Opérateur CALC_CHAMP

1 But

Créer ou compléter un résultat en calculant des champs par élément ou aux nœuds (contraintes, déformations, …).

Le concept résultat produit est soit créé, soit modifié, c'est-à-dire que l'appel à CALC_CHAMP se fait de la façon suivante :

```
resu = CALC_CHAMP  (  RESULTAT = resu … ,  reuse = resu , … )
```

ou bien

```
resu1 = CALC_CHAMP  (  RESULTAT = resu, … )
```


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

Table des Matières

1 But....1
2 Syntaxe....4
2.1 Opérandes RESULTAT/MODELE/CHAM_MATER/CARA_ELEM/EXCIT....10
2.1.1 Opérandes RESULTAT....10
2.1.2 Opérandes MODELE / CHAM_MATER / CARA_ELEM....10
2.1.3 Mot clé EXCIT....10
2.2 Sélection des mailles concernées par le calcul....10
2.3 Sélection des numéros d'ordre....11
2.4 Localisation des champs....11
2.5 Dépendance des champs....12
2.6 Opérandes pour les options mécaniques....12
2.6.1 Options de calcul des contraintes (Opérande CONTRAINTE)....12
2.6.2 Options de calcul des déformations (Opérande DEFORMATION)....15
2.6.3 Options d'extraction des variables internes (Opérande VARI_INTERNE)....17
2.6.4 Options de calcul d'énergie (Opérande ENERGIE)....18
2.6.5 Options de calcul de critères (Opérande CRITERES)....21
2.6.6 Option de calcul des flux hydrauliques (Opérande HYDRAULIQUE)....25
2.6.7 Options liées aux paramètres du calcul (Opérande PROPRIETES)....25
2.7 Opérandes pour les options thermiques....25
2.7.1 Opérande THERMIQUE....25
2.8 Opérandes pour les options acoustiques....26
2.8.1 Opérande ACOUSTIQUE....26
2.9 Opérande pour les forces et les réactions nodales....28
2.9.1 Opérande FORCE....28
2.9.2 Exemple 1 : structure chargée avec force nodale (2 éléments QUAD4)....32
2.9.3 Exemple 2 : structure avec chargement de température....33
2.9.4 Exemple 3 : structure sous chargement réparti (poutre)....34
2.9.5 Exemple 4 : calcul des réactions d'appui au pied d'un barrage....35
2.10 Calcul d'un champ utilisateur....37
2.10.1 Opérande NOM_CHAM....37
2.10.2 Opérande CRITERE....37
2.10.3 Opérande FORMULE....37
2.10.4 Opérande NORME....37
2.10.5 Opérande NUME_CHAM_RESU....38
2.10.6 Exemple de calcul d'un champ utilisateur....38
2.11 Opérande pour les sous-points....38
2.11.1 Opérande SOUS_POINT....38
2.12 Opérande TITRE....39

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

2.13 Performance des calculs : parallélisme en espace ou en temps ?.....39

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

2 Syntaxe

```
resu   [*] = CALC_CHAMP

          (  ◊  reuse = resu,
             ◊  MODELE =     mo,                          [modele]
             ◊  CHAM_MATER = chmater,                     [cham_mater]
             ◊  CARA_ELEM =  carac,                       [cara_elem]
             ◊  EXCIT = _F(  ♦  CHARGE = l_charge,        [l_char_meca]
                             ◊  FONC_MULT = fm,        [fonction]
                             # si résultat de type evol_noli
                             ◊  TYPE_CHARGE = /  'FIXE_CSTE', [DEFAUT]
                                             /  'FIXE_PILO',
                                             /  'SUIV',
                                             /  'DIDI'
                             # sinon
                             ◊  TYPE_CHARGE = /  'FIXE_CSTE', [DEFAUT]

                              # si résultat de type dyna_*, *_gene, *acou*
                             ◊ PHAS_DEG = / phas_deg       [R]
                                          / 0.             [DEFAUT]
                             ◊ PUIS_PULS =/ puis_puls      [I]
                                          / 0              [DEFAUT]
                             ◊ FONC_MULT_C =  / fonc_mult_c[fonction]
                             ◊ COEF_MULT_C =  / coef_mult_c[C]
                           )
             ◊  #  Sélection des mailles concernées par le calcul
                /  TOUT =    'OUI',                       [DEFAUT]
                /  GROUP_MA =   l_grma ,               [l_gr_maille]

             ◊  #  Sélection des numéros d'ordre
                /  TOUT_ORDRE =     'OUI',
                /  NUME_ORDRE =     l_nuor ,             [l_I]
                /  LIST_ORDRE =     l_nuor ,             [listis]
                /  NUME_MODE  =     l_numo ,             [l_I]
                /  NOEUD_CMP  =     l_nomo ,             [l_K16]
                /  NOM_CAS    =     nocas  ,             [K16]
                /  INST =    l_inst ,                    [l_R]
                /  FREQ =    l_freq ,                    [l_R]
                /  LIST_INST =  l_inst ,                 [listr8]
                /  LIST_FREQ =  l_freq ,                 [listr8]

             ◊  #  Accélérations des options REAC/FORC_NODA et ***_NOEU
                |  PARALLELISME_TEMPS =      / 'NON',     [DEFAUT]
                                             / 'OUI'.

             ◊  |  CRITERE =    / 'RELATIF',             [DEFAUT]
                                 / 'ABSOLU',
                |  PRECISION =  / prec,
                                 / 1.0E-6,                [DEFAUT]
```

 # options pour des résultats mécaniques linéaires

```
             ♦  RESULTAT =   resu,
```


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

options de calcul des contraintes (éléments de milieu continu 2D et 3D)

◊ CONTRAINTE = | 'EFGE_ELGA'
| 'EFGE_ELNO'
| 'EFGE_NOEU'
| 'SIEF_ELGA'
| 'SIEF_ELNO'
| 'SIEF_NOEU'
| 'SIGM_ELGA'
| 'SIGM_ELNO'
| 'SIGM_NOEU'
| 'SIPM _ELNO'
| 'SIPO_ELNO'
| 'SIPO_NOEU'
| 'SIRO_ELEM'

options de calcul des déformations

◊ DEFORMATION = | 'DEGE_ELGA'
| 'DEGE_ELNO'
| 'DEGE_NOEU'
| 'EPME_ELGA'
| 'EPME_ELNO'
| 'EPME_NOEU'
| 'EPSG_ELGA'
| 'EPSG_ELNO'
| 'EPSG_NOEU'
| 'EPSI_ELGA'
| 'EPSI_ELNO'
| 'EPSI_NOEU'
| 'EPVC_ELGA'
| 'EPVC_ELNO'
| 'EPVC_NOEU'

options de calcul d'énergies

◊ ENERGIE = | 'DISS_ELEM'
| 'DISS_ELGA'
| 'DISS_ELNO'
| 'DISS_NOEU'
| 'ECIN_ELEM'
| 'ENEL_ELEM'
| 'ENEL_ELGA'
| 'ENEL_ELNO'
| 'ENEL_NOEU'
| 'ENTR_ELEM'
| 'EPOT_ELEM'
| 'ETOT_ELEM'
| 'ETOT_ELGA'
| 'ETOT_ELNO'
| 'ETOT_NOEU'

options de calcul de critères

◊ CRITERES = | 'EPEQ_ELGA'
| 'EPEQ_ELNO'
| 'EPEQ_NOEU'
| 'EPMQ_ELGA'
| 'EPMQ_ELNO'

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

```
                                    |   'EPMQ_NOEU'
                                    |   'SIEQ_ELGA'
                                    |   'SIEQ_ELNO'
                                    |   'SIEQ_NOEU'
```

options d'interpolation et d'extraction des variables internes

```
◊   VARI_INTERNE =          |   'VARC_ELGA'
```

options concernant les propriétés du calcul

```
◊   PROPRIETES =        |   'MATE_ELGA'
                        |   ' MATE _EL EM '
```


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

 # options pour les résultats non linéaires (produits par STAT_NON_LINE ou DYNA_NON_LINE) :

♦ RESULTAT = resu, / [evol_noli]

options de calcul des contraintes (éléments de milieu continu 2D et 3D)

◊ CONTRAINTE = | 'EFGE_ELGA'
| 'EFGE_ELNO'
| 'EFGE_NOEU'
| 'SIEF_ELNO'
| 'SIEF_NOEU'
| 'SIGM_ELGA'
| 'SIGM_ELNO'
| 'SIGM_NOEU'
| 'SIPO_ELNO'
| 'SIPO_NOEU'
| 'SIRO_ELEM'

options de calcul des déformations

◊ DEFORMATION = | 'DEGE_ELGA'
| 'DEGE_ELNO'
| 'DEGE_NOEU'
| 'EPFD_ELGA'
| 'EPFD_ELNO'
| 'EPFD_NOEU'
| 'EPFP_ELGA'
| 'EPFP_ELNO'
| 'EPFP_NOEU'
| 'EPME_ELGA'
| 'EPME_ELNO'
| 'EPME_NOEU'
| 'EPMG_ELGA'
| 'EPMG_ELNO'
| 'EPMG_NOEU'
| 'EPSG_ELGA'
| 'EPSG_ELNO'
| 'EPSG_NOEU'
| 'EPSI_ELGA'
| 'EPSI_ELNO'
| 'EPSI_NOEU'
| 'EPSP_ELGA'
| 'EPSP_ELNO'
| 'EPSP_NOEU'
| 'EPVC_ELGA'
| 'EPVC_ELNO'
| 'EPVC_NOEU'

options de calcul d'énergies

◊ ENERGIE = | 'DISS_ELEM'
| 'DISS_ELGA'
| 'DISS_ELNO'
| 'DISS_NOEU'
| 'ENEL_ELEM'
| 'ENEL_ELGA'
| 'ENEL_ELNO'

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

```
                              |   'ENEL_NOEU'
                              |   'ENTR_ELEM'
                              |   'ETOT_ELEM'
                              |   'ETOT_ELGA'
                              |   'ETOT_ELNO'
                              |   'ETOT_NOEU'
```

options de calcul de critères

```
◊  CRITERES =                 |   'DERA_ELGA'
                              |   'DERA_ELNO'
                              |   'DERA_NOEU'
                              |   'ENDO_ELGA'
                              |   'ENDO_ELNO'
                              |   'ENDO_NOEU'
                              |   'EPEQ_ELGA'
                              |   'EPEQ_ELNO'
                              |   'EPEQ_NOEU'
                              |   'EPMQ_ELGA'
                              |   'EPMQ_ELNO'
                              |   'EPMQ_NOEU'
                              |   'INDL_ELGA'
                              |   'PDIL_ELGA'
                              |   'SIEQ_ELGA'
                              |   'SIEQ_ELNO'
                              |   'SIEQ_NOEU'
```

options de calcul des flux hydrauliques (éléments THM)

```
◊  HYDRAULIQUE =              |   'FLHN_ELGA'
```

options concernant les propriétés du calcul

```
◊  PROPRIETES =               |   'MATE_ELGA'
                              |   'MATE_ELEM'
```

 # options thermiques

```
♦  RESULTAT =    resu,                                  /   [evol_ther]

◊  THERMIQUE=                 |   'FLUX_ELGA'
                              |   'FLUX_ELNO'
                              |   'FLUX_NOEU'
                              |   'HYDR_NOEU'
                              |   'SOUR_ELGA'
                              |   'ETHE_ELEM'
```

 # options acoustiques

```
♦  RESULTAT =    resu,                                  /   [acou_harmo]
                                                        /   [mode_acou]

◊  ACOUSTIQUE =               |   'PRAC_ELNO'
                              |   'PRAC_NOEU'
                              |   'PRME_ELNO'
                              |   'INTE_ELNO'
                              |   'INTE_NOEU'
```


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

```
        # options pour les forces et les réactions nodales généralisées

        ♦ RESULTAT =    resu,

        ◊ FORCE =             |   'FORC_NODA'
                              |   'REAC_NODA'

        # calcul d'un champ utilisateur

        ◊ CHAM_UTIL = _F(

            ♦ NOM_CHAM =    ncham,
            ♦ /  CRITERE    =   |   'VMIS',
                                |   'INVA_2'
                                |   'TRACE',
              /  FORMULE    =   l_form,                 [formule]
              /  NORME      =   |   'L2',
                                |   'FROBENIUS'
            ♦ NUME_CHAM_RESU    =   nume,               [I]

            ),

        ◊ TITRE = titre,                                [l_Kn]
        ◊ INFO =     /  1,                              [DEFAUT]
                     /  2,
)
```


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

2.1 Opérandes RESULTAT/MODELE/CHAM_MATER/CARA_ELEM/EXCIT

2.1.1 Opérandes RESULTAT

♦ RESULTAT = resu

Nom de la structure de données résultat à enrichir. Cet argument peut être le même que celui utilisé pour le concept enrichi par l'opérateur, ou un nom différent, ce qui créera une nouvelle structure de données résultat.

