

Table des matières

1	Discrétisation, algorithmes et exemples	3
1.1	Représentation discrète	3
1.1.1	Maillage triangulaire	4
1.1.2	Champ de croix	4
1.2	Partitionnement de $\partial\Omega_h$	7
1.2.1	Recherche de points singuliers	7
1.2.2	Construction des séparatrices	7
1.2.3	Assemblage des partitions	10
1.3	Opération d'alignement	11
1.4	Etude de la méthode	12
1.4.1	Analyse de convergence	12
1.4.2	Lien entre \bar{u} et \bar{u}_h	12
1.5	Génération de champs de croix	13
	Références	19

Chapitre 1

Discrétisation, algorithmes et exemples

Sommaire

1.1	Représentation discrète	3
1.1.1	Maillage triangulaire	4
1.1.2	Champ de croix	4
1.2	Partitionnement de $\partial\Omega_h$	7
1.2.1	Recherche de points singuliers	7
1.2.2	Construction des séparatrices	7
1.2.3	Assemblage des partitions	10
1.3	Opération d'alignement	11
1.4	Etude de la méthode	12
1.4.1	Analyse de convergence	12
1.4.2	Lien entre \bar{u} et \bar{u}_h	12
1.5	Génération de champs de croix	13

Dans le précédent chapitre, la méthode a été élaborée dans un cadre continu, supposant un accès direct au domaine de calcul et à son champ de croix correspondant. Toutefois, dans la réalité, cette accessibilité directe n'est pas toujours envisageable. Ainsi, pour rendre cette méthode plus praticable, nous abordons dans ce chapitre une approche discrète, adaptant les différents algorithmes pour opérer sur des maillages triangulaires.

Étant donné un tel maillage représentant un domaine donné, le but de ce chapitre est de démontrer qu'à partir d'un champ de croix donné, il est possible de construire sur le maillage triangulaire un maillage quadrilatéral, et d'expliquer dans quelle mesure ce maillage représente fidèlement le domaine initial.

1.1 Représentation discrète

Soit Ω un domaine compact et connexe de \mathbb{R}^2 dont le bord $\partial\Omega$ est lisse par morceau. On désigne par \bar{u} un champ de croix presque- \mathcal{C}^1 défini sur Ω .

1.1.1 Maillage triangulaire

Considérons maintenant une triangulation Ω_h de Ω . Par là, nous entendons que Ω_h est une surface polygonale compacte de \mathbb{R}^2 et on note V l'ensemble des sommets de Ω_h , E l'ensemble des arêtes, et F l'ensemble des triangles fermés non vide composant Ω_h . On suppose que tout les sommets de Ω_h appartiennent à Ω ($V \subset \Omega$) et que tout intersection entre deux triangles est soit vide, soit un sommet, soit une arête. h représente le diamètre maximal des triangles constituant Ω_h . Pour tout $p \in \Omega_h$, on désigne par F_p l'ensemble des triangles de Ω_h contenant p .

1.1.2 Champ de croix

Nous cherchons maintenant à construire une représentation du champ de croix \bar{u} sur le maillage triangulaire Ω_h . Pour se faire, nous commençons par définir la notion d'angle signé entre deux croix.

Définition 1.1. Soient deux croix $\mathbf{c}_1, \mathbf{c}_2$ non nulles. L'angle signé entre \mathbf{c}_1 et \mathbf{c}_2 noté $\delta\theta(\mathbf{c}_1, \mathbf{c}_2)$ est l'unique élément de l'ensemble :

$$\{\delta\theta(\mathbf{c}_1, \mathbf{c}_2)\} := \left\{ \theta_{\mathbf{c}_2} - \theta_{\mathbf{c}_1} + k\frac{\pi}{2}, k \in \mathbb{Z} \right\} \cap \left] -\frac{\pi}{4}, \frac{\pi}{4} \right[.$$

Nous dirons que $\delta\theta(\mathbf{c}_1, \mathbf{c}_2)$ n'est pas défini si $|\theta_{\mathbf{c}_2} - \theta_{\mathbf{c}_1}| = \pi/4$. $\delta\theta(\mathbf{c}_1, \mathbf{c}_2)$ représente la variation d'angle entre \mathbf{c}_1 et \mathbf{c}_2 de \mathbf{c}_1 vers \mathbf{c}_2 . Autrement dit, on a :

$$\theta_{\mathbf{c}_2} = \theta_{\mathbf{c}_1} + \delta\theta(\mathbf{c}_1, \mathbf{c}_2).$$

Dans tout la suite, nous imposons les contraintes suivantes sur le maillage Ω_h :

- pour chaque arête $e \in E$ ayant des sommets s_1 et s_2 , soit $\bar{u}(s_1) \neq 0$ soit $\bar{u}(s_2) \neq 0$. De plus, si à la fois $\bar{u}(s_1) \neq 0$ et $\bar{u}(s_2) \neq 0$, alors $\delta\theta(\bar{u}(s_1), \bar{u}(s_2))$ est défini.
- pour chaque arête $e \in E$ ayant des sommets s_1 et s_2 , soit $\bar{u}(s_1) \neq 0$ soit $\bar{u}(s_2) \neq 0$. De plus, si à la fois $\bar{u}(s_1) \neq 0$ et $\bar{u}(s_2) \neq 0$, alors $\delta\theta(\bar{u}(s_1), \bar{u}(s_2))$ est défini.

Dans la pratique, il sera donc impératif d'affiner ou de modifier localement un maillage qui ne satisfait pas ces contraintes. L'importance de ces contraintes prend tout son sens par la suite, avec l'approche de construction que nous proposons pour représenter \bar{u} sur Ω_h . Nous introduisons maintenant la notion de triangle singulier via la définition suivante :

Définition 1.2. Un triangle T de Ω_h et de sommets s_1, s_2 et s_3 est dit singulier si l'une des assertions suivantes est vérifiée :

- i.) il existe $i \in \llbracket 1, 3 \rrbracket$ tel que $\bar{u}(s_i) = 0$
- ii.) $\sum_{i=1}^3 \delta\theta(\bar{u}(s_i), \bar{u}(s_{i+1})) \neq 0$.

Nous représentons le champ de croix \bar{u} sur Ω_h par un champ de croix noté \bar{u}_h défini pour tout $p \in \Omega_h$ par :

- si $p \in V$, alors $\bar{u}_h(p) = \bar{u}(p)$,

- si $p \in e$, où $e \in E$ est une arête de Ω_h de sommets s_1 et s_2 , alors on a :

$$\begin{cases} \bar{u}_h(p) = \left\{ \mathbf{R} \left(\theta_p + m \frac{\pi}{2} \right) (1, 0)^t, \ m \in \mathbb{Z} \right\}, \\ \theta_p = \theta_{\bar{u}_h}(s_1) + \frac{\|\overrightarrow{s_1 p}\|}{\|\overrightarrow{s_1 s_2}\|} \delta\theta(\bar{u}_h(s_1), \bar{u}_h(s_2)). \end{cases}$$

- sinon, soit T un triangle de Ω_h de sommets s_1, s_2 et s_3 tel que $p \in T$. On distingue trois cas :

— si on a

$$\begin{cases} (\bar{u}_h(s_1), \bar{u}_h(s_2), \bar{u}_h(s_3)) \neq (0, 0, 0), \\ \sum_{i \in \llbracket 1, 3 \rrbracket} \delta\theta(\bar{u}_h(s_i), \bar{u}_h(s_{i+1})) = 0 \quad (\text{avec } s_4 := s_1), \end{cases}$$

alors on pose

$$\begin{cases} \theta_1 = \theta_{\bar{u}_h}(s_1) \\ \theta_2 = \theta_1 + \delta\theta(\bar{u}_h(s_1), \bar{u}_h(s_2)) \\ \theta_3 = \theta_2 + \delta\theta(\bar{u}_h(s_2), \bar{u}_h(s_3)) \end{cases}$$

et on a

$$\begin{cases} \bar{u}_h(p) = \left\{ \mathbf{R} \left(\theta_p + m \frac{\pi}{2} \right) (1, 0)^t, \ m \in \mathbb{Z} \right\}, \\ \theta_p = \sum_{i \in \llbracket 1, 3 \rrbracket} \lambda_i \theta_i. \end{cases}$$

