Proposition de stage
Elliptic mesh smoothing

Dmitry Sokolov et Nicolas Ray

- **Titre :** Elliptic mesh smoothing
- **Thématique :** Géométrie numérique, remaillage hexahedral
- **Laboratoire, institution et université :** LORIA, Inria, Université de Lorraine
- **Ville et pays :** Nancy, France.
- **Equipe ou projet dans le labo :** Equipe PIXEL (`http://pixel.inria.fr/`)
- **Directeur de stage :** Dmitry Sokolov (dmitry.sokolov@univ-lorraine.fr) et Nicolas Ray (ray@loria.fr)
- **Directeur du laboratoire :** Jean-Yves Marion (jean-yves.marion@loria.fr)

- **Présentation générale du domaine (5 à 10 lignes) :**
En informatique, on représente souvent les objets géométriques par des maillages. Dans le cadre de ce stage, nous nous intéressons à la déformation de maillages. La figure 1 nous montre un exemple : étant donné un maillage initial, nous déformons son bord et nous voulons calculer la position des nœuds à l'intérieur du domaine pour que le maillage déformé soit le moins distordu possible.

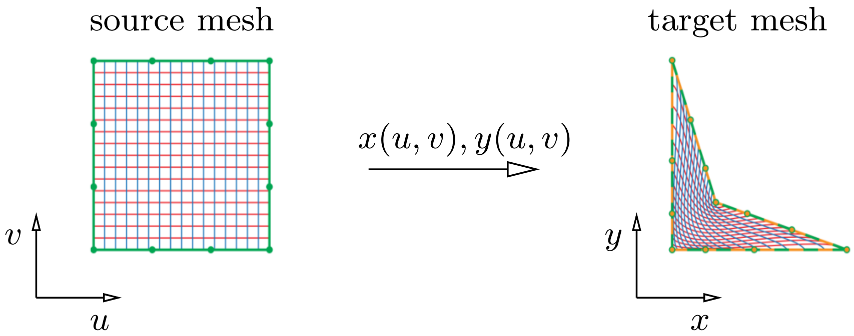

FIGURE 1 – Étant donné un maillage, nous déformons son bord et nous voulons que l'intérieur du nouveau maillage soit le moins distordu possible.

L'approche classique (*Laplacian smoothing*) est de minimiser une énergie $\min_x \left\{ \int_\Omega |\nabla x|^2 dV \right\}$ en fixant le bord du domaine. Cela équivaut à chercher la solution de

$$\begin{cases} \Delta x(u,v) = 0 \\ \Delta y(u,v) = 0 \end{cases} \tag{1}$$

Cette approche est très facile à implementer, cependant elle n'est pas sans défauts : elle peut produire des mailles retournées (Figure 2, colonne du milieu). Une autre approche (*Winslow smoothing*) serait d'annuler le laplacien de la transformation inverse [1] :

$$\begin{cases} \Delta u(x,y) = 0 \\ \Delta v(x,y) = 0 \end{cases} \tag{2}$$

Après calcul explicite des termes, on se retrouve à résoudre (en dimension 2) :

$$\begin{cases} \alpha u_{xx} - 2\beta u_{xy} + \gamma u_{yy} = 0 \\ \alpha v_{xx} - 2\beta v_{xy} + \gamma v_{yy} = 0 \end{cases} \tag{3}$$

avec :

$$\begin{cases} \alpha = u_y^2 + v_y^2 \\ \beta = u_y u_y + v_y v_x \\ \gamma = u_x^2 + v_x^2 \end{cases}$$

Cette approche nous donne de bien meilleurs résultats (Figure 2, colonne de droite).

