

**Manuel d'Utilisation**  
**Fascicule U7.0- : Echanges de données**  
**Document : U7.03.21**

## Procédure MACRO\_MADMACS

---

### 1 But

---

Impression au format IDEAS des données nécessaires au chaînage *Code\_Aster* - MADMACS.

Le logiciel MADMACS permet essentiellement :

- l'ajustement des modèles dynamiques sur des données d'essais,
- la comparaison de résultats de calcul avec des résultats d'essais,
- l'analyse du comportement dynamique des structures.

Les données imprimées sur le fichier au format IDEAS sont :

- le maillage,
- la définition d'une interface,
- des données de conditions aux limites (DDL\_IMPO, LIAISON\_DDL),
- les matrices élémentaires ou les matrices assemblées du modèle,
- les modes dynamiques,
- les modes statiques d'interface,
- des matrices et paramètres modaux (masse et rigidité condensées à l'interface, masse et rigidité généralisées, facteurs de participation).

## 2 Syntaxe

```
MACRO_MADMACS (

    ◇ FICHIER      :  nom_fich          [Kn]

    ◇ FORMAT       :  'IDEAS'           [DEFAULT]

    ◇ VERSION      :  5                  [DEFAULT]

    ◆ MAILLAGE     :  mail               [maillage]

    ◆ NUME_DDL     :  nu                  [nume_ddl]

    ◆ CHARGE       :  lcharge             [l_charge]

    ◆ / MATR_ELEM_RIGI : kel              [matr_elem_DEPL_R]
      / MATR_RIGI      : k                [matr_asse_DEPL_R]

    ◆ / MATR_ELEM_MASS : mel              [matr_elem_DEPL_R]
      / MATR_MASS      : m                [matr_asse_DEPL_R]

    ◇ / MATR_ELEM_AMOR : cel              [matr_elem_DEPL_R]
      / MATR_AMOR      : c                [matr_asse_DEPL_R]

    ◆ MODE_MECA    :  mode                [mode_meca]

    ◇ NMAX_MODE    :      / 10            [DEFAULT]
                      / nbm             [I]

    ◇ INTERFACE    :  (

                                ◇ NOM : nom_int          [Kn]

                                ◆ / NOEUD : l_no          [l_noeud]
                                  / GROUP_NO : l_grno      [l_gr_noeud]

                                ◆ / DDL_ACTIF : l_ddl       [l_Kn]
                                  / MASQUE : l_ddl         [l_Kn]
                                )

    );
```

### 3 Opérandes

La procédure MACRO\_MADMACS réalise automatiquement l'enchaînement de plusieurs commandes suivant la séquence :

```
DEFI_INTERF_DYNA [U4.64.01]
DEFI_BASE_MODELE [U4.64.02]
MACR_ELEM_DYNA [U4.64.01]
IMPR_RESU (pour le maillage) [U7.05.01]
IMPR_CHARGE [U7.04.31]
IMPR_MATRICE [U7.04.32]
IMPR_RESU (pour les modes) [U7.05.01]
IMPR_MACR_ELEM [U7.04.33]
```

#### 3.1 Opérande FICHIER

◇ FICHIER : nom\_fich [Kn]

Nom du fichier dans lequel on souhaite voir imprimer les données à transférer. Ce nom de fichier doit avoir été préalablement défini par les procédures DEBUT [U4.11.01] ou POURSUITE [U4.11.03].

#### 3.2 Opérande FORMAT

◇ FORMAT : 'IDEAS' [DEFAULT]

Format sous lequel on souhaite imprimer les données. Seul le format par défaut 'IDEAS' est supporté pour le moment.

#### 3.3 Opérande VERSION

◇ VERSION : 5 [DEFAULT]

Numéro de la version d'IDEAS correspondant aux format du fichier utilisé pour imprimer les données. Seule la version 5 (valeur par défaut) est supportée pour le moment.

#### 3.4 Opérande MAILLAGE

◆ MAILLAGE : mail

Nom du concept `maillage` qu'on souhaite transférer vers le logiciel MADMACS. A l'instar de la commande IMPR\_RESU [U7.05.02], la numérotation des noeuds est celle du *Code\_Aster* sauf lorsque le maillage est issu de IDEAS auquel cas on utilise la numérotation IDEAS.

#### 3.5 Opérande NUME\_DDL

◆ NUME\_DDL : nu

Nom du concept `nume_ddl` du modèle qu'on souhaite transférer vers le logiciel MADMACS.

### 3.6 Opérande CHARGE

- ◆ CHARGE : lcharge

Nom du concept de type `charge` ou liste de `charge` appliqué au modèle à transférer. Seuls les chargements de type déplacements imposés nuls ou degrés de liberté liés du type `ax1 + bx2 = 0` pourront être pris en compte par MADMACS.

### 3.7 Opérandes MATR\_ELEM\_RIGI / MATR\_RIGI

- ◆ / MATR\_ELEM\_RIGI : kel

- / MATR\_RIGI : k

Nom des matrices élémentaires de rigidité (type `matr_elem_DEPL_R`) ou de la matrice assemblée de rigidité (type `matr_asse_DEPL_R`) qu'on désire transférer dans MADMACS.

