Taking into account of the assumption of the constraints plane in the nonlinear behaviors

Summary:

This document describes a general method of integration of the nonlinear models of behaviors (elastoplastic, viscoplastic, damaging,…) in plane constraints.

This is carried out by a method of static condensation due to R. of Borst.

This method makes it possible to use modeling C_PLAN, or modelings COQUE_3D, DKT and PIPE for all the models of incrémentaux behaviors of (STAT/DYNA) _NON_LINE available into axisymmetric or plane deformations.

1 Introduction

One presents here a general method of integration of the nonlinear models of behaviors (plasticity, viscoplasticity, damage) in plane constraints. If the selected behavior is not integrated analytically in plane constraints, the method of R. De Borst is activated automatically for modelings C_PLAN, DKT, COQUE3D and PIPE.

2 Difficulty of integration of the nonlinear behaviors in plane constraints

Modeling C_PLAN, (as well as modelings COQUE_3D, DKT, PIPE) suppose that the local state of stresses is plan, i.e. that $\sigma_{zz}=0$, z representing the direction of the normal on the surface. The tensors of constraints and deformations thus take the following form (in C_PLAN):

$$\varepsilon=\begin{pmatrix}\varepsilon_{xx} & \varepsilon_{xy} & 0\\ \varepsilon_{xy} & \varepsilon_{yy} & 0\\ 0 & 0 & \varepsilon_{zz}\end{pmatrix}$$

$$\sigma=\begin{pmatrix}\sigma_{xx} & \sigma_{xy} & 0\\ \sigma_{xy} & \sigma_{yy} & 0\\ 0 & 0 & \sigma_{zz}\end{pmatrix} \qquad \text{eq.2-1}$$

Note:

For the hulls, it is necessary to add terms due to transverse shearing ($\sigma_{xzn}\sigma_{yz}$), but those are treated elastically and do not intervene in the resolution of the local behavior.

This assumption implies that the corresponding deformation is a priori unspecified (contrary to other two-dimensional modelings where one makes an assumption directly on ε_{zz}). It can be given only using the relation of behavior. However the condition $\sigma_{zz}=0$ is not alleviating for the integration of the behavior, where one calculates an increase in constraint $\Delta\sigma$ according to the increase in deformation $\Delta\varepsilon$ provided by the algorithm of Newton. In the case of linear elasticity, the taking into account of this condition is simple and makes it possible to find:

$$\varepsilon_{zz}=-\frac{\nu}{1-\nu}(\varepsilon_{xx}+\varepsilon_{yy})$$

But if the behavior is nonlinear, $\Delta\varepsilon_{zz}$ cannot be only calculated from Δu and does not result simply from the other components from the tensor of the deformations. The taking into account of this assumption must then be made (when it is realizable) in a way specific to each behavior, and very often brings to additional difficulties of resolution: it is the case in particular for the behavior of Von Mises to isotropic work hardening [R5.03.02]. So much of models of behavior are not available in plane constraints.

The method presented here has the large advantage of not requiring any particular development in the integration of the behavior to satisfy the assumption with the plane constraints. It is usable as soon as the model of behavior is available in axisymmetric or in plane deformations.

3 Principle of the treatment of the plane constraints by the method De Borst

3.1 Recall of the method of Newton

In non-linear statics one is brought to solve the following equation ([cf. `R5.03.01`]):

$$R(u)=f^{\text{int}}(u)-f^{ext}(t)\rightarrow 0 \;, \qquad \text{eq. 3.1-1}$$

where f^{ext} is the external force and f^{int} the interior force, definite like,

$$f^{\int(u)}=\int_V B^T\sigma(\varepsilon(u))\,dV . \qquad \text{eq. 3.1-2}$$

By using the finite element method (MEF), the interior forces, f^{int}, are obtained starting from the stress field, which is determined to him, by a law of behavior, starting from the field of deformation, the tensor of deformation discretizes, ε, being defined like:

$$\varepsilon=B u .$$

According to the method Newton one solves the system in [eq. 3.1-1] with the following iterative process:

1) $R^{(n)}=R(u^{(n)})$

2) $\Delta u^{(n+1)}=-\underbrace{\left(\frac{\partial R^{(n)}}{\partial u^{(n)}}\right)}_{K^{(n)^{-1}}}R^{(n)}=-K^{(n)^{-1}}R^{(n)}$

3) $u^{(n+1)}=u^{(n)}+\Delta u^{(n)}$, to repeat 1) until $R^{(n)}<tolérance$.

