Operator PRE_SEISME_NONL

1 Goal

The purpose of this operator is to simplify the implementation of the method Laplace-Time by allowing the realization of non-linear calculations with impedances produced by MISS3D.

The use of the method Laplace - Time (*cf*. [U2.06.05]) requires several operations:

1) the calculation of a base of modes of interface necessary for the projection of the impedances obtained with the operator CALC_MISS (*cf*. [U7.03.12]);
2) the creation of a new model which assembles these matrices of impedance, that it is within the framework of a resolution in physical base or with techniques of dynamic reduction (*cf*. [U2.07.04]);
3) possibly the transition statics-dynamics when a state of prestressed must be taken into account at the beginning of transitory calculation.

According to the arguments as starter, this order allows the realization of the operations mentioned previously, i.e., it makes it possible to produce the concepts necessary to call on the macro-order CALC_MISS [U7.03.12] or to exploit of them the results of type calculation of impedance.

Contents

2 Syntax

```
PRE_SEISME_NONL  (
      ♦  RESULT =  _F  (
         ◊  /  RESULT    =  CO ('resu'),                  [evol_noli]
         ◊  /  MODEL     =  CO ('Mo'),                      [model]
         ◊  /  GRID      =  CO ('e-mail'),                [grid]
          ◊ /  CHAM_MATER    = CO ('chmat'),                  [cham_mater]
         ◊  /  CARA_ELEM    =  CO ('carac'),               [cara_elem]
               # PRE_CALC_MISS is obligatory for creation
               # of the concepts of the type bases modal or macronutrient
         ◊  /  BASE_MODALE     =  CO ('bamo'),             [mode_meca]
         ◊  /  MACR_ELEM_DYNA  =  CO ('mael'),             [macr_elem_dyna]
               # POST_CALC_MISS is obligatory for creation
               # of a load with option 'LAPL_TEMPS'
         ◊  /  LOAD         = _F  (
                  ♦ NAME      =  CO ('l_char'),
               [l_char_meca]
                  ♦ OPTION =  / 'COND_LIM',              [DEFECT]
                              / 'LAPL_TEMPS',            [TXM]
                                 ),
         ),
      ◊  /  PRE_CALC_MISS = _F (
               ♦  CALC_MISS_OPTION =  /  'ISS',          [TXM]
                                      /  'ISFS',         [TXM]
               ♦  NMAX_MODE_ISS    =  nbiss,             [I]
               # If CALC_MISS_OPTION == 'ISFS'
                  ♦  NMAX_MODE_IFS    =  nbifs,          [I]
               ♦  GROUP_MA_INTERF  =  grma_int,          [grma]
               ◊  GROUP_NO_CENT    =  grno_cent,         [grno]
               ◊  REDUC_DYNA_ISS   =  /  'NOT',          [DEFECT]
                                      /  'YES',          [TXM]
               ◊  REDUC_DYNA_IFS   =  /  'NOT',          [DEFECT]
                                      /  'YES',          [TXM]
         ),
      ◊  /  POST_CALC_MISS = _F (
               ♦  MACR_ELEM_DYNA   =  mael,              [macr_elem_dyna]
               ♦  GROUP_MA_INTERF  =  grma_int,          [grma]
               ♦  GROUP_NO_CENT    =  grno_cent,         [grno]
               ◊  UNITE_RESU_RIGI  =  uresrig,           [I]
               ◊  UNITE_RESU_AMOR  =  uresamo,           [I]
               ◊  UNITE_RESU_MAS   =  uresmas,           [I]
         ),
      ◊  /  STAT_DYNA = _F (
               ♦  RESULTAT  = resu,                      [evol_noli]
               ♦  EXCIT     = _F (
                  ♦  LOAD      = tank,                /  [char_meca]
                                                      /  [char_cine_meca]
                  ◊ FONC_MULT = fonct,                /[fonction_sdaster]
                                                      /  [nappe_sdaster]
                                                      /  [formula]
                  ◊ TYPE_CHARGE =/'FIXE_CSTE',        [DEFECT]
                                / 'FIXE_PILO',           [TXM]
                                / 'SUIV',                [TXM]
                                / 'DIDI',                [TXM]
```

```
                        ),
                ♦  BEHAVIOR  = _F (see the document [U4.51.11]),
                ♦  CONVERGENCE  = _F (see the document [U4.51.03]),
                   ♦  BASE_MODALE = bamo,                          [mode_meca]
                   ♦  UNITE_IMPE_TEMPS = _F (
                         ◊ UNITE_RESU_RIGI = uresrig,              [I]
                         ◊ UNITE_RESU_AMOR = uresamo,              [I]
                         ◊ UNITE_RESU_MASS = uresmas,              [I]
                                          ),
                   ♦  UNITE_IMPE_FREQ = uimpfr,                    [I]
                   ♦  FORCE_SOL = fsol,                            [char_meca]
                   ◊  COEF_AMOR =   /  0.0,                        [DEFECT]
                                    /  coefamo,                    [R]
                   ◊  NB_INST   =   /  100.0                       [DEFECT]
                                    /  nbinst,                     [R]
                ),

       # the declaration of the keyword following must be coherent
       # with the keyword informed in RESULT
       ♦  AFFE_MODELE      =  _F ( to see the document [U4.41.01] ),

       ♦  AFFE_MATERIAL    =  _F (see the document [U4.43.03]  ),

       ♦  AFFE_CARA_ELEM   =  _F (see the document [U4.42.01]  ),

       ♦  AFFE_CHAR_MECA   =  _F (see the document [U4.44.01]  ),

       ◊  INFORMATION   =  /  1,
[DEFECT]
                    /  2,                                          [I]
   )
```

