第32回 FrontISTR研究会

FUJITSU

shaping tomorrow with you

FrontISTR接触解析プログラム解説（カスタマイズ編）

富士通アドバンストテクノロジ株式会社
構造技術統括部
稲垣和久

2016年11月28日

本日の講演内容

- FrontISTRの接触解析について、接触構造体や
  接触探索等を中心に、過去分と重複しない範囲で解説を行います
- 関連個所のカスタマイズ例を示します

概要

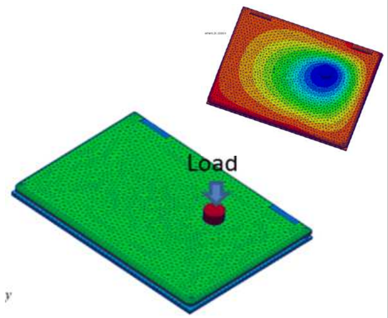
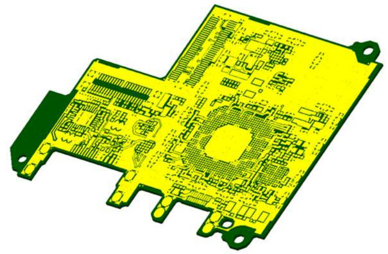
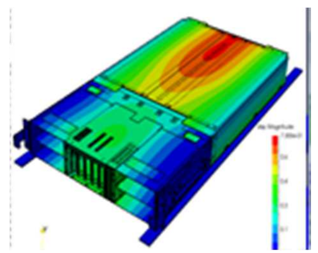
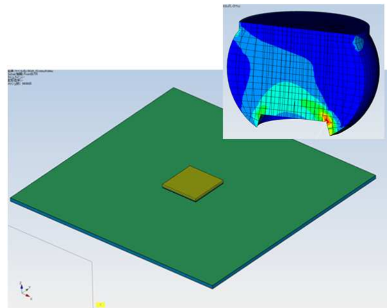
- はじめに
- FrontISTR接触解析の概要
- 接触解析プログラムの解説
  - 接触構造体と接触探索を中心に
- カスタマイズ実施例
  - ログ出力
  - 表面ベースの結合拘束
  - 接触アルゴリズムの新設（内点法接触の紹介）

はじめに

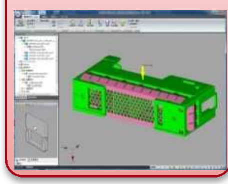
FUJITSU

■ 少しだけ宣伝

- 弊社では、FrontISTRに電子機器向け各種機能追加を行って活用しています
  - 特に接触解析を重点的に改良
- 弊社カスタマイズ版FrontISTR「**FS-Solver**」＋独自統合解析環境「**FS-Hub**」を提供中です
  - どうぞよろしくお願い致します。
- FrontISTR本体開発も行っています
  - 回転・トルク荷重機能の開発
  - 自動増分・カットバック機能開発（予定）

静解析 ：機械的荷重	**静解析 ：熱荷重**
 ノートPC圧迫・他 （2014 OpenCAE学会）	 プリント基板反り （2013計算工学講演会）
 サーバ自重反り （約1,000万節点）	 はんだ接合部疲労寿命 （2016計算力学講演会）

FS-Hub (Pre)

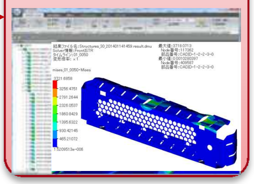


→ **FS-Solver (Solver)**

＋カスタム部

Front ISTR

→ **FS-Hub (Post)**



FrontISTR接触解析の概要

基礎方程式（静解析）

FUJITSU

- 連続体の釣合方程式を…

$$\mathrm{div}\boldsymbol{\sigma} + \rho\boldsymbol{g} = \boldsymbol{0} \quad \text{on} \quad \Omega^{(k)}, \quad \text{\# 釣合方程式}$$

$$\boldsymbol{u} = \boldsymbol{u}_0 \quad \text{on} \quad \Gamma_B^{(k)}, \quad \text{\# 変位境界条件}$$

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{t} \quad \text{on} \quad \Gamma_t^{(k)}, \quad \text{\# 荷重境界条件}$$

- 接触制約のもとで解く
  - 接触境界条件

$$\boldsymbol{\sigma} \cdot \boldsymbol{n} = \boldsymbol{t}_c \equiv t_N\boldsymbol{n} + \boldsymbol{t}_T \quad \text{on} \quad \Gamma_c^{(k)},$$

