Opérateur AFFE_MODELE

1 Drank

Définir the modelized physical phenomenon (mechanical, thermal or acoustic) and the type of finite elements.

This operator allows to affect modelizations on whole or part of the mesh, which defines:

- the degrees of freedom on the nodes (and the equation or the associated conservation equations),
- the types of finite elements on the meshes,

Les possibilities of the finite elements which can be selected are described in the booklets [U3].

The types of meshes are described in the document “Description of the mesh file of Code_Aster” [U3.01.00].

This operator also allows to define a distribution of the finite elements in order to parallel elementary computations and the assemblies.

Product data structure of a model `type`.

2 Syntax

```
Mo   [model] = AFFE_MODELE      (

       ♦  |  MAILLAGE=ma      ,                          /  [mesh]
                                                      /  [skeleton]
          |  GRILLE=grile        ,                          [grid]
       ♦  |  AFFE=_F          (
                           ♦  /  TOUT=        'OUI',
                              /MAILLE         =mail  ,
[l_maille]
                              /NOEUD       =noeu  ,            [l_noeud ]
                              /GROUP_MA    =g_mail   ,         [l_gr_maille]
                              /GROUP_NO    =g_noeu   ,         [l_gr_noeud
]
                           ♦  /  ♦PHENOMENE    =     'MECANIQUE',
                                 ♦MODELISATION =… (see [§3.2.1])

                              /  ♦PHENOMENE    =     'THERMAL'
                                 ♦MODELISATION =… (see [§3.2.1])

                              /  ♦PHENOMENE   :      “ACOUSTIC”,
                                 ♦MODELISATION =… (see [§3.2.1])

                      ),
          |  AFFE_SOUS_STRUC = _F (
                         ♦  /  TOUT=        'OUI',
                            /SUPER_MAILLE    =l_mail   ,          [l_maille]

                                  )

       ◊VERIF=          |  'NET'
                     |  'NODE',
       ◊VERI_JACOBIEN=        /  “OUI”                            [DEFECT]
                            /  “NON”
       ◊GRANDEUR_CARA=_F             (
                will ◊LONGUEUR=lcara            ,
[R]
                will ◊PRESSION=pcara            ,
[R]
                will ◊TEMPERATURE=tcara       ,                       [R]

       ◊PARTITION=_F           (
             ◊PARALLELISME  =
                       /“GROUP_ELEM”                            [DEFECT]
                       /“MAIL_CONTIGU”
                             ◊ CHARGE_PROC0_MA = / 100          [DEFAUT]
                                               / pct
                       /“MAIL_DISPERSE”
                             ◊ CHARGE_PROC0_MA = / 100          [DEFAUT]
                                               / pct
                       /“SOUS_DOMAINE”
                             ♦ PARTITION       =  share         [sd_feti]
                             ◊ CHARGE_PROC0_SD = / 0            [DEFAUT]
                                               / nbsd
                       /“CENTRALISE”
```

```
                    )
    ◊INFO=      /  1                         [DEFAUT]
                /2 ,

                      )
```

3 Operands

3.1 Opérande MESH

♦MAILLAGE = my

Nom of the associated mesh on which one affects the elements.

Note:

For the axisymmetric modelizations, the axis of revolution is the axis Y of the mesh. All the structure must be with a grid in $X \geq 0$.

3.2 Operand ROASTS

GRID = Nom

grid of the associated grid on which one affects the elements. The grid must be defined by operator DEFI_GRILLE (see U4.24.02).

3.3 Key word AFFE

♦ | AFFE

Définit entities of the mesh and types of elements which will be affected for them. For each occurrence, one can introduce a list of modelizations. The rule of overload applies between the various modelizations, from left to right.

For example:

```
AFFE=_F (TOUT='OUI', PHENOMENE='MECANIQUE',
                        MODELISATION= ("AXIS", "AXIS_SI"),)
```

Les various modelizations "overload" the ones the others: AXIS_SI overloads AXIS on the meshes where AXIS_SI exists (mesh QUAD4 and QUAD8).

Note:

The code stops in <F> error if the modelizations of the list are not very the same "dimension" (for example MODELISATION= ("3D", "D_PLAN")). Moreover, for an occurrence of AFFE, the specified meshes whose dimension is that of the dimension of the modelization must be all affected. If not the code emits a <A>larme. This alarm protects the user who uses modelizations "with holes". If for example, it uses only modelization AXIS_SI on a mesh containing only TRIA6.

