Mise en œuvre de calcul par sous-structuration dynamique

Résumé :

Ce document a pour but d'aider et de donner des conseils aux utilisateurs pour la réalisation d'une étude faisant intervenir la technique de sous-structuration dynamique.

On présente d'abord la démarche générale pour la mise en œuvre de la technique.

On décrit par la suite les enchaînements des opérations nécessaires afin d'obtenir des résultats exploitables. La présentation est illustrée par trois exemples traitant un calcul modal, un calcul de réponse transitoire et un calcul de réponse harmonique.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Table des matières

1En quoi consiste la sous-structuration dynamique?....3
2Pourquoi utiliser la sous-structuration dynamique?....3
3Un petit rappel de la méthodologie....3
4Mise en œuvre dans Code_Aster....4
4.1Notion de macro-élément dynamique....4
4.2Le déroulement d'une étude....4
4.3Conseils pratiques....5
5Quelques exemples de calcul....6
5.1Un exemple de calcul modal....6
5.2Un exemple de calcul de réponse transitoire....10
5.3Un exemple de calcul de réponse harmonique....14

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

1 En quoi consiste la sous-structuration dynamique?

La sous-structuration dynamique consiste à partitionner la structure en plusieurs sous-structures afin de réduire la taille du modèle numérique et de pouvoir étudier indépendamment chaque sous-structure. Le modèle d'ensemble s'obtient par la suite en assemblant les modèles des différentes sous-structures.
Chaque sous-structure est projetée sur une base de vecteurs judicieusement choisie . Cette projection permet de réduire la taille du modèle de chaque sous-structure et par conséquent la taille du modèle de la structure assemblée.

2 Pourquoi utiliser la sous-structuration dynamique?

La complexité croissante des structures mécaniques rend difficile l'estimation de son comportement dynamique sous sollicitation imposée. La modélisation numérique de la structure complète nécessiterait un investissement important tant au niveau espace mémoire que coût calcul.
En plus de la réduction de la taille du modèle, la technique de sous-structuration dynamique rend beaucoup plus modulaire la modélisation de la structure globale.

3 Un petit rappel de la méthodologie

On ne rappelle ici que les principes fondamentales de la technique. Pour plus de détail, le lecteur est invité à consulter le document de référence [R4.06.02].

Le fait de projeter la sous-structure sur une base réduite, entraîne nécessairement une approximation sur les résultats obtenus. Le rôle du modélisateur est de choisir une base assez riche afin de pouvoir saisir le comportement dynamique de la sous-structure. La qualité des résultats dépend du choix de cette base de projection. Les techniques les plus utilisées pour le choix de cette base sont la technique de Craig Bampton et la technique de Mac Neal. À ces deux techniques s'ajoute l'utilisation de modes d'interface qui est une variante de ces deux techniques où on effectue une condensation des degrés de liberté de liaison.

- La technique de Craig-Bampton consiste à choisir comme base de projection, des modes normaux et des modes contraints. Les modes normaux sont des modes propres de la sous-structure où on bloque tous les nœuds des interfaces avec les sous-structures adjacentes. Un mode contraint est défini par la déformée statique obtenue en imposant un déplacement unité sur un degré de liberté de liaison, les autres degrés de liberté de liaison étant bloqués. La base de projection contient des modes normaux et tous les tous les modes contraints associés à tous les degrés de liberté de liaison.
- La technique de Mac Neal consiste à choisir comme base de projection, des modes normaux et des modes d'attache. Dans ce cas, les modes normaux sont des modes propres de la sous-structure à interface libre. Un mode d'attache est défini par la déformée statique obtenue en imposant une force unitaire à un degré de liberté de liaison, les autres degrés de liberté de liaison étant libres. La base de projection contient des modes normaux et tous les modes d'attache associés à tous les degrés de liberté de liaison.
- Utilisation de modes d'interface : On peut constater que la dimension de la base de projection peut devenir très grande si le nombre de degré de liberté de liaison est important. On définit alors la notion de mode d'interface qui consiste à effectuer une condensation du comportement de la structure sur les degrés de liberté d'interface et résoudre le problème aux valeurs propres associé. Cela permet d'exprimer le comportement de la structure réduite à l'interface. On se limite de conserver comme vecteurs de base les premiers modes calculés en complément des modes normaux. En effet, si l'on reste dans le cadre de la dynamique basse fréquence, l'ensemble des modes contraints (ou d'attache) n'est pas nécessaire pour représenter la dynamique du problème. Néanmoins, il est recommandé de contrôler a posteriori la pertinence de la base choisie en vérifiant par exemple la continuité des déplacements aux interfaces.

On peut lister ci-après les différentes étapes de la démarche :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- Pour chaque sous-structure :
  - Calcul de la base de projection : modes normaux et modes de liaison ou modes d'interface
  - Calcul des matrices généralisées de la sous-structure : projection des matrices de masse et de rigidité de la sous-structure sur la base réduite
- Assemblage des sous-structures par la formulation de la continuité du déplacement aux interfaces et la nullité du travail des forces de liaison. On obtient ainsi un modèle généralisé de la structure assemblée
- Calcul de la réponse en coordonnées généralisées
- Restitution des résultats sur base physique
- contrôle des résultats obtenus
- post-traitement (calcul de contrainte, calcul d'énergie, ...)

4 Mise en œuvre dans Code_Aster

4.1 Notion de macro-élément dynamique

Un macro-élément dynamique est un modèle réduit (généralisé) d'une sous-structure. Il comprend :
- la projection du modèle (matrices de masse et de rigidité) de la sous-structure sur la base réduite,
- les informations sur les interfaces où vont être connectées les sous-structures adjacentes et les matrices de liaison associées à ces interfaces.

Normalement, on construit autant de macro-élément dynamique que de sous-structure. Dans le cas où on est en présence de sous-structures qui ne diffèrent que par leur position dans l'espace, un seul macro-élément est suffisant. On applique par la suite, une rotation et/ou translation du macro-élément lors de l'assemblage.

