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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Table des Matières

1 But....1
2 Syntaxe générale....5
3 Généralités....7
3.1 Principes....7
3.2 Hypothèses et limitations....7
3.2.1 Linéarité des relations cinématiques....7
3.2.2 Chargements de Neumann....8
3.3 Disponibilités des chargements suivant le type....8
3.4 Messages d'erreur possibles....9
3.5 Choix des unités....10
3.6 Cas des grandes transformations....10
3.6.1 Problématique....10
3.6.2 Chargements suiveurs....11
3.7 Désignation des entités topologiques d'affectation des chargements....11
3.8 Règles de surcharge et de rémanence....11
3.9 Définition des repères....12
3.9.1 Normales et tangentes aux mailles....12
3.9.2 Cas des éléments de structure....13
3.9.3 Définition d'un repère par les angles nautiques....13
4 Opérandes généraux....15
4.1 Opérande MODELE....15
4.2 Opérande DOUBLE_LAGRANGE....15
4.3 Opérande VERI_NORM....15
4.4 Opérande NUME_LAGR....15
4.5 Opérande INFO....15
4.6 Opérande ANGL_NAUT/CENTRE/TRAN....15
4.7 Opérande LIAISON_EPX....16
5 Chargements de type Dirichlet....17
5.1 Degrés de liberté....17
5.2 Conflits entre les degrés de liberté....19
5.3 Opérations d'appariement....19
5.3.1 Appariement nœud à nœud (maillages compatibles)....19
5.3.2 Appariement maille-à-nœud (maillages incompatibles)....20
5.4 Mot-clé DDL_IMPO....20
5.5 Mot-clé ARETE_IMPO....21
5.6 Mot-clé FACE_IMPO....22
5.7 Mot-clé LIAISON_DDL....23
5.8 Mot-clé LIAISON_OBLIQUE....24
5.9 Mot-clé LIAISON_UNIF....24

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5.10 Mot-clé LIAISON_CHAMNO....25
5.11 Mot-clé CHAMNO_IMPO....26
5.12 Mot-clé LIAISON_GROUP....26
5.13 Mot-clé LIAISON_MAIL....30
5.13.1 Usage et exemples....30
5.13.2 Quelques remarques et précautions d'usage....31
5.13.3 Mots-clefs....32
5.14 Mot-clé LIAISON_PROJ....33
5.14.1 But....33
5.14.2 Syntaxe (AFFE_CHAR_MECA seulement)....34
5.14.3 Opérandes....34
5.15 Mot-clé LIAISON_CYCL....34
5.16 Mot-clé LIAISON_SOLIDE....36
5.17 Mot-clé LIAISON_ELEM....37
5.17.1 Option '3D_POU'....38
5.17.2 Option '3D_POU_ARLEQUIN'....38
5.17.3 Option '2D_POU'....39
5.17.4 Option 'COQ_POU'....39
5.17.5 Option '3D_TUYAU'....39
5.17.6 Option 'COQ_TUYAU'....40
5.17.7 Option 'PLAQ_POUT_ORTH'....41
5.18 Mot-clé LIAISON_RBE3....41
6 Chargements de type Dirichlet pour les éléments de structure....43
6.1 Mot-clé DDL_POUTRE....43
6.2 Mot-clé LIAISON_COQUE....43
7 Chargements de type Neumann....46
7.1 Mot-clé FORCE_NODALE....46
7.2 Mot-clé FORCE_ARETE....46
7.3 Mot-clé FORCE_CONTOUR....47
7.4 Mot-clé FORCE_FACE....47
7.5 Mot-clé FORCE_INTERNE....47
7.6 Mot-clé PRES_REP....48
7.7 Mot-clé EVOL_CHAR....49
7.8 Mot-clé EFFE_FOND....49
7.9 Mot-clé PESANTEUR....50
7.10 Mot-clé ROTATION....50
7.11 Mot-clé PRE_SIGM....51
7.12 Mot-clé PRE_EPSI....51
7.13 Mot-clé FORCE_ELEC....53
7.14 Mot-clé INTE_ELEC....55

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

7.15 Mot-clé VECT_ASSE....56
8 Chargements de type Neumann pour les éléments de structure....57
8.1 Mot-clé FORCE_POUTRE....57
8.2 Mot-clé FORCE_TUYAU....58
8.3 Mot-clé FORCE_COQUE....58
9 Autres chargements....61
9.1 Mot-clé LIAISON_INTERF....61
9.2 Mot-clé RELA_CINE_BP....61
9.3 Mot-clé IMPE_FACE....62
9.4 Mot-clé VITE_FACE....63
9.5 Mot-clé ONDE_PLANE....63
9.6 Mot-clé ONDE_FLUI....64
9.7 Mot-clé FLUX_THM_REP....64
9.8 Mot-clé FORCE_SOL....65

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe générale

```
charg [char_meca] = AFFE_CHAR_MECA(
    ♦ MODELE           = mo                                    [modèle]
    ◊ INFO             = [1, 2]                                [défaut]
    ◊ DOUBLE_LAGRANGE  = /'OUI',                               [défaut]
                         /'NON'
    ◊ VERI_NORM        = /'OUI',                               [défaut]
                         /'NON'
    ◊ LIAISON_EPX      = ' OUI '
    ♦ | PESANTEUR        = voir mot clé PESANTEUR              [§7.9]
      | ROTATION         = voir mot clé ROTATION               [§7.10]
      | DDL_IMPO         = voir mot clé DDL_IMPO               [§5.4]
      | DDL_POUTRE       = voir mot clé DDL_POUTRE             [§6.1]
      | FACE_IMPO        = voir mot clé FACE_IMPO              [§5.6]
      | CHAMNO_IMPO      = voir mot clé CHAMNO_IMPO            [§5.11]
      | ARETE_IMPO       = voir mot clé ARETE_IMPO             [§5.5]
      | LIAISON_DDL      = voir mot clé LIAISON_DDL            [§5.7]
      | LIAISON_OBLIQUE  = voir mot clé LIAISON_OBLIQUE        [§5.8]
      | LIAISON_GROUP    = voir mot clé LIAISON_GROUP          [§5.12]
      | LIAISON_MAIL     = voir mot clé LIAISON_MAIL           [§5.13]
      | LIAISON_PROJ     = voir mot clé LIAISON_PROJ           [§5.14]
      | LIAISON_CYCL     = voir mot clé LIAISON_CYCL           [§5.15]
      | LIAISON_SOLIDE   = voir mot clé LIAISON_SOLIDE         [§5.16]
      | LIAISON_ELEM     = voir mot clé LIAISON_ELEM           [§5.17]
      | LIAISON_UNIF     = voir mot clé LIAISON_UNIF           [§5.9]
      | LIAISON_CHAMNO   = voir mot clé LIAISON_CHAMNO         [§5.10]
      | LIAISON_RBE3     = voir mot clé LIAISON_RBE3           [§5.18]
      | LIAISON_INTERF   = voir mot clé LIAISON_INTERF         [§9.1]
      | VECT_ASSE        = voir mot clé VECT_ASSE              [§ 7.15 ]
      | FORCE_NODALE     = voir mot clé FORCE_NODALE           [§ 7.1 ]
      | FORCE_FACE       = voir mot clé FORCE_FACE             [§ 7.4 ]
      | FORCE_ARETE      = voir mot clé FORCE_ARETE            [§ 7.2 ]
      | FORCE_CONTOUR    = voir mot clé FORCE_CONTOUR          [§ 7.3 ]
      | FORCE_INTERNE    = voir mot clé FORCE_INTERNE          [§ 7.5 ]
      | PRE_SIGM         = voir mot clé PRE_SIGM               [§ 7.11 ]
      | PRES_REP         = voir mot clé PRES_REP               [§ 7.6 ]
      | EFFE_FOND        = voir mot clé EFFE_FOND              [§ 7.8 ]
      | PRE_EPSI         = voir mot clé PRE_EPSI               [§ 7.12 ]
      | FORCE_POUTRE     = voir mot clé FORCE_POUTRE           [§ 8.1 ]
      | FORCE_TUYAU      = voir mot clé FORCE_TUYAU            [§ 8.2 ]
      | FORCE_COQUE      = voir mot clé FORCE_COQUE            [§ 8.3 ]
      | LIAISON_COQUE    = voir mot clé LIAISON_COQUE          [§ 6.2 ]
      | RELA_CINE_BP     = voir mot clé RELA_CINE_BP           [§ 9.2 ]
      | FORCE_ELEC       = voir mot clé FORCE_ELEC             [§ 7.13 ]
      | INTE_ELEC        = voir mot clé INTE_ELEC              [§ 7.14 ]
      | IMPE_FACE        = voir mot clé IMPE_FACE              [§ 9.3 ]
      | VITE_FACE        = voir mot clé VITE_FACE              [§ 9.4 ]
      | ONDE_FLUI        = voir mot clé ONDE_FLUI              [§ 9.6 ]
      | FLUX_THM_REP     = voir mot clé FLUX_THM_REP           [§ 9.7 ]
      | FORCE_SOL        = voir mot clé FORCE_SOL              [§ 9.8 ]
)

charg [char_meca] = AFFE_CHAR_MECA_C(
    ♦ MODELE           = mo                                    [modèle]
    ◊ INFO             = [1, 2]                                [défaut]
    ♦ | DDL_IMPO         = voir mot clé DDL_IMPO               [§5.4]
      | LIAISON_DDL      = voir mot clé LIAISON_DDL            [§5.7]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
      | FORCE_POUTRE     = voir mot clé FORCE_POUTRE                [§ 8.1 ]
      )
charg [char_meca] = AFFE_CHAR_MECA_F(
    ♦ MODELE           = mo                                       [modèle]
    ◊ INFO             = [1, 2]                                   [défaut]
    ◊ VERI_NORM        = /'OUI',                                  [défaut]
                         / 'NON'
    ♦ | DDL_IMPO         = voir mot clé DDL_IMPO                  [§5.4]
      | FACE_IMPO        = voir mot clé FACE_IMPO                 [§5.6]
      | LIAISON_DDL      = voir mot clé LIAISON_DDL               [§5.7]
      | LIAISON_OBLIQUE  = voir mot clé LIAISON_OBLIQUE           [§5.8]
      | LIAISON_GROUP    = voir mot clé LIAISON_GROUP             [§5.12]
      | LIAISON_UNIF     = voir mot clé LIAISON_UNIF              [§5.9]
      | FORCE_NODALE     = voir mot clé FORCE_NODALE              [§ 7.1 ]
      | FORCE_FACE       = voir mot clé FORCE_FACE                [§ 7.4 ]
      | FORCE_ARETE      = voir mot clé FORCE_ARETE               [§ 7.2 ]
      | FORCE_CONTOUR    = voir mot clé FORCE_CONTOUR             [§ 7.3 ]
      | FORCE_INTERNE    = voir mot clé FORCE_INTERNE             [§ 7.5 ]
      | PRES_REP         = voir mot clé PRES_REP                  [§ 7.6 ]
      | EFFE_FOND        = voir mot clé EFFE_FOND                 [§ 7.8 ]
      | PRE_EPSI         = voir mot clé PRE_EPSI                  [§ 7.12 ]
      | FORCE_POUTRE     = voir mot clé FORCE_POUTRE              [§ 8.1 ]
      | FORCE_TUYAU      = voir mot clé FORCE_TUYAU               [§ 8.2 ]
      | FORCE_COQUE      = voir mot clé FORCE_COQUE               [§ 8.3 ]
      | LIAISON_COQUE    = voir mot clé LIAISON_COQUE             [§ 6.2 ]
      | IMPE_FACE        = voir mot clé IMPE_FACE                 [§ 9.3 ]
      | VITE_FACE        = voir mot clé VITE_FACE                 [§ 9.4 ]
      | ONDE_ PLANE       = voir mot clé ONDE_ PLANE              [§ 9.5 ]
      | FLUX_THM_REP     = voir mot clé FLUX_THM_REP              [§ 9.7 ]
)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Généralités

3.1 Principes

Il y a trois grandes catégories d'opérande :

- Les opérandes appliquant des conditions cinématiques ou chargements de Dirichlet, c'est-à-dire des relations entre les degrés de liberté. Dans AFFE_CHAR_MECA, ces conditions sont appliquées par dualisation (méthode des doubles Lagrange, voir [R2.03.01]) ;
- Les opérandes appliquant des chargements de type « forces » ou chargements de Neumann, appliquées sous forme faible, ce qui implique l'utilisation d'un schéma d'intégration numérique. Certains chargements impliquent la présence d'éléments de bord dans le modèle ;
- Les opérandes appliquant des chargements spéciaux, y compris de type mixte Dirichlet/Neumann.

La plupart des opérandes sont construits sur le même principe :

- Spécification du lieu d'application des conditions limites par les mots-clefs standards GROUP_NO, GROUP_MA et parfois SANS_GROUP_NO et SANS_GROUP_MA.
- Spécification des composantes affectées, qui se répartissent en trois groupes :
  - Composantes standards de la grandeur considérée. Il s'agit de la grandeur DEPL_R (ou DEPL_C ou DEPL_F), représentant les degrés de liberté du problème de mécanique (voir § 5.1) ;
  - Composantes combinées DNOR et DTAN, qui construisent une combinaison entre les composantes de la grandeur DEPL_R sur des considérations relatives aux tangentes et à la normale ;
  - Composantes en efforts, moments ou pression utilisant soit la grandeur FORC_R (ou FORC_C ou FORC_F), la grandeur PRES_R (ou PRES_C ou PRES_F) ;
- Les composantes affectées doivent être du bon type selon l'opérateur utilisé :
  - Du type réel pour l'opérateur AFFE_CHAR_MECA ;
  - Du type complexe pour l'opérateur AFFE_CHAR_MECA_ C ;
  - Du type fonction (créé notamment par l'un des opérateurs DEFI_FONCTION, DEFI_NAPPE ou DEFI_CONSTANTE) pour l'opérateur AFFE_CHAR_MECA_F. Ceci est vrai à une exception près : l'argument de COEF_MULT pour le mot clé facteur LIAISON_DDL dans AFFE_CHAR_MECA_F est obligatoirement de type réel.

3.2 Hypothèses et limitations

En plus de la définition, des hypothèses et limitations propres à chaque chargement, il existe des hypothèses générales que l'on va rappeler ici.

3.2.1 Linéarité des relations cinématiques

On rappelle qu'une relation cinématique permet d'écrire une équation du type :

$$\sum_{i=1}^{r} \alpha_i U_i = \beta \qquad (1)$$

Avec U_i la liste des r degrés de liberté, α_i les coefficients et β le second membre.

Il convient de noter que les relations cinématiques de *Code_Aster* sont des relations ***linéaires***, c'est-à-dire :

- Elles ne peuvent dépendre *a priori* de la déformation ou du mouvement de la structure : elles restent valables uniquement dans l'hypothèse des petites perturbations, sauf mention contraire (voir LIAISON_SOLIDE).
- Les coefficients α_i de la relation linéaire ne peuvent pas être des fonctions du temps, car la matrice B des conditions de Dirichlet est constante pendant tout le transitoire. Par contre, ils peuvent être des fonctions de la géométrie ***initiale*** ;
- Le second membre β peut être une fonction du temps ou de la géométrie ***initiale*** ;

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.2.2 Chargements de Neumann

Contrairement aux conditions cinématiques, il est tout à fait possible que *certains* chargements de Neumann soient non-linéaires, et, en particulier, dépendent de la déformation de la structure. De tels chargements sont communément appelés chargements ***suiveurs***. Néanmoins, dans ce cas, le problème devenant non-linéaire, il est nécessaire d'utiliser un opérateur de calcul adéquat comme STAT_NON_LINE et DYNA_NON_LINE et de préciser que ces chargements sont effectivement considérés comme suiveurs (voir [U4.51.03]).

La plupart des chargements de Neumann (sauf FORCE_NODALE) sont appliqués sous forme faible, c'est-à-dire qu'on utilise une formule de quadrature numérique. De plus, on ne peut appliquer simultanément un chargement de Neumann et un chargement de Dirichlet sur le même nœud et dans la même direction. De ce fait, il peut exister une différence entre la solution théorique et la solution éléments finis.

Par exemple, sur une structure insuffisamment maillée, il est possible de constater un écart entre la somme des efforts nodaux correspondant au chargement de pesanteur et la valeur du poids réel, l'écart correspondant *grosso modo* au nombre de nœuds encastrés de la structure.

Un raffinement du maillage permet de minimiser cette différence. On peut également 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.
Une autre solution est de dédoubler les nœuds sur lesquels la condition cinématique est imposée et de faire par exemple un LIAISON_DDL entre les deux nœuds ou d'utiliser des éléments discrets.

3.3 Disponibilités des chargements suivant le type

Les chargements disponibles ne sont pas forcément applicables dans les trois opérateurs AFFE_CHAR_MECA. Voici la liste des disponibilités suivant le type de l'opérateur :

Mot-clef	AFFE_CHAR_MECA	AFFE_CHAR_MECA_C	AFFE_CHAR_MECA_F
ARETE_IMPO	OUI	NON	NON
CHAMNO_IMPO	OUI	NON	NON
DDL_IMPO	OUI	OUI	OUI
EFFE_FOND	OUI	NON	OUI
EVOL_CHAR	OUI	NON	NON
FACE_IMPO	OUI	NON	OUI
FLUX_THM_REP	OUI	NON	OUI
FORCE_ARETE	OUI	NON	OUI
FORCE_CONTOUR	OUI	NON	OUI
FORCE_COQUE	OUI	NON	OUI
FORCE_ELEC	OUI	NON	NON
FORCE_FACE	OUI	NON	OUI
FORCE_INTERNE	OUI	NON	OUI
FORCE_NODALE	OUI	NON	OUI
FORCE_POUTRE	OUI	OUI	OUI
FORCE_SOL	OUI	NON	NON
FORCE_TUYAU	OUI	NON	OUI
IMPE_FACE	OUI	NON	OUI

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

INTE_ELEC	OUI	NON	NON
LIAISON_CHAMNO	OUI	NON	NON
LIAISON_ CYCL	OUI	NON	NON
LIAISON_DDL	OUI	OUI	OUI
LIAISON_ ELEM	OUI	NON	NON
LIAISON_GROUP	OUI	NON	OUI
LIAISON_INTERF	OUI	NON	NON
LIAISON_MAIL	OUI	NON	NON
LIAISON_OBLIQUE	OUI	NON	OUI
LIAISON_RBE3	OUI	NON	NON
LIAISON_SOLIDE	OUI	NON	NON
LIAISON_UNIF	OUI	NON	OUI
ONDE_FLUI	OUI	NON	NON
ONDE_PLANE	NON	NON	OUI
PESANTEUR	OUI	NON	NON
PRE_EPSI	OUI	NON	OUI
PRE_SIGM	OUI	NON	NON
PRES_REP	OUI	NON	OUI
RELA_CINE_BP	OUI	NON	NON
ROTATION	OUI	NON	NON
VECT_ASSE	OUI	NON	NON
VITE_FACE	OUI	NON	OUI

3.4 Messages d'erreur possibles

Il arrive parfois qu'une commande de calcul mécanique 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 établit une correspondance entre des mots-clefs facteurs et les noms d'option de calcul qu'ils activent :

Mot-clef	Nom de l'option
EVOL_CHAR	CHAR_MECA_PRES_R CHAR_MECA_FR3D3D CHAR_MECA_FR2D2D CHAR_MECA_FR2D3D CHAR_MECA_FR1D2D
PESANTEUR	CHAR_MECA_PESA_R
ROTATION	CHAR_MECA_ROTA_R
PRE_SIGM	FORC_NODA
FORCE_NODALE	CHAR_MECA_FORC_R CHAR_MECA_FORC_F
FORCE_ARETE	CHAR_MECA_FR1D3D

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

	CHAR_MECA_FF1D3D
FORCE_CONTOUR	CHAR_MECA_FR1D2D CHAR_MECA_FF1D2D
FORCE_FACE	CHAR_MECA_FR2D3D CHAR_MECA_FF2D3D
FORCE_INTERNE	CHAR_MECA_FR2D2D CHAR_MECA_FR3D3D CHAR_MECA_FF2D2D CHAR_MECA_FF3D3D
PRES_REP	CHAR_MECA_PRES_R CHAR_MECA_PRES_F
EFFE_FOND	CHAR_MECA_EFON_R CHAR_MECA_EFON_F
PRE_EPSI	CHAR_MECA_EPSI_R CHAR_MECA_EPSI_F
FORCE_ELEC	CHAR_MECA_FRELEC
INTE_ELEC	CHAR_MECA_FRLAPL
FORCE_POUTRE	CHAR_MECA_FR1D1D CHAR_MECA_FC1D1D CHAR_MECA_FF1D1D
FORCE_TUYAU	CHAR_MECA_PRES_R CHAR_MECA_PRES_F
FORCE_COQUE	CHAR_MECA_FRCO2D CHAR_MECA_FRCO3D CHAR_MECA_FFCO2D CHAR_MECA_FFCO3D
FLUX_THM_REP	CHAR_MECA_FLUX_R CHAR_MECA_FLUX_F

3.5 Choix des unités

Pour les chargements de Neumann, les forces sont à fournir par unité de maillage pour les efforts linéiques, par unité de maillage au carré pour les efforts surfaciques et par unité de maillage au cube pour les efforts volumiques), en cohérence avec la définition des propriétés matériaux (module d'Young par exemple). Dans le cas axisymétrique, les forces à fournir sont ramenées à un secteur de 1 radian (diviser le chargement réel par 2π).

3.6 Cas des grandes transformations

3.6.1 Problématique

Lorsqu'on applique un chargement de Dirichlet sur une structure en grandes transformations (grands déplacements, grandes rotations), il faut prendre garde à la bonne prise en compte de celui-ci. C'est à dire qu'il faut veiller à ce que ce chargement soit applicable lorsque l'hypothèse des petites perturbations n'est plus vérifiée.

C'est naturellement le cas des chargements suivants :

- DDL_IMPO mais si on utilise AFFE_CHAR_MECA_F, il faut le faire avec des fonctions qui représentent le déplacement imposé **réel** (curviligne en général). Si on ne le fait pas, l'interpolation en temps nous fait passer par des états intermédiaires « faux » ;
- LIAISON_MAIL + TYPE_RACCORD='MASSIF' ;

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- LIAISON_UNIF ;
- LIAISON_OBLIQUE ;
- LIAISON_DDL
- LIAISON_CHAM_NO
- CHAMNO_IMPO
- LIAISON_SOLIDE à condition d'appliquer l'hypothèse d'un chargement suiveur (voir §3.6).

Par contre, pour les chargements suivants, l'hypothèse des grandes transformations n'est pas applicable et provoquera des résultats faux :

- FACE_IMPO avec DNOR=f(t) : la « normale » change en grandes rotations ;
- LIAISON_MAIL + TYPE_RACCORD='COQUE_MASSIF' et 'MASSIF_COQUE' ;
- LIAISON_ELEM ;
- LIAISON_RBE3.

3.6.2 Chargements suiveurs

Pour les opérateurs non-linéaires (STAT_NON_LINE et DYNA_ NON_LINE), certains chargements peuvent être « suiveurs », c'est-à-dire que leur application dépend du déplacement et donc change à chaque itération de Newton. Il est alors nécessaire que l'utilisateur le précise par l'opérande TYPE_CHARGE dans le mot-clef facteur EXCIT de ces commandes (voir [U4.51.03]). Le fait de préciser que le chargement est suiveur ajoute parfois une contribution dans la matrice de rigidité (voir par exemple [R3.03.04]) et peut la rendre non-symétrique. Toutefois, pour le chargement EVOL_CHAR (voir §7.7), il n'est pas nécessaire de préciser que les chargements sont suiveurs, ils le sont automatiquement par défaut. Le fait de le préciser va simplement activer la contribution matricielle supplémentaire et donc agir sur la vitesse de convergence (et non sur la précision du résultat).

3.7 Désignation des entités topologiques d'affectation des chargements

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

- Sur un seul nœud : on utilise l'opérande GROUP_NO qui ne doit évidemment contenir qu'un seul nœud ;
- Sur une liste de nœuds : on peut utiliser l'opérande GROUP_NO mais aussi l'opérande GROUP_MA ou TOUT='OUI' pour affecter sur tout le maillage ;
- Sur une seule maille : on utilise l'opérande GROUP_MA qui ne doit évidemment contenir qu'une seule maille ;
- Sur une liste de mailles : on peut utiliser l'opérande GROUP_MA ou TOUT='OUI' pour affecter sur tout le maillage ;

Certains mots-clefs ont besoin de définir plusieurs entités topologiques (des groupes de nœuds en vis-à-vis par exemple), dans ce cas, les noms peuvent varier légèrement (GROUP_NO_2, GROUP_MA_ESCL, etc.). Il est possible dans la plupart des mots clefs d'exclure des nœuds ou des mailles à l'aide d'opérateur de type SANS_*. Cette fonctionnalité évite de redéfinir des groupes dans votre maillage ou dans la commande DEFI_GROUP.

3.8 Règles de surcharge et de rémanence

Pour définir le domaine d'affectation le plus simplement possible, on utilise la règle de surcharge définie dans le document [U1.03.00] dont on rappelle les principes :

- Les affectations se font en superposant les effets des différents chargements ;
- En cas de conflit, le dernier chargement l'emporte sur les précédents ;

Si par exemple, l'utilisateur fait :

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

Et si la normale pour $G1$ est orientée selon X, alors tout se passera comme si on avait fait :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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

La règle de surcharge précédente doit être complétée par une autre règle pour préciser ce qui se passe lorsqu'on peut affecter plusieurs quantités pour chaque occurrence d'un chargement. Soit par exemple :

