Operator DYNA_LINE_TRAN

1 Drank

Compute the response transient dynamics with an unspecified temporal excitation.

The temporal loading must be given in the form of a linear combination of constant vectors forces assembled in time. They can be provided directly in the form of assembled vectors or loads which will be assembled in the algorithm.

Only the coefficients of the linear combination are function of time.

The implicit integration methods available are WILSON-theta and NEWMARK, and the explicit integration methods available are the diagram with the differences centered, and a version with adaptive step of this same diagram.

Times of archivage can be specified.

Product a result concept of the dyna_trans `type`.

Contents

2 Syntax

```
dyn [dyna_trans] = DYNA_LINE_TRAN

    ( ◊  reuse      =dyn  ,
      ◊  MODELS     =mo   ,                              [model]
      ◊  CHAM_MATER =chmat   ,                           [cham_mater]
      ◊  CARA_ELEM  =carac   ,                           [cara_elem]
      ♦  MATR_MASS  =m,                                  [matr_asse_DEPL_R]
      ♦  MATR_RIGI  =k,                                  [matr_asse_DEPL_R]
      ◊  MATR_AMOR  =c,                                  [matr_asse_DEPL_R]
      ◊  MODE_STAT  =modestat   ,                        [mode_meca]
      ◊  SCHEMA_TEMPS =_F  (
                          ♦ SCHEMA = ( |"NEWMARK", [DEFAULT]
                                       | "WILSON",
                                       | "DIFF_CENTRE"
                                       | "ADAPT_ORDRE2"
                        ),
                           ◊ BETA    =/0.25,          [DEFAULT]
                                      /beta,          [R]
                           ◊ GAMMA   =/0.5,           [DEFAULT]
                                      /gamma,         [R]
                           ◊ THETA   =/1.4,           [DEFAULT]
                                      /th,            [R]

                        ),
      ◊/ETAT_INIT       =   _F (
                ♦/RESULTAT        = Dy,               [dyna_trans]

                         ◊  /NUME_ORDRE  = nuord,     [I]
                            /INST_INIT   = to,        [R]
                         ◊  / CRITERE =   ' RELATIF', [DEFAULT]
                               ◊ accuracy = /1.E-06,  [DEFAULT]
                                            /prec,    [R]
                            / CRITERE =  "ABSOLU",
                               ♦ accuracy = prec,     [R]
                       /  DEPL = depl,                [cham_no_sdaster]
                          QUICKLY = quickly,
   [cham_no_sdaster]
                          ACCE = acce,                [cham_no_sdaster]
                          ),

      ◊EXCIT=_F         ( ♦/CHARGE      = chi,            [char_meca]
                               ◊ FONC_MULT = fi,      [function]
                           /VECT_ASSE   = goes,
[cham_no_sdaster]
                               ◊/COEF_MULT      = have,   [R]
                                 /FONC_MULT     = fi,  [function]
                                 /DEPL   = depl,       [function]
                                    QUICKLY = quickly,      [function]
                                    ACCE = acce,      [function]
                          ◊   MULT_APPUI=/       "OUI',
                                        /   "NON",  [DEFAULT]
                          ◊   DIRECTION=       (d1, d2, d3),    [l_R]
                          ◊   NOEUD=           lno,          [l_noeud]
                          ◊   GROUP_NO=        lgrno,        [l_gr_noeud]
                          ),
      ◊  EXCIT_RESU=_F  (
```

```
                    ♦RESULTAT  = resuforc,            [dyna_trans]
                    ♦COEF_MULT    = have,                  [R]
                              ),

          ◊    AMOR_MODAL =_F   (
                               ♦  AMOR_REDUIT=    l_amor, [l_R]
                               ♦  MODE_MECA = mode,   [mode_meca]
                               ◊  NB_MODE=/       nbmode,    [I]
                                                 /9999,  [DEFAULT]
                         ),
           ◊  solver =_F (see the document [U4.50.01] and the paragraph
    corresponding in present Doc.),

           ♦   INCREMENT =_F    (  ♦  /LIST_INST   = litps,  [listr8]
                                   /PAS         = dt,      [R]
                                   ◊INST_INIT    =ti  ,       [R]
                                   ◊  /INST_FIN    = tf,     [R]
                                      /NUME_FIN    = nufin,  [I]

                                ◊  PAS_CALCUL =/1     ,          [DEFAULT]
                                                /ipas  ,       [I]

                                ◊  VITE_MIN   =/"MAXI",
                                                /"NORM",       [DEFAULT]
                                ◊  COEF_MULT_PAS =  cmp,       [R]
                                                 /1.1,       [DEFAULT]
                                ◊  COEF_DIVI_PAS =  cdp   ,    [R]
                                                 /1.3334,    [DEFAULT]
                                ◊  PAS_LIMI_RELA =  plr,       [R]
                                                 /1.D-06,    [DEFAULT]
                                ◊  NB_POIN_PERIODE =/Np,   [I]
                                                  /50,       [DEFAULT]

                                ◊  NMAX_ITER_PAS =nip    ,       [I]
                                                  16,          [DEFAULT]
                                ◊  PAS_MINI =dtmin    ,       [R]

                             ),

          ◊ENERGIE=_F          ()


          ◊  ARCHIVAGE =_F (   ♦  /LIST_INST    = list       [listr8]
                               /INST        = in        [R]
                               /PAS_ARCH      = ipa          [I]
                               ◊/ CRITERE = ' RELATIF',  [DEFAULT]
                                 ◊ accuracy = /1.E-06,   [ DEFAULT]
                                              /prec,        [R]
                                 / CRITERE =   "ABSOLU",
                                 ♦ accuracy = prec,          [R]
                               ◊CHAM_EXCLU  = (  | "DEPL",
                                                 | "QUICKLY",
                                                 | "ACCE"
                                           ),
                             ),
          ◊ TITER = title,                              [l_Kn]
          ◊ INFO =  /1 ,
                    /2 ,
```

