

Manuel d'Utilisation
Fascicule U4.4- : Modélisation
Document : U4.44.01

Opérateurs AFFE_CHAR_MECA et AFFE_CHAR_MECA_F

1 But

Affecter des chargements et des conditions aux limites sur un modèle mécanique.

- Pour AFFE_CHAR_MECA, les valeurs affectées ne dépendent d'aucun paramètre et sont définies par des valeurs réelles.
- Pour AFFE_CHAR_MECA_F, les valeurs affectées sont fonction d'un ou plusieurs paramètres dans l'ensemble {INST, X, Y, Z}.

Ces fonctions doivent préalablement être définies notamment par l'appel à un des opérateurs :

- DEFI_CONSTANTE [U4.31.01],
- DEFI_FONCTION [U4.31.02],
- DEFI_NAPPE [U4.31.03],
- CALC_FONC_INTERP [U4.32.01].

Table des matières

1 But.....	1
2 Syntaxe générale	4
3 Généralités.....	6
4 Opérandes	7
4.1 Généralités sur les opérandes.....	7
4.2 Opérande MODELE	9
4.3 Opérande VERI_NORM	9
4.4 Opérande LIAISON_XFEM (AFFE_CHAR_MECA seulement)	9
4.5 Opérande TEMP_CALCULEE (AFFE_CHAR_MECA seulement)	9
4.6 Opérande EVOL_CHAR (AFFE_CHAR_MECA seulement).....	10
4.7 Opérande PESANTEUR (AFFE_CHAR_MECA seulement).....	10
4.8 Opérande ROTATION (AFFE_CHAR_MECA seulement).....	10
4.9 Mot-clé DDL_IMPO.....	11
4.10 Mot-clé FACE_IMPO	14
4.11 Mot-clé LIAISON_DDL	16
4.12 Mot-clé LIAISON_OBLIQUE	18
4.13 Mot-clé LIAISON_GROUP.....	20
4.14 Mot-clé LIAISON_MAIL.....	25
4.15 Mot-clé LIAISON_CYCL.....	28
4.16 Mot-clé CONTACT.....	31
4.17 Mot-clé FORCE_NODALE.....	47
4.18 Mot-clé LIAISON_SOLIDE	48
4.19 Mot-clé LIAISON_ELEM.....	49
4.20 Mot-clé LIAISON_UNIF.....	52
4.21 Mot-clé LIAISON_CHAMNO	53
4.22 Mot-clé CHAMNO_IMPO.....	54
4.23 Mot-clé LIAISON_UNILATER	55
4.24 Mot-clé VECT_ASSE	56
4.25 Mot-clé FORCE_FACE	57
4.26 Mot-clé FORCE_ARETE	58
4.27 Mot-clé FORCE_CONTOUR.....	59
4.28 Mot-clé FORCE_INTERNE.....	60
4.29 Mot-clé PRES_REP.....	61
4.30 Mot-clé EFFE_FOND	63
4.31 Mot-clé EPSI_INIT	64
4.32 Mot-clé FORCE_POUTRE.....	66
4.33 Mot-clé DDL_POUTRE	68
4.34 Mot-clé FORCE TUYAU	69
4.35 Mot-clé FORCE_COQUE	70

Titre : Opérateurs AFFE_CHAR_MECA et AFFE_CHAR_MECA_F

Date : 29/05/07

Auteur(s) : X. DESROCHES

Clé : U4.44.01-I

Page : 3/94

4.36	Mot-clé LIAISON_COQUE	73
4.37	Mot-clé RELA_CINE_BP	75
4.38	Mot-clé FORCE_ELEC.....	76
4.39	Mot-clé INTE_ELEC	79
4.40	Mot-clé IMPE_FACE (Phénomène 'ACOUSTIQUE').....	81
4.41	Mot-clé VITE_FACE (Phénomène 'ACOUSTIQUE').....	82
4.42	Mot-clé ONDE_PLANE.....	83
4.43	Mot-clé ONDE_FLUI (Phénomène 'ACOUSTIQUE').....	85
4.44	Mot-clé FLUX_THM_REP	86
4.45	Mot-clé ARLEQUIN	87
4.46	Mot-clé GRAPPE_FLUIDE	89

2 Syntaxe générale

```
ch [char_meca] = AFFE_CHAR_MECA
```

```
( ♦ MODELE = mo, [modele]

♦ | VERI_NORM = / 'OUI', [DEFAULT]
              / 'NON',
| LIAISON_XFEM= 'OUI'
| TEMP_CALCULEE= tempe, / [evol_ther]
                        / [cham_no_sdaster]
                        / [carte_sdaster]

| EVOL_CHAR = evch [evol_char]
| PESANTEUR= (g, ap, bp, cp) [l_R]
| ROTATION= (omega, ar, br, cr) [l_R]
| DDL_IMPO= _F (voir mot-clé DDL_IMPO [$ 4.9])
| FACE_IMPO= _F (voir mot-clé FACE_IMPO [$ 4.10])
| LIAISON_DDL= _F (voir mot-clé LIAISON_DDL [$ 4.11])
| LIAISON_OBLIQUE= _F (voir mot-clé LIAISON_OBLIQUE [$ 4.12])
| LIAISON_GROUP= _F (voir mot-clé LIAISON_GROUP [$ 4.13])
| LIAISON_MAIL= _F (voir mot-clé LIAISON_MAIL [$ 4.14])
| LIAISON_CYCL= _F (voir mot-clé LIAISON_CYCL [$ 4.15])
| CONTACT= _F (voir mot-clé CONTACT [$ 4.16])
| FORCE_NODALE= _F (voir mot-clé FORCE_NODALE [$ 4.17])
| LIAISON_SOLIDE= _F (voir mot-clé LIAISON_SOLIDE [$ 4.18])
| LIAISON_ELEM= _F (voir mot-clé LIAISON_ELEM [$ 4.19])
| LIAISON_UNIF= _F (voir mot-clé LIAISON_UNIF [$ 4.20])
| LIAISON_CHAMNO= _F (voir mot-clé LIAISON_CHAMNO [$ 4.21])
| CHAMNO_IMPO= _F (voir mot-clé CHAMNO_IMPO [$ 4.22])
| LIAISON_UNILATER= _F (voir mot-clé LIAISON_UNILATER [$ 4.23])
milieu continu | VECT_ASSE= _F (voir mot-clé VECT_ASSE [$ 4.24])
               | FORCE_FACE= _F (voir mot-clé FORCE_FACE [$ 4.25])
               | FORCE_ARETE= _F (voir mot-clé FORCE_ARETE [$ 4.26])
               | FORCE_CONTOUR= _F (voir mot-clé FORCE_CONTOUR [$ 4.27])
               | FORCE_INTERNE= _F (voir mot-clé FORCE_INTERNE [$ 4.28])
               | PRES_REP= _F (voir mot-clé PRES_REP [$ 4.29])
               | EFFE_FOND= _F (voir mot-clé EFFE_FOND [$ 4.30])
poutre coque | EPSI_INIT= _F (voir mot-clé EPSI_INIT [$ 4.31])
              | FORCE_POUTRE= _F (voir mot-clé FORCE_POUTRE [$ 4.32])
              | DDL_POUTRE= _F (voir mot-clé DDL_POUTRE [$ 4.33])
              | FORCE TUYAU= _F (voir mot-clé FORCE TUYAU [$ 4.34])
              | FORCE_COQUE= _F (voir mot-clé FORCE_COQUE [$ 4.35])
              | LIAISON_COQUE= _F (voir mot-clé LIAISON_COQUE [$ 4.36])
béton
électroméca | RELA_CINE_BP= _F (voir mot-clé RELA_CINE_BP [$ 4.37])
              | FORCE_ELEC= _F (voir mot-clé FORCE_ELEC [$ 4.38])
              | INTE_ELEC= _F (voir mot-clé INTE_ELEC [$ 4.39])
acoustique | IMPE_FACE= _F (voir mot-clé IMPE_FACE [$ 4.40])
            | VITE_FACE= _F (voir mot-clé VITE_FACE [$ 4.41])
            | ONDE_FLUI= _F (voir mot-clé ONDE_FLUI [$ 4.42])
            | ONDE_PLANE= _F (voir mot-clé ONDE_PLANE [$ 4.43])
thermo-hydrau | FLUX_THM_REP= _F (voir mot-clé FLUX_THM_REP [$ 4.44])
méth. Arlequin | ARLEQUIN= _F (voir mot-clé ARLEQUIN [$ 4.45])

forces fluides de chute de grappes
| GRAPPE_FLUIDE = _F (voir mot-clé GRAPPE_FLUIDE [$ 4.46])

♦ INFO = / 1, [DEFAULT]
         / 2,
) ;
```

```
ch [char_meca] = AFFE_CHAR_MECA_F
```

```
(
  ◆ MODELE= mo, [modele]
  ◆ DDL_IMPO=_F (voir mot-clé DDL_IMPO [§ 4.9])
  FACE_IMPO=_F (voir mot-clé FACE_IMPO [§ 4.10])
  LIAISON_DDL=_F (voir mot-clé LIAISON_DDL [§ 4.11])
  LIAISON_OBLIQUE=_F (voir mot-clé LIAISON_OBLIQUE [§ 4.12])
  LIAISON_GROUP=_F (voir mot-clé LIAISON_GROUP [§ 4.13])
  CONTACT=_F (voir mot-clé CONTACT [§ 4.16])
  FORCE_NODALE=_F (voir mot-clé FORCE_NODALE [§ 4.17])
  LIAISON_SOLIDE=_F (voir mot-clé LIAISON_SOLIDE [§ 4.18])
  LIAISON_UNIF=_F (voir mot-clé LIAISON_UNIF [§ 4.20])
  LIAISON_UNILATER=_F (voir mot-clé LIAISON_UNILATER [§ 4.23])
milieu continu
  FORCE_FACE=_F (voir mot-clé FORCE_FACE [§ 4.25])
  FORCE_ARETE=_F (voir mot-clé FORCE_ARETE [§ 4.26])
  FORCE_CONTOUR=_F (voir mot-clé FORCE_CONTOUR [§ 4.27])
  FORCE_INTERNE=_F (voir mot-clé FORCE_INTERNE [§ 4.28])
  PRES_REP=_F (voir mot-clé PRES_REP [§ 4.29])
  EFFE_FOND=_F (voir mot-clé EFFE_FOND [§ 4.30])
  EPSI_INIT=_F (voir mot-clé EPSI_INIT [§ 4.31])
poutre coque
  FORCE_POUTRE=_F (voir mot-clé FORCE_POUTRE [§ 4.32])
  FORCE TUYAU=_F (voir mot-clé FORCE TUYAU [§ 4.34])
  FORCE_COQUE=_F (voir mot-clé FORCE_COQUE [§ 4.35])
  LIAISON_COQUE=_F (voir mot-clé LIAISON_COQUE [§ 4.36])
acoustique
  IMPE_FACE=_F (voir mot-clé IMPE_FACE [§ 4.37])
  VITE_FACE=_F (voir mot-clé VITE_FACE [§ 4.40])
  ONDE_PLANE=_F (voir mot-clé ONDE_PLANE [§ 4.41])
  FLUX_THM_REP=_F (voir mot-clé FLUX_THM_REP [§ 4.44])
  VERI_NORM = / 'OUI', [DEFAULT]
               / 'NON', )
```

3 Généralités

Messages d'erreur possibles liés à la commande AFPE_CHAR_MECA

Il arrive parfois qu'une commande de calcul mécanique (MECA_STATIQUE, STAT_NON_LINE, ...) s'arrête en erreur fatale lors du calcul des seconds membres élémentaires dus aux chargements définis dans les commandes AFPE_CHAR_MECA_xx. Lorsque le code s'arrête pendant ces calculs élémentaires, une information importante du message d'erreur est le nom de l'option de calcul demandée par le code.

Le nom de cette option est en général inconnu de l'utilisateur et il lui est donc difficile de comprendre le message.

Dans le tableau ci-dessous, on donne en vis-à-vis des noms des options de calcul, le nom de la commande et du mot clé facteur qui permettent d'activer cette option.

Option de calcul élémentaire	Commande	Mot clé facteur
CHAR_MECA_EPSI_F	AFPE_CHAR_MECA_F	EPSI_INIT
CHAR_MECA_EPSI_R	AFPE_CHAR_MECA	EPSI_INIT
CHAR_MECA_FF1D1D	AFPE_CHAR_MECA_F	FORCE_POUTRE
CHAR_MECA_FF1D2D	AFPE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FF1D3D	AFPE_CHAR_MECA_F	FORCE_ARETE
CHAR_MECA_FF2D2D	AFPE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FF2D3D	AFPE_CHAR_MECA_F	FORCE_FACE
CHAR_MECA_FF3D3D	AFPE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FFCO2D	AFPE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FFCO3D	AFPE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FLUX_F	AFPE_CHAR_MECA_F	FLUX_THM_REP
CHAR_MECA_FLUX_R	AFPE_CHAR_MECA	FLUX_THM_REP
CHAR_MECA_FORC_F	AFPE_CHAR_MECA_F	FORCE_NODALE
CHAR_MECA_FORC_R	AFPE_CHAR_MECA	FORCE_NODALE
CHAR_MECA_FR1D1D	AFPE_CHAR_MECA	FORCE_POUTRE
CHAR_MECA_FR1D2D	AFPE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FR1D3D	AFPE_CHAR_MECA	FORCE_ARETE
CHAR_MECA_FR2D2D	AFPE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FR2D3D	AFPE_CHAR_MECA	FORCE_FACE
CHAR_MECA_FR3D3D	AFPE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FRCO2D	AFPE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRCO3D	AFPE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRELEC	AFPE_CHAR_MECA	FORCE_ELEC
CHAR_MECA_PESA_R	AFPE_CHAR_MECA	PESANTEUR
CHAR_MECA_PRES_F	AFPE_CHAR_MECA_F	PRES_REP
CHAR_MECA_PRES_R	AFPE_CHAR_MECA	PRES_REP
CHAR_MECA_ROTA_R	AFPE_CHAR_MECA_F	ROTATION

4 Opérandes

4.1 Généralités sur les opérandes

4.1.1 Deux catégories d'opérandes

Les opérandes sous un mot clé facteur sont de deux formes :

- les opérandes spécifiant les entités géométriques sur lesquelles sont affectés les chargements (mots clé GROUP_NO, GROUP_MA, etc ...). Les arguments de ces opérandes sont identiques pour les deux opérateurs,
- les opérandes spécifiant les valeurs affectées (DX, DY, etc ...). La signification de ces opérandes est la même pour les deux opérateurs. Les arguments de ces opérandes sont tous du type réel pour l'opérateur AFPE_CHAR_MECA et du type fonction (créé notamment par l'un des opérateurs DEFI_FONCTION, DEFI_NAPPE ou DEFI_CONSTANTE) pour l'opérateur AFPE_CHAR_MECA_F.

Ceci est vrai à une exception près : l'argument de COEF_MULT pour le mot clé facteur LIAISON_DDL dans AFPE_CHAR_MECA_F est obligatoirement de type réel.

Nous ne distinguerons donc pas dans ce document, sauf mention expresse du contraire, les deux opérateurs AFPE_CHAR_MECA et AFPE_CHAR_MECA_F.

4.1.2 Désignation des entités topologiques d'affectation des chargements

De façon générale, les entités sur lesquelles des valeurs doivent être affectées sont définies :

- par nœud et dans ce cas :
 - soit par l'opérande GROUP_NO permettant d'introduire une liste de groupes de nœuds : notons que dans certains cas un groupe de nœud ne doit contenir qu'un seul nœud,
 - soit par l'opérande NOEUD permettant d'introduire une liste de nœuds.
- par maille et dans ce cas :
 - soit par GROUP_MA permettant d'introduire une liste de groupes de mailles,
 - soit par MAILLE permettant d'introduire une liste de mailles.

4.1.3 Règle de surcharge

Pour définir le domaine d'affectation le plus simplement possible, on utilise la **règle de surcharge** définie dans le document "Règles de surcharge" [U1.03.00] :

c'est la dernière affectation qui prime.

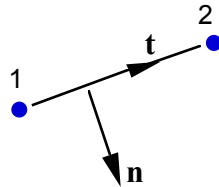
4.1.4 Eléments structuraux, milieux continus

Pour l'affectation des chargements répartis sur les éléments à feuillet moyen (plaque - coque) ou à fibre moyenne (poutre, câble, barre) les mots-clés facteurs sont distincts de ceux utilisés pour les milieux continus.

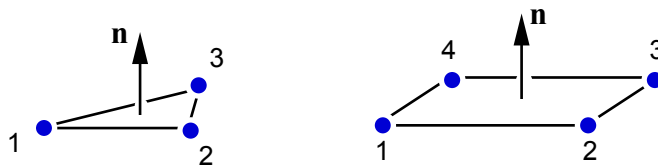
4.1.5 Normales et tangentes aux mailles

Normales :

- SEG2 ou SEG3 en 2D (coordonnées définies par COOR_2D dans le fichier de maillage au format *Aster*). La normale **n** est telle que (**n**, **t**) forment un repère direct, **t** étant porté par le segment orienté par les deux premiers nœuds du segment.

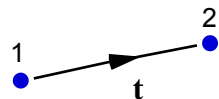


- QUAD4, ..., QUAD9, TRIA3, TRIA6 en 3D (coordonnées définies par COOR_3D dans le fichier de maillage au format *Aster*). L'orientation de la normale **n** est celle correspondant au sens direct de la description de la maille.

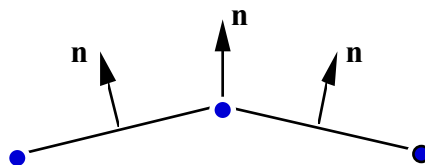


Tangentes :

Ne peut être spécifié que si la maille est du type SEG2 ou SEG3 en 2D. La tangente est celle définie par le segment orienté par ses deux premiers nœuds.



Si DNOR (ou DTAN) sont spécifiées, la normale (ou la tangente) sur un nœud est la moyenne des normales ou des tangentes des mailles qui ont ce nœud en commun (sauf pour les éléments quadratiques courbes où la normale est correctement calculée en tout point)



4.2 Opérande MODELE

◆ `MODELE= mo ,`

Concept produit par l'opérateur `AFFE_MODELE` où sont définis les types d'éléments finis affectés sur le maillage.

4.3 Opérande VERI_NORM

| `VERI_NORM= / 'OUI' [DEFAULT]`
`/ 'NON'`

Vérification de l'orientation des normales aux mailles surfaciques en 3D (mailles de peau `TRIA` ou `QUAD`) et linéiques en 2D (mailles de peau `SEG`). Ceci concerne les mot-clés `PRES_REP` et `FACE_IMPO 'DNOR'`.

Si une normale n'est pas sortante, il y a émission d'un message d'erreur fatale.

Pour réorienter les mailles de façon à avoir des normales sortantes, il faut utiliser l'opérateur `MODI_MALLAGE [U4.23.04]` mot-clés `ORIE_PEAU_2D` et `ORIE_PEAU_3D`.

Aucune vérification n'est faite sur les coques. Pour vérifier leur orientation, on renvoie également à l'opérateur `MODI_MALLAGE` mot-clé `ORIE_NORM_COQUE`.

4.4 Opérande LIAISON_XFEM (AFFE_CHAR_MECA seulement)

| `LIAISON_XFEM= 'OUI' ,`

Lors d'un calcul avec la méthode X-FEM [R7.02.12], il est nécessaire de créer une charge supplémentaire pour annuler certains ddls enrichis. Il faut donc impérativement indiquer `LIAISON_XFEM='OUI'` dans cette charge spécifique pour tout calcul X-FEM, comme sur l'exemple suivant :

```
chxfem = AFFE_CHAR_MECA ( MODELE = modele,
                          LIAISON_XFEM = 'OUI',
                          )
```

4.5 Opérande TEMP_CALCULEE (AFFE_CHAR_MECA seulement)

| `TEMP_CALCULEE= tempe ,`

Concept produit par un calcul thermique linéaire ou non linéaire (`THER_LINEAIRE [U4.54.01]`, `THER_NON_LINE [U4.54.02]`) ou créé à partir de valeurs de températures affectées par la commande `CREA_CHAMP [U4.72.04]` mot clé `AFFE` ou à partir de la commande `CREA_RESU [U4.44.12]`. Si le concept `tempe` est de type `cham_no_sdaster` alors le chargement thermique sera supposé constant en temps. S'il est de type `evol_ther`, les prolongements éventuels jusqu'aux bornes du calcul transitoire seront supposés constants.

4.6 Opérande EVOL_CHAR (AFFE_CHAR_MECA seulement)

/ EVOL_CHAR = evch,

Chargements évolutifs dans le temps de type 'evol_char' [U5.01.17] produits par LIRE_RESU [U7.02.01] et contenant des champs de pression, des densités de force volumique en 2D ou 3D et des densités de force surfacique en 2D ou 3D.

4.7 Opérande PESANTEUR (AFFE_CHAR_MECA seulement)

| PESANTEUR = (g, ap, bp, cp),

Accélération et direction de la pesanteur. Le chargement qui en résulte est de la forme :

$$\rho g \frac{(a_p \mathbf{i} + b_p \mathbf{j} + c_p \mathbf{k})}{\sqrt{a_p^2 + b_p^2 + c_p^2}}$$

où $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ est le repère cartésien global.

ρ est la masse volumique définie comme caractéristique du matériau (voir opérateurs DEFI_MATERIAU [U4.43.01] et AFFE_MATERIAU [U4.43.03]).

4.8 Opérande ROTATION (AFFE_CHAR_MECA seulement)

| ROTATION = (ω , ar, br, cr),

Vitesse de rotation et direction du vecteur rotation qui conduit à :

$$\boldsymbol{\omega} = \omega \frac{(a_r \mathbf{i} + b_r \mathbf{j} + c_r \mathbf{k})}{\sqrt{a_r^2 + b_r^2 + c_r^2}}$$

Le chargement qui en résulte est : $\rho (\boldsymbol{\omega} \wedge \mathbf{OM}) \wedge \boldsymbol{\omega}$ où \mathbf{O} est l'origine des coordonnées et \mathbf{M} un point courant de la structure avec ρ masse volumique définie comme caractéristique du matériau (voir opérateurs DEFI_MATERIAU [U4.43.01] et AFFE_MATERIAU [U4.43.03]).

◇ CENTRE = (x, y, z),

Si le centre n'est pas l'origine, on peut préciser ses coordonnées (x, y, z).

Remarque importante :

On peut faire varier dans le temps la vitesse de rotation en décomposant la rotation de façon multiplicative entre chargement spatial et évolution en temps $\omega(t) = \omega_0 f(t)$, puis en multipliant la CHARGE par une fonction multiplicatrice (mot clef FONC_MULT) dans le calcul transitoire (DYNA_TRAN_MODAL, DYNA_LINE_TRAN, DYNA_NON_LINE). Toutefois, il convient de faire attention : le chargement $[\rho (\boldsymbol{\omega} \wedge \mathbf{OM}) \wedge \boldsymbol{\omega}]$ étant proportionnel au carré de la vitesse de rotation, $\omega(t)^2$, il faut affecter le carré de l'évolution en temps, $f(t)^2$, derrière FONC_MULT.

4.9 Mot-clé DDL_IMPO

4.9.1 But

Mot-clé facteur utilisable pour imposer, à des nœuds introduits par un (au moins) des mots clés : TOUT, NOEUD, GROUP_NO, MAILLE, GROUP_MA, une ou plusieurs valeurs de déplacement (ou de certaines grandeurs associées).

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.9.2 Syntaxe

- pour AFFE_CHAR_MECA


```

      | DDL_IMPO=_F ( ♦ / TOUT      = 'OUI' ,
                      / NOEUD      = lno ,      [l_noeud]
                      / GROUP_NO= lgno ,      [l_gr_noeud]
                      / MAILLE     = lma ,      [l_maille]
                      / GROUP_MA= lgma ,      [l_gr_maille]
                      ♦ / | DX      = ux ,      [R]
                        | DY      = uy ,      [R]
                        | DZ      = uz ,      [R]
                        | DRX     = θx ,      [R]
                        | DRY     = θy ,      [R]
                        | DRZ     = θz ,      [R]
                        | GRX     = g ,      [R]
                        | PRES=    p ,      [R]
                        | PHI     = φ ,      [R]
                        | TEMP=    T ,      [R]
                        | PRE1=    pr1 ,      [R]
                        | PRE2=    pr2 ,      [R]
                        | GONF=    treps ,      [R]
                        | LAGS_C=  lag ,      [R]
                      / LIAISON=  'ENCASTRE'
                      )
      
```

La liste exhaustive des ddls pouvant être imposés est :

```

DX, DY, DZ, DRX, DRY, DRZ, GRX, PRES, PHI,
TEMP, PRE1, PRE2, UI2, UI3, VI2, VI3, WI2, WI3, UO2,
UO3, VO2, VO3, WO2, WO3, UI4, UI5, VI4, VI5, WI4,
WI5, UO4, UO5, VO4, VO5, WO4, WO5, UI6, UO6, VI6,
VO6, WI6, WO6, WO, WI1, WO1, GONF, LIAISON, DCX,
DCY, DCZ, H1X, H1Y, H1Z, E1X, E1Y, E1Z, E2X, E2Y, E2Z,
E3X, E3Y, E3Z, E4X, E4Y, E4Z, LAGS_C [R]

```

- pour AFFE_CHAR_MECA_F


```

      | DDL_IMPO=_F ( ♦ / TOUT      = 'OUI' ,
                      / NOEUD      = lno ,      [l_noeud]
                      / GROUP_NO= lgno ,      [l_gr_noeud]
                      / MAILLE     = lma ,      [l_maille]
                      / GROUP_MA= lgma ,      [l_gr_maille]
                      ♦ / | DX      =
                        | ...
                      / LIAISON = 'ENCASTRE'
                      )
      
```

4.9.3 Opérandes

DDL_IMPO

Toutes les valeurs imposées sont définies dans le repère GLOBAL de définition du maillage.

- $DX = u_x$ ou u_{xf} | Valeur de la composante de déplacement en **translation** imposée sur les nœuds spécifiés
- $DY = u_y$ ou u_{yf}
- $DZ = u_z$ ou u_{zf}

Uniquement si les nœuds spécifiés appartiennent à des éléments **discrets** de translation - rotation, de **poutre** ou de **coque** :

- $DRX = \theta_x$ ou θ_{xf} | Valeur de la composante de déplacement en **rotation** imposée sur les nœuds spécifiés
- $DRY = \theta_y$ ou θ_{yf}
- $DRZ = \theta_z$ ou θ_{zf}

Uniquement si les nœuds spécifiés appartiennent à des éléments de poutre 'POU_D_TG' :

- $GRX = g$ ou gf | Valeur du gauchissement de la poutre

Uniquement si les nœuds spécifiés appartiennent à des éléments fluide ou fluide structure :

- $PRES = p$ ou p_f | Pression acoustique dans le fluide (modélisation '3D_FLUIDE')
- $PHI = \phi$ ou ϕ_f | Potentiel des déplacements du fluide (modélisations '3D_FLUIDE' et 'FLUI_STRU')

Uniquement si les nœuds spécifiés appartiennent à des éléments de surface libre :

- $DZ = u_z$ ou u_{zf} | Déplacement imposé de la surface libre (modélisation '2D_FLUI_PESA')
- $PHI = \phi$ ou ϕ_f | Potentiel des déplacements du fluide (modélisation '2D_FLUI_PESA')

Uniquement si les nœuds spécifiés appartiennent à des éléments THM :

- $PRES = p$ | Pression du fluide interstitiel (modélisations '3D_JOINT_CT')
- $TEMP = T$ | Température (modélisations 'XXXX_YYYY' avec
XXXX = 3D ou AXIS ou D_PLAN
YYYY = THM ou THHM ou THH)
- $PRE1 = p1$ | Pression capillaire ou pression du liquide ou du gaz (modélisations 'XXXX_YYYY' avec
XXXX = 3D ou AXIS ou D_PLAN
YYYY = THM ou THHM ou THH ou HM ou HHM)
- $PRE2 = p2$ | Pression du gaz (modélisations 'XXXX_YYYY' avec
XXXX = 3D ou AXIS ou D_PLAN
YYYY = THH ou THHM ou HHM)
- $LAGS_C = lag$ | Pression de contact intervenant dans la méthode continue (modélisations 'XXXX' avec
XXXX = 3D ou AXIS ou D_PLAN)

Uniquement si les nœuds spécifiés appartiennent à des éléments 'TUYAU'.

