Data structures related to X-FEM

Summarized

This document describes the (SD) data structures related to the method X-FEM. These objects are in several SD:

1) `sd_fiss_xfem`, produced by the operator `DEFI_FISS_XFEM` [U4.82.08] and used for a computation of fracture mechanics in the frame of the method X-FEM [R7.02.12]. This data structure is also used and produced by the operator `PROPA_FISS` [U4.82.11],
2) `sd_modele_xfem`, is produced `MODI_MODELE_XFEM` [U4.41.11],
3) `sd_contact_xfem`, is by the operator produced by the operator `DEFI_CONTACT` [U4.44.01].

1 General information

During a computation X-FEM, information concerning crack not being contained in the mesh, it is necessary to store them in a specific (SD) data structure. Then, the enrichment of the elements leads to new finite elements, whose certain characteristics will be stored in the model. Then, if the contact is defined on the lips of crack, a contact load is then created, which contains information specific to X-FEM. Thus, the objects relating to the method X-FEM are distributed in 3 SD different: `sd_fiss_xfem` , `models` and `charges` . These three SD are the object of three most of this document.

An object of the sd_fiss_xfem `type` describes a crack contained in a 2D mesh or 3D, without this mesh not being inevitably in conformity with the geometry of crack. The data structure `sd_fiss_xfem` contains the description of crack by the means of level sets, the associated local bases, of the data related to enrichment of the finite elements around crack, of topological information concerning under-cutting of the finite elements nouveau riches, and of the data related to the contact between the lips of crack. Moreover, this structure is enriched by the addition of data necessary to the propagation of crack.

The model enriched allows to globalize local information with each crack on the total mesh. It contains also information on cracks of the model. This document does not describe that the objects of the model specific to X-FEM (for the description of `the sd_modele` , to see [D4.06.02]).

If the contact is defined on the lips, the SD of contact contains information specific to X-FEM, in particular concerning the algorithm of construction of the Lagrange multipliers to satisfy condition LBB: lists of relations to be imposed,… This document describes only the objects of the contact specific to X-FEM (for the description of the SD related to the contact friction, to see [D4.06.14]).

2 Tree structure

```
sd_fiss_xfem  (K8)  .: =record

    ♦  "                .INFO'OJBSVK16
       ″                 .MODELE'OJBSVK8

    ◊  ".CHAMPS.LVS'            OJBSVL

    ♦  ″                 .LTNO'CHAM_NO
       "                .LNNO'CHAM_NO
       ″                 .GRLTNO'CHAM_NO
       "                .GRLNNO'CHAM_NO
       ″                 .BASLOC'CHAM_NO
       "                .STNO'CHAM_NO
       ″                 .STNOR'CHAM_NO

    ♦  ".GROUP_MA_ENRI'         OJBSVI
       ″ .GROUP_NO_ENRI'           OJBSVI
       ".CARAFOND'              OJBSVR
```

◊ % if the structure contains so

```
       a crack tip″             .FONDFISS'
       OJBSVR "             .BASEFOND'
       OJBSVR″               .FONDMULT'
       OJBSVI                 "
       .LTNS'OJBSVK24                   ″
       .LTNT'OJBSVK16 "        .FOND.TAILLE_R'
```

OJBSVR ◊ % of meshes is entirely cut by crack

```
       ″ .MAILFISS.HEAV'           OJBSVI
```

◊ % so of meshes contain so

```
       crack tip "          .MAILFISS.CTIP'
```

OJBSVI ◊ % of meshes is cut by crack and contains crack tip

```
       ″ .MAILFISS.HECT'           OJBSVI
```

◊ % so of the contact is defined on certain cracks of the model

```
       ".MAILFISS.CONT'        OJBSVI
```

◊ % if the crack is resulting D” a propagation by the operator PROPA_FISS

```
       ".PRO.MES_EL″               CHAM_ELEM
       ".PRO.NORMAL″               CHAM_ELEM
       ".PRO.RAYON_TORE'           OJBSVR
       ″ .PRO.NOEUD_TORE'          OJBSVL
       ".PRO.NOEUD_PROJ'           OJBSVL
```

◊ % if a grid is associated with crack by DEFI_FISS_XFEM

```
       ″ .GRI.MODELE'                 OJBSV                      K8
       "                    .GRI.LNNO'CHAM_NO
       ″                     .GRI.LTNO'CHAM_NO
       "                    .GRI.GRLNNO'CHAM_NO
       ″                     .GRI.GRLTNO'CHAM_NO
```

◊ % if the grid contains so

```
       a crack tip "           .FONDFISG'
```

`OJBSVR` ◊ % of the relations D" equality between unknowns of contact are imposed
`".LISEQ'` `OJBSVI`

◊ % so relations D" equality between unknowns of contact are imposed and if there are elements multi-Heaviside
`".LISEQ_LAGR'` `OJBSVI`

◊ % if there are groups connected vital edges
`" .CNCTE'` `OJBSVI`

◊ % if key word `JONCTION` were used
`" .JONFISS'OJBSVK8`
`" .JONCOEF'` `OJBSVI`

```
sd_modele_xfem    (K8)  .: =record

    ♦  ‶            .TOPOSE.PIN'CHAM_ELEM
       ″             .TOPOSE.CNS'CHAM_ELEM
       ‶            .TOPOSE.HEA'CHAM_ELEM
       ″             .TOPOSE.LON'CHAM_ELEM


    ♦  ‶            .TOPOFAC.PI'CHAM_ELEM
       ″             .TOPOFAC.AI'CHAM_ELEM
       ‶            .TOPOFAC.CF'CHAM_ELEM
       ″             .TOPOFAC.LO'CHAM_ELEM
       ‶            .TOPOFAC.BA'CHAM_ELEM
       ″             .TOPOFAC.OE'CHAM_ELEM
       ‶            .TOPOFAC.GE'CHAM_ELEM
       ″             .TOPOFAC.GM'CHAM_ELEM

    ◊   % if there are elements multi-Heaviside
        ‶            .TOPOFAC.HE'CHAM_ELEM

    ♦  ″                 .LTNO'CHAM_ELNO
       ‶                .LNNO'CHAM_ELNO
       ″                 .BASLOC'CHAM_ELNO
       ‶                .STNO'CHAM_ELNO
       ″                 .NOXFEM'CHAM_ELNO

    ◊   % so of the elements stores the information of several cracks
       ‶                .FISSNO'CHAM_ELNO
       ″                 .HEAVNO'CHAM_ELNO

    ♦  ‶.XFEM_CONT'            OJBSVI

    ♦  ″ .NFIS'                OJBSVI
       ‶                .FISS'OJBSVK8
       ″                 .XMAFIS'CHAM_ELEM


sd_contact_xfem   (K16)    .: =record

    ♦  ‶.CARACF'           OJBSVR
       ″ .ECPDON'              OJBSVI
       ‶.METHCO'           OJBSVI

       ″             .XFIMAI'OJBSVK8
       ‶            .XNRELL'OJBSVK24

    ◊   % cards of contact for approach large sliding

       ″ .XFPO'            CARD
       ‶.XFST'             CARD
       ″ .XFPI'            CARD
       ‶.XFAI'             CARD
       ″ .XFCF'            CARD
       ‶.XFHF'             CARD
       ″ .XFPL'            CARD
```

3 Contained of the objects of the sd_fiss_xfem

Opening remark on the distinction enters 2D and 3D:

The presence of certain objects in the data structure sd_fiss_xfem depends sometimes on dimension of the mesh on which the crack is represented. Thus, the distinction enters 2D and 3D is referred corresponding to mesh contained in the model given by the user under the key word MODELS DEFI_FISS_XFEM of the command.

3.1 General objects

3.1.1 .INFO

concept .INFO is a vector $K16$ length 3 container of general information.
The 1st component of this vector informs about the type of discontinuity relating to the sd_fiss_xfem : either a crack, or an interface. The 2nd component indicates the discontinuous field: either the field of displacement, or the stress field. The 3rd component specifies if the bottom is closed or opened.
.INFO (1) = "CRACK" or "INTERFACE"
.INFO (2) = "DEPL" or "SIGM"
.INFO (3) = "OUVERT" or "FERME"
One reaches the values of this vector only by the routine dismoi.f. Example:
CAL
DISMOI ("F", "TYPE_DISCONTINUITE", FISS, "FISS_XFEM", IBID, TYPDIS, IRET)
CAL
DISMOI ("F", "CHAM_DISCONTINUITE", FISS, "FISS_XFEM", IBID, TYPCHA, IRET)
CAL
DISMOI ("F", "TYPE_FOND", FISS, "FISS_XFEM", IBID, TYPFON, IRET) .MODELE

3.1.2

concept .MODELE is a vector of K8 length 1 container the name of the model on which is built the crack (value of the MODEL key word in entry of DEFI_FISS_XFEM). In the event of propagation, this vector is duplicated old crack with new crack. One
reaches this vector only by the routine dismoi.f . Example:
CAL
DISMOI ("F", "NOM_MODELE", FISS, "FISS_XFEM", IBID, MODELS, IRET) Objects

3.2 relating to the level sets .CHAMPS

3.2.1 .LVS concept

.CHAMPS .LVS is a vector of Boolean length 1. If this vector is present (and its value is then always equal to TRUE), the fields level set of current crack were given directly by the user (by DEFI_FISS_XFEM by means of the key words FIELD _NO_LSN and FIELD _NO_LST) and they were not calculated. If
the two fields were extracted from a crack propagated by PROPA_FISS , information on the localization of field (.PRO.RAYON_TORE (§3.4 .3)14 .PRO. NOEUD_TORE (§3.4 .4)14 on the use of auxiliary grid for propagated crack (.GRI.MODELE (§3.4 .6)14. L*NO (§3.4 .7)15 .GRI. GRL*NO (§3.4 .8)15were lost and the SD of current crack does not contain them. Indeed this information is stored on the level of the SD fiss_xfem and not on the level of the SD champ_no. That poses problem if the current crack is propagated by PROPA_FISS because one could get false results. Indeed whenever the localization of the field or auxiliary grid were used, the level sets were put at days only in one small field around the bottom of crack: in the first case that is obvious, and in the second case the level sets were projected on the physical mesh with a field of localised projection

independently of the use of the localization of a field of computation in crack tip on the grid. Apart from the field of localization/projection the value of the levels set is the same one as that calculated to define crack: the level sets are thus not continuous in extreme cases of the field of localization /projection and the description of crack is thus not correct on all the mesh. If information on the field of localization and auxiliary grid is not stored in the SD of new crack, PROPA_FISS propagates crack by considering that the level sets are correct on all the mesh. When the bottom of crack propagated leaves the field localization/projection used for the initial crack before propagation, false results are got and one has problems of convergence in the rebootstrapping and the reorthogonalisation of the level set. If
the level sets given to define new crack were not extracted from a crack propagated by PROPA_FISS , or although the localization of the field or auxiliary grid were not used with PROPA_FISS , the fields level sets given correctly represent crack on all the mesh and the propagation of new crack can be calculated without problem by PROPA_FISS . If this
vector is present in the SD of crack, PROPA_FISS emits an alarm so that the user checks if it is not in the presence of a field extracted from a crack propagated with PROPA_FISS which it would use with localization of the field of computation or with one auxiliary grid. .LTNO

