

FrontISTR接触解析プログラムに ついての二、三の課題

第39回 FrontISTR研究会
2017年11月10日

はじめに

■ ベルトの接触解析に関連して、下記の話題提供をします

■ 接触関連の機能追加（v5.0a搭載予定）

1. 接触法線力、摩擦力、相対滑り量、接触状態の出力

■ 直近で見つかった接触（特に摩擦力）計算に関する課題 ※現在対応検討中

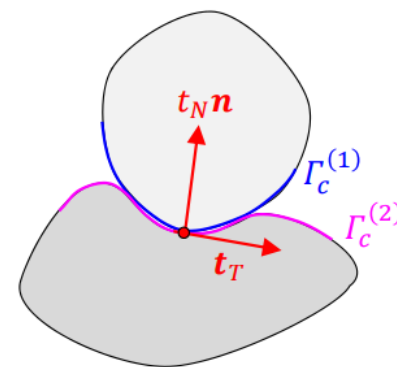
1. 接触面が大回転する場合における摩擦力計算の精度
2. 摩擦力の更新タイミング
3. 接触有り無しでの従動力の違い

接触関連の機能追加(v5.0 α 搭載予定)

機能追加

■ 下記の接触関連変数の出力を追加

- 接触法線力：節点における法線方向接触力 $t_N \mathbf{n}$ …※1
- 接触摩擦力：節点における摩擦力 $\mathbf{t}_T$ …※1
- 相対滑り速度：節点における相対滑り速度（スレーブのみ）
- 接触状態：フリー(-1)、未定義(0)、固着(1)、滑り(2)
のいずれか（スレーブのみ） …※2



※1 マスター面側は、各節点に分配された節点力の合計を出力

※2 節点が複数の接触ペアのスレーブである場合、固着or滑り優先で上書き

■ 解析制御ファイルの!OUTPUT_[RES/VIS]

!OUTPUT_VIS

CONTACT_NFORCE,ON

#接触法線力

CONTACT_FRICTION,ON

#接触摩擦力

CONTACT_RELVEL,ON

#相対滑り速度

CONTACT_STATE,ON

#接触状態

■ お試し版

■ GitLabのブランチ

kinagaki/contact_output
を取得下さい

■ 分散並列未対応の、プロト版です

- 接触法線力、接触摩擦力のマスター側出力以外は分散並列でも可です。

■ FrontISTRの摩擦力計算

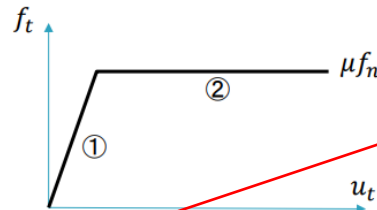
■ FrontISTRによる接触解析の理論とプログラムの解説 (ver.4.4)より

接触定式化(4/9)

▶ 摩擦状態

①固着状態 $\Delta f_t^{stick} = -\alpha_t \Delta u_t$

②滑り状態 $f_t^{slide} = \mu f_n t$



予想摩擦力計算においては、**前ステップの摩擦力はそのまに**、増分を加算する仕様になっている

▶ 摩擦力の計算 (増分ステップ n)

(1) 予想摩擦力

$$f_{t\,n+1}^{trial} = f_{t\,n} + \Delta f_t^{stick}$$

$$t^{trial} = \frac{f_{t\,n+1}^{trial}}{\|f_{t\,n+1}^{trial}\|}$$

(2) 摩擦状態のチェック

$$\phi^{trial} = |f_{t\,n+1}^{trial}| - \mu f_n$$

(3) 摩擦力の更新

$$\phi^{trial} < 0 \rightarrow \text{固着}$$

$$f_{t\,n+1} = f_{t\,n+1}^{trial}$$

$$\phi^{trial} \geq 0 \rightarrow \text{滑り}$$

$$f_{t\,n+1} = \mu f_n t^{trial} = \mu \lambda t^{trial}$$

■ プログラムの該当箇所

■ getTrialFricForceANDcheckFricState @ fstr_AddContactStiff.f90

```
397      dotP = dot_product(nTm,ndDu)
398      do i = 1, 3
399          relativeDisp(i) = - ndDu(i)
400          do j = 1, nnode
401              relativeDisp(i) = relativeDisp(i) + shapefunc(j)*ndDu(j*3+i)
402          enddo
403          relativeDisp(i) = relativeDisp(i) - dotP*normal(i)
404          ctstate%tangentForce_trial(i) = ctstate%tangentForce(i) - tPenalty*relativeDisp(i)
405      enddo
```

$$\mathbf{f}_{t,n+1}^{trial} = \mathbf{f}_{t,n} - \alpha \Delta \mathbf{u}_t \quad (\Delta \mathbf{f}_t^{stick} = -\alpha \Delta \mathbf{u}_t)$$

