Code_Aster

Version default

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Date : 04/11/2019 Page : 1/6
Clé : U4.84.32 Révision :
4261bd411366

Order `DEFI_CHAR_GROUND`

1   Goal

Produire an usable load in the operators of dynamic resolution `DYNA_VIBRA` and `DYNA_NON_LINE` for pto ouvoir to take into account seismic loads at the borders of fields of ground 2D or 3D. The loads created concern two sources potentially: either of the loads of type "plane wave" deduced from movements imposed on the side borders in the direction of propagation, or of the complements of second member of the type "forces nodal" imposed on the side borders in the transverse directions with the propagation.

The data of imposed movement or fields of force can also come from two sources: that is to say a file result of transitory evolutions resulting from an auxiliary calculation of column of ground 2D by `DEFI_SOL_EQUI`, that is to say an evolution of type result calculated on a model auxiliary and projected on the borders of the field.

She takes into account the concomitance of these alternative choices:

- source of the transitory data of movements or forces at the borders: that is to say a unit of transitory results got by `DEFI_SOL_EQUI` for the auxiliary calculation of column of ground 2D behind the keyword `UNITE_TRAN_UNIT` and in this case one informs coherent data with `DEFI_SOL_EQUI` ; that is to say a result of transitory evolutions at the borders of the basin calculated on an auxiliary model 2D or 3D entered behind the keyword `RESU_INIT` and coherent with the new syntax of `CREA_RESU`. In both cases, it is necessary to specify the type of load plane wave or force imposed by the keyword `LOADING`.

Code_Aster

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Version
default

Date : 04/11/2019 Page : 2/6
Clé : U4.84.32 Révision :
4261bd411366

2   Syntax

```
tank ground [ char_meca ] = DEFI_ CHAR_ GROUND (

        ♦       MODELE       = mO,                           [modelE]
        ♦       /♦    UNITE_TRAN_INIT  = utrin,        [ Links (In) ]
                /♦    RESU_INIT  = resui ,               / [dyna_trans]
                                                          / [ evol_char ]
                                                          / [ evol_noli ]

        ◊       LOADING              =/    'FORCE',       [DEFECT]
                                       /   'ONDE_PLANE',

    # If UNITE_TRAN_INIT
    {
        ◊       NOM_CMP     =/'DX',                        [DEFECT]
                            /'DY',
         ◊     LENGTH _ CARA    = lcar has,                      [R]
          ♦    | ◊  GROUP_MA_DROITE   = gmdr,      [grma]
               | ◊  GROUP_MA_GAUCHE   = gmga,      [grma]
          ♦    TABLE_MATER_ELAS  = tmela,           [table]
          ◊    Z0           = / z0 ,               [R]
                             = / 0. ,                    [DEFECT]
          ◊    | ♦   AXIS         =/'X',
                                  /'Y',
                                  /' Z ',
               | ◊   NOM_PARA    =/'Y',
                                  /' Z ',             [DEFECT]
    } # End If UNITE_TRAN_INIT

    # If RESU_INIT '
    {
        ◊       / ♦    MATR_AMOR = matr_has, [matr_asse_depl_r]
                / ◊    MATR_RIGI = matr_k,   [matr_asse_depl_r]
        ◊       / NUME_DDL = digital,         [nume_ddl]
                / MATR_RIGI = matr_k,        [matr_asse_depl_r]
        ♦       /   INST    =     inst,       [R]
                /   LIST_INST   =    linst,    [l_R]
        ♦       NOM_CHAM_INIT =  / 'FORC_NODA',     [DEFECT]
                                 / 'PULLEYC_NODA',
                                 / 'ACCE',
        ◊       DDL_EXCLUS    =  / 'DX',
                                 / 'DY',
                                 / 'DZ',
                                 / 'DRX',
                                 / 'DRY MARTINI',
                                 / 'DRZ',
        ◊       CRITERION    =   / 'ABSOLUTE',
                                 / 'RELATIVE',    [DEFECT]
        ◊       PRECISION    =   / prec,
                                 / 1.0E-6,         [DEFECT]
        ◊        COEFF     =     / coeff ,
                                 / 1.0,            [DEFECT]
    } # End If RESU_INIT
        ◊       INFORMATION =           /     1,                [DEFECT]
                                        /     2,
        ◊       TITLE =            title,                  [l_Kn]
    )
```

Code_Aster

Version
default

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Date : 04/11/2019 Page : 3/6
Clé : U4.84.32 Révision :
4261bd411366

3   Operands

3.1   Operand MODELE

♦   MODELE = mO

Permet to define the name of model to which will be affectée the load. According to the dimension of the problem, they must be in 2D a model plane deformations ('D_PLAN'), or in 3D of a voluminal model with a modeling'3D'.

