Chapitre III

Génération automatique

de maillage

Référence: P.L. George: *Génération automatique de maillages*, RMA 16, Masson, Paris.

Le chapitre III concerne la génération automatique de maillages à l'aide de Modulef.

Nous utilisons dans Modulef les deux programmes **apnoxx** pour appeler les mailleurs 2D et **apn3xx** pour appeler les mailleurs 3D.

Nous allons aborder les points suivants:

1. méthodologie générale
2. maillage des ensembles primaux
3. structuration de l'information dans Modulef
4. arbre du module APNOPO
5. aspect graphique de Modulef

III.1. Introduction

Soit $\Omega \subset \mathcal{R}^n$, $n = 1, 2, 3$, un ouvert borné. Il s'agit de décomposer $\overline{\Omega}$ en un ensemble de sous-domaines

1. segments
2. triangles ou quadrangles
3. tétraèdres ou pentaèdres ou hexaèdres.

On impose

- $\overline{\Omega} = \cup_{T \in \mathcal{T}_h} T$

- $T \cap T' = \emptyset$ ou sommet
  ou côté ou face

- pas d'angles trop grands.

En général, on n'a pas $\overline{\Omega} = \cup_{T \in \mathcal{T}_h} T$, mais

$$\overline{\Omega_h} = \bigcup_{T \in \mathcal{T}_h} \simeq \overline{\Omega}.$$

En fait nous allons utiliser chaque fois que cela est nécessaire des éléments finis isoparamétriques.

III.2. Méthodologie générale

Schématiquement nous adoptons la méthode :

analyse descendante et
réalisation ascendante.

1. On décompose le domaine à mailler en sous-ensembles (**les ensembles primaux**) faciles à mailler:

   suivant des propriétés géométriques,
   suivant des propriétés physiques.

2. On effectue alors une réalisation ascendante:

   en maillant les ensembles primaux,
   en recollant les maillages des ensembles primaux à l'aide d'utilitaires.

Illustrons cette méthodologie sur un exemple.

Nous voulons mailler la partie supérieure d'un réducteur homocinétique.

Étape 1. Les ensembles primaux à mailler sont des domaines bidimensionnels simples. N.B. Nous savons par translation passer d'un maillage 2D à un maillage 3D.

Étape 2.

1. On maille les ensembles primaux à l'aide de QUAC pour le sous-domaine 2 et de TRIH pour le sous-domaine 1.

2. On recolle.

3. On passe de la dimension 2 à la dimension 3 par translation des ensembles primaux.

4. Par symétrie, rotation et recollement, on obtient finalement la pièce.

III.3. Maillage des ensembles primaux

Nous allons détailler les concepts et les techniques qui permettent la création du maillage des parties primales.

a) Les points.

Ce sont *les éléments de plus bas niveau* dans la *description arborescente* du maillage.

point: point_numéro référence coordonnées

b) Les lignes.

Ce sont les domaines monodimensionnels qui relient des points.

ligne: ligne_numéro nombre_de_points extr1
extr2 référence type_de_ligne raison

Le type de la ligne est caractérisé par un numéro: 0 signifie segment, 2 arc de cercle par l'origine, 10 fonction interprétée.

c) Les faces.

Ce sont les domaines bidimensionnels qui se construisent à partir des lignes.

face: face_numéro lignes_contour référence

À une face nous rattachons aussi un maillage.

Passons en revue les différentes techniques.

Génération manuelle

Méthode fastidieuse mais utile pour des maillages grossiers.

Mot-clé: **MANU**.

Génération par subdivision

À partir d'un maillage grossier, on subdivise les éléments en éléments d'un type donné.

