

Manuel d'Utilisation**Fascicule U4.7- : Opérations sur les résultats et les champs****Document : U4.72.05**

Opérateur PROJ_CHAMP

1 But

Projeter les champs aux nœuds d'une structure de données de type `evol_xxx` sur un autre maillage. Cette commande peut servir par exemple à transférer sur un maillage "mécanique", le résultat d'un calcul thermique réalisé sur un maillage "thermique" différent. On peut également poursuivre un calcul thermique (ou mécanique) sur un autre maillage (plus ou moins raffiné).

La commande permet de définir des "zones" (géométriques) que l'on projette les unes sur les autres ce qui permet de résoudre le problème d'une discontinuité voulue du champ projeté (par exemple le long des lèvres d'une fissure).

De manière plus anecdotique La commande permet également de projeter les champs par éléments "ELNO" et "ELEM" (voir §3.7) mais cette possibilité est de moindre intérêt. De même, les méthodes (déconseillées) 'NUAGE_DEG_0/1' permettent de créer des `cham_no_*` "isolés"

Produit une structure de données `evol_*` (ou exceptionnellement `cham_no_*`)

2 Syntaxe

```

resu = PROJ_CHAMP (
#   Utilisation des fonctions de forme du 1er maillage (méthode fortement
#   conseillée) :
/   METHODE=      'ELEM' ,                                [DEFAULT]
    ♦ RESULTAT =   evol ,                                  / [evol_ther]
                                                    / [evol_elas]
                                                    / [evol_noli]
                                                    / [evol_char]
    ♦ MODELE_1 =    mo1 ,                                  [modele]
    ♦ MODELE_2 =    mo2 ,
    ◇ #   Sélection des noms des champs
      /   TOUT_CHAM = 'OUI' ,                                [DEFAULT]
      /   NOM_CHAM  = l_noch ,                                [l_Kn]
    ◇ #   Sélection des numéros d'ordre
      /   TOUT_ORDRE = 'OUI' ,                                [DEFAULT]
      /   NUME_ORDRE = l_nuor ,                                [l_I]
      /   ♦ /   INST  = l_inst ,                                [l_R]
              /   FREQ = l_freq ,                                [l_R]
              /   LIST_INST = l_inst ,                          [listr8]
              /   LIST_FREQ = l_freq ,                          [listr8]
          ◇ |   PRECISION = /   prec ,                          [DEFAULT]
              /   1.0E-3 ,
              |   CRITERE  = /   'RELATIF' ,                    [DEFAULT]
              /   'ABSOLU' ,
    ◇   DISTANCE_MAX = d_max ,                                [R]

#   "Lissage" d'un nuage de points (méthode fortement déconseillée !)
/   METHODE= /   'NUAGE_DEG_1' ,
              /   'NUAGE_DEG_0' ,
    ♦ CHAM_NO      = chno1 ,                                [cham_no_*]
    ♦ CHAM_NO_REFE = chno2 ,                                [cham_no_*]

    ◇ VIS_A_VIS = (_F(
        ♦ |   TOUT_1      = 'OUI' ,
          |   GROUP_MA_1  = lgma1 ,                                [l_gr_maille]
          |   MAILLE_1    = lmail1 ,                                [l_maille]
          |   GROUP_NO_1  = lgno1 ,                                [l_gr_noeud]
          |   NOEUD_1     = lnoe1 ,                                [l_noeud]
        ♦ |   TOUT_2      = 'OUI' ,
          |   GROUP_MA_2  = lgma2 ,                                [l_gr_maille]
          |   MAILLE_2    = lmail2 ,                                [l_maille]
          |   GROUP_NO_2  = lgno2 ,                                [l_gr_noeud]
          |   NOEUD_2     = lnoe2 ,                                [l_noeud]
        ),),
    ◇ SENSIBILITE = listpara ,                                [l_para_sensi]
    ◇ TITRE =      titr ,                                      [l_Kn]
    ) ;

```

Type du résultat de l'opérateur :

si on utilise le mot clé CHAM_NO, resu est un CHAM_NO de la même grandeur que chno1,
si on utilise le mot clé RESULTAT, resu est un evol_* de même type que evol.

3 Opérandes

3.1 Opérande METHODE

Deux familles de méthode de projection des nœuds d'un maillage sur un autre maillage sont disponibles.

