Opérateur THER_LINEAIRE

1 But

Résoudre un problème de thermique linéaire en régime stationnaire ou évolutif.

Le chargement thermique est défini par le mot clé EXCIT.

La discrétisation temporelle est fournie par la liste d'instants définie sous le mot clé LIST_INST. Ce calcul évolutif peut être initialisé, au premier instant, de trois manières différentes (mot clé ETAT_INIT) :

- par une température constante,
- par un champ de température, défini au préalable, ou extrait d'un calcul précédent,
- par un calcul stationnaire préalable.

Le concept produit par cet opérateur est de type evol_ther.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

2 Syntaxe

```
temper [evol_ther] = THER_LINEAIRE
 ( RESULTAT = temper,
   ♦  MODELE       =  mo,              [modele]
   ♦  CHAM_MATER   =  chmat,           [cham_mater]
   ◊  CARA_ELEM    =  carac,           [cara_elem]
   ♦  EXCIT        =_F(
         ♦  CHARGE    =  char,        [charge]
            FONC_MULT =  fonc,        [fonction, formule]
                  ),
   ◊   TYPE_CALCUL =      /   'TRAN' ,              [ DEFAUT]
                          /  'STAT',
   Si TYPE_CALCUL = 'TRAN'
      ♦  ETAT_INIT    =_F(
                  /  STAT            = 'OUI',
                  /  VALE            = tinit,     [R]
                  /  CHAM_NO         = tinit,     [cham_no]
                  /  EVOL_THER       = temp,      [evol_ther]
                     ◊  / NUME_ORDRE = nuini,     [I]
                        / INST       = instini,   [R]
                     ◊  PRECISION    = /1.0E-3,   [DEFAUT]
                                       /prec,     [R]
                     ◊  CRITERE      = /'RELATIF',[DEFAUT]
                                       /'ABSOLU',
                  ),
      ◊  PARM_THETA   =      /  theta,                [R]
                             /  0.57,                 [DEFAUT]
   ◊  INCREMENT    =_F(
         ... voir STAT_NON_LINE [U4.51.03] mot-clé INCREMENT ...
                   ),
   ◊  SOLVEUR      =_F(  . . . voir [U4.50.01] . . .  ),
   ◊  ARCHIVAGE    =_F(
         ... voir STAT_NON_LINE [U4.51.03] mot-clé ARCHIVAGE ...
                    ),
   ◊  TITRE        =         titre,                 [l_Kn]

)
```


Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3 Opérandes

3.1 Opérande RESULTAT

◊ RESULTAT= mo

Nom de l'objet résultat à enrichir en cas de poursuite de calcul (voir aussi ETAT_INIT).

3.2 Opérande MODELE

♦ MODELE = mo

Nom du modèle dont les éléments font l'objet du calcul thermique.

3.3 Opérande CHAM_MATER

♦ CHAM_MATER = chmat

Nom du champ de matériau affecté sur le modèle.

3.4 Opérande CARA_ELEM

◊ CARA_ELEM = carac

Le concept carac contient les caractéristiques des éléments de coque thermique, s'ils existent dans le modèle.

3.5 Mot clé EXCIT

♦ EXCIT =

Opérande permettant de définir plusieurs chargements. Pour chaque occurrence du mot clé facteur, on définit une charge éventuellement multipliée par une fonction de temps.

3.5.1 Opérande CHARGE

♦ CHARGE = char

Concept de type charge produit par AFFE_CHAR_THER ou par AFFE_CHAR_THER_F [U4.44.02].
Comme pour la mécanique, il est également possible de définir une charge de type «degré de liberté imposé », ici la température, par AFFE_CHAR_CINE [U4.44.03].

Remarque importante :

Pour chaque occurrence du mot clé facteur EXCIT les différents concepts char utilisés doivent être construits sur le même modèle mo .

3.5.2 Opérande FONC_MULT

◊ FONC_MULT = fonc

Coefficient multiplicatif fonction du temps (concept de type fonction, nappe ou formule) appliqué à la charge.

Remarque importante :

L'utilisation concomitante de FONC_MULT avec une charge contenant des chargements thermiques dépendant de la température est interdite ; c'est-à-dire pour des chargements de type ECHANGE_* .

3.5.3 Opérande TYPR_CALCUL

◊ TYPE_CALCUL = / 'TRAN' , [DEFAUT]
/ ' STAT' ,

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

Le calcul est évolutif si TYPE_CALCUL = 'TRAN' ou stationnaire si TYPE_CALCUL='STAT'. Dans le cas d'un calcul évolutif, il faut préciser le calcul de l'état initial.

3.6 Mot clé ETAT_INIT

Si TYPE_CALCUL= 'TRAN'
♦ ETAT_INIT =

Permet de définir le champ initial à partir duquel le calcul évolutif est effectué.

3.6.1 Opérande STAT

/ STAT = 'OUI'

La valeur initiale du champ de température est alors le résultat d'un calcul stationnaire préalable.

3.6.2 Opérande VALE

/ VALE = tinit

La valeur initiale de température est prise constante sur toute la structure.

3.6.3 Opérande CHAM_NO

/ CHAM_NO = tinit

La valeur initiale est définie par un cham_no de température (résultat de l'opérateur CREA_CHAMP [U4.72.04]).

3.6.4 Opérande EVOL_THER

/ EVOL_THER = temp

La valeur initiale est extraite d'une structure de données de type evol_ther.

3.6.5 Opérande NUME_ORDRE/INST

◊ /NUME_ORDRE = nuini_evol
/INST = instini_evol

Numéro d'ordre du champ à extraire de cette structure de donnée. Extraction de l'état thermique initial dans l'evol_ther_temp à partir du numéro d'archivage NUME_ORDRE ou de l'instant d'archivage INST pour effectuer la poursuite du calcul. Si NUME_ORDRE ou INST ne sont pas remplis, on prend le dernier numéro archivé existant dans evol.