3.2   Operand LOADING

◊   LOADING =   /   'ONDE_PLANE',
                          /   'FORCE',

Indicate the nature of the imposed loading :
- LOADING = 'ONDE_PLANE' : loading in the form of a wave planes vertical propagation imposed on the model and deduced from the movement in plane wave imposed on a column of ground 2D or another auxiliary model. The wave is of shearing if there is a horizontal movement of request, it is of pressure if there is a vertical movement of request.
- LOADING = 'FORCE' : loading in the form of opposite forces imposed on the side edges of the model.

3.3   Operand UNITE_TRAN_INIT

Fournit the unit of writing to the format 'TABLE' transitory evolutions in each level of a column 2D of laminated ground obtained at the time of a former call with DEFI_SOL_EQUI. At the time of this former call, Lbe evolutions are created using the keyword UNITE_RESU_TRAN. The loads imposed on the model are extracted in the forms from tablecloths from forces depending on time and the dimension on the stratification starting from the file corresponding to this unit.

3.3.1   Operand NOM_CMP

◊   NOM_CMP =   /   'DX',   [DEFECT]
                                   /   'DY',

Indicate the direction of the signal imposed on the column of ground 2D which was used to provide the results of the transitory evolutions to impose like boundary conditions :
- NOM_CMP = 'DX' : imposed signal is horizontal (corresponds to a wave of shearing of vertical propagation);
- NOM_CMP = 'DY' : imposed signal is vertical (corresponds to a wave of vertical pressure).

3.3.2   Operand AXIS

◊   AXIS =   /   'X',
                                          /   'Y',
                                          /   'Z',

Indicate the direction DE the load imposedE with the model. This parameter is necessary in 3D and makes it possible by its presence to activate this case of loading.

Note: this keyword is not necessary in 2D because the direction of the load depends on the type of loading and the direction of the signal: for NOM_CMP = 'DX', one has 'X' for LOADING = 'ONDE_PLANE' and 'Y' for LOADING = 'FORCE'. Conversely for NOM_CMP = 'DY'.

3.3.3   Operand NOM_PARA

Code_Aster

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Version default

Date : 04/11/2019 Page : 4/6
Clé : U4.84.32   Révision   :
4261bd411366

◊   NOM_PARA =   /   'Y',
                                             /   'Z',   [DEFECT]

Allows to only specify in 3D, the direction DE propagation of the signal of plane wave generating the force to be applied and corresponding to the vertical direction of the model. This operand is not necessary in most case because he generally results from the direction of the signal imposed on the column of ground by NOM_CMP and of the direction of the load imposed on the model by AXIS. But for NOM_CMP = 'DY', and AXIS = 'X', for example, one can have a parameter of propagation for a loading of the type 'FORCE'as well in the direction'Y' that in the direction 'Z'. And for NOM_CMP = 'DX', and AXIS = 'X', for a loading of the type 'ONDE_PLANE', one can have a wave as well HS in the direction 'Y' qu'oneE wave SV in the direction 'Z'.
One considers however a value by default ' Z' as a vertical direction for a 3D problem.

3.3.4   Operands GROUP_MA_DROITE/GROUP_MA_GAUCHE

♦ | ◊ GROUP_MA_DROITE = gmda
         | ◊ GROUP_MA_GAUCHE = gmga

Permettent respectively to define the names of the groups of meshs représentatifs flat rim and left edge model to which serhas affected the load. One must inform at least one of these operands. These groups of meshs are linear in 2D and surface in 3D, except in the case loading of the type 'FORCE' in 3D where one must define linear groups of meshs encircling the edges of the model.