4.2 Le déroulement d'une étude

- Pour chaque sous-structure : création du macro-élément dynamique
  - Calcul et assemblage des matrices de masse, de rigidité et du chargement de la sous-structure : (CALC_MATR_ELEM, ASSE_MATRICE, CALC_VECT_ELEM, ASSE_VECTEUR)
  - Calcul des vecteurs de base de projection : modes normaux, modes statiques, modes d'interface (CALC_MODES, MODE_STATIQUE, DEFI_BASE_MODALE)
  - Définition des interfaces (DEFI_INTERF_DYNA)
  - Projection des matrices de la sous-structure sur les vecteurs de base (MACR_ELEM_DYNA)

L'enchaînement de ces opérations peut être réalisé avec un seul appel à l'opérateur CREA_ELEM_SSD. Cet opérateur permet de créer un macro-élément dynamique en renseignant uniquement les informations sur le modèle et les interfaces. Il calcule les vecteurs de base de projection (modes normaux et modes de liaison), définit les interfaces et projette les matrices de rigidité et de masse sur la base calculée. Il enchaîne les opérateurs suivants : CALC_MATR_ELEM, ASSE_MATRICE, CALC_MODES, MODE_STATIQUE, DEFI_INTERF_DYNA, DEFI_BASE_MODALE, MACR_ELEM_DYNA.

- Assemblage des différentes sous-structures
  - Définition du modèle généralisé de la structure assemblée : liaison entre sous-structures et numérotation des degrés de liberté du système (DEFI_MODELE_GENE, NUME_DDL_GENE)
  - Assemblage des matrices généralisées des sous-structures (ASSE_MATR_GENE, ASSE_VECT_GENE)

Ces opérations d'assemblage de macro-éléments dynamiques peuvent être réalisées en utilisant l'opérateur ASSE_ELEM_SSD. Il définit le modèle généralisé et assemble les matrices généralisées des différentes sous-structures. Il enchaîne les opérateurs suivants : DEFI_MODELE_GENE, NUME_DDL_GENE, ASSE_MATR_GENE.

- Calcul de réponse sur le modèle généralisé (CALC_MODES, DYNA_VIBRA)

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- Restitution des résultats en coordonnées physiques de la sous-structure ou sur un maillage squelette (REST_SOUS_STRUCT)
- Contrôle a posteriori des résultats en vérifiant la continuité du champ de déplacement aux interfaces (POST_RELEVE_T, TEST_TABLE) et la nullité du travail des forces de liaison (CALC_CORR_SSD)

4.3 Conseils pratiques

- Pour les cas standards : utiliser l'opérateur CREA_ELEM_SSD pour la création du macro-élément dynamique et l'opérateur ASSE_ELEM_SSD pour l'assemblage des macro-éléments. Cela facilite de façon significative l'écriture du fichier de commande.
- Pour les cas non standards (base de projection particulière), il est des fois, nécessaire de revenir aux opérateurs élémentaires. Les limitations actuelles de CREA_ELEM_SSD et ASSE_ELEM_SSD sont :
  - structure conservative (sans amortissement)
  - base de projection : modes normaux + modes de liaison (Craig-Bampton, Mac Neal, les n premiers modes d'interface)
  - Toutes les composantes des degrés de liberté des nœuds interfaces sont considérées pour le calcul des modes de liaison (le mot clé MASQUE qui permet de sélectionner les composantes actives des nœuds interfaces n'est pas actif pour CREA_ELEM_SSD)
  - Les modes normaux sont calculés en utilisant l'opérateur CALC_MODES, les options sur la dimension du sous-espace de recherche de modes sont ceux pris par défaut par cet opérateur.
- Si on est en présence d'une interface ponctuelle ou à faible nombre de ddl, l'utilisation des méthodes classiques (Craig-Bampton ou Mac Neal) est préconisée.
- Les modes d'interface sont appropriés pour les interfaces réparties (non ponctuelles). La qualité des résultats dépend du nombre de modes d'interface pris en compte dans la base de projection. Un contrôle a posteriori des résultats est toujours indiqué. Ce contrôle peut se faire en vérifiant la continuité du champ de déplacement aux interfaces et/ou la nullité du travail des forces de liaison. Il faut augmenter le nombre de modes d'interface si on observe un décollement au niveau des interfaces ou si le travail des forces de liaison n'est pas négligeable.
- Bien choisir les modes normaux qui servent de vecteurs de base de projection. Le sous-espace engendré par ces vecteurs de base doivent contenir au minimum la déformée de la sous-structure après assemblage. On ne peut pas, par exemple, observer la torsion de la sous-structure si on ne met pas une déformée de torsion parmi les vecteurs de base.
- Pour une interface de type Mac Neal, il faut s'assurer que la sous-structure est maintenue quelque part afin d'éviter le mouvement de corps rigide.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5 Quelques exemples de calcul

On présente trois cas pour illustrer la mise en oeuvre de la technique.