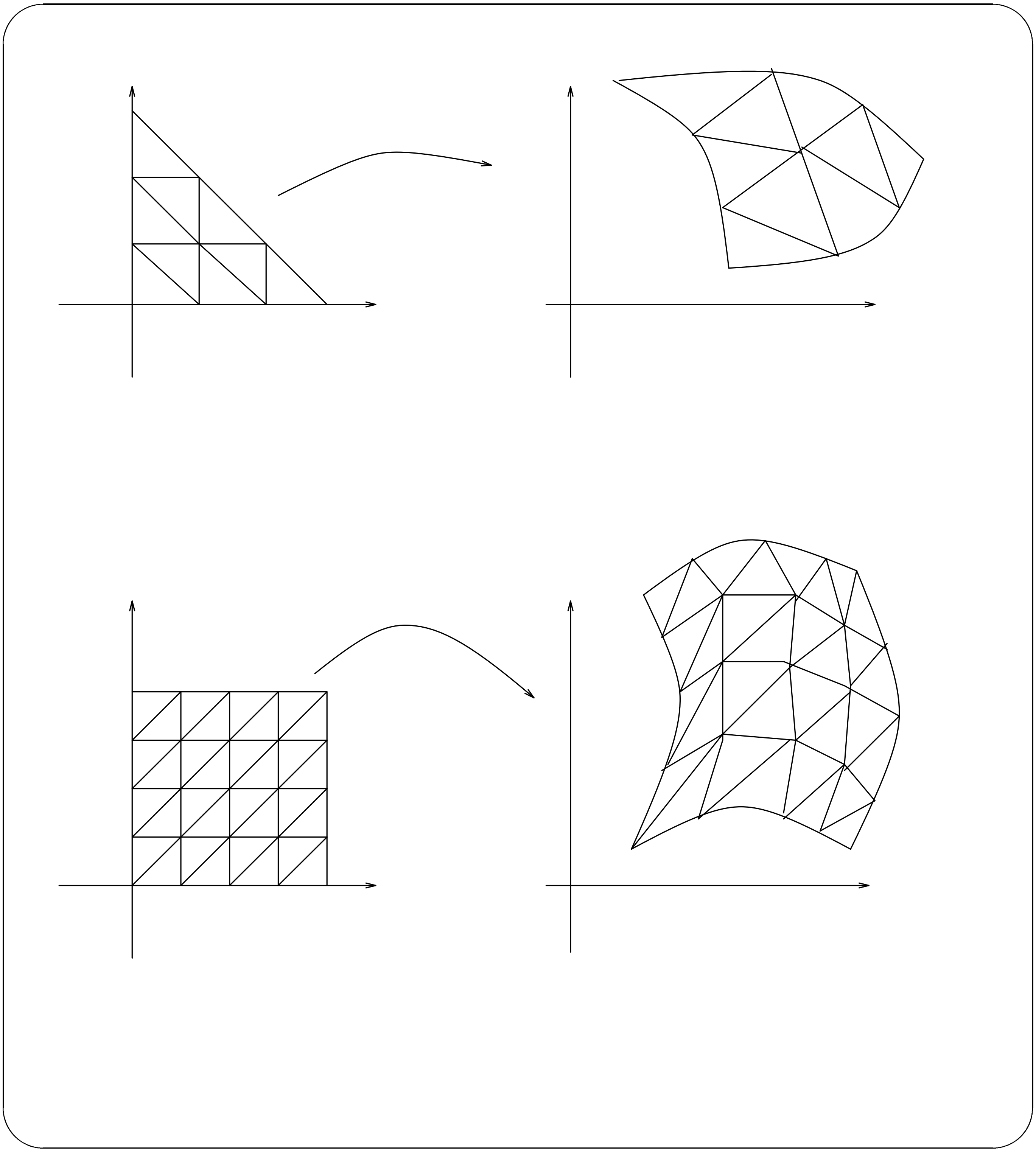
Mot-clé: **RETR** découpe chaque élément du maillage en sous-éléments du même type.

N.B. Dans Modulef on trouve des outils pour découper un quadrangle en quatre triangles...

Génération par transformation

La méthode est simple et elle permet de mailler raisonnablement des domaines élémentaires “équivalents” à des triangles ou des rectangles. Elle n’est pas très fiable, car elle peut donner des résultats absurdes.

Les maillages obtenus sont des **maillages structurés**.