Remarque :

Attention, il s'agit du numéro d'ordre dans la structure de donnée lue en reprise par le mot clé EVOL_THER précédent. Si cette structure de donnée a été calculée avec une liste d'instants différente de celle utilisée sous le mot clé facteur INCREMENT de la résolution courante, il est impératif de renseigner NUME_ORDRE sous INCREMENT, la même valeur de numéro d'ordre correspondant à des instants physiques différents. Dans le cas où les deux listes d'instants sont identiques, on peut se dispenser de renseigner deux fois le même NUME_ORDRE, sous ETAT_INIT et sous INCREMENT.

3.6.6 Opérande PRECISION/CRITERE

Cf. [U4.71.00].

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

3.7 Mot clé INCREMENT

◊ INCREMENT =

Permet de définir les instants de calcul qui déterminent les intervalles de temps pris pour intégrer l'équation différentielle.

Les opérandes du mot clé INCREMENT ont la même signification que dans l'opérateur STAT_NON_LINE, voir le document [U4.51.03].

Remarque :

Si le mot clé INCREMENT est absent, on crée une liste d'instants réduite au seul réel 0 dans le cas d'un calcul stationnaire.

3.8 Opérande PARM_THETA

Cette opérande n'est disponible que dans le cas TYPE_CALCUL = 'TRAN'.

◊ PARM_THETA = / theta, [R]
/ 0.57, [DEFAUT]

L'argument theta est le paramètre de la thêta-méthode appliquée au problème évolutif. Il doit être compris entre 0 (méthode explicite) et 1 (méthode totalement implicite). En l'absence du mot clé, la valeur utilisée est $\theta=0.57$, un peu supérieure à $\theta=0.5$ correspondant au schéma de Crank-Nicholson. L'incidence du choix de theta sur la stabilité de la méthode est détaillée dans [R5.02.02].

3.9 Mot clé SOLVEUR

◊ SOLVEUR =

Ce mot clé facteur est facultatif : il permet de définir la méthode de résolution des systèmes linéaires. Cet opérande est commun à l'ensemble des commandes globales [U4.50.01].

3.10 Mot clé ARCHIVAGE

◊ ARCHIVAGE =

Ce mot clé est facultatif : par défaut, l'ensemble des champs calculés pour tous les pas calculés est archivé dans le concept resultat issu de la commande. Il sert à stocker certains numéros d'ordre dans une structure de données resultat et/ou exclure du stockage certains champs.

Ce mot-clé est identique à son équivalent pour l'opérateur STAT_NON_LINE, se référer à la documentation [U4.51.03] pour la description des sous mots-clés.

Remarque :

En cas d'arrêt du calcul par manque de temps CPU, les pas de temps précédemment calculés sont sauvegardés dans la base.

3.11 Opérande TITRE

◊ TITRE = titre

Titre que l'on veut donner au résultat temp stocké dans la structure de données de type evol_ther [U4.03.01].

4 Modélisation

Les problèmes de thermique linéaire peuvent être traités avec des modèles utilisant les éléments finis 3D, 2D, AXIS ou COQUE décrits dans les documents [U3.22.01], [U3.23.01], [U3.23.02] et [U3.24.01].

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)

5 Exemple

5.1 Calcul transitoire

```
lr8 =  DEFI_LIST_REEL  (  DEBUT = 0.E0 ,
                       INTERVALLE =(
                          _F(JUSQU_A   =  2.E-4  ,  NOMBRE =  2  ),
                          _F(JUSQU_A   =  1.E-3  ,  NOMBRE =  10 ),
                          _F(JUSQU_A   =  1.E-2  ,  NOMBRE =  9  ),
                          _F(JUSQU_A   =  1.E-1  ,  NOMBRE =  9  ),
                          _F(JUSQU_A   =  1.E+0  ,  NOMBRE =  9  ),
                          _F(JUSQU_A   =  2.0,   NOMBRE =  10 ),))

tempe = THER_LINEAIRE  (  MODELE    =  moth,
                       CHAM_MATER   =  chmat,
                       EXCIT        =  _F(CHARGE       =  chth),
                       TYPE_CALCUL  = 'TRAN',
                       ETAT_INIT =  _F(STAT   =  'OUI'),
                       INCREMENT =  _F(LIST_INST    =  lr8,
                                       NUME_INST_FIN=  30)
                       )

tempe = THER_LINEAIRE  (  reuse  =  tempe,
                        RESULTAT =  tempe,
                       MODELE    =  moth,
                       CHAM_MATER   =  chmat,
                       EXCIT        =  _F(CHARGE =  chth),
                       TYPE_CALCUL  = 'TRAN',
                       ETAT_INIT =  _F(EVOL_THER    =  tempe
                                       NUME_ORDRE       =  30),
                       INCREMENT =  _F(LIST_INST    =  lr8),
                       )
```

Le premier appel à la commande THER_LINEAIRE permet d'effectuer un calcul stationnaire à l'instant 0. et d'enchaîner un calcul évolutif jusqu'à l'instant $0.1s$ (31 instants de calcul soit 30 calculs d'évolution).

Le second appel permet d'enrichir le concept tempe précédent, le calcul évolutif est poursuivi à partir du 31ième instant de calcul.

6 Remarque

La commande CALC_CHAMP [U4.81.04] permet de calculer les flux de chaleur, aux points d'intégration ou aux nœuds, à partir du champ aux nœuds de température ainsi obtenu par THER_LINEAIRE.

Document diffusé sous licence GNU FDL (http://www.gnu.org/copyleft/fdl.html)