Procédure IMPR_RESU au format 'GMSH'

1 But

Écrire un champ de grandeur ou un concept résultat au format 'GMSH'.

Actuellement cette procédure permet d'écrire au choix :

- des champs aux nœuds (de déplacements, de températures, de modes propres, de modes statiques, …),
- des champs par éléments aux nœuds (de contraintes, d'efforts généralisés, de variables internes…).

Pour les concepts de type resultat, on peut n'imprimer qu'une partie de l'information, en sélectionnant les champs et les numéros d'ordre que l'on désire exploiter.

Il est possible de sélectionner les entités topologiques (mailles ou groupes de mailles) sur lesquelles on veut imprimer les résultats.

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

2 Syntaxe

```
IMPR_RESU    (
    ◇ MODELE =  mo,                                          [modele]
    ◇ UNITE  =  / unit,
                / 37                                         [DEFAUT]
    ◊ PROC0 =   / 'OUI',                                     [DEFAUT]
                / 'NON',
    ◆ FORMAT =  /  'GMSH',

    ◆ RESU   =  _F (
       ◆ /      MAILLAGE  =  ma,                         /  [maillage]
                                                         /  [squelette]
         /  CHAM_GD   =  ch_gd,
         /  RESULTAT  =  resu,
                    ◇ /     TOUT_CHAM    =  /'OUI',       [DEFAUT]
                                            /'NON',
                      /     NOM_CHAM     =  l_nomsymb,    [l_K16]
                    ◇ /     TOUT_ORDRE   =  'OUI',        [DEFAUT]
                      /     NUME_ORDRE   =  lordre,       [l_I]
                      /     LIST_ORDRE   =  lenti,        [listis]
                      /     NUME_MODE    =  lmode,        [l_I]
                      /     NOEUD_CMP    =  lnoecmp,      [l_K16]
                      /     NOM_CAS      =  ncas,         [l_K16]
                      /     ANGLE        =  langl,        [l_K16]
                      /  /  FREQ         =  lfreq,        [l_R]
                         /  LIST_FREQ    =  lreel,        [listr8]
                         /  INST         =  linst,        [l_R]
                         /  LIST_INST    =  lreel,        [listr8]
                         ◇  |  PRECISION =  / prec,       [R]
                                            / 1.0D-3,     [DEFAUT]
                            |  CRITERE   =  / 'RELATIF',  [DEFAUT]
                                            / 'ABSOLU',
         # Sélection des entités topologiques et des composantes
         ◇ /  MAILLE    =  l_maille,                      [l_maille]
           /  GROUP_MA  =  l_grma,                        [l_gr_maille]
         ◇     TYPE_CHAM =  /  'SCALAIRE',               [DEFAUT]
                               ◇ NOM_CMP   =  lnomcmp,   [l_K8]
                            /  /  'VECT_3D',
                               /  'TENS_2D',
                               /  'VECT_2D',
                               /  'TENS_3D',
                               ♦ NOM_CMP   =  lnomcmp,   [l_K8]
                      ),
         ◇ PARTIE =  /  'REEL',
                     /  'IMAG',

       ◇ VERSION          =  / 1.2,                      [DEFAUT]
                             / 1.0,                      [R]
    )  ;
```


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

3 Mot clé RESU

Ce mot clé facteur permet de spécifier les résultats à imprimer et le format selon lequel on veut les imprimer.

3.1 Opérande MAILLAGE

Cet opérande permet de choisir le maillage que l'on souhaite imprimer. L'impression simultanée du maillage et de champs est impossible

3.2 Sélection des données à imprimer

Les valeurs des champs calculés sont stockées dans des structures de données appelées champs de grandeur. Ces champs de grandeur peuvent être directement accessibles (concept CHAM_GD) ou se trouver dans une structure de données regroupant plusieurs champs de grandeur (concept resultat).