Ces éléments ont 15 DDL de coque :

U : gauchissement
I : "in plane"

V, W : ovalisation
O : "out of plane"

Soit :

- UI2 VI2 WI2 UO2 VO2 WO2 | DDL liés au mode 2
- UI3 VI3 WI3 UO3 VO3 WO3 | DDL liés au mode 3
- WO WI1 WO1 | DDL de gonflement et mode 1 sur W

Uniquement si les nœuds spécifiés appartiennent à des éléments 'TUYAU_6M'.

- UI4 VI4 WI4 UO4 VO4 WO4 DDL liés au mode 4
- UI5 VI5 WI5 UO5 VO5 WO5 DDL liés au mode 5
- UI6 VI6 WI6 UO6 VO6 WO6 DDL liés au mode 6

Uniquement si les nœuds spécifiés appartiennent à des éléments 'XXX_INCO'.

- GONF gonflement

LIAISON = 'ENCASTRE'

Permet d'encastrier directement des nœuds, c.a.d. de forcer à zéro les ddl de translation et de rotation. Les autres ddl ne sont pas modifiés.

4.9.4 Vérifications et recommandations

On vérifie que le ddl spécifié existe en ce nœud pour les éléments affectés dans le MODELE aux mailles qui contiennent le nœud.

Cependant, si la même condition aux limites est spécifiée deux fois par deux appels à AFFE_CHAR_MECA (par exemple, avec deux valeurs de déplacement imposé), cela conduit à une matrice singulière.

Si elle est spécifiée deux fois (ou plus) dans un seul appel à AFFE_CHAR_MECA, la règle de surcharge s'applique et un message d'alarme (indiquant la surcharge) est émis.

4.10 Mot-clé FACE_IMPO

4.10.1 But

Mot-clé facteur utilisable pour imposer, à tous les nœuds d'une face définie par une maille ou un groupe de mailles, une ou plusieurs valeurs de déplacement (ou de certaines grandeurs associées).

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.10.2 Syntaxe

- pour AFFE_CHAR_MECA

```

|  FACE_IMPO=_F (  ♦ / MAILLE =      lma , [l_maille]
                   /  / GROUP_MA=      lgma , [l_gr_maille]
                   ◇   SANS_MAILLE =    lma , [l_maille]
                   ◇   SANS_GROUP_MA =  lgma , [l_gr_maille]
                   ♦ / | DX =          ux , [R]
                   |   DY =          uy , [R]
                   |   DZ =          uz , [R]
                   |   DRX =         θx , [R]
                   |   DRY =         θy , [R]
                   |   DRZ =         θz , [R]
                   |   GRX =          g , [R]
                   |   PRES=         p , [R]
                   |   PHI =         φ , [R]
                   |   TEMP=         T , [R]
                   |   PRE1=        pr1 , [R]
                   |   PRE2=        pr2 , [R]
                   /   | DNOR=         un , [R]
                   |   DTAN=         ut , [R]
                   )

```

- pour AFFE_CHAR_MECA_F

```

|  FACE_IMPO=_F (  ♦ / MAILLE =      lma , [l_maille]
                   /  / GROUP_MA=      lgma , [l_gr_maille]
                   ◇   SANS_MAILLE =    lma2 , [l_maille]
                   ◇   SANS_GROUP_MA =  lgma2 , [l_gr_maille]
                   ♦ / | DX =          uxf , [fonction]
                   |   DY =          uyf , [fonction]
                   |   DZ =          uzf , [fonction]
                   |   DRX =         θxf , [fonction]
                   |   DRY =         θyf , [fonction]
                   |   DRZ =         θzf , [fonction]
                   |   GRX =          gf , [fonction]
                   |   PRES=         pf , [fonction]
                   |   PHI =         φf , [fonction]
                   |   TEMP=         Tf , [fonction]
                   |   PRE1=        pr1f , [fonction]
                   |   PRE2=        pr2f , [fonction]
                   /   | DNOR=         un , [fonction]
                   |   DTAN=         ut , [fonction]
                   )

```

4.10.3 Opérandes

```

◇      SANS_MAILLE =    lma2,      [l_maille]
◇      SANS_GROUP_MA = lgma2,      [l_gr_maille]

```

Indique que l'on veut omettre les nœuds des listes lma2 et/ou lgma2 de la liste lma ou lgma.

Exemple : `FACE_IMPO = (_F (GROUP_MA=Gauche, DX=0, DY=0),
_F (GROUP_MA=Haut, SANS_GROUP_MA=Gauche, DNOR=0),)`

La signification de la 2ème occurrence de FACE_IMPO est : "pour tous les noeuds de Haut sauf ceux qui appartiennent à Gauche, DNOR=0."

Ceci permet de ne pas avoir de conditions aux limites redondantes.

```

♦ / | DX =
    | DY =
    | DZ =
    | DRX =
    | DRY =
    | DRZ =
    | GRX =
    | PRES=
    | PHI =
    | TEMP=
    | PRE1=
    | PRE2=

```

Les composantes, imposées sur tous les nœuds appartenant aux mailles spécifiées, sont définies dans le **repère** GLOBAL de définition du maillage.

Les faces considérées sont constituées :

- soit de TRIA3, TRIA6, QUAD4, QUAD8, QUAD9 en dimension 3,
- soit de SEG2 ou SEG3 en dimension 2 (la face se réduit à un bord).

Remarque :

*Les composantes de déplacement en rotation DRX, DRY, DRZ ne peuvent intervenir que sur des nœuds qui appartiennent à des éléments de **poutre** ou de **coque** (voir DDL_IMPO [§4.10]),*

la composante GRX sur des éléments de poutre 'POU_D_TG',

les composantes PRES et PHI sur des éléments des modélisations '3D_FLUIDE' et 'FLUI_STRU', les composantes DZ et PHI sur des éléments de la modélisation '2D_FLUI_PESA'.

Les composantes TEMP, PRE1, PRE2 sur des éléments des modélisations THM.

```

/ | DNOR =
  | DTAN =

```

Les composantes imposées sont définies selon la normale ou la tangente à une maille (**repère local**).

DNOR : composante normale (voir [U4.44.01 §4.1]),

DTAN : composante tangentielle (voir [U4.44.01 §4.1]).

4.11 Mot-clé LIAISON_DDL

4.11.1 But

Mot-clé facteur utilisable pour définir une relation linéaire entre des degrés de liberté de deux ou plusieurs nœuds.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.11.2 Syntaxe

- pour AFFE_CHAR_MECA
LIAISON_DDL=_F(
 - / NOEUD = lno , [l_noeud]
 - / GROUP_NO = lgrno , [l_gr_noeud]
 - DDL=
 - 'DX',
 - 'DY',
 - 'DZ',
 - 'DRX',
 - 'DRY',
 - 'DRZ',
 - COEF_MULT = α_i , [l_R]
 - COEF_IMPO = β , [R]
)
- pour AFFE_CHAR_MECA_F
LIAISON_DDL=_F(
 - / NOEUD = lno , [l_noeud]
 - / GROUP_NO = lgrno , [l_gr_noeud]
 - DDL=
 - 'DX',
 - 'DY',
 - 'DZ',
 - 'DRX',
 - 'DRY',
 - 'DRZ',
 - / COEF_MULT = α_i , [l_R]
 - / COEF_MULT_FONC = α_{if} , [l_fonction]
 - COEF_IMPO = β_f , [fonction]
)

4.11.3 Opérandes

GROUP_NO ou NOEUD : liste des nœuds N_i ($i = 1, r$) ordonnée de façon naturelle :

- dans l'ordre de la liste de groupes de nœuds, et pour chaque groupe de nœuds, dans l'ordre de définition du groupe par GROUP_NO,
- dans l'ordre de la liste de nœuds pour NOEUD.

DDL : liste de ddl U_i ($i = 1, r$) de r textes pris parmi :

'DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ'

COEF_MULT : liste α_i ($i = 1, r$) de coefficients (de type réel pour AFFE_CHAR_MECA et pour AFFE_CHAR_MECA_F).

COEF_MULT_FONC : liste α_i ($i = 1, r$) de coefficients de type fonction de la géométrie uniquement pour AFFE_CHAR_MECA_F.

COEF_IMPO : coefficient β pour AFFE_CHAR_MECA, fonction du temps pour AFFE_CHAR_MECA_F.

La condition cinématique suivante sera appliquée :
$$\sum_{i=1}^r \alpha_i U_i = \beta$$

4.11.4 Précautions d'utilisation

4.11.4.1 Composantes en rotation

Les composantes de déplacement en rotation DRX, DRY, DRZ ne peuvent intervenir que dans des combinaisons affectées **uniquement** à des nœuds qui appartiennent à des éléments **discrets** de translation-rotation, de **poutre** ou de **coque** (voir DDL_IMPO : cf. [§4.10]).

4.11.4.2 Relation linéaire entre les ddl d'un même nœud

Dans ce cas particulier, on répétera derrière le mot clé NOEUD le nom du nœud autant de fois qu'il y a de ddl dans la relation. Exemple : pour imposer $U_x = U_y$ sur le nœud N1, on écrira :

```
LIAISON_DDL = _F ( NOEUD      = ( 'N1' , 'N1' ),
                    DDL        = ( 'DX' , 'DY' ),
                    COEF_MULT  = ( 1. , -1. ),
                    COEF_IMPO  = 0. ,
                    )
```

4.11.4.3 Relation linéaire entre groupes de nœuds

Il est important de noter qu'à une occurrence du mot-clé facteur LIAISON_DDL correspond une et une seule relation linéaire.

Si on veut imposer la même relation entre 2 groupes de nœuds GRN01 et GRN02 (même déplacement U_x nœud à nœud par exemple) **on ne peut pas écrire** :

```
LIAISON_DDL = _F ( GROUP_NO   = ( 'GRN01' , 'GRN02' ),
                    DDL        = ( 'DX' , 'DX' ),
                    COEF_MULT  = ( 1. , -1. ),
                    COEF_IMPO  = 0. ,
                    )
```

Cette écriture n'a de sens que si GRN01 et GRN02 ne contiennent chacun qu'un seul nœud. Il faudra dans le cas ci-dessus expliciter chaque relation linéaire, nœud par nœud, ou utiliser LIAISON_GROUP [§4.14] qui permet de condenser l'écriture de mêmes relations linéaires entre deux groupes de nœuds en vis-à-vis.

4.12 Mot-clé LIAISON_OBLIQUE

4.12.1 But

Mot-clé facteur utilisable pour appliquer, à des nœuds ou des groupes de nœuds, la même valeur de déplacement définie composante par composante dans un repère oblique quelconque.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.12.2 Syntaxe

- pour AFFE_CHAR_MECA

```
| LIAISON_OBLIQUE =_F ( ♦ / NOEUD = no , [noeud]
                        / GROUP_NO = gno , [gr_noeud]
                        ♦ | DX = ux , [R]
                        | DY = uy , [R]
                        | DZ = uz , [R]
                        | DRX =  $\theta_x$  , [R]
                        | DRY =  $\theta_y$  , [R]
                        | DRZ =  $\theta_z$  , [R]
                        ♦ ANGL_NAUT = ( $\alpha, \beta, \gamma$ ) , [l_R]
                        )
```

- pour AFFE_CHAR_MECA_F

```
| LIAISON_OBLIQUE =_F ( ♦ / NOEUD = no , [noeud]
                        / GROUP_NO = gno , [gr_noeud]
                        ♦ | DX = uxf , [fonction]
                        | DY = uyf , [fonction]
                        | DZ = uzf , [fonction]
                        | DRX =  $\theta_{xf}$  , [fonction]
                        | DRY =  $\theta_{yf}$  , [fonction]
                        | DRZ =  $\theta_{zf}$  , [fonction]
                        ♦ ANGL_NAUT = ( $\alpha \ \beta \ \gamma$ ) , [l_R]
                        )
```

4.12.3 Opérandes

| LIAISON_OBLIQUE

- DX = ux ou uxf
- DY = uy ou uyf
- DZ = uz ou uzf

Valeur de la composante de déplacement en translation dans le repère oblique imposée sur les nœuds spécifiés

Uniquement si les nœuds spécifiés appartiennent à des éléments discrets de translation-rotation, de poutre ou de coque.

- DRX = θ_x ou θ_{xf}
- DRY = θ_y ou θ_{yf}
- DRZ = θ_z ou θ_{zf}

Valeur de la composante de déplacement en rotation dans le repère oblique imposée sur les nœuds spécifiés

- ♦ ANGL_NAUT = (α, β, γ),

Les angles nautiques $\alpha \ \beta \ \gamma$ définis en **degrés**, sont les angles permettant de passer du repère GLOBAL de définition des coordonnées des nœuds à un repère oblique quelconque (voir AFFE_CARA_ELEM [U4.42.01]).

4.12.4 Vérification

On vérifie que le `ddl` spécifié existe en ce nœud pour les éléments affectés dans le `MODELE` aux mailles qui contiennent le nœud.

4.12.5 Limitation

Dans une occurrence du mot-clé `facteur`, on ne peut introduire pour l'instant qu'un seul nœud ou un seul groupe de nœuds ne contenant qu'un seul nœud.

4.13 Mot-clé LIAISON_GROUP

4.13.1 But

Mot-clé facteur utilisable pour définir la même relation linéaire entre certains degrés de liberté de couples de nœuds, ces couples de nœuds étant obtenus en mettant en vis-à-vis deux listes de mailles ou de nœuds [§4.14.5].

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.13.2 Syntaxe

- pour AFFE_CHAR_MECA

```

LIAISON_GROUP=_F (  ♦ / ♦ / MAILLE_1      = lma1, [l_maille]
                    /   / GROUP_MA_1      = lgma1,
[l_gr_maille]
                    ♦ / MAILLE_2      = lma2, [l_maille]
                    /   / GROUP_MA_2      = lgma2,
[l_gr_maille]

                    / ♦ / NOEUD_1       = lno1, [l_noeud]
                    /   / GROUP_NO_1     = lgno1, [l_gr_noeud]
                    ♦ / NOEUD_2       = lno2, [l_noeud]
                    /   / GROUP_NO_2     = lgno2, [l_gr_noeud]

♦ / SANS_NOEUD       = lno , [l_noeud]
  / SANS_GROUP_NO    = lgno, [l_gr_noeud]

♦ DDL_1 = / | 'DX',
           | 'DY',
           | 'DZ',
           | 'DRX',
           | 'DRY',
           | 'DRZ',
           / 'DNOR',
♦ DDL_2 = / | 'DX',
           | 'DY',
           | 'DZ',
           | 'DRX',
           | 'DRY',
           | 'DRZ',
           / 'DNOR',

♦ COEF_MULT_1 = α1i , [l_R]
♦ COEF_MULT_2 = α2i , [l_R]
♦ COEF_IMPO   = β , [R]
♦ SOMMET      = 'OUI',
♦ CENTRE      = lr , [l_R]
♦ ANGL_NAUT   = lr , [l_R]
♦ TRAN        = lr , [l_R]
)

```

```

• pour AFFE_CHAR_MECA_F
  LIAISON_GROUP=_F ( ♦ / ♦ / MAILLE_1 = lma1, [l_maille]
                    / GROUP_MA_1 = lgma1,
[l_gr_maille]
                    ♦ / MAILLE_2 = lma2, [l_maille]
                    / GROUP_MA_2 = lgma2,
[l_gr_maille]
                    / ♦ / NOEUD_1 = lno1, [l_noeud]
                    / GROUP_NO_1 = lgnol, [l_gr_noeud]
                    ♦ / NOEUD_2 = lno2, [l_noeud]
                    / GROUP_NO_2 = lgnol, [l_gr_noeud]
♦ / SANS_NOEUD = lno, [l_noeud]
  / SANS_GROUP_NO = lgnol, [l_gr_noeud]
♦ DDL_1 = / | 'DX',
            | 'DY',
            | 'DZ',
            | 'DRX',
            | 'DRY',
            | 'DRZ',
            / 'DNOR',
♦ DDL_2 = / | 'DX',
            | 'DY',
            | 'DZ',
            | 'DRX',
            | 'DRY',
            | 'DRZ',
            / 'DNOR',
♦ COEF_MULT_1 = α1i, [l_R]
♦ COEF_MULT_2 = α2i, [l_R]
♦ COEF_IMPO = βf, [fonction]
♦ SOMMET = 'OUI',
♦ CENTRE = lr, [l_R]
♦ ANGL_NAUT = lr, [l_R]
♦ TRAN = lr, [l_R]
)

```

4.13.3 Opérandes

```

/ ♦ / GROUP_MA_1 =
/ MAILLE_1 =

```

Ces opérandes définissent la première liste de mailles en relation (notée Γ_1).

```

♦ / GROUP_MA_2 =
/ MAILLE_2 =

```

Ces opérandes définissent la deuxième liste de mailles en relation (notée Γ_2).

```

♦ / GROUP_NO_1 =
/ NOEUD_1 =

```

Ces opérandes définissent la première liste de nœuds en relation.

```

♦ / GROUP_NO_2 =
/ NOEUD_2 =

```

Ces opérandes définissent la deuxième liste de nœuds en relation.

Les deux listes doivent avoir la même longueur.

Titre : **Opérateurs AFFE_CHAR_MECA et AFFE_CHAR_MECA_F**
Auteur(s) : **X. DESROCHES**

Date : 29/05/07
Clé : U4.44.01-I Page : 22/94

◇ / SANS_GROUP_NO =
/ SANS_NOEUD =

Ces opérandes permettent de supprimer de la liste des couples de nœuds en vis-à-vis [§4.14.5] tous les couples dont au moins un des nœuds appartient à la liste de nœuds décrite par ces opérandes.

Cela permet d'éviter l'accumulation de relations linéaires sur un même nœud au cours de différentes répétitions du mot-clé facteur LIAISON_GROUP, ce qui conduit la plupart du temps à une matrice singulière.

◆ DDL_1 (_2) =

L'argument de DDL_1 ou _2 doit être une liste de textes pris parmi (DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ') ou 'DNOR'.

◆ COEF_MULT_1 (resp. COEF_MULT_2) =

Liste de réels exactement dimensionnée au nombre de degrés de liberté déclarés dans DDL_1 (resp. DDL_2) correspondant aux coefficients multiplicateurs de la relation linéaire.

◆ COEF_IMPO =

Coefficient de blocage de la relation linéaire :

β : réel pour AFFE_CHAR_MECA

βf : fonction pour AFFE_CHAR_MECA_F

Les opérandes CENTRE / ANGL_NAUT / TRAN permettent de définir une transformation virtuelle (rotation et/ou translation) approximative de Γ_1 en Γ_2 afin d'assurer la bijectivité de la fonction vis-à-vis [§4.14.5].

La commande effectue d'abord la rotation, puis la translation.

◇ CENTRE = coordonnées du centre de rotation (dans le repère global)

◇ ANGL_NAUT = angles nautiques définissant la rotation (en degrés)

◇ TRAN = composantes du vecteur translation

Remarques :

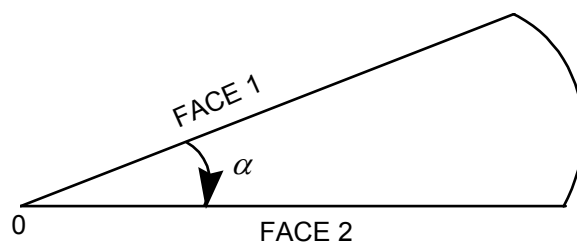
- On vérifie que les ddl spécifiés dans ces opérandes existent pour chacun des nœuds des éléments affectés dans le *MODELE* aux mailles qui contiennent le nœud,
- pour utiliser l'argument 'DNOR', il est obligatoire d'avoir déclaré les bords à l'aide de mailles et que le calcul d'une normale sur ces mailles soit possible.

◇ SOMMET = 'OUI'

Lorsque les mailles de bord sont quadratiques (donc des SEG3) l'utilisation de SOMMET : 'OUI' force l'algorithme d'appariement à associer les sommets des SEG3 à d'autres sommets, et les milieux des SEG3 à d'autres milieux. Dans le cas de maillages fins, cela permet dans certains cas d'éviter les problèmes de conflits de vis-à-vis.

4.13.4 Exemple d'utilisation

On veut imposer une condition de répétitivité cyclique (même déplacement normal) entre la FACE 1 et la FACE 2 de la géométrie ci-dessous :



Supposons que FACE 1 (resp. FACE 2) soit composée de la liste de mailles `lma1` (resp. `lma2`).

On veut écrire les relations linéaires suivantes :

$\forall N_i^1$ nœud de la face 1 de vis-à-vis N_i^2

$$\mathbf{u.n}(N_i^1) = \mathbf{u.n}(N_i^2) \quad \forall i = 1, \dots, \text{nbno}$$

où `nbno` est le nombre de nœuds de la face 1 (et de la face 2).

Les données de `LIAISON_GROUP` s'écriront :

```
LIAISON_GROUP=_F (  MAILLE_1      =  lma1,
                     MAILLE_2      =  lma2,
                     DDL_1         =  'DNOR',
                     DDL_2         =  'DNOR',
                     COEF_MULT_1   =  1.,
                     COEF_MULT_2   =  -1.,
                     COEF_IMPO     =  0,
                     CENTRE        =  (X0,Y0,Z0),
                     ANGL_NAUT     =  (α,0.,0.),
                     )
```

4.13.5 Détermination des couples de nœuds en vis-à-vis

Elle se fait de la même façon que dans `AFFE_CHAR_THER`.

Dans un premier temps, on établit les deux listes de nœuds à mettre en vis-à-vis (ie à apparier), pour chaque occurrence du mot-clé facteur `LIAISON_GROUP` :

- pour les mots-clés `GROUP_NO_1` et `GROUP_NO_2`, ce sont les nœuds constituant les groupes de nœuds,
- pour les mots-clés `GROUP_MA_1` et `GROUP_MA_2`, ce sont les nœuds des mailles constituant les groupes de mailles.

Les redondances étant éliminées, les deux listes de nœuds obtenues doivent avoir la même longueur.

La détermination des couples de nœuds en vis-à-vis se fait en plusieurs étapes :

- pour chaque nœud `N1` de la première liste, on cherche le nœud image `N2 = f(N1)` de la deuxième liste. Si `f` n'est pas injective (un nœud `N2` est l'image de deux nœuds distincts `N1` et `N1'`), le message d'erreur suivant est émis :

```
<F> <MODELISA8_85> CONFLIT DANS LES VIS-A-VIS DES NOEUDS
LE NOEUD N2 EST LE VIS-A-VIS DES NOEUDS N1 ET N1'
```

- pour chaque nœud `N2` de la deuxième liste, on cherche le nœud image `N1 = g(N2)` de la première liste. Si `g` n'est pas injective (un nœud `N1` est l'image de deux nœuds distincts `N2` et `N2'`), le message d'erreur suivant est émis :

```
<F> <MODELISA8_85> CONFLIT DANS LES VIS-A-VIS DES NOEUDS
LE NOEUD N1 EST LE VIS-A-VIS DES NOEUDS N2 ET N2'
```

- on vérifie que `g = f-1`, c'est-à-dire que les couples obtenus par les étapes a) et b) sont les mêmes (on veut avoir une bijection `f` entre les deux listes de nœuds). Si `f` n'est pas surjective, le message d'erreur suivant est émis :

```
<F> <MODELISA8_88> CONFLIT DANS LES VIS-A-VIS GENERES
SUCCESSIVEMENT A PARTIR DES LISTES LIST1 ET LIST2
LE NOEUD DE LA PREMIERE LISTE N1 N'EST L'IMAGE D'AUCUN NOEUD PAR LA
CORRESPONDANCE INVERSE
```

Pour un nœud N donné, on appelle nœud image $f(N)$ le nœud de l'autre liste de nœuds qui réalise le minimum de la distance avec N . Pour faciliter l'appariement, notamment dans le cas de géométries particulières (où les frontières Γ_1 et Γ_2 pourraient "presque" se déduire l'une de l'autre par la composition d'une translation et d'une rotation), on offre la possibilité de faire une transformation géométrique virtuelle du premier groupe de nœuds (translation et rotation avant de calculer les distances (mots-clés `TRAN`, `CENTRE` et `ANGL_NAUT`)).

Pour chaque occurrence du mot-clé facteur `LIAISON_GROUP`, on construit ainsi la liste des nouveaux couples en vis-à-vis. Lorsqu'on a balayé toutes les occurrences, on supprime de la liste les couples en double.

Remarque :

Dans les couples de nœuds en vis-à-vis, l'ordre des nœuds est important. Si pour la première occurrence de `LIAISON_GROUP`, un nœud N appartenait au premier groupe de nœuds et un nœud M au deuxième groupe de nœud, et que pour la seconde occurrence de `LIAISON_GROUP`, c'est l'inverse, on obtiendra à l'issue de l'appariement les couples (N, M) et (M, N) . Ils ne seront pas éliminés lors de la détection des redondances ; par contre, la matrice obtenue sera singulière. Ainsi, on conseille de garder la même logique lors de la description des bords en vis-à-vis.

4.14 Mot-clé LIAISON_MAIL

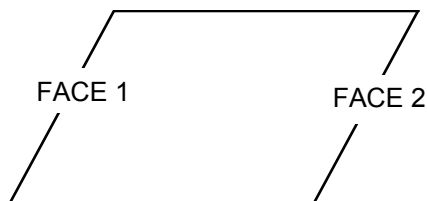
4.14.1 But

Mot-clé facteur utilisable pour définir des relations linéaires permettant de "recoller" deux "bords" d'une structure.

La particularité de ce mot-clé (par rapport à LIAISON_GROUP par exemple) est de permettre de lier les déplacements de nœuds sans contrainte sur le maillage. Les maillages de FACE 1 et FACE 2 peuvent être incompatibles.

Exemples :

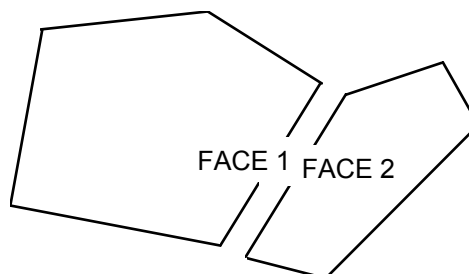
a) une condition de périodicité (étude d'une cellule d'homogénéisation)



b) une condition de répétitivité cyclique



c) une condition de simple recollement



Dans la suite de ce paragraphe, on parlera de la face "esclave" (FACE 2) et de la face "maître" (FACE 1).

Le "recollement" des 2 faces se fera par écriture de relations linéaires entre les ddls des 2 faces.

Les déplacements des nœuds de la face esclave seront reliés aux déplacements de leurs projections sur la face maître. Pour chaque nœud de la face esclave, on écrira 2 (en 2D) ou 3 (en 3D) relations linéaires.

Si FACE 1 et FACE 2 ne sont pas géométriquement confondues mais qu'il existe une isométrie (rotation + translation) entre les deux, l'utilisateur doit définir cette isométrie (celle qui transforme FACE 2 en FACE 1).

