Quasi-static nonlinear algorithm (STAT_NON_LINE)

Summarized:

Operator STAT_NON_LINE [U4.51.03] of *Code_Aster* allows a quasi-static request in the case of to integrate various types of non-linearities coming from the behavior of the material, of great geometrical transformations or the conditions of contact/friction. One describes the total algorithm of resolution here employed.

The integration of the behavior models itself is described in the documents [R5.03….] and [R7.01….], (for example [R5.03.02] for elastoplasticity), to which one will be able to refer for more details.

For computations in great geometrical transformations, one will be able to consult for example the document [R5.03.20] on nonlinear elasticity in large displacements, or the documents [R5.03.21], [R5.03.22] on the thermoelastoplasticity with isotropic hardening.

For the contact friction, there exist three documents: [R5.03.50] on the discrete formulation of the contact/friction, [R5.03.52] for the hybrid formulation by elements of contact/friction, and [R5.03.53] on the contact in great slidings with method XFEM.

For all that relates to control, it is necessary to refer to the document [R5.03.80].

Contents

1 Presentation.....
1.1 Généralités4.....
1.2 Types of nonlinéarités4.....
1.2.1 Large Behaviors not.....
linéaires4 1.2.2 transformations5.....
1.2.3 unilateral Contact and frottement5.....
1.3 Position of the quasi-static problem not linéaire5.....
2 Method of Newton.....
2.1 Principle of the méthode8.....
2.2 Adaptation of the method of Newton with the problem posé8.....
2.3 Phase of prediction of Euler10.....
2.3.1 Linéarisation10.....
2.3.2 Stamp 2.5 tangent.....
prédiction11 2.3.3 Second member vector of the variables of.....
commande11 2.3.4 Second member vector of the loading.....
mécanique12 2.3.5 System.....
linéaire13 2.3.6 Alternatives of.....
the prédiction14 2.4 Phase of correction of.....
Newton14 2.4.1.....
Principe14 2.4.2.....
Linéarisation15 2.4.3 System.....
linéaire15 2.4.4 Alternatives of.....
the correction16 2.4.4.1 Methods..... of
quasi-Newton16 2.4.4.2 Use of an evolutionary..... matrix
TANGENTE-sécante19 2.4.4.3 Method..... of
Newton-Krylov20 Criteria of convergence21.....
2.5.1 Choices of the components for convergence criteria 23.2.6
Difference of the matrixes in prediction and correction23.....
2.7 Cases of the loadings suiveurs24.....
3 linear Search.....
3.1 Principe26.....
3.2 Minimization of a fonctionnelle26.....
3.3 Method of minimisation26.....
3.4 Application to the minimization of the énergie28.....
3.5 Determination of the steps of avancement29.....
3.6 Computation of the coefficient of search linéaire30.....
3.6.1 secant Method (METHODE=' CORDE')..... 30
3.6.2 mixed Method (METHODE=' MIXTE')..... 30
3.6.3 Typical case: the method of..... Newton-Krylov31

4 Control.....
5 Bibliography.....
6 History of the versions of the document.....

1 Presentation

1.1 General information

STAT_NON_LINE is the operator of *Code_Aster* making it possible to carry out nonlinear mechanical computations when the effects of inertia are neglected (if one wants to take into account the effects of inertia, it is necessary to use DYNA_NON_LINE, see [R5.05.05]).

The computation relates a priori only to the mechanical variables (displacements, stresses, local variables) by excluding any coupling with other physical phenomena (thermal,…). Consequently, the associated fields influencing the structural mechanics behavior (thermal, hydrous, metallurgical fields) are calculated as a preliminary by other operators (THER_LINEAIRE [U4.33.01], THER_NON_LINE [U4.33.02]), even by other codes (for example CODE_SATURNE for the mechanics of the fluids,…).

There is an exception with regard to the modelization thermo-hydro-mechanics (modelization known as “THM”) for which STAT_NON_LINE treats the group of the coupled problem of the equations of diffusion of the thermal, of the pressure of (of) fluid (S) and of the mechanical equilibrium [R7.01.10].

> It should be noted that when one speaks about time of computation in this document, one almost always refers to a pseudo-TEMPS, which N `does not have physical meaning and which is only used to parameterize the incremental algorithm. However, time keeps a physical meaning in viscoplasticity and when the command variables depend on it.

1.2 Types of nonnonlinear

1.2.1 linearities Behaviors

the nonlinear behavior models are described in the documents [R5.03….], for the behavior generals, and [R7.01….] for the géo-materials. In STAT_NON_LINE, two families of behaviors are available:

- That which corresponds to the key word factor COMP_ELAS (Behavior Elastic) led through the balance equation to a nonlinear system explicitly depending on the field of displacements $\mathbf{u}$ compared to the reference configuration, and parameterized by the time of computation (through inter alia the thermal evolution). For more details, one will be able to refer, for example, with the document [R5.03.20] concerning nonlinear elasticity in great transformations (very-elasticity).
- The other family, which corresponds to the key word factor COMP_INCR (Behavior Incremental), is associated with behavior models expressed by ***an implicit differential equation*** (for example elastoplasticity, the visco - plasticity, hypo-elasticity, etc). In this case, the behavior model is integrated as presented for example in [R5.03.02]: by connecting a displacement increment $\Delta\mathbf{u}$ calculated from a mechanical state given (the mechanical state being represented by a field of displacements $\mathbf{u}$, a stress field σ and a field of local variables α) to the stress field at time of computation t. The balance equation thus leads to a nonlinear system in $\Delta\mathbf{u}$, but which is also parameterized by the time of computation through the facts of the case (variation of the mechanical loading and thermal evolution for example).

It is necessary to have for the spirit the basic difference between the two approaches. The elastic case referenced by supposes the existence of a state of ratio to which the elastic strain is written: this state corresponds in a state without strain, nor forced. It is the “absolute” value of the loading which creates the strain. The incremental case leans on the state previously calculated and “forgets” any reference to the former states except that given by the local variables. In this case, it is the variation of the loading which modifies the state of the system: in particular, one needs a variation of the field of temperature to create thermal strains.

In both cases, one calculates the solution gradually. It is theoretically not essential in the nonlinear elastic case because it is not necessary to keep the memory of the former state (not of local variables), nor to evaluate a differential equation, but it may be that it not linearity of the required solution either too strong for the algorithm of resolution used, and that it is essential, for numerical reasons, to operate step by step.

1.2.2 Great transformations

Whenever the assumption of the small disturbances (moderate displacements and strains) is not checked, the method of resolution of the problem must then integrate the evolution of the geometry of the problem, handle a particular kinematics and use an adequate formulation of the constitutive law.

In practice, the assumption of the small strains can be applied as long as the square of the modulus of deformation remains lower than the accuracy of computations considered. In the same way, the assumption of small rotations can be applied as long as the product between the square of the swing angle and the modulus of deformation remains lower than the accuracy of computations considered.

Various alternatives exist within *Code_Aster*; our purpose is not here to make a detailed presentation of it and we return to the various documents treating each problems specifically. It is advisable to distinguish the formulations which and the operate on massive isoparametric elements (2D or 3D) formulations being used for the structural elements (beams, plates and shells). For the cases of the massive isoparametric elements, one finds three great types of formulation of the kinematics for the case of the large deformations:

- Kinematics DEFORMATION= `PETIT_REAC' makes it possible to treat an unspecified constitutive law in large deformations. The model is written in small strains and the taking into account of the large deformations is done only by reactualizing the geometry. This formulation is not incrémentalement objective. Moreover, one can use it only if the behavior is isotropic, if the elastic strain are weak in front of plastic strains, if rotations remain weak (lower than $10°$) and if a sufficiently fine discretization in time is adopted. Moreover, the absence of the geometrical contribution in the tangent matrix can sometimes make convergence difficult (see [R5.03.21 for more details).
- Kinematics DEFORMATION= `SIMO_MIEHE' makes it possible elastoplastic constitutive law to treat one with isotropic hardening in large deformations, the model of ductility fracture known as “Rousselier” or the élasto (visco) plasticity with phase change for the metallurgy (see [R5.03.21], [R5.03.06] and [R4.04.03]). This formulation is incrémentalement objective, without limitation on the level of the transformations applied but it is available only for the three quoted constitutive laws and allows to treat only the cases where the behavior is isotropic.
- Kinematics DEFORMATION= `GDEF_HYPO_ELAS' makes it possible to treat any hypo-elastoplastic constitutive law. It is incrémentalement objective, without limitation on the level of the transformations applied but allows to treat only the cases where the elastic strain are small in front of the unrecoverable deformations (see [R5.03.24]) and if the behavior is isotropic elastic.

To treat the elastic large deformations, it is necessary to employ another formalism, called via DEFORMATION = "GROT_GDEP" , which is usable for the behavior models very-elastics nonlinear in large displacements (see [R5.03.20] and [R5.03.22] for the case of the small strains) or for the constitutive law hyper elastic (see [R5.03.19]).

Lastly, for the structural elements (beam, plates or shells), there exist specific formulations. One can quote:
- Beams in large displacements (see [R5.03.40]) or multifibre beams in large displacements (see [R3.08.09]). Keyword DEFORMATION = "GROT_GDEP".
- Voluminal shell elements in nonlinear geometrical (see [R3.07.05]).

It does not exist of structural elements (beam, plate or shell) usable in large deformations in *Code_Aster*.

1.2.3 Unilateral contact and friction

For the contact and friction, one will refer to three documents: [R5.03.51] on the discrete contact with friction, [R5.03.52] for the hybrid formulation by elements of contact/friction and [R5.03.53] on the contact in great slidings with method XFEM.

1.3 Position of the nonlinear quasi-static problem

As a consequence of paragraph 1.1, one sees that it is legitimate to consider that the nonlinear problem has like unknown a displacement and that it is parameterized by time. That is to say thus the quasi-static nonlinear problem according to, statement of the principle of the virtual wors:

$$\begin{cases} \mathbf{v}^T.\mathbf{L}^{\text{int}}(\mathbf{u},t) = \mathbf{v}^T.\mathbf{L}^{\text{ext}}(t) \quad \forall \mathbf{v} \text{ tel que } \mathbf{B}.\mathbf{v}=0 \\ \mathbf{B}.\mathbf{u}=\mathbf{u}^d(t) \end{cases} \quad (1)$$

where:
- t represent the variable of time
- $\mathbf{u}$ is the field of displacement taken from a reference configuration
- $\mathbf{v}$ is the kinematically admissible field of virtual displacement

the relation $\mathbf{B}.\mathbf{u}=\mathbf{u}^d(t)$ corresponds to the boundary conditions imposed in displacements (imposed displacements, connections between degrees of freedom,…). $\mathbf{B}$ is a linear *operator* of the space of the fields of displacements on a space of functions defined on part of edge of structure, $\mathbf{u}^d$ is a function given on this part.

This first equation (equation 1) is the classical statement of the principle of the virtual wors. $\mathbf{L}^{\text{ext}}$ is the external mechanical loading to which the structure is subjected (pressure, imposed force,…) and $\mathbf{L}^{\text{int}}$ represents the internal forces of the problem of quasi-static mechanics nonlinear. In the linear case, one has $\mathbf{L}^{\text{int}}(\mathbf{u},t) = \mathbf{K}.\mathbf{u}$, where $\mathbf{K}$ is the stiffness matrix of structure.

In fact, more precisely, $\mathbf{L}^{\text{int}}(\mathbf{u},t)$ is connected to the stress field σ by the operator work of the virtual strains $\mathbf{Q}^T$ according to the following relation:

$$\mathbf{L}^{\text{int}}(\mathbf{u},t)=\mathbf{Q}^T(\mathbf{u}).\sigma \quad (2$$

In small displacements, $\mathbf{Q}^T$ is independent of displacements; for large displacements, it is not any more the case. One gives oneself a discretization of the time interval to calculating:

$$t \rightarrow \left[t_0,\cdots,t_i,\cdots,t_n\right] \quad (3)$$

the stress field σ_i at time t_i is written $\sigma(\mathbf{u}_i,\beta_i,t_i,\mathbf{H}_{i-1})$, if one notes β_i the fields of command variables and $\mathbf{H}_{i-1}$ the last history of structure. For the elastic behaviors, the history does not intervene: the group $\mathbf{H}_{i-1}$ is thus empty. For incremental behaviors, the history is all the states (fields of displacements, stresses and local variables) at previous time: $\mathbf{H}_{i-1}=\left\{\mathbf{u}_{i-1},\sigma_{i-1},\alpha_{i-1},t_{i-1}\right\}$.

In the general case, the dependence of the operator $\mathbf{L}^{\text{int}}$ is, as we saw in [§1.11.1], implicit compared to time: it results from the integration of the behavior model in time (for the problems of elastoplasticity for example). The explicit dependence compared to time appears behavior models in particular in the case of taking into account a phenomenon of hardening by time (time-hardening) or in the case of the aging.

The dualisation of the boundary conditions of Dirichlet $\mathbf{B}.\mathbf{u}=\mathbf{u}^d(t)$ leads to the following problem [R3.03.01]:

$$\begin{cases} \mathbf{L}^{\text{int}}(\mathbf{u},t)+\mathbf{B}^T.\boldsymbol{\lambda}=\mathbf{L}^{\text{ext}}(t) \\ \mathbf{B}.\mathbf{u}=\mathbf{u}^d(t) \end{cases} \quad (4)$$

the unknowns are now, at any moment *t ,* the couple $(\mathbf{u},\boldsymbol{\lambda})$, where $\boldsymbol{\lambda}$ represents *the Lagrange multipliers* of the boundary conditions of Dirichlet [R3.03.01]. The vector $\mathbf{B}^{\mathbf{T}}.\lambda$ is interpreted like the opposite of the reactions of bearing to the corresponding nodes.