The entities of the mesh are specified by the operands:

Operands	Contained / meaning
TOUT	Affectation with the totality of meshes (but not the nodes!)
GROUP_MA	Affectation with a list of mesh groups
GROUP_NO	Affectation to a list of nodes groups (see remark)
NETS	Affectation with a list of meshes
Affectation	NODE to a list of nodes (see remark)

Remarque:

The use of elements being based only on nodes does not make it possible to affect materials via AFFE_MATERIAU . So these elements are usable neither in STAT_NON_LINE [U4.51.03] nor in DYNA_NON_LINE [U4.53.01]. In this case, meshes should be created as a preliminary. POI1 using the word - key CREA_POI1 of CREA_MAILLAGE [U4.23.02].

The use of such elements is thus reserved for linear computations, on discrete elements, of which all the characteristics are affected by `AFFE_CARA_ELEM`.

The type of element is specified by the operands:

Operands	Contained / physical
`meaning`	PHENOMENE Phénomène modelized (associated conservation equation)
`MODELISATION`	Type of interpolation or discretization

3.3.1 Opérandes PHENOMENE and compulsory

```
MODELISATION
♦PHENOMENE
```

♦MODELISATION Sont for each occurrence of the key word factor `AFFE`. This couple of key words defines in a bijective way the type of affected element in a kind of mesh. The possible modelizations are indicated below by listing them by “packages”:

ACOUSTIC

`ACOUSTICS 2D` ACOUSTIC
 `continuums`

`PLANU3.33.01 3D` 3DU3.33.01
 `continuums`

THERMAL

`THERMAL 2D` THERMAL
 `shell`
 `COQUE_AXISU3.22.01`

`COQUE_PLANU3.22.01 2D` THERMAL
 `continuums`
 `AXIS_DIAGU3.23.01`
 `AXIS_FOURIERU3.23.02`
 `AXISU3.23.01`
 `PLAN_DIAGU3.23.01`

`PLANU3.23.01 3D` THERMAL
 `shell`

`COQUEU3.22.01 3D` continuums
 `3D_DIAGU3.24.01`
 `3DU3.24.01`

MECANIQUE 2D

`MECANIQUE 2D` discrete elements
 `2D_DIS_TR`
 `2D_DIS_T`

`MECANIQUE 2D` fluid-structure
 `2D_FLUIDEU3.13.03`
 `2D_FLUI_ABSOU3.13.13`
 `2D_FLUI_PESAU3.14.02`
 `2D_FLUI_STRUU3.13.03`
 `AXIS_FLUIDEU3.13.03`

AXIS_FLUI_STRUU3.13.03
D_PLAN_ABSOU3.13.12

MECANIQUE 2D continuums
AXISU3.13.01
AXIS_FOURIERU3.13.02
AXIS_SIU3.13.05
C_PLAN_SIU3.13.05
C_PLANU3.13.01
D_PLAN_SIU3.13.05
D_PLANU3.13.01

MECANIQUE 2D quasi incompressible
AXIS_INCOU3.13.07
D_PLAN_INCOU3.13.07
AXIS_INCO_UPR3.06.08
D_PLAN_INCO_UPR3.06.08
AXIS_INCO_OSGSR3.06.08
D_PLAN_INCO_OSGSR3.06.08
AXIS_INCO_GDR3.06.08
D_PLAN_INCO_GDR3.06.08

MECANIQUE 2D nonlocal
C_PLAN_GRAD_EPSIU3.13.06
D_PLAN_GRAD_EPSIU3.13.06
D_PLAN_GRAD_VARI
D_PLAN_GVNOR5.04.04
AXIS_GVNOR5.04.04
D_PLAN_GRAD_SIGMR5.03.24

MECANIQUE 2D plates and Mechanical
shells
COQUE_AXISU3.12.02
COQUE_C_PLANU3.12.02

COQUE_D_PLANU3.12.02 2D elements joined for the Mechanical crack propagation

PLAN_JOINTU3.13.14
AXIS_JOINTU3.13.14
PLAN_INTERFACER3.06.13
PLAN_INTERFACE_SR3.06.13
AXIS_INTERFACER3.06.13

AXIS_INTERFACE_SR3.06.13 2D elements to internal discontinuities for the priming and the crack propagation
PLAN_ELDIU3.13.14
AXIS_ELDIU3.13.14

MECANIQUE 2D thermo-hydro-mechanics
AXIS_HH2MD
AXIS_HH2MS
AXIS_HHMD
AXIS_HHMS
AXIS_HHMU3.13.08
AXIS_HMDU3.13.08
AXIS_HMS
AXIS_HM
AXIS_THH2D
AXIS_THH2S