Une application de cette fonctionnalité est par exemple le recollement d'un maillage formé d'éléments linéaires (P1) sur un autre maillage quadratique (P2). Dans ce cas il est plutôt conseillé de choisir comme face "esclave" la face quadratique.

4.14.2 Syntaxe (dans AFPE_CHAR_MECA seulement)

```

LIAISON_MAIL =_F (
  ♦ | GROUP_NO_ESCL = lgn02 , [l_gr_noeud]
    | NOEUD_ESCL = lno2 , [l_noeud]
    | GROUP_MA_ESCL = lgma2 ,
  [l_gr_maille]
    | MAILLE_ESCL = lma2 , [l_maille]
  ♦ | GROUP_MA_MAIT = lgma1 ,
  [l_gr_maille]
    | MAILLE_MAIT = lma1 , [l_maille]
  ◇ | ♦ CENTRE = (xc, yc, [zc]), [l_R]
    | ♦ ANGL_NAUT = (alpha, [beta, gamma]), [l_R]
    | ♦ TRAN = (tx, ty, [tz]), [l_R]
  ◇ | ♦ DDL_MAIT = 'DNOR',
    | ♦ DDL_ESCL = 'DNOR',
  ◇ ELIM_MULT = / 'NON', [DEFAULT]
    / 'OUI',
)

```

4.14.3 Opérands

4.14.3.1 GROUP_NO_ESCL / NOEUD_ESCL / GROUP_MA_ESCL / MAILLE_ESCL

Ces mots-clés permettent de définir l'ensemble des nœuds de la face esclave. On prend tous les nœuds spécifiés par les mots-clés GROUP_NO_ESCL et NOEUD_ESCL plus tous les nœuds portés par les mailles spécifiées par les mots-clés GROUP_MA_ESCL et MAILLE_ESCL.

Remarque :

Quand on veut ne recoller que les déplacements normaux des faces (cf. mots-clés DDL_MAIT et DDL_ESCL), il faut pouvoir déterminer la direction normale des faces. La direction normale est calculée sur la face esclave. Il faut donc dans ce cas utiliser les mots-clés GROUP_MA_ESCL et MAILLE_ESCL avec des mailles de type "facette".

4.14.3.2 GROUP_MA_MAIT / MAILLE_MAIT

Ces mots-clés permettent de définir l'ensemble des mailles où l'on cherchera les vis-à-vis des nœuds de la face esclave.

Attention :

En 3D, il ne faut pas donner des mailles de surface, mais les mailles volumiques adjacentes à la face. Les mailles spécifiées sont des "candidates" pour la recherche des points vis-à-vis. On peut en donner trop, cela n'est pas gênant.

De la même façon, en 2D, les mailles "maîtres" doivent être surfaciques (QUAD, TRIA) et non linéiques

4.14.3.3 CENTRE / ANGL_NAUT / TRAN

Ces mots-clés permettent de définir la transformation géométrique (rotation et/ou translation) permettant de passer de la face esclave à la face maître.

Si ces mots-clés sont absents, c'est que la transformation géométrique est "l'identité" c'est-à-dire que les faces maître et esclave sont géométriquement confondues.

Il faut noter que le programme effectue d'abord la rotation et ensuite la translation. Attention : le sens de la transformation est esclave --> maître.

4.14.3.4 DDL_MAIT / DDL_ESCL

Si l'on veut ne recoller que les déplacements normaux aux faces, il faut spécifier :

```

DDL_MAIT = 'DNOR'
DDL_ESCL = 'DNOR'

```

Remarque :

La direction normale est calculée sur la face esclave (il faut donner des mailles de facette). Cette direction normale est transformée par l'éventuelle rotation de la transformation géométrique pour déterminer la direction normale sur la face maître.

4.14.3.5 Remarques

- Le mot-clé `LIAISON_MAIL` est en principe fait pour relier 2 surfaces a priori disjointes. Parfois ce n'est pas le cas et un nœud esclave peut appartenir à l'une des mailles maîtres. La relation linéaire que cherche à écrire le problème devient une tautologie ($X=X$), ce qui conduit à un pivot nul lors de la factorisation.

Pour éviter ce problème, on n'écrit pas les relations reliant un nœud esclave à sa maille maître si :

- ce nœud appartient à la connectivité de la maille
- les mots clés `CENTRE`, `ANGL_TRAN`, `TRAN` n'ont pas été utilisés

- Il faut être conscient que pour chaque occurrence de `LIAISON_MAIL`, on relie TOUS les nœuds esclaves aux mailles maîtres même si les distances de projection sont importantes (on émet toutefois des alarmes dans ce cas).

Ce serait une erreur d'écrire :

```
LIAISON_MAIL = ( _F(GROUP_MA_ESCL='GE', GROUP_MA_MAIT='GM1'),  
                  _F(GROUP_MA_ESCL='GE', GROUP_MA_MAIT='GM2'))
```

en pensant que le programme triera dans `GE` les nœuds proches de `GM1` et ceux proches de `GM2`. Dans cet exemple, les nœuds de `GE` seront éliminés 2 fois et on peut s'attendre à un problème de pivot nul lors de la factorisation.

L'utilisateur doit écrire :

```
LIAISON_MAIL = _F(GROUP_MA_ESCL='GE', GROUP_MA_MAIT=('GM1','GM2'))
```

4.14.3.6 `ELIM_MULT= 'OUI' / 'NON' [DEFAULT]`

Ce mot clé sert à résoudre le problème qui peut se poser lorsque l'on recolle plusieurs surfaces esclaves adjacentes (i.e. qui ont un ou plusieurs nœuds communs).

Imaginons par exemple que l'on écrive (en 2D) :

```
LIAISON_MAIL=(  
    _F(GROUP_MA_ESCL='LIGNE_AB', GROUP_MA_MAIT=...)  
    _F(GROUP_MA_ESCL='LIGNE_BC', GROUP_MA_MAIT=...)
```

Si l'utilisateur force `ELIM_MULT='OUI'`, le programme traitera chaque occurrence de `LIAISON_MAIL` de façon indépendantes. Le nœud B, appartenant à `LIGNE_AB` et `LIGNE_BC` sera éliminé 2 fois et il est malheureusement probable que le calcul s'arrêtera lors de la factorisation de la matrice avec le message "Pivot presque nul ..." car les relations linéaires générées par `LIAISON_MAILLE` sont redondantes.

La plupart du temps, le défaut (`ELIM_MULT='NON'`) est le bon choix. Le seul cas où l'utilisateur pourrait utiliser `ELIM_MULT='OUI'` est celui de l'utilisation du mot clé `DDL_ESCL='DNOR'` car si dans les 2 occurrences, les normales "esclaves" ne sont les mêmes, l'élimination n'est pas redondante.

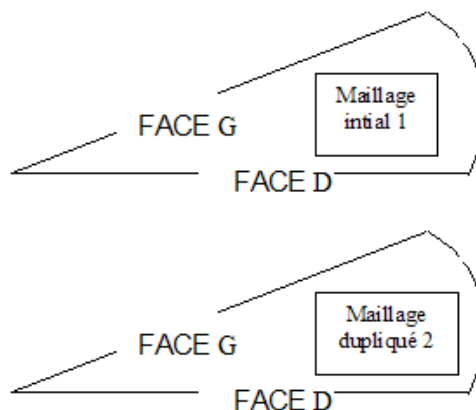
4.15 Mot-clé LIAISON_CYCL

4.15.1 But

Mot-clé facteur utilisable pour définir les relations linéaires permettant d'imposer des conditions de symétrie cyclique avec prise en compte d'un déphasage. Il est principalement dédié à être utilisé dans le cadre restrictif du calcul dynamique avec symétrie cyclique.

La particularité de ce mot-clé (à l'image de LIAISON_MAIL) est de permettre de lier les déplacements de nœuds sans contrainte sur le maillage. Les maillages de FACE G et FACE D peuvent être incompatibles.

La condition de répétitivité cyclique appliquée dans le cadre de la dynamique est basée sur la méthode de duplication de maillage. L'opérateur part donc sur le postulat que le maillage initial d'un secteur est dupliqué en deux maillages identiques à l'image de la figure suivante.



Dans la suite de ce paragraphe, on parlera de la face "esclave" et de la face "maître". Le "recollement" des 2 faces se fera par écriture de relations linéaires entre les ddls des 2 faces.

Les déplacements des nœuds de la face esclave seront reliés aux déplacements de leurs projections sur la face maître. Pour chaque nœud de la face esclave, on écrira 2 (en 2D) ou 3 (en 3D) relations linéaires.

Si FACE G et FACE D ne sont pas géométriquement confondues mais qu'il existe une isométrie (rotation + translation) entre les deux, l'utilisateur doit définir cette isométrie (celle qui transforme FACE G en FACE D).

Remarque :

Une application de cette fonctionnalité est par exemple le recollement d'un maillage formé d'éléments linéaires (P1) sur un autre maillage quadratique (P2). Dans ce cas il est plutôt conseillé de choisir comme face "esclave" la face quadratique.

L'expression de la condition de symétrie cyclique pour un déphasage inter-secteur β donné et en considérant G comme l'interface esclave est la suivante :

$$\begin{bmatrix} \mathbf{q}_g^1 \\ \mathbf{q}_g^2 \end{bmatrix} = \begin{bmatrix} \cos(\beta) & \sin(\beta) \\ -\sin(\beta) & \cos(\beta) \end{bmatrix} \begin{bmatrix} \mathbf{q}_d^1 \\ \mathbf{q}_d^2 \end{bmatrix}$$

Afin d'écrire les relations linéaires permettant de prendre en compte cette condition, il est nécessaire de donner **deux** occurrences du mot clé facteur LIAISON_CYCL :

- La première permet de lier les ddls de la face G du maillage 1 avec la face D du même maillage et la face D du maillage 2. Les coefficients ($\cos(\beta)$ et $\sin(\beta)$) doivent être renseignés par les mots clé COEF_MAIT1, COEF_MAIT2.
- La seconde permet de lier les ddls de la face G du maillage 2 avec la face D du même maillage et la face D du maillage 1. Les coefficients ($-\sin(\beta)$ et $\cos(\beta)$) doivent être renseignés par les mots clé COEF_MAIT1, COEF_MAIT2

4.15.2 Syntaxe (dans AFFE_CHAR_MECA seulement)

```
LIAISON_CYCL =_F (
  ♦ | GROUP_NO_ESCL      = lgrno2 ,          [lgrnoeud]
    | NOEUD_ESCL         = lno2 ,            [lnoeud]
    | GROUP_MA_ESCL      = lgma2 ,
  [lgrmaillage]
    | MAILLE_ESCL        = lma2 ,            [lmaillage]
  ♦ | GROUP_MA_MAIT1     = lgma1 ,
  [lgrmaillage]
    | MAILLE_MAIT1       = lma1 ,            [lmaillage]
    | GROUP_MA_MAIT2     = lgma1 ,
  [lgrmaillage]
    | MAILLE_MAIT2       = lma1 ,            [lmaillage]
  ◇ | ♦ CENTRE           = (xc, yc, [zc]),    [l_R]
    | ♦ ANGL_NAUT        = (alpha, [beta,gamma]), [l_R]
    | ♦ TRAN             = (tx, ty, [tz]),    [l_R]
  ◇ | ♦ COEF_MAIT1       = α ,               [R]
    | ♦ COEF_MAIT2       = β ,               [R]
    | ♦ COEF_ESCL        = χ ,               [R]
  ◇ | ♦ DDL_MAIT         = 'DNOR' ,
    | ♦ DDL_ESCL         = 'DNOR' ,
  )
```

4.15.3 Opérandes

4.15.4 GROUP_NO_ESCL / NOEUD_ESCL / GROUP_MA_ESCL / MAILLE_ESCL

Ces mots-clés permettent de définir l'ensemble des nœuds de la face esclave. On prend tous les nœuds spécifiés par les mots-clés GROUP_NO_ESCL et NOEUD_ESCL plus tous les nœuds portés par les mailles spécifiées par les mots-clés GROUP_MA_ESCL et MAILLE_ESCL.

Remarque :

Quand on veut ne recoller que les déplacements normaux des faces (cf. mots-clés DDL_MAIT et DDL_ESCL), il faut pouvoir déterminer la direction normale des faces. La direction normale est calculée sur la face esclave. Il faut donc dans ce cas utiliser les mots-clés GROUP_MA_ESCL et MAILLE_ESCL avec des mailles de type "facette".

4.15.5 GROUP_MA_MAIT1 / MAILLE_MAIT1

Ces mots-clés permettent de définir l'ensemble des mailles maîtres du maillage 1 (ou 2) où l'on cherchera les vis-à-vis des nœuds de la face esclave du maillage 1 ou 2.

Attention :

En 3D, il ne faut pas donner des mailles de surface, mais les mailles volumiques adjacentes à la face. Les mailles spécifiées sont des "candidates" pour la recherche des points vis-à-vis. On peut en donner trop, cela n'est pas gênant.

De la même façon, en 2D, les mailles "maîtres" doivent être surfaciques (QUAD, TRIA) et non linéiques

4.15.6 GROUP_MA_MAIT2 / MAILLE_MAIT2

Ces mots-clés permettent de définir l'ensemble des mailles de 1 (ou 2) où l'on cherchera les vis-à-vis des nœuds de la face esclave du maillage 1 ou 2.

Attention :

En 3D, il ne faut pas donner des mailles de surface, mais les mailles volumiques adjacentes à la face. Les mailles spécifiées sont des "candidates" pour la recherche des points vis-à-vis. On peut en donner trop, cela n'est pas gênant.

De la même façon, en 2D, les mailles "maîtres" doivent être surfaciques (QUAD, TRIA) et non linéiques

4.15.7 CENTRE / ANGL_NAUT / TRAN

Ces mots-clés permettent de définir la transformation géométrique (rotation et/ou translation) permettant de passer de la face esclave à la face maître.

Si ces mots-clés sont absents, c'est que la transformation géométrique est "l'identité" c'est-à-dire que les faces maître et esclave sont géométriquement confondues.

Il faut noter que le programme effectue d'abord la rotation et ensuite la translation. Attention : le sens de la transformation est esclave --> maître.

4.15.8 COEF_MAIT1 / COEF_MAIT2 / COEF_ESCL

Ces mots-clés permettent de définir les coefficients de la relation linéaire à appliquer, dans le cas de la symétrie cyclique il s'agit des cosinus et sinus de l'angle de déphasage inter-secteur considéré. Ces coefficients doivent donc être cohérents avec la définition des interfaces maîtres et esclaves. Le coefficient COEF_ESCL permet de passer un coefficient devant les ddl esclaves.

Par exemple :

$$\text{COEF_ESCL}(\mathbf{q}_g^1) = [\text{COEF_MAIT1} \times \text{COEF_MAIT2}] \begin{bmatrix} \mathbf{q}_d^1 \\ \mathbf{q}_d^2 \end{bmatrix} = [\cos(\beta) \cdot \sin(\beta)] \begin{bmatrix} \mathbf{q}_d^1 \\ \mathbf{q}_d^2 \end{bmatrix}$$

4.15.9 DDL_MAIT / DDL_ESCL

Si l'on veut ne recoller que les déplacements normaux aux faces, il faut spécifier :

DDL_MAIT = 'DNOR'
DDL_ESCL = 'DNOR'

Remarque :

La direction normale est calculée sur la face esclave (il faut donner des mailles de facette). Cette direction normale est transformée par l'éventuelle rotation de la transformation géométrique pour déterminer la direction normale sur la face maître.

4.16 Mot-clé CONTACT

4.16.1 But

Mot-clé facteur utilisable pour décrire les zones soumises à des conditions de contact unilatéral avec ou sans frottement. Ces zones (une pour chaque occurrence du mot-clé facteur), comprennent chacune deux surfaces pouvant entrer en contact qui sont décrites par la donnée des mailles qui les constituent.

Les ensembles de mailles potentiellement en contact sont : surfaciques et linéiques en dimension 3 (QUAD9, QUAD8, QUAD4 et TRIA7, TRIA6, TRIA3 et SEG2, SEG3), linéiques et concentrés en dimension 2 (SEG2, SEG3 et POI1).

Attention :

En dimension 3, le traitement du contact avec des mailles surfaciques quadratiques (QUAD8 et TRIA6 ou QUAD9 et TRIA7 associés à la modélisation COQUE_3D) nécessite de lier les nœuds milieux des côtés aux sommets de façon à avoir des résultats corrects. Cette opération est faite automatiquement dans le Code. Néanmoins, pour les calculs 3D milieux continus avec des éléments quadratiques, l'utilisation d'éléments HEXA27 (à faces QUAD9) est fortement conseillée.

Il existe une version modifiée de la projection avec des mailles surfaciques quadratiques dont les polynômes sont incomplets. Cette projection (utilisable avec des HEXA20 et des QUAD8) assure a minima que les réactions de contact sont cohérentes. On l'active via l'option PROJECTION = 'QUADRATIQUE'.

Les structures étudiées peuvent subir de grands glissements l'une par rapport à l'autre. Cette formulation, contact nodal ou nœud-facette en géométrie réactualisée, avec réactualisation de l'appariement pilotée par l'utilisateur, est décrite dans le document [R5.03.50] et implantée dans les opérateurs STAT_NON_LINE [U4.51.03] et DYNA_NON_LINE [U4.53.01].

Avant de faire un calcul avec contact utilisant le mot-clé CONTACT, il est **indispensable** d'avoir lu la documentation de référence [R5.03.50] et la note HI-75/97/034/A de conseils aux utilisateurs, qui explicitent le rôle de la plupart des mots-clés décrits ci-dessous et donnent les précautions d'utilisation.

Il est aussi recommandé de consulter la documentation d'utilisation du contact [U2.04.04].

Enfin, à noter que le solveur GCPC ne doit pas être utilisé avec le contact.

4.16.2 Syntaxe (AFPE_CHAR_MECA(_F))

Il existe plusieurs méthodes pour traiter les problèmes de contact/frottement. On a séparé ci-dessous les opérands propres à chacune.

```
CONTACT = _F( ◊ / MAILLE_MAIT = lma1, [l_maille]
               / GROUP_MA_MAIT = lgma1, [l_gr_maille]
               ♦ / MAILLE_ESCL = lma2, [l_maille]
               / GROUP_MA_ESCL = lgma2, [l_gr_maille]
               ◊ APPARIEMENT = / 'MAIT_ESCL', [DEFAULT]
                               / 'NODAL',
                               / 'MAIT_ESCL_SYME',
               ◊ RECHERCHE = / 'NOEUD_VOISIN', [DEFAULT]
                               / 'NOEUD_BOUCLE',
               ◊ NORMALE = / 'MAIT', [DEFAULT]
                           / 'MAIT_ESCL',
               ◊ LISSAGE = / 'NON', [DEFAULT]
                           / 'OUI',
               ◊ PROJECTION = / 'LINEAIRE', [DEFAULT]
                              / 'QUADRATIQUE',
               ◊ TOLE_PROJ_EXT = / 0.50, [DEFAULT]
                               / tole, [R]
               ◊ TOLE_PROJ_INT = / 0.001, [DEFAULT]
                               / tole, [R]
               / ◊ METHODE = / 'CONTRAINTTE', [DEFAULT]
```

Titre : Opérateurs AFPE_CHAR_MECA et AFPE_CHAR_MECA_F
Auteur(s) : X. DESROCHES

Clé : U4.44.01-1

Date : 29/05/07
Page : 32/94

```

◇ FROTTEMENT = / 'SANS' , [DEFAULT]
◇ REAC_GEOM = / 'AUTOMATIQUE' , [DEFAULT]
              / 'SANS' ,
              / 'CONTROLE' ,
    ◆ NB_REAC_GEOM = n, [I]
◇ SANS_NOEUD = lno, [l_noeud]
◇ SANS_GROUP_NO = lgno, [l_gr_noeud]
◇ SANS_NOEUD_QUAD = / 'OUI' , [DEFAULT]
                  / 'NON' ,
◇ GLISSIERE = / 'NON' , [DEFAULT]
              / 'OUI' ,
    ◇ ALARME_JEU / 0.0, [DEFAULT]
                / alarm_jeu, [R]
◇ DIST_MAIT = r, [R]([fonction])
◇ DIST_ESCL = r, [R]([fonction])
◇ VECT_NORM_ESCL = (Vx,Vy,Vz), [l_R]
◇ VECT_ORIE_POU = (Yx,Yy,Yz), [l_R]
◇ NB_RESOL = / 10, [DEFAULT]
              / n, [I]
◇ STOP_SINGULIER = / 'OUI' , [DEFAULT]
                  / 'NON' ,

/ ◇ METHODE = 'LAGRANGIEN' ,
  ◇ SANS_NOEUD = lno, [l_noeud]
  ◇ SANS_GROUP_NO = lgno, [l_gr_noeud]
  ◇ SANS_NOEUD_QUAD = / 'OUI' , [DEFAULT]
                    / 'NON' ,
  ◇ DIST_MAIT = r, [R]([fonction])
  ◇ DIST_ESCL = r, [R]([fonction])
  ◇ STOP_SINGULIER = / 'OUI' , [DEFAULT]
                    / 'NON' ,
  ◇ REAC_GEOM = / 'AUTOMATIQUE' , [DEFAULT]
                / 'SANS' ,
                / 'CONTROLE' ,
    ◆ NB_REAC_GEOM = n, [I]
  ◇ FROTTEMENT = / 'SANS' , [DEFAULT]
                / 'COULOMB' ,
    ◆ COULOMB = r, [R]
  ◇ COEF_MATR_FROT = / 0., [DEFAULT]
                    / r, [R]
  ◇ VECT_Y = (Yx,Yy,Yz), [R]
  ◇ VECT_ORIE_POU = (Yx,Yy,Yz), [l_R]
  ◇ ITER_MULT_MAXI = / 4, [DEFAULT]
                    / iter, [I]
  ◇ NB_RESOL = / 10, [DEFAULT]
              / n, [I]

```


Titre : Opérateurs AFFE_CHAR_MECA et AFFE_CHAR_MECA_F
Auteur(s) : X. DESROCHES

Clé : U4.44.01-I

Date : 29/05/07
Page : 33/94

```

/  ◇ METHODE = 'PENALISATION',
    ◇ E_N = r, [R]
    ◇ SANS_NOEUD = lno, [l_noeud]
    ◇ SANS_GROUP_NO = lgno, [l_gr_noeud]
    ◇ SANS_NOEUD_QUAD = / 'OUI', [DEFAULT]
                        / 'NON',
    ◇ DIST_MAIT = r, [R]([fonction])
    ◇ DIST_ESCL = r, [R]([fonction])
    ◇ STOP_SINGULIER = / 'OUI', [DEFAULT]
                        / 'NON',
    ◇ REAC_GEOM = / 'AUTOMATIQUE', [DEFAULT]
                  / 'SANS',
                  / 'CONTROLE',
    ◆ NB_REAC_GEOM = n, [I]
    ◇ FROTTEMENT = / 'SANS', [DEFAULT]
                  / 'COULOMB',
    ◆ COULOMB = r, [R]
    ◇ COEF_MATR_FROT = / 0., [DEFAULT]
                       / r, [R]
    ◇ VECT_Y = (Yx,Yy,Yz), [R]
    ◇ E_T = r, [R]
    ◇ VECT_ORIE_POU = (Yx,Yy,Yz), [l_R]
    ◇ ITER_MULT_MAXI = / 4, [DEFAULT]
                       / iter, [I]
    ◇ NB_RESOL = / 10, [DEFAULT]
                 / n, [I]

/  ◇ METHODE = / 'GCP',
    ◆ RESI_ABSO = resi, [R]
    ◇ PRE_COND = / 'SANS', [DEFAULT]
                  / 'DIRICHLET'
    ◇ COEF_RESI = / 1., [DEFAULT]
                  / coef [R]
    ◇ RECH_LINEAIRE = / 'ADMISSIBLE' [DEFAULT]
                      / 'NON_ADMISSIBLE'
    ◇ REAC_ITER = / 3, [DEFAULT]
                  / reac
    ◇ FROTTEMENT = / 'SANS', [DEFAULT]
    ◇ REAC_GEOM = / 'AUTOMATIQUE', [DEFAULT]
                  / 'SANS',
                  / 'CONTROLE',
    ◆ NB_REAC_GEOM = n, [I]
    ◇ SANS_NOEUD = lno, [l_noeud]
    ◇ SANS_GROUP_NO = lgno, [l_gr_noeud]
    ◇ SANS_NOEUD_QUAD = / 'OUI', [DEFAULT]
                        / 'NON',
    ◇ DIST_MAIT = r, [R]([fonction])
    ◇ DIST_ESCL = r, [R]([fonction])
    ◇ VECT_NORM_ESCL = (Vx,Vy,Vz), [l_R]
    ◇ VECT_ORIE_POU = (Vx,Vy,Vz), [l_R]
    ◇ STOP_SINGULIER = / 'OUI', [DEFAULT]
                        / 'NON',

```

```

/  ◇ METHODE = 'CONTINUE',
    ◇ GLISSIERE = / 'NON', [DEFAULT]
                  / 'OUI',
    ◇ ALARME_JEU / 0.0, [DEFAULT]
                  / alarm_jeu, [R]
    ◇ DIST_MAIT = r, [R]([fonction])
    ◇ DIST_ESCL = r, [R]([fonction])
    ◇ COEF_REGU_CONT = / 100., [DEFAULT]
                      / r,
    ◇ ITER_GEOM_MAXI = / 2, [DEFAULT]
                      / i,
    ◇ ITER_CONT_MAXI = / 30, [DEFAULT]
                      / i,
    ◇ INTEGRATION = / 'NOEUD', [DEFAULT]
                    / 'GAUSS',
                    / 'SIMPSON',
                    / 'SIMPSON1',
                    / 'SIMPSON2',
                    / 'NCOTES',
                    / 'NCOTES1',
                    / 'NCOTES2',
    ◆ MODL_AXIS = / 'NON',
                  / 'OUI',
    ◇ FORMULATION = / 'DEPL', [DEFAULT]
                    / 'VITE',
    ◇ DIRE_APPA = / (0, 0, 0), [DEFAULT]
                  / (x, y, z), [R]
    ◇ RACCOR_LINE_QUAD = / 'NON', [DEFAULT]
                        / 'OUI',
    ◇ GROUP_NO_RACC = / lgrno, [l_gr_noeud]
    ◇ NOEUD_RACC = / lno, [l_noeud]
    ◇ FOND_FISSURE = / 'NON', [DEFAULT]
                    / 'OUI',
    ◇ GROUP_MA_FOND = / lgma, [l_gr_maille]
    ◇ MAILLE_FOND = / lma, [l_maille]
    ◇ GROUP_NO_FOND = / lgrno, [l_gr_noeud]
    ◇ NOEUD_FOND = / lno, [l_noeud]
    ◇ FROTTEMENT = / 'SANS', [DEFAULT]
                  / 'COULOMB',
    ◆ COULOMB = r, [R]
    ◇ COEF_REGU_FROT = / 100., [DEFAULT]
                      / r,
    ◇ SEUIL_INIT = / 0., [DEFAULT]
                  / r,
    ◇ VECT_Y = (Yx,Yy,Yz), [R]
    ◇ VECT_Z = (Zx,Zy,Zz), [R]
    ◇ SANS_NOEUD = lno, [l_noeud]
    ◇ SANS_GROUP_NO = lgrno, [l_gr_noeud]
    ◇ ITER_FROT_MAXI = / 2, [DEFAULT]
                      / i,
    ◇ COMPLIANCE = / 'NON', [DEFAULT]
                  / 'OUI',
    ◆ ASPERITE = / asperite, [R]
    ◆ E_N = / e_n, [R]
    ◇ E_V = / 0, [DEFAULT]
            / e_v, [R]

```

Titre : Opérateurs AFFE_CHAR_MECA et AFFE_CHAR_MECA_F
Auteur(s) : X. DESROCHES

Clé : U4.44.01-I

Date : 29/05/07
Page : 35/94

```

/  ◇  METHODE          =      'VERIF' ,
                                ◇  STOP_INTERP      = /  'NON' ,          [DEFAULT]
                                                /  'OUI' ,
                                ◇  TOLE_INTERP       = /  0 ,          [DEFAULT]
                                                /  e_v ,          [R]
                                ◇  DIST_MAIT          = r ,          [R]([fonction])
                                ◇  DIST_ESCL          = r ,          [R]([fonction])
                                ◇  VECT_NORM_ESCL     = (Vx,Vy,Vz) ,    [l_R]
                                ◇  VECT_ORIE_POU      = (Yx,Yy,Yz) ,    [R]
                                )

```

4.16.3 Opérandes MAILLE_MAIT ou GROUP_MA_MAIT, MAILLE_ESCL ou GROUP_MA_ESCL

L'utilisateur fournit la liste des mailles de contact potentielles de la surface 1 (MAILLE_MAIT ou GROUP_MA_MAIT) et de la surface 2 (MAILLE_ESCL ou GROUP_MA_ESCL). Ces mailles doivent être surfaciques ou linéiques en dimension 3 (QUAD9, QUAD8, QUAD4 et TRIA7, TRIA6, TRIA3 et SEG2, SEG3), linéiques et concentrés en dimension 2 (SEG2, SEG3 et POI1). Le nombre de mailles et de nœuds des deux surfaces peut être différent.