- si $\bar{u}_h(s_1) \neq 0$ et $\bar{u}_h(s_2) \neq 0$ et $\bar{u}_h(s_3) \neq 0$ et $\delta\theta(\bar{u}_h(s_1), \bar{u}_h(s_2)) + \delta\theta(\bar{u}_h(s_2), \bar{u}_h(s_3)) + \delta\theta(\bar{u}_h(s_3), \bar{u}_h(s_1))$
- si $\bar{u}_h(s_1) \neq 0$ et $\bar{u}_h(s_2) \neq 0$ et $\bar{u}_h(s_3) \neq 0$ et $\delta\theta(\bar{u}_h(s_1), \bar{u}_h(s_2)) + \delta\theta(\bar{u}_h(s_2), \bar{u}_h(s_3)) + \delta\theta(\bar{u}_h(s_3), \bar{u}_h(s_1))$

dont la définition est exposée dans les lignes suivantes. On se donne un ensemble de valeurs de \bar{u} défini sur les sommets de Ω_h et on définit pour tout $p \in V$, $\bar{u}_h(p) = \bar{u}(p)$. Pour définir complètement \bar{u}_h , il reste à dire ce que vaut \bar{u}_h pour tout $p \in \Omega_h \setminus$ Nous introduisons alors la nomenclature suivante : Supposons de plus que

Points singuliers, indice et ligne de champs

L'ensemble $\mathcal{S}_{\bar{u}_h}$, défini comme l'ensemble des points singuliers de \bar{u}_h , est constitué des points $p \in \Omega_h$ tels que $\bar{u}_h(p) = 0$.

Lemme 1.1. *Les points singuliers de \bar{u}_h sont isolés.*

Preuve. Soit p un point singulier de \bar{u}_h . Le point p est isolé puisque par construction, on a $F_p \cap \mathcal{S}_{\bar{u}_h} = \{p\}$. En effet, pour tout $q \in F_p \setminus \{p\}$ on a $\bar{u}_h(q) \neq 0$. Par définition, $p \in F_p$. Supposons maintenant qu'il existe un point singulier q tel que $q \in F_p$ et $q \neq p$. Soit \mathcal{T} un triangle de Ω_h contenant q et inclus dans F_p . Puisque $p \in \mathcal{T}$ par définition de F_p , et que g est linéaire sur \mathcal{T} , cela implique que pour tout $r \in [pq]$, $g(r) = (0, 0)$, car $g(p) = g(q) = (0, 0)$. Autrement dit, pour tout $r \in [pq]$, $r \in \mathcal{S}_{\bar{u}_h}$ car $\bar{u}_h(r) = 0$. Par conséquent, les points singuliers de \bar{u}_h ne peuvent pas être isolés, ce qui contredit l'hypothèse selon laquelle les points singuliers de \bar{u}_h sont isolés. On peut donc en conclure que $F_p \cap \mathcal{S}_{\bar{u}_h} = \{p\}$. ■

Examinons à présent l'indice des points singuliers de \bar{u}_h . Soit p un point singulier de \bar{u}_h avec $p \in \Omega_h \setminus \partial\Omega_h$. L'indice de p est donné par :

$$id_{\bar{u}_h}(p) = \int_0^1 d\theta_{\bar{u}_h}^\gamma = \sum_{\mathcal{C} \in F \cap \gamma} \int_0^1 d\theta_{\bar{u}_h}^{\mathcal{C}}.$$

où γ est un chemin fermé englobant p et ne contenant aucun autre point singulier de \bar{u}_h . En pratique, nous calculerons l'indice d'un point p en utilisant une paramétrisation γ du bord ∂F_p de F_p . Soit $S_p = (s_i)_{i \in \llbracket 1, n_{S_p} \rrbracket}$ l'ensemble des sommets des triangles de F_p , privés du point p , et numérotés dans le sens anti-horaire. Il vient alors que :

$$id_{\bar{u}_h}(p) = \frac{1}{2\pi} \sum_{i=1}^{n_{S_p}} (\theta_{\bar{u}_h}^\gamma(s_{i+1}) - \theta_{\bar{u}_h}^\gamma(s_i)) = \frac{1}{2\pi} \sum \Delta, \quad (1.1)$$

où nous avons défini $s_{n_{S_p}+1} := s_1$. Si $p \in \partial\Omega_h$ alors l'indice de p est donné par :

$$id_{\bar{u}_h}(p) = \pi - \hat{p} + \frac{1}{2\pi} \sum_{i=1}^{n_{S_p}} (\theta_{\bar{u}_h}^\gamma(s_{i+1}) - \theta_{\bar{u}_h}^\gamma(s_i)), \quad (1.2)$$

où γ dans ce cas est la paramétrisation de $\partial F_p \setminus \partial\Omega_h$. C'est à dire la partie de ∂F_p se trouvant à l'intérieur de Ω_h .

Proposition 1.1. *Pour tout $p \in \Omega_h$, on a $4id_{\bar{u}_h}(p) \in \llbracket -d, d \rrbracket$ où $d = \#(\partial F_p \cap E)$ si $p \in \Omega_h \setminus \partial\Omega_h$ et $d = \#((\partial F_p \setminus \partial\Omega_h) \cap E)$ si $p \in \partial\Omega_h$.*

Preuve. Si F_p est réduit à un unique triangle, alors on a n ■

Illustrer les singularité se démultipliant en coloriant juste les triangles et en disant qu'on a pas besoin de la localisation spécifique.

Nous abordons à présent la représentation des lignes de champs de \bar{u}_h dans Ω_h . Étant donné un point $p_0 \in \Omega_h$ et un vecteur $\vec{u}_0 \in \mathbb{R}^2$, la ligne de champ $SL(p_0, \vec{u}_0)$ d'origine p_0 sera représenté par

1.2 Partitionnement de $\partial\Omega_h$

L'adaptation de l'algorithme ?? au maillage Ω_h est donné par :

Entrée: Ω_h un maillage triangulaire, champ de croix \bar{u}_h linéaire par morceau sur chaque triangle de Ω_h .

Sortie : Partition de Ω_h en ensembles de régions.

- 1.) Identification des points singuliers du champ de croix,
- 2.) Détermination du nombre de séparatrices pour chaque point singulier,
- 3.) Intégration des séparatrices,
- 4.) Identification des régions.

Algorithm 1: Algorithme de partitionnement Ω_h

On considère que l'algorithme a convergé si les séparatrices ne convergent pas vers un cycle limite. Examinons maintenant en détail certaines étapes de cet algorithme :

1.2.1 Recherche de points singuliers

Étant donné que pour tout $p \in \Omega_h$, $\bar{u}_h(p) = 0$ si et seulement si $g(p) = 0$, rechercher les points singuliers de \bar{u}_h revient à trouver les zéros de la fonction g . Puisque cette dernière est linéaire sur chaque triangle de Ω_h , la recherche de ces points peut être effectuée localement sur chaque triangle comme suit : Soit \mathcal{T} un triangle de Ω_h avec des sommets s_1 , s_2 et s_3 . S'il existe $(\lambda_i)_{i \in \llbracket 1,3 \rrbracket} \subset [0, 1]$ vérifiant le système suivant :

$$\begin{cases} \sum_{i=1}^3 \lambda_i g(s_i) = 0, \\ \sum_{i=1}^3 \lambda_i = 1, \end{cases} \quad (1.3)$$

alors le point $p = \sum_{i=1}^3 \lambda_i s_i$ est un point singulier de \bar{u}_h situé dans le triangle \mathcal{T} .

Remarque 1.1. *Grace à la proposition on peut choisir l'emplacement du point singulier dans F_p . en l'occurrence, pour ne pas avoir à en faire la recherche on peut choisir le barycentre du triangle.*

On peut rassembler plusieurs singularité voisine en une.. permet d'éviter les problématiques de point trop proche par exemple. Illustrer tout ça

1.2.2 Construction des séparatrices

Une fois les points singuliers de \bar{u}_h identifiés, nous procédons à la création des séparatrices sur Ω_h . Cette étape comprend le calcul du nombre de séparatrices à assigner à chaque point singulier, la détermination des directions initiales pour chaque séparatrice, ainsi que l'intégration de ces séparatrices.