3.2.2 and .LNNO concept

.LTNO (resp . .LNNO) is a field at nodes (CHAM_NO) scalar which contains for each node of the mesh the actual value of the level set tangent (resp. norm) with crack. .GRLTNO

3.2.3 and .GRLNNO concept

.GRLTNO (resp . .GRLNNO) is a field at nodes (CHAM_NO) with 3 real components. It contains for each node of the mesh the values of the gradient of the level set tangent (resp. norm) in the 3 directions of space. That is to say

the ème i i node of the mesh, V =.GRLTNO

(I); V (1)

Value	of the following gradient of x the level set tangent, calculated with the ème i node V (2)
Value	of the following gradient of y the level set tangent, calculated with the ème i node V (3)
Value	of the following gradient of z the level set tangent, calculated with the ème i node In 2D

, one has only 2 components according to and. x .BASLOC y

3.2.4 concept

.BASLOC is a field at nodes (CHAM_NO) with 9 real components (in 3D). It contains the origin and the vectors of the Local Base to the crack tip. For each node, the first three components are the coordinates of the project of the node on the bottom, which corresponds at the origin of the local base. The three following components are the coordinates of 1st vector (GRLT) of the base. The three last components are cordonnées of 2nd vector (GRLN) of the base. The 3rd vector of the base is not stored, because it is determined easily as being the cross product of the first 2 vectors. That is to say

the ème i i node of the mesh , V =.BASLOC

(I); V (1)

Coordinated	according to project x of node I on the bottom V (2)

Coordinated	according to project y of node I on the bottom V (3)
Coordinated	according to project z of node I on the bottom V (4)
Coordinated	according to 1st x vector of the local base V (5)
Coordinated	according to 1st y vector of the local base V (6)
Coordinated	according to 1st z vector of the local base V (7)
Coordinated	according to 2nd x vector of the local base V (8)
Coordinated	according to 2nd y vector of the local base V (9)
Coordinated	according to 2nd z vector of the local base In 2D

, one only has 2 components according to and, x are y 6 components for BASLOC . Let us point out that in 2D, there is one crack tip, which is then a point (case of cracks 2D emerging) and in this case all the nodes of the mesh have the same project on the crack tip, that is to say there are two crack tips, therefore two points (case of cracks 2D not emerging) and in this case, part of the nodes of the mesh projects itself on the 1st crack tip and another part of the nodes of the mesh project on the 2nd crack tip . .FONDFISS

3.2.5 vector

.FONDFISS is a vector of realities containing the coordinates of the points of the crack tip. The points are ordered according to the method described in the document [R7.02.12], so that a curvilinear abscisse can be defined. If NFON
is the number of points of the crack tip, then the length of vector .FONDFISS is 4xNFON . For each point of the crack tip, the first 3 components correspond to the 3 coordinates (in 3D) of the point, and the fourth component is its curvilinear abscisse. When
the bottom is closed, the last point is equal to the first. The last 4 terms of vector .FONDFISS are then identical to the 4 first. This

structure is not modified in 2D. However one uses only the first 2 components, because neither the curvilinear abscisse nor the last geometrical component are relevant in 2D. This object

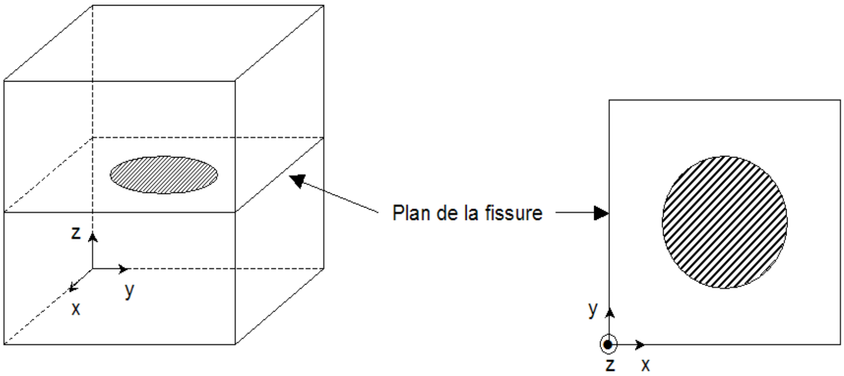
is created only if there exists at least a point of the crack tip. .BASEFOND

3.2.6 vector

.BASEFOND is a vector of realities containing the local base associated with each point of the crack tip. If NFON
is the number of points of the crack tip and if NDIM is the dimension of the model (2 in 2D and 3 in 3D), then the length of vector .BASEFOND is . In fact $2 \times NDIM \times NFON$ this vector contains, successively for all the points of the bottom, the components (2 in 2D and 3 in 3D) of the normal vector to the plane of crack and the direction of propagation (tangent vector to the plane of crack). In 3D
, when the bottom is closed, the last point is equal to the first. The last 4 terms of vector .BASEFOND are then identical to the 4 first. .FONDMULT

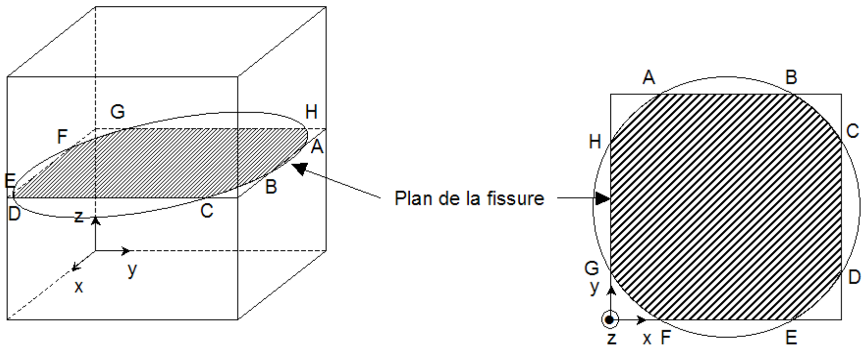
3.2.7 In 3D

, the crack tip either is line closed (a crack not emerging), or open (emerging crack). In most case, the crack tip is line continuous, like that of circular crack represented on the Figure 3.2.7 3.2.7-a



3.2.7 3.2.7-a of a continuous crack tip. However

, it can happen that the crack tip is in fact made up of several discontinuous pieces. It is the case for example circular crack represented on the Figure 3.2.7 3.2.7-b one always speaks about the crack tip , like all the pieces of the bottom. It is said that the crack tip is a multiple bottom. On the example of the Figure 3.2.7 3.2.7-b is composed of the curved lines, (BC) and (DE). (FG) Figure (HA)



3.2.7 3.2.7-b of a multiple crack tip. Vector

.FONDMULT is a vector of integers containing the indices of the starting and arrival points of the various pieces of the crack tip in the .FONDFISS. If NFONFI is the number of pieces of the crack tip, then vector .FONDMULT is of dimension 2x NFONFI . For example:

If FONDMULT
= (1; 8; 9; 12) AlorsMorceau
n°1 of the crack tip: item 1 as in point 8 of. FONDFISS Piece
n°2 of the crack tip: item 9 as in point 12 of. FONDFISS In 2D

, each point of the crack tip constitutes with him only a bottom, vector .FONDMULT nevertheless is created. For example for 2 crack tips, it will be worth: FONDMULT = (1; 1; 2; 2) the object

is created provided there exists at least a point of the crack tip. .CARAFOND

3.2.8 vector

.CARAFOND is a vector of realities of dimension 2. It contains the parameters users concerning the crack tip. The first component is the radius of enrichment of the elements and the second component is the number of layers of elements to enriching. As follows

: CARAFOND

(1) = RADIUS _ENRIsi RAYON_ENRI is provided by the user, 0sinon
. CARAFOND
(2) = NB_COUCHESsi NB_COUCHES is provided by the user, 0sinon
. These data

are those provided by the user in command DEFI_FISS_XFEM . They are stored because they are useful again in command PROPA_FISS to generate a propagated crack. .LTNS

3.2.9 vector

.LTNS is a vector of character strings containing the two arrays associated with crack. These arrays are accessible via command RECU_ COUNTS. First named array FOND_FISS contains the coordinates of the points of crack tips. In 2D , this one is consisted by the basic crack number and of the coordinates and. x Thus y , the display of the array will be: NUME_FOND

COOR_ X COOR_ Y Of 3D

, this one described each crack tip. Thus, for each crack tip I, the array will and the provide the curvilinear abscisse coordinated of the points in crack tip I. The display of the array will be: NUME_FOND

NUME_PT ABS_CURV COOR_ X COOR_ Y COOR_ Z second

named array NB_FOND _FISS contains the number of crack tips. The construction

of these arrays rests on the use of vectors .FONDFISS and. FONDMULT . .LTNT

3.2.10vector

.LTNT is a vector of character strings containing the names of the two arrays associated with crack required to command RECU_ COUNTS. It is of FOND_FISS and NB_FOND _FISS. .FOND

