

Titre: LUSS : sous-programmes pour la résolution d'un système
Title: d'équations algébriques stockées en ligne de ciel

Auteurs: Maryse Pagé, André Garon, & Ricardo Camarero
Authors:

Date: 1989

Type: Rapport / Report

Référence: Pagé, M., Garon, A., & Camarero, R. (1989). LUSS : sous-programmes pour la résolution d'un système d'équations algébriques stockées en ligne de ciel.
Citation: (Technical Report n° EPM-RT-89-13). <https://publications.polymtl.ca/9661/>

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/9661/>
PolyPublie URL:

Version: Version officielle de l'éditeur / Published version

Conditions d'utilisation: Tous droits réservés / All rights reserved
Terms of Use:

Document publié chez l'éditeur officiel

Document issued by the official publisher

Institution: École Polytechnique de Montréal

Numéro de rapport: EPM-RT-89-13
Report number:

URL officiel:
Official URL:

Mention légale:
Legal notice:

06 OCT. 1989

EPM/RT-89/13

LUSS: Sous-programmes pour la résolution d'un système
d'équations algébriques stockées en ligne de ciel.

Maryse Page, étudiante M.Sc.A.

André Garon, chercheur

Ricardo Camarero, professeur titulaire

Septembre 1989

gratuit

Ce projet a été réalisé grâce à l'apport financier du Conseil de Recherche en Sciences Naturelles et en Génie (CRSNG) et du Centre de Recherche Informatique de Montréal (CRIM).

Tous droits réservés. On ne peut reproduire ni diffuser aucune partie du présent ouvrage, sous quelque forme que ce soit, sans avoir obtenu au préalable l'autorisation écrite de l'auteur.

Dépôt légal, 3e trimestre 1989
Bibliothèque nationale du Québec
Bibliothèque nationale du Canada

Pour se procurer une copie de ce document, s'adresser au:

Service de l'édition
École Polytechnique de Montréal
Case postale 6079, Succursale A
Montréal (Québec) H3C 3A7
(514) 340-4000

Compter 0,10 \$ par page (arrondir au dollar le plus près) et ajouter 3, 00 \$ (Canada) pour la couverture, les frais de poste et la manutention. Régler en dollars canadiens par chèque ou mandat-poste au nom de l' École Polytechnique de Montréal. Nous n'honoreronrs que les commandes accompagnées d'un paiement, sauf s'il y a eu entente préalable dans le cas d'établissements d'enseignement, des sociétés ou d'organismes canadiens.

Table des Matières

Introduction	1
Méthode de stockage en ligne de ciel	2
Exemple: programme utilisant les sous-programmes LUSS	5
Description des variables	5
Programmes	6
Exemple: construction du vecteur d'adressage NUMVAR	9
Description des variables	9
Programme	10
Exemple de référence pour la description des sous-programmes	12
Description des sous-programmes LUSS	14
Sous-programme PRFSS	15
Sous-programme ADRSS	17
Sous-programme AGBSS	19
Sous-programme FLUSS	22
Sous-programme SLUSS	26

Introduction

Ce rapport présente les sous-programmes pour le stockage et la résolution d'un système d'équations algébriques. Il décrit la méthode de stockage en ligne de ciel et quelques extraits d'un programme qui utilise ces sous-programmes. Pour chacun des sous-programmes, le rapport décrit la fonction, la liste des arguments et leurs descriptions et l'algorithme. Un exemple de matrice à traiter permet d'illustrer chacun des sous-programmes.

Méthode de stockage en ligne de ciel

La méthode utilisée s'applique à une matrice dont la structure est symétrique mais dont le contenu n'est pas nécessairement symétrique. La ligne de ciel est l'enveloppe des lignes et des colonnes de longueur variable qui entoure tous les termes non nuls. Il n'est pas nécessaire de stocker les termes nuls hors de la ligne de ciel car ils sont invariables lors de la factorisation pour la résolution du système d'équations.

$$\begin{bmatrix} a_{11} & 0 & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & a_{25} \\ 0 & a_{32} & a_{33} & a_{34} & a_{35} \\ 0 & 0 & a_{43} & a_{44} & a_{45} \\ 0 & a_{52} & a_{53} & a_{54} & a_{55} \\ & & & \ddots & \end{bmatrix}$$

Figure 1

L'enveloppe de la ligne de ciel comprend globalement trois parties. Il y a les termes diagonaux, les termes en-dessous de la diagonale (partie inférieure) et les

termes au-dessus de la diagonale (partie supérieure). L'enveloppe de la ligne de ciel est constituée par plusieurs bras. Un bras comporte des termes inférieurs, des termes supérieurs et un terme diagonal. La figure 2 illustre le bras de la ligne de ciel correspondant à l'indice 5 de la matrice.

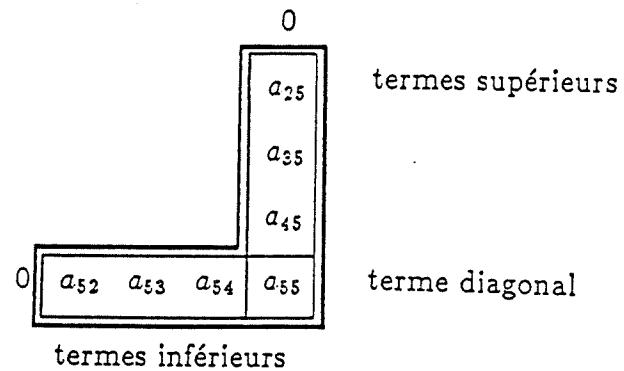


Figure 2

Les termes associés à un bras sont stockés dans l'ordre suivant: les termes inférieurs, les termes supérieurs et le terme diagonal.

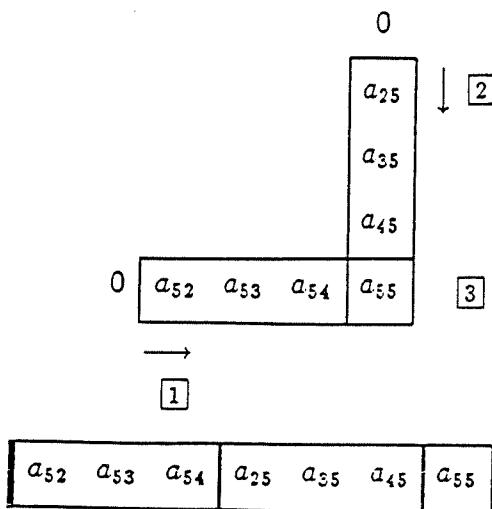


Figure 3

La hauteur de la ligne de ciel est la longueur des termes inférieurs ou des termes supérieurs.