```
FORCE_INTERNE=(
    _F(TOUT      = 'OUI',    FX = 1.              ),
    _F(GROUP_MA = 'GM1',              FY = 2.    ),
)
```

La règle de surcharge nous dit que la deuxième occurrence de FORCE_INTERNE surcharge la première. Mais que vaut FX sur une maille appartenant à GM1 ? A-t-il été effacé par la deuxième occurrence ? Si la seule règle de surcharge est appliquée, FX n'est pas défini sur GM1.
On utilise donc une deuxième règle dite de rémanence qui précise que lors de l'application de la règle de surcharge sur des occurrences, on conserve les composantes qui ne sont pas surchargées.
En appliquant la règle de rémanence sur l'exemple, FX conserve la valeur affectée au préalable. Tous les éléments du modèle ont donc une valeur pour FX et les éléments de GM1 ont une valeur à la fois pour FX et FY .

3.9 Définition des repères

La plupart des chargements sont définis dans le repère **global** du maillage, sauf :

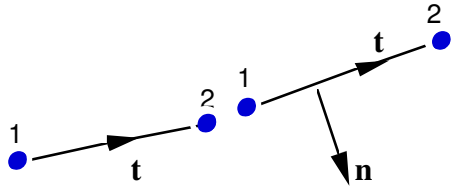
- Pour les éléments de structure (§3.9.2) ;
- Pour les mots-clefs DNOR, DTAN et les chargements de type pression. Dans ce cas, il est nécessaire de définir normales et tangentes (voir §3.9.1), voire éventuellement d'utiliser la commande MODI_MAILLAGE ;
- Quand le mot-clef ANGL_NAUT est utilisable (voir §3.9.3) ;

Dans les autres cas, il est généralement possible de définir des fonctions de l'espace (sous réserve de rester dans le cadre des hypothèses du §3.2).

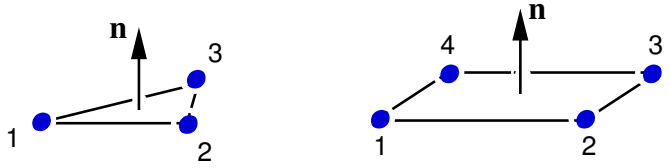
3.9.1 Normales et tangentes aux mailles

On donne ici la définition standard des normales et des tangentes suivant le type de maille de bord :

- Pour les éléments segments en 2D, la tangente est celle définie par le segment orienté par ses deux premiers nœuds, la normale n est alors telle que (n,t) forment un repère direct

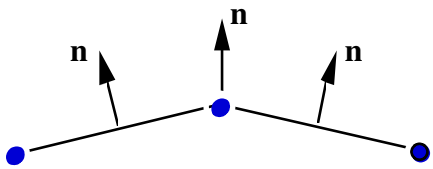


- Pour les éléments triangles ou quadrangles en 3D, l'orientation de la normale n est celle correspondant au sens direct de la description de la maille.



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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)



L'opérateur MODI_MAILLAGE permet de s'assurer de la continuité de l'orientation de la normale aux bords des éléments massifs de milieu continu.

3.9.2 Cas des éléments de structure

Les éléments de structure (poutres, plaques et coques) ont leur propre repère **local** dont la définition est donnée dans la documentation de la commande AFFE_CARA_ELEM [U4.42.01].

3.9.3 Définition d'un repère par les angles nautiques

Certains chargements offrent la possibilité de donner leur direction d'application en utilisant des angles nautiques dont on rappelle ici la définition.

Les angles nautiques α, β, γ fournis en degrés, sont les angles permettant de passer du repère global de définition des coordonnées des nœuds (P, X, Y, Z) au repère local (P, X_3, Y_3, Z_3). Celui-ci est obtenu par trois rotations :

- Une rotation d'angle α autour de Z, transformant $(X\,Y\,Z)$ en $(X_1\,Y_1\,Z_1)$ avec $Z_1 \equiv Z$ [Figure 3.9.3-1] ;
- Une rotation d'angle β autour de Y_1, transformant $(X_1\,Y_1\,Z_1)$ en $(X_2\,Y_2\,Z_2)$ avec $Y_2 \equiv Y_1$ [Figure 3.9.3-2] ;
- Une rotation d'angle γ autour de X_2, transformant $(X_2\,Y_2\,Z_2)$ en $(X_3\,Y_3\,Z_3)$ avec $X_3 \equiv X_2$ [Figure 3.9.3-3] ;

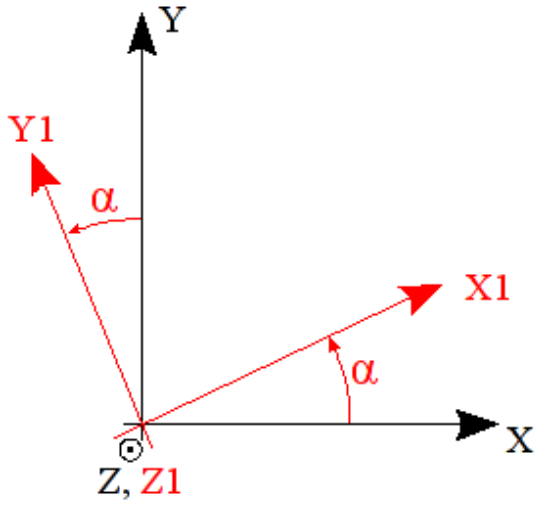


Figure 3.9.3-1 : angle α.

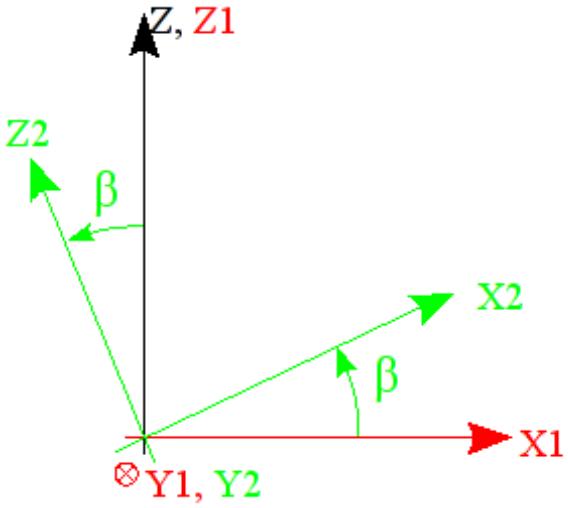


Figure 3.9.3-2 : angle β.

Remarque : pour la figure 3.9.3-2, l'angle de rotation β est négatif.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

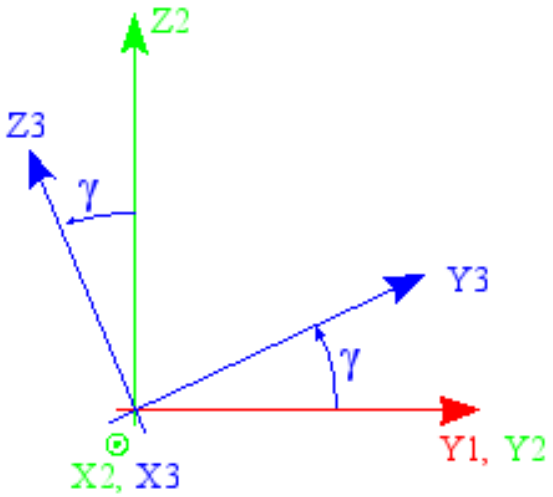


Figure 3.9.3-3 : angle γ .

Le repère local est : $(X_3 Y_3 Z_3)$

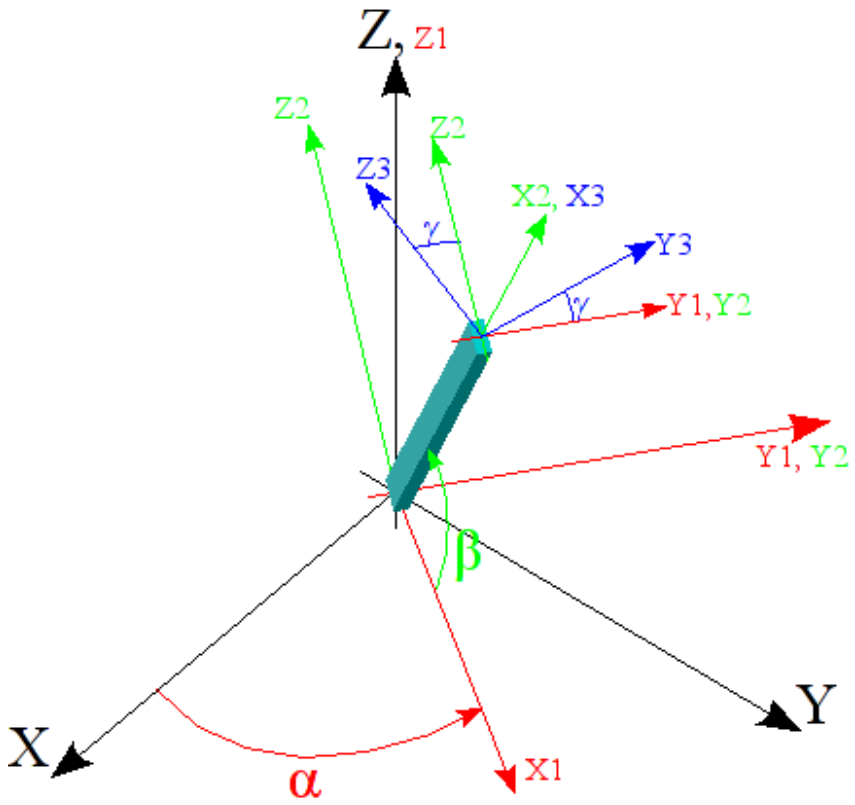


Figure 3.9.3-4 : Représentation des repères global et local.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

4 Opérandes généraux

4.1 Opérande MODELE

♦ MODELE
Le mot-clef MODELE attend un concept produit par l'opérateur AFFE_MODELE où sont définis les types d'éléments finis affectés sur le maillage. Le modèle est nécessairement du type MECANIQUE.

4.2 Opérande DOUBLE_LAGRANGE

◊ DOUBLE_LAGRANGE = 'OUI'/'NON'
Ce mot-clé permet de dire si l'utilisateur souhaite ou non dédoubler les multiplicateurs de Lagrange utilisés pour définir dualiser les conditions aux limites dans la matrice assemblée.
Concrètement, dédoubler les multiplicateurs de Lagrange permet d'utiliser des solveurs linéaires ne permettant pas le pivotage. Ne pas dédoubler les Lagrange permet de réduire le nombre de degré de liberté du problème (et donc la taille du problème à résoudre) mais son usage est limité aux solveurs MUMPS et Petsc.

4.3 Opérande VERI_NORM

◊ VERI_NORM = 'OUI'/'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). 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_MAILLAGE [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_MAILLAGE mot-clé ORIE_NORM_COQUE.

4.4 Opérande NUME_LAGR

◊ NUME_LAGR = 'NORMAL'/'APRES'
Ce mot-clef est utile dans l'imposition de conditions cinématiques complexes. Si NUME_LAGR ='NORMAL', alors 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 NUME_LAGR ='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.

4.5 Opérande INFO

◊ INFO
Niveau des impressions sur le fichier message.

4.6 Opérande ANGL_NAUT/CENTRE/TRAN

◊ ANGL_NAUT = (a,b,c)
Permet de définir un repère par les angles nautiques en degrés (voir § 3.9.3).

◊ CENTRE = (cx,cy,cz)
Coordonnées du centre de rotation (dans le repère global).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

◊ TRAN = (tx,ty,tz)
Composantes du vecteur translation (dans le repère global).

4.7 Opérande LIAISON_EPX

◊ LIAISON_EPX = / 'OUI'
La présence de ce mot-clé entraîne la création d'une table contenant la liste des relations linéaires produites par l'opérateur. Cette table est relue par CALC_EUROPLEXUS (si le chargement lui est fourni). Seul le mot-clé LIAISON_MAIL est compatible avec LIAISON_EPX.

Remarque :

il y a d'autres chargements autorisés dans CALC_EUROPLEXUS mais ceux-ci ont un équivalent direct parmi les fonctionnalités de EUROPLEXUS. Pour les liaisons n'ayant pas d'équivalent, cette fonctionnalité permet de transmettre directement à EUROPLEXUS la liste des relations linéaires créées par code_aster.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5 Chargements de type Dirichlet

5.1 Degrés de liberté

Les chargements de Dirichlet sont imposés sur les degrés de liberté de la grandeur DEPL_R (ou DEPL_C ou DEPL_F) , représentant les degrés de liberté du problème de mécanique (ou de thermo-hydro-mécanique ou d'hydraulique). Nous rappelons ici la signification de ces différents degrés de liberté :

Nom	Modélisation	Description
DX DY DZ	Toutes sauf 2D_FLUI_PESA	Composantes de déplacement en **translation** dans le repère global
DZ	2D_FLUI_PESA	Déplacement imposé de la surface libre
DRX DRY DRZ	Éléments discrets, de poutre, de coque ou de plaque	Composantes de déplacement en **rotation** dans le repère global
GRX	POU_D_TG	Valeur du gauchissement de la poutre
PRES	3D_FLUIDE	Pression acoustique dans le fluide
PRES	3D_JOINT_CT	Pression du fluide interstitiel
PRES	Formulation second gradient	Multiplicateur de Lagrange introduit pour la formulation mixte
PHI	3D_FLUIDE FLUI_STRU 2D_FLUI_PESA	Potentiel des déplacements du fluide
TEMP	THM THHM THH	Température
PRE1	THM THHM THH HM HHM	Pression capillaire ou pression du liquide ou du gaz
PRE2	THM THHM THH	Pression du gaz
UIx	Tuyaux	Gauchissement « in plane » du mode x [1]
VIx WIx	Tuyaux	Ovalisations « in plane » du mode x
UOx	Tuyaux	Gauchissement « out of plane » du mode x
VOx WOx	Tuyaux	Ovalisations « out of plane » du mode x
W0 WI1 WO1	Tuyaux	Degrés de liberté de gonflement et mode 1 sur l'ovalisation
GONF	Formulation incompressible _INCO_UPG Formulation second gradient	Gonflement

1 x vaut 2 et 3 pour les TUYAU_3M et TUYAU_6M, 4,5 et 6 uniquement pour les TUYAU_6M

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

V11 V12 V13 V21 V22 V23 V31 V32 V33	Formulation second gradient	Composantes du tenseur de déformation microscopique
PRES11 PRES12 PRES13 PRES21 PRES22 PRES23 PRES31 PRES32 PRES33	Formulation second gradient	Multiplicateurs de Lagrange introduits pour la formulation mixte
LAGS_C	Contact continue ou XFEM	Pression de contact
LAGS_F1 LAGS_F2	Contact continue ou XFEM	Densité surfacique d'effort de frottement (vectorielle)
H1X H1Y H1Z H2X H2Y H2Z H3X H3Y H3Z H4X H4Y H4Z	XFEM	Degrés de liberté enrichis d'Heaviside
H1PRE1 H2PRE1 H3PRE1	HM-XFEM	Degrés de liberté enrichis d'Heaviside pour la pression capillaire ou la pression du liquide ou du gaz.
K1 K2 K3	XFEM	Degrés de liberté enrichis vectoriels crack-tip
LAG2_C LAG3_C LAG4_C	XFEM (multi-fissuration)	Pressions de contact
LAG2_F1 LAG2_F2 LAG3_F1 LAG3_F2 LAG4_F1 LAG4_F2	XFEM (multi-fissuration)	Densités surfaciques d'effort de frottement (vectorielles)
PRE_FLU PR2_FLU PR3_FLU	HM-XFEM	Pression de fluide dans la fissure
LAG_FLI LAG_FLS LA2_FLI	HM-XFEM	Flux

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

LA2_FLS LA3_FLI LA3_FLS		
LH1	Éléments joints de type '_JHMS'	Multiplicateur de Lagrange hydraulique
GLIS	Éléments CABLE_GAINE	Déplacement relatif suivant la tangente (repère local)

5.2 Conflits entre les degrés de liberté

Au sein du même AFFE_CHAR_MECA, y compris entre les différentes occurrences, on vérifie qu'il n'y a pas redondance entre les conditions limites par application des règles de surcharge et de rémanence, (voir §3.8). En effet, une telle situation conduirait à une matrice singulière lors du calcul.
Cependant, si la même condition aux limites est spécifiée deux fois par deux appels ***différents*** à AFFE_CHAR_MECA (par exemple, avec deux valeurs de déplacement imposé), cela conduit à une matrice singulière. Les conditions limites cinématiques sont toujours imposées sur les ***nœuds***, pas sur les mailles.

5.3 Opérations d'appariement

5.3.1 Appariement nœud à nœud (maillages compatibles)

Ce type d'appariement (utilisé par exemple dans LIAISON_GROUP ou LIAISON_COQUE) permet d'établir des couples de nœuds deux-à-deux. Elle se fait de la même façon que dans AFFE_CHAR_THER. Dans un premier temps, on établit les deux listes de nœuds Γ_1 et Γ_2 à mettre en vis-à-vis (ie à apparier), pour chaque occurrence du mot-clé facteur. 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 $N3$), le message d'erreur suivant est émis :

Il y a un conflit dans les vis-à-vis des nœuds. *Le nœud $N1$ est à la fois le vis-à-vis du nœud $N2$ et du nœud $N3$.*

- 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 $N3$), le message d'erreur suivant est émis :

Il y a un conflit dans les vis-à-vis des nœuds. *Le nœud $N1$ est à la fois le vis-à-vis du nœud $N2$ et du nœud $N3$.*

- 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 :

Il y a un conflit dans les vis-à-vis des nœuds. *Le nœud $N1$ n'est l'image d'aucun nœud 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).
Dans les couples de nœuds en vis-à-vis, l'ordre des nœuds est important. Si pour la première occurrence du mot-clef facteur, un nœud N appartenait au premier groupe de nœuds et un nœud M au deuxième groupe de nœud, et que pour la seconde occurrence du même mot-clef facteur, c'est l'inverse, on obtiendra à l'issue de l'appariement les couples (N, M) et (M, N).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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.

5.3.2 Appariement maille-à-nœud (maillages incompatibles)

Dans la suite de ce paragraphe, on parlera de la face « esclave » (FACE2) et de la face « maître » (FACE1). Le « recollement » de deux faces se fera par écriture de relations linéaires entre les degrés de liberté des deux 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 (voir opérateur PROJ_CHAMP pour plus de détails).
Le principe de la liaison est d'éliminer les degrés de liberté esclaves en les écrivant comme des relations linéaires des degrés de liberté maîtres. 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 maître, il doit donner des mailles. De plus, les mailles maîtres 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 maîtres 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 lorsque l'on soude deux tôles, il vaut mieux multiplier les points de soudure.

Remarque s :

*- En 3D, il ne faut pas donner des mailles maîtres 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.*

*- Lorsque l'on recolle un maillage formé d'éléments linéaires $(P1)$ sur un autre maillage quadratique $(P2)$, il est plutôt conseillé de choisir comme face « esclave » **la face quadratique** .*

5.4 Mot-clé DDL_IMPO

