Macro order PROJ_BASE

1 Goal

To project matrices and/or vectors assembled on a modal basis or a basis of RITZ. The projected matrices and vectors results will be used by the calculation algorithms in generalized components (DYNA_TRAN_MODAL [U4.53.21] for example).

This macro-order replaces the following sequence controls:

- NUME_DDL_GENE [U4.65.03] which establishes the classification of the generalized degrees of freedom,
- one or more occurrences of PROJ_MATR_BASE [U4.63.12] to project one or more assembled matrices,
- one or more occurrences of PROJ_VECT_BASE [U4.63.13] to project one or more assembled vectors.

Contents

2 Syntax

```
PROJ_BASE (
      ♦  BASE   =  Ba,                                  [mode_meca]
                                                        [mode_gene]
      ◊  NB_VECT   =  Nm,                               [I]

      ◊  STORAGE   =  /  'FULL',                        [defect]
                      /  'DIAG',

      ◊  NUME_DDL_GENE    =  /numgen                    [nume_ddl_gene]
                             /CO ('numgen'),            [nume_ddl_gene]

      ◊  MATR_ASSE_GENE   =  _F (
                             ♦  MATRIX =  CO ('MT'),    [matr_asse_gene_r]
                             ♦  /  MATR_ASSE =  my,     [matr_asse_DEPL_r]
                                /  MATR_ASSE_GENE  =  my,[matr_asse_gene_r]
                                ),

      ◊  VECT_ASSE_GENE   =  _F (
                             ♦  VECTOR =  CO ('vt'),    [vect_asse_gene]
                             ◊  TYPE_VECT =  /  'FORC',  [defect]
                                             /  typ,    [KN]
                             ♦  /  VECT_ASSE =  goes,      [cham_no_depl_r]
                                /  VECT_ASSE_GENE  =  goes,[vect_asse_gene]
                                ),

      ◊  RESU_GENE        =  _F (
                             ♦  RESULT =  CO ('LMBO'), [resu_gene]
                             ◊  TYPE_VECT =  /  'FORC',  [defect]
                                             /  typ,    [KN]
                             ♦  /  RESU     =  goes,      [dyna_trans]
                                ),

      ◊  INFORMATION  =  /  1,                          [defect]
                         /  2,
            )
```

3 Operands

3.1 Operand BASE

♦ BASE = Ba

Concept of the type mode_meca or mode_gene (for the under-structuring), which contains the vectors defining the subspace of projection.

3.2 Operand NB_VECT

◊ NB_VECT = Nm

Many vectors used in the base (one take them Nm first). It is checked that the number Nm is quite lower than the number of vectors of the base, in the contrary case, one uses all the provided vectors.

3.3 Operand STORAGE

◊ STORAGE = /'FULL' [DEFECT]
/'DIAG'

Confer NUME_DDL_GENE [U4.65.03].

If a matrix presents a profile 'DIAG' and a another profile 'FULL', two classifications will be created with NUME_DDL_GENE.

The use keyword STOCKAGE=' DIAG' is licit when the base on which the matrices are projected is made up of clean modes. In this case, the projected matrices are indeed diagonal, and it is not necessary to save the other terms of the matrix, which are worthless.

Attention, if the base is made up of other types of vectors (of the static modes for example), then the use of the keyword STOCKAGE=' DIAG' conduit with false results.
In the case of calculations with use of the operators of fluid-structure, calculation must be made with the option of storage diagonal.

3.4 Operand NUME_DDL_GENE

◊ NUME_DDL_GENE = /numgen [nume_ddl_gene]
/CO ('numgen'), [nume_ddl_gene]

Classification associated with the model generalized. This operand can be a concept already existing or not (CO ('numgen')).

3.5 Keyword MATR_ASSE_GENE

◊ MATR_ASSE_GENE

Keyword factor defining the name of the projected matrix result and the name of the matrix to be projected. This keyword must be repeated as many times as there are matrices to project.

