

Structure s de données sd_corresp_2_mailla et sd_1_corresp_2_mailla

Résumé :

Table des matières

1 Généralités.....	3
1.1 Pour la méthode "COLLOCATION".....	3
1.2 Pour les méthodes "NUAGE_DEG_0/1".....	3
2 Arborescence.....	3
2.1 Contenu des objets JEVEUX.....	5
2.2 Exemple d'utilisation (méthode 'COLLOCATION').....	6

1 Généralités

La `sd_corresp_2_mailla` est la structure de données utilisée pour la projection des champs dans la commande `PROJ_CHAMP`. Elle stocke les informations géométriques permettant d'associer les entités géométriques des 2 maillages à projeter l'un sur l'autre. La SD est très différente en fonction de la méthode de projection retenue.

Pour l'instant cette SD est utilisée dans la commande `PROJ_CHAMP` ainsi que dans le mot clé `LIAISON_MAIL` d'`AFFE_CHAR_MECA`.

La structure `sd_l_corresp_2_mailla` est constituée de 2 `sd_corresp_2_mailla`. La première est utilisée pour toutes les méthodes de `PROJ_CHAMP` sauf `ECLA_PG`, la deuxième est utilisée pour la méthode `ECLA_PG`. Cette structure a été créée dans le but de projeter des résultats possédant à la fois des champs `ELGA` et non `ELGA`. On a ainsi les deux `sd_corresp_2_mailla` disponibles selon le type de champ à projeter. La structure `sd_l_corresp_2_mailla` est utilisée uniquement dans `PROJ_CHAMP`.

1.1 Pour la méthode "COLLOCATION"

Soit 2 maillages `M1` et `M2` qui occupent une même région de l'espace. La `sd_corresp_2_mailla` correspondant au couple (`M1`, `M2`) est la structure de données qui exprime la correspondance géométrique entre les nœuds du maillage `M2` et les mailles du maillage `M1`.

1.2 Pour les méthodes "NUAGE_DEG_0/1"

Soit 2 maillages `M1` et `M2` qui occupent une même région de l'espace. La `sd_corresp_2_mailla` correspondant au couple (`M1`, `M2`) contient les listes des nœuds à mettre en vis à vis.

2 Arborescence

```
sd_corresp_2_mailla (K16)      ::= record

    soit nno2 le nombre de nœuds du maillage M2

    (o)   '.PJXX_K1'           :   OBJ   S   V   K24       dim=5
    (o)   '.$VIDE'             :   / sd_corresp2_elem
                                / sd_corresp2_nuage

sd_corresp_2_elem (K16)       ::= record

    soit nno2 le nombre de nœuds du maillage M2

    (o)   '.PJEF_NB'           :   OBJ   S   V   I         dim=nno2
        '.PJEF_M1'             :   OBJ   S   V   I         dim=nno2
        '.PJEF_CF'             :   OBJ   S   V   R         dim=3*nno2
        '.PJEF_CO'             :   OBJ   S   V   R
        '.PJEF_NU'             :   OBJ   S   V   I

    (f)   '.PJEF_TR'           :   OBJ   S   V   I         dim=nno2
        '.PJEF_AM'             :   OBJ   S   V   I         dim=nno2

    % si METHODE='ECLA_PG' :
    (f)   '.PJEF_MP'           :   OBJ   S   V   K8         dim=1
        '.PJEF_EL'             :   OBJ   S   V   I

    % si METHODE='SOUS_POINT_MATER' :
        '.PJEF_SP'             :   OBJ   S   V   I
```

```
sd_corresp_2_nuage (K16)      ::= record
```

```
  (o) '.PJNG_I1'      :   OBJ   S   V   I  
      '.PJNG_I2'      :   OBJ   S   V   I
```

2.1 Contenu des objets JEVEUX

' .PJXX_K1 '

' .PJXX_K1' (1) : nom du maillage 1 : M1
' .PJXX_K1' (2) : nom du maillage 2 : M2
' .PJXX_K1' (3) : méthode de projection :
'COLLOCATION'/'NUAGE_DEG_0'/'NUAGE_DEG_1'
' .PJXX_K1' (4) : nom d'un cham_no "modèle" (si méthode= 'NUAGE_DEG_0/1')
' .PJXX_K1' (5) : inutilisé

' .PJEF_M1 '

' .PJEF_M1' (ino2) : imal : numéro de la maille de m1 qui doit
servir à l'interpolation du nœud ino2 de m2