```
DDL_IMPO=_F  (    ♦ /TOUT          =  'OUI',
                    /GROUP_NO      =  lgno,          [l_gr_noeud]
                    /GROUP_MA      =  lgma,          [l_gr_maille]
                  ◊ SANS_GROUP_MA  =  lgma1,         [l_gr_maille]
                  ◊ SANS_GROUP_NO  =  lgno1,         [l_gr_noeud]
                  ♦ /|DX           =  ux ,           [R] ou [C] ou [fonction]
.....................................................
                    /LIAISON       =  'ENCASTRE'
      )
```

Le mot-clé DDL_IMPO est utilisable pour imposer à des nœuds une ou plusieurs valeurs de degré de liberté.

♦ Affectation topologique : TOUT , GROUP_ MA , GROUP_ NO , SANS_ GROUP_ MA , SANS_ GROUP_ NO

Les conditions cinématiques sont imposées sur les nœuds donnés par les mots-clefs TOUT , GROUP_ MA , GROUP_ NO tout en excluant éventuellement grâce aux mots-clefs SANS_* .

♦ Composantes :

- Pour AFFE_CHAR_MECA : DX, DY, DZ, DRX, DRY, DRZ, GRX, PRES, PHI, TEMP, PRE1, PRE2, UI2, UI3, UI4, UI5, UI6, UO2, UO3, UO4, UO5, UO6, VI2, VI3, VI4, VI5, VI6, VO2, VO3, VO4, VO5, VO6, WI2, WI3, WI4, WI5, WI6, WO2, WO3, WO4, WO5, WO6, WO, WI1, WO1, GONF, H1X, H1Y, H1Z, H1PRE1, H2X, H2Y, H2Z, H2PRE1, H3X, H3Y, H3Z, H3PRE1, H4X, H4Y, H4Z, K1, K2, K3, PRE_FLU, LAG_FLI,

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

LAG_FLS, PR2_FLU, LA2_FLI, LA2_FLS, PR3_FLU, LA3_FLI, LA3_FLS, LAGS_C, LAGS_F1, LAGS_F2, LAG2_C, LAG2_F1, LAG2_F2, LAG3_C, LAG3_F1, LAG3_F2, LAG4_C, LAG4_F1, LAG4_F2, V11, V12, V13, V21, V22, V23, V31, V32, V33, PRES11, PRES12, PRES13, PRES21, PRES22, PRES23, PRES31, PRES32, PRES33, LH1 et GLIS.

- Pour AFFE_CHAR_MECA_C : DX, DY, DZ, DRX, DRY, DRZ, GRX, PRES, PHI et GLIS.
- DX , DY , DZ , DRX , DRY , DRZ , GRX , PRES , PHI , TEMP , PRE1 , PRE2 , GONF , H1X , H1Y , H1Z , H1PRE1 , K1 , K2 , K3 , LAGS_C et GLIS .

La signification de tous ces degrés de liberté est précisée dans le § 5.1 .
Remarques :

- 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). Si le nœud demandé est sur les lèvres, alors on impose la condition de blocage sur les nœuds des lèvres supérieure et inférieure.
- Le degré de liberté LH1 (multiplicateur de lagrange hydraulique pour les éléments joints de type '_JHMS') permet de neutraliser les degrés de liberté au bord du joint dans le cas où le massif d'appuis est purement mécanique.
- Les degrés de liberté imposés sont définis dans le repère ***global*** de définition du maillage

♦ 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.

5.5 Mot-clé ARETE_IMPO

```
ARETE_IMPO    =_F (    ♦ /TOUT          =   'OUI',
                         /GROUP_MA      =   lgma,          [l_gr_maille]
                       ◊ SANS_GROUP_MA  =   lgma1,         [l_gr_maille]
                       ◊ SANS_GROUP_NO  =   lgno1,         [l_gr_noeud]
                       ♦ /| DX          =   ux,            [R]
                         | DY          =   uy,            [R]
                         |  DZ         =   uz,            [R]
                         | PRES        =   p,             [R]
                         | PHI         =   phi,           [R]
                         | TEMP        =   T,             [R]
                         | PRE1        =   pr1 ,          [R]
                         | PRE2        =   pr2 ,          [R]
                        /| DTAN        =   ut,            [R]
                  )
```

Le mot-clef ARETE_IMPO permet imposer à tous les nœuds d'une arête sur des éléments volumiques, une ou plusieurs valeurs de degré de liberté.

♦ Affectation topologique : TOUT , GROUP_ MA , SANS_ GROUP_ MA , SANS_ GROUP_ NO
Les conditions cinématiques sont imposées sur les nœuds appartenant aux mailles données par les mots-clefs TOUT , GROUP_ MA tout en excluant éventuellement grâce aux mots-clefs SANS_* .
Les mailles sont nécessairement des **segments** .

♦ Composantes : DX , DY , DZ , GRX , PRE 1 , PRE 2 , PRES , PHI , TEMP
La signification de ces degrés de liberté est précisée dans le § 5.1 .
Remarque :

- Les degrés de liberté imposés sont définis dans le repère ***global*** de définition du maillage. Si on veut imposer un degré de liberté dans une autre direction, il est possible d'utiliser le mot-clef DTAN .

♦ DTAN
Permet d'appliquer une condition limite dans la direction tangente à l'arête (voir § 3.9.1). On modifie ainsi les valeurs des degrés de liberté de déplacement DX , DY et DZ .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Exemple :

```
ARETE_IMPO = (_F(GROUP_NO = 'LowSide',
                 DX = 0, DY = 0, DZ = 0),
               _F(GROUP_MA = 'RightSide', SANS_GROUP_NO = 'Corner',
                 DTAN = 10),)
```

La signification de la deuxième occurrence de ARETE_IMPO est : « pour tous les nœuds du groupe de mailles 'RightSide' , DTAN = 10 sauf pour ceux du groupe de nœuds 'Corner' . Ceci permet de ne pas avoir de conditions aux limites redondantes.

5.6 Mot-clé FACE_IMPO

```
FACE_IMPO     =_F (    ♦ /TOUT          =  'OUI',
                        /GROUP_MA       =  lgma,          [l_gr_maille]
                      ◊ SANS_GROUP_MA=  lgma1,          [l_gr_maille]
                      ◊ SANS_GROUP_NO=  lgno1,          [l_gr_noeud]
                      ♦ /| DX           =  ux,            [R] ou [fonction]
                        | DY           =  uy,            [R] ou [fonction]
                        |  DZ          =  uz,            [R] ou [fonction]
                        | DRX          =   r x,             [R] ou [fonction]
                        | DRY          =   r y,             [R] ou [fonction]
                        | DRZ          =   r z,             [R] ou [fonction]
                        | GRX          =  g,             [R] ou [fonction]
                        | PRES         =  p,             [R] ou [fonction]
                        | PHI          =  phi,           [R] ou [fonction]
                        | TEMP         =  T,             [R] ou [fonction]
                        |  PRE1        =  pr1 ,          [R] ou [fonction]
                        |  PRE2        =  pr2 ,          [R] ou [fonction]
                       /| DTAN         =  ut,            [R] ou [fonction]
                        | DNOR         =  un,            [R] ou [fonction]
                  )
```

Le mot-clef FACE_IMPO permet d'imposer à tous les nœuds d'une face sur des éléments surfaciques ou volumiques, une ou plusieurs valeurs de degré de liberté.

♦ Affectation topologique : TOUT , GROUP_ MA , SANS_ GROUP_ MA , SANS_ GROUP_ NO
Les conditions cinématiques sont imposées sur les nœuds appartenant aux mailles données par les mots-clefs TOUT , GROUP_ MA tout en excluant éventuellement grâce aux mots-clefs SANS_* .
Les mailles sont nécessairement des triangles **ou des quadrangles en 3D et des segments en 2D** .

♦ Composantes : DX , DY , DZ , DRX , DRY , DRZ , GRX , PRE1 , PRE2 , PRES , PHI , TEMP
La signification de ces degrés de liberté est précisée dans le § 5.1 .
Remarque :
- Les degrés de liberté imposés sont définis dans le repère ***global*** de définition du maillage. Si on veut imposer un degré de liberté dans une autre direction, il est possible d'utiliser le mot-clef DTAN .

♦ DTAN
Permet d'appliquer une condition limite dans la direction tangente à la face (voir § 3.9.1). On modifie ainsi les valeurs des degrés de liberté de déplacement DX, DY et DZ. **Cette condition ne peut fonctionner qu'en 2D, pas en 3D** .

♦ DNOR
Permet d'appliquer une condition limite dans la direction normale à la face (voir § 3.9.1). On modifie ainsi les valeurs des degrés de liberté de déplacement DX, DY et DZ .

Remarque concernant les mots clés DNOR et DTAN :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Si la géométrie des mailles (de peau) sélectionnées n'est pas de bonne qualité, le calcul des normales peut être perturbé et cela peut conduire à des résultats « chahutés ». Si la zone concernée est plane, on peut alors remplacer le mot clé FACE_IMPO / DNOR (ou DTAN) par le mot clé LIAISON_OBLIQUE. On est alors sûr que les nœuds se déplaceront rigoureusement dans la même direction.

5.7 Mot-clé LIAISON_DDL

```
LIAISON_DDL  =_F (  ♦  GROUP_NO            =  lgno,      [l_gr_noeud]
                    ♦  DDL                 =  lddl,      [l_K8]
     Si AFFE_CHAR_MECA
                    ♦  COEF_MULT           =  a_i,       [l_R]
                    ♦  COEF_IMPO           =   b ,       [R]
     Si AFFE_CHAR_MECA_ C
                    ♦  COEF_MULT           =  a_i,       [l_R]
                    ♦  COEF_IMPO           =   b ,    [C]
     Si AFFE_CHAR_MECA_ F
                    ♦  /  COEF_MULT        =  a_i,       [l_R]
                    ♦  /  COEF_MULT_ FONC  =   a _i ,        [ fonction ]
                    ♦  COEF_IMPO           =   b ,         [ fonction ]
               )
```

Le mot-clef LIAISON_DDL permet de définir une relation linéaire entre des degrés de liberté de deux ou plusieurs nœuds. La condition cinématique suivante sera appliquée :

$$\sum_{i=1}^{r} \alpha_i U_i = \beta \qquad (2)$$

Avec U_i la liste des r degrés de liberté, α_i les coefficients et β le second membre.

♦ Affectation topologique : GROUP_ NO
Donne la liste des nœuds N_i ***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,

Attention ! L'ordre des nœuds a une importance (voir les exemples).

♦ DDL
Donne la liste ***ordonnée*** des r degrés de liberté U_i (voir §5.1 pour les degrés de liberté possibles).
Attention ! L'ordre des degrés de liberté a une importance (voir les exemples).

♦ COEF_MULT
Donne la liste ***ordonnée*** des r coefficients **réels** α_i .
Attention ! L'ordre des coefficients a une importance (voir les exemples).

♦ COEF_MULT_FONC
Donne la liste ***ordonnée*** des r coefficients **fonctions** α_i (AFFE_CHAR_MECA_F). Les fonctions de peuvent dépendre que de la ***géométrie initiale*** (voir § 3.2.1).
Attention ! L'ordre des coefficients a une importance (voir les exemples).

♦ COEF_IMPO = a _i
Valeur de la relation linéaire β . Si c'est une fonction (AFFE_CHAR_MECA_F), celle-ci peut dépendre du ***temps ou de la géométrie initiale*** (voir § 3.2.1).

Exemple 1 : on veut imposer une relation linéaire entre les degrés de liberté d'un **même nœud**

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Dans ce cas particulier, on répétera derrière le mot clé GROUP_NO le nom (de groupe) 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 de nom (de groupe) $GN1$, on écrira :

```
LIAISON_DDL =_F  (  GROUP_NO  =  ('GN1', 'GN1'),
                    DDL       =  ('DX', 'DY'),
                    COEF_MULT =  (1., -1.),
                    COEF_IMPO =  0.,)
```

Exemple 2 : on veut imposer une 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 deux 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  =  ('GRNO1' , 'GRNO2'),
                    DDL       =  ('DX', 'DX'),
                    COEF_MULT =  (1. , -1.),
                    COEF_IMPO =  0.,)
```

Cette écriture n'a de sens que si $GRNO1$ et $GRNO2$ 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 [§ 5.12] qui permet de condenser l'écriture de mêmes relations linéaires entre deux groupes de nœuds en vis-à-vis.

5.8 Mot-clé LIAISON_OBLIQUE

```
LIAISON_OBLIQUE  =_F (  ♦  GROUP_NO     =  lgno,          [l_gr_noeud]
                        ♦  |  DX        =  ux,            [R] ou [fonction]
                           |  DY        =  uy,            [R] ou [fonction]
                           |   DZ       =  uz,            [R] ou [fonction]
                           |  DRX       =   r x,             [R]            ou
[fonction]
                           |  DRY       =   r y,             [R] ou
[fonction]
                           |  DRZ       =   r z,             [R] ou
[fonction]
                        ♦  ANGL_NAUT    =  (a,b,g),       [l_R])
```

Le mot-clef LIAISON_OBLIQUE permet d'appliquer, à des nœuds ou des groupes de nœuds, la même valeur de déplacement définie composante par composante dans un repère oblique quelconque.

♦ Affectation topologique : GROUP_NO
Le chargement est affecté sur les nœuds.

♦ Composantes : dx , dy , dz , rx , ry , rz
Valeurs des composantes.