3.3.5   Operand TABLE_MATER_ELAS

Indicate the table which contienT information on the geometrical description of the horizons soil laminated and that of the initial characteristics of materials constitutive of these layers, sleeps by layer. It corresponds to the table produced by the call to DEFI_SOL_EQUI who also produced the file entered by UNITE_TRAN_INIT.

3.3.6   Operand Z0

Indicate, the dimension of free surface, worthless by default, of the model to which to make correspond geometrically the worthless dimension of the model of column of ground 2D.

3.3.7   Operand LONG_CARA

Indicate, in the case of a loading plane wave, the value characteristic length defined in DEFI_MATERIAU keyword ELAS allowing to calculate of rigidity added on the absorbing border constituted by edge S right and left of the column of ground .

3.4   Operand RESU_INIT

Provides the name of the result of the transitory evolution calculated on a model to auxiliairE from which one must extract the fields from speed, displacement or nodal force necessary to the generation of the evolutions second member to be applied like limiting conditions to the edges of the model.

3.4.1   Operand COEFF

◊   COEFF =   /   1.0, [DEFECT]
                     /   coeF

Give the factor of combination to apply a loading in the case of of type 'FORCE'. It can be necessary to apply factors -1 in the case of wave of pressure P or in the case of wave of shearing S on the fields of nodal forces calculatedS on model to auxiliairE result of origin entered behind the keyword RESU_INIT.

Code_Aster

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Version default

Date : 04/11/2019 Page : 5/6
Clé : U4.84.32   Révision   :
4261bd411366

3.4.2 Operands **MATR_AMOR** / **MATR_RIGI**

♦   MATR_ AMOR = matr_ has
◊   MATR_RIGI = matr_k,

In the case of one loading of the type 'ONDE_PLANE', NomS of assembled matrices of damping and of rigidity (optional) that one uses in the products $KU + CV$ constituting the evolution second member produced from which one can create the load to be applied to the model.

3.4.3 Operands **NUME_DDL** / **MATR_RIGI**

◊   / MATR_ RIGI = matr_ K
       / NUME_DDL = digital ,

In the case of one loading of the type 'FORCE', Entrées from which one can obtain the classification of reference or conversion of the evolution second member produced from which one can create the load to be applied to the model.

3.4.4 Operands **INST** / **LIST_INST** / **PRECISION** / **CRITERION**
◊   INST = litps

List of actual values moments of calculation DU result of origin.

◊   LIST_INST = litps

List of realities produced by DEFI_LIST_REEL [U4.34.01] moments of calculation of the produced evolutions.

◊   PRECISION = / prec   [R]
                   / 1.0D-6   [DEFECT]

Precision used to search the moment specified in the result of origin .

◊   CRITERION =   / 'RELATIVE',
                       / 'ABSOLUTE,'

Criterion used to search the moment specified in the result of origin.

3.4.5 Operand **NOM_CHAM_INIT**

♦   NOM_CHAM_INIT =   / 'ACCE',
                         / 'FORC_NODA',
                         / 'PULLEYC_NODA',

Name of field of result DE L'evolution of departure calculated of type evol_char or dyna_tranS that one converts then in the field of the result of the evolution second member DE type evol_char transform in load.

3.4.6 Operand **DDL_EXCLUS**

◊   DDL_EXCLUS =   nom_cmp   [TXM]

Name of component to be excluded at once result DE L"evolution of departure calculated of type evol_char or dyna_tranS that one converts then in the field of the result of the evolution second member DE type evol_char transform in load. Applies in general to a field 'FORC_NODA'. All the field is taken if the keyword is not indicated.

3.5   Operand **TITLE**

◊   TITLE = title

Code_Aster

Titre : Commande DEFI_CHAR_SOL
Responsable : DEVESA Georges

Version default

Date : 04/11/2019 Page : 6/6
Clé : U4.84.32   Révision :
4261bd411366

Allows the user to define a title for LE loading.

3.6   Operand INFORMATION

◊   INFORMATION =

Indicate a level of impression for information in the file "MESSAGE":

INFORMATION = 1 : no impression
INFORMATION = 2 : impression information on the loading.