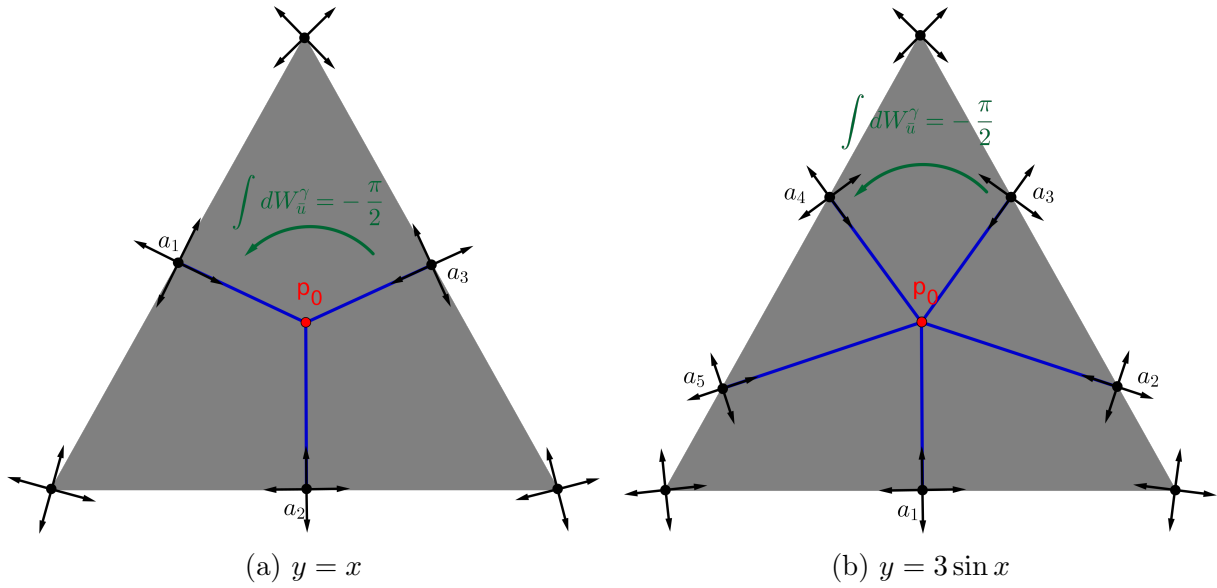


FIGURE 1.1 – Three simple graphs

Nombre de séparatrices : Si p est un point singulier de \bar{u}_h , alors le nombre de séparatrices $N_s(p)$ associées à p est donné par :

$$N_s(p) = \begin{cases} 4 - 4id_{\bar{u}_h}(p) & \text{si } p \in \Omega_h \setminus \partial\Omega_h \\ 2 - 4id_{\bar{u}_h}(p) & \text{si } p \in \partial\Omega_h \end{cases} \quad (1.4)$$

Directions initiales : Soit $p_0 \in \mathcal{S}_{\bar{u}_h} \setminus \partial\Omega_h$. Étant donné que \bar{u}_h s'annule en p_0 , notre première étape consiste à déterminer les orientations initiales des séparatrices. Les directions initiales des séparatrices émanant de p_0 sont données par les vecteurs $\overrightarrow{p_0 a_i}$, $i \in \llbracket 1, N_s(p_0) \rrbracket$ où la suite de points $(a_i)_{i \in \llbracket 1, N_s(p_0) \rrbracket} \subset \partial F_{p_0}$ est construite de la manière suivante :

- le premier point a_1 est tout point de ∂F_{p_0} tel que $\overrightarrow{p_0 a_1} \cdot \|\overrightarrow{p_0 a_1}\|^{-1} \in \bar{u}_h(a_1)$
- soit $t_1 \in [0, 1]$ tel que $\gamma(t_1) = a_1$. Pour tout $i \in \llbracket 2, N_s(p_0) \rrbracket$, on a :

$$a_i = \gamma(t_i) \text{ avec } \int_{t_{i-1}}^{t_i} dW_{p_0}^\gamma = -\frac{\pi}{2},$$

où γ est une paramétrisation de ∂F_{p_0} sur $[0, 1]$ dans le sens anti-horaire et la fonction $W_{p_0}^\gamma$ est donné pour tout $t \in [0, 1]$ par :

$$W_{p_0}^\gamma(t) = \theta_{\bar{u}_h}^\gamma(t) - \arg \overrightarrow{p_0 \gamma(t)}.$$

Si $p_0 \in \mathcal{S}_{\bar{u}_h} \cap \partial\Omega_h$ alors

On veut être mesh tri aussi grossier qu'il est possible donc mailler le vrai domaine et non le mesh tri

Intégration des séparatrices : Considérons $SL_{\bar{u}_h}(p_0, \overrightarrow{p_0 p_1})$ la séparatrice émanant de p_0 dans la direction du vecteur $\overrightarrow{p_0 p_1}$, où $p_1 \in \partial F_{p_0}$. Nous procédons à la construction d'une approximation $SL_{\bar{u}_h}^h(p_0, \overrightarrow{p_0 p_1})$ de $SL_{\bar{u}_h}(p_0, \overrightarrow{p_0 p_1})$ sur Ω_h sous la forme d'une

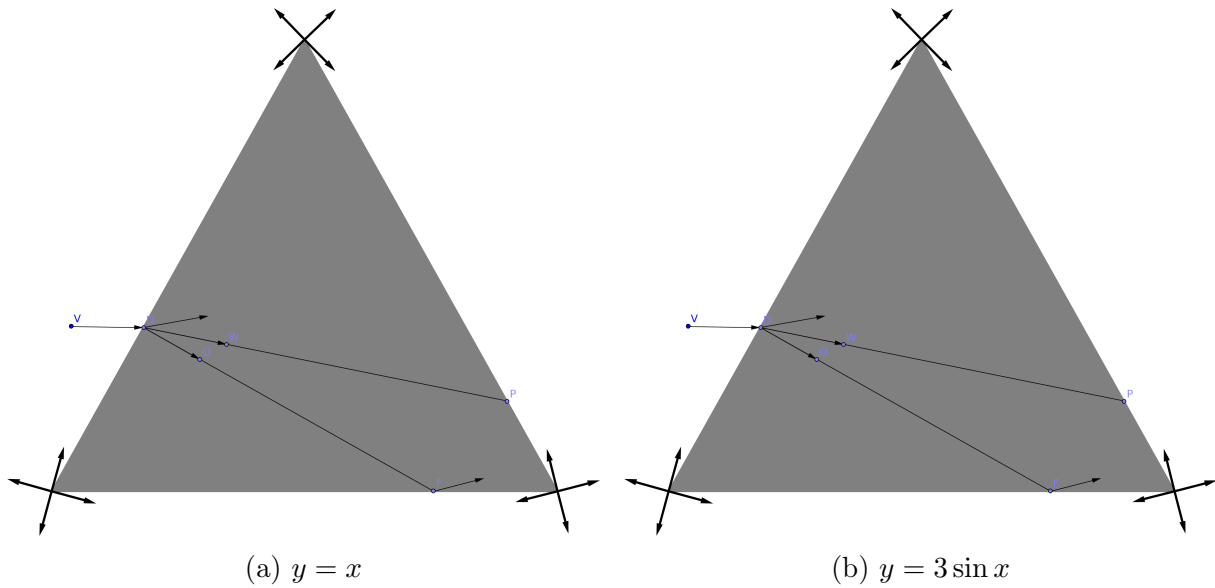


FIGURE 1.2 – Three simple graphs

succession de segments. Le premier segment est représenté par $[p_0p_1]$. Ensuite, pour tout $i \geq 1$, on construit le segment $[p_ip_{i+1}]$ en cherchant le point p_{i+1} comme le point d'intersection entre ∂F_{p_i} et la demi-droite d'origine p_i et dirigée par le vecteur $P(\bar{u}_h(p_i), \overrightarrow{p_{i-1}p_i}) + P(\bar{u}_h(p'_{i+1}), \overrightarrow{p_{i-1}p_i})$. Dans cette formule, p'_{i+1} est le point d'intersection entre ∂F_{p_i} et la demi-droite d'origine p_i et dirigé par le vecteur $P(\bar{u}_h(p_i), d_i)$ et pour tout $(\mathbf{c}, d) \in \mathbf{C} \setminus \{0\} \times \mathbb{R}^2$, $P(\mathbf{c}, d)$ désigne le vecteur \mathbf{c} qui s'aligne le mieux avec la direction d . Autrement dit, il s'agit de l'unique élément de l'ensemble

$$\left\{ c_k \mid c_k = \min_{k \in \llbracket 0, 3 \rrbracket} |c_k \cdot d - 1| \right\} = \underset{c_k \in \mathbf{c}, k \in \llbracket 1, 4 \rrbracket}{\operatorname{argmin}} |c_k \cdot d - 1|.$$

