

Manuel d'Utilisation
Fascicule U4.5- : Méthodes de résolution
Document : U4.52.04

Opérateur *MODE_ITER_INV*

1 But

Calculer des valeurs et vecteurs propres par la méthode des itérations inverses. Le cas du problème généralisé (calcul de type dynamique sans amortissement ou de type flambement d'Euler) et le cas du problème quadratique (calcul de type dynamique avec amortissement) sont traités.
Produit un concept *mode_meca_** (cas dynamique) ou *mode_flamb* (cas flambement d'Euler).

2 Syntaxe

```

mode      [*] = MODE_ITER_INV

# DONNEES DU PROBLEME MODAL
(  ♦ MATR_A =  A                                / [matr_asse_DEPL_R]
                                     / [matr_asse PRES_R]
                                     / [matr_asse_GENE_R]
    ♦ MATR_B =  B                                / [matr_asse_DEPL_R]
                                     / [matr_asse PRES_R]
                                     / [matr_asse_GENE_R]
    ♦ MATR_C =  C                                [matr_asse_DEPL_R]

# TYPE DE PROBLEME
    ♦ TYPE_RESU =      / 'DYNAMIQUE'              [DEFAULT]
                       / 'MODE_FLAMB'

# PHASE HEURISTIQUE
# TYPE DE CALCUL MODAL

    ♦ CALC_FREQ = _F(  ♦ OPTION =  /'PROCHE'
                                     /'SEPRE'
                                     /'AJUSTE'              [DEFAULT]
                       ♦ NMAX_FREQ =      / 0          [DEFAULT]
                                     / nf      [I]

#      SI TYPE_RESU = 'DYNAMIQUE'
                       ♦ FREQ =          lfreq          [1_R]
                       ♦ AMOR_REDUIT =    lamor          [1_R]

#      SI TYPE_RESU = 'MODE_FLAMB'
                       ♦ CHAR_CRIT =      lcharc         [1_R]

#      SI OPTION = 'SEPRE' ou 'AJUSTE'
                       ♦ NMAX_ITER_SEPRE =    / 30      [DEFAULT]
                                     / nis      [I]
                       ♦ PREC_SEPRE =        / 1.E-4   [DEFAULT]
                                     / ps       [R]

#      SI OPTION = 'AJUSTE'
                       ♦ NMAX_ITER_AJUSTE =    / 15      [DEFAULT]
                                     / nia      [I]
                       ♦ PREC_AJUSTE =        / 1.E-4   [DEFAULT]
                                     / pa       [R]

# SENSIBILITE
    ♦ SENSIBILITE =  (
                                     . . . voir [U4.50.02] . . .
                                     )

```

Titre : **Opérateur MODE_ITER_INV**
Auteur(s) : **E. BOYERE**

Clé : **U4.52.04-I** Date : **10/04/07**
Page : **3/14**

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

# PHASE ITERATIONS INVERSES
      ◇ CALC_MODE = _F( ◇ OPTION =      / 'DIRECT'      [DEFAULT]
                                   / 'RAYLEIGH'
      ◇ NMAX_ITER =      / 30      [DEFAULT]
                                   / nim    [I]
      ◇ PREC =           / 1.E-5   [DEFAULT]
                                   / pm     [R]
    )

# POUR VERIFICATION FINALE
      ◇ VERI_MODE = _F( ◇ STOP_ERREUR = / 'OUI'        [DEFAULT]
                                   / 'NON'
      ◇ SEUIL =         / 1.E-2    [DEFAULT]
                                   / r    [R]
    )

# DIVERS
      ◇ INFO =           / 1
                                   / 2
                                   [DEFAULT]
      ◇ TITRE =          ti
    );

