Operator DEFI_FONCTION

1 Drank

To define a real or complex function of a real variable. This operator allows to define, for example, of the characteristic materials function of the temperature, or the boundary conditions which depend on a variable of space or time.

The product concept by this operator is of standard `function`.

2 Syntax

```
F   [function]   =DEFI_FONCTION      (

                 ♦NOM_PARA=np        ,

                 ◊NOM_RESU=/            "TOUTRESU",
[DEFAULT]
                                  /nr ,                              [K8]

                 ♦/VALE=lv                  ,                          [l_R]
                    /VALE_C              =lv   ,                          [l_C]
                    /VALE_PARA           =la   ,
[listr8]
                       ♦VALE_FONC=lo           ,                       [listr8]
                    /ABSCISSE         =labs ,                          [l_R]
                       ♦ORDONNEE=lord             ,                    [l_R]
                    /NOEUD_PARA       =lno  ,                          [l_noeud]
                       ♦MAILLAGE=ma            ,                       [mesh]
                       ♦VALE_Y=ly                ,                        [l_R]

                 ◊PROL_DROITE=/             "CONSTANT",
                                       /   "LINEAIRE",
                                       /   "EXCLUDED",
[DEFAULT]

                 ◊PROL_GAUCHE=/             "CONSTANT",
                                       /   "LINEAIRE",
                                       /   "EXCLUDED",
[DEFAULT]

                 ◊INTERPOL=           |   ' LIN',
[DEFAULT]
                                       | "LOG",
                                       | "NON",

                 ◊INFO=/1               ,
[DEFAULT]
                                  /2  ,

                 ◊VERIF=              |   ' CROISSANT',
[DEFAULT]
                                  |   ' NON',

                 ◊TITER=ti           ,                                  [l_Kn]

                                  )
```

3 Operands

3.1 Operand NOM_PARA

♦NOM_PARA = Np

Indicates the name of the parameter (variable or X-coordinate) of the function.
The possible values for Np are:

"ABSC", "AMOR", "DRX", "DRY", "DRZ", "DSP", "DX", "DY", "DZ", "ENDO", "EPAIS", "EPSI", "FREQ", "HYDR", "INST", "META", "NEUT1", "NEUT2", "NORM", "PAD", "PCAP", "PGAZ", "PLIQ", "PORO", "PULS", "PVAP", "SAT", "SECH", "SIGM", "TEMP", "TSEC", "QUICKLY", "X", "Y", "Z"

3.2 Operand NOM_RESU

◊NOM_RESU = NR

Indicates the name of result (8 characters). The function thus created is NR = F (Np).

Note:

Some commands (CALC_FONCTION , DEFI_MATERIAU ...) check the coherence of the names of the parameter and result according to their context. For example, one expects a curve of tension defined by a function of which NOM_PARA=' EPSI' and NOM_RESU=' SIGM'.

3.3 Operand /VALE

VALE = lv

lv is the list of values (x1, y1,…, xn, yn) with in the order:

- x1, y1 (the first value of the parameter and the corresponding value of result),
- … ,
- xn, yn (the last value of the parameter and the corresponding value of result).

Note:

The list lv of values must be described in the order of the X-coordinates (X) increasing.

3.4 Operand VALE_C

/VALE_C = lv

lv is the list of the values (X, there, Z,…, xn, yn, Zn) with:

- xi values of the parameter
- … ,
- yi, zi the real part and the imaginary part of the function complexes for this parameter.

3.5 Operands ABSCISSE / ORDERED

/ABSCISSE = labs
/ORDONNEE = Lord

One provides the values of the X-coordinates and the Y-coordinates of the function separately in the shape of lists of actual values (x1, x2,..., xn) for ABSCISSE and (y1, y2,..., yn) for Y-coordinate. The two lists must have the same cardinal.

3.6 Operand VALE_PARA / VALE_FONC

```
/VALE_PARA     =
/VALE_FONC     = lo
```

Même operation that ABSCISSE, ORDERED except that the lists are provided in the form of concept listr8 produces by DEFI_LIST_REEL [U4.34.01].
VALE_PARA and VALE_FONC must be identical cardinals if not the command stops in error.

3.7 Operand NOEUD_PARA

```
/NOEUD_PARA    = lno
```

lno nodes list allowing to define the values of the X-coordinates of the function to be defined.
The X-coordinates will be equal to the curvilinear abscisses of the nodes on the curve which they define.

3.8 Operands PROL_DROITE and PROL_GAUCHE

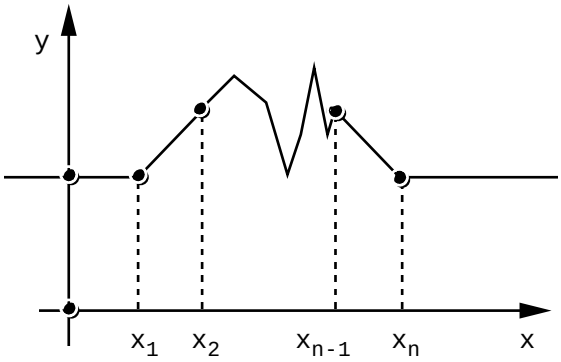
◊PROL_DROITE and PROL_GAUCHE =

Define the type of prolongation on the right (on the left) of the field of definition of the variable:

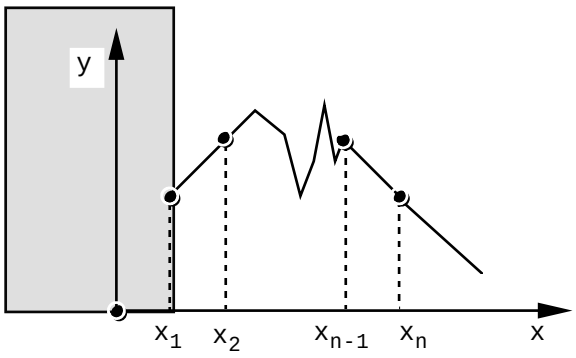
- "CONSTANT" for a prolongation with the last (or first) value of the function,
- "LINEAIRE" for a prolongation along the first definite segment (PROL_GAUCHE) or last definite segment (PROL_DROITE),
- "EXCLUDED" if the extrapolation of the values apart from the field of definition of the parameter is prohibited (in this case if a computation requires a value of the function out of field of definition, the code will stop in fatal error).

For example:

- PROL_DROITE = "CONSTANT", PROL_GAUCHE = "CONSTANT"



- PROL_DROITE = "LINEAIRE", PROL_GAUCHE = "EXCLUDED"



Note::

The type of prolongation and interpolation are independent one of the other.

3.9 Operand INTERPOL

◊INTERPOL =

Standard of interpolation of the function enters the values of the field of definition of the function: a type for the interpolation of the parameter and for the interpolation of the function. This is obtained by providing a list of texts among:

INTERPOL = ("LIN", "LOG")

"LIN" : linear,

"LOG" : logarithmic curve,

"NON" : one does not interpolate (and thus the program will stop if one asks for the value of the function for a value of the parameter where it was not defined).

Note:

If only one value is specified, she is taken into account at the same time by the interpolation of the parameter and the function. INTERPOL = "LOG" is equivalent to ("LOG", "LOG").

3.10 Operand INFO

◊INFO = Specifies the options of printing on the message file .

1: no the printing (option by default)

2: printing of the parameters plus the list of the first 10 values in the order ascending of the parameter

3.11 Operand VERIF

◊VERIF =

operator DEFI_ FUNCTION checks that the values of the X-coordinates are strictly increasing. If it is not the case, an error is started. This is the behavior by default, VERIF is worth "GROWING".
The user has the possibility of not making this checking by indicating VERIF=' NON'. In this case, the function is reordered by increasing X-coordinates. An alarm is emitted if the X-coordinates of the function were not increasing.
On the other hand, the X-coordinates must imperatively be strictly monotonous.

3.12 Operand TITER

◊TITER = Ti

Titrates attached to the product concept by this operator [U4.03.01].

3.13 Operands MAILLAGE and VALE_Y

these two key words should be informed if one defines the function from NOEUD_PARA.

MAILLAGE = my

Name of the mesh associated with the list with node lno.

VALE_Y = lv

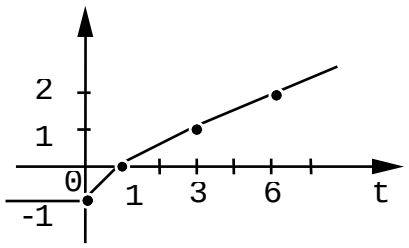
Liste of the values of the Y-coordinates of the function to be defined.

4 Definition of a function depending on time

4.1 Function and variables entered in the form of realities

Definition of a function (linear by pieces) depends on time (parameter INST).

```
EX_1 = DEFI_FONCTION  (
              NOM_PARA='INST',
              VALE = (  0. ,-1.,
                        1. , 0.,
                        3. , 1.,
                        6. , 2., ),
            PROL_GAUCHE='CONSTANT',
            PROL_DROITE='LINEAIRE',
                      )
```



4.2 Function and variables entered in the form of concepts listr8

It is possible to define this function using concepts of the listr8 type created via operator DEFI_LIST_REEL [U4.34.01]:

```
ABSCISSE = DEFI_LIST_REEL   ( DEBUT = 0. ,
                 INTERVALLE =  (_F (JUSQU_A = 1. , NOMBRE = 1,),
                                _F (JUSQU_A = 3. , NOMBRE = 1,),
                                _F (JUSQU_A = 6. , NOMBRE = 1,),)
                            )
ORDERED = DEFI_LIST_REEL    ( DEBUT = -1. ,
                 INTERVALLE =  (_F (JUSQU_A = 0. , NOMBRE = 1,),
                                _F (JUSQU_A = 1. , NOMBRE = 1,),
                                _F (JUSQU_A = 2. , NOMBRE = 1,),)
                            )
EX_2 = DEFI_FONCTION     ( NOM_PARA    = "INST",
                           VALE_PARA   = ABSCISSE,
                           VALE_FONC   = ORDERED,
                           PROL_DROITE = "CONSTANT",
                           PROL_GAUCHE = "LINEAIRE",
                         )
```

Note:

This example is obviously quite complicated to define the function suggested.
We wanted only to highlight the principle of use of the opportunity given.
This one becomes interesting when one uses functions defined in a large number of points.
Another reason to use the definition by DEFI_LIST_REEL is when the lists are necessary like argument for another operator: (list of times of an evolutionary computation THER_LINEAIRE, DYNA_LINE_TRAN, …), this avoids the duplication of information then.