The index (n) mean that the variable concerned corresponds to n - ième iteration, known as *total,* since it relates to the calculation of the field of displacement, contrary with the process known as *room*, which makes it possible to calculate the component ε_{zz} so that the constraints is plane and which will be detailed in the continuation.

The matrix of rigidity, $K^{(n)}$, is calculated like:

$$K^{(n)}(u)=\int_V B^T\underbrace{\frac{\partial\sigma^{(n)}}{\partial u^{(n)}}}_{D^{(n)}}B\,dV=\int_V B^T D^{(n)}B\,dV ,$$

where $D^{(n)}$ is the tangent matrix corresponding to the law of behavior used. To extend the method of Newton to the use of unspecified laws of behavior, in [bib1] one proposes to condense the value of ε_{zz} so that $\sigma_{zz}<tolérance$ at the converged state. Here two alternatives of this approach are presented: in the original approach, the variables $u^{(k+1)}$ and $\varepsilon_{zz}^{(k+1)}$ are corrected simultaneously for each total iteration, while in the modified approach a local iterative process is added, so that the condition $\sigma_{zz}<tolérance$ that is to say satisfied for each value with $u^{(k+1)}$ and non-pas only for the converged state. It is interesting to use the approach modified in particular when convergence on total balance is faster than convergence for the satisfaction of the plane constraints. In certain cases, but not always, the modified approach can reduce the iteration count total and thus accelerate calculations. In term of use, the original approach corresponds to the value of the parameter `ITER_CPLAN_MAXI` = 1 and approaches it modified with `ITER_CPLAN_MAXI` > 1.

Besides the calculation of ε_{zz} at the local level which results in modifying the local constraints, approach known as of DEBORST especially consist in intervening on the level of the matrix of rigidity,

$$\hat{K}^{(n)}(u)=\int_V B^T \hat{D}^{(n)} B\, dV ,$$

where D was replaced by $\hat{D}$. The calculation of $\hat{D}$ as well as of ε_{zz} are detailed in the continuation.

3.2 Approach of origin

The idea of the method due to R. of Borst [bib1] consists in not treating the condition of plane constraints with the level of the law of behavior but with the level of balance. One obtains thus during iterations of the algorithm of total resolution of STAT_NON_LINE stress fields which tend towards a plane stress field as iterations:

$$\sigma_{zz}^{(n)} \rightarrow 0$$

where n indicate the number of iteration of Newton.
One thus obtains the condition of constraint planes not exactly, but in an approached way, with convergence of the iterations of Newton, for each calculated increment. One checks, as specified thereafter, that the component above is lower than a given tolerance.

The method consists in breaking up the fields (strains or stresses) into a purely plane part (specified by a "hat") and a component according to z. One then reveals explicitly the components " zz " in the form of the tensors of strains and stresses:

$$\varepsilon=\begin{pmatrix}\hat{\varepsilon}\\ \varepsilon_{zz}\end{pmatrix} \quad \text{and} \quad \sigma=\begin{pmatrix}\hat{\sigma}\\ \sigma_{zz}\end{pmatrix}.$$

like in the expression of the tangent operator:

$$D^{(n)}=\frac{\partial \sigma^{(n)}}{\partial \varepsilon^{(n)}}=\begin{pmatrix} D_{11}^{(n)} & D_{12}^{(n)}\\ D_{21}^{(n)} & D_{22}^{(n)}\end{pmatrix} . \qquad \text{eq 3.2-1}$$

Lastly, with each iteration Newton one corrects the values of deformation,

$$\hat{\varepsilon}^{(n+1)}=\hat{\varepsilon}^{(n)}+\Delta\hat{\varepsilon}^{(n)} \quad \text{and} \quad \varepsilon_{zz}^{(n+1)}=\varepsilon_{zz}^{(n)}+\Delta\varepsilon_{zz}^{(n)} .$$

By using [eq. 3.2-1] one can write,

$$\begin{pmatrix}\Delta\hat{\sigma}\\ \Delta\sigma_{zz}\end{pmatrix}=D^{(n)}\begin{pmatrix}\Delta\hat{\varepsilon}\\ \Delta\varepsilon_{zz}\end{pmatrix} \text{, with } \Delta\sigma_{zz}=\sigma_{zz}^{(n+1)}-\sigma_{zz}^{(n)} ,$$

and one obtains:

$$\sigma_{zz}^{(n+1)}\approx\sigma_{zz}^{(n)}+D_{21}^{(n)}\Delta\hat{\varepsilon}^{(n)}+D_{22}^{(n)}\Delta\varepsilon_{zz}^{(n)}\rightarrow 0 , \qquad \text{eq 3.2-2}$$

