Macro-commande MACR_ADAP_MAIL

1 But

Adapter un maillage avec le logiciel HOMARD.

Cette opération est possible pour un maillage formé de mailles-points, de segments, de triangles, de quadrangles, de tétraèdres, d'hexaèdres, de pentaèdres. Un champ pilotant l'adaptation aura éventuellement été calculé. En fonction de sa valeur maille par maille ou nœud par nœud, ou en fonction d'une directive géométrique, le logiciel HOMARD modifiera le maillage. Il est également possible d'interpoler des champs, de l'ancien maillage vers le nouveau.

On peut enchaîner calcul et adaptation au fur et à mesure dans un processus d'amélioration du calcul. Ce processus peut avoir lieu en une seule passe, ou scindé en plusieurs étapes par une POURSUITE.

Le logiciel HOMARD est présenté sur le site : http://www.code-aster.org/outils/homard

On y trouve une description de la technique utilisée pour modifier les maillages ainsi que des exemples.
Pour en savoir plus sur HOMARD, on peut se référer aux documents cités dans la bibliographie.

Toute référence externe à HOMARD doit se faire avec :

G. Nicolas and T. Fouquet, « *Adaptive Mesh Refinement for Conformal Hexahedral Meshes* », Finite Elements in Analysis and Design, Vol. 67, pp. 1-12, 2013, doi:10.1016/j.finel.2012.11.008

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

Table des Matières

1 But....1
2 Syntaxe....5
3 Description d'une adaptation de maillage....11
3.1 Schéma général d'une adaptation....11
3.2 Fonctionnement de la macro-commande....11
3.3 Quelques commentaires....11
4 Opérandes....13
4.1 Opérande ADAPTATION....13
4.2 Opérande MAILLAGE_N....13
4.3 Opérande MAILLAGE_NP1....14
4.4 Opérande MAILLAGE_NP1_ANNEXE....14
4.5 Opérande IDENTIFICATION....14
4.6 Choix du champ de pilotage de l'adaptation....14
4.6.1 Opérande RESULTAT_N....15
4.6.2 Opérande CHAM_GD....15
4.6.3 Opérande NOM_CMP....15
4.6.4 Sélection du paramètre temporel du champ....15
4.6.5 Opérande USAGE_CMP....15
4.6.6 Opérande USAGE_CHAMP....16
4.6.7 Opérande ADAP_INIT....16
4.7 Opérande CRIT_RAFF_xxxx....17
4.7.1 Opérande CRIT_RAFF_PE....17
4.7.2 Opérande CRIT_RAFF_ABS....17
4.7.3 Opérande CRIT_RAFF_REL....17
4.7.4 Opérande CRIT_RAFF_MS....17
4.8 Opérande CRIT_DERA_xxxx....18
4.8.1 Opérande CRIT_DERA_PE....18
4.8.2 Opérande CRIT_DERA_ABS....18
4.8.3 Opérande CRIT_DERA_REL....18
4.8.4 Opérande CRIT_DERA_MS....18
4.9 Mot clé ZONE....18
4.9.1 Type de la zone....18
4.9.2 Usage de la zone....19
4.9.3 Cas du rectangle....19
4.9.4 Cas de la boîte parallélépipédique....19
4.9.5 Cas du disque....19
4.9.6 Cas de la sphère....20
4.9.7 Cas du cylindre....20

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

4.9.8 Cas d'un disque percé....20
4.9.9 Cas du tuyau....21
4.10 Opérandes GROUP_MA / GROUPE_NO....21
4.11 Opérande DIAM_MIN....22
4.12 Opérande NIVE_MAX....22
4.13 Opérande NIVE_MIN....22
4.14 Mot clé MAILLAGE_FRONTIERE....22
4.14.1 Opérande GROUP_MA_FRONT....24
4.15 Mot clé FRONTIERE_ANALYTIQUE....25
4.15.1 Nom de la frontière....25
4.15.2 Type de la frontière....25
4.15.3 Opérande GROUP_MA....25
4.15.4 Cas de la sphère....25
4.15.5 Cas du cylindre....26
4.15.6 Cas du cône défini par un angle....26
4.15.7 Cas du cône défini par des rayons....27
4.15.8 Cas du tore....27
4.16 Mot clé MAJ_CHAM....28
4.16.1 Opérande RESULTAT....28
4.16.2 Opérande CHAM_GD....28
4.16.3 Opérande NOM_CMP....28
4.16.4 Sélection du paramètre temporel du champ à mettre à jour....28
4.16.5 Opérande TYPE_MAJ....28
4.16.6 Opérande CHAM_MAJ....29
4.16.7 Opérande TYPE_CHAM....29
4.17 Mot clé ADD_CHAM....30
4.17.1 Opérande CHAM_GD....30
4.17.2 Opérande CHAM_CAT....30
4.18 Opérande MODELE....30
4.19 Opérande DEGRE....31
4.20 Opérande NOMBRE....31
4.21 Opérande QUALITE....31
4.22 Opérande DIAMETRE....31
4.23 Opérande INTERPENETRATION....32
4.24 Opérande TAILLE....32
4.25 Opérande CONNEXITE....32
4.26 Opérande PROP_CALCUL....32
4.27 Les historiques....32
4.28 Opérande LANGUE....32
4.29 Opérande VERSION_HOMARD....33

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

4.30 Opérande LOGICIEL....33
4.31 Opérande UNITE....33
4.32 Opérande ELEMENTS_ACCEPTES....33
4.33 Opérande INFO....34
5 Exemple....35
6 Bibliographie....39

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

2 Syntaxe

```
MACR_ADAP_MAIL   (
#   choix du type d'adaptation
♦   ADAPTATION    =   /   'RAFF_DERA'
                       /   'RAFFINEMENT'
                       /   'DERAFFINEMENT'
                       /   'RAFF_DERA_ZONE'
                       /   'RAFFINEMENT_UNIFORME'
                       /   'DERAFFINEMENT_UNIFORME'
                       /   'RIEN'
                       /   'MODIFICATION'
                       /   'LECTURE'

#   le maillage à modifier
♦   MAILLAGE_N    =   man    [maillage]

#   le nouveau maillage
♦   MAILLAGE_NP1  =   co (manp1)                          [K8]

#   un maillage annexe
◊   MAILLAGE_NP1_ANNEXE =   co (manp1ann)                 [K8]

#   maillage de référence pour l'historique
◊   IDENTIFICATION   =   refmesh                          [maillage]

#   Si l'adaptation est libre, (RAFFINEMENT, DERAFFINEMENT ou RAFF_DERA), choix de la
structure contenant le champ pilotant l'adaptation :
    ♦   / RESULTAT_N = resun                              [resultat]
           ♦ NOM_CHAM = nomsymb                           [K16]
        / CHAM_GD = cham_gd_i                             [cham_gd]
    ◊   NOM_CMP = l_cmp                                   [l_K8]
     #   Sélection du paramètre temporel
        / NUME_ORDRE = ordre                              [I]
        / INST = instant                                  [R]
           ◊ | PRECISION =  / prec                        [R]
                            / 1.0E-6                      [DEFAUT]
           ◊ | CRITERE =/ 'RELATIF                        [DEFAUT]
                        / 'ABSOLU'
    ◊   USAGE_CMP =   / 'NORME_L2'                        [DEFAUT]
                      / 'ABSOLU'
                      / 'NORME_INFINIE'
                      / 'RELATIF'
    ◊   USAGE_CHAMP =/ 'MAILLE'                           [DEFAUT]
                      / 'SAUT'
    ◊   ADAP_INIT =   / 'GARDER'                          [DEFAUT]
                      / 'RAFFINER'
                      / 'DERAFFINER'

#   Finsi

#   Si l'adaptation a lieu selon des zones géométriques, (RAFF_DERA_ZONE) :
    ♦   ZONE = _F (
    #  Type de la zone
    ♦   TYPE = / 'RECTANGLE'
               / 'BOITE'
               / 'DISQUE'
               / 'SPHERE'
```


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

```
                / 'CYLINDRE'
                / 'DISQUE_PERCE'
                / 'TUYAU'


        # Usage de la zone
        ♦  USAGE =   / 'RAFFINEMENT'     [DEFAUT]
                     / 'DERAFFINEMENT'

        #   pour une boîte rectangulaire : coordonnées extrêmes
            ♦  X_MINI = x_mini         [R]
            ♦  X_MAXI = x_maxi         [R]
            ♦  Y_MINI = y_mini         [R]
            ♦  Y_MAXI = y_maxi         [R]
        #   pour une boîte parallélépipédique : coordonnées extrêmes
            ♦  X_MINI = x_mini         [R]
            ♦  X_MAXI = x_maxi         [R]
            ♦  Y_MINI = y_mini         [R]
            ♦  Y_MAXI = y_maxi         [R]
            ♦  Z_MINI = z_mini         [R]
            ♦  Z_MAXI = z_maxi         [R]
        #   pour un disque : centre et rayon
            ♦  X_CENTRE  = x_centre       [R]
            ♦  Y_CENTRE  = y_centre       [R]
            ♦  RAYON  = rayon             [R]
        #   pour une sphère : centre et rayon
            ♦  X_CENTRE  = x_centre       [R]
            ♦  Y_CENTRE  = y_centre       [R]
            ♦  Z_CENTRE  = z_centre       [R]
            ♦  RAYON  = rayon             [R]
        #   pour un cylindre : axe, base, hauteur et rayon
            ♦  X_AXE  = x_axe             [R]
            ♦  Y_AXE  = y_axe             [R]
            ♦  Z_AXE  = z_axe             [R]
            ♦  X_BASE = x_base         [R]
            ♦  Y_BASE = y_base         [R]
            ♦  Z_BASE = z_base         [R]
            ♦  HAUTEUR   = hauteur        [R]
            ♦  RAYON  = rayon             [R]
        #   pour un disque percé : centre, rayons intérieur et extérieur
            ♦  X_CENTRE  = x_centre       [R]
            ♦  Y_CENTRE  = y_centre       [R]
            ♦  RAYON_INT = rayon_int   [R]
            ♦  RAYON_EXT = rayon_ext   [R]
        #   pour un tuyau : axe, base, hauteur et rayons intérieur et extérieur
            ♦  X_AXE  = x_axe             [R]
            ♦  Y_AXE  = y_axe             [R]
            ♦  Z_AXE  = z_axe             [R]
            ♦  X_BASE = x_base         [R]
            ♦  Y_BASE = y_base         [R]
            ♦  Z_BASE = z_base         [R]
            ♦  HAUTEUR   = hauteur        [R]
            ♦  RAYON_INT = rayon_int   [R]
            ♦  RAYON_EXT = rayon_ext   [R]
        #   Finsi
            )
    #   Finsi
```


