省略値: 0)

**!scale\_marking\_on** : color mapping barのメモリの有無を指定する  
 0: off 1: on (省略値: 0)

**!num\_of\_scales** : メモリの数を指定する。  
 (省略値: 3)

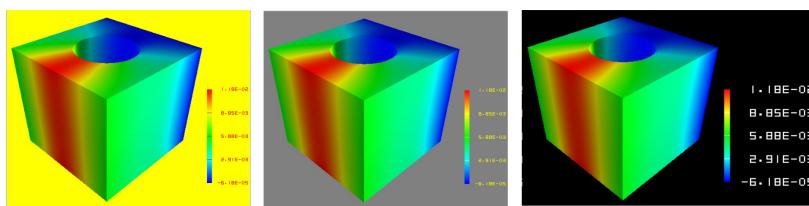


```
!color_mapping_bar_on=0 !color_mapping_bar_on=1 !color_mapping_bar_on=1
!scale_marking_on=0      !scale_markig_on =0      !scale_marking_on=1
                                         !num_of_scale = 5
```

図7.4.14 color\_mapping\_barの表示の例

#### 1.8.5.3.8 !font\_size, !font\_color, !background\_color (P2-19, P2-20, P2-21)

背景色や文字フォントを指定する。



```
!background_color=1.0,1.0,0.0 !background_color=0.5, 0.5, 0.5 !background_color
=0.0, 0.0,0.0
!font_color=1.0, 0.0, 0.0      !font_color=1.0, 1.0, 0.0      !font_color=1.0,
1.0, 1.0
!font_size=1.5                !font_size =1.5            !font_size=2.5
```

図7.4.15 backgroundとfontの設定例

#### 1.8.5.3.9 !data\_comp\_name, !data\_comp, !data\_subcomp (P3-1, P3-3, P3-4)

surface\_style=2の時、可視化する等値面の物理量を指定する。



```
!data_comp_name=pressure          !data_comp_name=vorticity
!data_subcomp=3
```

図7.4.16 data\_comp, data\_subcomp及びdata\_comp\_nameの設定例

#### 1.8.5.3.10 !method (P4-1)

面との切り口を指定する際、その面の設定方法を指定する。

```
!surface_num =2
```

```

!surface
!surface_style=3
!method=5
!coef=0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, -0.35
!color_comp_name = temperature

```

図7.4.17 methodの設定例

これにより平面 $z=0.35$  と $z=-0.35$ . の切り口が可視化される。

## 1.9 ユーザサブルーチン

ユーザーがFrontISTRの機能をプログラミングにより拡張するためのインターフェースを提供する。これらのインターフェースは、基本的にサブルーチンヘッダを含むFORTRANサブルーチンで、入出力変数の記述とこれらの変数のための宣言文である。ルーチンの主要部は、ユーザーによって書かなければならない。

FrontISTRは以下のユーザサブルーチンインターフェースを提供している。

### 1.9.1 ユーザ定義材料の入力

ユーザー定義材料を使用する場合、最大100のユーザー定義材料定数が使用可能である。材料定数の入力は以下のように、制御データファイル内の1行10数値、最大10行まで入力可能である。

2行目～最大10行目

v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

### 1.9.2 弾塑性変形に関わるサブルーチン (**uyield.f90**)

弾塑性剛性マトリクスおよび応力のreturn mappingを計算するためのサブルーチンを提供している。ユーザー定義降伏関数を利用する場合、まず入力ファイルに!PLASTIC, TYPE=USERを設定して必要な材料定数を入力し、次にサブルーチンElastoPlasticMatrixおよびBackwardEulerを作成する必要がある。

#### 1.9.2.1 (1) 弾塑性剛性マトリクスの計算サブルーチン

```

subroutine uElastoPlasticMatrix( matl, stress, istat, fstat, D )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(IN) :: stress(6)
    INTEGER, INTENT(IN) :: istat
    REAL(KIND=kreal), INTENT(IN) :: fstat(:)
    REAL(KIND=kreal), INTENT(OUT) :: D(:, :)

