

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 EVOL_CHAR (AFFE_CHAR_MECA seulement).....	9
4.6 Opérande PESANTEUR (AFFE_CHAR_MECA seulement).....	9
4.7 Opérande ROTATION (AFFE_CHAR_MECA seulement).....	10
4.8 Opérande SIGM_INTERNE (AFFE_CHAR_MECA seulement).....	10
4.9 Mot-clé DDL_IMPO.....	11
4.10 Mot-clé FACE_IMPO.....	13
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.....	29
4.16 Mot-clé CONTACT (voir commande DEFI_CONTACT [U4.44.11]).....	31
4.17 Mot-clé FORCE_NODALE.....	32
4.18 Mot-clé LIAISON_SOLIDE.....	33
4.19 Mot-clé LIAISON_ELEM.....	35
4.20 Mot-clé LIAISON_UNIF.....	38
4.21 Mot-clé LIAISON_CHAMNO.....	39
4.22 Mot-clé CHAMNO_IMPO.....	40
4.23 Mot-clé LIAISON_INTERF.....	41
4.24 Mot-clé VECT_ASSE.....	42
4.25 Mot-clé FORCE_FACE.....	43
4.26 Mot-clé FORCE_ARETE.....	44
4.27 Mot-clé FORCE_CONTOUR.....	45
4.28 Mot-clé FORCE_INTERNE.....	46
4.29 Mot-clé PRES_REP.....	47
4.30 Mot-clé EFFE_FOND.....	49
4.31 Mot-clé EPSI_INIT.....	50
4.32 Mot-clé FORCE_POUTRE.....	52
4.33 Mot-clé DDL_POUTRE.....	54
4.34 Mot-clé FORCE TUYAU.....	55
4.35 Mot-clé FORCE_COQUE.....	56

4.36 Mot-clé LIAISON_COQUE.....	59
4.37 Mot-clé RELA_CINE_BP.....	61
4.38 Mot-clé FORCE_ELEC.....	62
4.39 Mot-clé INTE_ELEC.....	65
4.40 Mot-clé IMPE_FACE (Phénomène 'ACOUSTIQUE').....	67
4.41 Mot-clé VITE_FACE (Phénomène 'ACOUSTIQUE').....	68
4.42 Mot-clé ONDE_PLANE.....	69
4.43 Mot-clé ONDE_FLUI (Phénomène 'ACOUSTIQUE').....	71
4.44 Mot-clé FLUX_THM_REP	72
4.45 Mot-clé ARLEQUIN.....	73

2 Syntaxe générale

```

ch      [char_meca] = AFFE_CHAR_MECA
      (
          ♦      MODELE = mo,
          [modele]
          ♦      | VERI_NORM      =      /
          'OUI',
          [DEFAULT]
          /      'NON',
          | LIAISON_XFEM=      'OUI'
          | EVOL_CHAR=      evch
          [evol_char]
          | ROTATION=
          (omega, ar, br, cr)      [l_R]
          | SIGM_INTERNE=      sigm
          /      [carte_sdaster]

          /      [cham_elem]
          | PESANTEUR=_F      (voir
mot-clé PESANTEUR      [$ 4.6])
          | DDL_IMPO=_F      [$ 4.9])
          (voir mot-clé DDL_IMPO
          | 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      [U4.44.11])
          (voir document CONTACT
          | 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_INTERF=_F (voir
mot-clé LIAISON_INTERF      [$ 4.23])
          | LIAISON_UNILATER=_F (voir document
CONTACT      [U4.44.11])
          | VECT_ASSE=_F (voir
mot-clé VECT_ASSE      [$ 4.24])
          milieu continu | FORCE_FACE=_F (voir mot-clé FORCE_FACE
          [$ 4.25])
          | FORCE_ARETE=_F
          (voir mot-clé FORCE_ARETE      [4.26])

```

```

mot-clé FORCE_CONTOUR      [$ 4.27])      | FORCE_CONTOUR=_F (voir
                                     | FORCE_INTERNE=_F (voir
mot-clé FORCE_INTERNE      [$ 4.28])      | PRES_REP=_F
                                     |   [$ 4.29])
                                     | EFFE_FOND=_F
                                     |   [$ 4.30])
                                     | EPSI_INIT=_F
                                     |   [$ 4.31])
poutre coque               | FORCE_POUTRE=_F (voir mot-clé FORCE_POUTRE
[$ 4.32])
                                     | DDL_POUTRE =_F
                                     |   [$ 4.33])
                                     | FORCE TUYAU=_F
                                     |   [$ 4.34])
                                     | FORCE COQUE=_F
                                     |   [$ 4.35])
                                     | LIAISON_COQUE=_F (voir
mot-clé LIAISON_COQUE      [$ 4.36])      |
béton                      | RELA_CINE_BP=_F (voir mot-clé
RELA_CINE_BP               [$ 4.37])      |
électroméca               | FORCE_ELEC=_F (voir mot-clé
FORCE_ELEC                 [$ 4.38])      |
                                     | INTE_ELEC=_F
                                     |   [$ 4.39])
                                     | (voir mot-clé
(voir mot-clé INTE_ELEC    |
acoustique                | IMPE_FACE=_F
IMPE_FACE                  [$ 4.40])      |
                                     | VITE_FACE=_F
                                     |   [$ 4.41])
                                     | ONDE_FLUI=_F
                                     |   [$ 4.42])
                                     | ONDE_PLANE=_F
                                     |   [$ 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])      |

                                     ◇      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 document CONTACT          [U4.44.11])
              | 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 document
              CONTACT          [U4.44.11])
              milieu continu          | FORCE_FACE=_F          (voir mot-clé FORCE_FACE
              [$ 4.25])
              (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          [$ 4.29])
              (voir mot-clé PRES_REP          [$ 4.29])
              | EFFE_FOND=_F          [$ 4.30])
              (voir mot-clé EFFE_FOND          [$ 4.30])
              | EPSI_INIT=_F          [$ 4.31])
              (voir mot-clé EPSI_INIT          [$ 4.31])
              poutre coque          | FORCE_POUTRE=_F (voir mot-clé FORCE_POUTRE
              [$ 4.32])
              (voir mot-clé FORCE_TUYAU          [$ 4.34])
              | FORCE_COQUE=_F          [$ 4.35])
              (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])
              (voir mot-clé VITE_FACE          [$ 4.40])
              | ONDE_PLANE=_F          [$ 4.41])
              (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 AFFE_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 AFFE_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	AFFE_CHAR_MECA_F	EPSI_INIT
CHAR_MECA_EPSI_R	AFFE_CHAR_MECA	EPSI_INIT
CHAR_MECA_FF1D1D	AFFE_CHAR_MECA_F	FORCE_POUTRE
CHAR_MECA_FF1D2D	AFFE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FF1D3D	AFFE_CHAR_MECA_F	FORCE_ARETE
CHAR_MECA_FF2D2D	AFFE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FF2D3D	AFFE_CHAR_MECA_F	FORCE_FACE
CHAR_MECA_FF3D3D	AFFE_CHAR_MECA_F	FORCE_INTERNE
CHAR_MECA_FFCO2D	AFFE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FFCO3D	AFFE_CHAR_MECA_F	FORCE_COQUE
CHAR_MECA_FLUX_F	AFFE_CHAR_MECA_F	FLUX_THM_REP
CHAR_MECA_FLUX_R	AFFE_CHAR_MECA	FLUX_THM_REP
CHAR_MECA_FORC_F	AFFE_CHAR_MECA_F	FORCE_NODALE
CHAR_MECA_FORC_R	AFFE_CHAR_MECA	FORCE_NODALE
CHAR_MECA_FR1D1D	AFFE_CHAR_MECA	FORCE_POUTRE
CHAR_MECA_FR1D2D	AFFE_CHAR_MECA_F	FORCE_CONTOUR
CHAR_MECA_FR1D3D	AFFE_CHAR_MECA	FORCE_ARETE
CHAR_MECA_FR2D2D	AFFE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FR2D3D	AFFE_CHAR_MECA	FORCE_FACE
CHAR_MECA_FR3D3D	AFFE_CHAR_MECA	FORCE_INTERNE
CHAR_MECA_FRCO2D	AFFE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRCO3D	AFFE_CHAR_MECA	FORCE_COQUE
CHAR_MECA_FRELEC	AFFE_CHAR_MECA	FORCE_ELEC
CHAR_MECA_PESA_R	AFFE_CHAR_MECA	PESANTEUR
CHAR_MECA_PRES_F	AFFE_CHAR_MECA_F	PRES_REP
CHAR_MECA_PRES_R	AFFE_CHAR_MECA	PRES_REP
CHAR_MECA_ROTA_R	AFFE_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] :
lorsque différentes occurrences d'un même mot clé facteur existent, c'est la dernière affectation qui prime.

Les mots clés facteurs différents se cumulent toujours.

Si par exemple, l'utilisateur fait :

