Operator RECU_FONCTION

1 Drank

To extract in the form of a function the evolution from a quantity according to another.

If the extraction is carried out from a data structure result, or of a field of variables cham_gd, or a resu_gene, the produced function corresponds to the temporal evolution of a component in a node or a Gauss point of the mesh.

Of a data structure tran_gene, one can also extract the evolution from two parameters in a node of shock.

Of an array, one can extract the evolution from 2 parameters in the columns of the array or a function contained in a box of the array.

Of a three-dimensions function, one can extract the function corresponding to a value given from the parameter.

From a data structure melasflu one can extract, the evolution of modal parameters according to the rate of flow of the fluid.

From a data structure interspectrum one can extract, the frequential evolution of the interspectrum associated with i - ième line j - the ième column with the interspectral matrix.

Product a data structure of type function or fonction_c.

In output of the command, the function is reordered by increasing X-coordinates. On the other hand, it is prohibited to have several identical X-coordinates (it would not be any more one function).

2 Syntax

```
Fr [function] =RECU_FONCTION         (
    ♦/RESULTAT       = resu,                                  /   [dyna_harmo]
                                                             /   [evol_elas]
                                                             /   [dyna_trans]
                                                             /   [evol_ther]
                                                             /   [evol_noli]
          # See extraction and localization of the /CHAM_GD
        field  = ch_gd,                                  /   [cham_no]
                                                          / [cham_elem]
          # See operands of localization of the /RESU_GENE
        field  = embarrassment,                              /   [tran_gene]
          # temporal Evolution of a physical component
            ♦/NOM_CHAM            =/"DEPL",
                                   /"QUICKLY",
                                   /"ACCE",
                                   /"PTEM",
                ♦NOM_CMP    = cmp,                    [K]
                /NOEUD     = No,                   [node]
                /GROUP_NO  = grno,                   [gr_noeud]
                /  NUME_CMP_GENE = val_n             [I]
                ◊/MULT_APPUI         =  ' OUI',
                   /   |  CORR_STAT =   ' OUI',
                       |  ACCE_MONO_APPUI =frap  ,    [function]
              /NOEUD_CHOC       = nd_choc,             [node]
              /GROUP_NO_CHOC       = no_choc,            [gr_noeud]
                ♦PARA_X=             nparax,            [kN]
                ♦PARA_Y=             nparay,            [kN]
                ◊INTITULE=           name,              [kN]
                ◊LISTE_PARA=         will li_para,          [listr8]
                ◊SOUS_STRUC=      nom_str,         [kN]
    /RESU_GENE     = embarrassment,                        /
[harm_gene]
    # frequential Evolution of a generalized or physical component
            ♦NOM_CHAM       = nomsymb,               [K16]
            ♦/NUMÉRIQUE_CMP_GENE        = numcmp,                  [K8]
              /NOM_CMP   = cmp,                      [K]
                ♦/NOEUD          = No,               [node]
                  /    GROUP_NO = grno,                [gr_noeud]
    /RESU_GENE     = embarrassment,                        /
[mode_gene]
    # frequential Evolution of a generalized or physical component
            ♦/NOM_PARA_RESU      = parameter,           [K8]
              /NOM_CHAM       = nomsymb,             [K16]
                ♦/      NUME_CMP_GENE = numcmp,       [K8]
                  /    NOM_CMP = cmp,                 [K]
                     ♦/NOEUD          = No,          [node]
                       /    GROUP_NO = grno,           [gr_noeud]
```

```
♦/SQUELETTE=          squ,          [squelette]
   /SOUS_STRUC        = sstru,  [K]
```

```
~    ARRAY =tabl ,                                  /  [array]
       ♦PARA_X=nparax                ,                  [kN]
       ♦PARA_Y=nparay                ,                  [kN]
       ◊NOM_PARA_TABL=/                "FONCTION",      [kN]
                                     /"FONCTION_C"
     ◊FILTRE    = _F (
              ♦NOM_PARA=                                 [kN]
              ◊CRIT_COMP=/          "EQ",          [DEFAULT]
                                  /  "LT",
                                  /  "GT",