The formulation of the quasi-static problem consists in expressing the equilibrium of structure (the internal forces are equal to the external forces) for a continuation of times of computation $\left(t_i\right)_{1\le i\le I}$ which parameterize the loading, one will note the quantities at time t_i by the index i (for example $\mathbf{L}^{\text{int}}(\mathbf{u_i},t_i)=\mathbf{L}_i^{\text{int}}$):

$$\begin{cases} \mathbf{L}_i^{\text{int}}+\mathbf{B}^T.\boldsymbol{\lambda}_i=\mathbf{L}_i^{\text{ext}} \\ \mathbf{B}.\mathbf{u}_i=\mathbf{u}_i^d \end{cases} \quad (5)$$

What amounts cancelling in $(\mathbf{u}_i, \boldsymbol{\lambda}_i, t_i)$ the vector $\mathbf{R}_i(\mathbf{u}_i, \boldsymbol{\lambda}_i, t_i)$ says vector residue of equilibrium, defined by:

$$\mathbf{R}_i(\mathbf{u}_i, \boldsymbol{\lambda}_i, t_i) = \begin{pmatrix} \mathbf{L}_i^{\text{int}} + \mathbf{B}^{\mathbf{T}}.\boldsymbol{\lambda}_i - \mathbf{L}_i^{\text{ext}} \\ \mathbf{B}.\mathbf{u}_i - \mathbf{u}_i^d \end{pmatrix} \quad (6)$$

the state of structure t_0 is supposed to be known by it. One carries out I increments (or not) of load defined on Figure 1 -1.3-a.

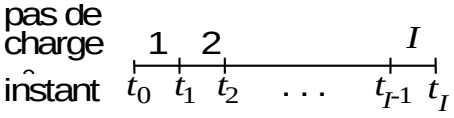


Figure 1 -1.3-a

unknowns is calculated in an incremental way by the total algorithm of resolution (even for the elastic behaviors). From $(\mathbf{u}_{i-1}, \boldsymbol{\lambda}_{i-1})$, solution satisfying the equilibrium in t_{i-1}, one determines $\Delta\mathbf{u}_i$ and $\Delta\boldsymbol{\lambda}_i$ which will make it possible to obtain the solution in t_i:

$$\begin{cases} t_i = t_{i-1} + \Delta t_i \\ \mathbf{u}_i = \mathbf{u}_{i-1} + \Delta\mathbf{u}_i \\ \boldsymbol{\lambda}_i = \boldsymbol{\lambda}_{i-1} + \Delta\boldsymbol{\lambda}_i \end{cases} \quad (7)$$

the increments $\Delta\mathbf{u}_i$ and $\Delta\boldsymbol{\lambda}_i$ are initially estimated by linearizing the problem compared to time around $(\mathbf{u}_{i-1}, \boldsymbol{\lambda}_{i-1}, t_{i-1})$ (phase known as of prediction or Eulerian). Then one uses a method of Newton or one of his alternatives to solve equation 5 in an iterative way (one calculates a continuation $(\Delta\mathbf{u}_i^n, \Delta\boldsymbol{\lambda}_i^n)$ where the exhibitor n is the number of the iteration). Besides these variables, for the incremental behavior models, one needs to know in t_{i-1} the stress field $\boldsymbol{\sigma}_{i-1}$ and the field of local variables $\boldsymbol{\alpha}_{i-1}$ (confer [R5.03.02] for an example).

2 Method of Newton

2.1 Principle of the method

the method of Newton is a classical method of resolution of the equations of the type searches of zero. Let us consider a nonlinear vector $\mathbf{F}$ function of the vector $\mathbf{x}$. One seeks the zero of this function, i.e.:

$$\mathbf{F}(\mathbf{x})=0 \tag{8}$$

the method of Newton consists in building a vector series $\mathbf{x}^n$ converging towards the solution $\mathbf{x}$. To find the new one reiterated $\mathbf{x}^{n+1}$, one approaches $\mathbf{F}(\mathbf{x}^{n+1})$ by a development limited to the order one around $\mathbf{x}^n$ and one expresses who $\mathbf{F}(\mathbf{x}^{n+1})$ must be null:

$$0=\mathbf{F}(\mathbf{x}^{n+1})\approx\mathbf{F}(\mathbf{x}^n)+\mathbf{F}'(\mathbf{x}^n)(\mathbf{x}^{n+1}-\mathbf{x}^n) \tag{9}$$

That is to say still:

$$\mathbf{F}'(\mathbf{x}^n)(\mathbf{x}^{n+1}-\mathbf{x}^n)=-\mathbf{F}(\mathbf{x}^n) \tag{10}$$

Finally:

$$\mathbf{x}^{n+1}=\mathbf{x}^n-\left[\mathbf{F}'(\mathbf{x}^n)\right]^{-1}.\mathbf{F}(\mathbf{x}^n) \tag{11}$$

$\mathbf{F}'(\mathbf{x})$ is the tangent linear application associated with the function $\mathbf{F}$. The derivative at the point $\mathbf{x}$ in the direction $\mathbf{h}$ is defined like following directional derivative:

$$\mathbf{F}'(\mathbf{x}).\mathbf{h}=\lim_{\theta\to 0}\frac{\mathbf{F}(\mathbf{x}+\theta.\mathbf{h})-\mathbf{F}(\mathbf{x})}{\theta} \tag{12}$$

the matrix of $\mathbf{F}'(\mathbf{x})$ in the bases chosen for the vector spaces concerned is called the jacobian matrix of $\mathbf{F}$ at the point $\mathbf{x}$. When $\mathbf{F}$ is a function of an Euclidean vector space with actual values, $\mathbf{F}'(\mathbf{x})$ is a linear form, and one can show that there exists a vector (single), noted $\nabla\mathbf{F}(\mathbf{x})$ and called the gradient of $\mathbf{F}$, such as:

$$\mathbf{F}'(\mathbf{x}).\mathbf{h}=\mathbf{h}^T.\nabla\mathbf{F}(\mathbf{x}) \tag{13}$$

I.e. the scalar product of $\mathbf{h}$ and the gradient of $\mathbf{F}$.

When one is close to the solution, the convergence of the method of Newton is quadratic i.e. the number of zeros after the comma in the error doubles with each iteration (0,19 – 0,036 – 0,0013 – 0,0000017 for example). But this method (using the true tangent) has several disadvantages:

- It requires the computation of the tangent to each iteration, which is all the more expensive as the size of the problem is large (especially if a direct solver is used),
- If the increment is large, the tangent (known as coherent or consistent) can lead to divergences of the algorithm,
- It can not be symmetric, which obliges to use particular solvers.

For this reason one can use other matrixes instead of the tangent matrix: the elastic matrix, a tangent matrix obtained before, the symmetrized tangent matrix,…

2.2 Adaptation of the method of Newton with the posed problem

If the boundary conditions of Dirichlet initially are forgotten, one must solve a system (nonlinear because dependant on $\mathbf{u}_i$) of the form:

$$\mathbf{L}_i^{\text{int}}(\mathbf{u}_i)=\mathbf{L}_i^{\text{méca}}(\mathbf{u}_i) \tag{14}$$

where $\mathbf{L}_i^{\text{méca}}$ from now on, at time, t_i the mechanical part *of* the total external loading will indicate $\mathbf{L}_i^{\text{ext}}$ in order to distinguish it from the thermal loading. Let us note that the mechanical loading $\mathbf{L}_i^{\text{méca}}(\mathbf{u}_i)$ can in the

case of depend on $\mathbf{u}_i$ displacements the forces known as "following" like the pressure or the centrifugal force (see [§2.72.7]). By means of the notations of [§2.12.1], that amounts cancelling the vector function $\mathbf{R}$ definite by:

$$\mathbf{R}(\mathbf{u}_i, t_i) = \mathbf{L}_i^{\text{int}} - \mathbf{L}_i^{\text{méca}} \tag{15}$$

the internal forces $\mathbf{L}_i^{\text{int}}$ can ***symbolically*** be noted $\mathbf{Q}_i^T.\boldsymbol{\sigma}_i$, where $\mathbf{Q}_i^T$ is the matrix associated with the operator divergence (statement of the work of the strain field virtual). The internal forces are expressed then:

$$\mathbf{L}_i^{\text{int}} = \mathbf{Q}_i^T.\boldsymbol{\sigma}_i = \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w}_i) : \boldsymbol{\sigma}(\mathbf{u}_i).d\Omega \tag{16}$$

And the forces of the mechanical loading:

$$\mathbf{L}_i^{\text{méca}} = \int_\Omega \mathbf{f}_i.\mathbf{w}_i.d\Omega + \int_\Gamma \mathbf{t}_i.\mathbf{w}_i.d\Gamma \tag{17}$$

Where:

- $\mathbf{w}_i$ indicate the field of virtual displacements;
- $\mathbf{f}_i$ indicate the volume forces applying to time t_i to Ω ;
- $\mathbf{t}_i$ indicate the surface forces applying to time t_i to the border Γ of Ω .

The application of the method of Newton results in solving a linear succession of problems of the type (n is the number of the iteration of Newton, i that of the variable of time):

$$\mathbf{K}_i^n.\Delta\mathbf{u}_i^{n+1} = \mathbf{L}_i^{\text{méca},n} - \mathbf{L}_i^{\text{int},n} \tag{18}$$

One notes $\delta\mathbf{u}_i^{n+1} = \mathbf{u}_i^{n+1} - \mathbf{u}_i^n$ the displacement increment between two successive iterations of Newton. The matrix $\mathbf{K}_i^n$ is the tangent stiffness matrix in $\mathbf{u}_i^n$ and the vector $\mathbf{L}_i^{\text{int},n}$ represents the internal forces with $n^{\text{ième}}$ the iteration of Newton of $i^{\text{ème}}$ the step of load. The quantity $\mathbf{R}_i^n = \left(\mathbf{L}_i^{\text{méca},n} - \mathbf{L}_i^{\text{int},n}\right)$ represents the not balanced forces, which one calls the "residue of equilibrium". The matrix $\mathbf{K}_i^n$ is the matrix of the tangent linear application of the function $\mathbf{R}_i^n$, it is thus worth:

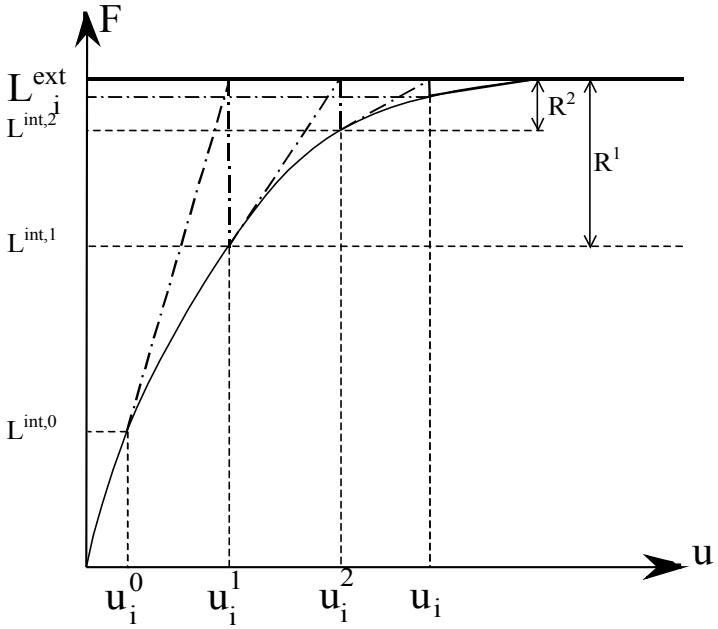
$$\mathbf{K}_i^n = \left.\frac{\partial\mathbf{R}_i^n}{\partial\mathbf{u}}\right|_{(\mathbf{u}_i^n, t_i)} = \left.\frac{\partial\mathbf{L}_i^{\text{int},n}}{\partial\mathbf{u}}\right|_{(\mathbf{u}_i^n, t_i)} - \left.\frac{\partial\mathbf{L}_i^{\text{méca},n}}{\partial\mathbf{u}}\right|_{(\mathbf{u}_i^n, t_i)} \tag{19}$$

In the absence of follower forces [§2.72.7], the second term is null. It thus remains of the matrix $\mathbf{K}_i^n$ only the derivative at the point $\mathbf{u}_i^n$ of the internal forces compared to displacements:

$$\mathbf{K}_i^n = \left.\frac{\partial\mathbf{L}_i^{\text{int},n}}{\partial\mathbf{u}}\right|_{(\mathbf{u}_i^n, t_i)} \tag{20}$$

a small error in the evaluating of the internal forces can have serious consequences, because it is their exact computation which guarantees, if one converges, that it will be towards the sought solution. On the other hand, it is not always necessary to use the true tangent matrix, whose computation and factorization are expensive. For example, an alternative of the method uses the elastic matrix $\mathbf{K}_{\text{élas}}$. The method using the true tangent matrix $\mathbf{K}_i^n$ (known as also coherent or consistent matrix) is called the method of Newton; the methods using of other matrixes (as for example the elastic matrix $\mathbf{K}_{\text{élas}}$) are called methods of Newton modified or methods of quasi-Newton. The choice between a tangent matrix (the last obtained or a preceding matrix) and an elastic matrix is carried out in *Code_Aster* via " `TANGENT" key word MATRICE=` or "`ELASTIC" MATRICE=` of factor key word `the NEWTON`. Moreover, it is possible to use a matrix of discharge, i.e. of a matrix with constant local variables (the evolution of nonthe linearities is thus not taken into account in this matrix), below some time step, for certain constitutive laws. One will refer to documentation [U4.51.03] for the use of this functionality.

The method of Newton with consistent tangent matrix can be illustrated simply using the diagram of [Figure 2.2.1-a].