)

3 Equations of behavior under transitory excitation

the operator carries out the direct temporal integration of a transitory linear mechanical problem of the form:

$$\mathbf{M}\ddot{\mathbf{x}}+\mathbf{C}\dot{\mathbf{x}}+\mathbf{K}\mathbf{x}=\sum_i \alpha_i(t)\mathbf{F_i}(\mathbf{x})$$

where the matrixes $\mathbf{M},\mathbf{C},\mathbf{K}$ are the assembled real matrixes of the problem finite elements (respectively) of mass, damping and stiffness of the system.
α_i Are functions of time (cf DEFI_FONCTION [U4.31.02]) and the $\mathbf{F}_i$ are assembled vectors resulting from loadings in imposed force (cf AFFE_CHAR_MECA [U4.44.01]); they can be provided directly in the form of assembled vectors or loads which will be assembled in the algorithm.

The solution $(\mathbf{X},\dot{\mathbf{X}},\ddot{\mathbf{X}})$ is calculated on a temporal discretization t_i of the interval of study specified by the user.

4 Operands

4.1 Operand MODELS

◊ MODELS = Mo

Name of the model whose elements are the object of dynamic computation.

This operand is compulsory when an excitation of the type is applied charges with the key word EXCIT (cf [§4.7]).

4.2 Operand CHAM_MATER

◊ CHAM_MATER = chmat

Name of the affected material field on the model Mo, necessary when an excitation of the type is applied with the key word charges EXCIT.

4.3 Operand CARA_ELEM

◊ CARA_ELEM = carac

Name of the characteristics of the beam elements, shell etc, necessary when an excitation of the type is applied with the key word charges EXCIT.

4.4 Matrixes of the problem

♦ MATR_MASS = m

Concept stamps assembled of matr_asse_DEPL_R type corresponding to the mass matrix of the system.

♦ MATR_RIGI = K

Concept stamps assembled of matr_asse_DEPL_R type corresponding to the stiffness matrix of the system.

◊ MATR_AMOR = C

Concept stamps assembled of matr_asse_DEPL_R type corresponding to the damping matrix of the system.

N.B: the three matrixes must lean on same classification and be built with the same mode of storage. It is true also of a damping matrix built like linear combination of the stiffness matrixes and mass by the method of Rayleigh: to use the matrix of the complete mass matrix to build the damping matrix and the diagonal mass matrix (explicit diagrams such as DIFF_CENTRE or ADAPT) for integration in time can lead to a numerical instability.

4.5 Diagrams of integration. Key word SCHEMA_TEMPS

Under this key word one can inform a diagram of integration with, possibly, its parameters. The diagrams available are to be declared under operand SCHEMA.

4.5.1 Operand SCHEMA

|' NEWMARK'

implicit Diagram of integration of type NEWMARK. It is the diagram by default for the transient analysis on physical base.
One can specify the parameters of integration β and γ :

◊ BETA = beta

Value of the parameter β for the method of NEWMARK. By default $\beta=0.25$.

