BL2D-V2 : mailleur bidimensionnel adaptatif

Patrick Laug, Houman Borouchaki

▶ **To cite this version:**

Patrick Laug, Houman Borouchaki. BL2D-V2 : mailleur bidimensionnel adaptatif. [Rapport de recherche] RT-0275, INRIA. 2003, pp.49. inria-00069902

HAL Id: inria-00069902
https://inria.hal.science/inria-00069902

Submitted on 19 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

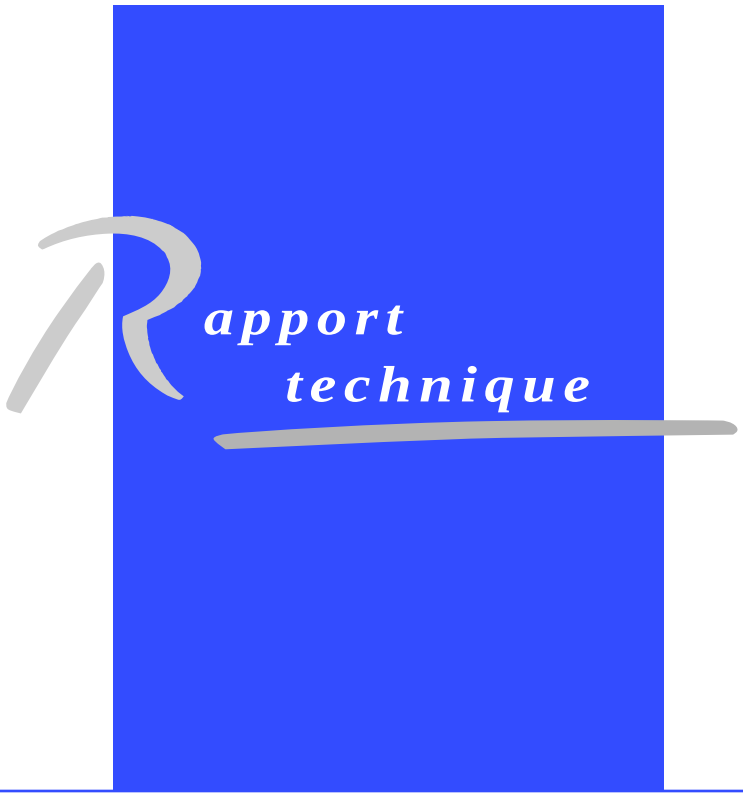
BL2D-V2 : mailleur bidimensionnel adaptatif

Patrick Laug — Houman Borouchaki

N° 0275

Janvier 2003

THÈME 4



ISSN 0249-0803

BL2D-V2 : mailleur bidimensionnel adaptatif

Patrick Laug [*] , Houman Borouchaki [**]

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet Gamma

Rapport technique n° 0275 — Janvier 2003 — 49 pages

Résumé : Le logiciel BL2D-V2 génère des maillages dans le plan, isotropes ou anisotropes, et peut être intégré dans un schéma adaptatif de calcul (notamment par éléments finis). Par rapport à la version V1, il offre de nombreuses possibilités nouvelles : méthode frontale, triangles quadratiques courbes, quadrilatères de degré 1 ou 2, frontières déformables, allocation dynamique de mémoire, etc. Ce rapport contient une initiation au logiciel, une description détaillée des programmes et des formats utilisés, ainsi que plusieurs exemples d'applications.

Mots-clés : maillage adaptatif, maillages isotrope et anisotrope, métrique riemannienne.

[*] E-mail : Patrick.Laug@inria.fr
[**] E-mail : Houman.Borouchaki@utt.fr

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau, Rocquencourt, BP 105, 78153 Le Chesnay Cedex (France)
Téléphone : +33 1 39 63 55 11 — Télécopie : +33 1 39 63 53 30

BL2D-V2: Adaptive Bidimensional Mesher

Abstract: The BL2D-V2 software package generates meshes in the plane, isotropic or anisotropic, and can be integrated within an adaptive computation scheme (especially using finite elements). Compared with the V1 version, it offers many new capabilities: advancing-front method, curved quadratic triangles, quadrilaterals of degree 1 or 2 , deformable boundaries, dynamic memory allocation, etc. This report contains an introduction to the software package, a detailed description of the programs and formats used, as well as several application examples.

Key-words: adaptive mesh, isotropic and anisotropic meshes, Riemannian metric.

Table des matières

1 Introduction **4**

2 Initiation au logiciel **6**
2.1 Exemple simple 6
2.2 Exemple de maillage adaptatif 10
2.3 Nommage des fichiers 15
2.4 Options ou variables d'environnement 15

3 Les programmes par ordre alphabétique **17**
3.1 Programme `blcv` 17
3.2 Programme `bldraw` 17
3.3 Programme `blg` 18
3.4 Programme `blh` 18
3.5 Programme `blinterpol` 19
3.6 Programme `blmc` 19
3.7 Programme `blqms` 19
3.8 Programme `blsmo` 19
3.9 Programme `bltms` 20

4 Les formats de fichiers par ordre alphabétique **21**
4.1 Format `.amdba` 21
4.2 Format `.bl2d` 22
4.3 Format `.data` 23
4.4 Format `.g` 25
4.5 Format `.h` (ou `.hg`, `.hc`, `.hs`) 30
4.6 Format `.is` 33
4.7 Format `.mc` (ou `.smo`) 34
4.8 Format `.ms` 36

5 Exemples d'applications **38**
5.1 Arobase : méthodes algébrique et frontale 38
5.2 Forgeage : frontières déformables 41
5.3 Quart de plaque trouée : choix des éléments 44
5.4 Mécanique des fluides : quadrilatères isotropes et anisotropes 45
5.5 Photos d'identité : triangles anisotropes 47

1 Introduction

Le logiciel BL2D-V2 [1] crée des maillages isotropes ou anisotropes dans le plan. Il peut être intégré dans un processus adaptatif. Son architecture logicielle est modulaire, et ses deux principaux composants permettent respectivement de discrétiser des courbes et de mailler des domaines du plan :

- **Discrétisation des courbes.** On suppose que la frontière Γ d'un domaine Ω de $\mathbb{R}^2$ est donnée sous une forme échantillonnée, constituée par un ensemble de segments droits. Cet échantillonnage est lissé par une *spline* cubique, qui est approchée de manière très fine par un segment polygonal, ce qui définit un support géométrique de la frontière Γ. Ce support géométrique est discrétisé en respectant des tailles spécifiées de manière externe [2].
- **Maillage des domaines.** À partir de la frontière Γ discrétisée précédemment, le mailleur insère des points sur des arêtes internes au domaine Ω (méthode algébrique) ou par couches successives (méthode frontale), puis connecte ces points entre eux par une méthode de Delaunay généralisée [3, 4]. Il génère des triangles droits (P^1) ou courbes de degré 2 (P^2), ou encore des quadrilatères Q^1 ou Q^2 par appariement.

La construction directe d'éléments de degré 2 est rendue possible via le contrôle du maillage des frontières du domaine, de façon à assurer la compatibilité désirée. Les nœuds milieux frontières sont placés en fonction de leurs abscisses curvilignes. Les nœuds milieux internes sont placés de manière à optimiser la qualité en forme des éléments.

En outre, le logiciel BL2D-V2 autorise le cas des domaines à frontières variables, dont la forme varie fortement au cours du temps.

Description fonctionnelle du logiciel

En entrée, un fichier (format `.g`) contient les données géométriques et physiques de plusieurs sous-domaines du plan. Il peut être créé directement par un préprocesseur intégré (`blg`) ou externe, ou encore être extrait de maillages donnés dans un autre format (`.nopo`, `.mesh`, `.amdba`, `.ms`, etc.). Un deuxième fichier d'entrée (`.hg`) permet de spécifier la taille et la forme des éléments à générer.

En sortie, un nouveau fichier (format `.ms`) contient le maillage demandé. Ce fichier peut également être converti en de nombreux autres formats.

Dans le cas d'une boucle d'adaptation, les données d'entrée comprennent en outre le maillage généré à l'étape précédente et une carte de tailles définies aux sommets de ce maillage (`.h`).

Remarques

Le logiciel BL2D-V2 est parfois référencé ailleurs sous le nom de BLMESH.

Il est issu d'une ancienne version, appelée maintenant V1 et datant de 1995 [5, 6]. Il est cependant très différent de sa version initiale, les principales nouveautés inclues dans la version V2 étant les suivantes : méthode frontale (placement optimal des points, meilleure qualité en forme des éléments), triangles de degré 2 (courbes), quadrilatères de degré 1 ou 2, frontières déformables, robustesse accrue (en particulier, détection des auto-intersections de la frontière), allocation dynamique de mémoire, etc.

Les formats de fichiers de la version V2 ont volontairement été maintenus très proches de ceux de la version V1.

Plan du rapport

Après cette introduction (section 1), une initiation à l'utilisation du logiciel est proposée à partir de plusieurs exemples (section 2). Ensuite sont données par ordre alphabétique la description détaillée des différents programmes du logiciel (section 3) et celle des formats de fichiers utilisés (section 4). Enfin, plusieurs exemples d'applications sont proposés (section 5).

2 Initiation au logiciel

Dans cette section, un premier exemple d'initiation est donné, suivi d'un exemple plus complexe avec adaptation de maillage. Ces exemples permettront de préciser le principe du nommage des fichiers et l'utilisation des variables d'environnement.

2.1 Exemple simple

Dans cet exemple d'initiation, le domaine à mailler est un simple triangle équilatéral de sommets A (-1, 0), B (1, 0) et C (0, $\sqrt{3}$) (voir figure 1).

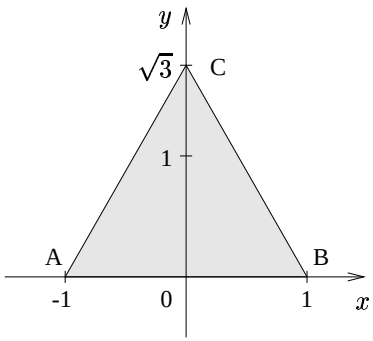


FIG. 1 – *Exemple simple de domaine (triangle équilatéral).*

Les données permettant de spécifier ce domaine se trouvent dans un fichier `x.0.0.data`, dont le contenu est présenté à gauche de la figure 2 et qui est généralement fourni avec le logiciel BL2D-V2. Pour obtenir un premier maillage, il suffit d'exécuter la commande `bl2d.sh` sans aucune option ou, pour cet exemple, `bl2d.sh -rigid 0` (le sens de cette option sera précisé par la suite). Le maillage obtenu peut être visualisé par le programme `bldraw` (figure 2, à droite), montrant ainsi sa totale régularité.

Cet exemple permet d'introduire quelques notions qui seront revues en détail par la suite. Dans le fichier `x.0.0.data`, les trois premiers entiers représentent un nombre maximal de points, de courbes et de sous-domaines à définir (ici, nous avons exactement 3 points, 3 courbes et 1 sous-domaine, et chaque majorant égal à 100 est donc très large). Ensuite, les sommets A, B, C et les côtés AB, BC et CA sont spécifiés (le mot-clé `NULL` indique que ces segments sont droits). La ligne `AB +1 0` signifie que le domaine à mailler est à gauche du segment AB et que son numéro de référence est 0. Le mot-clé `isotrope` précise la forme des éléments, qui doivent donc être le plus équilatéraux possible. Comme aucune taille n'est spécifiée, les segments droits qui constituent la frontière sont discrétisés avec

une taille maximale (par défaut $1/5^e$ de la diagonale du rectangle englobant), et le maillage correspondant à cette taille est généré.

Le maillage généré par le logiciel BL2D-V2 est décrit par un fichier dont le suffixe (ou extension) est `.ms`. Pour visualiser un tel fichier, il est possible de lancer le programme `bldraw` qui est piloté par des menus. Le premier menu détermine le terminal de sortie graphique. En donnant une ligne vide, on obtient un terminal par défaut. En tapant ensuite les données ci-dessous, on obtient le tracé de la figure 2.

```
% bldraw
 ...
 Files:  g | hg | smo | mc | hc | ms | ms- | bl2d | hs | h
 Others: b base | z zoom | t test | q quit bldraw
ms
 phe phys. ref. of edges | phd phys. ref. of subdomains | nd numbers associated with subdomains
 p points | f faces | cf colored faces | sf selected faces | Return quit draw_ms
f
 phe phys. ref. of edges | phd phys. ref. of subdomains | nd numbers associated with subdomains
 p points | f faces | cf colored faces | sf selected faces | Return quit draw_ms

 Files:  g | hg | smo | mc | hc | ms | ms- | bl2d | hs | h
 Others: b base | z zoom | t test | q quit bldraw
q
```

Pour obtenir un maillage uniforme plus fin, il est possible de spécifier une taille d'éléments aux points A, B et C (voir figure 3 avec une taille de 0.1).

Sur la figure 4, des tailles différentes sont spécifiées aux sommets A, B et C du triangle, respectivement 0.05, 0.1 et 0.2.

