

## Implémentation d'une méthode d'éléments finis

### T.P.3 : Assemblage

On écrit au cours de ce T.P. une fonction qui réalise l'assemblage des matrices et des seconds membres élémentaires. En parallèle, on construit les tableaux permettant la gestion des conditions de Dirichlet.

#### 1. Description des structures de données

La matrice notée  $A$  résultant de l'assemblage est construite et mémorisée selon la **Structure Morse Désordonnée (SMD)** décrite ci-dessous. Compte tenu de la symétrie de la matrice, seuls les coefficients non logiquement nuls <sup>(1)</sup> de la partie triangulaire inférieure, diagonale comprise, sont mémorisés à l'aide des variables et tableaux à une dimension (vecteurs) suivants :

- **NbLign**, nombre de lignes et de colonnes de la matrice  $A$  ;
- **NbCoef**, nombre de coefficients mémorisés de la partie triangulaire inférieure stricte de  $A$  ;
- **Matrice**, tableau (de type réel, de longueur  $\geq \text{NbLign} + \text{NbCoef}$ ) contenant les coefficients de la diagonale suivis des coefficients non logiquement nuls de la partie triangulaire inférieure stricte de  $A$  ; on désignera par **DiagMat** la première partie de **Matrice** (contenant les coefficients diagonaux), et par **LowMat** la deuxième partie (contenant les coefficients de la partie triangulaire inférieure stricte) ;
- **AdPrCoefLi**, tableau (de type entier, de longueur  $\geq \text{NbLign}$ ) contenant, pour chaque ligne de la partie triangulaire inférieure stricte de  $A$ , l'adresse (i.e. indice ou position dans **LowMat**) du premier coefficient non logiquement nul de cette ligne ; **AdPrCoefLi(I-1)** est l'adresse (la position dans **LowMat**) du premier élément stocké de la ligne  $I$  de  $A$  et **AdPrCoefLi(NbLign)=NbCoef+1** ; de plus, **AdPrCoefLi(I-1) = 0** si tous les éléments de la ligne  $I$  sont nuls jusqu'à l'élément diagonal non compris ;
- **AdSuccLi**, tableau (de type entier, de longueur  $\geq \text{NbCoef}$ ) contenant, pour chaque coefficient de la partie triangulaire inférieure stricte de  $A$ , l'adresse (position dans **LowMat**) du coefficient suivant sur la même ligne, ou 0 si le coefficient est le dernier mémorisé sur la ligne ;
- **NumCol**, tableau (de type entier, de longueur  $\geq \text{NbCoef}$ ) contenant, pour chaque coefficient de la partie triangulaire inférieure stricte de  $A$ , le numéro de colonne du coefficient.

Le second membre assemblé est mémorisé dans le tableau **SecMembre** (type réel, de longueur  $\geq \text{NbLign}$ ).

Les numéros des degrés de liberté de Dirichlet sont mémorisés dans le tableau **NumDLDir** et permettront de repérer ultérieurement de tels d.d.l. ; les valeurs associées aux conditions de Dirichlet en ces d.d.l. sont mémorisées dans le tableau **ValDLDir**.

#### 2. Assemblage de la matrice

On fournit la procédure Fortran **assmat\_** qui, étant donné un couple de numéros globaux de d.d.l. ( $I, J$  avec  $I > J$ ) d'un même élément, la valeur  $X$  du coefficient correspondant de la matrice élémentaire et un "pointeur", ici un indice – **NextAd** – dans le tableau **LowMat** indiquant la prochaine "adresse" (emplacement) disponible,

- crée le coefficient de la colonne  $J$  de la ligne  $I$  de la matrice  $A$  dans le tableau **LowMat** à l'adresse indiquée par **NextAd** si ce coefficient n'existe pas et lui affecte la valeur  $X$  (" $A_{IJ} = X$ ") ;
- incrémente de la valeur  $X$  le coefficient de la colonne  $J$  de la ligne  $I$  si ce coefficient existe déjà dans **LowMat** (" $A_{IJ} = A_{IJ} + X$ ").