♦ ANGL_NAUT
Liste des trois angles, en degrés, qui définissent le repère oblique d'application des degrés de liberté (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 §3.9.3). Par défaut les angles sont identiquement nuls et donc les composantes de forces sont définies dans le repère global.

5.9 Mot-clé LIAISON_UNIF

```
LIAISON_ UNIF    =_F (  ♦  /GROUP_NO    =  lgno,          [l_gr_noeud]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                /GROUP_MA     =  lgma,          [l_gr_maille]
            ◊   SANS_GROUP_MA=  lgma1,         [l_gr_maille]
            ◊   SANS_GROUP_NO=  lgno1,         [l_gr_noeud]
            ♦   DDL           =  lddl,         [l_K8]
        )
```

Le mot-clef LIAISON_UNIF permet d'imposer à tous les degrés de liberté de tous les nœuds fournis **la même valeur (inconnue)**. On aura donc :

$U_1(N_1)=U_i(N_k)$ pour tous les degrés de libertés U_i désignés sur les nœuds N_k (3)

♦ Affectation topologique : GROUP_MA , GROUP_NO , SANS_GROUP_MA , SANS_GROUP_NO

Les conditions cinématiques sont imposées sur les nœuds N_k définis par le mot-clé GROUP_ NO ou ceux appartenant aux mailles données par le mot-clé GROUP_MA, sans les nœuds définis sous SANS_GROUP_*.

♦ DDL

Donne la liste des degrés de liberté U_i (voir §5.1 pour les degrés de liberté possibles).

5.10 Mot-clé LIAISON_CHAMNO

```
LIAISON_CHAMNO=_F(  ♦  CHAM_NO   =  chamno ,          [cham_no]
                    ♦  COEF_IMPO =  b,                [R]
                    ◊  NUME_LAGR =  /  'NORMAL',      [DEFAUT]
                                    /  'APRES',
                  )
```

Le mot-clef LIAISON_CHAMNO permet de définir une relation linéaire entre tous les degrés de liberté 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].

♦ 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.

♦ COEF_IMPO
Valeur du coefficient réel β appliqué au second membre de la relation linéaire.

◊ NUME_LAGR
Voir § 4.4 .

Exemple :

Supposons que l'on ait un chamno portant sur deux nœuds de nom $N01$ et $N02$ respectivement porteurs des degrés de liberté '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 degrés de liberté :

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


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
        'DRY'     N02    5.
        'DRZ'     N02    2.
```

La relation linéaire que l'on va imposer est :

```
      2.*DX(N01)    +  1.*DY(N01)    +  3.*DZ(N01)
    +  1.*DX(N02)   +  4.*DY(N02)    +  2.*DZ(N02)
    +  3.*DRX(N02)  +  5.*DRY(N02)   +  2.*DRZ(N02) = b
```

5.11 Mot-clé CHAMNO_IMPO

```
CHAMNO_IMPO  =_F(  ♦ CHAM_NO   =  chamno ,          [cham_no]
                   ♦ COEF_IMPO =  b,                [R]
                   ◊ NUME_LAGR =  /  'NORMAL',      [DEFAUT]
                                  /  'APRES')
```

Il s'agit en fait d'une légère adaptation du mot clé LIAISON_CHAMNO (voir §5.10). 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.
Si le champ contient la valeur 2 pour DX, on impose la relation : DX = COEF_IMPO * 2. Le coefficient α est fixé à 1.

♦ CHAM_NO
Nom du cham_no qui sert à définir les seconds membres de la relation linéaire.

♦ COEF_IMPO = b
Coefficient multiplicateur du cham_no.

◊ NUME_LAGR
Voir § 4.4 .

5.12 Mot-clé LIAISON_GROUP

```
LIAISON_ GROUP  =_F(  ♦ /  GROUP_MA_1   =  lgma1,    [l_gr_maille]
                       GROUP_MA_2   =  lgma2,    [l_gr_maille]
                     /  GROUP_NO_1   =  lgno1,    [l_gr_noeud]
                        GROUP_NO_2   =  lgno2,    [l_gr_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     =   a 1i ,        [l_R]
                   ♦ COEF_MULT_2     =   a 2i ,          [l_R]
                   ♦ COEF_IMPO       =  b,        [R] ou [fonction]
                   ◊ SOMMET          =  'OUI',
                   ◊ CENTRE          =  centre ,  [l_R]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
◊  ANGL_NAUT          =  (a,b,g),  [l_R]
◊  TRAN               =   (x,y,z) ,   [l_R] )
```

Le mot-clef LIAISON_GROUP permet de 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 (voir §5.3.1). Le nombre total de relations imposées est égal au nombre de couple de nœuds.

♦ Affectation topologique
GROUP_MA_1 , GROUP_NO_1 :
Première liste de nœuds à mettre en relation (notée Γ_1).

GROUP_MA_2 , GROUP_NO_2 :
Seconde liste de nœuds à mettre en relation (notée Γ_2).

SANS_GROUP_NO :
Cette opérande permet de supprimer de la liste des couples de nœuds en vis-à-vis. C'est-à-dire tous les couples dont au moins un des nœuds appartient à la liste de nœuds décrite par cette opérande. 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
Donne la liste des d egrés de liberté pour le bord Γ_1 (voir § 5.1 pour les degrés de liberté possibles). Si DDL_1 = 'DNOR' , on lie les degrés de liberté de déplacement selon la normale à la surface de l'élément (voir § 3.9.1).

♦ DDL_2
Donne la liste des degrés de liberté pour le bord Γ_2 (voir § 5.1 pour les degrés de liberté possibles). Si DDL_2 = 'DNOR' , on lie les degrés de liberté de déplacement selon la normale à la surface de l'élément (voir § 3.9.1).

♦ COEF_MULT_1
Liste de réels exactement dimensionnée au nombre de degrés de liberté déclarés dans DDL_1 correspondant aux coefficients multiplicateurs de la relation linéaire.

♦ COEF_MULT_2
Liste de réels exactement dimensionnée au nombre de degrés de liberté déclarés dans DDL_2 correspondant aux coefficients multiplicateurs de la relation linéaire.

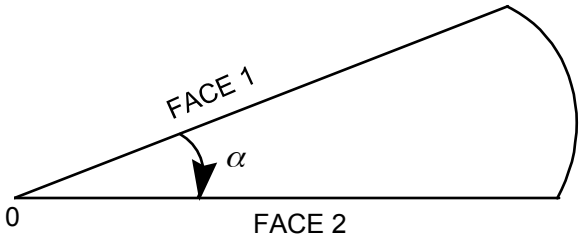
♦ COEF_IMPO
Coefficient de blocage de la relation linéaire.

◊ CENTRE/ANGL_NAUT/TRAN
Les opérandes CENTRE / ANGL_NAUT / TRAN (voir §4.6) 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 [§5.3.1]. La commande effectue d'abord la rotation, puis la translation.

◊ 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.

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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)



Supposons que FACE1 (respectivement FACE2) soit composée de la liste de mailles lma1 (respectivement lma2). Comme la relation doit être bijective, les deux faces comportent nécessairement le même nombre de nœuds $nbno$. On veut écrire les relations linéaires suivantes :

$$\begin{aligned} &\forall\, N_i^1 \text{ nœud de la face } 1 \text{ de vis-à-vis } N_i^2 \\ &\forall\, i=1,\ldots,nbno \\ &\boldsymbol{u.n}\left(N_i^1\right)=\boldsymbol{u.n}\left(N_i^2\right) \end{aligned} \tag{4}$$

Les données de LIAISON_GROUP s'écriront :

```
LIAISON_GROUP=_F (  GROUP_MA_1       =  lma1,
                    GROUP_MA_2       =  lma2,
                    DDL_1         =  'DNOR',
                    DDL_2         =  'DNOR',
                    COEF_MULT_1   =  1.,
                     COEF_MULT_2  =  -1.,
                    COEF_IMPO     =  0,
                    CENTRE        =  (X0,Y0,Z0),
                    ANGL_NAUT     =  (α,0.,0.),
                  )
```

Dans le cas ou FACE2 est perpendiculaire à l'axe X, l'exemple précédent peut aussi s'écrire :

```
LIAISON_GROUP=_F (
                        GROUP_MA_1 =  lma1,
                        GROUP_MA_2 =  lma2,
                        DDL_1 =  ('DX', 'DY'),
                        DDL_2 =  'DY',
                        COEF_MULT_1 =  (-sin(α), cos(α)),
                        COEF_MULT_2 =  -1.,
                        COEF_IMPO =  0
                  )
```

Remarque :

Le nombre total de relations imposées par une occurrence ne dépend pas du nombre d'arguments de DDL_1 ou de DDL_2, qui servent uniquement à enrichir la relation.

Dans le cas suivant, on impose $nbno$ relations :

```
LIAISON_GROUP=_F (
                        GROUP_MA_1 =  lma1,
                        GROUP_MA_2 =  lma2,
                        DDL_1 =  ('DX', 'DY'),
                        DDL_2 =  ('DX', 'DY'),
                        COEF_MULT_1 =  (1, 1),
                        COEF_MULT_2 =  (-1, -1),
                        COEF_IMPO =  0
                  )
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Dans le cas suivant, on impose $2 \cdot nbno$ relations :

```
LIAISON_GROUP=(_F (
                        GROUP_MA_1 =  lma1,
                        GROUP_MA_2 =  lma2,
                        DDL_1 =  'DX',
                        DDL_2 =  'DX',
                        COEF_MULT_1 =  1,
                        COEF_MULT_2 =  -1.,
                        COEF_IMPO =  0)
                      _F (
                        GROUP_MA_1 =  lma1,
                        GROUP_MA_2 =  lma2,
                        DDL_1 =  'DY',
                        DDL_2 =  'DY',
                        COEF_MULT_1 =  1,
                        COEF_MULT_2 =  -1.,
                        COEF_IMPO =  0
                      )
                    )
```

Il faut bien noter que ces deux derniers exemples ne sont pas équivalents.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5.13 Mot-clé LIAISON_MAIL

```
LIAISON_MAIL =_F (    ♦ |  GROUP_MA_ESCL    =  lgma1,        [l_gr_maille]
                        |  GROUP_NO_ ESCL   =  lgno1,        [l_gr_noeud]
                      ♦ GROUP_MA_ MAIT   =  lgma1,       [l_gr_maille]
                      ◊ TYPE_RACCORD       =  /  'MASSIF'  [DEFAUT]
                                              /  'COQUE'
                                              /  'COQUE_MASSIF'
                                              /  'MASSIF_COQUE'
                      ◊ ELIM_MULT          =  /  'NON',    [DEFAUT]
                                              /  'OUI',
                      ◊ DISTANCE_MAX       =  d_max,         [R]
                      ◊ DISTANCE_ALARME    =  d_ala,         [R]

   # si TYPE_RACCORD = 'MASSIF'
                     ◊  ANGL_NAUT          =  (a,b,c)      [l_R]
                     ◊  CENTRE             =  (cx,cy,cz)   [l_R]
                     ◊  TRAN               =  (tx,ty,tz)   [l_R]
                     ◊  DDL_MAIT           =  'DNOR',
                     ◊  DDL_ESCL           =  'DNOR',

   # si TYPE_RACCORD = 'COQUE_MASSIF'
                     ♦  EPAIS              =  epais,       [l_R]
                     ♦  CHAM_NORMALE       =  chanor,      [cham_no])
```

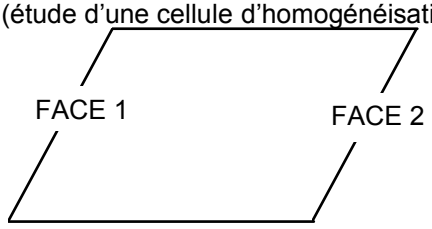
5.13.1 Usage et exemples

Remarque (cf. § 5.3.2) *:*

> *En 3D, il ne faut pas donner des mailles maîtres 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.*

Le mot-clef LIAISON_MAIL permet de définir des relations linéaires pour « 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 FACE1 et FACE2 peuvent être incompatibles.

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



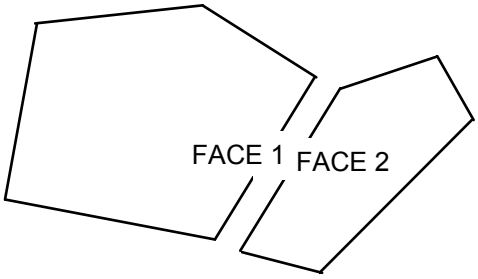
L'expérience a montré que pour les calculs d'homogénéisation périodique, les résultats sont beaucoup plus précis si les 2 faces ont des maillages compatibles (c'est à dire que les maillages de FACE1 et FACE2 sont superposables modulo une isométrie).

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

b) une condition de répétitivité cyclique



c) une condition de simple recollement



Dans la suite de ce paragraphe, on parlera de la face « esclave » (FACE2) et de la face « maître » (FACE1). Le « recollement » des deux faces se fera par écriture de relations linéaires entre les degrés de liberté des deux faces, voir §5.3.2.

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 deux (en 2D) ou trois (en 3D) relations linéaires.

5.13.2 Quelques remarques et précautions d'usage

- Si FACE1 et FACE2 ne sont pas géométriquement confondues mais qu'il existe une isométrie (rotation et translation) entre les deux, l'utilisateur doit définir cette isométrie (celle qui transforme FACE2 en FACE1), grâce aux mots-clefs CENTRE / ANGL_NAUT / TRAN.

- Une utilisation « classique » 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.

- Le mot-clé LIAISON_MAIL est en principe fait pour relier deux 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 ;
  - Et si les mots clés CENTRE / ANGL_NAUT / TRAN n'ont pas été utilisés.

- Il faut être conscient que pour chaque occurrence de LIAISON_MAIL, on relie *a priori* ***tous*** les nœuds esclaves aux mailles maîtres **même si les distances de projection sont importantes** (on émet toutefois des alarmes dans ce cas). On peut changer ce comportement en utilisant le mot clé DISTANCE_MAX.

- Il n'est pas possible d'utiliser le raccord MASSIF_COQUE avec des éléments de type COQUE_3D pour la partie coque.

- Si on écrit :

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


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Ce serait une erreur de penser 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 deux fois et on peut s'attendre à un problème de pivot nul lors de la factorisation.

L'utilisateur doit écrire :

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

5.13.3 Mots-clefs

♦ Affectation topologique

GROUP_MA_ESCL, GROUP_NO_ESCL :
Liste des nœuds **esclaves** à mettre en relation. 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 le mot-clé GROUP_MA_ESCL avec des mailles de type "facette".

GROUP_MA_MAIT :
Liste des **mailles maîtres** à mettre en relation.

◊ DISTANCE_MAX
Lorsque qu'un nœud esclave ne se trouve (géométriquement) dans aucune maille maître, le programme met en relation le nœud et le point (du bord) de la maille la plus proche.

Si l'on souhaite qu'un nœud esclave « lointain » des mailles maîtres ne soit pas concerné par la liaison, il faut utiliser l'opérande DISTANCE_MAX. Cet opérande permet de donner la distance maximale au delà de laquelle on ne fera pas de liaison.

Il n'y a pas de valeur par défaut pour DISTANCE_MAX. Ce qui veut dire que par défaut, la liaison concernera tous les nœuds esclaves.

◊ DISTANCE_ALARME = d_ala
Le mot clé DISTANCE_ALARME = d_ala permet d'être alarmé si un nœud esclave se trouve à une distance d > d_ala de la maille maître la plus proche.

Il n'y a pas de valeur par défaut pour DISTANCE_ALARME. Par défaut, on émet une alarme si la distance d est supérieure à 1/10 ème de la taille de la maille la plus proche (critère relatif).

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

- Si TYPE_RACCORD='MASSIF', les nœuds sont supposés porter des degrés de liberté de translation (DX,DY,DZ). Si l'utilisateur ne précise pas DDL_MAIT='DNOR', on écrira (par exemple en 2D), deux 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 degrés de liberté de translation (DX,DY,DZ) et des degrés de liberté de rotation (DRX,DRY,DRZ). On écrira six relations linéaires pour éliminer les six degrés de liberté de chaque nœud esclave.
- Si TYPE_RACCORD='MASSIF_COQUE', les nœuds esclaves sont supposés « massifs » (translations : DX,DY,DZ) et les nœuds maîtres sont supposés de type « coque » (trois translations et trois rotations). Les degrés de liberté de translation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux translations du point « maître » en vis à vis. Les translations du point maître 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» (six degrés de liberté : DX,DY, DZ, DRX, DRY, DRZ) et les nœuds maîtres sont supposés de type «massif» (DX, DY, DZ). Les degrés de liberté de translation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux translations du point « maître » en vis à vis.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Les degrés de liberté de rotation des nœuds esclaves sont éliminés en écrivant qu'ils sont égaux aux rotations du point « maître » 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).

◊ DDL_MAIT/ DDL_ESCL
Si l'on veut ne recoller que les déplacements normaux aux faces, il faut spécifier DDL_MAIT='DNOR' et DDL_ESCL='DNOR'. La direction normale étant calculée sur la face esclave, il faut donner des mailles de facette, voir GROUP_MA_ESCL). Cette direction normale est transformée par l'éventuelle rotation de la transformation géométrique (voir CENTRE/ANGL_NAUT/ TRAN) pour déterminer la direction normale sur la face maître.

◊ CENTRE/ANGL_NAUT/TRAN
Les opérandes CENTRE / ANGL_NAUT / TRAN (voir § 4.6) permettant de passer de la face esclave à la face maître . La commande effectue d'abord la rotation, puis la translation. 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. Attention ! **Le sens de la transformation est esclave vers maître** .

♦ EPAIS/CHAM_NORMALE
Ces deux mots clés sont obligatoires si TYPE_RACCORD = 'COQUE_MASSIF ': EPAIS donne l'épaisseur de la coque au niveau de la liaison (supposée constante) et CHAM_NORMALE donne un champ aux nœuds qui contient la direction de la normale à la coque sur les nœuds des mailles « esclaves ». Le champ peut être obtenu par la commande :

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

◊ ELIM_MULT
Ce mot clé sert à résoudre le problème qui peut se poser lorsque l'on recolle plusieurs surfaces esclaves adjacentes (c'est à dire 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 deux 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 deux occurrences, les normales « esclaves » ne sont pas les mêmes, l'élimination n'est pas redondante.

5.14 Mot-clé LIAISON_PROJ

5.14.1 But

Mot-clé facteur qui permet de définir des relations linéaires entre les nœuds d'un même modèle. Les coefficients des relations linéaires sont déterminés à l'aide de la commande PROJ_CHAMP qui permet d'avoir comme concept résultat la matrice des coefficients d'influence déterminée à partir des

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

fonctions de forme des éléments. C'est le même algorithme qui est utilisé avec le mot clef LIAISON_MAIL où l'on retrouve la notion de maille maître et esclave que l'on retrouve également dans la commande PROJ_CHAMP.

5.14.2 Syntaxe (AFFE_CHAR_MECA seulement)

```
LIAISON_PROJ = _F(
    ♦ MATR_PROJECTION = chamno,                          [corresp_2_mailla]
    ♦ DDL             = ['DX'|'DY'|'DZ'|'DRX'|'DRY'|'DRZ']
    ◊ TYPE = ['IDENTITE','EXCENTREMENT']                 [defaut]
)
```

5.14.3 Opérandes

♦ MATR_PROJECTION
Nom du concept issu de la commande PROJ_CHAMP. Le concept correspond à la matrice des coefficients obtenue par l'option PROJECTION = 'NON' de la commande PROJ_CHAMP.