```

- **matl:** 材料定数を保存する配列（最大100）
  - **stress:** 2nd Piola-Kirchhoff応力
  - **istat:** 降伏状態(0: 未降伏 ; 1: 降伏した)
  - **fstat:** 状態変数。fstat(1)=塑性ひずみ、fstat(2:7)= back stress(移動または複合硬化時)
  - **D:** 弹塑性マトリクス

#### 1.9.2.2 (2) 応力のReturn mapping計算サブルーチン

```

subroutine uBackwardEuler ( matl, stress, istat, fstat )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(INOUT) :: stress(6)
    INTEGER, INTENT(INOUT) :: istat
    REAL(KIND=kreal), INTENT(IN) :: fstat(:)

```

- matl: 材料定数を保存する配列（最大100）
  - stress: trial stress 弹性変形を仮定し得られた2nd Piola-Kirchhoff応力
  - istat: 降伏状態(0: 未降伏 ; 1: 降伏した)
  - fstat: 状態変数。fstat(1)=塑性ひずみ、fstat(2:7)= back
  - stress(移動または複合硬化時)

### 1.9.3 弹性変形に関するサブルーチン (**uelastic.f90**)

弾性および超弾性問題の弾性剛性マトリクスおよび応力の更新計算をするためのサブルーチンを提供している。ユーザー弾性または超弾性構成式を利用する場合、まず入力ファイルに!ELASTIC, TYPE=USERまたは!HYPERELASTIC, TYPE=USERを設定して必要な材料定数を入力し、次にサブルーチンuElasticMatrixおよびuElasticUpdateを作成する必要がある。

#### 1.9.3.1 (1) 弹性剛性マトリクスの計算サブルーチン

```
subroutine uElasticMatrix( matl, strain, D )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: strain(6)
  REAL(KIND=kreal), INTENT(OUT) :: D(6,6)
```

- matl: 材料定数を保存する配列（最大100）
  - strain: Green-Lagrangeひずみ
  - D: 弹性マトリクス

#### 1.9.3.2 (2) 応力の計算サブルーチン

```
subroutine uElasticUpdate ( matl, strain, stress )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: strain(6)
  REAL(KIND=kreal), INTENT(OUT) :: stress(6)
```

- matl: 材料定数を保存する配列（最大100）
  - strain: Green-Lagrangeひずみ
  - stress: 応力

### 1.9.4 ユーザー定義材料に関するサブルーチン (**umat.f**)

弾性、超弾性、弾塑性材に拘らず一般的な材料の変形解析のインターフェースを提供する。

#### 1.9.4.1 (1) 剛性マトリクスの計算サブルーチン

```
subroutine uMatlMatrix( mname, matl, ftn, stress, fstat, D, temperature, dtime )
  CHARACTER(len=*) , INTENT(IN) :: mname
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: ftn(3,3)
  REAL(KIND=kreal), INTENT(IN) :: stress(6)
  REAL(KIND=kreal), INTENT(IN) :: fstat(:)
  REAL(KIND=kreal), INTENT(OUT) :: D(:, :)
  REAL(KIND=kreal), optional :: temperature
  REAL(KIND=kreal), optional :: dtime
```

- mname: 材料名
  - matl: 材料定数を保存する配列（最大100）
  - ftn: 変形勾配テンソル
  - stress: 2nd Piola-Kirchhoff応力
  - fstat: 状態変数
  - D: 構成式
  - temperature: 温度
  - dtime: 時間増分

#### 1.9.4.2 (2) ひずみおよび応力の更新計算サブルーチン

```
subroutine uUpdate( mname, matl, ftn, strain, stress, fstat, temperature, dtime )
    character(len=*) , intent(in) :: mname
    real(KIND=kreal), intent(in) :: matl
    real(kind=kreal), intent(in)   :: ftn(3,3)
    real(kind=kreal), intent(inout) :: strain(6)
    real(kind=kreal), intent(inout) :: stress(6)
    real(kind=kreal), intent(inout) :: fstat(:)
    real(KIND=kreal), optional :: temperature
    real(KIND=kreal), optional :: dtime
```