■ 変数の対応

- tangentForce : $\mathbf{f}_{t,n}$ (前ステップでの摩擦力)
- tangentForce_trial : $\mathbf{f}_{t,n+1}^{trial}$ (現ステップの試行摩擦力)
- tangentForce_final : $\mathbf{f}_{t,n+1}$ (現ステップの摩擦力)

■ 現行仕様では、大回転が生じると精度が悪くなる

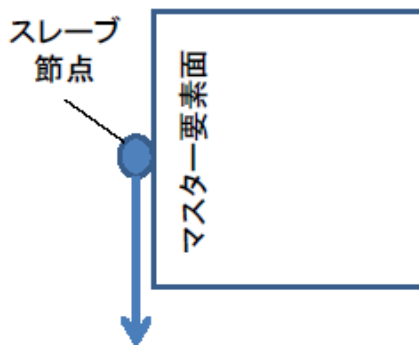
$$\mathbf{f}_{t,n+1}^{trial} = \mathbf{f}_{t,n} - \alpha \Delta \mathbf{u}_T$$

- 摩擦力の向きが前ステップのままなので新しい向きに対して面外方向成分が生じる（面内成分も目減りする）

- 下記のようなケースで現ステップでの修正量が少ないと、影響が目立つ

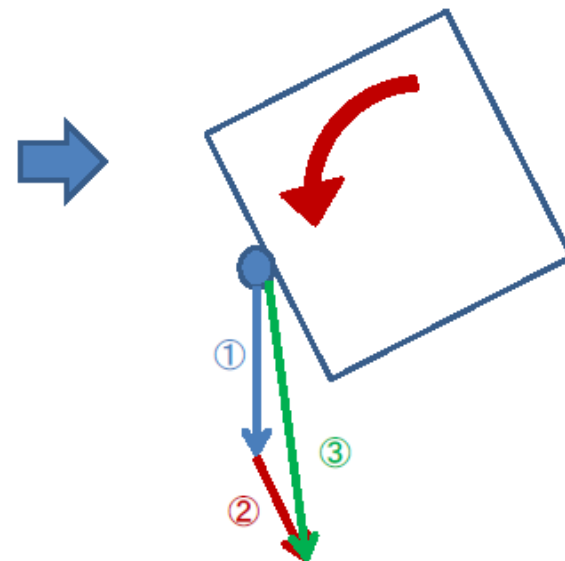
- α が小さい（Newton法の収束の都合などによる）
- 剛体回転（相対滑り $\Delta \mathbf{u}_T$ が発生しない）

サブステップ: n



摩擦力の発生時は、マスター要素面に沿っている。

サブステップ: n+1



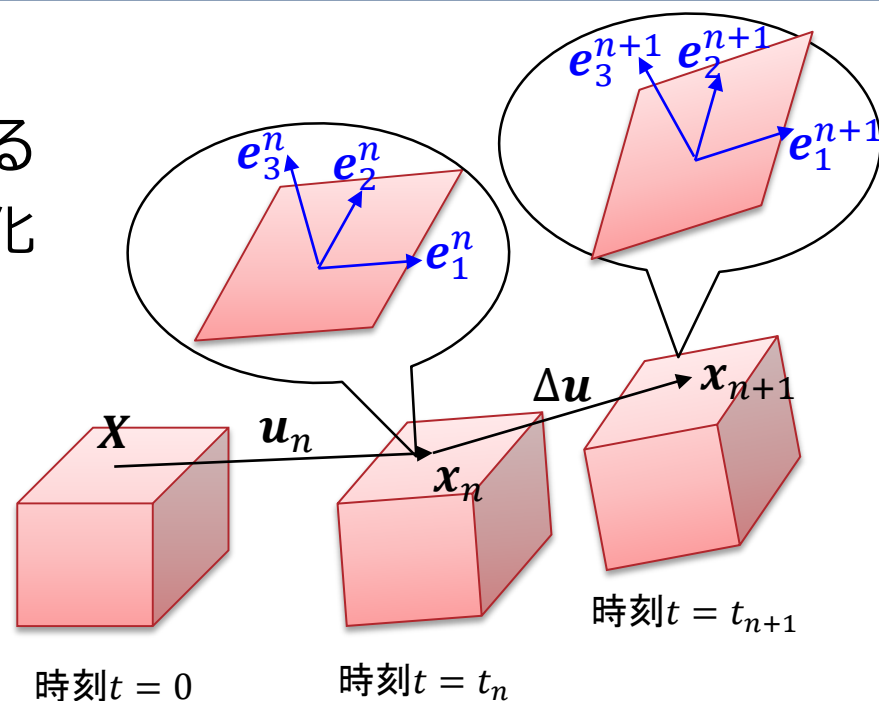
- ① マスター要素面が回転しても、それまでの摩擦力の向きは更新されず、前のサブステップの向きのまま持ち越される（その時点で、摩擦力はマスター面外へ飛び出している）。
- ② 今回サブステップの摩擦力増加分がそれに加算される（加算されない場合もある）。
- ③ 今回サブステップの加算分(②)の有無にかかわらず、摩擦力はマスター要素面外へずれる。

