Macro-command MACR_SPECTRE

1 Purpose

This macro-command produces a fast and efficient post-treatment of floor spectra of any building, after a seismic dynamical analysis.
It can be used for instance after building seismic transient dynamical calculations, where the connection between soil and foundation raft is idealised by soil stiffness (`DYNA_TRAN_MODAL` [U4.53.21], transient dynamics on reduced modal basis, calculated in the relative frame, produced concept: `resu_gene`) or after calculations using the *Code_Aster* - *ProMiss3D* software coupling (`LIRE_ MISS3D` [U7.02.31], produced concept: `dyna_trans`).

This macro-command executes successively:

1) extraction of the relative acceleration of one or more nodes of the finite element mesh in the `resultat` concept (`RECU_FONCTION` [U4.32.03]) ;
2) combination with the ground acceleration to get the absolute acceleration (`CALC_FONCTION` [U4.32.04]) ;
3) response acceleration spectra with several reduced damping coefficients (`CALC_FONCTION` [U4.32.04]) ;
4) envelope relative displacement function calculation corresponding to a specified floor to get the floor spectrum (optional).

We can consult test-cases sdll138a (post-treatment of a `dyna_tran_modal` analysis), or the tests miss01, miss05 with calculations using the *Code_Aster* - *ProMiss3D* software coupling, in order to get practical examples.

It produces a data structure of the type `table_sdaster` .

2 Syntax

```
table_sdaster = MACR_SPECTRE (
   ## response spectra calculation, seismic post-treatment
      ♦  MAILLAGE  =  ma                                         [maillage]
      ♦  PLANCHER = _F(
                          ♦      NOM          = floor            [Kn]
                          ◊   /  NOEUD        = l_no             [l_noeud]
                              /  GROUP_NO     = l_gno            [l_gr_noeud]
                           )
      ♦  CALCUL =      /   'ABSOLU'
                        /  'RELATIF'
      ♦  NOM_CHAM =    /   'ACCE'
                        /  'DEPL'
   ## IF NOM_CHAM ='ACCE' :
         ♦  AMOR_SPEC    =    l_amor                             [l_R]
         ◊  LIST_INST    =    linst                              [listr8]
         ◊  /  FREQ      =    l_fr                               [l_R]
            /  LIST_FREQ =    lfreq                              [listr8]
          ♦ NORME        =       r                               [R]
          ♦ RESU = _F (
                           ♦ /    ' RESU_GENE ' = tg             [ tran_gene ]
                              /   ' RESULTAT '  = resu           [dyna_trans,
evol_noli]
      ## IF CALCUL ='RELATIF' :
                           ♦ ACCE_X =        ac_x                [fonction]
                           ♦ ACCE_Y =        ac_y                [fonction]
                          ♦  ACCE_Z =        ac_z                [fonction]
                          )
          ◊ IMPRESSION = _F  (
                             ◊   TRI =     /   'AMOR_SPEC'       [DEFAUT]
                                           /   'DIRECTION'
                             ◊   FORMAT =  /   'TABLEAU'         [DEFAUT]
                                           /   'XMGRACE'
   ## logical unit number defining the file (fort.N) where result are
written:
                             ◊   UNITE =   /   29                [DEFAUT]
                                           /   u                 [I]
             ## IF FORMAT ='XMGRACE' :
                             ◊   PILOTE =  /   'POSTSCRIPT'
                                           /   'EPS'
                                           /   'MIF'
                                           /   'SVG'
                                           /   'PNM'
                                            /  'PNG'
                                            /  'JPEG'
                                           /   'PDF'
                                           /   'INTERACTIF'
                              ◊ TOUT =     /   'NON'             [DEFAUT]
                                           /   'OUI'
                                 )
```

```
## IF NOM_CHAM ='DEPL' :
    ◊  LIST_INST    =   linst                        [listr8]
     ♦ RESU = _F (
                 ♦ /   ‘ RESU_GENE ’ = tg            [ tran_gene ]
                   /   ‘RESULTAT’  = resu            [dyna_trans,
                                                      evol_noli]
   ## IF CALCUL ='ABSOLU' :
                 ♦ DEPL_X =        de_x              [fonction]
                 ♦ DEPL_Y =        de_y              [fonction]
                 ♦ DEPL_Z =        de_z              [fonction]
                )
 );
```