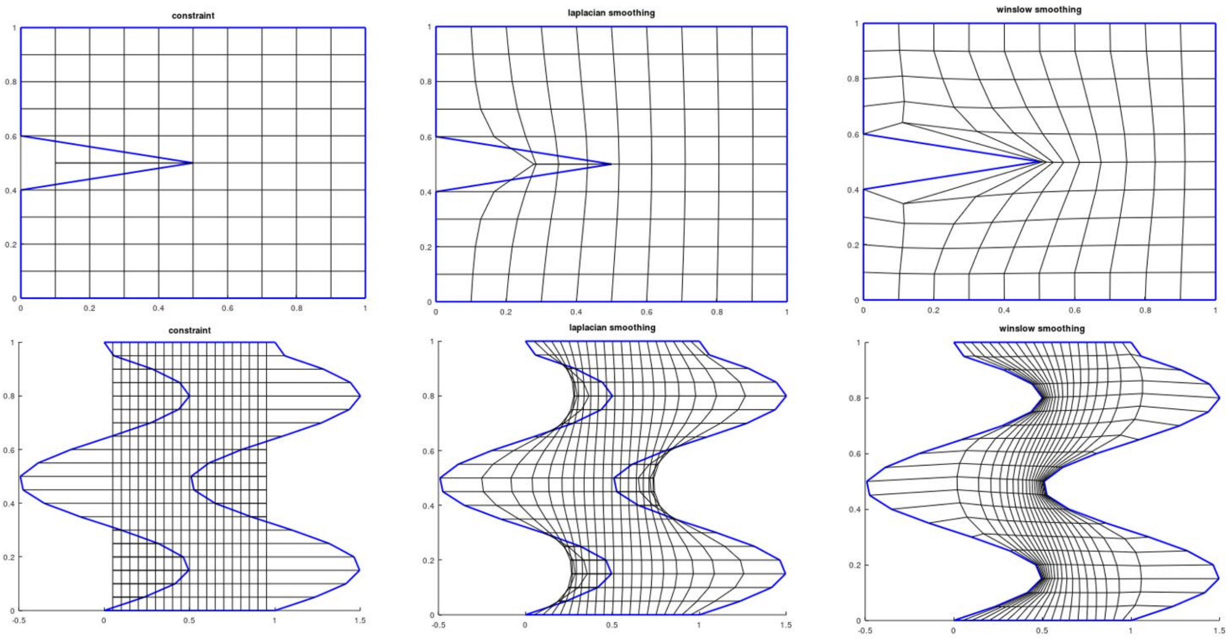

FIGURE 2 – A gauche : l'initialisation, un maillage régulier dont on a déplacé le bord (en bleu). Notre objectif est de déformer le maillage original pour qu'il ait ses mailles les moins distordues (le plus proche du carré possible). Au centre, l'approche classique (*Laplacian Smoothing*) contient des mailles retournées. A gauche, le *Winslow Smoothing* ne retourne aucune maille.

On peut remarquer deux choses : tout d'abord les termes u et v ne sont pas indépendants comme x et y pour le *Laplacian smoothing*. Ensuite, on remarque que cette équation est quasi-linéaire : on peut utiliser des solveurs itératifs en considérant localement α, β et γ constants et en les mettant à jour de temps en temps. Numériquement, cette équation aux dérivées partielles peut se résoudre avec des differences finies ou encore des volumes finis. En effet, on peut réécrire (3) sous la forme :

$$\mathtt{div}\vec{F} = 0 \tag{4}$$

avec $\vec{F} = \begin{pmatrix} \alpha u_x - \beta u_y \\ \gamma u_y - \beta u_x \end{pmatrix}$. Puis, en utilisant le théoreme de flux-divergence (Green-Ostrogradski), on peut discrétiser l'équation sur notre maillage et la résoudre avec des méthodes déjà très étudiées dans la littérature.

— **Objectifs du stage (10 à 20 lignes)**
L'objectif de ce stage est d'étudier la possibilité d'intégration d'autres types de contraintes dans ce système. Ça pourrait être, par exemple, l'orientation, la taille ou encore la position de certaines mailles à l'intérieur du domaine. Pour ce faire, il serait intéressant de re-formuler le problème (3) comme un problème d'optimisation. Lorsque nous aurons obtenu des résultats sur des maillages non structurés en dimension 2, il faudra les étendre en dimension 3 à des maillages tétraédriques.

— **Compétences espérées :** La principale qualité attendue est l'envie d'apprendre et de travailler en équipe. Une bonne intuition géométrique en $3D$ sera un plus. Être à l'aise en programmation sera nécessaire pour pouvoir tester différentes idées. Notre base de code est en C++, mais les difficultés seront principalement d'ordre algorithmiques (ni syntaxiques, ni architecturales).

— **Références bibliographiques :**
[1] Knupp, P. M. (1999). Winslow Smoothing on Two-Dimensional Unstructured Meshes. Engineering with Computers, 15(3) :263-268.