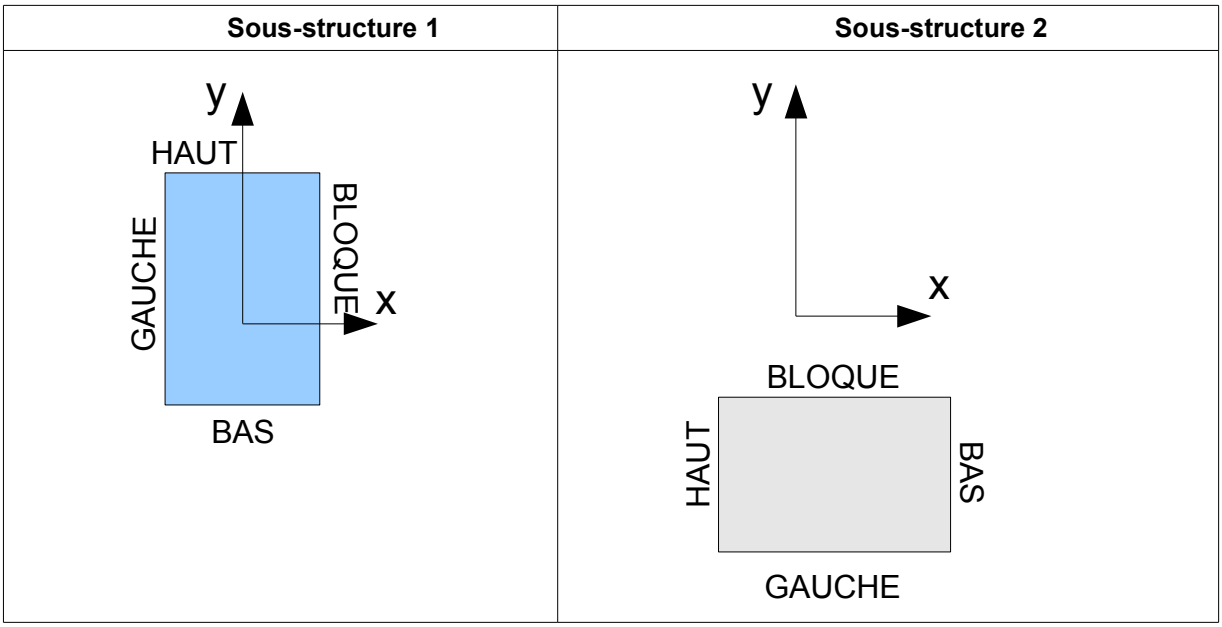
Le premier cas correspond à un calcul modal standard avec utilisation des opérateurs CREA_ELEM_SSD et ASSE_ELEM_SSD.
Le deuxième cas correspond à un calcul de réponse transitoire. On effectue une double projection. La première projection est exécutée lors de la création du macro-élément dynamique et la deuxième projection est réalisée sur le modèle généralisé de la structure assemblée pour le calcul de réponse.
Le troisième cas correspond à un calcul de réponse harmonique sur une structure dissipative. On décrit l'enchaînement des opérateurs élémentaires nécessaires pour la création des macro-éléments dynamiques et l'assemblage des matrices généralisées des sous-structures.

5.1 Un exemple de calcul modal

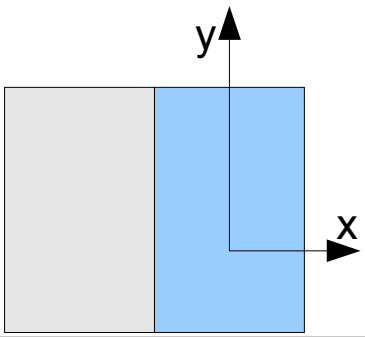
Il s'agit de la modélisation e du cas test SDLS106 [V2.03.106].

Dans ce cas test, on souhaite calculer les modes propres d'une structure composée de deux sous-structures définies dans deux repères différents. L'association de ces deux sous-structures est sensée obtenir une plaque en condition appuyée sur ses bords.

On peut le schématiser comme suit :

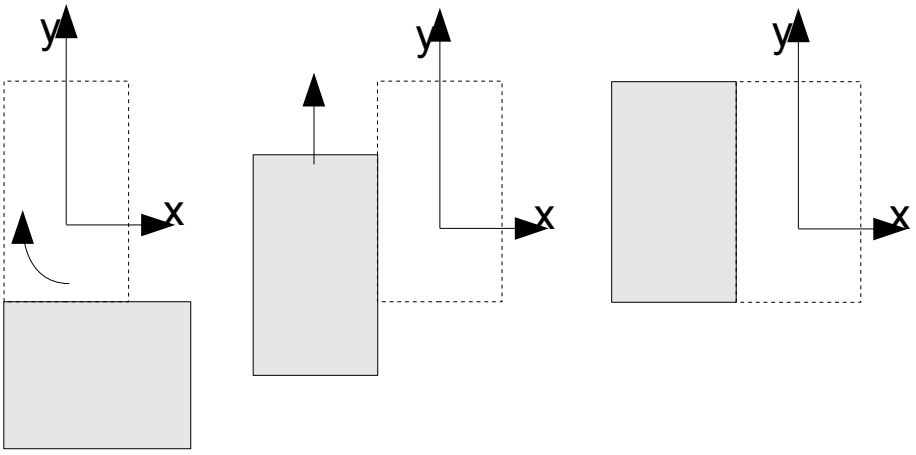


La structure assemblée est une plaque rectangulaire en appui sur ces bords.



Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Pour aboutir à cette structure assemblée, on effectue une rotation et une translation de la sous-structure 2. En effet, les deux sous-structures doivent être exprimées dans le même repère.



Rotation -90 degrés autour de DZ + Translation suivant DY

On va présenter les étapes pour la création du macro-élément dynamique de la sous-structure 1, le macro-élément dynamique de la sous-structure 2 est crée de façon analogue.

Définition des conditions aux limites de la sous-structure 1 :
On souhaite utiliser une base de type Craig-Bampton, il faut donc encastrer les degrés de liberté de l'interface qui sont ici associés au groupe de nœuds GAUCHE. Les autres conditions correspondent aux conditions aux limites de la sous-structure.

```
CHARGE_1=AFFE_CHAR_MECA(  MODELE=MODELE1,DDL_IMPO=(
                _F( GROUP_NO = 'BLOQUE', DX = 0.0, DY = 0.0, DZ = 0.0),
                _F( GROUP_NO = 'GAUCHE', DX = 0.0, DY = 0.0, DZ = 0.0,
                                         DRX = 0.0, DRY = 0.0, DRZ = 0.0),
                _F( GROUP_NO = 'BAS', DX = 0.0, DY = 0.0, DZ = 0.0),
                _F( GROUP_NO = 'HAUT', DX = 0.0, DY = 0.0, DZ = 0.0)))
```

- Création du macro-élément dynamique :