### 3.8 Opérandes MATR\_ELEM\_MASS / MATR\_MASS

- ◆ / MATR\_ELEM\_RIGI : mel

- / MATR\_RIGI : m

Nom des matrices élémentaires de masse (type `matr_elem_DEPL_R`) ou de la matrice assemblée de masse (type `matr_asse_DEPL_R`) qu'on désire transférer dans MADMACS.

### 3.9 Opérandes MATR\_ELEM\_AMOR / MATR\_AMOR

- ◇ / MATR\_ELEM\_AMOR : cel

- / MATR\_AMOR : c

Nom des matrices élémentaires d'amortissement (type `matr_elem_DEPL_R`) ou de la matrice assemblée d'amortissement (type `matr_asse_DEPL_R`) qu'on désire transférer dans MADMACS.

### 3.10 Opérandes MODE\_MECA / NMAX\_MODE

- ◆ MODE\_MECA : mode

Nom des modes propres dynamiques (concept de type `mode_meca`) calculés avec le *Code\_Aster* et qu'on souhaite transférer dans MADMACS.

- ◇ NMAX\_MODE : mode

Nombre maximum de modes à transférer dans le fichier d'interface, 10 par défaut.

### 3.11 Mot clé INTERFACE

Mot clé facteur définissant l'interface qui permet le calcul des modes statiques et des matrices de masse et de rigidité condensées. Le logiciel MADMACS n'autorise la prise en compte que d'une seule interface (voir aussi la commande `DEFI_INTERF_DYNA` [U4.64.02]).

### 3.11.1 Opérande NOM

◇ NOM : *nom\_int*  
Nom de l'interface.

### 3.11.2 Opérande NOEUD / GROUP\_NO

◆ / NOEUD : *l\_no*  
/ GROUP\_NO : *l\_grno*  
Ensemble des nœuds ou des groupes de nœuds qui vont servir de support à l'interface.

### 3.11.3 Opérandes DDL\_ACTIF / MASQUE

◆ / DDL\_ACTIF : *l\_ddl*  
Liste des degrés de liberté actifs de l'interface c'est-à-dire les degrés de liberté où seront calculés les modes statiques.

/ MASQUE : *l\_ddl*  
Liste des degrés de liberté de l'interface qui ne doivent pas engendrer de déformée statique (ddls masqués). Ce sont les degrés de liberté complémentaires aux degrés de liberté actifs.

## 4 Exemple

Pour illustrer l'utilisation de la procédure *MACRO\_MADMACS*, on donne l'exemple de l'écriture dans un fichier au format IDEAS des données nécessaires au logiciel *MADMACS*, de deux manières différentes : d'une part en utilisant la procédure *MACRO\_MADMACS*, d'autre part en réalisant la suite de procédures qui sont effectivement appelées par la macro *MACRO\_MADMACS*.

L'exemple présenté est un système à deux ressorts et deux masses ponctuelles, dont l'une se déplace suivant l'axe OX et l'autre se déplace suivant une droite du plan OXY qui fait un angle de 45° dans le sens trigonométrique par rapport à l'axe OX. L'interface pour le calcul des modes statiques est réduite au degré de liberté suivant X du nœud NO1.

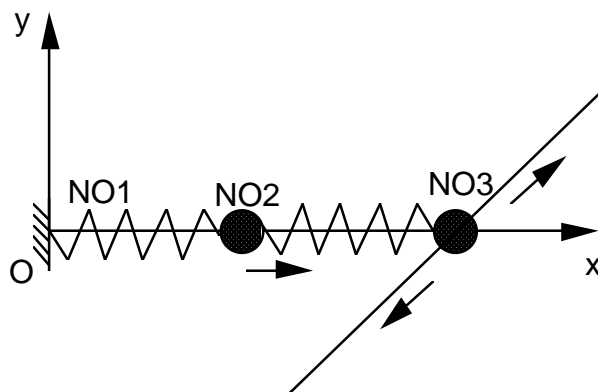


Schéma du cas test à 2 masses et 2 ressorts

Titre : Procédure MACRO\_MADMACS  
Auteur(s) : L. BILLET, L. VIVAN

Date : 19/09/00  
Clé : U7.03.21-C Page : 6/8

On donne le maillage correspondant à cet exemple.

```
COOR_3D
  NO1      0.    0.    0.
  NO2      0.1    0.    0.
  NO3      0.2    0.    0.
FINSF
%
SEG2
  S1      NO1      NO2
  S2      NO2      NO3
FINSF
%
GROUP_MA NOM=RESSORT
  S1 S2
FINSF
%
GROUP_NO NOM=MASSE
  NO2 NO3
FINSF
%
FIN
```