3.2.11.TAILLE_R This vector

contains for each node of the bottom, an estimate of the maximum size according to the direction of propagation, of meshes which are connected to them. These sizes are ordered according to the order of the nodes given in vector .FONDFISS. One note

formulated $\vec{V}_{P_i}$ of propagation of the local base to the node of bottom Ni and formulated a_{ijk} edge of the jème mesh to which node Ni belongs. For each node of bottom Ni, one projects the edges formulated a_{ijk} the vector of direction of propagation formulated $\vec{V}_{P_i}$ size formulated T_i connected to Nor is the maximum value of the absolute values of its projections. In other words, the size formulated T_i to formulated

$$T_i = \max_{\substack{1 \le j \le Nb_{mailles,i} \\ 1 \le k \le Nb_{arêtes,j}}} \left(\left| a_{ijk} \cdot V_{P_i} \right| \right)$$

formulated $Nb_{mailles,i}$ the number of meshes connected to node Ni and formulated $Nb_{arêtes,j}$ the number of edges of the jème mesh connected to node Ni. Objects

3.3 relating to enrichment .GROUP

3.3.1 _MA_ENRI vector

.GROUP_MA_ENRI contains the list of the numbers of meshes of the zone to enriching. .GROUP

3.3.2 _NO_ENRI vector

.GROUP_NO_ENRI contains the list of the numbers of nodes of the zone to enriching. Let us note that this vector can result from .GROUP_MA_ENRI. Note:

These the last two
vectors are stored because one wishes to re-use in PROPA_FISS information bound to the key word GROUP_MA_ENRI of DEFI_FISS_XFEM . And the only possibility of transmitting information is to store the numbers of meshes and the nodes of meshes of GROUP_MA_ENRI . .STNO

3.3.3 concept

.STNO is a field at nodes (CHAM_NO) with a whole component, corresponding to the statute of the node in question. If

- the node has its support entirely cut by crack, its statute is worth 1. If
- the node is in a zone "close" (this notion is defined according to a criterion defined by the user) to the crack tip, its statute is worth 2. If
- the node satisfies the two preceding conditions, its statute is worth 3. In
- all the other cases, the statute is worth 0. .STNOR

3.3.4 concept

.STNOR is exactly the same field as .STNO but with of the components real. It could thus be used for visualization with Salome for example . .MAILFISS

3.3.5 .HEAV vector

.MAILFISS.HEAV is a vector of integers containing the list of the numbers of meshes enriched by type Heaviside ("ROUND" mesh). .MAILFISS

3.3.6 .CTIP vector

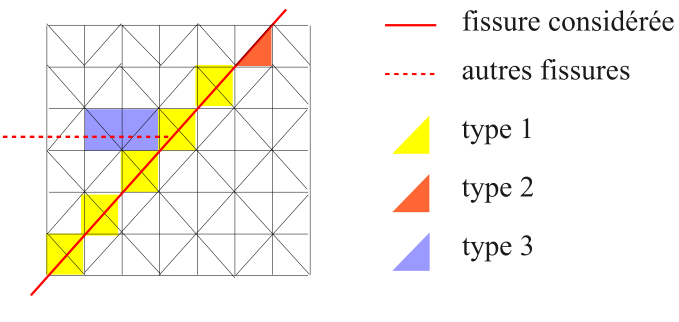
.MAILFISS.CTIP is a vector of integers containing the list of the numbers of meshes enriched by type Ace-Tip ("SQUARE" mesh). .MAILFISS

3.3.7 .HECT vector

.MAILFISS.HECT is a vector of integers containing the list of the numbers of meshes enriched by type Heaviside-Ace-Tip (mesh "RONDE-CARREE"). .MAILFISS

3.3.8 .CONT vector

.MAILFISS .CONT is a vector of integers containing the list of the numbers of meshes on which of the contact is defined for crack. It is about meshes: Either
–strictly intersected by crack (standard 1 on the fig.3.3.8-a3.3.8-a
–presenting an edge (in 2D) or a face (in 3D) confused with crack, and located on the side slave (standard 2 on the fig.3.3.8-a3.3.8-a ,
–in multi-cracking, of meshes presenting a Heaviside enrichment for crack considered and which enter one of the first two categories for another crack (standard 3 on the fig.3.3.8-a3.3.8-a
, of the contact to limit the introduction of new elements during the implementation with multi-cracking, the number of degrees of freedom of contact is always equal to the number of degrees of freedom of Heaviside, since it is necessary to then putting the superfluous degrees of freedom of contact at 0. From where need for including meshes this third type in the group of meshes "in contact" (but their degrees of freedom will be put at 0). Figure



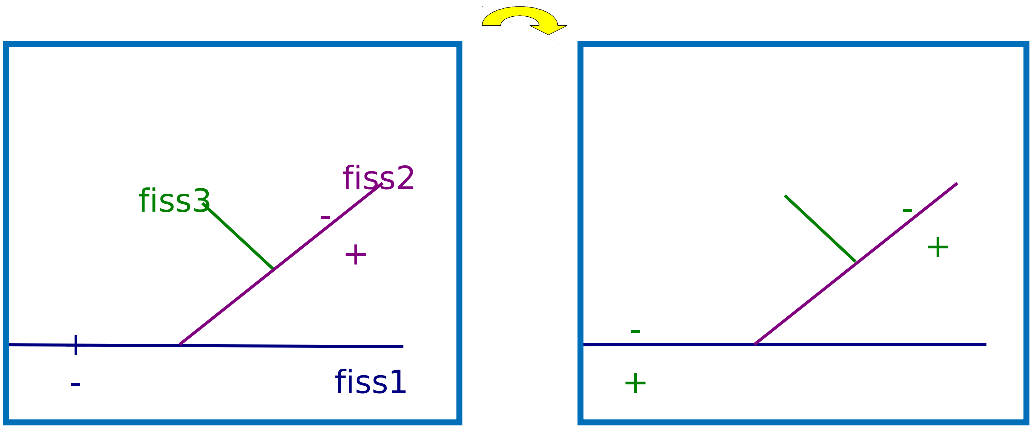
3.3.8 3.3.8-a of group .MAILFISS .CONT of a crack. .JONFISS

3.3.9 vector

.JONFISS contains the list of the cracks mothers on which the crack connects. It is in particular used in operator MODI_MODELE_XFEM for cutting and multiple enrichment. This vector is created if the user called on key word JONCTION in DEFI_FISS_XFEM. .JONCOEF

3.3.10vector

.JONCOEF is of the same length than .JONFISS . It contains the list of the coefficients to associate with the level sets norms cracks of .JONFISS to find the field in which the connected crack is defined. These coefficients are worth or and $+1$ -1 are calculated automatically. The connected crack is then defined in the zone where for all the cracks mothers, i products JONCOEF (I) *lsn (JONFISS (I)) are negative. Figure



3.3.10 3.3.10-a of the levels set of the cracks mothers on the left, sign obtained after multiplication by JONCOEF on the right, and definition of connected crack on the side less. One ensures

thus that the connected crack is in the zone “less” compared to the cracks mothers . Figure 3.3.10 3.3.10-a an example of use of .JONCOEF and .JONFISS in order to define crack 3. On this figure crack 3 is connected on crack 2, it even connected on crack 1, .JONFISS of crack 3 contains D onc cracks 1 and 2. The signs of the level sets of cracks 1 and 2 are given on the figure of left: one from of deduced the values from JONCOEF as well as $(-1,1)$ the répartiti one of the sign of and on $coef(1)\times lsn(1)$ $coef(2)\times lsn(2)$ the figure of right. Crack 3 is defined in the zone where the 2 signs are less. In conclusion
.JONFISS and. J ONCOEF make it possible to define crack 3 with its level sets usual norm and tangent as well as the level set norms of the cracks mother S, to which one has elementary access of way. Objects

3.4 relating to propagation .PRO.

3.4.1 MES_EL concept

.PRO. MES_EL is a field by element (CHAM_ELEM) with a real component. It corresponds worthy of the element (either its volume for an element 3D or its area for an element 2D). .PRO.

3.4.2 NORMAL concept

.PRO. NORMAL is a field by element (CHAM_ELEM) containing the norms opposed n_i to the node in i the element. This information is used for rebootstrapping and reorthogonalisation by the method known as “SIMPLEXE” (see Doc. [R7.02.12] where the notion of norm opposed to a node is explained). That is to say

the ème i i element of the mesh, and the ème j j node of the element. V = i .PRO

.NORMAL (I); V (3* (

j-1) +1) Coordinated	according to x norm opposed to the node V (3* (j
j-1) +2) Coordinated	according to y norm opposed to the node V (3* (j
j-1) +3) Coordinated	according to z norm opposed to node .PRO. j

3.4.3 RAYON_TORE Vector

of realities length equalizes to 1. If
this vector exists in the SD, propagation of crack was calculated by means of the localization of the field of update of level sets. The field coincides with a torus in 3D or a circle in 2D built around the bottom of crack. The radius of this torus or this circle is stored in this object. If
this vector does not exist in the SD, all the model were used for the computation of the propagation of crack. .PRO.

3.4.4 NOEUD_TORE Vector

the Boolean ones length equal to the number of nodes of the mesh on which the crack was defined or, if a grid is associated with crack, to the number of nodes of the grid. If
this vector exists in the SD, propagation of crack was calculated by means of the localization of the field of update of level sets. For each node of the mesh (or grid), a value TRUE means that the node is understood in the field used for computation. For example, if the value of the component i ème of the vector is TRUE, the ème i node of the mesh (or the grid) is understood in the field; on the other hand, if the value of the component i ème of the vector is FALSE , the ème i node of the mesh (or the grid) is apart from the field. If
this vector does not exist in the SD, all the model (or all the grid) were used for the computation of the propagation of crack. .PRO.

3.4.5 NOEUD_PROJ Vector

the Boolean ones length equal to the number of nodes of the mesh on which the crack was defined. If
this vector exists in the SD, the computation of the propagation of crack made on auxiliary grid was associated with crack and the level sets calculated were projected on the mesh. For each node of the mesh, a value TRUE means that the node is understood in the field of projection. For example, if the value of the component i ème of the vector is TRUE, the ème i node of the mesh is understood in the field; on the other hand, if the value of the component i ème of the vector is FALSE , the ème i node of the mesh is apart from the field and the values of the level sets carried by this node were not put up to date. If
this vector does not exist in the SD, the computation of the propagation of crack were made directly on the mesh. .GRI.