3 Operands

3.1 Operand RESULT

```
♦ RESULT = _F (
    ◊ / RESULT = CO ('resu'),                    [evol_noli]
```

Allows PRE_SEISME_NONL to produce the concepts at exit of the macro-order.

The keyword RESULT is to be used only with STAT_DYNA.

3.1.1 Operand MODEL/GRID/CHAM_MATER/CARA_ELEM

```
    ◊ / MODEL = CO ('Mo'),                          [model]
    ◊ / GRID = CO ('e-mail'),                     [grid]
     ◊ / CHAM_MATER = CO ('chmat'),                  [cham_mater]
     ◊ / CARA_ELEM = CO ('carac'),                 [cara_elem]
```

Allows PRE_SEISME_NONL to produce the concepts at exit of the macro-order.
The declaration of these concepts must be coherent with the data sunken in the operands AFFE_MODELE (*cf* . § 3.5) , AFFE_MATERIAU (*cf* . § 3.6), AFFE_CARA_ELEM (*cf.* § 3.7) and AFFE_CHAR_MECA (*cf* . § 3.8) defined inside this macro-order.

These keywords make it possible to inform the name of the concepts which the user wishes to create:

- the name of the model (Mo) whose elements are the object of mechanical calculation,
- the name of the grid (e-mail) possibly containing new nodes and meshs,
- the name of the material field (chmat) affected on the grid e-mail,
- the name of the characteristics (carac) elements of hull, beam, pipe, bars, cable, and discrete elements affected on the model Mo.

In case of a dynamic reduction (*cf.* [U2.07.04] and [U4.44.01]), these concEpts created will contain the additional elements necessary to later calculations.

3.1.2 Operand BASE_MODALE/MACR_ELEM_DYNA

```
    ◊ / BASE_MODALE = CO ('bamo'),                [mode_meca]
     ◊ / MACR_ELEM_DYNA = CO ('mael'),             [ macr_elem_dyna
]
```

These keywords cannot be used without informing the keyword factor PRE_CALC_MISS (*cf*. §3.2). Indeed, they make it possible to inform the following concepts at exit:

- the name of the modal base (bamo) containing the modes of interface used as projection of the impedances of ground bases,
- the name of the macronutrient (mael) who points on bamo and which can be directly used in CALC_MISS (*cf* . [U7.03.12]).

3.1.3 Operand LOAD

```
LOAD = _F (
    ♦ NAME = CO ('l_char'),
   [l_char_meca]
    ♦ OPTION = / 'COND_LIM',                      [DEFECT]
               / 'LAPL_TEMPS',                    [TXM]
                 ),
```

This keyword makes it possible to create a load related to the boundary conditions (OPTION = 'COND_LIM'), obligatory when one seeks to make dynamic reduction, or with the method Laplace-

Time (OPTION = 'LAPL_TEMPS'), which obliges the presence of the operands UNITE_RESU_RIGI, UNITE_RESU_AMOR or UNITE_RESU_MASS in the keyword factor POST_CALC_MISS (*cf.* §3.3).

