Note of use of the contact in *Code_Aster*

Summary:

This document describes the approach to be followed for the taking into account of conditions of contact-friction in the nonlinear studies. Initially, one points out what means to take into account contact-friction in mechanics of the structures, then one traces the broad outlines of a problem of contact in *Code_Aster* : pairing and resolution.

The definition of the contact is carried out with the order DEFI_CONTACT while the resolution is done with the orders STAT_NON_LINE or DYNA_NON_LINE. One formulates recommendations for the parameterization of pairing and the choice of the methods of resolution in these operators.

Finally various methodologies are evoked (contact with a rigid surface, to recover a contact pressure in postprocessing, great deformations and contact, movements of rigid bodies blocked by the contact,…). They make it possible to overcome the difficulties frequently encountered in the studies. In this section, are also approached alternative modelings of the phenomenon of contact-friction by elements of joints or elements discrete (through the law of behavior).

Contents

1 Introduction

1.1 Object of this document

To say that two solid bodies put in contact do not interpenetrate but that on the contrary a reciprocal effort is exerted one on the other and that this effort disappears when the bodies are not touched any more, concerns the good sense. It is the briefest definition which one can make of the problem of "contact": however to enforce these conditions in a computer code of the structures like *Code_Aster* request much for efforts.

To solve the problem of contact, it is finally to impose a boundary condition on certain degrees of freedom of displacement and to find an unknown factor additional, the reciprocal effort being exerted between the two bodies.

The difficulty comes from the strong non-linearity induced by this "pseudo-condition in extreme cases". Indeed the condition to be imposed on displacements (to prevent any interpenetration) depends it even on displacements (which will determine in which point surfaces make contact).

Non-linearity due to the taking into account of contact is separate in *Code_Aster* in two points:

- non-linearity of contact (- friction): it rises from the conditions of contact (- friction) which are not univocal. One calls here on an algorithm of optimization under constraints or with a transformation of the problem into a not forced version.
- geometrical non-linearity: it rises from the great relative slips likely to occur between surfaces in contact (ignorance *a priori* effective final surfaces of contact). One calls here on an algorithm of fixed point or Newton coupled to a geometrical research.

In *Code_Aster*, in the presence of contact, the user must *has minimum* to identify potential surfaces of contact. The technique of resolution rests then on two fundamental stages:

- Phase of pairing: it makes it possible to treat geometrical non-linearity as a succession of problems in small slips (where the problem is geometrically linear). The technique to determine effective surfaces of contact and the advices of parameter setting of this phase are given to the section 2.
- Phase of resolution: it makes it possible to solve the problem of optimization under constraints related to the non-linearity of contact and possibly of friction. The various algorithms of optimization available are presented in the section 3. One gives a advance to it to choose an algorithm adapted to his case of study.

It is essential to have understood that contact-friction is a non-linearity except for whole as well as non-linearities materials (law of nonlinear behavior) and kinematics (great displacements, great rotations). She thus asks at the same time to know the bases of the theory of the contact and to understand the treatment of this one in *Code_Aster* in order to make the good choices of modeling (grid and setting in data).
This document is there to assist the user in these choices.

1.2 A question of vocabulary

In order to facilitate the reading, one gives here some of the terms abundantly used in this document.
When one speaks about contact mechanics, one uses two characteristic sizes:

- often noted game g or d. It characterizes the distance signed between two surfaces of contact;
- density the effort of contact p. It is the reciprocal effort exerted by a solid on the other when the game is closed (null). It is carried by the normal on the surfaces of contact. One will also wrongly use the term of contact pressure.

These sizes intervene under the conditions of Hertz-Signorini-Moreau of respect of the contact (*cf.* [R5.03.50]).

In the presence of friction, one introduces in addition:
- direction of slip $\vec{t}$
- density the effort of friction τ, carried by $-\vec{t}$.

In *Code_Aster*, one uses a criterion of friction of Coulomb, the conditions of friction are described in [R5.03.50].

1.3 Alternative modelings of contact-friction

If the manner of treating the phenomenon of contact-friction described in introduction and in the essence of this document is most widespread, it is not only. *Code_Aster* thus propose two alternative modelings of the mechanical interactions:
- elements of joints (hydro) mechanical (modelings `*_JOINT*`) for the representation of the opening of a crack under the pressure of a fluid and friction enters the walls of the closed crack
- discrete elements of shock (modelings `*_DIS_T*`) for the representation of a specific contact by springs with possible taking into account of friction

These two other modelings are based both on finite elements and thus on specific laws of behavior (`JOINT_MECA_FROT` for the elements of joints and `DIS_CHOC` for the discrete elements).

More precise details on these elements are provided to the §4.8 and §4.9.

To finish, it will be noted that it is possible to model contact on the edges of a crack represented with method X-FEM. One will refer to the note [U2.05.02] for more information.

2 Pairing

2.1 Concept of zones and surfaces of contact

It is always to the user to define surfaces **potential** of contact: there does not exist in *Code_Aster* of automatic mechanism of detection of the possible interpenetrations in a structure.

The user thus provides in the command file a list of couples of surfaces of contact. Each couple contains one **surface** said “main” and one **surface** said “slave”. One calls “ **zone** of contact “such a couple.
The conditions of contact will be imposed zone by zone. To enforce the contact consists with **to prevent the nodes slaves from penetrating inside surfaces Masters** (on the other hand the reverse is possible).
On the example below (*cf.* Figure 2.1-1), the studied structure consists of three solids, one defined three potential zones of contact symbolized by the red ellipses. As their name indicates it these zones of contact determine parts of the structure where bodies are **likely** to make contact. That means that one enforced the conditions of contact-friction there, the effective activation of dependent contact *in fine* imposed loading.

There is no restriction on the number of zones of contact. The zones must however be separate, i.e. the intersection of two distinct zones must be empty[1]. In addition, within a zone, surfaces Masters and slaves of the same zone must also have a worthless intersection: if it is not the case, calculation is stopped. When a node is obligatorily common to surfaces Masters and slaves, because of a constraint of grid for example, to refer to the §2.3.4 for a solution. If a continuous formulation is used (*cf.* 3.1.3), surfaces slaves must imperatively be two to two disjoined.
One should not hesitate to describe broad zones of contact to avoid any interpenetration. It is the number of nodes of the surface slave which is determining in the cost of calculation. Surface Master can, it, being as large as it is wished.

1 More precisely it is the intersection of surfaces slaves which must be empty

It is imperative that the nodes of surfaces of contact (Masters and slaves) carry all of the degrees of freedom of displacement (DX, DY and possibly DZ), i.e. they belong to meshs of the model. An error message stops the user if it is not the case. One will refer to the §4.4 for the modeling of a contact with a rigid surface.

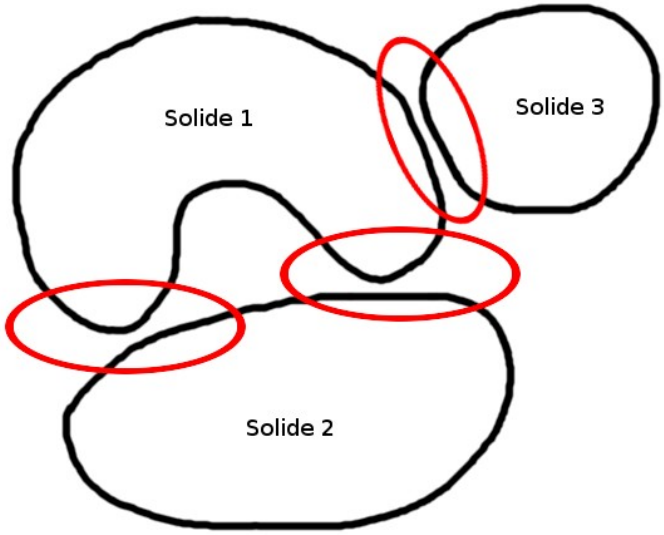

Figure 2.1-1: Definition of three zones of contact

2.2 Choices of surfaces main and slaves

As one has just said it, each zone of contact consists of a surface Master and a surface slave. In the actual position, one cannot make auto--contact in *Code_Aster* (except in the rare cases where one can predict the future zone of contact and thus define a slave and a Master).

The need to differentiate two surfaces comes from the technique adopted in calculation from the game. This calculation is carried out in a phase that one names **pairing**. The game is defined in any point of surface slave (for the discrete methods it is the nodes, for the continuous methods of the points of integration) as the minimal distance to surface Master. This dissymmetry implies a choice which can *a priori* to prove to be difficult (how to decide?). The points which must prevail in this choice are given in the following paragraphs.

One informs these surfaces in the operator DEFI_CONTACT under the keyword factor ZONE.

2.2.1 Case where a surface must be selected like mistress (GROUP_MA_MAIT)

When one of these conditions is joined together:

- one of two surfaces east **rigid** (A);
- one of two surfaces **recover** the other (b);
- one of two surfaces has an apparent rigidity **large** in front of the other (“apparent” with the direction where one does not speak about the Young moduli but about the stiffnesses in $N.m^{-1}$) (c);
- one of two surfaces is with a grid much more **coarsely** that the other (d);

then this one must be selected like surface Master.

2.2.2 Case where a surface must be selected like slave (GROUP_MA_ESCL)

When one of these conditions is joined together:

- one of two surfaces east **curve** (A);
- one of two surfaces is more **small** that the other (b);
- one of two surfaces has an apparent rigidity **small** in front of the other (c);
- one of two surfaces is with a grid much more **finely** that the other (d);

then this one must be selected like surface slave.

2.2.3 Case general

At the time of the study of complex structures, it happens that the rules given to the §2.2.1 and §2.2.2 are difficult to apply. For example when a solid is almost rigid (with respect to the other solid) and that it is curved, the rule (A) does not make it possible to decide: is it necessary to privilege the curved character or the rigid character?
In these situations “the art of the engineer” must prevail. In our example, if the two solids undergo weak slips, the curved character of the rigid solid will have only little influence and one will thus choose this main last like surface.