1. 要素面の回転変換を導出し、 tangentForceをこれで回転させる

■ 時刻 $t_n \rightarrow t_{n+1}$ における要素基底の変化
 $\mathbf{e}_i^{n+1} = f(\mathbf{e}_i^n)$

■ f を拡張と回転に分解
 (f の変換行列を極分解する)
 $\mathbf{e}_i^{n+1} = f_R(f_u(\mathbf{e}_i^n))$

■ 回転変換のみを $f_{t,n}$ にかける
 $f_{t,n+1} = f_R(f_{t,n})$



2. 新しい面にtangentForceを射影し、面外成分を除去

3. 接触要素面の基底ベクトルに成分分解する

1. $\mathbf{f}_{t,n} = f_{t,n}^1 \mathbf{e}_1^n + f_{t,n}^2 \mathbf{e}_2^n$: 基底ベクトルによって分解

2. $\mathbf{f}_{t,n+1}^{trial} = f_{t,n}^1 \mathbf{e}_1^{n+1} + f_{t,n}^2 \mathbf{e}_2^{n+1} - \alpha \Delta \mathbf{u}_t$: 成分を引き継ぎ、基底のみ更新

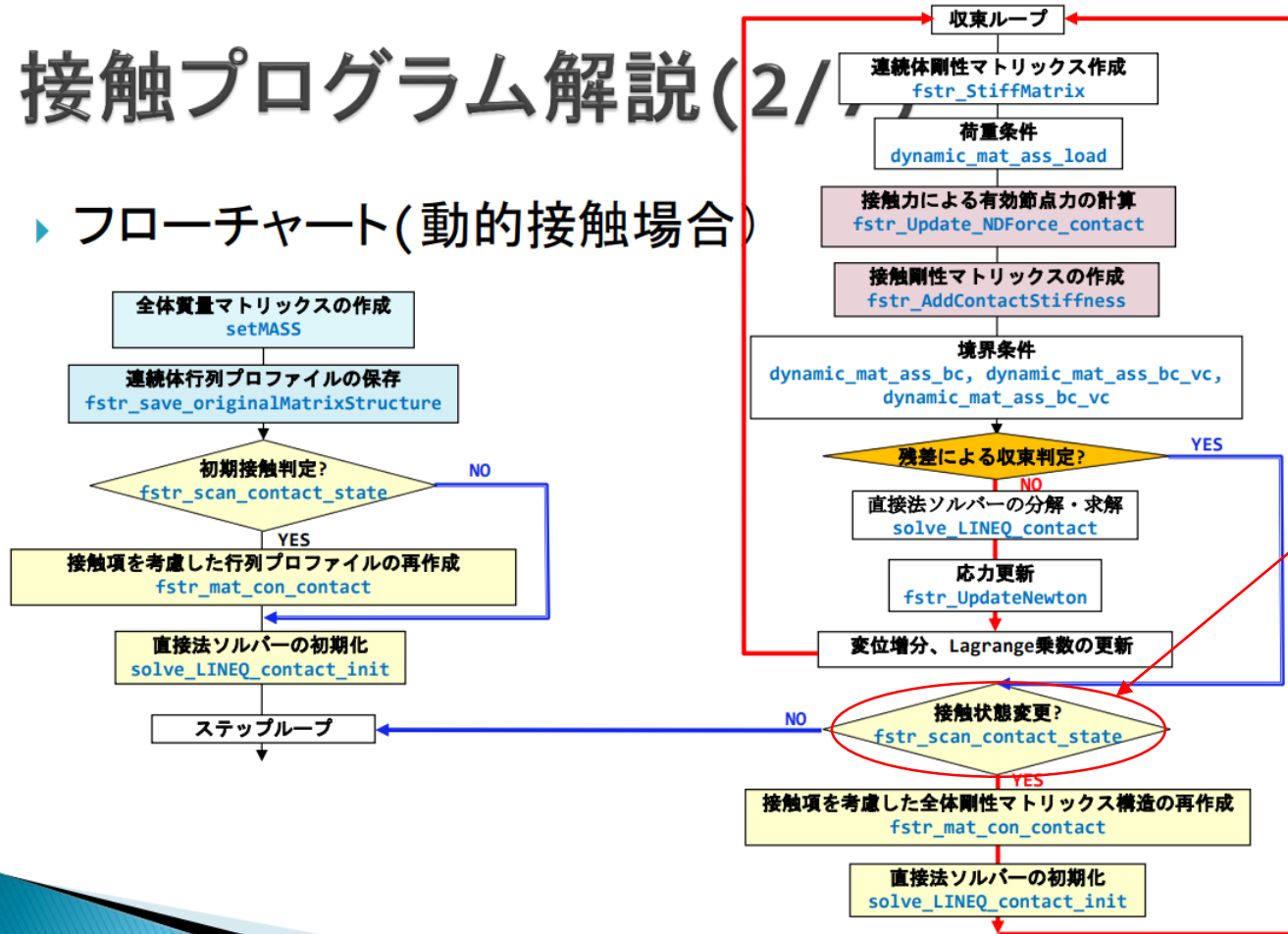
tangentForceの更新タイミングがおかしい？

課題2

■ tangentForceの更新タイミング

接触プログラム解説(2/)

▶ フローチャート(動的接触場合)



tangentForceは
scan_contact_state
の中で更新
= 毎接触ループごとに
更新

■ tangentForceの更新タイミング

■ scan_contact_stateの中

● 設定

- track_contact_position @ fstr_contact_def.f90 L524
contact%states(nslave)%tangentForce(1:3) = contact%states(nslave)%tangentForce_final(1:3)

● ゼロクリア

- fstr_contact_def.f90 L312 ※CONTACTからFREEに変化したとき
- fstr_contact_def.f90 L535 ※有限滑りで要素面集合からはみ出たとき

■ 疑問

- tangentForceが前ステップ終了時点での摩擦力なら、接触反復の途中でゼロクリアしてはいけない