Enfin, figure 5, des éléments anisotropes sont générés. En chaque sommet (respectivement A, B et C) sont spécifiées une direction (resp. 30°, 150° et 90° par rapport à l'axe horizontal), la taille souhaitée le long de cette direction (0.2), et la taille souhaitée perpendiculairement à cette direction (0.02).

Fichier `x.0.0.data` :

```
100   100   100
A -1 0
B  1 0
C  0 SQRT(3)
;
;
AB NULL A B NULL ;
BC NULL B C NULL ;
CA NULL C A NULL ;
;
;
;
AB +1 0
;
isotrope
;
```

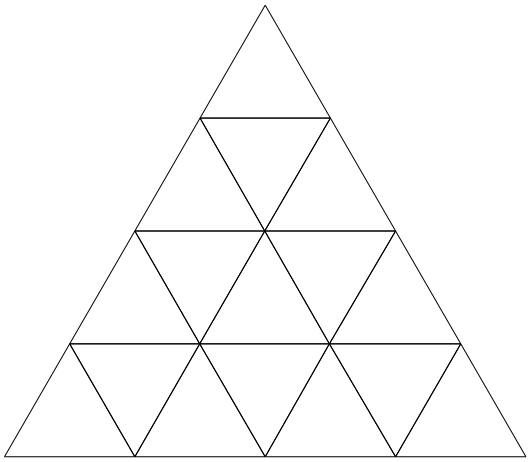


FIG. 2 – *Maillage uniforme par défaut.*

Fichier `x.0.0.data` :

```
100   100   100
A -1 0
B  1 0
C  0 SQRT(3)
;
;
AB NULL A B NULL ;
BC NULL B C NULL ;
CA NULL C A NULL ;
;
;
;
AB +1 0
;
isotrope
A 0.1   B 0.1   C 0.1 ;
```

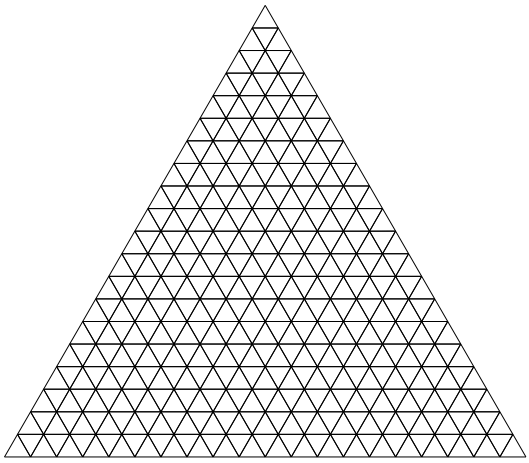


FIG. 3 – *Maillage uniforme raffiné.*

Fichier `x.0.0.data` :

```
100   100   100
A -1 0
B  1 0
C  0 SQRT(3)
;
;
AB NULL A B NULL ;
BC NULL B C NULL ;
CA NULL C A NULL ;
;
;
;
AB +1 0
;
isotrope
A 0.05   B 0.1   C 0.2 ;
```

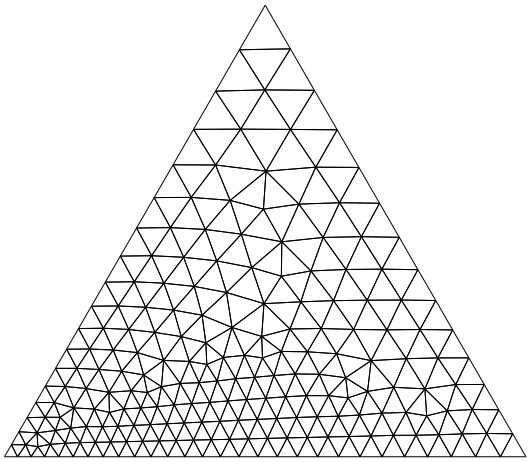


FIG. 4 – *Maillage isotrope à tailles variables.*

Fichier `x.0.0.data` :

```
100   100   100
A -1 0
B  1 0
C  0 SQRT(3)
;
;
AB NULL A B NULL ;
BC NULL B C NULL ;
CA NULL C A NULL ;
;
;
;
AB +1 0
;
anisotrope
A  30 0.2 0.02
B 150 0.2 0.02
C  90 0.2 0.02
;
```

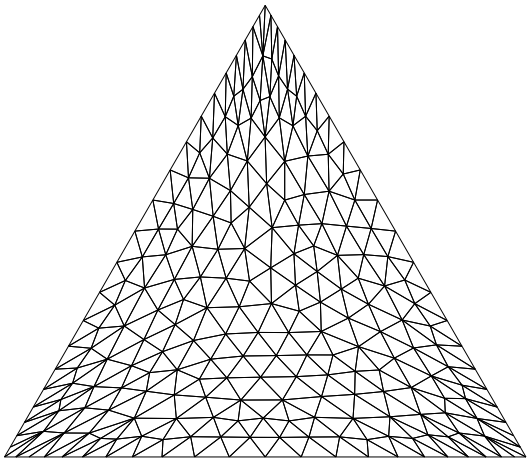


FIG. 5 – *Maillage anisotrope à directions variables.*

2.2 Exemple de maillage adaptatif

Considérons maintenant le cas plus complexe du maillage adaptatif d'un domaine de forme quelconque. Sa frontière est définie par des segments droits ou courbes, que nous représentons dans tous les cas par des *splines*. Nous devons spécifier ce domaine, discrétiser les splines, réaliser un maillage, et effectuer une simulation numérique (par exemple par la méthode des éléments finis). Enfin, selon les résultats obtenus, nous devons adapter le maillage en raffinant ou en déraffinant certaines régions, puis recommencer les calculs, ou bien arrêter le processus.

En pratique, ceci revient à créer un maillage initial en utilisant la même méthode que précédemment (fichier `x.0.0.data` et shell-script `bl2d.sh`), à spécifier une carte de tailles aux sommets de ce maillage, à rappeler le shell-script `bl2d.sh` avec l'option `-adapt 1`, à spécifier une nouvelle carte de tailles, et ainsi de suite. Ces différentes étapes sont précisées ci-dessous à partir d'un nouvel exemple mettant en jeu un maillage initial isotrope et des maillages adaptés anisotropes.

Définition du domaine

Le domaine choisi pour cet exemple est un "quart de plaque trouée", obtenu par différence entre un carré de côté 1 et un cercle de rayon 0.5 (figure 6).

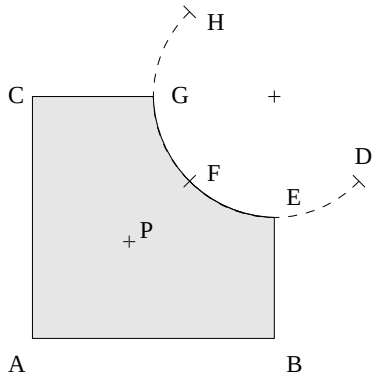


FIG. 6 – *Exemple du "quart de plaque trouée".*

Le fichier `x.0.0.data` qui spécifie ce domaine (figure 7) est distribué avec le logiciel BL2D-V2, comme précédemment. La forme générale d'un tel fichier est décrite en section 4.3. Dans le cas présent, les données sont explicitées par des commentaires commençant par deux tirets (--). La frontière est constituée de plusieurs segments droits joignant les points G, C, A, B, E, et d'un quart de cercle qui est approché par une spline dont les points de contrôle sont D, E, F, G, H. Un point P est imposé, ce qui signifie qu'il sera sommet d'un élément de tout maillage généré. Des références physiques permettent d'associer des

attributs aux points, segments et sous-domaines, en vue d'une simulation numérique. Enfin, une carte uniforme de tailles ($h = 0.1$) est spécifiée.

Fichier `x.0.0.data` :

```
100 -- nombre maximal de points
100 -- nombre maximal de segments
100 -- nombre maximal de sous-domaines

-- points
!X=SQRT(2)/4
A 0 0     B 1 0       C 0 1     P 0.4 0.4     D 1+X 1-X
E 1 0.5   F 1-X 1-X   G 0.5 1   H 1-X 1+X     ;

-- points imposes
P ;

-- segments
AB NULL A B NULL ;
BE NULL B E NULL ;
EG D E F G H ;
GC NULL G C NULL ;
CA NULL C A NULL ;
;

-- references physiques de certains points imposes ou extremites
P 10   A 20   B 30   C 40   E 50   G 60   ;

-- references physiques de certains segments
AB 110   BE 120   EG 130   GC 140   CA 150   ;

-- sous-domaines
AB +1 200 ;

-- carte de tailles ou de metriques
isotrope
A 0.1   B 0.1   C 0.1   P 0.1   E 0.1   F 0.1   G 0.1   ;
```

FIG. 7 – *Exemple du "quart de plaque trouée" : données.*

Maillage initial

Comme précédemment, le maillage initial est créé par la commande `bl2d.sh -rigid 0`. De manière interne, cette commande appelle successivement les programmes `blg`, `blsmo`, `blmc` et `bltms`. Tous ces programmes génèrent des fichiers qui peuvent être visualisés par le programme `bldraw`.

Le programme `blg` (définition du domaine de calcul) convertit le fichier `x.0.0.data` en deux fichiers initiaux `x.0.0.g` et `x.0.0.hg`, comme l'illustre la figure 8 :

- Le fichier `x.0.0.g` (données géométriques et physiques) reprend les données précédentes, les points étant maintenant numérotés de 1 à 9 et les splines de (1) à (5).
- Le fichier `x.0.0.hg` (carte initiale de tailles) requiert une taille $h = 0.1$ aux points 1, 2, 6, 8, 3 (extrémités), 7 (point de contrôle interne à une spline) et 4 (imposé).

Le programme `blsmo` calcule les splines à partir des points de contrôle fournis.

Le programme `blmc` discrétise les splines selon les tailles prescrites.

Enfin, le programme `bltms` maille le domaine en un ensemble de triangles (voir figure 9, en haut à gauche).

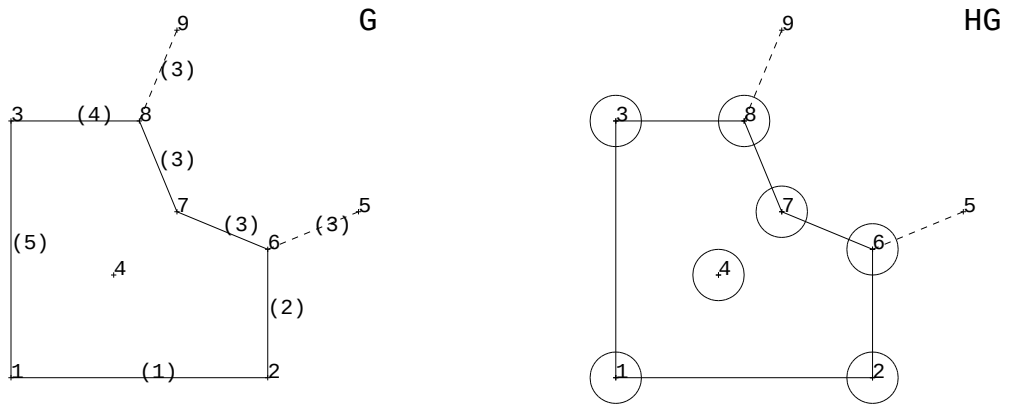


FIG. 8 – *Tracés à partir des fichiers initiaux* `x.0.0.g` *et* `x.0.0.hg`.

Première carte de tailles en vue d'une adaptation

Le maillage précédent peut être utilisé pour un calcul par éléments finis. Si la précision des résultats doit être améliorée en certains points, il est possible d'adapter le maillage et d'effectuer un nouveau calcul. Ce remaillage est habituellement basé sur une estimation d'erreur *a posteriori* de la solution physique cherchée. Ici, nous avons simulé cette étape à l'aide du programme `blh` :

```
% blh
  Test number?
  5
```

Ce programme crée un fichier `x.0.0.h` qui représente une première carte de tailles. Dans le cas prédéfini n° 5, une carte anisotrope est générée. À partir des coordonnées (x, y) d'un

sommet du maillage initial, une direction principale d'angle θ et deux tailles (h_1, h_2) sont calculées par les instructions Fortran suivantes :

```
theta = atan2(y-1, x-1)
h1 = 0.4*abs((x-1)**2 + (y-1)**2 - (0.75)**2) + 0.003
h2 = 0.1
```

Ces expressions sont choisies de telle manière que, sur le cercle de centre (1,1) et de rayon 0.75, on obtient des triangles étirés tangents à ce cercle, dans un rapport de $0.003/0.1 = 3\%$.

Boucle d'adaptation

Les commandes ci-dessous permettent de générer un premier maillage adapté et de recommencer le processus. La figure 9 montre les différents maillages obtenus. Tous les maillages conservent comme prévu le point imposé de référence 10.

```
% bl2d.sh -rigid 0 -adapt 1
% blh
  Test number?
  5
% bl2d.sh -rigid 0 -adapt 2
% blh
  Test number?
  5
% bl2d.sh -rigid 0 -adapt 3
```

Pour éviter de répéter les mêmes commandes, il est évidemment possible d'écrire un shell-script équivalent :

```
#!/bin/sh
for i in 1 2 3
do
   bl2d.sh -rigid 0 -adapt $i
   echo 5 | blh
done
```

Remarquons que certaines arêtes ont leurs deux extrémités sur la frontière. Si ceci n'est pas souhaité, ce qui est généralement le cas pour des calculs par éléments finis, il suffit de remplacer l'option `-rigid 0` par `-rigid 1`.

