

Operator INFO_MODE

1 Goal

This order allows to **count** and to save (in one `sd_table`) **the number of eigenvalues understood** in:

- One **interval of the real axis** (for the generalized problems GEPs standard with symmetrical real matrices).
- One **disc of the complex plan** (for all the other the quadratic cases, other GEPs and all problems QEPs).

When one deals with GEPs standards, the execution of this procedure requires generally only one factorization LDLT by mode. When they are disseminated in the complex plan, the procedure is much more expensive. In this case, case it is thus rather to hold for the simplified problems smalls (lower than 1000 degrees of freedom).

The call to this operator is recommended like **checking/calibration *a priori* model**. It also allows to define **balanced intervals of research and containing a reasonable number eigenvalues** (typically a few tens). And this, in order to optimize the costs of the later modal operators and not to degrade the precision of their results.

To treat a standard GEP effectively¹, one proposes to proceed in several stages:

- **To gauge** zones of interest by precalculations only using `INFO_MODE` (into sequential or, if possible, parallel) on lists of frequencies (resp. critical loads) given;
- **To start again** one or more calculations `CALC_MODES` with `OPTION='BAND'` cut out on several sub-bands, while basing itself on these initial calibrations.

For the supercomputing, the operator `INFO_MODE2`, can profit from two levels of parallelism. The savings of time can reach a factor 70 on a hundred processors, and the profits in peak report a factor 2. The implementation of parallelism is described in the dedicated paragraph of documentation [U2.06.01].

This operator produces one concept `sd_table`.

In a first approach one can be satisfied to inform the parameters: `MATR_*`, `TYPE_MODE` and `FREQ` (or `CHAR_CRIT` or `*_CONTOUR`).

This operator is complementary to the operator `CALC_MODES` [U4.52.02].

1 Currently, very majority of the cases.

2 Just like `CALC_MODES` with `OPTION='BAND'` cut out on several sub-bands.

Contents

1 Goal.....	1
2 Syntax.....	3
3 Operands.....	6
3.1 Principles.....	6
3.2 Characteristics of the modal problem: operands	
MATR_RIGI/MATR_A/MATR_MASS/MATR_RIGI_GEOM/MATR_B/MATR_AMOR/MATR_C.....	8
3.3 Operand TYPE_MODE.....	9
3.4 Operand FREQ.....	9
3.5 Operands CHAR_CRIT.....	10
3.6 Operands TYPE/RAYON/CENTRE_CONTOUR.....	10
3.7 Keyword factor COUNTING.....	11
3.8 Operand INFORMATION.....	13
3.9 Operand TITLE.....	13
3.10 Keyword factor SOLVEUR.....	13
3.11 Operand NIVEAU_PARALLELISME.....	14
4 Examples.....	17
4.1 N°1 example.....	17
4.2 N°2 example.....	17

2 Syntax

```
Tb [table_sdaster] = INFO_MODE
```

Standard of problem

```
◇ TYPE_MODE= / 'DYNAMIC' , [DEFECT]
              / 'MODE_FLAMB' ,
              / 'MODE_COMPLEXE' ,
              / 'GENERAL' .
```

```
# If TYPE_MODE= 'DYNAMIQUE'
```

Characteristic of calculation

```
◇ MATR_RIGI= With , / [matr_asse_depl_r]
              / [matr_asse_depl_c]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
              / [matr_asse_gene_c]
◇ MATR_MASS= B , / [matr_asse_depl_r]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
```

```
# the number of frequencies delimiting the intervals of calculation is noted nb_freq
```

```
◇ FREQ = l_f [l_R]
```

Parameters of the method of counting of STURM

```
◇ COMPTAGE=_F (
  ◇ METHOD = / 'CAR' , [DEFECT]
            / 'STURM' .
  ◇ SEUIL_FREQ = / f_seuil [R]
                / 0.01 [DEFECT]
  ◇ PREC_SHIFT = / p_shift [R]
                / 0.05 [DEFECT]
  ◇ NMAX_ITER_SHIFT=/ n_shift [I]
                    / 3 [DEFECT]
)
```

```
# If TYPE_MODE=
'MODE_FLAMB'
```

Characteristic of calculation

```
◇ MATR_RIGI= With , / [matr_asse_depl_r]
              / [matr_asse_depl_c]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
              / [matr_asse_gene_c]
◇ MATR_RIGI_GEOM= B , / [matr_asse_depl_r]
                  / [matr_asse_temp_r]
                  / [matr_asse_pres_r]
                  / [matr_asse_gene_r]
```

```
# the number of critical loads delimiting the intervals of calculation is noted nb_mode_flamb
```

```
◇ CHAR_CRIT=l_c [l_R]
```

Parameters of the method of counting of STURM

```

◇ COMPTAGE=_F (
  ◇ METHOD = / 'CAR', [DEFECT]
              / 'STURM'.
  ◇ SEUIL_CHAR_CRIT = / c_seuil [R]
                      / 0.01 [DEFECT]
  ◇ PREC_SHIFT = / p_shift [R]
                 / 0.05 [DEFECT]
  ◇ NMAX_ITER_SHIFT = / n_shift [I]
                     / 3 [DEFECT]
)

```

If TYPE_MODE= 'MODE_COMPLEXE'

Characteristic of calculation

```

◆ MATR_RIGI= With , / [matr_asse_depl_r]
              / [matr_asse_depl_c]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
              / [matr_asse_gene_c]
◆ MATR_MASS= B , / [matr_asse_depl_r]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
◇ MATR_AMOR= C , (only in QEP)/ [matr_asse_depl_r]
              / [matr_asse_temp_r]
              / [matr_asse_pres_r]
              / [matr_asse_gene_r]
◇ TYPE_CONTOUR = 'CIRCLE' [DEFECT]
◆ RAYON_CONTOUR = ray [R]
◇ CENTRE_CONTOUR = / center [C]
                  / 0.0+0.0j [DEFECT]

```

Parameters of the method of counting APM

```

◇ COMPTAGE=_F (
  ◇ METHOD = / 'CAR', [DEFECT]
              / 'APM'.
  ◇ NBPOINT_CONTOUR = / nb_point [I]
                     / 40 [DEFECT ]
  ◇ NMAX_ITER_CONTOUR= / n_contour [I]
                      / 3 [DEFECT]
)