AXIS_THH2MD
AXIS_THH2MS
AXIS_THHD
AXIS_THHS
AXIS_THHMD
AXIS_THHMS
AXIS_THMD
AXIS_THMS
AXIS_THMU3.13.08
AXIS _HHDR5.04.03
AXIS _HHSR5.04.03
AXIS _HH2DR5.04.03
AXIS _HH2SR5.04.03

D_PLAN_HH2MD
D_PLAN_HH2MS
D_PLAN_HHMD
D_PLAN_HHMS
D_PLAN_HHMU3.13.08
D_PLAN_HMD
D_PLAN_HMS
D_PLAN_HMU3.13.08
D_PLAN_HM_PU3.13.08
D_PLAN_THH2D
D_PLAN_THH2S
D_PLAN_THH2MD
D_PLAN_THH2MS
D_PLAN_THHD
D_PLAN_THHS
D_PLAN_THHMD
D_PLAN_THHMS
D_PLAN_THMD
D_PLAN_THMS
D_PLAN_THMU3.13.08
D_PLAN_HHDR5.04.03
D_ PLAN_HHS5.04.03
D_ hydraulic PLAN_HSR5.04.03
D_PLAN_HH2DR5.04.03
D_PLAN_HH2SR5.04.03
D_PLAN_2DGR5.04.03

D_PLAN_DILR5.04.03 MECANIQUE 2D unsaturated with finished volumes
D_PLAN_HH2SUC
D_PLAN_HH2SUDA
D_PLAN_HH2SUDM

MECANIQUE 2D elements joined with hydraulic coupling
AXIS_JHMS
PLAN_JHMS

Pour meshes 2D, makes it possible to inform the mesh groups or the meshes likely to be crossed by crack when the contact is defined on the lips of crack. Are allowed the following types of meshes: the QUAD8 and TRIA6 and edge the meshes of these elements, are the SEG3. If the meshes are linear, they should as a preliminary be transformed into quadratic meshes (with LINE_QUAD of operator CREA_MAILLAGE).

MECANIQUE 3D
MECANIQUE 3D bars and cables

2D_BARRE
BARREU3.11.01
CABLE_POULIEU3.11.03
CABLEU3.11.03

MECANIQUE 3D discrete elements
DIS_TRU3.11.02
DIS_TU3.11.02

MECANIQUE 3D fluid-structure
3D_FAISCEAU
3D_FLUIDEU3.14.02

MECANIQUE 3D absorbing border
3D_ABSOU3.14.09
3D_FLUI_ABSOU3.14.10

MECANIQUE 3D grids of concrete reinforcements
GRILLE_MEMBRANE
GRILLE_EXCENTREU3.12.04

MECANIQUE 3D continuums
3D_SIU3.14.01
3DU3.14.01

MECANIQUE 3D not room
3D_GRAD_EPSIU3.14.11
3D_GRAD_VARI
3D_GVNOR5.04.04

MECANIQUE 3D plates, shells and membranes
COQUE_3DU3.12.03
DKTU3.12.01
DSTU3.12.01
Q4GU3.12.01
DKTGU3.12.01
Q4GGU3.12.01
MEMBRANEU3.12.04

MECANIQUE 3D beams
FLUI_STRU U3.14.02
POU_C_T U3.11.01
POU_D_EM quasi
incompressible
U3.11.07
POU_D_EU3.11.01
POU_D_TGMU3.11.04
POU_D_TGU3.11.04

POU_D_T_GDU3.11.05 POU_D_TU3.11.01 MECANIQUE 3D
3D_INCOU3.14.06
3D_INCO_UPR3.06.08
3D_INCO_OSGSR3.06.08
3D_INCO_GDR3.06.08

MECANIQUE 3D thermo-hydro-mechanics
3D_HHMD
3D_HHMU3.14.07
3D_HMD

3D_HMU3.14.07
3D_THHD
3D_THHMD
3D_THHMU3.14.07
3D_THMD
3D_THMU3.14.07
3D_THVD
3D_THH2MD
3D_THH2M
3D_HH2MD
3D_HH2MS
3D_THH2S
3D_THH2D
3D_HHDR5.04.03
3D_HHSR5.04.03
3D_HSR5.04.03
3D_HH2DR5.04.03
3D_HH2SR5.04.03

MECANIQUE 3D hydraulics unsaturated with finished volumes
3D_HH2SUC
3D_HH2SUDA
3D_HH2SUDM

MECANIQUE 3D pipes
TUYAU_3MU3.11.06
TUYAU_6MU3.11.06

MECANIQUE 3D massive shell element
SHBU3.12.05

Pour meshes 3D, makes it possible to inform the mesh groups or the meshes likely to be crossed by crack when the contact is defined on the lips of crack. Are allowed the following types of meshes: HEXA20, PENTA15, TETRA10, and the edge meshes of these elements, are the QUAD8 and TRIA6. If the meshes are linear, they should as a preliminary be transformed into quadratic meshes (with LINE_QUAD of operator CREA_MAILLAGE).