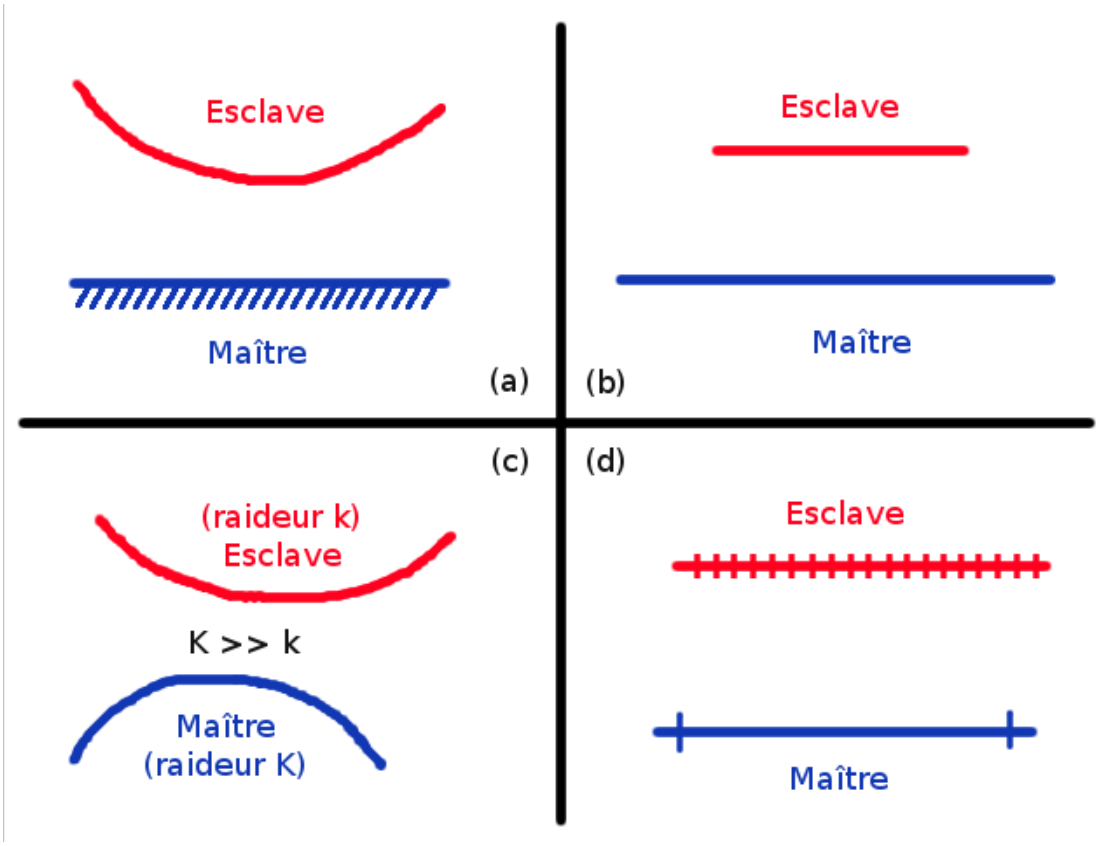

Figure 2.2.3-1: Choice of surfaces main and slaves according to various situations

2.2.4 Orientation of the normals

It is paramount always to direct them **normals** surfaces of contact so that they are **outgoing**. One can do it using the operator MODI_MAILLAGE. According to whether surface to be directed is a mesh of skin of a solid element, a hull or a beam, the keyword respectively will be used ORIE_PEAU_2D or ORIE_PEAU_3D, ORIE_NORM_COQUE, ORIE_LIGNE.
In the case of ORIE_LIGNE, one directs the tangent, of kind to being able systematically to produce the normal by a vector product.

By default (keyword VERI_NORM of DEFI_CONTACT), the good orientation of the normals is checked and one stops the user if need be.

2.2.5 Smoothness and degree of grid of curved surfaces

When surfaces of contact are curved, it is necessary to guarantee the good continuity of the normal to the facets. For that, one can is:
- to net finely into linear and to use the option of smoothing (*cf.* §2.3.2)
- to net into quadratic

So that the quadratic grid preserves its interest, it is necessary to have placed them **nodes mediums on the geometry** in the maillor and not to have used the operator CREA_MAILLAGE/LINE_QUAD of *Code_Aster*.

In the case of quadratic surfaces of contact, **in discrete formulation it is not necessary that surfaces of contact consist of quadrangular meshs with 8 nodes** (QUAD8) and one will thus prefer rather the meshs with 9 nodes (QUAD9). They then will be transformed HEXA20 in HEXA27 and them PENTA15 in PENTA18 (with the operator CREA_MAILLAGE). At present, mixed grids made up at the same time ofHEXA20 and of PENTA15 are not transformable by CREA_MAILLAGE.
If however the use of elements HEXA20 prove to be obligatory, Lbe linear relations written automatically on this occasion can be likely to enter in conflict with boundary conditions (in particular of symmetry), this is why it can be necessary to impose the boundary conditions only on the nodes tops of the meshs QUAD8 concerned (one will be able to use the operator DEFI_GROUP for the creation of the group of ad hoc nodes).
In formulation continues, for curved meshs of edge, the use of elements QUAD8 or TRIA6 can involve violations of the law of contact : this last is checked on average. One then observes games slightly positive or slightly negative in the presence of contact, which can disturb the results close to the zone of contact or calculations in recovery with initial state. For this reason it is **advised** to use elements HEXA27 or PENTA18 (with faces QUAD9) or many linear elements.

When at the end of a calculation one notices a strong rate of interpenetration of the main nodes inside surfaces slaves (what is possible contrary contrary), that generally means that the grid of one or two surfaces is too coarse or that there is a too great difference of smoothness between the two grids of surfaces. One can then either refine, or to reverse main and slave.

If a surface is rigid (and thus main), a coarse grid is sufficient except of course in the curved zones.

Finally in the typical case of one **contact cylinder-cylinder** or **sphere-sphere**, it is necessary to take care of **to net each surface sufficiently** to avoid leaving too much vacuum between them. Indeed in *Code_Aster*, one does not make for the moment not repositioning of nodes nor of projections on splines passing by surface Master, a too coarse grid will cause one then **strong oscillation of the contact pressure** (detection of the contact a node on two).

2.2.6 Sharp angles

The algorithms of pairing function less better in the presence of sharp angles, this is why one will as much as possible avoid having some in the grid of surfaces Masters and slaves. For example one will prefer to model a leave rather than a sharp angle.

If a sharp angle is essential, one will choose the surface which carries it like slave.

2.2.7 Quality of the grid

The quality of the surface elements which constitute the surface of main contact has a direct impact on the quality of pairing. Indeed distorted meshs, for example, can harm the precision of projections in spite of the robustness of the algorithm: the unicity of projection is not guaranteed any more.
For these reasons, it is recommended to check the quality of the produced grids and if necessary to correct their defects. In *Code_Aster*, the order MACR_INFO_MAIL allows to display the distribution of the elements according to their quality.

2.3 Control of pairing

2.3.1 Choice of the type of pairing

In *Code_Aster*, two types of pairing are available:

- "master-slave" (by default): it is generic, it makes it possible to prevent the nodes of surface slave from penetrating the meshs of surface Master using orthogonal projections (of a node on a mesh).
- "nodal": it makes it possible to prevent the nodes slaves from penetrating the main nodes according to a direction (given by the normal slave). It is a pairing reserved for the compatible grids of surfaces of contact for calculations in small slips. It is not available in continuous formulation (*cf.* §3.1.3).

One advises always to choose pairing " **master-slave** ", the value by default.

2.3.2 Smoothing of the normals

As its name indicates it this option makes it possible to smooth the normals. It is particularly useful in the case of curved surfaces with a grid into linear. This process is founded on average normals with the nodes, then their interpolation starting from the functions of form and realised normals, it makes it possible to ensure **continuity** normal with the nodes.

The normal is not then any more the geometrical normal, one will thus take the precaution (advised in any case) to check the results visually well.

A checking of the facettisation of surfaces is carried out automatically at the end of the step of time. She transmits a message of information when this one becomes too important and it is then advised to activate smoothing.

2.3.3 Choice of the normals

One always advises to leave the values by default: `NORMALE=' MAIT'`, `VECT_MAIT=' AUTO'`. I.e. the relation of nonpenetration is written starting from the normal Master, determined thanks to the grid.

However there exist some rare situations where one can want to impose the choice of the normal: it is primarily the treatment of the contact beam-beam (in 2D only) and of the case where surface Master is a mesh of the type `POI1`. One returns to the §3.1.6 of [U4.44.11] for more details.

2.3.4 Exclusion of nodes slaves of pairing

The keyword `SANS_GROUP_NO`/`SANS_NOEUD` serves to exclude from pairing as the nodes slaves. There can be several reasons with that:

- surface Master and slave have a nonempty intersection (bottom of crack, blocking of movements of rigid body); the common nodes do not need to be treated by the contact, they must thus be excluded.
- there already exists on the nodes slaves considered of the linear relations (boundary conditions, blocking of movements of rigid body); if those interfere with the direction of the contact (respectively of friction), one in general advises to privilege the boundary conditions and thus not to solve the contact on these nodes.

A fatal error is emitted when there exist nodes common to surfaces Masters and slaves and that the latter were not excluded.

2.4 To understand geometrical non-linearity

As one explained, geometrical non-linearity rises owing to the fact that one must apply conditions of contact-friction to a geometrical configuration which one does not know. In this section, one makes a small digression in order to explain the approach adopted to overcome this difficulty.

2.4.1 Assumption of small slips

The phase of pairing is a phase preliminary to the formulation of the conditions of contact to solve. In practice that means:

- for the discrete methods, the construction of a matrix A (for Pairing) as multiplied by the increment of displacement δu since the paired configuration, it gives the increment of game (linearized).
- for the method continues, association between a point of contact and its project in the parametric space of the mesh Master paired. It is by bringing up to date the coordinates of the mesh Master with displacement δu that obtains it the new coordinates (linearized) of the project.

