# Mise en œuvre de l'approche « grands glissements avec X-FEM »

**Résumé :**

Ce document décrit la mise en œuvre informatique de l'approche « grands glissements avec X-FEM ». Il s'agit d'enrichir la modélisation des structures fissurées par la X-FEM en lui donnant la possibilité de prendre en compte le contact au niveau des interfaces, sous grands déplacements, mais encore sous l'hypothèse des petites déformations. La formulation scientifique de cette approche est présentée dans le manuel [R5.03.53] et on recommande vivement sa lecture au préalable. Dans ce document on présente principalement les routines informatiques qui contribuent à l'implémentation numérique de cette approche (nouvelles routines, anciennes routines qui ont été modifiées ou juste anciennes routines qui n'ont pas été modifiées mais qui contribuent de façon générale à la modélisation X-FEM). On donne également les arbres d'appel de ces routines, un outil très important lorsqu'on se lance dans le développement de Code_Aster.

L'implémentation numérique de cette approche dans Code_Aster concerne les cas 2D et 3D.



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

# 1 Introduction

Ce document décrit l'implémentation numérique dans Code_Aster de l'approche « Grands glissements avec X-FEM ». On présente notamment les routines Fortran qui construisent les opérateurs de Code_Aster nécessaires à la construction du modèle X-FEM, ainsi que l'arborescence de ces opérateurs.

Le document est rédigé à l'intention des personnes qui vont continuer ce travail (stagiaires, doctorants, ou post-doctorants) et en faire la maintenance. Cette lecture est à compléter par les commentaires qui sont insérés un peu partout dans les routines Fortran (nouvelles ou juste modifiées).
Après cette introduction, les six chapitres qui suivent respectent l'ordre des opérateurs à définir dans un fichier de commande de Code_Aster afin de réaliser une analyse en utilisant l'approche « Grands glissements avec X-FEM ». Dans chaque chapitre correspondant à un opérateur, on présente également son arbre d'appel où seulement les routines concernant cette approche sont figurées.
On clôture avec la présentation des perspectives de développement pour rendre cette nouvelle approche la plus générale possible.

**Remarques importantes**:

1) L'implémentation numérique de cette approche concerne les cas 2D (linéaire et quadratique) et 3D (linéaire). Le traitement des éléments de fonds de fissure et multi-fissurés est possible. Les termes de frottement ont été implémentés dans tous les cas sauf pour les éléments multi-fissurés.
2) Cette approche a fait l'objet d'une évolution de Code_Aster pilotée par les fiches (10708, 10751, 10752 et 10753, 11957,11958, 12608,10556, 13178, 13179 et 15559). Elle est opérationnelle dans le code depuis la restitution 9.1.14 de la version STA9.1 et a été mise à jour depuis dans la version STA10.0.



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

# 2 Introduction des fissures

L'introduction des fissures X-FEM dans un modèle de Code_Aster se fait à l'aide de l'opérateur DEFI_FISS_XFEM [U4.82.08]. Cet opérateur n'a pas subi de modifications/ajouts suite à l'implémentation de l'approche *grands glissements avec X-FEM*. Toutefois, il est important de donner ici de courtes descriptions pour ses routines qui construisent en fait la structure de données FISS_XFEM [D4.10.02]. Le développeur pourra ainsi s'y retrouver si jamais il veut modifier ou enrichir ces structures de données (SD). Tout au long de ce document, le lecteur rencontrera l'appellation "routine chapeau". Il s'agit d'une routine qui sert seulement à lancer des opérations de calcul en appelant d'autres routines.

Ci-dessous, on présente un exemple d'introduction d'une fissure, par des fonctions level-sets analytiques, pour une analyse X-FEM. Le catalogue de la commande (defi_fiss_xfem.capy) évolue assez fréquemment, donc on recommande sa consultation avant la rédaction du fichier de commande.

```
LSN=FORMULE(NOM_PARA=('X','Y'),VALE='Y-2.5');
LST=FORMULE(NOM_PARA=('X','Y'),VALE='-X-10.');

FISS1=DEFI_FISS_XFEM(MODELE=MODELEIN,
                     DEFI_FISS=_F(FONC_LT=LST,FONC_LN=LSN,),
                     INFO=1,);
```

La description des routines Fortran est donnée ci-dessous et l'arbre d'appel est illustré sur la Figure 1.

OP0041 - routine de lancement de l'opérateur qui réalise principalement la lecture des données introduites dans le fichier de commande et le lancement des différentes routines qui suivent.

XLORIE – routine pour la lecture de données concernant le fond de fissure. Les paramètres introduits par l'utilisateur sous les mots clés 'RAYON_ENRI' et 'NB_COUCHES' sont lus ici afin de remplir la SD FISS//'.CARAFOND' avec le rayon d'enrichissement autour de la tête de fissure et le nombre de couches d'éléments à enrichir.

XLENRI – réalise la lecture de la liste de mailles potentiellement enrichies, liste fournie par l'utilisateur, et crée les SD: FISS//'.GROUP_MA_ENRI' et FISS//'.GROUP_NO_ENRI'.

XINILS – routine chapeau pour lancer le calcul des level-sets. Si les level-sets sont introduites par des fonctions analytiques, le calcul se fait directement dans cette routine et on remplit les SD: FISS//'.LNNO' et FISS//'.LTNO'

XINLSJ – routine pour la récupération des levels set jonction. Elle est appelée dans le cas où l'utilisateur a défini l'opérande JONCTION dans DEFI_FISS_XFEM . Dans ce cas on récupère localement les levels set normales et tangentes des fissures sur lesquelles FISS se branche dans un cham_no . Ces levels set sont ensuite utilisées dans la routine XSTANO pour le calcul du statut d'enrichissement des nœuds. Pour conserver l'information globale afin d'alimenter l'opérateur MODI_MODELE_XFEM , on crée ici les champs FISS//'.JONFISS' et FISS//'.JONCOEF' qui contiennent la liste des fissures sur lesquelles FISS se branche et les coefficients associés.

XLS3D – routine pour le calcul des level-sets si la fissure est introduite par des mailles surfaciques (cas 3D).

XLS2D - routine pour le calcul des level-sets si la fissure est modélisée par des mailles linéiques (cas 2D).

XAJULS – routine pour le réajustement des level-sets. Les valeurs des LS sont modifiées si elles sont trop proches de zéro pour éviter les erreurs d'intégration numérique dues au mauvais conditionnement de la matrice de rigidité.



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

XGRALS – routine pour le lancement du calcul des gradients des level-sets. On sort avec les SD: FISS//'.GRLNNO' et FISS//'.GRLTNO'.

TE0024 – réalise le calcul élémentaire des gradients des level-sets (option 'GRAD_NEUT_R').

XENRCH – lancement du calcul de l'enrichissement et des points du fond de fissure.

XMAFIS – routine pour trouver les mailles où la level-set normale change de signe. On sort avec la liste des mailles fissurées.

XSTANO – détermine le statut (enrichissement) des nœuds (HEAVISIDE, CRACK TIP ou HEAVISIDE et CRACK TIP). On remplit la SD FISS//'.STNO' .

XSTAMA – calcul du statut des mailles et actualisation éventuelle des statuts des nœuds (si enrichissement par couche).

XSTAM1 – calcul du statut des mailles en fonction du statut des nœuds.

XPTFON – recherche les points des fonds de fissure.

XORIFF – orientation des point des fonds de fissure.

XLMAIL – création et remplissage des SD: FISS//'.MAILFISS .INDIC', FISS//'.MAILFISS .HEAV', FISS//'.MAILFISS .CTIP', FISS//'.MAILFISS .HECT', FISS//'.FONDFISS', FISS//'.FONDMULT' .

XBASLO - création de la SD FISS//'.BASLOC' qui contient la base locale aux points du fond de fissure associé.