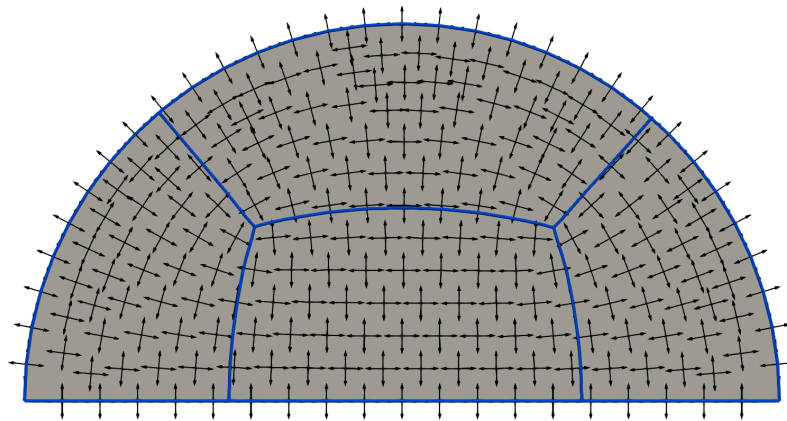
La figure 1.2 illustre la construction du segment $[p_ip_{i+1}]$.

direction de sortie mieux chez moi que chez les autres

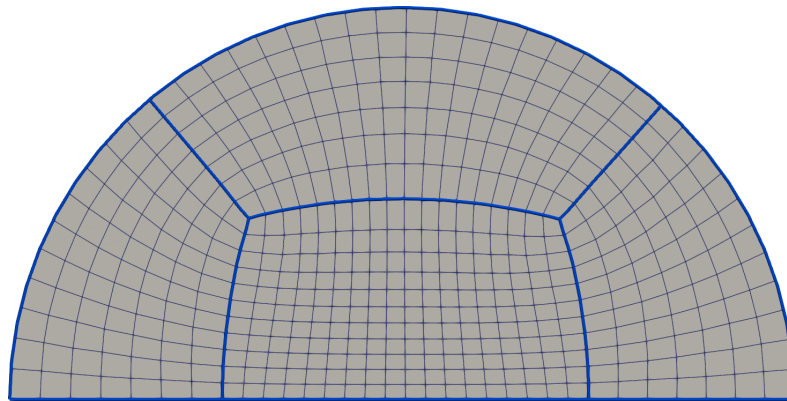
Traversé d'un triangle singulier : lorsque le segment $[p_ip_{i+1}]$ traverse un triangle singulier, l'angle du champ varie beaucoup. On remplace alors le segment en question par une succession d'autres segments calculés en raffinant localement le maillage dans le triangle singulier.

Fusion de séparatrices : De manière similaire à ce qui est réalisé dans [?], les séparatrices du champ de croix sont construites simultanément en incrémentant chacune d'elles progressivement, et la rencontre entre deux séparatrices est anticipée en comparant à chaque incrément, d'une part, la distance entre les derniers points calculés et, d'autre part, les directions des derniers segments construits. En d'autres termes, on cherche à déterminer si, à un moment donné, deux séparatrices données avancent dans des directions opposées et si elles sont suffisamment proches l'une de l'autre. On compare ces deux mesures à des seuils prédéfinis, et en fonction du résultat, on décide de fusionner ou non les deux séparatrices.

La fusion se réalise en créant une nouvelle séparatrice par une fusion linéaire des points des deux séparatrices impliquées. Pour se faire, chaque séparatrice est prolongée à



(a) Alignement du champ de croix présenté sur la figure ?? sur le bord du domaine et partitionnement du domaine.



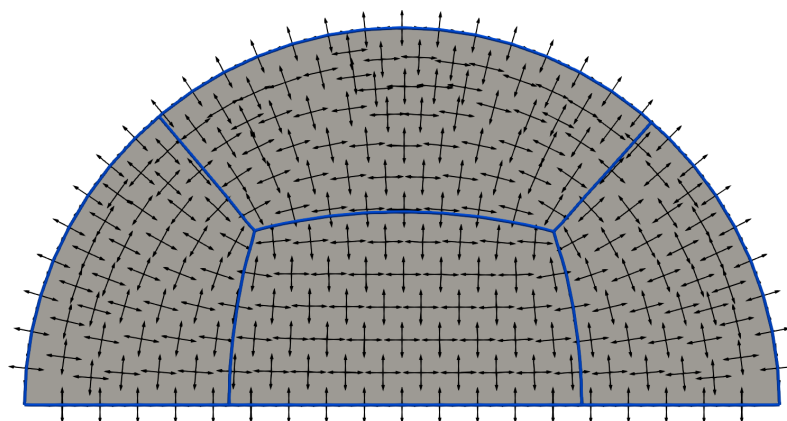
(b) Maillage quadrilatéral du domaine.

FIGURE 1.3 – Direction de sortie différentes.

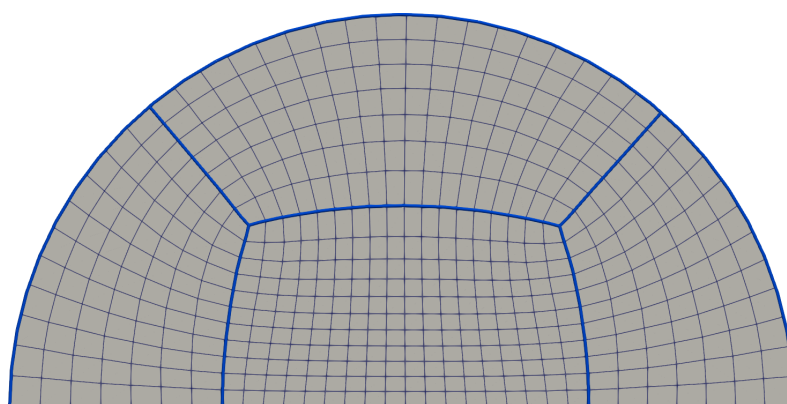
travers Ω_h jusqu'à atteindre la position de départ de l'autre, tout en maintenant le même nombre de points pour chacune des séparatrices. L'intérêt de fusionner les séparatrices réside dans la réduction de leur nombre, ce qui se traduit directement par une diminution du nombre de régions générées lors du découpage du domaine. Nous illustrons la fusion de deux séparatrices sur la figure 1.4. On observe sur que la non fusion donne plus de region qui sont notemment tres etirer et non homogene par rapport aux autres ce qui induit des mesh quad non homogene.