From [eq. 3.2-2] one can condense $\Delta\varepsilon_{zz}^{(n)}$ like,

$$\Delta\varepsilon_{zz}^{(n)}=-\frac{\sigma_{zz}^{(n)}+D_{21}^{(n)}\Delta\hat{\varepsilon}^{(n)}}{D_{22}^{(n)}} . \qquad \text{eq 3.2-3}$$

It is exactly the condensation in [eq 3.2-3] which enables us to use the framework 2D for the resolution with the MEF. With final, one seeks to correct the components 2D (noted by the index [1]), as well for the constraints as for the tangent operator, so that the condition of plane constraints is satisfied.

The algorithm of plane constraints entirely local, is applied within the framework of a total architecture finite elements corresponding to a modeling 2D equivalent to that of the plane deformation. One positions in a point of Gauss, where one knows the value of the deformation to the iteration (n+1),

$\hat{\varepsilon}^{(n+1)}=\hat{\varepsilon}^{(n)}+\Delta\hat{\varepsilon}^{(n)}$, and one must calculate the values of constraint $\hat{\sigma}^{(n+1)}$ and of the tangent operator, $\hat{D}^{(n+1)}$.

Algorithm 1:

1) To bring up to date $\varepsilon_{zz}^{(n)}$,

$$\varepsilon_{zz}^{(n+1)}=\varepsilon_{zz}^{(n)}-\frac{\sigma_{zz}^{(n)}+D_{21}^{(n)}\Delta\hat{\varepsilon}^{(n)}}{D_{22}^{(n)}}$$

2) To calculate the constraints and the tangent operator intermediaries $\sigma^{(n+1)}$, $D^{(n+1)}$ by the law of behavior 3D, imposing $\varepsilon_{xz}=\varepsilon_{yz}=0$,

$$\sigma^{(n+1)}=\sigma(\hat{\varepsilon}^{(n+1)},\varepsilon_{zz}^{(n+1)}) \quad \text{and} \quad D^{(n+1)}=D(\hat{\varepsilon}^{(n+1)},\varepsilon_{zz}^{(n+1)})$$

3) To calculate the final constraint by using the intermediate correction of $\tilde{\varepsilon}_{zz}^{(n+1)}$,

$$\Delta\tilde{\varepsilon}_{zz}^{(n+1)}=-\frac{\sigma_{zz}^{(n+1)}}{D_{22}^{(n+1)}} \qquad \text{eq 3.2-4}$$

which enables us to write the final constraint $\hat{\sigma}^{(n+1)}$ like:

$$\hat{\sigma}^{(n+1)}=\sigma^{(n+1)}+D_{12}^{(n+1)}\Delta\tilde{\varepsilon}_{zz}^{(n+1)}=\sigma^{(n+1)}-\frac{D_{12}^{(n+1)}\sigma_{zz}^{(n+1)}}{D_{22}^{(n+1)}}$$

4) To calculate the final tangent operator:

$$\hat{D}^{(n+1)}=D_{11}^{(n+1)}-\frac{D_{12}^{(n+1)}D_{21}^{(n+1)}}{D_{22}^{(n+1)}} .$$

Notice 1 *: In 2) the constraint* σ *is calculated as a tensor in 3D,*

$\sigma=(\sigma_{xx}\quad\sigma_{yy}\quad\sigma_{zz}\quad\sigma_{xy}\quad\sigma_{xz}\quad\sigma_{yz})^T$.

