

Opérateur `MODE_ITER_SIMULT`

1 But

Que cela soit pour étudier les vibrations d'une structure (éventuellement amortie ou tournante) ou rechercher ses modes de flambement, le mécanicien doit souvent résoudre un problème modal: soit **généralisé (GEP)** [R5.01.01], soit **quadratique (QEP)** [R5.01.02]. Pour ce faire, *Code_Aster* propose deux opérateurs: `MODE_ITER_SIMULT` et `MODE_ITER_INV`.

Le **premier**, qui nous occupe dans cette note, **est plutôt à utiliser lorsqu'on cherche une partie significative du spectre** (méthodes de sous-espace ou méthode globale). Le second est à privilégier lorsqu'on s'intéresse à seulement quelques modes propres (typiquement une demi-douzaine) ou lorsqu'on souhaite affiner quelques estimations (éventuellement provenant de `MODE_ITER_SIMULT`).

`MODE_ITER_SIMULT` détermine un ensemble de modes propres, soit par une méthode de type sous-espace (Bathe & Wilson, Lanczos ou Sorensen) ou une méthode globale de type QR (QZ pour les petits problèmes). Ses quatre méthodes sont accessibles pour traiter un **GEP symétrique réel**: calcul dynamique classique (sans amortissement et sans effet gyroscopique) ou un problème de flambement d'Euler.

En **QEP**, Bathe & Wilson est proscrit. **En GEP ou en QEP**, dès qu'une matrice est **réelle non symétrique** ou que **K** est **complexe symétrique** (pour prendre en compte de l'amortissement hystérétique), seuls Sorensen ou QZ restent disponibles.

Cet opérateur produit un **concept** `mode_meca_*` (cas dynamique) ou `mode_flamb` (cas flambement d'Euler, seulement en QEP) suivant la valeur renseignée dans le mot-clé `TYPE_RESU`. Ce document décrit les paramètres accessibles de chacune des méthodes propres à l'opérateur `MODE_ITER_SIMULT`.

Table des Matières

1 But.....	1
2 Syntaxe.....	3
3 Opérandes.....	6
3.1 Principes	6
3.2 Opérandes <code>MATR_A</code> , <code>B</code> , <code>C</code>	8
3.3 Mot clé <code>TYPE_RESU</code>	9
3.4 Mot clé <code>METHODE</code>	9
3.5 Mot clé <code>CALC_FREQ</code>	10
3.5.1 Opérande <code>OPTION</code>	10
3.5.2 Opérande <code>APPROCHE</code>	10
3.5.3 Opérande <code>FREQ</code>	10
3.5.4 Opérande <code>AMOR_REDUIT</code>	10
3.5.5 Opérande <code>CHAR_CRIT</code>	11
3.5.6 Opérande <code>NMAX_FREQ</code>	11
3.5.7 Opérande <code>DIM_SOUS_ESPACE</code>	11
3.5.8 Opérandes d'IRAM (si <code>METHODE='SORENSEN'</code>).....	12
3.5.9 Opérandes de la méthode de Lanczos (si <code>METHODE='TRI_DIAG'</code>).....	12
3.5.10 Opérandes de la méthode de Bathe & Wilson (si <code>METHODE='JACOBI'</code>).....	12
3.5.11 Opérandes de la méthode QZ (si <code>METHODE='QZ'</code>).....	13
3.6 Opérandes <code>SEUIL_FREQ</code> , <code>PREC_SHIFT</code> , <code>NMAX_ITER_SHIFT</code> et <code>NPREC_SOLVEUR</code>	13
3.7 Mot clé <code>VERI_MODE</code>	13
3.7.1 Opérande <code>PREC_SHIFT</code>	13
3.7.2 Opérande <code>STOP_ERREUR</code>	14
3.7.3 Opérande <code>SEUIL</code>	14
3.7.4 Opérande <code>STURM</code>	14
3.7.5 Opérande <code>PREC_SHIFT</code>	14
3.8 Opérandes <code>SENSIBLITE</code>	14
3.9 Opérande <code>STOP_FREQ_VIDE</code>	14
3.10 Opérande <code>INFO</code>	15
3.11 Opérande <code>TITRE</code>	15
4 Phase de vérification.....	16
5 Phase d'exécution.....	17
5.1 Vérification.....	17
5.2 Actions par défaut.....	17
6 Paramètres modaux/ Norme des modes/ Position modale.....	18
7 Impression des résultats	19
8 Tri de modes / Caractérisation de mode_meca_*.....	20
9 Exemples.....	21
9.1 Calcul des 5 modes propres les plus proches d'une fréquence donnée (100 Hz).....	21
9.2 Calcul des charges critiques contenues dans une bande.....	21

2 Syntaxe

```
mode_ [*]=MODE_ITER_SIMULT
```

#DONNEES DU PROBLEME MODAL

```
( ♦ MATR_A=A / [matr_asse_DEPL_R]
/ [matr_asse_DEPL_C]
/ [matr_asse_PRES_R]
/ [matr_asse_GENE_R]
/ [matr_asse_GENE_C]
♦ MATR_B=B / [matr_asse_DEPL_R]
/ [matr_asse_PRES_R]
/ [matr_asse_GENE_R]
♦ MATR_C=C (uniquement en QEP) / [matr_asse_DEPL_R]
/ [matr_asse_GENE_R]
```

#TYPE DE PROBLEME

```
♦ TYPE_RESU=/'DYNAMIQUE' [DEFAULT]
/'MODE_FLAMB'
```

#CHOIX DE LA METHODE

```
♦ METHODE=/'SORENSEN' [DEFAULT]
/'TRI_DIAG' (uniquement GEP/QEP symétriques réels)
/'JACOBI' (sauf en QEP)
/'QZ' (problème de petites tailles<103ddls)
```

#TYPE DE CALCUL MODAL

```
♦ CALC_FREQ=_F ( ♦ OPTION=/'CENTRE'
/'BANDE'
/'PLUS_PETITE' [DEFAULT]
/'TOUT' (uniquement avec QZ)
```