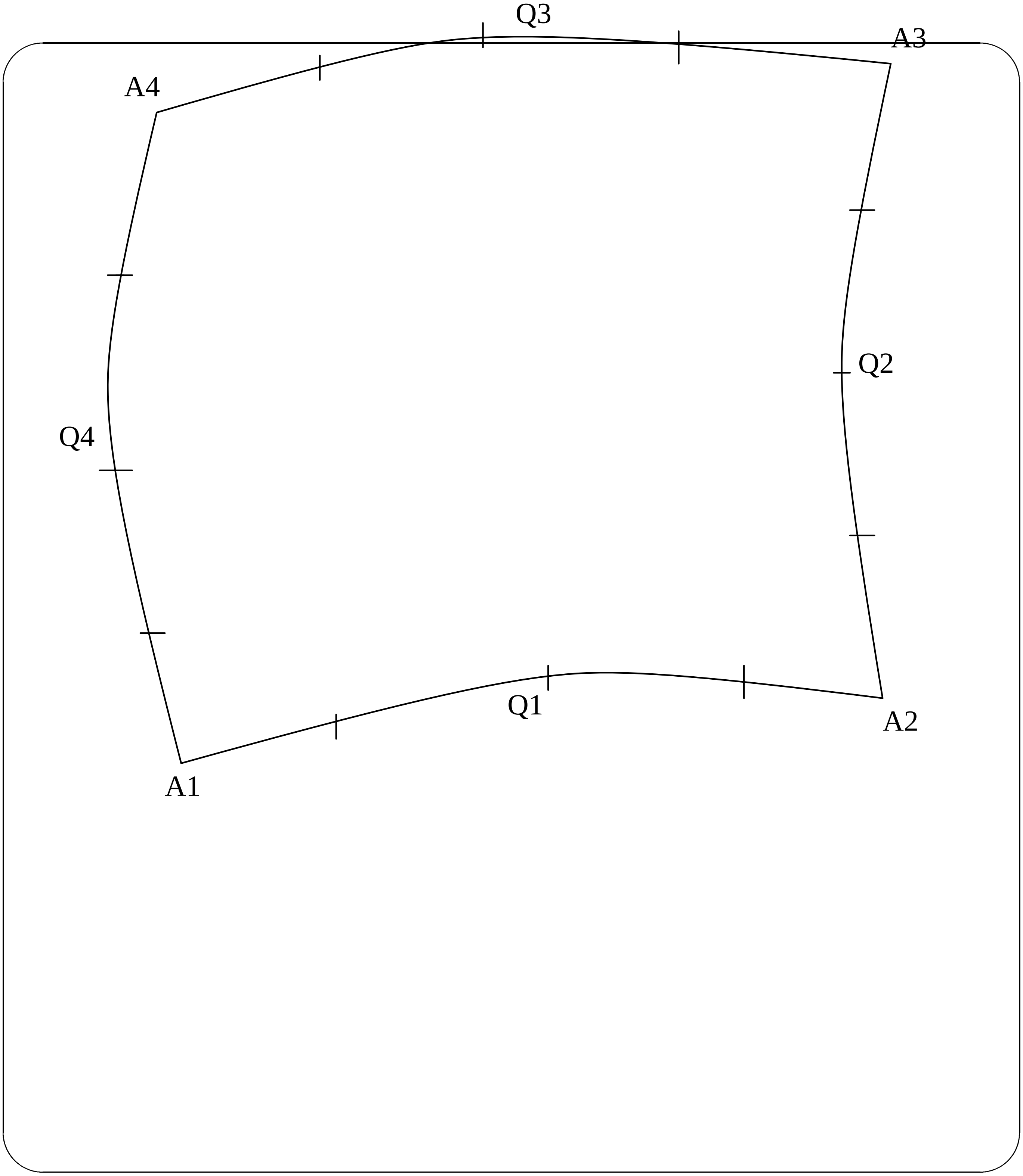
On distingue

1. le maillage d'un triangle curviligne,
2. le maillage d'un quadrilatère curviligne.

Analysons le maillage d'un quadrilatère curviligne.

Soit Q_j, $j = 1, 2, 3, 4$, les 4 côtés d'un quadrilatère curviligne.

Soit Q_j^i, $i = 1, \ldots, n_j$ donnés, avec $n_1 = n_3$, $n_2 = n_4$.


Q3
A3
A4
Q2
Q4
Q1
A2
A1

Étape 1. On définit sur $\hat{Q} = [0,1] \times [0,1]$ les points correspondants

$$\hat{Q}_j^i, i = 1, \ldots, n_j, j = 1, 2, 3, 4$$

par la formule

$$\hat{Q}_j^1 \hat{Q}_j^i = \frac{\Sigma_{k=1,\ldots,i} \, \|Q_j^1 Q_j^k\|}{\Sigma_{k=1,\ldots,n_j} \, \|Q_j^1 Q_j^k\|} \hat{Q}_j^1 \hat{Q}_j^{n_j}$$

Étape 2. On maille $\hat{Q}$ en joignant les points des côtés opposés. On a

$(n_1 - 1)(n_2 - 1)$ quadrilatères élémentaires.

