

---

## Operator THER\_NON\_LINE

---

### 1 Drank

---

Compute the thermal response with nonlinearities of behaviors and boundary conditions.

The equation of heat is solved in evolutionary mode (except if no list of time is provided, only the steady mode is then calculated). Nonthe linearities come either from the behavior (characteristics of the material depend on the temperature), or of the boundary conditions (radiation in infinite medium, nonlinear flux, thermal source depending on the temperature). A formulation in enthalpy was selected in order to more easily take into account the phase changes of the material.

The computation evolutionary can be initialized, at the first time in three different ways (key word `TEMP_INIT`):

- by a constant temperature,
- by a field of temperature, definite, or extracted as a preliminary from a preceding computation,
- by a steady computation.

This operator also allows to solve the problems of drying (nonlinear) by solving the equation of heat where the water concentration  $C$  is comparable to a temperature, for the resolution. Thermal conductivity is in this case the coefficient of diffusion, nonlinear in  $C$  and function, possibly, of a temperature calculated as a preliminary.

To model the hydration of the concrete, the operator also allows to add a term function source of the variable of hydration to the equation of heat. This term is then given by an equation of evolution where the temperature intervenes.

Product concept by the operator `THER_NON_LINE` is of `evol_ther` type as for a linear analysis by `THER_LINEAIRE` [U4.54.01].

When a sensitivity analysis of result compared to a parameter is required, there is production of as many data structures of the `evol_ther` type of necessary parameters.

## 2 Syntax

```
temper [evol_ther] = THER_NON_LINE

( ◊ reuse=temper          ,
  ◊ MODELE=mo             , [model]
  ◊ CHAM_MATER=chmat      , [cham_mater]
  ◊ EXCIT = _F (
      ◊CHARGE=char        , [load]
      ◊FONC_MULT=fonc     ,
[function]
  ),
  ◊ETAT_INIT=            _F (
[DEFAULT]
      ◊/STATIONNAIRE=' OUI',
      /VALE              =tinit  , [R]
      /CHAM_NO           =cinit  ,
[cham_no]
      /EVOL_THER         =temp   ,
[evol_ther]
      /NUME_ORDRE        =nuini_evol , [I]
      /INST              =inst_evol , [R]
      /CRITERE           = ' RELATIF', [DEFAULT]
      ◊PRECISION         = 1.E-6, [DEFAULT]
      /prec, [R]
      /CRITERE           = ' ABSOLU',
      ◊PRECISION         =prec   , [R]
      ◊INST_ETAT_INIT=inst_init  , [R]
  ),
  ◊INCREMENT=_F (
      ... to see STAT_NON_LINE [U4.51.03] key word INCREMENT ...
  ),
  ◊ COMP_THER_NL = _F (
      ◊ RELATION = "THER_NL", [DEFAULT]
                  /"THER_HYDR",
                  /"SECH_GRANGER",
                  /"SECH_MENSI",
                  /"SECH_BAZANT",
                  /"SECH_NAPPE",
                  ◊/TOUT=' OUI',
      /GROUP_MA    =l_grmail  ,
[l_gr_ma]
      MAILLE=l_maille      , [l_ma]
  ),
  ◊ EVOL_THER_SECH=resuther , [evol_ther]
```

```

    ◇ NEWTON=_F (
        ◇ REAC_ITER=/0 , [DEFAULT]
        /it, [I]
        ◇ RESI_LINE_RELA=/1.E-3 , [DEFAULT]
        /reslin, [R]
        ◇ ITER_LINE_MAXI=/0 , [DEFAULT]
        /iterl, [R]
    ),

    ◇ CONVERGENCE=_F (
        ◇ RESI_GLOB_RELA=/1.E-6 , [DEFAULT]
        /testr, [R]
        ◇ RESI_GLOB_MAXI=/testl , [R]
        ◇ ITER_GLOB_MAXI=/10 , [DEFAULT]
        /iterl, [R]
    ),

    ◇ PARM_THETA =/theta , [R]
        /0.57 , [DEFAULT]

    ◇ solver = _F (
        ... to see [U4.50.01]...
    ),

    ◇ ARCHIVAGE= _F (
        ... to see STAT_NON_LINE [U4.51.03] key word ARCHIVAGE...
    ),

    ◇ OBSERVATION= _F (
        ... to see STAT_NON_LINE [U4.51.03] key word OBSERVATION...
    ),

    ◇ TITER =titer , [l_Kn]

)

```

## 3 Operands

---

### 3.1 MODEL Operand

◆MODELE = Mo

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

### 3.2 Operand CHAM\_MATER

◆CHAM\_MATER = chmat

Name of the affected material field on the model.