3 Operands

The commonly used post-treatment of a seismic transient dynamics analysis consists in:

- Floor spectra obtained from absolute accelerations, for each direction X, Y, Z, calculated at a specified node of the finite element mesh (for instance in the case of a “stick-model”);
- Envelop of floor spectra calculated at some nodes on the same floor, for each direction X, Y, Z and H (max of X and Y), (for instance in the case of a 3D building model);
- Structural relative displacements envelops with respect to the ground motion.

Two *Code_Aster* operators can produce the `resultat` concept corresponding to the structural transient dynamics calculation:

- `DYNA_TRAN_MODAL` [U4.53.21] producing a `resu_gene` concept, including relative field values (accelerations and displacements). In that case, we need to add ground accelerations to get absolute accelerations suitable to compute the floor spectra.
- `LIRE_ MISS3D` [U7.02.31] producing a `dyna_trans` concept, including absolute field values (accelerations and displacements). In that case, accelerations are directly suitable to compute the floor spectra; conversely, deduction of ground displacements is compulsory (read with `LIRE_FONCTION` [U4.32.02] from a specified file logic unit) to get the relative displacements.

A quick sketch of the algorithm is presented in the following:

```
Loop #1 on the floors
    Loop #2 on the floor nodes
        Loop #3 on the 3 directions (X,Y,Z)
            Loop #4 on the results
            Function recovering: nodal relative accelerations… ( RECU_FONCTION )
                In case of accelerations:
                    Combination or not with ground acceleration function ( CALC_FONCTION COMB )
                    Response spectra calculation, with specified frequencies and damping values
                        ( CALC_FONCTION SPEC_OSCI )
                In case of displacements (absolute displacements):
                    Ground displacements are deducted to get relative displacements
                        ( CALC_FONCTION COMB )
            End of loop #4
                In case of accelerations:
                    result mean value calculation for given node and direction ( CALC_FONCTION
COMB )
                In case of displacements:
                    function maxima recovering
         End of loop #3
        Acceleration spectra printing at each node for each direction X, Y, Z ( IMPR_FONCTION )
    End of loop #2
        Acceleration spectra envelops on a specified floor or maximal displacements
            ( CALC_FONCTION ENVELOPPE )
        Acceleration spectra envelops printing at each floor for each direction X, Y, Z, H
                ( IMPR_FONCTION )
End of loop #1
```

3.1 Keyword MAILLAGE

This required keyword is used to specify the mesh defined by LIRE_MAILLAGE [U4.21.01].

3.2 Keyword PLANCHER

This required factor keyword is used to specify the floor names, where spectra will be computed. These names will be used to select or filter the parameter values in the table_sdaster data structure produced by this macro-command (in order to print or test numerical values…) .

3.2.1 Operand NOM

This required operand specifies the name of the considered floor.

3.2.2 Operand NOEUD / GROUP_NO

This operand specifies the individual nodes or node groups of the finite element mesh of the considered floor name where the spectra will be computed.

3.3 Keyword CALCUL

This required keyword is used to specify the nature of the transient dynamics calculation used for the post-treatment: in the absolute frame ('ABSOLU') or in the relative frame ('RELATIF').

3.4 Keyword NOM_CHAM

This required keyword is used to specify the nature of the field value used for the post-treatment: accelerations ('ACCE') or displacements ('DEPL').

3.5 Case NOM_CHAM ='ACCE'

In that case, we have to give the following data requested for the acceleration response spectra calculation: in the 3 directions X, Y, Z (vertical) and H (horizontal maximal value from the X and the Y values) .

3.5.1 Operand AMOR_SPEC

This required operand specifies the reduced damping coefficient values used by the response spectra calculation, see also CALC_FONCTION [U4.32.04], keyword SPEC_OSCI .