Attention :

Il est important de vérifier que la connectivité de ces mailles est telle que la normale est sortante à la structure (pour ce faire, voir MODI_MALLAGE mot clé ORIE_PEAU_2D, ORIE_PEAU_3D ou ORIE_NORM_COQU [U4.23.04]). Par ailleurs, il faut s'assurer que les structures « ne tiennent pas que par le contact » (notamment dans le cas d'un chargement en force imposée) : les mouvements de corps rigide doivent être bloqués par des conditions aux limites appropriées. Une bonne façon de le vérifier est d'effectuer un calcul avec l'opérateur STAT_NON_LINE sans prendre en compte le contact.

Dans toute la suite, on utilisera le concept maître-esclave : les nœuds de la surface esclave ne peuvent pas « pénétrer » dans les facettes (ou les nœuds) de la surface maître. Dans le cas de l'appariement de type 'MAIT_ESCL', la surface maître est celle définie par 'MAILLE_MAIT' ou 'GROUP_MA_MAIT' (pour des conseils sur le choix de la surface maître, se reporter à la note HI-75/97/034/A). Dans le cas de l'appariement de type 'NODAL', la surface maître est celle qui comporte le plus de nœuds.

Remarque :

Il est impossible de mélanger les modélisations purement bi-dimensionnelles (contraintes planes, déformations planes et axisymétriques) avec les modélisations tri-dimensionnelles. Les surfaces maître et esclave doivent être de même nature (2D/2D ou 3D/3D). Un message d'erreur vous arrêtera dans le cas contraire :

<CONTACT_84> Melange 2d et 3d dans le contact

Notons qu'une poutre, une plaque ou une coque sont de dimension 3 et qu'il est donc possible de faire du contact poutre/3D ou poutre/plaque.

4.16.4 Opérande APPARIEMENT

L'appariement peut être nodal ('NODAL') ou nœud-facette ('MAIT_ESCL'). Pour l'appariement nodal, on écrit une relation de non pénétration entre un nœud maître et un nœud esclave, alors que pour l'appariement nœud-facette, on écrit cette relation entre un nœud esclave et sa projection sur la maille maître la plus proche (voir [R5.03.50] pour les détails de la méthode d'appariement).

L'appariement nodal est déconseillé car la méthode nœud-facette est plus générale et est la seule à permettre de prendre en compte les grands glissements de façon précise.

L'appariement 'MAIT_ESCL' dispose d'une variante où l'on duplique et on échange les rôles des groupes de mailles GROUP_MA_MAIT et GROUP_MA_ESCL. Il s'agit de l'appariement 'MAIT_ESCL_SYME'. Néanmoins, son utilisation est déconseillée car elle conduit souvent à des problèmes de convergence et a tendance à sur-rigidifier le contact.

Dans les cas de contact rigide ou de relations unilatérales portant sur la température ou la pression, l'appariement peut aussi être inutile : on renseigne alors APPARIEMENT='NON'.

4.16.5 Opérande PROJECTION

Cet opérande indique le type de fonctions de forme utilisées lors de la projection d'un nœud esclave sur une maille maître. La prise en compte du caractère quadratique d'un élément permet de mieux décrire sa géométrie et d'améliorer la qualité du résultat.

- En 2D, on utilise soit des fonctions de forme linéaires soit des fonctions de forme quadratiques.
- En 3D, on utilise que des fonctions de forme linéaires. Sauf pour les éléments HEXA20 qui peuvent utiliser une version modifiée des fonctions de forme, via l'option PROJECTION='QUADRATIQUE'.

4.16.6 Opérande NB_RESOL

Nombre de résolutions simultanées faites lors du traitement du contact. Augmenter nb_resol fait gagner du temps cpu mais perdre de la place mémoire. nb_resol = 10 est un bon compromis.

4.16.7 Opérande RECHERCHE

Pour rechercher la maille maître qui sera appariée à chaque nœud esclave, on cherche d'abord le nœud maître le plus proche, soit par une boucle systématique sur tous les nœuds maîtres de la zone ('NOEUD_BOUCLE'), soit en n'examinant que les voisins de l'ancien nœud maître le plus proche ('NOEUD_VOISIN') : cette dernière approche suppose des petits glissements d'un pas de temps à l'autre (pas plus de deux mailles), mais permet de gagner du temps de calcul. Néanmoins, elle peut conduire à une non convergence, auquel cas il faut recourir à 'NOEUD_BOUCLE'.

4.16.8 Opérande REAC_GEOM

Cet opérande indique sur quelle configuration géométrique est traité le problème de contact.

- REAC_GEOM='SANS' : on travaille sur la géométrie initiale.
- REAC_GEOM='CONTROLE' : si cette option est renseignée, l'utilisateur doit en plus indiquer :
 - NB_REAC_GEOM=n : C'est le nombre de réactualisations géométriques qui seront effectuées par pas de charge. Plaçons-nous à un pas de charge donné.
 - La valeur 1 indique qu'à convergence, on réactualise la géométrie et on passe au pas de charge suivant.
 - La valeur 2 indique qu'à convergence, on ne passe pas au pas de charge suivant. On réactualise la géométrie et on réitère jusqu'à convergence.
 - La valeur n>2 indique que l'on fait n cycles réactualisation géométrique-itérations jusqu'à convergence.
- REAC_GEOM='AUTO' : on réactualise automatiquement la géométrie i.e. le nombre de cycles réactualisation géométrique-itérations jusqu'à convergence n'est pas fixé par avance mais obéit à un critère interne de convergence géométrique.

Remarque :

*Si vous avez choisi une réactualisation non-automatique et qu'Aster détecte la nécessité d'une réactualisation géométrique, il vous en avertira par une alarme :
<CFCONV> REAC_GEOM DU CONTACT SUPERIEURE A 5%
A vous de décider si cette erreur de 5% est acceptable ou non. Elle correspond grosso-modo à un déplacement relatif des deux surfaces de contact supérieur à 5% et donc un risque d'erreur d'appariement (la maille a été appariée sur une configuration qui a bougé).*

4.16.9 Opérandes **SANS_NOEUD / SANS_GROUP_NO / SANS_NOEUD_QUAD**

Ces opérandes permettent d'exclure des nœuds de la liste des nœuds esclaves, opération qui est recommandée pour des nœuds soumis à des conditions aux limites dans la direction attendue du contact (exemple encastrement).

L'opérande **SANS_NOEUD_QUAD** (qui vaut 'OUI' par défaut) permet d'autoriser ou non la liaison linéaire réalisée automatiquement par le code dans le cas de mailles quadratiques pour le contact. Cet opérande est utile dans le cas de redondance entre ces liaisons linéaires et certaines conditions limites de type Dirichlet. Quand l'opérande vaut 'NON', bien que les nœuds soient exclus du contact, les nœuds milieux des mailles quadratiques sont toujours linéairement liés aux sommets, ce qui peut provoquer ces interférences (pivots nuls).

4.16.10 Opérandes **RACCOR_LINE_QUAD / GROUP_NO_RACC / NOEUD_RACC**

Lorsqu'on a, au sein du même modèle, des maillages non compatibles avec des ordres d'interpolation EF différents (ici l'un est linéaire, l'autre est quadratique), des conditions de raccords y sont alors appliquées afin d'assurer la continuité des déplacements et d'éviter les « trous » en surface, ce qui peut engendrer des relations surabondantes (systèmes matriciels singuliers et donc pivots nuls).

On peut alors utiliser l'option **RACCOR_LINE_QUAD**='OUI' pour traiter ce conflit entre ces conditions de raccordement surfacique cinématique Linéaire/Quadratique et les conditions de contact. (cf. document [R5.03.52 § Gestion des incompatibilités dans le traitement du contact par la méthode continue]).

On renseigne alors la liste des nœuds sur lesquels portera le traitement théorique précisé dans la documentation de référence R5.03.52 par l'intermédiaire des mot-clés **GROUP_NO_QUAD** ou **NOEUD_QUAD**.

4.16.11 Opérandes **FOND_FISSURE / GROUP_NO_FOND / NOEUD_FOND / GROUP_MA_FOND / MAILLE_FOND**

Lorsqu'un modèle comporte un fond de fissure avec l'utilisation de la technique de Barsoum pour le calcul des coefficients de concentration des contraintes, on peut avoir des problèmes d'incompatibilités lors de la résolution du problème tangent numérisé du contact entre les lèvres de fissure.

En effet,

- l'interpolation polynomiale n'étant plus quadratique à cause du déplacement des nœuds milieux au quart, on doit veiller à contrôler l'influence des éléments de Barsoum sur l'intégration des termes de contact. Les schémas d'intégration de type Simpson ou Newton-Cotes avec subdivision en sous-éléments d'intégration virtuels permettent d'intégrer proprement les contributions élémentaires des éléments de contact en fond de fissure.
- les degrés de liberté de contact aux nœuds de fond de fissure n'ont aucune signification physique car appartenant à la fois aux surfaces maître et esclave.
- la transformation géométrique en certains nœuds particuliers des mailles modifiées en fond de fissure est singulière.

On peut alors utiliser l'option **FOND_FISSURE**='OUI' pour traiter ces problèmes d'incompatibilités entre les conditions de contact et la technique de Barsoum en fond de fissure. (cf. document [R5.03.52 § Gestion des incompatibilités dans le traitement du contact par la méthode continue]).

On renseigne la liste des nœuds (2D/3D) ou des mailles (3D) sur lesquels portera le traitement théorique précisé dans la documentation de référence R5.03.52 par l'intermédiaire des mot-clés **GROUP_NO_FOND**, **NOEUD_FOND**, **GROUP_MA_FOND** ou **MAILLE_FOND**.

4.16.12 Opérandes **DIST_MAIT / DIST_ESCL**

Ces opérandes permettent de prendre en compte des « trous » ou des « bosses » non maillés, ou l'épaisseur des coques (les relations de contact sont écrites entre les deux surfaces moyennes) pour les groupes de mailles 1 (**DIST_MAIT**) ou 2 (**DIST_ESCL**). On compte la distance positivement dans le sens de la normale sortante à la structure (cf. document [R5.03.50 § 3.3]). Les grandeurs renseignées sont soit des constantes, soit des fonctions des variables d'espace uniquement.

4.16.13 Opérandes DIST_POUTRE / DIST_COQUE

Semblables aux deux mots-clés précédents autour de la fibre moyenne d'une poutre ou du feuillet moyen d'une coque.

4.16.14 Opérande VECT_NORM_ESCL

Cet opérande permet le contact entre deux nœuds dans une direction $VECT_NORM_ESCL = (vx, vy, vz)$ donnée par l'utilisateur. Le contact est pris en compte entre des mailles de type POI1 et ne peut être utilisé que dans le cas d'un appariement nodal. Si ce mot-clé est absent, la direction de contact calculée lors de la procédure d'appariement est opposée à la normale maître.

4.16.15 Opérande NORMALE

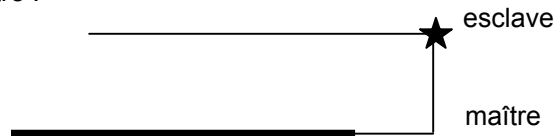
Cet opérande permet de sélectionner une méthode de calcul des normales suivant la maille maître considérée (par défaut ou explicitement par la commande : `NORMALE = 'MAIT'`) ou suivant une moyenne entre les mailles maîtres et esclaves avec la commande : `NORMALE = 'MAIT_ESCL'`.

4.16.16 Opérande STOP_SINGULIER

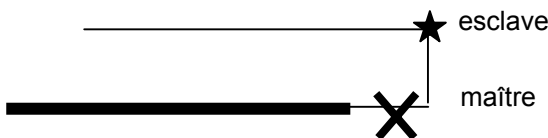
Cet opérande permet de désactiver l'erreur fatale apparaissant si la matrice de contact est singulière par `STOP_SINGULIER = 'NON'`. On conseille de n'utiliser cette opérande qu'en 3D en présence de mailles quadratiques dont les nœuds milieux sont liés aux nœuds sommets.

4.16.17 Opérande TOLE_PROJ_EXT

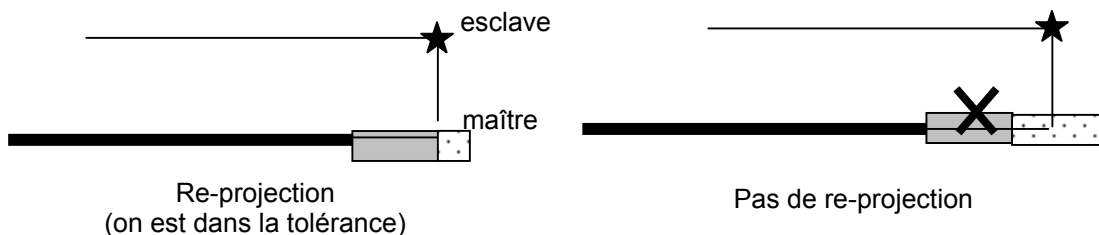
Dans certaines conditions, Aster détecte du contact entre deux surfaces alors qu'il y en a pas. Le problème vient d'abord d'une définition incorrecte et imparfaite des surfaces susceptibles d'entrer en contact. Prenons le cas du contact en 2D (les surfaces de contact sont donc des segments). Aster procède à une re-projection sur la surface maître lorsqu'un nœud esclave se projette en dehors de la surface maître :



Une solution consiste à interdire cette re-projection :



Cette solution ne tient pas compte des cas limites et peut provoquer des interpénétrations intempestives si jamais le maillage n'est pas « optimal » (ce qui est difficile à assurer dans le cadre des grandes transformations). On a donc opté pour une solution intermédiaire en limitant l'extension de la surface maître lors de la re-projection.



La valeur limite de cette re-projection est fixée par le mot-clef '`TOLE_PROJ_EXT`' qui prend pour argument la valeur (adimensionnelle) de l'extension de la maille maître dans laquelle on autorise la re-projection. Par défaut, cette valeur est fixée à 0.50. Ce qui signifie que tout nœud esclave se projetant à plus de 50% à droite ou à gauche (dans le cas d'un segment)

de la longueur de la maille maître ne sera pas reprojeté. Pour interdire complètement la re-projection, il suffit de fixer `TOLE_PROJ_EXT` à zéro. Cet opérateur est valable en 2D et en 3D (dans ce dernier cas, il s'agit de l'extension d'une maille surfacique de contact).

4.16.18 Opérande `TOLE_PROJ_INT`

Dans certains cas, la projection provoque des oscillations indésirables entre deux situations limites (mathématiquement, l'unicité de la normale n'est pas assurée). On a alors des problèmes de convergence, particulièrement lorsqu'on fait de la réactualisation géométrique automatique (`REAC_GEOM='AUTO'`). La valeur limite fixée par le mot-clef '`TOLE_PROJ_INT`' permet de régler finement la détection de la projection sur les entités géométriques internes. Ce réglage est réservé aux cas très pathologiques et exclusivement lorsqu'on utilise des surfaces de contact constituées de QUAD4.

4.16.19 Opérande `ITER_MULT_MAXI`

Cet opérande permet de fixer le nombre maximum d'itérations de contact/frottement. Le nombre d'itérations internes maximal N_{max} est fixé par la relation suivante :

$$N_{max} = ITER_MULT_MAXI \times N_{esclaves}$$

où $N_{esclaves}$ est le nombre de nœuds esclaves du couple de contact. Par défaut, `ITER_MULT_MAXI` est fixé à 4, sauf pour la méthode des contraintes actives où la valeur n'est pas modifiable et reste fixée à deux (valeur issue d'un résultat théorique de convergence).

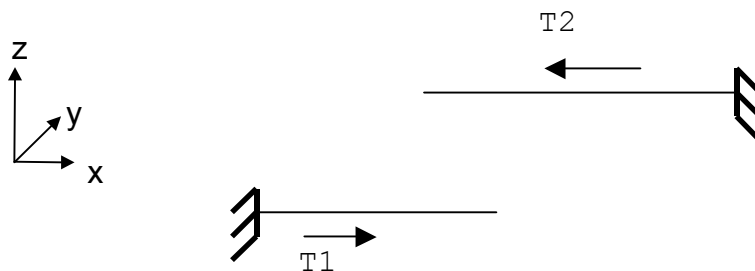
Si on dépasse le nombre maximum d'itérations de contact/frottement, on obtient le message d'erreur 'Echec dans le traitement du contact'. On peut alors tenter de raffiner le maillage, subdiviser le pas de temps, ou changer la valeur de `ITER_MULT_MAXI`.

4.16.20 Opérandes `VECT_ORIE_POU`

Cet opérande permet la modélisation du contact entre poutres **coplanaires** en 3D :

- Soient 2 poutres dans le plan xOy. `VECT_ORIE_POU` est le vecteur qui, par produit vectoriel avec le vecteur tangent à la poutre, donne la normale à utiliser. Soit :

$$VECT_ORIE_POU \wedge T = N$$



Comme ici $T1 = (1, 0, 0)$ et $T2 = (-1, 0, 0)$, avec `VECT_ORIE_POU = (0, 0, 1)`, on obtient la normale souhaitée pour chaque poutre : $N1 = (0, 1, 0)$ et $N2 = (0, -1, 0)$. Attention au fait que tout ceci est entièrement lié à l'orientation de chaque poutre.

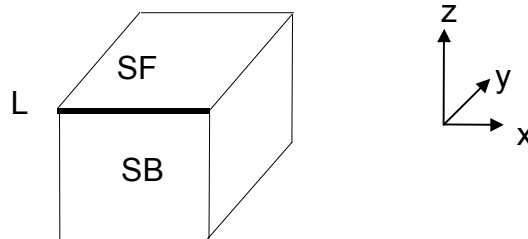
4.16.21 Opérandes `VECT_Y` et `VECT_Z`

Pour les méthodes discrètes (LAGRANGIEN et PENALISATION), l'opérande `VECT_Y` a une fonction liée au contact avec frottement en 3D.

- il permet de définir un repère local sur la surface d'un élément, repère sur lequel on décompose le déplacement des nœuds glissants. La construction de ce dièdre local est la

suivante : le premier vecteur $V1$ est obtenu par projection orthogonale de $VECT_Y$ sur la surface de l'élément considéré, le second $V2$ est obtenu par produit vectoriel de $V1$ avec le vecteur normal N ,

- soit le bloc ci-dessous dont la face SB est bloquée suivant y et la face SF est soumise à des conditions de contact-frottement. Suivant la ligne L peuvent apparaître des redondances entre blocages et conditions de frottement dans la direction y . Le problème est insoluble (pivots nuls). On peut lever ces redondances en renseignant $VECT_Y$ à $(0, 1, 0)$, direction dans laquelle les redondances apparaissent. Le problème peut alors être résolu. Ce type de difficulté n'apparaît qu'avec $METHODE = 'LAGRANGIEN'$.



Pour la méthode = 'CONTINUE', ces mots-clefs permettent à l'utilisateur d'exclure des directions de frottement qui risquent d'entrer en conflit avec d'autres conditions aux limites de Dirichlet. Ils s'utilisent donc en concordance avec `SANS_GROUP_NO` et `SANS_NOEUD`. Les directions de frottement exclues sont indiqués par $VECT_Y$ (en 2D) et $VECT_Y / VECT_Z$ (en 3D). L'exclusion de la direction de frottement permet néanmoins de garder le caractère contactant d'un nœud.

4.16.22 Opérande INTEGRATION/MODL_AXIS

L'opérande 'INTEGRATION' permet de sélectionner une méthode d'intégration numérique pour les termes de contact et de frottement. Plusieurs méthodes sont implémentées ; 'NOEUD' pour un schéma d'intégration aux nœuds, 'GAUSS' pour le schéma classique de Gauss, 'SIMPSON' pour le schéma de Simpson (intégration aux nœuds et aux milieux des éléments) et 'COTES' pour un schéma adaptatif dans le cas du contact linéaire/quadratique. L'opérande 'MODL_AXIS' permet de prendre en compte le caractère axisymétrique du problème. Ces opérandes sont utilisables uniquement avec la méthode 'CONTINUE'.

4.16.23 Opérande FORMULATION

L'opérande FORMULATION permet de choisir une formulation du problème en déplacement ou en vitesse. Ce choix concerne l'ensemble du calcul. On l'utilise en dynamique (il n'a aucun sens en statique). En dynamique, l'avantage de la formulation en vitesse est d'éliminer les oscillations numériques de la vitesse et de l'accélération au moment des impacts.

On utilise cette formulation avec un schéma d'ordre 1 en vitesse, disponible dans `DYNA_NON_LINE`, appelé `TETA_METHODE`. Ce schéma doit être choisi à la place du schéma de Newmark ou HHT. Il nécessite un paramètre `TETA` qui prend ses valeurs entre 0,5 et 1. `TETA = 1` donne le maximum d'amortissement numérique utilisable uniquement avec la méthode 'CONTINUE'.

4.16.24 Opérande ITER_CONT_MAXI/ITER_FROT_MAXI/ITER_GEOM_MAXI

Ces opérandes permettent de fixer respectivement le nombre maximal des itérations de contact, de frottement et géométriques. Rappelons que la boucle géométrique est une boucle qui sert à réactualiser l'appariement, la boucle de frottement est une boucle de point fixe sur le seuil de Coulomb alors que la boucle de contact est une boucle de type contraintes actives qui sert à déterminer les surfaces effectives de contact. Ces opérandes sont utilisables uniquement avec la méthode 'CONTINUE'.

4.16.25 Opérande DIRE_APPA

Cet opérande permet de préciser une direction de recherche pour l'appariement. La recherche des points susceptibles de rentrer en contact n'est plus fondée sur le principe de la proximité (les points les plus proches) mais selon la direction `DIRE_APPA = (vx,vy,vz)` donnée par l'utilisateur. Dans le cas des où l'appariement par direction n'est pas possible (pas de point maître dans la direction donnée), on cherche le point maître le plus proche dans une direction voisine. Cette opérande est utilisable uniquement avec la méthode 'CONTINUE'.

4.16.26 Opérande COMPLIANCE

Cet opérande permet d'activer le modèle de compliance pour la méthode 'CONTINUE'. Ce modèle prend en compte les aspect microscopiques des surfaces (aspérités) et permet une régularisation du modèle de contact de Signorini. En dynamique, l'apport de ce modèle consiste en la possibilité d'introduire une densité de percussion amortissante qui correspond à la dissipation de l'énergie du choc.

La loi de compliance introduite dans *Code_Aster* est une loi polynomiale (voir doc [R5.03.52]).

Les trois paramètres de la loi de compliance sont `ASPERITE`, `E_N` et `E_V`.

◆	ASPERITE	=	/	asperite,	[R]
◆	E_N	=	/	e_n,	[R]
◇	E_V	=	/	0.,	[DEFAULT]
			/	e_v,	[R]

4.16.27 Opérandes de résolution

Ces opérandes permettent de sélectionner une méthode de calcul suivant le type de contact (2D/3D et avec ou sans frottement) que l'on veut traiter.

◇	METHODE	=	/	'CONTRAINTE'	[DEFAULT]
			/	'LAGRANGIEN'	
			/	'PENALISATION'	
			/	'GCP'	
			/	'CONTINUE'	
			/	'VERIF'	

Cet opérande permet d'utiliser les différentes méthodes de résolution.

- | | | | |
|---|--------------|---|---|
| ◇ | CONTRAINTE | : | Par défaut on traite le problème du contact unilatéral exact sans frottement avec la méthode des contraintes actives de [R5.03.50]. |
| ◇ | LAGRANGIEN | : | La méthode lagrangienne permet de traiter de façon exacte, par multiplicateurs de Lagrange, des problèmes de contact avec ou sans frottement en 2D et 3D. |
| ◇ | PENALISATION | : | La méthode pénalisée permet de traiter soit :
- des problèmes de contact pénalisé sans frottement 2D ou 3D si on renseigne <code>E_N</code> ;
- des problèmes de contact avec frottement en 2D ou 3D avec une pénalisation sur les termes de frottement uniquement si on renseigne <code>E_T</code> et une pénalisation sur les termes de contact et de frottement si on renseigne <code>E_T</code> et <code>E_N</code> . |
| ◇ | GCP | : | C'est une méthode identique à la méthode 'CONTRAINTE' mais qui est particulièrement adaptée aux cas où le nombre de liaisons de contact est très élevé. |
| ◇ | CONTINUE | : | La méthode continue permet de traiter de façon exacte, par multiplicateurs de Lagrange augmentés, des problèmes de contact avec ou sans frottement en 2D et 3D. Les coefficients d'augmentation (ou de régularisation) sont précisés dans |

Titre : Opérateurs AFPE_CHAR_MECA et AFPE_CHAR_MECA_F
Auteur(s) : X. DESROCHES

Date : 29/05/07
Clé : U4.44.01-I Page : 43/94

COEF_REGU_CONT et COEF_REGU_FROT (ces coefficients sont de valeurs strictement positives.)

◇ VERIF

La méthode de vérification permet de contrôler si deux surfaces s'interpénètrent ou pas sans imposer les conditions de contact. C'est donc une méthode qui ne se préoccupe que de l'aspect géométrique et qui est peu coûteuse en termes de temps CPU. On peut l'utiliser par exemple pour contrôler que les deux lèvres d'une fissure ne s'interpénètrent.

Remarque :

La méthode 'CONTINUE' est une méthode moderne et prometteuse. Elle est cependant très récente dans le Code_Aster. Aussi conseille-t-on aux utilisateurs de plutôt choisir dans un premier temps une des autres méthodes et d'éventuellement tester la méthode 'CONTINUE' dans un second temps.

Pour les méthodes pénalisées et lagrangiennes, on renvoie pour plus de détails à [R5.03.51].

◇ FROTTEMENT = / 'SANS' [DEFAULT]
/ 'COULOMB'

Cet opérateur permet d'activer la prise en compte d'un frottement de Coulomb.

◇ COULOMB : valeur du coefficient de frottement pour le critère de Coulomb.