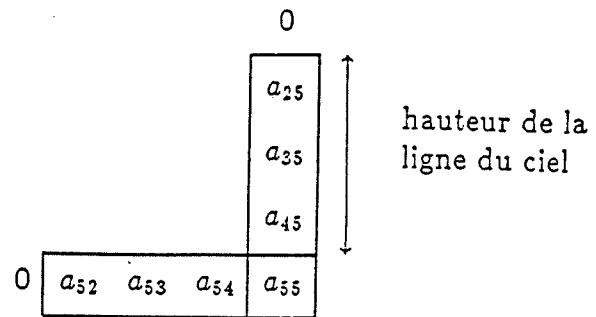


Figure 4

La matrice est stockée dans un vecteur. Le vecteur comprend consécutivement tous les bras qui forment la ligne de ciel.

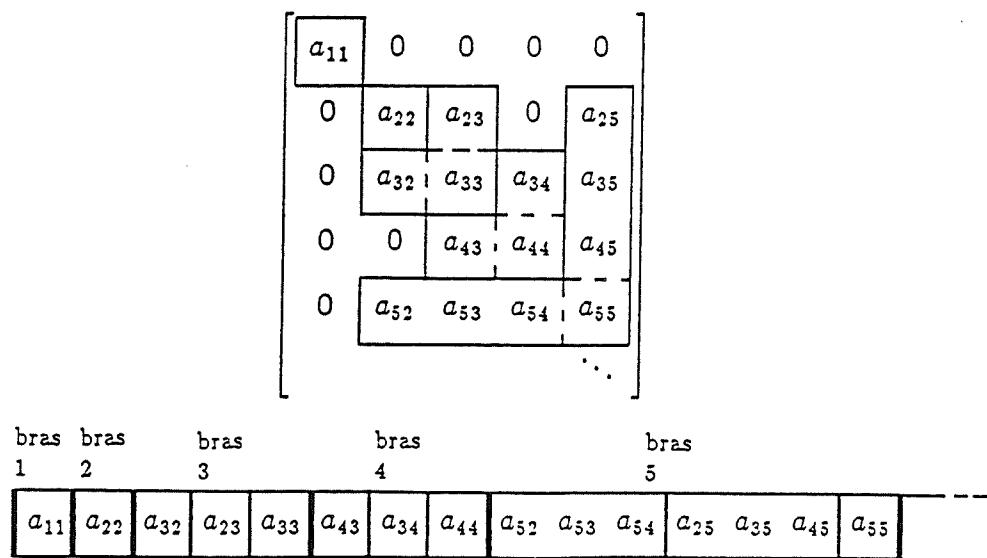


Figure 5

Exemple: programme utilisant les sous-programmes LUSS

Les extraits de ce programme permettent d'illustrer l'utilisation des sous-programmes LUSS dans le contexte de la méthode des éléments finis. Cette section présente aussi la description des variables utilisées en arguments des sous-programmes.

Description des variables

NBRDLE	Nombre de degrés de liberté de l'élément.
MAXDLE	Dimension maximale de la matrice élémentaire.
VADELM()	Vecteur d'adressage des degrés de liberté de l'élément.
MATELM(,)	Matrice élémentaire.
NBREQ	Nombre d'équations globales.
ADRMAT()	Vecteur d'adressage de la matrice globale en fonction de la ligne de ciel.
ESPMAT	Longueur effective utilisée du vecteur MATGLB.
HLCIEL()	Hauteur de la ligne de ciel de la matrice globale.

MATGLB() Matrice globale.
 MDDGLB() Membre de droite global.
 SOLUTN() Solution du système d'équations algébriques
 associées à la matrice globale.
 MSGERR Unité logique de sortie des messages d'erreurs.

Programme

PROGRAM UTLUSS

* Paramètres

INTEGER MAXDLE, MAXEQU, MAXVEC, MAXNOD
 PARAMETER (MAXDLE = 3, MAXEQU = 1000, MAXVEC = 10000)
 PARAMETER (MAXNOD = 1000)

* Variables utilisées en argument

INTEGER NBRDLE, MAXDLE, VADELM (MAXDLE)
 INTEGER NBREQ, ADRMAT (MAXEQU), ESPMAT
 INTEGER HLCIEL (MAXEQU)
 INTEGER MSGERR
 DOUBLE PRECISION MATELM (MAXDLE, MAXDLE)
 DOUBLE PRECISION MATGLB (MAXVEC)
 DOUBLE PRECISION MDDGLB (MAXEQU), SOLUTN (MAXEQU)

* Variables locales

INTEGER NBRELM
 INTEGER NUMVAR (MAXNOD)

...

* Initialisation du vecteur HLCIEL

DO 901 I = 1, MAXEQU

HLCIEL(I) = 0

901 CONTINUE

...

* Initialisation du vecteur MATGLB

DO 951 I = 1, MAXVEC

MATGLB(I) = 0.0D0

951 CONTINUE

...

** Construction du vecteur contenant le profil de la ligne

** de ciel (hauteur de la ligne de ciel HLCIEL)

DO 1001 NE = 1, NBRELM

* Construction du vecteur d'adressage de l'élément

DO 1002 J = 1, NBRDLE

VADELM(J) = NUMVAR (CNCELM (J, NE))

1002 CONTINUE

CALL PRFSS (NBRDLE, VADELM, HLCIEL)

1001 CONTINUE

** Construction du vecteur d'adressage de la matrice globale

** en fonction de la hauteur de la ligne de ciel (ADRMAT)

CALL ADRSS (NBREQ, HLCIEL, ADRMAT, ESPMAT)

...

** Assemblage de matrice globale (MATGLB)

DO 2001 NE = 1, NBRELM

* Construction de VADELM

DO 2002 J = 1, NBRDLE

VADELM(J) = NUMVAR (CNCELM (J, NE))

* Construction de la matrice élémentaire (MATELM)

DO 2003 I = 1, NBRDLE

MATELM (I, J) = ...

2003 CONTINUE

2002 CONTINUE

CALL AGBSS (MAXDLE, NBRDLE, VADELM, ADRMAT, MATELM, MATGLB)

2001 CONTINUE

...

** Factorisation LU de la matrice globale

CALL FLUSS (NBREQ, ADRMAT, MATGLB, MSGERR)

...

** Résolution du système d'équations algébriques pour

** le membre de droite MDDGLB (solution SOLUTN)

CALL SLUSS (NBREQ, ADRMAT, MATGLB, MDDGLB, SOLUTN)

...

Exemple: construction du vecteur d'adressage NUMVAR

Le vecteur d'adressage NUMVAR, utilisé dans l'exemple précédent, est construit selon une convention relative à la méthode des éléments finis. L'exemple suivant permet d'illustrer la convention qui a été utilisée pour les sous-programmes LUSS.