```
FORCE_FACE (GROUP_MA='G1', FX=12.)  
PRES_REP   (GROUP_MA='G1', PRES=13.)
```

et si la normale pour *G1* est orientée selon *X*,

alors tout se passera comme si on avait fait :

```
FORCE_FACE (GROUP_MA='G1', FX=25.)
```

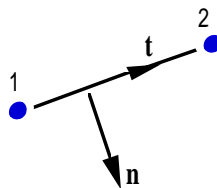

4.1.4 Éléments structuraux, milieux continus

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.

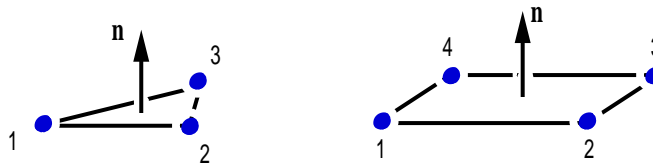
4.1.5 Normales et tangentes aux mailles

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.

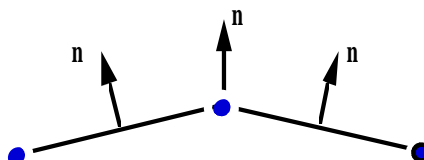


Tangentes :

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 degrés de liberté 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 EVOL_CHAR (AFFE_CHAR_MECA seulement)

| EVOL_CHAR = evch,

Chargements évolutifs dans le temps de type 'evol_char' 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.

4.6 Opérande PESANTEUR (AFFE_CHAR_MECA seulement)

```
| PESANTEUR =_F ( ◆ /
GRAVITE = G , [R]

◆ / DIRECTION = (ap, bp, cp) ,
[l_R]

◇ / MAILLE = lma ,
[l_maille]

/ GROUP_MA = lgma,
[l_gr_maille]
)
```

g représente l'intensité du champ de pesanteur et le vecteur $DIRECTION$ précise la direction et le sens d'application du champ. 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]`).

Par défaut, ce champ s'applique à tout le modèle. Il est possible de le restreindre à une partie du modèle à l'aide des mot-clés `MAILLE` et `GROUP_MA`, qui précisent les mailles sur lesquelles le champ s'applique.

Remarques :

Il peut exister une différence entre la solution théorique du calcul du poids de la structure et la solution EF. Cela est due à la discrétisation du problème.

Lorsque le chargement PESANTEUR est utilisé lors d'un MECA_STATIQUE, Code_Aster calcule les efforts aux nœuds en utilisant la matrice de raideur de l'élément et les déplacements précédemment calculés (option `EFGE_ELNO_DEPL`). On retrouve donc bien le poids de la structure là où les conditions de blocages sont imposées.

Si le chargement PESANTEUR est utilisé lors d'un STAT_NON_LINE, Code_Aster fait la somme des contraintes aux nœuds à partir des contraintes aux points de Gauss `SIEF_ELNO_ELGA`. Et cela ne donne pas la même chose que `MECA_STATIQUE`, car on s'impose, lors d'un `STAT_NON_LINE`, à un nœud à la fois des conditions de déplacement et d'effort (ici venant de la pesanteur), ces efforts ne sont pas pris en compte. La seule façon de retrouver le poids de la structure est :

- D'utiliser `MECA_STATIQUE`

- Lors d'une utilisation avec `STAT_NON_LINE` de faire en sorte que les éléments finis, sur lesquels des conditions cinématiques sont imposées, soient d'une taille suffisamment petite pour que leur poids soit négligeable devant celle de la structure totale.

- Lors d'une utilisation d'éléments de poutres avec un `STAT_NON_LINE`, une solution est de dédoubler les nœuds sur lesquels la condition cinématique est imposée et de faire par exemple une `LIAISON_ELEM` entre les 2 nœuds ou d'utiliser les discrets.

4.7 Opérande ROTATION (AFFE_CHAR_MECA seulement)

```
|      ROTATION =_F      (      ♦      VITESSE =
      omega ,      [R]
      AXE =      (ar, br, cr) ♦
      [1_R]
      =      (x, y, z),      [1_R] CENTRE
      /      MAILLE =      lma ,
      [1_maille]
/      GROUP_MA =      lgma,
      [1_gr_maille]
      /      TOUT =      'OUI',
      ♦      VITESSE =      omega ,
```

Vitesse de rotation

◆ AXE = (ar, br, cr) ,
direction de l'axe de rotation qui conduit à :

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

Le chargement qui en résulte est : $\rho(\omega \wedge \mathbf{OM}) \wedge \omega$ 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]`).

$$\diamond \quad \text{CENTRE} = (\bar{x}, \bar{y}, \bar{z}),$$

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

Limitations :

- **modélisations planes** : l'axe de rotation doit être dans la direction Oz (direction normale au plan), le centre peut être quelconque.
- **modélisations axisymétriques et Fourier** : l'axe de rotation doit être dans la direction Oy , le centre doit être l'origine (sinon le chargement n'est pas axisymétrique).

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 $\rho(\omega \wedge \mathbf{OM}) \wedge \omega$ 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`.

4.8 Opérande SIGM INTERNE (AFFE CHAR MECA seulement)

| SIGM INTERNE = sigm,

Ce chargement permet d'appliquer des contraintes volumiques (2D ou 3D) à un domaine volumique. Le champ de contraintes `sigm` est de type `carte` ou `chamelem elga`.

Il peut provenir de CREA CHAMP ou avoir été calculé par ailleurs.

Ce champ de contraintes internes, imposées, est utilisé comme second membre dans les résolutions de MECA STATIQUE et STAT NON LINE.

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).

Lors d'un calcul avec la méthode X-FEM, il est possible d'imposer le déplacement de nœuds enrichis. (AFFE_CHAR_MECA seulement). Cela se fait de manière habituelle (bien que ces nœuds ne possèdent pas de degré de liberté DX , DY ou DZ).

Remarque ; si le nœud demandé est sur les lèvres, alors on impose la condition de blocage sur les nœuds des lèvres supérieure et inférieure.

4.9.2 Syntaxe

- pour AFFE_CHAR_MECA

```

      TOUT | DDL_IMPO=_F ( ♦ /
      = '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 = thetax, [R]

      | DRY = thetay , [R]

      | DRZ = thetaz , [R]

      | GRX = g ,
      [R]

      | PRES= p ,
      [R]

      | PHI = phi ,
      [R]

      | TEMP= T ,
      [R]
```

```

[R]          |          PRE1=          pr1          ,
[R]          |          PRE2=          pr2          ,
              ...
              |          LAGS_C=    lag,          [R]
              |          V11=          v11,          [R]
              |          V12=          v12,          [R]
              |          V21=          v21,          [R]
              |          V22=          v22,          [R]
              |          PRES11=  pres11,  [R]
              |          PRES12=  pres12,  [R]
              |          PRES21=  pres21,  [R]
              |          PRES22=  pres22,  [R]
/          LIAISON=          'ENCASTRE'
)
```

La liste exhaustive des degrés de liberté 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, H1X, H1Y, H1Z, E1X, E1Y, E1Z, E2X, E2Y, E2Z, E3X, E3Y, E3Z, E4X, E4Y, E4Z, LAGS_C, V11, V12, V21, V22, PRES11, PRES12, PRES21, PRES22

- pour AFFE_CHAR_MECA_F


```

      |          DDL_IMPO=_F          (          ♦          /
TOUT          =          'OUI',

/          NOEUD          =          lno          ,
[l_noeud]

/          GROUP_NO =          lgrno,          [l_gr_noeud])

/          MAILLE          =          lma          ,
[l_maille]

/          GROUP_MA =          lgma,          [l_gr_maille]

♦          /          |          DX          =

...