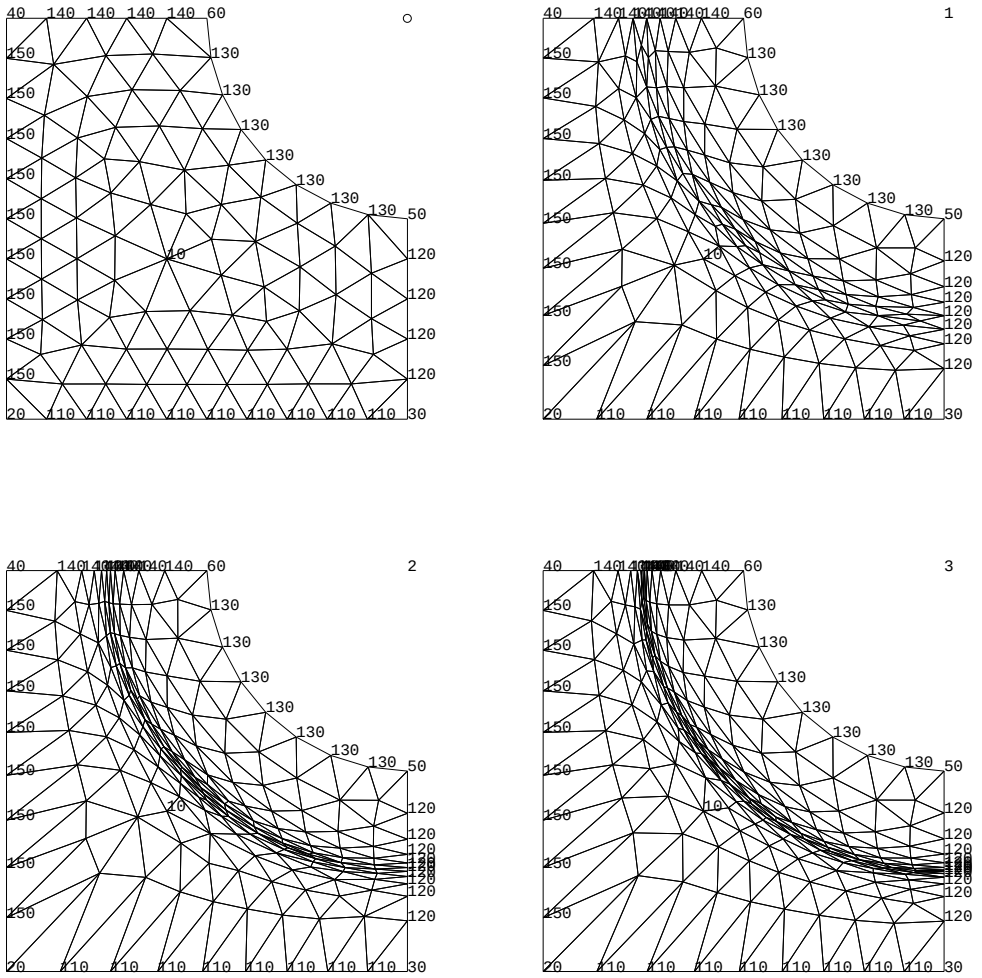


FIG. 9 – *Maillage initial ($i = 0$) et maillages adaptés ($i = 1$, 2 et 3).*

2.3 Nommage des fichiers

Dans les exemples précédents, le logiciel BL2D-V2 lit ou écrit des fichiers de noms `x.0.0.data`, `x.0.0.ms`, `x.0.0.h`, `x.0.1.ms`, etc. La gestion de ces fichiers est pratiquement transparente pour l'utilisateur grâce à une convention simple sur leurs noms, qui sont toujours de la forme :

base . *geom* . *adapt* . *format*

- *base* est un nom de base, ou préfixe, choisi par l'utilisateur pour identifier l'application en cours. Par défaut, le nom de base est `x`.
- *geom* est un numéro qui permet de prendre en compte les changements de formes géométriques de la frontière, dans le cas de domaines déformables (valeur 0 par défaut).
- *adapt* est le compteur de la boucle d'adaptation, qui vaut 0 pour le maillage initial (non adapté) et est incrémenté de 1 à chaque itération.
- *format* est un suffixe (ou extension) qui indique le format du fichier.

2.4 Options ou variables d'environnement

Toutes les options du shell-script `bl2d.sh` sont en fait transcrites dans un fichier de nom `bl2d.env`. Par exemple, la commande `bl2d.sh -rigid 0 -adapt 3` crée le fichier suivant :

```
! file generated by bl2d.sh
rigid 0
adapt 3
```

Ce fichier définit des variables d'environnement qui permettent de gouverner les différents programmes de BL2D-V2. S'il le souhaite, l'utilisateur peut créer lui-même un fichier `bl2d.env` et appeler manuellement ces programmes. La liste ci-dessous donne, pour chaque variable d'environnement, sa valeur par défaut (entre crochets [ ]) et sa signification.

- `adapt` [0] est le numéro d'adaptation des fichiers (cf. nommage, section 2.3).
- `adjacency` [0], dans le cas de triangles P^1, indique si le fichier de maillage (`.ms`) doit donner la liste des triangles voisins (1) ou non (0).
- `angle_geom` [8.] définit un angle θ_g (en degrés) qui représente la tolérance d'un maillage géométrique. Lorsqu'une courbe est discrétisée, θ_g est l'angle limite entre la tangente à la courbe et une arête sous-jacente.
- `angle_smo` [1.] définit un angle θ_s (en degrés) qui représente la tolérance du support des courbes (fichier `.smo`). Dans ce support, qui n'est autre qu'un segment polygonal très proche de la courbe, θ_s est l'angle limite entre la tangente à la courbe et une arête sous-jacente.
- `base` [x] (variable anciennement appelée `pref`) est le nom de base des fichiers (cf. nommage, section 2.3).

- `blsmo` [8] définit le type de spline utilisé pour définir une courbe à partir de points de contrôle. En appelant le programme `blsmo` avec une variable d'environnement `verb` ≥ 50, on obtient une liste d'options numérotées de 1 à 11 (Catmull-Rom, Hosaka, de Boor, ...). La valeur recommandée (8) correspond à la méthode de Boor prenant en compte les longueurs des cordes.
- `element` [`p1`] désigne le type des éléments du maillage à créer. Les valeurs possibles sont `p1` (triangles droits P^1), `p2` (triangles quadratiques courbes P^2), `q1.0` (quadrilatères droits Q^1 et quelques triangles P^1), `q1.1` (quadrilatères droits Q^1 uniquement), `q2.0` (quadrilatères courbes Q^2 et quelques triangles P^2) et `q2.1` (quadrilatères courbes Q^2 uniquement). Une illustration de ces possibilités est fournie en section 5.3.
- `file_h` [0] vaut 1 pour écrire la carte des tailles interpolées aux sommets du maillage (fichier `.hs`), et 0 sinon.
- `file_interpol` [1] vaut 1 pour générer un fichier `.is` qui, dans le cas d'une adaptation, donne les coordonnées barycentriques des sommets du nouveau maillage par rapport à l'ancien (cf. section 4.6), et 0 sinon.
- `frontal` [1] désigne la méthode de maillage, 1 frontale et 0 algébrique (cf. section 5.1).
- `geom` [0] est le numéro de géométrie des fichiers (cf. nommage, section 2.3).
- `hmax` [*diag*/5] est la taille maximale autorisée pour une arête du maillage (dans l'expression de la valeur par défaut, *diag* représente la longueur de la diagonale de la boîte englobante).
- `hmin` [*diag*/500] est la taille minimale autorisée pour une arête du maillage (dans l'expression de la valeur par défaut, *diag* représente la longueur de la diagonale de la boîte englobante).
- `insert` [1] indique si le mailleur doit insérer des points internes (1) ou non (0).
- `LSS` [0.5] est l'abréviation de "longueur d'un sous-segment". Cette variable est habituellement comprise entre 0.1 et 0.5. En pratique, une petite valeur augmente la précision des calculs mais également le temps CPU et la mémoire nécessaire.
- `memory` [100000]. La valeur spécifiée est le nombre maximal de points que le mailleur pourra générer, ce qui lui permet de calculer l'espace-mémoire nécessaire.
- `metric_interpol` [0] précise le type d'interpolation de la carte de tailles, linéaire (0) ou géométrique (1).
- `rigid` [1]. Si cette variable est nulle, une arête peut avoir ses deux extrémités sur la frontière du domaine. Si ceci n'est pas souhaité, ce qui est généralement le cas pour des calculs par éléments finis, il suffit de spécifier `rigid 1`.
- `verb` [10] est le degré de "verbosité" du logiciel, c'est à dire un pourcentage allant de 0 (impressions inexistantes ou très restreintes) à 100 (maximum d'impressions).

3 Les programmes par ordre alphabétique

La description détaillée des programmes que les utilisateurs peuvent activer est donnée ci-dessous par ordre alphabétique.

3.1 Programme `blcv`

Le programme `blcv` permet de convertir entre eux divers types de fichiers (cf. section 4), comme l'indique son menu :

```
 1 .g .mc .ms     -> AMDBA
 2 .g .mc .ms     -> AM
 3 .g .mc .ms     -> AM_FMT
 4 .g .mc .ms     -> .bl2d   ! blexport
 5 MESH           -> OFF
 6 MESH           -> POINTS, FACES
 7 INP            -> MESH
 8 VERT, FACE     -> MESH
 9 MS             -> AMDBA with references set to 0
10 MESH           -> AMDBA
11 MESH (0 based) -> MESH (1 based)
12 WRL            -> MESH
13 G, MC, MS      -> NOPOF (formatted) with triangles
14 G, MC, MS      -> NOPO with triangles
15 G, MC, FMT     -> NOPO with quadrilaterals
16 AMDBA          -> AM_FMT
18 .bl2d          -> .g .hg .smo .mc .ms-   ! blimport
20 AMDBA          -> MESH
22 TRI            -> MESH
23 .points .faces -> .mesh
24 .nopo          -> .0.0.bl2d   ! 2d triangles only
25 .bl2d          -> .mesh
26 .mesh          -> .stl
27 .mesh          -> .mesh after symmetry and rotation (e.g. Columbia, 747)
28 .mesh          -> .wrl
29 .mesh          -> .nopo 2D with triangles (ISMC)
30 .mesh          -> number of elements
31 .mesh          -> .points .faces (no allocation)
32 .mesh special  -> .blm (molecule)
```

3.2 Programme `bldraw`

Données :	dirigées par des menus
Fichiers d'entrée :	selon le tracé à effectuer
Fichiers de sortie :	écran ou Postscript

Le programme `bldraw` réalise des tracés à partir de différents fichiers. Il utilise la bibliothèque graphique Fortran 3D, qui permet un choix dynamique du terminal de sortie

graphique [8, partie 3]. L'utilisateur est constamment guidé par des menus. Le menu principal est le suivant :

```
Files:  g | hg | smo | mc | hc | ms | ms- | bl2d | hs | h
Others: b base | z zoom | t test | q quit bldraw
```

Le menu `Files` permet de choisir un ou plusieurs fichiers à tracer (cf. formats définis en section 4).

Le menu `Others` permet d'accéder à d'autres fonctionnalités :

- `b` : fonctions de base (effacer l'écran, changer la hauteur des caractères ou l'épaisseur des traits, afficher un texte, ...),
- `z` : définition d'une fenêtre de zoom,
- `t` : tests divers,
- `q` : quitter le programme `bldraw`.

La plupart des tracés de ce rapport (figures 2 à 5, 8 et 9 par exemple) ont été obtenus en utilisant le programme `bldraw`.

3.3 Programme `blg`

Fichiers d'entrée : x.0.0.data
Fichiers de sortie : x.0.0.g x.0.0.hg

Le programme `blg` permet de créer les fichiers initiaux `.g` et `.hg` (cf. section 2.2). Il utilise un fichier de données créé par l'utilisateur (cf. section 4.3). Un autre moyen de créer ces fichiers initiaux est d'utiliser un logiciel dédié à des applications particulières (par exemple, dans le domaine de la métallurgie, calcoMESH [9]).

3.4 Programme `blh`

Fichiers d'entrée : x.0.i.c x.0.i.ms
Fichiers de sortie : x.0.i.h

Le programme `blh` simule un *estimateur d'erreur*. Habituellement, un estimateur d'erreur analyse le résultat d'un calcul par éléments finis et en déduit une carte de tailles ou de métriques. Ici, la carte est définie analytiquement par l'utilisateur, sous forme d'instructions Fortran.

Par exemple, le cas numéro 5 du programme `blh` a été présenté en section 2.2. Il est facile d'ajouter d'autres cas dans le code source `blh.f90` ou d'écrire un nouveau programme.