Remarques :

- *Dans la majorité des situations, la structure de données resu contient toutes les informations nécessaires au calcul des options : le modèle, le champ de matériau, les caractéristiques élémentaires, les chargements. Les mots-clés MODELE , CHAM_MATER , CARA_ELEM et EXCIT sont donc inutiles ;*
- *Une exception notable concerne toutefois les structures de données dynamiques issues de DYNA_VIBRA pour lesquelles l'utilisateur doit assurer lui-même la cohérence des mots-clés entre l'opérateur de calcul et CALC_CHAMP . Une alarme l'en avertit.*

2.1.2 Opérandes MODELE / CHAM_MATER / CARA_ELEM.

◊ MODELE = mo

Nom du modèle sur lequel sont calculés les efforts, les contraintes, les déformations, etc.
Il est optionnel car il peut être extrait du résultat.

◊ CHAM_MATER = chmater

Champ de matériau associé au modèle mo. Ce mot-clé est optionnel et ne doit être fourni que dans des cas exceptionnels (modification volontaire du matériau par exemple).

◊ CARA_ELEM = carac

Caractéristiques élémentaires associées au modèle mo s'il contient des éléments de structure ou si les éléments iso-paramétriques sont affectés par un repère local d'anisotropie.
Ce mot-clé est optionnel car il peut être extrait du résultat.

2.1.3 Mot clé EXCIT

Ce mot clé facteur (optionnel) permet de spécifier les chargements thermiques ou mécaniques à utiliser pour le calcul des options, en lieu et place de ceux qui ont servi dans le calcul de la structure de données spécifiée sous le mot clé RESULTAT.

La définition de ce mot-clé est identique à celle des commandes qui ont construit la structure de données resu : voir les commandes MECA_STATIQUE [U4.51.01], STAT_NON_LINE [U4.51.03], DYNA_VIBRA [U4.53.03].

2.2 Sélection des mailles concernées par le calcul

Les mots clés TOUT et GROUP_MA permettent à l'utilisateur de choisir les mailles sur lesquelles il souhaite faire ses calculs élémentaires de post-traitement.

/ TOUT = 'OUI'

Toutes les mailles (porteuses d'éléments finis) seront traitées. C'est la valeur par défaut.

/ GROUP_MA = l_grma

Seules les mailles incluses dans l_grma seront traitées.

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

2.3 Sélection des numéros d'ordre

L'emploi des mots-clés TOUT_ORDRE, NUM_ORDRE, INST,FREQ est décrit dans le document [U4.71.00].
Dans le cas où il y a beaucoup de numéros d'ordre, il peut être plus performant (en mémoire notamment) d'appeler la commande plusieurs fois sur des sous-listes des numéros d'ordre.

2.4 Localisation des champs

Dans la suite du document on ne précisera pas explicitement la localisation des champs. En effet, la localisation est donnée dans le nom du champ (et donc de l'option) :

- Champ par élément : *_ELEM
- Champ aux points de Gauss par élément : *_ELGA
- Champ aux nœuds par élément : *_ELNO
- Champ aux nœuds : *_NOEU

Les champs sont, pour la plupart, calculés nativement aux points de Gauss (*_ELGA).

Les champs aux nœuds par élément (*_ELNO) sont obtenus par extrapolation à partir du champ aux points de Gauss (méthode détaillée dans [R3.06.03]).

Les champs aux nœuds (*_NOEU) sont obtenus à partir des champs aux nœuds par élément en faisant une moyenne arithmétique simple (non pondérée par la taille des mailles) des valeurs rencontrées sur les éléments en un nœud donné.

Remarque 1 :

*Pour le calcul des équivalents, les champs aux nœuds par élément (*_ELNO*) ne sont pas obtenus par extrapolation à partir du champ aux points de Gauss. L'extrapolation est faite sur le champ de contraintes ou de déformation puis on calcule le champ d'équivalent.*

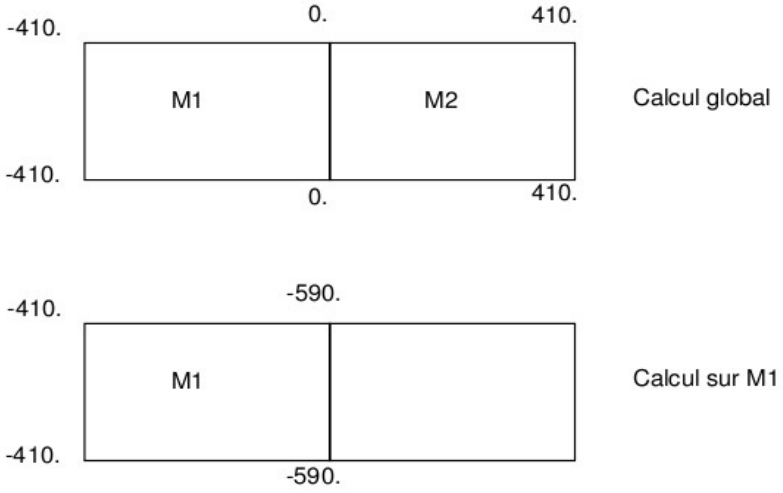
Remarque 2 :

Les moyennes aux nœuds de champs calculés dans des repères locaux ne sont licites que si les angles entre ces repères sont faibles. Dans le cas contraire, elles n'ont pas de sens.

Remarque 3 :

Quand le mot-clé GROUP_MA est *renseigné pour le calcul d'une option *_NOEU, la moyenne arithmétique est faite sur les mailles sélectionnées. Ainsi ce résultat est différent de celui obtenu en faisant un calcul global puis restreint aux seules mailles sélectionnées.*

Exemple : On considère une structure dont la contrainte de cisaillement σ_{xy} *vaut :*



Dans le calcul global, σ_{xy} *est nulle sur* $M1 \cap M2$ *comme moyenne de deux valeurs opposées. Ces valeurs sont loin d'être nulles, comme le montre le calcul sur* $M1$ *seul. Les valeurs sur la frontière du domaine demandé sont donc à interpréter avec précaution.*

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

Pour les éléments de structures qui possèdent des sous-points (plaques, coques, poutres multi-fibres, tuyaux), les champs de type *_ELGA et *_ELNO sont calculés sur tous les sous-points. Pour obtenir un champ sur un seul sous-point (une couche et un niveau par exemple), il faut faire une extraction *via* l'opérateur POST_CHAMP (options EXTR_COQUE, EXTR_PMF et EXTR_TUYAU). D'ailleurs cette opération intermédiaire est indispensable pour calculer un champ de type *_NOEU pour ces éléments de structure : les champs de type *_NOEU n'ont en effet jamais de sous-point.

Enfin les options de calcul d'énergie ne produisent jamais de champs à sous-point. En effet pour les éléments de structure, le champ est intégré dans l'épaisseur (intégration faite sur les sous-points).

2.5 Dépendance des champs

Le calcul d'un champ peut nécessiter le calcul préalable d'un ou plusieurs autres champs. Ainsi par exemple pour calculer un champ *_NOEU il est nécessaire de disposer du même champ par élément aux nœuds *_ELNO et aux points de Gauss *_ELGA.

Cette dépendance est résolue par l'opérateur CALC_CHAMP qui réalise automatiquement le calcul des champs intermédiaires. Il n'est donc pas nécessaire pour l'utilisateur de connaître l'arbre de dépendance des options.
Seuls les champs explicitement demandés par l'utilisateur sont sauvegardés dans la structure de données résultat.

2.6 Opérandes pour les options mécaniques

2.6.1 Options de calcul des contraintes (Opérande CONTRAINTE)

Les composantes des champs de contraintes et efforts généralisés sont détaillées dans le document [U2.01.05].

```
|   'EFGE_ELGA'
|   'EFGE_ELNO'
|   'EFGE_NOEU'
```

Calcul des efforts généralisés (éléments de structure).

Il s'agit soit d'une extraction des efforts contenus dans le champ SIEF_ELGA/STRX_ELGA (cas des éléments de poutres/tuyaux ou des discrets), soit d'un calcul par intégration des contraintes (cas des éléments de poutres multi-fibres ou de plaques et coques).

Remarque 1 :
Le champ EFGE_ELNO n'est pas toujours une extrapolation du champ EFGE_ELGA ; notamment pour un calcul linéaire où ce champ est calculé directement à partir du déplacement. C'est pourquoi certaines composantes ne sont pas calculées (mises à zéro) en non-linéaire.

Remarque 2 :
Pour les plaques excentrées, les efforts sont calculés dans le « plan » du maillage. Si on souhaite ces efforts dans le « plan » moyen de la plaque, il faut utiliser la commande POST_CHAMP / COQUE_EXCENT.

```
|   'SIEF_ELGA'
|   'SIEF_ELNO'
|   'SIEF_NOEU'
```

Calcul de l'état de contrainte (contraintes ou efforts généralisés selon la modélisation) à partir des déplacements (élasticité linéaire), voir [U2.01.05].
Pour les formulations incompressibles (INCO_UP, INCO_UPO et INCO_UPG), ce tenseur contient en plus des composantes SIXX, SIYY, SIZZ, SIXY, SIYZ et SIXZ en 3D, la composante SIP qui est la pression hydrostatique. Cette composante est calculée à partir de la trace des contraintes (partie de la loi de comportement) et des inconnues PRES et GONF (pour INCO_UPG) du champ des inconnues

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

nodales. Les composantes SIXX, SIYY et SIZZ sont alors les contributions purement déviatoriques du tenseur des contraintes

Remarque :
Le champ 'SIEF_ELGA' est calculé nativement par les opérateurs de résolution non-linéaires. Il est toujours présent dans une structure de données résultat de type evol_noli.

| 'SIGM_ELGA'
| 'SIGM_ELNO'
| 'SIGM_NOEU'

Calcul de l'état de contrainte.

Il s'agit en réalité d'une extraction des contraintes contenues dans le champ SIEF_ELGA, voir [U2.01.05].

| 'SIPO_ELNO'
| 'SIPO_NOEU'

Calcul des contraintes dans la section de poutre décomposée en contributions de chaque effort généralisé.

Liste des composantes du champ :

SN	Contribution de l'effort normal N à σ_{xx} , $\sigma_{xx}=\frac{N}{A}$
SMFY	Contribution du moment de flexion MFY à σ_{xx} , $\sigma_{xx}=z\frac{MFY}{I_Y}$
SMFZ	Contribution du moment de flexion MFZ à σ_{xx} , $\sigma_{xx}=-y\frac{MFZ}{I_Z}$
SVY	Contribution de l'effort tranchant VY à σ_{xy} , $\sigma_{xy}=\frac{VY\,a_Y}{A}$ a_Y coefficient de cisaillement dans la direction y
SVZ	Contribution de l'effort tranchant VZ à σ_{xz} , $\sigma_{xz}=\frac{VZ\,a_Z}{A}$ a_Z coefficient de cisaillement dans la direction z
SMT	Contribution du moment de torsion MX à σ_{yz} , $\sigma_{yz}=\frac{MX\,R_T}{J_x}$

Les contraintes ci-dessus s'expriment dans le repère local, c'est-à-dire le repère principal d'inertie de la section droite [R3.08.01].

Les valeurs de σ_{xx} dues aux deux moments de flexion sont les valeurs maximales de celles calculées en Y_{min} , Y_{max} d'une part, et en Z_{min} , Z_{max} d'autre part (sauf pour une section générale où c'est l'utilisateur qui fournit la localisation de l'extremum avec les mot-clés RY , RZ et RT *cf.* AFFE_CARA_ELEM [U4.42.01]).

Pour une section rectangulaire :

- on calcule la valeur de SMFY en $z=HZ/2$,
- on calcule la valeur de SMFZ en $y=HY/2$.

Pour une section circulaire, on calcule les valeurs de SMFY et SMFZ pour y et z valant R .

| 'SIPM_ELNO'

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

Calcul des contraintes maximum et minimum dans la section de poutre à partir des efforts généralisés (élasticité linéaire).
La même remarque que pour SIPO_ELNO s'applique dans le cas d'une section générale.

| 'SIRO_ELEM'

Calcul des contraintes projetées sur la peau d'un volume (par exemple sur les parements d'un ouvrage hydraulique) ou sur l'arête d'une surface.