Appear 2.2.1-a

When one takes into account the conditions of displacements imposed by dualisation1Dans[1], the system to be solved is written:

$$\begin{cases} \mathbf{L}_i^{\text{int}} + \mathbf{B}^T . \boldsymbol{\lambda}_i = \mathbf{L}_i^{\text{méca}} \\ \mathbf{B} . \mathbf{u}_i = \mathbf{u}_i^d \end{cases} \tag{21}$$

One will use the symbol Δ to note the increments since the preceding equilibrium (in t_{i-1}) of the various quantities. One will use a method of Newton to solve this system. However, the experiment shows that the convergence of the method of Newton is strongly dependant on a wise choice of the initial estimate: “more the initial estimate is close to the solution, plus the algorithm converges quickly”. To start the iterative process of the method, it is thus useful to determine “a good” initial increment $(\Delta \mathbf{u}_i^0, \Delta \boldsymbol{\lambda}_i^0)$. For that, one linearizes compared to time the continuous problem: it is what is called the phase of *prediction*. One connects with the loop of the iterations of Newton which makes it possible, with convergence, to obtain the values of $(\Delta \mathbf{u}_i, \Delta \boldsymbol{\lambda}_i)$, and thus those from $(\mathbf{u}_i, \boldsymbol{\lambda}_i)$ application of the equation (7):

$$\begin{cases} \mathbf{u}_i = \mathbf{u}_{i-1} + \Delta \mathbf{u}_i \\ \boldsymbol{\lambda}_i = \boldsymbol{\lambda}_{i-1} + \Delta \boldsymbol{\lambda}_i \end{cases} \tag{22}$$

2.3 Phase of prediction of Eulerian

2.3.1 Linearization

One will thus linearize system 21 compared to time around $(\mathbf{u}_{i-1}, \boldsymbol{\lambda}_{i-1})$. One starts by linearizing the internal forces $\mathbf{L}_i^{\text{int}}$:

$$\mathbf{L}_i^{\text{int}} \approx \mathbf{L}_{i-1}^{\text{int}} + \left. \frac{\partial \mathbf{L}^{\text{int}}}{\partial \mathbf{u}} \right|_{\mathbf{u}_{i-1}} . \Delta \mathbf{u}_i^0 + \left. \frac{\delta \mathbf{L}^{\text{int}}}{\delta t} \right|_{t_{i-1}} \tag{23}$$

the linearization of the reactions of bearing $\mathbf{B}^T . \boldsymbol{\lambda}_i$ is immediate because it is supposed that the matrix $\mathbf{B}$ is constant (it does not depend on displacements or time). Like $\boldsymbol{\lambda}_i = \Delta \boldsymbol{\lambda}_i^0 + \boldsymbol{\lambda}_{i-1}$, it comes immediately:

1 the case where the conditions of imposed displacements are treated by elimination (operator AFFE_CHAR_CINE), the system to be solved is given by the equation 14.

$$\mathbf{B}^T.\boldsymbol{\lambda}_i=\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0+\mathbf{B}^T.\boldsymbol{\lambda}_{i-1} \tag{24}$$

It is supposed that the mechanical loading does not depend on time (the following loads are excluded) and that the limiting conditions of Dirichlet are also linear, therefore:

$$\begin{cases}\mathbf{L}_i^{\text{méca}}=\mathbf{L}_{i-1}^{\text{méca}}+\Delta\,\mathbf{L}_i^{\text{méca}}\\ \mathbf{u}_i^d=\mathbf{u}_{i-1}^d+\Delta\,\mathbf{u}_i^d\end{cases} \tag{25}$$

By reinjecting 23 24 and 25 in the first equation of system 21, one obtains for the balance equation:

$$\mathbf{L}_{i-1}^{\text{int}}+\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}.\Delta\,\mathbf{u}_i^0+\left.\frac{\delta\mathbf{L}^{\text{int}}}{\delta t}\right|_{t_{i-1}}.\Delta t_i+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0+\mathbf{B}^T.\boldsymbol{\lambda}_{i-1}=\mathbf{L}_{i-1}^{\text{méca}}+\Delta\,\mathbf{L}_i^{\text{méca}} \tag{26}$$

There is equilibrium at time t_{i-1}, i.e.:

$$\mathbf{L}_{i-1}^{\text{int}}+\mathbf{B}^T.\boldsymbol{\lambda}_{i-1}=\mathbf{L}_{i-1}^{\text{méca}} \tag{27}$$

And it thus remains:

$$\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}.\Delta\,\mathbf{u}_i^0+\left.\frac{\delta\mathbf{L}^{\text{int}}}{\delta t}\right|_{t_{i-1}}.\Delta t_i+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0=\Delta\,\mathbf{L}_i^{\text{méca}} \tag{28}$$

If one now looks at the second equation of system 21, one obtains for the limiting conditions of Dirichlet:

$$\mathbf{B}.(\mathbf{u}_{i-1}+\Delta\,\mathbf{u}_i^0)=\mathbf{u}_{i-1}^d+\Delta\,\mathbf{u}_i^d \tag{29}$$

There is equilibrium at time t_{i-1}, i.e.:

$$\mathbf{B}.\mathbf{u}_{i-1}=\mathbf{u}_{i-1}^d \tag{30}$$

It remains finally:

$$\mathbf{B}.\Delta\,\mathbf{u}_i^0=\Delta\,\mathbf{u}_i^d \tag{31}$$

One obtains the system of equations allowing to calculate predictive values $(\Delta\,\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0)$:

$$\begin{cases}\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}.\Delta\,\mathbf{u}_i^0+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0=\Delta\,\mathbf{L}_i^{\text{méca}}-\left.\frac{\delta\mathbf{L}^{\text{int}}}{\delta t}\right|_{t_{i-1}}.\Delta t_i\\ \mathbf{B}.\Delta\,\mathbf{u}_i^0=\Delta\,\mathbf{u}_i^d\end{cases} \tag{32}$$

2.3.2 tangent Matrix of prediction

the quantity $\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}$ indicates derivative partial at constant time of $\mathbf{L}_{i-1}^{\text{int}}$, it can develop:

$$\mathbf{K}_{i-1}=\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}=\left(\mathbf{Q}_{i-1}^T.\left.\frac{\partial\boldsymbol{\sigma}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}+\left.\frac{\partial\mathbf{Q}^T}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}.\boldsymbol{\sigma}_{i-1}\right) \tag{33}$$

the matrix $\mathbf{K}_{i-1}$ is called tangent matrix of prediction. The dependence of the matrix $\mathbf{Q}$ compared to displacements is neglected on the assumption of small displacements: the term $\left.\frac{\partial\mathbf{Q}^T}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}$, known as term of geometrical stiffness, thus disappears from 33. This term is taken into account for the great transformations (see §1.2.25). For the developers, let us specify that the computation of the tangent matrix at the time of the phase of prediction is carried out via computation option `RIGI_MECA_TANG`.

2.3.3 Second member vector of the command variables

a command variable $\beta(t)$ is a scalar quantity, function of time and espace2Pour[2], given a priori by the user via keyword `AFFE_VARC` in operator `AFFE_MATERIAU`. It is *a parameter* of the problem and not *an unknown*. The

2 to be more precise, a command variable is assigned to a mesh. It thus is not about a function of the geometry (and thus of displacements), but of the topology of the mesh.

quantity $\left.\frac{\delta \mathbf{L}^{\text{int}}}{\delta t}\right|_{t_{i-1}}$ indicates the partial differential, compared to t and with $\mathbf{u}$ constant, of $\mathbf{L}^{\text{int}}=\mathbf{Q}.\boldsymbol{\sigma}(t,\beta(t))$. The purpose of this particular notation is drawing attention to the fact that for n_{varc} command variables, one writes the total differential:

$$\frac{\delta \boldsymbol{\sigma}}{\delta t}=\frac{\partial \boldsymbol{\sigma}}{\partial t}+\sum_{j=1,n_{varc}}\frac{\delta \boldsymbol{\sigma}}{\delta \beta^j}.\frac{\delta \beta^j}{\delta t} \tag{34}$$

If one takes as example the command variable describing the temperature θ:

$$\frac{\delta \boldsymbol{\sigma}}{\delta t}=\frac{\partial \boldsymbol{\sigma}}{\partial t}+\frac{\delta \boldsymbol{\sigma}}{\delta \theta}.\frac{\delta \theta}{\delta t} \tag{35}$$

It is supposed that the temperature varies linearly between two times:

$$\frac{\delta \theta}{\delta t}=\frac{\Delta \theta_i}{\Delta t_i} \tag{36}$$

the dependence from σ ratio with time and compared to the temperature makes it possible to write:

$$\left.\frac{\delta \mathbf{L}^{int}}{\delta t}\right|_{t_{i-1}}=\frac{\delta}{\delta t}\left(\mathbf{Q}_{i-1}^T.\boldsymbol{\sigma}_{i-1}\right)=\mathbf{Q}_{i-1}^T.\left(\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{t_{i-1}}+\left.\frac{\partial \boldsymbol{\sigma}}{\partial \theta}\right|_{t_{i-1}}.\frac{\Delta \theta_i}{\Delta t_i}\right) \tag{37}$$

Because $\mathbf{Q}$ does not depend on time and thus $\frac{\delta}{\delta t}\left(\mathbf{Q}^T\right)=0$. The vector $\Delta \mathbf{L}_i^{\text{varc}}$, whose statement is given by 38, is the increment of loading of temperature (attention to the change of sign!) which one generalized with all *the command variables*: metallurgical temperature, irradiation, phases (see *[R4.04.02]),*…

$$\Delta \mathbf{L}_i^{varc}=-\left.\frac{\delta \mathbf{L}^{int}}{\delta t}\right|_{t_{i-1}}.\Delta t_i=-\mathbf{Q}_{i-1}^T.\left(\left.\frac{\partial \boldsymbol{\sigma}}{\partial t}\right|_{t_{i-1}}.\Delta t_i+\sum_{j=1,n_{varc}}\left(\left.\frac{\partial \boldsymbol{\sigma}}{\partial \beta^j}\right|_{t_{i-1}}.\Delta \beta_i^j\right)\right) \tag{38}$$

One currently does not take account of the explicit dependence of the stresses compared to time and thus the first term of 38 is worth zero. And thus finally:

$$\Delta \mathbf{L}_i^{varc}=-\mathbf{Q}_{i-1}^T.\left(\sum_{j=1,n_{varc}}\left(\left.\frac{\partial \boldsymbol{\sigma}}{\partial \beta^j}\right|_{t_{i-1}}.\Delta \beta_i^j\right)\right) \tag{39}$$

the increment of loading *of the command variables* $\Delta \mathbf{L}_i^{\text{varc}}$, resulting from derivative of the internal forces compared to the command variables is an estimate of the effect of a variation of the command variables.

In the case of the temperature, if K the hydrostatic bulk modulus and α the thermal coefficient of thermal expansion are noted, the thermal stress is written:

$$\boldsymbol{\sigma}_i^{\text{ther}}=-3.K.\alpha.\Delta \theta_i.\mathbf{I}+\boldsymbol{\sigma}_{i-1}^{\text{ther}} \quad \text{if } \Delta \theta_i=\theta_i-\theta_{i-1} \tag{40}$$

Where $\mathbf{I}$ is the matrix identity. And thus, by applying 39:

$$\Delta \mathbf{L}_i^{ther}=-\left(\mathbf{Q}_{i-1}^T.\left.\frac{\partial \boldsymbol{\sigma}}{\partial \theta}\right|_{t_{i-1}}\right).\Delta \theta_i=3.K.\alpha.\Delta \theta_i.\left(\mathbf{Q}_{i-1}^T.\mathbf{I}\right) \tag{41}$$

In the elastic case, they are the internal forces associated with a thermal thermal expansion (it is not strictly speaking a loading, that is assimilated rather to the effect of an initial strain). This estimate is used in the phase of prediction and the stopping criteria. If thermal thermal expansions make leave structure of the elastic domain (plasticity for example), this estimate will be corrected during the iterations of Newton.

2.3.4 Second member vector of the mechanical loading

the mechanical increment of *loading* $\Delta \mathbf{L}_i^{\text{méca}}$ is composed of the dead loads (independent of the geometry, like gravity) and of the following loads. Actually, there exist cases (the first increment of load, for example) where $\mathbf{L}_{i-1}^{\text{méca}}$ is unknown. It is pointed out that the increment of loading (25) is written:

$$\mathbf{L}_i^{\text{méca}} = \mathbf{L}_{i-1}^{\text{méca}} + \Delta \mathbf{L}_i^{\text{méca}} \tag{42}$$

There is equilibrium at time t_{i-1} , therefore by applying 27:

$$\Delta \mathbf{L}_i^{\text{méca}} = \mathbf{L}_i^{\text{méca}} - \mathbf{L}_{i-1}^{\text{int}} - \mathbf{B}^T . \boldsymbol{\lambda}_{i-1} \tag{43}$$

the statement of the internal forces to time step preceding $\mathbf{L}_{i-1}^{\text{int}}$ implies either to save this vector of preceding computation if there exists (taken again of a former computation), or to integrate the constitutive law from the initial state given by the user (what can be expensive). By preoccupation of simplicity and an effectiveness, one chooses not to reinstate the constitutive law and one expresses simply the internal forces like the nodal forces by taking the stresses known at this time, that is to say:

$$\mathbf{L}_{i-1}^{int} = \mathbf{Q}_{i-1}^T . \boldsymbol{\sigma}_{i-1} \tag{44}$$

From where the new statement:

$$\Delta \mathbf{L}_i^{\text{méca}} = \mathbf{L}_i^{\text{méca}} - \mathbf{Q}_{i-1}^T . \boldsymbol{\sigma}_{i-1} - \mathbf{B}^T . \boldsymbol{\lambda}_{i-1} \tag{45}$$

The computation direct from 44 request with the user to take care of coherence enters the field of the stresses, the fields of displacements and local variables (DEPL , SIGM and VARI in ETAT_INIT). with respect to the integration of the constitutive law in the case of a resumption of computation.