◊ GAMMA = gamm

Value of the parameter γ for the method of NEWMARK. By default $\gamma=0.5$.

See [R5.05.02] for the choice of other values.

|' WILSON'

implicit Diagram of integration of type WILSON. With this diagram one can inform:
◊ THETA = HT

Value of the parameter θ for the method of WILSON. By default $\theta=1,4$.
This diagram should not be used when one imposes non-zero displacements via an assembled vector. See [R5.05.02].

| "DIFF_CENTRE"

Diagram of integration clarifies by central differences. The use of this diagram imposes certain enumerated restrictions of use on [§6.3]. The theoretical description of the diagram is made in [feeding-bottle 2].

| "ADAPT_ORDRE2"

Diagram of integration clarifies with time step adaptive, alternative of the diagram of the central differences. The use of this diagram imposes certain enumerated restrictions of use on [§6.3] (see [feeding-bottle 2]).

Foot-note bucket

> *One cannot use the explicit **diagrams** (DIFF_CENTRE , ADAPT_ORDRE2) with **the shell elements and shell** (except SHB).*

4.6 Key word ETAT_INIT

This functionality allows a poursuite of a transient computation, by taking as initial state one result obtained by a preceding computation with DYNA_LINE_TRAN. It also makes it possible to define initial conditions of the fields at nodes type.

Foot-note bucket

For the diagrams of a higher nature (NEWMARK or WILSON), initial acceleration (acce_init) cheek an important role in the initialization of the diagram.

4.6.1 Operands RESULTAT

♦ /RESULTAT = Dy

Concept of the dyna_trans type resulting from a preceding computation with DYNA_LINE_TRAN, and defining the initial conditions for new computation.

4.6.2 Operands DEPL/ VITE/ACCE

/ DEPL = C

Concept corresponding to initial displacements (field at nodes of quantity DEPL_R).

QUICKLY = vo

Concept corresponding at the initial speeds (field at nodes of quantity DEPL_R).

ACCE = ao

Concept corresponding to initial accelerations (field at nodes of quantity DEPL_R).
If the key key is present, one uses the field of acceleration entered to initialize the various diagrams of integration in time according to the algorithms described in the document [R5.05.02].
If it is absent one calculates an initial acceleration by the following formula:

$$M.ao = Fext(t = \mathrm{to}) - C.vo - K.xo$$

Notice important:

When the initial state of the dynamic system is defined by fields of DEPL, QUICKLY, and/or ACCE, the components of these fields which were not explicitly indicated during the creation of the fields are considered null during transient dynamic computation.

4.6.3 Operands NUME_ORDRE/ INST_INIT

◊ /NUME_ORDRE = nuord

nuord indicates the number of archivage of preceding computation to extract and take as initial state in the case of a recovery.

/INST_INIT = to

previous Time of computation in the case of to extract and take as initial state a recovery.

In the absence of NUME_ORDRE and INST_INIT, moment of recovery is taken equal to the last time of filed preceding computation.

4.6.4 Operand CRITERE

◊ CRITERE =

Indicates with which accuracy the search of time must be done:
"RELATIF" : interval of search [(1-PREC) .instant, (1+prec) .instant]
"ABSOLU" : interval of search [time-PREC, instant+prec]

the default value of the search criterion is "RELATIF".

4.6.5 Operand accuracy

◊ accuracy =/1.E-06 [DEFAULT]
/prec [R]

Indicates with which accuracy the search of time must be done.

4.7 Key word EXCIT

◊ EXCIT =

Operand allowing to define several space-time excitations. Either by indicating a vector assembled corresponding to a loading, or of the loads which will lead to the computation and the assembly of a second member. The assembled vector can be associated with a function with temporal evolution or a constant multiplying coefficient.

The total loading is the sum of the loadings defined by all the occurrences of key word EXCIT (cf [§4.7.2]).

4.7.1 Operands VECT_ASSE / CHARGE

♦ /VECT_ASSE = vecti

Vector assembled corresponding with a loading (concept of the cham_no_DEPL_R type).

◊ /COEF_MULT = Ci

multiplicative Coefficient of the vector assembled vecti.

/FONC_MULT = α_i

See [§4.7.2].

/CHARGE = chi

chi is the loading possibly comprising the evolution of a field of temperature specified by i the ème occurrence of EXCIT.

See [§4.7.2].