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

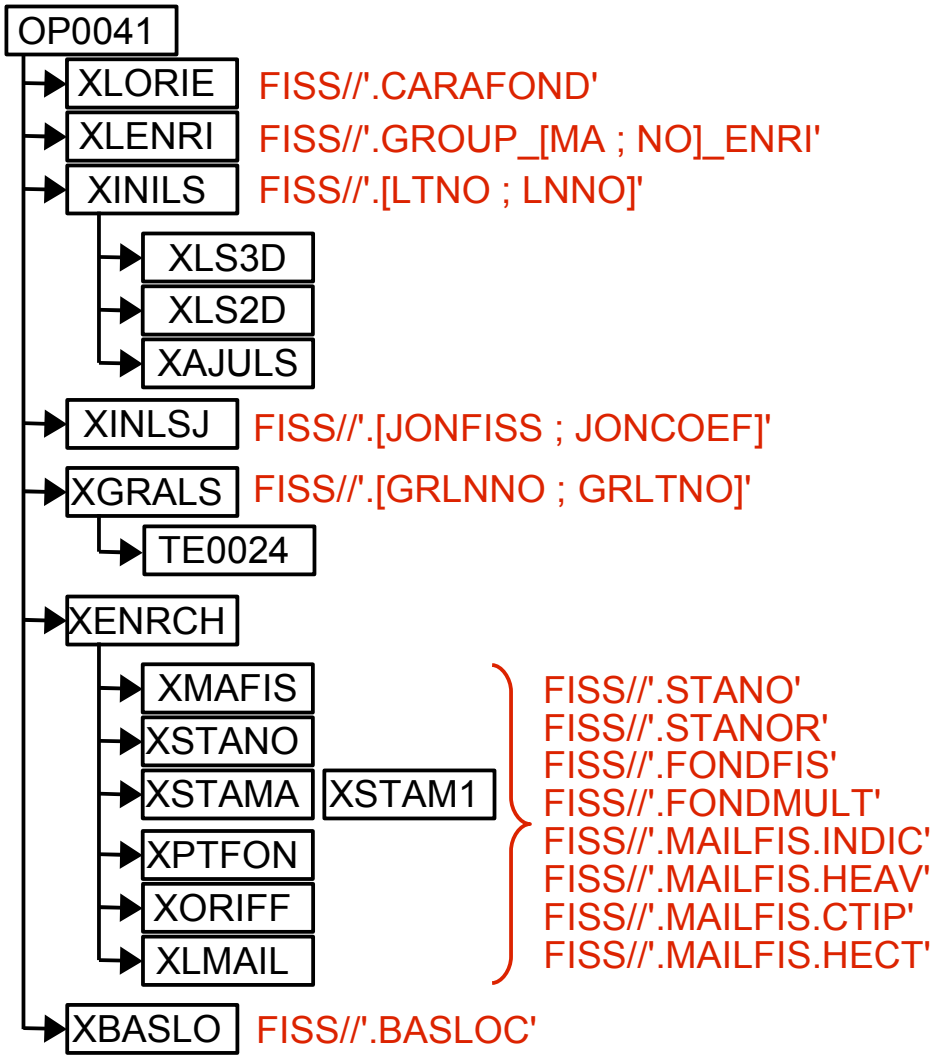


**Figure 1. L'arbre d'appel de l'opérateur `DEFI_FISS_XFEM`**



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

# 3 Enrichissement du modèle

Il s'agit de l'opérateur MODI_MODELE_XFEM [U4.41.11], développé spécialement pour l'analyse par X-FEM [R7.02.12] avec le but de modifier certains éléments finis classiques afin de les transformer en éléments finis enrichis. Cet opérateur a notamment pour but de préparer le modèle à un calcul X-FEM dans lequel l'utilisateur aurait introduit plusieurs fissures. Les fissures peuvent éventuellement découper la même maille voir se brancher l'une sur l'autre.. On fait appel à la routine XCONNO qui réalise la concaténation des SD X-FEM définies par fissure, le principe étant d'avoir des SD définies sur le modèle, pour l'ensemble des fissures. Ces SD concaténées servent ensuite au découpage des éléments en sous éléments (XTOPOI) et à déterminer les facettes de contact (XTOPOC).

Pour l'approche *grands glissements avec X-FEM*, les modifications apportées ici concernent la topologie des facettes de contact. Plus exactement, le contenu de l'objet MODELE//'.TOPOFAC.PI' (voir [D4.10.02]) a été changé: il ne stocke plus les coordonnées réelles des points d'intersection, mais les coordonnées de référence dans l'élément parent. On ajoute en même temps deux nouveaux objets: MODELE//'.TOPOFAC.GE' et MODELE//'.TOPOFAC.GM' qui vont stocker les coordonnées réelles des points d'intersection pour les facettes de contact, respectivement *esclave* et *maître*. On a donc dupliqué les facettes de contact par rapport à l'implémentation HPP. Par convention on désigne comme facette esclave celle formée par des points d'intersection attachés au côté des LSN négatifs (LSN<0 aux nœuds enrichis), tandis que la facette maître est formée par les points d'intersection attachés au côté des LSN positifs (LSN>0 aux nœuds enrichis). On observe que l'utilisateur a la possibilité de jouer avec le statut des facettes en changeant le signe de la level-set LSN dans le fichier de commande. Si la fissure n'est pas introduite par une fonction analytique mais par des mailles surfaciques, le changement du statut esclave/maître est possible par le changement d'ordre des nœuds qui composent ces mailles. Cette dernière opération peut s'avérer onéreuse et donc l'introduction d'un nouveau mot clé est envisageable pour permettre à l'utilisateur le choix facile des côtés positifs et négatifs, directement dans le fichier de commande. Pour plus de détails sur le calcul des level-sets pour les deux modalités d'introduction d'une fissure, voir le chapitre 2.2 de la documentation de référence de la X-FEM [R7.02.12].

Plusieurs SD qui sont mentionnées ici sont attachées au MODELE et présentées en détail dans le manuel [D4.06.02] « Structures de Données ligrel et modele ».

Voici un exemple d'appel de l'opérateur X-FEM dans le fichier de commande:

```
MODELEK=MODI_MODELE_XFEM(MODELE_IN=MODELEIN,
                         FISSURE=(FISS1,FISS2),
                         CONTACT='P1P1',
                          INFO=1,);
```

La description des routines Fortran pour l'opérateur MODI_MODELE_XFEM est donnée ci-dessous et l'arbre d'appel est présenté sur la Figure 2.

OP0113 – est la routine de base de l'opérateur. On crée ici trois SD: MODELE//'.XFEM_CONT' (stocke l'information sur l'existence ou non du contact), MODELE//'.NFIS' (stocke le nombre des fissures dans le modèle) et MODELE//'.FISS' (stocke les noms des fissures).

XTYELE– routine qui détermine le type d'élément X-FEM, suivant le type d'enrichissement et suivant le type de contact.

XTYHEA – routine qui compte le nombre de ddl Heavisides actifs dans la maille, dans le cas du Multi-Heaviside.

XMOLIG – routine pour la modification de la LIGREL X-FEM, suivant le type d'enrichissement.

XCODEC – routine chapeau pour le lancement du calcul du découpage en sous-tétraèdres, des facettes de contact et vérification des critères de conditionnement.



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

XCONNO – routine qui réalise la concaténation des champs nœuds pour passer à la multi-fissuration. Les nouveaux champs sont attachés au modèle, ils sont de type champ elno.

XFISNO – routine qui crée les sd MODELE'//.FISSNO' et MODELE'//.HEAVNO'. Ce sont des champs ELNO utiles dans le cas des éléments qui voient plusieurs fissures. La première SD retourne le numéro de fissure vu localement par l'élément associé au numéro de ddl Heaviside du nœud. La deuxième réalise l'opération inverse et retourne le numéro de ddl Heaviside à associer au numéro local de fissure au nœud.

XTOPOI – lancement du calcul de découpage des éléments fissurés en sous-tétraèdres.

TE0514 – calcul élémentaire des partitions des éléments fissurés.
On sort avec les SD :
MODELE//'TOPOSE.PIN',MODELE//'TOPOSE.CNS',MODELE//'TOPOSE.HEA', MODELE//'TOPOSE.LON' et MODELE//'TOPOSE.CRI'. Lorsque la maille voit plusieurs fissures, elle est découpée séquentiellement par chacune des fissures. A cet effet, les différentes opérations de ce TE sont encapsulées dans une boucle sur les fissures.

XDECOU – routine appelée depuis TE0514 pour trouver les points d'intersection entre les arêtes et le plan de fissure. Notons que c'est la routine XDECQU qui est appelée dans le cas quadratique.

XDECOV - routine appelée depuis TE0514 pour découper un tétraèdre en sous-tétraèdres. Notons que c'est la routine XDECQV qui est appelée dans le cas quadratique.