# DONNEE RESULTAT
      Si TYPE_RESU = 'MODE_FLAMB'      alors [*] -> mode_flamb
      Si MATR_C= [matr_asse_DEPL_R]     alors [*] -> mode_meca_C
      Si MATR_A= [matr_asse_DEPL_R]     alors [*] -> mode_meca
      Si MATR_A= [matr_asse_PRES_R]     alors [*] -> mode_acou
      Si MATR_A= [matr_asse_GENE_R]     alors [*] -> mode_gene
```

3 Opérands

3.1 Principes

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

Trouver (λ, \mathbf{x}) tels que $\mathbf{Ax} = \lambda \mathbf{Bx}$, $\mathbf{x} \neq 0$, où \mathbf{A} et \mathbf{B} sont des matrices symétriques à coefficients réels. 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 \mathbf{A} est la matrice de rigidité matérielle, notée \mathbf{K} , (éventuellement augmentée de la matrice de rigidité géométrique notée \mathbf{K}_g , si la structure est précontrainte) et \mathbf{B} est la matrice de masse ou d'inertie notée \mathbf{M} . Les valeurs propres obtenues sont les carrés des pulsations associées aux fréquences cherchées.

Le système à résoudre peut s'écrire: $\underbrace{(\mathbf{K} + \mathbf{K}_g)}_{\mathbf{A}} \mathbf{x} = \lambda \underbrace{\mathbf{M}}_{\mathbf{B}} \mathbf{x}$ où $\lambda = (2\pi f)^2$ est le carré de la

pulsation ω , f la fréquence propre et \mathbf{x} le vecteur de déplacement propre associé.

- **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 \mathbf{x} associé à ce niveau de charge critique $\mu = -\lambda$, se ramène à un problème généralisé aux valeurs propres de la forme: $(\mathbf{K} + \mu \mathbf{K}_g) \mathbf{x} = \mathbf{0} \Leftrightarrow \underbrace{\mathbf{K}}_{\mathbf{A}} \mathbf{x} = \lambda \underbrace{\mathbf{K}_g}_{\mathbf{B}} \mathbf{x}$ avec \mathbf{K} matrice de rigidité matérielle et

\mathbf{K}_g matrice de rigidité géométrique.

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 .

Cet opérateur permet aussi l'étude de la **stabilité dynamique d'une structure en présence d'amortissements et d'effets gyroscopiques**. Cela conduit à la résolution d'un problème modal d'ordre plus élevé, dit quadratique [R5.01.02]. On recherche alors des valeurs et vecteurs propres complexes par la méthode de Lanczos après avoir effectué une réduction linéaire du problème.

- Le problème consiste à trouver $(\lambda, \mathbf{x}) \in (C, C^N)$ tels que $(\lambda^2 \mathbf{B} + \lambda \mathbf{C} + \mathbf{A}) \mathbf{x} = \mathbf{0}$ où typiquement, en mécanique linéaire, \mathbf{A} sera la matrice de rigidité, \mathbf{B} la matrice de masse et \mathbf{C} la matrice d'amortissement. Les matrices \mathbf{A} , \mathbf{B} et \mathbf{C} sont des matrices à coefficients réels. La valeur propre complexe λ est liée à la fréquence propre f et à l'amortissement réduit ξ par : $\lambda = \xi (2\pi f) \pm i (2\pi f) \sqrt{1 - \xi^2}$.

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 le document [R5.01.01], 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				
<i>1^{ère} phase (heuristique)</i>				
Calcul de quelques modes	Bissection	'SEPRE'		
Calcul de quelques modes	Bissection + Sécante (géné.) Muller (quad.)	'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^{ème} phase (méthode des puissances proprement dite)</i>				
Méthode de base	Puissances inverses	'DIRECT'	Très bonne construction de vecteurs propres	Peu robuste
Option d'accélération	Quotient de Rayleigh	'RAYLEIGH'	Améliore la convergence	Coût calcul Non porté en quadratique
MODE_ITER_SIMULT				
Calcul d'une partie du spectre	Bathe & Wilson	'JACOBI'		Peu robuste Non porté en quadratique
	Lanczos (Newman- Pipano)	'TRI_DIAG'		Peu robuste
	IRAM (Sorensen)	'SORENSEN'	Robustesse accrue. Meilleures complexités calcul et mémoire. Contrôle de la qualité des modes.	Non porté en quadratique

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*, est souvent bien indiqué. Par contre, pour capturer une partie significatif du spectre, on a recourt à *MODE_ITER_SIMULT*, via les méthodes dites « de sous-espace ».

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