#CARACTERISTIQUES DU CALCUL

```
# Si TYPE_RESU='DYNAMIQUE'
♦ APPROCHE=/'REEL' [DEFAULT]
../'IMAG'
/'COMPLEXE' (uniquement Sorensen)
# Si OPTION='PLUS_PETITE'
♦ NMAX_FREQ=/10 [DEFAULT]
/nf [I]
# Si OPTION='CENTRE'
♦ FREQ=1_f [1_R]
♦ AMOR_REDUIT=1_a [1_R]
♦ NMAX_FREQ=/10 [DEFAULT]
/nf [I]
# Si OPTION='BANDE'
♦ FREQ=1_f [1_R]
# Si TYPE_RESU='MODE_FLAMB'
♦ APPROCHE=/'REEL' [DEFAULT]
/'IMAG'
# Si OPTION='PLUS_PETITE'
♦ NMAX_FREQ=/10 [DEFAULT]
/nf [I]
# Si OPTION='CENTRE'
♦ CHAR_CRIT=1_c [1_R]
♦ NMAX_FREQ=/10 [DEFAULT]
/nf [I]
# Si OPTION='BANDE' (uniquement GEP à modes réels)
♦ CHAR_CRIT=1_c [1_R]
```

#CARACTERISTIQUES DE L'ESPACE DE PROJECTION

```
        ◇ DIM_SOUS_ESPACE=dse                [I]
        ◇ COEF_DIM_ESPACE=mse                [I]
EXCLUS('DIM_SOUS_ESPACE', 'COEF_DIM_ESPACE')
```

#POUR PRE ET POST-TRAITEMENTS

```
        ◇ PREC_SHIFT=/0.05                    [DEFAULT]
          /ps                                [R]
        ◇ NMAX_ITER_SHIFT=/5                  [DEFAULT]
          /ns                                [I]
        ◇ NPREC_SOLVEUR=/8                   [DEFAULT]
          /ndeci                             [R]
        ◇ SEUIL_FREQ=/1.E-2                  [DEFAULT]
          /sf                                [R]
```

#PARAMETRAGE INTERNE DES METHODES

```
#      Si METHODE='SORENSEN'
        ◇ PREC_SOREN=/0                      [DEFAULT]
          /pso                               [R]
        ◇ NMAX_ITER_SOREN=/20                [DEFAULT]
          /nso                               [I]
        ◇ PARA_ORTHO_SOREN=/0.717            [DEFAULT]
          /porso                             [I]

#      Si METHODE='TRI_DIAG'
        ◇ PREC_ORTHO=/1.E-12                 [DEFAULT]
          /po                                [R]
        ◇ NMAX_ITER_ORTHO=/5                 [DEFAULT]
          /nio                               [I]
        ◇ PREC_LANCZOS=/1.E-8               [DEFAULT]
          /pl                                [R]
        ◇ NMAX_ITER_QR=/30                  [DEFAULT]
          /nim                               [I]
        ◇ OPTION='SANS'                     [DEFAULT]
          /'MODE_RIGIDE'

#      Si METHODE='JACOBI'
        ◇ PREC_BATHE=/1.E-10                [DEFAULT]
          /pbat                              [R]
        ◇ NMAX_ITER_BATHE=/40               [DEFAULT]
          /nbat                              [I]
        ◇ PREC_JACOBI=/1.E-2               [DEFAULT]
          /pjaco                             [R]
        ◇ NMAX_ITER_JACOBI=/12              [DEFAULT]
          /njaco                             [I]
      )

#      Si METHODE='QZ'
        ◇ TYPE_QZ='QZ_SIMPLE'                [DEFAULT]
          ~.'QZ_EQUI'
          /'QZ_QR' (si GEP à matrices SPD)
```

#POUR VERIFICATIONS FINALES

```
        ◇ VERI_MODE=_F(
          ◇ STOP_ERREUR='OUI'                [DEFAULT]
            /'NON'
          ◇ SEUIL= /1.E-6                    [DEFAULT]
            /r                                [R]
          ◇ PREC_SHIFT=/0.05                 [DEFAULT]
            /prs                              [R]
          ◇ STURM='OUI'                     [DEFAULT]
            /'NON'
        )
```

#SENSIBILITE

◇ SENSIBILITE= (voir [U4.50.02])

#DIVERS

◇ STOP_FREQ_VIDE=/'OUI' [DEFAULT]
/ 'NON'

◇ INFO=/1 [DEFAULT]
/2 [I]

◇ TITRE=ti

);

#RESULTATS DU PROBLEME MODAL

Si MATR_C=[matr_asse_DEPL_R]

Si TYPE_RESU='MODE_FLAMB'

Si MATR_A=[matr_asse_DEPL_C]

Si MATR_A=[matr_asse_DEPL_R]

Si MATR_A=[matr_asse_PRES_R]

Si MATR_A=[matr_asse_GENE_R]

Si MATR_A=[matr_asse_GENE_C]

alors [*]->mode_meca_c

alors [*]->mode_flamb

alors [*]->meca_c

alors [*]->mode_meca

alors [*]->mode_acou

alors [*]->mode_gene

alors [*]->mode_gene

3 Opérands

3.1 Principes

Cet opérateur résout le **problème généralisé (GEP)** aux valeurs propres suivant[R5.01.01]:

Trouver (λ, x) tels que $Ax = \lambda Bx$, $x \neq 0$, où A et B sont des matrices réelles, symétriques ou non. Pour modéliser un amortissement hystérétique dans l'étude des vibrations libres d'une structure, la matrice A peut être complexe symétrique[U2.06.03][R5.05.04].