XTOPOC – lancement du calcul pour le sous-découpage de la surface de contact et la création des SD MODELE//'.TOPOFAC.PI', MODELE//'.TOPOFAC.AI', MODELE//'.TOPOFAC.CF', MODELE//'.TOPOFAC.LO', MODELE//'.TOPOFAC.BA' et MODELE//'.TOPOFAC.OE' Notons que la SD MODELE//'.TOPOFAC.PI' contient les coordonnées des points d'intersection dans l'élément parent: elle est très utile pour retrouver la valeur des fonctions de forme des points d'intersection. La SD MODELE//'.TOPOFAC.OE' analogue contient les coordonnées des points d'intersection dans le repère global. Elle est utile pour alimenter le TE0519 lors de la réactualisation géométrique. Dans le cas de mailles multi-fisurées, toutes ces SD sont dupliquées autant de fois que l'élément voit de fissures. La SD MODELE//'.TOPOFAC.HE' est alors ajoutée pour stocker, par facette de contact, la valeur des fonctions Heaviside de chaque fissure de part et d'autre de la facette.

TE0510 – calcul élémentaire pour la topologie des facettes de contact. Notons que les facettes sont aussi utiles dans le cas des éléments sans contact, afin d'imposer des efforts de pression au niveau des lèvres de la fissure. Dans le cas des éléments contenant une jonction entre fissure, certaines facettes devraient être redécoupées de part et d'autre de la jonction. Ce travail n'a pas été implémenté et les facettes ne sont pas prises en compte.

XCFACE – routine appelée par TE0510 pour trouver les points d'intersection entre les arêtes et le plan de la fissure et déterminer le découpage des facettes. Notons que c'est la routine XCFAQ2 qui est appelée dans le cas quadratique.

XCFACF – routine pour trouver les points d'intersection entre le fond de fissure et les faces pour les éléments en fond de fissure.

XCFACJ – routine pour trouver les points d'intersection entre la jonction de fissures et les faces pour les éléments multi-fisurés.

XAINT2 – routine qui modifie la SD MODELE://'TOPOFAC.AI' dans le cas de mailles multi-fissurées. Il faut s'assurer que les facettes connectées à l'arête coupée possèdent bien les mêmes valeurs de fonction Heaviside. Dans le cas contraire l'arête est éliminée pour éviter tout conflit LBB.



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

XSTAN2 – routine qui modifie la SD MODELE://'.STNO' afin d'annuler les degrés de liberté HEAVISIDE pour des raisons de conditionnement de matrice et pour éviter des pivots nuls dans la matrice de rigidité.

XCRVOL – routine appelée par XSTAN2 aidant pour le calcul du critère volumique en lien avec le conditionnement des éléments X-FEM.

XORIPE – routine pour orienter les sous-éléments de peau des éléments X-FEM (en 2D les mailles de peau sont des segments de bord).

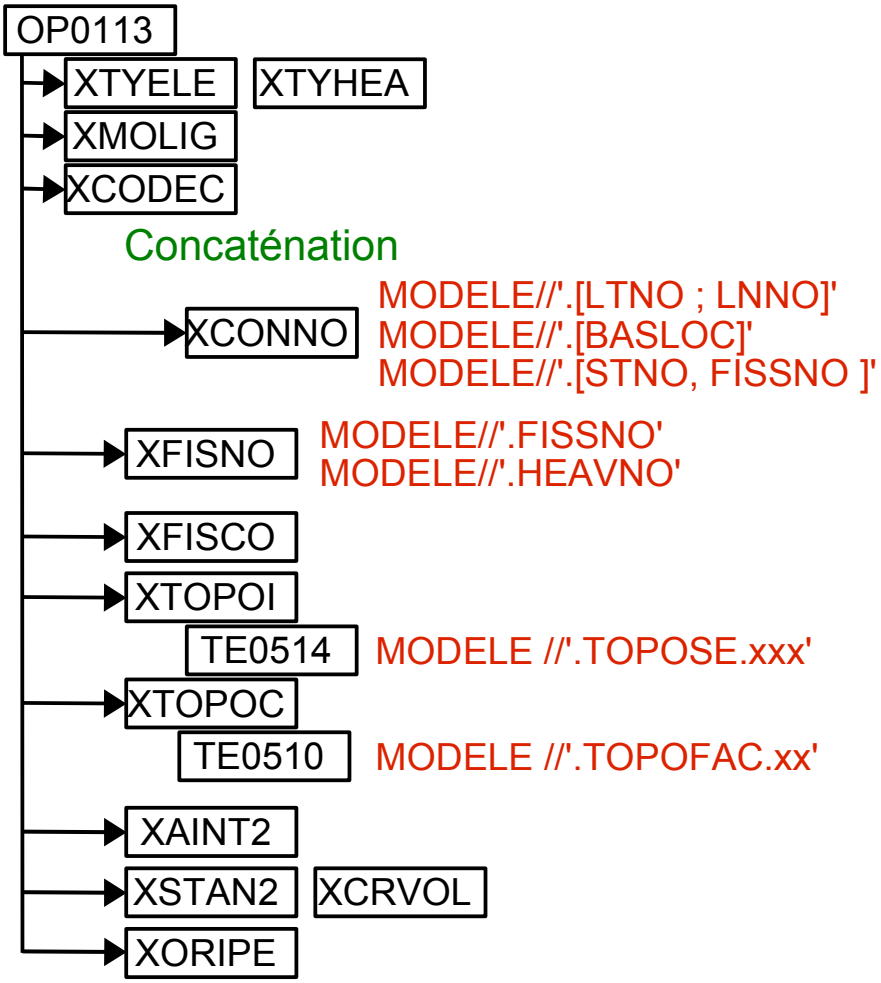

Figure 2. L'arbre d'appel de l'opérateur **MODI_MODELE_XFEM**


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

# 4 Affectation du contact

Au moment de la rédaction de ce document, l'affectation du contact sur les lèvres des fissures X-FEM se fait par l'intermédiaire de l'opérateur DEFI_CONTACT, sous le mot clé ZONE.
Le plus important pour l'implémentation X-FEM est le fait que les SD dédiées au contact, pour toute analyse avec Code_Aster, ont été unifiées à partir de la version 9.0.22 et que c'est ici qu'elles sont enrichies pour l'approche *grands glissements avec X-FEM*.
Lors de la restitution 9.1.6 cet opérateur a été enrichi avec les routines qui concernent l'algorithme LBB: XLAGSP, XLAGSC, XSELLA, XRELL1, etc. (voir [R7.02.12] pour plus de détails sur l'algorithme LBB), routines qui se trouvaient avant dans l'opérateur MODI_MODELE_XFEM.
La Figure 3 présente l'arbre d'appel actuel (en tenant compte aussi de la surcharge "grands glissements"). En effet, c'est ici que la première routine dédiée à cette nouvelle approche apparaît – XMACON. On crée alors la SD CHAR//'.CONTACT.MAESCL' contenant un certain nombre d'informations relatives aux mailles de contact X-FEM (voir [D4.06.14] pour la description de cette SD).
On traite le contact par la méthode continue et on suit exactement la même démarche que celle de la routine MMACON pour le cas classique, sauf que dans la SD ci-dessus mentionnée on a moins de positions occupées. Toujours dans cette routine, XMACON, on dimensionne l'objet CHAR//'.CONTACT.TABFIN' qui sera rempli plus tard, lors de l'appariement réalisé par la routine XAPPAR (voir l'opérateur STAT_NON_LINE). Enfin, deux nouvelles routines, XNEUVI et XBARVI, permettent la création des SD FISSi//'.CNCTE'. Chacune de ces SD stocke les informations utiles, sur les groupes d'arêtes vitales connectées entre elles, à l'amélioration de l'intégration des contributions de contact (voir [R5.03.53]). Plus de détails sur toutes les SD relatives au contact avec X-FEM sont aussi fournis dans la description des routines Fortran (voir CALICO ci-dessous).
Ci-dessous on donne un exemple d'introduction du contact dans le fichier de commande pour une analyse X-FEM avec l'approche grands glissements. Attention, le catalogue de la commande (defi_contact.capy) a été enrichi pour permettre la définition du nombre maximal d'itérations sur la boucle géométrique (ITER_GEOM_MAXI) ainsi que le niveau de tolérance pour la projection des points de contact sur la surface maître (TOLE_PROJ_EXT), qui a la même fonction que dans le cas classique.

```
CTXFEM = DEFI_CONTACT(MODELE=MODELEK,
                      FORMULATION='XFEM',
                      FROTTEMENT='COULOMB',
                      ITER_FROT_MAXI=10,
                      ITER_CONT_MAXI=10,
                      ITER_GEOM_MAXI=10,
                      ZONE=(_F(INTEGRATION='NOEUD',
                               COULOMB=0.5,
                               COEF_REGU_FROT=100,
                               COEF_REGU_CONT=100.,
                               TOLE_PROJ_EXT =0.0,
                               CONTACT_INIT='OUI',
                               FISS_MAIT = FISS1,
                               ALGO_LAGR='VERSION2',),
                            _F(INTEGRATION='NOEUD',
                               COULOMB=0.5,
                               COEF_REGU_FROT=100,
                               COEF_REGU_CONT=100.,
                               TOLE_PROJ_EXT =0.0,
                               CONTACT_INIT='OUI',
                               FISS_MAIT = FISS2,
                               ALGO_LAGR='VERSION2',), ),)
```

Description des routines Fortran:

OP0030 – la routine principale de l'opérateur DEFI_CONTACT. On appelle d'ici CALICO.



