

Opérateur CALC_TRANSFERT

1 But

Cette commande permet de calculer la fonction de transfert d'un système linéaire entre deux points. Il est aussi possible d'obtenir la réponse en un point quelconque du modèle numérique en fonction d'un signal de sortie renseigné par l'utilisateur.

Cet opérateur utilise en données d'entrée, les réponses du système mécanique à des sollicitations unidirectionnelles de préférence de type bruit blanc. Il produit une ou deux structure de données de type `table`, l'une pour la matrice de fonction de transfert, l'autre pour l'éventuel signal calculé.

Produit une ou plusieurs structures de données de type `table`.

2 Syntaxe

```
CALC_TRANSFERT (
  ♦ RESULTAT_X = resu_x , / [dyna_trans]
                        / [dyna_harmo]
                        / [harm_gene]
                        / [tran_gene]

  ♦ RESULTAT_Y = resu_y , / [dyna_trans]
                        / [dyna_harmo]
                        / [harm_gene]
                        / [tran_gene]

  ◊ RESULTAT_Z = resu_z , / [dyna_trans]
                        / [dyna_harmo]
                        / [harm_gene]
                        / [tran_gene]

  ♦ NOM_CHAM = / 'ACCE',
              / 'VITE',
              / 'DEPL',

  ♦ ENTREE = _F(
  ♦ GROUP_NO = lgn0, [l_gr_noeud]
  ),

  ♦ SORTIE = _F(
  ♦ GROUP_NO = lgn0, [l_gr_noeud]
  ),

  ◊ REPERE = / 'ABSOLU' ,
            / 'RELATIF' , [DEFAULT]
  ♦ ENTRAINEMENT = _F(
  ♦ DX = entr_x, [fonction]
  ♦ DY = entr_y, [fonction]
  ◊ DZ = entr_z, ), [fonction]

  ◊ SIGNAL = _F(
  ♦ MESURE_X = f_x, [fonction]
  ♦ MESURE_Y = f_y, [fonction]
  ◊ MESURE_Z = f_z, [fonction]
  ♦ TABLE_RESU = CO('table'), [table]
  ◊ TYPE_RESU = / 'TEMPOREL',
               / 'HARMONIQUE', [DEFAULT]
```

3 Description de la macro commande et étapes préalables

La commande `CALC_TRANSFERT` permet de calculer la matrice fonction de transfert entre deux points de la structure. Elle permet également à l'utilisateur de déterminer les signaux en un point de la structure connaissant les signaux en un autre point de la structure.

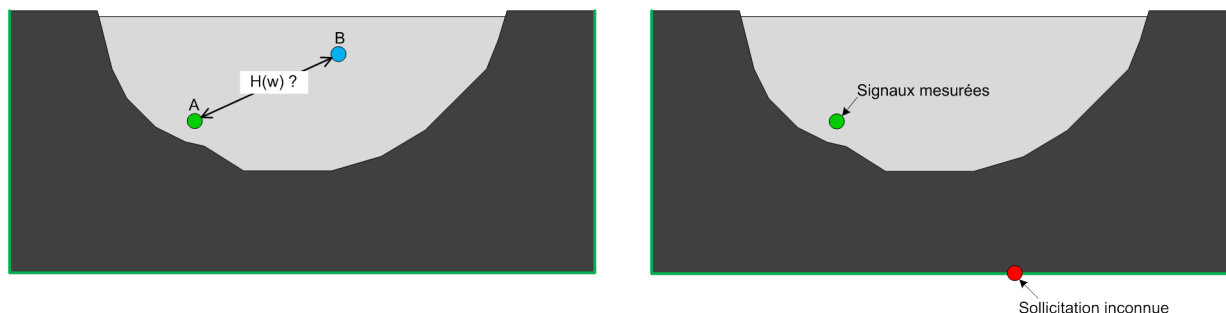


Figure 1 – Représentation schématique de l'utilisation de la macro commande

Préalablement à l'utilisation de la commande, l'utilisateur doit réaliser plusieurs calculs dynamiques sous sollicitation unidirectionnelle. Par exemple si l'étude est en 3D (resp. 2D), il doit réaliser 3 calculs (resp. 2 calculs) excitant l'une des trois (resp. 2) directions. Il est préférable de réaliser les calculs dynamiques via un bruit blanc mono-directionnel.

Les calculs dynamiques doivent être réalisés avec la même liste de fréquences si le calcul est harmonique et avec la même liste de temps si le calcul est transitoire. Pour un calcul transitoire, l'utilisateur doit veiller à avoir une liste à pas constant. Dans le cas contraire un message d'erreur lui sera transmis car la transformée de Fourier nécessaire aux différents calculs nécessite un pas constant.

Les calculs dynamiques devront être de même type, soit `tran_gene`, soit `harm_gene`, soit `dyna_trans` ou bien `dyna_harmo`.

Lorsque l'utilisateur doit renseigner les fonctions d'entraînement et/ou les signaux mesurés, il est préférable que ces derniers soit discrétisés sur la même liste de fréquences (cas d'un calcul harmonique) ou sur la même liste de temps (cas d'un calcul transitoire).

4 Opérandes

4.1 Concepts créés par la commande

Cette commande crée en sortie un concept de type table. Il est possible de sortir également une seconde table contenant les signaux calculés à l'aide des signaux mesurés.