                                  /  "NE",
                                  /  "LE",
                                  /  "GE",
                                 /   "VIDE",
                                 /   "NON_VIDE",
                                 /   "MAXI",
                                 /   "MAXI_ABS",
                                 /   "MINI",
                                 /   "MINI_ABS",
              ♦/VALE        =val_r   ,                   [R]
                  / VALE_I  =val_n   ,                   [I]
                 /VALE_C     =val_c  ,                   [C]
                  / VALE_K  =val_k   ,                   [kN]
              ◊   |  CRITERE =     "RELATIF",    [DEFAULT]
                                 /  "ABSOLU",
                 |  accuracy =/prec     ,
                                  /0.001 ,      [DEFAULT]
                 ),
  /BASE_ELAS_FLUI   = flui,                              [melasflu]
     ♦/TOUT_ORDRE        =    ' OUI',                     [DEFAULT]
       /NUME_ORDRE     =is ,                          [I]
     ♦NUMÉRIQUE_MODE     =im    ,                            [I]
     ♦PARA_X          =       ' VITE_FLU',                  [kN]
     ♦PARA_Y          =/  "FREQ",                    [kN]
                       /  "AMOR",

  /INTE_SPEC     = intespec,                             [interspectrum]
# frequential Evolution of a component of the interspectral matrix
     ◊NOM_CHAM       = nomsymb,                    [K16]
     ♦/♦NUMÉRIQUE_ORDRE_I            =numei ,                  [I]
          ◊NUMÉRIQUE_ORDRE_J     =numej ,                 [I]
        /♦NOEUD_I        = noei,                       [node]
          ♦NOM_CMP_I      = cmpi,                      [kN]
          ◊NOEUD_J       = noej,                       [node]
          ◊NOM_CMP_J      = cmpj,                     [KN]

  /NAPPE  = nap,                                        [three-dimensions
function]
     ♦VALE_PARA_FONC   = Np,                        [kN]
     ◊   |  CRITERE =     "RELATIF",                    [DEFAULT]
                     /   "ABSOLU",
        |  accuracy =/prec     ,
                        /0.001 ,                    [DEFAULT]

  # Operands of extraction of the field or the parameter if one
handles SD_resultat or RESU_GENE or a cham_gd
     ♦/NOM_CHAM             = nomsymb,            [K16]
       /NOM_PARA_RESU    = parameter,
     ◊/TOUT_ORDRE        =    ' OUI',                  [DEFAULT]
```

```
                /TOUT_INST       =   ' OUI',
                /NUME_ORDRE     =l_numeric,              [l_I]
                /LIST_ORDRE     =l_ord ,                 [listis]
                //INST          =l_inst,              [l_R]
                   /LIST_INST    =li_inst ,              [listr8]
                   /FREQ        =l_freq    ,            [l_R]
                   /LIST_FREQ    =li_freq ,             [listr8]
                   ◊   |  ACCURACY =/PREC     ,         [R]
                                       /1.0D-3   ,      [DEFAULT]
                       |  CRITERE   =  /  "RELATIF",    [DEFAULT]
                                       /  "ABSOLU",
                   ◊INTERP_NUMÉRIQUE     =    /  "NON",        [DEFAULT]
                                       /  "LIN",
```

Operands of localization of the field if one handles SD_resultat or a cham_gd

```
                ♦/NOEUD          = No,                   [node]
                   /GROUP_NO  = grno,                    [gr_noeud]
                   /♦/MAILLE=                  my,             [mesh]
                          /GROUP_MA     = grma,          [gr_maille]
                             ♦/NOEUD=             No,       [node]
                                /GROUP_NO     = grno,   [gr_noeud]
                                /POINT        = nupoint, [I]
                             ◊SOUS_POINT=     nusp,  [I]
          ♦NOM_CMP   = cmp,                              [K]
```

Overloads attributes of the function created

```
          ◊NOM_PARA  = nom_pa,                           [kN]
          ◊NOM_RESU  = nom_res,                          [kN]
          ◊INTERPOL  =     "NON",                        [kN]
                          /   |  ' LIN',
                              |  ' LOG',
          ◊PROL_DROITE  =         "CONSTANT",
                          /   "LINEAIRE",
                          /   "EXCLUDED",
          ◊PROL_GAUCHE  =         "CONSTANT",
                          /   "LINEAIRE",
                          /   "EXCLUDED",
   ◊TITER  =t ,                                          [l_K]
   ◊INFO   =/1        ,                                      [DEFAULT]
               /2 ,
                                )
```