Description des variables

NBRELM	Nombre d'éléments
NNDG	Nombre de noeuds globaux
NBRNOD	Nombre de noeuds par élément
NDRCLT	Nombre de conditions de Dirichlet
CNCELM(,)	Connectivité
NUMVAR()	Vecteur d'adressage des degrés de liberté
DRCLET()	Valeur des conditions de Dirichlet
NUMELM	Numéro de l'élément
NUMNOD	Numéro du noeud

Programme

PROGRAM CSTNVR

* Paramètres

```
    INTEGER          MAXNOD, MAXELM
    PARAMETER        (MAXNOD = 1000, MAXELM = 1001)
```

* Variables locales

```
    INTEGER          NBRELM, NNDG, NBRNOD, NDRCLT
    INTEGER          CNCELM (MAXNOD, MAXELM)
    INTEGER          NUMVAR (MAXNOD)
    INTEGER          NUMELM, NUMNOD
    DOUBLE PRECISION DRCLET (MAXNOD)
```

...

* Lecture de la connectivité

```
DO 1100 NE = 1, NBRELM
  READ (DONNEE, *) NUMELM, (CNCELM (J, NUMELM), J = 1, NBRNOD)
1100 CONTINUE
```

...

* Initialisation

```
DO 1200 I = 1, NNDG
  NUMVAR(I) = 1
1200 CONTINUE
```

...

* Identification des conditions de Dirichlet

```
DO 1300 I = 1, NDRCLT
  READ (DONNEE, *) NUMNOD, VALDRC
```

```
IF (VALDRC.NE.0.0D0) THEN
  NUMVAR (NUMNOD) = -I
  DRCLET(I) = VALDRC
END IF

1300 CONTINUE

...
* Numérotation des inconnues
NBREQ = 0
DO 1400 I = 1, NNDG
  IF (NUMD(I).EQ.1) THEN
    NBREQ = NBREQ + 1
    NUMVAR(I) = NBREQ
  END IF

1400 CONTINUE
```

Exemple de référence pour la description des sous-programmes

Dans la section suivante, la description détaillée de chacun des sous-programmes sera faite. Pour illustrer la fonction de ces sous-programmes, il est possible de considérer un exemple de trois matrices élémentaires à assembler dans une matrice globale selon les vecteurs d'adressage des degrés de liberté suivants. Les éléments ont trois degrés de liberté (NBRDLE = 3).

Degrés	VADELM		
	Élement 1	Élement 2	Élement 3
1	1	3	4
2	2	4	5
3	3	5	6

La matrice globale assemblée a la forme suivante

a_{11}	a_{12}	a_{13}	0	0	0
a_{21}	a_{22}	a_{23}	0	0	0
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	0
0	0	a_{43}	a_{44}	a_{45}	a_{46}
0	0	a_{53}	a_{54}	a_{55}	a_{56}
0	0	0	a_{64}	a_{65}	a_{66}

Cette matrice est stockée en ligne de ciel en utilisant les sous-programmes PRFSS, ADRSS et AGBSS. La matrice est factorisée selon LU à l'aide du sous-programme FLUSS. La résolution du système d'équations algébriques associées à cette matrice est faite avec SLUSS.

Description des sous-programmes

LUSS

Pour la description des sous-programmes, la convention suivante est utilisée pour détailler les arguments.

- : argument en entrée
- ← : argument en sortie
- ↔ : argument en entrée et en sortie

- (I) : argument de type entier
- (D) : argument de type réel, double précision

Un vecteur $VEC(I)$ est noté VEC_I . Un tableau $MAT(I,J)$ est noté $MAT_{I,J}$.

L'indice * correspond à l'énoncé FORTRAN qui associe la dimension du vecteur ou du tableau en argument à la dimension de la variable du programme appelant.

Sous-programme PRFSS

Fonction	Calcul du profil de la ligne de ciel de la matrice globale (hauteur de la ligne de ciel)
Utilisation	<div style="border: 1px solid black; padding: 2px;">CALL PRFSS ($\overrightarrow{\text{NBRDLE}}$, $\overrightarrow{\text{VADELM}_*}$, $\overleftarrow{\text{HLCIEL}}_*$)</div>
En entrée	<p>(I) NBRDLE Nombre de degré de liberté de l'élément</p> <p>(I) VADELM(*) Vecteur d'adressage des degrés de liberté de l'élément</p>
En entrée et en sortie	<p>(I) HLCIEL(*) Profil de la ligne de ciel de la matrice globale (hauteur de la ligne de ciel)</p>
Notes	<p>Avant le premier appel de PRFSS, le vecteur HLCIEL doit être initialisé à 0.</p> <p>Le vecteur HLCIEL est mis à jour à chaque fois que PRFSS est appelé pour un élément.</p>
Algorithme	Une combinaison locale de degrés de liberté de l'élément correspond à une position dans la matrice globale, à l'aide du vecteur d'adressage VADELM. Étant donné que la matrice est symétrique dans sa structure, le sous-programme teste seulement les combinaisons appartenant à la partie supérieure de la matrice globale. Pour ces combinaisons, il calcule leur hauteur par rapport à la

diagonale. Il conserve la hauteur maximale pour chacun des indices de la matrice globale.

Exemple

Le contenu du vecteur HLCIEL est présenté dans le tableau suivant après l'appel de PRFSS pour chacun des éléments.

No. de l'équation	HLCIEL		
	Élement 1	Élement 2	Élement 3
1	0	0	0
2	1	1	1
3	2	2	2
4	0	1	1
5	0	2	2
6	0	0	2

Sous-programme ADRSS

Fonction	Calcul du vecteur d'adressage de la matrice globale en fonction de la hauteur de la ligne de ciel
Utilisation	CALL ADRSS ($\overrightarrow{\text{NBREQ}}$, $\overrightarrow{\text{HLCIEL}_*}$, $\overleftarrow{\text{ADRMAT}_*}$, $\overleftarrow{\text{ESPMAT}}$)
En entrée	<p>(I) NBREQ</p> <p>Nombre d'équations globales</p> <p>(I) HLCIEL(*)</p> <p>Profil de la ligne de ciel de la matrice globale (hauteur de la ligne de ciel)</p>
En sortie	<p>(I) ADRMAT(*)</p> <p>Vecteur d'adressage de la matrice globale en fonction de la hauteur de la ligne de ciel</p> <p>(I) ESPMAT</p> <p>Longueur effective utilisée du vecteur MATGLB</p>
Notes	Avant l'appel de ADRSS, PRFSS doit être appelé pour chacun des éléments qui composent la matrice globale.
Algorithme	En connaissant la hauteur de la ligne de ciel pour tous les indices de la matrice globale, la longueur des bras peut être calculée. Un bras est composé par les termes inférieurs, par les termes supérieurs et par le terme diagonal. A l'aide de AGBSS, la matrice globale sera stockée dans un vecteur. Le vecteur contient consécutivement tous les bras, en indice croissant. Le sous-programme ADRSS permet de calculer la position de

l'élément diagonal de chacun des bras. Chacune des positions est stockée dans ADRMAT.

Exemple

Le vecteur d'adressage ADRMAT de l'exemple est le suivant.

