
Operator DYNA_NON_LINE

1 Drank

Compute the dynamic evolution of a structure whose material or geometry has a nonlinear behavior. They can be for example nonlinearities of material (plasticity or of geometry (large displacements)) [R5.05.05]. The syntax of this command is very similar to that of operator STAT_NON_LINE [U4.51.03].

The dynamic evolution is studied from an initial state, reference configuration, which can be produced by a quasi-static analysis (operator STAT_NON_LINE [U4.51.03]) or former dynamics (operator DYNA_NON_LINE).

The dynamic evolution can be studied in several successive works, by a poursuite from one time already calculated, if a data base were defined in the profile of study of the user.

Product a concept of the evol_noli type.

Contents

1	But1.....	
2	Syntaxe4.....	
3	Opérandes7.....	
3.1	Operands MODELE/CHAM_MATER/CARA_ELEM7.....	
3.2	Operands MODE_STAT/MASS_DIAG7.....	
3.3	Key word EXCIT7.....	
3.3.1	Operands CHARGE/FONC_MULT7.....	
3.3.2	Operand TYPE_CHARGE7.....	
3.3.3	Operands MULT_APPUI/ACCE/VITE/DEPL/DIRECTION/NOEUD/GROUP_NO7.....	
3.4	Key word CONTACT8.....	
3.5	Key word SOUS_STRUC8.....	
3.6	Key words COMP_INCR and COMP_ELAS8.....	
3.7	Key word ETAT_INIT8.....	
3.8	Key word INCREMENT8.....	
3.9	Key word NEWTON9.....	
3.10	Key word RECH_LINEAIRE9.....	
3.11	Key word SOLVEUR9.....	
3.12	Key word CONVERGENCE9.....	
3.13	Key word ARCHIVAGE9.....	
3.13.1	Operand CHAM_EXCLU9.....	
3.14	Key word AMOR_RAYL_RIGI9.....	
3.15	Key word AMOR_MODAL10.....	
3.15.1	Operands MODE_MECA/AMOR_REDUIT/LIST_AMOR/NB_MODE10.....	
3.15.2	Operand REAC_VITE10.....	
3.16	Description of the diagram of integration in time [bib2] [R5.05.05].....	10
3.16.1	Cases SCHEMA = "NEWMARK".....	11
3.16.2	Cases SCHEMA = "HHT".....	11
3.16.3	Cases SCHEMA = "THETA_METHODE".....	11
3.16.4	Cases SCHEMA = "KRENK".....	12
3.16.5	Cases SCHEMA = "DIFF_CENT".....	12
3.16.6	Cases SCHEMA = "TCHAMWA".....	12
3.16.7	Operand COEF_MASS_SHIFT12.....	
3.17	Key word CRIT_STAB12.....	
3.18	Key word MODE_VIBR13.....	
3.19	Operand ENERGIE13.....	
3.20	Operand PROJ_MODAL14.....	
3.20.1	Operands MODE_MECA, NB_MODE14.....	
3.20.2	Operands MASS_GENE, RIGI_GENE, AMOR_GENE.....	14
3.20.3	Operands DEPL_INIT_GENE, VITE_INIT_GENE, ACCE_INIT_GENE.....	14

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.

[3.21 Key word EXCIT_GENE15.....](#)

[3.22 Operand INFO15.....](#)

[3.23 Operand TITRE15.....](#)

[4 Example: motion of a pendulum of large amplitude16.....](#)