Just as the equilibrium conditions, **the conditions of contact are expressed on the deformed configuration** (or finale). This configuration is not known *a priori*.
The assumption of weak relative slips of surfaces in contact is the analogue of the assumption of small disturbances (for the writing of the relations of balance).

It consists in saying that the final configuration of surfaces in contact is not very different from the initial configuration, which thus makes it possible to once and for all carry out pairing at the beginning of calculation on the initial configuration. Then to use the conditions established on this configuration for all calculation.

Such a problem is then linear geometrically: only the non-linearity of contact-friction remains, it is treated with adapted algorithms (*cf.* section 3).

2.4.2 Case general

To deal with problems of great relative slips of surfaces in contact, two possibilities exist: the use of a fixed loop of point to be reduced to the cases of small slips or for the formulation continues (§3.1.3) the simultaneous resolution within the algorithm of Newton.

2.4.2.1 Buckle of point fixes (ALGO_RESO_GEOM=' POINT_FIXE')

The adopted approach is very similar to the resolution of a non-linear problem by the method of Newton. **One transforms a geometrical non-linear problem into a succession of geometrical linear problems**. For that one will solve a succession of problems on the assumption of small slips.
I.e. one carries out a pairing (on a balanced initial configuration) and a resolution of Newton (with resolution of the contact as one will explain it in the section 3). This gives us a new configuration; if this configuration is “close” to the initial configuration then one converged (it was thus the final configuration), if not one buckles: one remakes a pairing then a resolution… and so on until finding the configuration final (*cf.* Figure 2.4.3.2-1).

The difficulty is in the characterization of the convergence of this process of fixed point. What two “close” configurations? In *Code_Aster*, they are two configurations of which the “mechanical” vector displacement to pass from the one to the other (i.e. the increment of displacement obtained by Newton restricted with the degrees of freedom DX, DY, DZ) has a small infinite standard in front of the infinite standard of the vector preceding displacement.

That implies that one thus makes always at least two iterations of geometry with this criterion (in order to give a vector initial displacement). One returns in paragraph 3.7 of [R5.03.50] for the exact expression of the infinite standard.

2.4.2.2 Algorithm of generalized Newton (ALGO_RESO_GEOM=' NEWTON')

The formulation continues (§3.1.3) offer the possibility of treating geometrical non-linearity directly within the algorithm of Newton. For that a pairing is carried out with each iteration and the geometrical terms of the tangent matrix are also reactualized.
The geometrical convergence criteria become thus an additional criterion of the algorithm of Newton: the increment of displacement must tend towards zero.

2.4.3 Convergence of the loop of geometry

2.4.3.1 Linearization of the normal

Pairing provides two information:
- game on the paired configuration,
- coordinates of the point of potential contact on surface Master.

That one uses a discrete method of contact or continues, one must derive (to linearize) the principle from the virtual powers, in particular the terms of contact.

The fixed assumption of point led to also make the assumption of a weak variation of the normal during iterations of Newton (of a step of time). This assumption is coherent with the assumption of small slips. The operator “game” is thus linearized more easily. There remains in particular constant during an iteration of geometry. This has also another implication: geometrical convergence towards the final configuration in mode not fixes can sometimes be very slow.
Contrary, the resolution by a method of Newton generalized, possible only in formulation continues (§3.1.3), a convergence much faster presents but can prove less robust. This is why it is not the adjustment by default. To activate it, the keyword will be used `ALGO_RESO_GEOM`.

2.4.3.2 Geometrical convergence criteria

One saw higher than the convergence of the loop of geometry is done on a geometrical criterion: the difference of the vectors displacements between two successive geometrical configurations is small into relative. For the typical case of the algorithm of Newton generalized in continuous formulation, the criterion applies directly to the increment of displacement of Newton.

For the resolution by loop of fixed point, “small” by default corresponds to lower than 1% of displacement since the beginning of the step of time while with the algorithm of generalized Newton, they are 0.0001 % (this one profiting in the successful outcomes from a better convergence, it is possible to require a harder criterion).

When following a calculation, one observes an interpenetration of the nodes slaves in surface Master, the only explanation[2] is a not-checking of the geometrical criterion.
One then should not hesitate to harden the criterion. For that one uses the keyword `RESI_GEOM=0.005` of kind to lower the threshold around 0.5% for example.

Sometimes geometrical convergence is simply slow, in this case it is enough to increase the maximum number of tolerated iterations: `ITER_GEOM_MAXI=20` for example.

In certain situations, one in vain exploits the value of the criterion or the iteration count, calculation does not converge: it cycles. Several possibilities are offered then to the user:
- the most current explanation is a bad discretization of surfaces of contact (i.e. a too coarse grid, a difference of smoothness between two surfaces or a bad choice of surfaces main and slaves). One returns then to the §2.2.
- when surfaces of contact are curved and with a grid relatively coarsely, the explanation can come from a too great discontinuity of the normal (facettisation). **The activation of smoothing then facilitates very often convergence** (*cf.* §2.3.2). That should not however prevent the user from re-examining his grid.

2 In continuous formulation, that can occur with curved quadratic elements, *cf.* §2.2.5

- if the case is really pathological, oneself should be forced the number of reactualizations while using REAC_GEOM=' CONTRÔLE' and NB_ITER_GEOM=n. Afterwards n iterations of geometry, calculation will pass to the step of time following whatever the value of the geometrical criterion but it will emit an alarm when the criterion is not checked with less than 1% (for $n \geqslant 2$).

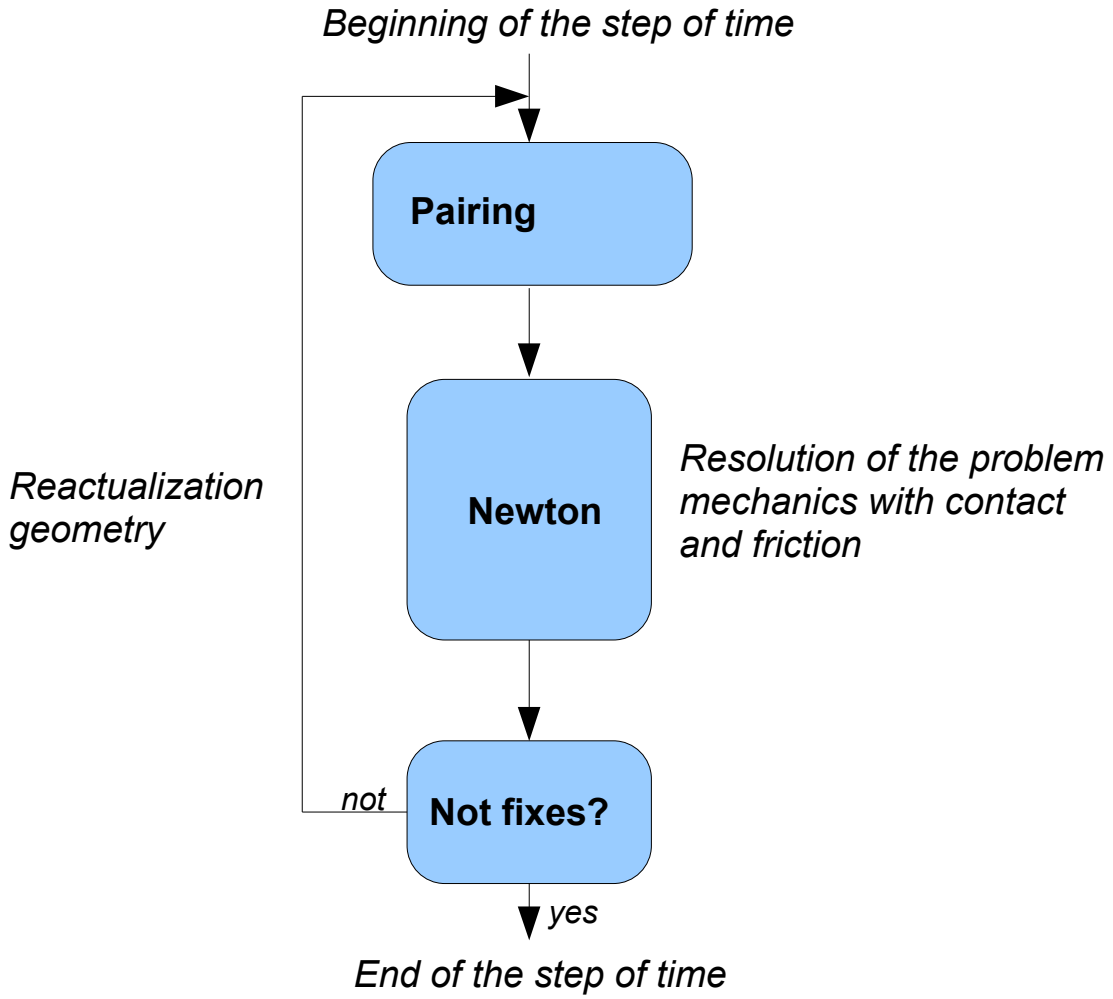

Figure 2.4.3.2-1: Treatment of the geometrical non-linearity of the contact by a fixed loop of point

3 Resolution

3.1 Outline general of the algorithm of resolution

3.1.1 Definition

What one calls “resolution of the contact”, it is the operation consisting in solving the system formed by the juxtaposition of the classical equations of the mechanics and the equations of contact-friction (the geometrical aspect being treated by pairing, it remains at this stage only the non-linearity of threshold of friction and the non-linearity of statute of the contact).

It should be noted that the two formulations available in the code differ notably on this point. Without going into the details, one briefly explains these differences for the continuation.

If the formulations discrete and continuous amount well solving the same physical problem, as their name indicates it they do not formulate it numerically same manner:

- in discrete method, the conditions of contact-friction are applied to the system discretized by the finite element method. One thus does not modify **that** the resolution of the linear system obtained by Newton $Ku=f$: one then obtains a linear system under constraints (these constraints are linear inequalities). The discrete methods thus call on algorithms of optimization to solve it.
- in continuous method, one writes a variational formulation for the equations of contact-friction, they are thus discretized as for the principle of virtual work. The approach adopted to solve the non-linear system obtained is a decoupling of non-linearities or a linearization of Newton associated with Lagrangian increased: the system thus becomes linear and nonconstrained, there is thus no algorithm specific to apply like of discrete method but rather a choice of judicious parameters (in the control of the loops and the Lagrangian one increased).