1.2.3 Assemblage des partitions

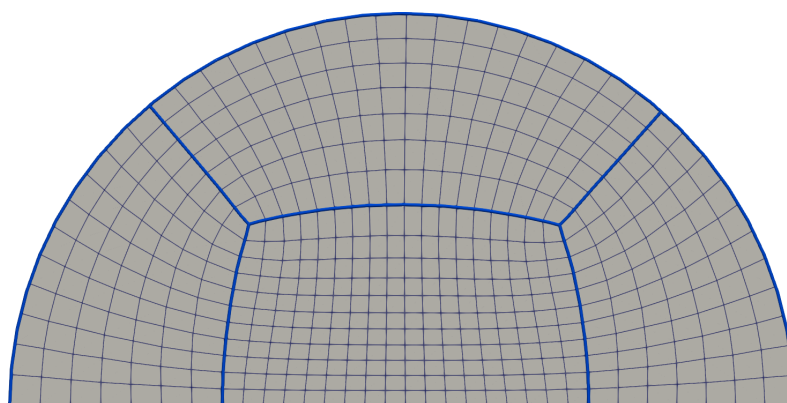
Une fois les séparatrices construites, le maillage triangulaire Ω_h sous-jacent à la méthode est divisé en plusieurs régions. Pour identifier ces régions, nous commençons par modifier Ω_h en un nouveau maillage. On récupère chaque région sous la forme d'un sous-maillage de Ω_h en faisant : Localement dans chaque triangle, on ajoute les points constituant les separatrices a Ω_h modifiant ainsi la topologie du maillage triangulaire initial, Pour se faire, on exécute localement l'algorithme d'insertion de point localement dans chaque triangle.
donné par



(a) Sans fusion



(b) detection fusion.



(c) fusion.

FIGURE 1.4 – Illustration de l'opération d'alignement à partir du champ d'angle donné par l'équation (??).

1.3 Opération d'alignement

cos laplacien rtheta...

Nous abordons maintenant la discrétisation de l'opération d'alignement présenté dans le chapitre ?? . Étant donné une représentation \bar{N}_h du champ de croix \bar{N} sur le bord $\partial\Omega_h$ de Ω_h , nous cherchons à modifier \bar{u}_h en construisant un nouveau champ de croix \bar{v}_h qui soit aligné avec $\partial\Omega_h$. Autrement dit, on veut que pour tout $p \in \partial\Omega_h$, $\bar{v}_h(p) \in \{\bar{N}_h(p), 0\}$. Pour se faire, on commence par définir l'ensemble des points singuliers de bord du champ de croix \bar{v} que l'on note \mathcal{B} . On associe un paramètre I_p pour tout $p \in \mathcal{B}$ représentant l'indice que nous souhaitons qu'il possède dans le champ de croix \bar{v} et vérifiant :

$$I_p = \begin{cases} \frac{k}{4} \text{ avec } k \in \mathbb{Z} \text{ et } k \leq 1 & \text{si } p \in \mathcal{B}, \\ 0 & \text{sinon .} \end{cases} \quad (1.5)$$

Le champ de croix \bar{u}_h choisit comme champ initial doit alors vérifié $0 < \#\mathcal{S}_{\bar{u}_h} < \infty$ et pour tout point $p \in \mathcal{S}_{\bar{u}_h}$, $id_{\bar{u}_h}(p) = k/4$, avec $k \in \mathbb{Z}$ et $k \leq 1$. On suppose de plus que :

$$\theta_{\bar{u}_h} - \theta_{\bar{u}_h} = \chi(\Omega) - \sum_{p \in \mathcal{B}} I(p). \quad (1.6)$$

Le champ de croix \bar{v}_h est alors donné par :

$$\bar{v}_h(p) = \begin{cases} \mathbf{R}(\phi_h(p))\bar{u}_h(p) & \text{si } p \in \Omega_h \setminus (\mathcal{B} \cup \mathcal{S}_{\bar{u}_h}), \\ \bar{N}_h(p) & \text{si } p \in (\mathcal{S}_{\bar{u}_h} \cap \partial\Omega_h) \setminus \mathcal{B}, \\ 0 & \text{si } p \in \mathcal{B}. \end{cases} \quad (1.7)$$

où ϕ_h est une approximation de la fonction ϕ définie par l'équation de Laplace suivant :

$$\begin{cases} \Delta\phi & = 0 & \text{dans } \Omega_h, \\ \phi_h(\gamma(t)) & = \theta_{\bar{N}_h}^\gamma(t) + \mathcal{I}(t) - \theta_{\bar{u}_h}(\gamma(t)) & \text{sur } \gamma^{-1}(\partial\Omega_h \setminus (\mathcal{B} \cup \mathcal{S}_{\bar{u}_h})), \end{cases} \quad (1.8)$$

où γ est une paramétrisation sur $[0, 1]$ de $\partial\Omega_h$ et la fonction \mathcal{I} est donnée par :

$$\mathcal{I}(t) = \sum_{s \in \gamma^{-1}(\mathcal{B})} \left[\left(\pi - \widehat{\gamma(s)} - 2\pi I_{\gamma(s)} \right) - \left(\lim_{r \rightarrow s^+} \theta_{\bar{N}_h}^\gamma(r) - \lim_{r \rightarrow s^-} \theta_{\bar{N}_h}^\gamma(r) \right) \right] \mathbb{1}_{[0,t]}(s),$$

avec $\widehat{\gamma(s)}$ la mesure de l'ouverture angulaire de la frontière en $\gamma(s)$.

ne pas oublié thetah pour le non-simplement connexe

1.4 Etude de la méthode

1.4.1 Analyse de convergence

1.4.2 Lien entre \bar{u} et \bar{u}_h

Lemme 1.2. $\mathcal{S}_{\bar{u}} = \mathcal{S}_{\bar{u}_h}$

Lorsque h tend vers 0, u_h tend vers u

champ de croix u isolé donc u_h isolé

vers quoi tend Q_h ?

On se rend compte que les normales ne match pas.

Je me suis aligné avec les normales d'une autre géométrie. Laquelle ?

On a eu Q_h .

Lorsque h tend vers 0

Le partitionnement Q_h obtenu suite au partitionnement est 'il une decomposition en région de quatres côtés de Ω_h ?

En quel sens le partitionnement Q_h obtenu est

1.5 Génération de champs de croix

edp
z-ai
somme flux triangle

Construction d'un maillage quadrilatéral

De manière similaire à ce qui a été exposé dans le chapitre précédent, nous entamons d'abord la mise en place du processus de partitionnement de Ω_h , puis nous revenons sur la discrétisation des différentes méthodes permettant d'aligner un champ de croix donné sur le bord du domaine sur lequel il est défini. Pour finir, nous abordons le maillage en quadrangles des régions à quatre côtés résultant du partitionnement de Ω_h .

Traitement des bords

Obtention du maillage quadrilatéral

Description de l'interpolation transfini.

Interprétation du partitionnement

En vrai v_h pas aligné avec ω_h donc pas de raison que ça marche
 v_h tend vers v S_h tend vers S R_h tend vers R chap 2 marche par v S et R donc partitionnement a 4 cotés Donc Q_h est un maillage de ω_h

Notons que le maillage Ω_h est construit de sorte que \mathbf{B} soit inclut dans l'ensemble des sommets de $\partial\Omega_h$.

Jusqu'à présent, nous n'avons pas supposé que nous travaillions avec une discrétisation particulière. Dans ce chapitre, nous discrétisons la méthode sur des maillages triangulaires.

Dans le chapitre précédent, les composantes de premier plan mise en jeu sont : le domaine bornée Ω , l'ensemble \mathbf{B} des points de $\partial\Omega$ caractérisant la géométrie de Ω et le champ de croix initial \bar{u} vérifiant l'hypothèse \mathbf{H}_1 . Avec ces données, nous construisons un champ de croix \bar{v} aligné avec le bord de Ω issue des différentes opérations abordés dans le chapitre précédent. \bar{v} est ensuite utilisé pour partitionner le domaine Ω en régions de quatre côtés.

Commençons par exécuter le processus d'alignement sur le champ de croix \bar{u}_h puisque le champ de croix u dont il est l'approximation n'a aucune raison d'être aligné par rapport à Ω . Pour se faire, nous cherchons un champ d'angle noté ϕ_h et vérifiant l'équation suivante :

$$\begin{cases} \Delta\phi_h &= 0 \text{ dans } \Omega_h, \\ \phi_h &= \theta_{\bar{N}_h} - \theta_{\bar{u}_h} \text{ sur } \partial\Omega_h. \end{cases} \quad (1.9)$$

où \bar{N}_h correspond au champ de croix de la normale extérieur de $\partial\Omega_h$. Remarquons que \bar{N}_h est défini sur les arêtes de $\partial\Omega_h$ mais n'est pas défini sur les sommets de $\partial\Omega_h$. Autrement dit, le champ de croix \bar{N}_h n'est pas compatible avec l'ensemble \mathbf{B} puisque les points singuliers de \bar{N}_h ne coïncident pas avec les points de l'ensemble \mathbf{B} .

