Operator TO SOLVE

1 Goal

Solving a system of equations linear (direct or iterative method)

the methods of resolutions established in *Code_Aster* and applicable by this command are:

1) the method MULT_FRONT (direct method),
2) method LDLT (direct method),
3) the method MUMPS (direct method),
4) the method PCG (iterative method),
5) method PETSC (iterative method).

The effective choice of the method is done through command NUME_DDL [U4.61.11].

For the direct methods, the matrix must be beforehand factorized by the command TO FACTORIZE [U4.55.01]. In the case of the iterative methods with prepacking, the matrix of pre conditioning is provided by the operator TO FACTORIZE it-also [U4.55.01].

The operator allows complex resolutions for the “direct” methods (not for the iterative methods).

Product a data structure of the cham_no type.

2 Syntax

```
U   [cham_no_*]  =RESOUDRE
                 (  ◊reuse=U          ,
                    ♦MATR=A           ,
                    #Si  method LDLT, MULT_FRONT, MUMPS:
                                                 /   [matr_asse_DEPL_R]
                                                 /   [matr_asse_DEPL_C]
                                                 /   [matr_asse_TEMP_R]
                                                 /   [matr_asse_TEMP_C]
                                                 /   [matr_asse_PRES_R]
                                                 /   [matr_asse_PRES_C]
                    #Si  method PCG or PETSC:
                                                 /   [matr_asse_DEPL_R]
                                                 /   [matr_asse_TEMP_R]
                                                 /   [matr_asse_PRES_R]

                    ♦CHAM_NO=B           ,          /          [cham_no]
                    ◊CHAM_CINE=vcine        ,       /          [cham_no]

                    #si  method PETSC :
                    ◊ALGORITHME=/           "GMRES",       [DEFAULT]
                                         /  "CG",
                                         /  "CR",
                                         /  "GCR",

                    #si  method MUMPS, PCG, PETSC :
                    ◊RESI_RELA=/1.e-6              ,              [DEFAULT]
                                      /eps      ,              [R]

                    #si  method PCG or PETSC :
                    ◊♦MATR_PREC=precond            ,   /      [matr_asse_DEPL_R]
                                                   /      [matr_asse_TEMP_R]
                                                   /      [matr_asse_PRES_R]
                       ◊NMAX_ITER=/niter                ,          [I]
                                          /0  ,              [DEFAULT]

                    ◊TITER=titr               ,                [l_K80]
                    ◊INFO=/1                  ,                [DEFAULT]
                                 /2     ,
                 )
```

If CHAM_NO: [cham_no_DEPL_R] then (*) →DEPL_R
[cham_no_TEMP_R] →TEMP_R
[cham_no_PRES_C] →PRES_C

3 General information

This command makes it possible to solve:

- by a direct method, the linear system $\mathbf{AX=B}$, where $\mathbf{A}$ is a matrix beforehand "factorized" by the command TO FACTORIZE [U4.51.01.],
- by an iterative method (PCG or PETSC), the linear system $\mathbf{P^{-1}AX=P^{-1}B}$, where $\mathbf{P^{-1}}$ is a matrix of prepacking determined by the command TO FACTORIZE [U4.51.01] and $\mathbf{A}$ stamps it assembled initial.

The resolution is possible for boundary conditions of Dirichlet (kinematical boundary conditions) dualized or eliminated [U2.01.02]. In this last case, if the loading $\mathbf{X=X_0}$ on "edge" Γ_0 is applied with a kinematical load (operator AFFE_CHAR_CINE [U4.44.03]) taken into account in the assembled matrix (operator ASSE_MATRICE [U4.61.22]), the "value" of this loading $\left(\mathbf{X_0}\right)$, calculated by the operator CALC_CHAR_CINE [U4.61.03] must be provided by key word CHAM_CINE.

4 Operands

4.1 Operand MATR

♦MATR=A ,

Name of the matrix assembled of the system to solve:

- For the direct methods, one provides to MATR the concept modified by the operator TO FACTORIZE ; this matrix can be real or complex, symmetric or not.
- For the iterative methods, one provides to MATR the initial assembled matrix. The matrix of prepacking is with being provided MATR_PREC with the key word.

4.2 Operand CHAM_NO

♦CHAM_NO=B ,

Name of the second member vector (in general obtained by the command ASSE_VECTEUR).

4.3 Operand CHAM_CINE

◊CHAM_CINE =vcine ,

Name of the vector representing the "value" of the boundary conditions of Dirichlet eliminated (i.e. applied with one of commands AFFE_CHAR_CINE or AFFE_CHAR_CINE_F).