- **mname:** 材料名
  - **matl:** 材料定数を保存する配列（最大100）
  - **ftn:** 変形勾配テンソル
  - **strain:** ひずみ
  - **stress:** 2nd Piola-Kirchhoff応力
  - **fstat:** 状態変数
  - **temperature:** 温度
  - **dtime:** 時間増分

#### 1.9.5 ユーザー定義外部荷重の処理サブルーチン (**uLoad.f**)

ユーザー定義外部荷重を処理するインターフェースを提供する。

ユーザー定義外部荷重を利用するため、まず外部荷重を定義するための数値構造tULoadを定義し、入力ファイルの!ULOADを利用してその定義を読み込む。その後、以下のインターフェースを利用して、外部荷重を組み込む。

##### 1.9.5.1 (1) 外部荷重の読み込みサブルーチン

```
integer function ureadload( fname )
    character(len=*) , intent(in) :: fname
```

- **fname:** 外部ファイル名。このファイルからユーザー定義外部荷重を読み込む。

##### 1.9.5.2 (2) 外部荷重を全体荷重ベクトルへ組み込むサブルーチン

```
subroutine uloading( cstep, factor, exForce )
    integer, INTENT(IN) :: cstep
    REAL(KIND=kreal), INTENT(IN) :: factor
    REAL(KIND=kreal), INTENT(INOUT) :: exForce(:)
```

- **cstep:** 現時点の解析ステップ数
  - **factor:** 現ステップの荷重係数
  - **exForce:** 全体荷重ベクトル

##### 1.9.5.3 (3) 残差応力の計算サブルーチン

```
subroutine uResidual( cstep, factor, residual )
    integer, INTENT(IN) :: cstep
    REAL(KIND=kreal), INTENT(IN) :: factor
    REAL(KIND=kreal), INTENT(INOUT) :: residual(:)
```

- **cstep:** 現時点の解析ステップ数
  - **factor:** 現ステップの荷重係数
  - **residual:** 全体残差力ベクトル

## 1.10 ユーザーサブルーチン

ユーザーがFrontISTRの機能をプログラミングにより拡張するためのインターフェースを提供する。これらのインターフェースは、基本的にサブルーチンヘッダを含むFORTRANサブルーチンで、入出力変数の記述とこれらの変数のための宣言文である。ルーチンの主要部は、ユーザーによって書かなければならない。

FrontISTRは以下のユーザサブルーチンインターフェースを提供している。

### 1.10.1 ユーザー定義材料の入力

ユーザー定義材料を使用する場合、最大100のユーザー定義材料定数が使用可能である。材料定数の入力は以下のように、制御データファイル内の1行10数値、最大10行まで入力可能である。

2行目～最大10行目

v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

### 1.10.2 弹塑性変形に関わるサブルーチン (**uyield.f90**)

弾塑性剛性マトリクスおよび応力のreturn mappingを計算するためのサブルーチンを提供している。ユーザー定義降伏関数を利用する場合、まず入力ファイルに!PLASTIC, TYPE=USERを設定して必要な材料定数を入力し、次にサブルーチンElastoPlasticMatrixおよびBackwardEulerを作成する必要がある。

#### 1.10.2.1 (1) 弹塑性剛性マトリクスの計算サブルーチン

```
subroutine uElastoPlasticMatrix( matl, stress, istat, fstat, D )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(IN) :: stress(6)
    INTEGER, INTENT(IN) :: istat
    REAL(KIND=kreal), INTENT(IN) :: fstat(:)
    REAL(KIND=kreal), INTENT(OUT) :: D(:, :)
```

- **matl:** 材料定数を保存する配列（最大100）
  - **stress:** 2nd Piola-Kirchhoff応力
  - **istat:** 降伏状態(0: 未降伏； 1: 降伏した)
  - **fstat:** 状態変数. **fstat(1)=**塑性ひずみ、**fstat(2:7)=** back stress(移動または複合硬化時)
  - **D:** 弹塑性マトリクス

