Operator CALC_TRANSFER

1 Goal

This operator allows to calculate the matrix of transfer functions transfer dynamic of a linear mechanical system between two points of the digital model. The transfer transfer functions can be in acceleration, speed or displacement. One can treat the case of absolute or relative movements, due to one request of training.

It is also possible to obtain the answer in an unspecified point of the digital model according to an output signal indicated by the user.

This operator uses in data input the answers of the mechanical system to one-way requests, preferably of type white vibration.

It produces one or two structures of data of the type table, one for the matrix of transfer transfer functions, the other for the possible calculated signal.
2 Syntax

CALC_TRANSFERT ( 

♦ RESULTAT_X = resu_x , 
  / [dyna_trans] 
  / [dyna_harmo] 
  / [harm_gene] 
  / [tran_gene] 

♦ RESULTAT_Y = resu_y , 
  / [dyna_trans] 
  / [dyna_harmo] 
  / [harm_gene] 
  / [tran_gene] 

◊ RESULTAT_Z = resu_z, 
  / [dyna_trans] 
  / [dyna_harmo] 
  / [harm_gene] 
  / [tran_gene] 

♦ NOM_CHAM = / 'ACCE', 
  / 'QUICKLY', 
  / 'DEPL', 

♦ ENTRY = _F ( 
  ♦ GROUP_NO = grno, 
  , [gr_noeud] 

♦ EXIT = _F ( 
  ♦ GROUP_NO = grno, 
  , [gr_noeud] 

◊ REFERENCE MARK = / 'ABSOLUTE' , 
  / 'RELATIVE' , 
  / [DEFECT] 
  ♦ TRAINING = _F ( 
    ♦ DX = entr_x, 
    , [function] 
    ♦ DY = entr_y, 
    , [function] 
    ♦ DZ = entr_z, 
    , [function] 

◊ SIGNAL = _F ( 
  ♦ MESURE_X = f_x, 
  , [function] 
  ♦ MESURE_Y = f_y, 
  , [function] 
  ♦ MESURE_Z = f_z, 
  , [function] 
  ♦ TABLE_RESU = CO ('table'), ), 
  , [table] 
  ♦ TYPE_RESU = / 'TEMPORAL', 
  / 'HARMONIC', 
  , [DEFECT] 

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.
Copyright 2020 EDF R&D - Licensed under the terms of the GNU FDL (http://www.gnu.org/copyleft/fdl.html)
3 Description of the macro order and preliminary stages

The operator **CALC_TRANSFERT** allows to calculate the matrix dynamic transfer function transfer between two points of the structure (entered => left). It also makes it possible to the user to determine the signal in a point of the structure (as starter) knowing the signal in another point of the structure (at exit).

Prior to the use of the order, the user must carry out several linear dynamic calculations under one-way request. For example if the study is in 3D (resp. 2D), it must carry out 3 calculations (resp. 2 calculations) exciting one of the three (resp. 2) directions. It is preferable to carry out these dynamic calculations via a mono-directional white vibration.

Dynamic calculations must be carried out with the same list of frequencies if calculation is harmonic and with the same list of moments if calculation is transitory. For a transitory calculation, the user must take care to have a list of moments to constant step. In the contrary case an error message will be transmitted to him because the transform of Fourier necessary to various calculations requires a constant step.

Dynamic calculations will have to be in the same way standard, that is to say **tran_gene**, that is to say **harm_gene**, that is to say **dyna_trans** or **dyna_harmo**.

When the user must inform the functions of training and/or the signals measured, it is preferable that the latter are discretized on the same list of frequencies (case of a harmonic calculation) or on the same list of time (case of a transitory calculation).

4 Operands

4.1 Concepts created by the order

This order creates at exit a concept of the type **table**. It is possible to also leave a second **table** containing the signals corresponding to dynamic stress (as starter) calculated using the measured signals (at exit).

The first concept produces whatever the use of the macro-order by the user is one **table** who contains the data of the matrix transfer functions transfer arranged as follows:

- The first column corresponds to the frequencies (**FREQ**); 
- The following ones correspond to the components of the matrix arranged in the order by lines **$H_{xx}$**, **$H_{xy}$**, **$H_{xz}$**, **$H_{yx}$**, **$H_{yy}$**, **$H_{yz}$**, **$H_{zx}$**, **$H_{zy}$** and **$H_{zz}$**. In the case 2D, one will have **$H_{xx}$**, **$H_{xy}$**, **$H_{yx}$** and **$H_{yy}$**.
### 4.2 Operands RESULTAT_X, RESULTAT_Y and RESULTAT_Z

- **RESULTAT_X** = resu_x
  
  Name of the concept of the type *Resultat* for which dynamic calculation linear was carried out with a request one-way according to $x$.