Étape 3. La maillage de Q est l'image du maillage de $\hat{Q}$ par

$$F(\hat{Q}^\ell) = \sum_{j=1}^{4} p_j(\hat{Q}^\ell)[f_j(p_{j+1}(\hat{Q}^\ell)) - p_{j-1}(\hat{Q}^\ell) A_j]$$

Dans la formule ci-dessous, nous avons les notations:

A_1, A_2, A_3, A_4 sont les sommets de Q,

f_j est la représentation paramétrique du côté Q_j,

$p_1(\hat{M}) = 1 - \hat{y}(\hat{M})$,

$p_2(\hat{M}) = \hat{x}(\hat{M})$,

$p_3(\hat{M}) = \hat{y}(\hat{M})$,

$p_4(\hat{M}) = 1 - \hat{x}(\hat{M})$.

Remarques

1. On vérifie que $F : \hat{Q}_j \to Q_j$, $j = 1, 2, 3, 4$.
2. On peut approcher f_j par $\tilde{f}_j$ linéaire par morceaux.
3. F n'est pas toujours bijective.

Mot-clé: **QUAC** construit un maillage structuré d'un quadrangle en triangles ou en quadrangles.

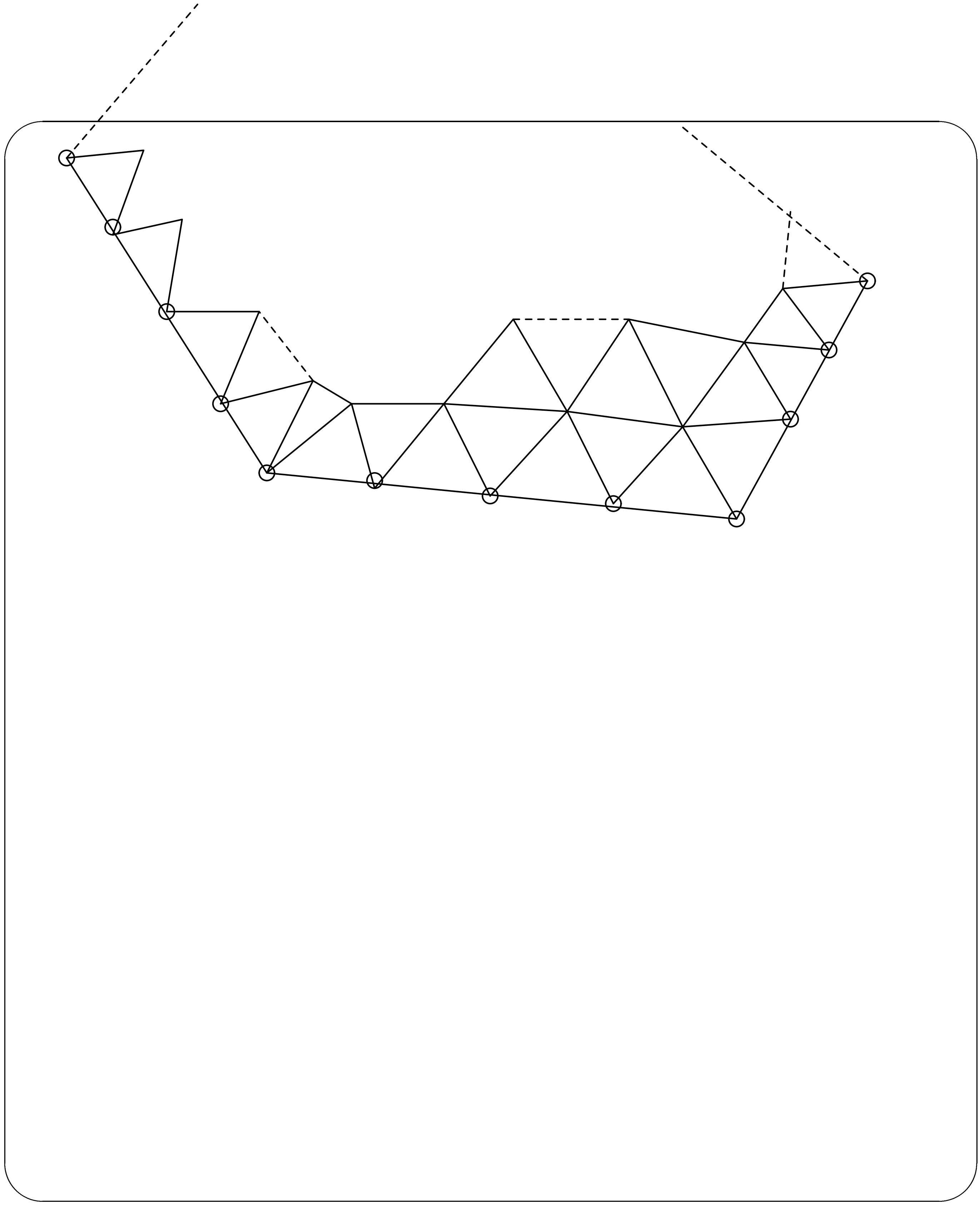
Mot-clé: **TRIC** construit un maillage structuré d'un triangle en triangles.