#### 1.10.2.2 (2) 応力のReturn mapping計算サブルーチン

```
subroutine uBackwardEuler ( matl, stress, istat, fstat )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(INOUT) :: stress(6)
    INTEGER, INTENT(INOUT) :: istat
    REAL(KIND=kreal), INTENT(IN) :: fstat(:)
```

- **matl:** 材料定数を保存する配列（最大100）
  - **stress:** trial stress弹性変形を仮定し得られた2nd Piola-Kirchhoff応力
  - **istat:** 降伏状態(0: 未降伏； 1: 降伏した)
  - **fstat:** 状態変数. **fstat(1)=**塑性ひずみ、**fstat(2:7)=** back stress(移動または複合硬化時)

### 1.10.3 弹性変形に関わるサブルーチン (**uelastic.f90**)

弾性および超弾性問題の弾性剛性マトリクスおよび応力の更新計算をするためのサブルーチンを提供している。ユーザー弾性または超弾性構成式を利用する場合、まず入力ファイルに!ELASTIC, TYPE=USERまたは!HYPERELASTIC, TYPE=USERを設定して必要な材料定数を入力し、

次にサブルーチンElasticMatrixおよびElasticUpdateを作成する必要がある。

### 1.10.3.1 (1) 弹性剛性マトリクスの計算サブルーチン

```
subroutine uElasticMatrix( matl, strain, D )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(IN) :: strain(6)
    REAL(KIND=kreal), INTENT(OUT) :: D(6,6)
```

- matl: 材料定数を保存する配列（最大100）
  - strain: Green-Lagrangeひずみ
  - D: 弹性マトリクス

### 1.10.3.2 (2) 応力の計算サブルーチン

```
subroutine uElasticUpdate ( matl, strain, stress )
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(IN) :: strain(6)
    REAL(KIND=kreal), INTENT(OUT) :: stress(6)
```

- matl: 材料定数を保存する配列（最大100）
  - strain: Green-Lagrangeひずみ
  - stress: 応力

## 1.10.4 ユーザー定義材料に関わるサブルーチン (umat.f)

弾性、超弾性、弾塑性材に拘らず一般的な材料の変形解析のインターフェースを提供する。

### 1.10.4.1 (1) 剛性マトリクスの計算サブルーチン

```
subroutine uMatlMatrix( mname, matl, ftn, stress, fstat, D, temperature, dtime )
    CHARACTER(len=*) , INTENT(IN) :: mname
    REAL(KIND=kreal), INTENT(IN) :: matl(:)
    REAL(KIND=kreal), INTENT(IN) :: ftn(3,3)
    REAL(KIND=kreal), INTENT(IN) :: stress(6)
    REAL(KIND=kreal), INTENT(IN) :: fstat(:)
    REAL(KIND=kreal), INTENT(OUT) :: D(:, :)
    REAL(KIND=kreal), OPTIONAL :: temperature
    REAL(KIND=kreal), OPTIONAL :: dtime
```

- mname: 材料名
  - matl: 材料定数を保存する配列（最大100）
  - ftn: 変形勾配テンソル
  - stress: 2nd Piola-Kirchhoff応力
  - fstat: 状態変数
  - D: 構成式
  - temperature: 温度
  - dtime: 時間増分

### 1.10.4.2 (2) ひずみおよび応力の更新計算サブルーチン

```
subroutine uUpdate( mname, matl, ftn, strain, stress, fstat, temperature, dtime )
    character(len=*) , intent(in) :: mname
    real(KIND=kreal), intent(in) :: matl
    real(kind=kreal), intent(in) :: ftn(3,3)
    real(kind=kreal), intent(inout) :: strain(6)
    real(kind=kreal), intent(inout) :: stress(6)
    real(kind=kreal), intent(inout) :: fstat(:)
    real(KIND=kreal), optional :: temperature
    real(KIND=kreal), optional :: dtime
```

- `mname`: 材料名
  - `matl`: 材料定数を保存する配列（最大100）
  - `ftn`: 変形勾配テンソル
  - `strain`: ひずみ
  - `stress`: 2nd Piola-Kirchhoff応力
  - `fstat`: 状態変数
  - `temperature`: 温度
  - `dtime`: 時間増分