No. de l'équation	ADRMAT
1	1
2	4
3	9
4	12
5	17
6	22

Il en est ainsi car la matrice sera stockée de la façon indiquée à la figure suivante

1 a_{11}	3 a_{12}	7 a_{13}	0	0	0
2 a_{21}	4 a_{22}	8 a_{23}	0	0	0
5 a_{31}	6 a_{32}	9 a_{33}	11 a_{34}	15 a_{35}	0 a_{36}
0 a_{41}	0 a_{42}	10 a_{43}	12 a_{44}	16 a_{45}	20 a_{46}
0 a_{51}	0 a_{52}	13 a_{53}	14 a_{54}	17 a_{55}	21 a_{56}
0 a_{61}	0 a_{62}	0 a_{63}	18 a_{64}	19 a_{65}	22 a_{66}

Pour cette matrice, ESPMAT = 22.

Sous-programme AGBSS

Fonction	Assemblage de la matrice élémentaire dans la matrice globale
Utilisation	<pre>CALL AGBSS (MAXDLE, NBRDLE, VADELM., ADRMAT, MATELM_{MAXDLE,*}, MATGLB*)</pre>
En entrée	<p>(I) MAXDLE Dimension maximale de la matrice élémentaire</p> <p>(I) NBRDLE Nombre de degré de liberté de l'élément</p> <p>(I) VADELM(*) Vecteur d'adressage des degrés de liberté de l'élément</p> <p>(I) ADRMAT(*) Vecteur d'adressage de la matrice globale en fonction de la hauteur de la ligne de ciel</p> <p>(D) MATELM (MAXDLE,*) Matrice élémentaire</p>
En entrée et en sortie	<p>(D) MATGLB(*) Matrice globale</p>
Notes	<p>Avant le premier appel de AGBSS, le vecteur MATGLB doit être initialisé à 0.0D0.</p> <p>Le vecteur MATGLB est mis à jour à chaque fois que AGBSS est appelé pour un élément.</p> <p>Avant l'appel de AGBSS, le sous-programme ADRSS</p>

doit être appelé pour définir le vecteur d'adressage ADR-MAT de la matrice globale.

Algorithme

Une combinaison locale de degrés de liberté de l'élément correspond à une position dans la matrice globale, à l'aide du vecteur d'adressage VADELM. Il y a trois cas possibles de position d'un terme dans la matrice globale.

1^o cas: Le terme appartient à la diagonale.

POSITION = position du terme diagonal du bras correspondant.

2^o cas: Le terme appartient à la partie inférieure.

POSITION = position du premier terme supérieur du bras correspondant
+ différence de position par rapport à la diagonale.

3^o cas: Le terme appartient à la partie supérieure.

POSITION = position du terme diagonal du bras correspondant
+ différence de position par rapport à la diagonale.

Lorsque la position du terme est connue, la contribution du terme de la matrice élémentaire est ajoutée au terme correspondant de la matrice globale.

Exemple

L'exemple suivant illustre la position de chaque terme de chacune des matrices élémentaires dans la matrice globale.

I_{local}	J_{local}	POSITION		
		Élement 1	Élement 2	Élement 3
1	1	1	9	12
1	2	3	11	16
1	3	7	15	20
2	1	2	10	14
2	2	4	12	17
2	3	8	16	21
3	1	5	12	18
3	2	6	14	19
3	3	9	17	22

Sous-programme FLUSS

Fonction	Factorisation LU de la matrice globale stockée en ligne de ciel
Utilisation	<div style="border: 1px solid black; padding: 2px; display: inline-block;">CALL FLUSS (NBREQ, ADRMAT, MATGLB, MSGERR)</div>
En entrée	<p>(I) NBREQ Nombre d'équations globales</p> <p>(I) ADRMAT(*) Vecteur d'adressage de la matrice globale en fonction de la hauteur de la ligne de ciel</p> <p>(I) MSGERR Indicateur de messages d'erreurs et unité logique de sortie des messages d'erreurs, si MSGERR > 0. Si MSGERR = 0, il n'y a pas de message. Il y a des messages d'erreurs si un ou plusieurs termes diagonaux sont plus petits que 1.0×10^{-8}.</p>
En entrée et en sortie	(D) MATGLB(*) Matrice globale
Sous-programmes utilisés	Fonction DDOT de la librairie ESSL
Notes	Avant l'appel de FLUSS, AGBSS, doit être appelé pour chacun des éléments qui composent la matrice globale. En entrée, le vecteur MATGLB représente la matrice globale. En sortie, il représente la matrice globale factorisée. Les matrices L et U remplacent la matrice globale.

Lors de la factorisation, il n'y a pas de pivotage des lignes de la matrice pour éviter les pivots nuls. Si le pivot est plus petit que 1×10^{-8} , le pivot est remplacé par la valeur 1×10^{-8} , tout en conservant le signe du pivot.

Algorithme

L'algorithme de ce sous-programme est celui d'une factorisation LU d'une matrice. La matrice L correspond à la matrice triangulaire inférieure ayant des termes unitaires sur sa diagonale. La matrice U correspond à la matrice triangulaire supérieure. Cette décomposition permet le stockage des matrice L et U dans le vecteur MATGLB, selon le même vecteur d'adressage de la ligne de ciel ADRMAT. La particularité de cette factorisation LU , c'est qu'elle procède en tenant compte de la structure de la matrice globale, qui est stockée en ligne de ciel.

Exemple

Pour la matrice globale de l'exemple, les deux matrices L et U ont les formes suivantes.

Matrice L

1	0	0	0	0	0
2					
ℓ_{21}	1	0	0	0	0
5	6				
ℓ_{31}	ℓ_{32}	1	0	0	0
		10			
0	0	ℓ_{43}	1	0	0
		13	14		
0	0	ℓ_{53}	ℓ_{54}	1	0
			18	19	
0	0	0	ℓ_{64}	ℓ_{65}	1

Matrice U

1	3	7			
u_{11}	u_{12}	u_{13}	0	0	0
	4	8			
0	u_{22}	u_{23}	0	0	0
		9	11	15	
0	0	u_{33}	u_{34}	u_{35}	0
			12	16	20
0	0	0	u_{44}	u_{45}	u_{46}
				17	21
0	0	0	0	u_{55}	u_{56}
					22
0	0	0	0	0	u_{66}

- Pour le calcul de u_{11} ,
 - $u_{11} = a_{11}$
 - MATGLB(1) n'est pas modifié
- Pour le calcul de u_{12} ,
 - $u_{12} = a_{12}$
 - MATGLB(3) n'est pas modifié
- Pour le calcul de ℓ_{21} ,
 - $\ell_{21} = a_{21}/u_{11}$
 - $\text{MATGLB}(2) = \text{MATGLB}(2)/\text{MATGLB}(1)$
- Pour le calcul de u_{22} ,
 - $u_{22} = a_{22} - \ell_{21}u_{12}$
 - $\text{MATGLB}(4) = \text{MATGLB}(4) - \text{MATGLB}(2) \times \text{MATGLB}(3)$
 - ...