On donne également le fichier de commande réalisant le chaînage entre le *Code\_Aster* et MADMACS

```
%
DEBUT ();
mailla = LIRE_MAILLAGE( UNITE : 20 ) ;
%
modele  = AFFE_MODELE (MAILLAGE: mailla
                      AFFE      : (GROUP_MA      : RESSORT
                                   PHENOMENE      : 'MECANIQUE'
                                   MODELISATION: 'DIS_TR' )
                      AFFE      : (GROUP_NO      : MASSE
                                   PHENOMENE      : 'MECANIQUE'
                                   MODELISATION: 'DIS_T' )
                      );
%
param   = AFFE_CARA_ELEM (MODELE: modele
                        DISCRET : (GROUP_MA: RESSORT
                                   CARA    : 'K_TR_D_L'
                                   VALE    : (10000. 0. 0. 0. 0. 0. )
                                   )
                        DISCRET : (GROUP_NO: MASSE
                                   CARA    : 'M_T_D_N'
                                   VALE    : (1. )
                                   )
                        );
%
% ----- DEFINITION DU CHARGEMENT
%
charge_1 = AFFE_CHAR_MECA (MODELE: modele
                        DDL_IMPO: (TOUT: 'OUI' DZ:0.0 DRX:0. DRY:0. DRZ:0.)
                        DDL_IMPO: (NOEUD:NO1 DX:0.0 DY:0.0 )
                        LIAISON_DDL: (NOEUD: (NO2 NO2) DDL: ( 'DX' 'DY' )
                                      COEF_MULT: (1.0 -1.0)
                                      COEF_IMPO: 0.0)
                        DDL_IMPO: (NOEUD:NO3 DY:0.0 ) );
%
```

Titre :       *Procédure MACRO\_MADMACS*  
Auteur(s) :   **L. BILLET, L. VIVAN**

Clé :    U7.03.21-C    Date :    19/09/00  
Page :   7/8

```
%
% ----- MATRICES ET VECTEURS ELEMENTS FINIS
%
merime_l = CALC_MATR_ELEM (MODELE      : modele
                           CHARGE       : charge_l
                           CARA_ELEM    : param
                           OPTION       : 'RIGI_MECA' );

%
memame_l = CALC_MATR_ELEM (MODELE      : modele
                           CHARGE       : charge_l
                           CARA_ELEM    : param
                           OPTION       : 'MASS_MECA' );

%
% ----- NUMEROTATION DU PROBLEME ELEMENTS FINIS
%
numddl_l = NUME_DDL ( MATR_RIGI: merime_l
%%                STOCKAGE :  'MORSE'
                    ) ;

%
% ----- ASSEMBLAGE DES MATRICES ET VECTEURS
%
raid_l    = ASSE_MATRICE (MATR_ELEM: merime_l
                        NUME_DDL : numddl_l);

%
mass_l    = ASSE_MATRICE (MATR_ELEM: memame_l
                        NUME_DDL : numddl_l);

%
% ----- CALCUL DES MODES PROPRES DES SS
%
modes_l = MODE_ITER_SIMULT( MATR_A   : raid_l   MATR_B: mass_l
                           METHODE  : 'JACOBI'
                           CALC_FREQ: (NMAX_FREQ: 2 )
                           ) ;

%-----

MACRO_MADMACS ( MAILLAGE   : mailla

                NUME_DDL   : numddl_l

                CHARGE     : charge_l

                MATR_ELEM_RIGI : merime_l
                MATR_ELEM_MASS : memame_l

                MODE_MECA   : modes_l
                NMAX_MODE   : 1

                INTERFACE : ( NOM       : 'TERRE'
                              NOEUD    : NO1
                              MASQUE   : ('DY' 'DZ' 'DRX' 'DRY' 'DRZ')
                              )

                ) ;

FIN( ) ;
```

On donne enfin la séquence de commandes équivalente à la commande MACRO\_MADMACS.

```
interdyn = DEFI_INTERF_DYNA ( NUME_DDL : numddl_1
                                INTERFACE ( NOM      : 'TERRE'
                                             TYPE     : 'CRAIGB'
                                             NOEUD    : NO1
                                             MASQUE   : ( 'DY' 'DZ' 'DRX'
                                                         'DRY' 'DRZ' )
                                )
                                );
bamo = DEFI_BASE_MODEALE ( CLASSIQUE : ( INTERF_DYNA : interdyn
                                         MODE_MECA    : modes_1
                                         NMAX_MODE    : 1
                                         )
                                );
maelem = MACR_ELEM_DYNA ( BASE_MODEALE : bamo
                          OPTION      : 'CLASSIQUE'
                          );

IMPR_RESU      ( RESU ( FORMAT      : 'IDEAS'
                        MAILLAGE   : mailla
                        )
                );

IMPR_CHARGE    ( CHARGE : charge_1
                );

IMPR_MATRICE   ( MATR_ELEM ( MATRICE : merime_1 )
                MATR_ELEM ( MATRICE : memame_1 )
                );

IMPR_RESU      ( RESU ( FORMAT : 'IDEAS'
                        RESU : (RESULTAT : modes_1
                                NUME_MODE : ( 1 )
                                )
                        )
                );

IMPR_MACR_ELEM ( MACR_ELEM_DYNA : maelem
                );
```