Liste des composantes du champ :

SIG_NX SIG_NY SIG_NZ	Composantes σ_X, σ_Y, σ_Z dans le repère global de $\vec{\sigma}_n$
SIG_N	Valeur SIG_N
SIG_TX SIG_TY SIG_TZ	Composantes σ_X, σ_Y, σ_Z dans le repère global de $\vec{\sigma}_t$
SIG_T1X SIG_T1Y SIG_T1Z	Composantes σ_X, σ_Y, σ_Z dans le repère global de $\vec{\sigma}_{t1}$
SIG_T1	Valeur propre SIG_{T1}
SIG_T2X SIG_T2Y SIG_T2Z	Composantes σ_X, σ_Y, σ_Z dans le repère global de $\vec{\sigma}_{t2}$
SIG_T2	Valeur propre SIG_{T2}
SIG_TN	Valeur SIG_{TN}

Ces champs sont évalués à partir d'un champ de contraintes calculés sur les mailles volumiques (MODELISATION='3D' ou '3D_SI') ou surfaciques (MODELISATION='D_PLAN' ou 'D_PLAN_SI'). Pour des mailles volumiques, la procédure est la suivante :

- Identification des mailles volumiques correspondant aux facettes du groupe de mailles surfaciques. Pour chaque facette (maille surfacique), on choisit la maille volumique située du côté « - » de la normale à la facette. S'il n'y a pas de maille volumique du coté « - » de la facette, on ne calcule pas SIRO_ELEM sur cette facette ;
- Récupération des contraintes 3D pour les affecter aux nœuds des faces ;
- Moyenne de chacune des composantes du tenseur des contraintes au centre des faces d'éléments ;
- On se place dans un repère composé par le vecteur normal $\vec{n}$ à la facette et le plan de la facette. On obtient un tenseur noté $[\sigma]$.
- On évalue $[\sigma]\vec{n}=\vec{\sigma}_n+\vec{\sigma}_t$, $\vec{\sigma}_n$ étant un vecteur colinéaire à $\vec{n}$. $\vec{\sigma}_t$ est alors un vecteur représentant le cisaillement qui est négligeable dans le cas des parements amont/aval d'un barrage. On note $\vec{\sigma}_n=SIG_N\vec{n}$ et SIG_N indique la présence de traction s'il est positif et de la compression s'il est négatif.
- On se place dans l'hypothèse d'un cisaillement négligeable donc $[\sigma]=\begin{bmatrix}\sigma_{2D} & 0\\ 0 & SIG_N\end{bmatrix}$

  On cherche les vecteurs de contraintes principales correspondant à σ_{2D} : on obtient donc les vecteurs $\vec{\sigma}_{t1}$ et $\vec{\sigma}_{t2}$ qui sont dans le plan de la facette et les valeurs propres SIG_{T1} et SIG_{T2}

Pour des mailles surfaciques, la procédure d'évaluation du champ SIRO_ELEM est la même que pour les mailles volumiques, en tenant compte des conventions suivantes :

- $\vec{\sigma}_{t2}$ colinéaire à la direction Z (un modèle D_PLAN étant défini exclusivement dans le plan XY), $\vec{\sigma}_{t1}$ étant donc dans le plan XY.

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

- SIG_NZ = SIG_TZ = 0. et SIG_T1Z = SIGT2X = SIGT2Y = 0.
- SIG_{TN} est la valeur de la contrainte de cisaillement dans le plan XY.

Remarque 1 :
Dans le cas de facettes/arêtes plongées dans le volume/surface, l'utilisateur a la possibilité grâce à la commande MODI_MAILLAGE / ORIE_PEAU_3D / GROUP_MA_VOLU ou MODI_MAILLAGE / ORIE_PEAU_2D / GROUP_MA_SURF de réorienter cette normale comme il le souhaite. Il peut ainsi choisir la maille volumique qui sera utilisée pour le calcul.
Si du côté « - », on trouve une maille de « joint » (qui est volumique/surfacique), le calcul de SIRO_ELEM est impossible car les contraintes stockées dans les éléments de joint ne permettent pas le calcul détaillé ci-dessus.

Remarque 2 :
Si l'on renseigne TOUT='OUI', la liste des mailles est filtrée pour ne garder que les mailles de peau/arête.

2.6.2 Options de calcul des déformations (Opérande DEFORMATION)

Les composantes des champs de déformation sont détaillées dans le document [U2.01.05].

| 'DEGE_ELGA'
Calcul des déformations généralisées à partir des déplacements. Cette option n'a de sens que pour les éléments de structure de plaque et tuyau, pas pour les poutres
Les déformations généralisées sont obtenues dans le repère local de l'élément.

| 'DEGE_ELNO'
| 'DEGE_NOEU'
Calcul des déformations généralisées à partir des déplacements. Cette option n'a de sens que pour les éléments de structure de poutre, plaque et tuyau.
Les déformations généralisées sont obtenues dans le repère local de l'élément.

| 'EPFD_ELGA'
| 'EPFD_ELNO'
| 'EPFD_NOEU'

Calcul des déformations de fluage de dessiccation, pour les modèles BETON_UMLV_FP et BETON_BURGER_FP.

| 'EPFP_ELGA'
| 'EPFP_ELNO'
| 'EPFP_NOEU'

Calcul des déformations de fluage propre associées au modèle GRANGER_FP, au modèle BETON_UMLV_FP ou au modèle BETON_BURGER_FP.

| 'EPME_ELGA'
| 'EPME_ELNO'
| 'EPME_NOEU'

Calcul des déformations « mécaniques » à partir des déplacements. Ce calcul est fait en théorie des « **petits déplacements** ». Les déformations calculées sont égales aux déformations totales moins les déformations thermiques. Les déformations de séchage et d'hydratation sont aussi soustraites ainsi que les déformations de pression de fluide et les déformations anélastiques. Par contre les déformations de fluage ne sont pas soustraites.

$$\varepsilon_{ij}^{m}(u)=\frac{1}{2}\left(u_{i,j}+u_{j,i}\right)-\varepsilon^{th}$$

| 'EPMG_ELGA'

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

| 'EPMG_ELNO'
| 'EPMG_NOEU'

Calcul des déformations « mécaniques » à partir des déplacements. Ce calcul est fait en théorie des « **grands déplacements** ». Les déformations calculées sont égales aux déformations totales moins les déformations thermiques.

$$E_{ij}^{m}(u)=\frac{1}{2}\left(u_{i,j}+u_{j,i}+u_{k,i}u_{k,j}\right)-\varepsilon^{th}$$

| 'EPSG_ELGA'
| 'EPSG_ELNO'
| 'EPSG_NOEU'

Calcul des déformations de Green-Lagrange.

$$E_{ij}(u)=\frac{1}{2}\left(u_{i,j}+u_{j,i}+u_{k,i}u_{k,j}\right)$$

| 'EPSI_ELGA'
| 'EPSI_ELNO'
| 'EPSI_NOEU'

Calcul des déformations à partir des déplacements.

$$\varepsilon_{ij}(u)=\frac{1}{2}\left(u_{i,j}+u_{j,i}\right)$$

Pour les éléments de structure, ces déformations sont obtenues dans le repère local de l'élément.

| 'EPSP_ELGA'
| 'EPSP_ELNO'
| 'EPSP_NOEU'

Calcul des déformations anélastiques à partir du champ de déplacement u, des contraintes σ, de la température T, des déformations anélastiques éventuelles ε^{a}, et des variables internes,

$$\varepsilon^{p}=\varepsilon(u)-A^{-1}\sigma-\varepsilon^{th}(T)-\varepsilon^{a}-\varepsilon^{fl}$$

où ε^{fl} est la déformation de fluage propre de Granger.

| 'EPVC_ELGA'
| 'EPVC_ELNO'
| 'EPVC_NOEU'

Calcul des déformations liées aux variables de commande. Pour l'instant seules les composantes suivantes sont définies :

- les déformations thermiques : EPTHER_L, EPTHER_T, EPTHER_N telles que : $\varepsilon_i^{th}=\alpha_i(T-T_{ref});i\in\{L,T,N\}$ (si le matériau est isotrope, les 3 composantes sont égales), T étant la température et α_i le coefficient de dilatation thermique ;
- le retrait de séchage EPSECH (utilisé pour les lois décrivant le comportement du béton) $\varepsilon^{sech}=-K_{dessic}(S_{ref}-S)$, S étant la variable de commande séchage et K_{dessic} le coefficient de retrait de dessiccation ;
- le retrait d'hydratation EPHYDR (utilisé pour les lois décrivant le comportement du béton) $\varepsilon^{hydr}=-B_{endog}h$, h étant la variable de commande hydratation, et B_{endog} étant le coefficient de retrait endogène.

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

- La déformation liée à la pression de fluide (pour la thermo-hydro-mécanique avec une résolution par chaînage) : EPPTOT telle que : $\varepsilon^{ptot}=\frac{b}{3K}p_{tot}$, p_{tot} est la variable de commande pression totale de fluide, b est le coefficient de Biot, K est le module d'élasticité.

| 'EPSL_ELGA'
| 'EPSL_ELNO'
| 'EPSL_NOEU'

Calcul des déformations logarithmiques.

$$\boldsymbol{E}_l(u)=\frac{1}{2}\left(\ln\left(\boldsymbol{F}^T\boldsymbol{F}\right)\right)$$

2.6.3 Options d'extraction des variables internes (Opérande VARI_INTERNE)

| 'VARC_ELGA'

Calcul des variables de commandes ayant servi à un calcul mécanique.

Liste des composantes du champ :

TEMP	Cf. documentation de la commande AFFE_MATERIAU [U4.43.03] pour la définition de chacune des composantes.
HYDR	
SECH	
CORR	
IRRA	
PTOT	
DIVU	
NEUT1	
NEUT2	

Remarque :

Toutes les composantes sont systématiquement calculées. Les variables qui n'ont pas été définies sont initialisées à la valeur R8VIDE() (nombre réel très grand de l'ordre de 1.0E+308).

| 'VARI_ELNO'
| 'VARI_NOEU'

Calcul des variables internes.

Liste des composantes du champ :

V1	Variable interne 1
...	
Vi	Variable interne i
...	
Vn	Variable interne n

Le nombre et le type de ces variables internes sont spécifiques à chaque modèle de comportement (*cf.* [U4.51.11]).

Remarque :

Le champ 'VARI_ELGA' est calculé nativement par les opérateurs de résolution non-linéaires. Il est toujours présent dans une SD résultat de type evol_noli.

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

2.6.4 Options de calcul d'énergie (Opérande ENERGIE)

| 'DISS_ELEM'

Calcul de l'énergie dissipée par l'endommagement. Le champ obtenu possède une seule composante de nom 'ENDO'.

Liste des composantes du champ :

ENDO	Énergie dissipée par l'endommagement

Remarque :
Valable uniquement pour les éléments DKTG et la loi GLRC_DM. Son expression est donnée dans [R7.01.32].

| 'DISS_ELGA'
| 'DISS_ELNO'
| 'DISS_NOEU'

Calcul de la densité d'énergie dissipée par l'endommagement. Le champ obtenu possède une seule composante de nom 'ENDO'.

Liste des composantes du champ :

ENDO	Énergie dissipée par l'endommagement

Remarque :
Valable uniquement pour les éléments DKTG et la loi GLRC_DM. Son expression est donnée dans [R7.01.32].

| 'ECIN_ELEM'

Calcul de l'énergie cinétique.

$$E_c = \frac{1}{2} m v^2$$

Liste des composantes du champ :

TOTALE	Énergie cinétique
Composantes supplémentaires pour les plaques et coques :	
MEMBRANE FLEXION	Contributions à l'énergie cinétique (cf. [R3.07.03])
Composantes supplémentaires pour les poutres courbes :	
PLAN_XY PLAN_XZ	Contributions à l'énergie cinétique (cf. [R3.08.01])
Composantes supplémentaires pour les discrets :	
DX DY DZ DRX DRY DRZ	Contributions à l'énergie cinétique

| 'ENEL_ELEM'

Calcul de l'énergie élastique.

$$E_p = \frac{1}{2} \sigma \, A^{-1} \sigma$$

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

Liste des composantes du champ :

TOTALE	Énergie élastique
Composantes supplémentaires pour les plaques et coques : Contributions à l'énergie élastique (cf. [R3.07.03])	
MEMBRANE	Énergie élastique en membrane
FLEXION	Énergie élastique en flexion
CISAILLE	Énergie élastique en cisaillement
COUPL_MF	Énergie élastique de couplage membrane-flexion

Remarque :

En non-linéaire (STAT_NON_LINE, DYNA_NON_LINE,...) les composantes CISAILLE et COUPL_MF sont nulles.

| 'ENEL_ELGA'
| 'ENEL_ELNO'
| 'ENEL_NOEU'

Calcul de la densité d'énergie élastique.