On utilise l'opérateur CREA_ELEM_SSD pour la création du macro-élément dynamique,
On a ici une interface répartie, on souhaite utiliser des **modes d'interface** comme modes de liaison. Sous le mot clé facteur BASE_MODALE, on choisit une base de type RITZ et on saisit le nombre de modes d'interface à utiliser.
Le mot clé facteur CALC_FREQ renseigne sur les nombres de modes normaux à utiliser.

```
MACEL1 = CREA_ELEM_SSD(
                  MODELE = MODELE1,
                  CHARGE = CHARGE_1,
                  CHAM_MATER = CHAMAT1,
                  CARA_ELEM = PARAM1,
                  INTERFACE = _F( NOM = 'GAUCHE',
                                  TYPE = 'CRAIGB',
                                  GROUP_NO = 'GAUCHE'),
                  BASE_MODALE = _F( TYPE = 'RITZ',
                                    NMAX_MODE_INTF = 20,),
                  CALC_FREQ = _F(NMAX_FREQ = 6,),
                 )
```

On crée de la même manière le macro-élément dynamique de la sous-structure 2.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

- Assemblage des différentes sous-structures :

On effectue l'assemblage des sous-structures en utilisant l'opérateur ASSE_ELEM_SSD

Le mot clé facteur RESU_ASSE_SSD permet d'affecter un nom aux concepts résultats (modèle, numérotation des ddl généralisés, matrices généralisées).
Le mot clé facteur SOUS-STRUCT permet de nommer les différentes sous-structures. C'est dans ce mot clé facteur que l'on affecte la rotation et la translation éventuelles du macro-élément (on effectue d'abord la rotation ANGL_NAUT et ensuite la translation TRANS).
Le mot clé LIAISON permet de définir les interfaces entre sous-structures. Dans le cas où on utilise des modes d'interfaces comme modes de liaison, il faut préciser REDUIT sous le mot clé OPTION.
Le groupe de mailles GROUP_MA_MAIT_* pilote le champ de déplacement à l'interface. Si les interfaces ne sont pas compatibles (les nœuds ne sont pas en vis-à-vis), il est conseillé de choisir comme maille maitresse, la maille la plus fine. Sinon, par défaut, les mailles de la sous-structure 1 sont définies comme maitresses.
Le choix ELIMINE sous le mot clé METHODE signifie qu'on numérote les degrés de liberté généralisés en prenant en compte les équations de liaison par la méthode d'élimination des contraintes.

```
ASSE_ELEM_SSD(
                  RESU_ASSE_SSD = _F(
                                MODELE = CO('MODEGE'),
                                NUME_DDL_GENE = CO('NUMEGE'),
                                RIGI_GENE = CO('RIGGEN'),
                                MASS_GENE = CO('MASGEN'),
                              ),
                  SOUS_STRUC = (
                             _F(NOM = 'CARRE1',
                                MACR_ELEM_DYNA = MACEL1,),
                             _F(NOM = 'CARRE2',
                                MACR_ELEM_DYNA = MACEL2,
                                ANGL_NAUT = (-90.,0.,0.),),
                                TRANS = (0.,0.5,0.),
                              ),
                  LIAISON = (
                            _F(SOUS_STRUC_1 = 'CARRE1',
                               INTERFACE_1 = 'GAUCHE',
                               GROUP_MA_MAIT_1 = 'CALCUL',
                               OPTION = 'REDUIT',
                               SOUS_STRUC_2 = 'CARRE2',
                               INTERFACE_2 = 'BLOQUE',),
                           ),
                  VERIF = _F( STOP_ERREUR = 'OUI',
                              PRECISION = 1.E-6,
                              CRITERE = 'RELATIF'),
                  METHODE = 'ELIMINE',
                ),
```

- Calcul des modes propres de la structure assemblée :

On effectue le calcul modal sur les matrices généralisées assemblées. Les déformées modales obtenues sont exprimées en coordonnées généralisées.

```
RESGEN=CALC_MODES( MATR_RIGI=RIGGEN,
                   MATR_MASS=MASGEN,
                   OPTION = 'PLUS_PETITE',
                   CALC_FREQ=_F( NMAX_FREQ=10,),
                   SOLVEUR_MODAL=_F( METHODE='SORENSEN',),
                   VERI_MODE=_F(PREC_SHIFT=5.0E-3,
                                STOP_ERREUR='NON',
                                STURM='NON',
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    SEUIL=9.9999999999999995E-01),
                      STOP_BANDE_VIDE='NON',
                    )
```

- Restitution sur base physique :

Pour l'exploitation des résultats, il nécessaire de les restituer sur la base physique.

```
MODCOT1=REST_SOUS_STRUC(RESU_GENE=RESGEN,
                        TOUT_ORDRE='OUI',
                        SOUS_STRUC='CARRE1',);

MODCOT2=REST_SOUS_STRUC(RESU_GENE=RESGEN,
                        TOUT_ORDRE='OUI',
                        SOUS_STRUC='CARRE2',);
```

- Contrôle de la qualité des résultats :