Mechanics 3D elements joined for the crack propagation
3D_JOINTU3.13.14
3D_INTERFACER3.06.13
3D_INTERFACE_SR3.06.13

3.4 Key word AFFE_SOUS_STRUC

♦ | AFFE_SOUS_STRUC

is usable only for one using model of static substructures [U1.01.04].

♦ / SUPER_MAILLE = l_mail

l_mail is the list of the super-meshes which one wants to affect in the model. As for the finite elements, it is not compulsory to affect all the meshes of the mesh. It is AFFE_MODELE which confirms which are the substructures which will be used in the model. The difference with the conventional finite elements is that on the super-meshes, one chooses neither the MODELISATION nor the PHENOMENE because the macro-element (built by operator MACR_ELEM_STAT [U4.62.01]) which will be affected on the super-mesh has its own modelization and its own phenomenon (those which were used to calculate it).

/TOUT = "OUI"

Toutes them (super) meshes are affected.

3.5 Operand VERIF

◊VERIF

Valeur	Contenu / meaning
"MESH"	checks the assignment with all the required meshes if not error
"NODE"	checks the assignment with all the required nodes if not downward bias

: no checking is carried out.

3.6 Operand VERI_JACOBIEN

◊VERI_JACOBIEN = "OUI" / "NON"

This key word is used to check that the meshes of the model are not distorted too much. One computes the jacobian of the geometrical transformation which transforms the element of reference into each real mesh of the model. So on the various points of integration of a mesh, the jacobian changes sign, it is that this mesh is very "badly rotten".
An alarm (CALCULEL_7) is then emitted.

3.7 Operand GRANDEUR_CARA

◊GRANDEUR_CARA = _F (LONGUEUR = will lcara,…)

This key word is used to define some physical quantities characteristic of the dealt with problem. These quantities are currently used "have-to dimension" certain terms of the estimators of error in "HM ". See [R4.10.05].

3.8 Key word PARTITION

◊PARTITION

This key word makes it possible to distribute the finite elements of the model for the parallelism of elementary computations, the assemblies and certain linear solvers. *Cf* [U2.08.06] "Notice of use of parallelism".

It defines how (or not) the meshes will be distributed/elements for the paralleled phases of *Code_Aster*. The user thus has the possibility of controlling this distribution between the processors.

Parallelism operates:

- on elementary computations and the assemblies of matrixes and vectors (it is what the key word factor PARTITION makes it possible to control),
- with the resolution of the linear system if the solver is paralleled (*cf* [U4.50.01]).

Note:

It is possible to modify the mode of distribution during its study. It is enough to use command MODI_MODELE [U4.41.02].

3.8.1 Operand PARALLELISME

3.8.1.1 PARALLELISME = / "CENTRALISE"

parallelism starts only on the level of the linear solver. Each processor builds and provides to the solver the entirety of the system to be solved. Elementary computations are not paralleled.

3.8.1.2 PARALLELISME = / "GROUP_ELEM" [DEFAUT]

It is the mode of distribution chosen by defect. It allows a perfect load balancing *a priori*, i.e. each processor will carry out, for a kind of element given, the same number of elementary computations (with near). Obviously that does not prejudge of anything the final load balancing in particular in nonlinear computations where the cost of an elementary computation depends on other parameters but the type of element.
In this mode, the elements of the model are gathered by "group" in order to mutualiser certain computations what makes it possible to gain in effectiveness. The number of elements by group can be selected in command DEBUT [U4.11.01].
In addition, it is a question of the only mode able of distributing elementary computations induced by the late elements, i.e. by the loadings such as the dualized boundary conditions or the continuous contact.

3.8.1.3 PARALLELISME = / "MAIL_DISPERSE"

the distribution takes place on the meshes. They are distributed equitably on the various processors available. The meshes are distributed on the various processors as it is made it when one distributes cards to several gamers. One also speaks about "cyclic" distribution.

For example, with a model comprising 8 meshes, carried out on 4 processors, one obtains the following distribution:

Mode of distribution	Nets 1	Mesh 2	Maille 3	Mesh 4	Maille 5	Mesh 6	Maille 7	Mesh 8
MAIL_DISPERSE	Proc. 0	Proc. 1	Proc. 2	Proc. 3	Proc. 0	Proc. 1	Proc. 2	Proc. 3

It is seen that with this mode of distribution, a processor will treat meshes regularly spaced in the order of the meshes of the mesh. The advantage of this distribution is that "statistically", each processor will treat as many hexahedrons, of pentahedrons,…, and of triangles.
The workload for elementary computations in general will be well distributed. On the other hand, the matrix assembled on a processor "will be very dispersed", contrary to what occurs for mode "MAIL_CONTIGU".