If RESULTAT is one [dyna_harmo] alorsfr is [fonction_c],
If RESU_GENE is one [harm_gene] alorsfr is [fonction_c],
If INTE_SPECet NUME_ORDRE_J or NOEU_Jalorsfr is [fonction_c],
If NOM_PARA_TABL = "FONCTION_C" alorsfr is [fonction_c],
In the other cases, Fr is [function].

3 Operands

3.1 Operand RESULTAT

♦RESULTAT = resu

Name of the concept of the type result to which the extraction relates.

For the operands allowing to extract the field, to refer to [§3.7].

For the operands allowing to locate the field, to refer to [§3.8].

3.2 Operand CHAM_GD

♦CHAM_GD = ch_gd

Name of the concept of a field to which the extraction relates. For the operands allowing to locate the field, to refer to [§3.7].

The field provided to key word CHAM_GD is:
- that is to say a field at nodes of quantity: DEPL_R, TEMP_R or PRES_R;
- that is to say a field by elements (with the nodes or Gauss points) of quantity: VARI_R, EPSI_R, FLUX_R, or PRES_R.

3.3 Operand RESU_GENE

3.3.1 temporal Evolution of a physical or generalized component, standard tran_gene

♦ RESU_GENE = embarrassment

Name of the concept of the resu_gene type produces by DYNA_TRAN_MODAL [U4.53.21] on which carries the extraction.

The recovered function is expressed
- according to the physical variables: one specified GROUP_NO = grno or NOEUD = Noah .
- according to the generalized variables: one specified NUME_CMP_GENE = n_val.

NOM_CHAM gives the name of the field which one wants to recover ("DEPL", "QUICKLY", "ACCE", or "PTEM"). Option "PTEM" makes it possible to extract, for each time (or sequence number) of recovery, the values of time step of computation.

Note:

> *To make this restitution on physical base is role of the command REST_GENE_PHYS. In the later versions, this functionality will be withdrawn from RECU_FONCTION, it will be necessary to make the restitution then extract the function.*

3.3.1.1 Operands MULT_APPUI and ACCE_MONO_APPUI

◊MULT_APPUI

If this key word is "OUI', one restores the evolution of the variables in physical space by dealing with the problem moving absolute motion in the case of an excitation multi-bearing. In the contrary case, the restitution in physical space is done by supposing that with the problem is dealt moving relative motion. This key word is not usable if key word CORR_STAT is used.

◊ACCE_MONO_APPUI

In the case of an acceleration mono-bearing, one must indicate here the acceleration imposed on all the supports in the direction considered in order to calculate the absolute acceleration of the point.
If the key word is not indicated, one obtains relative acceleration as a result of the command.

Note:

The name of the concept must be same as that well informed under FONC_MULT of DYNA_TRAN_MODAL.

3.3.1.2 Operands CORR_STAT

◊CORR_STAT

If this key word is "OUI', the evolution of the variables in physical space is obtained by taking account of the correction due to the catch in consideration of static modes (cf [R4.05.03]). This key word is not usable if key word MULT_APPUI is used.

3.3.1.3 Information concerning the nodes of shock

♦ RESU_GENE = embarrassment

Concept of the tran_gene type containing for the various nodes of shock: local displacements, normal and tangential velocities and the normal and tangential shock forces.

♦ NOEUD_CHOC = nd_choc,
GROUP_NO_CHOC = no_choc,

Name of the node or the nodes group (which contains one node) shock where the function is recovered.

This node of shock is defined in command DYNA_TRAN_MODAL [U4.53.21].

♦PARA_X = nparax

Name of the parameter defining the X-coordinates (argument taken among the list: "INST", "FN", "FT1, "FT2", "DXLOC", "DYLOC", "DZLOC", "VN", "VT1", "VT2").

