

Operator DEFI_BASE_MODAL

1 Goal

The goal of the operator is to define:

- the base of a dynamic under-structuring, when one cut out the complete structure in substructures;
- or bases it of a modal recombination, when the complete structure directly is treated.

The modal base obtained by this operator is of the type: 'CLASSICAL' if the modal base is made up of dynamic clean modes and of the static deformations calculated by the operator starting from a concept of the type `interf_dyna_clas` product by `DEFI_INTERF_DYNA` [U4.64.01]. The option `DIAG_MASS` allows to recompute a classification for the static modes so that the matrix of mass is diagonal.

The base is of type `RITZ` in the other cases.

The operator produces a concept of the type `mode_meca`.

Contents

1 Goal.....	1
2 Syntax.....	3
3 Operands.....	4
3.1 Keyword factor CLASSIC.....	4
3.1.1 Operand INTERF_DYNA.....	4
3.1.2 Operands MODE_MECA/NMAX_MODE.....	4
3.2 Keyword factor RITZ.....	4
3.2.1 Keyword MODE_MECA.....	4
3.2.2 Keyword MODE_INTF.....	5
3.2.3 Keyword BASE_MODALÉ.....	5
3.2.4 Operand NMAX_MODE.....	5
3.2.5 Operand INTERF_DYNA.....	5
3.2.6 Operand NUMÉ_REF.....	5
3.2.7 Operand LIST_AMOR.....	5
3.2.8 Operand ORTHO.....	5
3.2.9 Keyword MATRIX.....	5
3.3 Keyword factor DIAG_MASS.....	6
3.3.1 Keyword MODE_MECA.....	6
3.3.2 Keyword MODE_STAT.....	6
3.4 Keyword factor ORTHO_BASE.....	6
3.4.1 Keyword BASE.....	6
3.4.2 Keyword MATRIX.....	6
3.5 Keyword factor SOLVEUR.....	6
3.6 Operand TITLE.....	6
3.7 Operand INFORMATION.....	6
4 Examples.....	7
4.1 dynamic Under-structuring.....	7
4.2 Enrichment of a base of clean modes with static modes.....	7

2 Syntax

```
bamo [mode_meca] = DEFI_BASE_MODALÉ (

♦ / CLASSIC = _F ( ♦ INTERF_DYNA= intdyn, [interf_dyna_clas]
                  ♦ MODE_MECA = modes, [l_mode_meca]
                  ◇ NMAX_MODE = nbmodes, [l_I]
                  ),

/ RITZ = _F ( ♦ / MODE_MECA = modes [l_mode_meca]
              / MODE_INTF = modeintf, / [mode_meca]
              / BASE_MODALÉ= bamo, [mult_elas]
              / [mode_meca]
              ◇ NMAX_MODE = nbmodes, [l_I]
              )

/ DIAG_MASS = _F ( ♦ MODE_MECA = modes, [l_mode_meca]
                  ♦ MODE_STAT = modesta, [mode_meca]
                  ),

/ ORTHO_BASE = _F ( ♦ BASE = modes, / [mode_meca]
                   ♦ MATRIX = matrix, / [mult_elas]
                   [matr_asse_*]
                   ),

◇ INTERF_DYNA = intdyn, [interf_dyna_clas]

◇ NUME_REF = numddl, [nume_ddl]

◇ LIST_AMOR = listamor, [l_R8]

◇ ORTHO = / 'YES'
          / 'NOT' [DEFECT]
♦ MATRIX = matrix, [matr_asse_*]

◇ SOLVEUR = _F(see document [U4.50.01])

◇ TITLE = title, [l_Kn]

◇ INFORMATION = / 1, [DEFECT]
                / 2,

)
```

3 Operands

3.1 Keyword factor `CLASSIC`

- ◆ / `CLASSIC = _F (`
Keyword factor for the definition of a modal base of type 'CLASSICAL', when one carries out a calculation by dynamic under-structuring.