3.4.6 MODEL Vector

of K8 length equalizes to 1. Name of
the model of the grid associated with crack. If this object does not exist, no grid is associated with crack. .GRI.

3.4.7 LTNO and .GRI . LNNO concept

.GRI. LTNO (resp . .GRI. LNNO) is a field at nodes (CHAM_NO) scalar which contains for each node of the grid the actual value of the level set tangent (resp. norm) with crack. If this object does not exist, no grid is associated with crack. .GRI.

3.4.8 GRLTNO and. GRLNNO concept

.GRI. GRLTNO (resp . .GRI. GRLNNO) is a field at nodes (CHAM_NO) with 3 real components. It contains for each node of the grid the values of the gradient of the level set tangent (resp. norm) in the 3 directions of space. That is to say

it - i ème i node of the grid, V =.GRI

.GRLTNO (I); V (1)

Value	of the following gradient of x the level set tangent, calculated with the ème i node V (2)
Value	of the following gradient of y the level set tangent, calculated with the ème i node V (3)
Value	of the gradient following Z of the level set tangent, calculated with the ème i node In 2D

, one has only 2 components according to and. x If y this
object does not exist, no grid is associated with crack. .FONDFISG

3.4.9 vector

.FONDFISG is a vector of realities containing the coordinates of the points of the crack tip builds on auxiliary grid. The points are ordered according to the same method as that described in the document [R7.02.12] for creation of the vector of the points of the crack tip on real mesh .FONDFISS , so that a curvilinear abscisse can be defined. If NFON is the number of points of the crack tip on the grid, then the length of vector .FONDFISG is 4xNFON . For each point of the crack tip on the grid, the first 3 components correspond to the 3 coordinates (in 3D) of the point, and the fourth component is its curvilinear abscisse. This object is created only in 3D in the frame of the crack propagation using the methods Upwind or Simplex like tools of update of level sets. Objects

3.5 relating to the contact These objects

are created in the sd_fiss_xfem by DEFI_ CONTACT. .BASCO

3.5.1 concept

.BASCO is a field at nodes (CHAM_NO) with 12 real components (in 2D and 3D). It contains the origin and the vectors of the COvariante Base of the facets of contact. For each node, the first three components are the coordinates of the point of contact associated with this node, which corresponds at the origin of the base covariante. The three following components are the coordinates of the 1st vector of the base. The three following components are cordonnées of the 2nd vector of the base. The three last components are cordonnées of the 3rd vector of the base. That is to say

the ème i i node of the mesh , V =.BASCO

(I); V (1)

Coordinated	according to point x of contact associated with the node V (2) i

Coordinated	according to point y with contact associated with the node V (3) i
Coordinated	according to point z with contact associated with the node V (4) i
Coordinated	according to 1st x vector with the base at the point with contact associated with the node V (5) i
Coordinated	according to 1st y vector with the base at the point with contact associated with the node V (6) i
Coordinated	according to 1st z vector with the base at the point with contact associated with the node V (7) i
Coordinated	according to 2nd x vector with the base at the point with contact associated with the node V (8) i
Coordinated	according to 2nd y vector base at the point of contact associated with the node V (9) i
Coordinated	according to 2nd z vector with the base at the point with contact associated with the node V (10) i
Coordinated	according to 3rd x vector with the base at the point with contact associated with the node V (11) i
Coordinated	according to 3rd y vector with the base at the point with contact associated with the node V (12) i
Coordinated	according to 3rd z vector with the base at the point with contact associated with the node In 2D i

, the components according to are z null. .LISEQ

3.5.2 (addresses JLIS1) It is

the List of the relations of equality (EQuality) between the unknowns of contact. Vector of integers length NRELEQ *2 where NRELEQ is the number of relations of equality. For each relation of equality, one stores the numbers of the 2 nodes belonging to the equality: Either IE the ème i relation of equality, ZI (JLIS
1-1+2* (IE-1) +1) is the number of the 1st node which is part of the equality. ZI (JLIS
1-1+2* (IE-1) +2) is the number of the 2nd node which is part of the equality. .LISEQ

3.5.3 _LAGR (addresses JLISLA) It is

the List of the numbers of Lagrange associated with the relations with equality (EQuality) between the unknowns with contact. This list is built in the case of meshes the multi-Heaviside. It makes it possible to find the number of d.o.f. of Lagrange to be associated with the number of a node taken in a relation of equality. ZI (JLISLA

- 1+2* (IE-1) +1) is the number of Lagrange to be associated with the 1st node of the relation of equality. ZI (JLISLA
- 1+2* (IE-1) +2) is the number of Lagrange to be associated with the 2nd node of the relation of equality. For example

if it is necessary to impose relation LAGS_ C=LAG3_C between the nodes "N26" and "N32" for the ième relation of equality, one will aurra: ZI (JLIS

```
1-1+2* (IE-1) +1) = 26 ZI (JLIS
1-1+2* (IE-1) +2) = 32 ZI (JLISLA
- 1+2* (IE-1) +1) = 1 ZI (JLISLA
- 1+2* (IE-1) +2) = 3 .CNCTE
```

3.5.4 (addresses JCNTES) It

acts of the groups of connected vital edges. For each local number of edge in a mesh, which belongs to a group, one increments NCTE and one stores: ZI (JCNTES

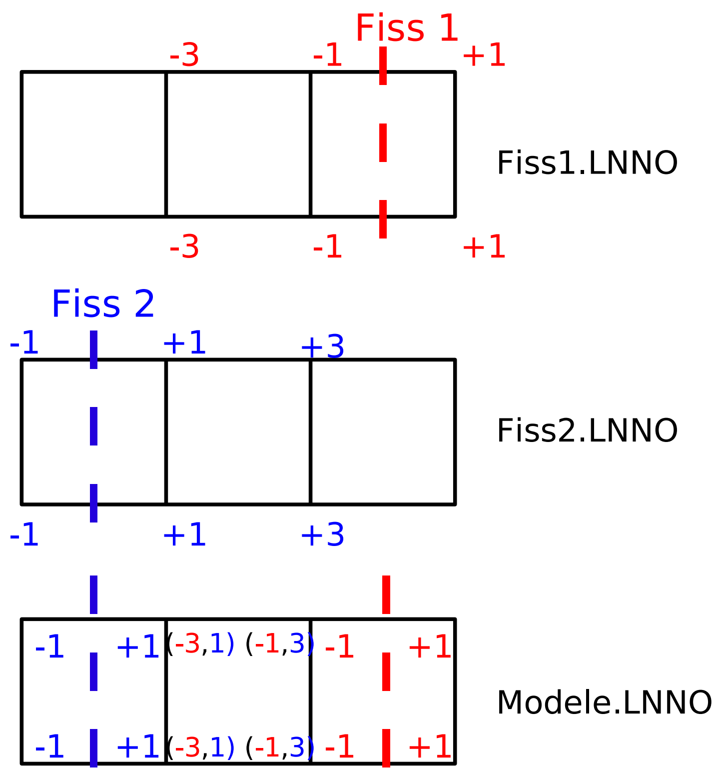
- `1+4* (NCTE-1) +1) the number` of the edge in the group, ZI (JCNTES
- `1+4* (NCTE-1) +2) the number` of group, ZI (JCNTES
- `1+4* (NCTE-1) +3) the number` of mesh, ZI (JCNTES
- `1+4* (NCTE-1) +4) the local` number of the edge in the mesh. Contents

4 of the objects of the sd_modele _xfem Contained

4.1 objects relating to concatenation .LNNO

4.1.1 , .LTNO , .BASLOC and. STNO Attention

, these objects do not have same structure as the objects of the same name contained in the sd_fiss _xfem ([§3.2 .2]3.2.2]3.2.4§3.3.3]3.3.3 is of the elementary fields of type ELNO and not of the CHAM_ NO the principal difference is that the CHAM_NO store information on the nodes of the mesh (a value by node) whereas CHAM_ ELNO store information by node on the element. A node being connected to elements N , the value will be stored time. N This choice has as a finality the information storage of several cracks in an element, one then uses the notion of under point. If an element is contained in the SD fiss (I)/".MAILFISS.HEAV", it is said that he is seen by crack I: it is then necessary to store in the model the relative information with this crack. If the element sees 2 cracks or more, it is necessary to store in the model information of all cracks. By defining the number of under points of $NBSP$ CHAM_ ELNO like the number of cracks seen by the element, one duplicates time $NBSP$ the memory capacity relating to the SD for the element which sees cracks $NBSP$. Thus one can store information of all cracks in this element. Figure 4.1.1 4.1.1-a an example of concatenation of 2 cracks in the model. One represents on this figure the SD .LNNO of 2 cracks and that of the model. The element of the center sees 2 cracks, the MODEL field thus contains 2 components per node in this element. Figure



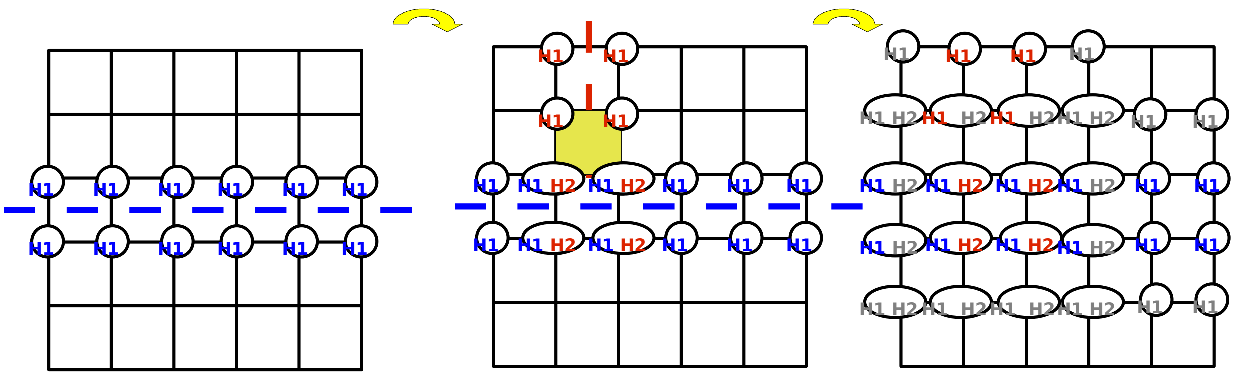
4.1.1 4.1.1-a of multiple storage of SD in the model. .NOXFEM