2.3.5 Linear system

By reinjecting the statement of $\left.\frac{\partial \mathbf{L}^{\text{int}}}{\partial \mathbf{u}}\right|_{\mathbf{u}_{i-1}}$ (equation 33), from $\left.\frac{\delta \mathbf{L}^{\text{int}}}{\delta t}\right|_{t_{i-1}}$ (equation 38) and $\Delta \mathbf{L}_i^{\text{méca}}$ (equation 45) in 32, the system of equations in prediction is written:

$$\begin{cases} \mathbf{K}_{i-1} . \Delta \mathbf{u}_i^0 + \mathbf{B}^T . \Delta \boldsymbol{\lambda}_i^0 = \mathbf{L}_i^{\text{méca}} - \mathbf{Q}_{i-1}^T . \boldsymbol{\sigma}_{i-1} - \mathbf{B}^T . \boldsymbol{\lambda}_{i-1} + \Delta \mathbf{L}_i^{\text{varc}} \\ \mathbf{B} . \Delta \mathbf{u}_i^0 = \Delta \mathbf{u}_i^d \end{cases} \tag{46}$$

It will be noticed whereas this statement utilized from now on the Lagrange multipliers at previous *time*, which are sometimes unknown (with the first increment of load, for example). What wants to say that with this new statement, one moved the problem of the knowledge of the internal forces at time t_{i-1} towards the ignorance of the Lagrange multipliers at this same time! But one will see that the fact that the limiting conditions are linear us "saves". Let us consider that the solution of 46 with regard to the Lagrange multipliers $\hat{\boldsymbol{\lambda}}_i$ is written in incremental form:

$$\hat{\boldsymbol{\lambda}}_i = \hat{\boldsymbol{\lambda}}_{i-1} + \Delta \hat{\boldsymbol{\lambda}}_i \tag{47}$$

This solution solves the first equation of the system:

$$\mathbf{K}_{i-1} . \Delta \mathbf{u}_i^0 + \mathbf{B}^T . \Delta \hat{\boldsymbol{\lambda}}_i^0 = \mathbf{L}_i^{\text{méca}} - \mathbf{Q}_{i-1}^T . \boldsymbol{\sigma}_{i-1} - \mathbf{B}^T . \hat{\boldsymbol{\lambda}}_{i-1} + \Delta \mathbf{L}_i^{\text{varc}} \tag{48}$$

the idea is to search it $\hat{\boldsymbol{\lambda}}_i$. As the operator $\mathbf{B}^T$ is constant, by applying 47, one a:

$$\mathbf{K}_{i-1} . \Delta \mathbf{u}_i^0 + \mathbf{B}^T . \hat{\boldsymbol{\lambda}}_i = \mathbf{L}_i^{\text{méca}} - \mathbf{Q}_{i-1}^T . \boldsymbol{\sigma}_{i-1} + \Delta \mathbf{L}_i^{\text{varc}} \tag{49}$$

One supposes that the limiting conditions are checked, therefore:

$$\mathbf{B} . \Delta \mathbf{u}_i^0 = \Delta \mathbf{u}_i^d \tag{50}$$

imposed displacements are also written in incremental form:

$$\mathbf{u}_i^d = \mathbf{u}_{i-1}^d + \Delta \mathbf{u}_i^d \tag{51}$$

the matrix $\mathbf{B}$ is constant, we thus had with the preceding increment (the problem was solved):

$$\mathbf{B} . \mathbf{u}_{i-1} = \mathbf{u}_{i-1}^d \tag{52}$$

By means of 51 and 52 in 50 :

$$\mathbf{B}.\Delta\mathbf{u}_i^0=\Delta\mathbf{u}_i^d=\mathbf{u}_i^d-\mathbf{u}_{i-1}^d=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_{i-1} \quad (53)$$

With the equilibrium, one thus has $\hat{\boldsymbol{\lambda}}_i$ who satisfies also the limiting conditions that one rewrites:

$$\mathbf{B}.\Delta\mathbf{u}_i^0=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_{i-1} \quad (54)$$

the vector of the Lagrange multipliers $\hat{\boldsymbol{\lambda}}_i$ can thus be found at the time of the phase of prediction by modifying the equation of imposition of the limiting conditions by statement 54. By analogy with the increment of the displacements found in prediction $\Delta\mathbf{u}_i^0$ one will note $\Delta\boldsymbol{\lambda}_i^0=\hat{\boldsymbol{\lambda}}_i$:

$$\begin{cases}\mathbf{K}_{i-1}.\Delta\mathbf{u}_i^0+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0=\mathbf{L}_i^{\text{méca}}-\mathbf{Q}_{i-1}^T.\boldsymbol{\sigma}_{i-1}+\Delta\mathbf{L}_i^{\text{varc}}\\ \mathbf{B}.\Delta\mathbf{u}_i^0=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_{i-1}\end{cases} \quad (55)$$

a cas particulier relates to the use of an excitation of the type TYPE_CHARGE= " DIDI " meaning Dirichlet differential, i.e. compared to the initial state. That consists, for the boundary conditions of the blockings type, to impose, not $\mathbf{B}.\mathbf{u}=\mathbf{u}^d$, but $\mathbf{B}.(\mathbf{u}-\mathbf{u}_{\text{didi}})=\mathbf{u}^d$ In this case, the system to be solved in the phase of prediction for the new increment of load is:

$$\begin{cases}\mathbf{K}_{i-1}.\Delta\mathbf{u}_i^0+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0=\mathbf{L}_i^{\text{méca}}-\mathbf{Q}_{i-1}^T.\boldsymbol{\sigma}_{i-1}+\Delta\mathbf{L}_i^{\text{varc}}\\ \mathbf{B}.\Delta\mathbf{u}_i^0=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_{i-1}+\mathbf{B}.\mathbf{u}_{\text{didi}}\end{cases} \quad (56)$$

2.3.6 Alternatives of the prediction

There exist other options of prediction available in STAT_NON_LINE.

- One can use an elastic matrix $\mathbf{K}_{\text{élas}}$ instead of the tangent matrix of velocity $\mathbf{K}_{i-1}$, it is option PREDICTION=' ELASTIQUE' (option RIGI_MECA).
- One can use a secant matrix $\mathbf{K}_{\text{sécante}}$ instead of the tangent matrix of velocity $\mathbf{K}_{i-1}$, it is option PREDICTION=' SECANTE' (option RIGI_MECA_ELAS). The secant matrix is an elastic matrix whose Young's modulus is used by applying the damage (see for example [R5.03.18] for more details)
- One can use a displacement increment previously calculated instead of the estimate, it is option PREDICTION=' DEPL_CALCULE'. In this case one makes no inversion of system and $\Delta\mathbf{u}_i^0$ is directly given. Confer to the documentation [U4.51.03] for its use.
- One can use a displacement increment extrapolated compared to the preceding step. One time step calculates the estimate of the displacement increment from the total increment obtained like solution with preceding (balanced by the ratio of time step). It is option PREDICTION=' EXTRAPOL'.

In these the last two cases, in order to ensure that initial displacement is kinematically admissible, one projects the estimate on all the kinematically admissible fields (i.e satisfying the boundary conditions with Dirichlet) according to the norm given by the elastic matrix ***, which*** must thus be calculated. Phase

2.4 of correction of Newton Principle

2.4.1 At the conclusion

of the phase of prediction, we find ourselves with an estimate of the increment of displacements and $\Delta\mathbf{u}_i^0$ increment of the Lagrange multipliers. $\Delta\boldsymbol{\lambda}_i^0$ If this estimate is exact (modulo the application of the convergence criteria described to the §2.521 then one obtains the solution converged ***for*** time step: t_i (57

$$\begin{cases}\mathbf{u}_i^{\text{convergé}}=\mathbf{u}_{i-1}+\Delta\mathbf{u}_i^0\\ \boldsymbol{\lambda}_i^{\text{convergé}}=\boldsymbol{\lambda}_{i-1}+\Delta\boldsymbol{\lambda}_i^0\end{cases} \quad)57$$

if it is not the case, one must find the values of $(\Delta\mathbf{u}_i,\Delta\boldsymbol{\lambda}_i)$ the displacement increments and the Lagrange multipliers since the values obtained $(\mathbf{u}_{i-1},\boldsymbol{\lambda}_{i-1})$ with the preceding equilibrium (urgent): t_{i-1} (58

$$\begin{cases}\mathbf{u}_i^{\text{convergé}}=\mathbf{u}_{i-1}+\Delta\mathbf{u}_i\\ \boldsymbol{\lambda}_i^{\text{convergé}}=\boldsymbol{\lambda}_{i-1}+\Delta\boldsymbol{\lambda}_i\end{cases} \quad (58$$

the phase of prediction converged, one thus has crudely: (59

$$(\Delta\mathbf{u}_i,\Delta\boldsymbol{\lambda}_i)=(\Delta\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0) \quad (59$$

, one takes as initial values obtained $(\Delta\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0)$ at the conclusion of the phase of prediction, before correcting by the iterations of $(\delta\mathbf{u}_i^n,\delta\boldsymbol{\lambda}_i^n)$ the method of Newton. With a sufficient number n_{CV} of iterations of Newton (always with the arbitration of the convergence criterion): (60

$$\begin{cases}\mathbf{u}_i^{\text{convergé}}=\mathbf{u}_{i-1}+\Delta\mathbf{u}_i^0+\sum_{j=1}^{n=n_{CV}}\delta\mathbf{u}_i^j\\ \boldsymbol{\lambda}_i^{\text{convergé}}=\boldsymbol{\lambda}_{i-1}+\Delta\boldsymbol{\lambda}_i^0+\sum_{j=1}^{n=n_{CV}}\delta\boldsymbol{\lambda}_i^j\end{cases} \quad (60$$

one did not converge (if the nombre of iterations of Newton is not sufficient), one notes: (61

$$(\Delta\mathbf{u}_i^n,\Delta\boldsymbol{\lambda}_i^n)=(\Delta\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0)+\sum_{j=1}^{n}(\delta\mathbf{u}_i^j,\delta\boldsymbol{\lambda}_i^j) \quad (61$$

total displacement, for time step and i the iteration of Newton n will thus be written: (62

$$(\mathbf{u}_i^n,\boldsymbol{\lambda}_i^n)=(\mathbf{u}_{i-1},\boldsymbol{\lambda}_{i-1})+(\Delta\mathbf{u}_i^n,\Delta\boldsymbol{\lambda}_i^n) \quad (62$$

each iteration, one must solve a system allowing to determine, $(\delta\mathbf{u}_i^n,\delta\boldsymbol{\lambda}_i^n)$ increments of displacements and the Lagrange multipliers since result of $(\mathbf{u}_i^{n-1},\boldsymbol{\lambda}_i^{n-1})$ the preceding iteration: (63

$$(\mathbf{u}_i^{n-1},\boldsymbol{\lambda}_i^{n-1})=(\mathbf{u}_{i-1},\boldsymbol{\lambda}_{i-1})+(\Delta\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0)+\sum_{j=1}^{n-1}(\delta\mathbf{u}_i^j,\delta\boldsymbol{\lambda}_i^j) \quad (63$$

also notes: (64

$$(\Delta\mathbf{u}_i^{n-1},\Delta\boldsymbol{\lambda}_i^{n-1})=(\Delta\mathbf{u}_i^0,\Delta\boldsymbol{\lambda}_i^0)+\sum_{j=1}^{n-1}(\delta\mathbf{u}_i^j,\delta\boldsymbol{\lambda}_i^j) \quad (64$$

still: (65

$$(\mathbf{u}_i^n,\boldsymbol{\lambda}_i^n)=(\mathbf{u}_i^{n-1},\boldsymbol{\lambda}_i^{n-1})+(\delta\mathbf{u}_i^n,\delta\boldsymbol{\lambda}_i^n)=(\mathbf{u}_{i-1},\boldsymbol{\lambda}_{i-1})+(\Delta\mathbf{u}_i^{n-1},\Delta\boldsymbol{\lambda}_i^{n-1})+(\delta\mathbf{u}_i^n,\delta\boldsymbol{\lambda}_i^n) \quad (65$$

2.4.2 One

must linearize system 21 21 the unknowns in with $(\mathbf{u}_i^n,\boldsymbol{\lambda}_i^n)$ constant t_i . One starts by linearizing the internal forces: $\mathbf{L}_i^{\text{int},n}$ (66

$$\mathbf{L}_i^{\text{int},n}\approx\mathbf{L}_i^{\text{int},n-1}+\left.\frac{\partial\mathbf{L}^{\text{int}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_i^{n-1}}.\delta\mathbf{u}_i^n \quad (66$$

the linearization of the reactions of bearing is $\mathbf{B}^T.\boldsymbol{\lambda}_i^n$ immediate: (67

$$\mathbf{B}^T.\boldsymbol{\lambda}_i^n=\mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}+\mathbf{B}^T.\delta\boldsymbol{\lambda}_i^n \quad (67$$

is supposed that the mechanical loading does not depend on time (the following loads are excluded) and that the limiting conditions of Dirichlet are also linear. Linear

2.4.3 system By

reinjecting 66 66 67 67 the first equation of system 21 21 obtains for the balance equation: (68

$$\mathbf{L}_i^{\text{int},n-1}+\left.\frac{\partial \mathbf{L}^{\text{int}}}{\partial \mathbf{u}}\right|_{\mathbf{u}_i^{n-1}}.\delta \mathbf{u}_i^n+\mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}+\mathbf{B}^T.\delta \boldsymbol{\lambda}_i^n=\mathbf{L}_i^{\text{méca}} \quad (68$$

the quantity $\left.\frac{\partial \mathbf{L}^{\text{int}}}{\partial \mathbf{u}}\right|_{\mathbf{u}_i^{n-1}}$ is called coherent tangent matrix and it is noted: $\mathbf{K}_i^{n-1}$ (69

$$\mathbf{K}_i^{n-1}=\left.\frac{\partial \mathbf{L}^{\text{int}}}{\partial \mathbf{u}}\right|_{\mathbf{u}_i^{n-1}} \quad (69$$

the limiting conditions, the linearization of system 21 21 us in a way similar to 54 5470

$$\mathbf{B}.\delta \mathbf{u}_i^n=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_i^{n-1} \quad (70$$

the system to be solved is written finally: (71

$$\begin{cases}\mathbf{K}_i^{n-1}.\delta \mathbf{u}_i^n+\mathbf{B}^T.\delta \boldsymbol{\lambda}_i^n=\mathbf{L}_i^{\text{méca}}-\mathbf{L}_i^{\text{int},n-1}-\mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}\\ \mathbf{B}.\delta \mathbf{u}_i^n=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_i^{n-1}\end{cases} \quad (71$$

the vector of the internal forces $\mathbf{L}_i^{\text{int},n-1}$ is calculated starting from the stresses. $\boldsymbol{\sigma}_i^{n-1}$ Those being calculated starting from displacements via $\mathbf{u}_i^{n-1}$ the behavior model of the material [§1.11.1In fact, in the case of incremental behaviors , $\boldsymbol{\sigma}_i^{n-1}$ is calculated from and $(\boldsymbol{\sigma}_{i-1},\boldsymbol{\alpha}_{i-1})$ of the increment of strain induced $\boldsymbol{\varepsilon}(\Delta \mathbf{u}_i^{n-1})$ by the displacement increment since the beginning of the iterative process (including the phase of prediction) or by the gradient of the transformation in the case of $\mathbf{F}$ the great transformations. Alternatives