Ce type de problème correspond, en mécanique, notamment à:

- **L'étude des vibrations libres d'une structure** non amortie et non tournante. Pour cette structure, on recherche les plus petites valeurs propres ou bien celles qui sont dans un intervalle donné pour savoir si une force excitatrice peut créer une résonance. Dans ce cas, la matrice A est la matrice de rigidité matérielle, notée K , symétrique réelle (éventuellement augmentée de la matrice de rigidité géométrique notée K_g , si la structure est précontrainte), et B est la matrice de masse ou d'inertie notée M (symétrique réelle). Les valeurs propres obtenues sont les carrés des pulsations associées aux fréquences cherchées. Le système à résoudre peut s'écrire

$$\underbrace{(K + K_g)}_A x = \lambda \underbrace{M}_B x$$

où $\lambda = (2\pi f)^2$ est le carré de la pulsation ω , f la fréquence propre et x le vecteur de déplacement propre associé. Les modes propres manipulés (I, x) sont à valeurs réelles. Ce type de problématique est activé par le mot-clé `TYPE_RESU='DYNAMIQUE'` et génère une structure de données Aster de type `mode_meca`, `mode_acou` ou `mode_gene` (suivant le type des données d'entrée).

- **La recherche de mode de flambement linéaire.** Dans le cadre de la théorie linéarisée, en supposant *a priori* que les phénomènes de stabilité sont convenablement décrits par le système d'équations obtenu en supposant la dépendance linéaire du déplacement par rapport au niveau de charge critique, la recherche du mode de flambement x associé à ce niveau de charge critique $\mu = -\lambda$, se ramène à un problème généralisé aux valeurs propres de la forme

$$\underbrace{(K + \mu K_g)}_A x = 0 \Leftrightarrow \underbrace{K}_A x = \lambda \underbrace{K_g}_B x$$

avec K matrice de rigidité matérielle et K_g matrice de rigidité géométrique. Les modes propres manipulés (I, x) sont à valeurs réelles. Ce type de problématique est activé par le mot-clé `TYPE_RESU='FLAMBEMENT'` et génère une structure de données Aster de type `mode_flamb`.

Attention:

- Dans le code, on ne traite que les valeurs propres du problème généralisé, les λ . Pour obtenir les véritables charges critiques, les μ , il faut les multiplier par -1 .
- En GEP, pour traiter des problèmes à modes complexes (matrices non symétriques et/ou à valeurs complexes), il faut utiliser `MODE_ITER_SIMULT (METHODE='SORENSEN'/'QZ')`.

Cet opérateur permet aussi l'étude de la **stabilité dynamique d'une structure en présence d'amortissements et/ou d'effets gyroscopiques**. Cela conduit à la résolution d'un problème modal d'ordre plus élevé, dit quadratique (QEP)[R5.01.02]. On recherche alors des valeurs et vecteurs propres complexes (I, x) .

- Le problème consiste à trouver $(\lambda, x) \in (C, C^N)$ tels que
- $$(\lambda^2 B + \lambda C + A)x = 0$$

où typiquement, en mécanique linéaire, A sera la matrice de rigidité, B la matrice de masse et C la matrice d'amortissement. Les matrices A , B et C sont des matrices symétriques et réelles. La valeur propre complexe λ est reliée à la fréquence propre f et à l'amortissement réduit ξ par $\lambda = \xi(2\pi f) \pm i(2\pi f)\sqrt{1-\xi^2}$. Ce type de problématique est activé par le mot-clé `TYPE_RESU='DYNAMIQUE'` et génère une structure de données Aster de type `mode_meca_c`.

Attention:

- En QEP, pour traiter des problèmes à matrices non symétriques et/ou à valeurs complexes, il faut utiliser `MODE_ITER_SIMULT (METHODE='SORENSEN'/'QZ')`.
- Le flambement (`TYPE_RESU='FLAMBEMENT'`) n'est pas licite en QEP.
- Le test de Sturm n'est opérant qu'en GEP à matrices symétriques réelles. En dehors de ce cadre (QEP, GEP à matrices réelles non symétriques ou à matrice A complexe symétrique), l'option 'BANDE' est proscrite et la post-vérification basée sur Sturm n'est pas activée (paramètre `VERI_MODE/STURM` inopérant).

Pour résoudre ces problèmes modaux généralisés ou quadratiques, Code_Aster propose différentes approches. Au delà de leurs spécificités numériques et fonctionnelles qui sont reprises dans les documents [R5.01.01/02], on peut les synthétiser sous la forme du tableau ci-dessous (**les valeurs par défaut sont matérialisées en gras**).

Opérateur/ Périmètre d'application	Algorithme	Mot-clé	Avantages	Inconvénients
MODE_ITER_INV				Uniquement symétrique réel (GEP et QEP).
<i>1^{ère} phase (heuristique)</i>				
Calcul de quelques modes	Bissection (sans objet en QEP).	'SEPRE'		
Calcul de quelques modes	Bissection+ Sécante(GEP) ou Müller-Traub (QEP).	'AJUSTE'	Meilleure précision	Coût calcul
Amélioration de quelques estimations	Initialisation par l'utilisateur	'PROCHE'	Reprise de valeurs propres estimées par un autre processus. Coût calcul de cette phase quasi-nul	Pas de capture de multiplicité
<i>2^{ième} phase (méthode des puissances inverses)</i>				Uniquement symétrique réel (GEP et QEP)
Méthode de base	Puissances inverses	'DIRECT'	Très bonne construction de vecteurs propres	Peu robuste
Option d'accélération	Quotient de Rayleigh (sans objet en QEP)	'RAYLEIGH'	Améliore la convergence	Coût calcul
MODE_ITER_SIMULT				
Calcul d'une partie du spectre	Bathe & Wilson	'JACOBI'		Peu robuste Uniquement symétrique réel (GEP)
	Lanczos	'TRI_DIAG'	Détection spécifique	Uniquement

Opérateur/ Périmètre d'application	Algorithme	Mot-clé	Avantages	Inconvénients
	(Newman-Pipano en GEP et Jennings en QEP)		des modes rigides.	<i>symétrique réel (GEP et QEP)</i>
	IRAM (Sorensen)	'SORENSEN'	Robustesse accrue. Meilleures complexités calcul et mémoire. Contrôle de la qualité des modes.	Méthode par défaut. <i>Portée en non symétrique et avec A complexe symétrique.</i>
Calcul de tout le spectre puis filtrage d'une partie.	QZ	'QZ'	Méthode de référence en terme de robustesse.	Très coûteuse en CPU et en mémoire. A réserver aux petits cas ($<10^3$ ddls). <i>Portée en non symétrique et avec A complexe symétrique.</i>

Tableau 3.1-1. Récapitulatif des méthodes modales de Code_Aster

Lorsqu'il s'agit de déterminer quelques valeurs propres simples bien discriminées ou d'affiner quelques estimations, l'opérateur `MODE_ITER_INV` (heuristique + puissance inverse), est souvent bien indiqué. Par contre, pour capturer une partie significatif du spectre, on a recourt à `MODE_ITER_SIMULT`, via les méthodes de sous-espace (Lanczos, IRAM, Jacobi) ou la méthode globale QZ (méthode très robuste mais coûteuse; à réserver aux petits cas).