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

CALICO – la routine qui lance la création et le remplissage des SD liées aux contact pour la méthode continue. On recommande particulièrement la lecture des commentaires au début de cette routine parce qu'on trouve ici la description détaillée de tous les objets (la documentation officielle [D4.06.14] n'étant pas encore mise à jour).

SYMECO - routine pour la détermination du nombre de zones de contact pour l'appariement symétrique et le remplissage de la SD associée CHAR//'.CONTACT.SYMECO'. Cette SD va être commune pour toutes les formulations (continue, discrète, X-FEM).

CARACO – routine chapeau pour récupérer les caractéristiques de contact introduites par l'utilisateur. On crée ici les SD CHAR//'.CONTACT.FORMCO' et CHAR//'.CONTACT.METHCO', la dernière étant aussi remplie ici.

CAZOCO – routine de lecture des caractéristiques de contact pour chaque zone. On remplit ici CHAR//'.CONTACT.FORMCO'.

CMFXSD – routine chapeau pour le lancement de la création des SD de définition du contact pour les trois formulations (continue, discrète, X-FEM).

CARACX – création des SD de définition du contact pour la formulation X-FEM.

CAZOCX – routine spécialisée pour la formulation X-FEM (de la même manière existe CAZOCD pour la formulation discrète et CAZOCC pour la formulation continue). Ici sont remplies, pour le cas X-FEM, les SD CHAR//'.CONTACT.ECPDON', CHAR//'.CONTACT.CARACF' et , CHAR//'.CONTACT.TOLECO' ayant la même structure que dans le cas classique, ainsi que , CHAR//'.CONTACT.MODELX' qui stocke le nom du modèle X-FEM utilisé par le contact. On ajoute aussi l'information d'existence du contact glissière dans CHAR//'.CONTACT.METHCO'.

LIMACO – routine chapeau pour la lecture des mailles de contact et la création des SD qui les concernent. On s'intéresse ici seulement aux appels des routines pour la formulation continue dans le cas X-FEM.

XMACON – nouvelle routine (similaire à MMACON pour le cas classique) pour la lecture des mailles de contact X-FEM et la création des SD CHAR//'.CONTACT.MAESCL' et CHAR//'.CONTACT.TABFIN'. La première est remplie ici tandis que la deuxième est juste dimensionnée.

XMELIN - nouvelle routine appelée par XMACON (semblable à MMELIN de la méthode continue) qui retourne le nombre de points d'intégration en fonction du type d'élément de contact et du schéma d'intégration de contact.

LIMACX – routine pour la lecture des fissures en contact. Ici sont créés et remplis deux nouveaux objets spécifiques pour la X-FEM: CHAR//'.CONTACT.XFIMAI' (contient, pour chaque zone de contact, le nom de la SD FISS_XFEM) et CHAR//'.CONTACT.NDIMCO' (la dimension du problème).

XCONTA – routine de préparation des données relatives au contact, concernant spécifiquement les relations linéaires à imposer pour satisfaire la condition LBB. On crée ici deux SD: CHAR//'.CONTACT.XNRELL' (stocke les noms des quatre objets nécessaires pour les relations linéaires créées par XRELL1 ou XRELL2) et CHAR//'.CONTACT.XNBASC' (stocke le nom du champ nodal pour la base covariante).

XDEFCO – routine chapeau pour le lancement du choix de l'espace des Lagrangiens de contact.

XLAGSP – préparation des objets de travail pour l'implémentation de l'algorithme LBB. On remplit la base covariante (seulement pour les points de contact correspondant aux éléments complètement coupés) SD FISSi//'.BASCO' .



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

XLAGSC – routine de lancement effectif pour la sélection des nœuds et l'écriture des relations d'égalité et linéaires pour l'algorithme LBB.

XSELLA – sélection des nœuds porteurs de *ddl* de contact qui seront impliqués dans les relations pour la LBB.

XRELL1 – création des listes de relations à imposer pour le respect de la condition LBB dans le cas d'application du premier algorithme: FISSi//'.LISEQ', FISSi//'.LISRL' et FISSi//'.LISCO'.

XRELL2 – fait la même chose que XRELL1 mais dans le cas de l'application du deuxième algorithme. Rappelons que pour la formulation aux nœuds sommets, il est préférable d'utiliser ce deuxième algorithme.

XNEUVI – appelé par XRELL2, crée pour chaque fissure les SD FISSi//'.CONNECTANT' et FISSi//'.CONNECTES'. La première SD stocke les nœuds ayant un score strictement plus grand que 1 (le score correspond au nombre d'arêtes vitales connectées au nœud à la fin de l'algorithme), chacun de ces nœuds définissant alors un groupe d'arêtes vitales, où le nombre d'arêtes vitales est égal au score du nœud 'vital'. La deuxième SD stocke les nœuds de chaque groupe qui, appariés à leur nœud vital, forment les couples de nœuds définissant les arêtes du groupe. Ces 2 SD sont temporaires. Elles sont utilisées puis détruites dans XBVARVI.

XLAG2C – routine permettant d'écrire pour chaque fissure la SD FISSi//'.LISEQ_LAGR' correspondant à la SD FISSi//'.CONTACT.LISEQ'. Cette SD stocke dans le cas de mailles multi-fissurées, les couples de numéros de Lagrangien qu'il faut associer lors de l'imposition des relations d'égalités.

XBARVI – appelé par XCONTA, c'est ici que l'on renseigne les arêtes vitales et les groupes d'arêtes vitales. On remplit la 5ème composante de la SD MODELE//'.TOPOFAC.AI', qui donne l'information vitale ou pas de l'arête intersectée, en se servant de la liste des nœuds formant les arêtes vitales. Et on crée pour chaque fissure la SD à 4 composantes FISSi//'.CNCTE', qui renseigne sur les groupes d'arêtes vitales. Elle stocke pour chaque arête appartenant à un groupe, son numéro de groupe, son numéro d'arête dans le groupe, son numéro de maille, son numéro d'arête locale dans la maille. Pour faire cela on utilise les SD FISSi//'.CONNECTANT' et FISSi//'.CONNECTES, que l'on détruit à la fin de la routine.