```

If TYPE_MODE= 'GENERAL'

Characteristic of calculation

```

◆ MATR_A= With , / [matr_asse_depl_r]
           / [matr_asse_depl_c]
           / [matr_asse_temp_r]
           / [matr_asse_pres_r]
           / [matr_asse_gene_r]
           / [matr_asse_gene_c]
◆ MATR_B= B , / [matr_asse_depl_r]
            / [matr_asse_temp_r]

```

```

/ [matr_asse_pres_r]
/ [matr_asse_gene_r]
◇ MATR_C= C , (only in QEP)/ [matr_asse_depl_r]
/ [matr_asse_temp_r]
/ [matr_asse_pres_r]
/ [matr_asse_gene_r]
◇ CHAR_CRIT =l_c [l_R]

# If CHAR_CRIT==None
◇ TYPE_CONTOUR = 'CIRCLE' [DEFECT]
◇ RAYON_CONTOUR = ray [R]
◇ CENTRE_CONTOUR = / center [C]
/ 0.0+0.0j [DEFECT]
◇ COMPTAGE=_F (
◇ METHOD = / 'CAR', [DEFECT]
/ 'STURM', (if CHAR_CRIT )
/ 'APM'. (if *_CONTOUR)

# Parameters of the method of counting_
# If STURM
◇ SEUIL_CHAR_CRIT = / c_seuil [R]
/ 0.01 [DEFECT]
◇ PREC_SHIFT = / p_shift [R]
/ 0.05 [DEFECT]
◇ NMAX_ITER_SHIFT = / n_shift [I]
/ 3 [DEFECT]

# If APM
◇ NBPOINT_CONTOUR = / nb_point [I]
/ 40 [DEFECT ]
◇ NMAX_ITER_CONTOUR= / n_contour [I]
/ 3 [DEFECT]
)

# Solveur linear and parallelism
◇ SOLVEUR=_F(For more details to see the document [U4.50.01]).
# In parallel, one advises the parameter setting particularly METHODE=' MUMPS'+RENUM=' QAMD' .
◇ NIVEAU_PARALLELISME = / 'COMPLETE', [DEFECT]
/ 'PARTIAL'.

# Activated only in parallel mode ( nb_proc >1) and sells by auction if METHODE=' STURM' .
# L 'option' COMPLETE 'functions some is the direct linear solvor, if nb_proc = nb_freq -1 (resp.
nb_mode_flamb -1). With the option 'PARTIAL , only ' SOLVEUR=_F (METHODE=' MUMPS') is
licit.

# Others
◇ INFO= / 1 [DEFECT]
/ 2
◇ TITRE= tit [KN]
) ;
```

3 Operands

3.1 Principles

This order makes it possible to count the number of eigenvalues understood in (cf appears 3ab):

- An interval of the real axis (for the generalized problems GEPs standard with symmetrical real matrices): keyword `TYPE_MODE=' DYNAMIQUE'` or `'MODE_FLAMB 'or' GENERAL'`.
- A disc of the complex plan (for all the other the quadratic cases, other GEPs and all problems QEPs): keyword `TYPE_MODE=' MODE_COMPLEXE'` or `'GENERAL'`.

This number of clean modes is traced in the file message for each interval of the real axis or for the portion of the plan complexes selected (cf figure 1). One points out also the characteristics of these zones of interest.

```

-----
CHECKING OF THE FREQUENCY SPECTRUM (METHOD OF STURM)

THE NUMBER OF FREQUENCIES IN THE BAND NR    1 OF
TERMINALS (0.000E+00, 1.000E+01) EAST    20
-----

THE NUMBER OF FREQUENCIES IN THE BAND NR    2 OF
TERMINALS (1.000E+01, 2.000E+01) EAST    43
.....

THE NUMBER OF FREQUENCIES IN THE BAND NR    6 OF
TERMINALS (5.000E+01, 6.000E+01) EAST   139
-----
    
```

Figure 1. *_Postings in the file message of one INFO_MODE on a series of frequencies $FREQ=(0. , 10. , 20. , ..., 50. , 60.)$ for a problem of DYNAMICS.*

All these elements are stored in one `sd_table` (cf appears 2ab) that one can print (via one `IMPR_TABLE`) or to re-use (by chainings with `CALC_MODES`). The overcost of this operator can then become almost negligible since one pools the total information released with other modal operators.

```

#TABLE_SDASTER
FREQ_MIN    FREQ_MAX    BORNE_MIN_EFFECT    BORNE_MAX_EFFECT    NB_MODE
0.00000E+00  1.00000E-02    0.00000E+00        1.00000E-02        3
1.00000E-02  5.00000E-02    1.00000E-02        5.00000E-02        3
5.00000E-02  5.50000E-02    5.00000E-02        5.50000E-02        1
5.50000E-02  6.00000E-02    5.50000E-02        6.00000E-02        0
6.00000E-02  1.00000E-01    6.00000E-02        1.00000E-01        3

#TABLE_SDASTER
CHAR_CRIT_MIN  CHAR_CRIT_MAX  BORNE_MIN_EFFECT  BORNE_MAX_EFFECT  NB_MODE
-1.00000E+06  -5.00000E+05  -1.00000E+06     -5.00000E+05     1
-5.00000E+05  0.00000E+00   -5.00000E+05     0.00000E+00     3
0.00000E+00   1.00000E+05   0.00000E+00     1.00000E+05     0
1.00000E+05   1.10000E+06   1.00000E+05     1.10000E+06     5
    
```

Figure 2ab. *_N°1 Example of table generated by one INFO_MODE on a series of frequencies $FREQ=(0. , 1.E-2, ..., 1.E-1)$ for a problem of dynamics. Extract of *sdlx201a.mess*. N°2 example of table generated by one INFO_MODE on a series of critical loads $CHAR_CRIT=(-1.E+6, -5.E+5, ..., 1.1E+6)$ for a problem of buckling. Extract of *sdlx504a.mess**

The call to this operator is recommended like checking/calibration *a priori* model. It also makes it possible to define intervals of research containing a reasonable number of eigenvalues (typically about forty modes) in order to optimize the costs of the later modal operators and not to degrade the precision of their results.