Liste des composantes du champ :

TOTALE	Énergie élastique
Composantes supplémentaires pour les plaques et coques : Contributions à l'énergie élastique (cf. [R3.07.03])	
MEMBRANE	Énergie élastique en membrane
FLEXION	Énergie élastique en flexion
CISAILLE	Énergie élastique en cisaillement
COUPL_MF	Énergie élastique de couplage membrane-flexion

Remarque :

En non-linéaire (STAT_NON_LINE, DYNA_NON_LINE,...) les composantes CISAILLE et COUPL_MF sont nulles.

| 'ENTR_ELEM'

Calcul de l'énergie élastique modifiée de traction. En mécanique de la rupture, il se peut que l'on est besoin d'estimer une énergie élastique dite de traction, ainsi l'idée consiste à calculer une énergie élastique modifiée, permettant d'annihiler la participation de la compression sphérique et de la compression suivant chaque directions propres de déformation. Ainsi l'énergie élastique devient:

$$E_{el}^{traction} = \frac{\lambda}{2} H\left(tr\left(\epsilon\right)\right) tr\left(\epsilon\right)^2 + \mu \sum_{i=1}^{3} H\left(\epsilon_i\right) \epsilon_i^2$$

où H représente la fonction Heaviside,
ϵ représente le tenseur des déformations élastiques,
ϵ_i représente les déformations élastique principales.

Liste des composantes du champ :

TOTALE	Énergie élastique modifiée de traction

Remarque :

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

Pour l'instant, uniquement valide en petites déformations (DEFORMATION = PETIT ou DEFORMATION = PETIT_REAC).

| 'EPOT_ELEM '

Calcul de l'énergie potentielle de déformation, à partir du champ de déplacement u et du champs de température T :

Liste des composantes du champ :

TOTALE	Énergie potentielle
Composantes supplémentaires pour les plaques et coques :	
MEMBRANE FLEXION	Contributions à l'énergie potentielle (cf. [R3.07.03])
Composantes supplémentaires pour les poutres droites :	
TRAC_COM TORSION FLEX_Y FLEX_Z	Contributions à l'énergie potentielle (cf. [R3.08.01])
Composantes supplémentaires pour les poutres courbes :	
PLAN_XY PLAN_XZ	Contributions à l'énergie potentielle (cf. [R3.08.01])
Composantes supplémentaires pour les discrets :	
DX DY DZ DRX DRY DRZ	Contributions à l'énergie potentielle

- pour les éléments de milieux continus 2D et 3D :

$$E_{pot}=\frac{1}{2}\int_{element}\boldsymbol{\varepsilon}(U).\mathbf{A}.\boldsymbol{\varepsilon}(U)dv-\int_{element}\boldsymbol{\varepsilon}(U).\mathbf{A}.\boldsymbol{\varepsilon}^{th}(U)dv+\frac{1}{2}\int_{element}\boldsymbol{\varepsilon}^{th}(U).\mathbf{A}.\boldsymbol{\varepsilon}^{th}(U)dv$$

- pour les éléments de poutres :

$$E_{pot}=\frac{1}{2}u^{T.}\mathbf{K}_e.u-u^{T.}\mathbf{B}^T.\mathbf{A}.\boldsymbol{\varepsilon}^{th}+\frac{1}{2}\boldsymbol{\varepsilon}^{th.}\mathbf{A}.\boldsymbol{\varepsilon}^{th}$$

- pour les éléments de plaques et coques :

$$E_{pot}=\frac{1}{2}u^T.\mathbf{K}_e.u-u^T.\mathbf{B}^T.\mathbf{A}.\boldsymbol{\varepsilon}^{th}$$

| 'ETOT_ELEM '

Calcul de l'incrément d'énergie de déformation totale entre l'instant courant et l'instant précédent.

Liste des composantes du champ :

TOTALE	Incrément d'énergie de déformation totale

| 'ETOT_ELGA'
| 'ETOT_ELNO'
| 'ETOT _NOEU'

Calcul de l'incrément de densité d'énergie de déformation totale entre l'instant courant et l'instant précédent.

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

Liste des composantes du champ :

TOTALE Incrément d'énergie de déformation totale

2.6.5 Options de calcul de critères (Opérande CRITERES)

| 'DERA_ELGA'
| 'DERA_ELNO'
| 'DERA_NOEU'

Calcul de l'indicateur local de décharge et indicateur de perte de radialité [R4.20.01].

Liste des composantes du champ :

DCHA_V	Indicateur de décharge calculé sur le tenseur déviateur des contraintes
DCHA_T	Indicateur de décharge calculé sur le tenseur total des contraintes
IND_DCHA	Indicateur permettant de savoir si la décharge resterait élastique ou s'il y aurait un risque de plastification si on utilisait un écrouissage cinématique pur
VAL_DCHA	Indique la proportion de sortie du critère dans le cas de décharge abusive
X11 X22 X33 X12 X13 X23	Composantes du tenseur cinématique utilisées pour le calcul de IND_DCHA
RADI_V	Indicateur de la variation de la direction des contraintes entre les instants t et $t+\Delta t$
ERR_RADI	Erreur η due à la discrétisation en temps, directement reliée à la rotation de la normale à la surface de charge

DCHA_V et DCHA_T expriment, dans les deux cas, la variation relative de la norme des contraintes au sens de Von Misès : $I_1=\frac{\|\boldsymbol{\sigma}(M,t+\Delta t)\|-\|\boldsymbol{\sigma}(M,t)\|}{\|\boldsymbol{\sigma}(M,t+\Delta t)\|}$, la norme étant fonction du comportement (écrouissage isotrope ou cinématique linéaire)

IND_DCHA peut prendre les valeurs suivantes :
- 0 : valeur initiale sans contrainte ;
- 1 : si charge élastique ;
- 2 : si charge plastique ;
- -1 : si décharge élastique licite (quel que soit le type d'écrouissage) ;
- -2 : si décharge abusive (on aurait plastifié avec un écrouissage cinématique).

RADI_V est donné par la relation suivante :

$$I_2=1-\frac{|\boldsymbol{\sigma}(M,t).\Delta\boldsymbol{\sigma}|}{\|\boldsymbol{\sigma}(M,t)\|\|\Delta\boldsymbol{\sigma}\|}$$

Cette quantité est nulle lorsque la radialité est conservée durant l'incrément de temps.

ERR_RADI est l'angle entre $\boldsymbol{n}^-$, la normale au critère de plasticité au début du pas de temps (instant t^-), et $\boldsymbol{n}^+$, la normale au critère de plasticité calculée à la fin du pas de temps (instant t^+) de la façon suivante :

$$\eta=\frac{1}{2}\|\Delta\boldsymbol{n}\|=\frac{1}{2}\|\boldsymbol{n}^+-\boldsymbol{n}^-\|=\left|\sin\left(\frac{\alpha}{2}\right)\right|$$

Cela fournit une mesure de l'erreur (également utilisée pour raffiner le pas de temps [U4.51.11]. Ce critère est opérationnel pour les comportements élastoplastiques de Von Mises à écrouissage isotrope, cinématique linéaire et mixte : VMIS_ISOT_LINE ,

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

VMIS_ISOT_TRAC , VMIS_ISOT_PUIS , VMIS_CINE_LINE , VMIS_ECMI_LINE , VMIS_ECMI_TRAC, et pour les comportements élasto-visco-plastiques de Chaboche : VMIS_CIN1_CHAB, VMIS_CIN2_CHAB, VMIS_CIN2_MEMO, VISC_CIN1_CHAB, VISC_CIN2_CHAB, VISC_CIN2_MEMO.

Remarque :

Le calcul de ces options nécessite de comparer les champs de contraintes aux instants t_i et t_{i+1}. Le résultat est rangé au numéro d'ordre associé à l'instant t_i.

L'indicateur de décharge est calculé par : $ID=\frac{\|\boldsymbol{\sigma}_{i+1}\|-\|\boldsymbol{\sigma}_i\|}{\|\boldsymbol{\sigma}_{i+1}\|}$.

Par défaut, le calcul se fait pour les numéros d'ordre 1 à $n-1$. Mais si on précise la liste d'instant (avec des "trous" éventuellement), le calcul ne concernera que les instants demandés mais il comparera toujours l'instant t_i avec l'instant t_{i+1} dans la liste des instants ayant servi à faire le calcul non-linéaire.

| 'ENDO_ELGA'
| 'ENDO_ELNO'
| 'ENDO_NOEU'

Calcul du dommage d à partir du tenseur des contraintes et de la déformation plastique cumulée p.

Liste des composantes du champ :

TRIAX	Taux de triaxialité
SI_ENDO	Contrainte équivalente d'endommagement de Lemaître-Sermage
COENDO	Contrainte d'endommagement de Lemaître-Sermage normalisée
DOM_LEM	Dommage de Lemaître-Sermage

Le taux de triaxialité α est donné par la relation suivante :

$$\alpha=\frac{\sigma_h}{\sigma_{eq}}$$

et la contrainte équivalente d'endommagement σ^* :

$$\sigma^*=\sigma_{eq}\sqrt{\frac{2}{3}(1+\nu)+3(1-2\nu)\alpha^2}$$

$$\mathbf{s} = \boldsymbol{\sigma}-\frac{1}{3}tr(\boldsymbol{\sigma})\cdot\mathbf{Id}$$

avec : $$\sigma_{eq} = \sqrt{\frac{3}{2}\mathbf{s}:\mathbf{s}}$$

$$\sigma_h = \frac{1}{3}tr(\boldsymbol{\sigma})$$

La cinétique d'endommagement est donnée par la loi de Lemaître-Sermage :

$$\dot{d}=\left[\frac{Y}{S}\right]^s\dot{p} \text{ si } p\geq p_{seuil} \text{ avec } Y=\frac{\sigma^{*2}}{2E(1-D)^2}$$

où S et s sont des coefficients caractéristiques du matériau et p_{seuil} le seuil d'endommagement lié à l'énergie stockée dans le matériau (si $s=1$ on obtient la loi de Lemaître classique).

| 'EPEQ_ELGA'

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

| 'EPEQ_ELNO'
| 'EPEQ_NOEU'
| 'EPMQ_ELGA'
| 'EPMQ_ELNO'
| 'EPMQ_NOEU'
| 'EPGQ_ELGA'
| 'EPGQ_ELNO'
| 'EPGQ_NOEU'

Calcul de la déformations équivalentes :

Les champs EPEQ_* sont calculées à partir des champs EPSI_* (déformations en petit déplacement), les champs EPGQ_* sont calculées à partir des champs EPSG_* (déformations de Green-Lagrange) et les champs EPMQ_* sont calculées à partir des champs EPME_* (déformations mécaniques).

Liste des composantes du champ :

INVA_2	Déformation équivalente de Von Mises
PRIN_1 PRIN_2 PRIN_3	Déformations principales, rangées dans l'ordre croissant
INVA_2SG	Déformation équivalente de Von Mises signée par la trace de ε
VECT_1_X VECT_1_Y VECT_1_Z VECT_2_X VECT_2_Y VECT_2_Z VECT_3_X VECT_3_Y VECT_3_Z	Directions principales

La déformation équivalente de Von Mises est donnée par l'expression suivante :

$$\text{INVA_2}=\sqrt{\frac{2}{3}dev(\boldsymbol{\varepsilon})_{ij}\,dev(\boldsymbol{\varepsilon})_{ji}} \text{ avec } dev(\boldsymbol{\varepsilon})_{ij}=\varepsilon_{ij}-\frac{1}{3}tr(\boldsymbol{\varepsilon})\delta_{ij}$$

Cette expression est équivalente à celle de la norme $\bar{\boldsymbol{\varepsilon}}$ du code RCC-CW.

$$\bar{\boldsymbol{\varepsilon}}=\frac{\sqrt{2}}{3}\sqrt{(\boldsymbol{\varepsilon}_{11}-\boldsymbol{\varepsilon}_{22})^2+(\boldsymbol{\varepsilon}_{22}-\boldsymbol{\varepsilon}_{33})^2+(\boldsymbol{\varepsilon}_{33}-\boldsymbol{\varepsilon}_{11})^2+6(\boldsymbol{\varepsilon}_{12}^2+\boldsymbol{\varepsilon}_{23}^2+\boldsymbol{\varepsilon}_{31}^2)}$$

Remarque :

On note que les déformations équivalentes obtenues à partir de EPSI_* et EPME_* sont identiques. En effet, la différence entre les deux tenseurs est un tenseur sphérique (déformation thermique). Comme la déformation équivalente est obtenue à partir du second invariant du déviateur, le tenseur sphérique « disparaît » lorsque l'on prend le déviateur.

| 'INDL_ELGA'

Calcul de l'indicateur de localisation, basé sur le tenseur acoustique (critère de RICE).