La première famille (`ELEM`) est classique : pour calculer la valeur sur un nœud du nouveau maillage, on cherche dans quel élément de l'ancien maillage se trouve ce nœud, puis on interpole la valeur à l'aide des fonctions de forme de cet élément. Lorsque le nœud n'est pas dans un des éléments de l'ancien maillage (en dehors du domaine géométrique modélisé), la méthode met en relation le nœud et le point de l'élément le plus proche et interpole la valeur à l'aide des fonctions de forme de l'élément. Il y a donc (par défaut) un "prolongement" du champ à l'extérieur du maillage initial. L'opérande `DISTANCE_MAX` permet de modifier ce comportement.

La deuxième famille (`NUAGE_DEG_1/0`) (**fortement déconseillée**) utilise la notion de nuages de points, en oubliant les éléments finis présents dans les modèles. Elle est plus générale que la famille (`ELEM`) car elle permet de projeter un champ aux nœuds sans que ces nœuds soient portés par des éléments finis (par exemple un ensemble de capteurs). En revanche, cette famille trouble souvent les utilisateurs familiers de la méthode des éléments finis car elle est purement géométrique et ne voit pas les frontières matérielles ni les trous : la valeur calculée sur un nœud du nouveau maillage dépend de **tous** les nœuds qui lui sont proches. Dans les zones de fort gradient, les valeurs peuvent être obtenues avec une forte extrapolation, ce qui est rarement très "physique".

3.1.1 Limitations actuelles (version 7.4)

3.1.1.1 méthode `ELEM`

La méthode `ELEM` est disponible pour les éléments "isoparamétriques" 2D et 3D ainsi que pour les surfaces en 3D. On ne peut l'utiliser que sur des concepts de type `evol_xxxx` (mais une généralisation serait facile). Les champs projetés doivent être réels (ou complexes).

Plus précisément : le maillage sur lequel on projette peut être quelconque car on ne se sert que de ses nœuds (leurs positions) ; en revanche, les éléments finis qui serviront à l'interpolation seront tous du même type (3D volumique, 2D surfacique ou 3D surfacique). Le choix 3D / 2D / surface en 3D est fait par le programme dès le début de la commande lorsque l'on n'utilise pas le mot clé `VIS_A_VIS`, ou au début de chaque occurrence du mot clé `VIS_A_VIS` quand on l'utilise :

si le maillage "1" est 3D (utilisation de la coordonnée "z") :

 s'il contient des mailles volumiques (hexaèdres, tétraèdres, ...), on utilise la procédure "3D".

 sinon on utilise la procédure "3D surfacique"

si le maillage "1" est 2D (utilisation des seules coordonnées "x" et "y") :

 on utilise la procédure "2D"

Pour en finir avec les précautions d'usage de la méthode '`ELEM`', il faut signaler qu'il a été observé (mais pas encore expliqué) qu'avec les maillages quadratiques et bords courbes (typiquement les cylindres), le champ de déplacement projeté (bien qu'apparemment assez "lisse") pouvait donner lieu à des "gradients" (déformations ou contraintes) assez chahutés. Il semble donc prudent, quand c'est possible, de projeter les champs de gradient plutôt que de les recalculer à partir du champ "primal" projeté.

3.1.1.2 méthode `NUAGE_DEG_1/0`

La méthode `NUAGE_DEG_1/0` ne peut traiter que les champs isolés (et pas les `SD_RESULTAT`).

3.2 Opérandes CHAM_NO et CHAM_NO_REFE

La commande projette les valeurs du `cham_no` `chno1` sur les nœuds du maillage sous-jacent au "modèle de champ" : `chno2`.

Elle crée alors le `cham_no` résultat sur le même modèle que `chno2` (les mêmes composantes portées par les mêmes nœuds) mais avec des valeurs obtenues par "interpolation" des valeurs de `chno1`.

On peut a priori projeter un `cham_no` de n'importe quelle grandeur réelle ou complexe, en 2D ou en 3D. Le champ à projeter et le champ modèle doivent être du même type (`DEPL_R`, `TEMP_R`, ...).

Le champ résultat sera alors du même type.

3.3 Mot clé VIS_A_VIS

Ce mot clé facteur facultatif permet de projeter le champ "par morceaux". Par défaut, c'est l'ensemble du champ qui est projeté.

Ce mot clé permet de projeter des champs discontinus ou de résoudre le problème de la projection de modèles complexes (voir [§3.3.3]).

S'il est utilisé ce mot clé permet à l'utilisateur de projeter un champ a priori discontinu (sur une ligne ou une surface) et de conserver cette discontinuité pour le champ projeté :

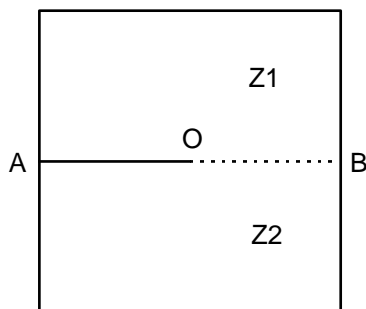


Figure 3.3-a

Par exemple, soit la structure fissurée de la [Figure 3.3-a]. Le champ de déplacement est discontinu sur la fissure : les lèvres supérieure et inférieure bien que confondues géométriquement possèdent chacune leur propre champ de déplacement.

