Notice d'utilisation des conditions aux limites traitées par élimination

Résumé

Le traitement des conditions aux limites du type Dirichlet par élimination (AFFE_CHAR_CINE) n'offre pas la même généralité que par dualisation (AFFE_CHAR_MECA par exemple).

Ce traitement est à utiliser lorsque l'on recherche à améliorer les temps d'éxécution d'un calcul ou si l'on souhaite travailler avec des matrices définies positives.

Notons que les conditions aux limites disponibles dans AFFE_CHAR_* (* = MECA/THER/ACOU) ne peuvent pas toutes être éliminées et traitées par AFFE_CHAR_CINE.

Dans ce document, on montre comment utiliser les « charges cinématiques » dans les jeux de commandes Aster.

Il y a 3 cas de figure (du plus simple au plus compliqué) :

- On utilise une commande de calcul « globale » (THER_LINEAIRE, STAT_NON_LINE, …). Dans ce cas, les charges cinématiques s'utilisent comme les autres charges.
- On souhaite faire un calcul de modes propres. Il faut alors ajouter un argument dans la commande ASSE_MATRICE.
- On souhaite faire un calcul « pas à pas » et résoudre les systèmes linéaires avec les commandes FACTORISER et RESOUDRE. Dans ce cas, il faut utiliser la commande CALC_CHAR_CINE.

1 Principe de l'élimination

On cherche à résoudre dans $\mathbb{R}^n$ le problème de minimisation sous contrainte (Pb1) suivant :

$$\min_{u \in U_G} \left(\frac{1}{2} u^T K u - u^T f \right) \quad \text{avec} \quad U_G = \{ u \in \mathbb{R}^n, u|_G = u_0 \}$$

où

- $u_0 \in \mathbb{R}^p$ est connu ($1 \leq p \leq n$)
- G est le sous ensemble de $N = \{1, \dots, n\}$, de cardinal p : $G = g_1 \dots g_p$
- $u|_G$ est la projection de u sur le sous espace engendré par $\{u_i\}_{i \in G}$
- où $(u_i)_j = \delta_{ij}, \forall j \in N$
- K est une matrice symétrique $n \times n$,
- $f \in \mathbb{R}^n$ est fixé.

La contrainte $u|_G = u_0$ représente des conditions aux limites de type Dirichlet homogène ou non.

Si on note $L = C_N G$ le complémentaire de G dans N , on peut, à l'aide des u_i définis précédemment, décomposer $\mathbb{R}^n$ en somme directe de V_G = espace vectoriel engendré par $\{u_i\}_{i \in G}$ et de V_L = espace vectoriel engendré par $\{u_i\}_{i \in L}$;

Dès lors, nous avons $\mathbb{R}^n = V_G \oplus V_L$
et l'on note $u = u_G \oplus u_K$ où $u_G = u|_G$ et $u_L = u|_L$
soit encore en notation vectorielle $u = \begin{Bmatrix} u_G \\ u_L \end{Bmatrix}$

Le problème (Pb1) peut donc s'écrire sous la forme du problème (Pb2) :

$$\begin{cases} \min \left(\frac{1}{2} u_G^T K_{GG} u_G + \frac{1}{2} u_L^T K_{LL} u_L + u_L^T K_{LG} u_G - u_L^T f_L - u_G^T f_G \right) \\ u_G \in V_G \\ u_L \in V_L \\ u_G = u_0 \end{cases}$$

Ce qui revient à écrire :

$$(\mathrm{Pb1}) \Leftrightarrow (\mathrm{Pb2}) \qquad \begin{cases} \min \left(\frac{1}{2} u_L^T K_{LL} u_L + u_L^T K_{LG} u_0 - u_L^T f_L \right) \\ u_L \in V_L \\ u = u_0 \oplus u_L \end{cases}$$

On a alors éliminé u_G du problème de minimisation.

Nous allons maintenant rechercher le problème matriciel associé à (Pb3).

On recherche u_L minimisant

$$\frac{1}{2} u_L^T K_{LL} u_L + u_L^T K_{LG} u_0 - u_L^T f_L$$

ce qui revient à résoudre le problème matriciel suivant :

$$K_{LL} u_L = F_L - K_{LG} u_0$$

On peut donc écrire :

$$(\mathrm{Pb1}) \Leftrightarrow (\mathrm{Pb2}) \Leftrightarrow (\mathrm{Pb3}) \Leftrightarrow \begin{bmatrix} K_{LL} & 0 \\ 0 & I_G \end{bmatrix} \begin{bmatrix} u_L \\ u_G \end{bmatrix} = \begin{bmatrix} f_L - K_{LG} u_0 \\ u_0 \end{bmatrix} \text{ , soit } K' \begin{bmatrix} u_L \\ u_G \end{bmatrix} = f'$$

2 Traitement dans *Aster*

2.1 Les charges cinématiques

Une charge cinématique (type *Aster* : char_cine_* [* = meca/ther/acou]) permet de caractériser l'ensemble G des ddl imposés et les $(u_0)_i$ pour $i \in G$ qui sont les valeurs affectées à ces ddl.