Sous-programme SLUSS

Fonction	Résolution du système d'équations algébriques associées à la matrice globale.
Utilisation	<pre>CALL SLUSS (<u>NBREQ</u>, <u>ADRMAT</u>*, <u>MATGLB</u>*, <u>MDDGLB</u>*, <u>SOLUTN</u>*)</pre>
En entrée	<p>(I) NBREQ Nombre d'équations globales</p> <p>(I) ADRMAT(*) Vecteur d'adressage de la matrice globale en fonction de la hauteur de la ligne de ciel</p> <p>(D) MATGLB(*) Matrice globale factorisée</p> <p>(D) MDDGLB(*) Membre de droite global</p>
En sortie	<p>(D) SOLUTN(*) Solution du système d'équations algébriques associées à la matrice globale</p>
Sous-programmes utilisées	Fonction DDOT de la librairie ESSL
Notes	Avant l'appel de SLUSS, le sous-programme FLUSS doit être appelé pour factoriser la matrice globale selon les matrices L et U . Le vecteur MATGLB contient les matrices L et U
Algorithme	En tenant compte de la convention de stockage de la matrice globale en ligne de ciel, le système d'équations

algébriques est résolu. Le système d'équation est

$$[A][x] = [b]$$

$$[L][U][x] = [b]$$

qui peut être écrit

$$[L][x'] = [b]$$

$$\text{avec } [U][x] = [x']$$

La descente triangulaire permet de résoudre $[L][x'] = [b]$. La montée triangulaire permet de résoudre $[U][x] = [x']$ pour déterminer le vecteur solution recherchée $[x]$ (SOLUTN)

Exemple

La descente triangulaire s'effectue pour le système d'équations suivant

$$[L][x'] = [b]$$

$$\begin{array}{|c|c|c|c|c|c|} \hline
 1 & 0 & 0 & 0 & 0 & 0 \\ \hline
 2 & & & & & \\ \hline
 \ell_{21} & 1 & 0 & 0 & 0 & 0 \\ \hline
 5 & 6 & & & & \\ \hline
 \ell_{31} & \ell_{32} & 1 & 0 & 0 & 0 \\ \hline
 0 & 0 & 10 & & & \\ \hline
 & & \ell_{43} & 1 & 0 & 0 \\ \hline
 0 & 0 & 13 & 14 & & \\ \hline
 & & \ell_{53} & \ell_{54} & 1 & 0 \\ \hline
 0 & 0 & 0 & 18 & 19 & 1 \\ \hline
 & & & \ell_{64} & \ell_{65} & \\ \hline
 0 & 0 & 0 & & & \\ \hline
 \end{array} = \begin{array}{|c|} \hline x'_1 \\ \hline x'_2 \\ \hline x'_3 \\ \hline x'_4 \\ \hline x'_5 \\ \hline x'_6 \\ \hline \end{array} = \begin{array}{|c|} \hline b_1 \\ \hline b_2 \\ \hline b_3 \\ \hline b_4 \\ \hline b_5 \\ \hline b_6 \\ \hline \end{array}$$

$$x'_1 = b_1,$$

$$\text{SOLUTN}(1) = \text{MDDGLB}(1)$$

$$\ell_{21}x'_1 + x'_2 = b_2 \rightarrow x'_2 = b_2 - \ell_{21}x'_1$$

$$\text{SOLUTN}(2) = \text{MDDGLB}(2) - \text{MATGLB}(2) \times \text{SOLUTN}(1)$$

...

La montée triangulaire du système d'équations est

$$[u][x] = [x']$$

$$\begin{array}{|c|c|c|c|c|c|} \hline
 1 & 3 & 7 & 0 & 0 & 0 \\ \hline
 u_{11} & u_{12} & u_{13} & & & \\ \hline
 0 & 4 & 8 & 0 & 0 & 0 \\ \hline
 & u_{22} & u_{23} & & & \\ \hline
 0 & 0 & 9 & 11 & 15 & 0 \\ \hline
 & & u_{33} & u_{34} & u_{35} & \\ \hline
 0 & 0 & 0 & 12 & 16 & 20 \\ \hline
 & & & u_{44} & u_{45} & u_{46} \\ \hline
 0 & 0 & 0 & 0 & 17 & 21 \\ \hline
 & & & & u_{55} & u_{56} \\ \hline
 0 & 0 & 0 & 0 & 0 & 22 \\ \hline
 & & & & & u_{66} \\ \hline
 \end{array} = \begin{array}{|c|c|} \hline
 x_1 & x'_1 \\ \hline
 x_2 & x'_2 \\ \hline
 x_3 & x'_3 \\ \hline
 x_4 & x'_4 \\ \hline
 x_5 & x'_5 \\ \hline
 x_6 & x'_6 \\ \hline
 \end{array}$$

$$u_{66}x_6 = x'_6 \rightarrow x_6 = x'_6/u_{66},$$

$$\text{SOLUTN}(6) = \text{SOLUTN}(6)/\text{MATGLB}(22)$$

$$u_{55}x_5 + u_{56}x_6 = x'_5 \rightarrow x_5 = x'_5 - u_{56}x_6$$

$$\text{SOLUTN}(5) = \text{SOLUTN}(5) - \text{MATGLB}(2) \times \text{SOLUTN}(6)$$

...

Annexe — Listing des sous-programmes LUSS

Sous-programme PRFSS

SUBROUTINE PRFSS (NBRDLE, VADELM, HLCIEL)

IMPLICIT NONE

*

*>>>>CALCUL DU PROFIL DE LA LIGNE DE CIEL DE LA MATRICE

*

*

* ALGORITHME DU SOUS-PROGRAMME

* -----

* UNE COMBINAISON LOCALE DE DEGRES DE LIBERTE DE L'ELEMENT

* (NDLI,NDLJ) CORRESPOND A UNE POSITION DANS LA MATRICE GLOBALE

* (EQI,EQJ), A L'AIDE DU VECTEUR D'ADRESSAGE VADELM.

* ETANT DONNE QUE LA MATRICE EST SYMETRIQUE DANS SA STRUCTURE,

* LE SOUS-PROGRAMME TESTE SEULEMENT LES COMBINAISONS

* APPARTENANT A LA PARTIE SUPERIEURE DE LA MATRICE GLOBALE.

* POUR CES COMBINAISONS, IL CALCULE LEUR HAUTEUR PAR RAPPORT
* A LA DIAGONALE (HLOC). IL CONSERVE LA HAUTEUR MAXIMALE POUR
* CHACUN DES INDICES DE LA MATRICE GLOBALE (HLCIEL).

*

*

* DESCRIPTION DES VARIABLES EN ARGUMENT

* -----

* - NBRDLE : NOMBRE DE DEGRES DE LIBERTE DE L'ELEMENT

*

* - VADELM() : VECTEUR D'ADRESSAGE DES DEGRES DE LIBERTE DE
* L'ELEMENT

*

* - HLCIEL() : PROFIL DE LA LIGNE DE CIEL DE LA MATRICE
* GLOBALE (HAUTEUR DE LA LIGNE DE CIEL)

*

* NOTES

* -----

* - AVANT LE PREMIER APPEL DE PRFSS, LE VECTEUR HLCIEL DOIT ETRE
* INITIALISE A 0.

*

* - LE VECTEUR HLCIEL EST MIS A JOUR A CHAQUE FOIS QUE PRFSS EST
* APPELE POUR UN ELEMENT.

*

```
*  
*      VARIABLES .. ARGUMENTS  
*  
*-----  
*      - ENTREE: NBRDLE, VADELM, HLCIEL  
*      - SORTIE: HLCIEL  
*  
*  
*      INTEGER      NBRDLE, VADELM(*), HLCIEL(*)  
*  
*      VARIABLES .. LOCALES  
*  
*-----  
*  
*      INTEGER      NDLI , NDLJ , EQI , EQJ , HLOC  
  
DO 1001 NDLI = 1, NBRDLE  
    EQI = VADELM(NDLI )  
    IF (EQI.GT.0) THEN  
        DO 1000 NDLJ = 1, NBRDLE  
            EQJ = VADELM(NDLJ )  
            IF (EQJ.GT.0) THEN  
                IF (EQJ.LT.EQI) THEN  
                    HLOC      = EQI - EQJ  
                    HLCIEL(EQI) = MAX(HLOC,HLCIEL(EQI))  
                END IF  
            END IF  
    END IF  
1000    CONTINUE  
END IF
```

1001 CONTINUE

END

Sous-programme ADRSS

SUBROUTINE ADRSS (NBREQ , HLCIEL, ADRMAT, ESPMAT)

IMPLICIT NONE

*

*>>>>CALCUL DU VECTEUR D'ADRESSAGE DE LA MATRICE EN FONCTION DE
* LA HAUTEUR DE LA LIGNE DE CIEL

*

*

* ALGORITHME DU SOUS-PROGRAMME

* -----

* EN CONNAISSANT LA HAUTEUR DE LA LIGNE DE CIEL POUR TOUS LES
* INDICES DE LA MATRICE GLOBALE, LA LONGUEUR DES BRAS PEUT ETRE
* CALCULEE. UN BRAS EST COMPOSE PAR LES TERMES INFERIEURS, PAR
* LES TERMES SUPERIEURS ET PAR LE TERME DIAGONAL. A L'AIDE DE
* AGBSS, LA MATRICE GLOBALE SERA STOCKEE DANS UN VECTEUR. LE
* VECTEUR CONTIENT CONSECUTIVEMENT TOUS LES BRAS, EN INDICE
* CROISSANT. LE SOUS-PROGRAMME ADRSS PERMET DE CALCULER LA
* POSITION DE L'ELEMENT DIAGONAL DE CHACUN DES BRAS. CHACUNE
* DES POSITIONS EST STOCKEE DANS ADRMAT.

*

*

* DESCRIPTION DES VARIABLES EN ARGUMENT

* -----

* - NBREQ : NOMBRE D'EQUATIONS GLOBALES

*

* - HLCIEL() : PROFIL DE LA LIGNE DE CIEL DE LA MATRICE GLOBALE

* (HAUTEUR DE LA LIGNE DE CIEL)

*

* - ADRMAT() : VECTEUR D'ADRESSAGE DE LA MATRICE GLOBALE EN

* FONCTION DE LA HAUTEUR DE LA LIGNE DE CIEL

*

* - ESPMAT : LONGUEUR EFFECTIVE UTILISEE DU VECTEUR MATGLB

*

*

*

* NOTES

*

* -----

* - AVANT L'APPEL DE ADRSS, PRFSS DOIT ETRE APPELE POUR CHACUN

* DES ELEMENTS QUI COMPOSENT LA MATRICE GLOBALE.

*

*

*

* VARIABLES .. ARGUMENTS

*

* -----

* - ENTREE: NBREQ , HLCIEL

* - SORTIE: ADRMAT, ESPMAT

*

INTEGER NBREQ , HLCIEL(*), ADRMAT(*), ESPMAT

*

* VARIABLES .. LOCALES

```
* -----
*
INTEGER      NUMEQ

ADRMAT(1) = 1
DO 1000 NUMEQ = 2, NBREQ
  ADRMAT(NUMEQ) = ADRMAT(NUMEQ-1) + 2 * HLCIEL(NUMEQ) + 1
1000 CONTINUE
ESPMAT = ADRMAT(NBREQ)
END
```

Sous-programme AGBSS

SUBROUTINE AGBSS (MAXDLE, NBRDLE, VADELM, ADRMAT, MATELM, MATGLB)

IMPLICIT NONE

*

*>>>>ASSEMBLAGE DE LA MATRICE ELEMENTAIRE DANS LA MATRICE GLOBALE

*

*

* ALGORITHME DU SOUS-PROGRAMME

* -----

* UNE COMBINAISON LOCALE DE DEGRES DE LIBERTE DE L'ELEMENT

* (I,J) CORRESPOND A UNE POSITION DANS LA MATRICE GLOBALE

* (EQI,EQJ), A L'AIDE DU VECTEUR D'ADRESSAGE VADELM. IL Y A
* TROIS CAS POSSIBLES DE POSITION D'UN TERME DANS LA MATRICE
* GLOBALE.

* - 1. CAS : LE TERME APPARTIENT A LA DIAGONALE

* POSITION (IJ) = POSITION DU TERME DIAGONAL DU BRAS

* CORRESPONDANT

*

* - 2. CAS : LE TERME APPARTIENT A LA PARTIE INFERIEURE

* POSITION (IJ) = POSITION DU PREMIER TERME SUPERIEUR

* DU BRAS CORRESPONDANT

* + DIFFERENCE DE POSITION PAR RAPPORT

* A LA DIAGONALE

*

* - 3. CAS : LE TERME APPARTIENT A LA PARTIE SUPERIEURE
* POSITION (IJ) = POSITION DU TERME DIAGONAL
* + DIFFERENCE DE POSITION PAR RAPPORT
* A LA DIAGONALE
*
*
* DESCRIPTION DES VARIABLES EN ARGUMENT
* -----
* - MAXDLE : DIMENSION MAXIMALE DE LA MATRICE ELEMENTAIRE
*
* - NBRDLE : NOMBRE DE DEGRES DE LIBERTE DE L'ELEMENT
*
* - VADELM() : VECTEUR D'ADRESSAGE DES DEGRES DE LIBERTE
DE L'ELEMENT
*
* - ADRMAT() : VECTEUR D'ADRESSAGE DE LA MATRICE GLOBALE EN
FONCTION DE LA HAUTEUR DE LA LIGNE DE CIEL
*
* - MATELM(,) : MATRICE ELEMENTAIRE
*
* - MATGLB() : MATRICE GLOBALE
*
*
* NOTES
* -----

* - AVANT LE PREMIER APPEL DE AGBSS, LE VECTEUR MATGLB DOIT ETRE
* INITIALISE A 0.000.