4.7.2 Operand FONC_MULT

◊FONC_MULT = α_i

α_i is the multiplicative function of the time of the assembled vector or the loading specified with i the ème occurrence of EXCIT.

The loading $\mathbf{ch}$ and the boundary conditions for n occurrences of factor key word the EXCIT are:

$$\mathbf{ch}(t)=\sum_{i=1}^{n}\alpha_i(t)\mathbf{ch}_i$$

The fields of temperature are not multiplied by α_i in thermomechanical analysis.

Notice important:

*The boundary conditions of type non-zero imposed displacement can be imposed with an assembled vector or a load **; it is then necessary to use the diagram of Newmark imperatively** .*

4.7.3 Operands MULTI_APPUI / ACCE / QUICKLY / DEPL / DIRECTION / NOEUD / GROUP_NO/MODE_STAT

In the case of an excitation multi-bearings (MULT_APPUI = "OUI'), the other operands have exactly the same meaning as in factor key word the EXCIT of operator DYNA_TRAN_MODAL [U4.53.21].

4.8 Key word EXCIT_RESU

Key word allowing to define several complements of loading in the form of a transitory evolution of assembled vectors second members.

4.9 Key word AMOR_MODAL

This key word makes it possible to take into account a damping equivalent to modal damping broken up on a basis of modes precalculated in the form of concept of the mode_meca type. This damping is taken overall into account in the dynamic balance equation like a correct force with the second member $-\mathbf{C}\dot{\mathbf{X}}$.

N.B.: This way of introducing modal damping into a problem calculated on physical base can reduce the properties of stability of the time schemes. In particular for the diagram of integration "NEWMARK" it can result in reducing time step compared to time step without damping to avoid numerical divergences.

4.9.1 Operands MODE_MECA / AMOR_REDUIT / NB_MODE

- ♦ MODE_MECA=mode
- ♦ AMOR_REDUIT=l_amor
- ◊ NB_MODE=nbmode

the concept mode of the mode_meca type (entered by operand MODE_MECA) represents the base of modes precalculated on which one breaks up modal damping. This base must imperatively have the same profile of classification as that of the dynamic system defined by the parameters of the key word solver [§4.11]. It is possible to truncate modal base with a number of modes defined by NB_MODE. A default, one takes all the modes of modal base.

Modal dampings in reduced form are given in the form of a list of realities of which the number of terms is lower or equal to the number of modes taken into account. If the number of terms of the list is strictly lower, one extends this list with the value of his last term until its size reaches the number of calculated modes.

4.10 Key word ENERGIE

◊ENERGIE= _F (…)

This key word makes it possible to activate the computation of the assessment of energy, its display in the course of computation and its storage in the array of name PARA_CALC. The assessment of energy can be extracted from this array using command RECU_TABLE [U4.71.02].

4.11 Key word solver

the syntax of this key word common to several commands is described in the document [U4.50.01].
However, it is advisable to pay attention to some singularities of the use of the solvers in DYNA_LINE_TRAN :
Renumbering RENUM=RCMK (usable for LDLT and PCG) modifies the order of unknowns (NUMÉRIQUE_EQUA), whereas the renumbering for MULT_FRONT (METIS/MDA/MD) is "internal" to the solver: the unknowns (of the fields solutions) are arranged in the natural order of the nodes of the mesh.
However, in DYNA_LINE_TRAN, the matrixes being provided in entry, they have already a classification ("SANS" or "RCMK") and one not always not to change this classification in DYNA_LINE_TRAN.

The various cases below are recapitulated:

1) if matrixes K and M are numbered with:
- METHODE = "MULT_FRONT" RENUM=METIS/MDA/MD
or - METHODE = "LDLT" RENUM=SANS
then:
- one can use in DYNA_LINE_TRAN :
  - METHODE = "MULT_FRONT" RENUM=METIS/MDA/MD
  - METHODE = "LDLT" RENUM=SANS
- one NE CANNOT use in DYNA_LINE_TRAN :
  - METHODE = "LDLT" RENUM=RCMK
  (actually, one can do it but RCMK is ignored)

2) if matrixes K and M are numbered with:
- METHODE = "LDLT" RENUM=RCMK
then:
- one can use in DYNA_LINE_TRAN :
  - METHODE = "MULT_FRONT" RENUM=METIS/MDA/MD
  - METHODE = "LDLT" RENUM=RCMK
- one NE CANNOT use in DYNA_LINE_TRAN :
  - METHODE = "LDLT" RENUM=SANS
  (actually, one can do it but SANS is ignored)

4.12 Key word INCREMENT

Factor key word defining times of computation.