  - 非貫入制約（KKT条件）

$$g_N \geq 0, \quad \text{\# 接触ギャップ}$$

$$t_N \leq 0, \quad \text{\# 法線方向接触圧}$$

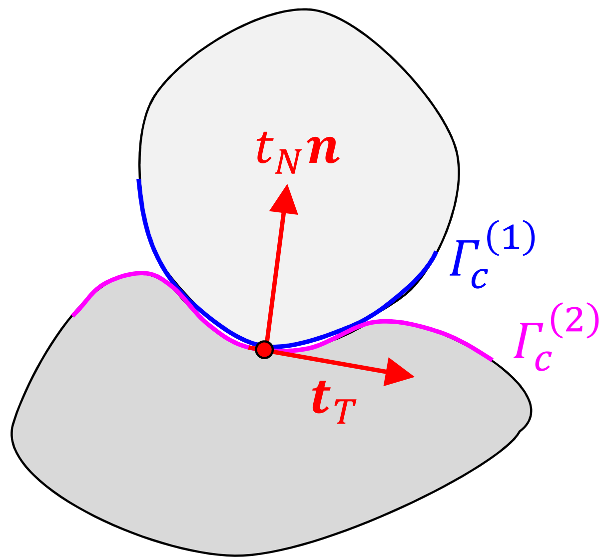
$$g_N t_N = 0,$$

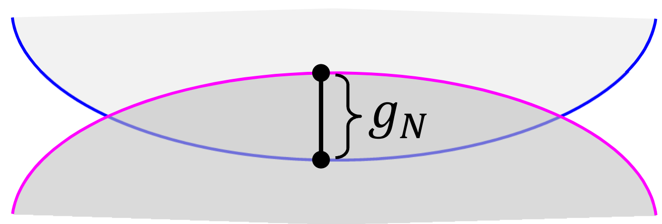
  - 摩擦制約（以降では省略）

$$\dot{\zeta} \geq 0, \quad \text{\# 滑り速度}$$

$$\Phi(t_N, \boldsymbol{t}_T) \leq 0, \quad \text{\# すべり限界関数}$$

$$\dot{\zeta}\Phi = 0.$$





g_N：接触ギャップ（貫入時は負値）

- ギャップが0で接触圧が発生(>0)
- ギャップが存在(>0)して接触圧が0

のいずれかであることを意味している

基礎方程式の離散化

FUJITSU

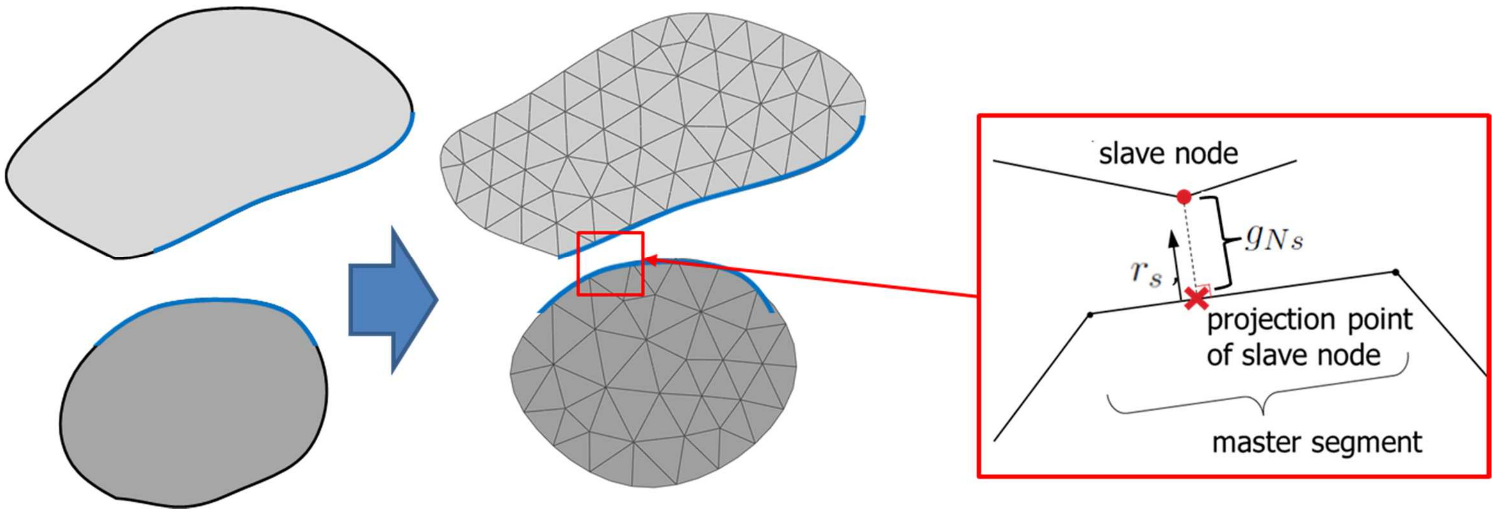
■ 有限要素法による離散化

■ 仮想仕事式の導出

$$\sum_{k=1}^{2}\int_{\Omega^{(k)}}\boldsymbol{\sigma}:\delta\boldsymbol{D}\,\mathrm{d}v=\sum_{k=1}^{2}\int_{\Omega^{(k)}}\rho\boldsymbol{g}:\delta\boldsymbol{u}\,\mathrm{d}v+\sum_{k=1}^{2}\int_{\Gamma_t^{(k)}}\boldsymbol{t}\cdot\delta\boldsymbol{u}\,\mathrm{d}s+\boxed{\sum_{k=1}^{2}\int_{\Gamma_c^{(k)}}\boldsymbol{t}_c\cdot\delta\boldsymbol{u}\,\mathrm{d}s}$$

接触の寄与

- ■ 連続体部分は、接触無しの場合と同様に有限要素離散化
- ■ 接触の離散化…FrontISTRでは**Node to Surface**の離散化を行う
  - 接触面対をマスター面、スレーブ面に区分
  - スレーブ面の節点が、各マスター要素面に貫入しない

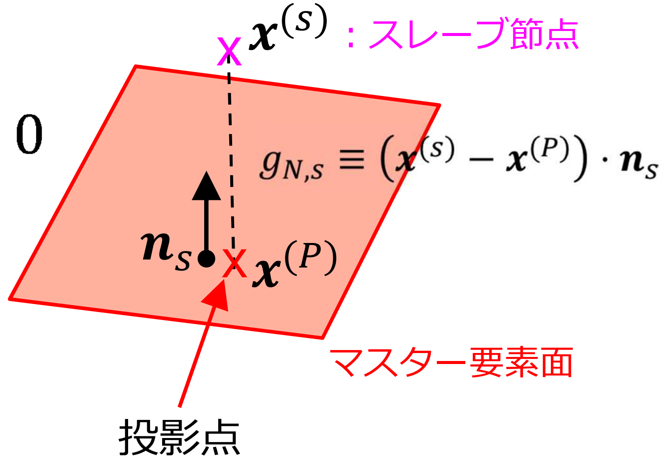


Node to Surface接触

■ 離散化された接触条件

- ■ 接触力、接触制約…スレーブ節点ごとに１つ

$$g_{N,s} \geq 0, t_{N,s} \leq 0, g_{N,s} t_{N,s} = 0$$



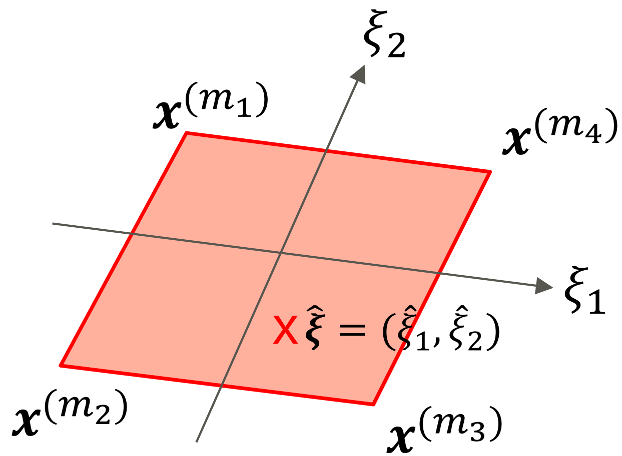
- ■ 接触ギャップ

$$g_{N,s} \equiv \left(\boldsymbol{x}^{(s)} - \boldsymbol{x}^{(P)}\right) \cdot \boldsymbol{n}_s$$