C'est la seconde classe de méthode qui va nous intéresser ici.

Pour les méthodes de sous-espace, elle consiste à projeter le problème sur un espace dont la taille est supérieure au nombre de valeurs propres souhaitées mais très inférieure à celle du problème. On s'arrange pour que ce problème ait un spectre très proche de celle du problème initial et qu'il prenne une forme canonique (tridiagonale, Hessenberg etc.). Puis on applique un solveur modal global (Jacobi pour Bathe & Wilson, QR pour Lanczos/IRAM) sur ce problème simplifié. Enfin on convertit les modes obtenus dans l'espace de travail initial.

Quant à la méthode globale QZ, elle résoud directement et entièrement le problème initial (GEP ou QEP linéarisé) pour améliorer la robustesse du processus. Elle présente toutefois l'inconvénient de calcul tout le spectre. Elle est donc à réserver aux petits cas ($<10^3$ ddls).

Il est d'ailleurs tout à fait recommandé de profiter des points forts des deux classes de méthode en affinant les vecteurs propres obtenus par `MODE_ITER_SIMULT`, via `MODE_ITER_INV` (OPTION='PROCHE'). Cela permettra de réduire la norme du résidu final (cf. §3.6.2).

Remarque:

On conseille fortement une lecture préalable des documentations de référence [R5.01.01] [R5.01.02]. Elle donne à l'utilisateur les propriétés et les limitations, théoriques et pratiques, des méthodes modales abordées tout en reliant ces considérations, qui peuvent parfois paraître un peu éthérées, à un paramétrage précis des options.

3.2 Opérandes `MATR_A`, `_B`, `_C`

- ♦ `MATR_A=A`
Matrice assemblée, réelle (symétrique ou non) ou complexe symétrique, de type `[matr_asse*_R/C]` du GEP/QEP à résoudre.

- ♦ `MATR_B=B`
Matrice assemblée, réelle (symétrique ou non), de type `[matr_asse*_R]` du GEP/QEP à résoudre.
- ◇ `MATR_C=C`
Matrice assemblée, réelle (symétrique ou non), de type `[matr_asse*_R]` du QEP à résoudre.

Remarque:

Si `MATR_A` est complexe symétrique ou si une des `MATR_A/B` ou `C` est non symétrique réelles, seule les méthodes de Sorensen et QZ sont licites. On ne peut alors utiliser l'option de calcul 'BANDE' et, pour le cas `MATR_A` complexe, une borne fréquentielle nulle ('OPTION=PLUS_PETITE' ou 'CENTRE' avec `f=0`).

3.3 Mot clé `TYPE_RESU`

- ◇ `TYPE_RESU=/ 'DYNAMIQUE' [DEFAULT]`
`/ 'MODE_FLAMB'`

Ce mot-clé permet de définir la nature du problème modal à traiter: recherche de fréquences de vibration (cas classique de dynamique avec ou sans amortissement et effets gyroscopiques) ou recherche de charges critiques (cas de la théorie du flambement linéaire, uniquement en GEP). Suivant cette classe d'appartenance, les résultats sont affichés et stockés différemment dans la structure de données:

- **En dynamique**, les fréquences sont ordonnées par ordre croissant du module de leur écart au shift (cf. [R5.01.01/02], §3.8/2.5). C'est la valeur de la variable d'accès `NUME_ORDRE` de la structure de donnée. L'autre variable d'accès, `NUME_MODE`, est égale à la véritable position modale dans la spectre de la valeur propre (déterminée par le test de Sturm cf. §3.6 [R5.01.01]). Ce test de Sturm n'est licite qu'en GEP à modes réels (matrices symétriques réelles), dans les autres cas de figures, GEP à modes complexes et QEP, on pose `NUME_MODE=NUME_ORDRE`.
- **En flambement**, les valeurs propres sont stockées par ordre croissant algébrique. Les variables `NUME_ORDRE` et `NUME_MODE` prennent la même valeur égale à cette ordre.

3.4 Mot clé `METHODE`

Quatre méthodes de résolution sont disponibles pour résoudre le problème aux valeurs propres (cf. tableau 3.1.1):

- ◇ `METHODE=/ 'SORENSEN' [DEFAULT]`
On utilise la méthode de Sorensen (package externe ARPACK) pour calculer les modes propres du GEP ou du QEP (cf. [R5.01.01/02] §7/4). Son périmètre englobe les matrices réelles, symétriques ou non, voire une matrice **A** complexe symétrique.
- ◇ `/ 'TRI_DIAG'`
On utilise la méthode de Lanczos (variante de Newmann-pipano en GEP, de Parlett & Saad en QEP) pour calculer les modes propres du GEP ou du QEP (cf. [R5.01.01/02] §6/4). Son périmètre est limité aux matrices symétriques réelles.
- ◇ `OPTION=/ 'MODE_RIGIDE' [DEFAULT]`
`/ 'SANS'`
Mot-clé utilisable seulement avec la méthode de Lanczos pour un GEP. Il permet de détecter et de calculer au préalable, par une méthode algébrique les modes de corps de rigide. Ils sont utilisés par la suite pour calculer les autres modes avec l'algorithme de Lanczos. Ils sont fournis à l'utilisateur seulement s'ils font partie des modes demandés. Si les modes de corps rigide sont calculés sans utiliser cette option, les valeurs propres calculées par l'algorithme de Lanczos ne sont pas nulles mais très voisines de zéro.
- ◇ `/ 'JACOBI'`

On utilise la méthode de Bathe & Wilson (puis la méthode de Jacobi sur le système projeté) pour calculer les modes propres du GEP (cf. [R5.01.01] §8). Son périmètre est limité aux matrices symétriques réelles.