La qualité des résultats obtenus peut être contrôlée en vérifiant la continuité des déplacements à l'interface. Cela permet de valider le nombre de modes d'interface choisi.
Dans ce cas test, on compare le déplacement du nœud N9 de la sous-structure 1 à celui du nœud N70 de la sous-structure 2. Ces deux nœuds sont en vis-à-vis.

```
T1_DZ=POST_RELEVE_T(ACTION=_F(INTITULE='CARRE1',
                        NOEUD=('N9'),
                        RESULTAT=MODCOT1,
                        NOM_CHAM='DEPL',
                        TOUT_ORDRE='OUI',
                        NOM_CMP='DZ',
                        OPERATION='EXTRACTION',
                        ),);

T2_DZ=POST_RELEVE_T(ACTION=_F(INTITULE='CARRE2',
                        NOEUD=('N70'),
                        RESULTAT=MODCOT2,
                        NOM_CHAM='DEPL',
                        TOUT_ORDRE='OUI',
                        NOM_CMP='DZ',
                        OPERATION='EXTRACTION',
                        ),);

F_DZ=FORMULE(NOM_PARA=('DZ_C1', 'DZ_C2',),
                 VALE='abs (DZ_C1 - DZ_C2)')

TC_DZ=CALC_TABLE(TABLE=T1_DZ,
                 ACTION=(_F(OPERATION='RENOMME',
                            NOM_PARA=('DZ', 'DZ_C1'),),
                         _F(OPERATION='COMB',
                            TABLE=T2_DZ,
                            NOM_PARA=('NUME_ORDRE',),),
                         _F(OPERATION='RENOMME',
                            NOM_PARA=('DZ', 'DZ_C2'),),
                         _F(OPERATION='OPER',
                            NOM_PARA='ECART',
                            FORMULE=F_DZ),
                         ),
                 TITRE='Comparaison DZ ssd 1 / ssd 2')

TEST_TABLE(
     TABLE     = TC_DZ,
     NOM_PARA  = 'ECART',
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
TYPE_TEST = 'SOMM',
VALE      = 0.,
CRITERE   = 'ABSOLU',
PRECISION = 3.E-3,
REFERENCE = 'ANALYTIQUE',)
```

On calcule ici la somme des valeurs absolues des écarts. Le mot clé PRECISION permet de définir l'écart maximal toléré.

On peut également vérifier la nullité du travail des forces de liaison en utilisant l'opérateur CALC_CORR_SSD. Ce calcul est assez consommateur de temps cpu.

```
CALC_ENR=CALC_CORR_SSD(MODELE_GENE=MODEGE,
                       RESU_GENE=RESGEN,
                       SHIFT=1.,
                       UNITE=6)
```

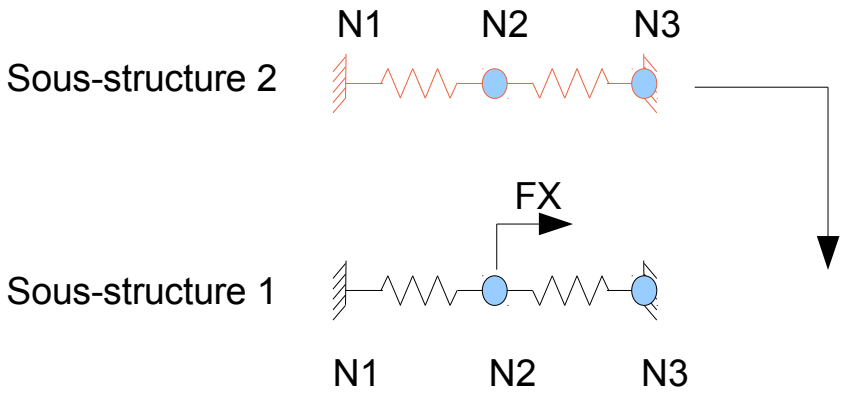
Un récapitulatif des résultats obtenus est déposé, par défaut, dans le fichier des messages. On peut enrichir la base si on constate que le travail des forces de liaison n'est pas négligeable. On peut s'inspirer de la modélisation a du cas test SDLS122 [V2.03.122] pour la mise en oeuvre de cet enrichissement de la base de projection.

5.2 Un exemple de calcul de réponse transitoire

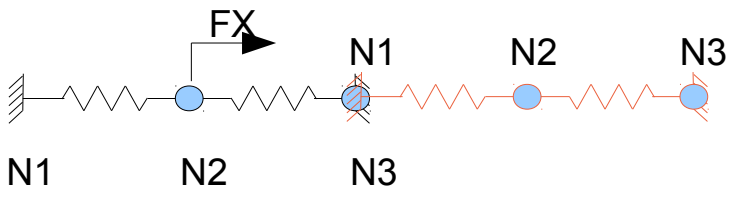
Il s'agit de la modélisation e du cas test SDLD102 [V2.01.102].

Les sous-structures sont composées d'éléments discrets. La première sous-structure est sollicitée par une force ponctuelle.

On utilise le même modèle et le même maillage pour la définition des deux sous-structures.



La structure complète est obtenue en assemblant les deux sous-structures.



Définition des conditions aux limites de la sous-structure :

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

On veut utiliser une base de type **Craig-Bampton classique**, le nœud à l'interface doit être bloqué (nœud $N3$ pour la sous-structure 1 et nœud $N1$ pour la sous structure 2).

```
CHARGE_L=AFFE_CHAR_MECA(MODELE=MODELE,
                        DDL_IMPO=(
                                  _F(NOEUD='N2',
                                     DY=0.,
                                     DZ=0.,
                                     DRX=0.,
                                     DRY=0.,
                                     DRZ=0.,),
                                  _F(NOEUD='N1',
                                     DX=0.,),
                                  _F(NOEUD='N3',
                                     DX=0.,),),) ;
```

- Création des macro-éléments dynamiques :

On applique le chargement FX sur cette sous-structure 1, il faut alors saisir une numérotation pour les degrés de liberté de la sous-structure. Cette numérotation sera utilisée ultérieurement lors de la définition du chargement sur le modèle assemblé.

```
MACEL1 = CREA_ELEM_SSD(
                 MODELE=MODELE,
                 CARA_ELEM=PARAM,
                 CHARGE=CHARGE_L,
                 CHAM_MATER = CHAMAT,
                 NUME_DDL=CO('NUMDDL_L'),
                 INTERFACE=_F(NOM='DROITE',
                              TYPE='CRAIGB',
                              NOEUD='N3',),
                 BASE_MODALE = _F( TYPE = 'CLASSIQUE',),
                 CALC_FREQ = _F(NMAX_FREQ = 1,),
                )
```