◇ E_T : coefficient de pénalisation sur le glissement pour la méthode pénalisée. Il n'est pas utilisé et n'est pas nécessaire lorsqu'une autre méthode de résolution que 'PENALISATION' est active. Une valeur de l'ordre du plus petit module d'Young des solides en contact est initialement recommandée. Un second calcul avec une valeur dix fois plus grande est vivement souhaitable pour voir la sensibilité des résultats par rapport à ce coefficient. Augmenter ensuite la valeur du coefficient jusqu'à l'obtention de résultats stables.

◇ E_N : coefficient de pénalisation sur l'interpénétration pour la méthode pénalisée. Il n'est pas utilisé et n'est pas nécessaire lorsqu'une autre méthode de résolution que 'PENALISATION' est active. Une valeur de l'ordre du plus petit module d'Young des solides en contact est initialement recommandée. En pratique le choix sur E_N est plus large que celui sur E_T et de grandes valeurs sont utilisables (10^7 ou 10^8 fois le plus petit module d'Young). On augmente la valeur du coefficient jusqu'à l'obtention de résultats stables. En outre il est possible de contrôler les distances d'interpénétration et donc d'affiner son choix de coefficient, ce qui n'est pas le cas du glissement, puisque l'on ne sait pas a priori quelles sont les zones glissantes et non glissantes alors qu'en cas de contact on peut vérifier que les distances d'interpénétration ne sont pas farfelues.

◇ COEF_MATR_FROT : coefficient, compris entre 0 et 1, de prise en compte de la partie négative de la rigidité géométrique. Plus ce coefficient est grand meilleure est la convergence lorsqu'on est proche de l'équilibre et plus la résolution est difficile loin de l'équilibre. Une valeur de 0.5 est donc initialement conseillée. Le défaut de 0 assure la convergence systématique pour un temps de calcul plus long. Ce coefficient est indispensable pour traiter des contacts surfaciques avec frottement en 3D. Il n'est pas utilisé le reste du temps.

◇ COEF_REGU_CONT : coefficient d'augmentation pour la méthode 'CONTINUE' (Lagrangien augmenté) relatif à la régularisation des lois de contact. Il peut prendre des valeurs de l'ordre de grandeur du pas de temps en dynamique (10^{-5} , 10^{-6} ...) jusqu'à beaucoup plus

Titre : **Opérateurs AFPE_CHAR_MECA et AFPE_CHAR_MECA_F**
Auteur(s) : **X. DESROCHES**

Date : **29/05/07**
Clé : **U4.44.01-I** Page : **44/94**

importante (500 par exemple). Valeur par défaut = 100.

- ◇ COEF_REGU_FROT coefficient d'augmentation pour la méthode 'CONTINUE' (Lagrangien augmenté) relatif à la régularisation des lois de frottement. Valeur par défaut = 100.
- ◇ SEUIL_INIT valeur de seuil initial de frottement pour la méthode 'CONTINUE' (Lagrangien augmenté). Elle est par défaut nulle ce qui correspond à traiter pendant la première itération de seuil le contact sans frottement.

4.16.28 Méthode VERIF

- ◇ STOP_INTERP = / 'NON' , [DEFAULT]
 / 'OUI' ,
- ◇ TOLE_INTERP = / 0. , [DEFAULT]
 / e_v , [R]

Cette méthode réalise un contrôle de l'interpénétration de deux surfaces sans imposer les conditions de contact (s'il y a interpénétration, elle restera). S'il y a interpénétration, on aura une ALARME. Le paramètre STOP_INTERP permet d'arrêter le calcul au lieu d'alarmer l'utilisateur. TOLE_INTERP règle la valeur d'interpénétration (homogène à une longueur).

4.16.29 Méthode GCP

Cette méthode permet de résoudre des problèmes de contact sans frottement. Elle résout avec une précision réglable (éventuellement très élevée) les conditions de contact à l'aide de multiplicateurs de Lagrange.

Elle est en tous points semblable à la méthode 'CONTRAINTTE' à la différence près qu'elle est de nature totalement itérative et donc très peu gourmande en mémoire. En d'autres termes, le surcoût lié à la prise en compte du contact est quasiment nul. Cette spécificité a pour effet de la rendre particulièrement adaptée aux cas impliquant des nombres importants de liaisons de contact.

- ◆ RESI_ABSO = resi , [R]
Ce mot-clé permet de régler la précision de la résolution des inéquations de contact. RESI_ABSO (qu'il faut comprendre par « résidu absolu ») représente le niveau d'interpénétration toléré pour les corps en contact. D'un point de vue pratique, il est conseillé lors de la mise au point d'une étude de partir d'une valeur assez grossière (typiquement de l'ordre de la taille des éléments au voisinage des surfaces de contact) et de la diminuer pour obtenir une solution précise.

- ◇ PRE_COND = / 'SANS' , [DEFAULT]
 / 'DIRICHLET'

Comme toute méthode de résolution itérative, la méthode GCP peut être accélérée par l'usage d'un préconditionneur. Un seul est aujourd'hui disponible et il s'agit d'un préconditionneur de Dirichlet. Son usage peut dans certains cas accélérer et diminuer sensiblement le temps de résolution. Il s'agit d'un développement en cours et son usage est déconseillé.

- ◇ COEF_RESI = / 1. [DEFAULT]
 / coef [R]

La phase de préconditionnement est réalisée par la résolution itérative d'un problème auxiliaire. Ce mot clé permet de spécifier la précision de la résolution de ce problème par : COEF_RESI*RESI_ABSO. Il s'agit d'un développement en cours et son usage est déconseillé.

- ◇ RECH_LINEAIRE = / 'ADMISSIBLE' [DEFAULT]
 / 'NON_ADMISSIBLE'

La méthode de résolution 'GCP' nécessite une phase appelée recherche linéaire. Deux variantes sont disponibles : admissible ou pas. Il s'agit d'un développement en cours et son usage est déconseillé.

```
◇ REAC_ITER      =  / 3 , [DEFAULT]
                   / reac
```

La méthode de résolution ‘GCP’ utilise des directions de recherche conjuguées les unes aux autres (dans la même logique qu’un gradient conjugué classique). Toutes les `reac` itérations, ces directions sont réinitialisées.

4.16.30 Méthode GLISSIERE

```

◇ GLISSIERE          = / 'NON', [DEFAULT]
                      / 'OUI',
◇ ALARME_JEU         / 0.0, [DEFAULT]
                      / alarm_jeu, [R]

```

Cette option est disponible uniquement pour la méthode 'CONTRAINTE' et la méthode 'CONTINUE'.

Elle permet d'activer le mode de contact bilatéral ou en glissière, dans lequel deux surfaces se trouvant en contact restent « collées » (c'est-à-dire avec un jeu nul) quelque soit l'évolution du chargement. Elle autorise de grands glissements relatifs et le mode glissière n'est pas activé avant que les surfaces soient effectivement en contact (elle ne colle pas a priori deux surfaces distantes d'un jeu non nul si le chargement ne l'implique pas).

L'opérande 'ALARME_JEU' permet d'activer une alarme dès que l'algorithme détecte que, sans la méthode glissière, il y aurait décollement des deux surfaces (un jeu virtuel supérieur à zéro). Sa valeur est réglée par défaut à 0, ce qui alarme l'utilisateur dès que les surfaces auraient dû se décoller sans l'option activée.

4.16.31 Opérande LISSAGE

Cet opérateur permet de lisser les normales aux surfaces de contact intervenant dans le calcul de la matrice de contact. On notera Q un nœud quelconque des surfaces de contact (maître ou esclave), P un nœud de la surface esclave et M le nœud maître obtenu par projection du nœud P.

◇ LISSAGE = / 'NON' [DEFAULT]

Pour un appariement de type "maître-esclave" la normale calculée est la normale entrante à la maille contenant M. Pour un appariement de type "nodal" la normale calculée est la normale sortante au nœud esclave P.

◇ LISSAGE = / 'OUI'

Le lissage se fait en deux étapes :

- la première étape du lissage consiste à effectuer une moyenne des normales aux mailles qui contiennent le nœud Q,
- la seconde étape consiste à calculer une moyenne des normales aux sommets de la maille contenant M. Cette moyenne étant pondérée par les fonctions de forme associées à M.

Remarque :

Pour un appariement de type NODAL le lissage n'apporte aucune différence.

Pour un appareillage de type `MAIT_ESCL` le lissage a un comportement qui varie en fonction du mot-clé `NORMALE`.

◇ NORMALE = / 'MAIT'

Le calcul de la matrice de contact se fait suivant la normale lissée au nœud maître.

◇ NORMALE = / 'MAIT ESCL'

Le calcul de la matrice de contact se fait suivant la moyenne des normales lissées au nœud esclave et au nœud maître.

4.16.32 Structure de données VALE_CONT

Toutes les méthodes de contact avec ou sans frottement produisent une structure de données de type VALE_CONT, qui est un `cham_no_s` avec les composantes suivantes, en chaque nœud esclave :

- CONT : indicateur de contact frottant
 - 0 : pas de contact
 - 1 : contact glissant
 - 2 : contact adhérent
- JEU : valeur du jeu
- RN : norme de la réaction normale de contact
- RNX : composante suivant DX de la réaction normale de contact
- RNY : composante suivant DY de la réaction normale de contact
- RNZ : composante suivant DZ de la réaction normale de contact
- GLIX : composante suivant t_1 du glissement tangentiel (repère local)
- GLIY : composante suivant t_2 du glissement tangentiel (repère local)
- GLI : norme du glissement tangentiel
- RTAX : composante suivant DX de la force tangentielle d'adhérence
- RTAY : composante suivant DY de la force tangentielle d'adhérence
- RTAZ : composante suivant DZ de la force tangentielle d'adhérence
- RTGX : composante suivant DX de la force tangentielle de glissement
- RTGY : composante suivant DY de la force tangentielle de glissement
- RTGZ : composante suivant DZ de la force tangentielle de glissement
- RX : composante suivant DX de la force de contact frottant (RNX+RTAX+RTGX)
- RY : composante suivant DY de la force de contact frottant (RNY+RTAY+RTGY)
- RZ : composante suivant DZ de la force de contact frottant (RNZ+RTAZ+RTGZ)
- R : norme de la force de contact frottant

Elle s'imprime comme suit sous forme de table :

```
MATABLE=POST_RELEVET(ACTION=_F( INTITULE='INFOS FROTTMNT',  
                                GROUP_NO='ESCLAVE',  
                                RESULTAT=U,  
                                INST=10.,  
                                TOUT_CMP='OUI',  
                                NOM_CHAM='VALE_CONT',  
                                OPERATION='EXTRACTION', ), ), );  
  
IMPR_TABLE(TABLE=MATABLE);
```

4.17 Mot-clé FORCE_NODALE

4.17.1 But

Mot-clé facteur utilisable pour appliquer, à des nœuds ou des groupes de nœuds, des forces nodales, définies composante par composante dans le repère GLOBAL ou dans un repère oblique défini par trois angles nautiques.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.17.2 Syntaxe

- pour AFFE_CHAR_MECA

```
FORCE_NODALE=_F ( ♦ | NOEUD      = lno , [l_noeud]
                  ♦ | GROUP_NO = lgno, [l_gr_noeud]
                  ♦ | FX=      fx , [R]
                  | FY=      fy , [R]
                  | FZ=      fz , [R]
                  | MX=      mx , [R]
                  | MY=      my , [R]
                  | MZ=      mz , [R]
                  ◇ ANGL_NAUT = (α,β,γ), [l_R]
                  ),
```

- pour AFFE_CHAR_MECA_F

```
FORCE_NODALE=_F ( ♦ | NOEUD      = lno , [l_noeud]
                  ♦ | GROUP_NO = lgno, [l_gr_noeud]
                  ♦ | FX=      fxf , [fonction]
                  | FY=      fyf , [fonction]
                  | FZ=      fzf , [fonction]
                  | MX=      mxf , [fonction]
                  | MY=      myf , [fonction]
                  | MZ=      mzf , [fonction]
                  ◇ ANGL_NAUT = (α_f,β_f,γ_f), [l_fonction]
                  ),
```

4.17.3 Opérandes

fx, fy, fz, mx, my, mz
ou fxf, fyf, fzf, mxf, myf, mzf

Valeurs des composantes des forces nodales appliquées aux nœuds spécifiés. Ces forces nodales viendront se superposer aux forces nodales issues, éventuellement, d'autres chargements. En axisymétrique, les valeurs correspondent à un secteur de 1 radian (diviser le chargement réel par 2π).

(α,β,γ)
ou (α_f,β_f,γ_f)

Liste des 3 angles, en degrés, qui définissent le repère oblique d'application des forces nodales (les derniers angles de la liste peuvent être omis s'ils sont nuls). Les angles nautiques permettent de passer du repère global de définition des coordonnées du maillage à un repère oblique quelconque (voir opérateur AFFE_CAR_ELEM [U4.42.01]). Par défaut les angles sont identiquement nuls et donc les composantes de forces sont définies dans le repère GLOBAL.

4.18 Mot-clé LIAISON_SOLIDE

4.18.1 But

Mot-clé facteur permettant de modéliser une partie indéformable d'une structure.

On impose des relations linéaires entre les degrés de liberté des nœuds de cette partie indéformable de telle sorte que les déplacements relatifs entre ces nœuds soient nuls.

Ces nœuds sont définis par les groupes de mailles, les mailles, les groupes de nœuds ou la liste de nœuds auxquels ils appartiennent.

4.18.2 Syntaxe

- pour AFPE_CHAR_MECA et AFPE_CHAR_MECA_F

```

LIAISON_SOLIDE =_F (
                                ♦ / MAILLE      = lma      ,      [l_maille]
                                /  GROUP_MA    = lgma     ,
[l_gr_maille]
                                /  NOEUD       = lno      ,      [l_noeud]
                                /  GROUP_NO    = lgn      ,      [l_gr_noeud]
                                ♦
                                NUME_LAGR = / 'NORMAL', [DEFAULT]
                                           / 'APRES' ,
                                ) ,

```

NUME_LAGR :

- Si 'NORMAL', les deux multiplicateurs de Lagrange associés à la relation seront tels que le premier sera situé avant tous les termes impliqués dans la relation et le second après, dans la matrice assemblée.
- Si 'APRES', les deux multiplicateurs de Lagrange associés à la relation seront situés après tous les termes impliqués dans la relation, dans la matrice assemblée.

Ce choix présente l'avantage d'avoir une matrice assemblée dont l'encombrement est plus faible mais a le désavantage de pouvoir faire apparaître une singularité dans la matrice.

Remarques :

De manière générale, on impose :

- en 2D ($nb_ddl * nb_noeud - 3$) relations
- en 3D ($nb_ddl * nb_noeud - 6$) relations

où

- nb_ddl est le nombre de degrés de liberté par nœud,
- nb_noeud est le nombre de nœuds de la liste donnée après LIAISON_SOLIDE

puisque un solide est déterminé par la position d'un de ses points et d'un repère en ce point.

Des relations sont écrites en prenant la formule vectorielle traduisant un mouvement de corps rigide en petites rotations :

$$\vec{u}(M) = \vec{u}(A) + \vec{\Omega}(A) \wedge \vec{AM}$$

où A est un nœud arbitraire du solide.

4.19 Mot-clé LIAISON_ELEM

4.19.1 But

En appelant "partie massive" un morceau de structure modélisé avec des éléments isoparamétriques 3D, ce mot-clé facteur permet de modéliser le raccord :

- d'une partie massive avec une partie poutre [R3.03.03] ou un élément de tuyau [R3.08.06],
- d'une partie coque avec une partie poutre [R3.06.03] ou un élément de tuyau [R3.08.06].

Le but de cette fonctionnalité n'est pas de rendre compte des échelles de longueur entre les parties à raccorder mais de permettre une simplification de la modélisation en remplaçant une partie massive par une partie poutre par exemple.

Le raccord est traité en imposant des relations linéaires entre les degrés de liberté des nœuds de la jonction des deux parties à raccorder, sans imposer de relations superflues.

4.19.2 Syntaxe (AFFE_CHAR_MECA uniquement)

```

LIAISON_ELEM =_F (  ♦ / OPTION          = / '3D_POU' ,
                                     / '3D_TUYAU' ,
                                     / 'COQ_POU' ,
                                     / 'COQ_TUYAU' ,
                                ♦ AXE_POUTRE= (x,y,z) ,    [l_R]
                                ♦ CARA_ELEM = cara ,      [cara_elem]
                                ♦ / MAILLE_1   = lma1 ,    [l_maille]
                                / GROUP_MA_1   = lgma1 ,   [l_gr_maille]
                                ♦ / NOEUD_2    = lno2 ,    [l_noeud]
                                / GROUP_NO_2   = lgno2 ,   [l_gr_noeud]
                                ◇ NUME_LAGR = / 'NORMAL' , [DEFAULT]
                                     / 'APRES' ,
                                ◇ ANGL_MAX  = / 1. ,      [DEFAULT]
                                     / angl ,    [R]
                                ) ,

```

4.19.3 Opérandes de l'option '3D_POU'

- ♦ OPTION = '3D_POU'

Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des poutres d'Euler ou de Timoshenko.

- ♦ / MAILLE_1 =
- ♦ / GROUP_MA_1 =

Ces opérandes définissent les mailles surfaciques de la partie massive modélisant la trace de la section de la poutre sur cette partie massive. Ces mailles doivent avoir été affectées par des éléments finis de faces d'éléments 3D auparavant.

- ♦ / NOEUD_2 =
- ♦ / GROUP_NO_2 =

Ces opérandes définissent le nœud de la poutre à raccorder à la partie massive. Donc si l'on utilise NOEUD_2, on ne doit donner qu'un seul nœud et si l'on utilise GROUP_NO_2, on ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

Précaution d'emploi :

La partie massive doit être maillée avec des éléments quadratiques car les coefficients des relations à imposer sont des quantités géométriques intégrées numériquement. Pour que ces intégrales soient évaluées correctement, il est nécessaire d'avoir des éléments quadratiques.

Remarque :

Un raccord entre une partie massive 3D et une partie poutre nécessite six relations linéaires.

4.19.4 Opérandes de l'option 'COQ_POU'

Cette option permet de raccorder une partie maillée en coque avec une partie poutre.

- ◆ **AXE_POUTRE =**
Permet de définir l'axe de la poutre à raccorder, dont l'extrémité est lno2 ou lgn2 (1 seul nœud).
- ◆ **CARA_ELEM = cara**
Concept créé par la commande AFPE_CARA_ELEM, contenant les caractéristiques géométriques de la coque.
- ◆ **/ MAILLE_1 =**
/ GROUP_MA_1 =
Ces opérandes définissent les mailles de bord de la partie maillée en coques (les mailles de bord sont donc des SEG2 ou SEG3 suivant la modélisation choisie). Ces mailles doivent avoir été affectées par des éléments finis de bord de coques auparavant.
- ◆ **/ NOEUD_2 =**
/ GROUP_NO_2 =
Ces opérandes définissent le nœud de la poutre à raccorder à la partie coque. Donc si l'on utilise NOEUD_2 on ne doit donner qu'un seul nœud, et si l'on utilise GROUP_NO_2, on ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

Précaution d'emploi :

La trace de la section de la poutre sur la partie coque doit correspondre exactement aux mailles de bord définies par MAILLE_1 ou GROUP_MA_1. Ceci implique l'identité des centres d'inertie, des surfaces des sections coque et poutre en vis-à-vis.

4.19.5 Opérandes de l'option '3D_TUYAU'

- ◆ **OPTION = '3D_TUYAU',**
Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des éléments TUYAU.
- ◆ **AXE_POUTRE =**
Définit l'axe du tuyau à raccorder, dont l'extrémité est un seul nœud (lno2 ou lgn2).
- ◆ **CARA_ELEM = cara**
Idem [§4.19.4].
- ◆ **/ MAILLE_1 =**
/ GROUP_MA_1 =
Ces opérandes définissent les mailles surfaciques de la partie massive modélisant la trace de la section du tuyau sur cette partie massive. Ces mailles doivent avoir été affectées par des éléments finis de faces d'éléments 3D auparavant.
- ◆ **/ NOEUD_2 =**
/ GROUP_NO_2 =
Ces opérandes définissent le nœud du tuyau à raccorder à la partie massive.

Remarque :

Un raccord entre une partie massive 3D et une partie tuyau nécessite six relations linéaires pour les degrés de liberté de poutre, plus une relation sur le mode de gonflement, plus douze relations correspondant à la transmission des modes de Fourier deux et trois d'ovalisation du tuyau.

4.19.6 Opérandes de l'option 'COQ_TUYAU'

- ◆ `OPTION = 'COQ_TUYAU'`

Cette option permet de raccorder une partie maillée en coque à une partie maillée avec des éléments tuyau.

- ◆ `AXE_POUTRE =`

Permet de définir l'axe du tuyau à raccorder, dont l'extrémité est `lno2` ou `ligno2` (un seul nœud).

- ◆ `CARA_ELEM = cara,`

Concept créé par la commande `AFPE_CARA_ELEM`, contenant les caractéristiques géométriques de la coque.

- ◆ `/ MAILLE_1 =`
`/ GROUP_MA_1 =`

Ces opérandes définissent les mailles de bord de la partie maillée en coques (les mailles de bord sont donc des `SEG2` ou `SEG3` suivant la modélisation choisie). Ces mailles doivent avoir été affectées par des éléments finis de bord de coques auparavant.

- ◆ `/ NOEUD_2 =`
`/ GROUP_NO_2 =`

Ces opérandes définissent le nœud du tuyau à raccorder à la partie coque. Donc si l'on utilise `NOEUD_2` on ne doit donner qu'un seul nœud, et si l'on utilise `GROUP_NO_2`, on ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

Précaution d'emploi :

La trace de la section du tuyau sur la partie coque doit correspondre exactement aux mailles de bord définies par `MAILLE_1` ou `GROUP_MA_1`. Ceci implique l'identité des centres d'inertie, des surfaces des sections coque et tuyau en vis-à-vis. Par conséquent des raccords de type "piquage" sont impossibles.

Remarque :

Un raccord entre une partie coque et une partie tuyau nécessite les mêmes relations linéaires que l'option '`COQ_POU`' sur les ddl de poutre de l'élément tuyau en plus des relations sur les ddl d'ovalisation, de gauchissement et de gonflement.

4.19.7 Opérande `ANGL_MAX`

- ◇ `ANGL_MAX =` `/ 1. ,` `[DEFAULT]`
 `/ angl,` `[R]`

Angle (en degré) permettant de vérifier si les mailles des listes `lma1` ou `lgma1` ont des normales faisant un angle supérieur à `angl` entre elles. Si c'est le cas, il y a émission d'un message d'alarme.

La programmation n'est faite que dans le cas 3D (donc '`3D_TUYAU`' et '`3D_POU`').

4.19.8 Opérande `NUME_LAGR`

- Si '`NORMAL`', les deux multiplicateurs de Lagrange associés à la relation seront tels que le premier sera situé avant tous les termes impliqués dans la relation et le second après, dans la matrice assemblée.
- Si '`APRES`', les deux multiplicateurs de Lagrange associés à la relation seront situés après tous les termes impliqués dans la relation, dans la matrice assemblée.

Ce choix présente l'avantage d'avoir une matrice assemblée dont l'encombrement est plus faible mais a le désavantage de pouvoir faire apparaître une singularité dans la matrice.

4.20 Mot-clé LIAISON_UNIF

4.20.1 But

Mot-clé facteur permettant d'imposer une même valeur (inconnue) à des degrés de liberté d'un ensemble de nœuds.

Ces nœuds sont définis par les groupes de mailles, les mailles, les groupes de nœuds ou la liste de nœuds auxquels ils appartiennent.

4.20.2 Syntaxe

- pour AFPE_CHAR_MECA et AFPE_CHAR_MECA_F
LIAISON_UNIF =_F (
 - ♦ / MAILLE = lma , [l_maille]
 - ♦ / GROUP_MA = lgma , [l_gr_maille]
 - ♦ / NOEUD = lno , [l_noeud]
 - ♦ / GROUP_NO = lgn , [l_gr_noeud]
 - ♦ DDL =

'DX'	,
'DY'	,
'DZ'	,
'DRX'	,
'DRY'	,
'DRZ'	

4.20.3 Opérande

- ♦ / MAILLE
- ♦ / GROUP_MA
- ♦ / NOEUD
- ♦ / GROUP_NO

Ces opérands permettent de définir une liste de n nœuds N_i dont on a éliminé les redondances, (pour MAILLE et GROUP_MA, il s'agit des connectivités des mailles).

- ♦ DDL

Cette opérande permet de définir une liste de degrés de liberté $u_i (i = 1, r)$ de r textes pris parmi : 'DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ'

Les $r \times (n - 1)$ conditions cinématiques résultantes sont :

$$u_i(N_1) = u_i(N_k)$$

$$\text{pour } k \in \{2, \dots, n\}$$

$$i \in \{1, \dots, r\}$$

4.21 Mot-clé LIAISON_CHAMNO

4.21.1 But

Mot-clé facteur utilisable pour définir une relation linéaire entre tous les ddls présents dans un concept CHAM_NO. Ce mot-clé peut également servir à imposer à la structure (ou à une partie) un travail donné, pour un chargement calculé au préalable avec un autre AFPE_CHAR_MECA et conduisant à un vecteur assemblé produit par ASSE_VECTEUR [U4.61.23].

4.21.2 Syntaxe (AFPE_CHAR_MECA seulement)

```
LIAISON_CHAMNO=_F(  ♦ CHAM_NO      =      chamno ,      [cham_no]
                    ♦ COEF_IMPO =      β      ,      [R]
                    ◇ NUME_LAGR =      /      'NORMAL' ,      [DEFAULT]
                                /      'APRES' ,
```

4.21.3 Opérandes

CHAM_NO =

Nom du cham_no qui sert à définir la relation linéaire. Les ddls reliés sont tous ceux présents dans le chamno. Les coefficients à appliquer aux ddls sont les valeurs du chamno pour ces ddls.

Exemple :

Supposons que l'on ait un chamno portant sur deux nœuds de nom N01 et N02 respectivement porteurs des ddls 'DX', 'DY' et 'DZ' pour le nœud N01 et 'DX', 'DY', 'DZ', 'DRX', 'DRY' et 'DRZ' pour le nœud N02.

Supposons aussi que le chamno ait les valeurs suivantes pour ces ddls :

```
2.      'DX'      N01
1.      'DY'      N01
3.      'DZ'      N01
1.      'DX'      N02
4.      'DY'      N02
2.      'DZ'      N02
3.      'DRX'     N02
5.      'DRY'     N02
2.      'DRZ'     N02
```

La relation linéaire que l'on va imposer est :

$$2.*DX(N01) + 1.*DY(N01) + 3.*DZ(N01) + 1.*DX(N02) + 4.*DY(N02) + 2.*DZ(N02) + 3.*DRX(N02) + 5.*DRY(N02) + 2.*DRZ(N02) = \beta$$

COEF_IMPO =

C'est la valeur du coefficient réel β au second membre de la relation linéaire.

NUME_LAGR =

- si 'NORMAL', les deux multiplicateurs de Lagrange associés à la relation seront tels que le premier sera situé avant tous les termes impliqués dans la relation et le second après, dans la matrice assemblée,
- si 'APRES', les deux multiplicateurs de Lagrange associés à la relation seront situés après tous les termes impliqués dans la relation, dans la matrice assemblée.

Ce choix présente l'avantage d'avoir une matrice assemblée dont l'encombrement est plus faible mais a le désavantage de pouvoir faire apparaître une singularité dans la matrice.

4.22 Mot-clé CHAMNO_IMPO

4.22.1 But

Il s'agit en fait d'une légère adaptation du mot clé LIAISON_CHAMNO de l'opérateur AFPE_CHAR_MECA. Celui-ci permet d'appliquer comme coefficients de relation linéaire le contenu d'un `cham_no`.