2 Syntax

```

dynamnl [evol_noli] = DYNA_NON_LINE
(
  ◊reuse=dynamnl , [evol_noli]
  ◊ MODELE=mo , [model]
  ◊CHAM_MATER=chmat , [cham_mater]
  ◊MODE_STAT=modestat , [mode_meca]
  ◊ CARA_ELEM=carac , [cara_elem]
  ◊MASS_DIAG=/ "OUI",
                /"NON",

  ◊EXCIT =_F (
    ◊TYPE_CHARGE = "FIXE_CSTE", [DEFAULT]
                /"SUIV",
                /"DIDI",
    ◊CHARGE =chi , [char_meca]
                [char_cine_meca]
    ◊/FONC_MULT=fi ,

[function]
                /DEPL =depl , [function]
                VITE=vite , [function]
                ACCE=acce , [function]
    ◊MULT_APPUI=/ "NON", [DEFAULT]
                /"OUI",
    ◊DIRECTION= (d1, d2, d3), [l_R]
    ◊ NOEUD=lno , [l_noeud]
    ◊ GROUP_NO=lgrno , [l_gr_noeud]
  ),

  ◊SOUS_STRUC =_F ( see the document [U4.51.03]),

  ◊ AMOR_RAYL_RIGI = /"TANGENT", [DEFAULT]
                  /"ELASTIC",

  ◊AMOR_MODAL =_F (
    ◊MODE_MECA=mode , [mode_meca]
    ◊/AMOR_REDUIT=l_amor , [l_R]
    /LIST_AMOR =lisamor [listr8]
    ◊NB_MODE=/nbmode , [I]
                /9999, [DEFAULT]
    ◊REAC_VITE=/ "OUI", [DEFAULT]
                /"NON",
  ),

  ◊ | COMP_INCR =_F ( see the document [U4.51.11]),
    | COMP_ELAS =_F ( see the document [U4.51.11]),

  ◊ETAT_INIT =_F ( see the document [U4.51.03]
  | QUICKLY = quickly,
[cham_no_DEPL_R]
  | ACCE = acce,
[cham_no_DEPL_R]
  ),

  ◊ INCREMENT =_F (see the document [U4.51.03])
  ◊EXCIT_GENE =_F (

```

```

                                ◇FONC_MULT=fomult
[fonction_sdaster]
                                ◆VECT_GENE=vecgen
[vect_asse_gene]
                                ),

                                ◇ NEWTON =_F ( see the document [U4.51.03]),

                                ◇ RECH_LINEAIRE=_F ( see the document [U4.51.03]),

                                ◇SOLVEUR =_F ( see the document [U4.50.01] ),

                                ◇CONVERGENCE =_F ( see the document [U4.51.03]),

                                ◇MODE_VIBR =_F (
                                ◇ NB_FREQ= / 3, [DEFAULT]
                                    /nbfreq, [I]
                                ◇ MATR_RIGI = "ELASTIC", [DEFAULT]
                                    /"TANGENT",
                                    /"SECANTE",

                                ◇ BANDE=intba , [listr8]
                                ◇ /LIST_INST =liste_r8 , [listr8]
                                    /INST =l_r8 , [R]
                                    /PAS_CALC =npas , [I]
                                ◇PRECISION=/1.E-6 ,
[DEFAULT]
                                    /prec, [R]
                                ◇CRITERE=/
[DEFAULT]
                                    "RELATIF",
                                    /"ABSOLU",
                                ),

                                ◇CRIT_STAB =_F ( see the document [U4.51.03]),
                                ◇ENERGIE =_F ()

                                ◇ ARCHIVAGE =_F ( see the document [U4.51.03]),

                                ◆ SCHEMA_TEMPS =_F (
                                    ◆ SCHEMA=/ "NEWMARK",
                                        / "HHT",
                                        / " THETA METHODE " ,
                                        / "KRENK " ,
                                        / "DIFF_CENT",
                                        / "TCHAMWA",
                                ◇ COEF_MASS_SHIFT =/0 ., [DEFAULT]
                                    /coef , [R]
                                { If SCHEMA=' NEWMARK"
                                    ◇ BETA =/0.25 , [DEFAULT]
                                        /beta , [R]
                                    ◇ GAMMA =/0.5 ,
[DEFAULT]
                                        /gamm , [R]
                                }
                                { If SCHEMA=' HHT'
                                    ◇ALPHA=/-0.3 ,
[DEFAULT]
                                        /alph , [R]
                                    ◇MODI_EQUI = "OUI",
                                        / "NON", [DEFAULT]