2.4.4 of the correction Methods

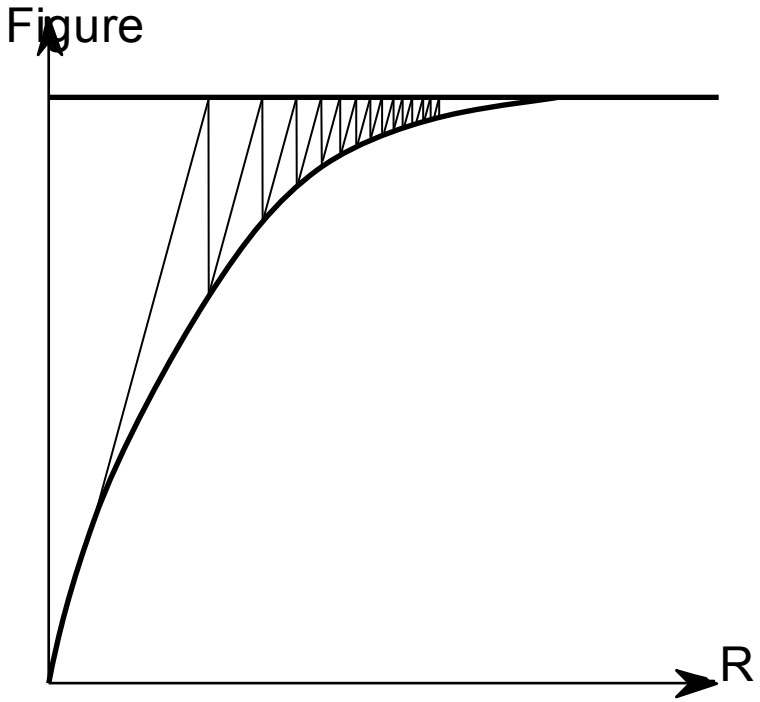
2.4.4.1 of quasi-Newton As

in the phase of prediction, one is not obliged to use the true tangent matrix. $\mathbf{K}_i^{n-1}$ In particular, operator STAT_NON_LINE `authorizes` the use of the elastic matrix, $\mathbf{K}_{\text{élas}}$ or the reactualization of the tangent matrix all time step i_0 (key word REAC_INCR) `or` all the iterations n_0 of Newton (key word REAC_ITER). Thus, the matrix perhaps $\mathbf{K}_i^{n-1}$ replaced by a matrix, $\mathbf{K}_j^{n-1}$ with, $j\leq i$ or a matrix, $\mathbf{K}_i^m$ with. $m\leq n-1$ Caution: a “stiff” matrix too does not pose problems of stability but can produce a very slow convergence; a “flexible” matrix too can lead to divergence, it is advised in this case to make linear search [§3].3It

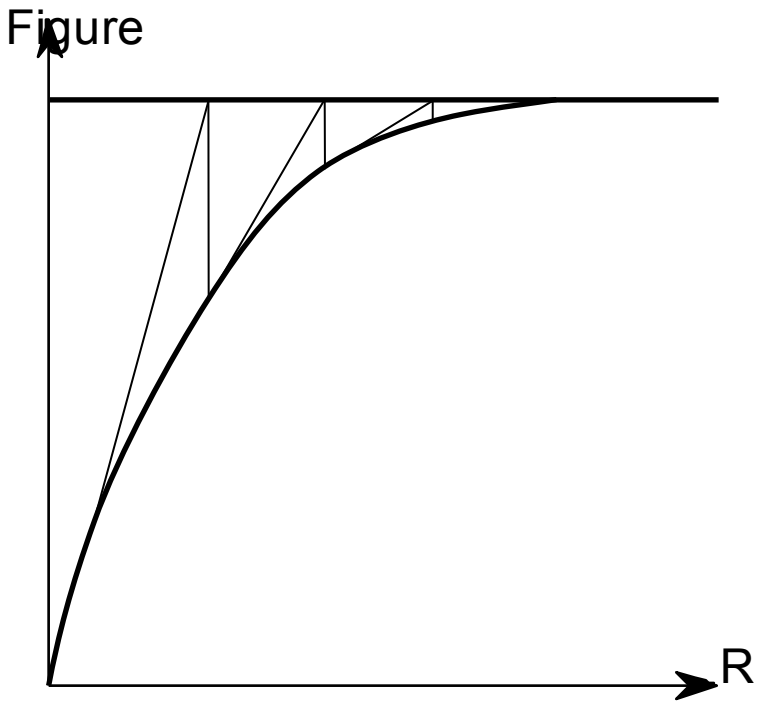
is difficult to give a rule making it possible to know when one must reactualize the tangent matrix: that strongly depends on the degree of nonlinearity of the problem and the size of the increments of load. In

discharge, it is recommended either to use the elastic matrix for the phase of prediction and resolution, or to use the elastic matrix for the phase of prediction then the tangent matrix for the resolution.

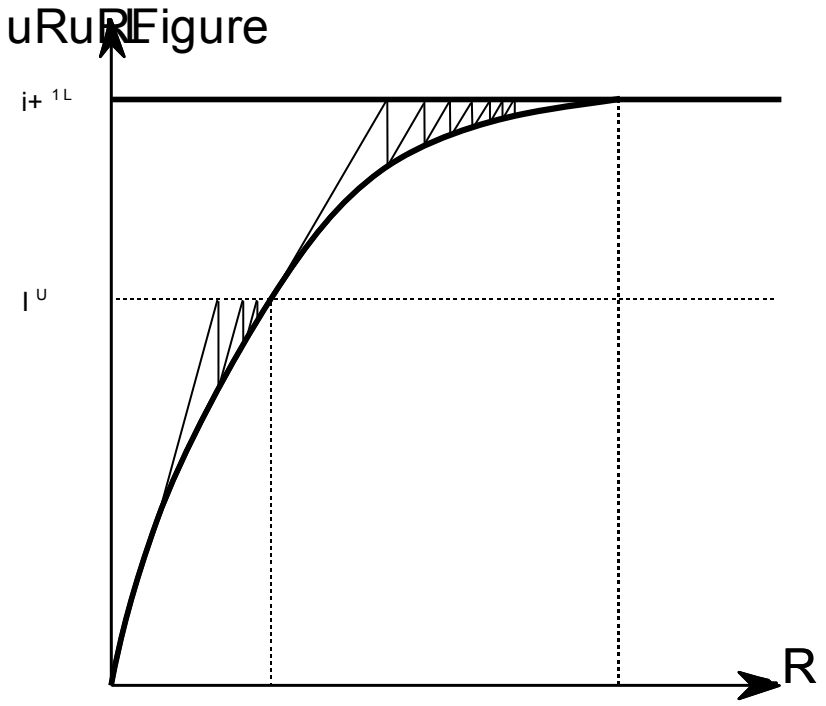
The figures hereafter illustrate the various possibilities of reactualization of the tangent matrix: stamp
•elastic used $\mathbf{K}_{\text{élas}}$ with each iteration Figure 2 - 2.4.4.12.4.4.1-a tangent
•matrix reactualized with $\mathbf{K}_i^{n-1}$ each iteration and each time step, reactualized
•tangent matrix all $\mathbf{K}_j^{n-1}$ time step i_0 () $i_0=1$, and
•stamps tangent reactualized all $\mathbf{K}_i^m$ the iterations n_0 of Newton (here $n_0=2$) Figure 2 - 2.4.4.12.4.4.1-d



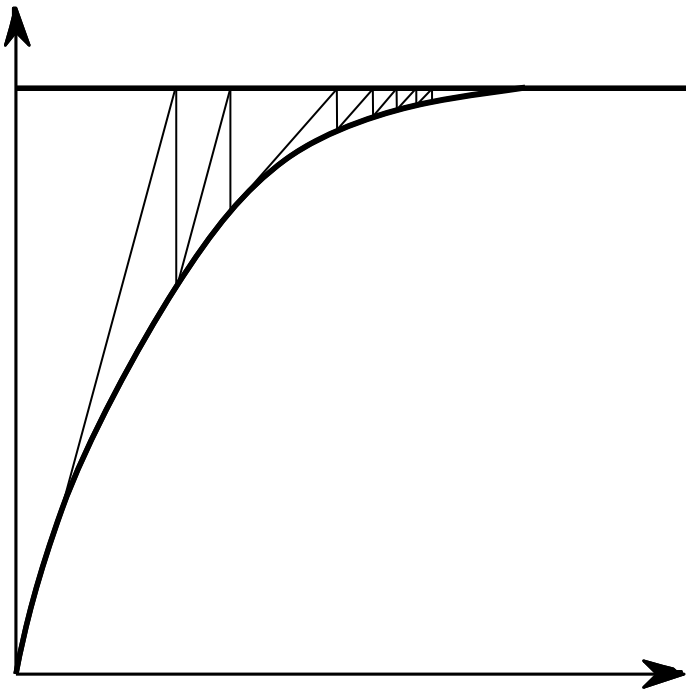
2 - 2.4.4.12.4.4.1-a of the elastic matrix U



2 - 2.4.4.12.4.4.1-b of the true tangent matrix revalued with each iteration U



2 - 2.4.4.12.4.4.1-c of the tangent matrix revalued with each time step Figure



2 - 2.4.4.12.4.4.1-d of the revalued tangent matrix all the 2 iterations of Newton

the method of Newton modified (using another matrix that the consistent tangent matrix) converges less quickly than the method of Newton classical, but is less expensive. It is all the more economic as the number of degrees of freedom of the system is high. This is why one can advise, when the system is of big size, to keep the same tangent matrix during some iterations. Lastly, let us not forget to announce that in certain cases, it is computation with the elastic matrix which is fastest in terms of computing time, even if the nombre of iterations carried out is much more important (they are cheap iterations since the matrix is calculated and factorized only once); moreover, it is the elastic matrix which it is recommended to use during the discharges. It

is necessary to carry out with each iteration of Newton the possible computation of the new tangent matrix and $\mathbf{K}_i^{n-1}$ the "nodal forces". $\mathbf{L}_i^{\text{int},n-1}+\mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}$ For the developers, let us specify that these operations are carried out by computation option FULL _MECA or RAPH _MECA if the tangent matrix is not recomputed. Use

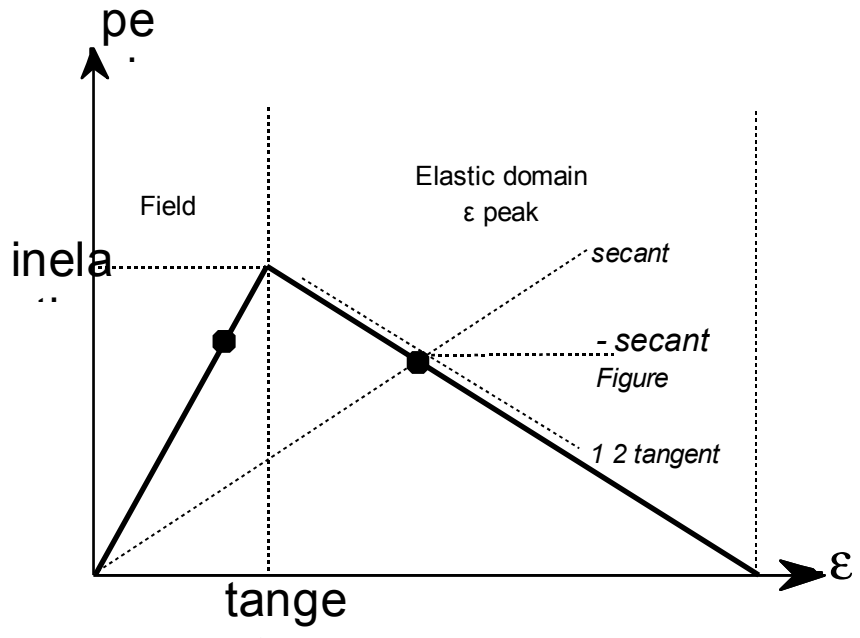
2.4.4.2 of an evolutionary matrix TANGENTE-secant

the method described in this paragraph applies exclusively to the strongly nonlinear problems, where a method of Newton classical fails for any type of choice of matrix, for the phase of prediction or correction. Typically, the usual method of Newton is put at fault for the problems badly-posed coming from the use of the lenitive constitutive laws (see for example R5.04.02). In

these situations, a NON-convergence appears when the algorithm does not manage to choose between several acceptable solutions, in an increment of (pseudonym) - time given. This default of convergence at the total level is generally translated at the local level (i.e at the point of integration) by an alternation repeated between a non-dissipative state (elastic) and a dissipative state (plasticity, damage,…) during consecutive iterations of Newton. One of

the strategies consists in using the notion of control (see R5.03.80) to mitigate the insufficiencies of Newton. The other strategy consists in modifying the tangent matrix. It is this last strategy which one details here. While following

the state of each point of integration of one iteration to the other, one can locate the points responsible for a total NON-convergence. Once these located points, one decides to modify the matrix by hoping that this modification will allow a total convergence. σ σ



2 - 2.4.4.2 2.4.4.2-a constitutive law, use of a mixed matrix TANGENTE-secant the baptized

approach TANGENTE-secant, activated under keyword COMP_INCR with TANGENTE _SECANTE=' OUI', is justified by the following reasoning: if the method of Newton using the coherent tangent matrix does not converge, it is often because a certain number (variable) of points of integration changes state (non-dissipative/ dissipative) between two total iterations of Newton. At the local level (see Figure 2 - 2.4.4.2 2.4.4.2-a wants to say that one continues to alternate between field 1 () and $\varepsilon < \varepsilon_{pic}$ field 2 (). Because of $\varepsilon > \varepsilon_{pic}$ the break of slope between 1 and 2, the use of a secant or tangent matrix coherent leads to a very poor approximation, from where interest to modify it. The choice which one present consists in building the tangent matrix like a linear combination between the coherent tangent matrix and the secant matrix, both being determined by the constitutive laws. Currently, the approach is available only for lenitive constitutive law ENDO_ISOT_BETON .