### 3.3 Key word EXCIT

◆EXCIT =

Key word factor allowing to define several loadings. For each occurrence of the key word factor, one defines a load possibly multiplied by a function of time.

#### 3.3.1 Operand CHARGE

◆CHARGE = tank

Concept of the type charges produces by AFFE\_CHAR\_THER or AFFE\_CHAR\_THER\_F [U4.44.02].

**Notice important:**

*For each occurrence of the key word factor EXCIT the various concepts tank used must be built on the same model Mo .*

#### 3.3.2 Operand FONC\_MULT

◆FONC\_MULT = fonc

multiplicative Coefficient function of the time (concept of type function, three-dimensions function or formula) applied to the load.

**Notice important:**

*The concomitant use of FONC\_MULT with a load containing of the thermal loadings depending on the temperature is prohibited; i.e. for loadings of the type ECHANGE\_ , RAYONNEMENT, SOUR\_NL or FLUNL .*

### 3.4 Key word ETAT\_INIT

◆ETAT\_INIT =

Makes it possible to define the initial field from which evolutionary computation is carried out. The initial field is affected sequence number 0 and initial time takes as value the first reality of the list of time as defined by INCREMENT.

**Note:**

*If key word ETAT\_INIT is absent, one carries out only steady computation at the first time defined under key word INCREMENT .*

## 3.4.1 Operand STATIONNAIRE

/STATIONNAIRE = "OUI"

the initial value is then result of a preliminary steady computation. This computation takes into account the boundary conditions defined under key word CHARGE.

## 3.4.2 Operand /VALE

VALE = tinit

the initial value of temperature is taken constant on all structure.

## 3.4.3 Operand CHAM\_NO

/CHAM\_NO = cinit

the initial value is defined by a cham\_no of temperature (result of operators CREA\_CHAMP [U4.72.04]).

## 3.4.4 Operand EVOL\_THER

/EVOL\_THER = temp

the initial value is extracted from a data structure of the evol\_ther type.

## 3.4.5 Operand NUME\_ORDRE/INST

◇/NUMÉRIQUE\_ORDRE = nuinievol  
/INST = instinievol

Sequence number of the field to be extracted from this data structure. Extraction of the initial thermal state in l'evol\_ther\_temp starting from the number of archivage NUME\_ORDRE or of the time of archivage INST to carry out the poursuite of computation. If NUME\_ORDRE or INSTS are not filled, one takes the last existing number filed in evol.

### Note:

*Attention, it acts of the sequence number in data structure read in recovery by key word EVOL\_THER precedent. If this data structure were calculated with a list of times different from that used under the key word factor INCREMENT of the current resolution, it is imperative to inform NUME\_ORDRE under INCREMENT, the same value of sequence number corresponding to different physical times. If the two lists of times are identical, one can exempt oneself to inform the same NUME\_ORDRE twice, under ETAT\_INIT and INCREMENT.*

## 3.4.6 Operand INST\_ETAT\_INIT

◇ INST\_ETAT\_INIT = istetaini

One can associate a value of time istetaini in this initial state. By default:

- when the initial state is defined by the data of the fields, associated time ago.
- when the state is given by a concept evol\_noli, it is time in preceding computation (istetaini = instinievol).

## 3.4.7 Operand PRECISION/CRITERE

Confer [U4.71.00].

## 3.5 Key word INCREMENT

◆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].

## 3.6 Key word COMP\_THER\_NL

◇ COMP\_THER\_NL =

the resolution of drying was added in *the Code\_Aster* because of analogy of the equations of the thermal and drying. That supposes to assimilate the variable of computation of drying, the water concentration, with a variable of the type "TEMP" during the resolution.