```

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.

```

}
{If SCHEMA=' THETA METHODE'
    ◇THETA=/1
[DEFAULT]
    /theta , [R]
}
{If SCHEMA=' KRENK'
    ◇KAPPA=/1
[DEFAULT]
    /kappa , [R]
}
{If SCHEMA=' TCHAMWA'
    ◇PHI=/1.05
[DEFAULT]
    /phi , [R]
}
{If SCHEMA=/
    "NEWMARK",
    / "HHT",
    / "THETA METHODE",
    / "KRENK",
    ◆FORMULATION=/ "DEPLACEMENT",
    / "VELOCITY",
    / "ACCELERATION",
}
{If SCHEMA=/
    "DIFF_CENT",
    / "TCHAMWA",
    ◆FORMULATION='ACCELERATION',
    ◇ STOP_CFL=/ "OUI", [DEFAULT]
    / "NON",
}, ),

◇ OBSERVATION = _F ( see the document [U4.51.03]),
◇ DISPLAY = _F ( see the document [U4.51.03]),
◇ PROJ_MODAL = _F (
    ◆MODE_MECA=mode , [mode_meca]
    ◇NB_MODE=/nbmode , [I]
    /9999 , [DEFAULT]
    ◇ /MASS_GENE =massgen [matr_asse_gene_R]
    RIGI_GENE=rigigen [matr_asse_gene_R]
    /AMOR_GENE =amorgen
[matr_asse_gene_R]
    ◇ DEPL_INIT_GENE = deplgen, [ vect_asse_gene ]
    ◇ VITE_INIT_GENE = vitegen, [ vect_asse_gene ]
    ◇ ACCE_INIT_GENE = accegen, [ vect_asse_gene ]
    ),

◇INFO=/ 1 , [DEFAULT]
    /2 ,

◇TITER = tx , [kN] )
    
```

3 Operands

3.1 Operands MODELE/CHAM_MATER/CARA_ELEM

These operands have the same meaning as in the document [U4.51.03].

3.2 Operands MODE_STAT/MASS_DIAG

◇MODE_STAT = modestat

Name of the static mode necessary in the case of a seismic computation with excitations multi-bearings [R4.05.01].

◇MASS_DIAG = / "OUI",
/ "NON",

Option to be used with a time scheme clarify [bib2] and which makes it possible to solve with a lumped mass matrix (diagonalized). This option is not available for all element types, in particular the discrete ones (in this case, it is necessary to solve with the consistent mass matrix). Into implicit it is not possible to use a lumped matrix (emission of a fatal error if the user is in this case)

3.3 Key word EXCIT

◇EXCIT = _F

This key word factor makes it possible for each occurrence to describe a load (requests and boundary conditions), and possibly a multiplying coefficient and/or a kind of load.

The operands have the same meaning as in the document [U4.51.03] but there are some specificities related to the dynamics.

Notice important for the time schemes:

If one imposes boundary conditions in displacement which evolve in the course of time, it is necessary to take account of the primal unknown of the diagram used. These conditions in fact are imposed in acceleration into explicit (because it is the primal unknown). That means that one must enter DYNA_NON_LINE the derivative second of the signal in displacement which one wants to impose. This evolution of imposed displacement must thus be differentiable at least twice in time. In the same way for the theta-diagram of velocity, the primal unknown is the velocity and one must enter DYNA_NON_LINE the derivative first of the signal in displacement which one wants to impose.

3.3.1 Operands CHARGE/FONC_MULT

◆ CHARGE = ch_i
◇ FONC_MULT = f_i

the operands have the same meaning as in the document [U4.51.03].

3.3.2 Operand TYPE_CHARGE

◇ TYPE_CHARGE = /"FIXE_CSTE" , [DEFAULT]
/ "SUIV",
/ "DIDI",

the operand has the same meaning as in the document [U4.51.03], except that a loading cannot be controlled in dynamics, and thus `tchi` cannot be `FIXE_PILO`.

3.3.3 Operands MULT_APPUI/ACCE/VITE/DEPL/DIRECTION/NOEUD/GROUP_NO

```
◇MULT_APPUI=/      "NON",                [DEFAULT]
                  /"OUI",
◆ACCE=ac           ,                      [function]
◆VITE=vi           ,                      [function]
◆DEPL=dp           ,                      [function]
◇DIRECTION=       (dx, Dy, dz, drx, dry, drz), [1_R]
◇ /NOEUD           =lno ,                 [1_noeud]
◇ /GROUP _ NO=lgrno ,                   [1_groupe_no]
```

In the case of an excitation multi-bearings (MULT_APPUI= "OUI"), the other operands have exactly the same meaning as in the key word factor EXCIT of operator DYNA_TRAN_MODAL [U4.53.21]. In this case, fields "DEPL", "QUICKLY", "ACCE" correspond respectively to displacements, velocities and accelerations of relative motion compared to the motion of training multi-bearings. New fields "DEPL_ABSOLU", "VITE_ABSOLU", "ACCE_ABSOLU" are then created and respectively correspond at displacements, the speeds and accelerations of absolute motion, adds motion of training multi-bearings and relative motion compared to this motion of training multi-bearings.