  - $\boldsymbol{x}^{(s)}$：スレーブ節点座標
  - $\boldsymbol{x}^{(P)}$：マスター投影点座標
  - $\boldsymbol{n}_s$：法線方向単位ベクトル

- ■ マスター投影点座標

$$\boldsymbol{x}^{(P)} = \sum_{j=1}^{n_e} N_j(\hat{\boldsymbol{\xi}}) \boldsymbol{x}^{(m_j)}$$



  - $\hat{\boldsymbol{\xi}}$：投影点の要素面内座標
  - N_j：要素面の形状関数（重み）
  - $\boldsymbol{x}^{(m_j)}$：要素面構成節点座標

- ■ 法線方向…$\boldsymbol{x}^{(P)}$の接ベクトルと直交する方向を計算（詳細割愛）

非貫入条件の処理

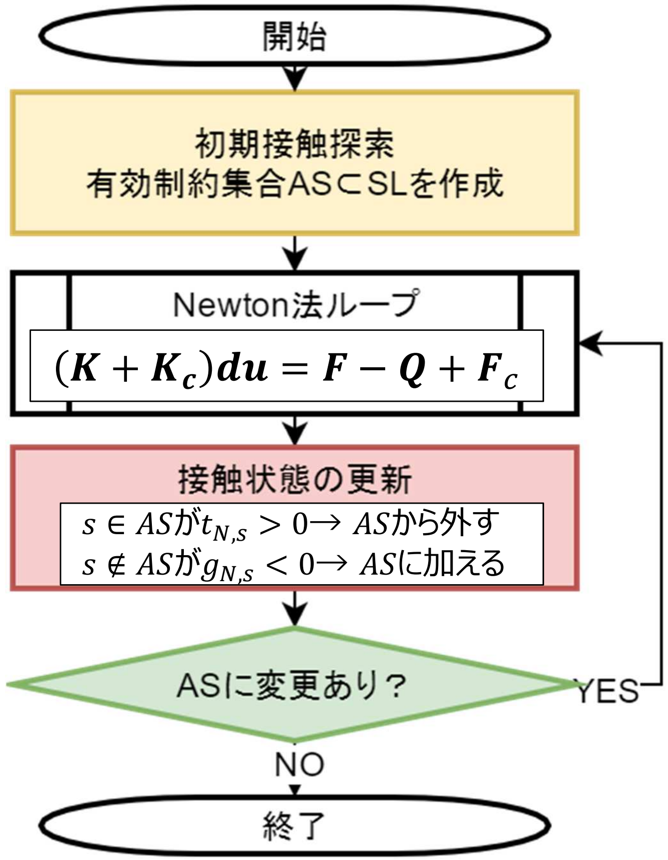
- 最終的に解くべき式は次の通り

$$\boldsymbol{Q} - \boldsymbol{F} - \boldsymbol{F}_c = \boldsymbol{0},$$ # 接触力を含む釣合方程式

$$\boldsymbol{u} = \boldsymbol{u}_0 \quad \text{on} \quad \Gamma_B^{(\mathrm{k})},$$ # 変位境界条件

$$g_{N,s} \geq 0, t_{N,s} \leq 0, g_{N,s}t_{N,s} = 0. (s \in SL)$$ # 非貫入制約

  - $\boldsymbol{Q}$ ：内力、 $\boldsymbol{F}$ ：外力、$\boldsymbol{F}_c$ ：接触力
  - SL ：スレーブ節点集合
- 非貫入制約の処理
  - FrontISTRでは**有効制約法**によって等式制約問題に帰着する
    - 貫入ギャップが0となるべきスレーブ節点集合 $AS \subset SL$を、試行錯誤的に決定する
    - 各試行の中では等式制約問題。FrontISTRでは**Lagrange乗数法**または**拡張Lagrange乗数法**で解く
  - 制約境界付近で発散するバリア関数を用い、不等式制約問題を直接解く方法もある



接触解析まとめ

FUJITSU

- 基礎方程式
  - 釣合方程式
  - 非貫入制約
  - 摩擦条件
- 有限要素離散化
  - 仮想仕事式
  - 接触の離散化(Node to Surface)
- 解法
  - 接触状態の決定（有効制約法）
  - 等式制約条件付きNewton法（Lagrange乗数法）
    - 接触剛性
    - 接触節点力
  - 線形方程式の解法
    - 直接法
    - 反復法
- プログラム
  - データ
    - 接触情報 tContact
      - 接触状態 tContactState
      - 面情報 tSurfaceElement
    - 接触剛性行列 hecMAT,conMAT
  - フロー
    - メインループ
    - 接触判定
      - 探索メインループ
      - 投影点探索
    - 接触剛性
    - 接触力
    - 行列プロファイル更新
  - 並列化