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

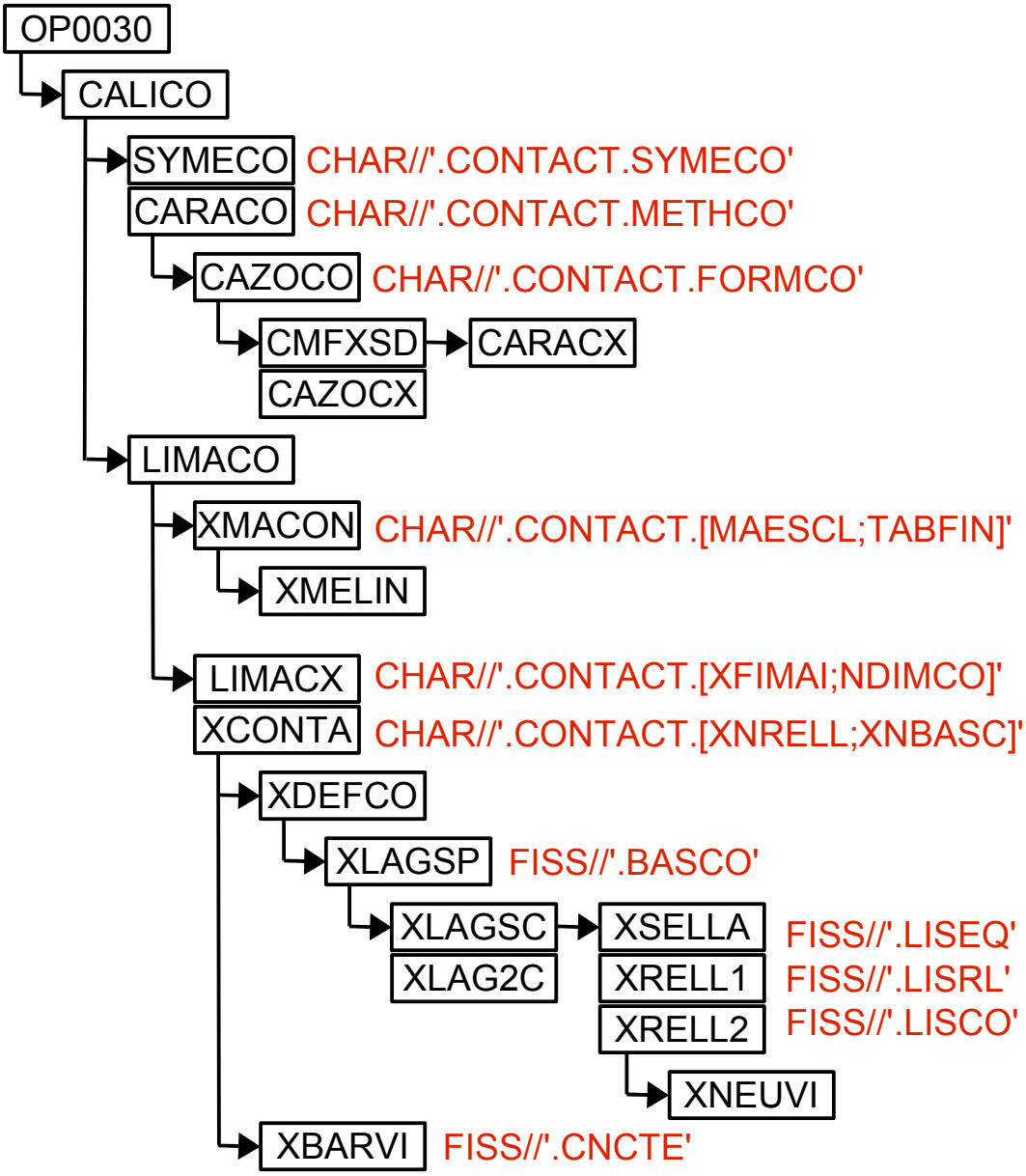


**Figure 3. L'arbre d'appel de l'opérateur `DEFI_CONTACT` pour le mot clé `FORMULATION='XFEM'`**



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

# 5 Affectation de la liaison X-FEM

Il s'agit ici du traitement du mot clé 'LIAISON_XFEM' dans 'AFFE_CHAR_MECA', spécifique pour la X-FEM, et aucun changement n'est intervenu pour le passage vers le contact en grands glissements. On présente brièvement les routines Fortran en précisant que le but principal est d'appliquer les relations entre les degrés de liberté de contact pour satisfaire la condition LBB. Le traitement sert aussi à éliminer les degrés de liberté de contact en trop dans le cas de l'ancienne formulation.

Voilà un exemple d'appel dans le fichier de commande:

```
CHXFEM=AFFE_CHAR_MECA(MODELE=MODELEK,
                      CONTACT_XFEM=CTXFEM,
                      LIAISON_XFEM='OUI',);
```

La description des routines Fortran et leur arbre d'appel (Figure 4) sont donnés ci-dessous.

CAXFEM – la routine de lancement pour les opérations d'annulation des degrés de liberté de contact en trop.

XDELCO - routine pour supprimer les degrés de liberté de contact en trop. Uniquement pour l'ancienne formulation du contact, où les Lagrangien sont stockés aux arêtes.

XRELCO – routine pour appliquer les relations entre les degrés de liberté de contact afin de satisfaire la condition LBB. On rappelle que ces relations sont créées maintenant dans l'opérateur DEFI_CONTACT (FORMULATION='XFEM'). Concernant le respect de la condition LBB pour les grands glissements, il se trouve que les algorithmes développés et implémentés pour le cas HPP [8] fonctionnent bien dans ce cas, ce qui est logique compte-tenu du fait que les inconnues de contact sont stockées seulement sur la partie esclave de la fissure.

AFLRCH – la routine qui affecte la liste de relations à l'objet de type charge. Autrement dit, cette routine rend effective l'application de la liste de relations créée précédemment.

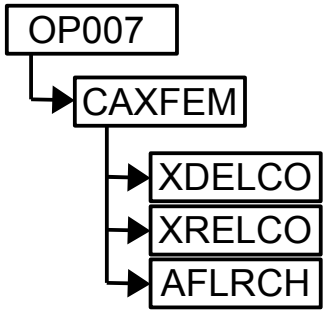


**Figure 4. L'arbre d'appel de l'opérateur AFFE_CHAR_MECA, mot clé LIAISON_XFEM**



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

# 6 Calcul des contributions de contact

Pour un problème de contact traité avec la méthode continue, la résolution se fait avec 5 boucles imbriquées dans l'opérateur STAT_NON_LINE [U4.51.03]:

- boucle sur les pas de charge,
- boucle géométrique,
- boucle sur les seuils de frottement,
- boucle de contraintes actives,
- itérations de Newton.

Pour comprendre le fonctionnement général de l'opérateur STAT_NON_LINE, on recommande vivement la lecture du document [D9.05.01]. La Figure 5 représente l'arbre d'appel de cet opérateur pour le cas spécifique d'un problème de contact X-FEM. On peut ainsi trouver sur cette figure les emplacements des boucles mentionnées ci-dessus.
La Figure 5 permet aussi de repérer où se situent les développements informatiques majeurs effectués pour l'approche *grands glissements avec X-FEM*, à savoir :

- l'arbre d'appel de la réactualisation géométrique (présenté Figure 6),
- l'arbre d'appel du calcul des contributions de contact-frottement (présenté Figure 7).
- l'arbre d'appel de la réactualisation des statuts de contact (présenté Figure 8),

Voilà, ci-dessous, un exemple d'appel à l'opérateur STAT_NON_LINE pour une analyse X-FEM:

```
UTOT1=STAT_NON_LINE( MODELE=MODELEK,
                     CHAM_MATER=CHAMPMAT,
                     EXCIT=(   _F(CHARGE=CTXFEM,),
                               _F(CHARGE=CHXFEM,),
                               _F(CHARGE=CH1,FONC_MULT=VAR1),),
                     COMPORTEMENT=_F(RELATION='ELAS',
                                  GROUP_MA='SURF',),
                     INCREMENT=_F(LIST_INST=L_INST,),
                     CONVERGENCE=_F(ITER_GLOB_MAXI=8,
                                 RESI_GLOB_MAXI=1E-06,),
                     SOLVEUR=_F(METHODE='MUMPS',
                               PCENT_PIVOT=500,),
                     ARCHIVAGE=_F(CHAM_EXCLU='VARI_ELGA',),
                     INFO=1,);
```

La description des routines Fortran:

OP0070 - un seul changement ici: l'appel à la routine de mise à zéro des Lagrangiens de contact. La routine XMISZL fait, pour le cas X-FEM, la même chose que l'ancienne routine MISAZL dans le cas standard du contact avec la méthode continue.

NMCTGE – routine chapeau qui lance le calcul de la géométrie des facettes de contact (XREACG) et l'appariement (XAPPAR).

XREACG – nouvelle routine qui réactualise la position des points d'intersections formant les facettes de contact. On utilise la SD MODELE//'.TOPOFAC.OE', contenant la position initiale des points d'intersection, pour calculer les SD MODELE//'.TOPOFAC.GE', et MODELE//'.TOPOFAC.GM', en appelant XGECFI. Ces 2 dernières SD correspondent respectivement aux coordonnées réactualisées des côtés esclaves et maîtres.

XGECFI – lance le calcul de la nouvelle géométrie des facettes de contact (esclave et maître). On passe par un calcul élémentaire (TE0519 avec l'option "GEOM_FAC").

TE0519 – calcule la nouvelle géométrie des points d'intersection formant les facettes de contact (esclave et maître). On rentre avec la géométrie initiale des facettes, stockée dans



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

MODELE//'.TOPOFAC.OE'. On sort avec les géométries réactualisées esclave et maître dans MODELE//'.TOPOFAC.GE' et MODELE//'.TOPOFAC.GM'. La réactualisation est effectuée à chaque début de boucle géométrique. Elle est utilisée ensuite pour apparier les points de contact esclaves aux mailles maîtres.