3.4 Key word CONTACT

◆ CONTACT = contact

This key word simple makes it possible to activate the resolution of contact-friction or the taking into account of a unilateral connection. `contact` is a concept resulting from operator DEFY_CONTACT [U4.44.11].

The operand has the same meaning as in the document [U4.51.03].

3.5 Key word SOUS_STRUC

◇SOUS_STRUC=_F

This key word factor makes it possible to specify which are the loadings to be used for under - static structures which then are obligatorily part of the model.

The operand has the same meaning as in the document [U4.51.03].

3.6 Key words COMP_INCR and COMP_ELAS

the syntax of these key words common to several commands is described in the document [U4.51.11]. All the behavior models supported by STAT_NON_LINE are available also in DYNA_NON_LINE, provided that the computation of the mass matrix of the elements concerned is envisaged.

3.7 Key word ETAT_INIT

◇ETAT_INIT =_F

Under this key word are defined the initial conditions of the problem. The operands of key word ETAT_INIT have the same meaning as in the document [U4.51.03].

In dynamics, one can define in more the initial velocity fields and of acceleration.

◆ / | QUICKLY = quickly
/ | ACCE = acce

If key keys EVOL_NOLI, DEPL, and QUICKLY are absent, one supposes that the initial state is with displacements, velocities and stresses null, and one calculates accelerations corresponding to the loading at time `instini` defined by the operand INST.

3.8 Key word INCREMENT

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.

◆ INCREMENT =_F

Defines the list of times of computation. The operands of key word INCREMENT have the same meaning as in the document [U4.51.03].

3.9 Key word NEWTON

◆ NEWTON =_F

Specifies the characteristics of the method of resolution of the nonlinear incremental problem (method of Newton-Raphson). The operands of key word NEWTON have the same meaning as in the document [U4.51.03].

3.10 Key word RECH_LINEAIRE

```
◆ RECHERCHE_LINEAIRE =_F (
  ◇ METHODE = "CORDE" [DEFAULT]
  /"MIXTE" )
```

Makes it possible to activate the linear search. The operands of key word RECH_LINEAIRE have the same meaning as in the document [U4.51.03], except that the method CONTROL does not exist.

3.11 Key word solver

◆ SOLVEUR =_F

the syntax of this key word common to several commands is described in the document [U4.50.01].

3.12 Key word CONVERGENCE

◆ CONVERGENCE =_F

This key word describes the parameters making it possible to appreciate the convergence of the method of NEWTON used to solve the nonlinear mechanical problem. The operands of key word CONVERGENCE have the same meaning as in the document [U4.51.03].

3.13 Key word ARCHIVAGE

◆ ARCHIVAGE =_F

Makes it possible to file or certain results at all or certain times of computation.

In the absence of this key word all time step are filed, including times of computations lately created by automatic recutting of time step. The operands of key word ARCHIVAGE have the same meaning as in the document [U4.51.03], except for keyword CHAM_EXCLU.

3.13.1 Operand CHAM_EXCLU

```
◇ CHAM_EXCLU= | ' DEPL '
               | ' VITE '
               | ' ACCE '
               | ' SIEF_ELGA '
               | ' VARI_ELGA '
```

Makes it possible to specify the fields which will not be filed, except with the last time step.

3.14 Key word AMOR_RAYL_RIGI

```
◇ AMOR_RAYL_RIGI = "TANGENT", [DEFAULT]
                  /"ELASTIC"
```

This key word makes it possible to specify the stiffness matrix \mathbf{K} which will be used to build the damping of Rayleigh $\mathbf{C} = \alpha \cdot \mathbf{K} + \beta \cdot \mathbf{M}$.

With the default value ("TANGENT"), the matrix \mathbf{K} will be the same one as that which is used for the computation of the internal forces. By choosing the "ELASTIC" value, one forces the computation of the damping of Rayleigh with the elastic stiffness matrix.

For the lenitive models or of type GLRC one advises to use the elastic matrix.

3.15 Key word AMOR_MODAL

```
◇ AMOR_MODAL =_F
```

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 $-\mathbf{C} \cdot \dot{\mathbf{X}}$.