### 1.10.5 ユーザー定義外部荷重の処理サブルーチン (`uLoad.f`)

ユーザー定義外部荷重を処理するインターフェースを提供する。

ユーザー定義外部荷重を利用するため、まず外部荷重を定義するための数値構造`tULoad`を定義し、入力ファイルの`!ULOAD`を利用してその定義を読み込む。その後、以下のインターフェースを利用して、外部荷重を組み込む。

#### 1.10.5.1 (1) 外部荷重の読み込みサブルーチン

```
integer function ureadload( fname )
  character(len=*) , intent(in) :: fname
```

- `fname`: 外部ファイル名。このファイルからユーザー定義外部荷重を読み込む。

#### 1.10.5.2 (2) 外部荷重を全体荷重ベクトルへ組み込むサブルーチン

```
subroutine uloading( cstep, factor, exForce )
  integer, INTENT(IN) :: cstep
  REAL(KIND=kreal), INTENT(IN) :: factor
  REAL(KIND=kreal), INTENT(INOUT) :: exForce(:)
```

- `cstep`: 現時点の解析ステップ数
  - `factor`: 現ステップの荷重係数
  - `exForce`: 全体荷重ベクトル

#### 1.10.5.3 (3) 残差応力の計算サブルーチン

```
subroutine uResidual( cstep, factor, residual )
  integer, INTENT(IN) :: cstep
  REAL(KIND=kreal), INTENT(IN) :: factor
  REAL(KIND=kreal), INTENT(INOUT) :: residual(:)
```

- `cstep`: 現時点の解析ステップ数
  - `factor`: 現ステップの荷重係数
  - `residual`: 全体残差力ベクトル

## 1.11 ステップ制御

### 1.11.1 解析上の時間について

ここではFrontISTRの解析上の時間について、以下の通り用語の定義を行う：

- 現時刻：解析初期からの総経過時間
- ステップ時刻：ステップ開始からの経過時間
- 時間幅：ステップで解析する時間
- 相対時刻：時間幅に対するステップ開始からの経過時間の割合
- 時間増分：現時刻からつり合いを求める時刻までの増分

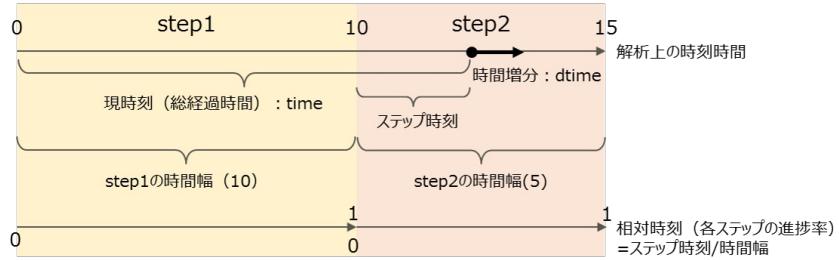


図4.7.1 解析上の時間に関する用語定義

### 1.11.2 静解析の制御

本開発コードにおいて、静解析は1つまたは複数の連続した解析ステップからなる。各解析ステップごとに基本境界条件の組を1つ与え、これを前述の増分解法によって解いていく。以下では解析ステップの反復をステップループ、増分解法の反復をサブステップループと呼ぶ。静解析の増分制御は、以下のいずれかから選択することができる。

- 固定増分による計算。収束に失敗した場合は直ちに計算を終了する。
- 自動増分・カットバックによる計算。収束状況に応じて増分量を変化させ、収束に失敗した場合には増分を小さくして再計算を行う。