XAPPAR – nouvelle routine pour l'appariement dans le cas X-FEM. C'est une routine complexe, qui appelle à son tour d'autres routines (voir l'arbre d'appel). Le principe est de boucler sur la zone de contact, ensuite sur les mailles esclaves, puis sur les facettes de contact de la maille esclave, et enfin sur les points de contact appartenant à la facette de contact. Pour chaque point de contact on remplit CHAR//'.CONTACT.TABFIN' avec des informations sur le point de contact esclave et sur la maille maître trouvée, afin de créer l'appariement. A retenir: il doit y avoir autant de mailles tardives que de points de contact.

MMGAUS – calcul les coordonnées de référence du point d'intégration dans la facette de contact.

XCOPCO – nouvelle routine appelée par XAPPAR pour calculer les COordonnées réelles d'un Point de COntact à partir de la connaissance de ses coordonnées de référence (déterminées avec l'ancienne routine MMGAUS) et des coordonnées des points d'intersections de la facette de contact de la maille esclave.

XMREPT – nouvelle routine pour la REcherche du PoinT d'intersection sur le côté maître, le plus proche du point de contact courant. Elle fournit un certain nombre d'informations sur ce point d'intersection: son numéro local, le numéro local de l'arête (ou du nœud si la fissure passe par un nœud) qui le contient, et le numéro global de la maille qui le contient. Ces informations vont nourrir la recherche du projeté sur la lèvre maître. Notons que cette étape permet de restreindre le nombre de facettes sur lesquelles le point esclave sera projeté dans XMREMA, afin de gagner en performances. Dans certains cas (même en éléments finis classiques) cette sélection est trop restrictive et conduit à des résultats faux. Il faudrait revoir cette approche.

XMREMA – nouvelle routine pour REchercher la MAille maître la plus proche connaissant le point d'intersection maître le plus proche du point de contact et pour faire ensuite la projection. On sort d'ici avec des informations essentielles sur le projeté et la maille maître (coordonnées de référence, numéro global de la maille maître, la base tangente au point de projection, l'information sur la projection hors de la surface maître, etc.). En 2D, l'unique facette de contact associée à la maille maître est toujours un SEG2. En 3D, la facette maître associée est toujours un TRI3, mais comme il y a plusieurs facettes par maille, il est aussi important de déterminer le numéro de facette. Dans le cas multi-fissure, on détermine aussi le numéro local de la fissure correspondant à la zone de contact dans l'élément maître. Ce numéro à été précédemment déterminé dans XMACON et stocké dans la SD CHAR//'.CONTACT.MAESCL'.

MMPROJ – ancienne routine de la méthode de contact FEM, qui réalise la projection d'un point de contact sur une maille donnée. Pour la méthode X-FEM, cette maille est "fictive" et définie par les coordonnées réactualisées des points d'intersections définissant la facette de contact maître.

XMCOOR – nouvelle routine qui calcule les coordonnées de référence d'un point de contact dans la maille parent (esclave ou maître) connaissant les coordonnées de référence de ce point dans la facette de contact, et les coordonnées de référence des points de la facette de contact dans la maille parente.

XMRLST – nouvelle routine qui calcule à partir de la level set tangente, la racine carrée de la distance au fond de fissure pour les points d'intégrations esclaves et maitres. Cela est utile lorsque l'élément est enrichi par les fonctions Crack Tip.

XPIVIT– nouvelle routine qui détermine si le point d'intégration est vital ou pas. On regarde si le point de contact est sur une arête, et si c'est le cas, on lui donne le statut de l'arête en question (statut stocké dans TOPOFAC.AI). On sort d'ici avec l'information « point vital » ou « point non vital », ainsi que le numéro de groupe et numéro d'arête de ce groupe si le point de contact se situe sur une arête appartenant à un groupe d'arêtes connectées.



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

NMCTCL – routine chapeau pour la préparation des données afin de lancer le calcul des contributions de contact. Des modifications ont été opérées afin de lancer d'ici la création de la LIGREL (LIste de GRoupes d'ELéments de même type) contenant les éléments hybrides de contact (routine XMLIGR) et des cartes de contact (routine XMCART).

XMLIGR – nouvelle routine pour la création de la LIGREL pour les nouveaux éléments tardifs de contact. C'est ici que les éléments de contact sont créés par l'association entre une maille esclave et une maille maître. On les range ensuite dans des GREL puis dans une LIGREL. La routine est semblable à l'ancienne routine MMLIGR. Les mailles tardives 2D de type QU8QU4 et TR6TR3 sont relatives à l'ancienne formulation (P1-P2). Elles seront résorbées ultérieurement. Pour la nouvelle formulation,on dispose des mailles tardives 2D QU4QU4, TR3TR3, QU4TR3 et TR3QU4 afin de traiter le contact en linéaire (P1-P1) ainsi que des mailles QU8QU8 et TR6TR6 pour le semi-quadratique (P2-P1). En 3D seules les mailles HE8HE8, PE6PE6 et TE4TE4 sont disponibles et le traitement du contact n'est possible qu'en linéaire (P1-P1). Enfin pour traiter les différents types d'éléments x-fem, chaque maille tardive décrite ci-dessus (uniquement pour le P1-P1) est associée à plusieurs éléments tardifs (spécificité de X-FEM). Ainsi on dispose en plus de l'élément $H-H$ des éléments $H-HCT$, $HCT-H$, $HCT-HCT$ et enfin $CT$ (qui est tout seul car on le traite en petits glissements pour le moment) pour traiter le contact grand glissement en pointe de fissure. Dans le cas de la multi-fissuration, il est aussi possible d'apparier des éléments ne possédant pas le même nombre de degrés de liberté Heaviside. Ainsi il est aussi possible de traiter les éléments $H-H_2$, $H-H_3$, $H-H_4$, $H_2-H$, $H_2-H_2$, $H_2-H_3$, $H_2-H_4$, $H_3-H$, $H_3-H_2$, $H_3-H_3$, $H_3-H_4$, $H_4-H$, $H_4-H_2$, $H_4-H_3$, $H_4-H_4$. Le nombre de types d'éléments multi-Heaviside étant élevé (car il faut aussi les générer pour chaque type de maille), les bibliothèques des éléments multi-Heaviside tardifs ont été générées à l'aide d'un script python.

XMELEL – fournit des informations sur l'élément tardif de contact. Cette routine fonctionne pour tous les éléments énoncés dans XMLIGR. Vu le nombre important d'éléments, cette routine retourne plusieurs informations sur l'élément tardif (type de mailles esclave et maître, type d'élément esclave et maître, type de modélisation) afin de reconstruire le type d'élément de manière dynamique dans XMLIGR.

XMCART – routine pour créer les cartes de contact (champs qui contiennent des informations sur les éléments de contact). On utilise des cartes mais il faudrait reprogrammer tout cela pour plutôt utiliser des CHAM_ELEM comme dans le cas classique (routine MMCHML). Dans le cas classique on ne crée qu'un seul champ. En X-FEM on crée plusieurs cartes car on doit fournir au calcul élémentaire des informations sur la topologie des éléments X-FEM. De plus, certaines de ces informations doivent être fournies à la fois pour la maille esclave mais aussi pour la maille maître. On crée la carte RESOCO//'.XFPO', qui est équivalente au CHAM_ELEM RESOCO//'.CHML' dans le cas classique ; la carte RESOCO//'.XFST' qui stocke les statuts d'enrichissement des nœuds des mailles esclave et maître ; la carte RESOCO//'.XFPI' contenant les positions des points d'intersection esclaves dans les coordonnées de référence; la carte RESOCO//'.XFAI' contenant les informations sur les arêtes intersectées esclaves ; la carte RESOCO//'.XFCF' contenant les informations sur la connectivité des facettes de contact esclaves; dans le cas où au moins un des éléments maître ou esclave est multi-fissuré, on ajoute les 2 cartes supplémentaires: RESOCO//'.XFHF' qui contient la valeur des fonctions Heaviside correspondant aux enrichissements des côtés maître et esclave et RESOCO//'.XFPL' qui contient la place des Lagrangiens associés aux nœuds dans la matrice.

NMELCM – routine pour lancer le calcul des matrices élémentaires de contact frottement. Elle est commune aux approches de contact continue (FEM, X-FEM HPP, X-FEM GG).

NMELCV – routine pour lancer le calcul des seconds membres de contact frottement. Elle est commune aux approches de contact continue (FEM, X-FEM HPP, X-FEM GG).

TE0366 – nouvelle routine de calcul élémentaire pour les contributions de contact frottant à la matrice de rigidité, dans l'approche *grands glissements avec X-FEM*. Un grand nombre de routines



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

(présentées par la suite) sont appelées ici. La plupart sont appelées également dans les TE0367 et TE0363.