♦PARA_Y = nparay

Name of the parameter defining the Y-coordinates (argument taken among the list: "INST", "FN", "FT1, "FT2", "DXLOC", "DYLOC", "DZLOC", "VN", "VT1", "VT2").

◊LISTE_PARA = will li_para

List of the values of the parameter in X-coordinate defining the function.

◊INTITULE = name

This name defines the connection of shock (this name if it is used, is defined in command DYNA_TRAN_MODAL [U4.53.21]).

◊SOUS_STRUC = nom_str

During a computation in dynamic substructuring, name of the substructure which contains the node of shock (cf orders DEFI_MODELE_GENE [U4.65.02]). In this case key word INTITULE must be also indicated.

3.3.2 Frequential evolution D" a generalized or physical, standard component harm_gene

♦RESU_GENE = embarrassment

Name of the concept of the harm_gene type produces by DYNA_LINE_HARM [U4.53.11].

The recovered function is expressed
- according to the physical variables: one specified GROUP_NO = grno or NOEUD = Noah .
- according to the generalized variables: one specified NUME_CMP_GENE = n_val.

NOM_CHAM gives the name of the field which one wants to recover ("DEPL", "QUICKLY" or "ACCE").

Note:

> *To make this restitution on physical base is role of the command REST_GENE_PHYS . In the later versions, this functionality will be withdrawn from RECU_FONCTION , it will be necessary to make the restitution then extract the function.*

3.3.3 Frequential evolution D" a generalized or physical, standard component mode_gene

♦RESU_GENE = embarrassment

Name of the concept of the mode_gene type produces by MODE_ITER_SIMULT [U4.53.03] or MODE_ITER_INV [U4.53.04].
The recovered function is expressed with the physical variables if NOM_CMP is present, with the generalized variables if NUME_CMP_GENE is present.

♦NOM_PARA_RESU /NOM_CHAMP See paragraph 3.8.
♦NOM_CMP /NOEUD/GROUP_NOVoir paragraph 3.9.

♦SQUELETTENom of the mesh squelette of the total structure on which result will be restored: to see operator DEFI_SQUELETTE [U4.24.01].
to ♦SOUS_STRUCVoir above.

3.4 Operand COUNTS

One can recover:
1) either a function defined from two columns of the array,
2) or a function whose name is indicated in a box of the array.

♦ ARRAY = tabl Name of the array result in which one carries out an extraction.

3.4.1 Function defined from two columns of the array

3.4.1.1 Operands PARA_X / PARA_Y

♦PARA_X = nparax

Name of the column of the array defining the X-coordinates.

♦PARA_Y = nparay

Name of the column of the array defining the Y-coordinates.

3.4.2 Function whose name is indicated in a box of the array

3.4.2.1 Operand NOM_PARA_TABL

♦NOM_PARA_TABL= "FONCTION" or "FONCTION_C"

the presence of this key word indicates that one recovers a function whose name is stored in a box of the array. The real functions are stored in column "FONCTION", the complex functions in column "FONCTION_C".

3.4.2.2 Key word FILTRE

◊ FILTRE
the operands of extraction are different from those used for the preceding cases. To carry out the extraction, it is necessary to use key word FILTRE and the operands NOM_PARA, CRIT_COMP, VALE_X, CRITERE, accuracy.

This factor key word allows to filter the information stored in the array. For the use of this key word to see command IMPR_TABLE [U4.91.03].

To recover a function whose name is indicated in a box of the array, It is necessary to at least twice use factor key word the FILTRE to select only the useful box.

3.5 Operand BASE_ELAS_FLUI

One recovers in a data structure of the type melasflu produced by the operator CALC_FLUI_STRU [U4.66.02], the evolutions of the frequency or damping, for a given mode, according to the various velocities of excitation of the fluid.

♦BASE_ELAS_FLUI = flui

Concept of the melasflu type produces CALC_FLUI_STRU by the command.

3.5.1 Operands NUME_ORDRE / TOUT_ORDRE

♦/NUMÉRIQUE_ORDRE = is,
/TOUT_ORDRE = "OUI',

the evolution of the frequency or that of damping is given for all the velocities of the fluid (TOUT_ORDRE) or for some sequence numbers velocities of the fluid (NUME_ORDRE).