*

* - LE VECTEUR MATGLB EST MIS A JOUR A CHAQUE FOIS QUE AGBSS EST
* APPELE POUR UN ELEMENT.

*

* - AVANT L'APPEL DE AGBSS, LE SOUS-PROGRAMME ADRSS DOIT ETRE
* APPELE POUR DEFINIR LE VECTEUR D'ADRESSAGE ADRMAT DE LA
* MATRICE GLOBALE.

*

*

* VARIABLES ... ARGUMENTS

* -----

* - ENTREE: MAXDLE, NBRDLE, VADELM, ADRMAT, MATELM, MATGLB
* - SORTIE: MATGLB

*

INTEGER MAXDLE, NBRDLE, VADELM(*), ADRMAT(*)

DOUBLE PRECISION MATELM(MAXDLE,*), MATGLB(*)

*

* VARIABLES .. LOCALES

* -----

*

INTEGER I, J, IJ, EQI, EQJ

DO 1001 I = 1, NBRDLE

EQI = VADELM(I)

```
IF (EQI.GT.0) THEN
  DO 1000 J = 1 , NBRDLE
    EQJ = VADELM(J)
    IF (EQJ.GT.0) THEN
*
*      ADRESSAGE DU COEFFICIENT DE LA MATRICE ELEMENTAIRE
*      DANS LA MATRICE GLOBALE
*
*      -----
*
*      IF (EQI.EQ.EQJ) THEN
*        IJ      = ADRMAT(EQI)
*      ELSE IF (EQI.GT.EQJ) THEN
*        IJ      = (ADRMAT(EQI)+ADRMAT(EQI-1)+1)/2 + EQJ-EQI
*      ELSE IF (EQI.LT.EQJ) THEN
*        IJ      = ADRMAT(EQJ) + EQI-EQJ
*      END IF
*      MATGLB(IJ) = MATGLB(IJ) + MATELM(I,J)
*    END IF
1000  CONTINUE
  END IF
1001 CONTINUE
END
```

Sous-programme FLUSS

SUBROUTINE FLUSS (NBREQ , ADRMAT, MATGLB, MSGERR)

IMPLICIT NONE

*

*>>>> FACTORISATION LU

*

*

* ALGORITHME DU SOUS-PROGRAMME

* -----

* L'ALGORITHME DE CE SOUS-PROGRAMME EST CELUI D'UNE FACTORISATION
* LU D'UNE MATRICE. LA MATRICE L CORRESPOND A LA MATRICE
* TRIANGULAIRE INFÉRIEURE AYANT DES TERMES UNITAIRES SUR SA
* DIAGONALE. LA MATRICE U CORRESPOND A LA MATRICE TRIANGULAIRE
* SUPÉRIEURE. CETTE DECOMPOSITION PERMET LE STOCKAGE DES
* MATRICES L ET U DANS LE VECTEUR MATGLB, SELON LE MEME VECTEUR
* D'ADRESSAGE DE LA LIGNE DE CIEL ADRMAT. LA PARTICULARITÉ DE
* CETTE FACTORISATION LU, C'EST QU'ELLE PROCEDE EN TENANT COMPTE
* DE LA STRUCTURE DE LA MATRICE GLOBALE, QUI EST STOCKÉE EN LIGNE
* DE CIEL.

*

*

* DESCRIPTION DES VARIABLES EN ARGUMENT

* -----

* - NBREQ : NOMBRE D'EQUATIONS GLOBALES

*

* - ADRMAT() : VECTEUR D'ADRESSAGE DE LA MATRICE GLOBALE EN

* FONCTION DE LA HAUTEUR DE LA LIGNE DE CIEL

*

* - MSGERR : UNITE LOGIQUE DE SORTIE DES MESSAGES D'ERREURS

* ET INDICATEUR DE MESSAGES D'ERREURS, SI MSGERR>0.

* SI MSGERR=0, IL N'Y A PAS DE MESSAGE. IL Y A DES

* MESSAGES D'ERREURS SI UN OU PLUSIEURS TERMES

* DIAGONAUX SONT PLUS PETITS QUE 1.OE-08.

*

* - MATGLB() : MATRICE GLOBALE

*

*

* NOTES

*

* -----

* - AVANT L'APPEL DE FLUSS, AGBSS DOIT ETRE APPELE POUR CHACUN

* DES ELEMENTS QUI COMPOSENT LA MATRICE GLOBALE.

*

* - EN ENTREE, LE VECTEUR MATGLB REPRESENTE LA MATRICE GLOBALE.

* EN SORTIE, IL REPRESENTE LA MATRICE GLOBALE FACTORISEE.

* LES MATRICES L ET U REMPLACENT LA MATRICE GLOBALE.

*

* - LORS DE LA FACTORISATION, IL N'Y A PAS DE PIVOTAGE DES LIGNES

* DE LA MATRICE POUR EVITER LES PIVOTS NULS. SI LE PIVOT EST

* PLUS PETIT QUE 1.OE-08, LE PIVOT EST REMPLACE PAR LA VALEUR

* 1.OE-08, TOUT EN CONSERVANT LE SIGNE DU PIVOT.

*

*

* DESCRIPTION DES VARIABLES LOCALES

*

* - I : NUMERO DU BRAS I , LONGUEUR MAXIMALE DE LA RANGEE

* - ID : POSITION DU TERME DIAGONAL DU BRAS I

* - IL : POSITION DU PREMIER TERME INFERIEUR DU BRAS I

* - IU : POSITION DU PREMIER TERME SUPERIEUR DU BRAS I

* - LONGI : HAUTEUR DE LA LIGNE DE CIEL DU BRAS I

* - IMIN : POSITION DU PREMIER ELEMENT NON NUL DANS LA PARTIE
* INFERIEURE DU BRAS I (RANGEE)

* - J : NUMERO DU BRAS J

* - JD : POSITION DU TERME DIAGONAL DU BRAS J

* - JL : POSITION DU PREMIER TERME INFERIEUR DU BRAS J

* - JU : POSITION DU PREMIER TERME SUPERIEUR DU BRAS J

* - LONGJ : HAUTEUR DE LA LIGNE DE CIEL DU BRAS J

* - LONG : HAUTEUR MINIMUM DE LA LIGNE DE CIEL (BRAS I OU BRAS J)