Typically, one advises to search the modes per packages of approximately forty³. Beyond that, consomsubdued in time and in memory are not optimal any more and the quality of the modes obtained is degraded. It is then to better use several wavebands (or of critical loads) directly with CALC_MODES and the option 'BAND' cut out on several sub-bands, or by several calls to CALC_MODES [U4.52.02].

Ideally, to treat a standard GEP effectively, one proposes to proceed in several stages:

- **To gauge** zones of interest by precalculations only using INFO_MODE (into sequential or, if possible, parallel) on lists of frequencies (resp. critical loads) given;
- **To start again** one or more calculations CALC_MODES with OPTION=' BANDE ' while basing itself on these initial calibrations.

To save time, one even can **to pool** (and it is strongly advised!) part of the cost calculation of INFO_MODE initial while notifying with CALC_MODES the name of sd_table generated. This chaining can thus return the overcost of INFO_MODE negligible and to guide modal calculation effectively.

Dimensioned supercomputing, the operator INFO_MODE, just like CALC_MODES with the option 'BAND' cut out in several sub-bands, profit potentially from two levels of parallelism (cf keyword NIVEAU_PARALLELISME). From where savings of time of a factor up to 70 on a hundred processors and of the profits in peak report until a factor 2.

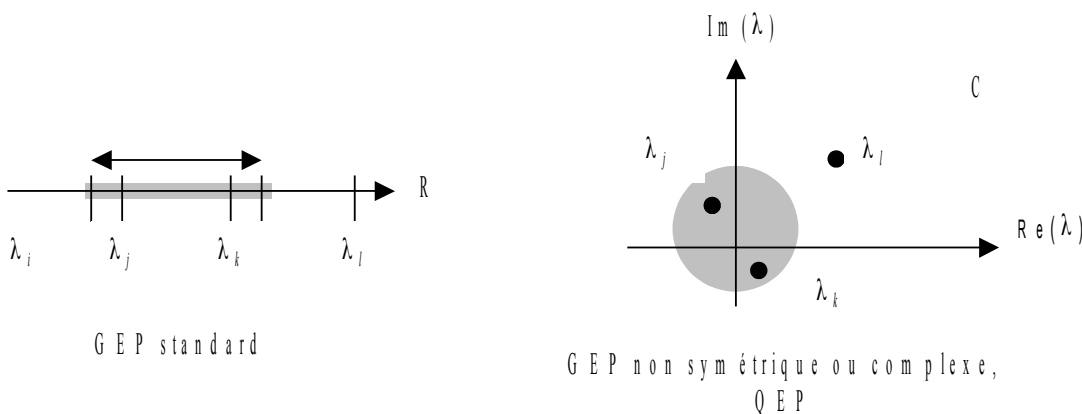


Figure 3ab. _Two distinct problems of enumeration: in a segment of the real axis and a finished portion of the complex plan (for the moment that a disc).

In the first case, when one seeks to count the eigenvalues strictly included in a segment of the real axis, one uses the method known as of "Sturm", in the second case method APM. Their theories and algorithmies are detailed in the reference material [R5.01.04].

When one deals with GEPs standards, the execution of this procedure generally does not require⁴ that a factorization LDLT by mode (METHODE=' STURM '). When they are disseminated in the complex plan, the procedure is much more expensive (METHODE=' APM ') because it requires, in general, of the hundreds of factorizations. In this case, case it is thus rather to hold for the simplified problems smalls (lower than 1000 degrees of freedom).

3 In sequential mode. With CALC_MODES option 'BAND' cut out on several sub-bands parallel, one can more quickly obtain a calculation by taking only about twenty modes.

4 Except if the value is close to a clean mode, in which case one it "shifts" (cf keyword NMAX_ITER_SHIFT) to limit certain digital instabilities.

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 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

Let us note that this last method APM is still the object of active research. Its robustness, for the moment, is thus not guaranteed. It is to be used while having taken well care to read associated documentation.

3.2 Characteristics of the modal problem: operands MATR_RIGI/MATR_A/MATR_MASS/MATR_RIGI_GEOM/MATR_B/MATR_AMOR/MATR_C

The table below represents the operands to be used according standard to the keyword TYPE_MODE.

TYPE_MODE			
'DYNAMIC'	'MODE_FLAMB'	'MODE_COMPLEXE'	'GENERAL'
◆ MATR_RIGI = With	◆ MATR_RIGI = WITH	◆ MATR_RIGI = WITH	◆ MATR_A = WITH
◆ MATR_MASS = B	◆ MATR_RIGI_GEOM = B	◆ MATR_MASS = B	◆ MATR_B = B
Without Object	Without Object	◇ MATR_AMOR = C	◇ MATR_C = C

These operands make it possible to inform the matrices (assembled or generalized⁵) characterizing the modal problem. These matrices can be symmetrical or not. They are real except the matrix **A** who can be, either real, or complex.

The data of **A** and **B** allows to define the generalized modal problem (GEP) (cf [R5.01.01]) studied

$$\mathbf{A}\mathbf{u} = \lambda \mathbf{B}\mathbf{u} \quad (\text{GEP}_{\text{dynamique}})$$

In the classical case of dynamics (TYPE_MODE=' DYNAMIQUE'), **A** is the matrix of rigidity (symmetrical real) and **B** and stamps it of mass (symmetrical real). The eigenvalue λ is then connected to the Eigen frequency f by the formula: $\lambda = (2\pi f)^2$. It is real positive.

In the case of the theory of linear buckling (TYPE_MODE=' MODE_FLAMB'),

$$\mathbf{A}\mathbf{u} - \lambda \mathbf{B}\mathbf{u} = \mathbf{0} \quad (\text{GEP}_{\text{flambement}})$$

A is the matrix of rigidity (symmetrical real) and **B** the geometrical matrix of rigidity (symmetrical real). The eigenvalue λ critical load is called. It is real.