Elle consiste à coupler une phase heuristique de localisation des valeurs propres (détermination d'une valeur approchée de chaque valeur propre contenue dans un intervalle donné par une technique de bisection, affinée ou non, par une méthode de la sécante, en généralisé, ou par une méthode de Muller en quadratique), avec une phase d'itérations inverses proprement dite (accélérée par un quotient de Rayleigh ou non), qui va améliorer ces estimations tout en exhumant les vecteurs propres associés.

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`, `MATR_B`, `MATR_C`

◆ `MATR_A = A`

Matrice assemblée de type `[matr_asse*_R]` du système généralisé ou quadratique à résoudre.

◆ `MATR_B = B`

Matrice assemblée de type `[matr_asse*_R]` du système généralisé ou quadratique à résoudre.

◇ `MATR_C = C`

Matrice assemblée de type `[matr_asse*_R]` du système quadratique à résoudre.

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) ou recherche de charges critiques (cas de la théorie du flambement linéaire). 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. [§2.9] [§4.4] [R5.01.01]). 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. [§2.5] [§2.6] [R5.01.01]).
- **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 `CALC_FREQ`

◆ `CALC_FREQ = _F(...`

Mot-clé facteur pour la définition des paramètres de la première phase de calcul (localisation des valeurs propres).

Pour le problème généralisé, la localisation des valeurs propres s'effectue généralement par une séparation dichotomique des fréquences (pour les options '`AJUSTE`' et '`SEPRE`'), suivie d'une méthode de la sécante (pour l'option : '`AJUSTE`').

Pour le problème quadratique, cette localisation s'effectue par une résolution du problème non amorti (problème généralisé) suivie d'une méthode de Muller (pour l'option : '`AJUSTE`').

3.4.1 Opérande OPTION

◇ **OPTION** =

'PROCHE'

On recherche le mode dont la valeur propre est la plus proche d'une valeur donnée. Cette valeur est indiquée par :

- l'argument *lfreq* du mot clé **FREQ** pour un problème généralisé de type dynamique (*TYPE_RESU* = 'DYNAMIQUE').
- l'argument *lcharc* du mot clé **CHAR_CRIT** pour un problème généralisé de type flambement linéaire (*TYPE_RESU* = 'MODE_FLAMB').
- les arguments *lfreq* et *lamor* des mot clé **FREQ** et **AMOR_REDUIT** pour un problème quadratique de type dynamique (*TYPE_RESU* = 'DYNAMIQUE').

Il y a autant de recherches de modes que de termes dans cette liste (ou ces listes). Si on souhaite calculer un mode multiple, il ne faut pas utiliser cette option car on ne trouvera qu'un seul mode.

'SEPRE'

On sépare les valeurs propres par une méthode de bisection basée sur le critère de Sturm. Les bornes de l'intervalle de recherche sont :

- les arguments de la liste *lfreq* du mot clé **FREQ** pour un problème généralisé ou quadratique de type dynamique (*TYPE_RESU* = 'DYNAMIQUE').
- les arguments de la liste *lcharc* du mot clé **CHAR_CRIT** pour un problème généralisé de type flambement linéaire (*TYPE_RESU* = 'MODE_FLAMB').

'AJUSTE'

[DEFAULT]

Après avoir séparé les fréquences propres, comme pour l'option 'SEPRE' on effectue des itérations supplémentaires soit par la méthode de la sécante (problème généralisé) soit par la méthode de Muller (problème quadratique) pour obtenir une meilleure précision sur la valeur propre.

3.4.2 Opérande FREQ

◇ **FREQ** = *lfreq*

Pour un problème de recherche de valeur propres de type dynamique (*TYPE_RESU* = 'DYNAMIQUE'), ce mot-clé correspond à la liste des fréquences dont l'utilisation dépend de l'OPTION choisie.

Si option 'PROCHE' : c'est la liste des fréquences dont on cherche le mode le plus proche. La liste a au moins 1 élément et est ordonnée par ordre croissant.

Si option 'SEPRE' ou 'AJUSTE' : ce sont les bornes des intervalles de recherche

FREQ : (*f*₁, *f*₂, ..., *f*_{*n*-1}, *f*_{*n*})

On cherchera à séparer les fréquences dans les intervalles

[*f*₁,*f*₂] , [*f*₂,*f*₃] [*f*_{*n*-2},*f*_{*n*-1}] , [*f*_{*n*-1},*f*_{*n*}]

La liste a au moins 2 éléments. Les fréquences sont positives. On vérifie que les fréquences sont données dans l'ordre croissant.

3.4.3 Opérande AMOR_REDUIT

◇ `AMOR_REDUIT = lamor`

Pour le problème quadratique de type dynamique (`TYPE_RESU = 'DYNAMIQUE'`), et si l'option `PROCHE` a été choisie, on peut initialiser la méthode des itérations inverses à partir d'une valeur propre initiale complexe. Pour construire cette valeur complexe, on utilise la liste des arguments donnés sous les mot-clés `FREQ` (liste de fréquences) et `AMOR_REDUIT` (liste d'amortissements). Ces deux listes doivent avoir le même nombre d'arguments.

