

To introduce a new non-linear law localised into DYNA_VIBRA

Summary:

DYNA_VIBRA allows to carry out calculation of a transitory or harmonic answer.

This document describes the method to introduce a new non-linear law localised into this operator.

This possibility is only available on a transitory calculation on reduced basis called by abuse language a modal base.

Contents

1 Introduction.....	3
2 Recall of the approach of calculation on reduced basis.....	3
3 Approach of programming.....	3
3.1 Preparation of the data.....	4
3.2 Evaluation of the force, resolution and filing of the results to each step of time.....	4
4 Introduction of a new non-linear law.....	4
5 Rules of programming.....	5
6 Activation of the new nonlinear law.....	6

1 Introduction

The operator `DYNA_VIBRA` allows to carry out calculation of a transitory or harmonic answer on physical basis or bases reduced (called bases modal by abuse language).

It is dedicated to the linear models, but a characteristic was introduced for the study of the structures having of localised non-linearities where one analyzes on a reduced basis, supposed to represent the behavior of the structure in the waveband of interest.

The taking into account of this non-linearity is only available for a transitory calculation of answer on modal basis. In *Code_Aster*, that corresponds to `BASE_CALCUL = 'GENE'` and `TYPE_CALCUL = 'TRAN'`.

2 Recall of the approach of calculation on reduced basis

The approach for the transitory calculation of answer on modal basis is the following one.

- Definition of the base of projection
- Projection of the matrices of the system on this basis
- Evaluation of the non-linear force (external force: second member)
- Projection of the force external on the basis of reduction
- Resolution of the system on reduced basis
- Restitution on physical basis

The characteristic of the resolution is at the level of the projection of the force on the reduced basis. Indeed, in the non-linear case, this force can depend on displacement, the speed even of acceleration at the moment of calculation. Here, one thus needs, with each step of calculation, to recover this information defined on the physical basis. That then requires a return on physical basis starting from the coordinates on reduced basis and then a projection of the non-linear effort calculated on the basis of reduction.

3 Approach of programming

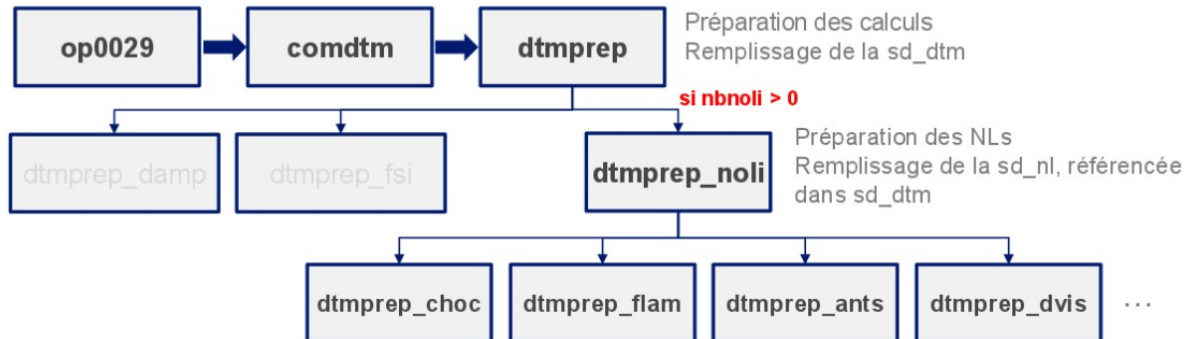
The approach of programming proceeds in two stages.

- The first stage consists in seizing all the parameters necessary for calculation.
- The second phase relates to the estimate of the non-linear force. This estimate is carried out with each step of computing time by summoning all the definite non-linear forces.

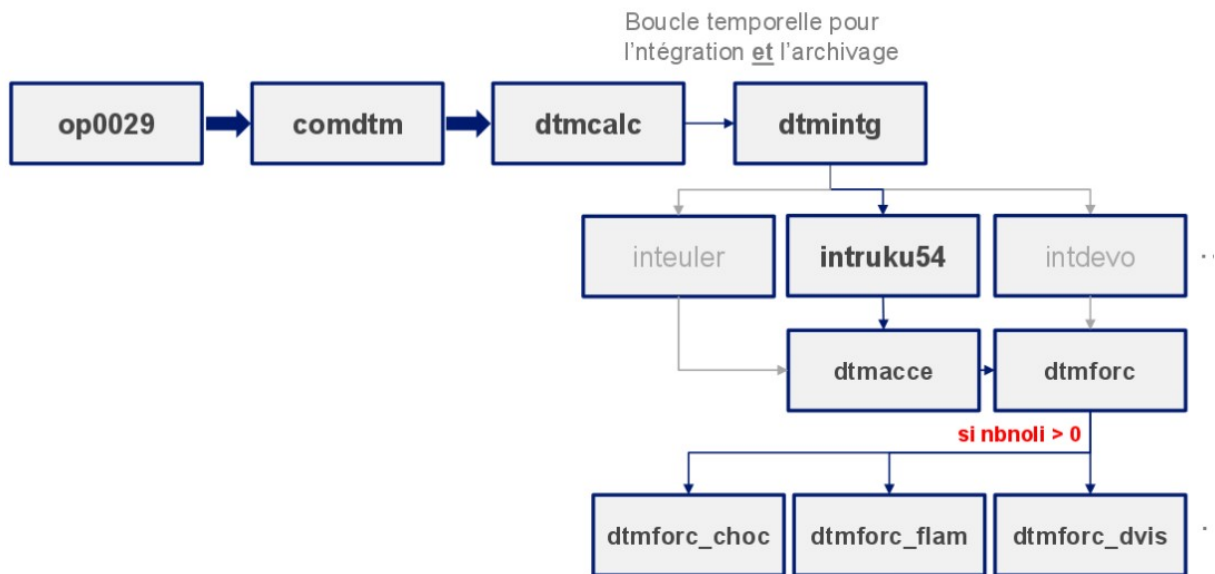
For each type of non-linearity, one seizes data necessary for calculation (first stage) with `dtmprep_noli_*`. The calculation of the non-linear force and the filing of the internal variables necessary for the treatment are carried out with `dtmforc_*`.