- **RESULTAT_Y** = resu_y
  
  Name of the concept of the type *Resultat* for which linear dynamic calculation was carried out with an one-way request according to $y$.

- **RESULTAT_Z** = resu_z
  
  Name of the concept of the type *Resultat* for which linear dynamic calculation was carried out with an one-way request according to $z$.

**Note:**

*Only the structures result of the type dyna_trans, dyna_harmo, tran_gene and harm_gene can be well informed in these operands.*

*The calculations produced by the operators of dynamics and who are to be informed in the operands RESULTAT_X, RESULTAT_Y and RESULTAT_Z must be in the same way standard: dyna_trans, or dyna_harmo or tran_gene or harm_gene.*

### 4.3 Operand NOM_CHAM

- **NOM_CHAM** = nomcha

  Reference symbol of the field which one wishes to extract from dynamic calculations linear realized as a preliminary by the user and informed in the operands RESULTAT_X, RESULTAT_Y and RESULTAT_Z.

  The possible names of the field are: ‘DEPL’, ‘QUICKLY’ and ‘ACCE’.

**Note:**

*Various functions indicated by the user such as the functions of training indicated in the word-key TRAINING or functions measured well informed in the keyword SIGNAL will have to be of the same type as nomcha.*

### 4.4 Operand REFERENCE MARK

- **REFERENCE MARK** = /'ABSOLUTE'/
  /'RELATIVE'/

  This obligatory operand makes it possible to inform the type of answer calculated by the dynamic operators. If the reference mark is *RELATIVE* (for one linear dynamic calculation moving relative), then the user must inform the signal of training in the keyword TRAINING (see paragraph 4.5).

### 4.5 Keyword factor TRAINING
The keyword factor `TRAINING` allows to inform the requests of training having been used to carry out various linear dynamic calculations moving relative.

**Note:**

> Various functions of training indicated in the word-key `TRAINING` will have to be of the same type as `nomcha` ("DEPL", "QUICKLY" or "ACCE").

### 4.5.1 Operands DX, DY and DZ

- **DX = entr_x**
  
  This operand makes it possible to inform the one-way request (according to the direction $x$) used for the dynamic emitter calculation.

- **DY = entr_y**
  
  This operand makes it possible to inform the one-way request (according to the direction $y$) used for the second dynamic calculation.

- **DZ = entr_z**
  
  This operand makes it possible to inform the one-way request (according to the direction $z$) used for the third dynamic calculation.

### 4.6 Keywords factor `ENTRY` and `EXIT`

#### 4.6.1 Operand `GROUP_NO`

- **GROUP_NO = grno**

  The obligatory keywords factor `ENTRY` and `EXIT` allow to inform the name of a group of node (grno, container only one node) on which carry out the extraction of the field "DEPL" or "QUICKLY" or "ACCE".

### 4.7 Keyword factor `SIGNAL`

This keyword factor (optional) makes it possible to determine the dynamic stress (as starter) having led to the measured signals (at exit) indicated by the user in the keyword factor `SIGNAL`.

#### 4.7.1 Operands `MESURE_X`, `MESURE_Y` and `MESURE_Z`

- **MESURE_X = f_x**
  
  This operand makes it possible to inform the signal measured according to the direction $x$.

- **MESURE_Y = f_y**
  
  This operand makes it possible to inform the signal measured according to the direction $y$.

- **MESURE_Z = f_z**
  
  This operand makes it possible to inform the signal measured according to the direction $z$.
Note:

The signals must be in the same way standard than nomcha and must be discretized over the same list of frequencies or time.

4.7.2 Operand TYPE_RESU

♦ TYPE_RESU = '/TEMPORAL',
   /'HARMONIC'

This operand makes it possible to inform the type of results wished at exit of table.

4.7.3 Operand TABLE_RESU

♦ TABLE_RESU = nom_table

This operand makes it possible to inform the name of table in which will appear the calculated signals knowing the measured signals.

table will be made up of several columns arranged like such:

If TYPE_RESU = ‘TEMPORAL’:
   INST, FX, FY, FZ

If TYPE_RESU = ‘HARMONIC’:
   FREQ, Re_FX, Im_FX, Re_FY, Im_FY, Re_FZ, Im_FZ