3.5.2 Operand NUME_MODE

♦NUMÉRIQUE_MODE = im

Number of the mode for which the extraction of the frequency or damping according to the velocity of the fluid is carried out.

3.5.3 Operands PARA_X / PARA_Y

♦PARA_X = "VITE_FLU"

In X-coordinate, the parameter is the velocity of excitation of the fluid, of name "VITE_FLU".

♦PARA_Y = / "FREQ",
/ "AMOR",

In Y-coordinate, one has the choice between the frequency (name of the parameter: "FREQ") or damping (name of parameter "AMOR").

3.6 Operand INTE_SPEC

One extracts in a data structure of type interspectrum, the frequential evolution of the interspectrum associated with the 2rd line j-ième column with the interspectral matrix intespec.

♦INTE_SPEC = intespec

3.6.1 Operand NOM_CHAM

◊NOM_CHAM = nomsymb

Symbolic name of the field to which the extraction relates.

3.6.2 Operands NUME_ORDRE_I, NUME_ORDRE_J

♦NUMÉRIQUE_ORDRE_I =numei
◊NUMÉRIQUE_ORDRE_J =numej

Indication of the couple of indices (line i , column j) to extract from the interspectral matrix intespec.

These operands are excluded with operands NOEUD_I NOM_CMP_I NOEUD_J NOM_CMP_J.

Note:

NUME_ORDRE_J is not informed if one wants to extract a diagonal term from the matrix.

3.6.3 Operands NOEUD_I, NOM_CMP_I, NOEUD_J, NOM_CMP_J

♦ NOEUD_I=noei
♦ NOM_CMP_I=cmpi
◊ NOEUD_J=noej
◊ NOM_CMP_J=cmpj

These operands correspond to the names of the nodes and of the components (line I, column J) of the matrix of the interspectral matrix intespec.

These operands are excluded with operands NUME_ORDRE_I NUME_ORDRE_J.

Note:

One does not inform NOEUD_J and NOM_CMP_J if one wants to extract a diagonal term from the matrix.

3.7 Operand THREE-DIMENSIONS FUNCTION

One recovers in a data structure of type three-dimensions function the function corresponding to a given value of the parameter of the three-dimensions function.

♦VALE_PARA_FONC = Np

np is the value of the parameter of the three-dimensions function for which one wishes to extract the function.
There is no interpolation on the parameter of the three-dimensions function. CRITERE and PRECISE DETAILS make it possible to provide np with a given accuracy.

3.8 Operands of extraction of the field or the parameter

3.8.1 Operand NOM_CHAM

♦ NOM_CHAM = nomsymb

Symbolic name of the field to which the extraction relates.

3.8.2 Operand NOM_PARA_RESU

♦NOM_PARA_RESU = parameter

Symbolic name of the parameter of the data structure which one wants to extract (for example: ETA_PILOTAGE, MASSE_EFFE_DX, MASSE_GENE …).

The extracted function will then have as a X-coordinate the variable of access (INST, FREQ…) and for Y-coordinate the value of parameter.

3.8.3 Operands TOUT_ORDRE/NUME_ORDRE/TOUT_INST/LIST_ORDRE

♦/TOUT_ORDRE = "OUI' (default value)

This key word indicates that one wants to extract for all the already calculated sequence numbers.

Example: all times for result of evol_* type.

/NUME_ORDRE = l_nume

the extraction will be done for the values of sequence number l_nume provided.

/TOUT_INST = "OUI'

This key word indicates that one wants to extract for all times.

/LIST_ORDRE = l_ord

This key word indicates that one wants to extract with the sequence numbers described in the concept l_ord of the listis type.

3.8.4 Operands INST/LIST_INST/FREQ/LIST_FREQ

◊/INST = l_inst

This key word indicates that one wants to extract at times l_inst.

/LIST_INST = li_inst

This key word indicates that one wants to extract at the times described in the concept li_inst of the listr8 type.

/ FREQ = l_freq

This key word indicates that one wants to extract with the frequencies l_freq.

/ LIST_FREQ = li_freq

This key word indicates that one wants to extract with the frequencies described in the concept li_freq of the listr8 type.