4.12.1 Operands LIST_INST / NOT

- For the diagrams of Newmark and Wilson:

  ♦ /LIST_INST = l_temp

  Concept lists realities of the listr8 type.

  List realities defining times t_i of computation of the solution

- For the diagrams of the central differences and with time step adaptive:

  /PAS = dt

Désigne time step used by the algorithm. This key word is compulsory for the diagram of the central differences and the diagram adaptive and nonavailable for the diagrams of Newmark and Wilson.

For the adaptive diagram, it indicates at the same time time step initial and the time step maximum one used by the algorithm.

This parameter must be sufficiently weak:

- to allow the computation of the static phases (which always use the maximum step),
- to start the algorithm correctly.

It must however be sufficiently high not to penalize the group of computation.

4.12.2 Operands INST_INIT / INST_FIN / NUME_FIN

For the diagrams of the central differences and with time step adaptive:

◊ INST_INIT = Ti

In the event of recovery one uses key word ETAT_INIT [§4.6]: under this key word, initial time is recovered with operand INST_INIT or taken equal to the last time of filed preceding computation.

Operand INST_INIT under INCREMENT must thus be only used if it did not begin again there of a preceding computation.

◊ /INST_FIN = tf

Instant of end of transient computation. Compulsory for the diagrams of the central differences and with time step adaptive.

/NUME_FIN = nufin

Number of the time of end of computation in LIST_INST (only for the diagrams of Newmark and Wilson).

If INST_INIT is not present, initial time is zero.

4.12.3 Operands VITE_MIN / COEF_MULT_PAS / COEF_DIV_PAS / PAS_LIMI_RELA / NB_POIN_PERIODE / NMAX_ITER_PAS / PAS_MINI

These operands time step relate to only the diagram with adaptive.

◊ VITE_MIN = "NORM" [DEFAULT]
/ "MAXI"

Méthode de calcul velocity of reference used to evaluate the apparent frequency.

When the denominator of the apparent frequency $(x_n - x_{n-1})$ becomes weak, the apparent frequency can become very high, which leads to an unjustified refinement of time step. To cure it, the algorithm uses the following criterion for each degree of freedom i:

$$\frac{\left|x_n^i - x_{n-1}^i\right|}{\Delta t} \leq v_{min}^i \;\Rightarrow\; f_{AP_n} = \frac{1}{2\pi}\sqrt{\left|\frac{\ddot{x}_n^i - \ddot{x}_{n-1}^i}{v_{min}^i \Delta t}\right|}$$

v_{min}^i can be calculated in two ways different according to the value from VITE_MIN:

"**NORM**" : $v^i_{min}(t_n)=Max\left(\frac{Max\left(\dot{x}^k_{n+1/2},\dot{x}^l_{n+1/2}\right)}{100},10^{-15}ms^{-1}\right)$ where k and l are the degrees of freedom of comparable nature that the degree of freedom i closest to i in classification (DX or DY or DZ ...).

"**MAXI**": $v^i_{min}(t_n)=\underset{0<t_p<t_n}{Max}\left(\frac{|v^i(t_p)|}{100},10^{-15}ms^{-1}\right)$ for the degree of freedom i .

Can be used if the order of magnitude velocity does not vary too much in the course of time.

◊ COEF_MULT_PAS = cmp

Coefficient of coarsening of time step (>1) when the error is sufficiently weak:

$\Delta t_n<\frac{0.75}{Nf_{AP_n}}$ since more than 5 consecutive steps $\Rightarrow \Delta t_{n+1}=min(\mathbf{cmp}\,\Delta t_n,\Delta t_{max})$

with $\Delta t_{max}=\Delta t_{initial}$

Its value by default ($\mathbf{cmp}=1.1$) guarantees stability and accuracy, but it can in general be increased (with more until 1.3) to accelerate integration.

◊ COEF_DIVI_PAS = cdp

Coefficient of refinement of time step (>1) when the error is higher than 1, that the nombre of iterations maximum (NMAX_ITER_PAS) is not reached and that the time step minimal one is not reached:

$$\Delta t_n>\frac{1}{Nf_{AP_n}},\quad Niter<Niter_{max} \text{ and } \Delta t_n>\mathbf{plr}*\Delta t_{initial}\Rightarrow \Delta t_n=\frac{\Delta t_n}{\mathbf{cdp}}$$

value by default formulates is of 1.3334 , that is to say a reduction of a factor $0,75$.