' .PJEF_NB '

' .PJEF_NB' (ino2) : nombre de nœuds de imal

' .PJEF_CO '

' .PJEF_CO' (3*(ino2-1)+1) : "ksi" de ino2 dans imal
' .PJEF_CO' (3*(ino2-1)+2) : "eta" de ino2 dans imal
' .PJEF_CO' (3*(ino2-1)+3) : "dzeta" de ino2 dans imal

Remarques :

*pjef_co ne sert que pour les liaisons 3d/coque et 3d/poutre
Les mailles SEG n'utilisent que ksi,
Les TRIA et QUAD n'utilisent que ksi et eta*

' .PJEF_NU '

' .PJEF_NU' : contient les numéros des nœuds de m1 servant à l'interpolation des nœuds de
m2 (mis bout à bout)

' .PJEF_CF '

' .PJEF_CF' : contient les coefficients pour les nœuds de m1 servant à l'interpolation des
nœuds de m2 (mis bout à bout)

' .PJEF_TR' et ' .PJEF_AM'

Les objets .PJEF_TR et PJEF_AM n'existent que dans les *corresp_2_mailla* temporaires
faites de TR3 (TR3 = SEG2, TRIA3 ou TETRA4)

' .PJEF_TR' (ino2) : numéro du TR3 associé au nœud ino2

' .PJEF_AM' (ino2) :

1 -> le nœud ino2 est inclus dans une maille de m1 on peut alors utiliser la routine
reereg.f pour améliorer la précision de l'interpolation.
0 -> sinon

' .PJEF_MP' et ' .PJEF_EL'

Les objets .PJEF_MP et PJEF_EL n'existent que pour METHODE = 'ECLA_PG'

' .PJEF_MP' :

(1) : nom du maillage 1 « prime »

```
' .PJEF_EL' : long >= 2*nb_PG(modèle « 2 »)
  Pour chaque point de Gauss du modèle « 2 », on stocke :
  V(2*(ipg-1)+1) = ima2 : numéro de la maille contenant ipg
  V(2*(ipg-1)+2) = kpg : numéro du point de Gauss dans ima2
```

' .PJEF_SP'

L'objet .PJEF_SP n'existe que pour METHODE = 'SOUS_POINT_MATER'.

```
' .PJEF_SP' : long = 3*nb_SP_MAT(modèle « 2 »)
  Pour chaque sous-point et chaque point de la liste de famille MATER du
  modèle « 2 », on stocke :
  V(3*(ispma-1)+1) = ima2 : numéro de la maille
  V(3*(ispma-1)+2) = kpg : numéro du point de Gauss
  V(3*(ispma-1)+3) = sspg : numéro du sous-point
```

' .PJNG_I1' et ' .PJNG_I2'

Ces 2 vecteurs d'entiers stockent les numéros des nœuds en correspondance via le mot clé facteur VIS_A_VIS.

Soit NOCC le nombre d'occurrences de VIS_A_VIS :

```
' .PJNG_I1' (1) : NOCC
' .PJNG_I1' (2) : nb1 : nombre de nœuds de MA1 pour l'occurrence 1 de VIS_A_VIS
' .PJNG_I1' (3) : nb2 : nombre de nœuds de MA1 pour l'occurrence 2 de VIS_A_VIS
...
' .PJNG_I1' (1+NOCC) : nbnocc
' .PJNG_I1' (1+NOCC+1, ..., 1+NOCC+nb1) : numéros des nœuds de MA1 pour
l'occurrence 1 de VIS_A_VIS
...
```

Remarques :

Si le mot clé VIS_A_VIS n'est pas utilisé : .PJNG_I1(1)=0

L'objet .PJNG_I2 a la même organisation que .PJNG_I1, mais il renseigne sur les nœuds du maillage MA2.

2.2 Exemple d'utilisation (méthode 'COLLOCATION')

On veut savoir comment interpoler INO2 à partir du maillage M1

```
soit nbno1=' .PJEF_NB'(INO2)
soit decal= somme pour ino<INO2 de ' .PJEF_NB'(ino)
valeur(INO2)=0
do i=1,nbno1
  nuno1=' .PJEF_NU' (decal+i)
  coefer=' .PJEF_CF' (decal+i)
  valeur(INO2)=valeur(INO2)+coefer*valeur(nuno1)
enddo
```