The Councils of use of STAT_NON_LINE

Résumé

the objective of this note is to give some advices to an user wishing to carry out nonlinear computations with *Code_Aster* by using operator STAT_NON_LINE [U4.51.03].

General advices are initially given. Then one specifies the solutions to be implemented for the main types of encountered problems. The problems thus it is question are of standard not-convergence, or failure of an algorithm. One also returns towards documentation specific to each problem.

The purpose of contents

1 Introduction

This document are to help the user to solve a problem of mechanical evolution or thermo-hydro-mechanics, into quasi-static, of a structure into nonlinear, *via* operator `STAT_NON_LINE`. A detailed presentation of the algorithm of this operator is made in his theoretical documentation [R5.03.01].
The 3 great types of nonlinearities are the following ones:

- non-linearity related to the behavior of the material (for example plastic);
- non-linearity related to the geometry (for example in large displacements);
- non-linearity related to contact-rubbing.

In the 2nd part of this document, one details the main tools which one lays out to help the convergence of computations and to improve quality of the results. The following parts are devoted to the advices of use. In the 3rd part, one gives general advices, commun runs with the 3 types of non-linearities. In 4th, 5th and 6th parts, one gives advices dedicated to each of the 3 types of non-linearities: materials, then geometrical and finally of contact-rubbing. These advices refer to the tools evoked in 2nd part.

2 Tools of aid to convergence

En cas de problems, one has several tools:

- under-spanning of the time step,
- change the matrixes,
- control,
- linear search,
- change finite elements formulation.

2.1 The under-spanning of the time step

In a general way, plus the time step is small, less the problem is nonlinear, therefore easier to solve. The under-spanning of the time step is thus essential tools which make it possible to pass the most current difficulties. The under-spanning activates by default in `DEFI_LIST_INST` [U4.34.03]. One advises always to activate it. Moreover, more the time step is small, and more the error of temporal discretization is small.

2.2 The wise choice of the matrixes

Dans the algorithm of Newton, it is possible to use various matrixes in prediction and correction and to more or less often reactualize them. These choices are specified in `STAT_NON_LINE` under key word `NEWTON` [U4.51.03]. The most general-purpose and ruggedized choice is to use the tangent matrix reactualized with each iteration of Newton and to choose an elastic prediction.

`PREDICTION='ELASTIQUE'`
`MATRICE='TANGENTE'` (by default)
`REAC_ITER=1`
`REAC_INCR=1` (by default)

En cas de problem of convergence to spend one delicate time, it can be interesting to toggle of the tangent matrix towards the matrix of décahrge if the time step becomes too small (i.e. when the time step in question under-was cut out many times successively). The value of the time step below which one takes the matrix of discharge is given by key word `PAS_MINI_ELAS`. The frequency of updating of the matrix of discharge is defined by `REAC_ITER_ELAS`. For a definition matrix of discharge specifies, to see the key word factor `NEWTON` of [U4.51.03]).

2.3 Control

the control of the loading is a method of continuation for the method of Newton. Control as well allows in particular compute the response of a structure which would have instabilities, of origins geometrical (buckling) as material (softening). Its employment is restricted with simulations for which time does not play of role physical, which excludes *a priori* the dynamic problems, or viscous or thermomechanical.

2.4 Linear search

the method of Newton provides an increment of the unknown factors, but this increment is valid only in one vicinity of initialization. The idea of linear search is to use the direction of the increment, but by controlling the length of advanced in this direction. The pitch of advance is then chosen by minimization of a functional calculus. That in particular makes it possible to avoid certain divergences of the algorithm of Newton. However, the activation of linear search is “expensive”. It is recommended to activate it only where necessary.

2.5 The wise choice of the formulation element-finish

Tous element-finish them are not worth themselves. Certain elements behave better than others in certain situations [U2.01.10]. The choice of element-finish is made in operator `AFFE_MODELE`.

3 The Councils common to the 3 types of Dans

non-linearities this part, one gives general advices on the resolution of nonlinear problems

3.1 Gestion of the list of times

the management of the list of times is specified via operator `DEFI_LIST_INST` [U4.34.03]. By default, the under-spanning of the list of times is activated in the event of error (in the contact, the constitutive law, control, factorization…). It is also possible to start the under-spanning of the time step when the increment of a quantity exceeds a built-in threshold (`DELTA_GRANDEUR`). That makes it possible for example to make sure that an intern variable does not vary too much during a pitch time. Another possible cause of under-spanning is a not-reduction in the residue during iterations of Newton (`DIVE_RESI`). That makes it possible under-to cut out in the event of problem without waiting to arrive at the end of the iteration count authorized of Newton.
The management known as manual of the list of times is conventional management: computation follows the list of times provided by the user (with possibility of under-spanning). It is also possible to choose an automatic management of the list of times. In this case, the user gives only a few times of transitions (urgent where the loading changes for example, or times of postprocessing). According to a first experience feedback, the automatic management of the time step [bib1]:

- have a real interest if the user is not informed any *a priori* temporal discretization,
- often allows to save a considerable time for the nonlinear studies “slightly”, i.e. the studies which converge without advanced techniques (control, linear search),
- often fails for the nonlinear studies “strongly” if it is used only; it appears necessary to combine it with another technique, as linear search.