3.1.2 Discrete formulation

To illustrate the definition of the preceding paragraph, one gives Figure 3.1.2-1 the outline general of the algorithm in the case of a discrete formulation. We can pass the following remarks on this diagram:

- it represents one step of time by supposing that one places oneself in small slips (one thus does not reveal the external loop, as in Figure 2.4.3.2-1, treating non-linearity geometrical and described with the §2.4);
- in this diagram, the three classical stages of an iteration of Newton appear: assembly and resolution of the linear system, integration of the law of behavior, analyzes convergence;
- the characteristic of the discrete formulation of the contact consists of **addition** of an additional stage enters the resolution of the linear system (without contact) and the integration of the law of behavior. **One can see this stage like a postprocessing of the solution of the system without contact.**

The purpose of the additional stage that carries out it limps “discrete contact” is construction then the resolution of the system increased by the conditions of contact and friction. Two approaches exist to formulate the discrete conditions of contact-friction:

- writing of Lagrangian and dualisation of the conditions of contact-friction, one then increases artificially the size of the total system to solve and one uses an algorithm of optimization to satisfy the constraints inequalities. This approach is treated with the §3.2.1.
- penalization (or regularization) of the conditions of contact-friction, one preserves the same size for the total system but one enriches the matrix, it does not have there a specific algorithm, it is the algorithm of Newton which ensures convergence. On the other hand it contact is solved only roughly and the user must provide parameters to control the algorithm. This approach is treated with the §3.2.2 and §3.3.2.

What produces it limps “discrete contact” at exit is a field of displacement checking the conditions of contact-friction as well as reactions of contact-friction. These reactions are used in the checking of balance.

The discrete formulation is thus based on the resolution of a mechanical problem without contact what has an important consequence: **one cannot simply treat the case of a structure where the contact as friction take part directly in blockings of the movements of rigid body** (*cf.* §4.2).

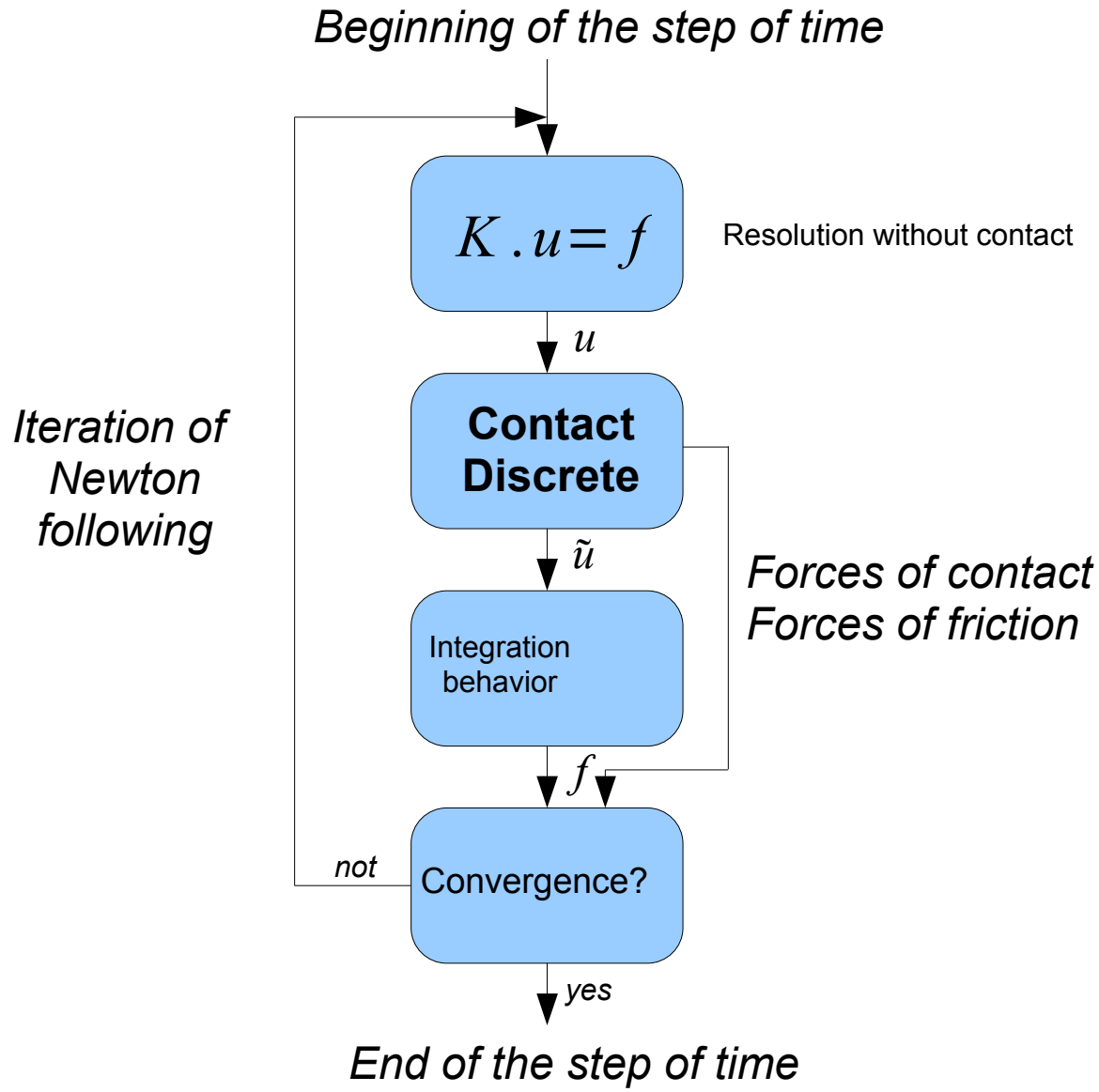

Figure 3.1.2-1: Algorithm general of a step of time in discrete formulation (small slips)

3.1.3 Continuous formulation

Figure 3.1.3-1 give the algorithm general of resolution of contact-friction with a continuous formulation, this one differs notably from the diagram in discrete formulation. Whereas with the latter contact-friction is solved by under-iterations (in limps “Discrete Contact”), the formulation continues is based on a decoupling of non-linearities:

- the non-linearity of friction (the threshold of Coulomb depends on the contact pressure which is itself an unknown factor) is treated by a fixed point on the value of the multiplier of contact or an algorithm of generalized Newton

- the non-linearity of contact is pressed on an algorithm of the statutes (with rocker per packages) or an algorithm of generalized Newton

When all non-linearities are uncoupled, there remains in the algorithm of Newton only classical non-linearities materials and kinematics.
Lagrangian increased makes it possible to transform the system constrained by the inequalities of contact-friction into a nonconstrained system. Each iteration of Newton in continuous formulation does not cost more in memory that in a calculation without contact of size equivalent contrary to the discrete formulation. Nevertheless the overlap of the loops or the treatment by the algorithm of generalized Newton implies one plus a large number of iterations (of Newton).
In continuous formulation, there exist additional degrees of freedom in modeling, consequence of the variational writing of the conditions of contact, as explained to the §4.1.1.

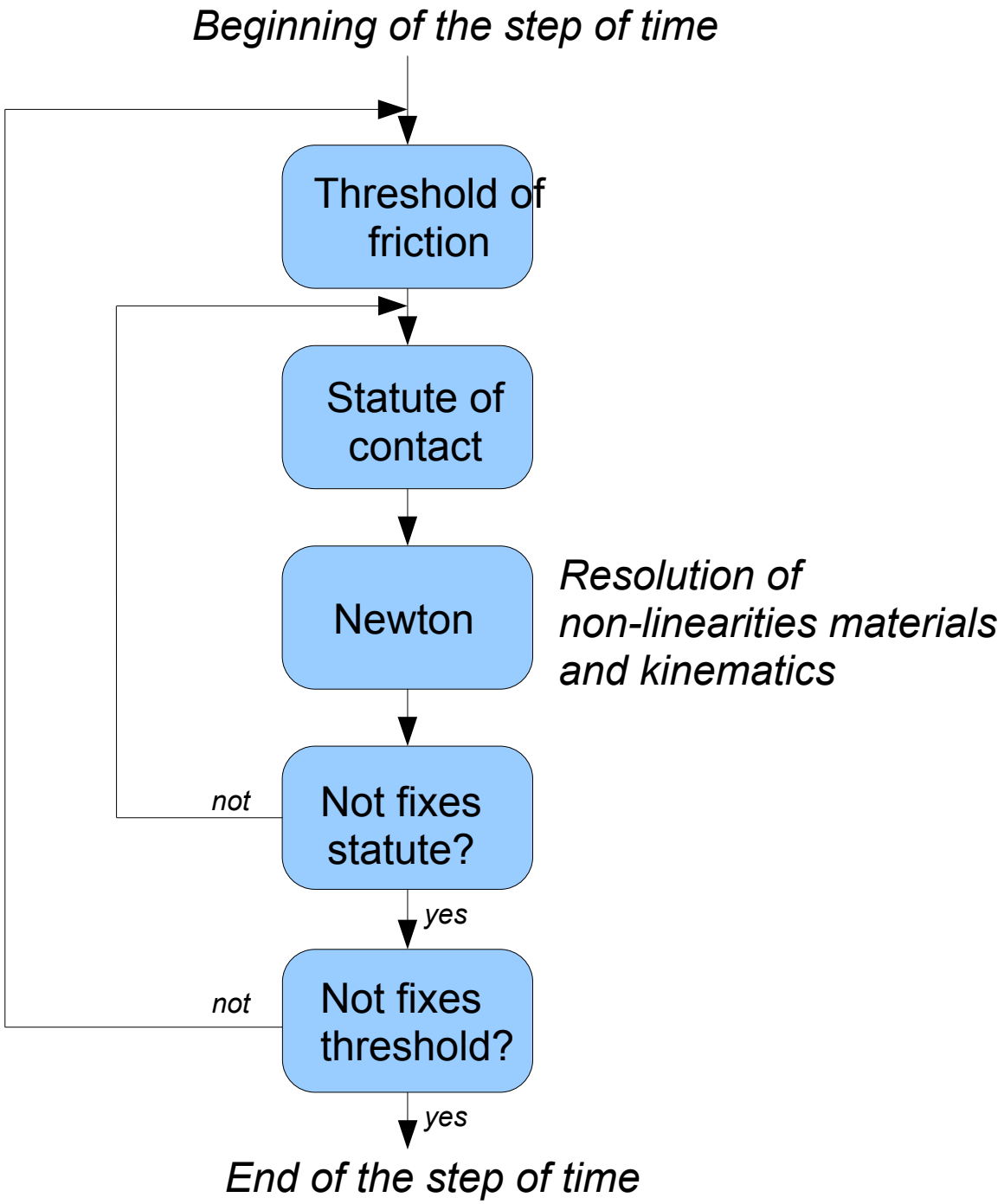

Figure 3.1.3-1: Algorithm general of a step of time in formulation continues with point fixes (small slips)