[fonction]

/          LIAISON =          'ENCASTRE'

)
```

4.9.3 Opérandes

```
|          DDL_IMPO
```

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

- $DX = ux$ ou uxf | Valeur de la composante de déplacement en **translation** imposée sur les nœuds spécifiés
- $DY = uy$ ou uyf
- $DZ = uz$ ou uzf

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 pf | 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 :

- $DZ = uz$ ou uzf | 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)
- $PRES1 = p1$ | 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)
- $PRES2 = p2$ | 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 degrés de liberté de coque :

U : gauchissement
 I : "in plane"

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

Soit :

- $UI2 \quad VI2 \quad WI2 \quad UO2 \quad VO2 \quad WO2$ Degrés de liberté liés au mode 2
- $UI3 \quad VI3 \quad WI3 \quad UO3 \quad VO3 \quad WO3$ Degrés de liberté liés au mode 3
- $WO \quad WI1 \quad WO1$ Degrés de liberté de gonflement et mode 1 sur W

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

- $UI4 \quad VI4 \quad WI4 \quad UO4 \quad VO4 \quad WO4$ Degrés de liberté liés au mode 4

- UI5 VI5 WI5 UO5 VO5 WO5 Degrés de liberté liés au mode 5
- UI6 VI6 WI6 UO6 VO6 WO6 Degrés de liberté liés au mode 6

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

- GONF gonflement

Uniquement si les nœuds spécifiés appartiennent à des éléments de régularisation second gradient :

- V11 V12 V21 Composante du tenseur de déformation microscopique
V22
- PRES11 PRES12 Multiplicateurs de Lagrange introduits pour la formulation mixte
PRES21 PRES22

Uniquement si les nœuds spécifiés appartiennent à des éléments de régularisation second gradient micro-dilatation :

- GONF Gonflement
- PRES Multiplicateur de Lagrange introduit pour la formulation mixte

LIAISON = 'ENCASTRE'

Permet d'encastrer directement des nœuds, c'est à dire de forcer à zéro les degrés de liberté de translation et de rotation. Les autres degrés de liberté ne sont pas modifiés.

4.9.4 Vérifications et recommandations

On vérifie que le degré de liberté 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

```

MAILLE | FACE_IMPO=_F ( ◆ /
[l_maille] = lma ,

/ GROUP_MA = lgma,
[l_gr_maille]

◇ SANS_MAILLE = lma1,
[l_maille]
```



```

    ◇
    [l_gr_maille]          SANS_GROUP_MA      =          lgma1,

    ◇
    [l_noeud]              SANS_NOEUD         =          lnol,

    SANS_GROUP_NO         =          lgnol,
    [l_gr_ noeud]

    ◆      /      |      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      =          phi,
              [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]

    ◇
    [l_maille]          SANS_MAILLE         =          lma1,

```

```

◇          SANS_GROUP_MA      =      lgma1,
[l_gr_maillage]

◇          SANS_NŒUD          =      lnol,
[l_noeud]

          SANS_GROUP_NO      =      lgnol,
[l_gr_noeud]

♦          /          |          DX          =          uxf
,

          |          DY          =          uyf          ,
          [fonction]

          |          DZ          =          uzf          ,
          [fonction]

          |          DRX          =          θ xf          ,
          [fonction]

          |          DRY          =          θ yf          ,
          [fonction]

          |          DRZ          =          θ zf          ,
          [fonction]

          |          GRX          =          gf          ,
          [fonction]

          |          PRES          =          pf          ,
          [fonction]

          |          PHI          =          ϕ f          ,
          [fonction]

          |          TEMP          =          Tf          ,
          [fonction]

          |          PRE1          =          pr1f,
[fonction]

          |          PRE2          =          pr2f,
[fonction]

/          |          DNOR          =          un          ,
          [fonction]

          |          DTAN          =          ut          ,
          [fonction])

```

```

◇                SANS_MAILLE                =                lma1,
[1_maillage]
◇                SANS_GROUP_MA                =lgm1,
[1_gr_maillage]
◇                SANS_NEUD                    =                lno1,
[1_noeud]
◇                SANS_GROUP_NO                =                lgno1,
[1 gr noeud]

```

[illegible]

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

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

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).

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]),

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(
    lno,
    /
    [l_gr_noeud]
    GROUP_NO = lgrno,
    DDL = lddl,
    [l_K8]
    COEF_MULT =  $\alpha_i$ ,
    [l_R]
    COEF_IMPO =  $\beta$ ,
    [R]
)

```
- pour AFFE_CHAR_MECA_F


```

LIAISON_DDL=_F(
    lno,
    /
    [l_gr_noeud]
    GROUP_NO = lgrno,
    DDL = lddl,
    [l_K8]
    /
    COEF_MULT =  $\alpha_i$ ,
    [l_R]
    /
    COEF_MULT_FONC =  $\alpha_{if}$ ,
    [l_fonction]
    COEF_IMPO =  $\beta^f$ ,
    [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 degrés de liberté U_i ($i=1,r$) de r textes à prendre dans la documentation des grandeurs simples [U2.01.04]

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 degrés de liberté dans la relation. Exemple : pour imposer $U_x = U_y$ sur le nœud *NI*, 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.

4.11.4.4 Coefficients multiplicateurs dépendant de la géométrie

Pour AFPE_CHAR_MECA_F, on peut rentrer des coefficients multiplicateurs dépendant de la géométrie avec COEF_MULT_FONC. Néanmoins, ces coefficients sont calculés à partir de la géométrie initiale, il n'y a pas de réactualisation éventuelle en non linéaire.

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]

[gr_noeud]      /      GROUP_NO =      gno      ,

      ♦      |      DX      =      ux
      ,      [R]

      |      DY      =      uy
      ,      [R]

      |      DZ      =      uz
      ,      [R]

      |      DRX      =      θ x
      ,      [R]

      |      DRY      =      θ y
      ,      [R]

      |      DRZ      =      θ z
      ,      [R]

      ♦      ANGL_NAUT =      ( α , β , γ ) ,
[l_R]

)
    
```

- pour AFFE_CHAR_MECA_F

```

/      |      LIAISON_OBLIQUE =_F      (      ♦
      NOEUD      =      no      ,      [noeud]

[gr_noeud]      /      GROUP_NO =      gno      ,

      ♦      |      DX      =      uxf
      ,      [fonction]

      |      DY      =      uyf
      ,      [fonction]

      |      DZ      =      uzf
      ,      [fonction]
    
```



```

[fonction] | DRX =  $\theta_{xf}$ ,

[fonction] | DRY =  $\theta_{yf}$ ,

[fonction] | DRZ =  $\theta_{zf}$ ,

♦ ANGL_NAUT =  $(\alpha, \beta, \gamma)$ ,
[l_R]
)

```

4.12.3 Opérandes

LIAISON_OBLIQUE	
• DX = ux ou uxf	Valeur de la composante de déplacement en translation dans le repère oblique imposée sur les nœuds spécifiés
• DY = uy ou uyf	
• DZ = uz ou uzf	

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

• DRX = θ_x ou θ_{xf}	Valeur de la composante de déplacement en rotation dans le repère oblique imposée sur les nœuds spécifiés
• DRY = θ_y ou θ_{yf}	
• DRZ = θ_z ou θ_{zf}	

♦ ANGL_NAUT = (α, β, γ) ,

Les angles nautiques (α, β, γ) 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 degré de liberté 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]

[1_gr_maille]      /      GROUP_MA_1      =      lgma1,

      ♦      /      MAILLE_2      =      lma2,
[l_maille]

[1_gr_maille]      /      GROUP_MA_2      =      lgma2,

/      ♦      /      NOEUD_1      =      lno1,
[l_noeud]

[l_gr_noeud]      /      GROUP_NO_1      =      lgno1,

      ♦      /      NOEUD_2      =      lno2,
[l_noeud]

[l_gr_noeud]      /      GROUP_NO_2      =      lgno2,

♦      /      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      =      α 1i
,      [1_R]

♦      COEF_MULT_2      =      α 2i
,      [1_R]

♦      COEF_IMPO      =
β ,      [R]

◊      SOMMET      =
'OUI',

◊      CENTRE      =
lr ,      [1_R]

◊      ANGL_NAUT      =      lr
,      [1_R]

◊      TRAN      =
lr ,      [1_R]
)
```

```

• pour AFFE_CHAR_MECA_F
  LIAISON_GROUPE=F (
    MAILLE_1 = lma1, / [l_maillage] /

    / GROUP_MA_1 = lgm

    [l_gr_maillage]

    [l_maillage] / MAILLE_2 = lma2,

    / GROUP_MA_2 = lgma2,

    [l_gr_maillage]

    / [l_gr_noeud] / NOEUD_1 = lno1, [l_noeud]

    / GROUP_NO_1 = lgno1,

    [l_gr_noeud]

    [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      =      α 1i ,
    [1_R]

COEF_MULT_2      =      α 2i ,
    [1_R]

COEF_IMPO        =      β f      ,
    [fonction]

SOMMET           =      'OUI' ,

CENTRE           =      lr      ,
    [1_R]

ANGL_NAUT        =      lr      ,
    [1_R]

TRAN             =      lr      ,
    [1_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.