4.1.2 It

acts of a CHAM_NO `with 2` components. The components are affected if the node has an enrichment X-FEM (i.e there exists a crack fiss (I) `for which` the field fiss (I)/ `/‶.STNO″ is not` null in this node). The first component corresponds to the number of mesh X-FEM containing the node. The second component is the local number of the node in this mesh X-FEM. This structure is useful for the imposition of limiting conditions of Dirichlet type. .FISSNO

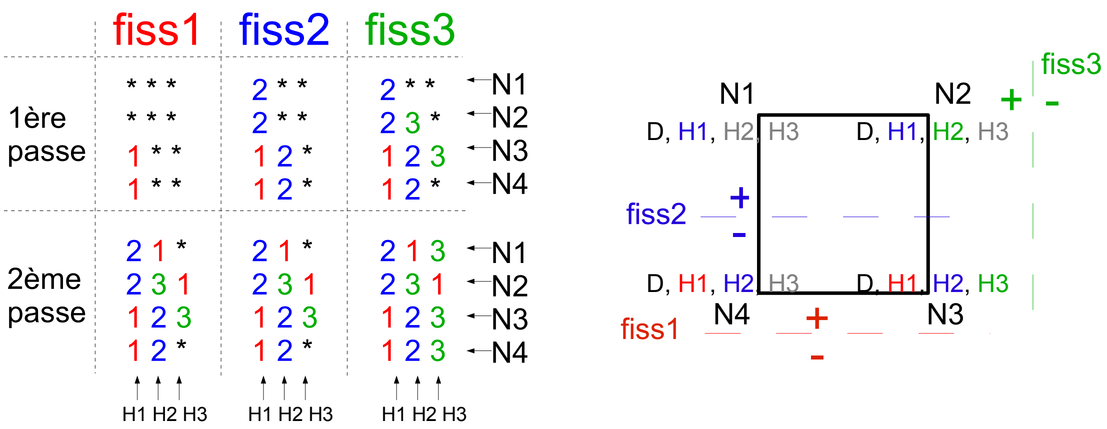
4.1.3 It acts

of a CHAM_ELNO `. This SD` is created when at least an element of structure must store information of several cracks. For the introduction of several Heaviside degrees of freedom, one made the choice sequentially to increment by nodes the numbers of degrees of freedom. For example figure 4.1.3-a, 4.1.3-a two cracks. One affects for first crack (in blue on the left) the d.o.f. For $H1$ second crack (in red in the center) one allots degrees of freedom to the nodes $H1$ which do not have already degrees of freedom affected $H1$ previously, and degrees of freedom for $H2$ the nodes having degrees of freedom already. This choice $H1$ is made with an aim of not weighing down the elementary catalogs. Indeed this strategy does not require an element having only Des. Appears $H2$



4.1.3-a 4.1.3-a Degrees of freedom, for two cracks which overlap However

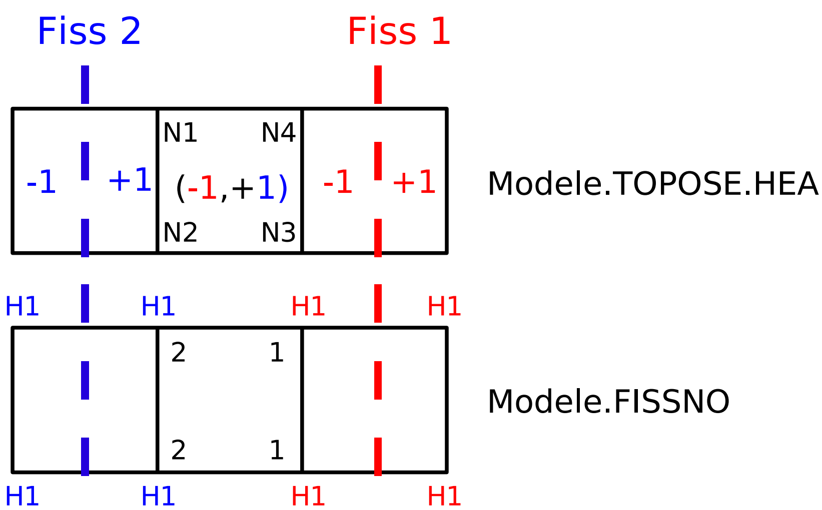
it is necessary to pay attention, because in certain elements, the incrementing of cracks does not ensure a direct link between the number of total Heaviside degree of freedom and the crack number (In the example, can correspond $H1$ to crack 1 or crack 2 in the yellow element). During the assembly in TE, it is thus necessary to be able to associate the degrees of freedom (with $H_{IFH}[X,Y,Z]$ the number IFH of total Heaviside degree of freedom) of the node with the crack INO right number. Moreover elements must be supplemented by Heaviside degrees of freedom which are then eliminated (degrees of freedom in gray on the figure of right). It is necessary to eliminate the degrees of freedom from the node $H_{IFH}[X,Y,Z]$ to have INO the statute of good crack in. The SD INO MODELS
/ `‶.FISSNO″ makes it possible` to do that. It turns over the crack number considering locally $IFISS$ to the node for INO the degrees of freedom. One A. $H_{IFH}[X,Y,Z]$ One then $IFISS=FISSNO(INO,IFH)$ recovers the good pointer for the SD MODELS `/‶[.LNNO, .STNO .TOPOSE.HEA]. Appear`



4.1.3-b 4.1.3-b of construction of the SD Modèle.FISSNO the figure

4.1.3-b 4.1.3-b the construction of table .FISSNO. `The lines` correspond to the numbers of node, and the column with the numbers of Heaviside degree of freedom. One fills out the table in a sequential way on the 2 ways. At the time of the first master key, one looks at statute (MODELE.STNO) `node` for INO crack. If $IFISS$ the statute is not null, one fills the first box by the data in their IFH order with path to which INO was not already assign with. One supplements $IFISS$ the table so necessary at the time of the second master key while proceeding as for the first master key but by filling the box by the data in if IFH the statute INO is null. To illustrate

the use of this SD, one takes again the simpler example of the figure 4.1.1-a 4.1.1-a 4.1.3-c 4.1.3-c built Modele.FISSNO `for the element` of the medium. If one wants (during the assembly for example) to recover the Heaviside function associated with the node for $N1$ the degrees of freedom. One recovers $H_1[X,Y]$ the crack number via. There is then $IFISS=FISSNO(INO=1, IFH=1)=2$ the Heaviside function which is worth. For $HE(IFISS=2)=+1$ the node, one will have $N4$ and thus $FISSNO(4,1)=1$. $HE(IFISS=1)=-1$



More generally 4.1.3-c 4.1.3-c Example for the SD Modèle.FISSNO

, the Heaviside function under element of integration, for SE a node and a Heaviside INO degree of freedom is obtained IFH in the following way: . This

$HE(SE, INO, IFH) = HE(SE, IFISS(INO, IFH))$ data structure

also makes it possible to remove the Heaviside degrees of freedom in excess in. One recovers TE statute (MODELE.STNO) node for INO the Heaviside degree of freedom, whereas IFH this structure is stored by crack : . This

$STATUT(INO, IFH) = STATUT(INO, IFISS(INO, IFH))$ structure

is also used for postprocessing X-FEM, during the interpolation of displacements on the lips of crack. .HEAVNO

4.1.4 It acts

of opposite structure of .FISSNO. This SD is created when at least an element of structure must store information of several cracks. It is built at the same time as .FISSNO and turns over the number of Heaviside degree of freedom associated with the node and INO crack. By keeping $IFISS$ the notations of the paragraph preceding one a: In the case of

$IFH = HEAVNO(INO, IFISS)$

the contact multi-Heaviside, the number of degree of freedom of Lagrange associated with a crack corresponds $IFISS$ to the number of corresponding Heaviside degree of freedom IFH Thus this data structure makes it possible it to find the name of the degree of freedom of Lagrange (and its position in the elementary matrix/vector) starting from the crack number. If one treats the contact on a facet of contact associated with the crack, the Lagrangian one $IFISS$ which works with the node carries INO the number: Contained

$ILAG = HEAVNO(INO, IFISS)$

4.2 objects relating to the topology of the subelements Notes with

the developers, certain values are in ma tough dan FORTRAN (te0514
, xbsigm , xmel3 D...) Recall on

the way in which the elements X-FEM are cut out (R7.02.12 Document left 3.3 Pennies cutting): 3D 2D Cutting

(1) preliminary Phase of cutting Each element	(hexahedron, pentahedron) is virtually cut out in tetrahedrons 6 tetrahedrons to the maximum Each element	(quadrangle) is virtually cut out in triangles 2 triangles to the maximum Cutting
(2) Under-cutting Each tetrahedron	is in its turn cut out under-tetrahedrons 6 under-tetrahedrons to the maximum Each triangle	is in its turn cut out under-triangles 3 under-triangles to the maximum Points D
” intersection Points D	“intersection between the edges of the tetrahedrons resulting from cutting (1) and crack 11 surfaces it points	“intersection between the edges of the triangles resulting from cutting (1) and the curve crack 3 points

	D” intersection to the maximum Points D	D” intersection to the maximum Points QUADRATIC
ELEMENTS	mediums NON MILKED Points in the middle of	the edges of the triangles resulting from cutting (2) [on condition that coinciding with no node of the initial element] 10 points mediums to the maximum Point central
of the quadratic quadrangle QUADRATIC	ELEMENTS NON MILKED Point of intersection	of the two right diagonals of a quadratic quadrangle 1 central point to the Process maximum
of cutting (1) and under cutting (2) 1 element	X-FEM 3D tetrahedron n°1 6 under-tetrahedrons with the max. tetrahedron n°2 (2) 6 pennies to max.… the 1 element - tetrahedr ons	X-FEM 2D triangle n°1 (2) 3 pennies with max.… the Note:

- triangles

: the number of under elements is not modified by the introduction of meshes quadratic. .TOPOSE.