This operator also allows to extract from a preceding computation the list of really computed times and to refine it of a factor 2. It is then possible to start again same computation with this new list 2 times finer in order to control the error due to the temporal discretization.

3.2 Management of the loadings

the application of a loading in force or displacement imposed not-no one is usually done in a gradual way (by bearings of load). For that, it is possible to define a loading of type function (time), or to multiply the loading by a function of the type crawls (with `FONC_MULT` under key word `EXCIT` of `STAT_NON_LINE`), except in the event of control.

The loads being used with the boundary conditions (of standard displacement imposed no one) or for the contact are applied directly and do not require the definition of a function for a gradual application.

3.3 Management of the convergence of the total algorithm nonlinear

By default, the algorithm of total resolution nonlinear is based on the method of Newton-Raphson. The parameter setting of the convergence of the algorithm of Newton is made under the key word factor CONVERGENCE of STAT_NON_LINE [U4.51.03].
The convergence of the algorithm of Newton is characterized by the data of a value threshold of the residue. When the value of the residue is below of this threshold, the algorithm converged. By default, it is the relative residue which is taken into account. The threshold is then given by key word RESI_GLOB_RELA (10^{-6} per defect). **It is strongly to disadvise modifying this threshold of convergence**.
When the imposed loading is null (in the event of total discharge for example), the relative residue does not function and one tries to pass automatically to an absolute convergence criterion (RESI_GLOB_MAXI). This operation is transparent for the user. In a general way it is disadvised informing RESI_GLOB_MAXI. When these convergence criteria are not enough, it is possible to use finer criteria: a criterion which reasons component by component (RESI_COMP_RELA) or a criterion which reasons starting from a value of reference for each quantity (RESI_REFE_RELA).

3.4 Management of filing

the question of the filing of the computed fields into implicit is a question less important than in explicit dynamics, where the number of time steps is generally very high.

According to the size of storage which one lays out, and the number of fields which one wishes to store, it is possible to evaluate the number of pitches time that one be able to store.
By defaults, all the time steps are archived, including times of computations lately created by automatic recutting of the time step. As this number cannot be given in advance, it is advised to archive only times of the initial list of times. If this list still contains too many times, it is advised to create one 2nd list, reduced, specific to filing. If the initial list of times contains many times and that the automatic under-spanning of the time step is not activated (it is not advised), one can also limit filing by a frequency of filing.

It is also possible to reduce the overall dimension of the data structure result by extracting certain fields (operator EXTR_RESU).

3.5 Observation and followed by certain quantities

It is possible to store in an array certain components of the computed fields on parts of the model at times of a list (known as of observation) generally finer than the list of archived times. This functionality can be activated in STAT_NON_LINE under key word OBSERVATION. That makes it possible to reduce the needs for filing considerably. However, it is advisable to be careful in the use of the observation. Indeed, this functionality was conceived for some points in structure and it is not advised to make several thousands of observations to each time step, for reasons of performance.

It is also possible to follow the evolution of certain components of the computed fields during iterations of Newton and to display them in the table of convergence. This functionality can be activated in STAT_NON_LINE under key word SUIVI_DDL.

3.6 Display of information and performance

Of much information are displayed by default in the files message and results (.mess and .resu).

Essential information is the table of convergence displayed at every moment of computation. This table is very useful and in particular makes it possible to identify certain faults of convergence.

Operator STAT_NON_LINE displays the size of the matrix (many equations of the system).
According to the size of the matrix, the use of an iterative solver (as PETSC) will allow a saving of time. For information, it is considered that an iterative solver is faster than a direct solver (by default) starting from approximately 200 000 equations in 3D. The counterpart of the use of an iterative solver is an unguaranteed robustness. It is to be noted that the use of PETSC requires a version MPI of *Code_Aster* (even if one uses one processor).