3.15.1 Operands MODE_MECA / AMOR_REDUIT / LIST_AMOR / NB_MODE

```
◆MODE_MECA=mode
◆/AMOR_REDUIT=l_amor ,
  /LIST_AMOR =lisamor
◆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 [§3.113.11]. 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 by 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.

3.15.2 Operand REAC_VITE

```
◇REAC_VITE=/ "OUI", [DEFAULT]
  /"NON",
```

If its value is "OUI", one modifies the correct force of modal damping to each internal iteration of Newton defined in key word NEWTON [§99].

If its value is "NON", one updates this term only to the beginning of each time step.

3.16 Description of the diagram of integration in time [bib2] [R5.05.05]

```
◆SCHEMA_TEMPS=_F ()
  ◇STOP_CFL=/ "OUI", [DEFAULT]
  /"NON",
  ◇FORMULATION=/ "DEPLACEMENT",
  /"VELOCITY",
  /"ACCELERATION",
```

One can is to use an implicit method of NEWMARK (key word SCHEMA='NEWMARK' or modified average acceleration: SCHEMA='HHT' with MODI_EQUI = "NON"), of HILBER-HUGHES-TAYLOR (SCHEMA=' HHT' with MODI_EQUI = "OUI') or a THETA_METHODE or, finally, the diagram of KRENK.

With an implicit scheme, the resolutions in displacement, velocity or acceleration are currently available (key word FORMULATION = "DEPLACEMENT", "VELOCITY" or "ACCELERATION").

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.

Conversely, one can choose an explicit method of central differences type (key word `SCHEMA='DIFF_CENT'`) or a dissipative diagram of type TCHAMWA (key word `SCHEMA='TCHAMWA'`). With an explicit diagram, one can solve only in acceleration (key word `FORMULATION = "ACCELERATION"`).

The explicit diagrams being conditionally stable, it can be useful to check if time step the given as starter one of computation observes the stability condition well (CFL condition). If `STOP_CFL = "OUI"` (default), then if the list of times provided by the user comprises one or more time step higher than the stability condition, computation stops in fatal error. If `STOP_CFL = "NON"`, one emits an alarm and continues computation.

In all the cases, time step criticizes is given in the file of messages for information.

The computation of the CFL is not programmed for all the elements (in particular the discrete elements are ignored.); the CFL estimated by *Code_Aster* can thus be larger (less penalizing) than the real CFL, with the risks of brutal divergence which results from this.

Into explicit, it is also recommended to use a lumped mass matrix (diagonalized): what one can obtain with the key word `MASS_DIAG = "OUI"` [§77].

Note:

- Choice `MASS_DIAG='NON'` is disadvised with shells `DKT`.
- With elements `DKT/DKTG` it is necessary to specify in `AFFE_CARA_ELEM`, under the key word factor `COQUE`, the simple key word `INER_ROTA = "OUI"`. If not the mass matrix is singular and the explicit diagram is unusable.
- For the multi-step diagrams (Newmark with `MODI_EQUI='OUI'` and theta-diagram), it is necessary to recompute the internal forces at previous time. This operating process is problematic at the first time of a resumption of computation. Indeed, if there are command variables (which thus depend on the time of computation), it is necessary to recover the value of previous time. It is not always possible, in particular if one is not in `reuse` and if `ETAT_INIT` is done from individual fields (not use of keyword `ETAT_INIT/EVOL_NOLI`). In this case, one does not calculate this contribution to the preceding step, which can possibly modify the convergence of the algorithm slightly.

3.16.1 Case `SCHEMA = "NEWMARK"`

```

◇ BETA      =/0.25 , [DEFAULT]
                /beta, [R]
◇ GAMMA     =/0.5 , [DEFAULT]
                /gamm, [R]

```

the integration method in time is that of `NEWMARK`, with the values given of the parameters `beta` and `gamm`.

When neither `beta` is specified, nor `gamm`, one has the method known as "trapezoidal rule" (`beta=0.25 ; gamm=0.5`) which, into linear, is unconditionally stable and no parasitic dissipation brings (*i.e numerical damping*), but which, in nonlinear, can be unstable [bib1] [bib2]. Case

3.16.2 `SCHEMA = "HHT"` ◇