4.2.1 PIN Informs

about the Points D "Intersection. C” is
a constant field by element (CHAM_ELEM) `with 33 real` components. For each element, this field comprises the coordinates of the points of intersection (with the meaning previously definite). As the maximum number of such points is 11, and considering one works in dimension 3, the number of components by element is dimensioned to 33. For each element, the components are ordered like this: where are $\{x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_{11}, y_{11}, z_{11}\}$ x_i, y_i, z_i the coordinates of the ème point of intersection i on the element. It will be noted
that if the element is not fissured, the components are null. In 2D, one

has to the maximum 3 points of intersection, with 2 components each one, therefore 6 components with the total. .TOPOSE.

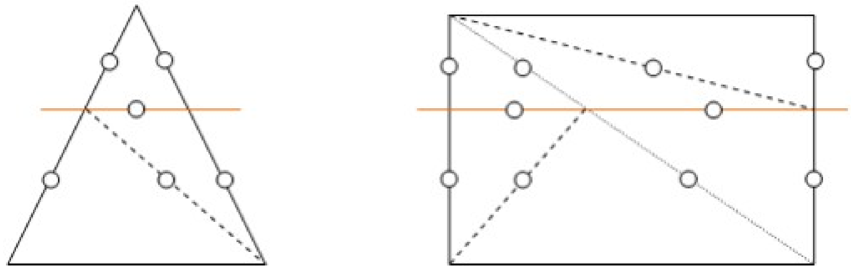
4.2.2 PMI Informs

about the Points Mediums in the presence of meshes quadratic. 2D It

4.2.2.1is

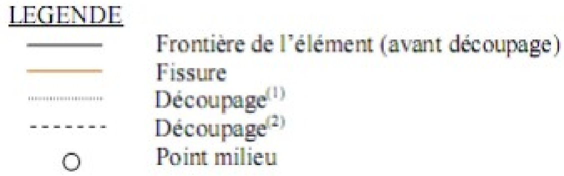
a constant field by element (CHAM_ELEM) `with 22 real` components. For each element, this field comprises: - the coordinates

of the points mediums Note



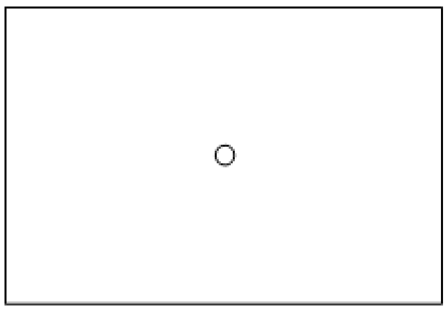
Points milieux d'un TRIA6 (6 au maximum)

Points milieux d'un QUAD8 (10 au maximum)



: it is pointed out that the coordinates of the points mediums which coincide with a node of the element relative are not stored in this data structure. - the coordinates

of the central point during introduces the cutting (1) of a QUAD8: Point central



of a QUAD8 (1 at the most
) That makes

a total of 11 points to the maximum. Like one places oneself in 2D, one can have component $11\times2=22$ to the maximum. The field with 22 thus is dimensioned. For each

element, the components are ordered like this: where are

$$\{x_1, y_1, x_2, y_2, \ldots, x_{11}, y_{11}\}$$

$(x_i, y_i)_{1\leqslant i\leqslant 10}$ the coordinates of the ème not i medium on the element and where are (x_{11}, y_{11}) the coordinates of the central node of a QUAD8. .TOPOSE.

4.2.3 CNS Informs

about the Connectivity of the Subelements . 3D It

4.2.3.1is

a constant field by element (CHAM_ELEM) with 128 whole components. For each element, this field contains the connectivity of the subelements of the element. By convention, all the subelements are tetrahedrons with 4 nodes (TETRA4), and there can be to the maximum 32 per element X-FEM (for a detailed explanation of this result, to see documentation of reference on the method X-FEM [R7.02.12]). There are to the maximum 6 tetrahedrons and 6 subelements per tetrahedron, but there is no configuration with 6 subelements for each tetrahedron, this is why the maximum is of 32. The components are arranged 4 by 4. Object .TOPOSE. CNS refers to 2 types of tops: either of the already existing nodes of the mesh, or of the points of intersection. Thus, for each element: - if
a top of a subelement is a node of the element X-FEM in question, then this node is located by its local number in the element; - if
a top is a point of intersection, then this point is located by a number higher than 1000, where the figure of the units returns to its local number in the list of the points of intersection of the element. Example:

For
element IE: CNS (IE)
= 1001 1005

6 1002	1003		3 1001	2	…	1004		4 5	6 1st		pen nies	

- will tétra : top N °1: 1001 - > at the 1st point of PIN (IE) top N °2
returns: 1005 - > at the 5th point of PIN (IE) top N °3
returns: 6 - > returns to the 6th node of element IE top N
°4: 1002 - > returns at the 2nd point of PIN (IE) 2nd pennies

- will tétra : top N °1: 1003 - > at the 3rd point of PIN (IE) top N °2
returns: 3 - > returns to the 3rd node of element IE top N
°3: 1001 - > at the 1st point of PIN (IE) top N °4
returns: 2 - > returns to the 2nd node of element IE. 2D It

4.2.3.2is

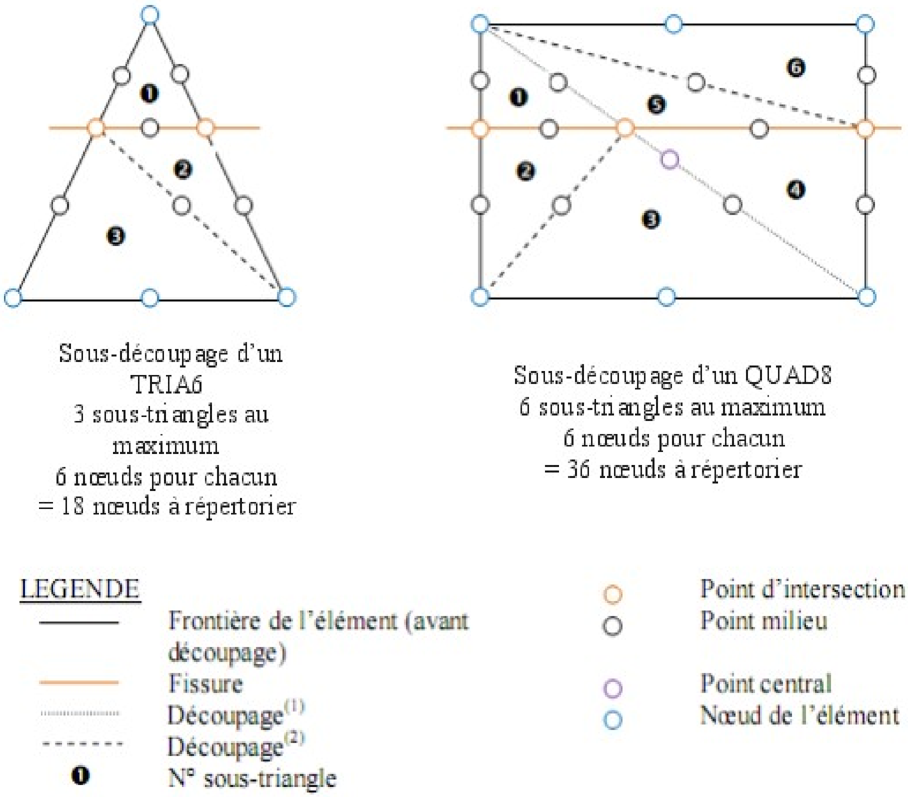
a constant field by element (CHAM_ELEM) with 36 whole components. For each element, it contains the connectivity of under elements of the element. One counts

: 6 under-triangles

– to the maximum (reached for a mesh quadrangle) 6 nodes

– to the maximum for each under-triangle (reached for a quadratic mesh) What makes

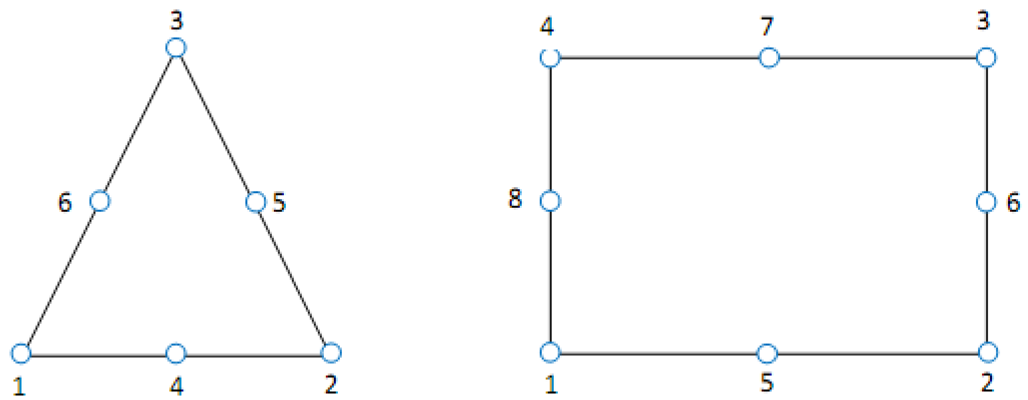
component $6\times6=36$ to the maximum. The field is thus dimensioned to 36 and one arranges these last 6 by 6. Object



.TOPOSE. CNS refers to 4 types of points: □ Standard

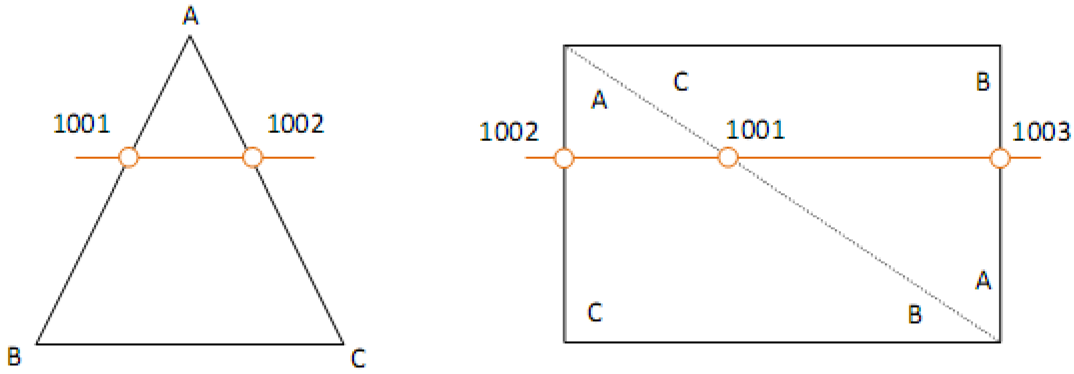
□ Node of the element (belonging to mesh) It is located

by its local number in the Standard element □



□□Point of intersection It is located

by a number, where returns $1000+u$ u to its local number in the list of the points of intersection of the element. 1001 →