The use of parallelism can be also effective if the time spent in `STAT_NON_LINE` is dominating compared to the total time of computation. For that, it is enough to choose an adequate solver (example: `MUMPS` , `PETSC`), a version MPI of *Code_Aster* and to specify the number of processes in Astk. Many advices on parallelism are given in the document [U2.08.06]. Finer information over the times spent in each part of the resolution is also displayed, and can be used for better parameterizing computation (time, memory, number of processor…). One returns for that to documentation [U1.03.03].

4 Non-linearity Cette

material left deals with problems most frequently encountered during the use of nonlinear constitutive laws.

4.1 Parameter related to the resolution of the behavior

La plupart des constitutive laws of *Code_Aster* is integrated in an analytical way. In the event of error in the integration of the constitutive law, there is thus no numerical parameter which one can exploit. The usual solution consists in Re-cutting out the total time step (or to choose a finer list of times).
Certain constitutive laws are solved locally via an iterative numerical method, for which the user can choose the authorized maximum residue and the iteration count local maximum. Most of the time, the default settings (`RESI_INTE_RELA` = 10^{-6} and `ITER_INTE_MAXI` = 20) are enough. In the event of error in the integration of the model, one can like previously under-cutting out the time step, but also authorizing local iterations in more (`ITER_INTE_MAXI`). For example, for `the MONOCRISTAL`, it is frequent to authorize 100 or 200 local iterations. The choice of the residue is more delicate. Indeed, it is important that the behavior is correctly integrated. That means that it is imperative not to increase the residue beyond the value by default (10^{-4} for example), except for the behaviors integrated by an explicit algorithm. On the other hand, it is sometimes advised to tighten the residue (for example `RESI_INTE_RELA` = 10^{-10}). In fact, according to the way in which the behavior is implemented, the residue does not have the same physical meaning (forced, strain…). It is then necessary to refer to the documentation of reference of the constitutive law in question.

4.2 Management of the plastic incompressibility

If the material is incompressible ($\nu > 0.45$) or in the event of strong plastic strains, of the oscillations on the stresses or on the trail of stresses can appear. Certain special finite elements, like the under-integrated elements or the mixed elements, make it possible to deal with these problems [U2.01.10]. If the use of under-integrated elements does not regulate the problem, then one advises the use of the mixed formulation at 2 fields (displacement and pressure) in small strains and of the mixed formulation at 3 fields in large deformation. To note that the latter functions only on quadratic meshes.

4.3 Décharge

En cas de discharges, one can encounter problems of convergence of the algorithm of Newton. One advises to use the elastic prediction (in `STAT_NON_LINE` under key word `NEWTON`). It is also useful to activate the under-spanning of the time step in the event of singular matrix (it is automatic in version 11, but to activate with the hand in the preceding versions *via* `STOP_SINGULIER = "DECOUPE"` in `STAT_NON_LINE` under the key word `SOLVER`).

4.4 Plane

plane stresses the processing of the constraint is carried out in the general frame by the method of Deborst. It is advised to often reactualize the tangent matrix (all the one with three iterations of Newton). In certain cases, convergence is reached for the algorithm of Newton, but not for the checking of the plane stress state, which leads to additional iterations, even an excessive recutting of the time step. It is then advised to activate an additional loop for better satisfying the plane stresses during iterations with Newton: ITER_CPLAN_MAXI must be selected at least equal to 5 [U4.51.11].

4.5 Damage (lenitive problems)

Pour process the problems of damage, of many advices are given in [U2.05.06]. In a general way, one advises to use control, with possibly mixed linear search.

For certain constitutive laws (ENDO_FRAGILE and ENDO_ISOT_BETON), method IMPLEX is proposed in alternative of the method of Newton. One activates this method in STAT_NON_LINE under key word NEWTON. This method is based on an explicit extrapolation of the intern variables to determine displacements from which the behavior is integrated implicitly. The nullity of the equilibrium is not checked. So it introduces an approximation of the resolution but makes it possible to guarantee the robustness of computation. A check out of the error and a tweaking of the time steps are possible via operator DEFI_LIST_INST by choosing an automatic management of the time step and a way of calculating of the time step specific to IMPLEX. From a practical point of view, the method imposes a reactualization of the matrix on each increment and only one iteration. Before any use of method IMPLEX, the document [R5.03.81] must imperatively be consulted.

4.6 Thermo-Hydro-mechanics (THM)

Les problems of THM utilize quite particular concepts. One advises to consult documentation [U2.04.05]. One gives here some general advices for the resolution. The initialization of fields is a delicate stage for which it is necessary well to pay attention (ETAT_INIT). It is necessary to use the reactualized tangent matrix. In the event of problem of convergence, it can be very useful to activate linear search (mixed preferably). Linear search does not improve however systematically convergence, it is thus to handle with precaution insofar as it can increase cost CPU.