Liste des composantes du champ :

INDICE	Indicateur de localisation 0 si $det(N.H.N)>0$ 1 sinon, ce qui correspond a l'initiation de la localisation
DIR1	Première normale à la zone de localisation
DIR2	Deuxième normale à la zone de localisation
DIR3	Troisième normale à la zone de localisation

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

DIR4 Quatrième normale à la zone de localisation

Cet indicateur définit un état à partir duquel le problème local d'intégration du comportement perd son caractère d'unicité. Il est défini par : $det(N.\mathbf{H}.N) \leq 0$, où $\mathbf{H}$ désigne l'opérateur tangent et N la normale aux directions de localisation.

Remarque :
La méthode n'est développée que dans le cas $2D$ *et pour les lois de comportement de type* DRUCK_PRAGER *et* HUJEUX.

| 'PDIL_ELGA'

Calcul du module de rigidité de micro-dilatation.

Liste des composantes du champ :

A1_LC2 Module de rigidité de micro-dilatation

L'option PDIL_ELGA fournit dans le cadre des milieux de second gradient de dilatation la valeur du module A1_LC2, permettant de maîtriser la périodicité de la solution non triviale du problème initialement homogène [R5.04.03].

Le calcul de A1_LC2 est obtenu via l'évaluation d'une fonction dépendant de l'orientation géométrique de la bande matérielle considérée. La discrétisation angulaire imposée actuellement est égale à $0.1\,°$.

Remarque :
La méthode n'est développée que pour les lois de comportement de type DRUCK_PRAGER *et* HUJEUX.

| 'SIEQ_ELGA'
| 'SIEQ_ELNO'
| 'SIEQ_NOEU'

Calcule des contraintes équivalentes calculées à partir des champs de contrainte.

Liste des composantes du champ :

VMIS	Contrainte équivalente de Von Mises
TRESCA	Contrainte de Tresca
PRIN_1 PRIN_2 PRIN_3	Contraintes principales, rangées dans l'ordre croissant
VMIS_SG	Contrainte équivalente de Von Mises signée par la trace de σ
VECT_1_X VECT_1_Y VECT_1_Z VECT_2_X VECT_2_Y VECT_2_Z VECT_3_X VECT_3_Y VECT_3_Z	Directions principales
TRSIG	Trace de σ
TRIAX	Taux de triaxialité

La contrainte équivalente de Von Mises est donnée par l'expression suivante :

$$\text{VMIS} = \sqrt{\frac{3}{2} s_{ij} s_{ji}} \text{ avec } s_{ij} = \sigma_{ij} - \frac{1}{3} tr(\boldsymbol{\sigma}) \delta_{ij}$$

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

Le taux de triaxialité est donné par l'expression suivante :

$$\text{TRIAX} = \frac{\text{TRSIG}}{3 \times \text{VMIS}}$$

2.6.6 Option de calcul des flux hydrauliques (Opérande HYDRAULIQUE)

| 'FLHN_ELGA'

Calcul des flux hydrauliques en THM $\Phi_{ij} = \boldsymbol{M}_{ij} . \boldsymbol{\nu}$ sur les éléments de bord (2D ou 3D) à partir du vecteur flux aux nœuds.
$\boldsymbol{M}_{ij}$ est le vecteur flux hydraulique du composant $\boldsymbol{ij}$.

Liste des composantes du champ :

FH11	
FH22	
FH12	
FH21	

2.6.7 Options liées aux paramètres du calcul (Opérande PROPRIETES)

| 'MATE_ELGA'
| 'MATE_ELEM'

Récupération des valeurs des paramètres matériaux élastiques E, NU, RH O, ALPHA et thermiques LAMBDA, RHO_CP avec prise en compte de la dépendance aux variables de commande et à l'instant de calcul. Une valeur nulle est affectée aux composantes du champ si le les paramètres associés n'ont pas été renseignés dans DEFI_MATERIAU. Les coordonnées des points d'intégration sont également présentes dans le champ.

Ces options sont uniquement disponibles pour les comportements ELAS, ELAS_FO, THER, THER_FO, THER_NL.

Pour le champ MATE_ELGA, on calcule les valeurs à chaque point de Gauss, pour le champ MATE_ELEM, on calcule les valeurs au barycentre de l'élément (famille FPG1).

Pour le champ MATE_ELGA et les éléments de frontière absorbante (3D_ABSO, D_PLAN_ABSO) les coordonnées spatiales prises en compte sont celles du barycentre de l'élément (comme dans les autres options de calcul). Les valeurs à chaque point de Gauss sont donc toujours égales entre elles et aussi avec les valeurs du champ MATE_ELEM.

2.7 Opérandes pour les options thermiques

2.7.1 Opérande THERMIQUE

| 'FLUX_ELGA'
| 'FLUX_ELNO'
| 'FLUX_NOEU'

Calcul des flux de chaleur à partir de la température.

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

Liste des composantes du champ :

FLUX FLUY FLUZ	Flux de chaleur dans les trois directions de l'espace (dans le feuillet moyen pour les coques)
Composantes supplémentaires pour les coques :	
FLUX_INF FLUY_INF FLUZ_INF	Flux de chaleur dans les trois directions de l'espace en peau inférieure
FLUX_INF FLUY_INF FLUZ_INF	Flux de chaleur dans les trois directions de l'espace en peau supérieure

| 'HYDR_NOEU'

Calcul de l'hydratation.

Liste des composantes du champ :

HYDR	Hydratation

Remarque :

Le champ 'HYDR_ELNO' est calculé nativement par l'opérateur de thermique non-linéaire THER_NON_LINE pour la modélisation du béton [R7.01.12].

| 'SOUR_ELGA'

Calcul d'une source de chaleur.

Liste des composantes du champ :

SOUR	Source de chaleur

Cette source est calculée à partir d'un potentiel électrique via la loi d'Ohm. Ce potentiel électrique doit avoir été calculé par l'opérateur THER_LINEAIRE [U4.54.01] en faisant les analogies nécessaires. Cette source peut être ensuite utilisée dans un calcul thermique via le mot-clé SOUR_CALCULEE de la commande AFFE_CHAR_THER [U4.44.02].

| 'ETHE_ELEM'

Calcul de l'énergie thermique à l'équilibre à partir du champ de température T.

Liste des composantes du champ :

TOTALE	Énergie thermique

2.8 Opérandes pour les options acoustiques

2.8.1 Opérande ACOUSTIQUE

| 'INTE_ELNO'
| 'INTE_NOEU'

Calcul de l'intensité acoustique. Les définitions se trouvent dans [R4.02.01].

Liste des composantes du champ :

INTX_R	Intensité acoustique, partie réelle selon l'axe x
INTY_R	Intensité acoustique, partie réelle selon l'axe y
INTZ_R	Intensité acoustique, partie réelle selon l'axe z
INTX_I	Intensité acoustique, partie imaginaire selon l'axe x
INTY_I	Intensité acoustique, partie imaginaire selon l'axe y
INTZ_I	Intensité acoustique, partie imaginaire selon l'axe z

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

| 'PRAC_ELNO'
| 'PRAC_NOEU'

Calcul de la pression aux nœuds en partie réelle, partie imaginaire et décibels.

Liste des composantes du champ :

PRES_I	Pression acoustique, partie réelle
PRES_R	Pression acoustique, partie imaginaire
DB	Pression acoustique en décibel

| 'PRME_ELNO'

Calcul de la pression aux nœuds pour les éléments FLUIDE.

Liste des composantes du champ :

DB	Pression acoustique en décibel

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

2.9 Opérande pour les forces et les réactions nodales

2.9.1 Opérande FORCE

| 'FORC_NODA'

Calcul des forces nodales généralisées à partir des contraintes aux points de Gauss. Liste des composantes du champ :

DX DY DZ	Forces nodales
Composantes supplémentaires pour les éléments de structure :	
DRX DRY DRZ	Forces nodales (moments)

Les forces nodales correspondent au sens éléments finis aux forces intérieures participant aux équations d'équilibre. Le calcul des forces nodales généralisées $\mathbf{F}_K$ se fait de la façon suivante :

$$\sum_K \mathbf{F}_K . \mathbf{v}_K = \int_\Omega \boldsymbol{\sigma} . \boldsymbol{\varepsilon}(\mathbf{v}_K) d\Omega = \sum_K \int_K \boldsymbol{\sigma}^K . \boldsymbol{\varepsilon}(\mathbf{v}_K) dK = \sum_K \int_K \boldsymbol{\sigma}^K . \mathbf{B} . \mathbf{v}_K dK \quad (1)$$

Avec :

- $\boldsymbol{\sigma}^K$ contraintes aux points de Gauss de l'élément K ;
- $\mathbf{B}$ l'opérateur éléments finis de déformations généralisées (matrice reliant les déformations du 1er ordre aux déplacements généralisés) ;
- $\mathbf{v}_K$ déplacement élémentaire unitaire généralisé.

D'où :

$$\mathbf{F}_K = \left\{ \int_K {}^t\mathbf{B} . \boldsymbol{\sigma}^K dK \right\} \quad (2)$$

La dimension des forces nodales est duale de celle des $\mathbf{v}_K$ pour donner un travail (en Joules). Pour les éléments massifs (3D, 2D et barres), les FORC_NODA en général ont la dimension d'une force. Il s'agit d'un champ sur les nœuds du maillage où la valeur en un nœud est obtenue à partir des contraintes calculées sur les éléments concourants à ce nœud, ainsi leurs valeurs varient donc lorsque le maillage change. En l'absence de chargement réparti, l'équilibre impose leur nullité en un nœud intérieur, tandis qu'elles correspondent à la réaction sur les appuis où l'on impose une relation cinématique (cas d'un déplacement imposé).

Cas bi-dimensionnel

Pour les éléments axisymétriques, l'intégration en θ se fait sur un secteur de $1\,radian$. Si on veut l'intégrale de l'effort surfacique sur tout le disque il faut donc multiplier par 2π.

Pour les éléments en déformation plane, le calcul est fait sur une bande de largeur unité. Les forces nodales calculées sont donc en fait des forces par unité de longueur. Si on veut calculer les forces nodales s'exerçant sur une structure de largeur l, il faut multiplier le résultat en D_PLAN par l, à ceci près que l'hypothèse de déformation plane n'est pas valide près des deux faces. On aura donc un résultat approximatif.

Cas des éléments de structures

Pour les éléments de poutre et les éléments discrets, les contraintes aux points de Gauss sont en fait les efforts nodaux généralisés dans le repère de l'élément (obtenus par le produit de la matrice de rigidité de l'élément par le déplacement et en tenant compte des efforts d'origine thermique et des efforts répartis). Le calcul des forces nodales se fait en projetant les efforts nodaux contenus dans le champ de nom symbolique 'SIEF_ELGA' dans le repère global. La sommation ci-dessus sur les éléments s'applique ensuite. Les composantes DX, DY et DZ donnent les forces et DRX, DRY et DRZ les moments.

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

Dans le cas des coques, les composantes DX, DY et DZ donnent les FORC_NODA (de dimension d'une force) dans le repère global du maillage. Ces composantes sont construites avec les efforts normaux et tranchants dans la coque. Les composantes DRX, DRY et DRZ donnent les FORC_NODA (de dimension d'un moment) dans le repère global du maillage, construites avec les moments fléchissants dans la coque. Dans le cas de comportements homogénéisés de coque type ELAS_COQUE, il est préférable de regarder EFGE_NOEU.

Cas bi-dimensionnel de la thermo-hydro-mécanique (THM)

En (thermo-)hydro-mécanique, cf. § 8, [R7.01.10], les forces nodales généralisées associées à chaque composante correspondent aux forces mécaniques et aux flux. Si on note $Q^T\boldsymbol{\sigma}_0$ le résultat de FORC_NODA, pour les équations hydrauliques, alors pour un pas de temps Δt, on a :

$$\int_\Omega \mathbf{Q}^T\boldsymbol{\sigma}_0\cdot p^* d\Omega = -\Delta t\int_\Omega \mathbf{M}^-\cdot\nabla p^* d\Omega \tag{3}$$

Les flux sont pris à l'instant t^- à cause du θ -schéma employé, cf. [R7.01.10].
Si on note $\mathbf{q}$ le flux thermique et $\mathbf{M}_w$, $\mathbf{M}_{vp}$, $\mathbf{M}_{as}$ et $\mathbf{M}_{ad}$ les flux hydrauliques de l'eau liquide, de la vapeur, de l'air (ou tout autre composant) sec et de l'air dissous dans le liquide (ces données correspondent respectivement aux contraintes généralisées $M_1^1, M_1^2, M_2^1, M_2^2$, cf. § 2.2, [R7.01.10]), alors l'expression de FORC_NODA est la suivante :

- au degré de liberté PRE1 en saturé par exemple est associé le flux d'eau

$$-\Delta t\int_\Omega \left(\mathbf{M}_{vp}+\mathbf{M}_w\right)^-\nabla p^* d\Omega \tag{4}$$

- au degré de liberté PRE2 en non saturé est associé le flux du composant gazeux

$$-\Delta t\int_\Omega \left(\mathbf{M}_{ad}+\mathbf{M}_{as}\right)^-\cdot\nabla p^* d\Omega \tag{5}$$

- au degré de liberté TEMP est associé le flux thermique

$$-\Delta t\int_\Omega \mathbf{q}^-\cdot\nabla T^* d\Omega \tag{6}$$

| 'REAC_NODA'

Calcul des forces nodales de réactions généralisées aux nœuds, à partir des contraintes aux points de Gauss et des chargements. Liste des composantes du champ :

DX DY DZ	Réactions nodales
Composantes supplémentaires pour les éléments de structure :	
DRX DRY DRZ	Réactions nodales (moments)

Les réactions nodales généralisées correspondent au sens éléments finis aux forces exercées sur les appuis (conditions aux limites) participant aux équations d'équilibre.