Le premier concept produit quelque soit l'utilisation de la macro_commande par l'utilisateur est une table qui contient les données de la matrice fonction de transferts rangées comme suit :

- La première colonne correspond aux fréquences (FREQ) ;
- Les suivantes correspondent aux éléments de la matrices rangés dans l'ordre par lignes H_{xx} , H_{xy} , H_{xz} , H_{yx} , H_{yy} , H_{yz} , H_{zx} , H_{zy} et H_{zz} . Dans le cas 2D, on aura H_{xx} , H_{xy} , H_{yx} et H_{yy} .

$$\underline{H}(\omega) = \begin{pmatrix} H_{xx} & H_{xy} & H_{xz} \\ H_{yx} & H_{yy} & H_{yz} \\ H_{zx} & H_{zy} & H_{zz} \end{pmatrix}$$

4.2 Opérandes RESULTAT_X, RESULTAT_Y et RESULTAT_Z

◆ RESULTAT_X = resu_x

Nom du concept de type `resultat` pour lequel le calcul a été menée avec une sollicitation unidirectionnelle suivant x.

◆ RESULTAT_Y = resu_y

Nom du concept de type `resultat` pour lequel le calcul a été menée avec une sollicitation unidirectionnelle suivant y .

◇ RESULTAT_Z = resu_z

Nom du concept de type `resultat` pour lequel le calcul a été menée avec une sollicitation unidirectionnelle suivant z .

Remarques :

*Seules les structures résultat de type `dyna_trans` , `dyna_harmo` , `tran_gene` et `harm_gene` peuvent être renseignées dans ces opérandes.
Les calculs produits par les opérateurs de dynamique et qui sont à renseigner dans les opérandes `RESULTAT_X` , `RESULTAT_Y` et `RESULTAT_Z` doivent être de même type : `dyna_trans` , ou `dyna_harmo` ou `tran_gene` ou `harm_gene`.*

4.3 Opérandes NOM_CHAM

◆ NOM_CHAM = nomcha

Nom symbolique du champ que l'on souhaite extraire des calculs dynamiques réalisés au préalable par l'utilisateur et renseignés dans les opérandes `RESULTAT_X`, `RESULTAT_Y` et `RESULTAT_Z`. Les noms possibles du champ sont : 'DEPL', 'VITE' et 'ACCE'.

Remarque :

*Les différentes fonctions renseignées par l'utilisateur telles que les fonctions d'entraînement renseignées dans le mot-clé `ENTRAINEMENT` ou bien les fonctions mesurées renseignées dans le mot clé `SIGNAL` devront être du même type que **nomcha**.*

4.4 Opérande REPERE

◆ REPERE = /'ABSOLU'
/'RELATIF'

Cet opérande obligatoire permet de renseigner le type de réponse calculées par les opérateurs dynamiques. Si le repère est `RELATIF`, alors l'utilisateur doit renseigner le signal d'entraînement dans le mot clé `ENTRAINEMENT` (voir paragraphe 4.5).

4.5 Mot clé ENTRAINEMENT

Le mot clé facteur `ENTRAINEMENT` permet de renseigner les sollicitations d'entraînement ayant été utilisées pour réaliser les différents calculs dynamiques linéaires.

Remarque :

*Les différentes fonctions d'entraînement renseignées dans le mot-clé `ENTRAINEMENT` devront être du même type que **nomcha** ('DEPL', 'VITE' ou 'ACCE').*

◆ DX = entr_x

Cette opérande permet de renseigner la sollicitation unidirectionnelle (suivant la direction x) utilisée pour le premier calcul dynamique.

◆ `DY = entr_y`

Cette opérande permet de renseigner la sollicitation unidirectionnelle (suivant la direction y) utilisée pour le deuxième calcul dynamique.

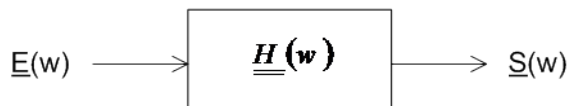
◇ `DZ = entr_z`

Cette opérande permet de renseigner la sollicitation unidirectionnelle (suivant la direction) z utilisée pour le troisième calcul dynamique.

4.6 Mots clé ENTREE et SORTIE

◆ `GROUP_NO = lgn0`

Les mots clés facteur ENTREE et SORTIE permettent de renseigner le nom d'un groupe de nœud (`grno`, contenant un seul nœud) sur lesquels portent l'extraction du champ 'DEPL' ou 'VITE' ou 'ACCE'



4.7 Mot clé SIGNAL

Ce mot clé facteur (optionnel) permet de déterminer la sollicitation dynamique ayant conduit aux signaux mesurés renseignés par l'utilisateur dans le mot clé facteur SIGNAL.

4.7.1 Opérandes MESURE_X, MESURE_Y et MESURE_Z

◆ `MESURE_X = f_x`

Cette opérande permet de renseigner le signal mesuré suivant la direction x.

◆ `MESURE_Y = f_y`

Cette opérande permet de renseigner le signal mesuré suivant la direction y.

◇ `MESURE_Z = f_z`

Cette opérande permet de renseigner le signal mesuré suivant la direction z.

Remarque :

*Les signaux doivent être de même type que **nomcha** et doivent être discrétisés sur une même liste de fréquences ou de temps.*

4.7.2 Opérande TYPE_RESU

◇ `TYPE_RESU = /'TEMPOREL',
/'HARMONIQUE'`

Cette opérande permet de renseigner le type de résultats souhaité en sortie de la table.

4.7.3 Opérande TABLE_RESU

◆ TABLE_RESU = nom_table

Cette opérande permet de renseigner le nom de la table dans laquelle figurera les signaux calculés connaissant les signaux mesurés.

La table sera composée de plusieurs colonnes rangées comme telles :

Si TYPE_RESU = 'TEMPOREL' :

INST, FX, FY, FZ

Si TYPE_RESU = 'HARMONIQUE' :

FREQ, Re_FX, Im_FX, Re_FY, Im_FY, Re_FZ, Im_FZ