#### 1.11.2.1 自動増分・カットバックの概要

自動増分・カットバックによる計算のフローは図4.7.2の通りである。

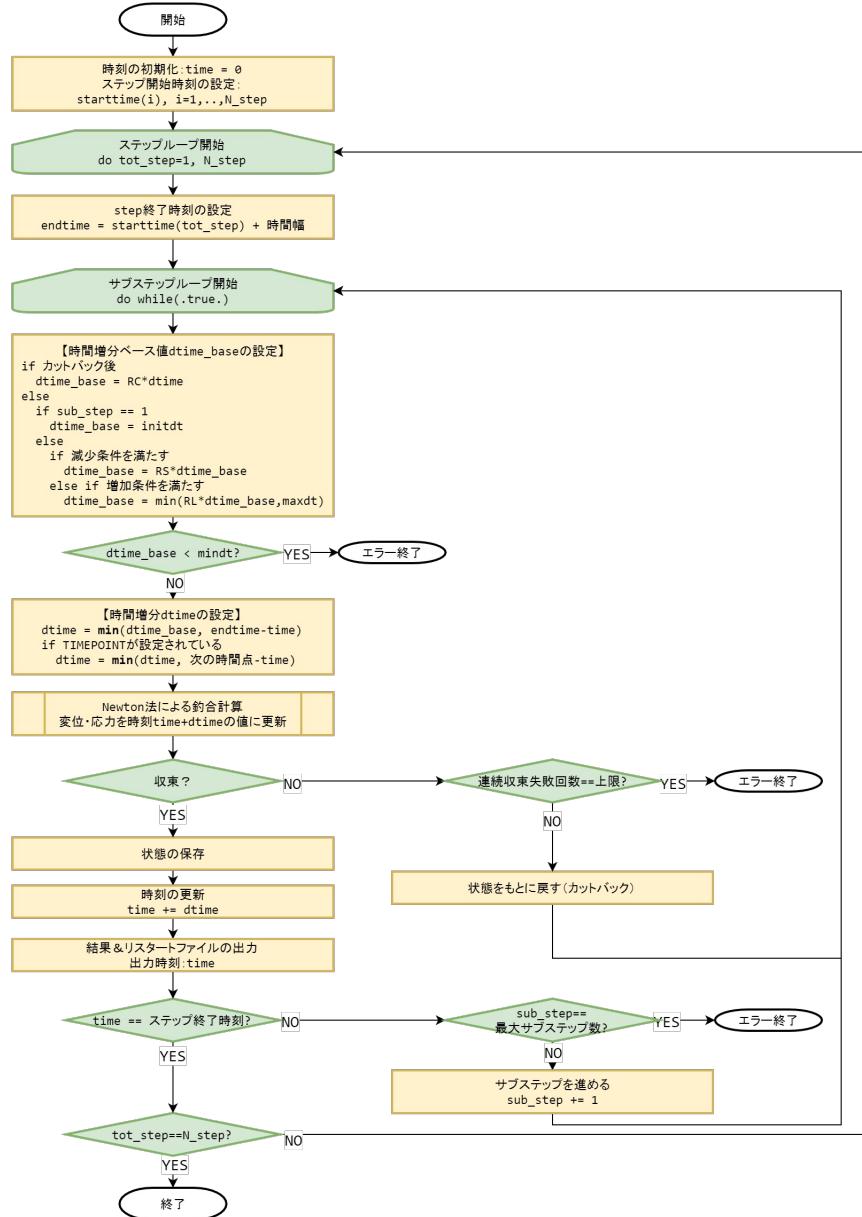


図4.7.2 増分割御フロー

フローの骨格は次の通りである。

1. ステップ1から最終ステップN\_stepまで、下記2.以降の手続きを繰り返す：
2. 時間増分の基準値dtime\_baseを、現在のdtime\_baseと前サブステップでの収束状況から定める。初回は初期時間増分initdtを用いる。
3. 実際の時間増分dtimeを、ステップ終了または直近の出力指定時刻までの残り時間とdtime\_baseの小さい方で定める。
4. 時刻time+dtimeでのつり合い計算を試みる
5. 収束に成功した場合は時刻をdtimeだけ進め、失敗した場合は時刻timeの状態を復元して2.に戻る。
6. timeがステップ終了時刻に到達したらステップを終了する