As mentioned in the preceding table, these matrices will be indicated, respectively, for dynamics by MATR_RIGI/MATR_MASS and, in buckling, by MATR_RIGI/MATR_RIGI_GEOM. Only method known as of 'STURM' is licit in these two cases.

When one of the two matrices is not symmetrical any more or comprises complex terms (e.g. to take into depreciation account hysteretic), the eigenvalues are potentially complex. To deal with this problem it is necessary to initialize TYPE_MODE with 'MODE_COMPLEXE' and only method APM is then licit.

The data of **A**, **B** and **C** allows to define the quadratic modal problem (QEP) (cf [R5.01.02]) studied

$$(\mathbf{A} + \lambda \mathbf{C} + \lambda^2 \mathbf{B})\mathbf{u} = \mathbf{0} \quad (\text{QEP})$$

In this case, case the eigenvalues are potentially complex. To deal with this problem it is necessary to initialize TYPE_MODE with 'MODE_COMPLEXE'. Often **A** is the matrix of rigidity, **C**

5 This concept of generalized matrix has nothing to do with that of generalized modal problem!

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 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

a matrix taking of account the viscous damping and/or of the effects gyroscopic and **B** the geometrical matrix of rigidity. As mentioned in the preceding table, these matrices will be indicated by MATR_RIGI/MATR_MASS/MATR_AMOR and only method APM is licit.

TYPE_MODE=' GENERAL' allows to solve a problem of eigenvalues in the case general. One then makes play matrices **A**, **B** and/or **C** the role which one wishes. As mentioned in the preceding table, these matrices will be indicated by MATR_A/MATR_B/MATR_C. According to the cases, only the method of Sturm (if MATR_C is not well informed and if the two other matrices are real symmetrical) or method APM (all other cases) are licit.

3.3 Operand TYPE_MODE

◇ TYPE_MODE= / 'DYNAMICS' [DEFECT]
/ 'MODE_FLAMB'
/ 'MODE_COMPLEXE'
/ 'GENERAL'

This keyword makes it possible to define the nature of the modal problem to treat: pre-to estimate the frequency spectrum of vibration (case classical of 'DYNAMIC'), to do it in term of critical loads (case of the theory of linear buckling 'MODE_FLAMB') or to gauge that of a nonstandard GEP or a QEP in a portion of the complex plan ('MODE_COMPLEXE'). One can too to search the eigenvalues and the modes associated with a matrix system general ('GENERAL'). For more precise details one will be able to refer to the description of §3.2.

With resulting from this choice, the user must specify the characteristics of his control field:

- for 'DYNAMIC': list of frequencies *via* FREQ,
- for 'MODE_FLAMB': list of critical loads *via* CHAR_CRIT,
- for 'MODE_COMPLEXE': zone of the complex plan *via* TYPE/RAYON/CENTRE_CONTOUR,
- for 'GENERAL': according to the case, one is proceeds as for 'MODE_FLAMB' (if MATR_C is not well informed and if the two other matrices are real symmetrical), maybe as for 'MODE_COMPLEXE' (all other cases).

3.4 Operand FREQ

◇ FREQ = l_f

List of frequencies (in Hertz) defining the intervals which one wants to gauge $l_f = (f_i)_i$ (one notes nb_freq the number of frequencies of this list). One then searches the number of eigenvalues in the sub-bands $[\lambda_i, \lambda_{i+1}]$ with $\lambda_* = (2\pi f_*)^2$. This list must comprise at least two values. These values must be arranged by order strictly growing and all positive. This keyword must be used if TYPE_MODE= 'DYNAMIQUE'. Only method of counting of the type STURM is then licit.

Note:

- *Each frequency is treated only once: as a lower limit of the first sub-band for the first of the list, as an upper limit of the sub-bands which follow for the other frequencies. In particular, if this frequency is judged too near to an eigenvalue, it is shifted (possibly several times of parameters PREC_SHIFT/NMAX_ITER_SHIFT etc). This shift "of the shift" is always operated towards the external one of the sub-band considered.*
- *If the shift of the lower limit makes it larger than the upper limit, the process stops in ERREUR_FATALE.*
- *This mode of shift makes more robust and consisting the later re-uses of this sd_table by one or more CALC_MODES. One avoids in particular coverings of sub-bands which could occur until V10.*

- Initial terminals of each one of these sub-bands are saved in the fields `FREQ_MIN/MAX` of each line of the produced table (cf appears 2a). For each one of these couples, one mentions the terminals actually used by the test of Sturm (i.e. after possible shifts), `BORNE_MIN/MAX_EFFECT`, as well as the number of found modes: `NB_MODE`.

3.5 Operands CHAR_CRIT

`CHAR_CRIT = l_c`

List of critical loads defining the intervals which one wants to gauge $l_c = (\lambda_i)_i$ (one notes `nb_mode_flamb` the number of critical loads of this list). One then searches the number of eigenvalues in the sub-bands $[\lambda_i, \lambda_{i+1}]$. This list must comprise at least two values. These values must be arranged by order strictly growing. They must be of real type.

This keyword must be used if `TYPE_MODE= 'MODE_FLAMB'` or if `TYPE_MODE= 'GENERAL'` (if `MATR_C` is not well informed and if the two other matrices are real symmetrical). Only method of counting of the type `STURM` is licit.

Note:

- Each critical load is treated only once: as a lower limit of the first sub-band for the first of the list, as an upper limit of the sub-bands which follow for the other frequencies. In particular, if this load is judged too near to an eigenvalue, it is shifted (possibly several times cf parameters `PREC_SHIFT/NMAX_ITER_SHIFT` etc). This shift "of the shift" is always operated towards the external one of the sub-band considered.
- If the shift of the lower limit makes it larger than the upper limit, the process stops in `ERREUR_FATALE`.
- This mode of shift makes more robust and consisting the later re-uses of `sd_table` by one or more `CALC_MODES`. One avoids in particular coverings of sub-bands which could occur until V10.
- Initial terminals of each one of these sub-bands are saved in the fields `CHAR_CRIT_MIN/MAX` of each line of the produced table (cf appears 2b). For each one of these couples, one mentions the terminals actually used by the test of Sturm (i.e. after possible shifts), `BORNE_MIN/MAX_EFFECT`, as well as the number of found modes: `NB_MODE`.