◇ / 'QZ'

On utilise la méthode QZ de la bibliothèque externe LAPACK pour calculer les modes propres du GEP ou du QEP (cf. [R5.01.01/02] §9/5). Son périmètre englobe les matrices réelles, symétriques ou non, voire une matrice **A** complexe symétrique. Cette méthode de référence très coûteuse est à réserver aux problèmes de petites tailles (<10³ ddl).

3.5 Mot clé **CALC_FREQ**

◇ **CALC_FREQ**= **F** (...)

Mot-clé facteur pour la définition des paramètres de calcul des modes propres et de leur nombre.

3.5.1 Opérande **OPTION**

◇ **OPTION**=
'BANDE'

On recherche toutes les valeurs propres dans une bande donnée. Cette bande est définie par l'argument de **FREQ**=(f_1 , f_2) ou par celui de **CHAR_CRIT**= (λ_1 , λ_2). Option uniquement disponible en GEP à matrices réelles symétriques.

'CENTRE'

On recherche les **NMAX_FREQ** valeurs propres les plus proches de la fréquence f (argument du mot-clé **FREQ**= f) ou les plus proches de la charge critique λ (argument du mot-clé **CHAR_CRIT**= λ).

'PLUS_PETITE'

On recherche les **NMAX_FREQ** plus petites valeurs propres.

[DEFAULT]

'TOUT'

On cherche tous les modes associés à des ddl physiques. Option utilisable qu'avec la méthode QZ.

Voir [R5.01.01/02] §2.5/3.8.

3.5.2 Opérande **APPROCHE**

◇ **APPROCHE**= /**REEL**' [DEFAULT]
/ 'IMAG'
/ 'COMPLEXE' (uniquement avec Sorensen)

Ce mot-clé définit le type d'approche (réelle, imaginaire ou complexe) pour le choix du pseudo-produit scalaire du QEP utilisé avec les méthode de Lanczos ou avec celle de Sorensen (cf. [R5.01.02]).

Cet opérande n'a de sens que pour l'analyse des vibrations libres d'une structure amortie ou tournantes (modes propres complexes; le mot-clé **MATR_C** doit être renseigné). En flambement, cela n'a aucun intérêt.

Remarque:

En quadratique, avec la méthode de Lanczos seule l'approche 'IMAG' est compatible avec une borne fréquentielle nulle ('OPTION=PLUS_PETITE' ou 'CENTRE' avec $f=0$). Avec Sorensen, aucune n'est compatible.

3.5.3 Opérande **FREQ**

◇ **FREQ**=**l_f**

Liste des fréquences (ne peut être utilisé que si **TYPE_RESU**='DYNAMIQUE'): son utilisation dépend de l'OPTION choisie.

OPTION='BANDE'

On attend deux valeurs (f_1 f_2) qui définissent la bande de recherche

`OPTION='CENTRE'`

On attend une seule valeur de fréquence

Les valeurs stipulées sous ce mot-clé doivent être positives.

3.5.4 Opérande `AMOR_REDUIT`

◇ `AMOR_REDUIT=1_a`

Valeur de l'amortissement réduit qui permet de définir la valeur propre complexe (le «shift») autour de laquelle on cherche les valeurs propres les plus proches (cf. [R5.01.01] §5.4). Cette option ne peut être utilisée que dans le cadre d'un problème modal à modes complexes: QEP ou GEP à matrices réelles non symétriques ou avec **A** complexe symétrique.

`OPTION='CENTRE'`

On attend une seule valeur d'amortissement réduit

La valeur stipulée sous ce mot-clé doit être positive et être comprise entre 0 et 1. En flambement, cela n'a aucun intérêt.

3.5.5 Opérande `CHAR_CRIT`

◇ `CHAR_CRIT=1_c`

Liste des charges critiques (ne peut être utilisé que si `TYPE_RESU='MODE_FLAMB'`): son utilisation dépend de l'`OPTION` choisie.

`OPTION='BANDE'`

On attend deux valeurs (\square_1 \square_2) qui définissent la bande de recherche

`OPTION='CENTRE'`

On attend une seule valeur de charge critique

Les valeurs stipulées sous ce mot-clé sont positives ou négatives. En dynamique et en QEP cela n'a aucun intérêt.

3.5.6 Opérande `NMAX_FREQ`

◇ `NMAX_FREQ=nf (10) [DEFAULT]`

Nombre maximum de valeurs propres à calculer.

Ce mot-clé est ignoré avec l'option '`BANDE`' car on calcule alors toutes les valeurs propres contenues dans la bande stipulée.

Dans les deux cas, si `nf` est strictement supérieur au nombre de «ddl-actifs», `nactif` (cf. [R5.01.01] §3.2), alors on le force à prendre cette valeur plafond.

3.5.7 Opérande `DIM_SOUS_ESPACE`

◇ `DIM_SOUS_ESPACE=des`

◇ `COEF_DIM_ESPACE=mse`

`EXCLUS('DIM_SOUS_ESPACE','COEF_DIM_ESPACE')`

Si le mot-clé `DIM_SOUS_ESPACE` n'est pas renseigné ou est initialisé à une valeur strictement inférieure au nombre de fréquences demandées `nf`, l'opérateur calcule automatiquement une dimension admissible pour le sous-espace de projection (cf. §5.2 de ce document et [R5.01.01] §5.3) à l'aide `COEF_DIM_ESPACE`.

Grâce à la donnée de ce facteur multiplicatif, `mse`, on peut projeter sur un espace dont la taille est proportionnelle au nombre de fréquences contenues dans l'intervalle d'étude. Dans l'encapsulation de `MODE_ITER_SIMULT`, `MACRO_MODE_MECA` [U4.52.02], on peut donc optimiser la taille des sous-espaces qui reste proportionnelle au nombre de fréquences recherchées: les sous-espaces riches en valeurs propres ne pénalisent ainsi pas les plus pauvres (en terme de CPU).