Supposons que l'on dispose de 2 maillages (M et M') différents de cette structure et que pour chacun de ces maillages, les zones Z1 et Z2 soient représentées par deux `GROUP_MA` nommés Z1 et Z2.

On pourra écrire :

```
VIS_A_VIS = ( _F( (GROUP_MA_1 = 'Z1' , GROUP_MA_2 = 'Z1' ) ,
                  _F( GROUP_MA_1 = 'Z2' , GROUP_MA_2 = 'Z2' ) ) ,
```

Les valeurs du champ projeté sur la lèvre supérieure (appartenant à Z1) ne tiendront compte que des valeurs du champ initial sur Z1. De même les valeurs obtenues sur la lèvre inférieure ne dépendent que du champ initial sur Z2.

Remarque :

Sur la ligne OB, les nœuds ne sont pas dédoublés leurs valeurs seront calculées deux fois (pour les 2 occurrences de VIS_A_VIS). C'est la dernière occurrence qui imposera sa valeur.

3.3.1 Opérandes TOUT_1 / GROUP_MA_1 / MAILLE_1 / GROUP_NO_1 / NOEUD_1

Ces opérandes servent à définir l'ensemble des mailles ('ELEM') ou des nœuds ('NUAGE ...') à prendre en compte dans le maillage initial.

3.3.2 Opérandes TOUT_2 / GROUP_MA_2 / MAILLE_2 / GROUP_NO_2 / NOEUD_2

Ces opérandes servent à définir l'ensemble des nœuds où l'on évalue le champ.

3.3.3 Utilisation du mot clé VIS_A_VIS pour projeter les modèles complexes

Le mot clé VIS_A_VIS permet par exemple de résoudre le problème de la projection d'un champ de température calculé sur un modèle contenant des éléments 3D et des éléments de coque (un solide dont la peau interne serait recouverte d'un "liner" maillé en éléments de coque).

Le problème est le suivant : les éléments 3D portent sur leurs nœuds le seul degré de liberté TEMP, alors que les éléments de coque (thermiques) portent les 3 degrés de liberté : TEMP, TEMP_INF et TEMP_SUP. Si on ne prend pas de précautions et que l'on projette tout le modèle d'un seul coup, les degrés de liberté TEMP_INF et TEMP_SUP seront "perdus" lors de la projection. En effet, comme le maillage initial contient des éléments volumique, c'est la procédure "3D" qui est appliquée : pour chaque nœud du maillage 2, on cherche une maille volumique qui contienne ce nœud (ou qui en est proche). Une fois cette maille trouvée, on interpole les degrés de liberté portés par **tous** les nœuds de cette maille. Pratiquement, aucune maille 3D ne peut porter sur tous ses nœuds les degrés de liberté TEMP_INF et TEMP_SUP ; ceux-ci sont donc perdus.

Pour résoudre ce problème il faut faire quelque chose comme :

```
evo2= PROJ_CHAMP( RESULTAT=evol, ...  
                  VIS_A_VIS=( _F(GROUP_MA_1: 'VOLU' , GROUP_MA_2='VOLU' , ),  
                              _F(GROUP_MA_1: 'LINER' , GROUP_MA_2='LINER' , ), )
```

De cette manière, les mailles de 'LINER' du maillage 2 seront affectées par les valeurs portées par les mailles de 'LINER' du maillage 1 et elles porteront les mêmes degrés de liberté.

3.4 Opérande RESULTAT

RESULTAT = evol,

Nom du concept résultat de type evol_xxx que l'on veut projeter.

Remarque :

Pour un evol_noli, le concept résultat ne contiendra (par défaut [§3.7]) que les champs de déplacements. Pour calculer les champs de contraintes et de variables internes correspondants, il faut utiliser la commande STAT_NON_LINE en utilisant le mot clé PREDICTION : 'DEPL_CALCULE'.

3.5 Opérande MODELE_1

MODELE_1 = mo1,

Nom du modèle associé au concept evol_xxx initial (evol)

3.6 Opérande MODELE_2

MODELE_2 = mo2,

Nom du modèle associé au concept evol_xxx résultat (resu). Ce modèle doit avoir été créé au préalable.