```

ALPHA = - 0.3 , [DEFAULT] ] /alph
                , [R] ◇
MODI_EQUI=/" OUI' ,/"
                NON", [DEFAULT] ] For

```

`MODI_EQUI = "NON"` (default value), the integration method in time (implicit diagram of integration) is that of the modified average acceleration (of the family of Newmark) ([bib1], [bib2]), with the negative value of `alph` given. More `|alph|` is large, more the numerical damping brought by computation is important. But this dissipation is sometimes necessary, in nonlinear, to ensure stability (unless assigning a damping by material to structure). For

`MODI_EQUI = "YES"`, the integration method in time (implicit diagram of integration) is that of Hilber - HUGHES - TAYLOR (HHT or) α -méthode [bib2], with the negative value of `alph` given.

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.

More $|\alpha|$ is large, more the numerical damping brought by computation is important. Compared to the preceding diagram (MODI_EQUI = "NON") of modified average acceleration, induced numerical damping is more "selective": it is weaker with low and intermediate frequency (asymptotically no one with frequency null) and it will grow more quickly when the frequency becomes large. This

second diagram is based on the first with, moreover, one modification of the balance equation (one shifts in time the internal forces and outsides) [bib2]. Case

3.16.3 SCHEMA = "THETA_METHODE" ◇

```
THETA=/1 . , [DEFAULT ] /theta  
 , [R]
```

the diagram of integration in time is an implicit theta-diagram of order one, of velocity. In the case of use with contact loads, one must also call on the continuous method (AFFE_CHAR_MECA /CONTACT/METHODE = "CONTINUE") and the formulation of velocity (FORMULATION = "VELOCITY"). theta

must be ranging between 0,5 and 1: 0,5 corresponds to a minimum of numerical dissipation, 1 corresponds to a maximum of numerical dissipation. theta = 1 makes it possible to find the diagram of Eulerian. This diagram is also usable with a formulation in displacement. Case

3.16.4 SCHEMA = "KRENK" ◇

```
KAPPA=/1 . , [DEFAULT ] /kappa  
 , [R] It
```

of integration in times of Krenk is implicit, of a nature 1 and dissipative. Its use is thus recommended, just like the theta-diagram, for the irregular problems like the shocks.

Numerical dissipation is controlled by the parameter kappa , which must be equal to or higher than 1. If it is worth 1, then the diagram will not bring dissipation. The larger kappa will be, the more dissipation will be high. Case

3.16.5 SCHEMA = "DIFF_CENT"

the diagram of the central differences is an explicit diagram of order two of the family of Newmark, parameters BETA = 0 and GAMMA = 0.5. It is about a diagram to a step which does not present numerical dissipation. Case

3.16.6 SCHEMA = "TCHAMWA" ◇

```
PHI=/1.05 , [DEFAULT ] /phi  
 , [R]
```

an alternative to the diagrams of the central differences is the diagram developed by Bertrand Tchamwa and Christian Wielgosz. This

explicit diagram has several interesting characteristics. It is not a derivative of Newmark, and the variation of its parameter PHI allows a controllable numerical dissipation of the high frequencies. When it is worth 1, dissipation is null. Not too much not to degrade the Flow condition and to preserve properties of stability comparable to the diagram of the central differences, it is recommended not to choose a PHI higher than 1.10. 1.05 is the value chosen by default. Operand

3.16.7 COEF_MASS_SHIFT ◇

```
COEF_MASS_SHIFT = 0. [DEFAULT ] /coef
```

the data of the coefficient coef makes it possible to carry out a shift of the mass matrix which M becomes:

$$M' = M + coef K$$

The value of this coefficient, by default null, must be non-zero to be able to reverse in dynamics with diagram clarifies the mass matrix when this one has null terms for certain specific degrees of freedom, for example the pressure for the elements of modelization HM .

The entry of this coefficient also makes it possible to strongly improve convergence in dynamics with implicit scheme in this same type of modelization by imposing a cut-off frequency inversely proportional to the value of coeff (at the cost of a light distortion of all the eigenfrequencies of the system) Key word

3.17 CRIT_STAB ◇

CRIT_STAB = _F This

key word makes it possible to start computation, at the end of each increment of time, of a criterion of stability, identical to what is proposed in STAT_NON_LINE . This criterion is useful to detect, during the loading, the point from which one loses stability (by buckling for example). It is not about a criterion of stability to the dynamic meaning (negative damping). The operands of key word CRIT_STAB have the same meaning as in the document [U4.51.03]. It

is however advisable to announce that the use of CRIT_STAB on models fluid - structure coupled (formulation, (u, p, ϕ) cf *documentation* [R4.02.02], which is available with DYNA_NON_LINE but not STAT_NON_LINE) obliges to exclude the fluid degrees of freedom because the total assembled stiffness matrix is singular for these degrees of freedom. For that, it is necessary to combine key word DDL_EXCLUS (by excluding all the degrees of freedom specific to the fluid elements, like "PRES ", "PHI " and , in the cases with free face, "DH ") and MODI_RIGI = "OUI". More details are given in documentations [U4.51.03], [U2.06.11] and [U2.08.04]. Key word

3.18 MODE_VIBR ◇