3.5 Programme `blinterpol`

Fichiers d'entrée : x.0.i.bb x.0.i+1.is
Fichiers de sortie : x.0.i.bbi

Le programme `blinterpol` utilise deux fichiers d'entrée qui contiennent respectivement :
- la solution d'un calcul par éléments finis sur un maillage i,
- les coordonnées barycentriques des points du maillage $i+1$ par rapport au maillage i (cf. section 4.6).

Il en déduit une solution interpolée qui permet d'initialiser le calcul suivant (la version fournie réalise une interpolation de type P^1 mais peut être modifiée).

3.6 Programme `blmc`

Fichiers d'entrée :	si $i = 0$: x.0.0.g	x.0.0.smo	x.0.0.hg	
	si $i \geq 1$: x.0.0.g	x.0.0.smo	x.0.i-1.c	x.0.i-1.h
Fichiers de sortie :	si $i = 0$: x.0.0.c	x.0.0.hc		
	si $i \geq 1$: x.0.i.c	x.0.i.hc	x.0.i.ic	

Le programme `blmc` est un mailleur de courbes : il discrétise chaque segment courbe en arêtes. Les tailles de ces arêtes sont déterminées par le fichier d'entrée `x.0.0.hg` si $i = 0$, et `x.0.i-1.h` si $i \geq 1$.

3.7 Programme `blqms`

Fichiers d'entrée : x.0.i.mc x.0.i.ms x.0.i.hs
Fichiers de sortie : x.0.i.fmt x.0.i.ms x.0.i.hs

Le programme `blqms` lit un maillage de triangles et le remplace par un maillage de quadrilatères, par une méthode d'appariement.

3.8 Programme `blsmo`

Fichiers d'entrée : x.0.0.g
Fichiers de sortie : x.0.0.smo

Le programme `blsmo` crée un fichier `.smo`. Il est gouverné par la variable d'environnement `blsmo` (cf. section 2.4).

3.9 Programme `bltms`

Fichiers d'entrée :	si $i = 0$: x.0.0.mc	x.0.0.hc		
	si $i \geq 1$: x.0.i.mc	x.0.i.hc	x.0.i-1.ms	x.0.i-1.h
Fichiers de sortie :	si $i = 0$: x.0.0.ms	x.0.0.hs		
	si $i \geq 1$: x.0.i.ms	x.0.i.hs	x.0.i.is	

Le programme `bltms` est un mailleur de domaines : il discrétise chaque sous-domaine en triangles. Les tailles et les formes de ces triangles sont déterminées par le fichier d'entrée x.0.i.hc créé par le mailleur de courbes `blmc`.

4 Les formats de fichiers par ordre alphabétique

Les programmes du logiciel BL2D-V2 créent des fichiers dans différents formats, en évitant de dupliquer inutilement des informations. Les principaux fichiers créés sont le maillage de courbes (format `.mc`) et le maillage de domaines (format `.ms`). Cependant, des informations présentes dans différents fichiers peuvent être regroupées dans un fichier unique, par exemple au format `.amdba` ou `.bl2d`.

4.1 Format `.amdba`

Objectif

L'objectif du format `.amdba` est de décrire un maillage de domaines et ses références physiques. Il est utilisé par certains codes de calcul par éléments finis. Cependant, il est moins général que le format `.bl2d`. Notamment, il ne donne pas les références physiques des arêtes et n'admet pas d'éléments autres que des triangles P^1.

Ces données sont générées par le programme `blcv`.

Définition du format `.amdba`

NP NT
Pour tous les points ..... $i = 1..NP$
i x_i y_i φ_i
Pour tous les triangles ..... $i = 1..NT$
i p_{i1} p_{i2} p_{i3} φ_i

Notations

NP	Nombre de points.
NT	Nombre de triangles.

Pour un point $i = 1..NP$:

x_i y_i	Coordonnées du point.
φ_i	Référence physique du point.

Pour un triangle $i = 1..NT$:

p_{i1} p_{i2} p_{i3}	Numéros des trois sommets donnés dans le sens direct.
φ_i	Référence physique du triangle.

Exemple de fichier au format `.amdba`

Le fichier `.amdba` ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
121   202
1   0.4000000   0.4000000   10
2   0.0000000   0.0000000   20
```

```
3   1.0000000   0.0000000   30
...
1   69   70   98   200
2   96   92   93   200
3   35   58   57   200
...
```

4.2 Format `.bl2d`

Objectif

Le format `.bl2d` sert d'interface générale entre le générateur de maillage BL2D-V2 et divers codes de calcul. Il peut être créé par le programme `blcv`.

Définition du format `.bl2d`

élément
NP NA NE
Pour tous les points ..... $i = 1..NP$
x_i y_i φ_i
Pour toutes les arêtes ..... $i = 1..NA$
p_{i1} p_{i2} ... φ_i
Pour tous les éléments ..... $i = 1..NE$
p_{i1} p_{i2} ... φ_i

Notations

élément	Type des éléments (`p1`, `p2`, ..., cf. section 2.4, variable `element`).
NP	Nombre de points.
NA	Nombre d'arêtes.
NE	Nombre d'éléments.

Pour un point $i = 1..NP$:

x_i y_i	Coordonnées du point.
φ_i	Référence physique du point.

Pour une arête $i = 1..NA$:

p_{i1} p_{i2} ...	Numéros des points qui définissent l'arête.
φ_i	Référence physique de l'arête.

Pour un élément $i = 1..NE$:

p_{i1} p_{i2} ...	Numéros des points qui définissent l'élément.
φ_i	Référence physique de l'élément.

Exemple de fichier au format `.bl2d`

Le fichier `.bl2d` ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
p1
105 38 170
 -1.4910049164396110E-08  -1.4910049164396110E-08 20
  0.9999999850899508  -1.4910049164396110E-08 30
 -1.4910049164396110E-08   0.9999999850899508 40
  0.9999999850899508   0.4999999850899509 50
  0.4999999850899509   0.9999999850899508 60
...
1 6 110
6 7 110
7 8 110
8 9 110
9 10 110
...
23 24 40 200
1 6 38 200
61 16 46 200
53 75 52 200
14 60 13 200
...
```

4.3 Format `.data`

Objectif

Rappelons que le programme `blg` permet de créer les fichiers initiaux `.g` (données géométriques et physiques) et `.hg` (carte de tailles ou de métriques). L'objectif du format `.data` est de permettre à l'utilisateur de décrire ces données de façon aussi simple que possible :

- Les objets géométriques (points, splines, sous-domaines, ...) peuvent être désignés par des noms (identificateurs) et non par des numéros.
- Les nombres d'objets sont calculés automatiquement.
- Les lectures sont en format libre [7]. Il est donc possible d'y inclure des commentaires et des expressions arithmétiques.
- Les métriques anisotropes sont données sous la forme θ, h_1, h_2 (et non pas a, b, c, cf. format `.h` en section 4.5). La valeur h_1 (resp. h_2) est la taille souhaitée le long de l'axe d'angle θ (resp. $\theta + 90°$) par rapport à l'axe horizontal, où θ est donné en degrés.

Définition du format `.data`

NP NS ND
Pour tous les points i :
  id_i x_i y_i
;
Pour tous les points imposés :
  id_i
;
Pour toutes les splines :
  id_i de la spline
  $id.$ du point de contrôle 1 (NULL s'il est absent)
  $id.$ de l'extrémité 1
  $id.$ des sommets intermédiaires
  $id.$ de l'extrémité 2
  $id.$ du point de contrôle 2 (NULL s'il est absent)
  ;
;
Pour tous les points imposés ou extrémités t.q. $\varphi_i \neq 0$:
  id_i φ_i
;
Pour toutes les splines t.q. $\varphi_i \neq 0$:
  id_i φ_i
;
Pour tous les sous-domaines t.q. $\varphi_i \neq 0$:
  id_i d'une spline o_i (+1 ou -1) φ_i
;
`isotrope` ou `anisotrope`
Pour tous les points imposés ou extrémités :
  id_i
  si `isotrope` : h au point i
  si `anisotrope` : θ h_1 h_2 au point i
;

Notations

NP	Nombre **maximal** de points.
NS	Nombre **maximal** de splines.
ND	Nombre **maximal** de sous-domaines.
id_i	Identificateur d'un objet i (point, spline ou sous-domaine).
x_i y_i	Coordonnées du point i.
φ_i	Référence physique d'un objet i (point, spline ou sous-domaine).
h	Taille prescrite

θ h_1 h_2 Angle et tailles prescrits (voir plus haut, section Objectif).

Exemple de fichier au format `.data`

Le fichier `.data` de la figure 7 a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

4.4 Format `.g`

Objectif

L'objectif du format `.g` est de décrire les données géométriques et physiques d'un domaine du plan. Ces données sont normalement générées par un système interactif graphique ou par un préprocesseur.

Rappels

Un domaine peut être composé d'un ou de plusieurs **sous-domaine(s)**. Chaque sous-domaine est délimité par sa **frontière**. Plusieurs maillages peuvent être réalisés sur un même domaine (par exemple pour adapter le maillage, ou encore pour résoudre un problème multi-physique). Tout maillage doit respecter les frontières des sous-domaines. On peut également imposer que le maillage s'appuie sur des segments ou des points internes à un sous-domaine (on parle alors de **segment imposé** ou de **point imposé**). Un segment interne a au plus une extrémité commune avec un segment de frontière (dans le cas contraire, il devient lui-même un segment de frontière).

Dans l'exemple de la figure 10, le domaine Ω comprend deux sous-domaines Ω_1 et Ω_2. Le sous-domaine Ω_1 a pour frontière la composante connexe $\Gamma_1 \cup \Gamma_2$. Le sous-domaine Ω_2 a pour frontière les deux composantes connexes $\Gamma_2 \cup \Gamma_3$ et Γ_4 (il comporte un "trou"). Il contient le segment imposé S et le point imposé P.

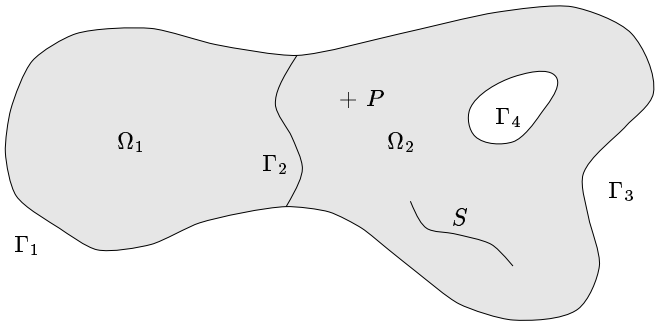


FIG. 10 – *Exemple de domaine* $\Omega = \Omega_1 \cup \Omega_2$.

Des **propriétés physiques** peuvent être associées aux entités que nous venons de citer (sous-domaines, frontières, segments imposés et points imposés). Par exemple, dans le cas de la simulation d'un problème thermique, on peut associer à un sous-domaine sa conductivité et sa source de chaleur, à une frontière son coefficient de transfert et sa température extérieure, et à un point imposé sa température.

Les splines

Il est commode de représenter chaque segment de frontière ou segment imposé par une ou plusieurs **spline(s)**. Une spline seule ne contient pas de point anguleux. En revanche, le point de jonction entre deux splines consécutives peut être anguleux (figure 11).

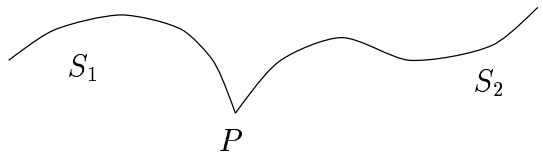


FIG. 11 – *Segment courbe formé de deux splines S_1 et S_2.*

Parmi les nombreuses méthodes utilisées pour définir des splines, l'une des plus simples est celle de Catmull-Rom [10]. Dans l'exemple de la figure 12, la spline d'extrémités P_2 et P_5 est définie par le polygone de contrôle $P_1P_2P_3P_4P_5P_6$. Le morceau de spline P_2P_3 est un morceau cubique défini par ses extrémités P_2 et P_3, par le vecteur tangent en P_2 égal à $\overrightarrow{P_1P_3}/2$, et par le vecteur tangent en P_3 égal à $\overrightarrow{P_2P_4}/2$. Les morceaux cubiques P_3P_4 et P_4P_5 sont définis de manière analogue. D'autres méthodes sont proposées par le programme `blsmo`, notamment celle de De Boor qui est prise par défaut.

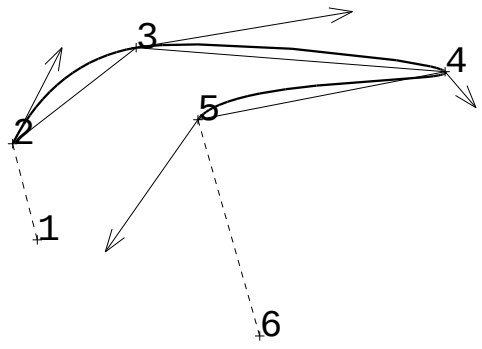


FIG. 12 – *Une spline et son polygone de contrôle.*

De façon plus générale, une spline est décrite par un polygone de contrôle $P_1 ... P_n$ ($n \geq 4$, P_1 et P_n facultatifs). Parmi les sommets de ce polygone, on peut distinguer :

- Les extrémités de la spline : P_2 et P_{n-1}.
- Les points de contrôle internes à la spline, si $n > 4$: P_3 ... P_{n-2}. La spline est dite interpolante car elle passe par les points P_2 P_3 ... P_{n-2} P_{n-1}.
- Les deux points de contrôle externes à la spline : P_1 et P_n. Si le point P_1 (resp. P_n) est présent, il sert à fixer la tangente à l'extrémité P_2 (resp. P_{n-1}) de la spline, en utilisant la méthode ci-dessus (Catmull-Rom). Si un point est absent, la tangente à l'extrémité associée est considérée comme libre.