3.4.4 Opérande CHAR_CRIT

◇ `CHAR_CRIT = lcharc`

Pour un problème de recherche de valeur propres de type flambement d'Euler (`TYPE_RESU = 'MODE_FLAMB'`), ce mot-clé correspond à la liste des charges critiques dont l'utilisation dépend de l'`OPTION` choisie.

Si option '`PROCHE`' : c'est la liste des charges critiques dont on cherche le mode le plus proche. La liste a au moins 1 élément.

Si option '`SEPARE`' et '`AJUSTE`' : ce sont les bornes des intervalles de recherche

`CHAR_CRIT` : $(\lambda_1, \lambda_2, \dots, \lambda_{n-1}, \lambda_n)$

On cherchera à séparer les charges critiques dans les intervalles

$[\lambda_1, \lambda_2]$, $[\lambda_2, \lambda_3]$ $[\lambda_{n-2}, \lambda_{n-1}]$, $[\lambda_{n-1}, \lambda_n]$

La liste a au moins 2 éléments. Les charges critiques sont négatives ou positives. On vérifie que les charges critiques sont données dans l'ordre croissant.

3.4.5 Opérande NMAX_FREQ

◇ `NMAX_FREQ = nf` **(0)** **[DEFAULT]**

Nombre maximum de valeurs propres à calculer. Cet opérande est ignoré pour l'option '`PROCHE`'.

Pour les autres options, si l'utilisateur ne renseigne pas ce mot-clé, toutes les valeurs propres contenues dans les intervalles précisés par l'utilisateur sont calculées. Sinon, les `NMAX_FREQ` premières valeurs propres, donc les plus basses, sont calculées.

3.4.6 Opérandes de la bisection (si `OPTION = 'SEPARE' ou 'AJUSTE'`)

◇ `NMAX_ITER_SEPARE = nis` **(30)** **[DEFAULT]**

◇ `PREC_SEPARE = ps` **(1.10^{-4})** **[DEFAULT]**

Paramètres d'ajustement du nombre d'itérations et de la précision de séparation pour la recherche par dichotomie. Ces opérandes sont ignorés pour l'option '`PROCHE`' (Cf. [R5.01.01 §3.2.1]).

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.

3.4.7 Opérandes de la sécante (si OPTION = 'AJUSTE')

◇ NMAX_ITER_AJUSTE = nia (15) [DEFAULT]
◇ PREC_AJUSTE = pa (1.10⁻⁴) [DEFAULT]

Paramètres d'ajustement du nombre d'itérations et de la précision de séparation pour la méthode de la sécante. Ces opérandes ne servent qu'à l'option 'AJUSTE' (Cf. [R5.01.01 §3.2.2]).

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.

3.4.8 Opérandes SEUIL_FREQ, PREC_SHIFT et NMAX_ITER_SHIFT

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

Pour les trois options possibles 'PLUS_PETITE', 'BANDE' ou 'CENTRE', on effectue une factorisation \mathbf{LDL}^T de la matrice $\left(A - (2\pi f_*)^2 B \right)$. f_* dépend de la méthode utilisée. Si f_* est détectée comme étant une fréquence propre ou étant située à proximité de fréquences propres (perte de plus de $n_{\text{deci}}=8$ décimales lors de la factorisation des matrices), la fréquence f_* est alors modifiée (cf. §2.6 et 2.9 [R5.01.01]):

$$f_*^- = f_* \times (1 - ps) \quad \text{ou} \quad f_*^+ = f_* \times (1 + ps)$$

Dans le cas où $\left(A - (2\pi f_*)^2 B \right)$ est non factorisable \mathbf{LDL}^T et $(|f_*| \leq sf)$, on effectue la modification suivante : $f_*^- = -sf$. On considère alors que f_* est associée à un mode de corps rigide. La modification de cette fréquence permet a priori de comptabiliser tous les modes de corps rigide. On n'effectue pas plus de ns modifications de la valeur f_* .

Dans le cas du flambement linéaire, la transposition est immédiate en remplaçant f_* (fréquence de vibration) par λ_* (charge critique), $(2\pi f_*)^2$ par λ_* et sf par $(2\pi sf)^2$.

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.

3.4.9 Opérande NPREC_SOLVEUR

◇ NPREC_SOLVEUR = ndeci (8) [DEFAULT]

ndeci représente le nombre de décimales qu'on s'autorise à perdre lors de la factorisation de la matrice shiftée $\left(A - (2\pi f_*)^2 B \right)$ ou $(A - \lambda B)$. Si on perd plus de ndeci décimales, la matrice est considérée comme non inversible (cf. [§2.6] et [§2.9] [R5.01.01]).

Remarque :

Lors des premiers passages, il est fortement conseillé de ne pas modifier ce paramètre qui concerne plutôt une arcane de l'algorithme et qui est initialisé empiriquement à une valeur standard.

3.5 Mot clé CALC_MODE

◇ CALC_MODE = _F(...