5 Geometrical non-linearity

Cette left deals with problems most frequently encountered during computations with geometrical non-linearities.

5.1 Buckling

operator STAT_NON_LINE allows of compute a criterion of stability *via* key word CRIT_STAB=_F (TYPE='FLAMBEMENT'). The documentation of use of buckling [U2.08.04] gives many advices on the installation of a computation of stability. Some general advices here are given.
In a general way, the analysis of stability is led on reactualized stiffness matrixes. In practice, it is necessary to limit the calls to CRIT_STAB for reasons of costs of computation, by limiting times over which the analysis of stability is carried out. In complement, it is judicious to use CRIT_STAB only on the time intervals where one suspects the possibility of instabilities. It is also advisable to refine well the time step with the approach of this area. By defect, one calculates 3 critical loads (NB_FREQ). Often the first can be enough. One of the particular items related to instability is the choice of the technique of control of the algorithm. Indeed, conventional control in force is not adapted. For the approach of a boundary point, it is necessary to reduce the increment of load and to increase the maximum number of iterations. It is also advised to use control by length of arc.

5.2 Large deformation

key word DEFORMATION under COMP_INCR makes it possible to define the assumptions used for the computation of the strains. By default, one considers small displacements and small strains. (DEFORMATION = "PETIT"). That means that one remains in Hypothèse of Petites Perturbations: small displacements, small rotations, small strains (lower than approximately 5%). When this assumption is not checked any more, it is necessary to change model of strains. For slender structures (shells, plates, beams), it frequently happens that one is in large displacements, large rotations but small strains. One uses then DEFORMATION = "GROT_GDEP".

The processing of the large deformation differs according to the type of element and the constitutive law. The model of large deformation of Simo and Miehe (`DEFORMATION = "SIMO_MIEHE"`) is advised for behavior models `VMIS_ISOT_LINE`, `VMIS_ISOT_TRAC`, `ROUSSELIER` and all the behaviours with isotropic hardening only, associated with an undergoing material of the metallurgical phase changes (relations `META_X_IL_XXX_XXX` and `META_X_INL_XXX_XXX`). For the other constitutive laws, one advises models it large deformation of Miehe and Appel (`DEFORMATION = "GDEF_LOG"`), theoretically applicable to any constitutive law for modelizations 3D and 2D. More detailed explanations are in the note of use of the nonlinear behaviors [U4.51.11] in paragraph `DEFORMATION`.

6 Non-linearity of contact-rubbing

the definition of the load of contact-rubbing been the subject of a specific operator: `DEFI_CONTACT` [U4.44.11]. In addition to the choice of the entities in contact, this operator allows to choose the type of formulation of the problem of contact (discrete or continues), of the parameters for the operation of pairing and of the parameters for the phase of resolution. Many advices of setting in data are available in the document [U2.04.04], in particular on the choice of surfaces main and slave.

The taking into account of the conditions of contact with or without friction led to nonlinear problems, which can cause difficulty in the resolution of `STAT_NON_LINE`. One of the first difficulties is the blocking of the modes of rigid body. Indeed the conditions of contact-rubbing do not intervene in the scrap-metal of the rigid modes. It is thus necessary to lock the rigid modes without taking account of the conditions of contact (except rare exceptions). A bad blocking of the modes of rigid body involves the appearance of null pivots during the factorization of the matrix. Advices are given in documentation [U2.04.04]. Attention, certain formulations of contact require a particular solver. In particular, in discrete formulation, the method `STRESS` requires a direct solver (`MULT_FRONT` or `MUMPS`). In a general way, one disadvises the use of linear search with the contact. In the event of coupling with other non-linearities, it is advised to make sure that each taken non-linearity separately converges. So problems appear only when non-linearities are coupled, it is advised to work with the reactualized tangent matrix and an elastic prediction.

A comparative benchmark between *Code_Aster* and other business software were carried out in 2010 on 5 problems of contact-rubbing [2]. This benchmark in particular made it possible to highlight oscillations of the contact pressure for the formulation continues in the event of incompatible meshes, defect from which also the business software used suffers. It is then recommended to extract contact pressures starting from the stresses on edge, or to use compatible meshes.

7 Bibliography

[1] S. Geniaut, Premier REX on the automatic management of the time step in `STAT_NON_LINE`, CR-AMA-09.268, 2009

[2] T. De Soza, Bilan of the batch "Contact/Friction" in 2010 (project MNAM), CR-AMA-11.049, 2011