Ainsi, chaque morceau de spline P_2P_3, ..., $P_{n-2}P_{n-1}$ est défini par ses deux extrémités et éventuellement les tangentes en ses deux extrémités. Si les tangentes sont toutes deux fixées, le morceau est cubique (degré ≤ 3). Si une seule tangente est fixée, le morceau est parabolique (degré ≤ 2). Enfin, si aucune tangente n'est fixée, le morceau est linéaire (degré ≤ 1). Il en résulte que, si seules les deux extrémités de la spline sont définies (c'est-à-dire qu'il n'existe aucun point de contrôle, ni interne ni externe), la spline se réduit à un segment de droite.

Dans le cas particulier d'un arc de cercle approché par une spline, il faut veiller à ce que le nombre n de sommets du polygone de contrôle soit suffisant pour atteindre la précision souhaitée (voir figure 13).

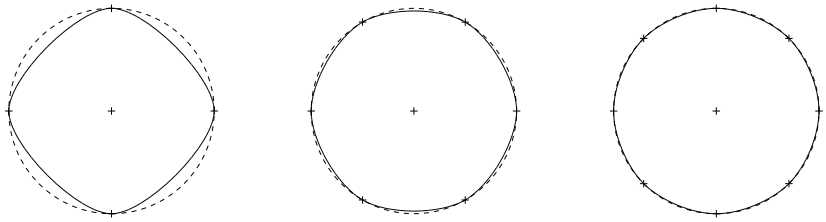


FIG. 13 – *Cercle défini par 4, 6 ou 8 points de contrôle.*

Définition du format .g

NP NPI NPE NPG NS ND
Pour tous les points ..... $i = 1..NP$
x_i y_i
Pour tous les points imposés ..... $i = 1..NPI$
p_i φ_i
Pour toutes les extrémités ..... $i = 1..NPE$
p_i φ_i
Pour toutes les splines ..... $i = 1..NS$

np_i
$(p_{ij})_{\ j=1..np_i}$ φ_i
Pour tous les sous-domaines ..... $i = 1..\mathrm{ND}$
s_i o_i φ_i

Notations

NP	Nombre total de points : NP = NPI + NPE + NPG.
NPI	Nombre de points imposés.
NPE	Nombre d'extrémités.
NPG	Nombre de points restants (ni imposés ni extrémités).
NS	Nombre de splines.
ND	Nombre de sous-domaines.

Pour un point $i = 1..\mathrm{NP}$:

x_i y_i	Coordonnées du point.

Pour un point imposé, $i = 1..\mathrm{NPI}$:

p_i	Numéro du point.
φ_i	Référence physique du point.

Pour une extrémité, $i = 1..\mathrm{NPE}$:

p_i	Numéro du point.
φ_i	Référence physique du point.

Pour une spline $i = 1..\mathrm{NS}$:

np_i	Nombre de points servant à définir la spline ($np_i \geq 4$).
p_{ij}	Numéros des points servant à définir la spline, donnés dans l'ordre suivant :

$p_{i,1}$	Numéro du point de contrôle externe 1. Si ce point est absent, $p_{i,1} = 0$.
$p_{i,2}$	Numéro de l'extrémité 1.
$p_{i,3..np_{i-2}}$	Numéros des points de contrôle internes à la spline, en allant de l'extrémité 1 vers l'extrémité 2.
$p_{i,np_{i-1}}$	Numéro de l'extrémité 2.
p_{i,np_i}	Numéro du point de contrôle externe 2. Si ce point est absent, $p_{i,np_i} = 0$.

φ_i	Référence physique de la spline.

Pour un sous-domaine $i = 1..\mathrm{ND}$:

s_i	Numéro d'une spline appartenant à la frontière.
o_i	Orientation telle que le sous-domaine soit à gauche de la spline s_i. Si $o_i = 1$, la spline est parcourue de l'extrémité 1 vers l'extrémité 2. Si o_i = -1, la spline est parcourue dans l'autre sens.
φ_i	Référence physique du sous-domaine.

Remarques

La première boucle $i = 1..NP$ définit les coordonnées de tous les points utilisés dans le format `.g`. Ces coordonnées doivent être toutes différentes.

Par définition, un point est dit :
- **imposé** s'il apparaît dans la boucle $i = 1..NPI$. Les numéros p_i des points doivent être tous différents.
- **extrémité** s'il apparaît dans la boucle $i = 1..NPE$. Les numéros p_i des points doivent aussi être tous différents. En outre, cette boucle doit contenir tous les points qui apparaissent en tant qu'extrémités dans la description des splines (points $p_{i,2}$ et $p_{i,np_{i-1}}$).
- **point de contrôle externe** s'il apparaît en tant que tel dans la description des splines (points $p_{i,1}$ et p_{i,np_i}).
- **point de contrôle interne** s'il apparaît en tant que tel dans la description des splines (points $p_{i,3..np_{i-2}}$).
- **avec métrique** si le fichier `.hg` définit en ce point une taille ou une métrique fixée ($h_i \neq 0$ ou $a_i \neq 0$, cf. section 4.5).

Les différentes combinaisons possibles sont précisées dans le tableau symétrique ci-dessous (avec l'abréviation p.c. = point de contrôle) :

—	imposé	extrémité	p.c. externe	p.c. interne	avec métrique
imposé	—	interdit	possible	possible	obligatoire
extrémité		—	possible	interdit	obligatoire
p.c. externe			—	possible	inutile
p.c. interne				—	possible
avec métrique					—

La combinaison "imposé - extrémité" est interdite afin de numéroter de façon systématique les points des maillages (cf. sections 4.7 et 4.8).

La combinaison "imposé - p.c. externe" est utile dans la définition de certaines splines.

La combinaison "imposé - p.c. interne" permet de d'imposer un point au mailleur de courbes sans diviser artificiellement une spline.

La combinaison "extrémité - p.c. externe" est utile dans le cas des courbes presque fermées.

La combinaison "extrémité - p.c. interne" est interdite car deux splines ne peuvent être adjacentes que si elles ont des extrémités communes.

La combinaison "p.c. externe - p.c. interne" est utile dans le cas des courbes fermées.

La combinaison "p.c. interne - avec métrique" permet de contrôler plus finement le mailleur de courbes.

Dans la description d'un point, d'une spline ou d'un sous-domaine, on appelle **référence physique** φ_i un nombre entier qui est utilisé comme index vers les propriétés physiques

de cette entité. Par convention, $\varphi_i = 0$ indique qu'aucune propriété physique ne doit être associée à l'entité.

Exemple de fichier au format .g

Le fichier .g ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
 9 1 5 3 5 1
   0.0000000000000000   0.0000000000000000
   1.0000000000000000   0.0000000000000000
   0.0000000000000000   1.0000000000000000
   0.4000000000000000   0.4000000000000000
   1.3535533905932737   0.6464466094067263
   1.0000000000000000   0.5000000000000000
   0.6464466094067263   0.6464466094067263
   0.5000000000000000   1.0000000000000000
   0.6464466094067263   1.3535533905932737

 4 10

 1 20
 2 30
 3 40
 6 50
 8 60

 4
 0 1 2 0 110
 4
 0 2 6 0 120
 5
 5 6 7 8 9 130
 4
 0 8 3 0 140
 4
 0 3 1 0 150

 1 1 200
```

4.5 Format .h (ou .hg, .hc, .hs)

Objectif

Un mailleur doit créer des éléments de courbes ou de domaines. L'objectif du format .h est de gouverner ce mailleur en donnant, en certains points d'un domaine, la taille souhaitée des éléments à créer au voisinage du point (cas isotrope) ou la métrique au voisinage du point (cas anisotrope). Les points du domaine sont eux-mêmes définis dans le format .g (données géométriques et physiques), .mc (maillage de courbes) ou .ms (maillage de domaines). Il en résulte des fichiers de suffixe .hg, .hc, .hs (carte de tailles calculée par interpolation) ou .h (carte de tailles générée par un estimateur d'erreur).

Définitions préalables

La carte de tailles ou de métriques est définie sur un ensemble de points, qui sont eux-mêmes donnés dans le format `.g`, `.mc` ou `.ms`.

Si le maillage à créer est **isotrope**, il suffit de donner en chaque point la taille souhaitée des éléments à générer au voisinage de ce point.

Si le maillage à créer est **anisotrope**, la méthode retenue consiste à fournir en chaque point une *métrique* dans laquelle la taille souhaitée est égale à l'unité [2]. Une métrique est représentée par une matrice symétrique définie positive à trois coefficients (a, b, c). En prenant pour origine le point où la métrique est donnée, tout point (x, y) situé à une distance 1 de l'origine, dans la métrique (a, b, c), vérifie l'équation :

$$\begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 1 \qquad \Longleftrightarrow \qquad a\,x^2 + 2\,b\,x\,y + c\,y^2 = 1$$

Il s'agit de l'équation d'une ellipse centrée à l'origine. Par une rotation d'angle θ, choisi de manière à rendre le repère parallèle à l'un des deux axes de l'ellipse (figure 14), l'équation se met sous la forme simplifiée :

$$\frac{X^2}{h_1^2} + \frac{Y^2}{h_2^2} = 1$$

h_1 et h_2 représentent les tailles souhaitées selon deux directions orthogonales, dans la métrique usuelle égale à l'identité. Inversement, si l'on connaît θ, h_1 et h_2, il est facile d'obtenir la métrique (a, b, c) grâce à la relation :

$$\begin{pmatrix} a & b \\ b & c \end{pmatrix} = P \begin{pmatrix} \dfrac{1}{h_1^2} & 0 \\ 0 & \dfrac{1}{h_2^2} \end{pmatrix} P^{-1}$$

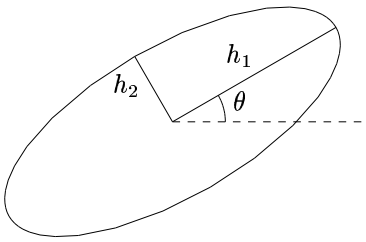


FIG. 14 – *Ellipse définie par θ, h_1 et h_2.*

où P est la "matrice de passage" :

$$P = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \qquad \text{et} \qquad P^{-1} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

Définition du format `.h`

NP NC
Pour chaque point ..... $i = 1..\mathrm{NP}$
si `NC` $= 1$: h_i
si `NC` $= 3$: a_i b_i c_i

Notations

NP	Nombre de points où sont définies les tailles ou les métriques.
NC	Nombre de colonnes (1 dans le cas isotrope, 3 dans le cas anisotrope).

Pour un point $i = 1..\mathrm{NP}$:

h_i	Taille au point i (cas isotrope).
$a_i\ b_i\ c_i$	Métrique au point i (cas anisotrope).

Remarques

Il est parfois utile de laisser libre la taille ou la métrique en certains points. Par convention, la taille ou la métrique est considérée comme libre en tout point p_i tel que $h_i = 0$ (cas isotrope) ou $a_i = 0$ (cas anisotrope).

Exemple de fichier au format `.h`

Le fichier `.h` ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
9
isotrope
  0.1000000000000000
  0.1000000000000000
  0.1000000000000000
  0.1000000000000000
  0.0000000000000000
  0.1000000000000000
  0.1000000000000000
  0.1000000000000000
  0.0000000000000000
```

4.6 Format `.is`

Objectif

Au cours d'une boucle d'adaptation, plusieurs maillages d'un même domaine sont générés. L'objectif du format `.is` est de donner les coordonnées barycentriques des sommets d'un "nouveau maillage" (*foreground mesh*) par rapport à ceux d'un "ancien maillage" (*background mesh*). Ces données sont créées en vue de l'interpolation de la solution d'un calcul par éléments finis.

Définition du format `.is`

NP
Pour tous les points du nouveau maillage ..... $i = 1..\mathrm{NP}$
p_{i1} p_{i2} p_{i3} λ_{i1} λ_{i2}

Notations

NP	Nombre de points du nouveau maillage.

Pour un point du nouveau maillage $i = 1..\mathrm{NP}$:

p_{i1} p_{i2} p_{i3}	Numéros de trois points de l'ancien maillage, ou 0 (cf. remarques).
λ_{i1} λ_{i2}	Coordonnées barycentriques (cf. remarques).

Remarques

Soit P_i le point numéro i (dans le nouveau maillage).