Pour palier à ce problème, nous modifions la définition du champ de croix normal. Supposons que l'on puisse définir un champ de croix \tilde{N}_h sur $\partial\Omega_h$ tel que les points singuliers de \tilde{N}_h correspondent exactement aux points de l'ensemble \mathbf{B} . L'équation (1.9) devient alors :

$$\begin{cases} \Delta\phi_h &= 0 \text{ dans } \Omega_h, \\ \phi_h &= \theta_{\tilde{N}_h} - \theta_{\bar{u}_h} \text{ sur } \partial\Omega_h. \end{cases} \quad (1.10)$$

La résolution de l'équation (1.10) permet de construire le champ de croix \bar{v}_h sur Ω_h définit par :

$$\bar{v}_h : p \in \Omega_h \longrightarrow \bar{v}_h(p) = \mathbf{R}(\phi_h(p))\bar{u}_h(p).$$

On ne peut interpoler directement le champ de croix, on passe par un champ de vecteur intermédiaire appelé champ de représentation ??,

Cependant, il est clair que \bar{v}_h n'est pas aligné avec $\partial\Omega_h$ puisque $\bar{v}_h = \tilde{N}_h$ sur $\partial\Omega_h$. Autrement dit, le champ de croix \bar{v}_h est aligné avec un domaine que nous notons $\tilde{\Omega}_h$ et dont la normale extérieure est associée au champ de croix \tilde{N}_h . \bar{v}_h n'étant pas défini sur $\tilde{\Omega}_h$ nous définissons une opération appelé *lift* et noté \mathbf{L} permettant de transporter une fonction définie sur un domaine vers un autre domaine. Nous désignons alors par $\tilde{v}_h = \mathbf{L}_{\tilde{\Omega}_h}^{\Omega_h} \bar{v}_h$ le lift de \bar{v}_h de Ω_h vers $\tilde{\Omega}_h$. Le champ de croix \tilde{v}_h ainsi défini est aligné avec $\partial\tilde{\Omega}_h$ (bord de $\tilde{\Omega}_h$) ce qui nous permet d'appliquer l'algorithme de partitionnement défini dans le chapitre ?? à \tilde{v}_h sur $\tilde{\Omega}_h$.

Définition de \tilde{N}_h et $\tilde{\Omega}_h$

Définition du Lift

Discussion sur Q_h

utiliser dans l'équation (1.9) en définissant le champ de croix \tilde{N} de la manière suivante :

$$\forall p \in \partial\Omega_h, \quad \tilde{N}(p) = \left\{ \begin{array}{l} \end{array} \right.$$

L'équation (1.9) devient alors :

Suite à la résolution de l'équation (1.10) sur Ω_h , on peut construire le champ de croix $\bar{v}_h = \mathbf{R}(\phi_h)\bar{u}_h$ sur Ω_h . On remarque alors que le champ de croix \bar{v}_h ainsi construit n'est pas aligné avec Ω_h .

Définition de $\tilde{\Omega}$

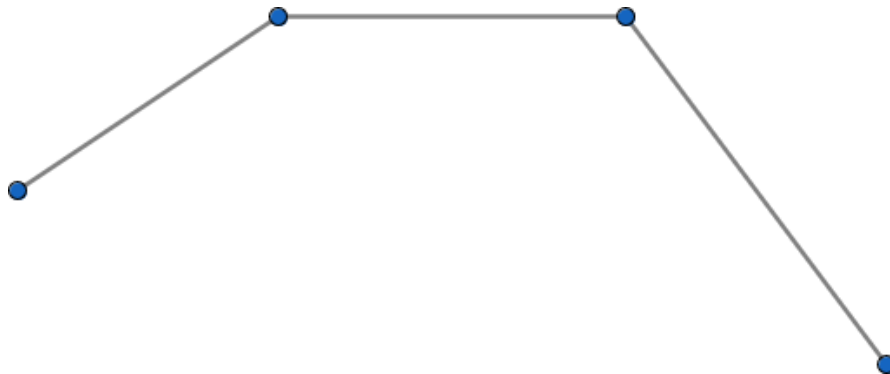


FIGURE 1.5

Définition du lift \mathbf{L}

Définition de $\tilde{v} = \mathbf{L}_{\Omega_h}^{\tilde{\Omega}} v_h$

Comparaison de v et \tilde{v}

$$\begin{aligned}
\|v - \mathbf{L}_{\Omega}^{\tilde{\Omega}} \tilde{v}\| &\leq \|\mathbf{R}(\phi)u - \mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h\| + \|\mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h - \mathbf{L}_{\Omega}^{\tilde{\Omega}} \tilde{v}\| \\
&\leq \|\mathbf{R}(\phi)u - \mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h\| + \|\mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h - \mathbf{R}(\mathbf{L}_{\Omega_h}^{\Omega} \phi_h)\mathbf{L}_{\Omega_h}^{\Omega} u_h\| + \\
&\quad \|\mathbf{R}(\mathbf{L}_{\Omega_h}^{\Omega} \phi_h)\mathbf{L}_{\Omega_h}^{\Omega} u_h - \mathbf{L}_{\Omega}^{\tilde{\Omega}} \mathbf{L}_{\Omega_h}^{\tilde{\Omega}} v_h\| \\
&\leq \|\mathbf{R}(\phi)u - \mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h\| + \|\mathbf{R}(\phi)\mathbf{L}_{\Omega_h}^{\Omega} u_h - \mathbf{R}(\mathbf{L}_{\Omega_h}^{\Omega} \phi_h)\mathbf{L}_{\Omega_h}^{\Omega} u_h\| + \\
&\quad \|\mathbf{R}(\mathbf{L}_{\Omega_h}^{\Omega} \phi_h)\mathbf{L}_{\Omega_h}^{\Omega} u_h - \mathbf{L}_{\Omega}^{\tilde{\Omega}} \mathbf{L}_{\Omega_h}^{\tilde{\Omega}} \mathbf{R}(\phi_h)u_h\|
\end{aligned}$$

Algorithmes

remarque sur le tracé des "streamlines" dans un triangle singulier en comparaison avec ce que fait viertel sachant que le champ n'est pas linéaire dans un tel triangle

Exemples

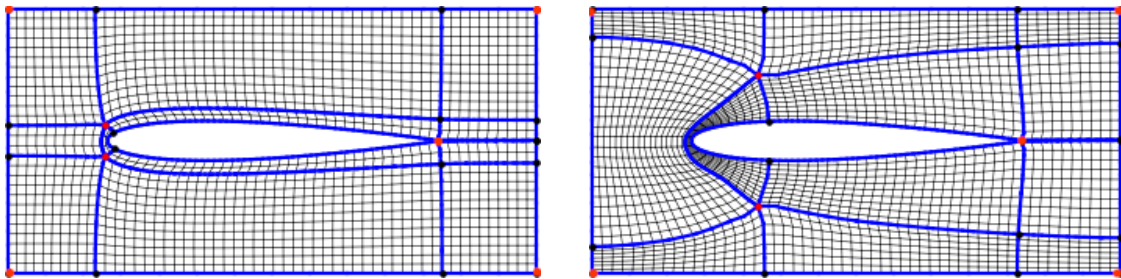


FIGURE 1.6 – Mesh of Naca0012 with two different configurations of singular points. The initial cross-fields were obtained using formula (??) .