3.7 Sélection des noms des champs

Les mots clés TOUT_CHAMP = 'OUI' ou NOM_CHAM = l_noch permettent de choisir quels sont les champs de la SD RESULTAT que l'on veut projeter (par défaut tous les champs aux noeuds). On peut aussi projeter les champs par éléments "ELNO" et "ELEM" (mais pas "ELGA") ; pour cela, il faut explicitement donner leurs noms via le mot clé NOM_CHAM.

3.7.1 Remarques sur la projection des champs par éléments

Cette possibilité a été introduite initialement pour permettre de projeter les champs de pression calculés par le Code Saturne (constants par éléments) sur la peau d'un maillage mécanique Aster. Ce développement étant général, il est disponible pour tous les champs "ELEM" et "ELNO" mais son utilisation peut parfois surprendre l'utilisateur.

Comme pour les champs aux nœuds, les champs par éléments projetés du concept `evol` dans le concept `resu` porteront les mêmes noms (par exemple : 'SIGM_ELNO_DEPL'). Les champs projetés (`resu`) sont donc de même "nature" que les champs "origine" (`evol`) : "ELEM" ou "ELNO".

Que fait-on numériquement ?

- Champs de type `ELNO` : pour chaque maille du maillage 2, on parcourt les nœuds de cette maille et on calcule la valeur de chaque nœud comme on le ferait pour un champ aux nœuds. Le résultat de cette projection est donc (par construction), un champ `ELNO` qui est continu entre les éléments.
- Champs de type `ELEM` (constant par maille) : la valeur portée par une maille du maillage 2 est obtenue par moyenne arithmétique des valeurs portées par ses nœuds (calculées comme pour un champ `ELNO`).

La projection de champs par éléments n'a de sens que si les éléments projetés les uns sur les autres sont de même "type" (coque, poutre, iso-paramétriques). En effet, si on projetait par exemple un champ 3D de 'SIGM_ELNO_DEPL' sur un modèle linéique formé d'éléments de barre, on ne saurait pas stocker les valeurs de contraintes SIXX, ... sur ces éléments (qui ne connaissent que les efforts généralisés : N, MX, ...)

3.8 Opérande `DISTANCE_MAX`

Pour projeter des nœuds d'un maillage `MA1` sur un maillage `MA2`, la méthode (`ELEM`) cherche dans quel élément du maillage `MA1` se trouve un nœud de `MA2`, puis interpole la valeur à l'aide des fonctions de forme de l'élément. Lorsque qu'un nœud de `MA2` n'est dans aucun élément du maillage `MA1`, la méthode met en relation le nœud et l'élément le plus proche et interpole la valeur à l'aide des fonctions de forme de l'élément et cela même si le nœud est "loin".

Si l'on souhaite que lorsque le nœud qui n'est dans aucun des éléments du maillage `MA1`, ne soit pas mis en relation avec l'élément le plus proche, on utilise l'opérande `DISTANCE_MAX`. Cet opérande permet de donner la distance maximale que l'on autorise entre le nœud et l'élément le plus proche. Si le nœud ne répond pas au critère de proximité non seulement le champ ne sera pas projeté mais la structure des données ne sera pas créée pour ce nœud (i.e. le nœud ne portera aucune composante dans le champ projeté).

3.9 Sélection des numéros d'ordre

cf. [U4.71.00].

3.10 Opérande `SENSIBILITE`

`SENSIBILITE = listpara,`

Ce mot-clé est suivi d'une liste de paramètres sensibles. Il précise que l'on se s'intéresse pas au résultat en lui-même, mais à la dérivée du résultat par rapport aux paramètres sensibles.

3.11 Opérande `TITRE`

`TITRE = titr,`

Titre que l'on veut donner au concept résultat.

4 Exemple

Calculs thermique et mécanique sur deux maillages différents.

Remarque :

Le modèle mécanique peut contenir (par exemple) des éléments de câble non présents dans le modèle thermique.

```
ma1=LIRE_MALLAGE      (... ) ;
mo1=AFFE_MODELE(MAILLAGE=ma1,AFFE=_F(TOUT='OUI',PHENOMENE='THERMIQUE',...));
...
evo1  = THER_LINEAIRE (MODELE = mo1,      ... );

ma2 = LIRE_MALLAGE (...);      # maillage plus raffiné
mo2 = AFFE_MODELE(MAILLAGE= ma2, ... 'THERMIQUE', ...);

evo2  = PROJ_CHAMP (METHODE= 'ELEM' ,   RESULTAT= evo1,   NOM_CHAM= 'TEMP' ,
                    MODELE_1= mo1,      MODELE_2= mo2,    );

ch2 = AFFE_CHAR_MECA( TEMP_CALCULEE= evo2, ...
```

Page laissée intentionnellement blanche.