On peut cependant fixer arbitrairement la taille de ce sous-espace, via la valeur `des` prise par le mot-clé `DIM_SOUS_ESPACE` (qui doit être supérieure à `nf` pour être prise en compte).

Dans les deux cas, si la taille du sous-espace de projection `ndim` est strictement supérieure au nombre de «ddl-actifs», `nactif` (cf. [R5.01.01] §3.2), alors on la force à prendre cette valeur plafond.

Remarques:

- Si on utilise la méthode de Sorensen (IRAM) et que $ndim - nf < 2$, des impératifs numérico-informatiques forcent à imposer $ndim = nf + 2$.
- En quadratique on travaille sur un problème réel de taille double: $2 * nf$, $2 * ndim$.
- Ces paramètres sont inutiles pour la méthode 'QZ'.

3.5.8 Opérandes d'IRAM (si `METHODE='SORENSEN'`)

◇ `PREC_SOREN=pso` (0.) [DEFAULT]

Remarque:

La méthode considère alors qu'elle doit travailler avec la plus petite précision possible, le «zéro machine». Pour en avoir un ordre de grandeur, en double précision sur les machines standards, cette valeur est proche de $2.22 \cdot 10^{-16}$.

◇ `NMAX_ITER_SOREN=nso` (20) [DEFAULT]
◇ `PARA_ORTHO_SOREN=porso` (0.717) [DEFAULT]

Il s'agit de paramètres d'ajustement de la précision requise sur les modes (par défaut, la précision machine est choisie), du nombre de redémarrages autorisé de la méthode de Sorensen (cf. [R5.01.01] §7) et du coefficient d'orthogonalisation de l'IGSM de Kahan-Parlett (cf. [R5.01.01] annexe 2).

Remarque:

Lors des premiers passages, il est fortement conseillé de ne pas modifier ces paramètres qui concernent plutôt les arcanes de l'algorithme et qui sont initialisés empiriquement à des valeurs standards.

3.5.9 Opérandes de la méthode de Lanczos (si `METHODE='TRI_DIAG'`)

◇ `PREC_ORTHO =po` (1.10^{-12}) [DEFAULT]
◇ `NMAX_ITER_ORTHO=nio` (5) [DEFAULT]
◇ `PREC_LANCZOS=pl` (1.10^{-8}) [DEFAULT]
◇ `NMAX_ITER_QR=nim` (30) [DEFAULT]

Les deux premiers paramètres permettent, respectivement, d'ajuster la précision d'orthogonalisation et le nombre de réorthogonalisations dans la méthode de Lanczos pour obtenir des vecteurs indépendants engendrant le sous-espace (cf. [R5.01.01] §6).

Le troisième est un paramètre d'ajustement pour déterminer la nullité d'un terme sur la surdiagonale de la matrice tridiagonale caractérisant le problème réduit obtenu par la méthode de Lanczos. C'est juste un critère de déflation et non, contrairement à ce que pourrait laisser croire son nom, un critère de qualité des modes.

Le dernier fixe le nombre d'itérations maximum pour la résolution du système réduit pour la méthode QR ([R5.01.01] annexe 1).

Remarque:

Lors des premiers passages, il est fortement conseillé de ne pas modifier ces paramètres qui concernent plutôt les arcanes de l'algorithme et qui sont initialisés empiriquement à des valeurs standards.

3.5.10 Opérandes de la méthode de Bathe & Wilson (si `METHODE='JACOBI'`)

◇ `PREC_BATHE =pbat` (1.10^{-10}) [DEFAULT]
◇ `NMAX_ITER_BATHE=nbat` (40) [DEFAULT]
◇ `PREC_JACOBI=pjaco` (1.10^{-2}) [DEFAULT]
◇ `NMAX_ITER_JACOBI=njaco` (12) [DEFAULT]

Les deux premiers paramètres permettent, respectivement, d'ajuster la précision de convergence et le nombre maximum d'itérations permises de la méthode de Bathe & Wilson (cf. [R5.01.01] §8).

Les deux autres ajustent la précision de la convergence et le nombre maximum d'itérations de la méthode de JACOBI (cf. [R5.01.01] annexe 3). Ce solveur modal global est utilisé pour calculer les modes propres de la matrice projetée par Bathe & Wilson.

Remarque:

Lors des premiers passages, il est fortement conseillé de ne pas modifier ces paramètres qui concernent plutôt les arcanes de l'algorithme et qui sont initialisés empiriquement à des valeurs standards.

3.5.11 Opérandes de la méthode QZ (si `METHODE='QZ'`)

```
◇ TYPE_QZ = / 'QZ_SIMPLE' [DEFAULT]
            / 'QZ_EQUI'
            / 'QZ_QR'
```

Ce paramètre permet de choisir une des variantes de l'algorithme QZ proposé par LAPACK. Le premier choix ('QZ_SIMPLE') désigne la méthode de base, le second ('QZ_EQUI') lui rajoute un pré-traitement d'équilibrage des termes de la matrice. Cela améliore souvent la qualité des modes mais, *a contrario*, si la matrice présente des termes très petits dus à des erreurs d'arrondis, cette phase engendre alors des modes parasites.

Quant au troisième choix ('QZ_QR'), il est réservé au cas symétrique défini positif (matrice de raideur réelle, condition de Dirichlet sans Lagrange, pas de flambement ou d'amortissement). Il est beaucoup plus rapide que les options précédentes.