```

MODE_VIBR=_F      (◇
  NB_FREQ=/3      , [DEFAULT
                   , [I]
                   ] /nbfreq
  MATR_RIGI      = " ELASTIC ", [DEFAULT
                               TANGENT"/"
                               SECANTE"/"
                               ] /"
  TAPE      =intba      , [listr8
                          /LIST_INST =liste_r8      , [listr8
                          =l_r8      , [R]
                          =npas      , [I]
                          ] /PAS_CALC
  accuracy      =/1.e-6      [DEFAULT
                              ] /prec
  CRITERE=/ " RELATIF ", [DEFAULT
                       ABSOLU"/"
                       ] /"
  This

```

key word makes it possible to start computation, at the end of each increment of time, of a search for vibratory eigen modes. This criterion is useful to follow, during transient computation, the evolution of the vibratory response of nonlinear structure. This

criterion is calculated in the following way: at the end of one time step, one solves. $\det(\mathbf{K} - \omega^2 \cdot \mathbf{M}) = 0$ can \mathbf{K} either be the elastic stiffness matrix, or the coherent tangent matrix at time running, or the secant matrix. is \mathbf{M} the mass matrix. This modal analysis is authorized only for the symmetric matrixes (mass and stiffness).

Key word NB_FREQ (3 per default) indicates the number of eigenfrequencies to calculating. One

stores the eigen mode the corresponding to smallest eigenfrequency in S. D. RESULTAT , under name DEPL_VIBR. This eigen mode can be extracted and visualized (like a field of displacements or a classical eigen mode). It is standardized with on 1 the largest component of displacement.

The key word `TAPE` makes it possible to specify on which bande de fréquence one wants to make the search for eigenfrequencies.

Times for which one wants to do a calculation of oscillatory mode are given by a list of times `LIST_INST` or `INST` (list `_r8` or `I_r8`) or by a frequency `PAS_CALC` (all of `npas` time). In the absence of

these keywords the vibratory modal analysis is carried out with all time step. Keywords accuracy and `CRITERE` make it possible to select times, cf [U 4.71.00]. Operand

3.19 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 the command `RECEIVED_TABLE` [U 4.71.02]. Operand

3.20 PROJ_MODAL ◇

`PROJ_MODAL` =_F This

key word makes it possible to calculate on a modal base (or Ritz) calculated beforehand. It is to be used with a diagram of integration in explicit time. Operands

3.20.1 MODE_MECA , NB_MODE ◆

```
MODE_MECA =mode , [mode_meca ] ◇  
NB_MODE=/nbmode , [I] /9999  
 , [DEFAULT ] One
```

specify the base to be used (`MODE_MECA`) and the number of modes (`NB_MODE`) . Notice

important:

Modal base must lean on a coherent classification with that of the calculated evolution (cf [S10]10: even profile of classification. Operands

3.20.2 MASS_GENE , RIGI_GENE , AMOR_GENE ◇

```
/MASS_GENE = massgen, [matr _asse_gene_R] RIGI_GENE  
 = rigigen, [matr _asse_gene_R] AMOR_GENE  
 = amorgen, [matr _asse_gene_R] These
```

operands are used together if one wants to dynamically condense part of the model to the linear behavior, by strictly calculating by `DYNA_NON_LINE` only fields with the behavior non-inéaire. This, in order to reduce the size of the model of computation. In this case, it is necessary to calculate a modal base of Ritz on all the fields: the nonlinear field modelled for computation which calls on `DYNA_NON_LINE` and the other dynamically condensed linear fields. This base must be orthogonalized compared to the mass and with a linear stiffness of all the fields. It must simply be representative of motions activating all the fields. On the other hand, one will not inform behind `MODE_MECA` that the modes obtained by reduction of the base of Ritz to the model of computation treated by `DYNA_NON_LINE` . An example of computation is provided by the case test `SDNV107A` [V5.03.107].

Operand `MASS_GENE` makes it possible to enter the projection of the mass matrix of all the fields on the basis of Ritz with a diagonal storage. Operand `RIGI_GENE` makes it possible to

enter the projection of the matrix stiffness of the linear fields condensed only on the basis of Ritz with a full storage. Operand AMOR_GENE makes it possible to possibly enter the projection of a damping matrix (if it exists) linear fields condensed only on the basis of Ritz with a full storage. Operands

3.20.3 DEPL_INIT_GENE , VITE_INIT_GENE , ACCE_INIT_GENE ◇

