

Manuel d'Utilisation
Fascicule U7.0- : Echanges de données
Document : U7.05.32

Procédure IMPR_RESU au format 'GMSH'

1 But

Ecrire 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.

2 Syntaxe

```

IMPR_RESU      (
  MODELE=      mo,                                [modele]
  ◊ UNITE = unit,
  ♦ FORMAT      = / 'GMSH',

  RESU =      (
    ♦ | / MAILLAGE      = ma,                      / [maillage]
                                              / [squelette]

    | / CHAM_GD      = ch_gd,

    / RESULTAT      = resu,

    # Extraction d'un champ de grandeur de resu
      ◊ / TOUT_CHAM      = / 'OUI',                [DEFAULT]
                                / 'NON',
      / NOM_CHAM      = l_nomsymb,                [l_K16]

      ◊ / TOUT_ORDRE      = 'OUI',                [DEFAULT]
      / NUME_ORDRE      = lordre,                [l_I]
      / LIST_ORDRE      = lenti,                 [listis]
      / NUME_MODE      = lmode,                  [l_I]
      / NOEUD_CMP      = lnoecmp,                [l_K16]
      / NOM_CAS      = ncas,                     [l_K16]
      / ANGL      = 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,        [DEFAULT]
      | CRITERE      = / 'RELATIF',              [DEFAULT]
                                / 'ABSOLU',

    | ♦ PARTIE = / 'REEL',
              / 'IMAG',

    # Sélection des entités topologiques

    ◊ | MAILLE      = l_maille,                    [l_maille]
      | GROUP_MA      = l_grma,                   [l_gr_maille]
      | NOM_CMP      = l_cmp,                     [l_K8]

    ◊ VERSION      = / 1.2,                        [DEFAULT]
                      / 1.0,                        [R]

  )
)

```

3 Mot clé RESU

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

4 Opérandes CHAM_GD et RESULTAT

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_DEPL',...). 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 (cf. [U5]).

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.

4.1 Opérande CHAM_GD

Pour les `CHAM_NO`, les composantes du champ aux nœuds sont éclatées en un champ vectoriel regroupant les composantes `DX`, `DY` et `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.

4.2 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].

5 Extraction d'un champ de grandeur

5.1 Opérandes TOUT_CHAM / NOM_CHAM

Cf. document [U4.71.00].

5.2 Opérandes TOUT_ORDRE / NUME_ORDRE / LIST_ORDRE / NUME_MODE / INST / LIST_INST / FREQ / LIST_FREQ / NOEUD_CMP / NOM_CAS / ANGL / PRECISION / CRITERE

Cf. document [U4.71.00].

6 Sélection des entités topologiques 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 désirer 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'.

On ne peut pas imprimer de champs complexes au format GMSH, il faut donc, pour de tels champs, sélectionner la PARTIE REELLE ou IMAGINAIRE à imprimer.

6.1 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`.

6.2 Opérande GROUP_MA

Ce mot clé 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`.

6.3 Opérande NOM_CMP

Ce mot clé permet de choisir la liste des composantes que l'on désire imprimer.

Si ce mot-clé n'est pas renseigné, on imprime le champ au format tenseur (9 composantes par nœud).

7 Opérande UNITE

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 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

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

Page laissée intentionnellement blanche.