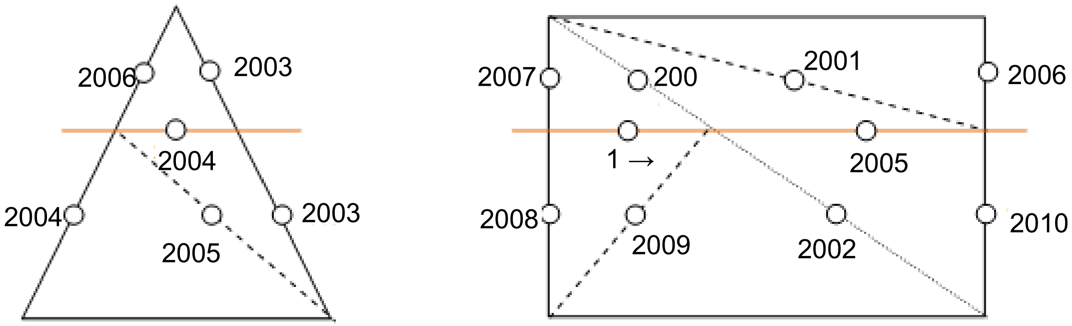


1st point stored in PINTTO 1002 →

2nd point stored in PINTTO… `Standard`

Not medium It is located

by a number, where returns $2000+u$ u to its local number in the list of the points mediums of the element. 2001 2002

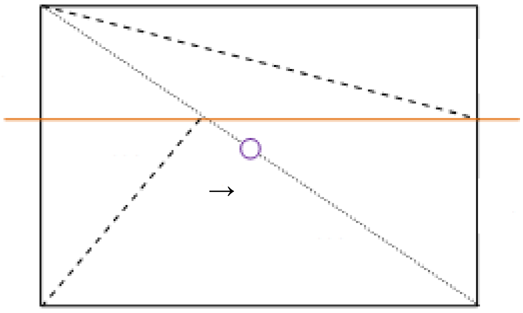


1st point stored in PMILTO 200 2 →

2nd point stored in PMILTO, `Standard etc`

Not medium quadrangle It is located

by a number, where returns $3000+u$ u to its local number in the list of the points mediums of the element. 3011 3011



11th point stored in PMILTO. TOPOSE.

4.2.4 HEA Informs

about the value of the Heaviside function . It is
a constant field by element (CHAM_ELEM) with 36 whole components corresponding to the value of the Heaviside function on the under-tetrahedrons (+1 or – 1), arranged by group of 6 values per tetrahedron: There can

$$\{H_1^1, H_2^1, H_3^1, 0, 0, H_1^2, H_2^2, H_3^2, H_4^2, H_5^2, H_6^2, H_1^3, H_2^3, \ldots, H_6^6\}$$

be of the 0 if the number of subelements (NSE) < 6 on a tetrahedron (as on the 1st tetrahedron of the example above). In 2D,

the subelements are with the maximum number of 6, field TOPOSE.HEA will thus have only 6 components arranged by groups of 3, by element. In

the case or the element sees several cracks, this field is duplicated as many times as there are cracks. One then uses the number of under points defines $NBSP$ in [§4.1.1]. 4.1.1 will be components $36 \times NBSP$ in 3D and components $6 \times NBSP$ in 2D. .TOPOSE.

4.2.5 LON Informs

about the value Length of fields used to define the subelements. It is
a constant field by element (CHAM_ELEM) with 8 whole components the 1st
component is the number of tetrahedrons (NIT) contents in the element X-FEM (6 if the element is a hexahedron, 3 if it is a pentahedron, 1 if it is a tetrahedron). In 2D,
the 1st component is the number of triangles contained in the element X-FEM (2 if the element is a quadrilateral, 1 if it is a triangle). The following
component NIT are the number of subelements (NSE) resulting from each tetrahedron. One must have: In 2D, this

$$\sum_{i=1}^{NIT} NSE^i \leq 32$$

sum is to the maximum of 6. The 8th
component is the number of points of intersection between the edges of the tetrahedrons and the crack (normally 11 in 3D $\leq$ and 3 in 2D). Objects relating

4.3 to the topology of the facets of contact fields

.TOPOFAC .PI, .TOPOFAC .AI, .TOPOFAC .CF, .TOPOFAC .LO and .TOPOFAC .BA, are duplicated by element as many times as the number of cracks seen by the mesh. One uses for that number of subpoints which corresponds to. Attention n_{fiss}

, the size of certain objects described in this part is difference between Heaviside element and element which can contain the crack tip (ace-tip or mixed), in 3D. One marks these objects by a star (*) beside their name. .TOPOFAC

4.3.1 .LO Informs

over the Length of the fields used for the contact. It is
a constant field by element (CHAM_ELEM) with 2 whole components, containing the number of points of intersection (NINTER) and the number of triangular facets of contact (NFACE). .TOPOFAC

4.3.2 .PI (*) Informs

about the Points D "Intersection. C" is
a constant field by element (CHAM_ELEM) with 18 or 21 real components, which are the coordinates within the space of **reference relative of the points of intersection** of the edges of the element with crack. For
the Heaviside elements, there are to the maximum 6 points of intersection in 3D, therefore the field is dimensioned to 18. For
the elements ace-tip or mixed, there are to the maximum 7 points of intersection in 3D, therefore the field is dimensioned to 21. That is to say

the ème i element i of the mesh , V =.TOPOFAC

.PI (I); V (1) Coordinated

reference following of the 1st point of intersection x V (2) Coordinated
reference following of the 1st point of intersection y V (3) Coordinated
reference following of the 1st point of intersection z V (4) Coordinated
reference following of the 2nd x point of intersection V (5) Coordinated
reference following of the 2nd y point of intersection ……
V (18) Coordinated
reference following of the 6th z point of intersection If the number

of points of intersection is strictly lower than the maximum number of points of intersection (6 or 7 following cases), the vector is supplemented by 0. The number of points of intersection east contained in vector .TOPOFAC .LO (see § 4.3.1). In4.3.1 linear

, one has to the maximum 2 points of intersection with each one 2 components, that is to say 4 real components. In 2D quadratic, one has to the maximum 2 points of intersection with each one 2 components plus the point medium of the interface on the element with 2 components, that is to say 6 real components. .TOPOFAC

4.3.3 .AI (*) Informs

about the Intersected Edges . It is
a constant field by element (CHAM_ELEM) with 30 or 35 real components containing of information on the intersected edges. For each
of the 6 or 7 points of intersection of each element: The 1st
component is the local number of the corresponding edge (0 if it is a top node). The 2nd
component is the local number of the node if it is a top node (0 if not). The 3rd
component is the length of the edge. The 4th
component is the position of the point of intersection on the edge, with an arbitrary meaning which depends of the mesh. This reality strictly lies between 0 and 1, and is worth 0 if the point of intersection is a top node. The 5th
component makes it possible to know if the intersected edge is vital or not. If it is vital, the component is worth 1, if it is not it, the component is worth 0. The number of components

by point of intersection (5) is accessible in FORTRAN by a call to function XXMMVD (" ZXAIN") That is to say

the ème i element i of the mesh , V =.TOPOFAC

.AI (I); V (1) local

Number of the edge corresponding to the 1st point of intersection V (2) local
Number of the node corresponding to the 1st point of intersection V (3) Length
of the edge corresponding to the 1st point of intersection V (4) Position
of the 1st point of intersection on the edge V (5) 1st
point of intersection : so on vital edge (1) if not (0) V (6) local
Number of the edge corresponding to the 2nd point of intersection ……
V (30) 6th
vital point of intersection (1) or not (0) If the number

of points of intersection is strictly lower than the maximum number of points of intersection (6 or 7 following cases), the vector is supplemented by 0. In 2D, there

are to the maximum 3 points (2 1 and points of intersection point medium) with in the same way that in 3D 5 information (number of the edge, number of the node, length of the edge and position of the point of intersection, vital intersected edge or not). Field TOPOFAC. AI thus has 15 components per element. .TOPOFAC

4.3.4 .CF Informs

about the Connectivity of the Facets of contact. It is
a constant field by element (CHAM_ELEM) with 15 whole components, containing the connectivity of the tops of the triangular facets. There are to the maximum 5 facets per element. That is to say

the ème i element i of the mesh , V =.TOPOFAC

.CF (I); V (1) local

Number of the 1st top of the 1st triangular facet V (2) local
Number of the 2nd top of the 1st triangular facet V (3) local
Number of the 3rd top of the 1st triangular facet V (4) local
Number of the 1st top of the 2nd triangular facet V (5) local
Number of the 2nd top of the 2nd triangular facet……
V (15) local
Number of the 3rd top of the 5th triangular facet If the number

of triangular facets is strictly lower than 5, the vector is supplemented by 0. In 2D linear

, there is only 1 facet of contact, which is a segment and thus has 2 tops. The vector thus has only 2 components. In 2D quadratic, there is 1 facet of contact, which is a parabola at 2 tops and a point medium. The vector thus has 3 components. .TOPOFAC

4.3.5 .BA Informs

on the covariante Basis facets of contact. It is a field by element (CHAM_ELEM) with 54 or 63 real components which are the coordinates of the vectors of the base covariante for each $(n, \tau_{1,} \tau_2)$ of the 6 or 7 points of intersection. For each element (the exhibitor who refers to the number of the point of intersection): That is to say

$$\{n_x^1, n_y^1, n_z^1, \tau_{1x}^1, \tau_{1y}^1, \tau_{1z}^1, \tau_{2x}^1, \tau_{2y}^1, \tau_{2z}^1, n_x^{2,} \ldots, \tau_{2z}^6\}$$

the ème i element i of the mesh , and the number j of the point of intersection V =.TOPOFAC