La définition d'une charge cinématique se fait par l'intermédiaire de l'opérateur `AFFE_CHAR_CINE` pour les $(u_0)_i$ constants ou fonctions de la géométrie ou du temps.

2.2 Les vecteurs cinématiques

Le vecteur cinématique est un cham_no_* qui représente le vecteur $\begin{bmatrix} 0 \\ u_0 \end{bmatrix}$.

A chaque charge cinématique correspond un vecteur cinématique.

Cette opération est effectuée par l'opérateur `CALC_CHAR_CINE`.

2.3 Calcul de K'

K' est directement calculée au moment de l'assemblage par l'opérateur `ASSE_MATRICE` sous réserve naturellement que l'on fournisse en argument une liste de charges cinématiques.

La structure de données `MATR_ASSE_*` a été modifiée de façon à pouvoir stocker K' quand cela est nécessaire.

2.4 Calcul de f'

Après l''opérateur `FACTORISER` le concept de type matr_asse_* produit, contient la factorisée de K' et la matrice K_{LG} inchangée.

Le calcul de f' s'effectue au moment de la résolution : il faut fournir à l'opérateur `RESOUDRE` en argument le vecteur cinématique correspondant à $\begin{bmatrix} 0 \\ u_0 \end{bmatrix}$ par l'intermédiaire du mot clé `CHAM_CINE`.

Cet opérateur calcule alors f' avant de résoudre $\mathrm{fact}(K') \begin{bmatrix} u_L \\ u_G \end{bmatrix} = f'$.

3 Exemples de fichiers de commandes

3.1 Calcul mécanique avec une commande globale (STAT_NON_LINE) :

```
DEPIMP=AFFE_CHAR_CINE( MODELE=MOD,
                     MECA_IMPO=_F(  GROUP_MA = 'LCD1', DY = -2.0))

RESU=STAT_NON_LINE( MODELE=MOD,  CHAM_MATER=CHMAT,
                    EXCIT= _F(  CHARGE = DEPIMP,   FONC_MULT = FONC),
                    ...)
```

3.2 Charges cinématiques pour un calcul de modes propres :

```
CHARCINE=AFFE_CHAR_CINE(MODELE=MODEL,
                        MECA_IMPO=_F(GROUP_MA='GM2', DX=0.0, DY=0.0))


KASS=ASSE_MATRICE(MATR_ELEM=KELEM,
                  NUME_DDL=NUME,
                  CHAR_CINE=CHARCINE,);

MASS=ASSE_MATRICE(MATR_ELEM=MELEM,
                  NUME_DDL=NUME,
                  CHAR_CINE=CHARCINE,);

# calcul des modes propres de la structure
MODES=MODE_ITER_SIMULT(MATR_RIGI=KASS,   MATR_MASS=MASS,
                       CALC_FREQ=_F(  NMAX_FREQ=10,))
```

3.3 Calcul "pas à pas" en utilisant les commandes FACTORISER et RESOUDRE :

```
CHCINE=AFFE_CHAR_CINE(   MODELE=MO,  MECA_IMPO=(
                _F( GROUP_NO = 'SUPY',    DY = 0.),
                _F( GROUP_NO = 'CHARGE', DX = -1.)))


MEL=CALC_MATR_ELEM(   MODELE=MO, CHAM_MATER=CHMAT,  OPTION='RIGI_MECA')


NU=NUME_DDL(  MATR_RIGI=MEL )

MATAS=ASSE_MATRICE( MATR_ELEM=MEL,  NUME_DDL=NU,  CHAR_CINE=CHCINE)


SCMBRE=CREA_CHAMP( ... )

VCINE=CALC_CHAR_CINE( NUME_DDL=NU, CHAR_CINE=CHCINE )

MATAS=FACTORISER(reuse=MATAS,  MATR_ASSE=MATAS )

DEP=RESOUDRE(MATR=MATAS, CHAM_NO=SCMBRE,  CHAM_CINE=VCINE)
```