3.1.1 Operand `INTERF_DYNA`

- ◆ `INTERF_DYNA = intdyn`
Name of the concept of the type `interf_dyna_clas` product by `DEFI_INTERF_DYNA` [U4.64.01].

The operator calculates the static deformations corresponding to the various defined interfaces, while being based on the classification used for the calculation of the clean modes.

3.1.2 Operands `MODE_MECA/NMAX_MODE`

- ◇ `MODE_MECA = modes`
List of the concepts of the type `mode_meca` containing the clean modes of the structure.
- ◇ `NMAX_MODE = nbmodes`
List of entirities giving, for each concept `mode_meca`, the maximum number of modes suitable to retain. From $I^{\text{ème}}$ base modal list `modes`, one retains the modes then corresponding to `nbmodes [I]` the lowest frequencies.

In general, the length of the list of entirities, well informed under the keyword `NMAX_MODE`, must correspond to that of the list of the modal bases (keyword `MODE_MECA`). The only exception to this rule is when a single value is given in `NMAX_MODE`. In this case, the value of `NMAX_MODE` is used for each modal base.

If `NMAX_MODE` is not well informed, one takes to by default all the modes of each one of modal bases given under the keyword `MODE_MECA`.

3.2 Keyword factor `RITZ`

- ◆ / `RITZ = _F (`
Keyword allowing factor:
 - in the case of a calculation by under-structuring, to build a modal base of substructure of the type 'RITZ';
 - in the case of a calculation directly with the complete structure, to build an enriched modal base (for example: clean modes of vibration, which one enriches by static modes associated with external forces).

The base is made up starting from 2 occurrences of the keyword `RITZ` (one with the keyword `MODE_MECA` or `BASE_MODAL`, one other with the keyword `MODE_INTF`).

3.2.1 Keyword `MODE_MECA`

Name of the concept of the type `mode_meca` containing the clean modes of vibration of the structure, or substructure in the case of a calculation by under-structuring.

One can give a list of `mode_meca` obtained for the same structure (with boundary conditions different for example).

3.2.2 Keyword `MODE_INTF`

Name of the concept of the type `mode_meca` (produced by `CALC_MODES` [U4.52.02] or by `MODE_STATIQUE` [U4.52.14]) or `mult_elas` (produced by `MACRO_ELAS_MULT` [U4.51.02]) container:

- in the case of a calculation by under-structuring: the modes which one wants to use as modes of interface of under - structure;
- in the case of a calculation directly on the complete structure: modes which enrich the base by clean modes of vibration given under the keyword `MODE_MECA` precedent. That can for example be static deformations associated with external forces (known imposed force; force of shock associated with an obstacle; etc).

3.2.3 Keyword `BASE_MODAL`

Name of concept of the type `mode_meca` product by a preceding call of the operator of `DEFI_BASE_MODAL` [U4.64.02]. It can be entered only at the time of the first occurrence of the keyword `RITZ`. Second occurrence of the keyword `RITZ` obligatorily the keyword will contain then `MODE_INTF`. The name of concept `mode_meca` result of the operator can be different from this one or identical (it is then réentrant).

3.2.4 Operand `NMAX_MODE`

Many modes to be retained in the dynamic modes (or statics) given by one of the preceding keywords under the occurrence of the keyword `RITZ`. If one informs a list of `mode_meca`, it is necessary to give either a list of the same size for the numbers of modes to retain, or a single value (in this case, this value is applied to all the concepts `mode_meca`).

If `NMAX_MODE` is not informed, one takes to by default all the modes.

3.2.5 Operand `INTERF_DYNA`

Dynamic interface of the substructure (to be informed if required and only if the keyword factor is used `'RITZ'`).

3.2.6 Operand `NUME_REF`

Classification of reference on which all fields of displacement (dynamic modes and statics) constituting the base of `'RITZ'` will be reordered.

3.2.7 Operand `LIST_AMOR`

List of modal depreciation that the user can provide to enrich the modes declare under the keyword `MODE_MECA`. That amounts adding the depreciation reduced for these same modes if in the beginning they are real modes. This option is useful to simulate experimental results.