3.2 Resolution of a problem with contact alone

3.2.1 Dualisation in discrete formulation (FORMULATION=' DISCRETE')

3.2.1.1 Principle

The dualisation of the discrete system consists of the introduction of Lagrangian (cf [R5.03.50]). The system to be solved takes the following shape when it is tiny room on the active connections:

$$\begin{cases} \boldsymbol{C}.\delta\tilde{\boldsymbol{u}} + \boldsymbol{A}_c^T.\mu_i = \boldsymbol{F}_i \\ \boldsymbol{A}_c.\delta\tilde{\boldsymbol{u}} = d_{i-1} \end{cases} \quad (1)$$

Knowing that the resolution of the system without contact was already carried out, one knows the solution of the following system:

$$\boldsymbol{C}.\delta\boldsymbol{u} = \boldsymbol{F}_i \quad (2)$$

The technique of resolution is based then on the use of the complement of Schur of the system (1) to transform the system:

$$\boldsymbol{S}_{\text{schur}} = -\boldsymbol{A}_c.\boldsymbol{C}^{-1}.(\boldsymbol{A}_c)^T \quad (3)$$

The problem thus transformed has the size amongst nodes slaves and it is full. Two algorithms with the choice are available to deal with this new problem:

- a method of active constraints (ALGO_CONT=' CONTRAINTE') being based on construction **explicit** and the factorization of the complement of Schur
- a method of gradient combined project (ALGO_CONT=' GCP') being based on the resolution **iterative** system formed by the complement of Schur of the system

It should be noted that the dualisation requires the use of a direct linear solvor: in *Code_Aster*, that means 'MULT_FRONT' or 'MUMPS'.
Each of the 2 algorithms quoted above indeed carries out under-iterations during which it is necessary to solve the linear system (2) with $\boldsymbol{C}$ the matrix of rigidity of the total system without contact (what is much faster if $\boldsymbol{C}$ is already factorized).

3.2.1.2 Method 'FORCED'

Being based on a factorization (thus a direct solvor) to solve the system associated with the complement with Schur, the method 'FORCED' **do not ask any parameter setting**. In addition its convergence[3] is shown, which explains why it is the method by default in the presence of contact.

Nevertheless the use of a direct solvor presents a major drawback: **this algorithm is not adapted as soon as the number of nodes slaves exceeds a few hundreds (500)**. Indeed the factorization of a full matrix very quickly becomes crippling.

The construction of the complement of Schur can be accelerated by using the parameter NB_RESOL (*cf.* [U4.44.11], value by default 10) to the detriment of the consumed memory (the larger the number of degrees of freedom total is, the more the increase of this parameter is expensive). In order to optimize a calculation with the method of the active constraints, it is advised to do a calculation on a step of time in order to find a compromise time/memory (*cf.* [U1.03.03] for the reading of information on the consumed memory).

3.2.1.3 Method 'GCP'

When that one cannot use the method of contact by default any more because it is too expensive, an alternative is the use of the method 'GCP'. As one mentioned above this method consists of the application of an iterative solvor (gradient combined project) to solve the dual problem.
The main advantage of such a method is not to be more limited in the face of problem (several thousands of nodes slaves are perfectly atteignables). The counterpart, specific to any iterative solvor, is an obligatory parameter setting for the user.
This method is usable in parallel calculation, it is besides the only discrete method with really benefitting from it.

3 One uses a direct solvor well to build the complement of Schur but the method of the active constraints consists in activating or to one by one disable the connections of contact until satisfying the total system, it is thus an iterative algorithm.

Like any iterative solvor, method 'GCP' use convergence criteria: it is about a criterion on the value of the game. Given by the keyword RESI_ABSO, it controls the tolerated maximum interpenetration.
It is obligatory and is expressed in the same unit as that used for the grid. One advises to initially use a criterion equal to 10^{-3} time average interpenetration when the contact is not taken into account (cf §4.6).

If one notes difficulties of convergence of the algorithm of the gradient combined project, there exist 2 parameters which, one advises to exploit (in an additive way, i.e. one then the other):

- to use an not-acceptable linear research (RECH_LINEAIRE=' NON_ADMISSIBLE')
- to use a pre-conditioner of Dirichlet (PRE_COND=' DIRICHLET')

The pre-conditioner has the advantage of being optimal and thus decreases appreciably the iteration count necessary to convergence. Moreover when one is close to the solution, it makes it possible to make decrease the residue very quickly and thus to reach very weak criteria of interpenetrations.
Its disadvantage is high costs which can often prevent a saving of time of calculation in spite of the reduction amongst iterations.
For this reason, it is possible to ask its activation only when the residue sufficiently decreased: the pre-conditioner then makes it possible ideally to converge in some iterations. The difficulty lies in the quantification of "sufficiently decreased" or in other words vicinity of the solution. One controls this release by the keyword COEF_RESI who is the coefficient (lower than 1) by which it is necessary to have multiplied the initial residue (initial maximum interpenetration thus) before applying the pre-conditioner. An example of implementation of this parameter is given in CAS-test SSNA102E.

3.2.2 Penalization in discrete formulation: algorithm 'PENALIZATION'

The penalization consists in regularizing the problem of contact: instead of seeking to solve exactly the conditions on the game and the pressure, one introduces a univocal approximate relation which implies that **an interpenetration will be always observed when the contact is established**.

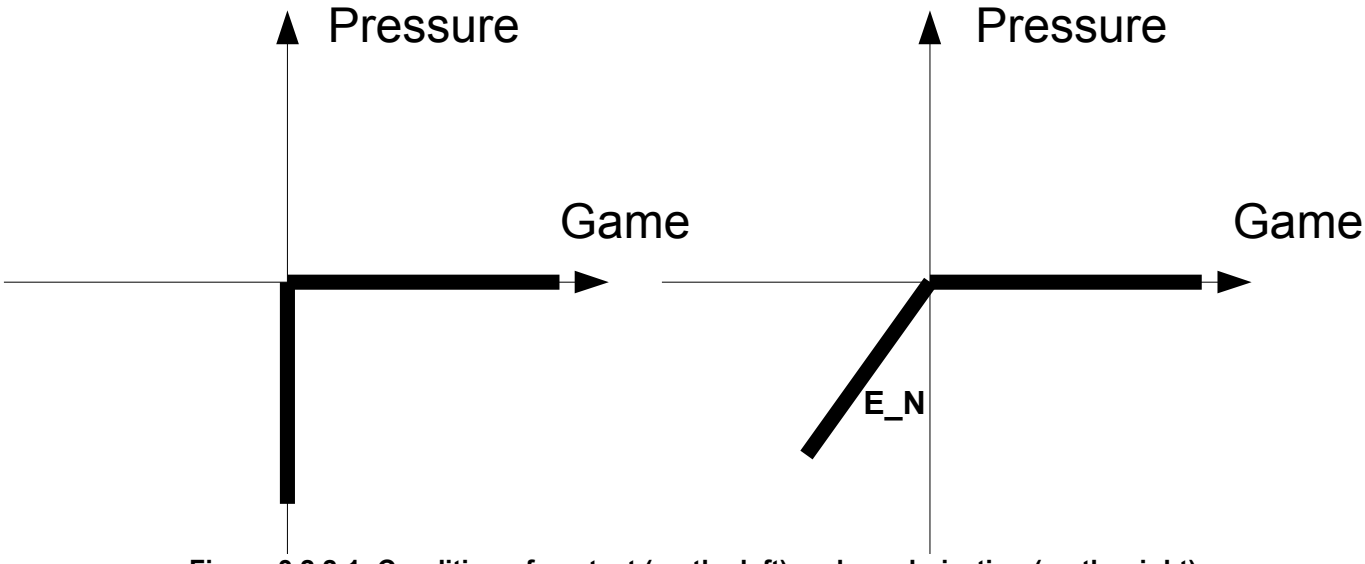

Figure 3.2.2-1: Condition of contact (on the left) and regularization (on the right)

Like shows it Figure 3.2.2-1 a parameter is added E_N to regularize the condition of contact: the larger it is, the more one tends towards the exact condition, the more it is small, the more one tolerates interpenetration.
In discrete formulation, the concept of contact pressure does not exist because one reasons on the nodes of the grid finite element: one thus works with nodal forces (*cf.* §4.1). The coefficient E_N known as of penalization thus the dimension of a stiffness has ($N.m^{-1}$).

One generally makes the analogy between the coefficient of penalization and the stiffness of unilateral springs which one would place between surface Master and slave where interpenetration is observed.

One generally chooses E_N by successive tests:

- first of all one will start by taking a value equalizes with 10 times the largest Young modulus of the structure multiplied by a length characteristic of this one;
- if calculation gives a result (satisfying or not), one will each time increase then the value by multiplying it by 10 until getting a stable result in terms of displacements and especially in terms of constraints.