3.6 Opérandes `SEUIL_FREQ`, `PREC_SHIFT`, `NMAX_ITER_SHIFT` et `NPREC_SOLVEUR`

```
◇ PREC_SHIFT      = ps      ( 0.05 )      [DEFAULT]
◇ SEUIL_FREQ      = sf      ( 0.01 )      [DEFAULT]
◇ NMAX_ITER_SHIFT = ns      ( 5 )         [DEFAULT]
◇ NPREC_SOLVEUR   = ndeci   ( 8 )         [DEFAULT]
```

Le déroulement d'un calcul modal dans cet opérateur requiert la factorisation LDL^T de matrices dynamiques $\mathbf{Q}(I)$ du type (cf. [R5.01.01/02] §2.5/3.8)

$$\mathbf{Q}(I) = \mathbf{A} - I \mathbf{B} \text{ (GEP)}$$

$$\mathbf{Q}(I) = I^2 \mathbf{B} + I \mathbf{C} + \mathbf{A} \text{ (QEP)}$$

Ces factorisations sont tributaires d'instabilités numériques lorsque le shift I est proche d'une valeur propre du problème. Cette détection s'opère en comparant la perte de décimales des termes diagonaux de cette factorisée par rapport à leurs valeurs initiales (en valeur absolue). Si le maximum de cette perte est supérieure à `ndeci`, la matrice est supposée singulière et on cherche une valeur décalée du shift (à chaque fois de `ps`%) procurant une matrice inversible. On réitère l'opération `ns` fois (cf. [R5.01.01] algorithme n°1). Si au bout de ces `ns` tentatives, la matrice décalée n'est toujours pas inversible, on émet une information, une alarme ou on s'arrête en erreur fatale, suivant les cas de figure.

Si au cours de ces décalages, le shift prend une valeur inférieure (en module) à `sf`, alors on lui impose la valeur $I = -sf$. Ce paramètre correspond à une valeur seuil en dessous de laquelle on considère qu'on a une valeur numériquement nulle. Cette imposition permet ainsi de distinguer ces modes rigides du reste du spectre.

Cette valeur `sf` sert aussi à détecter les valeurs propres quasi-nulles lors du post-traitement de vérification sur la norme du résidu (cf. [R5.01.01/02] algorithme n°2/n°1).

Remarque:

Lors des premiers passages, il est fortement conseillé de ne pas modifier ces paramètres qui concernent plutôt les arcanes de l'algorithme et qui sont initialisés empiriquement à des valeurs standards.

3.7 Mot clé `VERI_MODE`

- ◇ `VERI_MODE=F(...`
Mot clé facteur pour la définition des paramètres de la vérification des modes propres ([R5.01.01] §3.8).

3.7.1 Opérande `PREC_SHIFT`

- ◇ `PREC_SHIFT=ps2 (0,05) [DEFAULT]`
Dans la partie post-traitement, on se sert de la valeur paramétrée par `ps2`, pour déterminer les extrémités des bandes du test de Sturm (uniquement en GEP à modes réels, cf. [R5.01.01] algorithme n°3).

3.7.2 Opérande `STOP_ERREUR`

- ◇ `STOP_ERREUR=/ 'OUI' [DEFAULT]`
`/ 'NON'`
Permet d'indiquer à l'opérateur s'il doit s'arrêter ('OUI') ou continuer ('NON') dans le cas où l'un des critères `SEUIL` ou `STURM` (uniquement avec `MODE_ITER_SIMULT`) n'est pas vérifié.
Par défaut le concept de sortie n'est pas produit.

3.7.3 Opérande `SEUIL`

- ◇ `SEUIL=r (1.10-6) [DEFAULT]`
Seuil de tolérance pour la norme d'erreur relative du mode au dessus duquel il est considéré comme faux ou trop approximé (cf. [R5.01.01/02] algorithme n°2/n°1). Voir aussi paramètre `STOP_ERREUR`.

3.7.4 Opérande `STURM`

- ◇ `STURM=/ 'OUI' [DEFAULT]`
`/ 'NON'`
Vérification dite de `STURM` ('OUI') permettant de s'assurer que l'algorithme utilisé dans l'opérateur a déterminé le nombre exact de valeurs propres dans l'intervalle de recherche (§3.5/6/8 [R5.01.01]). Cette option n'a d'intérêt qu'en GEP à modes réels (donc pas avec **K** complexe et avec des matrices non symétriques). Voir aussi paramètre `STOP_ERREUR`.

3.7.5 Opérande `PREC_SHIFT`

- ◇ `PREC_SHIFT=prs (0.05) [DEFAULT]`
Ce paramètre (qui est un pourcentage) permet de définir un intervalle contenant les valeurs propres calculées, pour lequel la vérification de Sturm sera effectuée (§2.6 [R5.01.01]). Cette option a d'intérêt qu'en GEP à modes réels.

3.8 Opérandes `SENSIBLITE`

- ◇ `SENSIBILITE=`
Active le calcul de la dérivée des modes par rapport à un paramètre sensible du problème.
Il est à noter qu'à l'heure actuelle, la dérivée des modes multiples n'est pas disponible, car elle pose des problèmes théoriques et pratiques particuliers.
Le document [U4.50.02] précise le fonctionnement du mot clé.

3.9 Opérande `STOP_FREQ_VIDE`

◇ `STOP_FREQ_VIDE=/ 'OUI' [DEFAULT]`
`/ 'NON'`

'OUI' arrête le calcul si aucune valeur propre n'est détectée dans la bande stipulée par l'utilisateur: une exception (nommée `BandeFrequenceVide`) est émise. Elle peut être traitée pour continuer le déroulement de l'étude. On peut trouver un exemple sous le cas test SDLL11a:

```
try:
    MODE1=MODE_ITER_SIMULT (MATR_A=K_ASSE,MATR_B=M_ASSE,
                           CALC_FREQ=_F (OPTION='BANDE',
                           FREQ=(100.,200.)))
except aster.BandeFrequenceVideError:
    MODE1=MODE_ITER_SIMULT (MATR_A=K_ASSE,MATR_B=M_ASSE,
                           CALC_FREQ=_F (OPTION='BANDE',
                           FREQ=(200.,3500.,)))
```

'NON' n'arrête pas le calcul (émission seulement d'une `ALARME`) si aucune valeur propre n'est détectée dans la bande stipulée par l'utilisateur.

Ce mot-clé est utilisé dans la macro-commande `MACRO_MODE_MECA`[U4.52.02] afin de permettre l'absence de valeurs propres dans une bande de recherche. Cette option n'a pas d'intérêt avec la méthode QZ.

3.10 Opérande `INFO`

◇ `INFO=/1 [DEFAULT]`
`/2`

Indique le niveau d'impression dans le fichier `MESSAGE`.