3.2 Operand PRE_CALC_MISS

```
◊  /  PRE_CALC_MISS = _F (
         ♦  CALC_MISS_OPTION =  /  'ISS',            [TXM]
                                /  'ISFS',           [TXM]
         ♦  NMAX_MODE_ISS    =  nbiss,               [I]
         # If CALC_MISS_OPTION == 'ISFS'
            ♦  NMAX_MODE_IFS    =  nbifs,            [I]
         ♦  GROUP_MA_INTERF  =  grma_int,          [grma]
         ◊  GROUP_NO_CENT    =  grno_cent,         [grno]
         ◊  REDUC_DYNA_ISS   =  /  'NOT',          [DEFECT]
                                /  'YES',          [TXM]
         ◊  REDUC_DYNA_IFS   =  /  'NOT',          [DEFECT]
                                /  'YES',          [TXM]
      ),
```

3.2.1 Operand REDUC_DYNA_ISS/REDUC_DYNA_IFS

These operands indicate the manner of representing the kinematics of the interfaces of ISS or YEWS by means of a base of static modes (REDUC_DYNA_*= 'NOT') or of the dynamic modes (REDUC_DYNA_* = 'YES').

3.2.2 Operand NMAX_MODE_ISS/NMAX_MODE_IFS

These operands make it possible to inform the number of modes to in the case of calculate a calculation of ISS or ISFS. If REDUC_DYNA_* = 'YES', the value of the keyword NMAX_MODE_* corresponding must be a multiple of 6.

3.2.3 Operand GROUP_MA_INTERF

This keyword, which is obligatory, makes it possible to inform the group of meshs corresponding to the dynamic interface to consider (together interface of ISFS or that of ISS or that of YEWS) for the calculation of the modes of interface.

3.2.4 Operand GROUP_NO_CENT

This keyword indicates the group of nodes defined in the center of an interface making it possible to consider a rigid behavior of this interface. If this keyword is indicated a condition of the type LIAISON_SOLIDE (*cf.* [U4.42.01]) must, *a priori*, being seized by means of the operand AFFE_CHAR_MECA (*cf.* §3.8).

3.2.5 Operand CALC_MISS_OPTION

This keyword indicates if calculation to be realized is of type 'ISS' or of type 'ISFS'. The well informed value must thus be coherent with the other operands of PRE_CALC_MISS but also of the macro-order CALC_MISS (*cf* . [U7.03.12]).

3.3 Operand POST_CALC_MISS

```
◊  /  POST_CALC_MISS = _F (
         ♦  MACR_ELEM_DYNA   =  mael,              [macr_elem_dyna]
         ♦  GROUP_MA_INTERF  =  grma_int,          [grma]
         ♦  GROUP_NO_CENT    =  grno_cent,         [grno]
         ◊  UNITE_RESU_RIGI  =  uresrig,           [I]
         ◊  UNITE_RESU_AMOR  =  uresamo,           [I]
         ◊  UNITE_RESU_MAS   =  uresmas,           [I]
      ),
```

3.3.1 Operand MACR_ELEM_DYNA

This keyword expects as starter a concept of the type macr_elem_dyna who primarily corresponds to the impedance (frequential or temporal) to assemble with the digital model and who is obtained before with, for example, the sequence of the order LIRE_IMPE_MISS (*cf.* [U7.02.32]) and MACR_ELEM_DYNA (*cf.* [U4.65.01]).

3.3.2 Operand GROUP_MA_INTERF

This keyword same as that is used in § 3.2.3 .

3.3.3 Operand GROUP_NO_CENT

This operand same as that is used in §3.2.4. It is particularly important when a calculation with STAT_DYNA (*cf.* §3.4) must be carried out thereafter.

3.3.4 Operand UNITE_RESU_RIGI/UNITE_RESU_AMOR/UNITE_RESU_MASS

Numbers of logical unit of the files containing of the temporal impedances of ground (or its decomposition in terms of inertia, damping and rigidity) for the method Laplace-Time.

Operands UNITE_RESU_RIGI, UNITE_RESU_AMOR, UNITE_RESU_MASS, being of a use specific to the method Laplace-Time, are obligatory at the time a concept charges with type LAPL_TEMPS (*cf.* §3.1.3) is asked at exit of this macro-order.

3.4 Operand STAT_DYNA

```
◊  /  STAT_DYNA = _F (
          ♦  RESULTAT  = resu,                          [evol_noli]
          ♦  EXCIT     = _F (
             ♦  LOAD      = tank,                  /   [char_meca]
                                                   /   [char_cine_meca]
             ◊ FONC_MULT = fonct,                  /[fonction_sdaster]
                                                   /   [nappe_sdaster]
                                                   /   [formula]
             ◊ TYPE_CHARGE =/'FIXE_CSTE',          [DEFECT]
                           / 'FIXE_PILO',             [TXM]
                           / 'SUIV',                  [TXM]
                           / 'DIDI',                  [TXM]
                       ),
          ♦  BEHAVIOR  = _F (see the document [U4.51.11]),
          ♦  CONVERGENCE  = _F (see the document [U4.51.03]),
             ♦  BASE_MODALE = bamo,                     [mode_meca]
             ♦  UNITE_IMPE_TEMPS = _F (
                   ◊ UNITE_RESU_RIGI = uresrig,         [I]
                   ◊ UNITE_RESU_AMOR = uresamo,         [I]
                   ◊ UNITE_RESU_MASS = uresmas,         [I]
                                     ),
             ♦  UNITE_IMPE_FREQ = uimpfr,               [I]
             ♦  FORCE_SOL = fsol,                       [char_meca]
             ◊  COEF_AMOR =  /  0.0,                    [DEFECT]
                             /  coefamo,                [R]
             ◊  NB_INST   =  /  100.0                   [DEFECT]
                             /  nbinst,                 [R]
       ),
```