3.8.5 Operands accuracy/CRITERE

◊PRECISION = prec

This operand makes it possible to indicate that one searches the value of the field of which time or the frequency is in an interval defined by the absolute or relative position: "inst ± prec" (cf CRITERE).

By default prec = 1.0D-3

◊CRITERE =

"RELATIF" the interval of search is
[inst (1-PREC), inst (1+prec)]

"ABSOLU" the interval of search is
[INST-PREC, inst+prec]

3.8.6 Operand INTERP_NUME

◊ . INTER_NUMÉRIQUE

This key word defines the type of interpolation between two sequence numbers. It is valid only if the user defined a list of times or frequencies. It is possible to prohibit the interpolation "NON" or to admit a linear interpolation "LIN".
The interpolation cannot be used when one extracts the value from a parameter (key word NOM_PARA_RESU).

3.9 Operands of localization of the field

3.9.1 Operands NOEUD/GROUP_NO

♦ /NOEUD = No

Name of the node to which the extraction relates.

/GROUP_NO = grno

Name of the nodes group, containing 1 only node, to which the extraction relates.

3.9.2 Operands NETS/GROUP_MA/NOEUD/GROUP_NO/POINT

♦/MAILLE = my
/GROUP_MA = grma

Name of the mesh (ma) or name of a mesh group (grma), containing only one mesh, to which the extraction relates. These key words relate to only the cham_elem.

♦/NOEUD = No

Name of a node of the mesh to which the extraction (case of the cham_elem to the nodes) relates.

/GROUP_NO = grno

Indicates the name of the nodes group, container only one name of node, to which the extraction (case of the cham_elem to the nodes) relates.

/POINT = nupoint

the integer nupoint specifies the local number with the element of the Gauss point of which one wishes to obtain the value (case of the cham_elem to Gauss points).

◊SOUS_POINT = nusp

the integer nusp specifies the number of the subpoint of which one wishes to obtain the value (case of the cham_elem at subpoints, used by the structural elements: beam, pipes, shells).
In the case of the plates and of the multi-layer shells, the number of the subpoint corresponds to the level in all the layers. Each layer is described by a lower, average and higher skin. By convention, for N layers, this number varies between 1 and $3N$ where the first point is at the level of the skin lower of the first layer and $3N$ the ième point than the level of the higher skin of the last layer (cf [R3.07.03] and [R3.07.04] for the numeration of the layers).
In the case of the multifibre beams, this integer is the number of the fiber whose classification is described in documentation [U4.26.01] and [R3.08.08].
In the case of the pipes, it is necessary to refer to the description made in the document [R3.08.06].

3.9.3 Operand NOM_CMP

♦NOM_CMP = cmp

Name of the component of the quantity to which the extraction relates.

3.10 Attributes of the concept function created by RECU_FONCTION

3.10.1 Values by default

By default the attributes of the concept function created by the command RECU_FONCTION are:

Interpolation: "NON"
left Prolongation: "EXCLUDED"
right Prolongation: "EXCLUDED"
NOM_PARA : given in entry
NOM_RESU : given to as starter

3.10.2 Overload of the attributes

the user the attributes given by means of by default can overload the following key words:

3.10.2.1 Operand NOM_PARA

◊NOM_PARA = para

It indicates the name of the parameter (variable or X-coordinate) of the function. The values currently authorized for will lpara are:

/ "TEMP"	/ "INSTS"	/ "EPSI"
/ "X"	/ "Y"	/ "Z"
/ "FREQ"	/ "PULS"	/ "AMOR"
/ "DX"	/ "DY"	/ "DZ"
/ "DRX"	/ "DRY"	/ "DRZ"

plus those specific to the nodes of shock (cf [§ 3.3.2.2]).

3.10.2.2 Operand NOM_RESU

◊NOM_RESU = resu

It indicates the name of result, the function thus created is a function whose value is of name lresu (8 characters).

3.10.2.3 Standard operand

INTERPOL

◊INTERPOL of interpolation of the function enters the values of the parameter of the field of definition. Behind this key word one expects "a LIN", parameter list (two at the most) among "NON", "LOG". If only one value is given the interpolation will be identical for the X-coordinates and the ordered. If two values are given, the first corresponds to the interpolation of the X-coordinates and the second with the interpolation of the Y-coordinates.