◇ / 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 effectuée 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 :

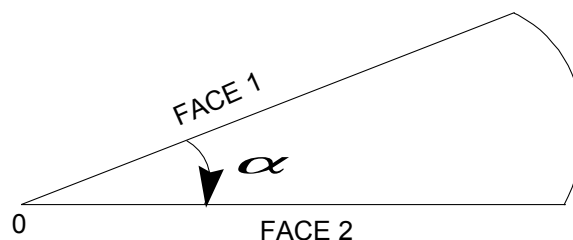
- On vérifie que les ddl spécifiés dans ces opérandes existent pour chacun des nœuds des éléments affectés dans le *MODELE* aux mailles qui contiennent le nœud,
- pour utiliser l'argument 'DNOR', il est obligatoire d'avoir déclaré les bords à l'aide de mailles et que le calcul d'une normale sur ces mailles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

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

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,
                     = lma2, MAILLE_2
                     = 'DNOR', DDL_1
                     = 'DNOR', DDL_2
                     = COEF_MULT_1
                     = 1., COEF_MULT_2
                     = -1., COEF_IMPO
                     = 0, CENTRE
                     = (X0,Y0,Z0), ANGL_NAUT
                     = (alpha,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 :

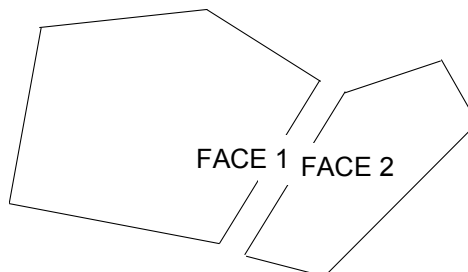
a) une condition de périodicité (étude d'une cellule d'homogénéisation)



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 AFFE_CHAR_MECA seulement)

```

LIAISON_MAIL      =_F      (
    ◇              TYPE_RACCORD      =      / 'MASSIF'
                                                    [DEFAULT]

/ 'COQUE'

/ 'COQUE_MASSIF'

/ 'MASSIF_COQUE'

    ◆              |      GROUP_NO_ESCL      =      lgn2
    ,
    [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]

    # si TYPE_RACCORD = 'MASSIF' :
    ◇              |      ◆      CENTRE      =
    (xc, yc, [zc]),      ◆      [l_R]
    (alpha, [beta, gamma]),      ANGL_NAUT      =
    |      ◆      [l_R]
    (tx, ty, [tz]),      ◆      TRAN      =
    ◇              ◆      DDL_MAIT      =
    'DNOR',      ◆      DDL_ESCL      =
    'DNOR',

    # si TYPE_RACCORD = 'COQUE_MASSIF' :
    ◆      EPAIS      =      epais,

    [l_R]
    CHAM_NORMALE      =      chanor,

    [cham_no]

    ◇      ELIM_MULT      =      /
    'NON',
    [DEFAULT]

/      'OUI',
)

```

4.14.3 Opérandes

4.14.3.1 Choix de la surface esclave et de la surface maître

Le principe de la liaison est d'éliminer les ddls esclaves en les écrivant comme des relations linéaires des ddls maitres. Il y a une certaine symétrie dans le problème et on pourrait croire que l'on peut choisir au hasard qui sera le maitre et qui sera l'esclave.

En réalité, il faut être attentif sur deux points particuliers :

- La syntaxe n'est pas symétrique : côté esclave, l'utilisateur doit préciser les nœuds à « souder », alors que côté maitre, il doit donner des mailles. De plus, les mailles maitres sont (pour l'instant) d'une dimension topologique à ce qui serait naturel. Par exemple, pour un maillage 2D, les surfaces à recoller sont des lignes, et on pourrait s'attendre à ce que les mailles maitres soient des segments. Le code attend des mailles surfaciques (quadrangles et triangles).
- Il est préférable (d'un point de vue mécanique) de choisir comme surface esclave la surface maillée la plus finement. De la même façon que lorsqu'on soude 2 tôles, il vaut mieux multiplier les points de soudure.

4.14.3.2 TYPE_RACCORD

Ce mot-clé permet de choisir le type des relations linéaires que l'on va écrire pour éliminer les ddls des nœuds esclaves.

• Si `TYPE_RACCORD='MASSIF'`, les nœuds sont supposés porter des ddls de translation (`DX, DY, DZ`). Si l'utilisateur ne précise pas `DDL_MAIT='DNOR'`, on écrira (par exemple en 2D), 2 relations linéaires pour chaque nœud esclave : l'une pour éliminer son '`DX`', l'autre pour éliminer son '`DY`'.

• Si `TYPE_RACCORD='COQUE'`, les nœuds sont supposés porter des ddls de translation (`DX, DY, DZ`) et des ddls de rotation (`DRX, DRY, DRZ`). On écrira 6 relations linéaires pour éliminer les 6 ddls de chaque nœud esclave.

• Si `TYPE_RACCORD='MASSIF_COQUE'`, les nœuds esclaves sont supposés « massifs » (translations : `DX, DY, DZ`) et les nœuds maitres sont supposés de type « coque » (3 translations et 3 rotations).

Les ddls de translation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux translations du point « maitre » en vis à vis. Les translations du point maitre sont calculées comme si le petit segment de normale à la coque restait rigide.

• Si `TYPE_RACCORD='COQUE_MASSIF'`, les nœuds esclaves sont supposés de type « coque » (6ddls : `DX, DY, DZ, DRX, DRY, DRZ`) et les nœuds maitres sont supposés de type « massif » (`DX, DY, DZ`).

Les ddls de translation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux translations du point « maitre » en vis à vis.

Les ddls de rotation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux rotations du point « maitre » en vis à vis (A). Les rotations du point A sont calculées à partir des translations de deux autres points A1 et A2 situés à $+h/2$ et $-h/2$, si h est un vecteur normal à la coque et dont la longueur est l'épaisseur de la coque (voir mots clés `EPAIS` et `CHAM_NORMALE`).

4.14.3.3 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".

4.14.3.4 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

4.14.3.5 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.

4.14.3.6 DDL_MAÎT / DDL_ESCL

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

```
DDL_MAÎT = '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.

4.14.3.7 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_MAÎT='GM1'),  
_F(GROUP_MA_ESCL='GE', GROUP_MA_MAÎT='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_MAÎT=('GM1','GM2'))
```

4.14.3.8 CHAM_NORMALE = chnor, EPAIS = epais

Ces deux mots clés sont obligatoires si TYPE_RACCORD = 'COQUE_MASSIF'.

Epais est l'épaisseur de la coque au niveau de la liaison (supposée constante).

Chnor est un champ aux nœuds qui contient la direction de la normale à la coque sur les nœuds des mailles « maitres ».

Le champ chnor peut être obtenu par la commande :

```
CHNOR = CREA_CHAMP( TYPE_CHAM = 'NOEU_GEOM_R', OPERATION = 'NORMALE',  
                    MODELE = MODEL, GROUP_MA = 'GMCOQU' )
```

4.14.3.9 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_MAIT= ...)  
    _F(GROUP_MA_ESCL='LIGNE_BC', GROUP_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 pas les mêmes, l'élimination n'est pas redondante.