Exemple :

```
matcoeff = PROJ_CHAMP(
    PROJECTION = 'NON', METHODE = 'COLLOCATION',
    MAILLAGE_1 = mail, MAILLAGE_2 = mail,
    VIS_A_VIS = _F(GROUP_MA_2 = 'ARMAT', GROUP_MA_1 = 'SDALLE',),
)
```

MAILLAGE_1 et MAILLAGE_2 doivent correspondre au même maillage. La commande AFFE_CHAR_MECA nécessite de renseigner le modèle qui s'appuie sur un maillage. Comme l'on va écrire des relations entre des degrés de liberté d'un modèle, il est obligatoire que les maillages soient identiques. La cohérence est vérifiée lors de l'utilisation de LIAISON_PROJ.
VIS_A_VIS : sous ce mot clef facteur les groupes permettent de définir les mailles maîtres et esclaves :

- le GROUP_MA_1 correspond aux mailles maîtres.
- le GROUP_MA_2 correspond aux mailles esclaves.

♦ DDL
Liste des degrés de liberté des nœuds esclaves auxquels on impose les relations. L'existence des degrés de libertés sur les nœuds esclaves et maîtres est vérifiée.

◊ TYPE = 'EXCENTREMENT'
Permet dans le cas d'existence de degrés de liberté de rotation sur les mailles maîtres d'imposer des déplacements aux nœuds esclaves qui tiennent compte de ces rotations.

Les relations sont de la forme suivante :

- Pour les degrés de libertés des nœuds esclaves donnés sous DDL.
$$DDL(N_{escl}) = \sum_i Coeff_i * DDL(N^i_{maître})$$
avec i : nœud de la maille maître contenant le nœud esclave.
- Dans le cas où TYPE='EXCENTREMENT', la relation sur les degrés de libertés de translation des nœuds esclaves devient :
$$DDL(N_{escl}) = \sum_i Coeff_i * \left(DDL(N^i_{maître}) + \overrightarrow{\omega(N^i_{maître})} \wedge \overrightarrow{N^i_{maître} N_{escl}} \right)$$

L'existence des degrés de libertés sur les nœuds maîtres et esclaves est vérifiée.

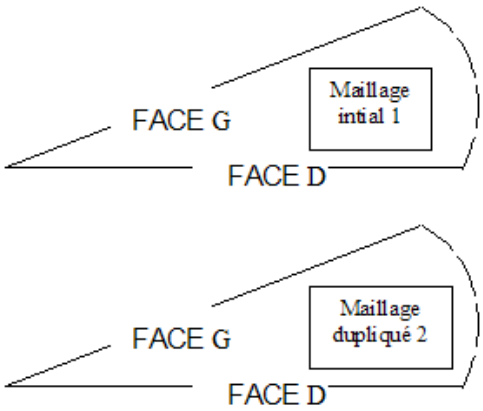
5.15 Mot-clé LIAISON_CYCL

```
LIAISON_ CYCL    =_F (
                    ♦  |  GROUP_NO_ESCL    =  lgno2,        [l_gr_noeud]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
    |   GROUP_MA_ESCL    =   lgma2  ,      [l_gr_maille]
♦   |   GROUP_MA_MAIT1   =   lgma1  ,      [l_gr_maille]
    |   GROUP_MA_MAIT2   =   lgma2  ,      [l_gr_maille]
◊   DDL_MAIT             =   'DNOR',
◊   DDL_ESCL             =   'DNOR',
◊   ANGL_NAUT            =    (a,b,c)      [l_R]
◊    CENTRE                 =    (cx,cy,cz)    [l_R]
◊    TRAN                 =    (tx,ty,tz)    [l_R]
◊   COEF_MAIT1           =   cm1 ,         [R]
◊   COEF_MAIT2           =   cm2 ,         [R]
◊   COEF_ESCL            =   ce ,             [R] )
```

Le mot-clef LIAISON_CYCL permet de 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 $FACEG$ et $FACED$ 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 deux faces se fera par écriture de relations linéaires entre les degrés de liberté des deux 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 deux (en 2D) ou trois (en 3D) relations linéaires.
Si $FACEG$ et $FACED$ 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 $FACEG$ en $FACED$) grâce aux mots-clefs CENTRE/ANGL_NAUT/TRAN.
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 :

$$\begin{bmatrix} q_g^1 \\ q_g^2 \end{bmatrix} = \begin{bmatrix} \cos\beta & \sin\beta \\ -\sin\beta & \cos\beta \end{bmatrix} \begin{bmatrix} q_d^1 \\ q_d^2 \end{bmatrix} \quad (5)$$

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 degrés de liberté 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.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- La seconde permet de lier les degrés de liberté 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.

♦ Affectation topologique

GROUP_NO_ESCL/GROUP_MA_ESCL :

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

GROUP_MA_MAIT_1 :

Ce mot-clé permet 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.

GROUP_MA_MAIT_2 :

Ce mot-clé permetde 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.

◊ COEF_MAIT_1/COEF_MAIT_2/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 :

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

◊ DDL_MAIT/ DDL_ESCL

Si l'on veut ne recoller que les déplacements normaux aux faces, il faut spécifier DDL_MAIT= 'DNOR' et DDL_ESCL='DNOR'. La direction normale étant calculée sur la face esclave, il faut donner des mailles de facette, voir GROUP_MA_ESCL). Cette direction normale est transformée par l'éventuelle rotation de la transformation géométrique (voir CENTRE/ANGL_NAUT/ TRAN) pour déterminer la direction normale sur la face maître.

◊ CENTRE/ANGL_NAUT/TRAN

Les opérandes CENTRE / ANGL_NAUT / TRAN (voir § 4.6) permettant de passer de la face esclave à la face maître . La commande effectue d'abord la rotation, puis la translation. 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. Attention ! **Le sens de la transformation est esclave vers maître** .

5.16 Mot-clé LIAISON_SOLIDE

```
LIAISON_ SOLIDE =_F(     ♦ /  GROUP_MA  =  lgma,          [l_gr_maille]
                           /  GROUP_ NO    =  lg no ,     [l_gr_noeud]
                     ◊  DIST_MIN     =  dmin,         [R]
                     ◊  NUME_LAGR    =  /  'NORMAL', [DEFAUT]
                                        /  'APRES' )
```

Le mot-clef LIAISON_SOLIDE permet 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 et les groupes de nœuds auxquels ils appartiennent. Pour les restrictions d'usage, voir le §3.2.1 .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

En petites perturbations, on impose en 2D $(nb_{ddl} \times nb_{noeud} - 3)$ relations et en 3D $(nb_{ddl} \times nb_{noeud} - 6)$ relations, où nb_{ddl} est le nombre de degrés de liberté par nœud et nb_{noeud} est le nombre de nœuds de la liste donnée après LIAISON_SOLIDE . 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 :

$$\vec{u}(M) = \vec{u}(A) + \vec{\Omega}(A) \wedge \overrightarrow{AM} \quad (6)$$

où A est un nœud arbitraire du solide. Pour plus de détails, voir la doc [R3.03.02].
Il est possible d'utiliser LIAISON_SOLIDE en grandes transformations dans STAT_NON_LINE (puisque le problème devient non-linéaire) uniquement en imposant que le chargement soit suiveur (TYPE_CHARGE='SUIV' , voir [U4.51.03]).

♦ Affectation topologique : TOUT , GROUP_ MA , SANS_ GROUP_ MA , SANS_ GROUP_ NO
Les conditions cinématiques de corps rigide sont imposées sur les nœuds appartenant aux mailles données par les mots-clefs TOUT, GROUP_MA tout en excluant éventuellement grâce aux mots-clefs SANS_* .

◊ DIST_ MIN
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, $dmin = 0.001 \times armin$, où $armin$ est la plus petite arête du maillage.

Remarques :

Si un élément a tous ses nœuds dans une zone « solidifiée » par LIAISON_SOLIDE, sa déformation est nulle. L'état de contrainte est alors également nul ainsi que les efforts généralisés s'il s'agit d'un élément de structure.
Si la liste des nœuds à lier ne comporte que des nœuds alignés ne port ant aucun degré de liberté de rotation, alors, si le système doit être soumis à une rotation, il faut rajouter un nœud fictif non aligné dans la liste . L'autre solution est de se placer dans l'hypothèse des grandes rotations et grands déplacement s (TYPE_CHARGE='SUIV'), on impose en effet explicitement à la liaison solide de vérifier que les distances sont constantes.

5.17 Mot-clé LIAISON_ELEM

```
LIAISON_ ELEM    =_F (  ♦  GROUP_MA_1    =  lgma1,            [l_gr_maille]
                        ♦  /  GROUP_NO_2    =  lgno2,            [gr_noeud]
                           /  GROUP_MA_2    =  lgma2,            [l_gr_maille]

                        ♦  /  OPTION        =  /'3D_POU',
                                               /'3D_TUYAU',
                                               /'3D_POU_ARLEQUIN',
                                               /'COQ_POU',
                                               /'COQ_TUYAU',
                                               /'PLAQ_POUT_ORTH',
                                               /'2D_POU',
                     ◊  NUME_LAGR        =  /'NORMAL',        [DEFAUT]
                                               /'APRES'
                     ◊  ANGL_MAX         =  /1.,              [DEFAUT]
                                               /angl,           [R]
   Si option == '3D_POU_ARLEQUIN'
                     ♦  CARA_ELEM        =  cara,             [cara_elem]
                     ♦  CHAM_MATER       =  mater,            [cham_mater]
   Si option == 'PLAQ_POUT_ORTH'
                     ◊  EXCENT_POUTRE    =  /'OUI',           [DEFAUT]
                                               /'NON',

   Si option == 'COQ_POU' ou option == 'COQ_TUYAU' ou option == '3D_TUYAU'
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
◊  AXE_POUTRE      =  (x,y,z),        [l_R]
◊  CARA_ELEM       =  cara,           [cara_elem] )
```

Le mot-clef LIAISON_ELEM permet de relier des morceaux de structure de modélisations différentes. 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 [R3.03.06] ou un élément de tuyau [R3.08.06] ;

Ce mot-clé permet aussi de raccorder le bord d'une structure 2D avec une poutre ou un élément discret.

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 ou surfacique 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.

◊ NUME_LAGR
Voir § 4.4 .

5.17.1 Option '3D_POU'

Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des poutres d'Euler ou de Timoshenko. Un raccord entre une partie massive 3D et une partie poutre nécessite six relations linéaires.

♦ Affectation topologique

GROUP_ MA_ 1 :

Cette opérande définit 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. **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.

GROUP_ NO _2, GROUP_MA_2 :

Ces opérandes définissent le nœud de la ***poutre*** à raccorder à la partie massive. Donc si l'on utilise GROUP_NO_2 , on ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.
Si on utilise GROUP_MA_2, il faut que la maille soit unique et de type POI1.

◊ ANGL_MAX

Donne l'angle (en degré) permettant de vérifier si les mailles de la liste GROUP_ MA_1 ont des normales faisant un angle supérieur à ANGL_MAX entre elles. Si c'est le cas, il y a émission d'un message d'alarme.

5.17.2 Option '3D_POU_ARLEQUIN'

Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des poutres de Timoshenko dans le cadre Arlequin.

♦ Affectation topologique

GROUP_MA_1 :

Ces opérandes définissent les mailles volumiques de la partie ***massive*** incluses dans la zone de recouvrement. Ces mailles doivent avoir été affectées par des éléments volumiques 3D auparavant. La partie massive peut être maillée avec des éléments linéaires ou quadratiques.

GROUP_MA_2 :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Ces opérandes définissent les mailles de type ***poutre*** à raccorder à la partie massive. Ces mailles doivent avoir été affectées par des éléments 1D de poutres de Timoshenko auparavant.

- CARA_ELEM
  Concept créé par la commande AFFE_CARA_ELEM , contenant les caractéristiques géométriques de la poutre, servant à la construction des matrices de couplage Arlequin.

- CHAM_MATER
  Concept créé par la commande AFFE_MATERIAU , contenant les caractéristiques matériaux de la poutre, servant à la construction des matrices de couplage Arlequin. Ces caractéristiques sont censées être pondérées par l'utilisateur, au sens de la partition de l'unité nécessaire au cadre Arlequin.

5.17.3 Option '2D_POU'

Cette option permet de raccorder une partie surfacique 2D à une partie modélisée avec une poutre d'Euler ou un discret.

- Affectation topologique

GROUP_ MA_1 :

Cette opérande définit les mailles de bord de la partie **2D** à raccorder à l'élément 1D. **La partie surfacique 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.

GROUP_ NO _ 2, GROUP_MA_2 :

Ces opérandes définissent le nœud de la ***poutre*** à raccorder à la partie massive. Si l'on utilise GROUP_NO_2 , on ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud, si l'on utilise GROUP_MA_2, il faut que la maille soit unique et de type POI1.

5.17.4 Option 'COQ_POU'

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

- Affectation topologique

GROUP_ MA_1 :

Cette opérande définit 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.

GROUP_ NO _ 2 :

Cette opérande définit le nœud de la **poutre** à raccorder à la partie coque . On ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

- AXE_POUTRE
  Permet de définir l'axe de la poutre à raccorder, dont l'extrémité est lno2 ou lgno2 (1 seul nœud).
- CARA_ELEM
  Concept créé par la commande AFFE_CARA_ELEM, contenant les caractéristiques géométriques de la coque.

5.17.5 Option '3D_TUYAU'

Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des éléments tuyau. Un raccord entre une partie massive 3D et une partie tuyau nécessite six relations

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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.

♦ Affectation topologique
GROUP_MA_1 :

Cette opérande définit 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.

GROUP_NO_2 :
Cette opérande définit le nœud du **tuyau** à raccorder à la partie massive. On ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

♦ AXE_POUTRE
Définit l'axe du tuyau à raccorder, dont l'extrémité est un seul nœud (lgno2).

♦ CARA_ELEM
Concept créé par la commande AFFE_CARA_ELEM, contenant les caractéristiques géométriques du tuyau.

◊ ANGL_MAX
Donne l'angle (en degré) permettant de vérifier si les mailles de la liste GROUP_MA_1 ont des normales faisant un angle supérieur à ANGL_MAX entre elles. Si c'est le cas, il y a émission d'un message d'alarme.

5.17.6 Option 'COQ_TUYAU'

Cette option permet de raccorder une partie maillée en coque à une partie maillée avec des éléments tuyau. La trace de la section du tuyau sur la partie coque doit correspondre exactement aux mailles de bord définies par 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**. Un raccord entre une partie coque et une partie tuyau nécessite les mêmes relations linéaires que l'option COQ_POU sur les degrés de liberté de poutre de l'élément tuyau en plus des relations sur les degrés de liberté d'ovalisation, de gauchissement et de gonflement.

♦ Affectation topologique
GROUP_MA_1 :
Cette opérande définit 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.
GROUP_NO_2 :
Cette opérande définit le nœud du **tuyau** à raccorder à la partie massive. On ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

♦ AXE_POUTRE
Définit l'axe du tuyau à raccorder, dont l'extrémité est un seul nœud (lgno2).

♦ CARA_ELEM
Concept créé par la commande AFFE_CARA_ELEM, contenant les caractéristiques géométriques du tuyau et de la coque.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5.17.7 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. Dans le but de simplifier l'entrée des données les vérifications suivantes ne sont pas pas réalisées[2] :

- 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.

♦ Affectation topologique

GROUP_MA_1 :

Cette opérande définit 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 .

GROUP_NO_2 :

Cette opérande définit le nœud de la **poutre** à raccorder à la plaque. On ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

◊ ANGL_MAX

Donne l'angle (en degré) permettant de vérifier si les mailles de la liste GROUP_MA_1 ont des normales faisant un angle supérieur à ANGL_MAX entre elles. Si c'est le cas, il y a émission d'un message d'alarme.

◊ VERIF_EXCENT = 'NON'/'OUI'

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, deux 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').

5.18 Mot-clé LIAISON_RBE3

```
LIAISON_RBE3 =_F (
                ♦  GROUP_NO_ MAIT      =  lgno1 ,       [l_gr_noeud]
                ♦  GROUP_NO_ESCL       =  lgno2,        [l_gr_noeud]
                ♦  DDL_MAIT            =   ddlm ,
                ♦  DDL_ESCL            =   ddle ,
                ◊  COEF_ ESCL          =   b i ,        [l_R]
                ◊  NUME_LAGR           =  /'NORMAL',    [DEFAUT]
                                          /'APRES',
                )
```

Le mot-clef LIAISON_RBE3 permet de définir des relations linéaires de type RBE3 entre les degrés de liberté d'un nœud maître et de nœuds esclaves. Il s'agit de relations permettant de spécifier la valeur de certains degrés de libertés d'un nœud maître comme étant la moyenne pondérée de certains déplacements et de certaines rotations de nœuds esclaves.
Les relations linéaires produites sont telles que les efforts vus par le nœud maître sont distribués aux nœuds esclaves proportionnellement à leur distance au centre de gravité des nœuds esclaves.

2 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.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

D'éventuelles pondérations supplémentaires fournies par l'utilisateur peuvent être prises en compte. Pour plus de précisions, on pourra se reporter à la doc de référence [R3.03.08].

Cette option permet de raccorder une partie massive 3D avec une partie modélisée avec des poutres d'Euler ou de Timoshenko. Un raccord entre une partie massive 3D et une partie poutre nécessite six relations linéaires.

- Affectation topologique

  GROUP_NO_MAIT :
  Identification le nœud maître de la relation linéaire. On ne doit donner qu'un seul groupe, celui-ci ne contenant qu'un seul nœud.

  GROUP_ NO_ ESCL :
  Identification des nœuds esclaves de la relation linéaire.

- DDL_MAIT
  Identification des degrés de liberté du nœud maître impliqués dans la relation linéaire. On attend une liste comprenant au plus six entrées parmi 'DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ'.

- DDL_ESCL
  Identification des degrés de liberté des nœuds esclaves impliqués dans la relation linéaire. La liste doit avoir une longueur égale aux nombre de nœuds esclaves Chaque terme de la liste doit être une combinaison des entrées 'DX', 'DY', 'DZ', 'DRX', 'DRY', 'DRZ', séparées par un tiret '-'.

◊ COEF_ESCL
Liste de coefficients de pondérations des termes de la relation linéaire pour chaque nœud esclave. La liste doit :
  - Soit avoir la même longueur que le nombre de nœuds esclaves ;
  - Soit être de longueur 1 , auquel cas ce coefficient est utilisé pour tous les nœuds esclaves

◊ NUME_LAGR
Voir § 4.4 .

