Operator CALC_MODE_ROTATION

1  Goal

To calculate the modes and the frequencies of the system following according to the number of revolutions,

\[ M \ddot{\delta} + (C + \Omega G) \dot{\delta} + K \delta = 0 \]

Where \( M \) is the matrix of mass of the system, \( C \) is a matrix of damping, \( G \) is the matrix of gyroscopy (antisymmetric), and \( K \) is the matrix of stiffness of the system. \( \Omega \) represent the number of revolutions.

The data necessary for this macro are:
1) matrices: \( K, C, G \) and \( M \)
2) A list number of revolutions

This operator returns a list of concept mode_meca_c: a concept for each number of revolutions. She calls on the order CALC_MODES.
2 Syntax

```
CALC_MODE_ROTATION (

# Matrix of rigidity
   ♦ MATR_RIGI = K [matr_asse_depl_r]

# Matrix masses
   ♦ MATR_MASS= M [matr_asse_depl_r]

# Matrix damping
   ♦ MATR_AMOR = C [matr_asse_depl_r]

# Gyroscopic matrix
   ♦ MATR_GYRO = G [matr_asse_depl_r]

# List number of revolutions
   ♦ VITE_ROTA = List [R]

# Choice of the method
   ♦ METHOD = / ‘QZ’ [DEFECT]
   ♦          / ‘SORENSEN’

# Type of modal calculation
   ◊ CALC_FREQ = _F (  
      ◊ OPTION = / ‘CENTER’
      ◊          / ‘PLUS PETITE’ [DEFECT]
      ◊ NMAX_FREQ = nbF [I]
      ◊ SEUIL_FREQ= /1.E-2 [DEFECT]
      ◊          /f_seuil [R]
   )

# For final checks
   ◊ VERI_MODE = _F (  
      ◊ STOP_ERREUR = / ‘YES’ [DEFECT]
      ◊          / ‘NOT’
      ◊ THRESHOLD = / 1.E-6 [DEFECT]
      ◊          / R [R]
      ◊ PREC_SHIFT = / 0.05 [DEFECT]
      ◊          / prs [R]
      ◊ STURM = / ‘YES’ [DEFECT]
      ◊          / ‘NOT’
   )
```

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 2019 EDF R&D - Licensed under the terms of the GNU FDL (http://www.gnu.org/copyleft/fdl.html)
3 Operands

3.1 Operands

\[
\text{MATR_RIGI/MATR_MASS/}
\]
\[
\text{MATR_AMOR/MATR_GYRO/INFORMATION/METHOD/OPTION}
\]

They have the same meaning as in the order \text{CALC_MODES} [U4.52.02].

Note:

Because of presence of the matrices of damping and gyroscopy, only methods \text{QZ} and \text{SORENSEN} are usable.

3.2 Keyword \text{CALC_FREQ}

Play the same part as in the order \text{CALC_MODES} [U4.52.02], has the same internal keywords with the same values by default.

Note:

The number of modes \( nbF \) is the same one for all the number of revolutions.

3.3 Operand \text{VITE_ROTA}

List number of revolutions \( \Omega \) in \( \text{rad/s} \).

3.4 Operand Keyword \text{VERI_MODE}

The internal operands have the same meaning as in of the same keyword name of order \text{CALC_MODES} [U4.52.02].

4 Example

```python
# Calculation of the first 5 modes in rotation by using the method \text{QZ}:

Lmod=CALC_MODE_ROTATIONR (MATR_RIGI = RIGIDITY,
                         MATR_MASS = MASS,
                         MATR_AMOR=AMOR,
                         MATR_GYRO =GYASS,
                         VITE_ROTA=L_VITROT,
                         METHOD = 'QZ',
                         CALC_FREQ=_F (OPTION=' PLUS_PETITE', NMAX_FREQ=5),
                         VERI_MODE=_F (STOP_ERREUR=' NON'));
```

\text{CALC_MODE_ROTATION} return a table (table_contenor) containing the modal bases calculated for each number of revolutions.

\text{mode_meca_c product are named as follows: mod_0,… mod_i. .mod_nbV, i} is the index number of revolutions in \text{VITE_ROTA}.

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 2019 EDF R&D - Licensed under the terms of the GNU FDL (http://www.gnu.org/copyleft/fdl.html)