TE0367 - nouvelle routine de calcul élémentaire pour les contributions de contact frottant au second membre, dans l'approche *grands glissements avec X-FEM*.

XMELET – nouvelle routine pour récupérer des informations sur l'élément de contact courant. On rentre avec le nom de l'élément tardif. On sort avec les noms des mailles esclave, maître; le numero de la facette de contact; les nombres de ddls, esclaves et maîtres, de la facette de contact; le nombre total de ddls; l'information sur le type de formulation (nœuds milieu ou nœuds sommets).

XMPINT – nouvelle routine pour le calcul des coordonnées réelles des points d'intersection constituant la facette de contact.

XTFORM – calcule les fonctions de formes et ses dérivées du point d'intégration à l'aide des coordonnées paramétriques dans l'élément parent. Elle est appelée par les TE0366 et TE0367, les dérivées des fonctions de forme servent ensuite à calculer le jacobien du point de contact dans XMMJAC.
XLACTI – calcule le statut actif ou non des Lagrangiens de contact. Un nœud est actif si il appartient à une arête coupée ou si il est directement coupé par la fissure.

XMOFFC – répartit les fonctions de formes des nœuds ou les Lagrangiens sont à éliminer sur les nœuds actifs.

XMMJAC – nouvelle routine utilisée pour calculer le jacobien du point de contact.

MMNORM – ancienne routine pour calculer la normale sortante au projeté du point de contact. Attention, en 3D on n'appelle pas MMNORM, car les facettes sont orientées dans l'autre sens pour la méthode de contact X-FEM par rapport à la méthode de contact FEM.

XTCALN – calcule le vecteur normal.

PROVEC – ancienne routine qui sert à faire le produit vectoriel des vecteurs tangents pour obtenir le vecteur normal en 3D.

NORMEV – ancienne routine qui normalise un vecteur. Elle sert à normaliser le vecteur normal en 3D, mais elle est aussi appelée pour projeter le vecteur frottement sur la boule unité dans le cas glissant du frottement.

XOULA – routine propre à X-FEM, qui sert pour la connectivité des ddl de contact. Elle ne sert plus pour la formulation aux nœuds sommets.

XPLMA2 – nouvelle routine qui fournit la place d'un *ddl* de contact dans la matrice de rigidité élémentaire, correspondant aux éléments tardifs X-FEM. La routine correspondante en HPP est XPLMAT.

XTLAGM – calcule les Lagrangiens de contact et de frottement pour le point de contact.

XTLAGC – calcule le Lagrangien de contact pour le point de contact

XTLAGF – calcule le Lagrangien de frottement pour le point de contact

XTDEPM – calcule les incrément de déplacement depuis le pas de temps précédent du point de contact esclave et de son projeté sur la facette maître.

XMMJEU – nouvelle routine appelée par les TE0366 et TE0367 pour calculer le jeu entre le point de contact et son projeté. Au cours des itérations de Newton, on a accès aux déplacements totaux du pas



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

de temps précédent (idepl) et aux incréments de déplacements depuis le pas de temps précédent jusqu'au Newton en cours (idepm). La position est donc calculée en faisant géométrie initiale + déplacement total du pas de temps précédent + incrément jusqu'au Newton en cours.

XMMJEC – nouvelle routine appelée par le TE0363 pour calculer le jeu entre le point de contact et son projeté. Lors de la réactualisation des statuts de contact, on a accès aux déplacements totaux (idepl): la position est donc calculée en faisant géométrie initiale + déplacement total.

TTPRSM – ancienne routine qui calcule $g_\tau$, $\|g_\tau\|$ et $[P_\tau]$.

XMMAA1 – nouvelle routine pour le calcul des termes $A_u$, $A$ et $A^T$ (cas avec contact) de la matrice de rigidité due au contact. Les expressions des termes de contact-frottement dans la matrice de rigidité et dans le second membre sont données dans la documentation [R5.03.53].

XMMAA0 – nouvelle routine pour le calcul des termes dus à la modélisation du contact dans la matrice de rigidité (cas sans contact).

XMMAB0 - nouvelle routine pour le calcul des termes dus à la modélisation du contact dans la matrice de rigidité (cas du contact sans frottement).

XMVEC1 - nouvelle routine pour le calcul des termes du second membre (cas avec contact).

XMVEC0 - nouvelle routine pour le calcul des termes du second membre (cas sans contact).

XMMAB2 - nouvelle routine pour le calcul des termes $Bu$, $B$, $BT$ et $F$ dans la matrice de rigidité due au frottement (cas du contact frottement glissant).

MKKVEC – ancienne routine qui applique une transformation à un vecteur. Cette routine est appelée par XMMAB2, elle est utile pour calculer les matrices de frottement dépendantes de $K(g_\tau)$.

XMMAB1 - nouvelle routine pour le calcul des termes $Bu$, $B$, $BT$ dans la matrice de rigidité due au frottement (cas du contact frottement adhérent).

XMMAB0 - nouvelle routine pour le calcul du terme de frottement $F$ dans la matrice de rigidité due au frottement (cas du contact sans frottement ou cas sans contact).

XMVEF1 - nouvelle routine pour le calcul des termes de frottement $L^{1\text{frott}}$ et $L^{3\text{frott}}$ du second membre (cas du contact frottant).

XMVEF0 - nouvelle routine pour le calcul du terme de frottement $L^{3\text{frott}}$ du second membre (cas sans contact et cas sans frottement).

XTEDD2 – nouvelle routine qui élimine les degrés de liberté Heaviside et Crack Tip en trop pour les contributions de contact en grands glissements X-FEM. Elle élimine aussi les ddls de contact en trop.