- Un champ de grandeur est une structure de données qui permet de stocker des champs définis aux nœuds (cham_no_*) ou des champs définis par éléments (cham_elem_*). Pour les champs par éléments, on distingue les champs définis aux nœuds des éléments et les champs définis aux points de GAUSS des éléments.
- Un concept resultat se compose d'un ou plusieurs champs de grandeur. Par exemple, à chaque pas de calcul on stocke dans la structure de données resultat, le champ de grandeur déplacement. Cette structure est donc une structure matricielle d'ordre 2, dont un indice est, par exemple, la liste des instants de calcul et l'autre l'ensemble des champs calculés (déplacements, contraintes, déformations, ...).

On accède dans ce cas à un champ de grandeur en spécifiant une valeur d'une variable d'accès (numéro d'ordre, instant, fréquence, numéro de mode, ...) et un nom de champ ('DEPL', 'SIGM_ELNO',...). Il existe plusieurs types de concept resultat : evol_elas, evol_noli, mode_meca,.... A chacun correspond une liste de champs et une liste de variables d'accès licites.

Compte-tenu de la structure de donnée resultat, on comprend aisément que les possibilités d'impression dont on dispose sont celles des champs de grandeur, complétées par des possibilités spécifiques.

Attention

On ne peut imprimer dans un même fichier le MAILLAGE et le RESULTAT/CHAM_GD , GMSH ne sait pas lire un tel fichier. La commande s'arrête donc si les deux mots-clés sont présents.

3.2.1 Opérande CHAM_GD

Pour les CHAM_NO, les composantes du champ aux nœuds sont éclatées en un champ vectoriel regroupant les composantes DX, DY, DZ et en autant de champs scalaires qu'il y a d'autres composantes.

Exemple : Dans le cas d'un champ aux nœuds de grandeur DEPL et dont la modélisation est de type poutre, on écrit :

- 1 champ vectoriel (regroupant l'ensemble des composantes de translation),
- 3 champs scalaires (un champ par degré de rotation).

Pour les CHAM_ELEM, on imprime seulement les champs par éléments aux noeuds (_ELNO_). Un champ par éléments aux nœuds est éclaté en autant de champs scalaires qu'il y a de composantes.

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

Remarque

Lorsque l'on écrit un résultat au format GMSH, le titre des champs, qui apparait dans GMSH, est constitué du nom du résultat Aster + le nom du champ + la composante.
Exemple : RESU_DEPL_DX : la composante 'DX' du champ 'DEPL' du résultat 'RESU'
Si on écrit un champ dérivé, on intercale le nom du paramètre sensible par rapport auquel on dérive, entre le nom du résultat et le nom du champ.
Exemple : RESU_PS1_DEPL_DX : la dérivée par rapport à 'PS1' de la composante 'DX' du champ 'DEPL' du résultat 'RESU'.

3.2.2 Opérande RESULTAT

L'opérande RESULTAT permet d'imprimer les champs contenus dans un concept resultat. On peut par exemple choisir de n'imprimer que certains champs (Cf. le mot clé : NOM_CHAM).
Le concept resultat est écrit champ de grandeur par champ de grandeur comme décrit dans le [§4.1].

4 Choix des champs à imprimer

4.1 Opérandes TOUT_CHAM / NOM_CHAM

Cf. document [U4.71.00].

4.2 Opérandes TOUT_ORDRE / NUME_ORDRE / LIST_ORDRE / NUME_MODE / INST / LIST_INST / FREQ / LIST_FREQ / NOEUD_CMP / NOM_CAS / ANGLE / PRECISION / CRITERE

Cf. document [U4.71.00].

5 Sélection des zones géométriques et des composantes

Afin de réduire le volume des impressions, il est parfois nécessaire de n'imprimer qu'une partie du résultat. Pour ce faire on peut n'imprimer un champ aux nœuds ou un champ par élément qu'en certains éléments.

Cette sélection que l'on notera "sélection sur des entités topologiques" est possible au format 'GMSH'.

5.1 Opérande MAILLE

Ce mot clé permet d'indiquer la liste des mailles sur lesquelles on désire imprimer un cham_elem. Pour un cham_no, il permet d'indiquer la liste des nœuds, sommets des mailles auxquels on désire imprimer le cham_no.

5.2 Opérande GROUP_MA

Cet opérande permet d'indiquer la liste des groupes de mailles sur lesquelles on désire imprimer un cham_elem. Pour un cham_no, il permet d'indiquer la liste des nœuds, sommets des mailles auxquels on désire imprimer le cham_no.