The advantage of the method of penalization is **not to increase the size of the system contrary to the dualisation, but also not to restrict the choice of the linear solvor**. The counterpart is a sensitivity to the coefficient of penalization which implies systematically to conduct a parametric study before launching out in long calculations (*cf.* [U1.04.00] and [U2.08.07] for the launching of distributed parametric calculations).

To help to gauge the coefficient of penalization, there exists an automatic adaptation mechanism being based on the order DEFI_LIST_INST [U4.34.03]. One will find an example of implementation in CAS-test SDNV103I [V5.03.103].

3.2.3 Formulation 'CONTINUES'

For the problem of contact alone, the method continues has the advantage like the method (discrete) of the active constraints of not requiring any adjustment by the user.

Moreover, Comme it **is not dependent on a solvor linear direct**, it is possible to use a solvor linear iterative (like 'GCPC' or 'PETSC') to gain enormously over the computing time. However, insofar as the iterative solveurs can prove less robust, one does not advise to turn to such a solvor that once calculation with contact-friction was developed and validated. In any event, it is strongly to advise to return to a direct solvor in the event of difficulties of convergence.
When one uses an iterative solvor with the formulation continues contact-friction, it is advised to activate the method of Newton-Krylov (*cf.* keyword METHOD of STAT_NON_LINE [U4.51.03]) which makes it possible to adapt the convergence criteria of the solvor automatically linear.

The main advantage of the continuous method is to propose via the degree of freedom LAGS_C (in the field DEPL) **access to the contact pressure on surface slave**.
One however draws attention to the fact that this quantity is in fact only**a density of force of contact per unit of area expressed on the configuration of reference**. In particular, in great deformations, one cannot any more qualify it pressure because it does not have any more a physical direction.

In formulation continues two algorithms exist:

- method of point fixes on the statutes of contact: the state of the statutes of contact is evaluated in an external loop with the loop of Newton
- method of Newton generalized: the statutes of contact are evaluated with each iteration of Newton (it is the defect)

To choose the algorithm, should be used the total keyword ALGO_RESO_CONT.
The method of the point fixes (ALGO_RESO_CONT=' POINT_FIXE') is most robust but also most expensive since the non-linear problem (plasticity for example) is solved with each change of the statutes of contact.
Method of Newton generalized (ALGO_RESO_CONT=' NEWTON') is more powerful but poses sometimes problems of convergence in dynamics. In this case one will return towards a method of point fixed.

3.3 Resolution of a problem with friction

3.3.1 Treatment of the non-linearity of threshold

In *Code_Aster*, the only model of friction available is that of Coulomb (cf [R5.03.50]). An additional non-linearity must be treated in the presence of friction: it is the non-linearity of threshold.
The threshold of friction depends indeed on the contact pressure which is itself unknown.
The law of Coulomb utilizes a coefficient μ, called coefficient of Coulomb. During the phase known as of adherence, a point in contact does not move (it has a worthless speed and there exists a tangential reaction). During the phase of slip, the point has a nonworthless speed and is subjected to a tangential reaction equalizes with μ time normal reaction.

In general, **if the coefficient of friction is very low, it is advised to neglect frictions**. In addition, it is advised in the studies not to treat **initially that the contact**, this in order to introduce non-linearities ones after the others.

The discrete methods that they work by penalization or dualisation press on algorithms dedicated in the presence of friction (distinct from those used for the contact) while the method continues uses two different algorithms:

- method of point fixes on the thresholds of friction: the threshold is brought up to date in an external loop with the loop of Newton (and with the loop on the statutes of contact);
- method of Newton generalized: the non-linearity of friction is treated in the process of Newton, by explicit derivation of all the non-linear terms.

3.3.2 Discrete formulation: penalization of friction (algorithm '**PENALIZATION**')

For the 3D problems or of big size, it is advised to deal with the problem of friction by penalization. That requires, as for the penalization of the contact, the entry of a parameter of penalization (E_T). More difficult to choose than its equivalent E_N, it requires to carry out a small parametric study.

To make the analogy with the case of the penalization of the contact it will be noticed that the phase of adherence strictly speaking disappears (as soon as the contact is activated there is interpenetration, in friction there is always slip).

Convergence can also be accelerated by the use of the keyword COEF_MATR_FROT.

3.3.3 Formulation '**CONTINUOUS**'

It is the method of choice when one must deal with a problem of contact-friction : it is most robust moreover it tolerates well the great coefficients of friction (larger than $0,3$).

It is possible to choose among two algorithms of resolution for friction with the keyword ALGO_RESO_FROT.
The method of the point fixes (ALGO_RESO_FROT=' POINT_FIXE') is robust but expensive since the non-linear problem (plasticity for example) is solved with each change of the threshold of friction and with each change of the statutes of contact.
Method of Newton generalized (ALGO_RESO_FROT=' NEWTON', by default choice) is very powerful and offers a good level of robustness.
The large advantage of this algorithm is its least dependence with the value of the coefficient of friction, since there is no loop on the thresholds. One produces a not-symmetrical matrix tangent, which represents a light overcost during factorization and limit the range of the iterative solveurs usable.
It is preferable to use the generalized method of Newton since the coefficient of friction is not negligible. The savings of time calculation are very important (up to 80% of profit compared to the fixed point).
The two algorithms give identical results.

When however difficulties of convergence appear, in particular in the presence of important slips, the user will be able to parameterize the coefficient COEF_FROT (which has the dimension of the reverse of a distance). This parameter takes a value of 100 by defaults: one will test values understood enters

10^{-6} and 10^{6}. For studies where adherence is dominating, one will support values of COEF_FROT lower than the value by default while for cases where the slip is dominating, one will choose higher values.
It is also possible to activate an automatic algorithm of check of this coefficient (via the analysis of cyclings of type adherence/slip). One will use for that the keyword ADAPT_COEF. This method can fail in the direction where control can not be effective, but it will act only on the speed of convergence and not on the quality of the results.

3.4 Summary for the choice of the methods of resolution

3.4.1 For contact-friction

For the problems with low number of degrees of freedom in contact (lower than 500 degrees of freedom), one will privilege a discrete formulation with algorithm of the active constraints ('FORCED'). If friction must be activated, one will turn to a formulation 'CONTINUES'.

For the problems with a large number of degrees of freedom in contact (higher than 500 degrees of freedom), the iterative algorithm of resolution by active constraints 'GCP' is most suitable. If however one must take into account friction, one will be able to turn once again to the formulation 'CONTINUES'.

For the problems of big size (independently amongst degrees of freedom in contact), the resolution of the linear system consumes most of the computing time, the choice of the linear solvor is thus paramount. Method 'CONTINUES' (and to a lesser extent method by 'PENALIZATION') is well adapted in the sense that it leaves to the user the choice of the linear solvor and that it is well paralleled.

3.4.2 For the linear system

If one uses a discrete formulation (except penalization), only the direct linear solveurs are accessible. The solvor will thus be chosen 'MULT_FRONT' except if one carries out a parallel calculation in which case one will select 'MUMPS'. Method 'GCP' allied with the linear solvor 'MUMPS' benefit from a good level of parallelization in the algorithm of contact.

If a continuous formulation is used, it is advised, as soon as the total problem exceeds 100,000 degrees of freedom, to use an iterative solvor associated with the pre-conditioner 'LDLT_SP' and with the method of Newton-Krylov (cf § 3.2.3). If calculation implements friction or is parallel, the iterative solvor 'PETSC' is the best choice.

4 Methodologies

In this part, one answers the questions frequently asked at the time of the studies with contact-friction. The techniques installation in this part are often pressed on other operators that DEFI_CONTACT, one will briefly describe the keyword to be used but the user will be able advantageously to refer to documentations of use of these orders.

4.1 To recover the contact pressure

In postprocessing of a calculation of contact, one generally wishes to reach the efforts of contact-friction. More precisely, one wishes to know the normal and tangential constraint on the edge of the solids in contact.
The formulation continues contact directly gives access to the pressure of contact-friction, while the discrete formulations require to approximate it by the constraints on the edge.
An example of implementation for the two formulations exists in CAS-test SSNP154 [V6.03.154].

4.1.1 Continuous formulation

In continuous formulation, the field DEPL contains one or more additional unknown factors:

- LAGS_C represent the surface density the effort of contact expressed on the configuration of reference.
- LAGS_F1 and LAGS_F2 the coordinates of a directing vector in the tangent plan represent. This vector of standard lower or equal to 1 indicates the direction of slip or adherence when which one takes into account friction.

These sizes are defined in any point of surface slave of contact. One can thus easily reach the contact pressure. One will however note that in great displacements, configuration initial and final being confused more, the degree of freedom LAGS_C any more the significance of a pressure does not have.

To reach the surface density the effort of friction (in phase of adherence like slip), should be carried out an additional calculation: the standard of the directing vector in the tangent plan indeed gives the amplitude compared to the threshold of friction.
If one notes λ the contact pressure then density the effort of friction τ is written:

$$\tau = \mu . \lambda . \sqrt{\left(LAGS_{F1}^2 + LAGS_{F2}^2 \right)} \quad (4)$$

In penalized formulation (ALGO_CONT=' PENALISATION'), the degrees of freedom of pressure continue to exist, one can thus apply what precedes.

It happens sometimes that the contact pressure raised by this method present of the oscillations, in particular for curved geometries. In this case, the got results remain right but it becomes difficult to exploit the contact pressure directly. One returns then in the following paragraph for the statement of the contact pressure by the constraints.

4.1.2 Discrete formulation

In discrete formulation, no degree of freedom is added with the principal unknown factors. The problem of contact being formulated on the discrete system, the possible multipliers of Lagrange used do not even have the dimension of a pressure but that of nodal forces.
This absence obliges to calculate the tensor of the constraints of Cauchy on the edge of surfaces in contact. The contact pressure is written indeed:

$$\lambda = (\sigma . \underline{n}) . \underline{n} \quad (5)$$

where $\underline{n}$ is the normal on the surface of contact and σ the tensor of the constraints of Cauchy.