This cham_no comes from the execution of operator CALC_CHAR_CINE on the list of the char_cine (kinematical loadings) associated with the matrix assembled A [U2.01.02].

4.4 Operand ALGORITHME

◊ ALGORITHME =/"GMRES" [DEFAULT]
/"CG"
/"CR"
/"GCR"

This key word is used to choose the algorithm of iterative method PETSC. The various algorithms available are documented in the key word solver [U4.50.01].

4.5 Operand MATR_PREC

◊MATR_PREC=precond

Matrix of prepacking, obtained by the operator TO FACTORIZE [U4.55.01].
Prepacking is necessary in the iterative methods to obtain a good convergence in a minimum of iterations.
With the method PCG, the matrix of prepacking is a matrix distinct from the matrix of the problem (key word MATR).
On the other hand, with method PETSC, it is advised to use the same matrix for MATR_PREC and MATR, which wants to say that the command TO FACTORIZE must be made “places from there” (with the key word reuse). See example below.

4.6 Operand RESI_RELA

```
◊RESI_RELA     = /1.e-6 ,              [DEFAULT]
                 /eps      ,              [R]
```

This key word is described in [U4.50.01]

For the iterative methods PCG and PETSC, it acts of the convergence criterion of the algorithm.
For the method MUMPS , this key word makes it possible to check the quality of the solution .

4.7 Maximum operand

NMAX_ITER

◊NMAX_ITER=niter Nombre of iterations of the iterative algorithm.

If niter = the 0 then algorithm chooses a by default nombre of iterations.

4.8 Operand TITER

◊TITER=titr ,

Title which one wants to give to result produced [U4.03.01].

4.9 Operand INFO

◊INFO=

1 : no printing.
2 : printings

5 Resolution

5.1 Examples by direct method MULT_FRONT

- Constitution of the assembled matrixes:

  One calculated before elementary terms KEL, FEL.

```
NU=NUMÉRIQUE_DDL  (MATR_RIGI=KEL, METHODE=' MULT_FRONT')
K=ASSE_MATRICE    (MATR_ELEM=KEL,  NUMÉRIQUE_DDL=NU,)
F=ASSE_VECTEUR    (MATR_ELEM=FEL,  NUMÉRIQUE_DDL=NU,)
```

- Factorization:

```
K=FACTORISER   (reuse=K,     MATR_ASSE=K,)
```

- Resolution:

```
U=RESOUDRE     (MATR=K,  CHAM_NO=F,)
```

- for the use of the kinematical loads (with elimination of the imposed degrees of freedom), to see the example set in command AFFE_CHAR_CINE [U4.44.03].

5.2 Resolution by the method MUMPS

```
NU=NUMÉRIQUE_DDL        (MATR_RIGI= KEL, METHODE= "MUMPS", RENUM=' METIS')

K=ASSE_MATRICE             ( MATR_ELEM= KEL, NUME_DDL= NU)
F=ASSE_VECTEUR             ( VECT_ELEM= FEL, NUME_DDL= NU)
K=FACTORISER            ( reuse= K, MATR_ASSE= K)
dep=RESOUDRE               ( CHAM_NO  = F ,   MATR= K )
```

5.3 Resolution by the method of the pre conjugate gradient conditioned

```
NU=NUMÉRIQUE_DDL        (MATR_RIGI= KEL, METHODE= "PCG")

K=ASSE_MATRICE             ( MATR_ELEM= KEL, NUME_DDL= NU)
F=ASSE_VECTEUR             ( VECT_ELEM= FEL, NUME_DDL= NU)
KPREC=FACTORISER        ( MATR_ASSE= K)
dep=RESOUDRE               ( CHAM_NO  = F ,   MATR= K,
                          NMAX_ITER= 1000  ,   RESI_RELA= 1e-07
                       )
```

5.4 Resolution by method PETSC

```
NU=NUMÉRIQUE_DDL        (MATR_RIGI= KEL, METHODE= "PETSC")

K=ASSE_MATRICE             ( MATR_ELEM= KEL, NUME_DDL= NU)
F=ASSE_VECTEUR             ( VECT_ELEM= FEL, NUME_DDL= NU)
K=FACTORISER            ( reuse=K, MATR_ASSE= K)
dep=RESOUDRE               ( CHAM_NO  = F ,   MATR= K, MATR_PREC= K,
                          ALGORITHME=' GMRES',
                          NMAX_ITER= 1000  ,   RESI_RELA= 1e-07
                       )
```