Cas de la statique

En statique, pour les concepts résultat de type evol_elas, mult_elas, fourier_elas ou evol_noli, le calcul des réactions nodales généralisées $\mathbf{R}_K$ se fait par :

$$\sum_K \mathbf{R}_K.\mathbf{v}_K = \sum_K \mathbf{F}_K.\mathbf{v}_K - \mathbf{L}(\mathbf{v}_K) \tag{7}$$

Avec :

- $\sum_K \mathbf{F}_K\cdot\mathbf{v}_K$ l'expression des forces nodales (option FORC_NODA , voir (1) et (2)) ;

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

- $\mathbf{v}_K$ déplacement élémentaire unitaire généralisé.

Et on a :

$$\mathbf{L}(\mathbf{v}_K)=\int_\Omega \mathbf{f}\cdot\mathbf{v}_K\,d\,\Omega+\int_\Gamma \mathbf{F}_s\cdot\mathbf{v}_K\,d\,\Gamma+\sum_i \mathbf{F}_i\cdot\mathbf{v}_{Ki} \tag{8}$$

Avec :

- $\mathbf{f}$ sont les forces volumiques ;
- $\mathbf{F}_s$ les forces généralisées surfaciques ;
- $\mathbf{F}_i$ les forces généralisées ponctuelles au nœud i.

Le vecteur des réactions nodales généralisées sur l'élément K s'obtient donc à partir des forces nodales généralisées :

$$\mathbf{R}_K=\mathbf{F}_K-\int_K \mathbf{f}\,dK-\int_{\partial K}\mathbf{F}\,\partial K-\sum_i \mathbf{F}_{iK} \tag{9}$$

autrement dit, on retranche aux forces nodales les forces extérieures généralisées appliquées à l'élément K.
Remarque : à noter que le chargement de température ne figure pas dans les forces extérieures : il intervient dans l'expression des contraintes via la loi de comportement.

Cas de la dynamique transitoire non-linéaire

Pour les concepts résultat de type `evol_noli` issus de calculs dynamiques traansitoires non-linéaires, pour obtenir les réactions nodales, il convient d'ôter de surcroît les efforts d'inertie (accélération) et d'amortissement liés à la vitesse :

$$\sum_K \mathbf{R}_K.\mathbf{v}_K=\sum_K \mathbf{F}_K.\mathbf{v}_K+(\mathbf{M}\ddot{\mathbf{u}})\cdot\mathbf{v}_K-\mathbf{L}(\mathbf{v}_K) \tag{10}$$

Où :

- $\sum_K \mathbf{F}_K\cdot\mathbf{v}_K$ l'expression des forces nodales (option `FORC_NODA`, voir (1) et (2)) ;
- $\mathbf{v}_K$ déplacement élémentaire unitaire généralisé.
- $\mathbf{M}$ est la matrice de masse ;
- $\ddot{\mathbf{u}}$ le champ d'accélération généralisée ;
- $\mathbf{L}$ le vecteur des forces extérieures généralisées appliquées (équation (8)).

Remarques :

- les contributions de l'amortissement liées directement à la vitesse sur les réactions nodales sont négligées ;
- le calcul des efforts d'inertie requiert la présence du champ d'accélération `ACCE` dans la structure de données `resultat`. Dans le cas d'une structure de type `evol_noli`, il n'est pas possible pour le code de vérifier sa présence, il convient donc de prendre garde.

Cas de la dynamique modale

Pour les concepts résultat de type `mode_meca` (issus de calculs modaux) la formule est :

$$\sum_K \mathbf{R}_K.\mathbf{v}_K=\int_\Omega \boldsymbol{\sigma}(\boldsymbol{\varepsilon}(\mathbf{u})).\varepsilon(\mathbf{v}_K)\,d\,\Omega-\omega^2(\mathbf{M}\cdot\mathbf{u})\cdot\mathbf{v}_K \tag{11}$$

Où :

- $\boldsymbol{\sigma}(\boldsymbol{\varepsilon}(\mathbf{u}))$ sont les contraintes modales généralisées ;
- $\mathbf{M}$ est la matrice de masse ;
- ω la pulsation propre ;
- $\mathbf{u}$ le champ de déplacement généralisé du mode ;
- $\mathbf{v}_K$ déplacement élémentaire unitaire généralisé.

Cas de la dynamique transitoire linéaire

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

Pour les concepts résultat de type dyna_trans issus de calculs dynamiques transitoires linéaires (DYNA_VIBRA/TYPE_CALCUL='TRAN') ou de type dyna_harmo issus de calculs harmoniques (DYNA_VIBRA/TYPE_CALCUL='HARM') l'expression des réactions nodales généralisées s'obtient par :

$$\sum_K \mathbf{R}_K.\mathbf{v}_K = \sum_K \mathbf{F}_K.\mathbf{v}_K + (\mathbf{M}\ddot{\mathbf{u}})\cdot\mathbf{v}_K - \mathbf{L}(\mathbf{v}_K) \quad (12)$$

Où :

- $\sum_K \mathbf{F}_K.\mathbf{v}_K$ l'expression des forces nodales (option FORC_NODA , voir (1) et (2)) ;
- $\mathbf{v}_K$ déplacement élémentaire unitaire généralisé.
- $\mathbf{M}$ est la matrice de masse ;
- $\ddot{\mathbf{u}}$ le champ d'accélération généralisée ;
- $\mathbf{L}$ le vecteur des forces extérieures généralisées appliquées (équation (8)).

Utilisation de GROUP_MA dans FORC_NODA et REAC_NODA

Si le mot clé GROUP_MA est renseigné, $\mathbf{F}_K$ (option FORC_NODA) est calculé uniquement sur les éléments demandés puis assemblé. Le résultat est différent d'un calcul global sur tout le domaine puis réduit aux éléments demandés. La méthode implantée permet de mesurer la réaction d'un morceau de modèle sur un autre.

Attention ! Le calcul de 'REAC_NODA' sur un sous-ensemble du modèle (via le mot clé GROUP_MA) doit être fait avec prudence. L'exemple 4 ci-dessous illustre ce type de calcul.

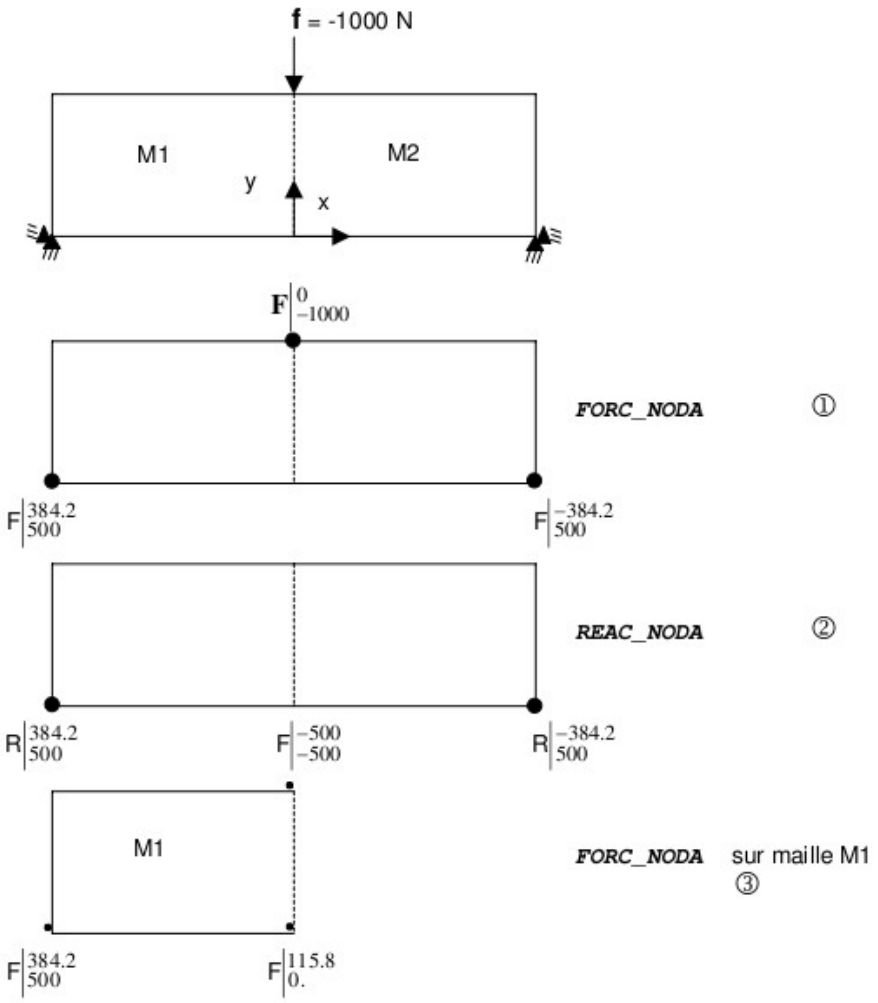
Pour les résultats de type dyna_harmo , le 'REAC_NODA' se calcule uniquement sur le modèle en entier. Le calcul sur un sous-ensemble du modèle peut être réalisé manuellement en effectuant un calcul de 'FORC_NODA' sur le groupe de mailles en question puis en retranchant la force extérieure sur les résultats obtenus.

Remarques :

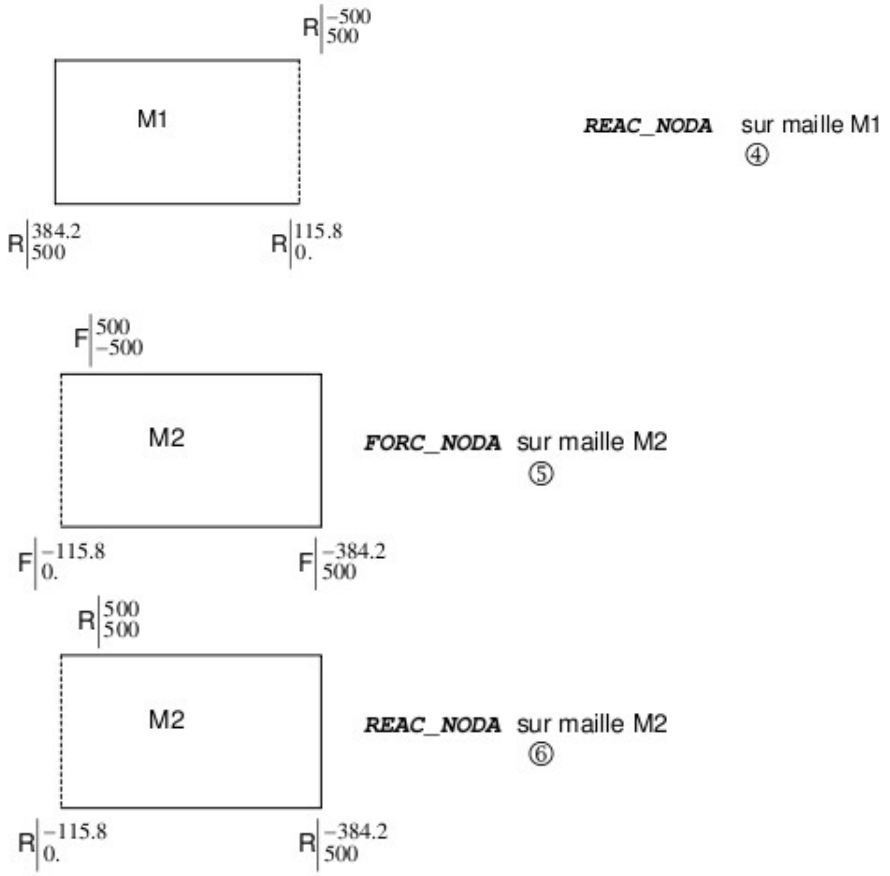
- Les réactions nodales sont nulles en tout point intérieur du modèle et ne sont pas nulles a priori en un point du bord soumis à une condition aux limites cinématique (ou à des efforts de contact) ;
- L e fait de négliger l'apport de l' ***amortissement en dynamique*** peut créer un léger écart par rapport au résultat exact ;
- Seule la résultante des forces ou des réactions nodales sur un groupe de nœuds a un sens physique (ce groupe doit correspondre à au moins un élément du modèle, par exemple un bord soumis à une condition aux limites). Elle peut être calculée avec POST_RELEVE_T [U4.81.21] . Ponctuellement, le champ FORC_NODA ou REAC_NODA ne doit pas être interprété car la valeur en un nœud est directement liée à la finesse du maillage. De plus le signe de ces forces sur les nœuds d'un même élément peut être contre-intuitif alors qu'il est parfaitement en accord avec la théorie des éléments finis (par exemple sur des mailles QUAD8 situées à l'interface d'une zone en compression pure, les signes des forces nodales aux sommets et aux milieux sont opposés) ;
- Le calcul de REAC_NODA tient compte des chargements répartis sur les poutres. Dès lors que vous faites varier ce chargement sur un calcul non-linéaire (changement de EXCIT d'un pas de temps à l'autre), le calcul de REAC_NODA est interdit. Il est nécessaire de diviser son post-traitement en « paquets » de numéros d'ordre, pour lesquels le chargement est constant (c'est-à-dire qu'il utilise le même concept AFFE_CHAR_MECA dans EXCIT).
- Certains chargements ne sont pas pris en compte lors du calcul de cette option, il s'agit de : INTE_ELEC, FORCE_SOL et RELA_CINE_BP.