By default, the resolution carried out will be nonlinear thermal. This key word factor thus makes it possible to distinguish the resolution of drying from the thermal. Moreover, the equation of drying is characterized by a coefficient of diffusion which can be expressed in various forms. This key word factor also makes it possible to choose one of the equations of the drying, defined by the statement of its coefficient of diffusion, available in Aster. To carry out a nonlinear computation of thermal, this key word becomes optional, and the notion of behavior is transparent for the user.

### Note:

*If key word COMP\_THER\_NL is absent, one carries out a computation of nonlinear thermal "standard".*

## 3.6.1 Operand RELATION

◆RELATION : / "THER\_NL" [DEFAULT]  
/ "THER\_HYDR"  
/ "SECH\_GRANGER"  
/ "SECH\_MENSI"  
/ "SECH\_BAZANT"  
/ "SECH\_NAPPE"

the syntax and the processing of this key word are similar to the use of key keys COMP\_INCR and COMP\_ELAS of operator STAT\_NON\_LINE.

/ "THER\_NL"

standard nonlinear thermal Resolution.

Supported modelizations:

- continuums 3D: 3D
- continuums 2D: 2D, AXIS

/ "THER\_HYDR"

Resolution of the equation of heat with an additional source term:  $Q\dot{\xi}$

$Q$  is the heat of hydration, presumedly constant. The variable of hydration  $\xi$  is solution of the nonlinear law of evolution, solved simultaneously with the problem of thermal:

$$\dot{\xi} = A(\xi)e^{-E/RT}$$

One will refer to the documentation of operator DEFI\_MATERIAU for the meaning of the various parameters.

At the initial time of computation, the variable of hydration takes for value the field found under name "HYDR\_ELNO" in the data structure evol\_ther definite under ETAT\_INIT. By default, the initial hydrous state is taken virgin  $\xi = 0$ .

Supported modelizations:

- continuums 3D: 3D
- continuums 2D: 2D, AXIS

/ "SECH\_GRANGER"

Resolution of the drying characterized by the equation  $\frac{\partial C}{\partial t} - \text{Div}[D(C, T)\nabla C] = 0$

It is the equation of nonlinear heat where the variable of drying  $C$  holds the role of the temperature. The choice of the behavior model makes it possible to define the coefficient of diffusion selon  $D(C, T)$  various usual forms of the literature. The formulation of Granger of the coefficient of diffusion is given by the statement:

$$D(C, T) = A \exp(BC) \frac{T}{T_0} \cdot \exp\left(\frac{Q_s}{R}\right) \left(\frac{1}{T} - \frac{1}{T_0}\right)$$

One will refer to the documentation of operator DEFI\_MATERIAU for the meaning of the various parameters. In the case of the use of this model SECH\_GRANGER, it is necessary to make sure of coherence between the material used and the constitutive law: i.e. that key word SECH\_GRANGER indeed was indicated in DEFI\_MATERIAU for the material used.

Supported modelizations:

- continuums 3D: 3D
- continuums 2D: 2D, AXIS

As the resolution of drying is carried out by an operator of thermal, the supported modelizations are thermal modelizations, but which do not have whereas a conceptual value of a geometrical nature.

/ "SECH\_MENSI"

Resolution of the drying characterized by the model of MENSI.

In the case of the use of this model SECH\_MENSI, it is necessary to make sure of coherence between the material used and the constitutive law: i.e. that key word SECH\_MENSI indeed was indicated in DEFI\_MATERIAU for the material used. Supported modelizations: analog with SECH\_GRANGER.

/ "SECH\_BAZANT"

Resolution of the drying characterized by the model of BAZANT.

In the case of the use of this model SECH\_BAZANT, it is necessary to make sure of coherence between the material used and the constitutive law: i.e. that key word SECH\_BAZANT indeed was indicated in DEFI\_MATERIAU for the material used. Supported modelizations: analog with SECH\_GRANGER.

/ "SECH\_NAPPE"

Resolution of drying with a coefficient of diffusion defined by a three-dimensions function Aster.