On the other hand, one only uses $\sigma=(\sigma_{xx}\quad\sigma_{yy}\quad\sigma_{zz}\quad\sigma_{xy})^T$ *, components* σ_{xz} *and* σ_{yz} *not needing to be extracted. Thus,* D *is of size* 4×4 *.*

Notice 2 *: In 3) the expression of* $\Delta\tilde{\varepsilon}_{zz}^{(n+1)}$ *is obtained by supposing that the value extrapolated of* $\sigma_{zz}^{n+2}\approx0$ *,*

$$\sigma_{zz}^{(n+2)}\approx\sigma_{zz}^{(n+1)}+D_{22}^{(n+1)}\Delta\tilde{\varepsilon}_{zz}^{(n+1)}=0 .$$

Notice 3 *: In 4) one calculates the tangent operator like:*

$$\hat{D}^{(n+1)}=\frac{\delta\sigma^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}}=\frac{\partial\sigma^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}}+\frac{\partial\sigma^{(n+1)}}{\partial\varepsilon_{zz}^{(n+1)}}\frac{\partial\varepsilon_{zz}^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}}=D_{11}^{(n+1)}-\frac{D_{12}^{(n+1)}D_{21}^{(n+1)}}{D_{22}^{(n+1)}} ,$$

where δ *mean the total derivative contrary to* ∂ *who means partial derivative. In general, this tangent operator is not coherent compared to* $\hat{\sigma}^{(n+1)}$ *,*

$$\hat{D}^{(n+1)} \neq \frac{\delta\hat{\sigma}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} \quad .$$

The coherent tangent operator can be obtained in the following way:

$$\frac{\delta\hat{\sigma}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} = \frac{\delta\sigma^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} - \frac{D_{12}^{(n+1)}}{D_{22}^{(n+1)}} \frac{\delta\sigma_{zz}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} \quad ,$$

where

$$\frac{\delta\sigma^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} = \frac{\partial\sigma^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}} + \frac{\partial\delta^{(n+1)}}{\partial\varepsilon_{zz}^{(n+1)}} \frac{\partial\varepsilon_{zz}^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}} = D_{11}^{(n+1)} - \frac{D_{12}^{(n+1)} D_{\mathbf{21}}^{(n)}}{D_{22}^{(n)}}$$

$$\frac{\delta\sigma_{zz}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} = \frac{\partial\sigma_{zz}^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}} + \frac{\partial\sigma_{zz}^{(n+1)}}{\partial\varepsilon_{zz}^{(n+1)}} \frac{\partial\varepsilon_{zz}^{(n+1)}}{\partial\hat{\varepsilon}^{(n+1)}} = D_{\mathbf{21}}^{(n+1)} - \frac{D_{22}^{(n+1)} D_{\mathbf{21}}^{(n)}}{D_{22}^{(n)}} \quad .$$

With final, one obtains:

$$\frac{\delta\hat{\sigma}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} = D_{11}^{(n+1)} - \frac{D_{12}^{(n+1)} D_{\mathbf{21}}^{(n)}}{D_{22}^{(n)}} - \frac{D_{12}^{(n+1)}}{D_{22}^{(n+1)}} \left(D_{\mathbf{21}}^{(n+1)} - \frac{D_{22}^{(n+1)} D_{\mathbf{21}}^{(n)}}{D_{22}^{(n)}} \right) \quad .$$

The operator used, $\hat{D}^{(n+1)}$, tends towards the coherent tangent operator at the time of the convergence of the criterion of the plane constraints, since $D^{(n+1)} \to D^{(n)}$, when $\sigma_{zz}^{(n)} \to 0$, which thus carries out to:

$$\hat{D}^{(n+1)} \overset{\sigma_{zz}^{(n+1)} \to 0}{\longrightarrow} \frac{\delta\hat{\sigma}^{(n+1)}}{\delta\hat{\varepsilon}^{(n+1)}} \quad .$$

3.3 Modified approach

In the modified algorithm one proposes to introduce an additional loop compared to the process describes above for better satisfying the plane constraints for each total iteration with Newton, (n). This new loop includes the points 2) and 3) algorithm presented in 3.2. Thus the new algorithm is written like:

Algorithm 2:

1) To bring up to date $\varepsilon_{zz}^{(n)}$,

$$\varepsilon_{zz}^{(n+1)} = \varepsilon_{zz}^{(n)} - \frac{\sigma_{zz}^{(n)} + D_{\mathbf{21}}^{(n)} \Delta\hat{\varepsilon}^{(n)}}{D_{22}^{(n)}}$$

2) To initialize the loop

$$\tilde{\varepsilon}_{zz}^{(k=0,n+1)} = \varepsilon_{zz}^{(n+1)}$$

Beginning buckles $k=0, K_{\max}$

3) To calculate the constraints and the tangent operator intermediaries $\sigma^{(k,n+1)}$, $D^{(k,n+1)}$ by the law of behavior 3D,

$$\sigma^{k,(n+1)} = \sigma(\hat{\varepsilon}^{(n+1)}, \tilde{\varepsilon}_{zz}^{(k,n+1)}) \quad \text{and} \quad D^{(k,n+1)} = D(\hat{\varepsilon}^{(n+1)}, \tilde{\varepsilon}_{zz}^{(k,n)})$$