3.6 Operands TYPE/RAYON/CENTRE_CONTOUR

If `TYPE_MODE= 'MODE_COMPLEXE'` or if `TYPE_MODE= 'GENERAL'` (if `MATR_C` ou/et is indicated if one of the matrices is nonsymmetrical or complex)

```
◇ TYPE_CONTOUR = 'CIRCLE' [DEFECT]
◆ RAYON_CONTOUR= ray [R]
◇ CENTRE_CONTOUR = / center [C]
/ 0.0+0.0j [DEFECT]
```

These keywords define the type of contour inside which one seeks the eigenvalues like his characteristics. For the moment one can choose only one contour of the type 'CIRCLE'(which thus delimits the disc of control). Sound is mentioned `center` (a complex number) and sound `ray` (a positive reality).

Note:

- In any rigour, this ray must be higher than "the zero modal one". I.e. higher than the value below which one considers than an eigenvalue is worthless or than two eigenvalues are confused if one speaks about their variation.
- Contrary to the values of *FREQ*, the units are not here in frequency, but in pulsation if one makes the analogy with the problem *DYNAMICS*. As for one *TYPE_MODE='MODE_FLAMB'*, the unit of the characteristics of contour does not undergo any transformation. That makes it possible method *APM* to treat all the types of problems.
- The table produced with the option *TYPE_MODE='MODE_COMPLEXE'* contains, besides the number of eigenvalues (*NB_MODE*), three real allowing to describe the search criteria for method *APM*. The two first are *CENTRE_R* and *CENTRE_I*, parts real and imaginary of the complex number which corresponds to the center of the disc of research. The third parameter, *RAY*, is the ray (real strictly positif > 1.D-2) of the disc.

3.7 Keyword factor COUNTING

Once defined the type of control field (segment or disc), it remains to fix the method of enumeration and its parameters.

```
◇ METHOD = / 'CAR', [DEFECT]
           / 'STURM',
           / 'APM' .
```

If one deals with a problem of the type 'DYNAMICS' or 'MODE_FLAMB', only methods 'STURM' and 'CAR' are authorized.

In the same way for 'MODE_COMPLEXE' and 'APM' / 'CAR'.

In the case 'GENERAL', the three methods are available and the value 'CAR' takes all its direction then! It chooses for the user enters *STURM* and *APM* according to the characteristics of the modal problem:

- Either it is about a standard GEP (two real symmetrical matrices): method of *STURM*.
- In all other cases (atypical GEP with nonsymmetrical and/or complex matrix or QEP): method *APM*.

If one leaves the mode 'CAR' parameterized by default, the operator automatically chooses the method according to the type of selected study. One left this value knowingly, even in the cases which comprise only one alternative, to facilitate ergonomics.

If one is mistaken in *METHOD*, an alarm appears and the operator chooses the most adapted method automatically or stops in *ERREUR_FATALE*, according to the cases.

Note:

- One can use the method *APM* to count the modes of a standard GEP while posing *TYPE_MODE='MODE_COMPLEXE'*. That makes it possible to compare and validate the two methods of counting (cf *SDLL123a CAS-test*).

```
# If TYPE_MODE= 'DYNAMIQUE'

◇ SEUIL_FREQ = / f_seuil [R]
                / 0.01 [DEFECT]
◇ PREC_SHIFT = / p_shift [R]
                / 0.05 [DEFECT]
◇ NMAX_ITER_SHIFT = / n_shift [I]
                    / 3 [DEFECT]
```

```
# If TYPE_MODE= 'MODE_FLAMB' or 'GENERAL' (if MATR_C is not well informed and if the
two other matrices are symmetrical real)
```

◇ SEUIL_CHAR_CRIT =	/	c_seuil	[R]
	/	0.01	[DEFECT]
◇ PREC_SHIFT =	/	p_shift	[R]
	/	0.05	[DEFECT]
◇ NMAX_ITER_SHIFT =	/	n_shift	[I]
	/	3	[DEFECT]

These parameters are used only by the method of STURM.

Parameters SEUIL_* allow to define “the zero modal one”, i.e. the value in on this side which it is considered that the eigenvalue is worthless. If one is in dynamics one transforms this value into pulsation

$$omecor = (2\pi f_seuil)^2$$

while in buckling one keeps it such as it is

$$omecor = c_seuil.$$

Other parameters, PREC_SHIFT and NMAX_ITER_SHIFT, are related to the algorithm of shift of the terminals of the interval (cf [R5.01.04] §3.2) when one realizes that those are very close to an eigenvalue. Coarsely these terminals f_{min} (or λ_{min} in buckling) or f_{max} (resp. λ_{max}) are then shifted towards the outside of the segment of p_shift%. If the dynamic matrix thus rebuilt is always considered to be “numerically singular”, one again Re-shifts after having emitted one ALARM. This shift is tried n_shift time.

$$\lambda_{min}^- = \lambda_{min} - \max(omecor, 2^{(i-1)} \times p_shift \times |(\lambda_{min})|) \quad (I \text{ ème attempt})$$
$$\lambda_{max}^- = \lambda_{max} + \max(omecor, 2^{(i-1)} \times p_shift \times |(\lambda_{max})|) \quad (I \text{ ème attempt})$$

In fact, in dynamics as in buckling the shift takes place same manner. *Stricto-censu*, in dynamics these are thus not the frequencies that one shifts, but pulsations.

Another precision, the shift is in fact, by preoccupations with an effectiveness, dichotomic: p_shift% the first time, 2xp_shift% second time etc This process must make it possible quickly to move away from the “zone of singularity” to lower costs. *A contrario*, one should not too much increase the values of these parameters, because through shifts, the resulting terminals can prove to be very different from the initial terminals.

Moreover, to remain coherent with “the zero modal one” (noted here omechor):

- One does not shift a value lower than this minimum (from where the max in the formulas above),
- So from the beginning, the measured terminal is lower than this “zero” $|\lambda_*| < omechor$ (in absolute value) one fixes it at more or less this value (according to whether this terminal is positive or negative). One then allows any more no shift.