To manage the starting of the option TANGENTE-secant, one introduces an additional local variable compared to the existing local variables, (with $\boldsymbol{\alpha} = (\alpha_1, \ldots \alpha_n, \alpha_{n+1})$ the number n of local variables of the model used). This variable represents possible alternation between the elastic state and the lenitive state of a Gauss point. One time step initializes it with the first iteration of Newton each again, then one makes it evolve to know, in each Gauss point, the number of successive alternations between the two states. By having this information, one can estimate that from a certain threshold (for example alternations $S_0 = 3$), the algorithm of Newton will not be able to converge any more for the increment of current time and that it is necessary to modify the tangent matrix. To modify the matrix, one is based directly on the way in which the coherent tangent matrix in ENDO_ISOT_BETON is built (see [R7.01.04]). It is a question of making the sum between the dissipative part and the nondissipative part. (72)

$$\mathbf{K}_{\text{tg}} = \mathbf{K}_{\text{sc}} + \mathbf{K}_{\text{end}} \quad \text{where72}$$

is $\mathbf{K}_{\text{tg}}$ the tangent matrix, the secant $\mathbf{K}_{\text{sc}}$ matrix (left non-dissipative) and the damaging $\mathbf{K}_{\text{end}}$ correction (left dissipative). For the modified matrix, one $\mathbf{K}_{\text{t-s}}$ replaces the statement in $\mathbf{K}_{\text{tg}}$ (72) by72 (73)

$$\mathbf{K}_{\text{t-s}} = \mathbf{K}_{\text{sc}} + \eta . \mathbf{K}_{\text{end}} \quad \text{where73}$$

is η a function of with α_{n+1} values between 0 and 1. The function retained η in the continuation is the following one: (74)

$$\eta = \frac{1}{A^{(\alpha_{n+1} - S_0)}} \quad \text{where74}$$

is A a constant and the threshold S_0 on the value amongst successive alternations undergone from which the tangent matrix is modified. For the matrix $\alpha_{n+1} = S_0$ remains tangent coherent and for, it $\alpha_{n+1} \gg S_0$ becomes secant. Values considered to be satisfactory for these parameters are and ($A = 1{,}5$ values $S_0 = 3$ by default). The choice on the evolution of the value of is paramount α_{n+1} for the performance of the algorithm. One chooses an increase in, α_{n+1} for $G = 1{,}0$ $\alpha_{n+1} \rightarrow \alpha_{n+1} + G$ each new alternation between an elastic state and a damaging state, then a reduction in α_{n+1} , P when $\alpha_{n+1} \rightarrow \alpha_{n+1} - P$ the state remains damaging twice continuation. The purpose of the use of is P to allow the return to the coherent tangent matrix when a Gauss point remains damaging on several iterations, since the coherent tangent matrix makes convergence quadratic, provided that one is close to the solution. The value of the rate of reduction compared to P the rate of increase, is G crucial for the behavior of L `evolutionary algorithm. The total idea consists in increasing, lorsqu α_{n+1} "one is far from the solution to have an operator closer to secant than of tangent coherent, then once "near" to the solution, to rock out of coherent tangent matrix (which is the best within the meaning of Newton). The ratio (keyword P/G TAUX_RETOUR - 0.05 per default) represents the third parameter of L" algorithm, besides (keyword A AMPLITUDE) and (keyword S_0 SEUIL). Method

2.4.4.3 of Newton-Krylov general

Principle the method
of Newton-Krylov is part of the inaccurate methods of Newton. It is only usable when the solver of the linear system (71) is71 iterative (in opposition to a direct solver). This approach does not modify the choice of the tangent matrix of the system like the preceding methods. She exploits the convergence criterion of the iterative solver used for the linearized system. By as well as possible adapting the convergence criterion of the iterative method to convergence of Newton, one avoids making useless iterations (in the linear solver) and one gains thus in computing times. Put

in work the general
principle of the inaccurate methods of Newton is to replace the condition which the increment of solution is $(\delta \mathbf{u}_i^n, \delta \boldsymbol{\lambda}_i^n)$ the exact solution of the system (71) by71 a weaker condition. It is asked that $(\delta \mathbf{u}_i^n, \delta \boldsymbol{\lambda}_i^n)$ the condition check: (75)

$$\left\| \begin{matrix} \left(\mathbf{K}_i^{n-1}.\delta \mathbf{u}_i^n + \mathbf{B}^T.\delta \boldsymbol{\lambda}_i^n\right) - \left(\mathbf{L}_i^{\text{méca}} - \mathbf{L}_i^{\text{int},n-1} - \mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}\right) \\ \mathbf{B}.\delta \mathbf{u}_i^n - \left(\mathbf{u}_i^d - \mathbf{B}.\mathbf{u}_i^{n-1}\right) \end{matrix} \right\| \leqslant \eta_n \left\| \begin{matrix} \mathbf{L}_i^{\text{méca}} - \mathbf{L}_i^{\text{int},n-1} - \mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1} \\ \mathbf{u}_i^d - \mathbf{B}.\mathbf{u}_i^{n-1} \end{matrix} \right\| \quad \text{where75}$$

η_n the sustained pressure term *is called. One can*

show that the method suggested is convergent and that when the continuation tends η_n towards, 0 convergence is super-linear (see [8] p.9631). The smaller this value is, the more the solution will be close to that obtained by an exact resolution, but less one will save time with the resolution of the linear system. It is thus necessary to find a good compromise between solving the linear systems quickly and not too much not to degrade the convergence of the iterations of Newton. By examining
the condition (75), onewhere75 notes that it is identical to the relative convergence criterion of the iterative solvers used to solve the linearized system of Newton. To check this condition, it is thus a question of using the sustained pressure *term like* convergence criterion of an iterative linear solver. As

one saw previously, it as should be made sure as the continuation tends η_n towards 0 preserving the super-linear convergence of the method of Newton. With this intention, one will control to η_n the decrease of the residue of Newton by the law of evolution (cf [8*] p* .10531): (76)

$$\eta_{n+1}^{Res} = \gamma \frac{\left\|\mathbf{R}_i^n\right\|^2}{\left\|\mathbf{R}_i^{n-1}\right\|^2} \quad \text{where76}$$

the constant is selected such as. This $\gamma = 0.1$

simple formula is not sufficient in practice because it is necessary to guarantee an adequate decrease of. For η_n that, one determines completely by η_n the following statement: (77)

$$\eta_{n+1} = \begin{cases} \eta_0 & n = -1 \\ max\left(min\left(0.4\ \eta_n,\ \eta_{n+1}^{Res}\right),\ \eta_{min}\right) & n \geq 0,\quad (1-\gamma)\ \eta_n^2 \leq 0.2 \\ min\left(0.4\ \eta_n, max\left(\eta_{n+1}^{Res}, (1-\gamma)\ \eta_n^2\right)\right) & n \geq 0,\quad (1-\gamma)\ \eta_n^2 > 0.2 \end{cases} \quad \text{where77}$$

the constant is worth and corresponds $\eta_0 = 0.9$ to the convergence criterion used for the first linear resolution. , as for
η_{min} it, is the value of the convergence criterion of the iterative linear solver provided by the user (key word RESI_RELA) . Convergence criteria

2.5 At the end of

the prediction and each iteration of Newton, one must estimate if the iterative process converged (the equilibrium of structure is reached). One places oneself at time step running and t_i the iteration of Newton (given that n the value corresponds $n = 0$ to the prediction). There exist four convergence criteria: Criterion

RESI_GLOB_MAXI consists in checking that the norm infinie3La infinite[3] is lower than the value specified γ by the user. (78)

$$\left\|\mathbf{Q}^T.\boldsymbol{\sigma}_i^n + \mathbf{B}^T.\boldsymbol{\lambda}_i^n - \mathbf{L}_i^{\text{méca}}\right\|_\infty \leq \gamma \quad \text{It78}$$

is not advised to use this criterion alone, because one cannot easily have an idea of the acceptable orders of magnitude absolute. Criterion

3 norm corresponds simply to the maximum component of the absolute value of the vector of the residue

RESI_GLOB_RELA (chosen by default) amounts checking that the residue is sufficiently small, like previously, and this compared to a quantity representative of the loading. (79)

$$\frac{\|\mathbf{Q}^T.\boldsymbol{\sigma}_i^n+\mathbf{B}^T.\boldsymbol{\lambda}_i^n-\mathbf{L}_i^{\text{méca}}\|_\infty}{\|\mathbf{L}_i^{\text{méca}}+\mathbf{L}_i^{\text{varc}}-\mathbf{B}^T.\boldsymbol{\lambda}_i^n\|_\infty}\leq\eta$$ being79

η desired relative accuracy given by the user under keyword RESI_GLOB_RELA (or the default value of). One 10^{-6} can

notice that, in the case of use of RESI_GLOB_RELA , the criterion can become singular if the external loading becomes $\mathbf{L}_i^{\text{méca}}+\mathbf{L}_i^{\text{varc}}-\mathbf{B}^T.\boldsymbol{\lambda}_i^n$ null. This can arrive in the event of total discharge of structure. If such a case arises (i.e loading time 10^6 smaller than the smallest loading observed until this increment), the code uses then criterion RESI_GLOB_MAXI with as value that observed with the convergence of the preceding increment. When the loading becomes again non-zero, one returns to the initial criterion. The third

criterion is criterion RESI_REFE_RELA : the idea of this criterion is to build a nodal force of reference, which will be used to estimate term in the long term, the nullity (approximate) of the residue: (80)

$$\forall\, j\in\{\text{ddl}\}\qquad\left|\left(\mathbf{Q}^T.\boldsymbol{\sigma}_i^n+\mathbf{B}^T.\boldsymbol{\lambda}_i^n-\mathbf{L}_i^{\text{méca}}\right)_j\right|\leq\varepsilon\ .\mathbf{F}_j^{\text{ref}}$$ 80

, the nodal force of reference is built $\mathbf{F}_j^{\text{ref}}$ from the data of an amplitude of reference which can A^{ref} be: A stress
•; A pressure
•, a temperature in the case of the THM; A generalized force
•in the case of beams or shells; others…
•the list
is accessible in documentation from use of the command STAT_NON_LINE [U4.51.03], description of operand RESI_REFE_RELA . If one takes

as example a stress, the amplitude of reference being given σ^{ref} by the user via keyword SIGM_REFE . From this amplitude of stress of reference, one defines the tensor: it is $\boldsymbol{\sigma}^{\text{test}}$ null for all these components, except j-ième *which* is worth. One defines σ^{ref} then, for each node of each element the nodal force (the goal $\tilde{R}_i^e$ being giving an idea of the importance of a component in a Gauss point of the stress on the nodal force): (81) With

$$\tilde{R}_i^e=\frac{1}{N}.\sum_{\alpha=1}^{N}\sum_{j=1}^{M}|B_{i,j}^{\alpha}.\sigma_j^{\text{test}}|.\omega^{\alpha}$$ 81

the number N of Gauss points of the element, the number of components M of the tensor of the stresses; the exhibitor being used α to note the evaluating of quantity at the Gauss point. For example are ω^{α} the weights of Gauss points. Note:

For certain elements, as the bars, the grids or the membranes, this definition leads to residues of null reference on certain axes. To cure it, one determines the nodal forces of reference via a computation of order of magnitude based on the size of the element. The nodal

force of reference is then defined by: (82) where

$$\mathbf{F}_i^{\text{ref}}=\min_{e\in\Gamma_i}\tilde{R}_i^e$$ 82

Γ_i all the elements connected to the node. The fourth i

criterion is criterion RESI_COMP_RELA : the idea of this criterion is to separate the various component contributions of the residue by components (with the meaning DX, DY, DZ, PRE1 , PRE2 , TEMP). Each vector obtained will be then normalized by the internal force corresponding to this residue. This choice of

convergence criterion has meaning only for the strongly evolutionary problems, typically problems THM. Who more is, this choice will be effective since one will have problems with strong contrasts. In fact indeed the zones with "strong gradient" will control convergence. One définitla left $\mathbf{F}^n(c)$ the residue corresponding $\mathbf{Q}^T.\boldsymbol{\sigma}_i^n+\mathbf{B}^T.\boldsymbol{\lambda}_i^n-\mathbf{L}_i^{\text{méca}}$ to the component and the vector c $\mathbf{L}^{\text{int},n}(c)$ the internal forces at time corresponding n to this same component. Criterion c RESI_COMP_RELA then amounts checking that this residue is sufficiently small, i.e.: (83)

$$\max_{c=1,\ldots,nbcmp}\left(\frac{\max_{d=1,\ldots,nbddl}|\mathbf{F}^n(c,d)|}{\max_{d=1,\ldots,nbddl}|\mathbf{L}^{\text{int},n}(c,d)|}\right)<\varepsilon \qquad \text{convergence83}$$

is issued carried out when all the criteria specified by the user are checked simultaneously. By default, one makes a test on the relative total residue (RESI_GLOB_RELA) and the maximum number of iterations of Newton (ITER_GLOB_MAXI). Choices of the components

2.5.1 for the convergence criteria For

residues RESI_GLOB_RELA and RESI_GLOB_MAXI, all the components of the field of displacement are used in the evaluating of the norm formulates $\|.\|_\infty$ two cases where a particular processing is made on the level of the choice of the components: For

- the loadings of the type AFFE_CHAR_CINE, the degree of freedom concerned is ignored in the evaluating of the norm of the residue because the procedure of elimination of the unknowns does not make it possible to reach the reactions of bearings; For
- the continuous contact, components LAGR_C and LAGR_F1/ LAGR_F2 are ignored in the evaluating of the norm because the model of Signorini-Coulomb is already checked in the algorithm (see [R5.03.52]) and that these terms are dimensionally incoherent with those relating to displacements; On the other hand, for the case of the contact in XFEM, these components are preserved because they are used to check condition LBB; Difference

2.6 of the matrixes in prediction and correction It is important

to stress that the tangent matrix resulting from option RIGI_MECA_TANG (phase of prediction) and the tangent matrix resulting from option FULL_MECA (phase of correction) are in general not identical. Let us suppose

that one reached convergence for time and that L t_{i-1} "one now seeks to obtain L" balances for next time. The matrix t_i resulting from RIGI_MECA_TANG comes from a linearization from the balance equations compared to time around i.e around $(\mathbf{u}_{i-1},\boldsymbol{\lambda}_{i-1})$ *L* "balances with L" urgent. It is thus t_{i-1} the tangent matrix of the system converged at time . On the other hand t_{i-1},
the matrix resulting from FULL_MECA comes from a linearization from the balance equations compared to displacement ***around*** i.e around $(\mathbf{u}_i^n,\boldsymbol{\lambda}_i^n)$ the equilibrium at time. One can interpret t_i

the differences between RIGI_MECA_TANG and FULL_MECA in D "other terms. One can thus show that the matrix resulting from RIGI_MECA_TANG corresponds to the tangent operator of the continuous problem in time **, known as** also problem of velocity (and connects the velocity of stress at the strainrate), whereas the matrix resulting from FULL_MECA corresponds to the tangent operator of the problem discretized in **time. The document** [R5.03.02] gives the statement in each of the two cases for the relation of elastoplasticity of Von Mises to isotropic or kinematical hardening linear. It is pointed out that

the processing of a behavior model [R5.03.02 § 5] consists with: And to calculate