To calculate the constraints of Cauchy on the edge, they should be interpolated starting from the constraints at the points of Gauss. For then obtaining the pressure, it is necessary to calculate the normals on the edge (of the geometry deformed in great displacements) then to create a field starting from the interpolated constraints and normals.
Framed below the watch how one can proceed for a calculation 2D. That supposes to have already calculated in the result (RESU) the field of the constraints of Cauchy to the nodes (SIEF_NOEU).

```
sigm=CRÉA_CHAMP (TYPE_CHAM=' NOEU_SIEF_R',
                 OPERATION=' EXTR',
                 RESULTAT=RESU,
                 NOM_CHAM=' SIEF_NOEU',
                 INST=1.0,
                 );

depl=CRÉA_CHAMP (TYPE_CHAM=' NOEU_DEPL_R',
                 OPERATION=' EXTR',
                 RESULTAT=RESU,
                 NOM_CHAM=' DEPL',
                 INST=1.0,
                 );

# normal on the initial configuration
NormaleI=CRÉA_CHAMP (TYPE_CHAM=' NOEU_GEOM_R',
                     OPERATION=' NORMALE',
                     MODELE=MO,
                     GROUP_MA= ('Slavic', 'Master degree',),
                     );

Pressure = FORMULA (VALE=' SIXX*X*X+SIYY*Y*Y+2*SIXY*X*Y',
                    NOM_PARA= ('SIXX', 'SIYY', 'SIXY', 'X', 'Y',),);

Pres=CRÉA_CHAMP (TYPE_CHAM=' NOEU_NEUT_F',
                 OPERATION=' AFFE',
                 MAILLAGE=MESH,
                 AFFE=_F (GROUP_MA= ('Slavic', 'Master degree',),
                          NOM_CMP=' X1',
                          VALE_F=Pression,),);

pI=CRÉA_CHAMP (TYPE_CHAM=' NOEU_NEUT_R',
               OPERATION=' EVAL',
               CHAM_F=Pres,
               CHAM_PARA= (NormaleI, sigm,),);
```

For a calculation in great displacements, the normal must be calculated on the deformed configuration. For that it is necessary to relocate the grid with the order MODI_MAILLAGE.
In the example above, one uses the formula drawn from (5) to calculate the contact pressure explicitly. In the typical case where the edge on which one extracts the pressure is parallel to the axes of the reference mark, the pressure is directly equal to one of the diagonal components of the tensor of the constraints of Cauchy (SIXX, SIYY or SIZZ).

4.2 Movements of rigid bodies blocked by the contact

This paragraph applies only to the studies in statics. In dynamics, the movements of rigid body are allowed.

It arrives in the studies that the contact makes it possible to block the movements of rigid body of certain solids (and to make so that those become deformed). The initial not-catch in account of this phenomenon will thus involve the singularity of the matrix of rigidity (and thus impossibility of solving).

The discrete formulations are not adapted to an initial taking into account of the contact, the realization of studies with solids only held by the contact will thus require in this case an enrichment of modeling.
The formulation continues makes it possible to take account of an initial contact naturally and for this reason thus is well adapted under investigation mechanisms.

For studies in three dimensions, there exist 6 movements of possible rigid bodies: 3 translations, 3 rotations. For studies in two dimensions (modelings D_PLAN, C_PLAN), there exist 3 movements of rigid body: 2 translations and a rotation. Axisymmetric modeling (AXIS) is particular: there exists one movement of rigid body, the translation along the axis Oy (cylindrical axis of symmetry).

When one notes the existence of movements of rigid body in his modeling, **one will always start by checking that there do not exist symmetries in the structure and its loading**. The conditions of symmetry indeed make it possible to remove most of the movements of rigid body.

An example of blocking of movements of rigid body in formulation continues (by CONTACT_INIT) and in discrete formulation (by springs) is available in CAS-test SSNA122 [V6.01.122].

4.2.1 Continuous formulation

In continuous formulation, the taking into account of an initial contact is assured zone by zone with the keyword CONTACT_INIT. By defaults at the beginning of a calculation all the connections with null game (or interpenetrated) are activated (CONTACT_INIT=' INTERPENETRE'). The tolerance, to determine if a game null or is interpenetrated, is fixed in-house in the program at $10^{-6} \times a_{min}$ where a_{min} represent the smallest nonworthless edge of the grid.

It is possible to disable this automatic activation (CONTACT_INIT=' NON'). When one does non-linear calculations with recovery (i.e. with the keyword ETAT_INIT of STAT_NON_LINE), it is essential to use the value by default ('INTERPENETRATES') in order to ensure a recovery starting from the true state of contact (and not of a virgin state).

Finally if one wants to stick initially all surfaces of contact independently of the initial game, one can select CONTACT_INIT=' OUI' (that can be useful if the grids are not perfectly in contact).

In all the cases where an initial contact is declared, of the efforts will be generated: **it is not about a simple geometrical repositioning** aiming at sticking the grids.

The activation of an initial contact blocks the movements of rigid body following the normal direction to surface. If one wants to take into account an initial adherent state in order to block the tangent direction, one will be able to specify an initial threshold of contact not no one *via* SEUIL_INIT. This parameter informs the initial value of the contact pressure (homogeneous with a density of surface force). By default, if calculation is resulting from a continuation then one automatically rebuilds the value of the initial threshold while using values of LAGS_C contents in ETAT_INIT/STAT_NON_LINE.

It should be noted that the use of an initial contact in continuous formulation as makes it possible to be freed from not-convergence when as a structure is subjected only to displacements. For example, when two solids initially in contact are in a hurry one against the other by displacements (it is thus about a rigid movement of body).

4.2.2 Discrete formulation

In discrete formulation, one is obliged to manually block the movements of rigid body of the solid accused by springs of low stiffness. By “weak” one understands sufficiently small to generate only negligible nodal forces in front of the nodal forces put concerned in calculation.
The goal of the springs is to make so that calculation without contact is able to turn in linear mechanics (i.e. in the operator MECA_STATIQUE or in STAT_NON_LINE once withdrawn the conditions of contact).

There exist two approaches for the addition of springs:

- to add a spring of low stiffness in any point of the structure
- to add springs quite selected points to into cubes block the movements of rigid body of the structure

The first approach has the advantage of the generics but can disturb sometimes too much the solution (whatever the stiffness of the springs). Indeed such an approach amounts adding on all the diagonal terms of the matrix a positive term which makes it invertible.
The second approach only adds springs where it is necessary. When there exist points of the structure which will be brought to have a weak displacement (thus not to generate that a weak nodal force in the spring), this approach is adapted more.

To apply a spring in *Code_Aster*, it is necessary to create meshs of the type POI1 starting from nodes. For that one uses the operator CREA_MAILLAGE/CRÉA_POI1. To use the first approach one will choose to create this group of meshs on all the structure (TOUT=' OUI'), while for the second approach, one will indicate the group of nodes wished. The group of meshs lately created will be used to affect a modeling of the type 'DIS_T' or '2D_DIS_T' in AFFE_MODELE.

The definition of the characteristics of the spring is carried out in the operator AFFE_CARA_ELEM. By defaults, the stiffnesses entered the total reference mark. If for example one wishes to block a rigid movement of body in a direction parallel with the axes of the total reference mark, one will define a nonworthless stiffness only according to this direction. Below an example of definition of a stiffness for a calculation 2D according to the direction DY.

```
RESSORT=AFFE_CARA_ELEM (MODELE=model,
                        DISCRET_2D=_F (CARA=' K_T_D_N',
                                       GROUP_MA=' SPRING',
                                       VALE= (0. , 1.0e-1,),),);
```

Whenever the direction to be blocked is not parallel to the axes, two alternatives are possible:

- to define a stiffness according to all the directions
- to define the stiffness in a local reference mark. It is then necessary to lay down the direction of this reference mark (keyword ORIENTATION of AFFE_CARA_ELEM) or to use springs being pressed either on meshs POI1 but SEG2.

For an example of use of springs, one will consult CAS-test ZZZZ237 and his documentation [V1.01.237].

4.3 Great deformations, great displacements and contact

The taking into account of conditions of contact-friction is completely uncoupled from the taking into account of great displacements or great deformations. More generally any non-linearity which is of order material or geometrical is *a priori* compatible with the use of the contact.

In practice, one often notes difficulties of convergence in studies mixing three non-linearities. One gives in the continuation of this section the approach to be adopted in this case.

Examples of calculation mixing three non-linearities are available in CAS-tests SSNP155 [V6.03.155], SSNP157 [V6.03.157] and SDNV103 [V5.03.103].

4.3.1 To uncouple non-linearities

When that such a calculation fails, the first approach is to retrogress: by uncoupling non-linearities and while trying to apply the good practices into non-linear (*cf.* [U2.04.01]).
That means:

- to carry out an elastic design in small disturbances with the activated contact. If this calculation fails, apply the advices delivered in the first part of this document (orientation normals, main choices of surfaces and slaves, choice of the algorithm of resolution,…)
- to carry out a calculation with a nonlinear law of behavior but without contact. If this one fails, then the problem comes from the integration of the behavior. One will refer then to documentations [U2.04.02] and [U2.04.03].
- if necessary to carry out a calculation in great displacements but without contact and non-linearity material. If this calculation does not function, try to use another model among

those of great displacements available in *Code_Aster* (‘SIMO_MIEHE’, ‘GDEF_LOG’, ‘PETIT_REAC’).

4.3.2 To parameterize the algorithm of Newton well

If complete calculation (mixing all non-linearities) does not converge in spite of the application of the preceding advices then one can try to exploit the parameters of the algorithm of Newton. That leaves the following report:
When one couples contact and non-linearity material for example, it is possible (by the “abrupt” correction of the contact) to start in the law of behavior of the mechanisms (left the elastic range, discharges) which should not be active in the final solution and which are likely to degrade the tangent matrix (until making it noninvertible). That makes then any convergence impossible.