Note:

- A terminal of the interval σ is close to an eigenvalue, when factorization LDLT of the dynamic matrix associated on this terminal (for example that of a GEP is written $\mathbf{Q}(\sigma) := \mathbf{A} - \sigma \mathbf{B}$), led to a loss of decimal of more than NPREC digits (value parameterized under the keyword SOLVEUR). While exploiting the value of this parameter (NPREC=7, 8 or 9), one can then avoid the expensive refactorisations which these shifts imply when this digital singularity is not very marked.
- In the same way, while exploiting the digital parameters of the linear solveurs (for example: METHOD, RENUM, PRETREATMENTS...), one can also influence this criterion of singularity.

If TYPE_MODE= 'MODE_COMPLEXE' or 'GENERAL' (if MATR_C ou/et is indicated if one of the two other matrices is nonsymmetrical or complex)

◇ NBPOINT_CONTOUR =	/	nb_point	[I]
	/	40	[DEFECT]

◇ NMAX_ITER_CONTOUR = / n_contour [I]
/ 3 [DEFECT]

These parameters are used only by the method APM .

The two parameters precedent, NB_POINT_CONTOUR and NMAX_ITER_CONTOUR, are used only with the method APM. They are crucial for its robustness and they dimension its time thatlcul (a calculation with nb_point=80 will last longer than a calculation with nb_point=40).

The value nb_point give the number of point of discretization which is positioned along contour. This discretization must be enough fine "to capture" all spectral information. Ideally, this value must be fixed at least six times the number of modes which one discounts to find in the circle. If one has no idea *a priori* this number, it should be fixed at a value not too low: for example, 40 or 60.

Note:

- Attention, if this figure is too large (for example 1000), according to the size of the problem, as it implies as many factorizations LDLT, calculation can be very long!

Established algorithm APM will converge only when 3 successive evaluations amongst eigenvalues provide the same result. The differences between these three evaluations lies only in the degree of discretization of contour. One starts by discretizing with $k_- = \text{nb_point}/2$, then with $k = \text{nb_point}$ and finally with $k_+ = 2 \text{nb_point}$. If these three levels of discretization of contour produce an estimate amongst identical eigenvalues, the algorithm is regarded as converged. Its result is then this integer.

If not, one doubles these three levels of discretizations according to the following permutation

$$k_- \leftarrow k, \quad k \leftarrow k_+, \quad \text{and} \quad k_+ \leftarrow 2k_+$$

and one revalues the numbers of eigenvalues with these three levels of discretization⁶. If they are identical, convergence is reached, if not one continues. One reiterates this dichotomic heuristics to the maximum n_contour time.

If this process did not converge or if its result is incoherent (strictly negative entirety), one stops in ERREUR_FATALE.

3.8 Operand INFORMATION

◇ INFORMATION = / 1 [DEFECT]
/ 2

Indicate the level of impression in the file MESSAGE.

- 1: Impression of the result (and the principal stages of the algorithm if APM).
- 2: Rather more detailed impression for developer.

3.9 Operand TITLE

The title which will be given to the produced table.

3.10 Keyword factor SOLVEUR

◇ SOLVEUR=_F(),

One has access to all the parameters of the direct linear solveurs (METHODE='LDLT'/'MULT_FRONT'/'MUMPS') except those explicitly related to the final stage of descent-increase. This restriction relate to only the two parameters following of the solvor MUMPS: POSTTRAITEMENTS and RESI_REL.

⁶ Of course to save time calculation, one revalues only for the new discretization, finest.

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 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

In parallel, the parameter setting particularly is advised ⁷ METHODE=' MUMPS ' and RENUM=' QAMD '.

For more details on the solveurs, one will be able to consult the document [U4.50.01].

3.11 Operand NIVEAU_PARALLELISME

◇ NIVEAU_PARALLELISME = / 'COMPLETE' [DEFECT]
/ 'PARTIAL'

When with problems are dealt **mean sizes or large** (> 0.5M degrees of freedom) and/or that one seeks one **good part of their spectra** (> 50 modes), the recourse to parallelism gets appreciable profits in time and memory. And this, with a functional behavior and precise details of unchanged results compared to the sequential mode.

Natural calculation division INFO_MODE in nb_sbande independent (number of sous-bandes=nb_freq-1 or nb_mode_flamb-1), a first very effective level of parallelism gets (in time). It does not impact consumption memory⁸.

In addition, like the essence of the cost calculation is due to the digital phase of factorization⁹ of the linear solvor, if this stage is paralleled *via* MUMPS (METHODE=' MUMPS '), one adds notable profits in time and memory. It is a second level of parallelism activable.

With the value by default of this keyword (NIVEAU_PARALLELISME=' COMPLETE '), when calculation is parallel, the two levels of parallelism cumulate. As the second (that of the linear solvor) is less effective than the first, it starts only if the number of processors allows it (nb_procnb_sbande) and that if the selected linear solvor supports parallelism MPI (SOLVEUR=_F (METHODE=' MUMPS ')).

If one selects the other value of the keyword (NIVEAU_PARALLELISME=' PARTIEL '), one benefits only from the parallelism of the linear solvor (the second level). That can have interest to test methods or to center the profits on the reduction of the peak report (after having tested the other arms of lever proposed in the keyword SOLVEUR).

For more information on the implementation of parallelism, one will refer to documentation [U2.08.06] or in the dedicated paragraph of documentation [U2.06.01].

⁷ In order to at least reduce the cost in time of the phase of analysis (sequential) of MUMPS. This parameter setting is done however to the detriment of consumption memory. But this overcost quickly proves compensated by the distribution of the data on the processors which parallelism implies.

⁸ What is the case for CALC_MODES with OPTION='BAND' cut out on several sub-bands paralleled.

⁹ Stage of the linear solvor who gains the most profit from parallelism.

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.