赤字…本資料発表部分
それ以外の部分は下記の過去資料を参照ください
資料1: FrontISTRによる接触解析の理論とプログラムの解説 (ver.4.4), 第21回FrontISTRユーザ会
資料2: FrontISTRによる並列接触解析のプログラム解説(ver.4.4), 第23回FrontISTRユーザ会
資料3: 接触問題向け反復法線形ソルバーの解説と適用事例, 第21回FrontISTRユーザ会

接触解析プログラムの解説

- 接触構造体と接触探索を中心に

接触構造体

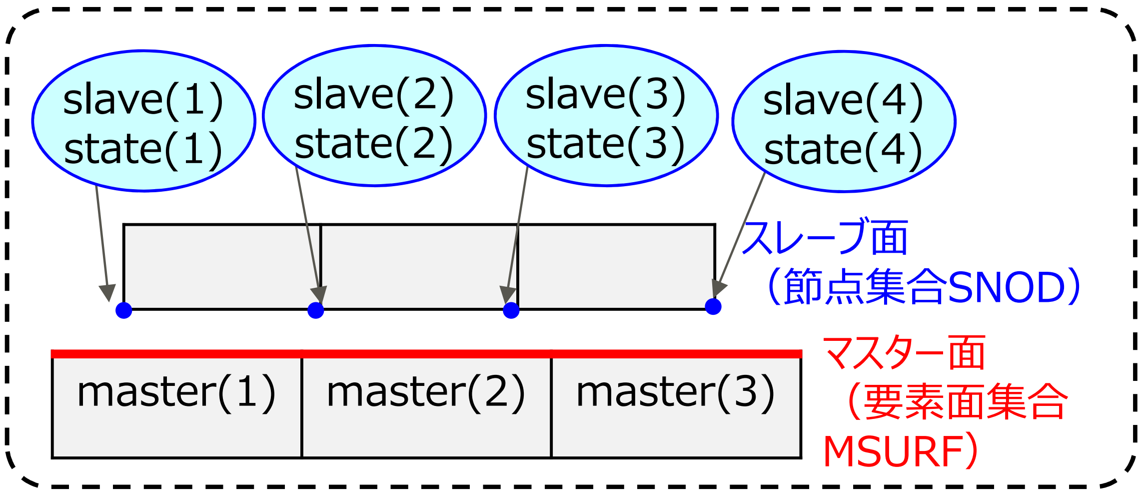
■ 接触ペアの構造体：tContact

- !CONTACT 1つにつき1つ
- 定義：lib/contact/fstr_contact_def.f90
- 主な変数
  - スレーブ節点番号　：integer　:: slaves(:)
  - スレーブ節点情報　：type(tContactState)　:: states(:)
  - マスター面構成情報　：type(tSurfEelemt)　:: master(:)
  - 接触特性情報　：ctype（接触タイプ），fcoeff（摩擦係数）など