Génération à partir du contour: méthode frontale

On se donne les points du bord de Ω. Ici Ω est un domaine limité par des **lignes orientées**, pour distinguer l'intérieur de l'extérieur. On maille alors par couches successives en respectant des critères d'angles et de diamètre d'éléments.

Il s'agit d'une méthode simple à programmer, mais qui a de nombreux inconvénients.

1. On ne domine pas le nombre de triangles.
2. On peut avoir des problèmes pour terminer l'algorithme.
3. L'algorithme est très sensible à la distribution de points sur $\partial\Omega$.



Modulef-Maillage-20

Méthode de Voronoï

On parle de la méthode de Voronoïou de la méthode de Delaunay-Voronoï.

Il s'agit de l'approche la plus générale pour obtenir des maillages, qu'il s'agisse de domaines bidimensionnels ou de domaines tridimensionnels. Partant du maillage du bord du domaine, la méthode de Voronoï–Delaunay permet de construire un maillage de l'intérieur.

Pour les problèmes tridimensionnels, une difficulté majeure demeure: comment construire le maillage de la surface ?

Les travaux majeurs concernant cette technique sont

- Voronoï [1908], géométrie algébrique,
- Delaunay [1934],
- Hermeline [1980], analyse numérique.

Décrivons les idées fondamentales concernant cette méthode.

Supposons avoir Σ un ensemble de points du plan (un nuage). Nous voulons définir une triangulation $\mathcal{T}_h$ ayant ces points comme sommets.

Pour S_i, S_j dans Σ, on définit l'ensemble

$$V_{ij} = \{M \in \mathcal{R}^2; d(M, S_i) \leq d(M, S_j)\}.$$

À chaque sommet S_i on associe l'ensemble convexe

$$V_i = \bigcap_j V_{ij}.$$

Nous avons aussi

$$V_i = \{M \in \mathcal{R}^2; \forall j \neq i \ d(M, S_i) \leq d(M, S_j)\}.$$

Les ensembles V_i recouvrent le plan. C'est une **triangulation de Delaunay**.

On obtient alors $\mathcal{T}_h$ en joignant 2 à 2 les sommets S_i et S_j tels que V_i et V_j ont 1 côté en commun.

Un telle triangulation satisfait à la *propriété de Delaunay*: le cercle circonscrit à chaque triangle ne contient pas d'autres points du nuage.

N.B. Cette propriété assure la stabilité de certaines méthodes d'éléments finis pour des problèmes évolutifs.

Une triangulation de Delaunay-Voronoï possède la propriété suivante. Si nous rajoutons au nuage de points de la triangulation un nouveau point, il est possible de construire à partir de la triangulation existante un nouvelle triangulation de Delaunay-Voronoï pour le nouveau nuage de points. L'algorithme est un algorithme local. Cette propriété est fondamentale pour construire des maillages.

Pour trianguler on opère de la façon suivante.