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

```
#   Si l'adaptation inclut le raffinement libre (RAFFINEMENT ou RAFF_DERA) :
    ♦  /   CRIT_RAFF_PE =      crp [R]
       /   CRIT_RAFF_REL =  crr [R]
       /   CRIT_RAFF_ABS =  cra [R]
       /   CRIT_RAFF_MS =   crms   [R]
#   Finsi


#   Si l'adaptation inclut le déraffinement libre (DERAFFINEMENT ou RAFF_DERA) :
    ♦  /   CRIT_DERA_PE =      cdp    [R]
       /   CRIT_DERA_REL =  cdr    [R]
       /   CRIT_DERA_ABS =  cda    [R]
       /   CRIT_DERA_MS =   cdms       [R]
#   Finsi

#   Si l'adaptation inclut du raffinement :
    ◊  NIVE_MAX = nivmax             [I]
    ◊  DIAM_MIN = diamin             [R]
#   Finsi

#   Si l'adaptation inclut du déraffinement :
    ◊  NIVE_MIN = nivmin             [I]
#   Finsi

#   Si l'adaptation inclut du raffinement ou du déraffinement :
    ◊  GROUP_MA = l_grma          [l_gr_maille]
    ◊  GROUP_NO = l_grno          [l_gr_nœud]
#   Finsi

#    Suivi d'une frontière maillée
◊   MAILLAGE_FRONTIERE = maf         [maillage]
    ◊  GROUP_MA_FRONT = l_grma    [l_gr_maille]

#    Suivi d'une frontière analytique
◊   FRONTIERE_ANALYTIQUE = _F  (
    # Nom de la frontière
    ♦  NOM =  nom       [K]
    ♦  GROUP_MA = l_grma          [l_gr_maille]
    # Type de la frontière
    ♦  TYPE = / 'SPHERE'
              / 'CYLINDRE'
              / 'CONE_A'
              / 'CONE_R'
              / 'TORE'
    #    pour une sphère : centre et rayon
       ♦  X_CENTRE  = x_centre       [R]
       ♦  Y_CENTRE  = y_centre       [R]
       ♦  Z_CENTRE  = z_centre       [R]
       ♦  RAYON  = rayon             [R]
    #   pour un cylindre : axe, base et rayon
       ♦  X_AXE  = x_axe             [R]
       ♦  Y_AXE  = y_axe             [R]
       ♦  Z_AXE  = z_axe             [R]
       ♦  X_CENTRE  = x_centre       [R]
       ♦  Y_CENTRE  = y_centre       [R]
       ♦  Z_CENTRE  = z_centre       [R]
       ♦  RAYON  = rayon             [R]
```


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

```
    #   pour un cône défini par son angle : axe, centre et angle
        ♦   X_AXE  = x_axe              [R]
        ♦   Y_AXE  = y_axe              [R]
        ♦   Z_AXE  = z_axe              [R]
        ♦   X_CENTRE  = x_centre        [R]
        ♦   Y_CENTRE  = y_centre        [R]
        ♦   Z_CENTRE  = z_centre        [R]
        ♦   ANGLE  = angle              [R]
    #   pour un cône défini par ses rayons : centres et rayons
        ♦   X_CENTRE  = x_centre        [R]
        ♦   Y_CENTRE  = y_centre        [R]
        ♦   Z_CENTRE  = z_centre        [R]
        ♦   RAYON  = rayon              [R]
        ♦   X_CENTRE2 = x_centre2       [R]
        ♦   Y_CENTRE2 = y_centre2       [R]
        ♦   Z_CENTRE2 = z_centre2       [R]
        ♦   RAYON2 = rayon2             [R]
    #   pour un tore : centre, axe et rayons
        ♦   X_CENTRE  = x_centre        [R]
        ♦   Y_CENTRE  = y_centre        [R]
        ♦   Z_CENTRE  = z_centre        [R]
        ♦   X_AXE  = x_axe              [R]
        ♦   Y_AXE  = y_axe              [R]
        ♦   Z_AXE  = z_axe              [R]
        ♦   RAYON  = rayon              [R]
        ♦   RAYON2 = rayon2             [R]
    #   Finsi
              )


#   Si l'adaptation est une modification, (MODIFICATION), choix du type :
    ◊   Changement de degré
        DEGRE =   / 'OUI'
                  / 'NON'           [DEFAUT]
#   Finsi


#   Mise à jour de champs sur le nouveau maillage
◊   MAJ_CHAM = _F (
    #   choix de la structure contenant le champ à mettre à jour
    ♦   / RESULTAT = resu               [resultat]
            ♦ NOM_CHAM = nomsymb        [K16]
        / CHAM_GD = cham_gd             [cham_gd]
    ◊   NOM_CMP = l_cmp             [l_K8]
     #   Sélection du paramètre temporel
        / NUME_ORDRE = ordre        [I]
        / INST = instant        [R]
            ◊ | PRECISION =  / prec [R]
                             / 1.0E-3   [DEFAUT]
            | CRITERE =    / 'RELATIF    [DEFAUT]
                           /   'ABSOLU'
    #   choix du type de mise à jour
    ◊ TYPE_MAJ = / 'AUTO'           [DEFAUT]
                 / 'ISOP2'
    # nom du champ de grandeurs qui contiendra le nouveau champ
    ♦ CHAM_MAJ = co (chpmaj)                [K8]
    #   type du champ mis à jour
    ♦ TYPE_CHAM =/ 'NOEU_TEMP_R'
```


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

```
                / 'NOEU_DEPL_R'
                / etc ...
      )
```

```
# Ajout de champs en sortie sur le nouveau maillage
◊ ADD_CHAM = _F(
  # nom du champ de grandeurs qui contiendra le nouveau champ
  ♦ CHAM_MAJ = co (chpadd)            [K8]
  #  catégorie du champ à créer
  ♦ CHAM_CAT = / 'NIVEAU'
               / 'QUALITE'
               / 'DIAMETRE'
  )

# Si l'adaptation est une lecture, (LECTURE), choix du modèle :
  ♦ MODELE = modele    [modele]
# Finsi

◊ NOMBRE =      / 'OUI'               [DEFAUT]
                / 'NON'

◊ QUALITE =     / 'NON'               [DEFAUT]
                / 'OUI'

◊ DIAMETRE =    / 'NON'               [DEFAUT]
                / 'OUI'

◊ CONNEXITE =   / 'NON'               [DEFAUT]
                / 'OUI'

◊ TAILLE =      / 'NON'               [DEFAUT]
                / 'OUI'

◊ PROP_CALCUL =/ 'NON'                [DEFAUT]
                / 'OUI'

◊ INTERPENETRATION =     / 'OUI'

◊ ELEMENTS_ACCEPTES = / 'HOMARD'      [DEFAUT]
                      /IGNORE_PYRA'

◊ LANGUE =      / 'FRANCAIS' [DEFAUT]
                / 'FRENCH'
                / 'ANGLAIS'
                / 'ENGLISH'

◊ VERSION_HOMARD =/ 'V11_10'          [DEFAUT]
                  / 'V11_N'
                  / 'V11_N_PERSO'

◊ LOGICIEL =   logiciel     [K]
```


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

```
#   Si la version est la version de développement, (V11_N, V11_N_PERSO) :
    ◊ UNITE =     unite                    [I]
#   Finsi



◊   INFO =     / 1                        [DEFAUT]
               / 2
               / 3
               / 4

)
```


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

3 Description d'une adaptation de maillage

3.1 Schéma général d'une adaptation

Le principe général d'un calcul avec adaptation de maillage est le suivant :

Phase 1 : Lecture du maillage initial, M0
Définition des matériaux
Phase 2 :
- définition du modèle, des chargements sur ce maillage M0
- calcul produisant un résultat RESU0
- calcul éventuel d'un champ pilotant le raffinement, CHAMP0

Cette phase initiale est la phase standard de tout calcul

Phase 3 : Adaptation du maillage M0. On récupère un nouveau maillage, M1
Phase 4 :
- définition du modèle, des chargements sur le maillage M1,
- calcul produisant un résultat RESU1,
- calcul éventuel d'un champ pilotant le raffinement, CHAMP1.

La phase 4 est similaire à la phase 2. La seule chose qui a changé est le maillage. De ce fait, tous les concepts en dépendant doivent être repris. Aujourd'hui, il n'y a pas de possibilité ni de réutiliser les anciens concepts, ni de les détruire automatiquement.

Ensuite, on peut poursuivre, autant de fois que l'on veut, le tandem phase 3/phase 4. Cela se fait soit en dupliquant les instructions, soit en écrivant une boucle python.

Voir la référence [bib1] pour une présentation générale de l'adaptation de maillage et de HOMARD, accompagnée d'exemples.

3.2 Fonctionnement de la macro-commande

La phase 3 réalise l'adaptation du maillage. Elle est activée par la macro-commande MACR_ADAP_MAIL, décrite dans ce document. Elle a pour argument essentiel le nom du concept du maillage courant et le nom que l'on donnera au concept du futur maillage. L'autre donnée obligatoire est le type d'adaptation que l'on souhaite : du raffinement ou du déraffinement libre, c'est-à-dire en fonction des valeurs que prend un champ sur les mailles du maillage, ou d'une zone géométrique, ou du raffinement ou du déraffinement uniforme, c'est-à-dire que toutes les mailles sont traitées de la même manière.

Les autres données dépendent ensuite des options retenues.

En complément à l'adaptation, HOMARD peut fournir sur demande des bilans sur la qualité ou le diamètre des mailles du maillage, la connexité du domaine de calcul, les tailles caractéristiques, les éléments sur-contraints ou un contrôle de la non-interpénétration des mailles. Ces renseignements s'obtiennent par l'activation des mots-clés associés. On regardera avec profit la commande MACR_INFO_MAIL [U7.03.02] qui permet d'obtenir toutes ces informations, indépendamment de tout calcul.

3.3 Quelques commentaires

Le maillage adapté contient les mêmes groupes que le maillage en entrée, avec la règle suivante : un groupe définit un même lieu géométrique dans les deux maillages.

- Utiliser un groupe de nœuds revient à définir des lieux ponctuels. Le groupe dans le maillage adapté sera la liste des mêmes nœuds, ni plus, ni moins, pour représenter les mêmes points ; seuls leurs numéros auront éventuellement changé.
- Utiliser un groupe de segments revient à définir des lignes. Le groupe dans le maillage adapté sera la liste des segments qui représentent les mêmes lignes. Selon le mode d'adaptation, ces segments seront soit les mêmes, au numéro près, soit les moitiés des segments initiaux.

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

• Utiliser un groupe de triangles et/ou de quadrangles revient à définir des surfaces. Le groupe dans le maillage adapté sera la liste des triangles et/ou quadrangles qui représentent les mêmes surfaces. Selon le mode d'adaptation, ces mailles seront soit les mêmes, au numéro près, soit les fractions des mailles 2D initiales.
• De même, utiliser un groupe de mailles 3D revient à définir des volumes. Le groupe dans le maillage adapté sera la liste des mailles 3D qui représentent les mêmes volumes. Selon le mode d'adaptation, ces mailles seront soit les mêmes, au numéro près, soit les fractions des mailles 3D initiales.

La conséquence est la suivante. Les chargements du calcul mécanique ou thermique doivent exclusivement être définis par des groupes de la dimension cohérente avec le phénomène que l'on veut modéliser.
Tout autre fonctionnement conduira à une erreur dans le calcul sur le maillage adapté. Utiliser des mailles définies par leur numéro est impossible car la numérotation va changer. Utiliser des groupes de nœuds ou de mailles de la mauvaise dimension ne décrira pas complètement le lieu.
Pour une explication plus détaillée et illustrée, regarder :
http://www.code-aster.org/outils/homard/usage/regles.fr.htm#CL

Quand on veut adapter plusieurs fois de suite un maillage, il est fondamental de bien respecter la chaîne des maillages. À la première itération, le maillage d'entrée de MACR_ADAP_MAIL est le maillage initial du cas que l'on traite. Ensuite, le maillage d'entrée d'un MACR_ADAP_MAIL doit être le maillage de sortie du MACR_ADAP_MAIL précédent. Attention : il ne suffit pas de donner un maillage qui est formellement le même, suite à une copie par exemple. Il est impératif de fournir le même concept (voir aussi IDENTIFICATION). Si on ne procède pas ainsi, on perdra l'historique de raffinement des mailles et il sera impossible de déraffiner ultérieurement. Plus grave, HOMARD n'ayant plus connaissance des découpages supplémentaires qui ont été introduits pour assurer la conformité, on sera conduit à découper des mailles en dégradant fortement leur qualité. Si cette mauvaise mise en données apparaît, une alarme est émise.