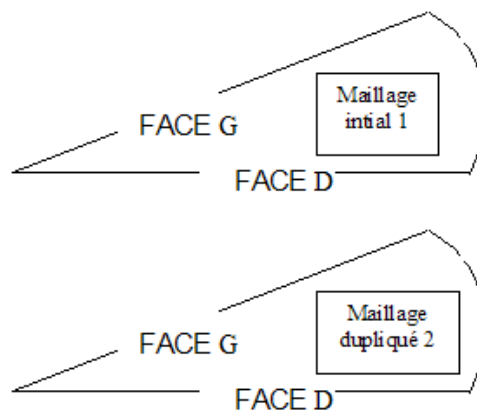
4.15 Mot-clé LIAISON_CYCL

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 :

$$\begin{bmatrix} q_g^1 \\ q_g^2 \end{bmatrix} = \begin{bmatrix} \cos \beta & \sin \beta \\ -\sin \beta & \cos \beta \end{bmatrix} \begin{bmatrix} q_d^1 \\ q_d^2 \end{bmatrix}$$

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      =      lgn2
    ,
    [l_gr_noeud]    |      NOEUD_ESCL          =      lno2
    ,
    [l_noeud]       |      GROUP_MA_ESCL       =      lgma2
    ,
    [l_gr_maille]   |      MAILLE_ESCL         =      lma2
    ,
    [l_maille]      |      ♦              |      GROUP_MA_MAIT1      =      lgma1      ,
    |              |      MAILLE_MAIT1        =      [l_gr_maille]
    |              |                        =      lma1
    ,
    [l_maille]      |      GROUP_MA_MAIT2      =      lgma2      ,
    |              |      MAILLE_MAIT2        =      [l_gr_maille]
    |              |                        =      lma1
    ,
    [l_maille]      |      ♦              |      CENTRE              =
    |              |      (xc, yc, [zc]),    |      [l_R]
    |              |      [beta,gamma]),    |      (alpha,
    |              |      |                  |      [l_R]
    |              |      (tx, ty, [tz]),    |      =
    |              |      ♦              |      TRAN                  =
    |              |      |                  |      [l_R]
    |              |      ♦              |      COEF_MAIT1            =
    |              |      |                  |      alpha
    ,
    [R]              |      ♦              |      COEF_MAIT2            =
    |              |      |                  |      beta      ,
    |              |      |                  |      [R]
    |              |      ♦              |      COEF_ESCL              =
    |              |      |                  |      X      ,
    ♦              |      ♦              |      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_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 degrés de liberté esclaves.

Par exemple :

$$\text{COEF_ESCL} \begin{pmatrix} q_g^1 \\ q_g^2 \end{pmatrix} = [\text{COEF_MAIT1} \times \text{COEF_MAIT2}] \begin{bmatrix} q_d^1 \\ q_d^2 \end{bmatrix} = [\cos \beta \cdot \sin \beta] \begin{bmatrix} q_d^1 \\ q_d^2 \end{bmatrix}$$

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 (voir commande DEFI_CONTACT [U4.44.11])

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 =      lgn,
      [l_gr_noeud]
      ◆      |
      FX=      fx      ,
      [R]
      |
      FY=      fy      ,
      [R]
      |
      FZ=      fz      ,
      [R]
      |
      MX=      mx      ,
      [R]
      |
      MY=      my      ,
      [R]
      |
      MZ=      mz      ,
      [R]
      ◇      ANGL_NAUT =      (  $\alpha, \beta, \gamma$  )      [l_R]
      ,
• pour AFFE_CHAR_MECA_F
=      FORCE_NODALE=_F      (      ◆      |      NOEUD
      lno      ,      [l_noeud]
      |
      GROUP_NO =      lgn,
      [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 axisymétrie, 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_CARA_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 et on impose éventuellement les déplacements aux valeurs résultant de la translation et/ou rotation.

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 AFFE_CHAR_MECA et AFFE_CHAR_MECA_F

```

LIAISON_SOLIDE =_F (
    ♦ / MAILLE = lma
    , [l_maille]
    / GROUP_MA = lgma ,
[l_gr_maille]
    / NOEUD = lno
    , [l_noeud]
    / GROUP_NO = lgno ,
[l_gr_noeud]
    ♦ NUME_LAGR = / 'NORMAL', [DEFAULT]
    /
'APRES' ,
    ♦ | ♦ CENTRE = (xc, yc, [zc]
    , [l_R]
    ♦ ANGL_NAUT=(alpha,[beta,gamma]),[l_R]
    | ♦ TRAN =
    (tx, ty, [tz]), [l_R]
    ♦
    DIST_MIN= dmin ,
    [R]
),

```

4.18.3 Opérandes

♦ 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 :

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

où A est un nœud arbitraire du solide.

◇ CENTRE / ANGL_NAUT / TRAN :

Ces mots-clés permettent de définir la transformation géométrique (rotation et/ou translation) permettant de déterminer les déplacements imposés à la structure.

Si ces mots-clés sont absents, les déplacements imposés sont nuls.

Il est déconseillé actuellement d'utiliser les mots-clé `CENTRE` et `ANGL_NAUT`.

TRAN = (tx, ty, [tz]) : composantes de la translation imposée à la structure.

◇ DIST_MIN : dmin

Ce mot-clé sert à définir une distance (dans les unités du maillage) en dessous de laquelle on considère que les points du maillage sont confondus. Cette distance sert aussi à déterminer si des points sont alignés, c'est-à-dire s'ils se trouvent dans un cylindre de diamètre inférieur à $dmin$.

Par défaut, $dmin = 0.001 * armin$, où $armin$ est la plus petite arête du maillage.

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',
                                / 'PLAQ_POUT_ORTH',
  # debut condition option == PLAQ_POUT_ORTH
  ◇ EXCENT_POUTRE = / 'NON',
                                / 'OUI',
                                [DEFAULT]
  # fin condition
  # debut condition option <> PLAQ_POUT_ORTH
  ◇ AXE_POUTRE = (x,y,z),
                                [l_R]
  ◇ CARA_ELEM = cara,
                                [cara_elem]
  # fin condition
  ♦ / 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 `lgn2` (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 Opérandes de l'option 'PLAQ_POUT_ORTH'

♦ `OPTION = 'PLAQ_POUT_ORTH'`

Cette option permet de raccorder une partie maillée avec des éléments `TRI3` et `QUA4` (modélisations `DKT`, `DST` et `DKTG`) avec une partie modélisée par un élément de poutre ou un discret.

♦ `/ MAILLE_1 =`
 `/ GROUP_MA_1 =`

Ces opérandes définissent les mailles de la plaque qui modélisent la trace de la section de la poutre sur cette partie. Ces mailles doivent avoir été affectées par des éléments finis de plaque, modélisations `DKT`, `DST` et `DKTG`.

♦ `/ NOEUD_2 =`
 `/ GROUP_NO_2 =`

Ces opérandes définissent le nœud à raccorder à la plaque. 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. Le nœud doit porter les degrés de libertés suivants : `DX`, `DY`, `DZ`, `DRX`, `DRY`, `DRZ`.

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

Le nœud de la poutre doit coïncider, à une tolérance près, avec le centre de gravité des mailles qui modélisent la trace de cette poutre sur la dalle. En cas de non respect de cette règle, 2 comportements sont possibles :

- si VERIF_EXCENT = 'OUI', comportement par défaut, un message d'erreur est émis et le code s'arrête en erreur fatale.
- si VERIF_EXCENT = 'NON', un message d'information est émis.

Cet opérande permet de ne pas être obligé de positionner exactement les poutres au centre de gravité de la trace de la section, qui n'est pas forcément connue lors de la réalisation du maillage. Dans le cas, où cette règle n'est pas respectée, l'utilisateur est informé de la distance entre le nœud de la poutre et ce centre de gravité soit par une erreur fatale (VERIF_EXCENT = 'OUI') soit par l'émission d'un message d'information (VERIF_EXCENT = 'NON').

Remarques :

Dans le but de simplifier l'entrée des données les vérifications suivantes ne sont pas réalisées :

- Il n'y a pas de vérification que l'axe de la poutre soit perpendiculaire à la plaque.
- Il n'y a pas de vérification entre le calcul des caractéristiques mécaniques (S,I,...) réalisé sur les mailles de la trace de la section de poutre et les caractéristiques mécaniques affectées à la poutre à l'aide CARA_ELEM.

Pour faire ces vérifications il faudrait que l'utilisateur donne en plus du nœud de la poutre, le nom de la maille affectée par le CARA_ELEM qui a pour extrémité le nœud de raccord. Dans la grande majorité des cas cette maille est inconnue de l'utilisateur, c'est le logiciel de maillage qui définit son nom.

4.19.8 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 est faite dans les cas 3D : '3D_TUYAU', '3D_POU', 'DAL_COL'.

4.19.9 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 AFFE_CHAR_MECA et AFFE_CHAR_MECA_F


```

LIAISON_UNIF =_F (
    ♦ / MAILLE = lma ,
      [l_maille]
    / GROUP_MA = lgma,
      [l_gr_maille]
    / NOEUD = lno ,
      [l_noeud]
    / GROUP_NO = lgno,
      [l_gr_noeud]
    ♦ DDL = | 'DX',
              | 'DY',
              | 'DZ',
              | 'DRX',
              | 'DRY',
              | 'DRZ',
              )
      
```

4.20.3 Opérande

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

Ces opérandes 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 avec $i=1, r$ de r textes pris parmi : 'DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ'

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

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

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 AFFE_CHAR_MECA et conduisant à un vecteur assemblé produit par ASSE_VECTEUR [U4.61.23].