<sup>(1)</sup> Dans le cadre des éléments finis, le coefficient  $A_{IJ}$  de la matrice  $A$  est non logiquement nul si les degrés de liberté de numéros globaux  $I$  et  $J$  sont des d.d.l. d'un même élément (cf. cours).

Plus précisément, cette procédure réalise les étapes suivantes :

- si la ligne  $I$  n'a jamais été rencontrée lors du parcours des d.d.l. précédents, on crée dans **LowMat** le premier coefficient de la ligne  $I$  à la prochaine adresse disponible (indiquée par **NextAd**), on affecte la valeur  $X$  au coefficient  $A_{IJ}$  (avec  $I > J$ ) et on actualise **NextAd** ;
- si la ligne  $I$  a déjà été rencontrée et si ce coefficient de la colonne  $J$  n'a jamais été rencontré, on crée dans **LowMat** le coefficient suivant de la ligne  $I$  à la prochaine adresse disponible (indiquée par **NextAd**), on affecte la valeur  $X$  au coefficient  $A_{IJ}$  (avec  $I > J$ ) et on actualise **NextAd** ;
- si la ligne  $I$  a déjà été rencontrée et si ce coefficient de la colonne  $J$  y est trouvé, le coefficient  $A_{IJ}$  est incrémenté de la valeur  $X$ .

Les coefficients diagonaux ( $I=J$ ) ne sont pas traités par **assmat** ; ils devront être assemblés directement dans la partie **DiagMat** du tableau **Matrice**.

### 3. Prise en compte des conditions de Dirichlet

Les tableaux globaux de description des conditions de Dirichlet seront définis comme suit, à partir des tableaux **NuDElem** et **uDElem** relatif à chaque élément :

- **NumDLDir(I)**= $-I$  si le nœud  $I$  appartient à un bord portant une condition de Dirichlet non homogène et dans ce cas **ValDLDir(I)** contient la valeur de la donnée de Dirichlet en ce d.d.l. ;
- **NumDLDir(I)**= 0 si le nœud  $I$  appartient à un bord portant une condition de Dirichlet homogène ;
- **NumDLDir(I)**=  $I$  sinon.

### 4. Procédure d'assemblage

1°) Écrire la procédure **Assemblage** qui réalise l'assemblage de la matrice  $A$  (tableau **Matrice**) selon la **Structure Morse Désordonnée**, l'assemblage du second membre **SecMembre** et produit les tableaux nécessaires au traitement de conditions de Dirichlet **NumDLDir** et **ValDLDir**. Cette procédure effectue donc les tâches suivantes :

- initialisation des tableaux utiles,
- lors d'un parcours de tous les éléments du maillage, appel de la procédure **cal1Elem**, assemblage des matrice et second membre élémentaires par appel à **assmat**, assemblage de la diagonale et remplissage des tableaux décrivant les conditions de Dirichlet **NumDLDir** et **ValDLDir**.

2°) Écrire la procédure **EcrSMD** qui sauvegarde la **Structure Morse Désordonnée**, constituée de la matrice et des vecteurs associés, sous la forme d'un fichier séquentiel **binaire** contenant les enregistrements suivants :

- ① **NbLign**
- ② **SecMembre** (de taille **NbLign**)
- ③ **NumDLDir** (de taille **NbLign**)
- ④ **ValDLDir** (de taille **NbLign**)
- ⑤ **AdPrCoefLi** (de taille **NbLign**)
- ⑥ **Matrice** (de taille **NbLign+NbCoef**)
- ⑦ **NumCol** (de taille **NbCoef**)
- ⑧ **AdSuccLi** (de taille **NbCoef**)

3°) Écrire la procédure **LecSMD** qui relit le fichier construit en 2°). On rappelle à ce sujet que **NbCoef** = **AdPrCoefLi(NbLign)**-1. A l'aide de la procédure Fortran **affsmd** fournie, afficher la matrice  $A$  calculée. La procédure **affsmd** affiche à la demande une ou plusieurs lignes consécutives de la partie triangulaire inférieure de la matrice  $A$ , du second membre et des tableaux de description des conditions de Dirichlet.