L'adaptation de maillage est possible en mode POURSUITE. Les données nécessaires à la reprise sont automatiquement archivées puis relues dans le répertoire de conservation de la base nécessaire à *Code_Aster*. Utiliser HOMARD en poursuite se fait donc de la même façon qu'utiliser *Code_Aster* en poursuite.

De manière générale, les impressions essentielles fournies par HOMARD sont insérées dans le fichier "mess" au fil de l'exécution. En cas d'erreur ou en mode d'information 3 ou 4, des impressions plus détaillées ont lieu.

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

4 Opérandes

4.1 Opérande ADAPTATION

♦ ADAPTATION = / 'RAFF_DERA'
/ 'RAFFINEMENT'
/ 'DERAFFINEMENT'
/ 'RAFF_DERA_ZONE'
/ 'RAFFINEMENT_UNIFORME'
/ 'DERAFFINEMENT_UNIFORME'
/ 'RIEN'
/ 'MODIFICATION'
/ 'LECTURE'

Cet opérande permet de définir le type d'adaptation souhaité.

En premier lieu, on trouve les modes d'adaptations qui sont pilotées par un champ. En d'autres termes, la décision de (dé) raffiner une maille se prend en fonction de la valeur d'un champ calculé auparavant sur cette maille. Le choix peut se faire entre trois variantes :
• 'RAFF_DERA' : le maillage est raffiné et déraffiné en fonction du champ. C'est l'option recommandée.
• 'RAFFINEMENT' : seule la fonction de raffinement est activée. Aucune maille ne sera déraffinée.
• 'DERAFFINEMENT' : c'est l'inverse ; seule la fonction de déraffinement est activée. Aucune maille ne sera raffinée.

En deuxième lieu, on peut décider d'adapter le maillage dans des zones géométriques définies par des boîtes. Toutes les mailles dont au moins deux nœuds sont présents dans l'une de ces boîtes seront retenues. Cela permet de faire des raffinements ou déraffinements *a priori*, sans avoir fait de calcul.
• 'RAFF_DERA_ZONE' : les mailles de chacune des boîtes définies sont raffinées ou déraffinées.

Enfin, on peut activer une adaptation uniforme d'un maillage. En d'autres termes, toutes les mailles du maillage sont traitées de la même manière. Le choix peut se faire entre trois variantes :
• 'RAFFINEMENT_UNIFORME' : toutes les mailles sont raffinées,
• 'DERAFFINEMENT_UNIFORME' : toutes les mailles sont déraffinées,
• 'RIEN' : toutes les mailles sont conservées ; le maillage est le même à la sortie qu'à l'entrée.

Remarques :

Quand on applique une option de déraffinement, on ne fait que revenir en arrière sur des raffinements antérieurs. Il faut comprendre cette option comme du dé-raffinement. En particulier, on ne pourra jamais obtenir un maillage plus grossier que le maillage initial.
Les options de raffinement ou de déraffinement peuvent ne s'appliquer que sur une partie du maillage. Cela s'obtient par l'option de filtrage GROUP_MA ou GROUP_NO.

Deux options complémentaires existent :
La première permet la modification de maillage, pour changer le degré du maillage :
• 'MODIFICATION' : le maillage est modifié globalement.
La seconde permet de lire des champs aux points de Gauss qui ont été mis à jour sur le nouveau maillage :
• 'LECTURE' : les champs aux points de Gauss sont lus.

4.2 Opérande MAILLAGE_N

♦ MAILLAGE_N = man

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

Le maillage de type [maillage] à adapter ou à modifier. Attention, l'adaptation ne peut porter que sur les mailles suivantes : mailles-points, segments, triangles, quadrangles, tétraèdres, hexaèdres ou pentaèdres. Si on fournit un maillage comportant d'autres mailles, par exemple des pyramides, deux cas de figure sont possibles : soit un arrêt en erreur, soit une adaptation sur la zone autorisée et une restitution à l'identique du reste du maillage. Le choix entre ces deux modes de fonctionnement est fait par le mot-clé ELEMENTS_NON_HOMARD.
Le maillage est en degré 1 ou 2, mais il n'est pas possible de mélanger les deux.
Dans tous les cas, la présence des mailles enrichies HEXA27 est interdite.

Quand le choix a été fait de lire des champs aux points de Gauss, on donne ici le maillage sur lequel ils se trouvent.

4.3 Opérande MAILLAGE_NP1

◊ MAILLAGE_NP1 = co (manp1)

Le nom du concept de type [maillage] qui contiendra le maillage issu de l'adaptation. Ce nom doit respecter les contraintes habituelles des noms de concept (8 caractères au maximum) et ne pas avoir déjà été utilisé.

4.4 Opérande MAILLAGE_NP1_ANNEXE

♦ MAILLAGE_NP1_ANNEXE = co (manp1ann)

Cette opérande permet de produire un maillage analogue au maillage obtenu par l'opérande MAILLAGE_NP1, mais de degré différent. C'est utile en thermo-mécanique où le calcul thermique a lieu sur le maillage en degré 1 et la mécanique sur le même maillage mais en degré 2. Ce nom doit respecter les contraintes habituelles des noms de concept (8 caractères au maximum) et ne pas avoir déjà été utilisé.

4.5 Opérande IDENTIFICATION

◊ IDENTIFICATION = refmesh

Supposons que le maillage issu d'une adaptation (M1) est ensuite légèrement modifié (en M1bis). Si on adapte ce maillage modifié, on ne pourra naturellement pas bénéficier du déraffinement.
Dans cette situation, il est possible d'indiquer que le maillage modifié (MAILLAGE_N=M1bis) peut être considéré de la même lignée que M1 en indiquant IDENTIFICATION=M1. Ainsi, l'historique de M1 pourra servir à déraffiner M1bis.
Bien sûr, il est essentiel que les maillages soient très proches et de bien vérifier le résultat de l'adaptation.

4.6 Choix du champ de pilotage de l'adaptation

Dans le cas d'une adaptation libre, le pilotage des mailles à raffiner ou déraffiner est réalisé par un champ. Ce champ est contenu soit dans une structure de résultat, soit dans un champ de grandeurs. Ce champ peut être un champ d'indicateur d'erreur au sens numérique du terme (QIRE_ELEM par exemple) mais ce n'est pas obligatoire ; n'importe quel champ peut être utilisé. On peut par exemple piloter l'adaptation par le champ des contraintes ou par un champ construit exprès comme une distance ou un critère d'endommagement. Il suffit que ce champ soit défini par son nom tel que décrit dans les documents [U4.81.01], [U4.81.02] ou [U4.81.03].

Si le champ est un champ aux nœuds, la décision de raffinement/déraffinement sera prise sur chaque arête en fonction des valeurs du champ sur ses nœuds. Si le champ est un champ constant par élément, c'est cette valeur qui pilotera le raffinement/déraffinement de la maille. Si le champ est un

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

champ aux nœuds par élément ou aux points de Gauss, l'algorithme se basera sur la valeur maximale dans la maille pour décider du raffinement/déraffinement.

4.6.1 Opérande RESULTAT_N

```
/ ◊ RESULTAT_N = resun
```

Cet opérande permet de désigner le concept de type [resultat] qui contient le champ à utiliser pour de l'adaptation libre.

4.6.1.1 Opérande NOM_CHAM

```
◊ NOM_CHAM = nomsymb
```

On précise ici quel est le champ qui est utilisé pour piloter l'adaptation.

Attention :
Le champ doit être présent dans le résultat ; s'il est absent, il n'est pas calculé d'office.

4.6.2 Opérande CHAM_GD

```
/ ◊ CHAM_GD = cham_gd_i
```

Cet opérande permet de désigner le concept de type [cham_gd] qui contient le champ à utiliser pour piloter l'adaptation libre.

4.6.3 Opérande NOM_CMP

```
◊ NOM_CMP = l_cmp
```

Nom de la composante du champ qui doit être utilisée pour piloter l'adaptation de maillage. Si plusieurs composantes sont souhaitées, donner ici la liste.
Si aucune composante n'est définie ici, la commande prendra toutes celles qui existent dans le champ transmis.

Le type de prise en compte de la ou des composantes est pilotée par USAGE_CMP.

4.6.4 Sélection du paramètre temporel du champ

Si la structure de résultat ne contient le champ requis que pour un seul numéro d'ordre, rien n'est à préciser. Ce sont les valeurs du champ à ce numéro d'ordre qui seront utilisées.
Sinon, il faut préciser de quel numéro il s'agit. Cela se fait par la désignation d'un numéro d'ordre ou d'une valeur d'instant. Se référer au document [U4.71.00] pour les détails sur ces mots-clés.

4.6.5 Opérande USAGE_CMP

```
◊ USAGE_CMP = / 'NORME_L2' [DEFAUT]
              / 'NORME_INFINIE'
              / 'ABSOLU'
              / 'RELATIF'
```

On précise ici comment traiter les différentes composantes du champ pilotant l'adaptation. On part du principe que le raffinement porte sur les grandes valeurs examinées et, symétriquement, le déraffinement porte sur les petites valeurs. Par défaut, on filtrera le raffinement et le déraffinement en examinant la norme L2 des composantes du champ sur les mailles (ou les nœuds), c'est-à-dire la racine carrée de la somme des carrés des valeurs des composantes (norme dite euclidienne).

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

Si plusieurs composantes ont été retenues, on peut choisir entre deux types de norme : soit la norme L2, choix par défaut, soit la norme infinie, c'est-à-dire la plus grande des valeurs absolues des composantes.

Si une seule composante est retenue pour piloter l'adaptation, les choix NORME_L2, NORME_INFINIE et ABSOLU sont équivalents : on examinera la valeur absolue du champ. Une alternative est possible : utiliser RELATIF permet de piloter l'adaptation avec les valeurs brutes du champ. Dans ce cas-là, pour un champ dont les valeurs sont négatives, le raffinement portant sur les valeurs maximales, ce seront les zones où la valeur est proche de 0 qui seront raffinées ; symétriquement, le déraffinement portera sur les zones où la valeur est très grande négativement.

4.6.6 Opérande USAGE_CHAMP

```
◊  USAGE_CHAMP =/ 'MAILLE'       [DEFAUT]
                / 'SAUT'
```

Par défaut, le pilotage de l'adaptation se fait par le tri des valeurs du champ transmis, maille par maille ou nœud par nœud.
Avec la variante SAUT, HOMARD on triera sur le saut du champ entre mailles, selon le procédé suivant. Pour chaque maille, HOMARD commence par calculer le maximum de l'écart absolu entre la valeur du champ sur la maille courante et sa valeur sur chacune des mailles voisines. Ce maximum est attribué à la maille courante. Ensuite, on trie les mailles sur ces écarts maximums selon les critères habituels.
En 2D, les voisins examinés sont les triangles/quadrangles qui partagent une arête avec la maille en cours.
En 3D, ce sont les mailles volumiques qui partagent une face triangulaire ou quadrangulaire avec la maille courante.
Si le champ est défini par nœud, les voisins sont les nœuds qui partagent une arête avec le nœud courant.

Remarque :

Cette option permet d'adapter aisément le maillage en se fixant comme objectif une variation régulière d'un champ d'une maille à l'autre. Ainsi, choisir le type SAUT et le champ SIEF_ELGA permet d'obtenir un maillage où s'atténueront les fortes variations de contraintes d'une maille à sa voisine.