3.5.1 Operand MATRIX

♦ MATRIX = CO ('MT')

Concept of the type matr_asse_gene_R, generalized matrix result.

3.5.2 Operands MATR_ASSE/MATR_ASSE_GENE

♦ / MATR_ASSE = my

Concept of the type matr_asse_DEPL_R, assembled matrix which one wishes to project.

/ MATR_ASSE_GENE = my

Concept of the type matr_asse_gene_R, matrix assembled resulting from under - structuring, which one wishes to project.

3.6 Keyword VECT_ASSE_GENE

◊ VECT_ASSE_GENE

Keyword factor defining the name of the vector result project and the name of the vector to be projected. This keyword must be repeated as many times as there are vectors to project.

3.6.1 Operand VECTOR

♦ VECTOR = CO ('vt')

Concept of the type vect_asse_gene, vector generalized result.

3.6.2 Operand TYPE_VECT

◊ TYPE_VECT = typ

Character string describing the type of the field represented by the assembled vector, by default one expects a field of the type forces 'FORC', the other possibilities are 'DEPL', 'QUICKLY' and 'ACCE'. The treatment is different according to whether the option is used FORC or others.

- With the option FORC, simple projection is carried out $\Phi^T f$, where Φ is the base of modes and f effort,
- With the other options, one calculates by problem reverses the modal coefficients of participation associated with a displacement given. It is supposed that one can write displacement x in the form $x=\eta^T\Phi$. One calculates then $\eta=\Phi^T(\Phi^T\Phi)^{-1}x$ (pseudo-opposite of Moore-Penrose).

3.6.3 Operands VECT_ASSE/VECT_ASSE_GENE

♦ / VECT_ASSE = goes

Concept of the type cham_no_DEPL_R, assembled vector which one wishes to project.

/ VECT_ASSE_GENE = goes

Concept of the type vect_asse_gene, assembled vector resulting from the under-structuring, which one wishes to project.

3.7 Keyword RESU_GENE

◊ VECT_ASSE_GENE

Allows to project a structure of data result of the type dyna_trans (exit of a linear calculation of dynamics, or reading of a data file). This keyword must be repeated as many times as there are vectors to project.

3.7.1 Operand RESULT

♦ RESULT = CO ('LMBO')

Concept of the type resu_gene, vector generalized result.

3.7.2 Operand TYPE_VECT

◊ TYPE_VECT = typ

Character string describing the type of the field represented by the assembled vector, by default one expects a field of the type forces 'FORC', the other possibilities are 'DEPL', 'QUICKLY' and 'ACCE'. The treatment is different according to whether the option is used FORC or others.

- With the option FORC, simple projection is carried out $\Phi^T f$, where Φ is the base of modes and f effort
- With the other options, one calculates by problem reverses the modal coefficients of participation associated with a displacement given. It is supposed that one can write displacement x in the form $x=\eta^T \Phi$. One calculates then $\eta=\Phi^T(\Phi^T\Phi)^{-1}x$ (pseudo-opposite of Moore-Penrose).

3.7.3 Operands RESU

♦ / RESU= LMBO

Concept of the type dyna_trans, structure of data result which one wishes to project.

3.8 Operand INFORMATION

◊ INFORMATION = / 1 [DEFECT]
/ 2

Level of impression of information for the order NUME_DDL_GENE (confer [U4.65.03]).

4 Example of use

```
# dynamic transitory on modal basis system masses and arises
PROJ_BASE (       BASE=MODES,
                  MATR_ASSE_GENE= (
                       _F (MATRIX = CO ('MASSEGEN'),
                          MATR_ASSE = MATRMASS),
                       _F (MATRIX = CO ('RIGIDGEN'),
                          MATR_ASSE = MATRRIGI),
                       _F (MATRIX = CO ('AMORTGEN'),
                          MATR_ASSE = MATRAMOR,
                          STORAGE = 'FULL')),
                  VECT_ASSE_GENE=
                       _F (VECTOR = CO ('EFFOGENE'),
                          VECT_ASSE = VECTASS))
                  );
```