3.5.2 Operand LIST_INST

This facultative operand specifies the list, produced by DEFI_LIST_REEL [U4.34.01], defining all the time steps for which the transient dynamics result is requested for the response spectra calculation .

3.5.3 Operand FREQ / LIST_FREQ

/ ◊ FREQ = l_fr
l_fr = f_1,..., f_i. Frequency list.
/ ◊ LIST_FREQ = lfreq
Frequency list defined before by a listr8 concept.

This facultative operand specifies the frequency values list used by the response spectra calculation, see also CALC_FONCTION [U4.32.04], keyword SPEC_OSCI .

3.5.4 Operand NORME

♦ NORME = r

The response spectrum will be normalised at the r value (pseudo-acceleration value). Calculations are in many cases done with International Units and input acceleration time histories are often given into m/s². Response spectra are usually given in g = 9.81m/s².
So this compulsory operand NORME can be used for a unit conversion between calculated accelerations and response spectra, see also CALC_FONCTION [U4.32.04], keyword SPEC_OSCI .

3.5.5 Keyword RESU

This compulsory factor keyword is used to specify the names of the resultat concepts where nodal accelerations are taken. The operand can be:

♦ / ' RESU_GENE ' = tg [tran_gene]

if the post-treatment is made on a DYNA_TRAN_MODAL [U4.53.21] produced concept (transient dynamics on reduced modal basis). However, this result has to be computed in the relative frame.

or:

/ 'RESULTAT' = resu [dyna_trans, evol_noli]

if the post-treatment is made on a transient dynamics result (coming for instance from LIRE_ MISS3D [U7.02.31], produced concept: dyna_trans).

3.5.6 Case CALCUL ='RELATIF'

♦ ACCE_X = / ac_x [fonction]
♦ ACCE_Y = / ac_y [fonction]
♦ ACCE_Z = / ac_z [fonction]

In that case, we have to introduce the ground acceleration functions, defined on the same time list, in each spatial direction, to be combined with the relative accelerations to compute the absolute accelerations.

3.6 Keyword IMPRESSION

This facultative keyword is used to specify the nature of the curves printing (spectra, envelops).

3.6.1 Operand TRI

This facultative keyword is used to specify the nature of the curves sorting: by damping coefficient value ('AMOR_SPEC') or by spatial direction ('DIRECTION').

3.6.2 Operand FORMAT

This facultative keyword is used to specify the nature of the curves format: by printed table ('TABLEAU') or readable by the Xmgrace software ('XMGRACE'). The abscissa scale (frequency) is logarithmic.

3.6.3 Operand UNITE

◊ UNITE = u

Logical number of unit of the file where result are written (integer to be chosen between 10 and 90). Unit 29 by default. This unit number has to be the same as specified in the astk interface.

3.6.4 Case FORMAT ='XMGRACE'

In that case we can choose, using the PILOTE keyword, the file format driver for Xmgrace plotting software among the following list, see IMPR_FONCTION [U4.33.01]:

'POSTSCRIPT' : postscript type file,
'EPS' : encapsulated postscript type file,
'MIF' : specific type file,
'SVG' : specific type file,
'PNM' : image specific type file,
'PNG' : image specific type file,
'JPEG' : image compressed type file,
'PDF' : portable document format type file,
'INTERACTIF' : the curves are directly drawn on the screen by Xmgrace, if open, without file.

3.6.5 Operand TOUT

This facultative operand TOUT='OUI' can be used to print all the spectra calculated at the end of loop #4 : mean values at each node for all directions and all damping values.

3.7 Case NOM_CHAM ='DEPL'

In that case, we have to give the following data requested for the displacement envelope calculation: in the 3 directions X, Y, Z (vertical) and H (horizontal maximal value from the X and the Y values) .

3.7.1 Operand LIST_INST

This operand specifies the list, produced by DEFI_LIST_REEL [U4.34.01], defining all the time steps for which the transient dynamics result is requested for the displacement envelope calculation .