解析途中で下記に挙げるケースに該当した場合、非線形静解析の手続きは失敗と判断され、エ

ラー終了する。

- timeがステップ終了時刻に到達する前にサブステップ数が上限に到達した場合
- 時間増分の基準値dtime\_baseが時間増分下限mindtを下回った場合
- 指定されたN\_C回連続して収束に失敗した場合

### 1.11.2.2 時間増分基準値dtime\_baseの調整

ステップ初回のdtime\_baseは、指定された初期時間増分initdtの値に設定される。それ以外の場合は、前のサブステップの収束状況に応じて次の通り設定される。

1. 収束に失敗した場合（カットバックされた場合）...dtime\_baseにカットバック縮小率R\_Cを乗じた値
2. 収束に成功した場合
  1. 減少条件に該当する場合：dtime\_baseに減少率R\_Sを乗じた値
  2. 減少条件に該当せず、増加条件に該当する場合：dtime\_baseに増加率R\_Lを乗じた値と、時間増分上限maxdtの小さい方
  3. 減少条件にも増加条件にも該当しない場合：dtime\_baseは変化しない

### 1.11.2.3 増加・減少条件

自動増分調整機能では、増分を増加・減少させる条件を以下の変数を用いて判定する：

- N\_max：前サブステップにおけるNewton法反復回数の最大値
- N\_sum：前サブステップにおけるNewton法反復回数の合計値（接触反復が無い場合はN\_maxに一致）
- N\_cont：前サブステップにおける接触反復回数

減少条件は以下の両方が満たされるときである：

- N\_max, N\_sum, N\_contの「いずれか一つ」が、各々の閾値NS\_max, NS\_sum, NS\_contを上回る
- 上記の状態が、N\_S回以上連続したサブステップで満たされる

増加条件は以下の両方が満たされるときである：

- N\_max, N\_sum, N\_contの「すべて」が、各々の閾値NL\_max, NL\_sum, NL\_cont以内である
- 上記の状態が、N\_L回以上連続したサブステップで満たされる

### 1.11.2.4 計算および出力時刻の指定

自動時間増分は収束状況によって増分が変化するため、どの時刻につりあい計算および結果出力が行われるか事前に決定できない。これが不便である場合に、出力時刻のリストを与えることによって、希望する時刻におけるつり合い計算および結果出力を実行させることができる。出力時刻のリストが与えられたステップでは、指定された時刻にて必ず計算が行われるように、時間増分dtimeの値が調整される。

### 1.11.2.5 時間増分の使用方法

本機能に関する設定はすべて解析制御ファイルで行う。自動増分調整・カットバック機能は!STEPカードでTYPE\_INC=AUTOを指定することで有効になる。時間増分の調整関連のパラメータは、!AUTOINC\_PARAMを定義の上、!STEP, AUTOINCPARAMパラメータで各ステップごとに指定する。指定がない場合は!AUTOINC\_PARAMのデフォルトパラメータが使用される。出力時刻の指定は、!TIME\_POINTSで時刻リストの定義の上、!STEP, TIMEPOINTSパラメータで各ステップごとに行う。

#### 1.11.2.5.1 使用例

自動増分調整を有効にし、初期時間増分0.01、ステップ時間幅2.5、時間増分下限1E-5、時間増分上限0.3、最大サブステップ数を200に設定する。自動増分のパラメータセットはAP1（セット名）、出力時刻1.5, 2.7, 3.9を指定する。

```
!AUTOINC_PARAM, NAME=AP1
0.25, 10, 50, 10, 1
1.25, 1, 1, 1, 2
0.25, 5
!TIME_POINTS, TIME=TOTAL, TIME=,NAME=TP1
1.5
2.7
3.9
!STEP, INC_TYPE=AUTO, SUBSTEPS=200, AUTOINCPARAM=AP1, TIMEPOINTS=TP1
0.01, 2.5, 1E-5, 0.3
```

### 1.11.3 動解析の制御

本開発コードにおいて、動解析直接時間積分法は1つの解析ステップからなる。また動解析の増分制御は固定増分のみであり、自動時間増分機能を使用することはできない。