4.21.2 Syntaxe (AFFE_CHAR_MECA seulement)

```

LIAISON_CHAMNO=_F(
    chamno ,
    COEF_IMPO =  $\beta$  ,
    NUME_LAGR = / 'NORMAL' , [DEFAULT]
    / 'APRES' ,
    )

```

4.21.3 Opérandes

CHAM_NO =

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

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 :

$$\begin{aligned}
 & 2.*DX(N01) + 1.*DY(N01) + 3.*DZ(N01) \\
 & + 1.*DX(N02) + 4.*DY(N02) + 2.*DZ(N02) \\
 & + 3.*DRX(N02) + 5.*DRY(N02) + 2.*DRZ(N02) = \beta
 \end{aligned}$$

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 AFFE_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 (AFFE_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 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_INTERF

4.23.1 But

Mot-clé facteur répétable et utilisable avec un modèle contenant à la fois des éléments finis et des macro-éléments statiques condensant certains sous-domaines. Il permet de définir des relations linéaires entre les degrés de liberté physiques des interfaces de la partie de modèle en éléments finis et les coordonnées généralisées de modes de représentation réduite des mouvements d'interface contenus dans certains macro-éléments de condensation statique.

4.23.2 Syntaxe (AFFE_CHAR_MECA seulement)

```
                                LIAISON_INTERF=_F(                ♦                MACR_ELEM_DYNA =
                                macrel,                            [macr_elem_dyna]
♦                                TYPE_LIAISON =                    /                'RIGIDE',                [DEFAULT]
                                /                'SOUPLE'
                                ,
                                )
```

4.23.3 Opérandes

MACR_ELEM_DYNA =

Nom du `macr_elem_dyna` qui sert à définir les relations linéaires entre les degrés de liberté physiques de l'interface entre le domaine non condensé modélisé en éléments finis et un domaine condensé par le macro-élément et les composantes des nœud assimilés à des coordonnées généralisées de modes de mouvements d'interface. Cela est nécessaire seulement quand les modes de mouvements d'interface sont une base réduite de tous les modes contraints correspondant chacun à un mode de déplacement pour chaque degré de liberté physique de l'interface. On génère ainsi des relations de type `LIAISON_DDL` dont les coefficients sont calculés de façon transparente pour l'utilisateur entre les nœuds de l'interface dynamique du macro-élément et ceux associés à la base de réduction qui a servi à constituer le macro-élément.

TYPE_LIAISON =

- si 'RIGIDE', on écrit la relation entre les degrés de liberté physiques de l'interface U_Σ et les composantes des nœud assimilés à des coordonnées généralisées q de modes de mouvements d'interface Φ sous la forme de produit simple : $U_\Sigma = \Phi q$. Ce choix permet d'avoir une liaison plus rigide qu'en prenant en compte tous les modes contraints correspondant chacun à un mode de déplacement pour chaque degré de liberté physique de l'interface.
- si 'SOUPLE', on écrit la relation entre les degrés de liberté physiques de l'interface U_Σ et les composantes des nœud assimilés à des coordonnées généralisées q de modes de mouvements d'interface Φ sous la forme de produit double : $\Phi^T U_\Sigma = \Phi^T \Phi q$. Ce choix permet d'avoir une liaison plus souple qu'en prenant en compte tous les modes contraints correspondant chacun à un mode de déplacement pour chaque degré de liberté physique de l'interface.

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 = AFFE_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 (AFFE_CHAR_MECA) ou par l'intermédiaire d'un concept fonction (AFFE_CHAR_MECA_F).

4.26.2 Syntaxe

- pour AFFE_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 AFFE_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=
      [fonction]
      mzf
      ,
      )
```

4.26.3 Opérandes

$f_x, f_y, f_z, m_x, m_y, m_z$: valeurs des composantes dans le repère GLOBAL
 $f_{xf}, f_{yf}, f_{zf}, m_{xf}, m_{yf}, m_{zf}$: 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	DKT, DST, Q4G
SEG2, SEG3	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(□)

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 (`AFFE_CHAR_MECA`) ou par l'intermédiaire d'un concept fonction (`AFFE_CHAR_MECA_F`).

4.28.2 Syntaxe

- pour AFFE CHAR MECA

```

=      'OUI',
      /
      [l_maille]
      |
      [l_gr_maille]
      ◆
      ,
      |
      FY=
      [R]
      |
      FZ=
      [R]
      FORCE_INTERNE=_F
      (
      ◆
      /
      TOUT
      lma
      ,
      lgma,
      fx
      [R]
      fy
      ,
      fz
      ,
      )

```

- pour AFFE CHAR MECA F

```

=          'OUI',
          /
          |          MAILLE      =          lma          ,
          [l_maille]
          |          GROUP_MA =          lgma,
          [l_gr_maille]
          ♦          |          FX=          fx
          ,          [fonction]          fxf
          |          FY=          fyf          ,
          [fonction]
          |          FZ=          fzf          ,
          [fonction]
          )

```

4.28.3 Opérandes

$f_x, f_y, f_z,$	valeurs des composantes dans le repère GLOBAL des forces
$f_{xf}, f_{yf}, f_{zf}:$	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

```

TOUT      |      PRES_REP=_F      (      ♦      /
          =      'OUI',

/          |      MAILLE      =      lma      ,
[l_maille]

          |      GROUP_MA      =      lgma,
[l_gr_maille]

          |      FISSURE      =      fiss,
[fiss_xfem]

♦          |      PRES      =      P      ,
[R]

|          CISA_2D      =      T      ,
[R]
    )
    
```

- pour AFFE_CHAR_MECA_F

```

TOUT      |      PRES_REP=_F      (      ♦      /
          =      'OUI',

/          |      MAILLE      =      lma      ,
[l_maille]

          |      GROUP_MA      =      lgma,
[l_gr_maille]

          |      FISSURE      =      fiss,
[fiss_xfem]

♦          |      PRES      =      Pf      ,
[fonction]

|          CISA_2D      =      Tf      ,
[fonction]
    )
    
```

4.29.3 Opérandes

```
|      PRES      =      P      (Pf)
```

Valeur de la pression imposée

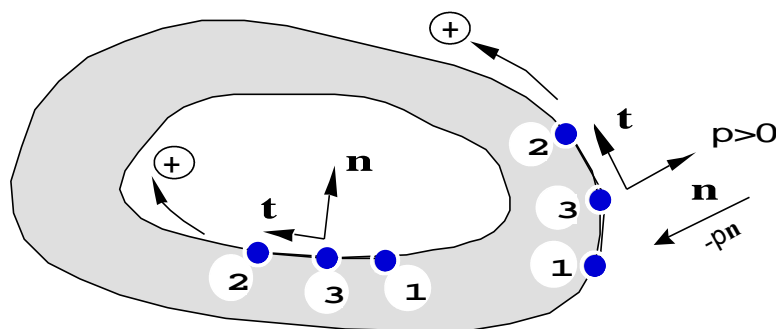
P (ou P_f) est positif suivant le sens contraire de la normale à l'élément : soit σ le tenseur des contraintes, le chargement imposé est : $\sigma_{ij} n_i n_j = -p n_i n_j$.

| CISA_2D = T (Tf)

Valeur du cisaillement imposé

T (ou T_f) 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 :



| FISSURE = fiss, [fiss_xfem]

L'imposition d'une pression sur les lèvres d'une fissure X-FEM se fait par le mot-clé spécifique FISSURE, puisque aucun groupe de maille ne correspond aux lèvres. On renseigne alors le ou les noms des fissures (provenant de la commande DEFI_FISS_XFEM [U4.82.08]) sur lesquelles on souhaite appliquer la pression.

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	3D
TRIA6, QUAD8, QUAD9	

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


```

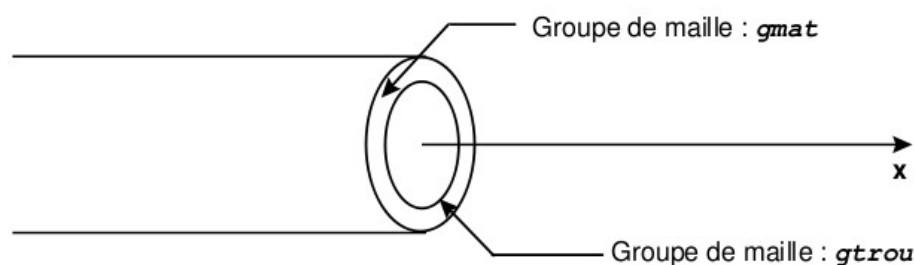
      |      EFFE_FOND =_F      (      ♦      |      MAILLE =
      |      lma      ,      [l_maille]

      |      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 :