In the case of the use of this model SECH\_NAPPE, it is necessary to make sure of coherence between the material used and the constitutive law: i.e. that key word SECH\_NAPPE indeed was indicated in DEFI\_MATERIAU for the material used. Supported modelizations: analog with SECH\_GRANGER.

## 3.6.2 Operands TOUT/GROUP\_MA/MESH

```
◇/TOUT      =      "OUI"  
  /GROUP_MA =l_grmail  
  /MAILLE   =      l_maille
```

Specifies meshes to which the behavior model is applied. In a way similar to the mechanics, one can use several different laws of drying, applied to complementary mesh groups. On the other hand, the thermal cannot be mixed with drying. Behavior "THER\_NL" is necessarily applied to all the mesh, option TOUT: "OUI", default option, which is in fact, in the general case, transparent for the user.

## 3.7 Operand EVOL\_THER\_SECH

```
◇ EVOL_THER_SECH= resuther
```

This operand is specific with the resolution of drying. Drying is solved after a preliminary thermal computation in the general case, (computation not coupled but chained thermal/drying), the thermal field intervening like auxiliary variable, making it possible to calculate the coefficient of diffusion of certain models. It is an input datum of the computation of the drying, which must be a data structure of the evol\_ther type. This key word is compulsory only for models "SECH\_GRANGER" and "SECH\_NAPPE", whose coefficient of diffusion depends on the temperature. The data structure of here well informed thermal evolution will have been obtained by a preceding execution of an operator of thermal, linear or not.

## 3.8 Key word NEWTON

```
◇NEWTON     =
```

Specifies the characteristics of the method of resolution of the nonlinear problem (method of NEWTON-RAPHSON).

### 3.8.1 Operand REAC\_ITER

◇ REAC\_ITER =/0 [DEFAULT]  
/it

the matrix used for the total iterations of the method is the tangent matrix which are revalued at the beginning of each increment of time and all the `it` iterations of NEWTON for an increment of time given (precisely with the iterations of number `it`, `2it`, `3it`...). Thus with the first iteration of NEWTON, one reassembles the tangent matrix only if `it` is worth 1: if not one keeps the matrix used in the phase of prediction. By convention if `it` is worth 0 the matrix is not revalued during all time step.

### 3.8.2 Operand RESI\_LINE\_RELA/ITER\_LINE\_MAXI

◇ RESI\_LINE\_RELA=/1.E-3 [DEFAULT]  
/reslin  
◇ ITER\_LINE\_MAXI=/0 [DEFAULT]  
/itlin

In fact the parameters of the linear search make it possible to ensure a better convergence of the method of NEWTON (cf [R5.03.01] for more details). One gives the maximum nombre of iterations `itelin` to be carried out (the value by default 0 indicates that one does not make a linear search) and the accuracy `reslin` to reach to carry out the convergence of the linear search.

**Note:**

*It is not necessary to specify an accuracy nor a nombre of iterations very high, the practice showing that 2 or 3 iterations of linear search are sufficient. One can thus be satisfied to ask for 3 iterations with the accuracy by default.*

## 3.9 Key word CONVERGENCE

◇ CONVERGENCE :

Allows to define the values associated with the convergence criteria:

*If none of the two operands following is present, then all occurs like if: RESI\_GLOB\_RELA = 1.E-6.*

### 3.9.1 Operand RESI\_GLOB\_RELA

◇ RESI\_GLOB\_RELA = 1.e-6  
/testr

the algorithm continues the external iterations as long as the relative residue is higher than `testr`.

$$\left( \sum_{i=1, \dots, nb\ ddl} (F_i^n)^2 \right)^{1/2} / \left( \sum_{i=1, \dots, nb\ ddl} (S_i)^2 \right)^{1/2} > testr$$

où désigne  $F_i$  the residue and  $S_i$  the thermal loading, the index  $n$  indicates  $n^{ième}$  the iteration.

### 3.9.2 Operand RESI\_GLOB\_MAXI

◇ RESI\_GLOB\_MAXI=/ 1.e-3  
/testl

the algorithm continues the external iterations as long as the absolute residue is higher than `testl`.

$$\max_{i=1,\dots,nb\ ddt} |F_i^n| > test1$$

where  $F_i$  indicates the residue, the index  $n$  indicates  $n^{ième}$  the iteration.