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

2.9.2 Exemple 1 : structure chargée avec force nodale (2 éléments QUAD4)



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



Sur cet exemple, les réactions aux nœuds (2) sont bien égales aux forces nodales (1) moins le chargement. Elles représentent les réactions aux appuis de la structure.

Si on restreint le calcul à la maille $M1$, les forces (3) aux nœuds appartenant à la frontière entre $M1$ et $M2$ sont différentes. Elles représentent la réaction du modèle formé de $M1$ au modèle formé de $M2$. À noter que le chargement nodal est divisé par deux car les deux mailles y contribuent. Les réactions nodales (4) sont encore égales aux forces nodales moins le chargement.

Sur le calcul restreint à la maille $M2$, les forces nodales (5) suivant OX sont de signe contraire au calcul restreint à la maille $M1$, illustrant le principe de l'action et la réaction.

2.9.3 Exemple 2 : structure avec chargement de température

Données :

$E=1.10^{9}\,\mathrm{Pa}$
$\nu=0.3$
$\alpha=1.10^{-6}$

Résultats :

$F_{y}=-3.410^{4}\,N$
$F_{1x}=7.8\,10^{3}\,N$
$F_{2x}=-1.2\,10^{3}\,N$

Sur cet exemple, les forces nodales et les réactions nodales coïncident car le seul chargement est un chargement température.

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

Si on restreint le calcul à la maille $M2$, les forces suivant OY restent les mêmes mais sont différentes suivant OX.

2.9.4 Exemple 3 : structure sous chargement réparti (poutre)

On considère une structure de type poutre encastrée et soumise à un chargement de pesanteur sur sa moitié supérieure.

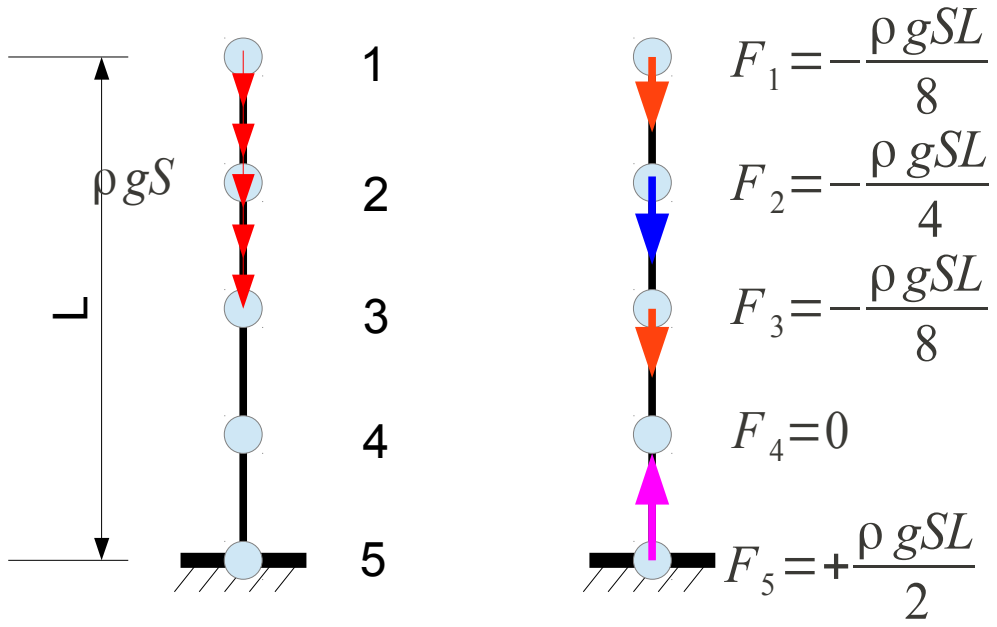


Figure 2.9.4-1: structure sous chargement réparti (gauche), FORC_NODA (droite)

Sur ce type de structure, si l'on restreint le calcul des forces et des réactions nodales à une sous-partie des éléments, FORC_NODA et REAC_NODA ne donneront pas le même résultat sur l'interface entre la partie isolée et le reste de la structure comme le montre la figure 2.9.4-2 (sur la force F_3).

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

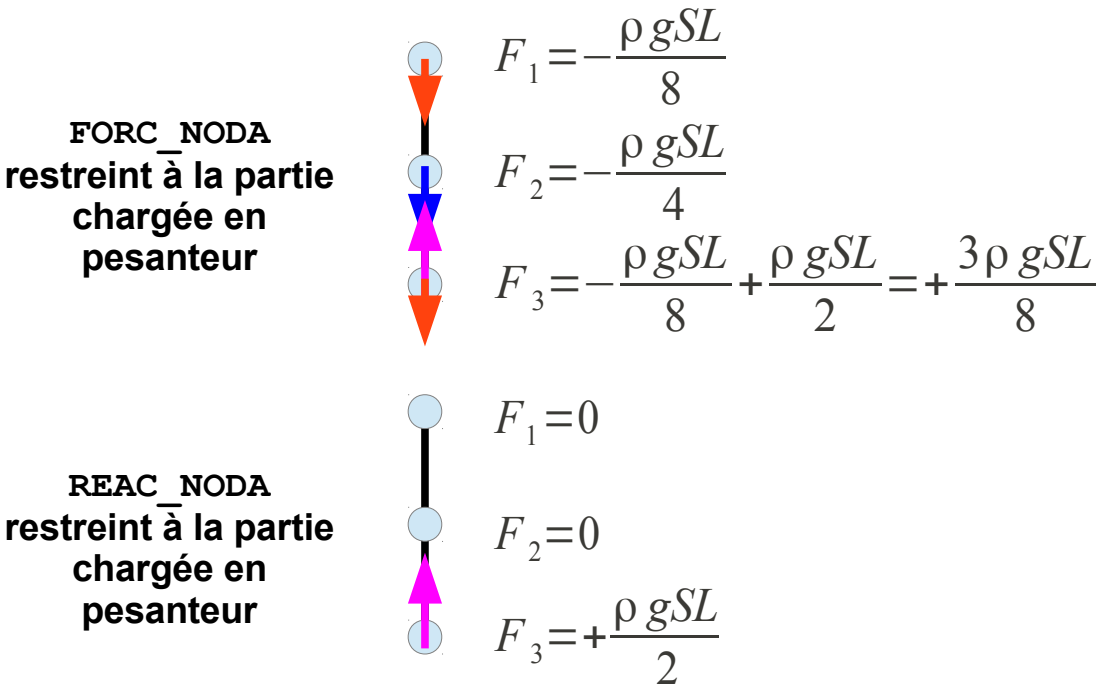


Figure 2.9.4-2: isolement d'une partie de la structure

2.9.5 Exemple 4 : calcul des réactions d'appui au pied d'un barrage

Dans cet exemple, on schématise (très grossièrement !) un barrage en 2D. Le barrage est représenté par un triangle DEF posé sur une fondation représentée par un rectangle ABCG (voir schéma ci-dessous).

On aimerait calculer les réactions d'appui du barrage sur sa fondation.

Les chargements sont :
- La pesanteur (qui s'applique au barrage et à sa fondation)
- Le chargement de pression du à la retenue d'eau (côté amont) appliqué sur les bords CD et DE.
- La fondation est encastrée sur AB.

Sur les illustrations suivantes, on a représenté la norme du champ REAC_NODA lorsqu'on utilise le mot clé GROUP_MA de différentes façons :
- Illustration 1 : on n'utilise pas GROUP_MA (TOUT='OUI')
- Illustration 2 : GROUP_MA='BARRAGE'
- Illustration 3 : GROUP_MA=('BARRAGE','DE')
- Illustration 4 : GROUP_MA=('BARRAGE','DE','CD')

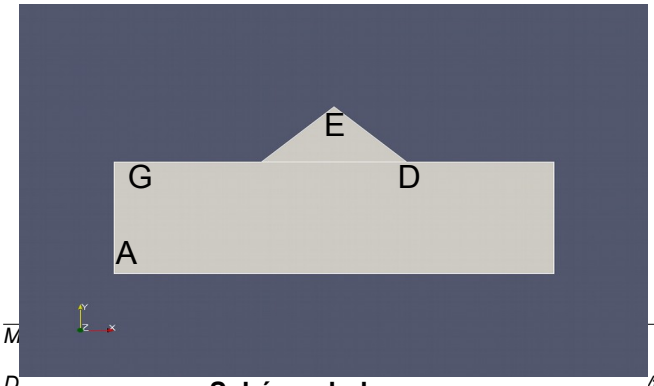


Schéma du barrage

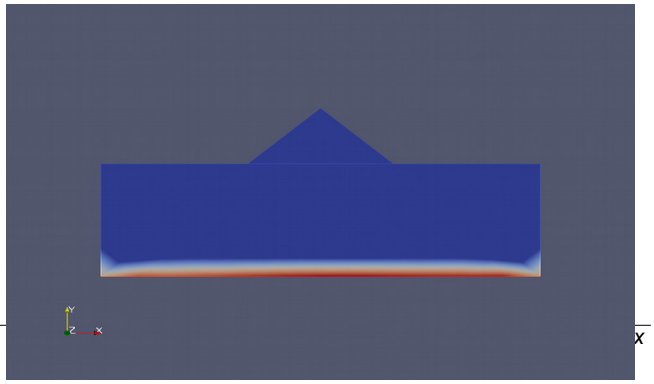


Illustration 1: REAC_NODA / TOUT='OUI'

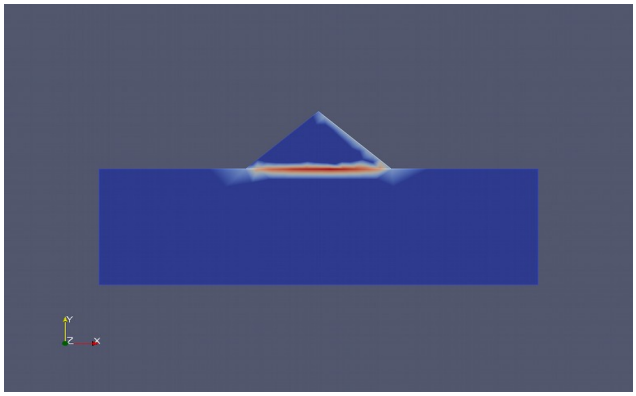


Illustration 2: REAC_NODA / GROUP_MA='BARRAGE'

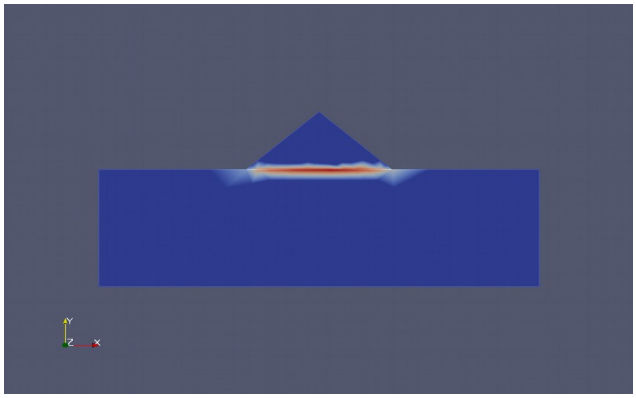


Illustration 3: REAC_NODA / GROUP_MA=('BARRAGE','DE')