Dans le cas du mot clé CHAMNO_IMPO, on prend le contenu d'un `cham_no` comme second membre de la relation linéaire. C'est donc strictement équivalent à une procédure manuelle où on récupère les valeurs du `cham_no` à la main puis on les impose via DDL_IMPO.

4.22.2 Syntaxe (AFPE_CHAR_MECA seulement)

```
CHAMNO_IMPO  =_F(  ♦  CHAM_NO    =      chamno ,      [cham_no_sdaster]
                   ♦  COEF_MULT  =      β      ,      [R]
                   ◇  NUME_LAGR  =      /  'NORMAL' ,  [DEFAULT]
                   /  'APRES'  ,
                   )
```

4.22.3 Opérandes

CHAM_NO =

Nom du `cham_no` qui sert à définir les valeurs imposées.

COEF_MULT =

Coefficient multiplicateur β du `cham_no`.

NUME_LAGR =

- si 'NORMAL', les deux multiplicateurs de Lagrange associés à la relation seront tels que le premier sera situé avant tous les termes impliqués dans la relation et le second après, dans la matrice assemblée,
- si 'APRES', les deux multiplicateurs de Lagrange associés à la relation seront situés après tous les termes impliqués dans la relation, dans la matrice assemblée.

Ce choix présente l'avantage d'avoir une matrice assemblée dont l'encombrement est plus faible mais a le désavantage de pouvoir faire apparaître une singularité dans la matrice.

4.23 Mot-clé **LIAISON_UNILATER**

4.23.1 But

Mot clé facteur utilisable pour définir une condition unilatérale de type nodal sur un degré de liberté quelconque tel que $\Sigma(\alpha_i(t).p_i) < r(t)$ avec p_i la valeur du degré de liberté d'un nœud (déplacement, pression, température), r une valeur numérique imposée (constante ou fonction) et α_i un coefficient réel quelconque.

L'imposition d'une condition unilatérale de type contact mécanique se fait par l'opérateur «CONTACT». Par rapport à ce dernier, **LIAISON_UNILATER** est purement nodal et ne procède à aucun appariement des surfaces.

NB : Le paramètre α permet aussi, si son signe est négatif, d'inverser le sens de la relation unilatérale.

4.23.2 Syntaxe

```
LIAISON_UNILATER = _F(♦ / MAILLE      = lma,          [l_maille]
                        / GROUP_MA     = lgma,         [l_gr_maille]
                        / NOEUD        = lno,          [l_noeud]
                        / GROUP_NO     = lgno,         [l_gr_noeud]
                        ♦ NB_RESOL     = / 10,         [DEFAULT]
                        / n,           [I]
                        ♦ NOM_CMP      = lgd,          [l_TXM]
                        ♦ SANS_NOEUD   = lno,          [l_noeud]
                        ♦ SANS_GROUP_NO= lgno,         [l_gr_noeud]
                        ♦ COEF_IMPO    = r,            [R]
                        ♦ COEF_MULT    = alpha,         [l_R]
                        ♦ METHODE       = /'CONTRAINTE' [DEFAULT]
                        )
```

Pour *AFFE_CHAR_MECA_F*, on aura comme fonctions :

```
♦ COEF_IMPO = r,          [fonction]
♦ COEF_MULT = alpha,      [l_fonction]
```

4.23.3 Opérandes **MAILLE**, **GROUP_MA**, **NOEUD** et **GROUP_NO**

La condition unilatérale s'exprime sur les nœuds du maillage donnés sous les mot-clefs **NOEUD** ou **GROUP_NO**. On peut néanmoins donner les mailles portant les nœuds grâce aux mots-clefs **MAILLE** et **GROUP_MA**.

NB : contrairement au cas du contact, et du fait que les conditions soient nodales, il est inutile d'orienter les normales aux mailles de peau dans le cas **LIAISON_UNILATER**.

4.23.4 Opérande **NOM_CMP**

Liste des composantes p_i (degrés de libertés) sur lesquelles s'exerce la relation unilatérale. Ce peut être n'importe quel ddl porté par le nœud. Par exemple : PRE, PRE1, PRE2, TEMP ou encore DX, DY ou DZ.

4.23.5 Opérandes **METHODE** et **NB_RESOL**

L'opérande **METHODE** permet de sélectionner une méthode de calcul de la condition unilatérale. Pour l'instant seule la méthode des contraintes actives ('CONTRAINTE') est programmée.

NB_RESOL est le nombre de résolutions simultanées faites lors du traitement. Augmenter **nb_resol** fait gagner du temps cpu mais perdre de la place mémoire. **nb_resol = 10** est un bon compromis. Ce paramètre est réservé à un usage d'expert.

4.23.6 Opérands COEF_IMPO et COEF_MULT

COEF_IMPO est la valeur $r(t)$ imposée à droite de la relation unilatérale. COEF_MULT est la liste des coefficients multiplicateurs $\alpha_i(t)$ utilisés devant chacun des ddls. Les longueurs des listes COEF_MULT et NOM_CMP doivent bien entendu être identiques.

Ces valeurs peuvent être une fonction du temps si on utilise LIAISON_UNILATER dans AFPE_CHAR_MECA_F.

Exemple : On veut imposer la condition $1.3*PRE1 - 5.2*PRE2 < 4.0$, on aura alors :

```
NOM_CMP = ('PRE1', 'PRE2')
COEF_IMPO = 4.0
COEF_MULT = (1.3, -5.2)
```

4.24 Mot-clé VECT_ASSE

4.24.1 But

Mot-clé permettant d'affecter un second membre sous la forme d'un CHAM_NO dans les commandes STAT_NON_LINE et DYNA_NON_LINE. Ce CHAM_NO est transmis à ces commandes via le nom du chargement.

4.24.2 Syntaxe

```
VECT_ASSE = chamno [cham_no_DEPL_R]
```

4.24.3 Opérande VECT_ASSE

chamno est le nom du CHAM_NO qui va servir de second membre dans les commandes STAT_NON_LINE ou DYNA_NON_LINE.

Le mode d'utilisation peut se voir de la manière suivante :

```
char = AFPE_CHAR_MECA (
    MODELE = modele,
    VECT_ASSE = chamno,
) ;
resu = STAT_NON_LINE (
    MODELE = modele,
    EXCIT = _F (CHARGE = char ),
    ...
) ;
```


4.25 Mot-clé FORCE_FACE

4.25.1 But

Mot-clé facteur utilisable pour appliquer des **forces surfaciques** sur une **face** (d'élément volumique) définie par une ou plusieurs mailles ou des groupes de mailles de type **triangle** ou **quadrangle**.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.25.2 Syntaxe

- pour AFFE_CHAR_MECA

```
FORCE_FACE=_F (  ◆ | MAILLE   =      lma ,   [l_maille]
                  |   GROUP_MA =      lgma , [l_gr_maille]
                  ◆ | FX=          fx ,      [R]
                  |   FY=          fy ,      [R]
                  |   FZ=          fz ,      [R]
                  )
```

- pour AFFE_CHAR_MECA_F

```
FORCE_FACE=_F (  ◆ | MAILLE   =      lma ,   [l_maille]
                  |   GROUP_MA =      lgma , [l_gr_maille]
                  ◆ | FX=          fxf ,      [fonction]
                  |   FY=          fyf ,      [fonction]
                  |   FZ=          fzf ,      [fonction]
                  )
```

4.25.3 Opérandes

fx, fy, fz	valeurs des composantes dans le repère GLOBAL des forces surfaciques appliquées à la face.
fxf, fyf, fzf	

4.25.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
TRIA3, TRIA6, QUAD4, QUAD8, QUAD9, QUAD8, TRIA6	3D, 3D_SI, 3D_INCO 3D_HHMD, 3D_HMD, 3D_THHD, 3D_THHMD, 3D_THMD

4.26 Mot-clé FORCE_ARETE

4.26.1 But

Mot-clé facteur utilisable pour appliquer des forces **linéiques**, à une **arête** d'élément **volumique** ou de coque. Cette arête est définie par une ou plusieurs mailles ou des groupes de mailles de type **segment**.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFPE_CHAR_MECA) ou par l'intermédiaire d'un concept *fonction* (AFPE_CHAR_MECA_F).

4.26.2 Syntaxe

- pour AFPE_CHAR_MECA

```

FORCE_ARETE =_F ( ♦ | MAILLE      = lma ,           [l_maille]
                  | GROUP_MA   = lgma ,        [l_gr_maille]
                  ♦ | FX=       fx ,           [R]
                  | FY=       fy ,           [R]
                  | FZ=       fz ,           [R]
                  | MX=       mx ,           [R]
                  | MY=       my ,           [R]
                  | MZ=       mz ,           [R]
                  )

```
- pour AFPE_CHAR_MECA_F

```

FORCE_ARETE =_F ( ♦ | MAILLE      = lma ,           [l_maille]
                  | GROUP_MA   = lgma ,        [l_gr_maille]
                  ♦ | FX=       fxf ,          [fonction]
                  | FY=       fyf ,          [fonction]
                  | FZ=       fzf ,          [fonction]
                  | MX=       mxf ,          [fonction]
                  | MY=       myf ,          [fonction]
                  | MZ=       mzf ,          [fonction]
                  )

```

4.26.3 Opérandes

fx, fy, fz, mx, my, mz valeurs des composantes dans le repère GLOBAL
fxf, fyf, fzf, mxf, myf, mzf : des forces linéiques appliquées à l'arête.

4.26.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
SEG2 SEG2, SEG3	DKT, DST, Q4G 3D, 3D_SI, 3D_INCO COQUE_3D

4.27 Mot-clé FORCE_CONTOUR

4.27.1 But

Mot-clé facteur utilisable pour appliquer des forces **linéiques**, au **bord d'un domaine** (2D, AXIS ou AXIS_FOURIER) défini par une ou plusieurs mailles ou des groupes de mailles.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.27.2 Syntaxe

- pour AFFE_CHAR_MECA

```
FORCE_CONTOUR=_F (  ◇ | MAILLE = lma , [l_maille]
                    |   GROUP_MA= lgma , [l_gr_maille]
                    |   ◆ | FX= fx , [R]
                    |   | FY= fy , [R]
                    |   | FZ= fz , [R]
                    |   | MX= mx , [R]
                    |   | MY= my , [R]
                    |   | MZ= mz , [R]
                    )
```

- pour AFFE_CHAR_MECA_F

```
FORCE_CONTOUR=_F (  ◇ | MAILLE = lma , [l_maille]
                    |   GROUP_MA= lgma , [l_gr_maille]
                    |   ◆ | FX= fxf , [fonction]
                    |   | FY= fyf , [fonction]
                    |   | FZ= fzf , [fonction]
                    |   | MX= mxf , [fonction]
                    |   | MY= myf , [fonction]
                    |   | MZ= mzf , [fonction]
                    )
```

4.27.3 Opérandes

fx, fy, fz, mx, my, mz valeurs des composantes dans le repère GLOBAL des forces linéiques appliquées sur le contour.
 fxf, fyf, fzf, mxf, myf, mzf

4.27.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation	Composante
SEG2, SEG3	C_PLAN	Fx, Fy
	D_PLAN	Fx, Fy
	AXIS	Fx, Fy
SEG2, SEG3	AXIS_FOURIER	Fx(r), Fy(z), Fz(θ)

Remarque :

En plan, les forces sont à fournir par unité de longueur du maillage, en axisymétrique, les forces à fournir sont ramenées à un secteur de 1 radian (diviser le chargement réel par 2π).

4.28 Mot-clé FORCE_INTERNE

4.28.1 But

Mot-clé facteur utilisable pour appliquer des **forces volumiques** (2D ou 3D), à un **domaine** défini par une ou plusieurs mailles ou des groupes de mailles de type **volumique**.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFPE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFPE_CHAR_MECA_F).

4.28.2 Syntaxe

- pour AFPE_CHAR_MECA

```
FORCE_INTERNE=_F ( ♦ / TOUT      = 'OUI',
                   / | MAILLE    = lma ,      [l_maille]
                   / | GROUP_MA  = lgma,
[l_gr_maille]
                   ♦ | FX=        fx ,          [R]
                   | FY=        fy ,          [R]
                   | FZ=        fz ,          [R]
                   )
```

- pour AFPE_CHAR_MECA_F

```
FORCE_INTERNE=_F ( ♦ / TOUT      = 'OUI',
                   / | MAILLE    = lma ,      [l_maille]
                   / | GROUP_MA  = lgma,
[l_gr_maille]
                   ♦ | FX=        fxf ,        [fonction]
                   | FY=        fyf ,        [fonction]
                   | FZ=        fzf ,        [fonction]
                   )
```

4.28.3 Opérandes

fx, fy, fz, valeurs des composantes dans le repère GLOBAL des forces
fxf, fyf, fzf : volumiques appliquées sur le domaine.

4.28.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
HEXA8, HEXA20, HEXA27 PENTA6, PENTA15 TETRA4, TETRA10 PYRAM5, PYRAM13	3D, 3D_SI, 3D_INCO 3D_HHMD, 3D_HMD, 3D_THHD, 3D_THHMD, 3D_THMD, 3D_THHM, 3D_THM, 3D_HM, 3D_THH, 3D_HHM
TRIA3, TRIA6, QUAD4, QUAD8, QUAD9	C_PLAN D_PLAN AXIS AXIS_FOURIER AXIS_SI AXIS_INCO AXIS_THHM, AXIS_HM, AXIS_THH, AXIS_HHM, AXIS_THM D_PLAN_THHM, D_PLAN_HM, D_PLAN_THH, D_PLAN_HHM, D_PLAN_THM

Remarque :

En 2D (resp 3D), les forces sont à fournir par unité de surface (resp volume), en axisymétrique, les forces à fournir sont ramenées à un secteur de 1 radian (diviser le chargement réel par 2π).

4.29 Mot-clé PRES_REP

4.29.1 But

Mot-clé facteur utilisable pour appliquer une **pression** à un domaine de milieu continu **2D ou 3D** et/ou un **cisaillement** à un domaine de milieu continu **2D**.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.29.2 Syntaxe

- pour AFFE_CHAR_MECA


```

      | PRES_REP=_F ( ♦ / TOUT      = 'OUI',
                      / | MAILLE    = lma ,      [l_maille]
                      | GROUP_MA   = lgma ,      [l_gr_maille]
                      ♦ | PRES      = P ,        [R]
                      | CISA_2D    = T ,        [R]
                      )
      
```
- pour AFFE_CHAR_MECA_F


```

      | PRES_REP=_F ( ♦ / TOUT      = 'OUI',
                      / | MAILLE    = lma ,      [l_maille]
                      | GROUP_MA   = lgma ,      [l_gr_maille]
                      ♦ | PRES      = Pf ,       [fonction]
                      | CISA_2D    = Tf ,       [fonction]
                      )
      
```

4.29.3 Opérandes

| PRES = P (Pf)

Valeur de la pression imposée

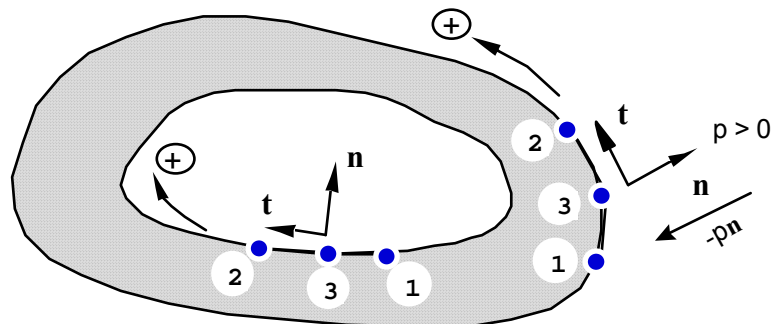
P (ou Pf) est positif suivant le sens contraire de la normale à l'élément : soit σ le tenseur des contraintes, le chargement imposé est : $\sigma_{ij} n_i n_j = -p n_i n_j$.

| CISA_2D = T (Tf)

Valeur du cisaillement imposé

T (ou Tf) est positif suivant la tangente à l'élément.

Pour la définition des normales et des tangentes, on se référera aux définitions données au [§4.1].
Exemple :



4.29.4 Modélisations et mailles

Le chargement de pression s'applique aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
SEG2 SEG3	AXIS, D_PLAN, C_PLAN, AXIS_FOURIER D_PLAN_HHM, D_PLAN_HM, D_PLAN_THHM, D_PLAN_THM
SEG3	AXIS_HHM, AXIS_HM, AXIS_THHM, AXIS_THM
TRIA6 QUAD8	3D_HHM, 3D_HM, 3D_THHM, 3D_THM
TRIA3, QUAD4 TRIA6, QUAD8, QUAD9	3D

Le chargement de cisaillement s'applique aux mailles et aux modélisations suivantes :

Type de Maille	Modélisation
SEG2 SEG3	AXIS, D_PLAN, C_PLAN, AXIS_FOURIER

4.30 Mot-clé EFFE_FOND

Mot-clé facteur utilisable pour calculer l'effet de fond sur une branche de tuyauterie (modélisation 3D exclusivement) soumise à une pression interne P.

4.30.1 Syntaxe

- pour AFFE_CHAR_MECA

```

| EFFE_FOND =_F ( ♦ | MAILLE = lma , [l_maille]
|                 | GROUP_MA= lgma, [l_gr_maille]
|                 ♦ GROUP_MA_INT = gtrou, [l_gr_maille]
|                 ♦ PRES = p, [R]
|                 )

```

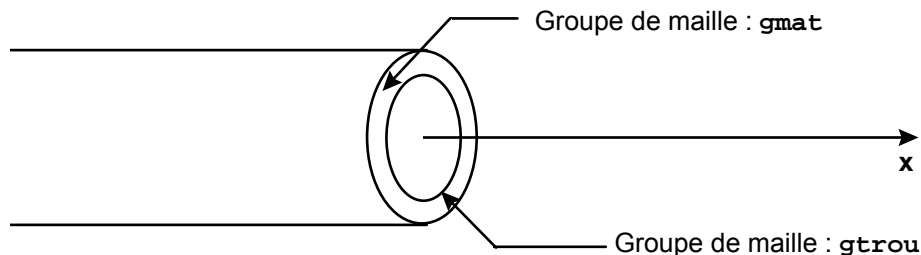
- pour AFFE_CHAR_MECA_F

```

| EFFE_FOND =_F ( ♦ | MAILLE = lma , [l_maille]
|                 | GROUP_MA= lgma, [l_gr_maille]
|                 ♦ GROUP_MA_INT = gtrou, [l_gr_maille]
|                 ♦ PRES = pf, [fonction]
|                 )

```

4.30.2 Opérandes



- ♦

GROUP_MA	=	gmat,
MAILLE	=	lma,

Ensemble des mailles surfaciques modélisant la section matérielle de tuyauterie (gmat sur la figure) où sera appliquée la pression.

- ♦ GROUP_MA_INT = gtrou,

Ensemble des mailles linéiques (SEG2 ou SEG3) modélisant le contour du trou (option sur la figure).

La connaissance de ces mailles est nécessaire car on a besoin de calculer l'aire du trou.

En effet, l'effort résultant (ou effet de fond) dû au bouchage du trou à l'extrémité vaut :

$$F_b = \pi R_i^2 P.x$$

Cet effort ou effet de fond s'applique sur la paroi du tube (gmat). L'effort réparti correspondant vaut :

$$F_p = \frac{\pi R_i^2}{\pi(R_e^2 - R_i^2)} P.x = P \frac{S_{trou}}{S_{mat}} .x$$

- ♦ PRES : p (ou pf)

Pression interne à la tuyauterie. On applique en fait F_p à gmat (avec $p > 0$ suivant le sens contraire de la normale à l'élément).

4.31 Mot-clé EPSI_INIT

4.31.1 But

Mot-clé facteur utilisable pour appliquer un chargement de déformation initiale à un élément 2D, 3D ou de structure. Cette déformation "initiale" est utilisable par exemple pour résoudre les problèmes élémentaires déterminant les correcteurs élastiques dans la cellule de base (2D, 3D), en homogénéisation périodique. Les coefficients d'élasticité homogénéisée sont obtenus en calculant par l'opérateur POST_ELEM [U4.81.22] mot-clé ENER_POT l'énergie potentielle de déformation élastique à l'équilibre à partir des correcteurs. Mais cela peut servir pour d'autres applications.

L'affectation peut se faire sur une ou plusieurs mailles, un ou plusieurs groupes de mailles ou sur tous les éléments du modèle.

4.31.2 Syntaxe

- pour AFPE_CHAR_MECA

```

EPSI_INIT =_F(  ♦ / TOUT      = 'OUI',
                  / | MAILLE    = lma ,      [l_maille]
                  / | GROUP_MA  = lgma ,     [l_gr_maille]
                  ♦ | EPXX      = epsxx ,      [R]
                  | | EPYY      = epsyy ,      [R]
                  | | EPZZ      = epszz ,      [R]
                  | | EPXY      = epsxy ,      [R]
                  | | EPXZ      = epsxz ,      [R]
                  | | EPYZ      = epsyz ,      [R]
                  | | EPX       = epsx ,       [R]
                  | | KY        = ky ,        [R]
                  | | KZ        = kz ,        [R]
                  | | EXX       = exx ,       [R]
                  | | EYY       = eyy ,       [R]
                  | | EXY       = exy ,       [R]
                  | | KXX       = kxx ,       [R]
                  | | KYY       = kyy ,       [R]
                  | | KXY       = kxy ,       [R]
                  )

```
- pour AFPE_CHAR_MECA_F

```

EPSI_INIT =_F(  ♦ / TOUT      = 'OUI',
                  / | MAILLE    = lma ,      [l_maille]
                  / | GROUP_MA  = lgma ,     [l_gr_maille]
                  ♦ | EPXX      = epsxxf ,    [fonction]
                  | | EPYY      = epsyyf ,    [fonction]
                  | | EPZZ      = epszzf ,    [fonction]
                  | | EPXY      = epsxyf ,    [fonction]
                  | | EPXZ      = epsxzf ,    [fonction]
                  | | EPYZ      = epsyzf ,    [fonction]
                  )

```

4.31.3 Opérandes

EPXX = epsxx ou epsxxf	
EPYY = epsyy ou epsyyf	
EPZZ = epszz ou epszzf	
EPXY = epsxy ou epsxyf	
EPXZ = epsxz ou epsxzf	
EPYZ = epsyz ou epsyzf	
	composantes du tenseur des déformations initiales dans le repère GLOBAL
	(en 3D seulement)

Remarques :

Le second membre élémentaire calculé sera $\int_{V_e} \wedge \varepsilon_{ini} \cdot \varepsilon(v^*) dV_e$ où \wedge désigne le tenseur

d'élasticité.

Il correspond à un chargement et ne sera pas pris en compte dans le calcul des contraintes en non linéaire. Il ne correspond donc pas à une déformation initiale en non linéaire.

Pour les éléments poutres seulement : champ de déformations généralisées constant par élément :

| EPX = epsx :
élongation selon l'axe de la poutre

| KY = ky :
courbure selon l'axe y local $\left(-\frac{d\theta_y}{dx} \right)$

| KZ = kz :
courbure selon l'axe z local $\left(\frac{d\theta_z}{dx} \right)$

Pour les poutres courbes, seul EPX est pris en compte actuellement. Emission d'un message d'erreur fatale si l'utilisateur fournit KY ou KZ.

Pour les éléments coques seulement : champ de déformations initiales constant par élément :

| EXX, EYY, EXY : déformations de membrane
| KXX, KYY, KXY : courbures

4.31.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
TRIA3, TRIA6 QUAD4, QUAD8, QUAD9	C_PLAN, AXIS, D_PLAN
HEXA8, HEXA20, HEXA27 PENTA6, PENTA15 PYRAM5, PYRAM13 TETRA4, TETRA10	3D
SEG2	POU_D_E, POU_D_T, POU_D_TG, POU_C_T
TRIA3, QUAD4	DKT, DST, Q4G
HEXA20	3D_SI
QUAD8	AXIS_SI, D_PLAN_SI

4.32 Mot-clé FORCE_POUTRE

4.32.1 But

Mot-clé facteur utilisable pour appliquer des forces **linéiques**, sur des éléments de type poutre (POU_D_T_*, POU_D_E, ...) définis sur tout le maillage ou sur une ou plusieurs mailles ou des groupes de mailles. Les forces sont définies composante par composante, soit dans le repère GLOBAL, soit dans le repère local de l'élément défini par l'opérateur AFPE_CARA_ELEM [U4.42.01].

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFPE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFPE_CHAR_MECA_F).

4.32.2 Syntaxe

- pour AFPE_CHAR_MECA

```

FORCE_POUTRE =_F (  ♦ / TOUT      = 'OUI',
                    / | MAILLE    = lma ,      [l_maille]
                    / | GROUP_MA = lgma ,
[l_gr_maille]
                    ♦ / | FX      = fx ,      [R]
                    / | FY      = fy ,      [R]
                    / | FZ      = fz ,      [R]
                    / | N       = n  ,      [R]
                    / | VY      = vy ,      [R]
                    / | VZ      = vz ,      [R]
                    ◇ TYPE_CHARGE = / 'FORCE', [DEFAULT]
                    / 'VENT' ,
                    )

```
- pour AFPE_CHAR_MECA_F

```

FORCE_POUTRE =_F (  ♦ / TOUT      = 'OUI',
                    / | MAILLE    = lma ,      [l_maille]
                    / | GROUP_MA = lgma ,
[l_gr_maille]
                    ♦ / | FX      = fxf ,      [fonction]
                    / | FY      = fyf ,      [fonction]
                    / | FZ      = fzf ,      [fonction]
                    / | N       = nf  ,      [fonction]
                    / | VY      = vyf ,      [fonction]
                    / | VZ      = vzf ,      [fonction]
                    ◇ TYPE_CHARGE = / 'FORCE', [DEFAULT]
                    / 'VENT' ,
                    )

```

4.32.3 Opérandes

♦ /	FX	: Force suivant X	[R] ou [fonction]
	FY	: Force suivant Y	[R] ou [fonction]
	FZ	: Force suivant Z	[R] ou [fonction]
/	N	: Effort de traction - compression	[R] ou [fonction]
	VY	: Effort transversal suivant Y	[R] ou [fonction]
	VZ	: Effort transversal suivant Z	[R] ou [fonction]

Notons que l'on doit rester homogène dans chaque occurrence du mot-clé facteur FORCE_POUTRE : soit toutes les composantes sont définies dans le repère GLOBAL soit toutes les composantes sont définies dans le repère de définition de la poutre.

◇ TYPE_CHARGE = 'VENT'

Si p est la pression exercée par le vent sur une surface plane normale à sa direction,

$\mathbf{v} = (v_x, v_y, v_z)$ le vecteur unitaire ayant la direction et le sens de la vitesse du vent,

Ø le diamètre du câble sur lequel s'exerce le vent,

alors :

$$FX = p \otimes v_x$$

$$FY = p \otimes v_y$$

$$FZ = p \otimes v_z$$

TYPE_CHARGE = 'FORCE'

[DEFAULT]

Cas d'une force linéique quelconque.

4.32.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
SEG2	POU_D_T, POU_C_T, POU_D_E POU_D_TGM

Ce chargement n'est pas actuellement disponible pour la modélisation POU_D_TG.

4.33 Mot-clé DDL_POUTRE

4.33.1 But

Mot-clé facteur utilisable pour bloquer des DDL dans un repère local d'une poutre.

