

Macro-order MACRO_EXPANS

1 Goal

Lmacro-order has MACRO_EXPANS allows to carry out the expansion of experimental data on a digital model starting from a base of expansion. It consists of the succession of the operators EXTR_MODE, PROJ_MESU_MODAL, REST_GENE_PHYS, and PROJ_CHAMP.

2 Syntax

```
MACRO_EXPANS (
    ♦ MODELE_CALCUL = _F ( ♦ MODEL = modelnum,           [modele_sdaster]
                          ♦ BASE = bases,
    [mode_meca]
                          ♦ NUME_MODE = digital
    [l_I]
                          ♦ NUME_ORDRE = numord           [l_I]
                          )
    ♦ MODELE_MESURE = _F ( ♦ MODEL = modelexp,           [modele_sdaster]
                          ♦ MEASUREMENT = my,           /
    [mode_meca]
                                                              / [dyna_harmo]
                          ♦ NOM_CHAM =/'DEPL'           [DEFECT]
                                                              / 'QUICKLY'
                                                              / 'ACCE'
                                                              / 'SIEF_NOEU'
                                                              / 'EPSI_NOEU'
                          ♦ NUME_MODE = digital
    [l_I]
                          ♦ NUME_ORDRE = numord           [l_I]
                          )
    ♦ RESOLUTION = _F ( ♦ METHOD =/'LU'                 [DEFECT]
                        / 'SVD'
    # If METHOD = 'SVD' then :
        ♦ EPS =/0.0           [DEFECT]
        / eps                 [R]
        ♦ REGUL =/'NOT'       [DEFECT]
        / 'NORM_MIN'
        / 'TIK_REL'
    # If REGUL! = 'NOT' then :
        ♦/COEF_PONDER = /0.           [DEFECT]
        /W                 [l_R]
        / COEF_PONDER_F = w_f         [l_fonction]
        ),
    ♦ NUME_DDL = num_ddl,           [nume_ddl]
    ♦ RESU_NX = res_nx,             [mode_meca]
    ♦ RESU_EX = res_ex,             / [mode_meca]
        / [dyna_harmo]
    ♦ RESU_ET = res_et,             / [mode_meca]
```

Code_Aster

Version
default

Titre : Macro-commande MACRO_EXPANS
Responsable : ANDRIAMBOLOLOŌNA Harinaivo

Date : 19/01/2011 Page : 3/5
Clé : U4.90.02 Révision :
b0c155132f2e

```
◇ RESU_RD = res_rd,
```

```
)
```

/ [dyna_harmo]

/ [mode_meca]

/ [dyna_harmo]

3 Operands

3.1 Keyword **MODELE_CALCUL**

◆ MODELE_CALCUL

Keyword factor gathering the whole of the keywords relating to the base of expansion, in general obtained by calculation (from where the name).

3.1.1 Keyword **MODEL**

◆ MODEL = modelnum

modele_sdaster indicating the model on which one will extend measurement

3.1.2 Keyword **BASE**

◆ BASE = bases

mode_meca being used as a basis for the expansion.

The base should not have vectors colinéaires, and the number of modes used must be lower than the number of DDL of measurement (preferably, $N_{\text{modes}} \ll N_{\text{my}}$) or else, the system to be solved is under-given, which can lead to a fatal error, and a stop of the code.

3.1.3 Keyword **NUME_ORDRE/NUME_MODE**

List of the sequence numbers or the modal positions of the modes which one wishes to use for the expansion.

3.2 Keyword **MODELE_MESURE**

◆ MODELE_MESURE

Keyword factor gathering the whole of the keywords relating to the experimental base which one wishes to extend

3.2.1 Keyword **MODEL**

◆ MODEL = modelexp

modele_sdaster indicating the model associated with the experimental grid. The knowledge of the nodes is in general enough to determine an experimental grid. The associated model can be then defined in the following way:

```
MODELEXP = AFFE_MODELE (GRID = MAIEXP,  
                        AFFE = _F (GROUP_MA = 'SENSORS',  
                                  PHENOMENON = 'MECHANICAL',  
                                  MODELING = 'DIS_T',),),);  
  
CAREXP = AFFE_CARA_ELEM (MODEL = MODELEXP,  
                        DISCRETE = _F (GROUP_MA = 'SENSORS',  
                                       REFERENCE MARK = 'TOTAL',  
                                       CARA = 'K_T_D_N',  
                                       VALE = (100.0, 100.0, 100.0,)),  
                        ),);
```

The value of the stiffnesses given is arbitrary, it is not useful in calculation.

NB: to use the operator `PROJ_CHAMP` in the macro one, one needs to generate a `nume_ddl` associated with this grid. For that, it is moreover necessary to assign one material to the model, to calculate the elementary matrices (rigidity for example) and to create classification with `NUME_DDL`.

3.2.2 Keyword MEASUREMENT

◆ `MEASUREMENT = my`
`dyna_harmo` or `mode_meca` to extend. These data are in general imported of a result of measurement (file `.unv`) with the operator `LIRE_RESU`.

3.2.3 Keyword NUME_ORDRE/NUMÉRIQUE_MODE

List of entreties. Allows to select the modes which one wishes to extend.

3.2.4 Keyword NOM_CHAM

◆ `NOM_CHAM = 'DEPL' ...`
Experimental size to extend.

3.3 Keyword RESOLUTION

Two techniques of resolution are proposed for the opposite problem: Truncated SVD, method LU. For the SVD, one can choose to truncate the singular values smallest to improve conditioning of the problem (choice of 'eps'), or to use a regularization of the Tikhonov type. One will be able to refer to the documents [U4.73.01] (Doc. of `PROJ_MESU_MODAL`) and [R6.03.01] (Doc. of reference on the decomposition in singular values).

3.4 Keyword NUME_DDL

Allows to force classification to be used for the operator `PROJ_CHAMP`. For more precision, to refer to the documentation of `PROJ_CHAMP` [U4.72.05].

3.5 Keywords RESU_XX

Keywords `RESU_XX` pre-déclarer names of the outgoing concepts allow:

- `RESU_NX` is the truncation of the digital base (keyword `BASE` under the keyword factor `MODELE_CALCUL`) with the modes chosen in `NUME_MODE`,
- `RESU_EX` is the truncation of the experimental base (keyword `MEASUREMENT` under the keyword factor `MODELE_MESURE`) with the modes chosen in `NUME_MODE`,
- `RESU_ET` is the result of the expansion,
- `RESU_RD` is the reprojction on the experimental model of the wide base: it is intéressaant to check if the reprojction of the wide result is comparable to the initial experimental data.