4.6.7 Opérande ADAP_INIT

```
◊  ADAP_INIT =  /  'GARDER'        [DEFAUT]
                /  'RAFFINER'
                /  'DERAFFINER'
```

Quand le champ pilotant l'adaptation est défini sur tout le maillage, cette option est sans effet.
Dans le cas contraire, elle permet de préciser comment sont traitées les régions où ce champ n'est pas défini.
Avec la variante GARDER, les mailles des régions où l'indicateur n'est pas défini sont *a priori* gardées telles quelles. C'est l'option par défaut.
Avec la variante RAFFINER , les mailles des régions où l'indicateur n'est pas défini sont *a priori* découpées.
Avec la variante DERAFFINER , les mailles des régions où l'indicateur n'est pas défini sont *a priori* réactivées.
Évidemment, cela n'est qu'une initialisation et ces décisions peuvent évoluer pour assurer la conformité du maillage, compte-tenu des décisions apportées aux mailles voisines.

Remarque :

Cette option est particulièrement utile quand on utilise un indicateur construit à partir d'un champ. Par exemple, on utilise l'endommagement calculé dans une région donnée alors qu'ailleurs, il n'est pas défini. On pourra choisir de ne pas toucher aux autres mailles ou de tenter de les déraffiner.

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

Par exemple : dans les calculs d'excavation, des mailles sont retirées du modèle pour simuler le creusement ; ces mailles-là ne portent plus de valeur d'indicateur. Si on ne fait rien, elles restent dans le maillage. Si on choisit la variante `DERAFFINER` *, elles seront déraffinées au fur et à mesure de la progression de l'excavation.*

4.7 Opérande CRIT_RAFF_xxxx

Dans le cas d'adaptation libre impliquant du raffinement de maillage, il faut définir un critère haut de raffinement. Toutes les mailles pour lesquels le champ est supérieur à ce critère seront raffinées. Il est important de regarder a posteriori l'allure de la répartition du champ. Cela est possible grâce aux impressions réalisées par HOMARD dans le fichier `mess`. On y trouvera en particulier un tableau présentant cette répartition sous forme d'histogramme ; voir le chapitre 5 pour un exemple commenté.

Pour le choix du critère, quatre variantes sont possibles :

4.7.1 Opérande CRIT_RAFF_PE

◊ / CRIT_RAFF_PE = crp

Le critère est défini par une proportion de mailles à raffiner. C'est un nombre réel compris entre 0 et 1. Le processus est le suivant :

- calcul du nombre de mailles `n` correspondant à la proportion définie par `crp` soit `n` = `crp` x nombre total de mailles
- raffinement des `n` mailles avec la plus forte valeur du champ.

4.7.2 Opérande CRIT_RAFF_ABS

/ CRIT_RAFF_ABS = cra

Le critère est défini par une valeur absolue du champ. Toutes les mailles avec une valeur supérieure à cette valeur seront raffinées.

4.7.3 Opérande CRIT_RAFF_REL

/ CRIT_RAFF_REL = crr

Le critère est défini par une valeur relative du champ. C'est un nombre compris entre 0 et 1. Le processus est le suivant :

- calcul des valeurs minimales et maximales de l'indicateur,
- calcul de la valeur correspondant à la proportion requise : `v= vmin + crr(vmax-vmin)`,
- raffinement de toutes les mailles où le champ est supérieur à cette valeur.

4.7.4 Opérande CRIT_RAFF_MS

◊ / CRIT_RAFF_MS = crms

Le critère est défini par une valeur absolue du champ, calculée en fonction de la moyenne et de l'écart-type de ce champ. Toutes les mailles avec une valeur supérieure à cette valeur seront raffinées. Le critère vaut : moyenne + n*sigma, où `n` est le coefficient fourni, strictement positif.

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

4.8 Opérande CRIT_DERA_xxxx

Dans le cas d'adaptation libre impliquant du déraffinement, il faut définir un critère bas de déraffinement. Toutes les mailles où le champ est inférieur à ce critère seront déraffinées. Trois variantes sont possibles.

4.8.1 Opérande CRIT_DERA_PE

◊ / CRIT_DERA_PE = cdp

Le critère est défini par une proportion de mailles à déraffiner. C'est un nombre compris entre 0 et 1. Le processus est le suivant :
- calcul du nombre de mailles n correspondant à la proportion définie par cdp soit n = cdp x
- déraffinement des n mailles avec la plus faible valeur de champ.

4.8.2 Opérande CRIT_DERA_ABS

/ CRIT_DERA_ABS = cda

Le critère est défini par une valeur absolue du champ. Toutes les mailles avec une valeur de champ inférieure à cette valeur seront déraffinées.

4.8.3 Opérande CRIT_DERA_REL

/ CRIT_DERA_REL = cdr

Le critère est défini par une valeur relative du champ. C'est un nombre compris entre 0 et 1. Le processus est le suivant :
- calcul des valeurs minimales et maximales de l'indicateur,
- calcul de la valeur d'erreur *v* correspondant à la proportion cdr telle que : v= vmin + cdr(vmax-vmin),
- déraffinement de toutes les mailles où le champ est inférieur à cette valeur.

4.8.4 Opérande CRIT_DERA_MS

◊ / CRIT_DERA_MS = cdms

Le critère est défini par une valeur absolue du champ, calculée en fonction de la moyenne et de l'écart-type de ce champ. Toutes les mailles avec une valeur inférieure à cette valeur seront déraffinées. Le critère vaut : moyenne - n*sigma, où n est le coefficient fourni, strictement positif.

4.9 Mot clé ZONE

♦ ZONE = _F (

Dans le cas d'une demande d'adaptation par zone, il faut définir au moins une zone. Ce mot-clé est à employer autant de fois que l'on veut définir de zones d'adaptation. Le principe est le suivant : on définit une zone par des coordonnées puis chaque maille dont au moins une des arêtes se trouve dans cette zone sera sélectionnée.
On a le choix entre plusieurs types de zones.

Attention :

Pour un calcul qui serait 2D, les types de zone sont de fait des rectangles ou des cercles. Mais comme la notion de maillage strictement 2D est inconnue dans Code_Aster au moment de la création des commandes, on supposera que la 3 ème coordonnée z est nulle.

4.9.1 Type de la zone

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

♦ TYPE= / 'RECTANGLE'
/ 'BOITE'
/ 'DISQUE'
/ 'SPHERE'
/ 'CYLINDRE'
/ 'DISQUE_PERCE'
/ 'TUYAU'

Cet opérande permet de définir le type de zone souhaité.

4.9.2 Usage de la zone

♦ USAGE= / 'RAFFINEMENT' [DEFAUT]
'DERAFFINEMENT'

Cet opérande permet de définir l'usage attribué à la zone. Avec le choix 'RAFFINEMENT', toutes les arêtes dont les deux extrémités appartiennent à la zone seront coupées, Avec le choix 'DERAFFINEMENT', toutes les arêtes dont les deux extrémités appartiennent à la zone seront réactivées.

4.9.3 Cas du rectangle

4.9.3.1 Opérandes X_MINI, X_MAXI, Y_MINI, Y_MAXI

♦ X_MINI = x_mini
♦ X_MAXI = x_maxi
♦ Y_MINI = y_mini
♦ Y_MAXI = y_maxi

Ce sont les valeurs extrêmes des coordonnées du rectangle englobant les mailles à raffiner.

4.9.4 Cas de la boîte parallélépipédique

4.9.4.1 Opérandes X_MINI, X_MAXI, Y_MINI, Y_MAXI, Z_MINI, Z_MAXI

♦ X_MINI = x_mini
♦ X_MAXI = x_maxi
♦ Y_MINI = y_mini
♦ Y_MAXI = y_maxi
♦ Z_MINI = z_mini
♦ Z_MAXI = z_maxi

Ce sont les valeurs extrêmes des coordonnées de la boîte englobant les mailles à raffiner.

4.9.5 Cas du disque

4.9.5.1 Opérandes X_CENTRE, Y_CENTRE

♦ X_CENTRE = x_centre
♦ Y_CENTRE = y_centre

Ce sont les coordonnées du centre du disque.

4.9.5.2 Opérande RAYON

♦ RAYON = rayon

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

C'est le rayon du disque.

4.9.6 Cas de la sphère

4.9.6.1 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

- X_CENTRE = x_centre
- Y_CENTRE = y_centre
- Z_CENTRE = z_centre

Ce sont les coordonnées du centre de la sphère.

4.9.6.2 Opérande RAYON

- RAYON = rayon

C'est le rayon de la sphère.

4.9.7 Cas du cylindre

Le cylindre est défini par un axe et un rayon. Il est limité par deux plans perpendiculaires à l'axe. Le premier plan est positionné par un point sur l'axe. Le second plan est distant du premier d'une hauteur, dans le sens du vecteur axial défini.

4.9.7.1 Opérandes X_AXE, Y_AXE, Z_AXE

- X_AXE = x_axe
- Y_AXE = y_axe
- Z_AXE = z_axe

Ce sont les coordonnées du vecteur directeur de l'axe du cylindre. L'orientation n'a pas d'importance. Le vecteur n'est pas nécessairement normé.

4.9.7.2 Opérandes X_BASE, Y_BASE, Z_BASE

- X_BASE = x_base
- Y_BASE = y_base
- Z_BASE = z_base

Ce sont les coordonnées d'un point à la base du cylindre et situé sur l'axe.

4.9.7.3 Opérande RAYON

- RAYON = rayon

C'est le rayon du cylindre.

4.9.7.4 Opérande HAUTEUR

- HAUTEUR = hauteur

C'est la hauteur du cylindre.

4.9.8 Cas d'un disque percé

4.9.8.1 Opérandes X_CENTRE, Y_CENTRE

- X_CENTRE = x_centre

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

♦ Y_CENTRE = y_centre

Ce sont les coordonnées du centre du disque.

4.9.8.2 Opérandes RAYON_INT, RAYON_EXT

♦ RAYON_INT = rayon_int
♦ RAYON_EXT = rayon_ext

Ce sont les rayons intérieur et extérieur du disque percé.

4.9.9 Cas du tuyau

Le tuyau est défini par un axe et ses rayons intérieur et extérieur. Il est limité par deux plans perpendiculaires à l'axe. Le premier plan est positionné par un point sur l'axe. Le second plan est distant du premier d'une hauteur, dans le sens du vecteur axial défini.

4.9.9.1 Opérandes X_AXE, Y_AXE, Z_AXE

♦ X_AXE = x_axe
♦ Y_AXE = y_axe
♦ Z_AXE = z_axe

Ce sont les coordonnées du vecteur directeur de l'axe du tuyau. L'orientation n'a pas d'importance. Le vecteur n'est pas nécessairement normé.

4.9.9.2 Opérandes X_BASE, Y_BASE, Z_BASE

♦ X_BASE = x_base
♦ Y_BASE = y_base
♦ Z_BASE = z_base

Ce sont les coordonnées d'un point à la base du tuyau et situé sur l'axe.

4.9.9.3 Opérandes RAYON_INT, RAYON_EXT

♦ RAYON_INT = rayon_int
♦ RAYON_EXT = rayon_ext

Ce sont les rayons intérieur et extérieur du tuyau.

4.9.9.4 Opérande HAUTEUR

♦ HAUTEUR = hauteur

C'est la hauteur du tuyau.

4.10 Opérandes GROUP_MA / GROUPE_NO

◊ GROUP_MA = l_grma
◊ GROUP_NO = l_grno

Si cette option est absente, le pilotage de l'adaptation s'applique à tout le maillage. Si on souhaite restreindre ce pilotage à une partie du maillage, on donne ici la liste des groupes qui définissent cette partie.
Exemple 1, pour raffiner uniformément une région du maillage : on demande du raffinement uniforme et on donne la liste des groupes de mailles formant cette région.

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

Exemple 2, pour n'appliquer le champ de pilotage de l'adaptation que sur certaines régions : on demande du raffinement/déraffinement avec le champ et on fournit la liste des groupes de mailles formant cette région.