En général, (p_{i1}, p_{i2}, p_{i3}) sont les numéros des sommets (P_{i1}, P_{i2}, P_{i3}) du triangle de l'ancien maillage qui englobe le point P_i. Alors P_i est défini vectoriellement par :

$$P_i = \lambda_{i1}\, P_{i1} + \lambda_{i2}\, P_{i2} + (1 - \lambda_{i1} - \lambda_{i2})\, P_{i3}$$

Cependant, il est possible qu'il n'existe pas de triangle englobant (par exemple si une frontière courbe est remaillée, ou si un segment courbe est défini extérieurement au domaine). Dans ce cas, seuls les deux points englobants dans le maillage de **courbes** sont considérés. Par convention, le numéro du troisième point est nul. Les coordonnées barycentriques sont alors calculées en fonction des abscisses curvilignes. Par exemple, si $p_{i3} = 0$, on aura :

$$\lambda_{i1} = \lambda \qquad \lambda_{i2} = 1 - \lambda \qquad (1 - \lambda_{i1} - \lambda_{i2}) = 0$$

Exemple de fichier au format `.is`

Le fichier `.is` ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
204
 49     1    65   0.000000   1.000000
```

```
  2    39     0   1.000000   0.000000
  3    16     0   1.000000   0.000000
  4    31     0   1.000000   0.000000
  5    20   101   1.000000   0.000000
  6   105    26   1.000000   0.000000
  7     8     0   0.672134   0.327866
  8     9    61   0.422772   0.577228
  9    10     0   0.242179   0.757821
...
 30     4    71   0.051587   0.172009
  3    16    48   0.087147   0.073293
  5   101    19   0.457181   0.475098
```

4.7 Format `.mc` (ou `.smo`)

Objectif

L'objectif du format `.mc` est de décrire toutes les données géométriques nécessaires au mailleur de domaines `bltms`. Il décrit donc essentiellement un maillage de courbes. Ces données sont générées par le programme `blmc`.

Les fichiers de suffixe `.smo`, générés par le programme `blsmo` pour approcher les splines par des segments polygonaux suffisamment fins, utilisent ce même format.

Définition du format `.mc`

élément
NP NA ND
bb_1 bb_2 bb_3 bb_4
Pour tous les points ..... $i = 1..NP$
  x_i y_i
Pour toutes les arêtes ..... $i = 1..NA$
  typc p_{i1} p_{i2} ...
Pour tous les sous-domaines ..... $i = 1..ND$
  q_{i1} q_{i2}

`supplement` (ce mot-clé et les informations qui suivent sont optionnels)
NP NPI NPE NPC NS
Pour toutes les splines ..... $i = 1..NS$
  np_i
  $(p_{ij}$ $s_{ij})$ $_{j=1..np_i}$
NTG `tgvectors`
Pour toutes les splines, si NTG $\neq 0$ ..... $i = 1..NTG$
tg_{i1} tg_{i2} tg_{i3} tg_{i4}

Notations

élément	Type des éléments (`p1`, `p2`, ..., cf. section 2.4, variable `element`).
NP	Nombre de points.
NA	Nombre d'arêtes.
ND	Nombre de sous-domaines.
$bb_1..bb_4$	Boîte englobante (*bounding box*) : (bb_1, bb_2) sont les coordonnées du coin inférieur gauche, (bb_3, bb_4) sont les coordonnées du coin supérieur droit.

Pour un point $i = 1..\text{NP}$:

$x_i\ y_i$	Coordonnées du point.

Pour une arête $i = 1..\text{NA}$:

typc	Type de l'arête, forcé ici à 1.
$p_{i1}\ p_{i2}$ ...	Numéros des points qui definissent l'arête.

Pour un sous-domaine $i = 1..\text{ND}$:

$q_{i1}\ q_{i2}$	Numéros des deux extrémités d'une arête frontière ou interne au sous-domaine. L'ordre de ces extrémités est tel que le sous-domaine soit à gauche en allant de q_{i1} vers q_{i2}.

Sauf exception (notamment un fichier généré par le mailleur surfacique BLSURF [11, 12]), la fin de ce fichier contient le mot-clé `supplement` suivi des informations suivantes :

NP, NPI, NPE, NPC, NS	Même signification que l'entête du format `.g`.

Pour une spline $i = 1..\text{NS}$:

np_i	Nombre de points.
$p_{ij}\ s_{ij}$	Numéro et abscisse curviligne du point.
NTG	Nombre de splines dont les vecteurs tangents sont donnés, c'est-à-dire 0 pour un fichier `.mc` ou NS pour un fichier `.smo`.

Pour une spline $i = 1..\text{NTG}$:

$tg_{i1}..tg_{i4}$	Vecteurs tangents, de coordonnées (tg_{i1}, tg_{i2}) à l'extrémité 1 et (tg_{i3}, tg_{i4}) à l'extrémité 2.

Exemple de fichier au format `.mc`

Le fichier `.mc` ci-dessous a été utilisé dans l'exemple du "quart de plaque trouée" (cf. section 2.2).

```
p1
        39          38           1
   0.00000000000000000E+00   0.00000000000000000E+00
   0.10000000000000000E+01   0.10000000000000000E+01

   0.40000000000000002E+00   0.40000000000000002E+00
   0.00000000000000000E+00   0.00000000000000000E+00
   0.10000000000000000E+01   0.00000000000000000E+00
   0.00000000000000000E+00   0.10000000000000000E+01
   0.10000000000000000E+01   0.50000000000000000E+00
```

```
...
          1          2          7
          1          7          8
          1          8          9
          1          9         10
          1         10         11
...
 supplement
 39 1 5 33 5
 11
 2  0.0000000000000000E+000
 7   0.1000000000000000
 8   0.2000000000000000
 9   0.3000000000000000
 10   0.4000000000000000
...
 0  tgvectors
```

4.8 Format `.ms`

Objectif

Dans un maillage de domaines, chaque sous-domaine est subdivisé en triangles ou en quadrilatères. L'objectif du format `.ms` est de décrire cet ensemble d'éléments.

Ces données sont générées par le programme `bltms`.

Définition du format `.ms`

élément
NP NE NS
Pour tous les points ..... $i = 1..NP$
x_i y_i
Pour tous les éléments ..... $i = 1..NE$
p_{i1} p_{i2} ... d_i

`supplement` (ce mot-clé et les informations qui suivent sont optionnels)
Pour tous les triangles ..... $i = 1..NE$
v_{i1} v_{i2} v_{i3}

Notations

élément	Type des éléments (`p1`, `p2`, ..., cf. section 2.4, variable `element`).
NP	Nombre de points.
NE	Nombre d'éléments.
ND	Nombre de sous-domaines.

Pour un point $i = 1..NP$:

x_i y_i	Coordonnées du point.

Pour un élément $i = 1..\mathrm{NE}$:

$p_{i1}\ p_{i2}\ \ldots$ Numéros des points qui définissent l'élément.

d_i Numéro du sous-domaine :
- $d_i = 0$ pour le sous-domaine compris entre l'objet et la *bounding box*,
- $1 \leq d_i \leq \mathrm{ND}$ pour les sous-domaines définis dans `.g`,
- $d_i \geq \mathrm{ND} + 1$ pour les autres sous-domaines ("trous").

Si les variables d'environnement sont telles que `element` = `p1` et `adjacency` = 1, la fin de ce fichier contient le mot-clé `supplement` et, pour chaque triangle $i = 1..\mathrm{NE}$, les informations suivantes :

$v_{i1}\ v_{i2}\ v_{i3}$ Numéros des trois triangles voisins.

Remarques

Les points sont numérotés de 1 à NP en considérant successivement :
- les NPI points imposés définis dans `.g`,
- les NPE extrémités définies dans `.g`,
- les NPC points créés par le mailleur de courbes (cf. section 4.7),
- les points créés par le mailleur de domaines,
- les 4 points de la boîte englobante (*bounding box*).

Les références physiques des points, des arêtes et des éléments, qui sont déjà décrites dans le format `.g`, ne sont pas dupliquées dans le format `.ms`. Elles peuvent être relues dans un fichier `.g` ou obtenues dans un fichier `.bl2d`.

Exemple de fichier au format `.ms`

Le fichier `.ms` ci-dessous a été obtenu dans l'exemple du "quart de plaque trouée" (cf. section 2.2), en lançant la commande `bl2d.sh -rigid 0 -adjacency 1`.