Création du macro-élément dynamique de la sous-structure 2 :

```
MACEL2 = CREA_ELEM_SSD(
                 MODELE=MODELE,
                 CARA_ELEM=PARAM,
                 CHARGE=CHARGE_L,
                 CHAM_MATER = CHAMAT,
                 INTERFACE=_F(NOM='GAUCHE',
                              TYPE='CRAIGB',
                              NOEUD='N1',),
                 BASE_MODALE = _F( TYPE = 'CLASSIQUE',),
                 CALC_FREQ = _F(NMAX_FREQ = 1,),
                )
```

- Assemblage des différentes sous-structures :

```
ASSE_ELEM_SSD(
                 RESU_ASSE_SSD = _F(
                              MODELE = CO('MODEGE'),
                              NUME_DDL_GENE = CO('NUMEGE'),
                              RIGI_GENE = CO('RIGGEN'),
                              MASS_GENE = CO('MASGEN'),
                             ),
                       SOUS_STRUC=(_F(NOM='COTE1',
                                      MACR_ELEM_DYNA=MACEL1,),
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    _F(NOM='COTE2',
                                       MACR_ELEM_DYNA=MACEL2,
                                       TRANS=(0.2,0.,0.),),),
                            LIAISON=_F(SOUS_STRUC_1='COTE1',
                                       INTERFACE_1='DROITE',
                                       SOUS_STRUC_2='COTE2',
                                       INTERFACE_2='GAUCHE',),
                            VERIF=_F(STOP_ERREUR='OUI',
                                     PRECISION=1.E-6,
                                     CRITERE='RELATIF',),
                    );
```

- Calcul de la réponse transitoire de la structure assemblée :

Définition et assemblage du vecteur chargement :

```
CHARTRAN=AFFE_CHAR_MECA(MODELE=MODELE,
                        FORCE_NODALE=_F(NOEUD='N2',
                                        FX=1.,),);

V_ELEM=CALC_VECT_ELEM(OPTION='CHAR_MECA',
                      CHARGE=CHARTRAN,);
```

Le chargement doit être exprimé dans la même numérotation que les vecteurs de projection (numérotation des degrés de liberté de la sous-structure 1).

```
V_ASSE=ASSE_VECTEUR(VECT_ELEM=V_ELEM,
                    NUME_DDL=NUMDDL_L,);
```

Assemblage du vecteur chargement sur le modèle généralisé (chargement appliqué sur la sous-structure COTE1) :

```
FGEN=ASSE_VECT_GENE(NUME_DDL_GENE=NUMEGE,
                    CHAR_SOUS_STRUC=_F(SOUS_STRUC='COTE1',
                                       VECT_ASSE=V_ASSE,),);
```

Définition de la base de projection pour le calcul de réponse transitoire (dans la majorité des cas, la base est composée de modes propres du système) :

```
RESGEN=CALC_MODES(MATR_RIGI=RIGGEN,
                  MATR_MASS=MASGEN,
                  OPTION='PLUS_PETITE',
                  CALC_FREQ=_F(NMAX_FREQ=3,),
                  TITRE='CALCUL DES MODES PROPRES DE LA STRUCTURE GLOBALE',
                 );
```

Numérotation des degrés de liberté de la base de calcul :

```
NUMEMODE=NUME_DDL_GENE(BASE=RESGEN,
                       STOCKAGE='DIAG',);
```

Projection des matrices assemblées sur la base de calcul :

```
MASPRO=PROJ_MATR_BASE(BASE=RESGEN,
                      NUME_DDL_GENE=NUMEMODE,
                      MATR_ASSE_GENE=MASGEN,);

RIGPRO=PROJ_MATR_BASE(BASE=RESGEN,
                      NUME_DDL_GENE=NUMEMODE,
                      MATR_ASSE_GENE=RIGGEN,);
```

Projection du chargement sur la base de calcul :

```
VECPRO=PROJ_VECT_BASE(BASE=RESGEN,
                      NUME_DDL_GENE=NUMEMODE,
                      VECT_ASSE_GENE=FGEN,);
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Par défaut, le déplacement et la vitesse sont nuls à l'instant initial. On montre ci-après comment saisir la condition initiale du système.
On crée le champ sur le maillage de la sous-structure avec la même numérotation que celle du macro-élément.

```
CHAMP_0 = CREA_CHAMP( OPERATION='AFFE', TYPE_CHAM='NOEU_DEPL_R',
                      MAILLAGE=MAILLA,
                      NUME_DDL=NUMDDL_L,
                      AFFE=_F(  TOUT = 'OUI',
                           NOM_CMP = ( 'DX', 'DY','DZ','DRX','DRY','DRZ', ),
                           VALE = (0.,0.,0.,0.,0.,0.,))
                          )
```

Projection du champ sur le modèle généralisé assemblé :

```
VGEN_0=ASSE_VECT_GENE(NUME_DDL_GENE=NUMEGE,
                      CHAR_SOUS_STRUC=_F(SOUS_STRUC='COTE1',
                                         VECT_ASSE=CHAMP_0,),);
```

Deuxième projection du champ sur la base de calcul de réponse :

```
DEPINI=PROJ_VECT_BASE( BASE=RESGEN, VECT_ASSE_GENE=VGEN_0,
                       NUME_DDL_GENE=NUMEMODE,
                       TYPE_VECT='DEPL')

VITINI=PROJ_VECT_BASE( BASE=RESGEN, VECT_ASSE_GENE=VGEN_0,
                       NUME_DDL_GENE=NUMEMODE,
                       TYPE_VECT='VITE')
```

Le calcul de réponse transitoire peut s'effectuer en utilisant la commande DYNA_VIBRA. On utilise la technique de résolution par projection sur une base de calcul.

Calcul de la réponse transitoire sur le modèle assemblé projeté :