3.8.1.4 PARALLELISME = /"MAIL_CONTIGU"

the distribution takes place on the meshes. They are divided into packages of contiguous meshes on the various processors available.

For example, with a model comprising 8 meshes, a machine of 4 processors available, one obtains the following distribution:

Mode of distribution	Nets 1	Mesh 2	Maille 3	Mesh 4	Maille 5	Mesh 6	Maille 7	Mesh 8
MAIL_CONTIGU	Proc. 0	Proc. 0	Proc. 1	Proc. 1	Proc. 2	Proc. 2	Proc. 3	Proc. 3

For this mode of distribution, the workload for elementary computations can be less balanced. For example, a processor can have to treat only "easy" edge meshes. On the other hand, the matrix assembled on a processor is in general more compact.

3.8.1.5 Key word CHARGE_PROC0_MA

```
◊ CHARGE_PROC0_MA = / 100 [DEFAUT]
                    / pct
```

This key word is accessible only for the modes from parallelism "MAIL_DISPERSE" and "MAIL_CONTIGU". Indeed these modes of distribution do not distribute in general equitably the load of computations because of dualized boundary conditions whose elementary computations are treated by processor 0.

If one wishes to relieve processor 0 (or on the contrary overload it), one can use key word CHARGE_PROC0_MA. This key word makes it possible to the user to choose the percentage of load which one wishes to assign to processor 0.

For example, if the user chooses CHARGE_PROC0_MA = 80, processor 0 will treat 20% of elements of less than the other processors, is 80% of the load which it should support if the sharing were equitable between the processors.

3.8.1.6 PARALLELISME = / "SOUS_DOMAINE"

the distribution of the meshes is based on a decomposition in subdomains built upstream via operator DEFI_PART_FETI.

```
♦ PARTITION       =  share  [feti]
◊ CHARGE_PROC0_SD = / 0  [DEFAUT]
                    / nbsd
```

key word PARTITION receives the product concept by DEFI_PART_FETI which describes partitioning in subdomains.

Key word CHARGE_PROC0_SD makes it possible to allot the number of subdomains for processor 0 (main processor). If CHARGE_PROC0_SD = 1 , then processor 0 will deal with one subdomain.

For example, with a data structure SD_FETI comprising 5 subdomains and a machine having 2 processors, and CHARGE_PROC0_SD = 2 , one obtains the following distribution:

Mode of distribution	Under-DOM. 1	Under-DOM. 2	Under-DOM. 3	Under-DOM. 4	Under-DOM. 5
SOUS_DOMAINE	Proc. 0	Proc. 0	Proc. 1	Proc. 1	Proc. 1

4 Production run

À partir des key words PHENOMENE and MODELISATION, one creates a data structure specifying the type of element attached to each mesh. There are possibly additional creations of meshes of type POI1 when assignments are made on nodes or nodes groups. This meshes are not accessible to the user. This is why it is strongly advised to use CREA_MAILLAGE [U4.23.02] to create meshes POI1 usable in the command file (for STAT_NON_LINE for example).

A brief recall of the assignments is systematically printed (INFO=1) in the file message.

For example:

```
SUR LES          612 MESHES OF the MESH MY
   ONE A DEMANDE the AFFECTATION OF          612
   ONE A PU IN AFFECTER                    612

   MODELISATION      ELEMENT FINI      TYPE NETS          NOMBRE
   3D                MECA_TETRA4       TETRA4                   52
   3D                MECA_PENTA6       PENTA6                   16
…
   3D                MECA_FACE3        TRIA3                    60
```

5 Exemple

```
   mo= AFFE_MODELE      ( MESH =ma  ,
                          VERIF =        ( “MESH”, “NODE”),
                          AFFE  = ( _F   (        GROUP_MA=gma  ,
                                           PHENOMENE=          'MECANIQUE',
                                           MODELISATION=    '3D'   ),
                                    _F   (        GROUP_NO=gno  ,
                                           PHENOMENE=          'MECANIQUE',
                                           MODELISATION=    'DIS_T'   ),
                      ))
```

For a modelization of phenomenon “MECANIQUE”, one affects:

- on the gma mesh group of isoparametric elements 3D,
- on the gno nodes group of the discrete elements to 3 degrees of freedom of translation.