Operator DEFI_FONCTION

1 Goal

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 depends on a variable of space or time.

The concept produced by this operator is of type `function`.

2 Syntax

```
F  [function]   =  DEFI_FONCTION (

                ♦  NOM_PARA  =  Np,

                ◊  NOM_RESU  =  /  'TOUTRESU',              [DEFECT]
                                /  NR,                      [K8]

                ♦  /  VALE          =  lv,                  [l_R]
                   /  VALE_C        =  lv,                  [l_C]
                   /  VALE_PARA     =  ,                    [listr8]
                      ♦  VALE_FONC     =  lo,               [listr8]
                   /  X-COORDINATE     =  labs,                [l_R]
                      ♦  ORDINATE      =  Lord,             [l_R]
                   /  NOEUD_PARA    =  lno,                 [l_noeud]
                      ♦  GRID       =  my,               [grid]
                      ♦  VALE_Y        =  ly,                  [l_R]

                ◊  PROL_DROITE  =  /  'CONSTANT',
                                   /  'LINEAR',
                                   /  'EXCLUDED',                [DEFECT]

                ◊  PROL_GAUCHE  =  /  'CONSTANT',
                                   /  'LINEAR',
                                   /  'EXCLUDED',                [DEFECT]

                ◊  Interpol  =  |  'FLAX',                    [DEFECT]
                                |  'LOG',

                ◊  INFORMATION  =     /  1,
   [DEFECT]
                                /  2,

                ◊  VERIF  =  |  'GROWING',                    [DEFECT]
                             |  'NOT',

                ◊  TITLE  =  Ti,                              [l_KN]

                             )
```

3 Operands

3.1 Operand NOM_PARA

♦ NOM_PARA = Np

Indicate the name of the parameter (variable or X-coordinate) of the function.
Possible values for Np are:

'ABSC':	Withbscisse curvilinear
'AMOR':	Damping
'DRX':	Rotation around axis X
'DRY MARTINI':	Rotation around the axis there
'DR.Z':	Rotation around axis Z
'DSP':	
'DX':	Displacement along axis X
'DY':	Displacement along axis X
'DZ':	Displacement along axis X
'ENDO':	Damage
'THICK':	Thickness
'EPSI':	Deformation
'FREQ':	Frequency
'HYDR':	Hydration
'INST':	Moment
'META':	
'NEUT1':	Neutral parameter 1
'NEUT2':	Neutral parameter 2
'NORM':	
'NUMÉRIQUE_ORDRE':	NRuméro of order
'PAD':	
'PCAP':	Capillary pressure
'PGAZ':	Gas pressure
'PLIQ':	Pressure of liquid
'PORO':	Porosity
'SWEATERS':	Pulsation
'PVAP':	Steam pressure
'SAT':	Saturation
'SECH':	Drying
'SIGM':	Constraint
'TEMP':	Temperature
'TSEC':	

'QUICKLY':	Speed
'X':	Coordinate X
'Y':	Coordinate there
'Z':	Coordinate Z
'XF':	Coordinate X finale (in the case of following loading)
'YF':	Coordinate there final (in the case of following loading)
'ZF':	Coordinate Z finale (in the case of following loading)

3.2 Operand NOM_RESU

◊ NOM_RESU = NR

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

Note:

Certain orders (CALC_FONCTION , DEFI_MATERIAU ...) check the coherence of the names of the parameter and result according to their context. For example, one expects a traction diagram 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 (X_1, there$_1$,…, X_N, there$_N$) with in the order:

- X_1, there$_1$ (the first value of the parameter and the corresponding value of the result),
- … ,
- X_N, there$_N$ (the last value of the parameter and the corresponding value of the result).

Note:

The list lv 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,…, X_N, there$_N$, Z_N) with:

- X_I values of the parameter
- … ,
- there$_I$, Z_I the real part and the imaginary part of the function complex for this parameter.

3.5 Operands X-COORDINATE / ORDINATE

/ X-COORDINATE = labs
/ ORDINATE = Lord

One provides the values of the X-coordinates and the ordinates of the function separately in the shape of lists of actual values (X_1, X_2,..., X_N) for X-COORDINATE and (there$_1$, there$_2$,..., there$_N$) for ORDINATE. The two lists must have the same cardinal.

3.6 Operand VALE_PARA / VALE_FONC

/ VALE_PARA =
/ VALE_FONC = lo

Even operation that X-COORDINATE, ORDINATE except that the lists are provided in the form of concept listr8 product by DEFI_LIST_REEL [U4.34.01].
VALE_PARA and VALE_FONC must be identical cardinals if not the order stops in error.