```
TRANGEN1=DYNA_VIBRA(TYPE_CALCUL='TRAN',BASE_CALCUL='GENE',
                    SCHEMA_TEMPS=_F(SCHEMA='EULER',),
                    MATR_MASS=MASPRO,
                    MATR_RIGI=RIGPRO,
                    INCREMENT=_F(INST_INIT=0.,
                                 INST_FIN=82.,
                                 PAS=0.01,),
                    ETAT_INIT=_F( DEPL = DEPINI,
                                  VITE = VITINI,),
                    ARCHIVAGE=_F(PAS_ARCH=10,),
                    EXCIT=_F(VECT_ASSE=VECPRO,
                             COEF_MULT=1.,),);
```

- Restitution sur base physique :

On peut restituer les résultats directement sur la sous-structure ou sur un maillage de visualisation appelé maillage squelette qu'il faut définir au préalable.

Création du maillage de visualisation :

```
SQUEL=DEFI_SQUELETTE(MODELE_GENE=MODEGE,
                     SOUS_STRUC=(_F(NOM='COTE1',
                                    GROUP_MA='RESSORT'
                                    ),
                                 _F(NOM='COTE2',
                                    GROUP_MA='RESSORT'
                                    ),),);
```

Restitution de la base de calcul sur le maillage de visualisation :

```
MODGLO=REST_SOUS_STRUC(RESU_GENE=RESGEN,
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                        TOUT_ORDRE='OUI',
                        SQUELETTE=SQUEL,);
```

Restitution de la réponse sur le maillage de visualisation (pour une résolution avec double projection, ne pas oublier de spécifier la restitution de la base de calcul sur la sous-structure ou sur le squelette) :

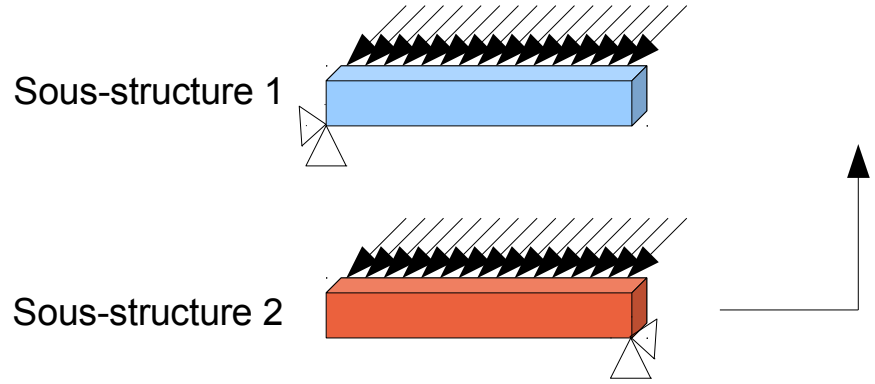
```
TRAN1=REST_GENE_PHYS(RESU_GENE=TRANGEN1,
                     MODE_MECA=MODGLO,
                     LIST_INST=LIST_R,
                     INTERPOL='LIN',
                     TOUT_CHAM='OUI',
                     NOEUD='NO3',);
```

On rappelle donc que pour un calcul de réponse temporelle, on effectue une double projection. Une première projection de la sous-structure sur la base composée de modes normaux et de modes de liaison et une deuxième projection du modèle assemblé réduit sur la base de calcul de réponse.

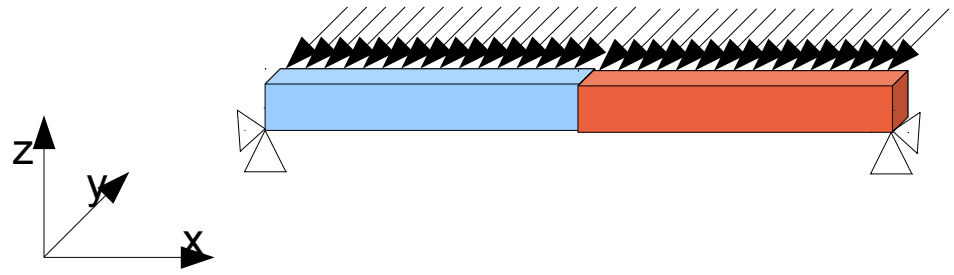
5.3 Un exemple de calcul de réponse harmonique

Dans la majorité des cas, un calcul de réponse harmonique s'effectue sur un modèle dissipatif. Les opérateurs CREA_ELEM_SSD et ASSE_ELEM_SSD ne traitent que les structures conservatives. On détaille dans cet exemple les opérations élémentaires sous-jacentes, nécessaires pour la réalisation d'un calcul de réponse harmonique par sous-structuration dynamique.

Le cas test SHLV301A est utilisé pour illustrer la présentation. Il s'agit d'une barre à section rectangulaire modélisée avec des éléments volumiques quadratiques. On applique une pression sinusoïdale sur une face latérale de la barre.



La structure assemblée :



Définition et affectation du matériau :

```
MAT=DEFI_MATERIAU(  ELAS=_F( RHO = 7.8E03,  NU = 0.3,  E = 2.1E11,
                             AMOR_ALPHA = 1.6E-5, AMOR_BETA = 16.))

CHMAT=AFFE_MATERIAU(  MAILLAGE=MAYA,
                      AFFE=_F( TOUT = 'OUI',
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    MATER = MAT))
```

Description du chargement :

```
PRESSION=AFFE_CHAR_MECA(
                  MODELE=POVOL,
                  PRES_REP=_F(  GROUP_MA = 'PRESSION',
                                PRES = 50.E4))
```

Description des conditions aux limites (On souhaite utiliser une base de projection de type **Mac Neal** : les nœuds de l'interface est libre) :

```
FIXAT1=AFFE_CHAR_MECA(
                   MODELE=POVOL,
                   DDL_IMPO=(
                       _F(  GROUP_NO = 'COND1',  DY = 0.),
                       _F(  GROUP_NO = 'CONDZG',  DZ = 0.),
                       _F(  NOEUD = 'N26',  DX = 0.),
                       _F(  GROUP_NO = 'DROITE', DX = 0.)))
```

- Création des macro-éléments dynamiques :

Création des matrices élémentaires de la sous-structure 1 :