## 3.9.3 Operand ITER\_GLOB\_MAXI

◇ ITER\_GLOB\_MAXI = 10  
/iter1

the algorithm continues the iterations as long as their number is lower than `iter1`.

## 3.10 Operand PARM\_THETA

◇ PARM\_THETA = theta

the argument `theta` is the parameter of the theta-method applied to the evolutionary problem. It must be ranging between 0 (explicit method) and 1 (completely implicit method). In the absence, key word, the value used is  $\theta=0.57$ , a little higher than  $\theta=0.5$  corresponds to the diagram of Crank-Nicholson. The incidence of the choice of `theta` on the stability of the method is detailed in [R5.02.02].

## 3.11 Key word solver

◇ SOLVEUR =

This key word factor is optional: it makes it possible to choose another solver of resolution of system.

This operand is common to all the global commands [U4.50.01].

## 3.12 Key word ARCHIVAGE

◇ ARCHIVAGE =

This key word is optional: by default, all the computed fields for all the calculated steps is filed in the result concept resulting of the command. It is used for storing certain sequence numbers in a data structure `result` and/or excluding from storage certain fields.

This key word is identical to its equivalent for operator `STAT_NON_LINE`, to refer to documentation [U4.51.03] for the description of under key words.

### Note:

*In the event of stop of computation by lack of time CPU, time step previously calculated are saved in the base.*

## 3.13 Key word OBSERVATION

◇ OBSERVATION =

This key word makes it possible post-to treat certain fields at nodes or with the elements on parts of model times of a list (known as of observation) generally more refined than the list of the times filed defined in the key word `ARCHIVAGE` [§3.13] (where one the model stores all the fields on all). It is used primarily with economies as storage.

This key word is répétable and allows the creation of an array of the same observation name than the result concept of `THER_NON_LINE`.

For the description of syntax of this factor key word, to refer to the documentation of the key word are equivalent of `DYNA_NON_LINE` [U4.53.01].

## 3.14 Operand TITER

◇ TITER = title

Titrate that one wants to give to result stored in `temper`, data structure of the `evol_ther` type [U4.03.01].

## 4 Modelization

the problems of nonlinear thermal can be treated with models using the finite elements 3D, 2D or AXIS described in the documents [U3.22.01], [U3.23.01] and [U3.23.02] and [U3.24.01].

## 5 Example

One defined below the principal commands used to carry out a computation of thermal not - linear transient. The example indicates how to continue computation by enriching the result concept and how to specify the "initial" field.

```
lr8=DEFI_LISTE_REEL          (... )

conduc=DEFI_FONCTION          (          NOM_PARA=' TEMP',
                                VALE= ...
                                )

enthal=DEFI_FONCTION          (          NOM_PARA=' TEMP',
                                VALE= ... )

alu=DEFI_MATERIAU            (THER_NL=_F  ( LAMBDA=conduc    ,
                                           BETA=enthal      ))

...

tempe=THER_NON_LINE          ( MODELE=moth, CHAM_MATER=chmat,
                                ETAT_INIT=_F      ( VALE=20.0          ),
                                INCREMENT=_F      ( LISTE_INST=lr8    ,
                                                    NUME_FIN= 20),
                                EXCIT=_F         ( CHARGE=chth      ),
                                CONVERGENCE=_F   ( RESI_GLOB_RELA= 1.E-8,
                                                    ITER_GLOB_MAXI= 20),
                                )

...

tempe=THER_NON_LINE          ( reuse=tempe      ,
                                MODELE=moth, CHAM_MATER=chmat,
                                ETAT_INIT=_F      ( EVOL_THER=tempe  ,
                                                    NUME_ORDRE=20),
                                INCREMENT=_F      ( LISTE_INST=lr8    ),
                                EXCIT=_F         ( CHARGE=chth      ),
                                CONVERGENCE=_F   ( RESI_GLOB_RELA= 1.E-6,
                                                    ITER_GLOB_MAXI= 10 ),
                                )

...
```