- 1: Impression sur le fichier 'MESSAGE' des valeurs propres, de leur position modale, de l'amortissement réduit, de la norme d'erreur a posteriori et de certains paramètres utiles pour suivre le déroulement du calcul (Cf. §5.2)
- 2: Impression plutôt réservée aux développeurs.

3.11 Opérande `TITRE`

◇ `TITRE=ti`
Titre attaché au concept produit par cet opérateur [U4.03.01].

4 Phase de vérification

On vérifie selon les options:

`OPTION='BANDE'`

L'argument du mot-clé `FREQ` ou du mot-clé `CHAR_CRIT` doit fournir exactement **deux** valeurs,

`OPTION='CENTRE'`

L'argument du mot-clé `FREQ` ou du mot-clé `CHAR_CRIT` doit fournir exactement **une** seule valeur,

`OPTION='PLUS_PETITE'`

L'argument du mot clé `FREQ` ou du mot-clé `CHAR_CRIT`, est ignoré.

Si les précisions et les nombres maximaux d'itérations sont irréalistes (par exemple des précisions inférieures à la précision machine ou des nombres d'itérations négatifs), on n'effectue pas le calcul.

5 Phase d'exécution

5.1 Vérification

Les matrices A , B (et C) arguments des mots-clé `MATR_A` et `MATR_B` (et `MATR_C`), doivent être cohérentes entre elles (c'est à dire s'appuyer sur la même numérotation et le même mode de stockage).

5.2 Actions par défaut

Si le mot-clé `DIM_SOUS_ESPACE` n'est pas renseigné ou est initialisé à une valeur strictement inférieure au nombre de fréquences demandées `nf` (opérande `NMAX_FREQ`), l'opérateur calcule automatiquement une dimension admissible pour le sous-espace de projection *via* les formules empiriques (cf §3.6.7):

```
METHODE='SORENSEN'
```

```
ndim=MIN(MAX(2+nf,mse*nf),nactif) avec mse=2 par défaut.
```

```
METHODE='TRI_DIAG'
```

```
ndim=MIN(MAX(7+nf,mse*nf),nactif) avec mse=4 par défaut.
```

```
METHODE='JACOBI'
```

```
ndim=MIN(MAX(7+nf,mse*nf),nactif) avec mse=2 par défaut.
```

où `nactif` est le nombre de ddl actifs (c'est-à-dire le nombre total de ddl moins le nombre de ddls de `LAGRANGE` et moins le nombre de relations linéaires qui lient des ddls entre eux, [R5.01.01] §3.2) et `mse` est le facteur de proportionnalité fixé par `COEF_DIM_ESPACE`.

Si l'on résout un GEP, la dimension du sous-espace est doublée. Les valeurs de ces différents paramètres sont imprimées dans le fichier `MESSAGE`.

6 Paramètres modaux/ Norme des modes/ Position modale

En sortie de cet opérateur, les modes propres réels ou complexes sont normalisés à la plus grande des composantes qui n'est pas un multiplicateur de `LAGRANGE`. Pour choisir une autre norme, il faut utiliser la commande `NORM_MODE` [U4.52.11].

Dans le cas d'un calcul dynamique, la structure de données `mode_meca_*`, contient, en plus des fréquences de vibration et des déformées modales associées, des paramètres modaux (masse généralisée, raideur généralisée, facteur de participation, masse effective). On trouvera la définition de ces paramètres dans [R5.01.03].

Dans le cas d'un calcul de flambement linéaire, la structure de données `mode_flamb`, ne contient que les charges critiques et les déformées associées.

Dans le cas d'un calcul dynamique généralisé à matrices réelles symétriques, la position modale correspond à la position du mode dans l'ensemble du spectre défini par les matrices initiales.

Dans tous les autres cas, les positions modales sont attribuées de 1 à `nf` (`nf` étant le nombre de modes retenus) en les classant par ordre croissant algébrique. Toutes les positions modales sont donc positives.

7 Impression des résultats

Pour afficher les paramètres modaux associés à chaque mode et les coordonnées des modes, il faut utiliser l'opérateur `IMPR_RESU[U4.91.01]` de la manière suivante:

- Affichage des paramètres modaux seulement sous forme de table:

```
IMPR_RESU (RESU=_F (RESULTAT=mode,  
                    TOUT_PARA='OUI',  
                    TOUT_CHAM='NON' ) ) ;
```

- Affichage des paramètres modaux et des vecteurs propres:

```
IMPR_RESU (RESU=_F (RESULTAT=mode,  
                    TOUT_PARA='OUI',  
                    TOUT_CHAM='OUI' ) ) ;
```

8 Tri de modes / Caractérisation de `mode_meca_*`

Par exemple, lors de sollicitations sismiques en analyse modale, la base modale utilisée doit contenir les modes qui ont une masse effective unitaire importante dans la direction du séisme.

La commande `EXTR_MODE` [U4.52.12] permet d'extraire dans une structure de données de type `mode_meca_*` des modes qui vérifient un certain critère et de concaténer plusieurs structures de données de type `mode_meca_*`.

Une macro-commande, permettant d'enchaîner les commandes `MODE_ITER_SIMULT`, `NORM_MODE` et `EXTR_MODE` a été créée : `MACRO_MODE_MECA` [U4.52.02].

9 Exemples

9.1 Calcul des 5 modes propres les plus proches d'une fréquence donnée (100 Hz)

```
mode=MODE_ITER_SIMULT(MATR_A=rigid,  
                      MATR_B=masse,  
                      CALC_FREQ=_F(OPTION='CENTRE',  
                                   FREQ=100.,  
                                   NMAX_FREQ=5 )  
                      );
```

9.2 Calcul des charges critiques contenues dans une bande

```
mode=MODE_ITER_SIMULT(MATR_A=rigid,  
                      MATR_B=riggeo,  
                      TYPE_RESU='MODE_FLAMB',  
                      CALC_FREQ=_F(OPTION='BANDE',  
                                   CHAR_CRIT=(-1.E8,1.5E8))  
                      );
```