Structure de donnée
représentation des multimatériau

structure de donnée mesh

generation champ de croix
discretisation par vertex
alignement champ de croix

index
seul type 1 0 -1

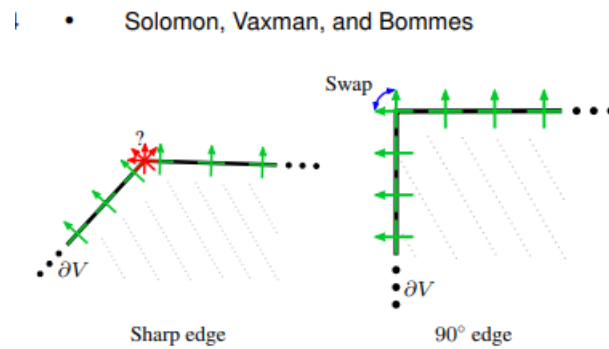


FIGURE 1.7 – Caption

N'importe quel champ \rightarrow mesh quad
 préserver les points singuliers
 Montrer simplement $\phi = N \cdot U$ et sa trace
 déplacer ensuite les singularité vers où on veut

Complementary Operations Irregular Vertex Cancellation : We can move a v_3 vertex to collide with a v_5 vertex, or vice versa, by applying multiple pair-wise movement operations. When one v_3 and one v_5 vertex collide they cancel each other and both become regular. At least one other irregular vertex needs to be involved in this cancellation. In this fashion we develop a $3-5$ pair cancellation operation. It is possible that the last step of a $3-5$ pair cancellation is equivalent to one $3-3-5-5$ removal operation and two pairs of irregular vertices are canceled at once. Examples can be found in Figures 16 and 17. Irregular Vertex Merging : A $3-3$ pair can be merged to a v_2 vertex and a $5-5$ pair can be merged to a v_6 vertex when their graph distance is even. Theorems 7.2 and 7.3 provide the theoretical analysis that is related to such a merge. Irregular Vertex Alignment : Under the assumptions of Theorems 7.2 and 7.3, arbitrary $3-3$ and $5-5$ pairs can be aligned by applying multiple movement operations until $d_1 = 0$ or $d_2 = 0$. Smoothing : We use iterative Laplacian mesh smoothing to improve the geometry if the connectivity edits degrade the shape of the mesh above a user-defined tolerance. The user can select uniform weights or cord-length weights, and elect to preserve sharp features by constraining the positions of vertices on sharp edges. The smoothing scheme can improve the aspect ratios of modified faces. After each iteration all vertices are projected back onto the original mesh. We have also experimented with a scheme in which newly generated vertices are pulled towards vertices in the original mesh if the distance between the new and original vertices is above a threshold. The projection and pulling scheme can narrow the difference to the original mesh.

9 Connectivity Editing for Quadrilateral Meshes
 Chi-Han Peng Arizona State University Eugene Zhang† Oregon State University Yoshihiro Kobayashi‡ Arizona State University Peter Wonka§ Arizona State University / KAUST

Références

- [Allaire, 2005] Allaire, G. (2005). Analyse numérique et optimisation : Une introduction à la modélisation mathématique et à la simulation numérique. *Editions Ecole Polytechnique*, 22.
- [Alliez et al., 2003] Alliez, P., Cohen-Steiner, D., Devillers, O., Lévy, B., and Desbrun, M. (2003). Anisotropic polygonal remeshing. In *ACM SIGGRAPH 2003 Papers*, pages 485–493.
- [Arnold et al., 2000] Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D. (2000). Discontinuous galerkin methods for elliptic problems. In *Discontinuous Galerkin Methods : Theory, Computation and Applications*, pages 89–101. Springer.
- [Baker, 2005] Baker, T. J. (2005). Mesh generation : Art or science ? *Progress in Aerospace Sciences*, 41(1) :29–63.
- [Barth and Ohlberger, 2003] Barth, T. and Ohlberger, M. (2003). Finite volume methods : foundation and analysis.
- [Beaufort et al., 2017] Beaufort, P.-A., Lambrechts, J., Henrotte, F., Geuzaine, C., and Remacle, J.-F. (2017). Computing cross fields a pde approach based on the ginzburg-landau theory. *Procedia engineering*, 203 :219–231.
- [Blacker and Stephenson, 1991] Blacker, T. D. and Stephenson, M. B. (1991). Paving : A new approach to automated quadrilateral mesh generation. *International journal for numerical methods in engineering*, 32(4) :811–847.
- [Bommes et al., 2013] Bommes, D., Lévy, B., Pietroni, N., Puppo, E., Silva, C., Tarini, M., and Zorin, D. (2013). Quad-mesh generation and processing : A survey. In *Computer graphics forum*, volume 32, pages 51–76. Wiley Online Library.
- [Bommes et al., 2009] Bommes, D., Zimmer, H., and Kobbelt, L. (2009). Mixed-integer quadrangulation. *ACM Transactions On Graphics (TOG)*, 28(3) :1–10.
- [Botella, 2016] Botella, A. (2016). *Génération de maillages non structurés volumiques de modèles géologiques pour la simulation de phénomènes physiques*. PhD thesis, Université de Lorraine.
- [Bowyer, 1981] Bowyer, A. (1981). Computing dirichlet tessellations. *The computer journal*, 24(2) :162–166.
- [Bunin, 2008] Bunin, G. (2008). Towards unstructured mesh generation using the inverse poisson problem. *arXiv preprint arXiv :0802.2399*.
- [Campen et al., 2015] Campen, M., Bommes, D., and Kobbelt, L. (2015). Quantized global parametrization. *Acm Transactions On Graphics (tog)*, 34(6) :1–12.
- [Catmull and Clark, 1998] Catmull, E. and Clark, J. (1998). Recursively generated b-spline surfaces on arbitrary topological meshes. In *Seminal graphics : pioneering efforts that shaped the field*, pages 183–188.

- [Cohen-Steiner et al., 2002] Cohen-Steiner, D., De Verdiere, E. C., and Yvinec, M. (2002). Conforming delaunay triangulations in 3d. In *Proceedings of the eighteenth annual symposium on Computational geometry*, pages 199–208.
- [D’azevedo, 2000] D’azevedo, E. F. (2000). Are bilinear quadrilaterals better than linear triangles? *SIAM Journal on Scientific Computing*, 22(1) :198–217.
- [de Moura Pinto and Freitas, 2009] de Moura Pinto, F. and Freitas, C. M. D. S. (2009). Fast medial axis transform for planar domains with general boundaries. In *2009 XXII Brazilian Symposium on Computer Graphics and Image Processing*, pages 96–103. IEEE.
- [Ern and Guermond, 2004] Ern, A. and Guermond, J.-L. (2004). *Theory and practice of finite elements*, volume 159. Springer.
- [Fogg et al., 2015] Fogg, H. J., Armstrong, C. G., and Robinson, T. T. (2015). Automatic generation of multiblock decompositions of surfaces. *International Journal for Numerical Methods in Engineering*, 101(13) :965–991.
- [Forsythe et al., 1961] Forsythe, G. E., Wasow, W. R., and Nachbar, W. (1961). Finite-difference methods for partial differential equations.
- [George and Borouchaki, 1997] George, P. L. and Borouchaki, H. (1997). *Triangulation de Delaunay et maillage : applications aux éléments finis*. Hermes.
- [George and Frey, 2013] George, P. L. and Frey, P. (2013). *Mesh generation : application to finite elements*. John Wiley & Sons.
- [George and Seveno, 1994] George, P. L. and Seveno, É. (1994). The advancing-front mesh generation method revisited. *International Journal for Numerical Methods in Engineering*, 37(21) :3605–3619.
- [Gurung et al., 2011] Gurung, T., Laney, D., Lindstrom, P., and Rossignac, J. (2011). Squad : Compact representation for triangle meshes. In *Computer Graphics Forum*, volume 30, pages 355–364. Wiley Online Library.
- [Hecht, 2007] Hecht, F. (2007). Maillage 2d, 3d, adaptation.
- [Höhring et al., 1997] Höhring, R. H., Müller-Hannemann, M., and Wiehe, K. (1997). Mesh refinement via bidirected flows : Modeling, complexity, and computational results. *Journal of the ACM (JACM)*, 44(3) :395–426.
- [Ives, 1995] Ives, D. (1995). Geometric grid generation. surface modeling, grid generation, and related issues in computational fluid dynamic (cfd) solutions. In *Proceedings NASA-conference, Cleveland, Ohio, NASA CP-3291*.
- [Jezdimirovic et al., 2021] Jezdimirovic, J., Chemin, A., Reberol, M., Henrotte, F., and Remacle, J.-F. (2021). Quad layouts with high valence singularities for flexible quad meshing. *Proceedings of the 29th Meshing Roundtable*.
- [Kälberer et al., 2007] Kälberer, F., Nieser, M., and Polthier, K. (2007). Quadcover-surface parameterization using branched coverings. In *Computer graphics forum*, volume 26, pages 375–384. Wiley Online Library.
- [Kirkup, 2019] Kirkup, S. (2019). The boundary element method in acoustics : A survey. *Applied Sciences*, 9(8) :1642.
- [Kowalski et al., 2013] Kowalski, N., Ledoux, F., and Frey, P. (2013). A pde based approach to multidomain partitioning and quadrilateral meshing. In *Proceedings of the 21st international meshing roundtable*, pages 137–154. Springer.