Nota : Pour les entrées-sorties binaires utiliser les fonctions **fread** et **fwrite** (taper **man fread** pour avoir le mode d'emploi). L'ouverture du fichier se fait toujours à l'aide de **fopen**.

## 5. Procédure principale et dimensionnement des tableaux

Le programme principal correspondant à ce TP pourra se déduire par enrichissement de celui du TP 2, et qui mettra en œuvre ces procédures, par exemple par ajout des actions suivantes dans son menu :

ASSEMBLER LE SYSTÈME  
AFFICHER LE SYSTÈME ASSEMBLÉ  
QUITTER

Les procédures d'entrées-sortie **EcrSMD** et **LecSMD** sont réciproques l'une de l'autre et permettent simplement de s'assurer que les valeurs calculées sont correctes. Elles ont pour but de valider les étapes qui conduisent à la construction du système linéaire final.

⚠ La description faite dans cet énoncé utilise implicitement les indices de ligne et de colonne naturels, mathématiques, qui commencent à 1 (et qui sont aussi ceux utilisés en Fortran). Il ne faut pas perdre de vue que le numéro d'une ligne correspond au numéro d'un nœud dans la numérotation globale, qui provient du maillage lu par la fonction **lecfima** et cette numérotation commence à 1. Afin d'assurer la cohérence de l'ensemble, il y a donc lieu dans la procédure **Assemblage** écrite en langage C, de décaler certains indices et la variable **NextAd** doit quant à elle être initialisée à 1.

⚠ Les dimensions maximales des tableaux **Matrice**, **NumCol**, **AdSuccLi**, **AdPrCoefLi**, **SecMembre**, **NumDLDir**, **ValDLDir** peuvent être calculées assez précisément en fonction des valeurs maximales de **N1** et **N2** (cf. TP 1) si on n'utilise pas l'allocation dynamique.

⚠ Si on utilise l'allocation dynamique, on peut exploiter les informations issues du fichier de maillage (valeur de **NbLign** par exemple). Mais même dans ce cas, on est amené à surdimensionner les tableaux, notamment **Matrice**, **NumCol** et **AdSuccLi** car leur contenu exact dépend du maillage et n'est connu qu'une fois cette information entièrement exploitée. On peut (sur)estimer la taille nécessaire en comptant, pour un nœud donné, le nombre de contributions non nulles avec des nœuds de l'élément courant et de ses voisins. Pour cette raison, dans la boucle d'assemblage, penser à contrôler, à l'aide de **NextAd**, qu'il n'y a pas de débordement de tableau.

⚠ A propos de **NbCoef**.  
Ce nombre peut avoir été passé sous silence en cours car cette information n'est pas nécessaire pour définir la SMD. Pour la même raison, il n'est pas non plus nécessaire pour afficher le contenu de la SMD, ni pour construire la SMO à partir de la SMD ; il suffit de consulter les fonctions **affsmd\_** et **cdesse\_** pour le voir (ici, on anticipe car cette dernière fonction intervient dans le TP suivant relatif à la SMO).

Par contre, dans le cadre de la mise en œuvre informatique proposée dans le cadre de ces TPs, et en particulier pour la création du fichier de stockage de la SMD, il est nécessaire de connaître ce nombre. Celui-ci étant connu à l'issue de l'assemblage, il est judicieux de le mémoriser dans le dernier élément du tableau **AdPrCoefLi**, inutilisé autrement. C'est ce qui est indiqué au début de l'énoncé.

En dehors de la création du fichier de stockage de la SMD, il peut malgré tout avoir une utilité pour dimensionner les tableaux de la SMO par allocation dynamique, car c'est un majorant du "NbCoef de la SMO" pour le dire rapidement, mais ne doit surtout pas être confondu avec lui car ils ne sont égaux qu'en l'absence de condition de Dirichlet. Cette dernière remarque concerne le TP suivant, mais est faite par anticipation pour regrouper les explications portant sur le même sujet.

### Nota

Les différents fichiers fournis sont situés dans le répertoire `/usr/local/anamum/csmef/tp3`. En particulier, le fichier **forfun.h** contient les prototypes qui correspondent aux sous-programmes Fortran fournis, relatifs à ce T.P. et aux suivants.