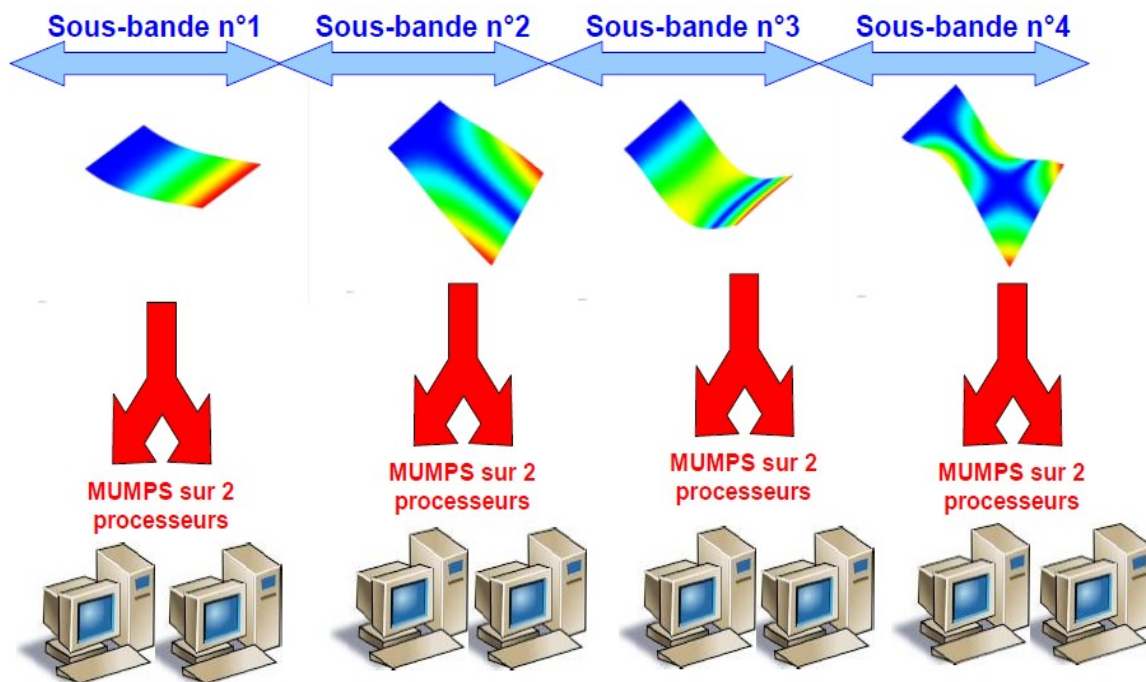


Figure 3.11-1. Example of distribution of calculations D 'INFO_MODE' on 8 processors with a cutting in 4 frequential sub-bands.

Thus, by activating only the first level of parallelism, a calculation cut out in nb_sbande and paralleled on $nb_proc=nb_sbande$ can gain in time at least a factor $nb_sbande/2$ (without overcost report and loss of precision).

If one distributes calculation on more processors (nb_procnb_sbande) by using solver MUMPS (adds the 2 thus of themNd level), the savings of time will be improved of a factor close to 2 for each $2^{*}nb_sbande$ additional processors¹⁰. And this, with profits in memory being able to go until a factor 2^{11} .

In order to preserve the parallel effectiveness of calculations, one advises:

- With 'COMPLETE', to select a number of processors which is a multiple amongst sub-bands. Typically 2.4 or 8. That limits the imbalance of load.
- With 'PARTIAL', to reserve at least 10^5 degrees of freedom per processor in order to feed linear solver MUMPS sufficiently.

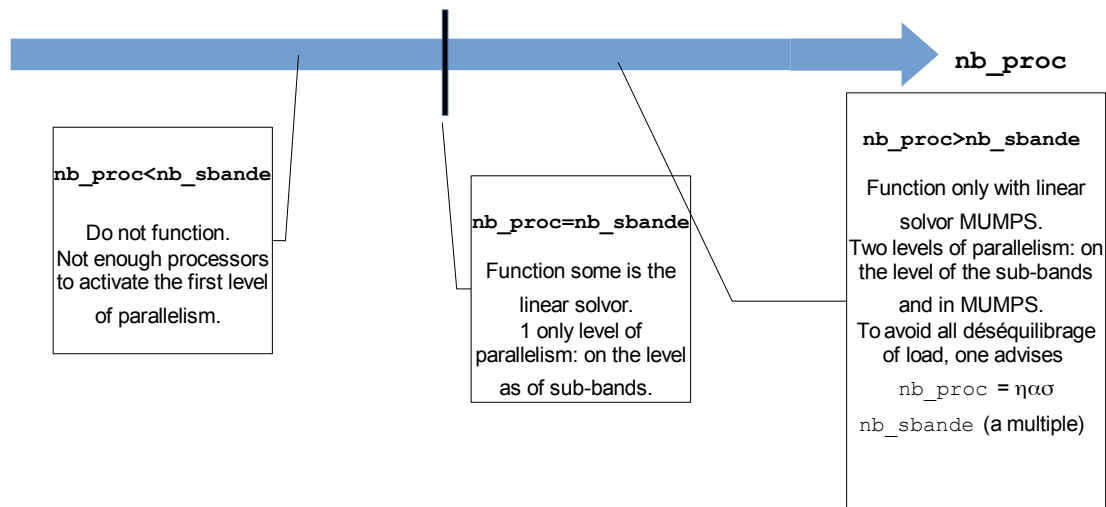
Functional rules are the following ones, while noting nb_proc the number of processors parameterized (mitre `option/mpi_nbcpu` of Astk) and nb_sbande the number of nonempty sub-bands:

- With `NIVEAU_PARALLELISME=' COMPLETE'` (**defect**): very large saving of time/improvement of the peak RAM report.

10 On sufficiently large problems ($N>0.5M$), the parallel effectiveness of the phase of factorization of MUMPS is often close to 0.5 (especially with a renumerotor frustrates standard 'QAMD'cf. §3.10).

11 In INFO_MODE, consumption memory of MUMPS is particularly optimized: operation in IN-CORE do not consume more memory than itOUT-OF-CORE (profit x3). This behavior is adapted for this operator. The profits which can get to him parallelism are thus, in proportion, less important.

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.



Figure

3.11-2. Perimeter of use with `NIVEAU_PARALLELISME='COMPLET'`.

- With `NIVEAU_PARALLELISME='PARTIEL'` : profit moderated in time/important profit on the peak RAM report.