【メッシュファイル】
!NGROUP, NGRP=**SNOD**
!SGROUP, SGRP=**MSURF**
!CONTACT PAIR, NAME=CP1
SNOD, **MSURF**

【解析制御ファイル】
!CONTACT
CP1, 0.2

接触面対 tContact



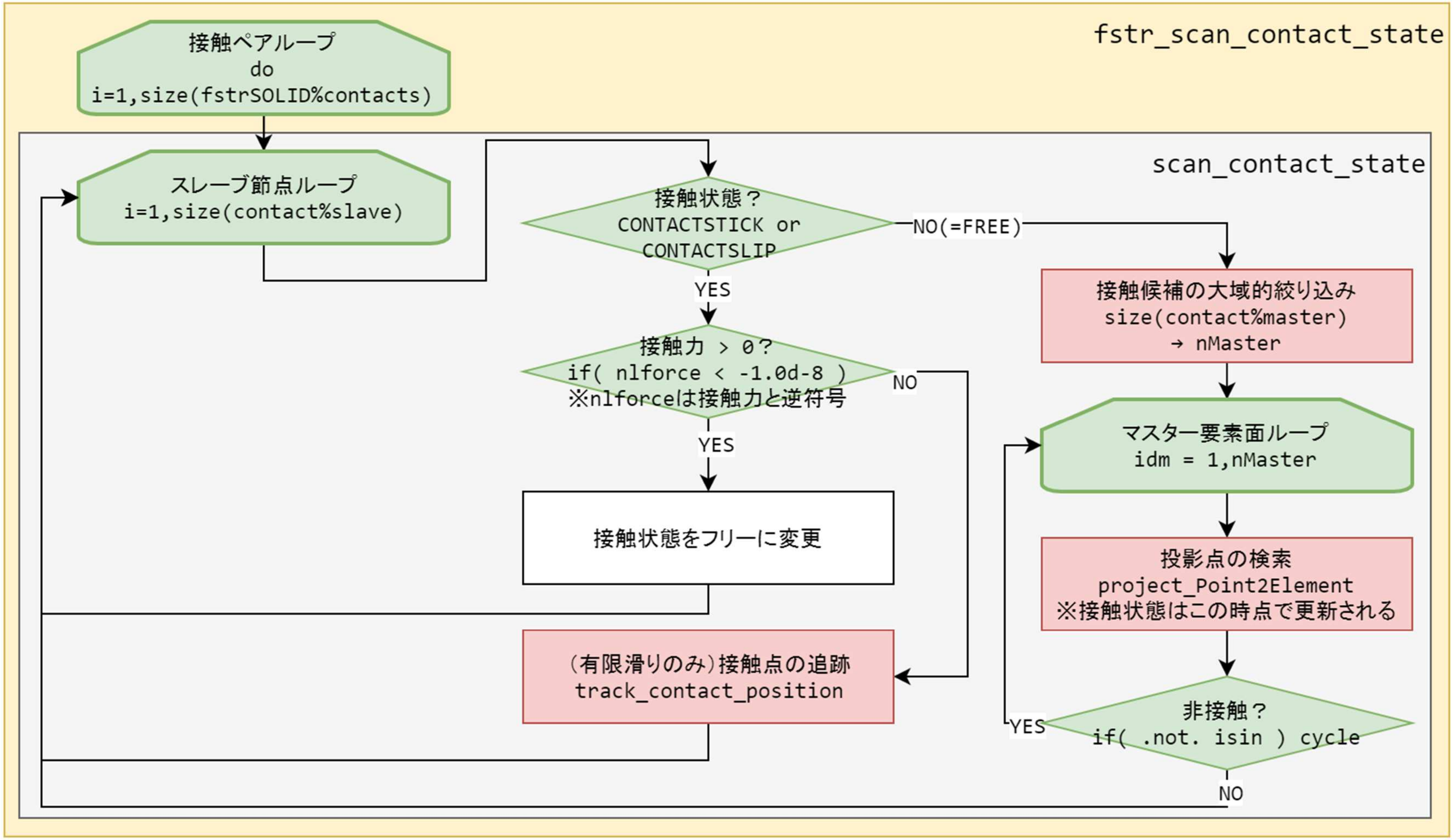
接触構造体

- fstr_info_contactChange
  - 接触状態の変化に関する統計情報
  - 定義：lib/contact/fstr_contact_def.f90
  - 主な変数（すべてint）
    - contact2free ： 接触からフリーに変化した数
    - contact2neighbor ： （有限滑りのみ）隣接要素に移動した数
    - contact2diffLpos ： （有限滑りのみ）要素内で閾値以上に変位した数
    - free2contact ： フリーから接触に変化した数
    - contactNode_previous ： 前反復での接触数
    - contactNode_current ： 現在の接触数

接触探索

■ fstr_scan_contact_state：接触探索のメイン関数

- 定義：
  - common/fstr_contact.f90
  - lib/contact/fstr_contact_def.f90（ scan_contact_state ）



接触解析のカスタマイズ例

【例1】接触状態のモニタリング

FUJITSU

■ 接触状態の統計情報（時系列出力）

- デフォルトの接触ログは、個別の接触状態の変化が全部出力される
- 接触点数が多いときは、統計情報がほしい
- fstr_info_contactChangeの内容を毎時出力するだけ
  - 一方で、scan_contact_stateにあるスレーブ節点ごとの出力は削除してもよい

```
subroutine print_analysis_status( infoCTChange )
  type (fstr_info_contactChange),intent(in) :: infoCTChange

  logical, save :: header_flag = .false.
  integer(kind=kint) :: c2f, f2c, cnt, mg, ml

  c2f = infoCTChange%contact2free
  f2c = infoCTChange%free2contact
  cnt = infoCTChange%contactNode_current
  mg = infoCTChange%contact2neighbor
  ml = infoCTChange%contact2difflpos

  write(IDBG,'(9I8)') cstep, sub_step, c_iter, n_iter, f2c, c2f, cnt, mg, ml

end subroutine
```