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

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

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

◆ 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 AFFE_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]

```

4.31.3 Opérandes

Fascicule u4.44 : Conditions aux limites et chargements

Remarques :

Le second membre élémentaire calculé sera $\int_{V_e} \varepsilon_{ini} : \varepsilon(v^*) dV_e$ où ε 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 $-\frac{d\theta_y}{dx}$

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

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 AFFE_CARA_ELEM [U4.42.01].

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).

4.32.2 Syntaxe

```

• pour AFFE_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 AFFE_CHAR_MECA_F
  FORCE_POUTRE =_F ( ♦ / TOUT
= 'OUI',
  / | MAILLE = lma ,
  [l_maille]
  / | GROUP_MA = lgma,
  [l_gr_maille]
  ♦ / | FX =
  fx , [fonction]
  / | FY = fy
  , [fonction]

```

```

, | FZ = fzf
[fonction]

/ | N = nf
[fonction]

, | VY = vyf
[fonction]

, | VZ = vzf
[fonction]

◇ TYPE_CHARGE = / 'FORCE',
[DEFAULT]

/ 'VENT' ,
)

```

4.32.3 Opérandes

suivant x	◇	/		FX	:	Force
[fonction]				[R]	ou	
suivant y				FY	:	Force
[fonction]				[R]	ou	
suivant z				FZ	:	Force
[fonction]				[R]	ou	
traction - compression		/		N	:	Effort de
			[R]	ou	[fonction]	
transversal suivant y				VY	:	Effort
[fonction]				[R]	ou	
transversal suivant z				VZ	:	Effort
[fonction]				[R]	ou	

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,

\varnothing le diamètre du câble sur lequel s'exerce le vent,

alors :

$$FX = p \varnothing v_x$$

$$FY = p \varnothing v_y$$

$$FZ = p \varnothing v_z$$

```
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 AFFE_CHAR_MECA
      DDL_POUTRE =_F      (      ◆      |      NOEUD
=      lno      ,      [l_noeud]

      |      GROUP_NO =      lgno,
      [l_gr_noeud]

      ◆      |      DX =      ux      ,
      [R]

      |      DY =      uy      ,
      [R]

      |      DZ =      uz      ,
      [R]

      |      DRX =       $\theta_x$       ,
      [R]

      |      DRY =       $\theta_y$       ,
      [R]

      |      DRZ =       $\theta_z$       ,
      [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
DY = uy
DZ = uz

Valeur de la composante de déplacement en **translation** imposée sur les nœuds spécifiés

DRX = θ_x

DRY = θ_y

DRZ = θ_z

Valeur de la composante de déplacement en **rotation** imposée sur les nœuds spécifiés

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 :  
  
= 'OUI', | FORCE_TUYAU=_F( ♦ / TOUT  
  
/ | MAILLE = lma ,  
[l_maille]  
  
| GROUP_MA = lgma ,  
[l_gr_maille]  
  
♦ PRES = p ,  
[R]  
 )  
  
• AFPE_CHAR_MECA_F :  
  
= 'OUI', | FORCE_TUYAU=_F( ♦ / TOUT  
  
/ | 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',
    / [l_maille] MAILLE = lma ,
    / [l_gr_maille] GROUP_MA = lgma,
    / FX = fx
    , [R]
    / FY = fy ,
    / FZ = fz ,
    / MX = mx ,
    / MY = my ,
    / MZ = mz ,
    / PLAN = 'MOY',
    / 'INF',
    / 'SUP',
    / 'MAIL',
    [DEFAULT]
    / PRES = p,
    / [R]
    / F1 = f1 ,
    / F2 = f2 ,
    / 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 = m1f ,
[fonction]

```

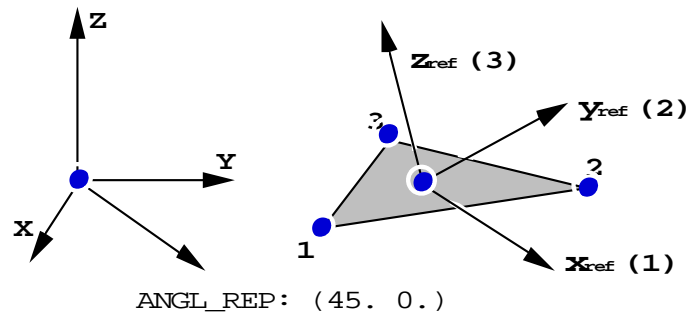


```
| MF2 = mf2f,  
[fonction]  
)
```

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 `AFPE_CARA_ELEM` [U4.42.01]).



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

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 :

$$F2 = F1$$

$$M2X = M1X - dx \cdot F1Y$$

$$M2Y = M1Y + dx \cdot F1X$$

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

$$F2 = F1$$

$$M2X = M1X - d + \frac{h}{2} \cdot F1Y$$

$$M2Y = M1Y + d + \frac{h}{2} \cdot F1X$$

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

$$F2 = F1$$

$$M2X = M1X - d - \frac{h}{2} \cdot F1Y$$

$$M2Y = M1Y + d - \frac{h}{2} \cdot F1X$$

/	'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`).

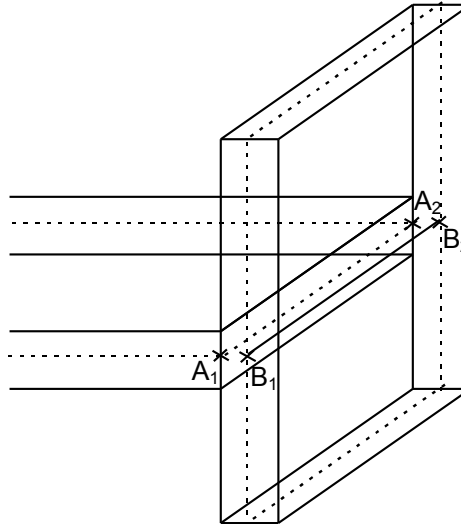
4.36 Mot-clé LIAISON_COQUE

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_maillage]
        |
        = l_ma1 ,
        [l_maillage]
        |
        = l_gno1 ,
        [l_gr_noeud]
        |
        = l_no1 ,
        [l_noeud]
      )
  
```

```

=          l_gma2      ,      ♦      |      GROUP_MA_2
[l_gr_maille]

=          l_ma2       ,      |      MAILLE_2
[l_maille]

=          l_gno2      ,      |      GROUP_NO_2
[l_gr_noeud]

=          l_no2       ,      |      NOEUD_2
[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 AFFE_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 (AFFE_CHAR_MECA seulement)

```

RELA_CINE_BP                =_F      (
                                ◆      CABLE_BP
                                =      cabl_pr,      [cabl_precont]
                                ◇      SIGM_BPEL
                                =      /      'OUI',
                                /      'NON',
                                [DEFAULT]
                                ◇      RELA_CINE
                                =      /      'OUI',      [DEFAULT]
                                /      'NON',
                                ◇
                                DIST_MIN      =      dmin,
                                [R]
                                )
    
```

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 AFFE_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 ; valeur par défaut 'OUI'.

```
◇          DIST_MIN=          dmin          ,  
          [R]          (voir LIAISON_SOLIDE 4.18)
```

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,) ,
[1_R]

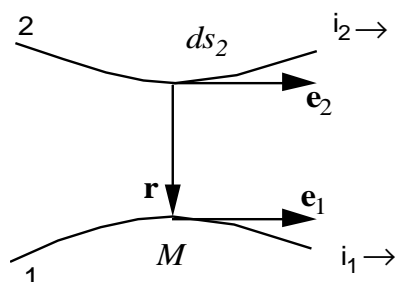
♦          POINT2          =          (x2,y2,z2,) ,
[1_R]
)

```

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 :

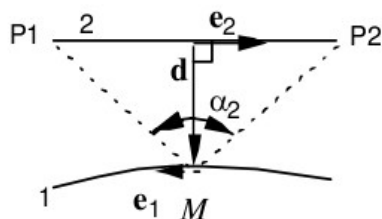
$$f(M) = \frac{e_1}{2} \wedge \int_2 \frac{e_2 \wedge r}{\|r\|^3} ds_2$$



avec $\|e_1\| = \|e_2\| = 1$

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

$$f(M) = \frac{e_1}{2} \wedge \frac{n}{d} (\sin \alpha_1 - \sin \alpha_2)$$



avec $n = \frac{e_2 \wedge d}{d}$, $d = \|d\|$, $\|d\| = 1$

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

$$f(M) = e_1 \wedge \frac{n}{d}$$

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.