```
K_ELEM1=CALC_MATR_ELEM(  MODELE=POVOL,
                         CHAM_MATER=CHMAT,
                         OPTION='RIGI_MECA',
                         CHARGE=( FIXAT1, )
                           )

M_ELEM1=CALC_MATR_ELEM(  MODELE=POVOL,
                         CHAM_MATER=CHMAT,
                         OPTION='MASS_MECA',
                         CHARGE=( FIXAT1, )
                           )

A_ELEM1=CALC_MATR_ELEM(  MODELE=POVOL,
                         CHAM_MATER=CHMAT,
                         OPTION='AMOR_MECA',
                         RIGI_MECA=K_ELEM1,
                         MASS_MECA=M_ELEM1,
                         CHARGE=( FIXAT1, )
                           )
```

Création du vecteur élémentaire du chargement pour la sous-structure 1 :

```
V_ELEM1=CALC_VECT_ELEM(  OPTION='CHAR_MECA',
                         CHARGE=PRESSION )
```

Assemblage des matrices et vecteur élémentaires :

```
NUM1=NUME_DDL(  MATR_RIGI=K_ELEM1)

K_ASSE1=ASSE_MATRICE(  MATR_ELEM=K_ELEM1,
                       NUME_DDL=NUM1)

M_ASSE1=ASSE_MATRICE(  MATR_ELEM=M_ELEM1,
                       NUME_DDL=NUM1)

A_ASSE1=ASSE_MATRICE(  MATR_ELEM=A_ELEM1,
                       NUME_DDL=NUM1)

V_ASSE1=ASSE_VECTEUR(  VECT_ELEM=V_ELEM1,
                       NUME_DDL=NUM1)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Calcul des modes normaux :
```
MODES1=CALC_MODES( MATR_RIGI=K_ASSE1,
                   MATR_MASS=M_ASSE1,
                   CALC_FREQ=_F( NMAX_FREQ = 20) )
```

Définition de l'interface pour une base de type Mac Neal :
```
INTERF1=DEFI_INTERF_DYNA(  NUME_DDL=NUM1,
                           INTERFACE=_F( NOM = 'DROITE',
                                         TYPE = 'MNEAL',
                                         GROUP_NO = 'DROITE'),
                           INFO=2)
```

Définition de la base de projection :
```
BAMO1=DEFI_BASE_MODALE(  CLASSIQUE=_F( INTERF_DYNA = INTERF1,
                                       MODE_MECA = MODES1,
                                       NMAX_MODE = 20),
                         INFO=2)
```

Création du macro-élément dynamique :
On précise que l'on souhaite prendre en compte l'amortissement en saisissant la matrice d'amortissement à projeter.
```
MACRO1=MACR_ELEM_DYNA(  BASE_MODALE=BAMO1,
                        MATR_AMOR=A_ASSE1 )
```

On construit de la même manière le macro-élément dynamique de la sous-structure 2.

- Assemblage des différentes sous-structures :

Définition du modèle généralisé : on affecte un nom à chaque sous-structure et on précise les interfaces entre les différentes sous-structures
```
MODEGE=DEFI_MODELE_GENE( SOUS_STRUC=( _F( NOM = 'COTE1',
                                          MACR_ELEM_DYNA = MACRO1,
                                          TRANS = (0., 0., 0.,),
                                          ANGL_NAUT = (0., 0., 0.,),),
                                      _F( NOM = 'COTE2',
                                          TRANS = (1., 0., 0.,),
                                          ANGL_NAUT = (0., 0., 0.,),
                                          MACR_ELEM_DYNA = MACRO2)),
                                 LIAISON=_F( SOUS_STRUC_1 = 'COTE1',
                                          SOUS_STRUC_2 = 'COTE2',
                                          INTERFACE_1 = 'DROITE',
                                          INTERFACE_2 = 'GAUCHE'))
```

Définition d'une numérotation des degrés de liberté du système généralisé assemblé :
```
NUMEGE=NUME_DDL_GENE(  MODELE_GENE=MODEGE)
```

Assemblage des matrices généralisées des différentes sous-structures :
```
RIGGEN=ASSE_MATR_GENE(  NUME_DDL_GENE=NUMEGE,
                        OPTION='RIGI_GENE' )

MASGEN=ASSE_MATR_GENE(  NUME_DDL_GENE=NUMEGE,
                        OPTION='MASS_GENE' )

AMOGEN=ASSE_MATR_GENE(  NUME_DDL_GENE=NUMEGE,
                        OPTION='AMOR_GENE' )
```

Assemblage du vecteur chargement généralisé :
```
VECTGEN=ASSE_VECT_GENE(  NUME_DDL_GENE=NUMEGE,
                         CHAR_SOUS_STRUC=(
                                _F(  SOUS_STRUC = 'COTE1',
                                     VECT_ASSE = V_ASSE1),
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

```
                                    _F(  SOUS_STRUC = 'COTE2',
                                         VECT_ASSE = V_ASSE2)))
```

- Calcul de la réponse harmonique de la structure assemblée :

```
DYNGEN=DYNA_VIBRA(TYPE_CALCUL='HARM',BASE_CALCUL='GENE',
                  MATR_MASS=MASGEN,
                  MATR_RIGI=RIGGEN,
                  MATR_AMOR=AMOGEN,
                  SOLVEUR=_F( METHODE='LDLT', ),
                  FREQ=F,
                  TOUT_CHAM='OUI',
                  EXCIT=_F( VECT_ASSE = VECTGEN,
                            COEF_MULT = 1.))
```

- Restitution sur base physique :
  On effectue ici une restitution sur la sous-structure 1 (COTE1) :

```
DYNHAR1=REST_SOUS_STRUC(  RESU_GENE=DYNGEN,
                          SOUS_STRUC='COTE1')
```

On rappelle donc que pour un calcul de réponse harmonique, comme pour le calcul modal, on effectue directement le calcul sur le modèle généralisé de la structure assemblée sans faire une deuxième projection sur une base de calcul.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)