Remarques :

Pour toutes les mailles 1D, 2D ou 3D contenues dans les groupes de la liste, il y a raffinement selon les critères retenus. Pour les mailles 0D ou les nœuds contenus dans les groupes, on retient les arêtes dont les deux extrémités sont dans ces listes.
Les mailles retenues sont adaptées, mais l'adaptation ira certainement un plus loin pour pouvoir fournir un maillage conforme en sortie.

4.11 Opérande DIAM_MIN

◊ DIAM_MIN = diamin

On rappelle que le diamètre d'une maille est la longueur du plus grand segment qu'il est possible de tracer à l'intérieur. Pour un triangle ou un tétraèdre, le diamètre est la longueur du plus grand côté. Pour un quadrangle, un hexaèdre ou un pentaèdre, le diamètre est la longueur de la plus grande diagonale.
Donner une valeur à diamin permet de ne pas rendre un maillage extrêmement fin. Une maille qui serait sélectionnée comme devant être raffinée à cause du champ de pilotage ou de la zone géométrique mais dont le diamètre est déjà inférieur à cette valeur minimale diamin ne sera pas découpée ; elle sera gardée telle quelle. Attention : il est néanmoins possible qu'au final elle soit quand même découpée si ses voisines le sont, pour respecter la conformité du maillage final.
Par défaut, aucune limite n'est donnée : on peut obtenir des mailles aussi petites que l'on veut.

4.12 Opérande NIVE_MAX

◊ NIVE_MAX = nivmax

C'est le niveau maximal de raffinement du maillage. Autrement dit une maille du maillage initial ne pourra pas être divisée plus de nivmax fois dans l'ensemble du processus. Cela permet d'assurer que le maillage ne va pas devenir extrêmement fin au voisinage d'une singularité : la taille minimale d'une arête sera sa taille initiale divisée par 2^{nivmax}.
Par défaut, aucune limite n'est donnée : on peut découper autant que l'on veut.

4.13 Opérande NIVE_MIN

◊ NIVE_MIN = nivmin

C'est le niveau minimal de déraffinement du maillage. C'est-à-dire que seules les mailles issues d'au moins nivmin découpages de maillage peuvent être déraffinées. Cela permet d'assurer que l'on en va pas remonter trop haut dans le déraffinement : on garde ainsi une finesse minimale au maillage.
Par défaut, aucune limite n'est donnée : on peut déraffiner jusqu'à retrouver le maillage initial.

4.14 Mot clé MAILLAGE_FRONTIERE

◊ MAILLAGE_FRONTIERE = maf

Le choix de cette option permet au processus d'adaptation de suivre la courbure des bords du maillage. L'option s'applique exclusivement à des bords 1D. Pour des bords 2D, il faut utiliser l'option FRONTIERE_ANALYTIQUE. On fournit ici un concept *Code_Aster* de type maillage qui contient un maillage fin des bords unidimensionnels de la géométrie. Ce maillage n'est donc formé *a priori* que de segments. Leurs longueurs sont très inférieures à celles des segments de bord du maillage à adapter. Si le processus d'adaptation est amené à couper un segment de bord, le nouveau nœud sera placé sur le maillage de la frontière. Ainsi les angles seront adoucis au fur et à mesure des adaptations.

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

Le repérage des différents bords se fait par les groupes avec les règles suivantes :
- les bords sont décrits par des groupes de segments ;
- un bord est décrit par le même nom de groupe dans le maillage de calcul et dans le maillage de la frontières ;
- un bord ne peut avoir que zéro ou deux extrémités ;
- il n'est ni indispensable ni déconseillé d'inclure les bords rectilignes ;
- le bord peut aussi bien être externe, le plus courant, qu'interne, pour séparer deux matériaux.
- le bord n'est pas nécessairement plan ; ce peut être une courbe dans l'espace 3D comme l'intersection de deux cylindres par exemple.

Dit autrement, un groupe de segments de bord doit comporter une liste de segments formant une ligne.

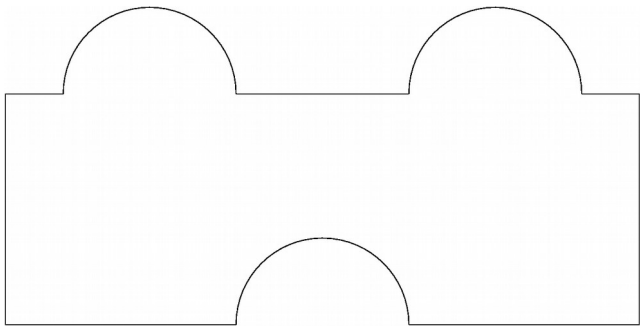
Remarque :

On regardera les cas-tests `zzzz121d`, `zzzz175a`, `zzzz175c` *et* `zzzz259a` *pour des exemples de pilotage du suivi de frontière ainsi que le site WEB de HOMARD pour une illustration graphique du résultat obtenu.*

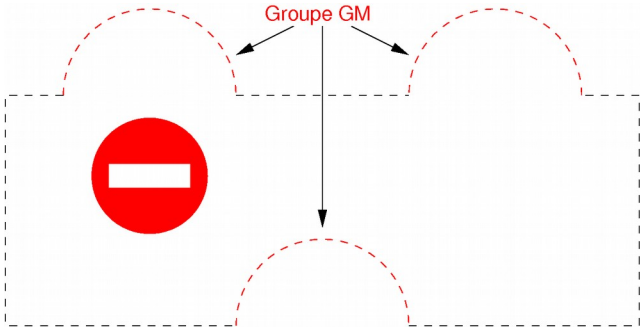
Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Exemple :

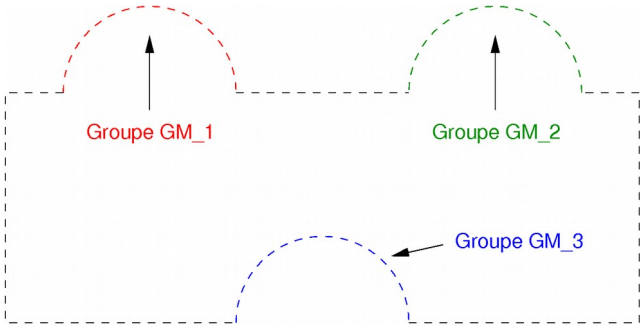
Considérons un objet bidimensionnel dont la frontière n'est pas toujours rectiligne. Cette frontière aura été maillée par des éléments SEG2 ou SEG3 aussi bien dans le maillage de calcul que dans le maillage annexe. Ces mailles de bord sont mises dans les mêmes groupes.



La mauvaise solution est celle-ci : repérer les mailles des bords courbes et les stocker toutes dans le même groupe. HOMARD ne sait pas gérer un bord fractionné ; il y aura arrêt avec un message signifiant que la ligne est en plusieurs morceaux.

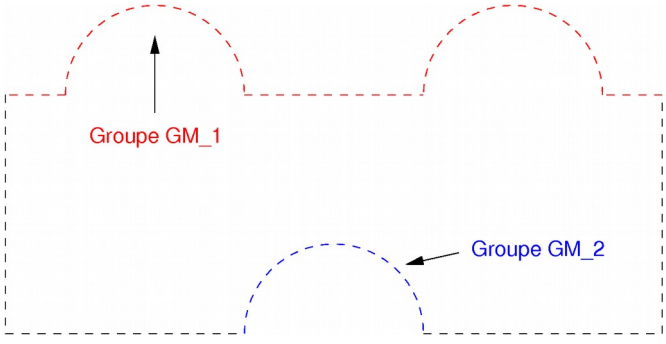


Une première bonne façon de faire consiste à créer autant de groupes que de zones d'intérêt.



Une autre solution acceptable consiste à regrouper par tronçon.

Entre les deux méthodes, pas de différence pour HOMARD : l'essentiel est de ne pas faire le tour complet (sinon, pas d'extrémité) et de ne pas couper (sinon, trop d'extrémités !). On choisira la méthode la plus facile à réaliser dans le mailleur.



4.14.1 Opérande GROUP_MA_FRONT

◊ GROUP_MA_FRONT = l_grma

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

Si cette option est absente, le suivi de la frontière se fait pour tous les groupes définis dans le maillage de la frontière. Si on souhaite restreindre ce suivi à une partie de la frontière, on donne ici la liste des groupes de segments qui définissent cette partie de frontière.

4.15 Mot clé FRONTIERE_ANALYTIQUE

◊ FRONTIERE_ANALYTIQUE = _F (

Ce mot-clé est à renseigner autant de fois que l'on veut définir de frontières analytiques. Le choix de cette option permet au processus d'adaptation de suivre la courbure des bords du maillage. L'option s'applique exclusivement à des bords 2D. Pour des bords 1D, il faut utiliser l'option MAILLAGE_FRONTIERE. On fournit ici la description analytique de chaque frontière à suivre. Si le processus d'adaptation est amené à couper une maille de bord, le nouveau nœud sera positionné sur la frontière, via sa description. Ainsi les angles seront adoucis au fur et à mesure des adaptations.

Remarque :

On regardera les cas-tests zzzz175a, zzzz175c et zzzz259a pour un exemple de suivi de frontière analytique.
Quand on lance plusieurs fois de suite l'adaptation, il est indispensable que chaque frontière analytique soit définie de la même manière à chaque invocation de MACR_ADAP_MAIL : nom, type, liste de groupes, caractéristiques géométriques.

4.15.1 Nom de la frontière

♦ NOM = nom [K]

Cet opérande permet de définir le nom associé à la frontière. Le choix de ce nom est libre.

4.15.2 Type de la frontière

♦ TYPE = / 'SPHERE'
/ 'CYLINDRE'
/ 'CONE_A'
/ 'CONE_R'
/ 'TORE'

Cet opérande permet de définir le type de frontière souhaité : sphère, cylindre.

4.15.3 Opérande GROUP_MA

♦ GROUP_MA = l_grma

On donne ici la liste des groupes de mailles qui définissent la partie de frontière représentée par cette définition analytique.

4.15.4 Cas de la sphère

4.15.4.1 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

♦ X_CENTRE = x_centre
♦ Y_CENTRE = y_centre
♦ Z_CENTRE = z_centre

Ce sont les coordonnées du centre de la sphère.

4.15.4.2 Opérande RAYON

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

♦ RAYON = rayon

C'est le rayon de la sphère.

4.15.5 Cas du cylindre

Le cylindre est défini par un axe, un point sur l'axe et un rayon.

4.15.5.1 Opérandes X_AXE, Y_AXE, Z_AXE

♦ X_AXE = x_axe
♦ Y_AXE = y_axe
♦ Z_AXE = z_axe

Ce sont les coordonnées du vecteur directeur de l'axe du cylindre. L'orientation n'a pas d'importance. Le vecteur n'est pas nécessairement normé.

4.15.5.2 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

♦ X_CENTRE = x_centre
♦ Y_CENTRE = y_centre
♦ Z_CENTRE = z_centre

Ce sont les coordonnées d'un point situé sur l'axe du cylindre.

4.15.5.3 Opérande RAYON

♦ RAYON = rayon

C'est le rayon du cylindre.

4.15.6 Cas du cône défini par un angle

Le cône est défini par un axe, un centre sur l'axe et un angle.

4.15.6.1 Opérandes X_AXE, Y_AXE, Z_AXE

♦ X_AXE = x_axe
♦ Y_AXE = y_axe
♦ Z_AXE = z_axe

Ce sont les coordonnées du vecteur directeur de l'axe du cône. L'orientation n'a pas d'importance. Le vecteur n'est pas nécessairement normé.