3.10.2.4 Operands PROL_DROITE/ PROL_GAUCHE

◊PROL_DROITE and PROL_GAUCHE

They 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.

3.11 Operand TITER

◊TITER

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

3.12 Operand INFO

◊INFO

Specifies the options of printing on the message file .

1 step of printing (by default)

2 printing of the descriptor of the function and the list of the first 10 values of the function in the order ascending of the first 10 parameters

4 Extractions

4.1 Examples of function on the dynamic response of a network of pipework

```
tran_gen = DYNA_TRAN_MODAL (…)

l_inst = DEFI_LIST_REEL (debut = 0. ,
                         INTERVALLE = _F (JUSQU_A = 3. , NOT = 0.005))

dyn_tran = REST_GENE_PHYS (RESU_GENE = tran_gen, NOM_CHAM = "DEPL",
                           LIST_INST = l_inst,   INTERPOL = "LIN")

dyn_tran = CALC_CHAMP (…,
                       FORCED = "SIEF_ELGA"                    )


tab_rele = POST_RELEVE_T (ACTION=_F (INTITULE=   "sixx_254",
                                     CHEMIN= line,
                                     RESULTAT=   dyn_tran,
                                     NOM_CHAM=   "SIEF_ELGA",
                                     INST=       2.54,
                                     TOUT_CMP=   "OUI',
                                     OPERATION=  "EXTRACTION"  )  )
```

4.1.1 Evolution of the displacement of component node NO01 "DX" at all times of computation

```
f1 = RECU_FONCTION (RESU_GENE = tran_gen,   NOM_CHAM = "DEPL",
                    NOEUD    = "NO01"  ,   NOM_CMP = "DX"    )
```

4.1.2 Evolution of quantity "SIXX" on mesh MA01 with node NO01 at all times of computation

```
f2 = RECU_FONCTION (RESULTAT= dyn_tran, NOM_CHAM= "SIEF_ELGA",
                    =  "MA01 NETS"  , NOEUD   = "NO01", NOM_CMP=' SIXX')
```

4.1.3 Evolution of quantity "SIXX" along line of pipework at the time of computation 2.54 S

```
f3 = RECU_FONCTION (ARRAY = tab_rele,
                    PARA_X = "ABSC_CURV",  PARA_Y = "SIXX")
```

4.1.4 Evolution of quantity "SIXX" along line of pipework (curvilinear abscisse higher than 10) at the time of computation 2.54 S

```
f4 = RECU_FONCTION (ARRAY = tab_rele,
                    FILTRE = _F (NOM_PARA  = "ABSC_CURV",
                                 CRIT_COMP = "GE",
                                 VALE      = 10. ,  ),
                    PARA_X = "ABSC_CURV",  PARA_Y = "SIXX")
```

4.2 Extraction of function in a data structure melasflu

```
meles1   = CALC_FLUI_STRU (  …  )
```

```
f_freq   = RECU_FONCTION (  BASE_ELAS_FLUI = meles1,
                            PARA_X         = "VITE_FLU",
                            PARA_Y         = "FREQ",
                            TOUT_ORDRE     = "OUI',
                            NUME_MODE      = 2)
```

4.3 Extraction of `function` whose name is indicated in a box of an array of the type `table_fonction`

```
reppx_ac  = REST_SPEC_PHYS (…)

statx_ac  = POST_DYNA_ALEA (INTE_SPEC  = reppx_ac,
                            TOUT_ORDRE = "OUI',
                            OPTION     = "DIAG"     )

f_freq   = RECU_FONCTION (  ARRAY    = statx_ac,
                            NOM_PARA_TABL = "FONCTION_C",
                            FILTRE   = ( _F (NOM_PARA = "NOEUD_I",
                                             VALE_K   = "N_TUB_01"),
                                         _F (NOM_PARA = "NOEUD_J",
                                             VALE_K   = "N_TUB_01"),
                                         _F (NOM_PARA = "NUME_VITE_FLUI",
                                             VALE_I   = 3          )
                                       )
                          )
```