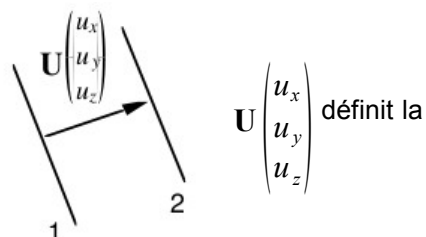
/		FX = fx,	fx2 + fy2
+ fz2 = 1.		FY = fy,	
(fx, fy, fz)		colinéaire à la force de LAPLACE	
		FZ = fz,	

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'

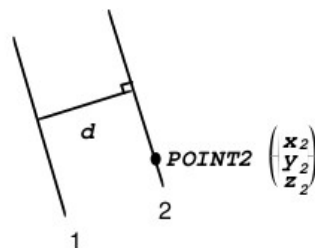
Le conducteur secondaire est considéré infini et parallèle au conducteur principal. On peut définir sa position de deux manières :

/ TRANS : (ux uy uz)



translation amenant du conducteur principal 1 au conducteur secondaire 2

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



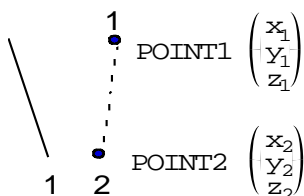
Le conducteur

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 et POINT2.

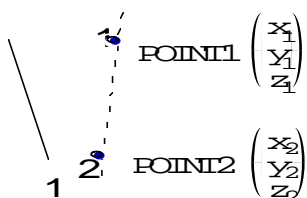
POINT1 = (x1, y1, z1),
POINT2 = (x2, y2, z2),



/ '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.

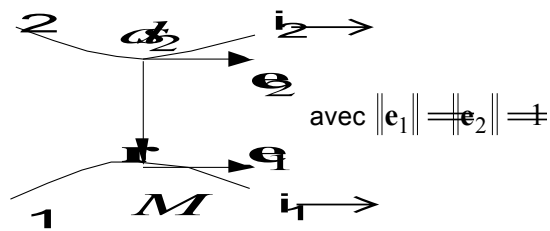
4.39.2 Syntaxe

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

4.39.3 Fonction d'espace

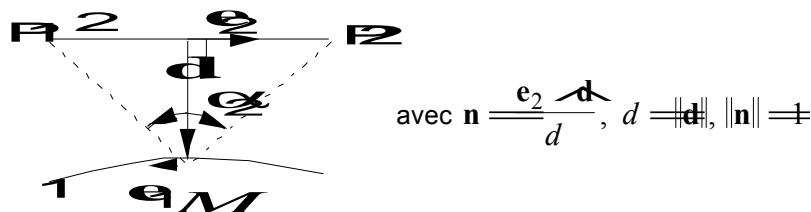
La fonction d'espace composant la densité linéique de forces de LAPLACE exercée en un point M du conducteur 1 (conducteur principal) par les éléments du conducteur 2 (conducteur secondaire) peut s'exprimer :

$$f(M) = \frac{e_1}{2} \wedge \int_2 \frac{e_2 \wedge r}{\|r\|^3} ds_2$$



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

$$f(M) = \sum_i \frac{e_i}{2} \wedge \frac{n}{d} (\sin \alpha_1 - \sin \alpha_2)$$



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 AFFE_CHAR_MECA ; si c'est AFFE_CHAR_MECA_F, elles proviennent d'un concept de type fonction.

4.40.2 Syntaxe

- pour AFFE_CHAR_MECA

IMPE_FACE =_F

MAILLE = lma

(

♦

|

[l_maille]

,

GROUP_MA= lgma,

[l_gr_maille]

IMPE = Q,

[R]

)

- pour AFFE_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,
=      (kx, ky, kz) ,      [l_R]      ◆      DIRECTION
=      H,      ◆      DIST_ORIG
[FONC_SIGNAL]      =      f,      ◆      [R]
[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.

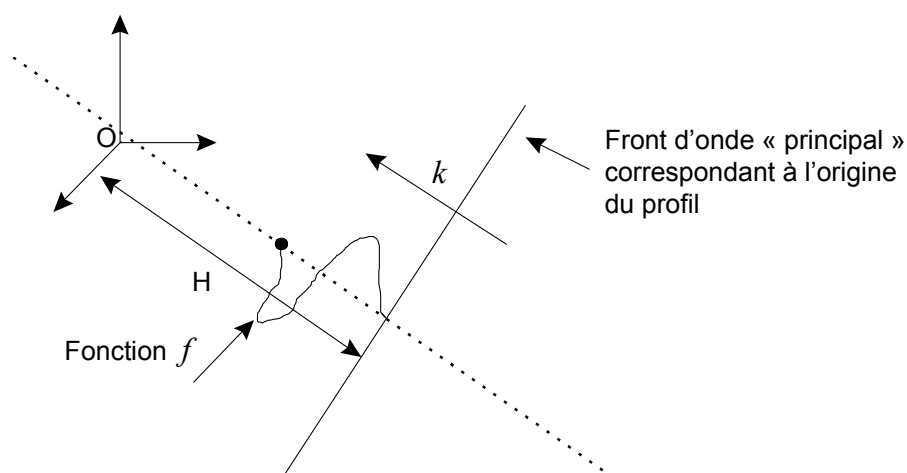
Plus précisément, on caractérise :

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

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 AFFE_CHAR_MECA

ONDE_FLUI =_F

(

◆

|

MAILLE =

lma

,

[l_maille]

GROUP_MA=

lgma,

|

[l_gr_maille]

PRES =

P,

[R]

)

- pour AFFE_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	3D_FLUIDE
QUAD4, QUAD8, QUAD9	
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 AFFE_CHAR_MECA
= 'OUI',
  / | MAILLE = lma ,
  [l_maille]
  | GROUP_MA = lgma,
  [l_gr_maille]
  ♦ | FLUN = phiT,
  , [R]
  | FLUN_HYDR1 = phie,
  [R]
  | FLUN_HYDR2 = phiv
  [R]
)

• pour AFFE_CHAR_MECA_F
= 'OUI',
  / | MAILLE = lma ,
  [l_maille]
  | GROUP_MA = lgma,
  [l_gr_maille]
  ♦ | FLUN = phiTf,
  [fonction]
  | FLUN_HYDR1 = phief,
  [fonction]
  | FLUN_HYDR2 = phivf,
  [fonction]
)

```

4.44.3 Opérandes

| FLUN = phiT,
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 = phie,

Valeur du flux hydraulique associé au constituant eau

| FLUN_HYDR2 = phiv,

Valeur du flux hydraulique associé au constituant air

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

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

avec : ρ_e : masse volumique du liquide

ρ_v : masse volumique de la vapeur

P_e : pression du liquide (PRE1)

P_v : 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 AFFE_CHAR_MECA seulement

```

      ARLEQUIN = _F( ♦ GROUP_MA_1
= gma1,
[gr_maille] ♦ GROUP_MA_2 = gma2,
♦ GROUP_MA_COLL = [gr_maille] gma,
♦ COLLAGE = [gr_maille] /
'GROSSIER', [DEFAULT]
/ 'FIN',
/ 'GROUP_MA_1',
/ 'GROUP_MA_2',
♦
/ POIDS_1 = rho1 ,
[R]
/ POIDS_2 = rho2 ,
[R]
/ POIDS_GROSSIER = rhof ,
[R]
/ POIDS_FIN = rhog ,
[R]
♦
CARA_ELEM = cara ,
[cara_elem]
♦ UNITE_GMSH = / 0
, [DEFAULT]

```

[I] / unit ,
)

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 = rho1,
/ POIDS_2 = rho2,
/ POIDS_FIN = rhof,
/ POIDS_GROSSIER = rhog,

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 $(\alpha_1, \alpha_2) = (1 - \rho_2, \rho_2)$.

◇ CARA_ELEM =

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

◇ UNITE_GMSH = / 0
[DEFAULT]

/ unit , [I]

Numéro d'unité du fichier au format gmsh qui permet l'écriture du maillage de la zone de collage après découpe des éléments.

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)