◊ PAS_LIMI_RELA = plr

Coefficient applied to time step initial to define the limit of refinement and thus the time step minimal one:

$$\Delta t_{min}=\mathbf{plr}*\Delta t_{initial}$$

◊ NB_POIN_PERIODE = N

Number of points per apparent period. It is this parameter which fixes the accuracy of computation. It must be at least equal to 20; its value by default (50) guarantees a satisfactory accuracy (about 1à 2%) in most case.

◊ NMAX_ITER_PAS

maximum Number of reductions of time step per computation step:

if $err>1$ and $N_{iter}<N_{iter}max$: $\Delta t_n=cdp*\Delta t_n$

It is by default equal to 16, which limits the coefficient of reduction of the step to $(1/1,33)^{16}=10^{-2}$ by iteration. NMAX_ITER_PAS can be:

- increased to allow time step to fall in a more brutal way,
- decreased if time step seems excessively refined.

◊ PAS_MINI = dtmin

minimal Value of time step. If the conditions of reduction in time step are met, time step flow will be able to then decrease up to this limiting value.

If the user does not give a value to this optional parameter, then the code will calculate the time step minimal one from PAS_LIMI_RELA.

4.13 Key word ARCHIVAGE

◊ ARCHIVAGE =

Factor key word defining the archivage. In the absence of this factor key word, all time step are filed.

Whatever the option of archivage chosen, one time step files the last and all the associated fields to allow a possible poursuite.

4.13.1 Operands LIST_INST/INST

◊ /LIST_INST = list

Lists realities defining times of computation for which the solution must be filed in the result concept dyna_tran.

◊ /INST

Times of computation for which the solution must be filed in the result concept dyna_tran .

4.13.2 Operand PAS_ARCH

/PAS_ARCH = ipa

Whole defining the periodicity of archivage of the solution of transient computation in the result concept dyna_trans.

If ipa = 5 one files all the 5 computation steps.

4.13.3 Operand CRITERE

◊ CRITERE =

Indicates with which accuracy the search of time to be filed must be done:
"RELATIF" : interval of search [(1-PREC) .instant, (1+prec) .instant]
"ABSOLU" : interval of search [time-PREC, instant+prec]
the default value of the search criterion is "RELATIF".

4.13.4 Operand accuracy

◊ accuracy =/1.E-06 [DEFAULT]
/prec [R]

Indicates with which accuracy the search of time to be filed must be done.

4.13.5 Operand CHAM_EXCLU

◊ CHAM_EXCLU = (| "DEPL",
| "QUICKLY",
| "ACCE",
)

Makes it possible to exclude the archivage from one or more fields among "DEPL", "QUICKLY" and "ACCE".

This exclusion is ignored for the last moment of computation: the three fields are necessary for a POURSUITE.

4.14 Operand TITER

◊ TITER = title

Titrates data structure result [U4.03.01].

5 Product concept

dyn is a product concept of the dyna_trans type which will contain from 1 to 3 fields at nodes for each filed time.

These cham_no has as a symbolic name:

DEPL : displacement
QUICKLY : velocity
ACCE : acceleration

6 Stage of execution

the use of the diagrams of the central differences and adaptive imposes certain restrictions of use:

- these two diagrams require the use of a diagonal mass matrix. A test checks that the mass matrix was created with option "MASS_MECA_DIAG" of CALC_MATR_ELEM. In addition, the mass matrix must be stored on sky line,
- it should not have other boundary conditions there only blocked degrees of freedom.
  A test checks that there are no boundary conditions of the connections type between degrees of freedom.
  It is not either possible to impose non-zero displacements via an assembled vector,
- for the diagram of the central differences, one makes sure that the time step selected one checks the stability conditions:
  $dt < 0,05 / f_{max}$ with $f_{max} = \max_{1 \le i \le nddl} \left(\frac{1}{2\pi} \sqrt{\frac{k_{ii}}{m_{ii}}} \right)$ and k_{ii} m_{ii} diagonal terms of the stiffness matrixes and mass.

7 Bibliography

1) BATHE K.J. : Finite Element Procedures in Analysis engineering. Prentice-hall, 1982.

2) LIGHT A.C.: Introduction of the explicit diagrams “central differences” and “time step adaptive” into operator DYNA_LINE_TRAN of *the Code_Aster*. Note EDF HP51/97/067/A 1997.