- the stresses local variables $\boldsymbol{\sigma}_i^n$ from $\boldsymbol{\alpha}_i^n$ the initial state and of the increment $(\boldsymbol{\sigma}_{i-1},\boldsymbol{\alpha}_{i-1})$ of strain induced by the displacement increment $\boldsymbol{\varepsilon}(\Delta\mathbf{u}_i^{n-1})$ since the beginning of the iterative process (including the phase of prediction). To calculate the internal forces
- . To calculate (possibly $\mathbf{L}_i^{\text{int},n}=\mathbf{Q}_i^T.\boldsymbol{\sigma}_i^n$

•) the tangent matrix (option RIGI_MECA_TANG for the phase of prediction, option FULL_MECA for the iterations of Newton). Case of the following

2.7 loadings a following

loading (in mechanics) is a loading which depends on the geometry of structure, as for example the pressure which is exerted in the direction opposed to the norm (or inertia forces in a nonGalilean reference). Thus, when the structure becomes deformed with the evolution of the load, the loading, expressed in an absolute coordinate system, is transformed. The loads which do not depend on the geometry of structure are called dead or fixed loads (for example, gravity). To indicate that a load must be treated like a following load in STAT_NON_LINE, one indicates TYPE_CHARGE=' SUIV " under key word EXCIT. A mechanical loading comprising of $\mathbf{L}_i^{\text{méca}}(\mathbf{u}_i)$ the following loads thus breaks up into two parts: (84) the exhibitor

$$\mathbf{L}_i^{\text{méca}}(\mathbf{u}_i)=\mathbf{L}_i^{\text{fixe}}+\mathbf{L}_i^{\text{suiv}}(\mathbf{u}_i) \qquad 84$$

the died $^{\text{fixe}}$ loads here, and the following loads $^{\text{suiv}}$. The system of equations to be solved (21) becomes then21: (85) the derivative

$$\begin{cases}\mathbf{L}_i^{\text{int}}+\mathbf{B}^T.\boldsymbol{\lambda}_i=\mathbf{L}_i^{\text{fixe}}+\mathbf{L}_i^{\text{suiv}}(\mathbf{u}_i)\\ \mathbf{B}.\mathbf{u}_i=\mathbf{u}_i^d\end{cases} \qquad 85$$

operations making it possible to write the phase of prediction and the iterations of the method of Newton thus utilize the derivatives from ratio with $\mathbf{L}_i^{\text{suiv}}(\mathbf{u}_i)$ displacements. The phase of prediction $(\mathbf{u}_i)$ becomes: (86) And the iterations

$$\begin{cases}\left(\mathbf{K}_{i-1}-\left.\frac{\partial\mathbf{L}^{\text{suiv}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_{i-1}}\right).\Delta\mathbf{u}_i^0+\mathbf{B}^T.\Delta\boldsymbol{\lambda}_i^0=\Delta\mathbf{L}_i^{\text{fixe}}+\mathbf{L}_{i-1}^{\text{suiv}}+\Delta\mathbf{L}_i^{\text{varc}}\\ \mathbf{B}.\Delta\mathbf{u}_i^0=\Delta\mathbf{u}_i^d\end{cases} \qquad 86$$

of Newton consist in solving the system: (87) Thus, at

$$\begin{cases}\left(\mathbf{K}_i^{n-1}-\left.\frac{\partial\mathbf{L}^{\text{suiv}}}{\partial\mathbf{u}}\right|_{\mathbf{u}_i^{n-1}}\right).\delta\mathbf{u}_i^n+\mathbf{B}^T.\delta\boldsymbol{\lambda}_i^n=\mathbf{L}^{\text{fixe}}+\mathbf{L}_i^{\text{suiv},n-1}(\mathbf{u}_i)-\mathbf{L}_i^{\text{int},n-1}-\mathbf{B}^T.\boldsymbol{\lambda}_i^{n-1}\\ \mathbf{B}.\delta\mathbf{u}_i^n=\mathbf{u}_i^d-\mathbf{B}.\mathbf{u}_i^{n-1}\end{cases} \qquad \text{the beginning87}$$

of each step of load (prediction) and with each iteration of Newton, one must calculate a stiffness matrix and a vector $\left.\frac{\partial\mathbf{L}^{\text{suiv}}}{\partial\mathbf{u}}\right|_{\mathbf{u}}$ related to the following $\mathbf{L}^{\text{suiv}}$ loadings. The only loads

which can be dealt as with the following loads in the actual position of operator STAT_NON_LINE are: Pressure for

•the modelizations 3D, 3D_SI, D_PLAN , D_PLAN_SI , AXIS , AXIS_SI , C_PLAN , C_PLAN_SI [R3.03 .04] and COQUE_3D [R3.03 .07]; The loading

•of gravity for elements CABLE_POULIE [R3.08.05], elements with three nodes comprising a pulley and two bits of cables: the force of gravity being exerted on the element depends on the respective lengths of the two bits; The centrifugal force

•in large displacements, which for a rotational speed is given by $\boldsymbol{\omega}$: . Available for $\int_\Omega\rho.\boldsymbol{\omega}\wedge[\boldsymbol{\omega}\wedge\mathbf{OM}].d\Omega=\int_\Omega\rho.\boldsymbol{\omega}\wedge[\boldsymbol{\omega}\wedge(\mathbf{OM_0}+\mathbf{u})].d\Omega$ the modelizations 3D and AXIS_FOURIER ; The loading

•of gravity for all modelizations THM of the unsaturated porous environments [R7.01.10]: indeed, the density depends on the nodal variables to take account of the behavior models of the géomatériaux one. Linear search

3 the linear search

here exposed relates to the linear search in L" absence of control. Principle the introduction

3.1

of the linear search into operator STAT_NON_LINE `results from` a report: the method of Newton with consistent matrix does not converge in all the cases, in particular when one leaves too much far from the solution. In addition, the use of matrixes other than the consistent tangent matrix can, when they are too "flexible", lead to divergence. The linear search makes it possible to secure such divergences against. It consists in

considering, either like $(\delta \mathbf{u}_i^n, \delta \boldsymbol{\lambda}_i^n)$ the increment of displacements and the Lagrange multipliers, but as a direction of search in which one will seek to minimize a functional calculus (the energy of structure). One will find a step of advance in this direction ρ , and the actualization of the unknowns will consist in making: (88) In the absence of

$$\begin{cases} \mathbf{u}_i^n = \mathbf{u}_i^{n-1} + \rho.\ \delta \mathbf{u}_i^n \\ \boldsymbol{\lambda}_i^n = \boldsymbol{\lambda}_i^{n-1} + \rho.\ \delta \boldsymbol{\lambda}_i^n \end{cases} \quad 88$$

linear search (by default) the scalar is of course ρ equal to 1. Minimization of

3.2 a functional calculus In order to

be better convinced of the founded good of the linear search, one can interpret the method of Newton as a method of minimization of a functional calculus (if the tangent matrixes are symmetric). We insist on the fact that the equations obtained are rigorously those of the method of Newton exposed in [the §2.2] and that only2.2 the way of reaching that point is different. "

The talk the dualisation of the boundary conditions of Dirichlet and we place Forget" to simplify on the assumption of the small strains. The functional calculus is considered: (89) where the density

$$J : V \to \mathbb{R}$$
$$\mathbf{u} \to J(\mathbf{u}) = \int_\Omega \Phi(\boldsymbol{\varepsilon}(\mathbf{u})).d\Omega - \int_\Omega \mathbf{f}.\mathbf{u}.d\Omega - \int_\Gamma \mathbf{t}.\mathbf{u}.d\Gamma \quad 89$$

of free energy makes it possible in the case of Φ to connect the tensor of the stresses to the tensor $\boldsymbol{\sigma}$ of the strains linearized by $\boldsymbol{\varepsilon}$ the relation $\boldsymbol{\varepsilon} = \frac{\partial \Phi}{\partial \boldsymbol{\sigma}}$ L" hyperelasticity (one generalizes this situation with the others nonlinearities in the continuation of the document). The functional calculus being convex J , to find the minimum of is equivalent J to cancel its gradient, that is to say: (90) What is

$$\nabla J(\mathbf{u}).\mathbf{v} = 0 \quad \forall \mathbf{v} \in V \quad 90$$

the Principle of the Virtual wors since: (91) Thus, to solve

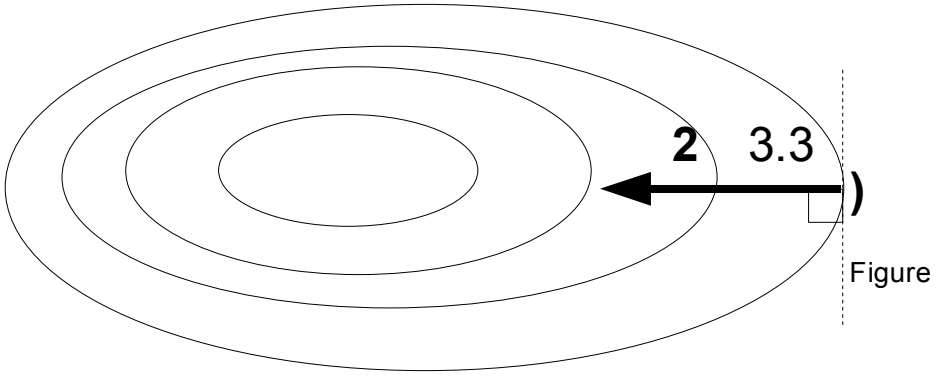
$$\nabla J(\mathbf{u}).\mathbf{v} = \int_\Omega \boldsymbol{\sigma}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}).d\Omega - \int_\Omega \mathbf{f}.\mathbf{v}.d\Omega - \int_\Gamma \mathbf{t}.\mathbf{v}.d\Gamma \quad 91$$

the equations resulting from the Principle of the Virtual wors (bases problem formulated in [§1.3]) is equivalent1.3 to minimize the functional calculus which represents J the energy of the structure (decreased internal energy of the work of the external forces and). Method $\mathbf{f}$ of $\mathbf{t}$ minimization

3.3 minimization

is made in an iterative way, classically in two times with each iteration: Computation of a direction
•of search along which δ one will seek reiterated according to, Computation of the best

•step of advance in this direction ρ : In a problem $\mathbf{u}^{n+1}=\mathbf{u}^{n}+\rho.\boldsymbol{\delta}$
of minimization, the natural idea is to advance in the direction opposed to the gradient of the functional calculus, which is locally the best direction of descent since this direction is normal with the lines of isovaleurs and directed in the meaning of the decreasing values 2-3.3-a U - has However 3.3-a



- has However 3.3-a

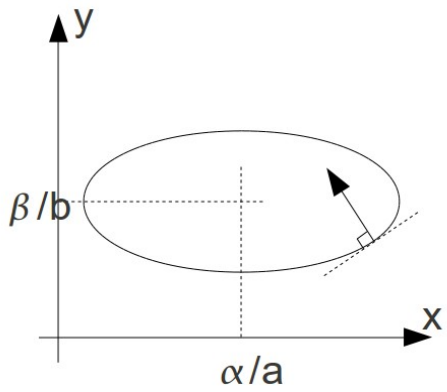
is possible by means of to improve the choice of the direction of descent this method of gradient in a metric news. It is what will enable us to find the classical equations of the method of Newton. Let us take the simple example of a problem with two variables and for which x y the functional calculus has the shape of an ellipse whose minimum is in: (92) While choosing $\left(\frac{\alpha}{a},\frac{\beta}{b}\right)$

$$J(x,y)=\frac{1}{2}.ax^{2}+\frac{1}{2}.by^{2}-\alpha x-\beta y \qquad 92$$

like direction of descent the reverse of the gradient of, one passes from one J reiterated to the following (let us reason on only) by x : (93) which does not point

$$x^{n+1}=x^{n}-ax^{n}+\alpha \qquad 93$$

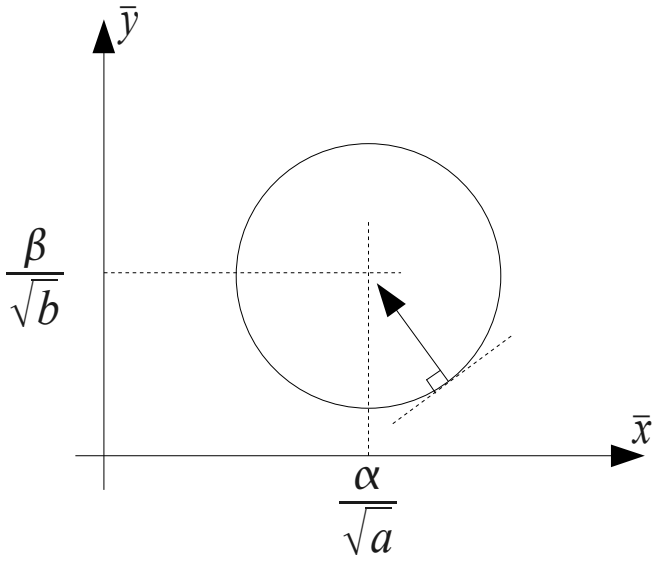
towards the solution since the norm in a point of an ellipse does not pass in general by the center of the ellipse Figure 2 -



3.3-b On the other hand 3.3-b

a change of variables is carried out so that the isovaleurs of become J circles: (94) Figure 2

$$\begin{cases} \bar{x}=\sqrt{a}.x \\ \bar{y}=\sqrt{b}.y \\ \bar{J}(\bar{x},\bar{y})=\frac{1}{2}.(\bar{x}^2+\bar{y}^2)-\frac{\alpha}{\sqrt{a}}.\bar{x}-\frac{\beta}{\sqrt{b}}.\bar{y} \end{cases}$$ -94