3.4.1 Operand RESULT

This operand expects as starter a concept of the type evol_noli corresponding to the result of a static calculation.

In particular, within the framework of the method Laplace-Time when one seeks to carry out a non-linear seismic analysis with taking into account of the interaction ground-structure, the expected concept corresponds to that produced by the operator STAT_NON_LINE (*cf.* [U4.51.03]).

3.4.2 Operand EXCIT

Cf. [U4.51.03].

In practice and in the framework of the method Laplace-Time, the keyword EXCIT the whole waits of loads used for the non-linear calculation whose result is seized in RESULT (*cf.* §3.4.1).

3.4.3 Operand BEHAVIOR/CONVERGENCE

Cf. [U4.51.03].
Syntax is the same one as that of the keywords factor defined in the operator STAT_NON_LINE and it must be coherent with the result seized in RESULT (*cf.* §3.4.1).

3.4.4 Operand BASE_MODALE

This keyword expects as starter the modal base being used to project the impedances of ground. This base must be the same one as that used to project the impedances of ground indicated in UNITE_IMPE_TEMPS (*cf.* §3.4.5) and UNITE_IMPE_FREQ (*cf.* §3.4.6) in the form of logical units.

3.4.5 Operand UNITE_IMPE_TEMPS

The three keywords of this operand have the same meaning as in § 3.3.4 .

3.4.6 Operand UNITE_IMPE_FREQ

This keyword expects as starter the logical unit of the file containing the impedance of ground calculated in the frequential field.

3.4.7 Operand FORCE_SOL

This keyword expects as starter a concept of load of the type 'LAPL_TEMPS' (*cf.* §3.1.3) such as created as a preliminary by the use of the keyword POST_CALC_MISS (*cf.* §3.3).

3.4.8 Operand COEF_AMOR

This keyword is a coefficient which makes it possible to control the value of viscous damping to add on the level of the interface ground-structure in the shape of discrete elements of type DIS_TR .
When COEF_AMOR is worth 1.0, the value of damping which is assigned to the whole of the degrees of freedom of each discrete element is of 1.E16 $N.s.m^{-1}$.

3.4.9 Operand NB_INST

This keyword expects as starter the number of steps of time necessaries to stabilize the transition statics-dynamics. The value by default is often sufficient.

3.5 Operand AFFE_MODELE

Allows to define the modelled physical phenomenon (mechanical, thermal, acoustic) and the type of finite elements used. Syntax related to the keyword AFFE_MODELE is the same one as that of the operator bearing the same name [U4.41.01].

Only one occurrence of this keyword is authorized.

3.6 Operand AFFE_MATERIAU

Allows to assign materials to geometrical zones of a grid or a model. Syntax related to the keyword AFFE_MATERIAU is the same one as that of the operator bearing the same name [U4.43.03]. The only difference relates to the operand MODEL who is not obligatory when PRE_CALC_MISS (*cf.* §3.2) is present.

Only one occurrence of this keyword is authorized.

3.7 Operand AFFE_CARA_ELEM

Allows to assign to elements of structure of the geometrical and material characteristics. Syntax related to the keyword AFFE_CARA_ELEM is the same one as that of the operator bearing the same name [U4.42.01]. The only difference relates to the operand MODEL who is not obligatory when PRE_CALC_MISS (*cf.* §3.2) is present.

Only one occurrence of this keyword is authorized.

3.8 Operand AFFE_CHAR_MECA

Allows to affect loadings and boundary conditions on a mechanical model. Syntax related to the keyword AFFE_CHAR_MECA is the same one as that of the operator bearing the same name [U4.44.01]. The only difference relates to the operand MODEL who is not obligatory at the time only PRE_CALC_MISS (*cf.* §3.2) is present.

Only one occurrence of this keyword is authorized.

3.9 Operand INFORMATION

Allows to control the level of message of the macro-order.