1. On part de Σ_0 les points sur $\partial\Omega$ et on construit $\mathcal{T}_h^0$,
2. on crée un point à l'intérieur de chaque triangle T tant que T est trop grand
$$\frac{\rho_c(T)}{\rho_i(T)} \geq \delta.$$

d) Les volumes.

Ce sont les objets tridimensionnels, qui se construisent à partir des faces.

volume: volume_numéro faces_contour
référence_volume.

À un volume nous rattachons aussi un maillage.

Nous avons des techniques semblables à celles présentées pour les domaines en dimension 2.

À savoir

1. génération manuelle,
2. génération par subdivision,
3. génération par transformation,
4. génération à partir du contour.

Difficulté: il est nécessaire de partir du maillage du contour, qui est souvent difficile à obtenir.

III.4. Les maillages dans Modulef

Notion de référence. Pour retrouver les différentes particularités d'un domaine:

1. lignes courbes,
2. milieu,
3. conditions aux limites,

il est nécessaire de référencer.

On modifie la notion de *partie primale*:

il s'agit d'une partie géométrique simple avec une référence physique unique.

Un maillage est en général un ensemble de valeurs et de tableaux de valeurs contenant la liste des éléments géométriques le constituant. Pour chaque élément, les informations suivantes sont nécessaires:

1. la *nature* de l'élément : segment, triangle, quadrangle, tétraèdre, pentaèdre, hexaèdre ou autres,
2. la liste des *sommets* de l'élément,
3. la *connectivité* et la *topologie*: la connectivité décrit la liaison entre les sommets, la topologie définit les arêtes et les faces de l'élément à partir de ses sommets,

4. le *nombre* et la liste de ses *nœuds*,
5. les *coordonnées* de ses sommets,
6. les *attributs physiques* de l'élément, que ce soit le numéro de sous-domaine ou le numéro de référence.

Le code Modulef stocke ces informations dans une structure de données, qui constitue l'interface entre les différents modules traitant de maillage.

S.D. NOPO. Les informations concernant un maillage sont généralement stockées dans une structure de données NOPO. Il s'agit de la structure originelle de stockage de maillages.

S.D. GEOM. La S.D. GEOM est utilisée pour les représentations graphiques.

S.D. MAIL et **COOR**. Les S.D. MAIL et COOR contiennent les informations sur le maillage et sur l'interpolation choisie.

Le module APNOPO

Le module APNOPO permet l'enchaînement et l'exécution de tous les modules travaillant sur une structure de données NOPO contenant un maillage **bidimensionnel**.

Remarques

La numérotation des niveaux correspond à un ordre logique sauf pour le niveau 6 qui peut être exécuté à tout moment.

Certains niveaux peuvent être sautés.

Le préprocesseur APNOXX permet de créer les données utiles au module APNOPO.

Les opérations sont de 6 niveaux logiques.

1. *Préparation*: saisie des points et lignes caractéristiques du maillage.
2. *Génération*: appel des mailleurs.
3. *Manipulation*: modifications géométriques des maillages créés.
4. *Définition des nœuds*: création des nœuds autres que les sommets si nécessaire.
5. *Renumérotation*: minimisation de la taille des matrices.
6. *Sauvegarde sur fichier*: sauvegarde des maillages créés.

Il y a deux manières d'appeler le module APNOPO:

1. via le préprocesseur APNOXX,
2. directement.

Dans le premier cas, on devra réaliser l'édition de liens du programme APNOXX avec les bibliothèques CONV, CONVW, NOP2, NOPO, UTSD, UTIL ET UTIF (utilitaires généraux non portables).

Dans le second cas, les bibliothèques utiles sont les mêmes que ci-dessus , à l'exception des bibliothèques CONV et CONVW devenues inutiles.

Un programme d'appel en F77 contiendra les étapes suivantes:

- COMMON M(LM) (déclaration du super tableau de travail M de LM mots)
- EXTERNAL FFRONT (déclaration de la fonction utilisée)
- CALL INITI(L,LM,IMPRE,NNN) (initialisation de tout travail Modulef)
- CALL APNOPO(M,M,FFRONT) (appel du module)
- STOP
- END

Les seuls arguments de ce module sont le super tableau M et le nom formel d'une éventuelle fonction de projection. Les données sont contenues dans un fichier de commandes sous la forme de mots-clés et des valeurs associées.