```
DEPL_INIT_GENE      = deplgen, [vect      _asse_gene] ◇
VITE_INIT_GENE      = vitegen, [vect      _asse_gene] ◇
ACCE_INIT_GENE      = accegen, [vect      _asse_gene] These
```

operands are associated with the use of operands MASS_GENE , RIGI_GENE and possibly AMOR_GENE in key word PROJ_MODAL . It are used to introduce a vector generalized resulting from projection by PROJ_VECT_BASE (TYPE = ' DEPL') from a field displacement or velocity or acceleration of the complete model (including the field with the linear behavior) on modal base of this complete model. This generalized vector will the model serve as initial condition of the evolution of the generalized coordinates of computation on reduced to the nonlinear field. It will then be necessary also to inform the operands of key word ETAT_INIT with corresponding fields displacement, velocity, acceleration, stress, local variable of the small-scale model. An example of computation is provided by the case test SDNV107C [V5.03.107]. Key word

3.21 EXCIT_GENE ◇

```
EXCIT_GENE          = _F (◇
  FONC_MULT          =fomult , [fonction_sdaster ] ◇
  VECT_GENE          =vecgen , [ vect_asse_gene])
This
```

key word répétable is associated with the use of operands MASS_GENE , RIGI_GENE and possibly AMOR_GENE in key word PROJ_MODAL . It is used to introduce the applied forces on fields of linear behavior condensed dynamically and not modelled into computation which calls on a diagram of integration in explicit time. These forces are projected on the basis of Ritz calculated on all the fields. VECT_GENE

is used to inform the force vectors projected on the basis of Ritz. FONC_MULT is used to inform the multiplying function depending on time associated with each vector within an occurrence with key word EXCIT_GENE . Operand

3.22 INFO ◇

```
INFO =inf Makes it possible
```

to carry out in the message file various intermediate printings.

Other printings are done systematically during the nonlinear calculation, independently of value affected to the key word INFO : they are the printings of the residues and the relative increments of displacement during iterations of Newton. Attention , the files .mess can become very important with INFO = 2. Operand

3.23 TITER ◇

```
TITER = tx tx
```

is the title of computation. It will be printed at the top of the results. See [U4.03.01]. Example:

4 motion of a pendulum of great simple amplitude

```
#TITER Pendulum in great oscillation #
#PENDULE
  CONSTITUTES Of a Cable element (test SDNL100A). #
RESU
=DYNA_NON_LINE (MODELE=MO, CHAM_MATER=CHMAT, CARA_ELEM=CARA, EXCIT
  = (_F (CHARGE = CHA1), _F
    (CHARGE = CHA2)), INCREMENT
  = _F (INST_INIT = 0. , LIST_INST = L_INST1), ARCHIVAGE
  = _F (LIST_INST = L_INST2), SCHEMA
  _TEMPS=_F (SCHEMA=' NEWMARK', FORMULATION
    = ' DEPLACEMENT'), COMP
  _ELAS=_F (RELATION = "CABLE", DEFORMATION
    = "GREEN"), CONVERGENCE
  = _F (RESI_GLOB_RELA = 1.E-6, ITER_GLOB_MAXI
    = 100), NEWTON
  = _F (REAC_ITER = 1))
  FIN
```

()

- the load cha 1 forces node 1 to remain fixed and with node 2 to move in the vertical plane XZ
- the load cha 2 is gravity,
- command DYNA_NON_LINE specifies that:
 - the integration method of time will be that of "NEWMARK ", "trapezoidal rule" (also called average acceleration), because it there has no argument under "NEWMARK ",
 - the initial state, at time 0, is with null displacement, i.e. displacements will be evaluated from the initial position, and at velocity null,
 - iterative computation will continue as much as the relative residue will be, $> 10^{-2}$ but the number of the iterations will be restricted to 100, finally
 - the tangent matrix of the linear system to solve will be revalued with each iteration (by default since key word NEWTON is absent).