Le repère local d'une poutre est défini :

- par l'axe X déterminé par la maille à laquelle appartient le nœud. La maille est orientée vers le nœud spécifié. Pour éviter l'indétermination, il faut que le nœud sur lequel porte la condition appartienne à un seul SEG. Dans le cas où il appartient à plusieurs mailles, l'utilisateur définit la maille donnant l'orientation locale.
- par VECT_Y : un vecteur dont la projection sur le plan orthogonal à l'axe X définit l'axe Y. L'axe Z est déterminé à l'aide de X et Y
- par ANGL_VRIL : angle de vrille, donné en degrés, permet d'orienter un repère local autour de l'axe X.

4.33.2 Syntaxe

```

• pour AFPE_CHAR_MECA
  DDL_POUTRE =_F ( ♦ | NOEUD      = lno ,          [l_noeud]
                  | GROUP_NO = lgn ,          [l_gr_noeud]
                  ♦ | DX      = ux ,          [R]
                  | DY      = uy ,          [R]
                  | DZ      = uz ,          [R]
                  | DRX     = θx ,          [R]
                  | DRY     = θy ,          [R]
                  | DRZ     = θz ,          [R]
# définition du repère local
                  ◇ | MAILLE     = lma ,          [l_maille]
                  | GROUP_MA = lgma ,          [l_gr_maille]
                  ♦ / ANGL_VRIL = G ,          [R]
                  / VECT_Y   = (V1, V2, V3) [l_R]
                  )

```

4.33.3 Opérandes

DX = ux	Valeur de la composante de déplacement en translation imposée sur les nœuds spécifiés
DY = uy	
DZ = uz	
DRX = θx	Valeur de la composante de déplacement en rotation imposée sur les nœuds spécifiés
DRY = θy	
DRZ = θz	

ANGL_VRIL = G
 angle de vrille, donné en degrés, permet d'orienter un repère local autour de l'axe X.

VECT_Y = (V1, V2, V3)
 vecteur dont la projection sur le plan orthogonal à l'axe X définit l'axe Y.
 L'axe Z est déterminé à l'aide de X et Y

4.33.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
SEG2	POU_D_T, POU_C_T, POU_D_TG, POU_D_E, POU_D_TGM

4.34 Mot-clé FORCE_TUYAU

4.34.1 But

Mot-clé facteur utilisable pour appliquer une pression sur des éléments tuyau, définis par une ou plusieurs mailles ou des groupes de mailles.

4.34.2 Syntaxe

- AFPE_CHAR_MECA :

```
|  FORCE_TUYAU=_F(  ♦  /  TOUT      =  'OUI',
                   /  |  MAILLE   =  lma ,      [l_maille]
                   |  GROUP_MA  =  lgma ,
[l_gr_maille]
                   ♦  PRES      =  p ,          [R]
                   )
```

- AFPE_CHAR_MECA_F :

```
|  FORCE_TUYAU=_F(  ♦  /  TOUT      =  'OUI',
                   /  |  MAILLE   =  lma ,      [l_maille]
                   |  GROUP_MA  =  lgma ,
[l_gr_maille]
                   ♦  PRES      =  pf ,          [fonction]
                   )
```

4.34.3 Opérande

PRES = p (pf),

Valeur de la pression imposée (réel ou fonction).

p est positif lorsque la pression est interne à la tuyauterie.

4.34.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
SEG3, SEG4	'TUYAU_3M'
SEG3	'TUYAU_6M'

4.35 Mot-clé FORCE_COQUE

4.35.1 But

Mot-clé facteur utilisable pour appliquer des efforts surfaciques, sur des éléments de type coque (DKT, DST, Q4G, ...) définis sur tout le maillage ou sur une ou plusieurs mailles ou des groupes de mailles.

Suivant le nom de l'opérateur appelé, les valeurs sont fournies directement (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.35.2 Syntaxe

- pour AFFE_CHAR_MECA


```

FORCE_COQUE =_F (
  / TOUT      = 'OUI',
  / | MAILLE   = lma ,           [l_maille]
  / | GROUP_MA = lgma ,
  [l_gr_maille]
  / | FX =      fx ,           [R]
  / | FY =      fy ,           [R]
  / | FZ =      fz ,           [R]
  / | MX =      mx ,           [R]
  / | MY =      my ,           [R]
  / | MZ =      mz ,           [R]
  / PLAN      = / 'MOY',
                  / 'INF',
                  / 'SUP',
                  / 'MAIL',           [DEFAULT]
  / PRES      = p ,           [R]
  / | F1 =      f1 ,           [R]
  / | F2 =      f2 ,           [R]
  / | F3 =      f3 ,           [R]
  / | MF1 =     mf1 ,           [R]
  / | MF2 =     mf2 ,           [R]
)
```
- pour AFFE_CHAR_MECA_F

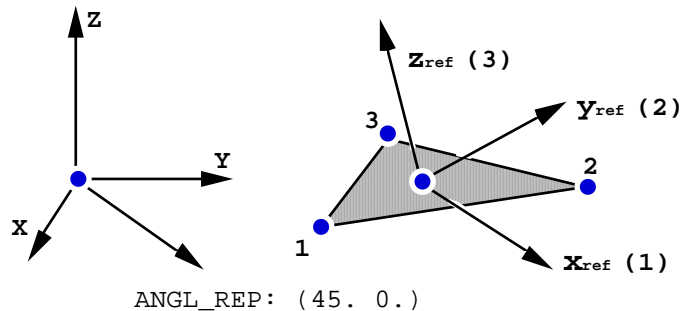

```

FORCE_COQUE =_F (
  / TOUT      = 'OUI',
  / | MAILLE   = lma ,           [l_maille]
  / | GROUP_MA = lgma ,
  [l_gr_maille]
  / | FX =      fxf ,           [fonction]
  / | FY =      fyf ,           [fonction]
  / | FZ =      fzf ,           [fonction]
  / | MX =      mxf ,           [fonction]
  / | MY =      myf ,           [fonction]
  / | MZ =      mzf ,           [fonction]
  / PLAN      = / 'MOY',
                  / 'INF',
                  / 'SUP',
                  / 'MAIL',           [DEFAULT]
  / PRES      = pf ,           [fonction]
  / | F1 =      f1f ,           [fonction]
  / | F2 =      f2f ,           [fonction]
  / | F3 =      f3f ,           [fonction]
  / | MF1 =     mf1f ,          [fonction]
  / | MF2 =     mf2f ,          [fonction]
)
```

4.35.3 Opérandes

Les opérandes de `FORCE_COQUE` peuvent être définies :

- dans le repère GLOBAL d'axes X, Y et Z ,
- dans un repère de référence défini sur chaque maille ou groupe de maille (repère défini sur la variété) ; ce repère est construit autour de la normale à l'élément de coque (z_{ref}) et d'une direction fixe (x_{ref}) (pour le groupe de maille) définie par le mot-clé `ANGL_REP` en même temps que l'épaisseur de la coque (voir mot-clé facteur `COQUE` opérateur `AFFE_CARA_ELEM` [U4.42.01]).



♦	/		FX	:	Force suivant X	[R] ou [fonction]
			FY	:	Force suivant Y	[R] ou [fonction]
			FZ	:	Force suivant Z	[R] ou [fonction]
			MX	:	Moment d'axe X	[R] ou [fonction]
			MY	:	Moment d'axe Y	[R] ou [fonction]
			MZ	:	Moment d'axe Z	[R] ou [fonction]
	/		PRES	:	Pression normale à la coque	[R] ou [fonction]
	/		F1	:	Effort de membrane suivant x_{ref}	[R] ou [fonction]
			F2	:	Effort de membrane suivant y_{ref}	[R] ou [fonction]
			F3	:	Effort normal suivant z_{ref}	[R] ou [fonction]
			MF1	:	Moment fléchissant d'axe X	[R] ou [fonction]
			MF2	:	Moment fléchissant d'axe Y	[R] ou [fonction]

Notons que l'on doit rester homogène dans chaque occurrence du mot-clé facteur `FORCE_COQUE` : soit tout en composante d'effort dans le repère GLOBAL soit tout en composante d'effort dans le repère de définition de la coque.

La pression appliquée est positive suivant le sens contraire de la normale à l'élément (définie par les 3 premiers nœuds de chaque maille (cf. [§4.25.3])).

```

◇ PLAN =      / 'MOY',
               / 'INF',
               / 'SUP',
               / 'MAIL', [DEFAULT]
  
```

Permet de définir un torseur d'efforts sur le plan moyen, inférieur, supérieur ou du maillage.

Si on note d l'excentrement et h l'épaisseur de la coque,

($F2X, F2Y, F2Z, M2X, M2Y, M2Z$) le torseur des efforts sur le plan défini par l'utilisateur (i.e. excentré)

($F1X, F1Y, F1Z, M1X, M1Y, M1Z$) le torseur des efforts dans le plan du maillage

Les formules de passage sont les suivantes :

- si le plan de calcul est le plan du maillage :

$$F2 = F1$$

$$M2 = M1$$

- si le plan de calcul est le feuillet moyen excentré :

$$F2 = F1$$

$$M2X = M1X - dx F1Y$$

$$M2Y = M1Y + dx F1X$$

- si le plan de calcul est le feuillet supérieur excentré :

$$F2 = F1$$

$$M2X = M1X - \left(d + \frac{h}{2}\right) \times F1Y$$

$$M2Y = M1Y + \left(d + \frac{h}{2}\right) \times F1X$$

- si le plan de calcul est le feuillet inférieur excentré :

$$F2 = F1$$

$$M2X = M1X - \left(d - \frac{h}{2}\right) \times F1Y$$

$$M2Y = M1Y + \left(d - \frac{h}{2}\right) \times F1X$$

- / 'MOY' on applique le torseur d'efforts sur le feuillet moyen excentré
- / 'INF' on applique le torseur d'efforts sur la peau inférieure
- / 'SUP' on applique le torseur d'efforts sur la peau supérieure
- / 'MAIL' on applique le torseur d'efforts au niveau du plan du maillage

4.35.4 Modélisations et mailles

Ce chargement s'applique aux types de mailles et aux modélisations suivantes :

Maille	Modélisation
TRIA3 QUAD4	DKT, DST
QUAD4	Q4G
TRIA7 QUAD9	COQUE_3D

Remarque :

Ce chargement n'est disponible que sur un maillage tridimensionnel (défini par COOR_3D).

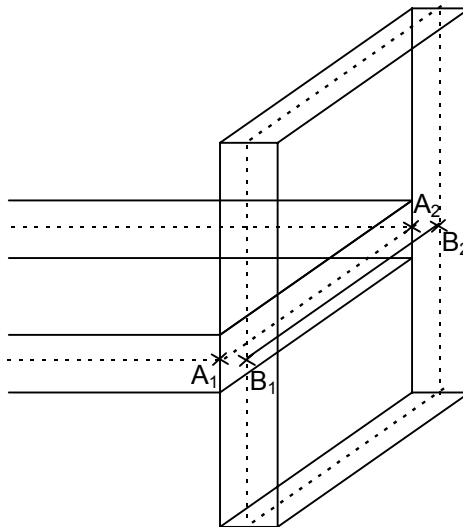
4.36 Mot-clé LIAISON_COQUE

4.36.1 But

Mot-clé facteur permettant de représenter le raccord entre des coques au moyen de relations linéaires. L'approche classique admet que deux plans maillés en coques se coupent selon une droite qui appartient au maillage de la structure.

Cela a l'inconvénient de compter deux fois le volume qui est l'intersection des deux coques.

L'idée est donc d'arrêter le maillage d'une coque perpendiculaire à une coque donnée au niveau de la peau supérieure ou inférieure de cette dernière.



On a représenté en traits pleins le volume des coques et en pointillés les plans moyens de ces coques (qui sont issus du maillage).

La coque horizontale s'arrête en $A_1 A_2$ et la projection de $A_1 A_2$ sur le plan moyen de la coque verticale est $B_1 B_2$ (que l'on a représentée en traits pleins).

La liaison entre les 2 coques se fait par des liaisons de corps solide entre les nœuds en vis-à-vis des segments $A_1 A_2$ et $B_1 B_2$.

Par exemple pour les nœuds A_1 et B_1 , on va écrire la formule (valable en petites rotations) :

$$U(B_1) = U(A_1) + \Omega(A_1) \wedge A_1 B_1$$

et l'égalité des rotations :

$$\Omega(B_1) = \Omega(A_1)$$

4.36.2 Syntaxe

- pour AFFE_CHAR_MECA et AFFE_CHAR_MECA_F


```

LIAISON_COQUE = _F
(
  ♦ | GROUP_MA_1      = l_gma1 ,      [l_gr_maille]
    | MAILLE_1        = l_ma1 ,      [l_maille]
    | GROUP_NO_1      = l_gno1 ,      [l_gr_noeud]
    | NOEUD_1         = l_no1 ,      [l_noeud]

  ♦ | GROUP_MA_2      = l_gma2 ,      [l_gr_maille]
    | MAILLE_2        = l_ma2 ,      [l_maille]
    | GROUP_NO_2      = l_gno2 ,      [l_gr_noeud]
    | NOEUD_2         = l_no2 ,      [l_noeud]

  ◇ NUME_LAGR         = / 'NORMAL' , [DEFAULT]
                      / 'APRES' ,

)
      
```

4.36.3 Opérandes

```
| GROUP_MA_1  
| MAILLE_1  
| GROUP_NO_1  
| NOEUD_1
```

A l'aide des mots-clés `GROUP_MA_1`, `MAILLE_1`, `GROUP_NO_1` et `NOEUD_1`, on constitue la première liste de nœuds (non redondante) représentant la trace de la coque perpendiculaire sur la coque courante.

Sur notre exemple, il s'agirait des nœuds du segment $B_1 B_2$ ou du segment $A_1 A_2$.

```
| GROUP_MA_2  
| MAILLE_2  
| GROUP_NO_2  
| NOEUD_2
```

A l'aide des mots-clés `GROUP_MA_2`, `MAILLE_2`, `GROUP_NO_2` et `NOEUD_2`, on constitue la seconde liste de nœuds (non redondante) appartenant à la coque perpendiculaire et en vis-à-vis des nœuds de la première liste. Le vis-à-vis est ajusté par le programme selon le critère de plus petite distance.

Sur notre exemple si la première liste est constituée des nœuds de $A_1 A_2$, la seconde liste est constituée des nœuds de $B_1 B_2$.

```
◇ NUME_LAGR          = / 'NORMAL' , [DEFAULT]  
                      / 'DEFAULT' ,
```

Voir mot-clé `LIAISON_SOLIDE` [§4.19].

Remarques importantes :

- 1) *Après les mots-clés `GROUP_MA_`, `MAILLE_`, `GROUP_NO_` et `NOEUD_`, un nœud peut apparaître plusieurs fois, c'est le programme qui se charge d'éliminer les occurrences inutiles et ainsi d'obtenir une liste non redondante de nœuds.*
- 2) *Après l'élimination des occurrences inutiles des nœuds dans les deux listes de nœuds, ces deux listes doivent être impérativement de longueur égale.*
- 3) *Les mailles données après les mots-clés `GROUP_MA_1`, `GROUP_MA_2`, `MAILLE_1` et `MAILLE_2` sont des mailles de bord de type `SEG2` ou `SEG3` des éléments de coque et pour lesquelles on n'a pas forcément affecté de modélisation mécanique.*

4.37 Mot-clé RELA_CINE_BP

4.37.1 But

Mot-clé facteur permettant la définition d'un chargement de type RELA_CINE_BP.

Ce type de chargement peut être défini pour un système mécanique comprenant une structure béton et ses câbles de précontrainte. Les profils initiaux de tension dans les câbles, ainsi que les coefficients des relations cinématiques entre les ddl des nœuds des câbles et les ddl des nœuds de la structure béton sont déterminés préalablement par l'opérateur DEFI_CABLE_BP [U4.42.04]. Les concepts `cabl_precont` produits par cet opérateur apportent toutes les informations nécessaires à la définition du chargement.

Les occurrences multiples sont autorisées pour le mot-clé facteur RELA_CINE_BP, afin de permettre dans un même appel à l'opérateur AFPE_CHAR_MECA de définir les contributions de chacun des groupes de câbles ayant fait l'objet d'appels distincts à l'opérateur DEFI_CABLE_BP [U4.42.04]. A chaque groupe de câbles considéré, défini par un concept `cabl_precont`, est associée une occurrence du mot-clé facteur RELA_CINE_BP.

Le chargement ainsi défini sert ensuite à calculer l'état d'équilibre de l'ensemble structure béton / câbles de précontrainte. Cependant, la prise en compte de ce type de chargement n'est pas effective dans tous les opérateurs de résolution. Le chargement de type RELA_CINE_BP n'est reconnu pour l'instant que par l'opérateur STAT_NON_LINE [U4.51.03], option COMP_INCR exclusivement.

4.37.2 Syntaxe (AFPE_CHAR_MECA seulement)

```

RELA_CINE_BP      =_F (
    ♦ CABLE_BP      =   cabl_pr,                [cabl_precont]
    ◇ SIGM_BPEL      =   /   'OUI',
                        /   'NON',                [DEFAULT]
    ◇ RELA_CINE      =   /   'OUI',                [DEFAULT]
                        /   'NON',
)

```

4.37.3 Opérandes

♦ CABLE_BP = cabl_pr

Concept de type `cabl_precont` produit par l'opérateur DEFI_CABLE_BP [U4.42.04]. Ce concept apporte d'une part la carte des contraintes initiales dans les éléments des câbles d'un même groupe, et d'autre part les listes des relations cinématiques entre les ddl des nœuds de ces câbles et les ddl des nœuds de la structure béton.

◇ SIGM_BPEL = / 'OUI',
/ 'NON', [DEFAULT]

Indicateur de type texte par lequel on spécifie la prise en compte des contraintes initiales dans les câbles ; la valeur par défaut est 'NON'.

Dans le cas 'NON', seul le liaisonnement cinématique est pris en compte. C'est utile si on enchaîne des STAT_NON_LINE alors qu'on a des câbles de précontrainte. Pour le premier STAT_NON_LINE il faut avoir mis 'OUI', de telle sorte que l'on met en place la tension dans les câbles. En revanche, pour les STAT_NON_LINE suivants, il ne faut considérer comme chargement que les liaisons cinématiques et donc définir le chargement avec SIGM_BPEL = 'NON', sinon la tension est comptée deux fois.

Depuis la restitution de la macro pour mettre en tension les câbles, l'utilisateur ne devrait plus avoir besoin de faire un AFPE_CHAR_MECA avec SIGM_BPEL = 'OUI', cela devrait ainsi éviter les risques d'erreur.

◇ RELA_CINE = / 'OUI', [DEFAULT]
/ 'NON',

Indicateur de type texte par lequel on spécifie la prise en compte des relations cinématiques entre les ddl des nœuds des câbles et les ddl des nœuds de la structure béton ; la valeur par défaut est 'OUI'.

4.38 Mot-clé FORCE_ELEC

4.38.1 But

Mot-clé facteur utilisable pour appliquer la force de LAPLACE agissant sur un conducteur principal, due à la présence d'un conducteur secondaire droit (ne s'appuyant pas sur une partie de maillage Aster) par rapport à ce conducteur principal.

En fait, le chargement défini par FORCE_ELEC a un module qui doit être multiplié par la fonction temporelle d'intensité spécifiée par l'opérateur DEFI_FONC_ELEC [U4.MK.10] pour représenter réellement la force de LAPLACE.

Le conducteur principal s'appuie sur tout ou partie du maillage Aster constitué d'éléments linéiques dans l'espace et défini dans cet opérateur par une ou plusieurs mailles, des groupes de mailles ou la totalité du maillage.

Remarque :

Lorsque le conducteur secondaire n'est pas rectiligne on utilisera le mot-clé INTE_ELEC [§4.40].

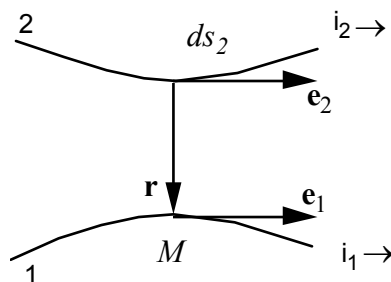
4.38.2 Syntaxe

```
FORCE_ELEC = _F
(
  ♦ / TOUT = 'OUI',
  / | MAILLE = lma, [l_maille]
  / | GROUP_MA = lgma,
  [l_gr_maille]
  ♦ / | FX = fx, [R]
  / | FY = fy, [R]
  / | FZ = fz, [R]
  / ♦ POSITION = 'PARA',
  ♦ / TRANS = (ux,uy,uz, ), [l_R]
  / DIST = d, [R]
  / POINT2 = (x2,y2,z2, ), [l_R]
  / ♦ POSITION = 'FINI',
  ♦ POINT1 = (x1,y1,z1, ), [l_R]
  ♦ POINT2 = (x2,y2,z2, ), [l_R]
  / ♦ POSITION = 'INFI',
  ♦ POINT1 = (x1,y1,z1, ), [l_R]
  ♦ POINT2 = (x2,y2,z2, ), [l_R]
)
```

4.38.3 Fonction d'espace

La fonction d'espace composant la densité linéique de force de LAPLACE exercée en un point M du conducteur 1 (conducteur principal) par les éléments du conducteur 2 (conducteur secondaire) est :

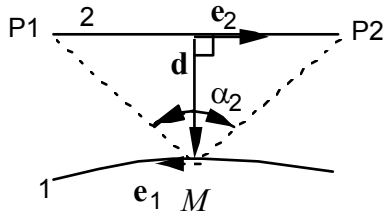
$$\mathbf{f}(M) = \frac{1}{2} \mathbf{e}_1 \wedge \int_2 \frac{\mathbf{e}_2 \wedge \mathbf{r}}{\|\mathbf{r}\|^3} \cdot ds_2$$



avec $\|\mathbf{e}_1\| = \|\mathbf{e}_2\| = 1$

Dans le cas d'un conducteur secondaire droit et fini, cette expression devient :

$$\mathbf{f}(M) = \frac{\mathbf{e}_1}{2} \wedge \frac{\mathbf{n}}{d} \cdot (\sin \alpha_1 - \sin \alpha_2)$$



avec $\mathbf{n} = \frac{\mathbf{e}_2 \wedge \mathbf{d}}{d}$, $d = \|\mathbf{d}\|$, $\|\mathbf{n}\| = 1$

Dans le cas particulier du conducteur secondaire droit infini, α_1 et α_2 tendent vers $\pi/2$, on a alors :

$$\mathbf{f}(M) = \mathbf{e}_1 \wedge \frac{\mathbf{n}}{d}$$

4.38.4 Opérandes

| FORCE_ELEC

Dans le cas où il y a plusieurs conducteurs secondaires infinis et parallèles au conducteur principal (mots-clés COUR_PRIN et COUR_SECO dans la commande DEFI_FONC_ELEC) on précise directement les composantes de la direction de la force de LAPLACE qui doivent être normées à 1.

/ | FX = fx, fx2 + fy2 + fz2 = 1.
| FY = fy, (fx, fy, fz) colinéaire à la force de LAPLACE
| FZ = fz,

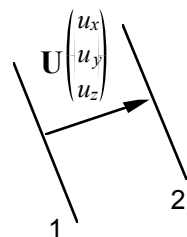
Sinon, la direction de la force de LAPLACE peut être définie par la position du conducteur unique secondaire par rapport aux éléments du conducteur principal.

/ ♦ POSITION

/ 'PARA'

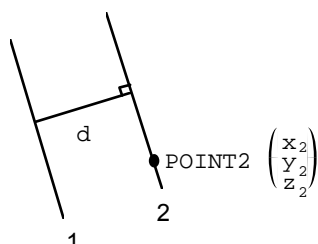
Le conducteur secondaire est considéré infini et parallèle au conducteur principal. On peut définir sa position de deux manières :

/ TRANS : (ux uy uz)



$\mathbf{U} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}$ définit la translation amenant du conducteur principal 1 au conducteur secondaire 2

/ DIST = d,
/ POINT2 = (x2, y2, z2),



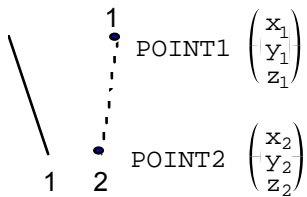
Le conducteur secondaire 2 est défini par sa distance au conducteur 1 et un deuxième point.

/ 'FINI'

Le conducteur secondaire est défini par deux points correspondant à ses extrémités

POINT1 $\begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix}$ POINT2 $\begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix}$
 et

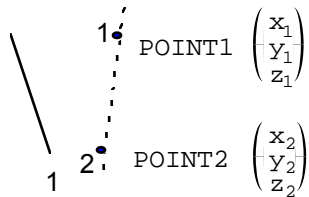
POINT1 = (x1,y1,z1),
 POINT2 = (x2,y2,z2),



/ 'INFI'

Le conducteur secondaire est défini par deux points quelconques POINT1 et POINT2.

POINT1 = (x1,y1,z1),
 POINT2 = (x2,y2,z2),



Dans les deux cas, il est préférable de choisir POINT1 et POINT2 tels que le courant circule de POINT1 à POINT2.

4.39 Mot-clé INTE_ELEC

4.39.1 But

Mot-clé facteur utilisable pour appliquer la force de LAPLACE agissant sur un conducteur principal, due à la présence d'un conducteur secondaire non nécessairement droit par rapport à ce conducteur principal.

En fait, le chargement défini par INTE_ELEC a un module qui doit être multiplié par la fonction temporelle d'intensité spécifiée par l'opérateur DEFI_FONC_ELEC [U4.MK.10] pour représenter réellement la force de LAPLACE.

Le conducteur principal s'appuie sur une partie de maillage Aster constitué d'éléments linéiques dans l'espace et défini dans cet opérateur par une ou plusieurs mailles, des groupes de mailles ou la totalité du maillage.

Le conducteur secondaire s'appuie également sur une partie de maillage Aster constitué d'éléments linéiques dans l'espace et spécifié également dans cet opérateur par une ou plusieurs mailles, des groupes de mailles, ou bien par une translation (ou une symétrie plane) par rapport au conducteur principal.

Remarque :

La différence de l'utilisation du mot-clé INTE_ELEC par rapport au mot-clé FORCE_ELEC réside dans le fait que la géométrie du conducteur secondaire peut ne pas être rectiligne et s'appuie sur une partie de maillage Aster qu'on décrit ici.

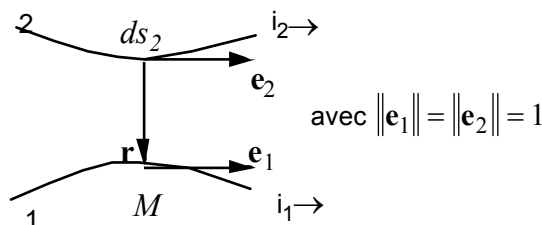
4.39.2 Syntaxe

```
INTE_ELEC =_F(  ♦ / TOUT      = 'OUI' ,
                / | MAILLE    = lma ,           [1_maille]
                / | GROUP_MA  = lgma ,
[1_gr_maille]  ♦ / | MAILLE2   = lma ,           [1_maille]
                / | GROUP_MA2 = lgma ,
[1_gr_maille]  / TRANS    = (ux,uy,uz) ,        [1_R]
                / SYME     = (x0,y0,z0,ux,uy,uz) , [1_R]
                )
```

4.39.3 Fonction d'espace

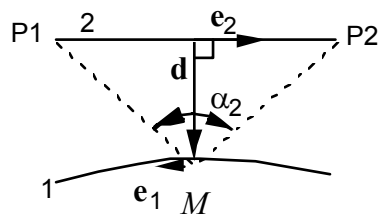
La fonction d'espace composant la densité linéique de forces de LAPLACE exercée en un point M du conducteur 1 (conducteur principal) par les éléments du conducteur 2 (conducteur secondaire) peut s'exprimer :

$$\mathbf{f}(M) = \frac{1}{2} \mathbf{e}_1 \wedge \int_2 \frac{\mathbf{e}_2 \wedge \mathbf{r}}{\|\mathbf{r}\|^3} \cdot ds_2$$



Pour chaque élément i du conducteur secondaire, on calcule sa contribution à partir de l'expression précédente et on somme :

$$\mathbf{f}(M) = \sum_i \frac{\mathbf{e}_1}{2} \wedge \frac{\mathbf{n}}{d} \cdot (\sin \alpha_1 - \sin \alpha_2)$$



$$\text{avec } \mathbf{n} = \frac{\mathbf{e}_2 \wedge \mathbf{d}}{d}, \quad d = \|\mathbf{d}\|, \quad \|\mathbf{n}\| = 1$$

4.39.4 Opérands TOUT / MAILLE / GROUP_MA / MAILLE2 / GROUP_MA2 / TRANS / SYME

TOUT, MAILLE, GROUP_MA :

Définit la géométrie du conducteur principal où le chargement est affecté.