3.2.8 Operand `ORTHO`

Operand allowing to make a orthonormalisation of the base of Ritz (be informed if this D-orthonormalisation is wished and only if the keyword factor is used `'RITZ'`). This orthonormalisation is made with an algorithm of iterative the Graam-Schmidt type (IGS) according to the version of Kahan-Parlett.

3.2.9 Keyword `MATRIX`

Name of the concept of the type `matr_asse_*` who will be taken into account for the scalar products at the time of the reorthonormalisation of the base of `RITZ`. It is an obligatory keyword if `ORTHO='OUI'`.

3.3 Keyword factor **DIAG_MASS**

◆ / DIAG_MASS = _F (

Keyword allowing to recompute the static modes by eliminating the dynamic contribution and while proceeding to a orthogonalisation of Graam-Schmidt.

3.3.1 Keyword **MODE_MECA**

Name of the concept of the type `mode_meca` containing the dynamic clean modes of the treated substructure.

3.3.2 Keyword **MODE_STAT**

Name of the concept of the type `mode_meca` product by the operator `MODE_STATIQUE` [U4.52.14] which contains the static modes.

3.4 Keyword factor **ORTHO_BASE**

◆ / ORTHO_BASE = _F (

Allowing keyword of orthonormaliser the modes of a base This orthonormalisation is made with an algorithm of iterative the Graam-Schmidt type (IGS) according to the version of Kahan-Parlett.

3.4.1 Keyword **BASE**

Name of the concept of the type `mode_meca` or `mult_elas` containing dynamic clean modes.

3.4.2 Keyword **MATRIX**

Name of the concept of the type `matr_asse_*` who contains will be taken into account for the scalar products at the time of the reorthogonalisation.

3.5 Keyword factor **SOLVEUR**

◇ SOLVEUR = _F (...)

This keyword factor is optional: it makes it possible to choose another solvor of resolution of system. In the case of this order, the syntax of the keyword is restricted with two methods: one can choose between the method by default, `MULT_FRONT`, and methods `LDLT` or `MUMPS`. Syntax being common to several orders, please consult the handbook [U4.50.01].

3.6 Operand **TITLE**

◇ TITLE = title

Title of the concept created.

3.7 Operand **INFORMATION**

◇ INFORMATION =

Level of the furnished information in the file `MESSAGE` :

- 1 pas d' impression;
- 2 writing of the general information (concepts upstream, basic standard).

4 Examples

4.1 dynamic Under-structuring

Various modelings of CAS-test SDLS106 present implementation the complete of a calculation by under-structuring, with various methods (Craig-Bampton, Mc Neal, compatible grids or not, etc).

4.2 Enrichment of a base of clean modes with static modes

```
# calculation of the assembled matrices and the classification of the DDL
...

# calculation of the clean modes of vibration
modesvib = CALC_MODES (MATR_RIGI=K_ASSE,
                      MATR_MASS=M_ASSE,
                      CALC_FREQ=_F (NMAX_FREQ = nb_freq),
                      );

# application of an external force
force = AFFE_CHAR_MECA (MODELE=modele,
                       FORCE_NODALE=_F (NODE = 'N11',
                                         FX=800., FY = -1000.)) ;

# calculation of the static deformation associated with this external force
modestat = MODE_STATIQUE (MATR_RIGI=K_ASSE,
                          MATR_MASS=M_ASSE,
                          FORCE_NODALE=_F (NOEUD=' N11',
                                           AVEC_CMP= ('DX', 'DY')),
                          );

# construction of the enriched modal base
basemode = DEFI_BASE_MODALÉ (RITZ = (_F (MODE_MECA=modesvib),
                                       _F (MODE_INTF=modestat),
                                       ),
                             NUME_REF=NUMDDL,
                             );
```

The SDNL104c CAS-test presents implementation the complete of this technique.

Another example is given by the SDNL301a CAS-test, where the static deformations which enrich the base by the clean modes, are those related to external forces generated by shocks.