```
p1
107 208 1
 3.99999985089950830e-01   3.99999985089950830e-01
-1.49100491643961130e-08  -1.49100491643961130e-08
 9.99999985089950800e-01  -1.49100491643961130e-08
-1.49100491643961130e-08   9.99999985089950800e-01
 9.99999985089950800e-01   4.99999985089950860e-01
...
     12      11     105   0
     17      44      43   1
    104       5     105   0
...
supplement
     24       5     132
     80      71      98
     54       0      33
...
```

5 Exemples d'applications

5.1 Arobase : méthodes algébrique et frontale

Dans ce premier exemple, la frontière du domaine est un cercle de rayon 50. À l'intérieur, une courbe en forme d'arobase (@) est imposée. La figure 15 représente un maillage "grossier" avec une taille requise de 10 sur le cercle extérieur et de 0.5 sur la courbe imposée, avec une gradation de 1.4. La figure 16 représente un maillage plus fin avec une taille requise de 5 sur le cercle extérieur et de 0.25 sur la courbe imposée, avec une gradation de 1.25. Sur chacune de ces deux figures, la partie gauche représente le maillage obtenu par une méthode algébrique, et la partie droite par une méthode frontale. Par ailleurs, la partie inférieure est une vue agrandie de la partie supérieure. Ces figures illustrent le fait que la méthode frontale produit un maillage de meilleure qualité (triangles presque équilatéraux, meilleure régularité, degré moins élevé des nœuds, fronts visibles partant de la frontière et de la courbe imposée).

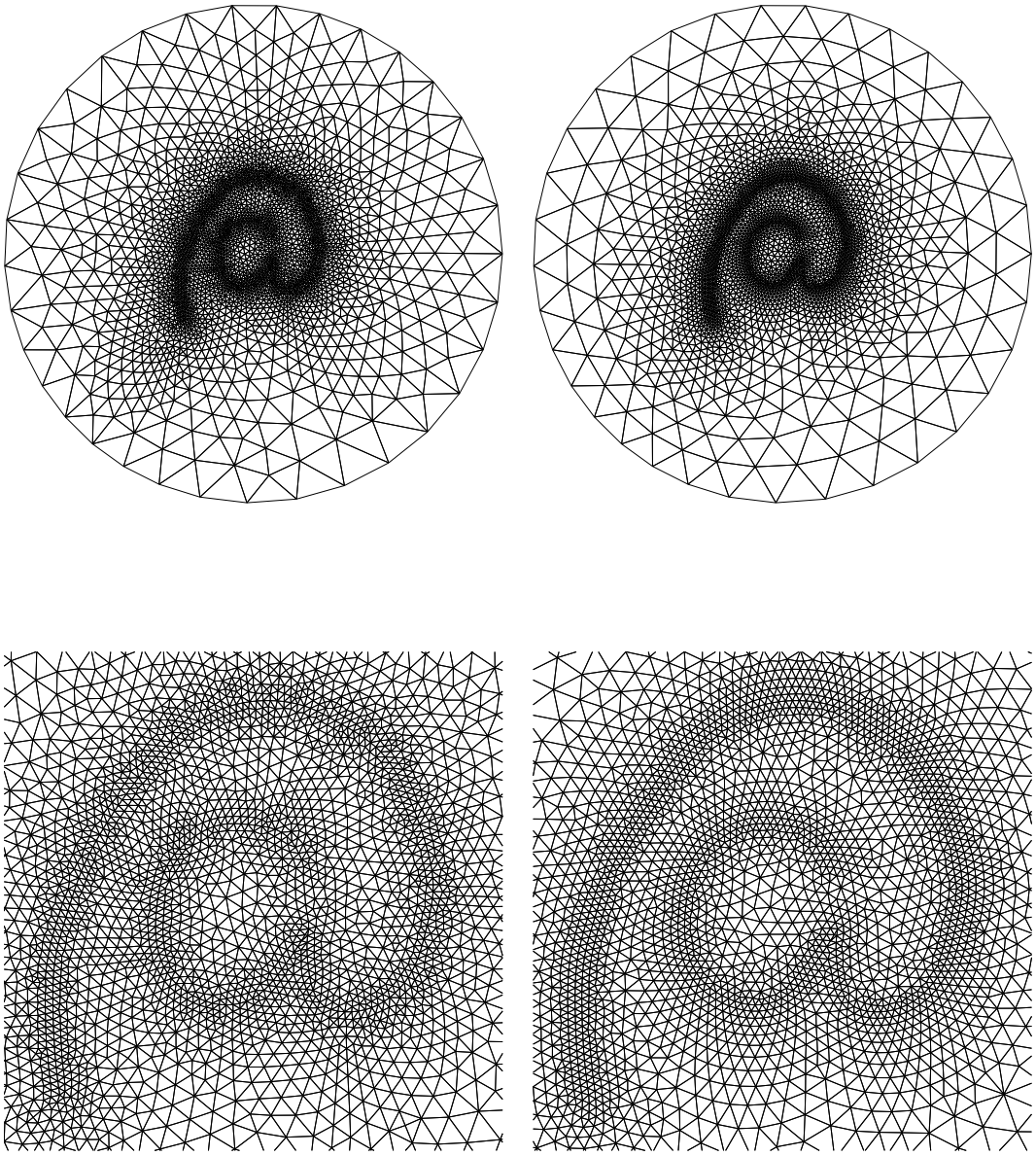


FIG. 15 – *Maillage "grossier". À gauche, méthode algébrique. À droite, méthode frontale. La partie inférieure est une vue agrandie de la partie supérieure.*

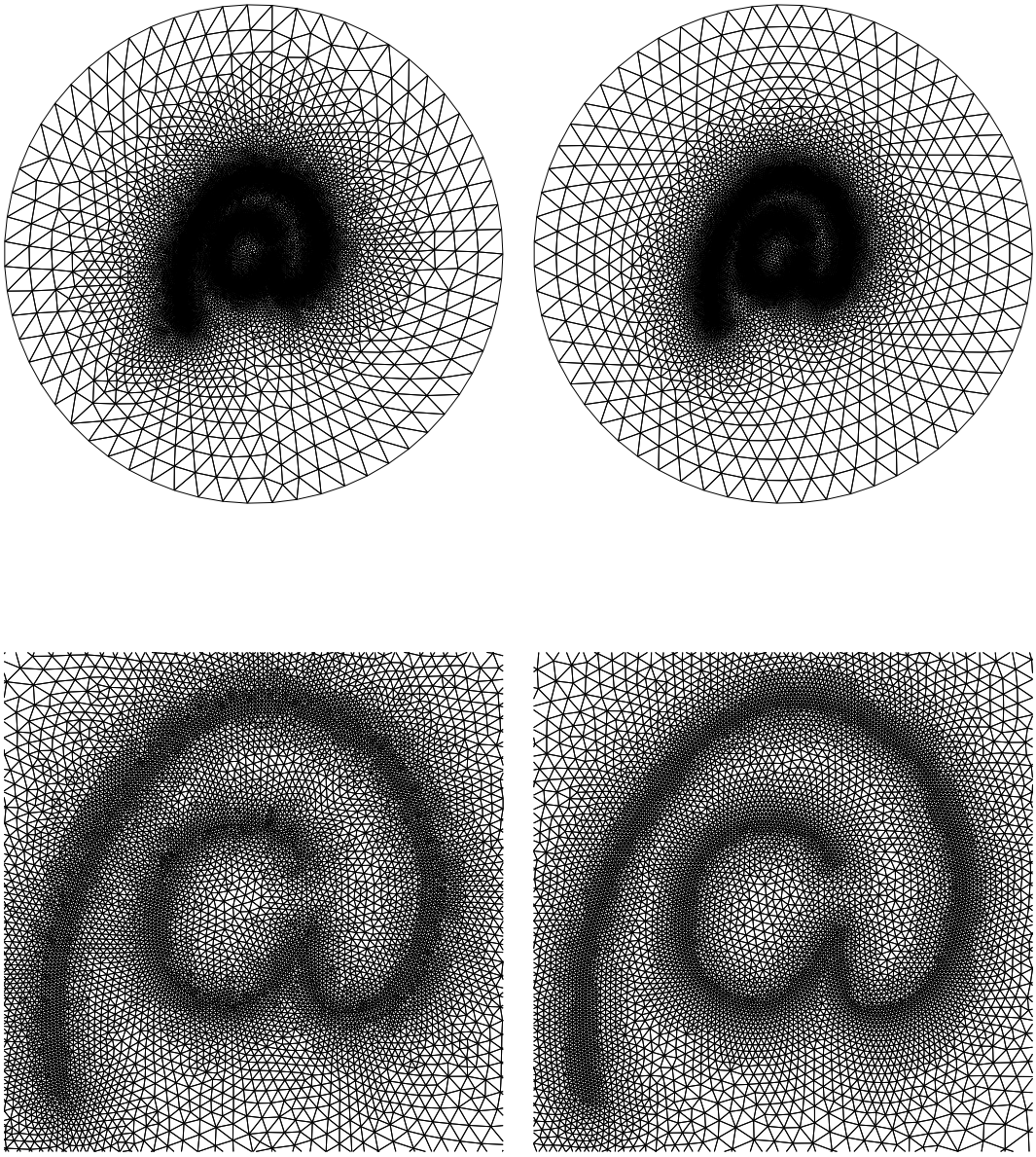


FIG. 16 – *Maillage "fin". À gauche, méthode algébrique. À droite, méthode frontale. La partie inférieure est une vue agrandie de la partie supérieure.*

5.2 Forgeage : frontières déformables

Pour illustrer le cas des frontières déformables, on considère le problème de forgeage d'une pièce mécanique. La configuration initiale de la pièce et de l'outil est présentée sur la figure 17. L'outil est composé de deux parties, l'une haute et l'autre basse. Le procédé consiste à déformer par compression la pièce en descendant la partie haute de l'outil (la partie basse restant fixe) jusqu'à ce que la pièce épouse totalement la forme de l'outil. La déformation de la pièce est obtenue par contact avec l'outil (voir figures 18 à 23).

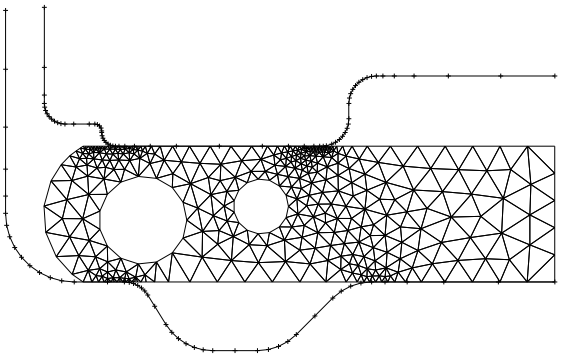


FIG. 17 – *Maillage initial de la pièce*

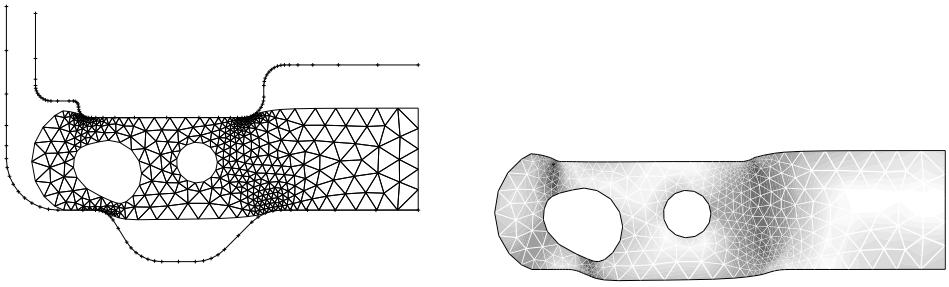


FIG. 18 – *Remaillage et contraintes de Von-Mises à* $\delta = 10$ *mm*

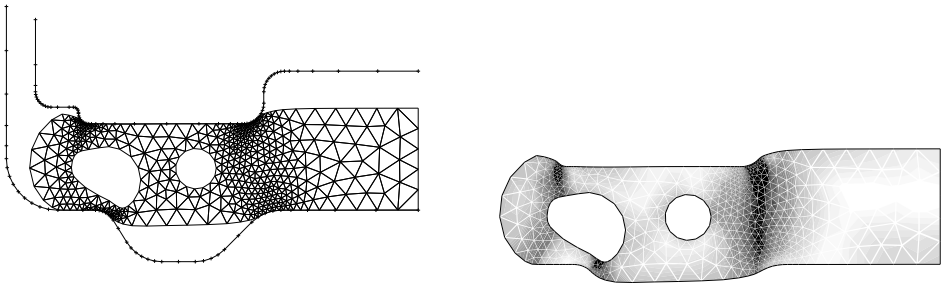


FIG. 19 – *Remaillage et contraintes de Von-Mises à* $\delta = 16$ *mm*

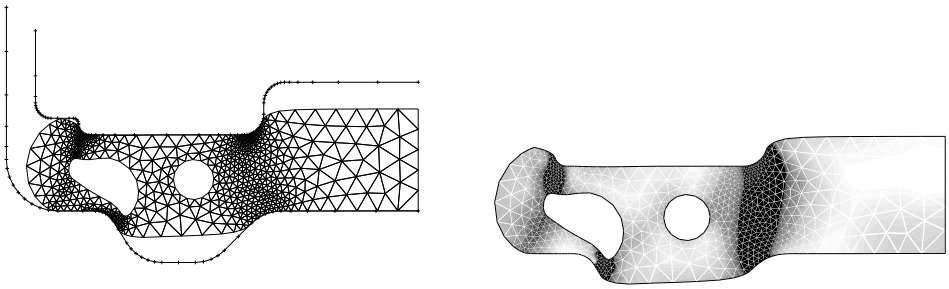


FIG. 20 – *Remaillage et contraintes de Von-Mises à* $\delta = 26$ *mm*

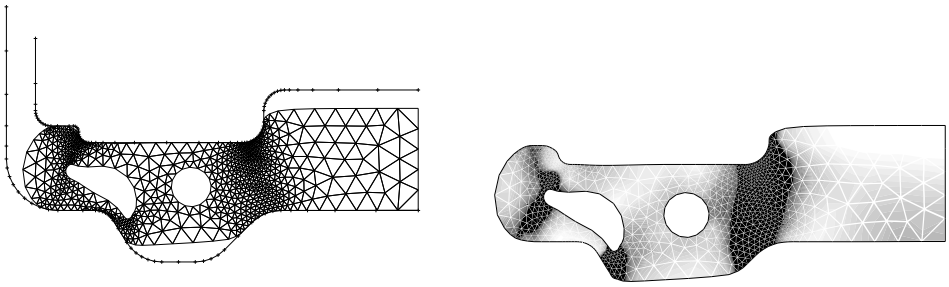


FIG. 21 – *Remaillage et contraintes de Von-Mises à* $\delta = 34$ *mm*

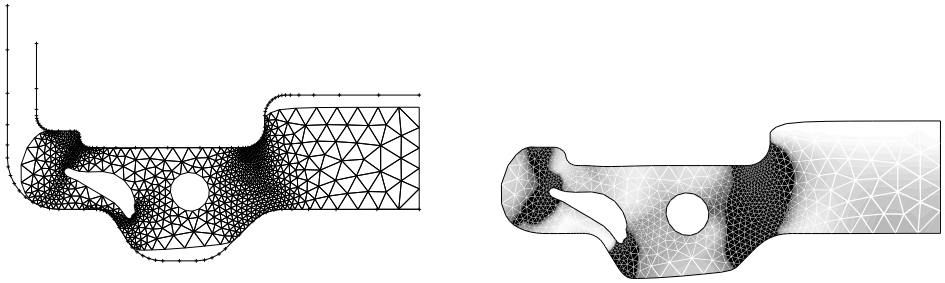


FIG. 22 – *Remaillage et contraintes de Von-Mises à* $\delta = 40$ *mm*