4.15.6.2 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

♦ X_CENTRE = x_centre
♦ Y_CENTRE = y_centre
♦ Z_CENTRE = z_centre

Ce sont les coordonnées du centre sur l'axe du cône.

4.15.6.3 Opérande ANGLE

♦ ANGLE = angle

C'est l'angle en degré du cône.

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

4.15.7 Cas du cône défini par des rayons

Le cône est défini par deux points sur son axe et les deux rayons correspondant à ces positions.

4.15.7.1 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

- X_CENTRE = x_centre
- Y_CENTRE = y_centre
- Z_CENTRE = z_centre

Ce sont les coordonnées du premier point situé sur l'axe du cylindre.

4.15.7.2 Opérande RAYON

- RAYON = rayon

C'est le rayon du cône pour cette première position sur l'axe.

4.15.7.3 Opérandes X_CENTRE2, Y_CENTRE2, Z_CENTRE2

- X_CENTRE2 = x_centre2
- Y_CENTRE2 = y_centre2
- Z_CENTRE2 = z_centre2

Ce sont les coordonnées du second point situé sur l'axe du cylindre.

4.15.7.4 Opérande RAYON2

- RAYON2 = rayon2

C'est le rayon du cône pour cette seconde position sur l'axe.

4.15.8 Cas du tore

Le tore est défini par un axe, un centre et les deux rayons.

4.15.8.1 Opérandes X_AXE, Y_AXE, Z_AXE

- X_AXE = x_axe
- Y_AXE = y_axe
- Z_AXE = z_axe

Ce sont les coordonnées du vecteur directeur de l'axe du tore. L'orientation n'a pas d'importance. Le vecteur n'est pas nécessairement normé.

4.15.8.2 Opérandes X_CENTRE, Y_CENTRE, Z_CENTRE

- X_CENTRE = x_centre
- Y_CENTRE = y_centre
- Z_CENTRE = z_centre

Ce sont les coordonnées du centre du tore.

4.15.8.3 Opérande RAYON

- RAYON = rayon

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

C'est le rayon de révolution du tore.

4.15.8.4 Opérande RAYON2

♦ RAYON2 = rayon2

C'est le rayon du cercle primaire qui tourne autour de l'axe du tore.

4.16 Mot clé MAJ_CHAM

◊ MAJ_CHAM = _F (

Ce mot-clé est à employer autant de fois que l'on a de champs à mettre à jour de l'ancien maillage vers le maillage adapté. Ce champ est contenu soit dans une structure de résultat, soit dans un champ de grandeurs.

Remarque :
Il faut veiller à ce que le champ soit défini sur un maillage identique au maillage rentré sous l'opérande MAILLAGE_N.

4.16.1 Opérande RESULTAT

/ ◊ RESULTAT = resu

Nom du concept [resultat] contenant le champ à mettre à jour.

4.16.1.1 Opérande NOM_CHAM

◊ NOM_CHAM = nomsymb [K16]

Nom symbolique du champ que l'on souhaite exprimer sur le nouveau maillage.

4.16.2 Opérande CHAM_GD

/ ◊ CHAM_GD = cham_gd

Nom du concept [cham_gd] contenant le champ à mettre à jour.

4.16.3 Opérande NOM_CMP

◊ NOM_CMP = l_cmp

Nom de la composante du champ qui doit être mise à jour. Si plusieurs composantes sont souhaitées, donner ici la liste.

Si aucune composante n'est définie ici, la commande prendra toutes celles qui existent dans le champ transmis.

4.16.4 Sélection du paramètre temporel du champ à mettre à jour

La sélection du numéro d'ordre associé au champ à interpoler se fait par la désignation d'un numéro d'ordre ou d'une valeur d'instant. Se référer au document [U4.71.00] pour les détails sur ces mots-clés.

4.16.5 Opérande TYPE_MAJ

◊ TYPE_MAJ = / 'AUTO' [DEFAUT]

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

```
                    /   'ISOP2'
```

On précise ici le type de mise à jour souhaité.
Par défaut, le fonctionnement 'AUTO', est ... automatique : l'interpolation est faite selon la nature du champ.
Pour un champ constant par élément, si une maille est découpée, la valeur du champ est reportée telle quelle sur les mailles filles.
Pour un champ aux nœuds, l'interpolation a lieu avec les fonctions de forme P1 ou P2 selon le support du champ. Une variante est possible pour les champs aux nœuds exprimés sur un maillage de degré 2. En précisant 'ISOP2', l'interpolation est faite par des fonctions de forme P1 exprimées sur les sous-mailles de l'élément. Cette technique garantit que le champ interpolé respecte les valeurs extrêmes du champ initial sur une maille.
Pour un champ aux points de Gauss, la mise à jour a lieu seulement dans les cas où les types de mailles sont homogènes entre l'entrée et la sortie : triangle ou tétraèdre. Pour les quadrangles, les hexaèdres ou les pentaèdres, rein n'est fait car on ne sait pas définir les familles de points de Gauss sur les triangles ou tétraèdres produits. Pour un maillage en triangles, on procède ainsi :

- pour un triangle inchangé après l'adaptation, les valeurs aux points de Gauss sont reconduites à l'identique.
- pour un triangle découpé, en 2 ou en 4, on calcule la valeur moyenne du champ sur tous les points de Gauss et cette valeur est attribuée à tous les points de Gauss de tous les triangles fils.
- pour un triangle qui disparaît (en cas de déraffinement), on calcule la moyenne du champ sur tous les points de Gauss de ce triangle et de ses frères et cette valeur est attribuée à tous les points de Gauss du triangle père qui réapparaît.

Ce mécanisme permet de garantir que les bornes extrêmes ne sont pas franchies. La même technqiue est appliquée aux tétraèdres.

Attention :

Il ne peut pas y avoir de contrôle de cohérence entre le type demandé et le type véritable du champ à interpoler.

4.16.6 Opérande CHAM_MAJ

♦ CHAM_MAJ = co(chpmaj) [K8]

Nom du concept qui contiendra le champ exprimé sur le nouveau maillage. Ce concept ne doit pas exister. Il sera automatiquement créé.
Un champ aux nœuds ou aux éléments sera lu automatiquement par la macro-commande qui demande l'adaptation ou la modification du maillage. Il sera disponible dans le jeu de calcul dès la fin de l'exécution de la macro-commande. En revanche, cela n'est pas possible pour un champ exprimé aux points de Gauss car Code_Aster a besoin de connaître le modèle pour lire. Il faut pour cela procéder en 3 temps. Le champ est calculé par la macro-commande qui a demandé l'adaptation ou la modification du maillage, comme pour un champ aux nœuds. Ensuite, un nouveau modèle doit être appliqué sur le nouveau maillage par la commande AFFE_MODELE. Enfin, la lecture sera faite par une nouvelle invocation de MACR_ADAP_MAIL moyennant le paramètre ADAPTATION = 'LECTURE' et la fourniture du maillage et du modèle.

Remarque :

On regardera le cas-test zzzz175b pour un exemple de mise à jour et de lecture de champs de différents types.

4.16.7 Opérande TYPE_CHAM

```
♦ TYPE_CHAM =   / 'NOEU_DEPL_R'
                / 'NOEU_TEMP_R'
                / 'ELGA_SIEF_R'
                / etc ...
```

On désigne ici le type du concept à mettre à jour sur le nouveau maillage. Le nom de ce type est construit avec la logique habituelle de *Code_Aster*. Les 4 premiers caractères sont 'NOEU', 'ELEM'

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

ou 'ELGA'. On trouve ensuite '_'. La séquence suivante définit le type de champ : 'TEMP', 'DEPL', etc. Le nom se termine par '_R' pour un champ réel.

Exemple : 'NOEU_TEMP_R', 'NOEU_DEPL_R', etc.

Attention :
Il ne peut pas y avoir de contrôle de cohérence entre le type demandé et le type véritable du champ à interpoler.

4.17 Mot clé ADD_CHAM

◊ ADD_CHAM = _F (

Ce mot-clé est à employer pour produire des champs particuliers sur le maillage adapté.

4.17.1 Opérande CHAM_GD

♦ CHAM_GD = co(chpmaj) [K8]

Nom du concept qui contiendra le champ exprimé sur le nouveau maillage. Ce concept ne doit pas exister. Il sera automatiquement créé.

Remarque :
On regardera les cas-tests zzzz121b et zzzz121f pour un exemple d'ajouts de champs de différents types.

4.17.2 Opérande CHAM_CAT

♦ CHAM_CAT = / 'NIVEAU'
/ 'QUALITE'
/ 'DIAMETRE'

On désigne ici la catégorie du champ que l'on veut produire.
Avec 'NIVEAU ', le champ produit contiendra pour chaque maille son niveau. On rappelle qu'une maille du niveau initial est du niveau 0. Ensuite une maille issue de *n* découpages standard d'une maille initiale portera le niveau *n*. Une maille qui assure la transition de conformité entre le niveau *n* et le niveau *n+1* portera le niveau *n+0,5*.
Avec 'QUALITE ', le champ produit contiendra la valeur de la qualité pour chaque maille.
Avec 'DIAMETRE ', le champ produit contiendra la valeur du diamètre pour chaque maille.

4.18 Opérande MODELE

◊ MODELE = modele [modele]

Attention :
Cet opérande est actif uniquement quand on a choisi 'LECTURE' comme type d'adaptation.

Cet opérande permet de préciser le modèle qui a été affecté au maillage sur lequel a eu lieu la mise à jour de champs exprimés aux points de Gauss.

Remarque :
On regardera le cas-test zzzz175b pour un exemple de mise à jour et de lecture de champs aux points de Gauss.

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

4.19 Opérande DEGRE

◊ DEGRE = / 'OUI'

Attention :
Cet opérande est actif uniquement quand on a choisi 'MODIFICATION' comme type d'adaptation.

Si le choix est 'OUI', le degré du maillage est globalement changé.

Remarque :
Combiné avec MAJ_CHAM , l'opérande DEGRE peut être utilisé par exemple pour le post-traitement de la pression en THM (cf doc [U2.04.05]). Pour certaines études 3D avec des maillages volumineux, il peut se révéler parfois plus performant que PROJ_CHAMP .

4.20 Opérande NOMBRE

Remarque :
On consultera le document [U7.03.02] décrivant la commande MACR_INFO_MAIL pour des commentaires sur les restitutions des opérandes QUALITE , INTERPENETRATION , NOMBRE , CONNEXITE et TAILLE .

◊ NOMBRE = / 'OUI' [DEFAUT]
/ 'NON'

Si le choix est 'OUI', un bilan des nombres de nœuds et de mailles est imprimé sur le fichier de messages.

4.21 Opérande QUALITE

◊ QUALITE = / 'NON' [DEFAUT]
/ 'OUI'

Si le choix est 'OUI', un bilan de la qualité des mailles est imprimé sur le fichier de message. La qualité d'un triangle est définie comme étant le rapport entre la longueur du plus grand côté et le rayon du cercle inscrit. La qualité d'un quadrangle est définie comme le quotient du produit de la plus grande longueur et des moyennes sur les côtés et les diagonales par la plus petite des surfaces des triangles internes aux quadrangles. De même, la qualité d'un tétraèdre est définie comme étant le rapport entre la longueur du plus grand côté et le rayon de la sphère inscrite. Ces rapports sont normalisés pour valoir 1 dans le cas d'un triangle équilatéral, d'un carré, d'un tétraèdre ou d'un hexaèdre équilatéral. Pour tout élément non équilatéral, la qualité est supérieure à 1. Voir la référence [bib1] pour des explications détaillées.
Le résultat est présenté sous forme de tableaux, avec les valeurs extrêmes.