 - 接触反復内においてtangentForce_finalで上書きすると、2反復目以降は「前ステップ終了時点での摩擦力」でなくなってしまう
 - 例えば、増分ステップ $n \rightarrow n+1$, k 回目の接触反復の場合に、固着摩擦力が以下ようになる
 - 本来計算すべき量： $\mathbf{f}_{t,n+1}^{trial} = \mathbf{f}_{t,n} - \alpha \Delta \mathbf{u}_T^{(k)}$
 - 今計算されている量： $\mathbf{f}_{t,n+1}^{trial} = \mathbf{f}_{t,n} - \alpha \Delta \mathbf{u}_T^{(1)} - \alpha \Delta \mathbf{u}_T^{(2)} - \dots - \alpha \Delta \mathbf{u}_T^{(k)}$
- ※ $\Delta \mathbf{u}_T^{(k)}$ は1増分ステップでの相対滑り変位（毎接触反復で生じる差分ではない）

■ 接触有りの解析は従動力が使えない？ (v4.6)

■ fstrNewton

```
L81  call fstr_ass_load(cstep, hecMESH, hecMAT, fstrSOLID, fstrPARAM)
...
L89  do iter=1,fstrSOLID%step_ctrl(cstep)%max_iter #Newton Loop
...
L127  if( fstrSOLID%DLOAD_follow /= 0 ) & #追従力の呼び出し
L128    call fstr_ass_load(cstep, hecMESH, hecMAT, fstrSOLID, fstrPARAM )
...
L151 enddo
```

■ fstr_Newton_contactSLag

```
L524 call fstr_ass_load(cstep, hecMESH, hecMAT, fstrSOLID, fstrPARAM)
...
L545 loopFORcontactAnalysis: DO WHILE( .TRUE. ) #Contact Loop
...
L552  do iter = 1, fstrSOLID%step_ctrl(cstep)%max_iter #Newton Loop
...
    #追従力の呼び出しが無い
...
L668  enddo
```