Exemple :

Si on veut créer une relation de type RBE3 entre :
- Les degrés de liberté 'DX', 'DY', 'DZ', 'DRX' du nœud maître 'NO1' ;

Et :
- Les degrés de liberté 'DX', 'DY', 'DZ' du nœud esclave 'NO2' avec le coefficient de pondération 0.1 ;
- Les degrés de liberté 'DX', 'DY', 'DZ', 'DRX' du nœud esclave 'NO3' avec le coefficient de pondération 0.2 ;
- Les degrés de liberté 'DX', 'DY', 'DZ', 'DRX' du nœud esclave 'NO4' avec le coefficient de pondération 0.3 ;

On doit écrire la commande :

```
LIAISON_RBE3=_F( GROUP_NO_MAIT    ='NO1',
                 DDL_ MAIT        =('DX', 'DY', 'DZ', 'DRX'),
                 GROUP_NO_ESCL    =('NO2', 'NO3', 'NO4'),
                 DDL_ESCL         =('DX-DY-DZ',
                                    'DX-DY-DZ-DRX',
                                    'DX-DY-DZ-DRX'),
                 COEF_ESCL        =(0.1, 0.2, 0.3),)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

6 Chargements de type Dirichlet pour les éléments de structure

6.1 Mot-clé DDL_POUTRE

```
DDL_POUTRE    =_F (  ♦  /TOUT          =  'OUI',
                        /GROUP_NO      =  lgno,            [l_gr_noeud]
                        /GROUP_MA      =  lgma,            [l_gr_maille]
                     ◊  SANS_GROUP_MA  =  lgma1,           [l_gr_maille]
                     ◊  SANS_GROUP_NO  =  lgno1,           [l_gr_noeud]
                     ♦  |DX            =  ux,              [R]
                        |DY            =  uy,              [R]
                        |DZ            =  uz,              [R]
                        |DRX           =  drx,             [R]
                        |DRY           =  dry,             [R]
                        |DRZ           =  drz,             [R]
                     ◊  GROUP_MA_REPE  =  lgma,            [l_gr_maille]
                     ◊  /ANGL_VRIL     =  G,               [R]
                        /VECT_Y        =  (V1,V2,V3)       [ l_ R]
                      )
```

Le mot-clef DDL_POUTRE permet de bloquer des degrés de liberté 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 ou 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, l'angle de vrille, donné en degrés, permet d'orienter un repère local autour de l'axe X.

♦ Affectation topologique : TOUT , GROUP_ MA , GROUP_ NO , SANS_ GROUP_ MA , SANS_ GROUP_ NO

Les conditions cinématiques sont imposées sur les nœuds donnés par les mots-clefs TOUT , GROUP_ MA , GROUP_ NO tout en excluant éventuellement grâce aux mots-clefs SANS_* .

♦ Composantes : ux , uy , uz , drx , dry , drz
Voir leur signification § 5.1 .

◊ GROUP_MA_REPE
Définition du repère de la poutre sur la dernière maille.

◊ ANGL_VRIL
Angle de vrille, donné en degrés, permet d'orienter un repère local autour de l'axe X.

◊ VECT_Y
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.

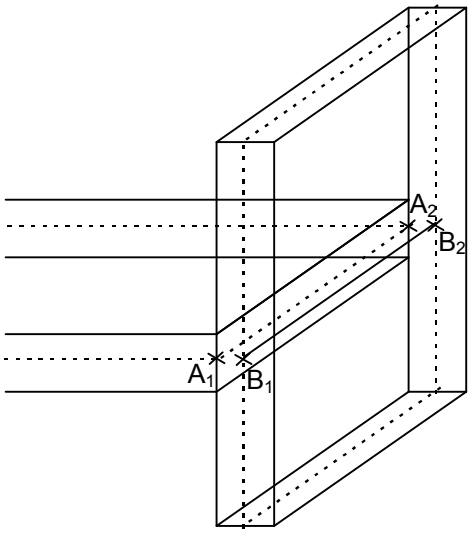
6.2 Mot-clé LIAISON_COQUE

```
LIAISON_COQUE    =_F ( ♦    | GROUP_MA_1      =  l_gma1 ,      [l_gr_maille]
                           | GROUP_NO_1      =  l_gno1 ,      [l_gr_noeud]
                      ◊  SANS_GROUP_MA_1  =  lgma1,        [l_gr_maille]
                      ◊  SANS_GROUP_NO_1  =  lgno1,        [l_gr_noeud]
                      ♦   |GROUP_MA_ 2     =  l_gma 2    ,      [l_gr_maille]
                          |GROUP_NO_2      =  l_gno2 ,      [l_gr_noeud]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
◊  SANS_GROUP_MA_2 =  lgma2,        [l_gr_maille]
◊  SANS_GROUP_NO_ 2   =  lgno 2 ,        [l_gr_noeud]
◊  NUME_LAGR       =  /'NORMAL',    [DEFAUT]
                      /'APRES', )
```

Le mot-clef LIAISON_COQUE permet 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. Les couples de nœuds étant obtenus en mettant en vis-à-vis deux listes de nœuds (voir §5.3.1). 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_1A_2 et la projection de A_1A_2 sur le plan moyen de la coque verticale est B_1B_2 (que l'on a représentée en traits pleins). La liaison entre les deux coques se fait par des liaisons de corps solide (voir §5.16) entre les nœuds en vis-à-vis des segments A_1A_2 et B_1B_2.
Par exemple pour les nœuds A_1 et B_1, on va écrire la formule (valable en petites rotations) :

$$U(B_1)=U(A_1)+\Omega(A_1)\wedge A_1B_1 \quad (7)$$

Et l'égalité des rotations :

$$\Omega(B_1)=\Omega(A_1) \quad (8)$$

♦ Affectation topologique

GROUP_MA_1, GROUP_NO_1 :

Première liste de nœuds à mettre en relation. Les nœuds dont donnés par les mots-clefs GROUP_MA_1, GROUP_NO_1 tout en excluant éventuellement grâce aux mots-clefs SANS_*. Ces nœuds représentent la trace de la coque perpendiculaire sur la coque courante. Sur notre exemple, il s'agirait des nœuds du segment B_1B_2 ou du segment A_1A_2.

GROUP_MA_2, GROUP_NO_2 :

Seconde liste de nœuds à mettre en relation. Les nœuds dont donnés par l es mots-clefs GROUP_MA_2, GROUP_NO_2 tout en excluant éventuellement grâce aux mots-clefs SANS_*. Ces nœuds appartiennent 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_1A_2, la seconde liste est constituée des nœuds de B_1B_2.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

◊ NUME_LAGR
Voir § 4.4 .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

7 Chargements de type Neumann

7.1 Mot-clé FORCE_NODALE

```
FORCE_NODALE=_F  (  ♦  GROUP_NO      =  lgno,          [l_gr_noeud]
                    ♦  |  FX         =  fx,            [R] ou [fonction]
                       |  FY         =  fy,            [R] ou [fonction]
                       |  FZ         =  fz,            [R] ou [fonction]
                       |  MX         =  mx,            [R] ou [fonction]
                       |  MY         =  my,            [R] ou [fonction]
                       |  MZ         =  mz,            [R] ou [fonction]
                    ◊     ANGL_NAUT =  (a,b,g)        [l_R] ou [l_fonction]
                 ),
```

Le mot-clé facteur FORCE_NODALE est 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. En toute rigueur, l'application d'un chargement nodal est physiquement incorrect et peut provoquer des concentrations de contraintes. Il est préférable d'utiliser des chargements répartis.

♦ Affectation topologique : GROUP_NO
Le chargement est affecté sur les nœuds.

♦ Composantes : fx , fy , fz , mx , my , mz
Valeurs des composantes des forces nodales des moments nodaux appliqués aux nœuds spécifiés. Ces forces nodales viendront se superposer aux forces nodales issues, éventuellement, d'autres chargements. En axisymétrique, les valeurs correspondent à un secteur d'un radian (diviser le chargement réel par 2π).

◊ ANGL_NAUT
Liste des trois 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 §3.9.3). Par défaut les angles sont identiquement nuls et donc les composantes de forces sont définies dans le repère global.

7.2 Mot-clé FORCE_ARETE

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

Le mot-clé facteur FORCE_ARETE est utilisable pour appliquer des forces **linéiques**, à une **arête** d'élément **volumique** ou de coque, définies composante par composante dans le repère **global**. Cette arête est définie par une ou plusieurs mailles ou des groupes de mailles de type **segment**.

♦ Affectation topologique : GROUP_ MA
Le chargement est affecté sur les mailles qui sont nécessairement des segments.

♦ Composantes : fx , fy , fz , mx , my , mz
Valeurs des composantes des forces et des moments linéiques appliqués aux mailles spécifiées. Ce chargement s'applique aux modélisations suivantes : DKT , DST , Q4G , 3D et COQUE_3D .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

7.3 Mot-clé FORCE_CONTOUR

```
FORCE_CONTOUR=_F (  ♦  GROUP_MA     =  lgma,        [l_gr_maille]
                    ♦  |  FX        =  fx,          [R] ou [fonction]
                       |  FY        =  fy,          [R] ou [fonction]
                       |  FZ        =  fz,          [R] ou [fonction]
                  ),
```

Le mot-clé facteur FORCE_CONTOUR est utilisable pour appliquer des forces **linéiques** au **bord d'un domaine 2D**, définies composante par composante dans le repère **global**. Ce contour est défini par une ou plusieurs mailles de type **segment**.
Remarque : fondamentalement il s'agit d'une force linéique mais l'unité est une force surfacique car on raisonne pour une épaisseur unitaire (contraintes planes, déformations planes) ou divisé par 2π pour les modélisations axisymétriques (voir §3.5).

- Affectation topologique : GROUP_ MA
  Le chargement est affecté sur les mailles qui sont nécessairement des segments .

- Composantes : fx , fy , fz
  Valeurs des composantes des forces linéiques appliquées aux mailles spécifiées. Ce chargement s'applique aux modélisations suivantes : D_PLAN, AXIS et AXIS_FOURIER, y compris XFEM.

7.4 Mot-clé FORCE_FACE

```
FORCE_FACE =_F   (  ♦  GROUP_MA     =  lgma,        [l_gr_maille]
                    ♦  |  FX        =  fx,          [R] ou [fonction]
                       |  FY        =  fy,          [R] ou [fonction]
                       |  FZ        =  fz,          [R] ou [fonction]
                  ),
```

Le mot-clé facteur FORCE_FACE est utilisable pour appliquer des forces **surfaciques** (donc de la dimension d'une pression) sur une **face d'élément 3D**, définies composante par composante dans le repère **global**. Cette face est définie par une ou plusieurs mailles de type **triangle** ou **quadrangle**.

- Affectation topologique : GROUP_ MA
  Le chargement est affecté sur les mailles qui sont nécessairement des triangles ou des quadrangles .
- Composantes : fx , fy , fz
  Valeurs des composantes des forces surfaciques appliquées aux mailles spécifiées. Ce chargement s'applique aux modélisations suivantes : 3D , 3D_HHM , 3D_HM , 3D_THHM , 3D_THM , 3D_HH2 , 3D_THH2M et XFEM .

7.5 Mot-clé FORCE_INTERNE

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

Le mot-clé facteur FORCE_INTERNE est utilisable dans deux cas :

- Pour appliquer des forces **volumiques** sur un **domaine 3D** , définies composante par composante dans le repère **global** . Ce domaine est définie par une ou plusieurs mailles de type **hexaèdre** , **tétraèdre** , **pyramide** ou **pentaèdre** .
- Pour appliquer des forces **volumiques** sur un **domaine 2D** , définies composante par composante dans le repère **global** . Ce domaine est définie par une ou plusieurs mailles de type **triangle** ou **quadrangle** .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur les mailles qui sont nécessairement des triangles ou des quadrangles en 2D et des hexaèdres , tétraèdres , pyramides ou pentaèdres en 3D.

- Composantes : fx , fy , fz
  Valeurs des composantes des forces volumiques appliquées aux mailles spécifiées.
  Pour le cas 3D , ce chargement s'applique aux modélisations suivantes : 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.
  Pour le cas 2D , ce chargement s'applique aux modélisations suivantes : 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 .

7.6 Mot-clé PRES_REP

```
PRES_REP      =_F( ♦  /  TOUT      =  'OUI',
                   /  GROUP_MA  =  lgma,        [l_gr_maille]
                    | FISSURE   =  fiss,        [fiss_xfem]
                 ♦  |  PRES      =  P,          [R] ou [fonction]
                    |  CISA_2D   =  T,          [R] ou [fonction]
                  ),
```

Le mot-clé facteur PRES_REP est utilisable pour appliquer une **pression** à un domaine de milieu continu **2D ou 3D**, une **pression** sur une coque de type COQUE_3D ou un **cisaillement** à un domaine de milieu continu **2D**.

- Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur les mailles qui sont nécessairement des segments en 2D et des triangles ou quadrangles en 3D.

- FISSURE
  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. Attention, il n'est pas possible d'appliquer un chargement de pression sur les lèvres d'un modèle XFEM de type cohésif.

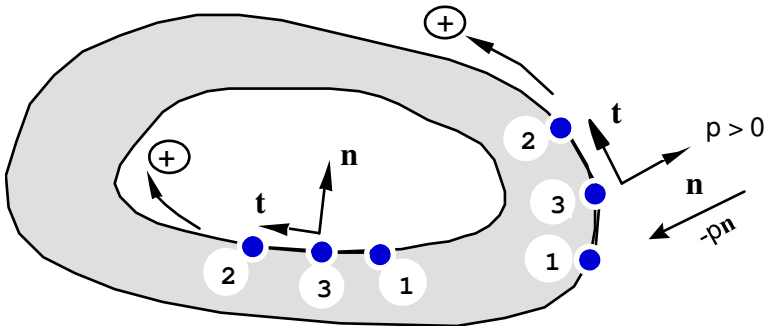
- PRES
  Valeur de la pression imposée. La pression est positive suivant le sens contraire de la normale à l'élément. Soit σ le tenseur des contraintes, le chargement imposé est : $\sigma_{ij} n_i n_j = -p n_i n_j$. Ce chargement s'applique aux modélisations suivantes : AXIS, D_PLAN, C_PLAN, AXIS_FOURIER, D_PLAN_HHM, D_PLAN_HM, D_PLAN_THHM, D_PLAN_THM, AXIS_HHM, AXIS_HM, AXIS_THHM, AXIS_THM, TUYAU_3M, TUYAU_6M, 3D_HHM, 3D_HM, 3D_THHM, 3D_THM, 3D et COQUE_3D.

  Dans le cas d'une pression « fonction », la dépendance à la géométrie peut être faite par rapport à la géométrie initiale avec les paramètres X, Y, et Z ou par rapport à la géométrie réactualisée (uniquement si TYPE_CHAR = 'SUIV') avec les paramètres XF, YF, et ZF .

  Remarque :
  Les pressions de type « fonction », ne sont pour l'instant pas compatibles avec la modélisation COQUE_3D.

- CISA_2D
  Valeur du cisaillement imposé. Le cisaillement est positif suivant la tangente à l'élément. Ce chargement s'applique aux modélisations suivantes : AXIS, D_PLAN, C_PLAN et AXIS_FOURIER.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)



7.7 Mot-clé EVOL_CHAR

EVOL_CHAR = evch

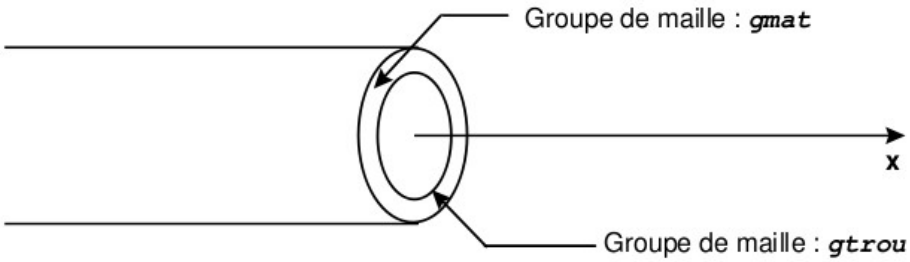
Le mot-clé facteur EVOL_CHAR est utilisable pour appliquer des chargements évolutifs dans le temps de type evol_char produits par LIRE_RESU [U7.02.01] et contenant des champs de pression (correspondant à un chargement de type PRES_REP), des densités de force volumique en 2D ou 3D (correspondant à un chargement de type FORCE_INTERNE) et des densités de force surfacique en 2D ou 3D (correspondant à un chargement de type FORCE_FACE et FORCE_CONTOUR).

Ces chargements sont toujours de type « suiveur » (voir §3.6.2).

7.8 Mot-clé EFFE_FOND

```
EFFE_FOND     =_F (    ♦ GROUP_MA  =  lgma,          [l_gr_maille]
                       ♦ GROUP_MA_INT =  gtrou,      [l_gr_maille]
                       ♦ PRES         =  P   ,       [R] ou [fonction]
                   ),
```

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



- Affectation topologique : GROUP_ MA
  Ensemble des mailles surfaciques (triangles ou quadrangles) modélisant la section matérielle de tuyauterie (gmat sur la figure) où sera appliquée la pression. Cette surface doit avoir été orientée suivant la même convention que pour le chargement PRES_REP (cf. §7.6).

- GROUP_MA_INT
  Ensemble des mailles linéiques (segments) modélisant le contour du trou (gtrou 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 :

$$F_b = \pi R_i^2 P \qquad (9)$$

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

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$F_p = \frac{\pi R_i^2}{\pi\left(R_e^2 - R_i^i\right)} P = P \frac{S_{trou}}{S_{mat}} \tag{10}$$

- PRES
  Pression interne à la tuyauterie. On appliquera en pratique F_p à gmat (une valeur positive correspondant à un effort orienté dans le sens contraire à la normale sortante et un effet de fond induisant une traction sur la tuyauterie).

7.9 Mot-clé PESANTEUR

```
PESANTEUR     =_F (  ◊  GROUP_MA  =  lgma,          [l_gr_maille]
                     ♦  /  GRAVITE   =  g,             [R]
                      ♦  / DIRECTION =  (ap ,bp ,cp),     [l_R]
                  )
```

Le mot-clé facteur PESANTEUR est utilisable pour appliquer un champ de pesanteur sur le modèle. g représente l'intensité du champ de pesanteur et (a_p, b_p, c_p) précise la direction et le sens d'application du champ. Le chargement qui en résulte est de la forme :

$$\rho\, g \frac{\left(a_p \boldsymbol{i} + b_p \boldsymbol{j} + c_p \boldsymbol{k}\right)}{\sqrt{a_p^2 + b_p^2 + c_p^2}} \tag{11}$$

où $(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})$ est le repère cartésien global. et ρ est la masse volumique définie comme caractéristique du matériau (voir opérateurs DEFI_MATERIAU [U4.43.01] et AFFE_MATERIAU [U4.43.03]).
Il ne peut y avoir qu'une seule occurrence de ce mot-clef dans AFFE_CHAR_MECA .

- Affectation topologique : GROUP_ MA
  Par défaut, ce champ s'applique à tout le modèle. Il est possible de le restreindre à une partie du modèle à l'aide du mot-clé GROUP_MA, qui précise les mailles sur lesquelles le champ s'applique (cette possibilité, qui n'a pas de sens physique, est néanmoins très utile pour appliquer progressivement un champ de pesanteur).

- GRAVITE
  Accélération de la pesanteur.

- DIRECTION
  Direction de la pesanteur.

7.10 Mot-clé ROTATION