*

*

* VARIABLES .. ARGUMENTS

*

* - ENTREE: NBREQ , ADRMAT, MATGLB, MSGERR

* - SORTIE: MATGLB

*

```
INTEGER          NBREQ , ADRMAT(*), MSGERR
DOUBLE PRECISION MATGLB(*)  
  
*  
*      VARIABLES .. LOCALES
*      -----
*  
*      INTEGER          I      ,J      ,IJ
*      INTEGER          ID     ,IL     ,IU     ,LONGI ,IMIN
*      INTEGER          JD     ,JL     ,JU     ,LONGJ
*      INTEGER          DEBI   ,DEBJ   ,LONG
*      DOUBLE PRECISION EPSLU ,LUDAG
*      PARAMETER        (EPSLU=1.0D-08)
*  
*      FONCTIONS .. ESSL
*      -----
*  
*      DOUBLE PRECISION DDOT
*  
*  
DO 1001 I = 2 , NBREQ
*      ID    = ADRMAT(I  )
*      IL    = ADRMAT(I-1)+ 1
*      IU    = ( ID + IL ) / 2
*      LONGI = IU - IL
*      IMIN = I - LONGI
*  
*      CALCUL DU PREMIER TERME INFERIEUR S'IL NE S'AGIT PAS DU
```

```
*      TERME DIAGONAL DU BRAS

*
IF (IMIN.NE.I) MATGLB(IL) = MATGLB(IL)/MATGLB(ADRMAT(IMIN))

*
BOUCLE DU DEUXIEME TERME DE LA PARTIE INFERIEURE AU DERNIER
TERME DE LA PARTIE INFERIEURE

*
DO 1000 J = IMIN+1 , I-1
    JD      = ADRMAT(J  )
    JL      = ADRMAT(J-1)+ 1
    JU      = ( JD + JL ) / 2
    LONGJ = JU - JL
    LONG   = MIN(J-IMIN, LONGJ)

*
*      CALCUL DE L(I,J) POUR I DIFFERENT DE J
*
IJ      = IL + J - IMIN
DEBI   = IJ - LONG
DEBJ   = JD - LONG
MATGLB(IJ) = (MATGLB(IJ)-
& DDOT(LONG,MATGLB(DEBI),1,MATGLB(DEBJ),1))/MATGLB(JD)

*
*      CALCUL DE U(J,I) POUR I DIFFERENT DE J
*
IJ      = IU + J - IMIN
DEBI   = IJ - LONG
DEBJ   = JU - LONG
```

```
        MATGLB(IJ) = MATGLB(IJ)-
& DDOT(LONG,MATGLB(DEBI),1,MATGLB(DEBJ),1)
1000  CONTINUE
*
*      CALCUL DE U(I,J) POUR I EGAL A J ET DETECTION DES PIVOTS
*      TROP PETITS
*
LUDAG = MATGLB(ID) - DDOT(LONGI,MATGLB(IL),1,MATGLB(IU),1)
IF ( DABS(LUDAG) .LT. EPSLU) THEN
    MATGLB(ID) = SIGN(EPSLU,LUDAG)
    IF (MSGERR.GT.0) WRITE(MSGERR,9000) LUDAG, I, MATGLB(ID)
    ELSE
        MATGLB(ID) = LUDAG
    ENDIF
1001  CONTINUE
9000  FORMAT(T1,'*** AVERTISSEMENT *** FLUSS - PIVOT =',G16.8,
&           'A L''EQUATION ',I10, 'REPLACE PAR ',G16.8)
END
```

Sous-programme SLUSS

SUBROUTINE SLUSS (NBREQ , ADRMAT, MATGLB, MDDGLB, SOLUTN)

IMPLICIT NONE

*

* ALGORITHME DU SOUS-PROGRAMME

* -----

* EN TENANT COMPTE DE LA CONVENTION DE STOCKAGE DE LA MATRICE
* GLOBALE EN LIGNE DE CIEL, LE SYSTEME D'EQUATIONS ALGEBRIQUES
* EST RESOLU. LE SYSTEME D'EQUATIONS EST

* $[A] [X] = [B]$,

* $[L] [U] [X] = [B]$,

* QUI PEUT ETRE ECRIT

* $[L] [X'] = [B]$

* AVEC $[U] [X] = [X']$

* LA DESCENTE TRIANGULAIRE PERMET DE RESOUDRE $[L] [X'] = [B]$.

* LA MONTEE TRIANGULAIRE PERMET DE RESOUDRE $[U] [X] = [X']$ POUR
* DETERMINER LE VECTEUR SOLUTION RECHERCHE $[X]$ (SOLUTN).

*

*

* DESCRIPTION DES VARIABLES EN ARGUMENT

* -----

* - NBREQ : NOMBRE D'EQUATIONS GLOBALES

*

* - ADRMAT() : VECTEUR D'ADRESSAGE DE LA MATRICE GLOBALE EN

* FONCTION DE LA HAUTEUR DE LA LIGNE DE CIEL.

*

* - MATGLB() : MATRICE GLOBALE FACTORISEE

* - MDDGLB() : MEMBRE DE DROITE GLOBAL

* - SOLUTN() : SOLUTION DU SYSTEME D'EQUATIONS ALGEBRIQUES

* ASSOCIEES A LA MATRICE GLOBALE.

*

*

* NOTES

* -----

* - AVANT L'APPEL DE SLUSS, LE SOUS-PROGRAMME FLUSS DOIT ETRE
* APPELE POUR FACTORISER LA MATRICE GLOBALE SELON LES MATRICES
* L ET U.

*

* - LE VECTEUR MATGLB CONTIENT LES MATRICES L ET U.

*

*

* VARIABLES .. ARGUMENTS

* -----

* - ENTREE: NBREQ , ADRMAT, MATGLB, MDDGLB
* - SORTIE: SOLUTN

*

INTEGER NBREQ , ADRMAT(*)

DOUBLE PRECISION MATGLB(*), MDDGLB(*), SOLUTN(*)

*

* VARIABLES .. LOCALES

* -----

*

INTEGER I, J, IL, JD, LONGI, LONGJ
DOUBLE PRECISION XJ

*

* FONCTIONS .. ESSL

* -----

*

DOUBLE PRECISION DDOT

*

*>>>>DESCENTE TRIANGULAIRE

*

SOLUTN(1) = MDDGLB(1)

DO 1000 I = 2, NBREQ

IL = ADRMAT(I-1)+ 1

LONGI = (ADRMAT(I) - IL) / 2

SOLUTN(I) = MDDGLB(I)

& - DDOT(LONGI,MATGLB(IL),1,SOLUTN(I-LONGI),1)

1000 CONTINUE

*

*>>>>REMONTREE TRIANGULAIRE

```
*  
DO 1002 J = NBREQ , 2 , -1  
    JD      = ADRMAT(J  )  
    LONGJ = (JD-ADRMAT(J-1)-1) / 2  
    XJ      = SOLUTN(J) / MATGLB(JD)  
    DO 1001 I = J-LONGJ , J-1  
        SOLUTN(I) = SOLUTN(I) - MATGLB(JD+I-J) * XJ  
1001    CONTINUE  
        SOLUTN(J) = XJ  
1002    CONTINUE  
        SOLUTN(1) = SOLUTN(1) / MATGLB(ADRMAT(1))  
END
```

ÉCOLE POLYTECHNIQUE DE MONTRÉAL



3 9334 00223419 1