- C the use3.3-c

of the opposite direction of the gradient of allows D then $\bar{J}$ "to obtain the solution in an iteration: (95) Thus,

$$\bar{x}^{n+1}=\bar{x}^n-\left(\bar{x}^n-\frac{\alpha}{\sqrt{a}}\right)=\frac{\alpha}{\sqrt{a}}\Rightarrow x^{n+1}=\frac{\alpha}{a}$$ the use95

of the method of gradient in the metric news allows an immediate convergence. In a more complicated case (functional calculus convex but different from an ellipse), convergence is not instantaneous but the change of variables makes it possible to reduce the nombre of iterations appreciably necessary. Application to

3.4 the minimization of energy to simplify

, one will be placed in the discretized linear case where the functional calculus is worth: (96) One J notes

$$J(\mathbf{u})=\frac{1}{2}.\mathbf{u}^T.\mathbf{K}.\mathbf{u}-\mathbf{u}^T.\mathbf{L}$$ 96

$\mathbf{K}$ of structure, and the vector of $\mathbf{L}$ the imposed loadings. To minimize, we J *will use* the same method of descent as previously by making a completely similar change of variables as a preliminary. The definite symmetric matrix $\mathbf{K}$ being positive, its eigenvalues are real positive: one can thus define the "square root" of which one will note $\mathbf{K}$ (also symmetric $\sqrt{\mathbf{K}}$). One poses, the minimization $\bar{\mathbf{u}}=\sqrt{\mathbf{K}}.\mathbf{u}$ of is then equivalent J to that of: (97) By means of

$$\bar{J}(\bar{\mathbf{u}})=\frac{1}{2}.\bar{\mathbf{u}}^T.\bar{\mathbf{u}}-\bar{\mathbf{u}}^T.\left(\sqrt{\mathbf{K}}\right)^{-T}.\mathbf{L}$$ 97

a decomposition by the diagonal: (98) With diagonal

$$\mathbf{K}=\mathbf{P}.\mathbf{D}.\mathbf{P}^{-1}$$ 98

, therefore $\mathbf{D}$: (99) What leads

$$\sqrt{\mathbf{K}}=\mathbf{P}.\sqrt{\mathbf{D}}.\mathbf{P}^{-1} \quad 99$$

to: (100) By taking over

$$\sqrt{\mathbf{K}^T}.\sqrt{\mathbf{K}}=\mathbf{P}.\sqrt{\mathbf{D}^T}.\sqrt{\mathbf{D}}.\mathbf{P}^{-1}=\mathbf{K} \quad 100$$

direction of descent the opposite leadership of the gradient of, one obtains $\bar{J}$: (101) That is to say, while

$$\bar{\mathbf{u}}^{n+1}=\bar{\mathbf{u}}^{n}-\left(\bar{\mathbf{u}}^{n}-\sqrt{\mathbf{K}^{-1}}.\mathbf{L}\right) \quad \text{returning101}$$

to the initial variables: (102) Or

$$\mathbf{u}^{n+1}=\mathbf{u}^{n}-\mathbf{K}^{-1}.\left(\mathbf{K}.\mathbf{u}^{n}-\mathbf{L}\right) \quad 102($$

103) One finds

$$\mathbf{K}.\left(\mathbf{u}^{n+1}-\mathbf{u}^{n}\right)=\mathbf{L}-\mathbf{K}.\mathbf{u}^{n} \quad 103$$

the equations of the method of Newton: the matrix is Hessienne $\mathbf{K}$ of the functional calculus (matrix of J derivative second) and the second member is the difference of the loading and the internal forces, also called residue of equilibrium. Thus method of Newton perhaps interpreted like resulting from the minimization of the energy of structure via a method of gradient applied after a change of metric. Determination

3.5 of the step of advance Let us return to the real

problem, that of the resolution of. This problem $\mathbf{L}_i^{\text{int}}\left(\mathbf{u}_i\right)=\mathbf{L}_i^{\text{ext}}$ can be interpreted like the minimization of the following functional calculus: (104) where corresponds

$$J=W\left(\mathbf{u}_i\right)-\mathbf{u}_i^T.\mathbf{L}_i^{\text{ext}} \quad 104$$

to $W\left(\mathbf{u_i}\right)$ the discretization, on the basis of the shape functions, the internal energy of structure: (105) One calculated

$$W=\int_\Omega \Phi\left(\boldsymbol{\varepsilon}\left(\mathbf{u}\right)\right).d\,\Omega \quad 105$$

by the method of Newton a displacement increment which, in the problem $\delta\mathbf{u}_i^n$ of minimization, is interpreted like a direction of search, according to what precedes. One will calculate the step of advance in this direction ρ allowing to minimize the value of the functional calculus: (106) to find

$$\underset{\rho\in\mathbb{R}}{\text{Min}}\left\{W\left(\mathbf{u}_i^{n-1}+\rho.\delta\mathbf{u}_i^n\right)-\mathbf{L}_i^{\text{ext}}\left(\mathbf{u}_i^{n-1}+\rho.\delta\mathbf{u}_i^n\right)\right\} \quad \text{106the minimum of}$$

this function scalar of that one will note ρ , one seeks $f\left(\rho\right)$ the point where its derivative is cancelled (that amounts making orthogonal the final residue and the direction of search): (107) is the projection

$$f'(\rho)=\left[\delta\mathbf{u}_i^n\right]^T.\left[\mathbf{Q}^T.\boldsymbol{\sigma}\left(\mathbf{u}_i^{n-1}+\rho.\delta\mathbf{u}_i^n\right)-\mathbf{L}_i^{\text{ext}}\right]=0 \quad 107$$

$f'(\rho)$ of the residue on the direction of search. With the notations of [§2.2] and in taking into account2.2 the reactions of bearing, the scalar equation to solve determine the step of advance, is written: (ρ 108) At the end of

$$f'(\rho)=\left[\delta\mathbf{u}_i^n\right]^T.\left[\mathbf{Q}^T.\boldsymbol{\sigma}\left(\mathbf{u}_i^{n-1}+\rho.\delta\mathbf{u}_i^n\right)+\mathbf{B}^T.\left(\boldsymbol{\lambda}_i^{n-1}+\rho.\delta\boldsymbol{\lambda}_i^n\right)-\mathbf{L}_i^{\text{ext}}\right]=0 \quad 108$$

the linear search, one brings up to date displacements and parameters of Lagrange by: (109) the test

$$\begin{cases}\mathbf{u}_i^n=\mathbf{u}_i^{n-1}+\rho.\ \delta\mathbf{u}_i^n\\ \lambda_i^n=\lambda_i^{n-1}+\rho.\ \delta\lambda_i^n\end{cases} \quad \text{of109}$$

carries: On the maximum
•number of iterations of linear search indicated by the user under the key - key ITER_LINE_MAXI of factor key word the NEWTON (the default value 0 inhibits the linear search, and is worth 1 then), ρ On criterion
•RESI_LINE_RELA given by, where is worth by $f(\rho) \leq \tau . f(0)$ default τ 0.1. The linear search

is to some extent a “insurance” making it possible to guard itself against serious divergences of the method of Newton. When the direction of search is “bad” (if the tangent matrix is too flexible, for example), the linear algorithm of search ends in a low value of, which avoids ρ moving away too much from the solution. It is not necessary to do many iterations in the method of secant (two or three are enough to avoid the catastrophes) because each one is rather expensive (the internal forces should be recomputed) and there is not the ambition to find with each iteration of Newton the value of really optimal ρ . Computation of the linear

3.6 coefficient of search There exist two

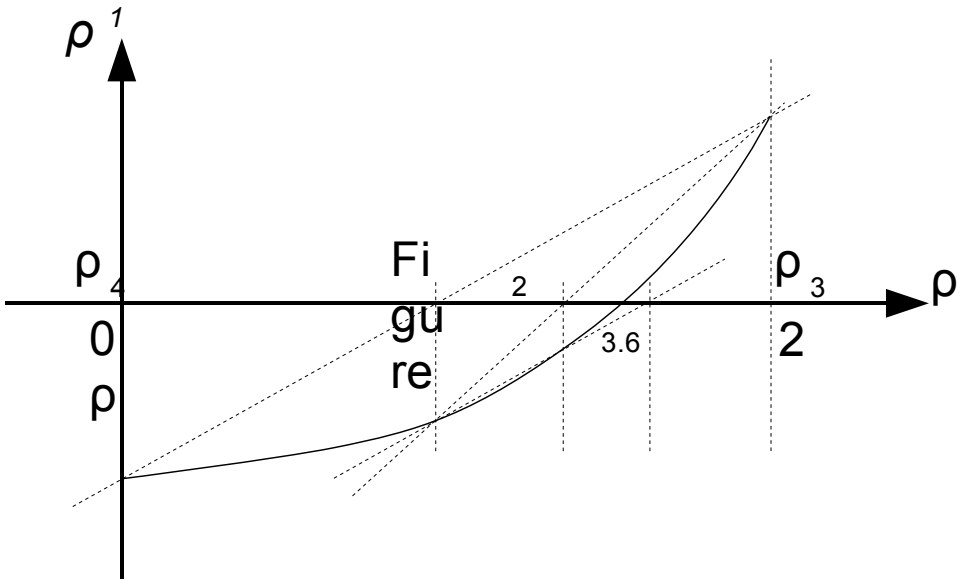
methods to compute: the optimal one in ρ the linear search. Secant method

3.6.1 (METHODE=' CORDE') So that the determination

of is not too ρ expensive, one uses a method of secant of which the maximum number of iterations is fixed by the user. The method of secant can be interpreted as a method of Newton where the derivative at the point running is approached by the direction uniting the point running and the preceding point: (110) Where one

$$\rho^{p+1} = \rho^p - \frac{\rho^p - \rho^{p-1}}{g^p - g^{p-1}} \cdot g^p = \frac{\rho^{p-1} . g^p - \rho^p . g^{p-1}}{g^p - g^{p-1}} \qquad 110$$

. One leaves $g^p = f'(\rho^p)$ and. The method $\rho^0 = 0 \quad \rho^1 = 1$ of secant has an order of convergence of about 1.6. It is represented schematically on Figure 2-3.6.1 .1-a mixed 3.6.1-a



.1-a mixed 3.6.1-a

3.6.2 (METHODE=' MIXTE') This method

mixes several techniques of resolution to be more robust. It consists primarily of the application of a secant method (see paragraph the preceding one) between two predetermined limits. It is by way of the application the method of secant with variable limits. Here the algorithm used: It is supposed that

1.. If it is not $f'(0)>0$ the case, one changes the meaning of the direction of descent (by examining the negative ones, which ρ amounts defining as being equal f' to One seeks $-f'$

2.positive such as $\rho_{\max}$. The method $f'(\rho_{\max})<0$ is simply iterative while making with (stage $\rho_{n+1}=3.\rho_n$ of bracketage $\rho_0=1$ or framing) One thus has

3.the two new limits between which the function changes sign. If it is supposed that the function is continuous, f' there thus exists a solution between these limits. One applies

4.the secant method to this interval: one leaves and. Typical case $\rho^0=0 \quad \rho^1=\rho_{max}$

3.6.3 : the method of Newton-Krylov It was specified

higher than the linear search is carried out simultaneously on the unknowns and as the formula $\mathbf{u} \quad \lambda$ (109) of actualization of the variables109 shows it. However the functional calculus to be minimized does not present minimum according to the unknowns formulates, it $(\mathbf{u}, \lambda)$ about Lagrangian which presents a point saddles, i.e. a minimum in formula and $\mathbf{u}$ in formula (see λ]). This way make is thus not licit in the general case. However, one can
show that if the system in prediction is solved “exactly” (all at least with a numerically satisfactory accuracy), this approach is licit. It is generally the case in the usual use of Code_Aster. It *is on the other hand not*
the case in the frame of the use of the method of Newton-Krylov, where the linear systems are precisely solved in a voluntarily inaccurate way. In this situation, to circumvent the problem, only the unknowns formula are affected $\mathbf{u}$ by the linear search and the formula of update of the variables thus becomes: formulate (111)

$$\begin{cases} \mathbf{u}_i^n = \mathbf{u}_i^{n-1} + \rho.\ \delta\mathbf{u}_i^n \\ \lambda_i^n = \lambda_i^{n-1} + \delta\lambda_i^n \end{cases} \qquad 111$$

the functional calculus to be minimized has well a minimum in formula, procedure $\mathbf{u}$ of linear search is licit. Control One

4 will refer

to documentation [R5.03.80]. Bibliography

5 A. Ibrahimbegovic

1) – “Nonlinear Solid Mechanics: Theoretical Formulations and Finite Element Methods Solution” – Springer – 2009 M.A. Crisfield

2) – “Non-Linear Finite Element Analysis of Solids and Structures: Essentials” – Wiley Professional Software – 1996 M.A. Crisfield

3) – “Non-Linear Finite Element Analysis of Solids and Structures: Advanced Topics” – Wiley Professional Software – 1997 J.C. Simo, T.J.R. Hughes

4) – “Computational inelasticity ” – Springer – 2000 “Elements of analysis

5) and numerical resolution of the relations of elastoplasticity” – EDF – Bulletin of the Management of the Studies and Searches – Series C – N° 3 1986 – p. 57 - 89. J.F. Master – “

6) numerical Analysis” – duplicated course of the ENTPE. J. Shi, Mr. A. Crisfield

7) – “Combining arc-length control and line searches in path following” – Communications in Numerical Methods in Engineering, vol. 11,793-803 (1995). Work Card Kelley - “Iterative

8) Methods for Solving Linear and Nonlinear Equations”, vol. 16 in Frontiers in Applied Mathematics, Siam, Philadelphia, 1995. History of the versions

6 of the document Version Aster Author

(S) or contributor	(S), organization Description of the modifications	5 N. Tardieu, I. Vautier
	, E. Lorentz EDF R & D MN 7.4 P.Badel, J.Laverne	
,	N. Tardieu EDF R & D AMA 8.5 M.Abbas EDF R & D	
Updated	AMA of the notations	, addition of the command variables. 9.2 M.Abbas EDF R & D
Updated	AMA of the §	2.2. 9.4 M.Abbas, D.Markovic
	EDF R & D AMA Addition of the matrix	TANGENTE-secant and the mixed linear search. 10.1 M.Abbas EDF R & D
AMA	Updated of the paragraph	on the large deformations. 11.1 N. Tardieu EDF
R & D	AMA Addition of the method	of Newton-Krylov. 11.3 Mr. David EDF R & D
AMA	Precise details concerning	criterion RESI_REFE_RELA