```
ROTATION      =_F (  ◊  /  TOUT      =  'OUI',
                     /  GROUP_MA  =  lgma,        [l_gr_maille]
                   ♦  VITESSE     =  omega,       [R]
                   ♦  AXE         =  (ar,br,cr),  [l_R]
                   ◊  CENTRE      =  (x,y,z),     [l_R]
                 )
```

Le mot-clé facteur ROTATION est utilisable pour appliquer un champ de force équivalent à la force centrifuge s'appliquant sur une structure en rotation . Soit ω la vitesse de rotation et (a_r, b_r, c_r) l 'axe de rotation.

$$\boldsymbol{\omega} = \omega \frac{\left(a_r \boldsymbol{i} + b_r \boldsymbol{j} + c_r \boldsymbol{k}\right)}{\sqrt{a_r^2 + b_r^2 + c_r^2}} \tag{12}$$

Le chargement qui en résulte est de la forme :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

$$f = \rho(\omega \wedge OM) \wedge \omega \quad (13)$$

Où O est l'origine des coordonnées et M un point courant de la structure avec ρ la masse volumique définie comme caractéristique du matériau (voir opérateurs DEFI_MATERIAU [U4.43.01] et AFFE_MATERIAU [U4.43.03]) .
Il ne peut y avoir qu'une seule occurrence de ce mot-clef dans AFFE_CHAR_MECA .

- ♦ Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur ces mailles .
- ♦ VITESSE
  Vitesse de rotation.
- ♦ AXE
  Axe de rotation. Pour les modélisations planes, l'axe de rotation doit être dans la direction Oz et pour les modélisations axisymétriques et Fourier, il doit être dans la direction Oy .
- ◊ CENTRE
  Si le centre de rotation n'est pas l'origine (défaut), on peut préciser ses coordonnées (x,y,z). Pour les modélisations axisymétriques et Fourier, le centre doit être l'origine.

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. Toutefois, il convient de faire attention : le chargement $\rho(\omega \wedge OM) \wedge \omega$ 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.

7.11 Mot-clé PRE_SIGM

PRE_SIGM
Le mot-clé facteur PRE_SIGM est utilisable pour appliquer une pré-contrainte σ_{pre} . Ce chargement permet d'appliquer des contraintes volumiques moyennes, globalement uniforme (2D ou 3D) à un domaine volumique. Le second membre élémentaire calculé sera :

$$\int_{Ve} \sigma_{pre} : \varepsilon(v^*) dV_e \quad (14)$$

Ce chargement est connu sous l'option FORC_NODA que l'on retrouve dans la commande CALC_CHAMP ou pendant la phase de prédiction de Newton de l'opérateur STAT_NON_LINE .

Le champ de contraintes sigm est de type carte ou cham_elga. Il peut provenir de CREA_CHAMP ou avoir été calculé par ailleurs.

Il ne faut pas confondre cette pré-contrainte avec la contrainte initiale σ_{ini} utilisée en non linéaire, car cette pré-contrainte n'intervient pas directement dans l'expression de la loi de comportement. Ce champ de pré-contrainte est utilisé comme second membre dans les résolutions de MECA_STATIQUE et STAT_NON_LINE.

7.12 Mot-clé PRE_EPSI

```
PRE_EPSI      =_F (  ♦  /  TOUT      =  'OUI',
                        /  GROUP_MA  =  lgma,        [l_gr_maille]
                     ♦  |  EPXX      =  epsxx,       [R] ou [fonction]
                        |  EPYY      =  epsyy,       [R] ou [fonction]
                        |  EPZZ      =  epszz,        [R] ou [fonction]
                        |  EPXY      =  epsxy,        [R] ou [fonction]
                        |  EPXZ      =  epsxz,        [R] ou [fonction]
                        |  EPYZ      =  epsyz,        [R] ou [fonction]
                        |  EPX       =  epsx,        [R]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                    | KY        =  ky,          [R]
                    |  KZ       =  kz,          [R]
                    |  EXX          =  exx,          [R]
                    | EYY       =  eyy,         [R]
                    | EXY       =  exy,         [R]
                    | KXX       =  kxx,         [R]
                    | KYY       =  kyy,         [R]
                    | KXY       =  kxy,         [R]
                )
```

Le mot-clé facteur PRE_EPSI est utilisable pour appliquer une pré-déformation ε_{pre}. C'est un chargement de déformation moyen, globalement uniforme appliqué à un élément 2D, 3D ou de structure. Le second membre élémentaire calculé sera :

$$\int_{Ve} A\,\varepsilon_{pre} : \varepsilon(v^*)\,dV_e \qquad (15)$$

où A désigne le tenseur d'élasticité (récupéré dans le champ matériau pour toutes les lois pour lesquelles sont définies les caractéristiques élastiques).

Il ne faut pas confondre cette pré-déformation avec la déformation initiale ε_{ini} utilisée en non linéaire, car cette pré-déformation n'intervient pas directement dans l'expression de la loi de comportement.

Cette pré-déformation 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.

- Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur ces mailles .

- Composantes : epsxx, epsyy, epszz, epsxy, epsxz, epsyz
  Valeurs des composantes du tenseur des déformations initiales dans le repère global pour les éléments isoparamétriques 2D ou 3D (C_PLAN , AXIS , D_PLAN , 3D , 3D_SI , AXIS_SI , D_PLAN_SI)

- Composante : epsx
  Valeur constante par élément de l'élongation selon l'axe local de la poutre (POU_D_E , POU_D_T , POU_D_TG).

- Composante : ky

  Valeur constante par élément de la variation de courbure selon l'axe y local $-\frac{d\theta_y}{dx}$ de la poutre (POU_D_E , POU_D_TG).

- Composante : kz

  Valeur constante par élément de la variation de courbure selon l'axe z local $\frac{d\theta_z}{dx}$ de la poutre (POU_D_E, POU_D_TG).

- Composantes : exx, eyy, exy
  Valeurs constantes par élément des déformations de membrane dans le repère local de la coque (DKT , DST , Q4G).

- Composantes : kxx, k yy, kxy

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Valeurs constantes par élément des variations de courbure dans le repère local de la coque (DKT , DST , Q4G).

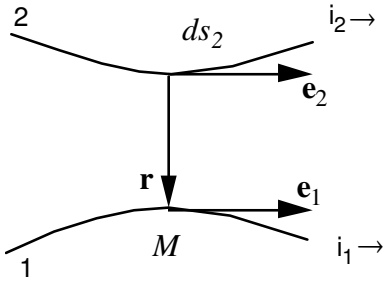
7.13 Mot-clé FORCE_ELEC

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

Le mot-clé facteur FORCE_ELEC est 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) par rapport à ce conducteur principal. Lorsque le conducteur secondaire n'est pas droit, on utilisera le mot-clé INTE_ELEC (voir §7.14).
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.
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 :

$$\boldsymbol{f}(M)=\frac{\boldsymbol{e_1}}{2}\wedge\int_2\frac{\boldsymbol{e_2}\wedge r}{\|r\|^3}ds_2 \tag{16}$$

Dont voici la représentation graphique :



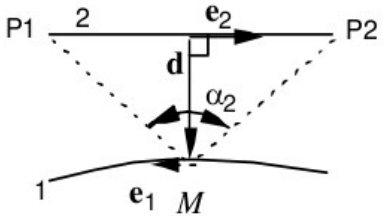
avec $\|\mathbf{e}_1\| = \|\mathbf{e}_2\| = 1$

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

$$\boldsymbol{f}(M)=\frac{\boldsymbol{e_1}}{2}\wedge\frac{\boldsymbol{n}}{d}\left(\sin\alpha_1-\sin\alpha_2\right) \tag{17}$$

Dont voici la représentation graphique :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)



avec $\boldsymbol{n}=\dfrac{\boldsymbol{e}_2\wedge\boldsymbol{d}}{d}$, $d=\|\boldsymbol{d}\|$, $\|\boldsymbol{d}\|=1$

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

$$\boldsymbol{f}(M)=\boldsymbol{e}_1\wedge\frac{\boldsymbol{n}}{\boldsymbol{d}} \tag{18}$$

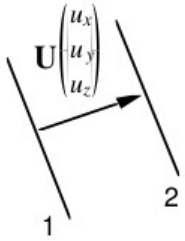
- Affectation topologique : TOUT , GROUP_ MA
  Le conducteur principal s'appuie sur tout ou partie du maillage constitué d'éléments linéiques dans l'espace.

- Composantes : fx, f y, fz
  Dans ce 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 (fx, fy, fz) de la direction de la force de Laplace qui doivent être normées à 1 (soit $fx^2+fy^2+fz^2=1$)

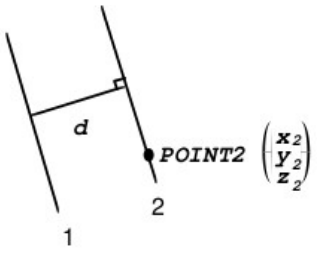
- POSITION = 'PARA'/ 'FINI'/'INFI'
  Précise la manière dont on va définir la position du second conducteur.

  - POSITION ='PARA'
    Pour un conducteur secondaire infini et parallèle au conducteur principal, il y a lors deux manières de définir le conducteur secondaire.

    - TRANS= (u_x ,u_y ,u_z)
      (u_x ,u_y ,u_z) va définir la translation amenant du conducteur principal 1 au conducteur secondaire 2 .



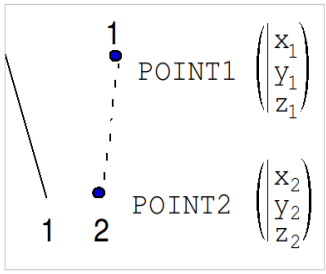
    - DIST = d
      POINT2 = (x_2 ,y_2 ,z_2)
      Le conducteur secondaire 2 est défini par sa distance d au conducteur principal 1 et un deuxième point (x_2 ,y_2 ,z_2).



  - POSITION =' FINI '
    Pour un conducteur secondaire fini et non-parallèle au conducteur principal.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

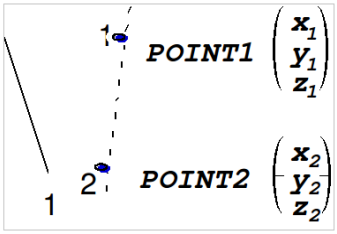
♦ POINT 1 = (x_1, y_1, z_1)
POINT2 = (x_2, y_2, z_2)
Le conducteur secondaire 2 est défini par deux points (x_1, y_1, z_1) et (x_2, y_2, z_2) correspondant à ses extrémités.



Il est préférable de choisir POINT1 et POINT 2 tels que le courant circule de POINT1 à POINT 2.

♦ POSITION =' INFI '
Pour un conducteur secondaire infini et non-parallèle au conducteur principal.

♦ POINT 1 = (x_1, y_1, z_1)
POINT2 = (x_2, y_2, z_2)
Le conducteur secondaire 2 est défini par deux points (x_1, y_1, z_1) et (x_2, y_2, z_2).



Il est préférable de choisir POINT1 et POINT 2 tels que le courant circule de POINT1 à POINT 2.

7.14 Mot-clé INTE_ELEC

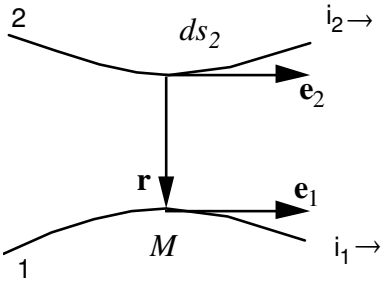
```
INTE_ELEC     =_F (  ♦  /  TOUT       =  'OUI',
                        /  GROUP_MA   =  lgma,                [l_gr_maille]
                     ♦  /  GROUP_MA_2=  lgma2,                [l_gr_maille]
                        /  TRANS      =  (ux,uy,uz),          [l_R]
                        /  SYME       =  (x0,y0,z0,ux,uy,uz), [l_R]
                    )
```

Le mot-clé facteur INTE_ELEC est utilisable pour appliquer la force de Laplace agissant sur un conducteur principal, due à la présence d'un **conducteur secondaire non-droit** s'appuyant sur une partie de maillage, par symétrie ou par translation par rapport au 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.
On rappelle que 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 :

$$\boldsymbol{f}(M)=\frac{\boldsymbol{e}_1}{2}\wedge\int_2\frac{\boldsymbol{e}_2\wedge r}{\|r\|^3}ds_2 \qquad (19)$$

Dont voici la représentation graphique :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

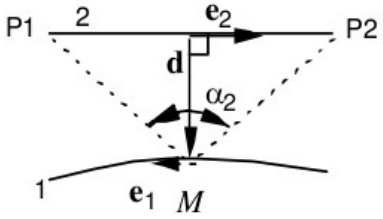


avec $\|\mathbf{e}_1\| = \|\mathbf{e}_2\| = 1$

C'est la même fonction que dans le cas du mot-clef FORCE_ELEC. Par contre, pour chaque élément i du conducteur secondaire, on calcule sa contribution à partir de l'expression précédente et on somme :

$$\boldsymbol{f}(M) = \sum_i \frac{\boldsymbol{e}_1}{2} \wedge \frac{\boldsymbol{n}}{d} \left(\sin \alpha_1 - \sin \alpha_2 \right) \tag{20}$$

Dont voici la représentation graphique :



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

- Affectation topologique : TOUT , GROUP_ MA
  Définition de la géométrie du conducteur principal où le chargement est affecté.

- Affectation topologique : GROUP_ MA_2
  Définition de la géométrie du conducteur secondaire.

- TRANS = (u_x , u_y , u_z)
  (u_x , u_y , u_z) va définir la translation amenant du conducteur principal 1 au conducteur secondaire 2 .

- SYME = (x_0 , y_0 , z_0 , u_x , u_y , u_z)
  Définit une symétrie par rapport au plan donné par le point (x_0 , y_0 , z_0) et la normale (u_x , u_y , u_z) , commune au conducteur principal et au conducteur secondaire.

7.15 Mot-clé VECT_ASSE

VECT_ASSE = chamno [cham_no_DEPL_R]
Le mot-clé facteur VECT_ASSE est utilisable pour appliquer un second membre sous la forme d'un CHAM_NO. Ce CHAM_NO est transmis à ces commandes via le nom du chargement. Le champ de déplacements chamno est de type cham_no. Il peut provenir de CREA_CHAMP ou avoir été calculé par ailleurs.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

8 Chargements de type Neumann pour les éléments de structure

8.1 Mot-clé FORCE_POUTRE

```
FORCE_POUTRE=_F  (  ♦  /  TOUT      =  'OUI',
                       /  GROUP_MA  =  lgma,          [l_gr_maille]
                    ◊  TYPE_CHARGE  =  /'FORCE',      [DEFAUT]
                                       /'VENT',
                    # si TYPE_CHARGE = 'FORCE'
                       ♦  /|  FX     =  fx,           [R] ou [fonction]
                          |  FY     =  fy,            [R] ou [fonction]
                          |   FZ    =  fz,            [R] ou [fonction]
                          |   MX    =  mx,            [R] ou [fonction]
                          |  MY     =  my,           [R] ou [fonction]
                          |  MZ     =  mz,           [R] ou [fonction]
                         / |     N    =  n,              [R] ou [fonction]
                          |   VY    =  vy,            [R] ou [fonction]
                          |   VZ    =  vz,            [R] ou [fonction]
                          |   MT    =  mt,            [R] ou [fonction]
                          |  MFY    =  mfy,          [R] ou [fonction]
                          |  MFZ    =  mfz,          [R] ou [fonction]
                    # si TYPE_CHARGE = 'VENT'
                       ♦  /|  FX     =  fx,           [R] ou [fonction]
                          |  FY     =  fy,            [R] ou [fonction]
                          |   FZ    =  fz,            [R] ou [fonction]
                         / |     N    =  n,              [R] ou [fonction]
                          |   VY    =  vy,            [R] ou [fonction]
                          |   VZ    =  vz,            [R] ou [fonction]
                    )
```

Le mot-clé facteur FORCE_POUTRE est utilisable pour appliquer des forces **linéiques**, sur des éléments de type poutre définis sur tout le maillage ou sur une ou plusieurs 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].
On définit ainsi la force de type VENT . Si p est la pression exercée par le vent sur une surface plane normale à sa direction $\nu=(\nu_x,\nu_y,\nu_z)$ le vecteur unitaire ayant la direction et le sens de la vitesse du vent, et d le diamètre du câble sur lequel s'exerce le vent, alors :

$$\begin{aligned} Fx &= p\,d\,\nu_x \\ Fy &= p\,d\,\nu_y \\ Fz &= p\,d\,\nu_z \end{aligned} \qquad (21)$$

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 ***local*** de définition de la poutre.

- ♦ Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur ces mailles qui sont nécessairement des segments .
- ♦ TYPE_CHARGE = ' FORCE '/ ' VENT '
  Précise le type de charge .
  - ♦ TYPE_CHARGE = ' FORCE '
    - ♦ Composantes : fx , fy , fz , mx , my , mz
      Valeurs des composantes des forces ou des moments linéiques (dans le repère ***global***) appliquées aux mailles spécifiées avec :
      - fx , fy , fz : forces suivant x , y et z

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

• mx , my , mz : moments suivant x , y et z

Ce chargement s'applique aux modélisations suivantes : POU_D_T, POU_D_E, POU_D_TGM et POU_D_TG . Les moments répartis ne peuvent être appliqués que sur les poutres droites à section constante.

♦ Composantes : n, vy, vz, mt, mfy, mfz

Valeurs des composantes des efforts généralisés (dans le repère ***local*** de la poutre) linéiques appliquées aux mailles spécifiées avec :

• n : effort de traction/compression
• vy : effort tranchant suivant y
• vz : effort tranchant suivant y
• mt : moment de torsion
• mfy : moment de flexion suivant y
• mfy : moment de flexion suivant z

Ce chargement s'applique aux modélisations suivantes : POU_D_T, POU_D_E , POU_D_TGM et POU_D_TG . Les moments répartis ne peuvent être appliqués que sur les poutres droites à section constante.

♦ TYPE_CHARGE = ' VENT '

♦ Composantes : fx , fy , fz

Valeurs des composantes des forces ou des moments linéiques (dans le repère ***global***) appliquées aux mailles spécifiées avec :

• fx , fy , fz : forces suivant x , y et z

Ce chargement s'applique aux modélisations suivantes : POU_D_T, POU_D_E , POU_D_TGM et POU_D_TG.

♦ Composantes : n, vy, vz

Valeurs des composantes des efforts généralisés (dans le repère ***local*** de la poutre) linéiques appliquées aux mailles spécifiées avec :

• n : effort de traction/compression
• vy : effort tranchant suivant y
• vz : effort tranchant suivant y

Ce chargement s'applique aux modélisations suivantes : POU_D_T, POU_D_E, P

8.2 Mot-clé FORCE_TUYAU