L'interprétation des valeurs produites dépend de la méthode numérique employée pour le calcul. Selon que le problème est isotrope ou non, selon la rapidité de variation spatiale des données, selon la technique de calcul, une même maille peut conduire à un bon jacobien ou non. L'essentiel dans un premier temps est de repérer les mailles franchement mauvaises. Si on observe que le maximum de qualité dépasse 100, voire 1 000 ou 100 000, on doit s'inquiéter : une ou plusieurs mailles sont très déformées et le maillage est certainement à reprendre. Dans un second temps, cette information de qualité doit permettre de comparer deux maillages a priori corrects, sans grande valeur. SI le problème est isotrope, on aura intérêt à utiliser le maillage avec la répartition de qualité la plus proche de 1.
On trouvera des illustrations de valeurs de qualité de différentes mailles dans [Réf.5].

4.22 Opérande DIAMETRE

◊ DIAMTRE = / 'NON' [DEFAUT]

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

```
                    /   'OUI'
```

Si le choix est 'OUI', un bilan des diamètres des mailles est imprimé sur le fichier de message. Le diamètre d'une maille est définie comme la longueur du plus grand segment qu'il est possible d'insérer dans la maille.
Pour un triangle ou un tétraèdre, le diamètre correspond à la longueur du plus grand côté.

Pour un quadrangle, un hexaèdre, un pentaèdre ou une pyramide, le diamètre est le maximum entre la longueur du plus grand côté et la longueur de la plus grande diagonale.

Le résultat est présenté sous forme de tableaux, avec les valeurs extrêmes.

4.23 Opérande INTERPENETRATION

```
◊  INTERPENETRATION =  /   'NON'                    [DEFAUT]
                       /   'OUI'
```

Si le choix est 'OUI', on vérifie que le maillage est correct du point de vue du recouvrement : aucune maille n'entre dans une autre.

4.24 Opérande TAILLE

```
◊  TAILLE =  /   'NON'                    [DEFAUT]
             /   'OUI'
```

Si le choix est 'OUI', un bilan des tailles des sous-domaines est imprimé sur le fichier de messages. Un sous-domaine est défini comme un ensemble de mailles de même dimension et appartenant aux mêmes groupes.

4.25 Opérande CONNEXITE

```
◊  CONNEXITE =  /   'NON'                    [DEFAUT]
                /   'OUI'
```

Si le choix est 'OUI', un bilan des connexités est imprimé sur le fichier de messages. On saura alors si les segments, les éléments 2D (triangles et quadrangles réunis) ou les mailles 3D (tétraèdres, hexaèdres, pentaèdres et pyramides réunis) sont d'un seul tenant ou répartis en plusieurs blocs. On connaîtra également le nombre de trous de la structure : les trous traversants ou les trous internes.

4.26 Opérande PROP_CALCUL

```
◊  PROP_CALCUL  =  /   'NON'                    [DEFAUT]
                   /   'OUI'
```

Si le choix est 'OUI', un diagnostic sur les propriétés des mailles en tant qu'éléments pour le calcul est imprimé sur le fichier de messages. On dénombre le nombre d'éléments surcontraints : les éléments dont tous les sommets sont situés sur le bord. On dénombre les mailles volumiques (resp. surfaciques) qui touchent le bord du domaine mais qui ne sont pas bordées par des mailles surfaciques (resp. linéiques).

4.27 Les historiques

Il faut bien noter que le maillage ne pourra être déraffiné qu'au sein d'une même étude. Si on exécute un nouveau calcul (utilisation de DEBUT et non POURSUITE), l'historique de raffinement sera perdu.

4.28 Opérande LANGUE

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

```
◊  LANGUE =  /  'FRANCAIS'          [DEFAUT]
             /  'FRENCH'
             /  'ANGLAIS'
             /  'ENGLISH'
```

Cet opérande précise la langue dans laquelle sont imprimés les messages issus de HOMARD.

4.29 Opérande VERSION_HOMARD

```
◊  VERSION_HOMARD  =  /  'V11_10'  [DEFAUT]
                      /  'V11_N'
                      /  'V11_N_PERSO'
```

Cet opérande permet de sélectionner la version de HOMARD qui est utilisée pour l'adaptation. Par défaut, HOMARD 11.10 est lancé. C'est la version de référence. Le choix 'V11_N' active la version 11.n de HOMARD qui est la version de développement. Le choix 'V11_N_PERSO' active une version de développement propre à l'utilisateur. Cette option permet à l'équipe de développement de HOMARD de mettre au point de nouvelles fonctionnalités. Elle permet aussi de faire bénéficier l'utilisateur d'une innovation dans HOMARD avant la mise en service dans *Code_Aster*.

4.30 Opérande LOGICIEL

```
◊  LOGICIEL =  logiciel          [K]
```

Cette option propose d'utiliser une autre interface de couplage entre *Code_Aster* et HOMARD que celle fournie par défaut dans le répertoire des outils associés à *Code_Aster*. Cette option est de fait réservée à l'équipe de développement de HOMARD pour mettre au point de nouvelles fonctionnalités. Elle permet de tester des nouveautés avant d'avoir modifié la macro-commande de pilotage.

4.31 Opérande UNITE

```
◊  UNITE  =  unite           [I]
```

Cette option n'est possible que si on a activé la version de développement de HOMARD, 11.n. Le fichier de données transmis par l'utilisateur sous ce numéro d'unité logique sera directement transmis comme complément au fichier de configuration de HOMARD. Cette option est de fait destinée à l'équipe de développement de HOMARD pour mettre au point de nouvelles fonctionnalités. Elle permet de tester des nouveautés avant d'avoir modifié la macro-commande de pilotage.

4.32 Opérande ELEMENTS_ACCEPTES

```
◊  ELEMENTS_ACCEPTES  =  /  'HOMARD'      [DEFAUT]
                         /  'IGNORE_PYRA'
```

Dans sa version actuelle, HOMARD sait lire tous les types de mailles mais ne fait porter l'adaptation que sur certaines : mailles-points, segments, triangles, quadrangles, tétraèdres, hexaèdres ou pentaèdres. Le maillage est en degré 1 ou 2, mais il n'est pas possible de mélanger les deux.
En retenant l'option 'HOMARD', la transmission d'un maillage contenant autre chose que ces types de mailles entraînera un arrêt en erreur. C'est l'option par défaut.
En choisissant l'option 'IGNORE_PYRA', on pourra transmettre un maillage comportant des pyramides. L'adaptation ne portera que sur les zones autorisées par HOMARD. Si par suite de propagation du raffinement, une zone interdite vient à être touchée, il y aura un arrêt en erreur. Sinon, quand le raffinement se limite à la zone autorisée, les autres mailles sont restituées sans changement.

Dans tous les cas, la présence des mailles enrichies HEXA27 est interdite.

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

4.33 Opérande INFO

◊ INFO = / 1
/ 2
/ 3
/ 4

Si INFO vaut 1, les impressions sont minimales ; on n'obtient que celles qui ont explicitement été demandées, la qualité des mailles par exemple, et les éventuels messages d'erreur.
Si INFO vaut 2, on obtiendra les messages émis par les commandes sous-jacentes à la macro-commande : IMPR_RESU, LIRE_MAILLAGE, LIRE_RESU.
Si INFO vaut 3, on aura les messages standard de HOMARD, récapitulant l'exécution.
Si INFO vaut 4, on aura tous les messages émis par HOMARD, en vue de déboggage.

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

5 Exemple

On regardera avec profit les fichiers de commandes associés aux cas-tests zzzz121a, b, c, d, e, f et zzzz175a, b. Ils expriment les processus d'adaptation de maillage soit par enchaînement des commandes soit sous la forme d'une boucle en langage Python.

Voici un exemple de paramétrage de la macro-commande.

```
MACR_ADAP_MAIL   (
          ADAPTATION = 'RAFF_DERA',
          MAILLAGE_N = mun,
          MAILLAGE_NP1 = CO ("mdeux"),
          RESULTAT_N = remeun,
          NOM_CHAM = 'QIRE_ELEM',
          NOM_CMP = 'ERREST'
          NUME_ORDRE = 3,
          CRIT_RAFF_PE = 0.01,
          CRIT_DERA_PE = 0.25,
          NIVE_MAX = 5
          MAJ_CHAM  = _F   (
                RESULTAT = rethun,
                NOM_CHAM = 'TEMP',
                TYPE_CHAM = 'NOEU_TEMP_R',
                INST = 12.5,
                CHAM_MAJ = CO ("tempdeux")
                ),
          QUALITE = 'OUI',
       )
```

Cette séquence va adapter le maillage contenu dans le concept mun et restituer un concept maillage de nom mdeux. L'adaptation se fait par raffinement et déraffinement libre, selon le champ contenu dans le champ QIRE_ELEM du résultat remeun, au 3ème instant ; la composante utilisée est ERREST. Les mailles seront classées en fonction de leur niveau d'erreur décroissant. Le premier % sera raffiné ; les 25% dernières seront candidates au déraffinement. Aucune maille du maillage final ne devra être issue de plus de 5 raffinements.

Le champ TEMP du résultat rethun à l'instant 12,5 est exprimé sur le maillage mun. Il sera exprimé sur le maillage mdeux sous la forme du champ de température aux nœuds tempdeux.

Un récapitulatif de la qualité des mailles du nouveau maillage est produit. On ne contrôle pas l'interpénétration des mailles.