MAILLE2, GROUP_MA2 :

Définit la géométrie du conducteur secondaire.

TRANS :

Définit une translation du conducteur principal au conducteur secondaire.

SYME :

Définit une symétrie par rapport à un plan (donné par un point $(x_0 \ y_0 \ z_0)$ et la normale $(u_x \ u_y \ u_z)$ commune au conducteur principal et au conducteur secondaire).

4.40 Mot-clé IMPE_FACE (Phénomène 'ACOUSTIQUE')

4.40.1 But

Le mot-clé facteur IMPE_FACE permet d'appliquer une impédance acoustique, à une face définie par une ou plusieurs mailles ou groupes de mailles de type triangle ou quadrangle.
Les valeurs sont directement données si l'opérateur appelé est AFPE_CHAR_MECA ; si c'est AFPE_CHAR_MECA_F, elles proviennent d'un concept de type fonction.

4.40.2 Syntaxe

- pour AFPE_CHAR_MECA

```
IMPE_FACE =_F ( ♦ | MAILLE = lma , [l_maille]
                | GROUP_MA= lgma , [l_gr_maille]
                ♦ IMPE = Q , [R]
                )
```

- pour AFPE_CHAR_MECA_F

```
IMPE_FACE =_F ( ♦ | MAILLE = lma , [l_maille]
                | GROUP_MA= lgma , [l_gr_maille]
                ♦ IMPE = Qf , [fonction]
                )
```

4.40.3 Opérande IMPE_FACE

IMPE_FACE = Q (Qf)

Impédance acoustique appliquée à la face.

4.40.4 Modélisations et mailles

Le chargement s'applique aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
TRIA3, TRIA6 QUAD4, QUAD8, QUAD9	3D_FLUIDE

4.41 Mot-clé *VITE_FACE* (Phénomène 'ACOUSTIQUE')

4.41.1 But

Le mot-clé facteur *VITE_FACE* permet d'appliquer des vitesses normales, à une face définie par une ou plusieurs mailles ou groupes de mailles de type triangle ou quadrangle.

Les valeurs sont directement données si l'opérateur appelé est *AFFE_CHAR_MECA*, si c'est *AFFE_CHAR_MECA_F*, elles proviennent d'un concept de type fonction.

4.41.2 Syntaxe

- pour *AFFE_CHAR_MECA*

```
VITE_FACE =_F (  ♦ | MAILLE = lma , [l_maille]
                  | GROUP_MA= lgma , [l_gr_maille]
                  ♦ VNOR = V , [R]
                  )
```

- pour *AFFE_CHAR_MECA_F*

```
VITE_FACE =_F (  ♦ | MAILLE = lma , [l_maille]
                  | GROUP_MA= lgma , [l_gr_maille]
                  ♦ VNOR = Vf , [fonction]
                  )
```

4.41.3 Opérande *VNOR*

$VNOR = V (Vf)$

Vitesse normale appliquée à la face.

4.41.4 Modélisations et mailles

Le chargement s'applique aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
TRIA3 , TRIA6 QUAD4 , QUAD8 , QUAD9	3D_FLUIDE

4.42 Mot-clé ONDE_PLANE

4.42.1 But

Mot-clé facteur utilisable pour imposer un chargement sismique par onde plane, correspondant aux chargements classiquement rencontrés lors des calculs d'interaction sol-structure par les équations intégrales (voir [R4.05.01]).

4.42.2 Syntaxe (AFFE_CHAR_MECA_F seulement)

```
ONDE_PLANE =_F (
  ♦ TYPE_ONDE      = ty,           [txm]
  ♦ DIRECTION      = (kx,ky,kz),   [l_R]
  ♦ DIST_ORIG      = H,           [R]
  ♦ FONC_SIGNAL    = f,           [fonction]
)
```

4.42.3 Opérandes

- ♦ TYPE_ONDE = ty,
 Type de l'onde : 'P' onde de compression
 'SV' ondes de cisaillement
 'SH' ondes de cisaillement
- ♦ DIRECTION = (kx,ky,kz),
 Direction de l'onde.
- ♦ DIST_ORIG = H,
 Distance du front d'onde principal à l'origine à l'instant initial.
- ♦ FONC_SIGNAL = f,
 Dérivée du profil de l'onde : $f(x)$ pour $x \in]\infty, +\infty[$.

En harmonique, une onde plane élastique est caractérisée par sa direction, sa pulsation et son type (onde P pour les ondes de compression, ondes SV ou SH pour les ondes de cisaillement). En transitoire, la donnée de la pulsation, correspondant à une onde stationnaire en temps, doit être remplacée par la donnée d'un profil de déplacement dont on va prendre en compte la propagation au cours du temps dans la direction de l'onde.

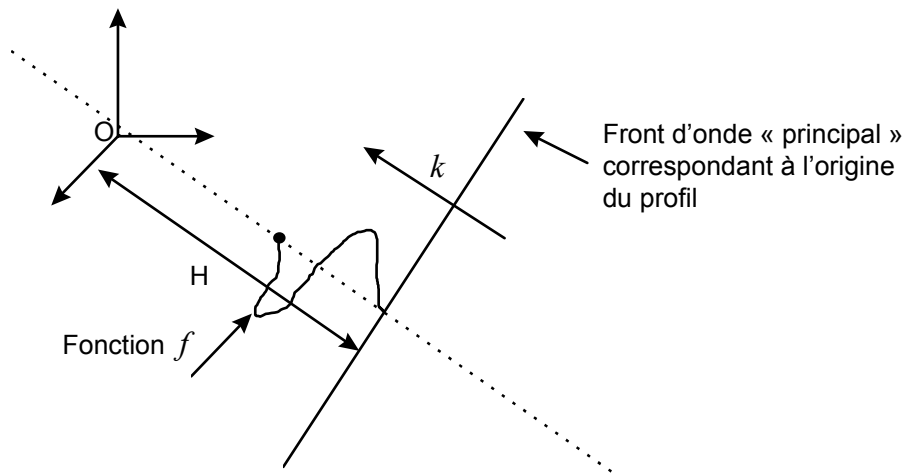
Plus précisément, on caractérise :

- une onde P par la fonction $u(x,t) = f(k.x - C_p t)k$
- une onde S par la fonction $u(x,t) = f(k.x - C_s t) \wedge k$

Avec :

- k , vecteur unitaire de direction
- f représente alors le profil de l'onde donné selon la direction k .

Attention : c'est la dérivée f' que l'utilisateur donne dans FONC_SIGNAL.



H_0 est la distance du front d'onde principal à l'origine O, portée par le vecteur directeur de l'onde à l'instant initial du calcul, H la distance du front d'onde principal à l'origine O, à un instant quelconque.

4.42.4 Modélisations et mailles

Type de Maille	Modélisation
MECA_FACE_*	3D_ABSO
MEPLSE2, MEPLSE3	2D_ABSO

4.43 Mot-clé ONDE_FLUI (Phénomène 'ACOUSTIQUE')

4.43.1 But

Le mot-clé facteur ONDE_FLUI permet d'appliquer une amplitude de pression d'onde incidente sinusoïdale arrivant normalement à une face définie par une ou plusieurs mailles ou groupes de mailles.

4.43.2 Syntaxe

- pour AFPE_CHAR_MECA

```
ONDE_FLUI =_F (  ♦ | MAILLE = lma , [l_maille]
                  | GROUP_MA= lgma , [l_gr_maille]
                  ♦ PRES = P , [R]
                  )
```

- pour AFPE_CHAR_MECA_F

Non développé.

4.43.3 Opérande PRES

PRES = P ,

Amplitude de pression d'onde incidente sinusoïdale arrivant normalement à la face.

4.43.4 Modélisations et mailles

Le chargement s'applique aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
TRIA3, TRIA6 QUAD4, QUAD8, QUAD9	3D_FLUIDE
SEG2, SEG3	2D_FLUIDE, AXIS_FLUIDE

4.44 Mot-clé FLUX_THM_REP

4.44.1 But

Mot-clé facteur utilisable pour appliquer à un domaine de milieu continu 2D ou 3D défini par des mailles ou groupes de mailles un flux de chaleur et/ou un apport de masse fluide (flux hydraulique).

4.44.2 Syntaxe

- pour AFPE_CHAR_MECA

```

FLUX_THM_REP =_F (  ♦ / TOUT      = 'OUI' ,
                    / | MAILLE     = lma ,      [l_maille]
                    / | GROUP_MA   = lgma ,
[l_gr_maille]
                    ♦ | FLUN        = φT ,      [R]
                    | FLUN_HYDR1   = φe ,      [R]
                    | FLUN_HYDR2   = φv ,      [R]
                    )

```
- pour AFPE_CHAR_MECA_F

```

FLUX_THM_REP =_F (  ♦ / TOUT      = 'OUI' ,
                    / | MAILLE     = lma ,      [l_maille]
                    / | GROUP_MA   = lgma ,
[l_gr_maille]
                    ♦ | FLUN        = φTf ,     [fonction]
                    | FLUN_HYDR1   = φef ,     [fonction]
                    | FLUN_HYDR2   = φvf ,     [fonction]
                    )

```

4.44.3 Opérandes

| FLUN = φT ,

Valeur du flux de chaleur

$$\phi_T = \lambda_T \frac{\partial T}{\partial n} + h_m^e \phi^e + h_m^v \phi^v + h_m^a \phi^a$$

avec : h_m^l : enthalpie massique du liquide

h_m^v : enthalpie massique de la vapeur

h_m^a : enthalpie massique de l'air

ϕ^e et ϕ^v sont les flux hydrauliques définis ci-dessous

| FLUN_HYDR1 = φe ,

Valeur du flux hydraulique associé au constituant eau

| FLUN_HYDR2 = φv ,

Valeur du flux hydraulique associé au constituant air

$$\phi^e = \rho_e (\nabla P_e - \rho_e \mathbf{g}) \cdot \mathbf{n}$$

$$\phi^v = \rho_v (\nabla P_v - \rho_g \mathbf{g}) \cdot \mathbf{n}$$

avec : ρ_e : masse volumique du liquide

ρ_v : masse volumique de la vapeur

P_e : pression du liquide (PRE1)

P_g : pression de la vapeur (PRE2)

4.44.4 Modélisations et mailles

Les flux normaux s'appliquent aux types de mailles et aux modélisations suivantes :

Type de Maille	Modélisation
SEG2	D_PLAN_YYYY
SEG3	AXIS_YYYY, D_PLAN_YYYY
FACE8	3D_YYYY

avec YYYY = THM ou THH ou THHM ou HM ou HHM.

4.45 Mot-clé ARLEQUIN

4.45.1 But

Mot-clé facteur définissant les paramètres de la méthode Arlequin. Cette méthode consiste à relier des modèles par le volume. A priori, n'importe quelle combinaison de maillages, interpolations et cinématiques est envisageable. Elle permet ainsi d'enrichir localement un modèle, créer la jonction entre deux modèles, substituer localement un modèle par un autre. Cette jonction est assurée par pondération du travail des forces élastiques sur espace médiateur : la trace d'un des deux modèles sur la zone de collage.

4.45.2 Restrictions d'usage

Seulement deux modèles peuvent être superposés au même endroit. Seuls les modèles volumiques (3D), surfaciques (2D) et de coques (2D et 3D) sont autorisés.

4.45.3 Syntaxe

- pour AFPE_CHAR_MECA seulement

```
ARLEQUIN = _F( ♦ GROUP_MA_1 = gma1 , [gr_maille]
                ♦ GROUP_MA_2 = gma2 , [gr_maille]
                ◇ GROUP_MA_COLL = gma , [gr_maille]
                ◇ COLLAGE = / 'GROSSIER' , [DEFAULT]
                        / 'FIN' ,
                        / 'GROUP_MA_1' ,
                        / 'GROUP_MA_2' ,
                ♦ / POIDS_1 = p1 , [R]
                  / POIDS_2 = p2 , [R]
                  / POIDS_GROSSIER = pf , [R]
                  / POIDS_FIN = pg , [R]
                ◇ CARA_ELEM = cara , [cara_elem]
                )
```

4.45.4 Opérandes

- ♦ GROUP_MA_1 = gma1
- ♦ GROUP_MA_2 = gma2

Noms des groupe de mailles définissant les modèles se chevauchant. La position relative des frontières, des nœuds et des mailles appartenant à ces deux modèles est a priori indépendante. Aucun nœud ni aucune maille ne doit être partagé par les deux modèles. GROUP_MA_1 et GROUP_MA_2 jouent le même rôle, sans distinction.

◇ GROUP_MA_COLL = gma

Groupe de mailles définissant la zone où les deux modèles sont reliés. Cette zone doit normalement correspondre à des mailles de GROUP_MA_1 ou de GROUP_MA_2 appartenant à la zone de recouvrement des deux domaines. Il est toutefois possible qu'elle déborde de cette zone de superposition. Si le mot clé n'est pas renseigné, la zone exacte de superposition est retenue dans l'algorithme.

◇ COLLAGE =

Choix du modèle utilisé pour définir les multiplicateurs de Lagrange de collage (GROUP_MA_1 ou GROUP_MA_2 restreint à la zone de collage). Une comparaison du volume moyen des mailles des deux modèles permet aussi l'emploi de FIN ou GROSSIER.

◆ / POIDS_1 = ρ_1 ,
/ POIDS_2 = ρ_2 ,
/ POIDS_FIN = ρ_f ,
/ POIDS_GROSSIER = ρ_g ,

Réel compris strictement entre 0. et 1. Il permet de définir la valeur des fonctions de pondération dans la zone de superposition. POIDS_i correspond au poids du modèle i.

La donnée de ρ_1 ou ρ_2 permet de décrire le couple (α_1, α_2) tel que :

$(\alpha_1, \alpha_2) = (\rho_1, 1 - \rho_1)$ ou $(1 - \rho_2, \rho_2)$

◇ CARA_ELEM =

Champ de caractéristiques élémentaires issu de AFPE_CARA_ELEM (épaisseurs pour les coques, sections pour les poutres). A préciser obligatoirement quand un des deux modèles est composé de coques.

4.45.5 Exemples et conseils d'utilisation

Opération de jonction de modèles :

```
ARLEQUIN = _F(
    GROUP_MA_1 = gma1,
    GROUP_MA_2 = gma2,
    [CARA_ELEM = cara,]
    COLLAGE    = 'GROSSIER',
    POIDS_FIN   = 0.99,
)
```

Opération de substitution (introduction de défauts ...) :

```
ARLEQUIN = _F(
    GROUP_MA_1 = gma1,      # modèle sans défaut
    GROUP_MA_2 = gma2,      # modèle avec défaut
    GROUP_MA_COLL = gmac,   # couronne encerclant le défaut,
                                # suffisamment loin du défaut pour que les 2
                                # modèles soient compatibles mécaniquement

    COLLAGE    = 'GROUP_MA_2',
    POIDS_2     = 0.9999,
)
```


4.46 Mot-clé GRAPPE_FLUIDE

4.46.1 But

Mot-clé facteur permettant l'entrée des données de calcul des forces fluides lors des études de chute de grappe.

Ce type de chargement est spécifique à ce genre d'étude. Il est associé à un maillage prédéfini. Les données sont entrées par l'intermédiaire d'un fichier « include » propre au type d'assemblage dont on étudie la chute de grappe. Des exemples de fichiers « include » et de maillage sont disponibles avec les cas tests associés à ce type de chargement. Ils recensent, sous forme de variables pythons, les données entrées sous les différents mots clefs simples du mot clef facteur GRAPPE_FLUIDE.

Si l'on souhaite modifier une, ou plusieurs, valeurs, il est possible de faire à nouveau appel à GRAPPE_FLUIDE. Selon la règle de surcharge, la dernière valeur entrée est alors celle employée pour le calcul.

Attention :

Même si, pour des raisons purement informatiques, les mots clefs simples apparaissent comme facultatifs, il est nécessaire que toutes les données soient entrées, soit directement par l'utilisateur, soit, ainsi qu'il l'est préconisé, par l'intermédiaire du fichier include. Un test est effectué qui permet de vérifier si les données géométriques de la grappe sont cohérentes. Ainsi, on teste si la masse totale de la grappe et les densités volumiques affectées à ses composantes (crayon, tige, araignée) correspondent. L'erreur relative acceptée est de 10^{-3} , sinon un message d'alarme est émis.

4.46.2 Syntaxe

Applicable à AFPE_CHAR_MECA seulement :

```
GRAPPE_FLUIDE = _F(
```

définition du groupe de mailles modélisant le tube, l'araignée et le crayon :

```
    ◇ GROUP_MA          = 'magrap',                                [gr_maille]
```

définition du noeud supérieur du tube :

```
    ◇ / GROUP_NO_ORIG    = 'grnori',                                [gr_noeud]
      / NOEUD_ORIG       = 'nonori',                                [noeud]
```

définition du noeud inférieur du crayon :

```
    ◇ / GROUP_NO_EXTR    = 'grnoex',                                [gr_noeud]
      / NOEUD_EXTR       = 'nonoex',                                [noeud]
```

profondeur d'enfoncement initial du crayon dans le coeur :

```
    ◇ Z0                  = z0,                                     [R]
```

définition des données hydrauliques :

```
    ◇ CARA_HYDR = ( 'Q', 'ROC', 'ROD', 'ROP', 'ROM', 'ROML', 'ROG',
                    'NUC', 'NUM', 'NUML', 'NUG', 'P2', 'P3', 'P4', 'CGG', 'G' ),
    ◇ VALE_HYDR = ( q, roc, rod, rop, rom, roml, rog,
                    nuc, num, numl, nug, p2, p3, p4, cgg, g ),    [R]
```

définition des données géométriques de grappe :

```
    ◇ CARA_GRAPPE=( 'M', 'DTIGE', 'DTMOY', 'ROTIGE', 'LTIGE', 'LLT', 'LCT',
                    'VARAI', 'RORAI', 'DCRAY', 'ROCRAY', 'LCRAY', 'LCHUT',
                    'CFM', 'CFCI', 'CFGG', 'HRUGC', 'HRUGTC', 'NCA' ),
    ◇ VALE_GRAPPE=( m, dtige, dtmoy, rotige, ltige, llt, lct,
                    varai, rorai, dcray, rocray, lcray, lchut,
                    cfm, cfc, cgg, hrugg, hrugtc, nca ),    [R]
```

définition des données géométriques du mécanisme de commande :

```
    ◇ CARA_COMMANDE = ( 'LI', 'LML', 'LG', 'LIG', 'DIML', 'DEML', 'DCSP',
                        'DG', 'HRUGML', 'HRUGCSP', 'HRUGG' ),
```

Titre : **Opérateurs AFPE_CHAR_MECA et AFPE_CHAR_MECA_F**Date : **29/05/07**Auteur(s) : **X. DESROCHES**Clé : **U4.44.01-I**Page : **90/94**

```

    ◇ VALE_COMMANDE = ( li, lml, lg, lig, diml, deml, dcsp,
                        dg, hrugml, hruggcsp, hrugg), [R]

```

définition des données géométriques de la manchette et son adaptateur:

```

    ◇ CARA_MANCHETTE = ( 'LM', 'LA', 'LIM', 'DINT', 'DEML', 'DCMT', 'VMT',
                        'ROMT', 'DA', 'HRUGM', 'HRUGA'), [R]
    ◇ VALE_MANCHETTE = ( lm, la, lim, dint, deml, dcmt, vmt, romt,
                        da, hrugmg, hruga), [R]

```

définition des données géométriques des tubes guides :

```

    ◇ CARA_GUIDE = ( 'NRET', 'L0', 'L1', 'L2', 'L3', 'L4',
                    'DTG', 'DR', 'DOR', 'D0', 'D00', 'HRUGTG'),
    ◇ VALE_GUIDE = ( nret, l0, l1, l2, l3, l4,
                    (dtg, dr, dor, d0, D00, hrugtg), [R]

```

définition des données géométriques des assemblages :

```

    ◇ CARA_ASSEMBLAGE = ( 'SASS', 'DCC', 'DTI', 'NGM', 'NGMDP',
                        'KM', 'KS', 'KI', 'KES', 'KEI', 'KF'),
    ◇ VALE_ASSEMBLAGE = ( sass, dcc, dti, ngm, ngmp,
                        km, ks, ki, kes, kei, kf), [R]

```

définition des coefficients de perte de charge singulière :

```

    ◇ CARA_PDC = ( 'CD0', 'CD1', 'CD2', 'CDELARG', 'CDRET',
                    'CDM', 'CDA', 'CDML', 'CDI', 'CDG'),
    ◇ VALE_PDC = ( cd0, cd1, cd2, cdelarg, cdret,
                    cdm, cda, cdml, cdi, cdret), [R]

```

définition du point d'application des diverses forces fluides :

```

    ◇ APPL_FORC_ARCHI = / 'REPARTIE', [DEFAULT]
                        / 'CDG',

    ◇ APPL_FORC_FPLAQ = / 'REPARTIE', [DEFAULT]
                        / 'CDG',
                        / 'ZONE',
                        / 'MILIEU',
                        / 'DISTR',

    ◇ APPL_FORC_FMEC = / 'REPARTIE', [DEFAULT]
                       / 'CDG',
                       / 'ZONE',
                       / 'PTREP',

    ◇ APPL_FORC_FTG = / 'REPARTIE', [DEFAULT]
                      / 'CDG',
                      / 'ZONE',
                      / 'PTREP',

```

si APPL_FORC_ARCHI = 'CDG' ou

si APPL_FORC_FPLAQ = 'CDG' ou

si APPL_FORC_FMEC = 'CDG' ou

si APPL_FORC_FTG = 'CDG'

```

    ◆ MASS_INER = mass_iner, [table]

```

définition de la direction de la force de plaquage :

```

    ◇ DIRE_FORC_FPLAQ = (n1,n2,n3), [1_R]

```

définition du numéro d'unité d'impression des forces :

```

    ◇ UNITE_IMPR_FORCE = i1, [I]

```

définition du numéro d'unité d'impression des nœuds par zone :

```

    ◇ UNITE_IMPR_NOEUD = i2, [I]
    ),

```

4.46.3 Opérandes

La signification des données géométriques et hydrauliques est expliquée dans le document [R4.07.06], Chargements fluides sur une grappe de commande en cours de chute. Pour une description détaillée du rôle de chacune de ces variables, on se référera donc à ce document.

4.46.3.1 Mot clé APPL_FORC_ARCHI

```
◇ APPL_FORC_ARCHI = / 'REPARTIE', [DEFAULT]
                    / 'CDG',
```

Ce mot clé permet de définir le point d'application de la force d'Archimède :

- soit répartie sur toute la grappe ;
- soit concentrée au centre de gravité.

Dans le second cas il faut renseigner le mot clé MASS_INER.

4.46.3.2 Mot clé APPL_FORC_FPLAQ

```
◇ APPL_FORC_FPLAQ = / 'REPARTIE', [DEFAULT]
                    / 'CDG',
                    / 'ZONE',
                    / 'MILIEU',
                    / 'DISTRIB',
```

Ce mot clé permet de définir le point d'application de la force de plaquage :

- soit répartie uniformément sur toute la grappe ('REPARTIE') ;
- soit concentrée au centre de gravité ('CDG') ;
- soit répartie uniformément dans la partie de la grappe qui se situe dans le guidage continu ('ZONE') ;
- soit au milieu du guidage continu ('MILIEU') ;
- soit répartie selon une distribution particulière ('DISTRIB').

Comme pour le paragraphe précédent, si le mot clé prend la valeur 'CDG', il faut renseigner obligatoirement le mot clé MASS_INER.

4.46.3.3 Mot clé APPL_FORC_FMEC

```
◇ APPL_FORC_FMEC = / 'REPARTIE', [DEFAULT]
                   / 'CDG',
                   / 'ZONE',
                   / 'PTREP',
```

Ce mot clé permet de définir le point d'application de la force dans le mécanisme de levée :

- soit répartie uniformément sur toute la grappe ('REPARTIE') ;
- soit concentrée au centre de gravité ('CDG') ;
- soit répartie uniformément dans la partie de la grappe qui se situe dans le mécanisme de levée ;
- soit appliquée à l'extrémité de la grappe ('PTREP').

Si le mot clé prend la valeur 'CDG', il faut renseigner obligatoirement le mot clé MASS_INER.

4.46.3.4 Mot clé APPL_FORC_FTG

```
◇ APPL_FORC_FTG = / 'REPARTIE', [DEFAULT]
                  / 'CDG',
                  / 'ZONE',
                  / 'PTREP',
```

Ce mot clé permet de définir le point d'application de la force avant et après le retreint :

- soit répartie uniformément sur toute la grappe ('REPARTIE') ;
- soit concentrée au centre de gravité ('CDG') ;
- soit répartie uniformément dans la partie de la grappe qui se situe avant et après le retreint ;
- soit appliquée à l'extrémité de la grappe ('PTREP').

Si le mot clé prend la valeur 'CDG', il faut renseigner obligatoirement le mot clé MASS_INER.

4.46.3.5 Mot clé MASS_INER

```
si APPL_FORC_ARCHI = 'CDG' ou
si APPL_FORC_FPLAQ = 'CDG' ou
si APPL_FORC_FMEC = 'CDG' ou
si APPL_FORC_FTG = 'CDG'

◆ MASS_INER = mass_iner, [tabl_mass_iner]
```

Ce mot clé n'est à renseigner que dans les cas où le point d'application d'une force est le centre de gravité.

4.46.3.6 Mot clé DIRE_FORC_PLAQ

```
◇ DIRE_FORC_FPLAQ = (n1,n2,n3), [l_R]
```

La direction du vecteur de la force de plaquage est éventuellement donnée sous ce mot clé. Si ce mot clé n'est pas renseigné, la direction de la force est colinéaire à la grappe.

4.46.3.7 Mot clé UNITE_IMPR_FORCE

◇ UNITE_IMPR_FORCE = i1, [I]

Unité logique d'impression des forces fluides :

- Force d'Archimède ;
- Force de plaquage (FPLAQ) ;
- Forces dans le mécanisme de levée : force de pression (FPMEC) et une force visqueuse (FMEC) ;
- Forces dans le tube guide : force de pression (FPTG) et une force visqueuse (FTG) ;
- Forces dans le dashpot : force de pression (FPTG), une force visqueuse (FTG) et force du retreint (FTG').

4.46.3.8 Mot clé UNITE_IMPR_NOEUD

◇ UNITE_IMPR_NOEUD = i2, [I]

Unité logique d'impression des nœuds de la grappe par zone :

- Nœuds situés dans le mécanisme de levée (zone 1) ;
- Nœuds situés dans le guidage continu (zone 2) ;
- Nœuds situés dans le tube guide, dans le dashpot, dans le retreint (zone 3).

4.46.4 Position des points d'application des forces

Zone 1 : mécanisme de levée
Zone 2 : guidage continu
Zone 3 : tube guide / dahpot