```
FORCE_TUYAU=_F   (  ♦  /  TOUT      =  'OUI',
                       /  GROUP_MA  =  lgma,        [l_gr_maille]
                    ♦  PRES         =  P,           [R] ou [fonction]
                  )
```

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

♦ Affectation topologique : TOUT , GROUP_ MA

Le chargement est affecté sur ces mailles qui sont nécessairement des segments .

♦ PRES

Valeur de la pression imposée (réel ou fonction). La pression est positive lorsque la pression est interne à la tuyauterie. Ce chargement s'applique aux modélisations TUYAU_3M et TUYAU_6M.

8.3 Mot-clé FORCE_COQUE

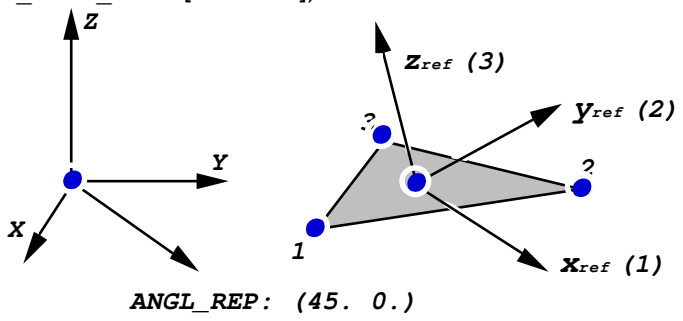
```
FORCE_COQUE =_F  (  ♦  /  TOUT      =  'OUI',
                       /  GROUP_MA  =  lgma,        [l_gr_maille]
                    ◊     PLAN      =  /'MAIL',     [DEFAUT]
                                        / ' MOY ',
                                       /'INF',
                                       /'SUP',
                    ♦  /| FX        =  fx ,         [R] ou [fonction]
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                    | FY        =  fy ,          [R] ou [fonction]
                    | FZ        =  fz ,          [R] ou [fonction]
                    | MX        =  mx ,          [R] ou [fonction]
                    | MY        =  my ,          [R] ou [fonction]
                    | MZ        =  mz ,          [R] ou [fonction]
                    / PRES      =  p,            [R] ou [fonction]
                    /| F1       =  f1 ,          [R] ou [fonction]
                    | F2        =  f2 ,          [R] ou [fonction]
                    | F3        =  f3 ,          [R] ou [fonction]
                    | MF1       =  mf1,          [R] ou [fonction]
                    | MF2       =  mf2,          [R] ou [fonction]
)
```

Le mot-clé facteur FORCE_COQUE est utilisable pour appliquer des efforts **surfaciques** sur des éléments de type coque (DKT, DST, Q4G, COQUE_3D...). Les opérandes de FORCE_COQUE peuvent être définis :

- Dans le repère **global** avec les mot-clefs FX, FY, FZ, MX, MY et MZ si l'on veut imposer un torseur des efforts surfaciques par composante. La composante PRES impose une pression **normale** à la surface ;
- Dans un repère **de référence défini sur chaque maille** avec les mot-clefs F1, F2, F3, MF1 et MF2 si l'on veut imposer un torseur des efforts **surfaciques** par composante ; le 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 de l'opérateur AFFE_CARA_ELEM [U4.42.01]).



Il est aussi possible de définir un torseur d'efforts **surfaciques** sur le plan moyen, inférieur, supérieur ou du maillage. Si on note d l'excentrement, h l'épaisseur de la coque, $(F2X, F2Y, F2Z, M2X, M2Y, M2Z)$ le torseur des efforts sur le plan défini par l'utilisateur (soit excentré) et $(F1X, F1Y, F1Z, M1X, M1Y, M1Z)$ le torseur des efforts dans le plan du maillage.
Les formules de passage sont les suivantes :

- Si le plan de calcul est le plan du maillage :
$$F2 = F1$$
$$M2 = M1$$
- Si le plan de calcul est le feuillet moyen excentré :
$$F2 = F1$$
$$M2X = M1X - dx\,F1Y$$
$$M2Y = M1Y + dx\,F1X$$
- Si le plan de calcul est le feuillet supérieur excentré :
$$F2 = F1$$
$$M2X = M1X - d + \frac{h}{2}.F1Y$$
$$M2Y = M1Y + d + \frac{h}{2}.F1X$$

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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

$$F2 = F1$$
$$M2X = M1X - d - \frac{h}{2}.F1Y$$
$$M2Y = M1Y + d - \frac{h}{2}.F1X$$

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 l'orientation de chaque maille (cf. [§3.9.1]).

♦ Affectation topologique : TOUT , GROUP_ MA
Le chargement est affecté sur ces mailles qui sont nécessairement des triangles ou des quadrangles .

♦ PRES
Valeur de la pression imposée (réel ou fonction), normale à la coque.

♦ Composantes : fx , fy , fz , mx , my , mz
Valeur des efforts et des moments dans le repère global.

♦ Composantes : f1 , f2
Efforts de membrane suivant x_{ref} et y_{ref} .

♦ Composantes : f3
Effort normal suivant z_{ref} .

♦ Composantes : mf1 , mf2
Moments fléchissants d'axe X et Y

◊ PLAN = 'MAIL'/'MOY'/'INF'/'SUP'
Définition du plan pour écrire le torseur d'efforts :
- '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.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

9 Autres chargements

9.1 Mot-clé LIAISON_INTERF

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

Le mot-clé facteur LIAISON_INTERF 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. Il est utilisable avec un modèle contenant à la fois des éléments finis et des macro-éléments statiques condensant certains sous-domaines.

♦ 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 = 'RIGIDE'/'SOUPLE'

- 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.

9.2 Mot-clé RELA_CINE_BP

```
RELA_CINE_BP =_F(  ♦  CABLE_BP      =  cabl_pr,          [cabl_precont]
                   ◊  SIGM_BPEL     =  /' NON ',          [DEFAUT]
                                       /' OUI '
                   ◊  RELA_CINE     =  /'OUI',           [DEFAUT]
                                       /'NON'
                   ◊  DIST_MIN      =  dmin,             [R]
                   ◊  TYPE_EPX      =  /'ADHE'           [DEFAUT]
                                       /'GLIS'
                                       /'FROT'
                 )
```

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 degrés de liberté des nœuds des câbles et les degrés de liberté

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

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]. À 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], comportements incrémentaux exclusivement.

♦ CABLE_BP
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 degrés de liberté des nœuds de ces câbles et les degrés de liberté des nœuds de la structure béton.

◊ SIGM_BPEL = 'OUI'/'NON'
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'/'NON'
Indicateur de type texte par lequel on spécifie la prise en compte des relations cinématiques entre les degrés de liberté des nœuds des câbles et les degrés de liberté des nœuds de la structure béton ; valeur par défaut 'OUI'.

◊ DIST_MIN = dmin
Voir LIAISON_SOLIDE §5.16.

◊ TYPE_EPX = 'ADHE'/'GLIS'/'FROT'
Ce mot-clé n'a d'effet que dans CALC_EUROPLEXUS. Il permet d'indiquer si on souhaite des liaisons câble-béton totales (c'est-à-dire dans les 3 directions de l'espace, correspondant à ce chargement Aster), des liaisons glissantes ou frottantes.

9.3 Mot-clé IMPE_FACE

```
IMPE_FACE      =_F( ♦  GROUP_MA      =  lgma,          [l_gr_maille]
                    ♦  IMPE          =  Q,             [R] ou [fonction]
                  )
```

Le mot-clé facteur IMPE_FACE permet d'appliquer une impédance acoustique dans le cas de la modélisation 3D_FLUIDE.

♦ Affectation topologique : GROUP_ MA
Le chargement est affecté sur ces mailles qui sont nécessairement des triangles ou des quadrangles.

♦ IMPE
Valeur de l'impédance acoustique appliquée à la face.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

9.4 Mot-clé VITE_FACE

```
VITE_FACE      =_F(  ♦ GROUP_MA      =  lgma,          [l_gr_maille]
                     ♦  VNOR         =   V ,           [R] ou [fonction]
                   )
```

Le mot-clé facteur VITE_FACE permet d'appliquer des vitesses normales à une face dans le cas de la modélisation 3D_FLUIDE.

♦ Affectation topologique : GROUP_ MA
Le chargement est affecté sur ces mailles qui sont nécessairement des triangles ou des quadrangles .

♦ VNOR
Valeur de la vitesse normale appliquée à la face.

9.5 Mot-clé ONDE_PLANE

```
ONDE_PLANE     =_F(  ◊  GROUP_MA      =  lgma,          [l_gr_maille]
                     ♦  TYPE_ONDE     =  ty,            [txm]
                     ♦  DIRECTION     =  (k x ,k y ,k z ), [l_R]
                     ♦  FONC_SIGNAL   =  f,             [fonction]
                     ◊  DIST          =  /dist          [R]
                     ◊  DIST_REFLECHI =  dist_r,        [R]
                     ◊  DEPL_IMPO     =  fd,            [fonction]
                   )
```

Le mot-clé facteur ONDE_PLANE (une seule occurrence est permise) permet d'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]). En harmonique, une onde plane élastique est caractérisée par sa direction, sa pulsation et son type (onde P pour les ondes de compression, ondes S , SV ou SH pour les ondes de cisaillement). En transitoire, la donnée de la pulsation, correspondant à une onde stationnaire en temps, doit être remplacée par la donnée d'un profil de déplacement dont on va prendre en compte la propagation au cours du temps dans la direction de l'onde. Plus précisément, on caractérise :

- Une onde P par la fonction $u(x,t)=f\left(k\cdot x-C_p t\right)$;
- Une onde S par la fonction $u(x,t)=f\left(k\cdot x-C_s t\right)\wedge k$;

Avec :

- k le vecteur unitaire de direction ;
- f le profil de l'onde donné selon la direction k ;

Il ne peut y avoir qu'une seule occurrence de ce mot-clef dans AFFE_CHAR_MECA .

♦ Affectation topologique : GROUP_ MA
Le chargement est affecté sur les mailles de frontières absorbantes concernées par l'introduction de l'onde incidente. Si rien n'est donné, par défaut, ce sont toutes les mailles de la modélisation ABSO qui sont concernées.

◊ DIST
Ce paramètre réel permet de déterminer le déphasage en temps dû au passage de l'onde, en indiquant le point d'entrée du chargement par onde plane dans la structure donné par le produit scalaire du vecteur d'onde et de la position du point source d'entrée $\mathbf{k}.\mathbf{x_0}$. Si ce paramètre n'est pas présent, alors on renseignera l'opérande FONC_SIGNAL avec une nappe de signaux temporels.

◊ DIST_REFLECHI
Ce paramètre réel permet la prise en compte de l'onde réfléchie, activée si on donne la position du point de sortie $\mathbf{k}.\mathbf{x_1}$. Dans ce cas, l'expression du profil de l'onde en tenant compte du déphasage spatial lié au passage de l'onde $\dot{f}\left(\mathbf{k}.\left(\mathbf{x}-\mathbf{x_0}\right)-C_m t\right)$ devient alors $\dot{f}\left(\mathbf{k}.\left(\mathbf{x}-\mathbf{x_0}\right)-C_m t\right)+\dot{f}\left(\mathbf{k}.\left(2\mathbf{x_1}-\mathbf{x_0}-\mathbf{x}\right)-C_m t\right)$ avec $m=S$ ou P .

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

♦ TYPE_ONDE = 'P'/'S'/'SV'/'SH'
Type de l'onde (compression ou cisaillement).
P onde de compression
SV onde de cisaillement (uniquement en 3D)
SH onde de cisaillement (uniquement en 3D)
S onde de cisaillement (uniquement en 2D)
Les directions des ondes P, SV et SH sont déterminées à partir du vecteur V renseigné par DIRECTION. A savoir :
– P est colinéaire à V et normé à 1,
– SH est l'intersection du plan horizontal et du plan normal à V, et normé à 1,
– SV est le produit vectoriel de SH et de P. Il existe un cas d'indétermination avec cette règle quand le plan horizontal et le plan normal sont confondus. Dans ce cas, si V = Z purement vertical, on impose SH = Y, et SV = X.

♦ DIRECTION
Direction de propagation de l'onde.

♦ FONC_SIGNAL
Dérivée du profil de l'onde $f(t)$ pour $t \in [0,+\infty[$. **Attention :** c'est la fonction correspondant à la vitesse $v(t)=\dot{u}(t)$ que l'utilisateur donne dans FONC_SIGNAL. Si l'opérande DIST est absent, on donne une nappe de signaux temporels en vitesse paramétrés par des valeurs de coordonnées de propagation $\mathbf{k}.\mathbf{x}$ pour la variable X.

◊ DEPL_IMPO
Profil de l'onde $fd(t)$ pour $t \in [0,+\infty[$. **Attention :** c'est la fonction correspondant à l'intégrale de la vitesse $v(t)=\dot{u}(t)$ que l'utilisateur donne **obligatoirement** dans DEPL_IMPO. Cette donnée **facultative** n'est activée que si on a des raideurs ajoutées sur la frontière absorbante, activées par la donnée d'une valeur non nulle de l'opérande LONG_CARA du mot clé ELAS de DEFI_MATERIAU. Si l'opérande DIST est absent, on donne une nappe de signaux temporels en déplacement paramétrés par des valeurs de coordonnées de propagation $\mathbf{k}.\mathbf{x}$ pour la variable X.

Remarque concernant l'utilisation de cette charge :
Il est fortement conseillé d'orienter au moyen des mots clés ORIE_PEAU_2D ou ORIE_PEAU_3D de MODI_MAILLAGE les mailles de peau affectées par une modélisation de frontière absorbante vers l'extérieur du domaine qu'elles délimitent.

9.6 Mot-clé ONDE_FLUI

```
ONDE_FLUI      =_F(   ♦ GROUP_MA      =  lgma,              [l_gr_maille]
                      ♦ PRES          =   p ,               [R]
                   )
```

Le mot-clé facteur ONDE_FLUI permet d'appliquer une amplitude de pression d'onde incidente sinusoïdale arrivant normalement à une face sur les modélisations 3D_FLUIDE, 2D_FLUIDE et AXIS_FLUIDE.

♦ Affectation topologique : GROUP_ MA
L'onde est appliquée sur ces mailles . Ce sont nécessairement des mailles de bord (segments en 2D et triangles/quadrangles en 3D).

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

9.7 Mot-clé FLUX_THM_REP

```
FLUX_THM_REP =_F (   ♦   /   TOUT          =   'OUI',
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                    /   GROUP_MA      =   lgma,          [l_gr_maille]
                ♦   |   FLUN          =   phiT,          [R] ou [fonction]
                    |   FLUN_HYDR1    =   phie,          [R] ou [fonction]
                    |   FLUN_HYDR2    =   phiv           [R] ou [fonction]
                    |   FLUN_FRAC     =   phif           [fonction]
                )
```

Le mot-clé facteur FLUX_THM_REP permet d'appliquer à un domaine de milieu continu 2D ou 3D un flux de chaleur et/ou un apport de masse fluide (flux hydraulique). Les flux hydrauliques (eau et vapeur) sont définis par :

$$\begin{aligned}\phi^e &= \rho_e\left(\nabla P_e - \rho_e \boldsymbol{g}\right)\cdot\boldsymbol{n}\\ \phi^v &= \rho_v\left(\nabla P_v - \rho_v \boldsymbol{g}\right)\cdot\boldsymbol{n}\end{aligned} \tag{22}$$

Avec la masse volumique de l'eau ρ_e , la masse volumique de la vapeur ρ_v , la pression de l'eau P_e (degré de liberté PRE1) et la pression de la vapeur P_v (degré de liberté PRE2).
Le flux de chaleur est défini par :

$$\phi_T = \lambda_T \frac{\partial T}{\partial n} + h_m^e \phi^e + h_m^v \phi^v + h_m^a \phi^a \tag{23}$$

Avec les enthalpies massiques de l'eau h_m^l , de la vapeur h_m^v , de l'air h_m^a et le flux d'air ϕ^a .

- ♦ Affectation topologique : TOUT , GROUP_ MA
  Le chargement est affecté sur ces mailles Ce sont nécessairement des mailles de bord (segments en 2D et triangles/quadrangles en 3D). Ce chargement n'est utilisable qu'en (thermo)-hydraulique ou en hydraulique pur (THM , THH , THHM , HM ou HHM).
- ♦ FLUN
  Valeur du flux de chaleur.
- ♦ FLUN_HYDR1
  Valeur du flux hydraulique associé au constituant eau.
- ♦ FLUN_HYDR2
  Valeur du flux hydraulique associé au constituant vapeur.
- ♦ FLUN_FRAC
  Valeur du flux hydraulique injecté dans une interface cohésive pour les modèles HM-XFEM. Ce chargement est imposé exclusivement à l'aide d'une fonction de l'espace sur les éléments principaux et non pas les éléments de bord. Pour plus de détails concernant l'utilisation de ce mot-clef, se reporter aux documentations [R7.02.18] et [U2.05.02].

9.8 Mot-clé FORCE_SOL

```
FORCE_SOL =_F(  ♦   |  UNITE_RESU_RIGI  =  uniresri,        [I]
                    |  UNITE _RESU_AMOR  =  uniresam,        [I]
                    |  UNITE_RESU_MASS   =  uniresma,        [I]
                 ◊  TYPE                =  /  'ASCII',      [DEFAUT]
                                           /  'BINAIRE',
                 ◊  UNITE_RESU_FORC      =  uniresfo,        [I]
                 ◊  NB_PAS_TRONCATURE    =  nb_pas,          [I]
                 ♦  / SUPER_MAILLE          =  sup_ma,       [super_maille]
                    / GROUP_NO_INTERF       =  gnintf,          [group_no]
              )
```

Le mot-clé facteur FORCE_SOL permet de prendre en compte la force interne d'un domaine de sol en utilisant les évolutions temporelles des contributions en rigidité, masse et amortissement de l'impédance de sol. L'impédance de sol extraite à l'instant initial permet de constituer par MACR_ELEM_DYNA un macro-élément représentant le comportement du domaine de sol que l'on ajoute au modèle de structure. L'interface dynamique du macro-élément est décrite soit par une super-maille du modèle contenant à la fois la structure et ce macro-élément, soit par un groupe de nœuds si l'interface physique coïncide avec l'interface dynamique modale.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

On peut également prendre en compte, si elle existe, l'évolution temporelle des forces sismiques, affectée à cette même interface dynamique sous forme d'unité logique .
Ce type de charge est pris en compte dans la commande DYNA_NON_LINE. Un exemple d'utilisation est fourni dans le cas test MISS03B [V1.10.122].
Il ne peut y avoir qu'une seule occurrence de ce mot-clef dans AFFE_CHAR_MECA .

♦ Affectation topologique : SUPER_MAILLE/GROUP_NO_INTERF
Ces opérandes permettent de décrire l'interface dynamique du macro-élément représentant le comportement du domaine de sol que l'on ajoute au modèle de structure soit par une super-maille du modèle contenant à la fois la structure et ce macro-élément par le mot clé SUPER_MAILLE, soit par un groupe de nœuds par le mot clé GROUP_NO_INTERF si l'interface physique coïncide avec l'interface dynamique modale.
♦ UNITE_RESU_RIGI/UNITE_RESU_AMOR/UNITE_RESU_MASS
Ces opérandes permettent d'introduire les évolutions temporelles des contributions en rigidité, masse et amortissement de l'impédance de sol sous forme d'unités logiques.
◊ Opérande TYPE
Cet opérande permet d'introduire le format d'écriture commun ('ASCII' ou 'BINAIRE') des évolutions temporelles des contributions en rigidité, masse et amortissement de l'impédance de sol renseignées sous forme d'unités logiques par les opérandes précédents UNITE_RESU_*. La valeur de l'opérande doit être cohérente avec la valeur entrée dans l'opérande TYPE_FICHIER_TEMPS lors du calcul de l'option TYPE_RESU='FICHIER_TEMPS' par CALC_MISS.
◊ Opérande UNITE_RESU_FORC
Cet opérande permet d'introduire, si elle existe et sous forme d'unité logique, l'évolution temporelle des forces sismiques, affectée à l'interface dynamique du macro-élément représentant le comportement du domaine de sol que l'on ajoute au modèle de structure.
◊ Opérande NB_PAS_TRONCATURE
Cet opérande permet d'introduire (s'il est présent) une troncature du calcul de la contribution au second membre de la force de sol dans DYNA_NON_LINE . Cette troncature s'opère sur la sommation à partir des champs de déplacements stockés aux instants précédents. Suite à des tests, on se rend compte qu'il suffit souvent de tronquer la somme sur les 50 à 100 derniers pas de temps au lieu de faire la somme complète sur tous les instants stockés précédents, qui est l'option par défaut. Il est donc conseillé d'utiliser comme valeur pratique la valeur entière du maximum entre 50 et le dixième du nombre total de pas de temps stockés.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)