3.7 Operand NOEUD_PARA

```
/   NOEUD_PARA = lno
```

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

3.8 Operands PROL_DROITE and PROL_GAUCHE

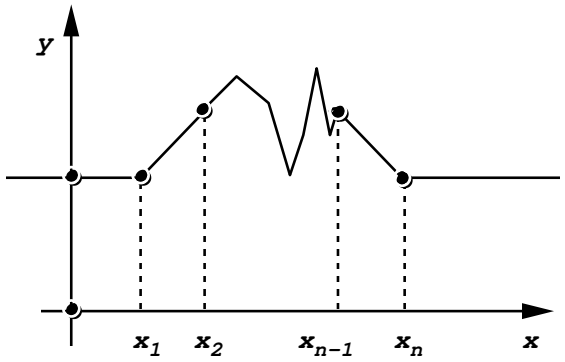
◊ PROL_DROITE and PROL_GAUCHE =

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

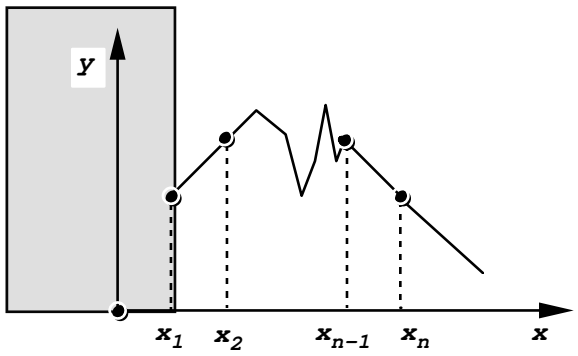
- 'CONSTANT' for a prolongation with the last (or first) value of the function,
- 'LINEAR' for a prolongation along the first definite segment (PROL_GAUCHE) or of the last definite segment (PROL_DROITE),
- 'EXCLUDED' if the extrapolation of the values apart from the field of definition of the parameter is prohibited (in this case if a calculation 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 = 'LINEAR', PROL_GAUCHE = 'EXCLUDED'



Note:

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

3.9 Operand Interpol

◊ Interpol =

Type 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 = ('FLAX', 'LOG')

'FLAX' linear,
:
'LOG' : logarithmic curve,

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 INFORMATION

◊ INFORMATION = Specify the options of impression on the file MESSAGE.

1: pas d' impression (option by default)

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

3.11 Operand VERIF

◊ VERIF =

The operator DEFI_FUNCTION check 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 while 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 TITLE

◊ TITLE = Ti

Title attached to the concept produced by this operator [U4.03.01].

3.13 Operands GRID and VALE_Y

These two keywords should be informed if one defines the function from NOEUD_PARA.

GRID = my

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

VALE_Y = lv

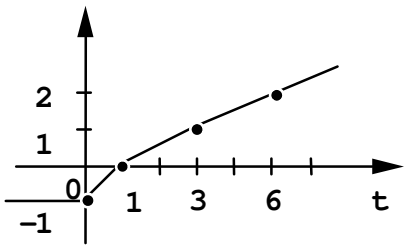
List of the values of the ordinates of the function to be defined.

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 type listr8 created via the operator DEFI_LIST_REEL [U4.34.01]:

```
X-COORDINATE = DEFI_LIST_REEL (  BEGINNING = 0. ,
                 INTERVAL =  (_F (JUSQU_A = 1. , NUMBER = 1,),
                                 _F (JUSQU_A = 3. , NUMBER = 1,),
                                 _F (JUSQU_A = 6. , NUMBER = 1,),)
                              )
ORDINATE = DEFI_LIST_REEL  (  BEGINNING = -1. ,
                 INTERVAL =    (_F (JUSQU_A = 0. , NUMBER = 1,),
                                 _F (JUSQU_A = 1. , NUMBER = 1,),
                                 _F (JUSQU_A = 2. , NUMBER = 1,),)
                              )
EX_2 = DEFI_FONCTION   (  NOM_PARA    = 'INST',
                          VALE_PARA   = X-COORDINATE,
                          VALE_FONC   = ORDINATE,
                          PROL_DROITE = 'CONSTANT',
                          PROL_GAUCHE = 'LINEAR',
                        )
```

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 the moments of an evolutionary calculation THER_LINEAIRE , DYNA_LINE_TRAN , …), this avoids the duplication of information then.