One thus proposes to use the following adjustments in the algorithm of Newton (operator STAT_NON_LINE or DYNA_NON_LINE):

- reactualization of the tangent matrix to each iteration (REAC_ITER=1)
- use of an elastic prediction (PREDICTION=' ELASTIQUE')
- in great deformations (DEFORMATION= ' SIMO_MIEHE'), the tangent matrix is nonsymmetrical, it is thus necessary well to take care to inform SYME=' NON' in the keyword SOLVEUR.

When calculation has still difficulty converging, it is necessary to return to modeling:

- does my calculation cause problems of incompressibility? In this case, consult documentations [U2.04.01] [U2.04.02] and try to use adapted finite elements (under-integrated, with mixed formulation).
- the behavior that I use have a coherent tangent matrix? If it is not the case, one can try as a last resort to use a matrix ‘ELASTIC’ and to increase the iteration count of Newton.

4.3.3 Resolution of a quasi-static problem in slow dynamics

As a last resort, for the quasi-static problems, to carry out a dynamic calculation in long time can bring a solution. The matrix of mass causes to stabilize the structure, it should however be made sure that the inertial forces remain weak then in front of the internal forces of the system.

One advises for this kind of modeling to assign to the structure his true density (it is obligatory in any case in the presence of loading of gravity) and to carry out calculation by using great steps of time.
An example of implementation is available in CAS-test SSNP155 [V6.03.155].

4.4 Rigid surface and contact

Sometimes one wishes to model in the studies of the rigid solids which make contact with deformable solids. In this section, it is explained how to optimize such studies.

In order not to weigh down modeling the rigid solids will not be entirely modelled: **only their edge will carry degrees of freedom**. In order to facilitate the orientation of the normals of this rigid solid the grid will comprise the complete solid however.

After having directed the normals, one will thus affect in AFFE_MODELE only elements of edge to the skin of the rigid solid: as the elements of edge do not carry rigidity, an alarm is emitted to prevent risk of noninvertible matrix of rigidity. **This alarm is normal in this case** and can be been unaware of.

To prevent that the matrix of rigidity is singular, **it is necessary to impose the displacement of all the degrees of freedom carried by the rigid edge**. That is done with the orders:

- AFFE_CHAR_CINE/MECA_IMPO from which the advantage is to eliminate the unknown factors
- AFFE_CHAR_MECA/DDL_IMPO who adds additional unknown factors to the problem.

One thus advises to eliminate the unknown factors (AFFE_CHAR_CINE).

Rigid surface will be declared like **surface Master** in DEFI_CONTACT as explained with the §2.2.1.

One will be able to refer to CAS-test SSNV506 [V6.04.506] for an example of contact with rigid surface.

4.5 Redundancy between conditions of contact-friction and boundary conditions (symmetry)

In the presence of symmetries in the studied structure, it is current that the conditions of friction enter in conflict with the boundary conditions of symmetry. Figure 4.5-1 watch the example of two cubes in contact-friction, the hatched part represents the faces of the cubes subjected to a condition of symmetry (DX=0).
In this example, the edge of the higher cube **in thick feature** belongs to surface slave and also carries the condition of symmetry. This condition enters in conflict with the condition of friction written the tangent plan (here the plan xOz). In practice calculation will once stop the established contact because the tangent matrix will be singular.
Mechanically it is seen that the condition of symmetry implies that adherence or the slip will occur only according to the direction DZ (green tangent vector). To eliminate the redundancy it is thus necessary to exclude the direction from following friction DX (red tangent vector).

For that one will use the keyword SANS_GROUP_NO_FR to indicate the list of nodes of the edge slave then one will inform (in the total reference mark) DIRE_EXCL_FROT= (1,0,0) that is to say direction DX to exclude.

CAS-test ZZZZ292 implements the functionality SANS_GROUP_NO_FR.

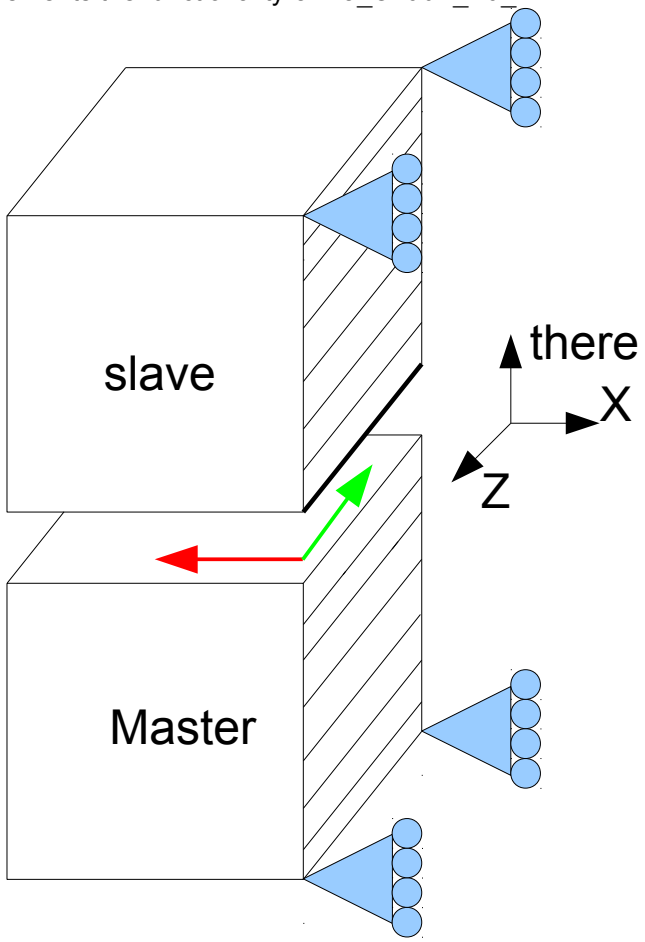

Figure 4.5-1: Elimination of directions of friction

4.6 To measure the interpenetration without solving the contact

The resolution of a problem of contact sometimes which can be expensive it can be advantageous to replace the imposition of the conditions of contact by a simple checking of the interpenetration. It is all the more interesting whenever one simply wishes to check that solids will not make contact.
For each zone of contact defined in the operator DEFI_CONTACT, it is possible to choose if one wishes to enforce the contact there (RESOLUTION=' OUI') or not (RESOLUTION=' NON').

The interest of such an approach is not to weigh down a calculation: when that a calculation carried out without resolution **on the entirety of the zones of contact** watch that there is no interpenetration then one can be unaware of the modeling of the contact.
Attention however: so at least one of the zones of contact "is solved" and the another "unsolved" then existence of an interpenetration does not prejudge a solution of a complete calculation with contact (because of possible interactions between zones of contact).

Finally this technique can be also used to measure the rate of interpenetration on the level of the zones of contact to gauge a criterion like the coefficient of penalization or the maximum interpenetration tolerated in the method of resolution 'GCP' .

4.7 To display the results of a calculation of contact

When one displays the results of a calculation of contact-friction in a software of postprocessing, it is necessary to take care of several things:

- for the posting of the deformations, **a factor of amplification different from 1 can result in visualizing nonreal interpenetrations**
- for calculations 2D in formulation 'CONTINUES', one will pay attention during the posting of deformations to the software of postprocessing which regards the first three components of a field as the components according to X, Y and Z displacement. In 2D, the third component corresponds to LAGS_C and must thus be been unaware of

4.8 Specific contact with discrete elements (springs)

- during the visualization of the field of postprocessing of the contact (CONT_NOEU) and more particularly of the component CONT who indicates the state of the contact, one will sometimes automatically pay attention to the interpolation of the fields to the nodes realized. Indeed this component takes values 0 (not contact), 1 (adherent contact) or 2 (slipping contact). The adherent state is not possible that in the presence of friction: if one visualizes such a value for a calculation of contact without friction it is that there is interpolation of the field.

Discrete elements (or springs) 2D_DIS_T* or DIS_T* associated with the law of behavior DIS_CHOC [R5.03.17] allow to give an account of a specific contact in a fixed direction. They are well adapted to the modeling of shocks and for this reason are often used in dynamics on modal basis [U4.53.21] and in explicit dynamics [U4.53.01].
The springs can be based indifferently on a specific mesh or a segment. In all the cases, it is necessary to correctly direct each element with the order AFFE_CARA_ELEM [U4.42.01].
The contact as friction are solved by penalization (cf §3.2.2). The stiffnesses of penalization, the coefficient of friction as well as the initial games are specified in material DIS_CONTACT (order DEFI_MATERIAU, [U4.43.01]).

This kind of element is not usable in great displacements because the direction of contact fixed and is given by the initial orientation of the discrete element.

CAS-tests SSNL130A and SDND100C implement springs of contact.

4.9 Elements of joints (hydro) mechanical with contact and friction

Elements of joints (hydro) mechanical PLAN_JOINT (_HYME) and 3D_JOINT (_HYME) allow to model the opening of a crack under the pressure of a fluid and friction on the edges of the crack closed with the law JOINT_MECA_FROT [R7.01.25]. It is possible to couple the opening of the crack and the propagation of the fluid with modelings *_HYME.
The formulation of contact-friction is penalized and the related parameters are indicated under the keyword JOINT_MECA_FROT order DEFI_MATERIAU [U4.43.01].

CAS-tests SSNP142C and SSNP142D provide an example of application of such elements on the modeling of a stopping.

5 Bibliography

[1] MR. ABBAS. “Discrete formulation of contact-friction”. Reference material of *Code_Aster*. [R5.03.50].

[2] MR. ABBAS. “Elements of contact derived from a continuous hybrid formulation”. Reference material of *Code_Aster*. [R5.03.52].

[3] T. SOZA. “Operator DEFI_CONTACT ”. Documentation of use of *Code_Aster*. [U4.44.11].

[4] A. KONTER. “Advanced Finite Element Benchmarks Contact”. NAFEMS, 2006.

[5] A. KONTER. “How to undertake has contact and friction analysis”. NAFEMS, 2000.