Mot-clé facteur pour la définition des paramètres de calcul de la deuxième phase de calcul (méthode des puissances inverses).

3.5.1 Opérande OPTION

◇ OPTION =

Définition de variante pour l'itération inverse proprement dite (cf. [R5.01.01 §3.3]):

'DIRECT'	Itération inverse standard (seule option admise pour le problème quadratique),
[DEFAULT]	
'RAYLEIGH'	Itération inverse avec quotient de Rayleigh (sans effet sur le problème quadratique).

3.5.2 Opérande NMAX_ITER

◇ NMAX_ITER = nim (30) [DEFAULT]

Nombre maximum d'itérations pour la recherche des vecteurs propres.

3.5.3 Opérande PREC

◇ PREC = pm (1.10^{-5}) [DEFAULT]

L'itération se poursuit tant que l'écart relatif de norme sur les modes propres, entre deux itérés, est supérieur à pm.

3.6 Opérandes SENSIBLITE

◇ SENSIBLITE =

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.

3.7 Mot clé **VERI_MODE**

◇ `VERI_MODE = _F(...`

Mot clé facteur pour la définition des paramètres de la vérification des modes propres ([§2.9] [R5.01.01]).

Remarques :

- 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.
- Contrairement à son alter-ego, *MODE_ITER_SIMULT*, ce mot-clé facteur ne comporte pas de mot-clé du type *STURM* et *PREC_SHIFT*. La phase de post-traitement et de vérification ne comporte en effet pas de test de Sturm qui serait redondant avec la première partie heuristique. Les méthodes de type « puissance » étant moins robustes que celles de type « sous-espace », la valeur par défaut du seuil r est moins exigeante (10^{-2} au lieu de 10^{-6}).

3.7.1 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* n'est pas vérifié.

Par défaut le concept de sortie n'est pas produit.

3.7.2 Opérande **SEUIL**

◇ `SEUIL = r (1.10-2) [DEFAULT]`

Seuil de tolérance pour la norme d'erreur relative du mode au dessus duquel le mode est considéré comme faux.

La norme d'erreur relative du mode est :

$$\frac{\|(\mathbf{A} - \lambda \mathbf{B})\mathbf{x}\|_2}{\|\mathbf{A}\mathbf{x}\|_2}, \text{ pour } \lambda \neq 0 \text{ pour le problème généralisé et}$$

$$\frac{\|(\lambda^2 \mathbf{B} + \lambda \mathbf{C} - \mathbf{A})\mathbf{x}\|_2}{\|\mathbf{A}\mathbf{x}\|_2}, \text{ pour le problème quadratique}$$

3.8 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.
- 2 : Impression plutôt réservée aux développeurs.

3.9 Opérande **TITRE**

◇ `TITRE = ti`

Titre attaché au concept produit par cet opérateur [U4.03.01].

4 Phase d'exécution

4.1 Vérification

Les matrices **A** et **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).

L'opérateur vérifie que pour les options '`SEPRE`' et '`AJUSTE`', la liste des valeurs des arguments du mot clé `FREQ` a, au moins, deux termes.

Il vérifie aussi une certaine cohérence des paramètres des différents algorithmes.

4.2 Exécution

Pour l'option '`AJUSTE`', si la séparation n'est pas possible et que dans un intervalle donné il y a plus d'une valeur de fréquence propre, on n'applique pas la méthode d'ajustement à cet intervalle. Par contre, on effectuera lors du calcul des modes des réorthogonalisations par rapport aux modes précédents contenus dans l'intervalle (ceci permet de calculer des modes associés à une fréquence multiple).

Pour l'option '`SEPRE`', ayant obtenu un intervalle cernant une fréquence propre, on prend pour le calcul du mode le milieu de l'intervalle. Lors du calcul du mode, la valeur de la fréquence propre est encore affinée. C'est le résultat de l'itération inverse proprement dit.

5 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, la position modale des modes correspond à la position du mode dans l'ensemble du spectre défini par les matrices **A** et **B**.

Dans le cas d'un calcul de flambement linéaire, les positions modales des charges critiques sont attribuées de 1 à `nf` (`nf` étant le nombre de charges critiques calculées) en classant les charges critiques par ordre croissant en valeur absolue. Toutes les positions modales sont donc positives.

Pour l'option `PROCHE`, les positions modales sont attribuées de 1 à `nf` (`nf` étant le nombre de valeurs propres calculées), en prenant les valeurs propres dans l'ordre de la liste renseignée sous `FREQ` ou `CHAR_CRIT`.

6 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' ) ) ;
```