4) To calculate the intermediate correction of $\tilde{\varepsilon}_{zz}^{(k,n+1)}$,

$$\Delta\tilde{\varepsilon}_{zz}^{(k,n+1)} = -\frac{\sigma_{zz}^{(k,n+1)}}{D_{22}^{(k,n+1)}} \quad \text{and} \quad \tilde{\varepsilon}_{zz}^{(k+1,n+1)} = \tilde{\varepsilon}_{zz}^{(k,n+1)} + \Delta\tilde{\varepsilon}_{zz}^{(k,n+1)} \quad .$$

<u>To finish the loop</u> if $|\sigma_{zz}^{(k,n+1)}|<\sigma_{tol}$ or if $k=K_{\max}$.

(With the parameter σ_{tol} one defines the tolerance on the value of σ_{zz} .)

5) To allot to the tensor 2D constraints converged values:

$$\hat{\sigma}^{(n+1)}=\sigma^{(k,n+1)}+D_{12}^{(k,n+1)}\Delta\tilde{\varepsilon}_{zz}^{(k,n+1)}=\sigma^{(k,n+1)}-\frac{D_{12}^{(k,n+1)}\sigma_{zz}^{(k,n+1)}}{D_{22}^{(k,n+1)}}$$

In theory, the second term of the equation above can be omitted, if the parameter of the convergence criteria, σ_{tol} , is selected sufficiently small.

6) To calculate the final tangent operator:

$$\hat{D}^{(n+1)}=D_{11}^{(n+1)}-\frac{D_{12}^{(n+1)}D_{21}^{(n+1)}}{D_{22}^{(n+1)}} .$$

***Notice 4** : In the modified version, the tangent operator, $\hat{D}^{(n+1)}$, is coherent compared to $\hat{\sigma}^{(n+1)}$, contrary to the operator of the version of origin (see remark 3), since $|\sigma_{zz}^{(n+1)}|<\sigma_{tol}$.*

4 Practical aspects of use

This method is used automatically as soon as the selected behavior is not available in plane constraints, for modelings `C_PLAN` or of standard hull: `COQUE_3D`, `DKT`, `PIPE`. In practice, that increases (automatically) by 4 the number of internal variables of the behavior.

For converging well, it is advised to reactualize the tangent matrix if possible (, with all the iterations: `REAC_ITER = 1`, or all them n iterations, with n small).

This method thus allows a great flexibility in use compared to the behaviors: it is enough that a behavior is available in axisymetry or plane deformation so that it is also usable in plane constraints.

As for all integrations of models of behaviors nonlinear, it is highly advised to give small convergence criteria (to leave the value by default with 10^{-6} .).

The advantage of the modified approach is a better satisfaction of the condition of plane constraints in each point of Gauss ($|\sigma_{zz}^{\mathrm{mod}if,(n)}|<<|\sigma_{zz}^{orig,(n)}|$ for each n). In certain cases, it is essential to make converge a calculation, in particular for the lenitive laws of behavior.

On the other hand, because of an additional loop the modified procedure is more expensive, especially because the loop includes the call to the module "law of behavior 3D". Nevertheless, the overcost because of heavier local calculations can be compensated by a profit on the level of the iteration count total of Newton, which is generally less low for the modified algorithm. This profit of iteration count total is not guaranteed, which has as a consequence that the additional iterative loop is not activated by default (`ITER_CPLAN_MAXI`=1). It was also observed that moment when one chooses `ITER_CPLAN_MAXI` > 1, it is preferable to use `ITER_CPLAN_MAXI` > 5.

5 Bibliography

1 R de Borst "The zero normal stress condition in plane stress and Shell elastoplasticity" Communications in applied numerical methods, Flight 7, 29-33 (1991)

6 History of the versions of the document

Version Aster	Author (S) or contributor (S), organization	Description of the modifications
5.4	J.M. PROIX, E. LORENTZ	Initial version.
9.2	D. MARKOVIC	Addition of the iterative loop interns to improve convergence.
11.2	J.M.PROIX	card 18398