The following flow charts present the adopted approach.

3.1 Preparation of the data



3.2 Evaluation of the force, resolution and filing of the results to each step of time



- A la sortie de dtmforc est le cumul des efforts externes (EXCIT) et internes venant des non linéarités localisées
- Les variables internes sont stockés dans un vecteur dédié de la sd_nl : INTERNAL_VARS

18

4 Introduction of a new non-linear law

A temporary structure of data `sd_nl` was created in order to be able to store all the parameters necessary to calculation.

Several utilities are placed at the disposal:

- `nlinivec` : allows to initialize a vector with zero
- `nlsav` : allows to store the parameters of calculation in the structure of data `sd_nl`

Warning : The translation process used on this website is a "Machine Translation". It may be imprecise and inaccurate in whole or in part and is provided as a convenience.

Copyright 2021 EDF R&D - Licensed under the terms of the GNU FDL (<http://www.gnu.org/copyleft/fdl.html>)

- `nlget` : allows to recover the information stored in `sd_nl`
- `togene` : projection of a vector on reduced basis
- `tophys` : expansion of a vector generalized on physical basis
- `tophys_ms` : expansion on physical basis with taking into account of the movement of training (calculation multi-support)

To introduce a new non-linear law, the developer thus needs only to write the routine of preparation of the data: `dtmprep_noli_newf.F90` and that which carries out the calculation and the filing of the internal variables: `dtmforc_newf.F90` (without forgetting to write the files of interface `dtmprep_noli_newf.h` and `dtmforc_newf.h`).

For example for the definition of the relation effort – displacement (`RELA_EFFO_DEPL`), one needs only `dtmprep_noli_rede.F90` and of `dtmforc_rede.F90`. One can be inspired some for the writing by a new law.

One puts then up to date:

- the file `nldef.h` for the definition of non-linearity – to associate an entirety with the type of non-linearity (`NL_NEWF`) and to define the number of internal variables a priori to store (`NBVARINT_NEWF`),
- the file `nlinc.h` for the definition of the parameters of calculation (if the addition of new parameters in `sd_nl` prove to be necessary),
- the catalogue of order `dyna_vibra.capy` (keyword `COMPORTEMENT/RELATION`).

5 Rules of programming

If the developer has the free choice in the use and the naming of its parameters of work in the dedicated structure of data (`sd_nl`), there remains however obligatory to comply with the following rules in order to allow the processes of filing and resumption of calculation to function correctly with new non-linearity.

- The parameter `MX_LEVEL sd_nl` must be incremented in the routine preparing the data `dtmprep_noli_newf`. That makes it possible to avoid the crushing of data during the combination of several non-linearities.
- The parameter `NB_NONL sd_dtm` must also be incremented in `dtmprep_noli_newf`. That makes it possible to activate the calculation of nonlinear forces overall and the filing of the associated internal variables.
- The parameter `NL_TYPE sd_nl` must be saved with the whole value describing the non-linearity treated in `dtmprep_noli_newf`. This value is defined in the file `nldef.h` (e.g. `NL_CHOC = 1`, `NL_BUCKLING = 2`, etc). The parameter `NL_TYPE` is necessary in `dtmforc` in order to dispatcher correctly towards the routine `dtmforc_newf`. The use of an entirety makes it possible to optimize this very repetitive stage in the loop of integration.
- The internal variables (realities) are concaténées for all nonthe linearities and are stored under the total parameter `INTERNAL_VARS sd_nl`. In order not to crush the variables of the others nonlinearities, it is necessary to recover the index of safeguard generated and stored automatically under the parameter `INTERNAL_VARS_INDEX`. In practice:

```
cal nlget (sd_nl, _INTERNAL_VARS      , vr= came )
cal nlget(sd_nl, _INTERNAL_VARS_INDEX, vi=vindx)
start = vindx(nl_ind)
came(start+0) = valeur1
came(start+1) = valeur2
came(start+2) = valeur3
```

6 Activation of the new nonlinear law

Once routines of preparation and calculation of the nonlinear force created (dtmprep_noli_newf and dtmforc_newf), it is necessary to activate them by adding a "box":

- In dtmprep_noli, dispatchant towards dtmprep_noli_newf according to the value of the keyword `RELATION` under the keyword `factor BEHAVIOR` of `DYNA_VIBRA`.
- In dtmforc, routine responsible for the calculation of the force, according to the value of the parameter `NL_TYPE`, allowing to pass by dtmforc_newf.