7 Exemples

Soient masse et rigidite deux matrices préalablement assemblées par l'opérateur ASSE_MATRICE à partir de matrices élémentaires de masse (OPTION = 'MASS_MECA') et de rigidité (OPTION = 'RIGI_MECA').

On calcule les modes de fréquence propre compris dans la bande 50 Hz à 150 Hz avec l'opérateur MODE_ITER_INV comme suit :

```
mode          = MODE_ITER_INV
                (  MATR_A=  rigidite,
                  MATR_B=  masse,
                  CALC_FREQ=_F (  OPTION = 'AJUSTE',
                                FREQ = (  50. , 150. ))
                );
```

On calcule les modes de fréquence propre les plus proches des fréquences 20 Hz et 50 Hz avec l'opérateur MODE_ITER_INV comme suit :

```
mode          = MODE_ITER_INV
                (  MATR_A=  rigidite,
                  MATR_B=  masse,
                  CALC_FREQ=_F (  OPTION = 'PROCHE',
                                FREQ = (  50. , 150. )),
                  CALC_MODE =_F (  OPTION = 'RAYLEIGH')
                );
```

L'accélération de convergence en utilisant le coefficient de Rayleigh a été sélectionnée.

8 Remarques d'utilisation

Le coût de cet opérateur peut être élevé car :

- chaque dichotomie nécessite une factorisation (si `OPTION = 'SEPRE'`),
- chaque itération de sécante (si `OPTION = 'AJUSTE'`) nécessite aussi une factorisation.

Il peut être plus judicieux de faire :

- une recherche de valeurs propres par l'opérateur `MODE_ITER_SIMULT` [U4.52.03],
- puis d'affiner les résultats obtenus par `MODE_ITER_INV` en utilisant l'option '`PROCHE`' de `CALC_FREQ` et l'option '`RAYLEIGH`' de `CALC_MODE` pour améliorer les vecteurs propres.