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

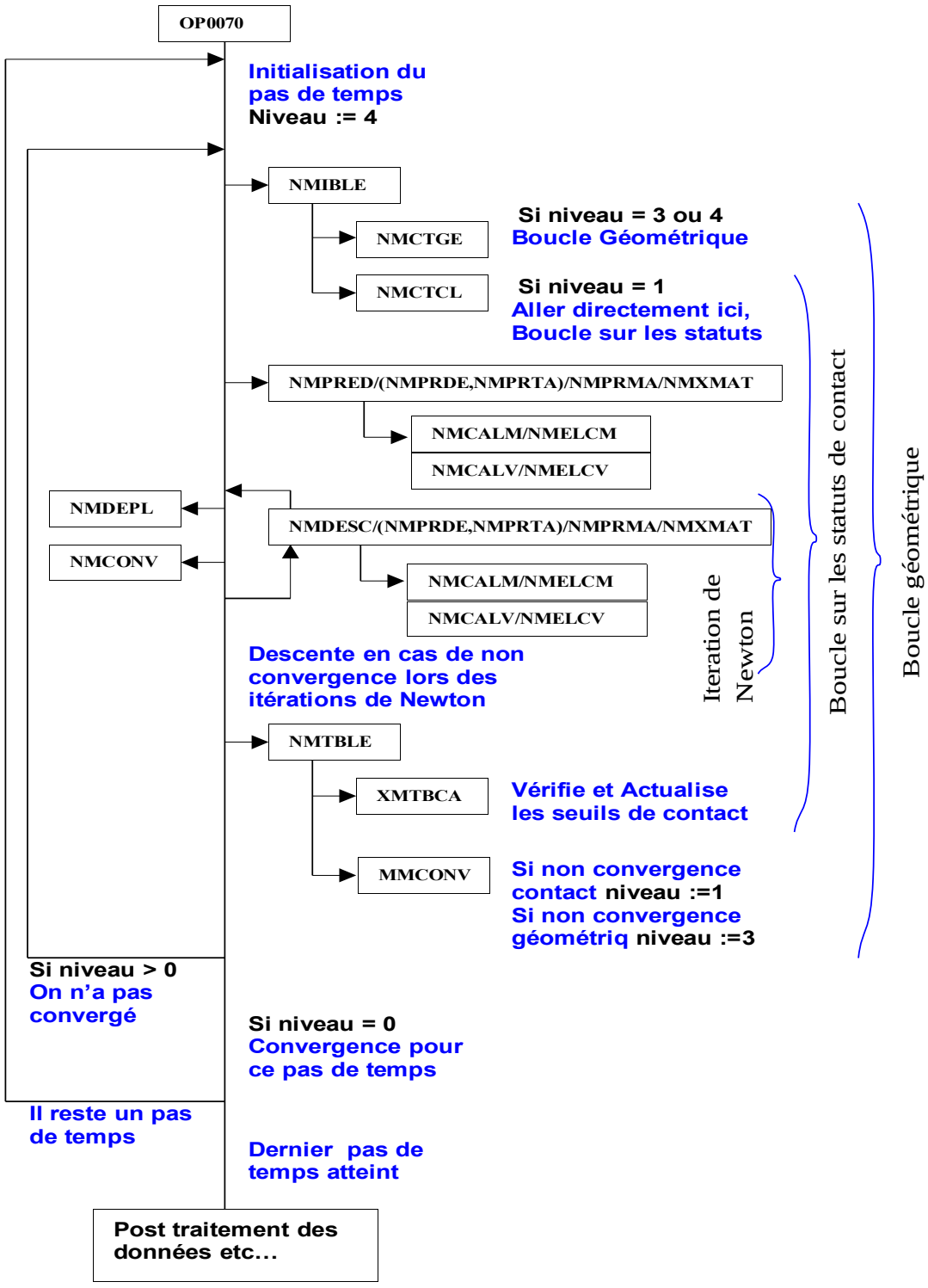


**Figure 5. L'arbre d'appel de l'opérateur STAT_NON_LINE pour le traitement d'un problème de contact en grands glissements avec XFEM.**



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

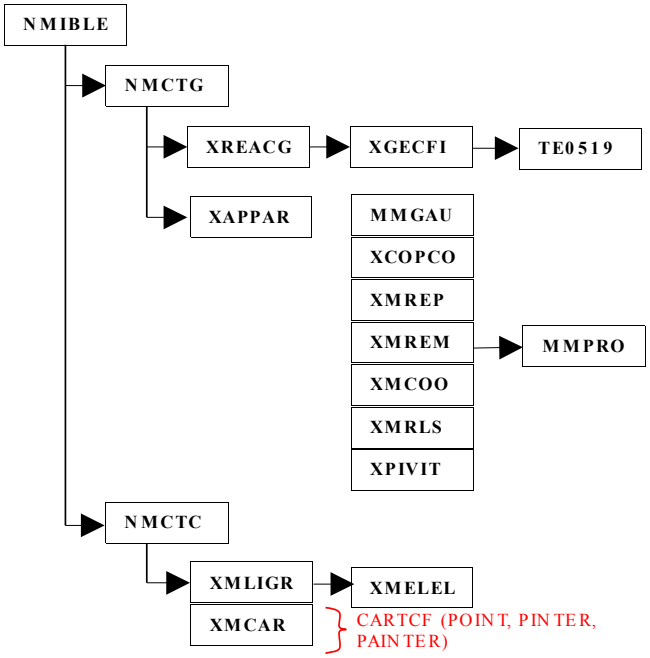


**Figure 6. L'arbre d'appel lors d'un réappariement géométrique pour un problème de contact en grand glissement avec X-FEM.**

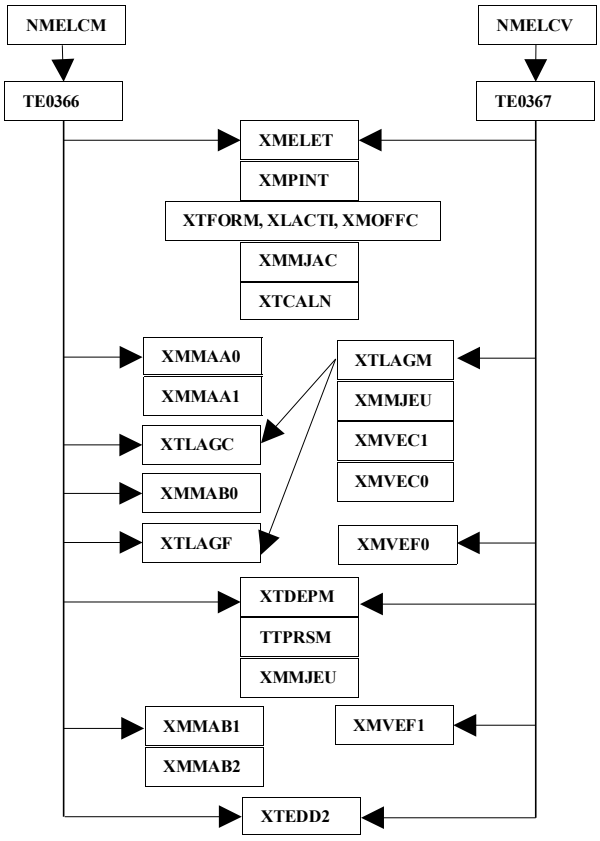


**Figure 7. L'arbre d'appel pour le calcul des contributions de contact dans la matrice et le second membre pour un problème de contact en grand glissement avec X-FEM.**



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

# 7 Réactualisation des statuts de contact et des seuils de frottement

Les routines appelées lors de la réactualisation des statuts de contact et des seuils de frottement (voir l'arbre d'appel dans la Figure 8) sont décrites ci-dessous.

| |
|---|
| NMTBLE – routine de lancement pour la boucle de contraintes actives et celle sur les seuils de frottement (cousine de NMIBLE). On a ici changé l'appel à la routine XMMBCA (valable pour le cas HPP) par l'appel à la nouvelle routine XMTBCA (pour les grands glissements). Pour les seuils de frottement, on ne fait rien, car cette boucle n'est plus active en grands glissements X-FEM. |
| XMTBCA – nouvelle routine qui lance le calcul pour vérifier la convergence sur la boucle de contraintes actives et pour corriger le statut des points de contact si besoin. La vérification se fait par l'appel du TE0363 pour chaque point d'intégration de contact. Pour chaque point d'intégration, on stocke la valeur de changement de statut. Cette valeur vaut 1 si il y a changement de statut pour ce point, elle vaut 0 sinon. Ensuite, il y a sommation de toutes ces valeurs, si la somme est strictement positive, on a un code de retour négatif donc pas de convergence. C'est ici que l'on ajoute aussi la vérification sur le statut des points d'intégration de contact qui appartiennent à un groupe d'arêtes vitales. |
| TE0363 – nouvelle routine de calcul élémentaire associée à XMTCBA pour vérifier la convergence sur la boucle de contraintes actives. Attention, cette routine n'a pas de correspondance pour le cas classique (la vérification se fait alors dans la routine MMALGO). |
| MMCONV – ancienne routine qui assure la gestion de la convergence pour la méthode continue. Dans le cas HPP on avait toujours imposé la convergence géométrique (car il n'y avait pas de réactualisation géométrique). On a donc modifié la routine pour que la vérification soit faite dans le cas *grands glissements avec X-FEM*. |

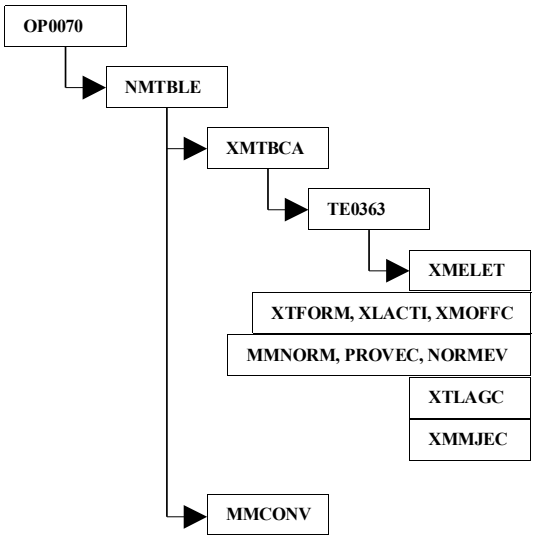


**Figure 8. L'arbre d'appel de l'opérateur STAT_NON_LINE pour la boucle sur les seuils de frottement et celle des contraintes actives**



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

# 8 Perspectives de développement

En conclusion, l'implémentation numérique de l'approche *grands glissements avec X-FEM* a été validée dans les conditions énoncées au début de ce document. L'approche est donc utilisable avec frottement dans Code_Aster en 2D (contraintes planes et déformations planes pour les types d'éléments TRI3, QUAD4, TRI6 et QUAD8) et en 3D (pour des éléments HEXA8, PENTA6 et TETRA4). L'approche a aussi été généralisée au fond de fissure.

L'approche *grands glissement avec X-FEM* est aussi validée pour les éléments multi-fissurés (jonctions notamment) sans frottement en 2D et 3D. Il reste toutefois des efforts à faire pour garantir la robustesse de la méthode dans ce cas.



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