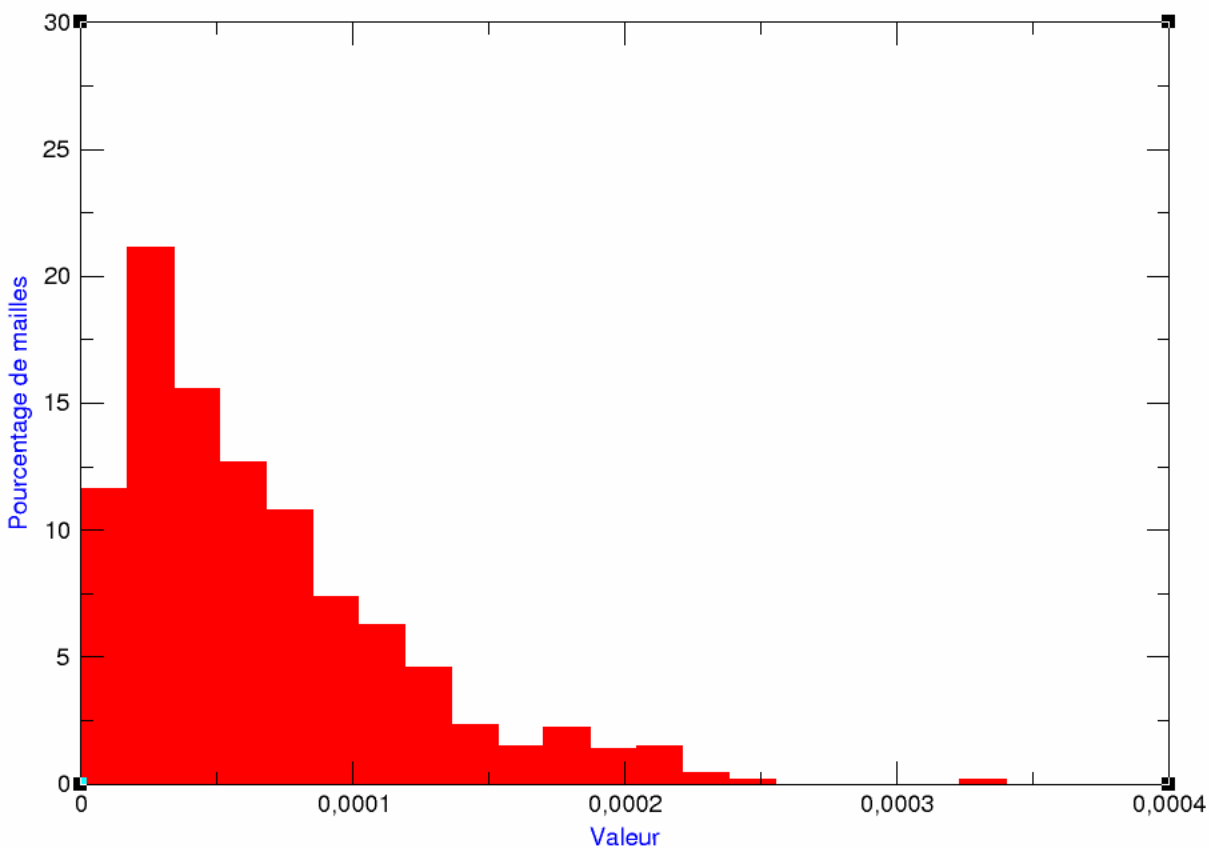
Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Voici un exemple du tableau présentant la répartition du champ pilotant l'adaptation du maillage.

```
************************************************************
*               Champ pilotant l'adaptation                *
*            Valeur sur les         936 triangles         *
************************************************************
*   Minimum :  0.35358E-05        Maximum :  0.33395E-03   *
*   Moyenne :  0.66371E-04     Ecart-type :  0.51323E-04   *
************************************************************
*                  Fonction de repartition                 *
*      Valeurs       *         Nombre de mailles           *
*    Mini <  < Maxi  *    par classe    *      cumul       *
*     * 10**-4       *  en %  . nombre  *  en %  . nombre  *
************************************************************
*    0.000 <   0.170 *  11.65 .     109 *  11.65 .     109 *
*    0.170 <   0.340 *  21.15 .     198 *  32.80 .     307 *
*    0.340 <   0.510 *  15.60 .     146 *  48.40 .     453 *
*    0.510 <   0.680 *  12.71 .     119 *  61.11 .     572 *
*    0.680 <   0.850 *  10.79 .     101 *  71.90 .     673 *
*    0.850 <   1.020 *   7.37 .      69 *  79.27 .     742 *
*    1.020 <   1.190 *   6.30 .      59 *  85.58 .     801 *
*    1.190 <   1.360 *   4.59 .      43 *  90.17 .     844 *
*    1.360 <   1.530 *   2.35 .      22 *  92.52 .     866 *
*    1.530 <   1.700 *   1.50 .      14 *  94.02 .     880 *
*    1.700 <   1.870 *   2.24 .      21 *  96.26 .     901 *
*    1.870 <   2.040 *   1.39 .      13 *  97.65 .     914 *
*    2.040 <   2.210 *   1.50 .      14 *  99.15 .     928 *
*    2.210 <   2.380 *   0.43 .       4 *  99.57 .     932 *
*    2.380 <   2.550 *   0.21 .       2 *  99.79 .     934 *
*    2.550 <   2.720 *   0.00 .       0 *  99.79 .     934 *
*    2.720 <   2.890 *   0.00 .       0 *  99.79 .     934 *
*    2.890 <   3.060 *   0.00 .       0 *  99.79 .     934 *
*    3.060 <   3.230 *   0.00 .       0 *  99.79 .     934 *
*    3.230 <   3.400 *   0.21 .       2 * 100.00 .     936 *
************************************************************
```

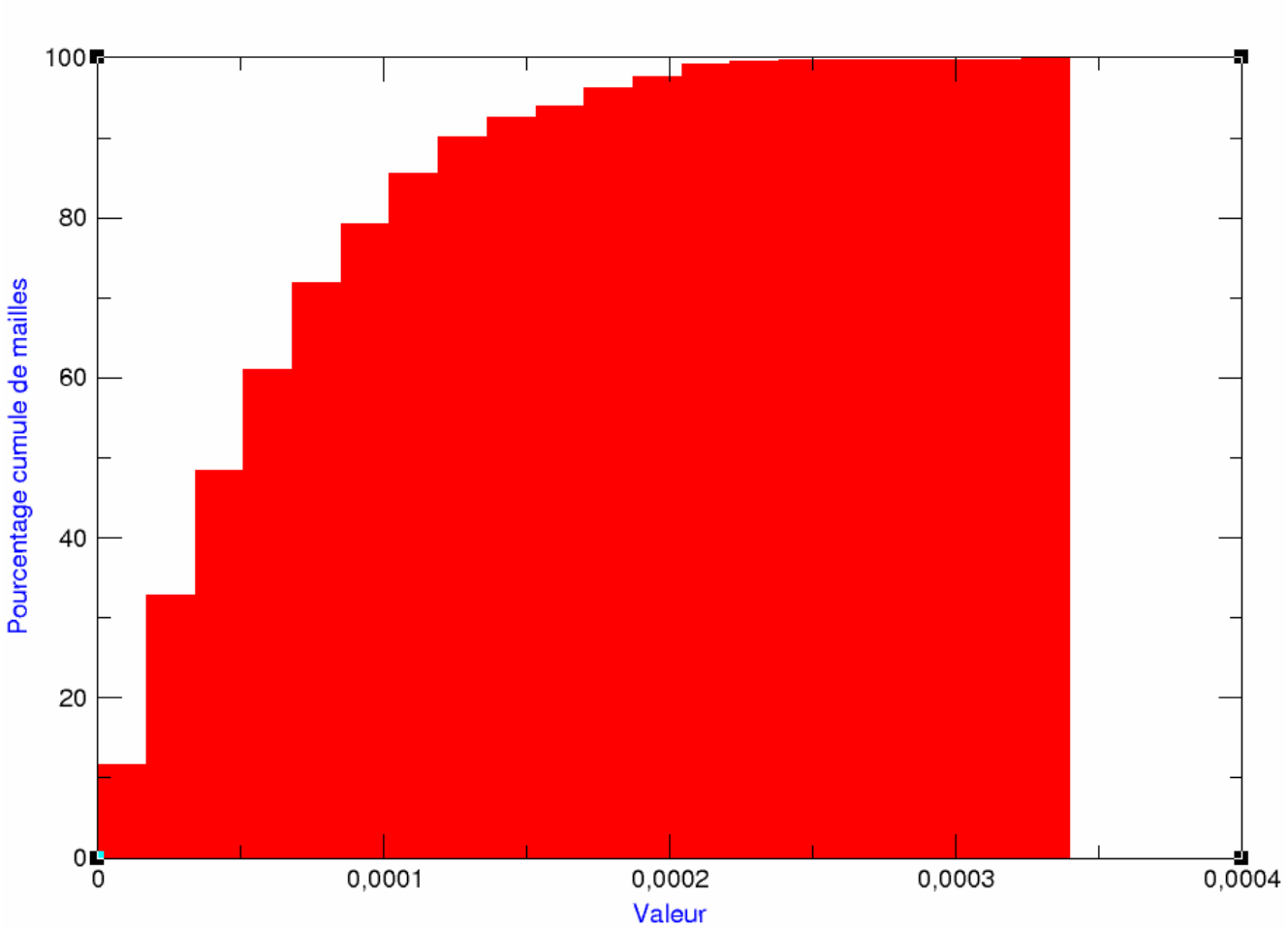

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Le diagnostic sur la répartition du champ pilotant l'adaptation de maillage rappelle d'abord les valeurs extrêmes rencontrées dans le calcul en cours. Ici le minimum est de $0{,}353585 \times 10^{-5}$ et le maximum de $0{,}33395 \times 10^{-3}$. On précise la valeur moyenne, $0{,}66371 \times 10^{-4}$, et l'écart-type, $0{,}51323 \times 10^{-4}$. Ensuite on présente la répartition par tranche équidistante à partir de la valeur optimum, 0. On voit que pour 880 triangles, la valeur du champ est inférieure à $1{,}70 \times 10^{-4}$, soit 94,02 % du nombre total de triangles. Ensuite, pour 21 triangles la valeur du champ est comprise entre $1{,}70 \times 10^{-4}$ et $1{,}87 \times 10^{-4}$, soit 2,24 % du nombre total de triangles. En cumulé, on constate donc que pour 901= 880 + 21 triangles, la valeur du champ est inférieure à $1{,}87 \times 10^{-4}$, soit 96,26 % du total. Et ainsi de suite. Par exemple, pour 99,79 % des mailles,la valeur du champ est inférieure à $2{,}55 \times 10^{-4}$.



Sur la figure précédente, on peut voir la représentation sous forme d'histogramme des pourcentages de mailles dans chacune des plages de valeur concernées. Comme on pouvait également le constater dans le tableau précédent, on constate que très peu de mailles concentrent les fortes valeurs.
En visualisant une représentation du pourcentage cumulé de mailles dans une plage de valeur donnée, on a la figure suivante.

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



De cette répartition des valeurs, on peut déduire deux conséquences sur les stratégies de raffinement.
Si on demande un raffinement sur un critère relatif de la valeur du champ, mot-clé `CRIT_RAFF_REL`, cela revient à sélectionner les mailles les éléments qui se trouvent à droite de la ligne verticale passant par ce critère. Par exemple si on demande `CRIT_RAFF_REL = 0.77`, on sélectionnera toutes les mailles dont l'erreur est supérieure à $0.35358 \times 10^{-5} + 0,77 \times (0.33395 \times 10^{-3} - 0.35358 \times 10^{-5})$, soit $2,58 \times 10^{-3}$. On constate que cela correspond à très peu de mailles : 2 seulement dépassent cette valeur, soit 0,21% du total. On avait l'impression de demander un raffinement important, 0,77 soit un quart *grosso modo*, mais en fait on ne raffine quasiment rien.
Si on demande un raffinement sur un pourcentage de mailles, mot-clé `CRIT_RAFF_PE`, cela revient à sélectionner les mailles qui se trouvent au-dessus de la ligne horizontale passant par ce critère. Par exemple si on demande `CRIT_RAFF_PE = 0.10`, on sélectionnera les 10% de mailles les pires, soit 93 mailles. C'est la ligne horizontale à 90%. Parmi ces mailles-là, les « moins pires » portent une valeur inférieure à $1,36 \times 10^{-4}$, soit 40% de la valeur maximum. C'est assez efficace puisqu'on aura piégé les gros écarts.
La conséquence de ces remarques est qu'il convient de faire une première analyse de la répartition des valeurs du champ avant de choisir le type et les valeurs des critères de raffinement. Il est en effet inutile, voire coûteux en terme d'augmentation de la taille de maillage, de raffiner dans des zones où le champ n'est pas très fort. L'adaptation sera d'autant plus performante que l'on aura su réduire les mailles à forte valeur jusqu'à obtenir un équilibre dans le maillage.

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

6 Bibliographie

1) G. Nicolas, T. Fouquet : “Logiciel HOMARD - Volume 1 - Présentation générale”, rapport EDF H-I23-2008-04107-FR, octobre 2014.

2) G. Nicolas, T. Fouquet : “Logiciel HOMARD - Volume 2 – Algorithmes de raffinement et déraffinement de maillages”, rapport EDF H-I23-2008-04108-FR, octobre 2014.

3) G. Nicolas, T. Fouquet : “Logiciel HOMARD - Volume 3 – Interfaces avec les codes de calcul”, rapport EDF H-I23-2008-04118-FR, octobre 2014.

4) G. Nicolas, T. Fouquet : “Logiciel HOMARD - Volume 4 – Structures de données”, rapport EDF H-I23-2008-04120-FR, octobre 2014.

5) G. Nicolas and T. Fouquet, « *Adaptive Mesh Refinement for Conformal Hexahedral Meshes* », Finite Elements in Analysis and Design, Vol. 67, pp. 1-12, 2013, doi:10.1016/j.finel.2012.11.008

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