.BA (I); V (9* (j-1

) +1) Coordinated	according to 1st vector x of the base to the ème point of intersection j V (9* (j-1
) +2) Coordinated	according to 1st vector y of the base to the ème point of intersection j V (9* (j-1
) +3) Coordinated	according to 1st vector z of the base to the ème point of intersection j V (9* (j-1
) +4) Coordinated	according to 2nd x vector of the base to the ème point of intersection j V (9* (j-1
) +5) Coordinated	according to 2nd y vector of the base to the ème point of intersection j V (9* (j-1
) +6) Coordinated	according to 2nd z vector of the base to the ème point of intersection j V (9* (j-1
) +7) Coordinated	according to 3rd x vector of the base to the ème point of intersection j V (9* (j-1
) +8) Coordinated	according to 3rd y vector of the base to the ème point of intersection j V (9* (j-1
) +9) Coordinated	according to 3rd z vector of the base to the ème point of intersection j If the number

of points of intersection is strictly lower than the maximum number of points of intersection (6 or 7 following cases), the vector is supplemented by 0. In 2D linear

, the base covariante consists of 2 vectors having each one (n, τ_1) 2 components and there are 2 points of intersection. The number of components is 8. In 2D quadratic, the base covariante consists of 3 vectors having each one (n, τ_1) 2 components and there are 2 1 and points of intersection point medium. The number of components is 12. TOPOFAC.

4.3.6 HE Informs

about the value of the Heaviside functions on both sides of the facet of contact. This SD is useful only in the case of multi-Heaviside elements. It is not created so the model does not have such elements. That is to say the number n_{fiss} of cracks seen by the mesh. Number of subpoints for this object is then (for each n_{fiss}^2 crack, one will want to know the values coming from other cracks). The number of components is of with $2 \times n_{face}$ the preset n_{face} maximum number of facets of contact is in 2D and $n_{face}=1$ 3D. That is to say $n_{face}=5$

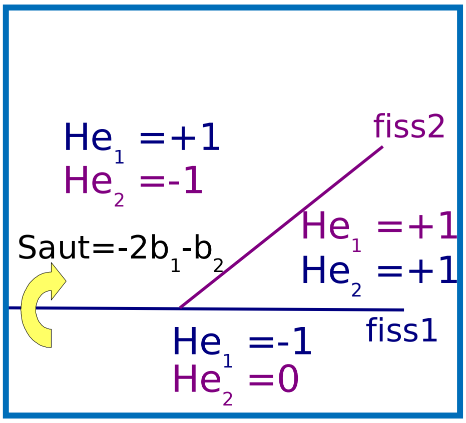
the ème i element i of the mesh , the number j of facet for crack V =.TOPOFAC k

.HE (I); V (2*nface

* (nfiss* (k-1) +1-1) +2* (j-1) +1) Value of	the Heaviside function of crack 1 on the side slave of the facet V (2*nface
* (nfiss* (k-1) +1-1) +2*	the Heaviside function of crack 1 on the main side of the

(j-1) +2) Value of	facet V (2*nface
* (nfiss* (k-1) +2-1) +2* (j-1) +1) Value of	the Heaviside function of crack 2 on the side slave of the facet V (2*nface
* (nfiss* (k-1) +2-1) +2* (j-1) +2) Value of nface	the Heaviside function of crack 2 on the main side of the facet… V (2*
* (nfiss* (k-1) +nfiss-1) +2* (j-1) +1) Value of	the Heaviside function of the crack nfiss on the side slave of the facet V (2*nface
* (nfiss* (k-1) +nfiss-1) +2* (j-1) +2) Value of	the Heaviside function of the crack nfiss on the main side of the facet This SD

is useful to compute: the jump of displacement between the sides Master and slave in the case of the multi-Heaviside elements. In this case the jump does not depend solely on the Heaviside function associated with crack cutting out the facet but on all the functions Heaviside (one by crack) present in the element. An example is shown on the figure 4.3.6-a. 4.3.6-a



appear 4.3.6-a of the jump a junction. The jump associated with crack 1 also depends on the Heaviside function associated with crack 2. .TOPOFAC

4.3.7 .OE (*) It acts

of a field of which the data-processing structure is rigorously the same one as TOPOFAC. Pi. As this last it contains the coordinates of the points of intersection in real space, but in **the initial configuration. This field** serves in the frame of the approach great slidings with X-FEM. .TOPOFAC

4.3.8 .GE (*) It acts

of a field of which the data-processing structure is rigorously the same one as TOPOFAC. OE. As this last it contains the coordinates of the points of intersection in real space, but brought up to date **according to the kinematics of the facets slaves. This field** serves in the frame of the approach great slidings with X-FEM. .TOPOFAC

4.3.9 .GM (*) It acts

of a field of which the data-processing structure is rigorously the same one as TOPOFAC. OE. As this last it contains the coordinates of the points of intersection in real space, but brought up to date **according to the kinematics of the facets Masters. This field** serves in the frame of the approach great slidings with X-FEM. Other objects

4.4 .XFEM_CONT

4.4.1 Vector

of integers length equalizes to 1. This value is worth 0 if the contact is not defined on any the cracks X-FEM, and is worth 1 if not (contact on at least a crack X-FEM) .NFIS Vector

4.4.2

of integers length equalizes to 1. It corresponds to the number of cracks X-FEM of the model. .FISS Vector

4.4.3

of K8 length equal to the number of cracks X-FEM of the model. It corresponds in the name of the sd_fiss_xfem `of each` crack. .XMAFIS

4.4.4 CHAM_ELEM

`of K8.` For `each` mesh, this CHAM_ELEM `corresponds` in the name of cracks seen by this mesh, in the order defined by the user. Number of subpoints of this SD corresponds to the number of cracks seen by the mesh. Contained

5 objects of the sd_contact _xfem .CARACF

5.1 (addresses IAD=JCMCF +ZCMCF* (IZONE-1)) ZR (IAD+1

) = diagram of integration (=1 if INTEGRATION = "GAUSS" (FPG12) =4 if INTEGRATION
= "FPG4" =6 if INTEGRATION
= "FPG6" =7 if INTEGRATION
= "FPG7") ZR (IAD+2
) = COEF_ REGU _CONT ZR (IAD+3
) = COEF_ REGU _FROT ZR (IAD+4
) = coefficient of Coulomb for friction ZR (IAD+5
) = coefficient characterizing the presence of friction and its type (=1 if FROTTEMENT
= ' SANS' =3 if FROTTEMENT
= ' COULOMB') ZR (IAD+6
) = initial value of the threshold of friction (SEUIL_INIT) ZR (IAD+23
) = coefficient of setting at the level of contact pressures (COEF_ECHELLE) ZR (IAD+24
) = algorithm of restriction of the space of the Lagrange multipliers (=0 if ALGORITHME
_LAGR = "NON" =1 if ALGORITHME
_LAGR = "VERSION1" =2 if ALGORITHME
_LAGR = "VERSION2") .ECPDON

5.2 (addresses IAD=JECPD +ZECPD* (IZONE-1)) ZI (IAD +1

) = indicating of axi-symmetry (=0 because MODL _AXIS = ' NON') ZI (IAD+2
) = ITER_CONT_MAXI ZI (IAD+3
) = ITER_FROT_MAXI ZI (IAD+4
) = ITER_GEOM_MAXI (=1 because it N "does not have there a geometrical loop) ZI (IAD+5
) = indicating of the contact initial (=0 if CONTACT_INIT=" NON' =1 if CONTACT
_INIT=' OUI') .METHCO

5.3 (addresses IAD=JMETH +ZMETH* (IZONE-1)) ZI (IAD +6

) = number of method used (=11 if
continuous method =12 if
continuous method with contact slide) .XFIMAI

5.4 Vector

of K8 length equalizes with the number of contact zones. For each contact zone, is stored the main name of crack (name of the sd_fiss_xfem). .XNRELL

5.5 Vector

of K24 length equalizes with 3 times the number of cracks. For each crack, one stores the name of the objects for relations LBB of the sd_fiss_xfem (see [§ 3.5.2])3.5.2

- /".LISEQ", Objects related

5.6 to SD RESOCO relative to the great slidings It acts

of the large cards of contact sliding X-FEM which is built once the pairing made between the points of integrations of the elements slaves and Masters. For each pairing, a late element is created in operator STAT_NON_LINE . One stores in these various fields the relative information with these useful late elements during the integration of the contributions of contact-friction. ".XFPO"

5.6.1 Card of

R which stores for each late element of information on pairing (see the comments in routine XMCART for more details). ".XFST"

5.6.2 Card of

I which stores in the order of the nodes of the late element the statutes of enrichment of the nodes of the elements slave and Master of the late element. It corresponds to a copy of the SD MODELS /".STNO" for the elements slave and Master of the late element. ".XFPI"

5.6.3 Card of

R which stores the local coordinates of the points of intersection of the element slave. It corresponds to a copy of the SD MODELS /".TOPOFAC.PI" for the element slave of the late element. ".XFAI"

5.6.4 Card of

R which stores information on the intersected edges of the element slave. It corresponds to a copy of the SD MODELS /".TOPOFAC.AI" for the element slave of the late element. ".XFCF"

5.6.5 Card of

I which stores information on the connectivity of the facets of the element slave. It corresponds to a copy of the SD MODELS /".TOPOFAC.CF" for the element slave of the late element. ".XFHF"

5.6.6 Card of

I which stores the value of the Heaviside functions on both sides of discontinuity, if the element slave or Master is multi-Heaviside. It corresponds to a copy of the SD MODELS /".TOPOFAC.HE" side slave of the element slave and main side of the main element for the late element. ".XFPL"

5.6.7 Card of

I which stores core of Lagrange of contact to the nodes of the element slave, if the element slave or Master is multi-Heaviside. It corresponds to a copy of the SD MODELS /".HEAVNO" for crack associated with the contact zone, for the element slave of the late element.