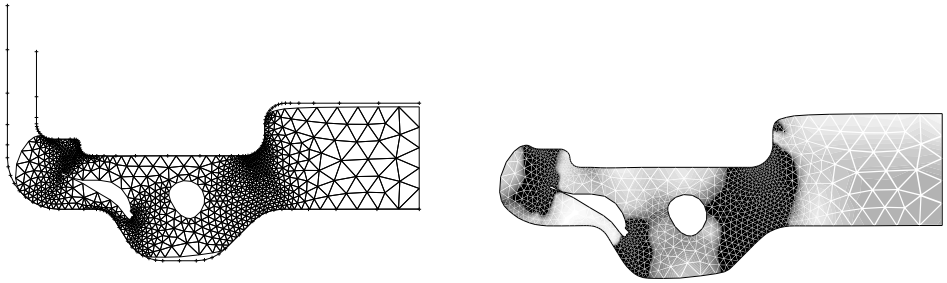


FIG. 23 – *Remaillage et contraintes de Von-Mises à* $\delta = 48$ *mm*

5.3 Quart de plaque trouée : choix des éléments

La figure 24 a été obtenue en reprenant les données du "quart de plaque trouée" de la figure 7), en supprimant le point imposé P, et en lançant la commande `bl2d.sh -rigid 0 -element p1` (resp. `p2`, `q1.0`, ..., cf. section 2.4, variable `element`).

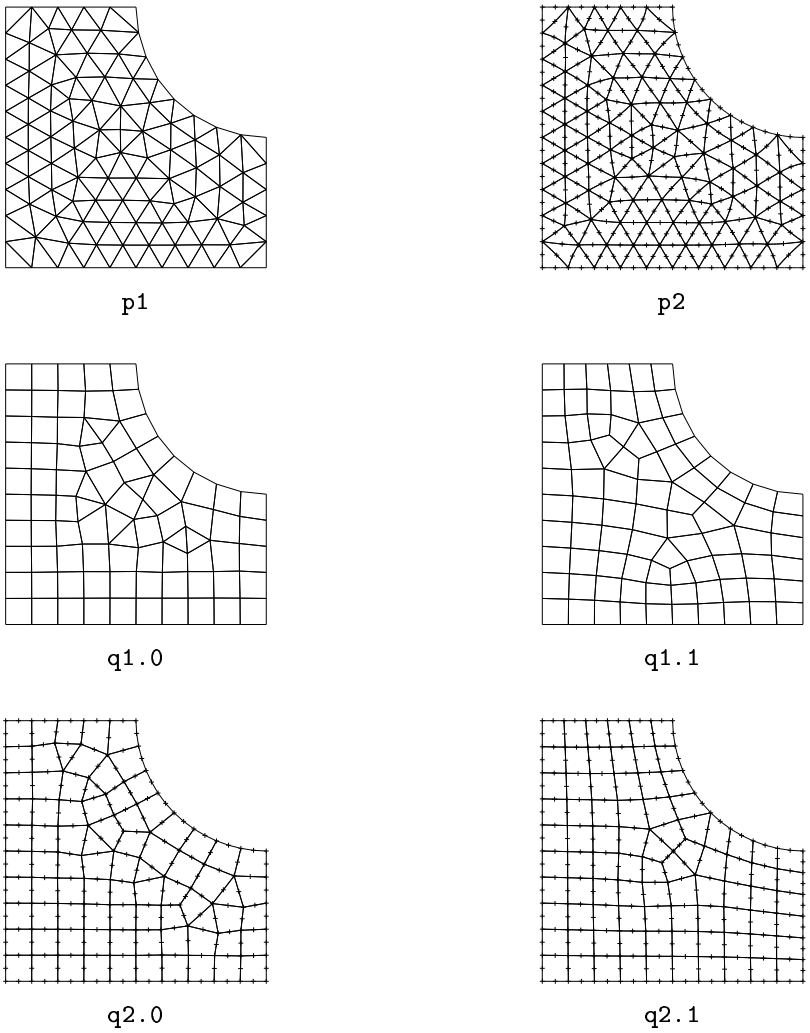


FIG. 24 – *Maillages du "quart de plaque trouée" constitués de divers éléments.*

5.4 Mécanique des fluides : quadrilatères isotropes et anisotropes

Les figures 25 et 26 montrent des maillages en quadrilatères obtenus par appariement de triangles, respectivement isotropes et anisotropes, pour une application en mécanique des fluides (CFD).

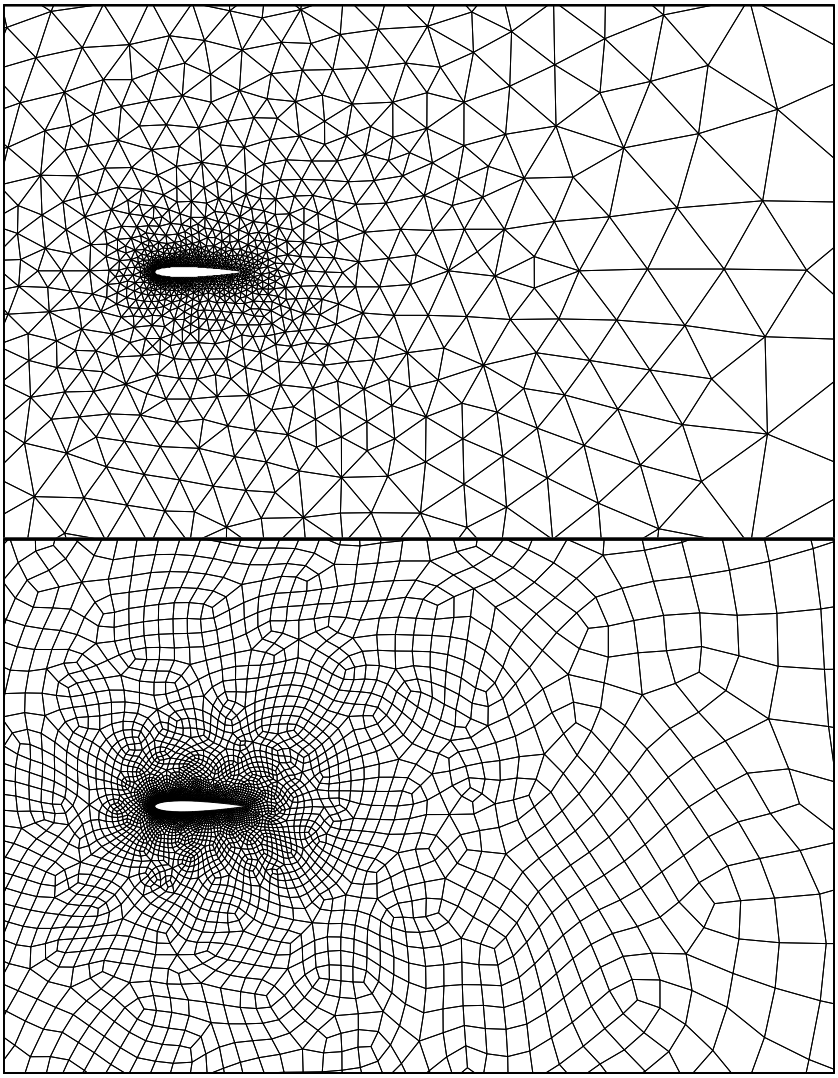


FIG. 25 – *Maillages CFD* **isotropes***: triangles initiaux (en haut) et quadrilatères (en bas).*

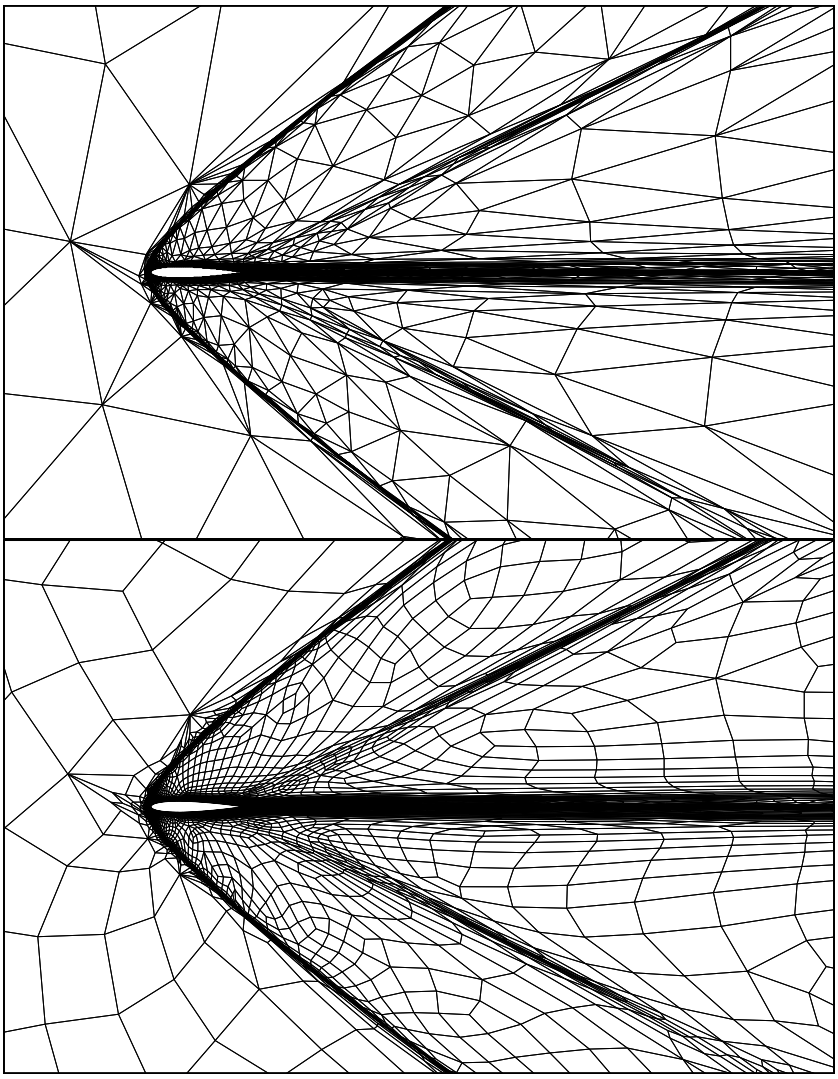


FIG. 26 – *Maillages CFD* **anisotropes***: triangles initiaux (en haut) et quadrilatères (en bas).*

5.5 Photos d'identité : triangles anisotropes

Pour conclure de façon moins formelle, les figures 27 et 28 montrent des maillages anisotropes adaptés à des photos d'identité des deux auteurs.

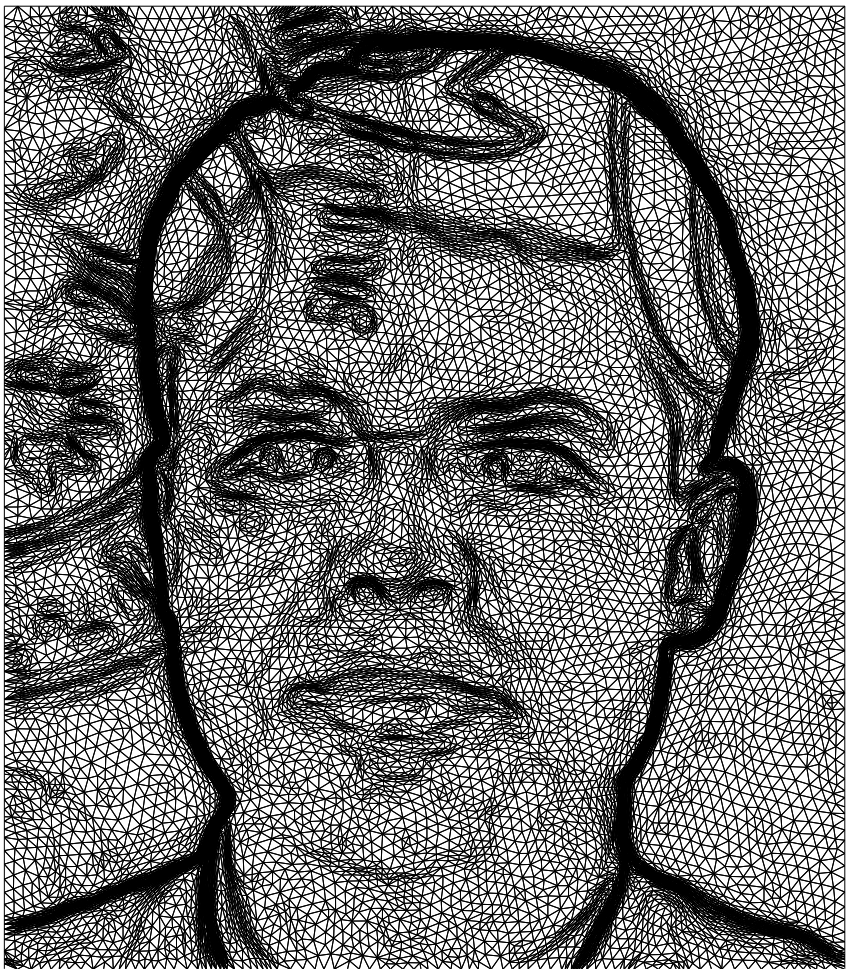


FIG. 27 – *P. Laug : maillage anisotrope.*

FIG. 28 – *H. Borouchaki : maillage anisotrope.*

Références

[1] http://www-rocq.inria.fr/Patrick.Laug. *Le logiciel BL2D-V2.*

[2] P. Laug, H. Borouchaki, P.L. George. *Maillage de courbes gouverné par une carte de métriques.* Rapport INRIA RR-2818, mars 1996.

[3] H. Borouchaki, P.L. George, F. Hecht, P. Laug, E. Saltel. *Mailleur bidimensionnel de Delaunay gouverné par une carte de métriques. Partie I: Algorithmes.* Rapport INRIA RR-2741, décembre 1995.

[4] H. Borouchaki, P.L. George, F. Hecht, P. Laug, B. Mohammadi, E. Saltel. *Mailleur bidimensionnel de Delaunay gouverné par une carte de métriques. Partie II: Applications.* Rapport INRIA RR-2760, décembre 1995.

[5] H. Borouchaki, P. Laug. *Le mailleur adaptatif bidimensionnel BL2D : manuel d'utilisation et documentation.* Rapport INRIA RT-0185, décembre 1995.

[6] H. Borouchaki, P. Laug. *The BL2D Mesh Generator: Beginner's Guide, User's and Programmer's Manual.* Rapport INRIA RT-0194, juillet 1996.

[7] P.L. GEORGE et P. LAUG, *Normes d'utilisation et de programmation*, Guide Modulef n° 2, INRIA, 1992.

[8] P.L. GEORGE et E. SALTEL, *Post-traitements et graphiques*, Guide Modulef n° 6, INRIA, 1992.

[9] http://www.calcom.ch. CalcoMESH. *Calcom SA, Lausanne, Switzerland.*

[10] E. CATMULL, *A Subdivision Algorithm for Computer Display of Curved Surfaces*, Univ. Utah Comp. Sci. Dept. UTEC-CSC-74-133, 1974.

[11] P. Laug, H. Borouchaki. *BLSURF - Mailleur de surfaces composées de carreaux paramétrés - Manuel d'utilisation.* Rapport INRIA RT-0232, juin 1999.

[12] P. Laug, H. Borouchaki. *BLSURF - Mesh Generator for Composite Parametric Surfaces - User's Manual.* RT-0235, Nov. 1999.

Unité de recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-0803