3.7.2 Keyword RESU

This required factor keyword is used to specify the names of the resultat concepts where nodal accelerations are taken. The operand can be:

```
♦ /   ‘ RESU_GENE ’ = tg                [ tran_gene ]
```

if the post-treatment is made on a DYNA_TRAN_MODAL [U4.53.21] produced concept (transient dynamics on reduced modal basis). However, this result has to be computed in the relative frame.

or:

```
  /   ‘RESULTAT’  = resu            [dyna_trans,
                                      evol_noli]
```

if the post-treatment is made on a transient dynamics result (coming for instance from LIRE_MISS3D [U7.02.31], produced concept: dyna_trans).

3.7.3 Case CALCUL ='ABSOLU'

```
♦ DEPL_X =      / de_x            [fonction]
♦ DEPL_Y =      / de_y            [fonction]
♦ DEPL_Z =      / de_z            [fonction]
```

In that case, we have to introduce the ground displacement functions, defined on the same time list, in each spatial direction, to be deducted from the absolute displacements to compute the relative displacements.

4 Example

The following examples come from sdll138a test case, first to illustrate the acceleration spectra computation, then to illustrate the displacement envelopes computation:

```
TAB=MACR_SPECTRE(
        MAILLAGE  = MA1,
        PLANCHER  = (_F(NOM = 'NIV1',
                        GROUP_NO  = ('N4_NIV1',), ),
                     _F(NOM  = 'NIV8',
                        GROUP_NO  = ('N4_NIV8','N5_NIV8',), ),
                      ),
        NOM_CHAM = 'ACCE',
        CALCUL = 'RELATIF',
        AMOR_SPEC = L_AMOR_S,
          LIST_FREQ = L_FREQ,
          LIST_INST = LISTE,
          RESU=(_F(RESU_GENE = TRAN_GE1,
                   ACCE_X    = acceH1,
                   ACCE_Y    = acceH2,
                   ACCE_Z    = acceV3,),
                _F(RESU_GENE = TRAN_GE2,
                   ACCE_X    = acceH2,
                   ACCE_Y    = acceH3,
                   ACCE_Z    = acceV1,),
                _F(RESU_GENE = TRAN_GE3,
                   ACCE_X    = acceH3,
                   ACCE_Y    = acceH1,
                   ACCE_Z    = acceV2,), ),
         IMPRESSION =_F(
                  FORMAT ='TABLEAU',
                  UNITE = 16,)
                ),

TAB2=MACR_SPECTRE(
        MAILLAGE  = MA1,
        PLANCHER  = (_F(NOM = 'NIV1',
                      GROUP_NO  = ('N4_NIV1','N5_NIV1','N6_NIV1','N7_NIV1',),
                          ),
                     _F(NOM  = 'NIV8',
                      GROUP_NO  = ('N4_NIV8','N5_NIV8','N6_NIV8','N7_NIV8',),
                         ),
                      ),
        NOM_CHAM = 'DEPL',
        CALCUL = 'RELATIF',
        LIST_INST = LISTE,
        RESU=(_F(RESU_GENE = TRAN_GE1,),
              _F(RESU_GENE = TRAN_GE2,),
              _F(RESU_GENE = TRAN_GE3,),
             ),
                ),

TAB3=MACR_SPECTRE(
        MAILLAGE  = MA1,
        PLANCHER  = (_F(NOM = 'NIV1',GROUP_NO  = ('N4_NIV1',), ),
                      _F(NOM = 'NIV8',GROUP_NO  = ('N4_NIV8','N5_NIV8',), ), ),
        NOM_CHAM = 'ACCE',
        CALCUL = 'ABSOLU',
        AMOR_SPEC = L_AMOR_S,
          LIST_FREQ = L_FREQ,
          RESU=(_F(RESULTAT = DYNAT_K1,),
                _F(RESULTAT = DYNAT_K2,),
                _F(RESULTAT = DYNAT_K3,),),
         IMPRESSION =_F(
                  FORMAT ='TABLEAU',
                  UNITE = 16,), )
```