- [LeVeque, 2002] LeVeque, R. J. (2002). *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press.
- [Liu et al., 2006] Liu, Y., Vinokur, M., and Wang, Z. J. (2006). Spectral difference method for unstructured grids i : Basic formulation. *Journal of Computational Physics*, 216(2) :780–801.
- [Lo, 1989] Lo, S. (1989). Generating quadrilateral elements on plane and over curved surfaces. *Computers & structures*, 31(3) :421–426.
- [Löhner, 1996] Löhner, R. (1996). Progress in grid generation via the advancing front technique. *Engineering with computers*, 12 :186–210.
- [Löhner, 2014] Löhner, R. (2014). Recent advances in parallel advancing front grid generation. *Archives of Computational Methods in Engineering*, 21 :127–140.
- [Macq et al., 2020] Macq, A., Reberol, M., Henrotte, F., Beaufort, P.-A., Chemin, A., Remacle, J.-F., and Van Schaftingen, J. (2020). Ginzburg-landau energy and placement of singularities in generated cross fields. *arXiv preprint arXiv :2010.16381*.
- [Marinov and Kobbelt, 2004] Marinov, M. and Kobbelt, L. (2004). Direct anisotropic quad-dominant remeshing. In *12th Pacific Conference on Computer Graphics and Applications, 2004. PG 2004. Proceedings.*, pages 207–216. IEEE.
- [Mitchell, 2000] Mitchell, S. A. (2000). High fidelity interval assignment. *International Journal of Computational Geometry & Applications*, 10(04) :399–415.
- [Mitchell, 2014] Mitchell, S. A. (2014). Simple and fast interval assignment using nonlinear and piecewise linear objectives. In *Proceedings of the 22nd International Meshing Roundtable*, pages 203–221. Springer.
- [Myles et al., 2014] Myles, A., Pietroni, N., and Zorin, D. (2014). Robust field-aligned global parametrization. *ACM Trans. Graph.*, 33(4) :135–1.
- [Nackman and Srinivasan, 1989] Nackman, L. R. and Srinivasan, V. (1989). Method of generating finite elements using the symmetric axis transform. US Patent 4,797,842.
- [Nédélec, 2001] Nédélec, J.-C. (2001). *Acoustic and electromagnetic equations : integral representations for harmonic problems*, volume 144. Springer.
- [Nieser et al., 2011] Nieser, M., Reitebuch, U., and Polthier, K. (2011). Cubecover—parameterization of 3d volumes. In *Computer graphics forum*, volume 30, pages 1397–1406. Wiley Online Library.
- [Norrie and De Vries, 2014] Norrie, D. H. and De Vries, G. (2014). *The finite element method : fundamentals and applications*. Academic Press.
- [Owen, 1998] Owen, S. J. (1998). A survey of unstructured mesh generation technology. *IMR*, 239(267) :15.
- [Owen et al., 1998] Owen, S. J., Staten, M. L., Canann, S. A., and Saigal, S. (1998). Quad-morphing : advancing front quad meshing using triangle transformations. *Proceedings 7th International Meshing Roundtable, Dearborn, MI*.
- [Owen et al., 1999] Owen, S. J., Staten, M. L., Canann, S. A., and Saigal, S. (1999). Q-morph : an indirect approach to advancing front quad meshing. *International journal for numerical methods in engineering*, 44(9) :1317–1340.
- [Palacios and Zhang, 2007] Palacios, J. and Zhang, E. (2007). Rotational symmetry field design on surfaces. *ACM Transactions on Graphics (TOG)*, 26(3) :55–es.

- [Ray et al., 2006] Ray, N., Li, W. C., Lévy, B., Sheffer, A., and Alliez, P. (2006). Periodic global parameterization. *ACM Transactions on Graphics (TOG)*, 25(4) :1460–1485.
- [Ray et al., 2008] Ray, N., Vallet, B., Li, W. C., and Lévy, B. (2008). N-symmetry direction field design. *ACM Transactions on Graphics (TOG)*, 27(2) :1–13.
- [Reberol, 2018] Reberol, M. (2018). *Maillages hex-dominants : génération, simulation et évaluation*. PhD thesis, Phd Thesis). Université de Lorraine.
- [Remacle et al., 2012] Remacle, J.-F., Lambrechts, J., Seny, B., Marchandise, E., Johnen, A., and Geuzainet, C. (2012). Blossom-quad : A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm. *International journal for numerical methods in engineering*, 89(9) :1102–1119.
- [Rivière, 2008] Rivière, B. (2008). *Discontinuous Galerkin methods for solving elliptic and parabolic equations : theory and implementation*. SIAM.
- [Schneiders, 1996] Schneiders, R. (1996). A grid-based algorithm for the generation of hexahedral element meshes. *Engineering with computers*, 12 :168–177.
- [Shepherd and Johnson, 2008] Shepherd, J. F. and Johnson, C. R. (2008). Hexahedral mesh generation constraints. *Engineering with Computers*, 24(3) :195–213.
- [Smith, 1985] Smith, G. D. (1985). *Numerical solution of partial differential equations : finite difference methods*. Oxford university press.
- [Stummel, 2006] Stummel, F. (2006). Difference methods for linear initial value problems. In *Numerische Behandlung nichtlinearer Integrodifferential-und Differentialgleichungen : Vorträge einer Tagung im Mathematischen Forschungsinstitut Oberwolfach, 2. 12.–7. 12. 1973*, pages 123–135. Springer.
- [Taghavi, 1994] Taghavi, R. (1994). Automatic parallel and fault tolerant mesh generation from cad on cray research super computers. In *Proc. CUG Conference, Tours, France*.
- [Tam, 1991] Tam, T. (1991). 2d finite element mesh generation by medial axis subdivision. *Advances in engineering software and workstations*, 13(5-6) :313–324.
- [Van den Abeele et al., 2008] Van den Abeele, K., Lacor, C., and Wang, Z. J. (2008). On the stability and accuracy of the spectral difference method. *Journal of Scientific Computing*, 37 :162–188.
- [Viertel and Oosting, 2019] Viertel, R. and Oosting, B. (2019). An approach to quad meshing based on harmonic cross-valued maps and the ginzburg–landau theory. *SIAM Journal on Scientific Computing*, 41(1) :A452–A479.
- [Watson, 1981] Watson, D. F. (1981). Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *The computer journal*, 24(2) :167–172.
- [White and Kinney, 1997] White, D. R. and Kinney, P. (1997). Redesign of the paving algorithm : Robustness enhancements through element by element meshing. In *6th international meshing roundtable*, volume 10, page 830.
- [Wu, 2002] Wu, T. (2002). Boundary element acoustics fundamentals and computer codes.
- [Yerry and Shephard, 1983] Yerry, M. and Shephard, M. (1983). Finite element mesh generation based on a modified-quadtrees approach. *IEEE Computer Graphics and Applications*, 3(1) :36–46.
- [Zienkiewicz and Taylor, 2005] Zienkiewicz, O. C. and Taylor, R. L. (2005). *The finite element method for solid and structural mechanics*. Elsevier.