Nous avons une liste des mots-clés classés selon le niveau mentionné ci-dessus.

Par exemple pour le mot-clé Q4T (appel du module QUATRI pour découper chaque quadrangle en quatre triangles), on précisera: Q4T : le mot-clé à activer,

IMPRE : le paramètre d'impressions,

NIVEAU1 : le niveau de la S.D. de départ,

NIVEAU2 : le niveau de la S.D. résultat.

Le module APNOP3

Le module APNOP3 permet l'enchaînement et l'exécution de tous les modules travaillant sur une structure de données NOPO contenant un maillage **tridimensionnel**.

Comme pour le cas bidimensionnel, les opérations possibles sont de 6 niveaux logiques.

Le préprocesseur APN3XX permet de créer les données utiles au module APNOP3.

III.5. Post-traitements et graphiques

Modulef possède des outils pour effectuer du post-traitement numérique et graphique. Dans la documentation, un guide complet (sur 7) est consacré à ce sujet.

On distingue le traitement numérique et le traitement graphique des résultats.

Traitement numérique.

Nous trouvons des outils pour manipuler des maillages.

Par exemple, il est possible à l'aide du programme DEFNXX de construire une S.D. NOPO à partir de la S.D. NOPO contenant le maillage initial et de la S.D. B contenant les déplacements.

Avec QUALXX, on vérifie que les éléments d'un maillage 2D sont de surface positive. On peut obtenir la qualité des éléments.

Nous trouvons aussi des outils permettant de manipuler des S.D. B.

Le préprocesseur COSDXX permet de construire une S.D. B à partir
– d'une valeur donnée: tous les degrés de liberté sont affectés à cette valeur,
– d'une fonction des coordonnées de chacun des nœuds.

Le préprocesseur SYMBXX permet de construire une S.D. B symétrique par rapport à une droite (un plan) d'une S.D. B donnée.

Le préprocesseur RECOXX permet de construire une S.D. B issue du recollement de deux structures B en parallèle du recollement des structures NOPO correspondantes.

Le préprocesseur STREXX permet de construire la S.D. TAE des contraintes obtenue à partir des tableaux élémentaires utilisés pour résoudre le problème de l'élasticité en déplacement.

Le module STRESS calcule les contraintes dans chaque élément. Pour un problème 2D, on peut visualiser les contraintes à l'aide de TRSTXX.

Mentionnons encore les préprocesseurs:

FLUXXX, IMAGXX, NORMXX, MAXDXX,...

Traitement graphique.

La visualisation des objets utilisés par une méthodes d'éléments finis,

maillage, déformés, températures, vitesses, pressions, contraintes,...

est un point essentiel dans

l'analyse et la communication des résultats.

Voici une liste des préprocesseurs existants:

- TRSTXX: contraintes 2D, zones de plasticité,

- TRMCXX: maillages 2D, vitesses, coupes, isovaleurs, déformées,

- COUPXX: maillages 3D et coupes,
- TRACXX: courbes,
- TRNOXX: maillages 2D et 3D,
- TRC3XX: maillages 3D, déformées, coupes, iso-valeurs, vitesses,
- REFEXX: maillages 3D et transformations,
- TRPOXX: points et lignes caractéristiques,
- TRPRXX: profil d'une matrice,
- ISOPXX: piézométrie,
- VIS3XX: surfaces.

Méthodologie: tous les modules de tracé utilisent le *logiciel* Fortran 3D (F3D).

On peut distinguer les étapes:

1. récupérer dans les S.D. les informations utiles,
2. réaliser le tracé à l'aide de F3D,
3. réaliser le dessin sur le terminal choisi.

F3D est un ensemble de programmes permettant de visualiser des objets dans un espace à deux ou trois dimensions sur différents terminaux (ou drivers) graphiques.

Les idées principales de F3D sont:

• la description dynamique des objets (leur description résulte de l'exécution du programme,

• le choix dynamique du terminal de sortie graphique,

• choix dynamique du dispositif d'entrée.

F3D est **un logiciel graphique de base**. Pour assurer l'indépendance vis à vis des terminaux, l'image est conceptuellement produite et modifiée sur un écran virtuel. Finalement l'image sera envoyée sur un terminal de sortie ou périphérique.