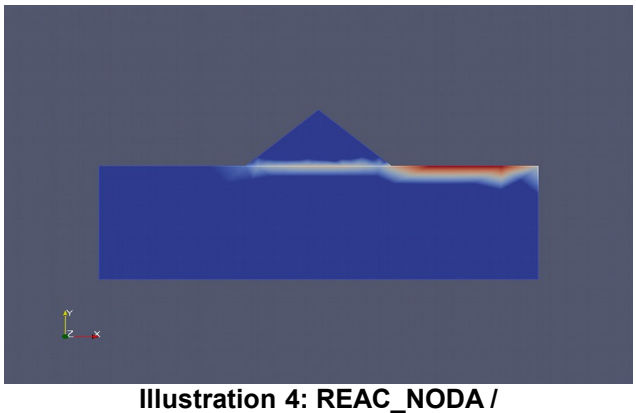


Illustration 4: REAC_NODA / GROUP_MA=('BARRAGE','DE','CD')

Commentaires :

- L'illustration 1 montre que quand on n'utilise pas le mot clé GROUP_MA, les réaction d'appui ne sont non-nulles que sur le bord du modèle où l'on impose des déplacements (ici AB).
- L'illustration 2 montre que si l'on calcule REAC_NODA sur le seul groupe BARRAGE (formé seulement d'éléments « volumiques ») des réactions non nulles apparaissent sur DF (ce que l'on cherche) mais aussi sur DE, ce qui doit nous alerter. La raison de ce comportement est que le chargement de pression sur DE n'a pas été calculé, car les éléments de bord qui calculent ce chargement ne font pas partie du groupe BARRAGE.
- L'illustration 3 montre que si l'on calcule REAC_NODA sur les groupes BARRAGE et DE, les réactions sont celles que l'on cherche : elles ne sont non nulles que sur DF. Remarque : l'image semble montrer que des réactions non nulles existent sur une partie de CD et FG, mais c'est une illusion due à la visualisation : les forces nodales en D et F « bavent » sur les éléments adjacents.
- L'illustration 4 montre qu'il ne faut pas ajouter trop de groupes de bord (ici CD). On voit que des réactions d'appuis non nulles apparaissent sur CD. Dans ce cas, la réaction sur le point D est fausse.

Conclusion

> *Si l'on veut calculer les réactions d'appui du barrage sur sa fondation, il faut demander le calcul de REAC_NODA en précisant exactement **toutes** les mailles du barrage et **toutes** les mailles de son bord soumis à un chargement réparti.*

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

Il est vivement conseillé de visualiser les réactions calculées pour vérifier qu'elles ne sont non nulles que sur l'interface entre le barrage et sa fondation.

2.10 Calcul d'un champ utilisateur

Le mot-clé facteur CHAM_UTIL permet de calculer des champs quelconques, dits « utilisateur » en raison du nom qui leur sera affecté dans le concept résultat.
Il peut y avoir plusieurs occurrences de CHAM_UTIL afin d'enchaîner le calcul de plusieurs champs.
Le traitement étant effectué à la fin de la commande CALC_CHAMP, les champs calculés par les mots-clés précédents (CONTRAINTE, DEFORMATION...) sont disponibles ici.
Soit on demande le calcul d'un critère prédéfini, soit on applique une ou plusieurs formules pour calculer un autre champ.

2.10.1 Opérande NOM_CHAM

Il s'agit du champ à partir duquel on fait les calculs. Le champ produit aura le même type : ELGA, ELNO ou NOEU.

2.10.2 Opérande CRITERE

Demande le calcul d'un critère prédéfini. Les critères sont (le paragraphe 2.6.5 fournit les expressions de chaque critère) :
- VMIS (pour les champs de contraintes),
- INVA_2 (pour les champs de déformations),
- TRACE (pour les champs de contraintes ou de déformations).

Chacun de ces critères produit une composante (nommée X1).
Un des intérêts est de pouvoir calculer INVA_2 de n'importe quel champ de déformations.

2.10.3 Opérande FORMULE

Ceci permet de calculer n'importe quelle expression fonction des composantes du champ fourni pour NOM_CHAM.
Le champ produit contiendra autant de composantes que de formules fournies : à la première formule correspondra la composante X1, à la deuxième X2, etc. Jusqu'à 30 composantes peuvent être ainsi créées.
Le champ en sortie est systématiquement prolongé à zéro sur les autres composantes et là où le champ en entrée n'est pas défini.

Des exemples de formules permettant de retrouver les critères VMIS et INVA_2 peuvent être trouvés dans la deuxième partie du test sslv104a.

2.10.4 Opérande NORME

Demande le calcul d'une norme prédéfinie. Les normes sont :
- L2 : norme L_2 d'un tenseur
- FROBENIUS : norme de Frobenius d'un tenseur

La norme calculée est une quantité intégrée (au sens faible) sur un élément Ω_e . Pour un tenseur $\boldsymbol{A}$, la norme L_2 vaut :

$$N_{L_2} = \int_{\Omega_e} \sqrt{(\boldsymbol{A} : \boldsymbol{A})} . d\Omega_e \tag{13}$$

La norme de Frobenius sur les composantes A_{ij} d'un tenseur $\boldsymbol{A}$:

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

$$N_F = \int_{\Omega_e} \sum_{i,j} |A_{ij}| . |A_{ij}| . d\Omega_e \quad (14)$$

Chacune de ces normes produit une composante (nommée X1) sur un élément. Il s'agit donc d'un champ ELEM. Il est à noter que la différence entre les deux normes vient des termes croisées (hors diagonale).

2.10.5 Opérande NUME_CHAM_RESU

Le champ produit doit être rangé, de manière unique, dans le concept résultat. Les champs « utilisateur » sont donc numérotés en utilisant NUME_CHAM_RESU et le type du champ.
Le nom du champ sera donc du type UT01_ELGA, UT22_NOEU, etc.

2.10.6 Exemple de calcul d'un champ utilisateur

Produit le champ UT02_ELGA à deux composantes. X1 est la trace de SIGM_ELGA (comparable à la composante TRSIG de SIEQ_ELGA) et X2 est la contrainte équivalente de Von Mises (composante VMIS de SIEQ_ELGA).

```
fTrace = FORMULE(NOM_PARA=('SIXX', 'SIYY', 'SIZZ'),
                 VALE="""SIXX+SIYY+SIZZ""")

fVonMis = FORMULE(NOM_PARA=('SIXX', 'SIYY', 'SIZZ', 'SIXY', 'SIXZ', 'SIYZ'),
                  VALE="""sqrt(3./2. * (SIXX**2 + SIYY**2 + SIZZ**2
                            + 2*SIXY**2 + 2*SIXZ**2 + 2*SIYZ**2 )
                            - 1./2. * fTrace(SIXX, SIYY, SIZZ)**2)""")

RES = CALC_CHAMP(reuse=RES,
                 RESULTAT=RES,
                 CHAM_UTIL=_F(NOM_CHAM='SIGM_ELGA',
                              FORMULE=(fTrace, fVonMis),
                              NUME_CHAM_RESU=2,),)
```

2.11 Opérande pour les sous-points

2.11.1 Opérande SOUS_POINT

' DEPL _ELGA'
Calcul des déplacements aux sous-points.
Pour visualiser un CHAMP_ELGA sur la structure "déformée", un champ DEPL_ELGA est calculé à partir du champ DEPL (aux nœuds) à l'aide des fonctions de forme de l'élément.
Pour les éléments à sous-points, la connaissance des fonctions de forme ne suffit pas, il faut également connaître la position des sous-points ainsi que les équations cinématiques reliant les déplacements de l'élément support aux déplacements des sous-points. Ces informations sont seulement disponibles via le CARA_ELEM.
Le champ DEPL_ELGA a pour composantes DX, DY, DZ.

' T EMP _ELGA'
Calcul de la température aux sous-points. Cette option permet de connaître la température qui est utilisée lors d'un calcul mécanique dans la loi de comportement.
Pour les éléments à sous-points, il est nécessaire de connaître la position des sous-points. Ces informations sont seulement disponibles via le CARA_ELEM.
Le champ TEMP_ELGA a pour composante TEMP.

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

2.12 Opérande TITRE

◊ TITRE = titre
Titre que l'on veut donner au résultat de la commande [U4.02.01].

2.13 Performance des calculs : parallélisme en espace ou en temps ?

◊ PARALLELISME_TEMPS = 'OUI'/'NON' [DEFAUT] (actif uniquement pour les options REAC/FORC_NODA et ***_NOEU).

Par défaut, lorsque le parallélisme MPI est activé[1] (cf. [U2.08.06]), les options de calcul bénéficient généralement ici d'un **parallélisme dit « en espace »**. C'est-à-dire que les calculs sont distribués sur les processus parallèles par paquets de mailles. Un processus MPI va gérer, pour tous les pas de temps, les calculs élémentaires requis par un groupe de mailles donné.
Mais, outre que ce parallélisme est peu efficient pour les options impliquant directement peu ou pas de calculs élémentaires (telles les options REAC/FORC_NODA et ***_NOEU), il suscite d'autre part des communications MPI importantes et répétées[2] .

Pour les options pré-citées, on lui préfère donc le **parallélisme dit « en temps »** qui, lui, distribue, pour toutes les mailles, les calculs (élémentaires ou non) requis par un paquet de pas de temps. Les communications sont donc moins coûteuses[3] et le parallélisme est étendu à des portions de code au-delà des calculs élémentaires.
Avec ce parallélisme, les accélérations peuvent aller de X5 à X50 suivant la configuration des calculs.
L'activation de ce parallélisme est effectué dans un CALC_CHAMP donné[4] et elle concerne toutes les options de calcul susceptibles d'en bénéficier : REAC/FORC_NODA et ***_NOEU et toutes les options de calculs sous-jacentes[5] .
Sur un CALC_CHAMP coûteux, comportant des calculs REAC/FORC_NODA ou ***_NOEU sur de longs transitoires et/ou sur de gros modèles, on conseille donc d'activer ce parallélisme en temps : PARALLELISME_TEMPS='OUI'. Et, contrairement à celui en espace, il n'a pas besoin d'une granularité minimum[6] pour être efficace. Dès la dizaine de pas de temps par processus MPI, l'accélération procurée par le parallélisme en temps peut être notable[7].

Le type de parallélisme actif (en espace, en temps ou aucun) est précisé, option par option, dans le fichier de message. Pour plus de détails on pourra consulter la notice d'utilisation du parallélisme [U2.08.06].

Remarque s sur le parallélisme :

- *Le parallélisme en temps, lorsqu'il est activé dans l'opérateur, concerne tous les traitements informatiques susceptibles d'en bénéficier. Pour ce faire, on « débranche » en sous-main et ponctuellement, option par option , le parallélisme en espace. Celui-ci est ensuite réactivé, « comme si de rien n'était », dans les options suivantes si celles-ci ne sont pas, à leur tour, concernées par le parallélisme en temps. Ainsi l'appel*

```
CALC_CHAMP(EFGE_ELGA, REAC_NODA, ETOT_ELGA, EPSI_NOEU,
    PARALLELISME_TEMPS='OUI')
```

  activera le parallélisme en espace pour les options n°1 et n°3 et celui en temps pour les autres . Il est équivalent, d'un point de vue parallélisme, à la séquence d'appels :

```
CALC_CHAMP(EFGE_ELGA, ETOT_ELGA)
CALC_CHAMP(REAC_NODA, EPSI_NOEU, PARALLELISME_TEMPS='OUI')
```

- *Pour simplifier la programmation[8] et ne pas chercher inutilement des gains mineurs, lorsque le nombre de pas de temps n'est pas un multiple du nombre de processus MPI, le reliquat de pas temps est effectué par tous les processus. Et bien sûr, aucune communication n'est organisée entre ceux-ci.*

1 Version mpi du code + menu d'Astk Options/mpi_nbcpu>1.
2 Sur toutes les composantes du champs et à chaque pas de temps.
3 Sur uniquement des vecteurs et à chaque mpi_nbcpu pas de temps.
4 Elle n'est donc plus active dans un CALC_CHAMP suivant. Si nécessaire, elle devra être réactivée.
5 A ppelées si nécessaire, en sous-main, par ces options de calculs.
6 Quelques dizaines de milliers de ddls par processus MPI pour le parallélisme en espace.
7 Evidemment, d'autant plus que le calcul unitaire d'un pas de temps est coûteux.
8 Donc la maintenance et l'évolutivité de l'opérateur.

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

- *Dans certaines situations, le parallélisme en temps ne fonctionne pas. Suivant les cas, le calcul s'arrête ou émet une alarme : option non encore couverte (CHAM_UTIL), contexte de calcul variant au cours du transitoire (MODELE…), calcul impossible en séquentiel, pas assez de pas de temps...*

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