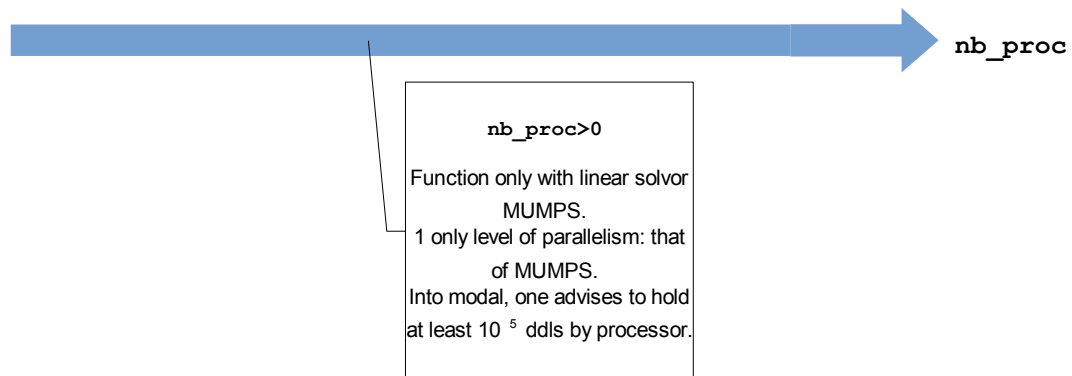


Figure 3.11-3. Perimeter of use with `NIVEAU_PARALLELISME='PARTIEL'`.

4 Examples

4.1 N°1 example

Comparison of the two methods **STURM** and **APM** on the standard GEP of **SDLD02a** (for **STURM** the terminals are mentioned in frequency). In the first calculation, one to seek to count the number of modes contained in the frequential band $[0; 5]$ with the usual method: **STURM**. In the second, one makes the same thing with method **APM**, but in the disc centered in the beginning ($centre=0+0j$) and of ray $rayon=(2\pi.5)^2$ (because there is no change of unit with **APM**). The two results are displayed in the file **MESSAGE**.

```
f1=5.0
nbmod01 = INFO_MODE (MATR_RIGI=MATASSR, MATR_MASS=MATASSM, TYPE_MODE='DYNAMICS',
                    FREQ= (0. , f1), COMPTAGE=_F (METHODE='STURM'),)
w1= (2*pi*f1) ** 2
nbmod11 = INFO_MODE (MATR_RIGI=MATASSR, MATR_MASS=MATASSM, TYPE_MODE='MODE_COMPLEXE',
                    TYPE_CONTOUR='CERCLE', CENTRE_CONTOUR=0.0+0.0j,
                    RAYON_CONTOUR=w1, COMPTAGE=_F (METHODE='APM'),)
```

With **INFO=1**, that causes postings following in the file **MESSAGE**:

```
-----
CHECKING OF THE FREQUENCY SPECTRUM (METHOD OF STURM)
PAS DE FREQUENCE IN the BAND (0.000E+00, 5.000E+00)
-----
.....

(METHOD APM) FOR THE 3 FOLLOWING LEVELS OF DISCRETIZATION
--- 20 --- 40 --- 80 ---
MANY EIGENVALUES DETECTEES
--- 0 --- 0 --- 0 ---
(METHOD APM) CONVERGENCE OF THE HEURISTICS
-----
CHECKING OF THE FREQUENCY SPECTRUM (METHOD OF THE PRINCIPAL ARGUMENT)
PAS DE FREQUENCE IN the DISC CENTERS IN (0.000E+00, 0.000E+00)
AND OF RAY 9.870E+02
```

Here, the method of Sturm required only two factorizations. Method **APM** has Convergé immediately with the first iteration. But this one required $20 + 40 + 80 = 140$ factorizations. The enumeration of eigenvalues in the complex plan has a price (which one does not manage for the moment to reduce)!

number of the Eigen frequencies (0 in this case) as well as the search criteria are saved in one table. Impression, by **IMPR_TABLE**, concepts **NBMOD01** and **NBMOD11** products by **INFO_MODE** in the preceding example shows the following composition:

```
nbmod01
FREQ_MIN      FREQ_MAX      NB_MODE
0.000000E+00  5.000000E+00      0
...
nbmod11
CENTRE_R      CENTRE_I      RAY      NB_MODE
0.000000E+00  0.000000E+00  9.86960E+02  0
```

When **INFO_MODE** is called with the option **TYPE_MODE='MODE_FLAMB'**, the produced tables contain three columns: **NB_MODE** (the number of eigenvalues) like **CHAR_CRIT_MIN** and **CHAR_CRIT_MAX**, search criteria for the dynamic problems with linear buckling.

4.2 N°2 example

Enumeration concerning the QEP of SDLL123a. This time, only method APM proves to be licit. One counts the number of modes contained in the circle centered in the beginning ($centre=0+0j$) and of ray $rayon=124 \times 2\pi$.

```
f1=124.*2.*pi
nbmod4=INFO_MODE (MATR_RIGI=RIGIDITE,          MATR_MASS=MASSE,          MATR_C=GYOM,
TYPE_MODE='MODE_COMPLEXE',
                TYPE_CONTOUR=' CERCLE', CENTRE_CONTOUR=0.0+0.0j,
                RAYON_CONTOUR=f1, COMPTAGE=_F (METHODE='APM',),)
```

In INFO=1 that causes postings following in the file MESSAGE:

```
(METHOD APM) FOR THE 3 LEVELS OF FOLLOWING DISCRETIZATION
---  20 ---  40 ---  80 ---
MANY EIGENVALUES DETECTEES
---  4 ---  4 ---  4 ---
(METHOD APM) CONVERGENCE OF THE HEURISTICS
-----
CHECKING OF THE FREQUENCY SPECTRUM (METHOD OF THE PRINCIPAL ARGUMENT)
THE NUMBER OF FREQUENCIES IN the DISC CENTERS IN (0.000E+00, 0.000E+00)
AND OF RAY 7.791E+02 EAST 4
```

And impression of the produced table (nbmod4) by IMPR_TABLE give:

CENTRE_R	CENTRE_I	RAY	NB_MODE
0.00000E+00	0.00000E+00	7.79115E+02	4