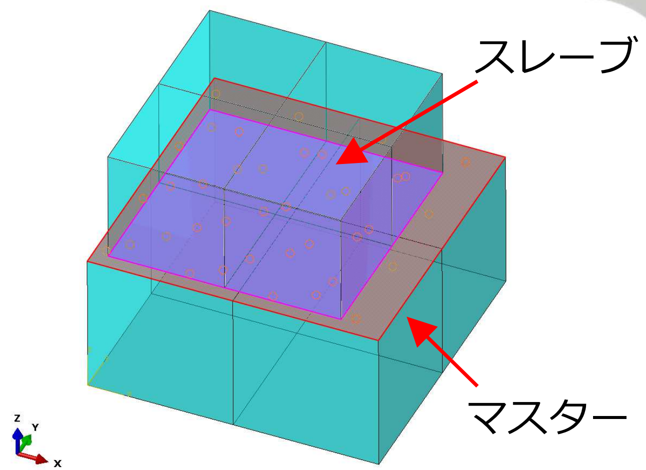
【例2】表面ベースの結合拘束

FUJITSU

■ 対向する表面同士の結合

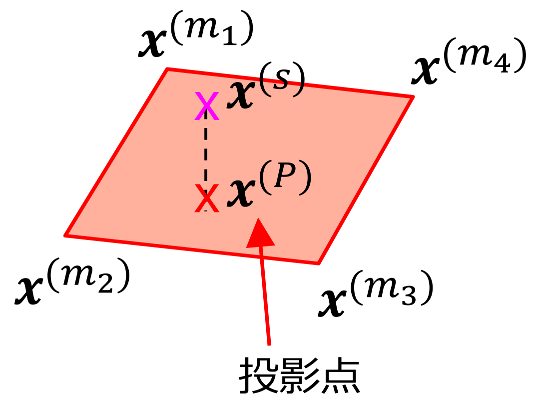
■ どんな機能か

- 対向する２つの面（マスター面とスレーブ面）を与えるとき
  スレーブ面内の各節点について、その変位が
  マスター面投影点の変位と一致するよう拘束する（MPC）
  $\boldsymbol{u}^{(s)} = \boldsymbol{u}^{(P)} = N_1\boldsymbol{u}^{(m_1)} + N_2\boldsymbol{u}^{(m_2)} + \cdots + N_k\boldsymbol{u}^{(m_k)}$
  N_i ：補間関数@投影点



■ FrontISTRは!EQUATIONカードで
スレーブ節点の各自由度ごとに
MPCの係数N_iを入力する必要がある（大変）

!EQUATION
$-1, N_1,\ N_2,\ \cdots, N_k$
…



- 接触と同じような入力で、面対の結合がしたい
  （係数の算出などはソルバ内部で実行してほしい）

【例2】表面ベースの結合拘束

FUJITSU

- 接触解析を次のように変えればOK
  - 接触タイプのフラグを増やして、通常の接触と分岐すると良い
    - mContactDefの接触タイプCONTACTTIEDを使う

```
!CONTACT, TYPE=SSLID
⇒!CONTACT, TYPE=TIED
```

  - 接触探索…初回だけ実行、接触状態は変えない
  - 拘束式…法線方向拘束を、全自由度の相対変位拘束に変える
    - 接触の拘束式：$g_{N,s} \equiv \left(\boldsymbol{x}^{(s)} - \boldsymbol{x}^{(P)}\right) \cdot \boldsymbol{n}_s = 0$
    - 結合の拘束式：$\boldsymbol{u}^{(s)} - \boldsymbol{u}^{(P)} = \boldsymbol{0}$

【例3】接触アルゴリズムの新設(内点法)

- 当日発表内容の詳細
  - INAGAKI, Kazuhisa, Gaku HASHIMOTO, and Hiroshi OKUDA. "Interior point method based contact analysis algorithm for structural analysis of electronic device models." Mechanical Engineering Journal (2015).

shaping tomorrow with you