

Manuel d'Utilisation**Fascicule U4.6- : Matrices/Vecteurs élémentaires et assemblage****Document : U4.63.11**

Macro commande `MACRO_PROJ_BASE`

1 But

Projeter des matrices et/ou vecteurs assemblés sur une base modale ou sur une base de RITZ. Les matrices et vecteurs projetés résultats seront utilisés par les algorithmes de calcul en composantes généralisées (`DYNA_TRAN_MODAL` [U4.53.21] par exemple).

Cette macro-commande se substitue aux commandes successives suivantes :

- `NUME_DDL_GENE` [U4.65.03] qui établit la numérotation des degrés de liberté généralisés,
- une ou plusieurs occurrences de `PROJ_MATR_BASE` [U4.63.12] pour projeter une ou plusieurs matrices assemblées,
- une ou plusieurs occurrences de `PROJ_VECT_BASE` [U4.63.13] pour projeter un ou plusieurs vecteurs assemblés.

2 Syntaxe

MACRO_PROJ_BASE

```
(
  ♦ BASE = ba,
  [mode_meca]
  [base_modale]
  [mode_gene]

  ♦ NB_VECT = / 9999,
               / nm,
  [DEFAULT]
  [I]

  ♦ PROFIL = / 'DIAG',
               / 'PLEIN',
  [DEFAULT]

  ♦ MATR_ASSE_GENE =_F( ♦ MATRICE = CO('mt'),
                        ♦ / MATR_ASSE = ma,
                        / MATR_ASSE_GENE = ma,
                        ),
  [matr_asse_gene_R]
  [matr_asse_DEPL_R]
  [matr_asse_gene_R]

  ♦ VECT_ASSE_GENE =_F (♦ VECTEUR = CO('vt'),
                        ♦ TYPE_VECT = / 'FORC',
                        / typ,
                        ♦ / VECT_ASSE = va,
                        / VECT_ASSE_GENE = va,
                        ),
  [vect_asse_gene]
  [DEFAULT]
  [Kn]
  [cham_no_DEPL_R]
  [vect_asse_gene]

  ♦ INFO = / 1,
            / 2,
  [DEFAULT]
)
```

3 Opérandes

3.1 Opérande `BASE`

- ◆ `BASE = ba`

Concept du type `mode_meca`, `base_modale` ou `mode_gene` (pour la sous-structuration), qui contient les vecteurs définissant le sous-espace de projection.

3.2 Opérande `NB_VECT`

- ◇ `NB_VECT = nm`

Nombre de vecteurs utilisés dans la base (on prend les `nm` premiers). On vérifie que le nombre `nm` est bien inférieur au nombre de vecteurs de la base, dans le cas contraire (`nm = 9999`) on utilise tous les vecteurs fournis.

3.3 Opérande `PROFIL`

- ◇ `PROFIL = / 'DIAG' [DEFAULT]
 / 'PLEIN'`

Cf. `NUME_DDL_GENE` [U4.65.03].

Si une matrice présente un profil '`DIAG`' et une autre un profil '`PLEIN`', deux numérotations seront créées avec `NUME_DDL_GENE`.

3.4 Mot clé `MATR_ASSE_GENE`

- ◇ `MATR_ASSE_GENE`

Mot clé facteur définissant le nom de la matrice projetée résultat et le nom de la matrice à projeter. Ce mot clé doit être répété autant de fois qu'il y a de matrices à projeter.

3.4.1 Opérande `MATRICE`

- ◆ `MATRICE = CO('mt')`

Concept du type `matr_asse_gene_R`, matrice généralisée résultat.

3.4.2 Opérandes `MATR_ASSE` / `MATR_ASSE_GENE`

- ◆ `/ MATR_ASSE = ma`

Concept du type `matr_asse_DEPL_R`, matrice assemblée que l'on souhaite projeter.

- `/ MATR_ASSE_GENE = ma`

Concept du type `matr_asse_gene_R`, matrice assemblée issue de la sous-structuration, que l'on souhaite projeter.

3.5 Mot clé VECT_ASSE_GENE

◇ VECT_ASSE_GENE

Mot clé facteur définissant le nom du vecteur projeté résultat et le nom du vecteur à projeter. Ce mot clé doit être répété autant de fois qu'il y a de vecteurs à projeter.

3.5.1 Opérande VECTEUR

◆ VECTEUR = CO('vt')

Concept du type vect_asse_gene, vecteur généralisé résultat.

3.5.2 Opérande TYPE_VECT

◇ TYPE_VECT = typ

Chaîne de caractères décrivant le type du champ représenté par le vecteur assemblé, par défaut on attend un champ de type force 'FORC', les autres possibilités sont 'DEPL', 'VITE' et 'ACCE'.

3.5.3 Opérandes VECT_ASSE / VECT_ASSE_GENE

◆ / VECT_ASSE = va

Concept du type cham_no_DEPL_R, vecteur assemblé que l'on souhaite projeter.

/ VECT_ASSE_GENE = va

Concept du type vect_asse_gene, vecteur assemblé issu de la sous-structuration, que l'on souhaite projeter.

3.6 Opérande INFO

◇ INFO = / 1 [DEFAULT]
 / 2

Niveau d'impression d'informations pour la commande NUME_DDL_GENE (cf. [U4.65.03]).

4 Exemple d'utilisation

dynamique transitoire sur base modale système masse et ressort

```
MACRO_PROJ_BASE(  BASE=MODES,
                  MATR_ASSE_GENE=(
                    _F(
                      MATRICE = CO('MASSEGEN'),
                      MATR_ASSE = MATRMASS),
                    _F(
                      MATRICE = CO('RIGIDGEN'),
                      MATR_ASSE = MATRRIGI),
                    _F(
                      MATRICE = CO('AMORTGEN'),
                      MATR_ASSE = MATRAMOR,
                      PROFIL = 'PLEIN')),
                  VECT_ASSE_GENE=
                    _F(
                      VECTEUR = CO('EFFOGENE'),
                      VECT_ASSE = VECTASS))
```

séquence des commandes équivalentes à la macro-commande MACRO_PROJ_BASE :

```
_0000000 = NUME_DDL_GENE (  BASE=MODES,
                             STOCKAGE='DIAG'      )
MASSEGEN  = PROJ_MATR_BASE (  BASE=MODES,
                             NUME_DDL_GENE=_0000000,
                             MATR_ASSE=MATRMASS   )
RIGIDGEN  = PROJ_MATR_BASE (  BASE=MODES,
                             NUME_DDL_GENE=_0000000,
                             MATR_ASSE=MATRRIGI   )
_0000001  = NUME_DDL_GENE (  BASE=MODES,
                             STOCKAGE='PLEIN'     )
AMORTGEN  = PROJ_MATR_BASE (  BASE=MODES,
                             NUME_DDL_GENE=_0000001,
                             MATR_ASSE=MATRAMOR    )
EFFOGENE  = PROJ_MATR_BASE (  BASE=MODES,
                             NUME_DDL_GENE=_0000001,
                             VECT_ASSE=VECTASS     )
```

PAGE LAISSEE BLANCHE INTENTIONNELLEMENT