5.3 Opérande TYPE_CHAM

GMSH car il est capable de visualiser des vecteurs et des tenseurs ; il est ainsi possible de visualiser les autres choses que chacune des composantes indépendamment. Cet opérande permet de choisir le

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

type de champ à imprimer (`'VECT_2D'`, `'VECT_3D'`, `'SCALAIRE'`, `'TENS_2D'` ou `'TENS_3D'`). La valeur par défaut est `'SCALAIRE'`.

5.4 Opérande NOM_CMP

Ce opérande permet de choisir la liste des composantes que l'on désire imprimer en fonction du type de champ choisi.
Si l'on a choisi d'imprimer des champs scalaires il n'est pas nécessaire de renseigner `NOM_CMP`, dans ce cas, chacune des composantes seront imprimées indépendamment les unes des autres.

5.5 Opérande PARTIE

On ne peut pas imprimer de champs complexes au format GMSH, il faut donc, pour de tels champs, sélectionner la partie `REEL`le ou `IMAG`inaire à imprimer.

6 Opérande VERSION

Les mailles acceptées par `GMSH` (en post-traitement) sont, depuis la version 1.35, les mailles de type `POI1`, `SEG2`, `TRIA3`, `QUAD4`, `TETRA4`, `HEXA8`, `PENTA6`, `PYRAM5`, ce qui correspond à :

`VERSION = '1.2'` qui est la valeur par défaut

Si on a besoin de visualiser les champs avec une ancienne version de GMSH ou bien si l'on souhaite découper toutes les mailles en `POI1`, `SEG2`, `TRIA3` et `TETRA4`, on utilise `VERSION = '1.0'`.

Pour pouvoir visualiser les vues avec `GMSH`, le maillage est modifié pour n'avoir que des mailles `GMSH`. Actuellement, les mailles `PYRAM13` ne sont pas traitées.

- le `SEG3` est décomposé en 2 `SEG2`,
- le `SEG4` est décomposé en 3 `SEG2`,
- le `TRIA6` est décomposé en 4 `TRIA3`,
- le `TRIA7` est décomposé en 4 `TRIA3`,
- le `QUAD4` est décomposé en 2 `TRIA3` (découpé uniquement si `VERSION='1.0'`),
- le `QUAD8` est décomposé en 6 `TRIA3`,
- le `QUAD9` est décomposé en 6 `TRIA3`,
- le `TETRA10` est décomposé en 8 `TETRA4`,
- le `PENTA6` est décomposé en 3 `TETRA4` (découpé uniquement si `VERSION='1.0'`),
- le `PENTA15` est décomposé en 16 `TETRA4`,
- le `PYRAM5` est décomposé en 2 `TETRA4` (découpé uniquement si `VERSION='1.0'`),
- le `HEXA8` est décomposé en 6 `TETRA4` (découpé uniquement si `VERSION='1.0'`),
- le `HEXA20` est décomposé en 24 `TETRA4`,
- le `HEXA27` est décomposé en 48 `TETRA4`.

Remarque 1

Le `TRIA7` (respectivement `QUAD9`) est découpé comme le `TRIA6` (resp. `QUAD8`) car le nœud central ne porte pas de degré de liberté de déplacement en modélisation `COQUES_3D`.

Remarque 2

Avec les versions récentes de `GMSH` (>1.60), on pourrait éviter de découper les éléments. Le choix n'est pas disponible dans Code_Aster car il est possible de lire le format MED dans GMSH depuis la version 2.0.

7 Opérande UNITE

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

UNITE permet de choisir sur quel fichier les résultats seront écrits. Par défaut, UNITE = 37 au format 'GMSH'.

On peut ainsi imprimer des résultats sur des fichiers différents à chaque commande IMPR_RESU.
Le suffixe naturel des fichiers de post traitement GMSH est .pos et l'unité logique habituelle (par défaut dans astk) est 37.

8 Opérande PROC0

L'opérande PROC0 dont la valeur par défaut est 'OUI', permet de restreindre l'impression sur le processeur de rang 0. Si on lui affecte la valeur 'NON', les impressions seront effectuées sur tous les processeurs.

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