Master GIG 2

Alexandra Bac

TP de modélisation géométrique : Courbures discrètes

A rendre : code + un compte rendu illustrant le travail réalisé. Ce compte rendu fera l'objet d'une note intervenant dans la note finale sur l'enseignement. Temps estimé : 6 heures

Dans ce TP, on réalisera une application OpenMesh mettant en application les notions vues dans le cours.

1 Exercice de base (TP1/2)

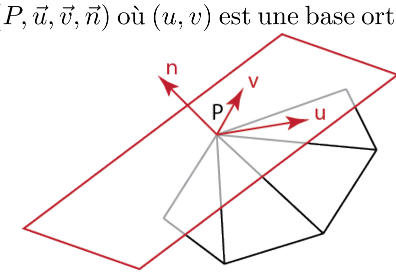
Récupérez le code disponible en matériel de TP sur ma page (fournissant un starting code simple incluant la bibliothèque Eigen ainsi que des fichiers `courbures.h` et `courbures.c`). Le bouton `pushButton_courbures` lancera vos fonctions dans un premier temps.

Le but du TP est d'implémenter un calcul de courbures par approximation par un patch quadratique. La classe `Coubures` vous fournit une bonne partie du le matériel nécessaire dans un premier temps. Elle utilise la bibliothèque template Eigen, fournie.

On veut ajuster localement une surface quadratique au voisinage du point P du maillage, pour cela, on va faire une approximation aux moindres carrés. La figure 1 illustre les étapes de ce calcul.

Précisément :

- Pour calculer un repère local :
  - On calcule tout d'abord la normale $\vec{n}$ au sommet P (une version basique est implémentée dans la classe)
  - Puis on se place dans le repère local $(P, \vec{u}, \vec{v}, \vec{n})$ où (u, v) est une base orthonormale du plan tangent :

    Le passage de la base canonique à ce repère se fait par une translation $\vec{t}$ et une rotation r (calculées dans la fonction `fit_quad`).
- Dans ce repère, on va alors ajuster une surface d'élévation quadratique passant par P aux points du cercle des premiers ou seconds voisins. L'équation de cette surface est :

  $$z = g(x, y) = a_0x^2 + a_1xy + a_2y^2 + a_3x + a_4y$$

  Ayant 5 coefficients, il faut au moins 5 points (plus est mieux) pour pouvoir ajuster une quadrique. Dans `get_two_neighborhood` vous trouverez le code de calcul du 2-voisinage d'un point. Vous l'ajusterez pour vous contenter des premiers voisins quand il y en a assez ... Peut-on appliquer cette méthode sur des maillages "sparse" ?

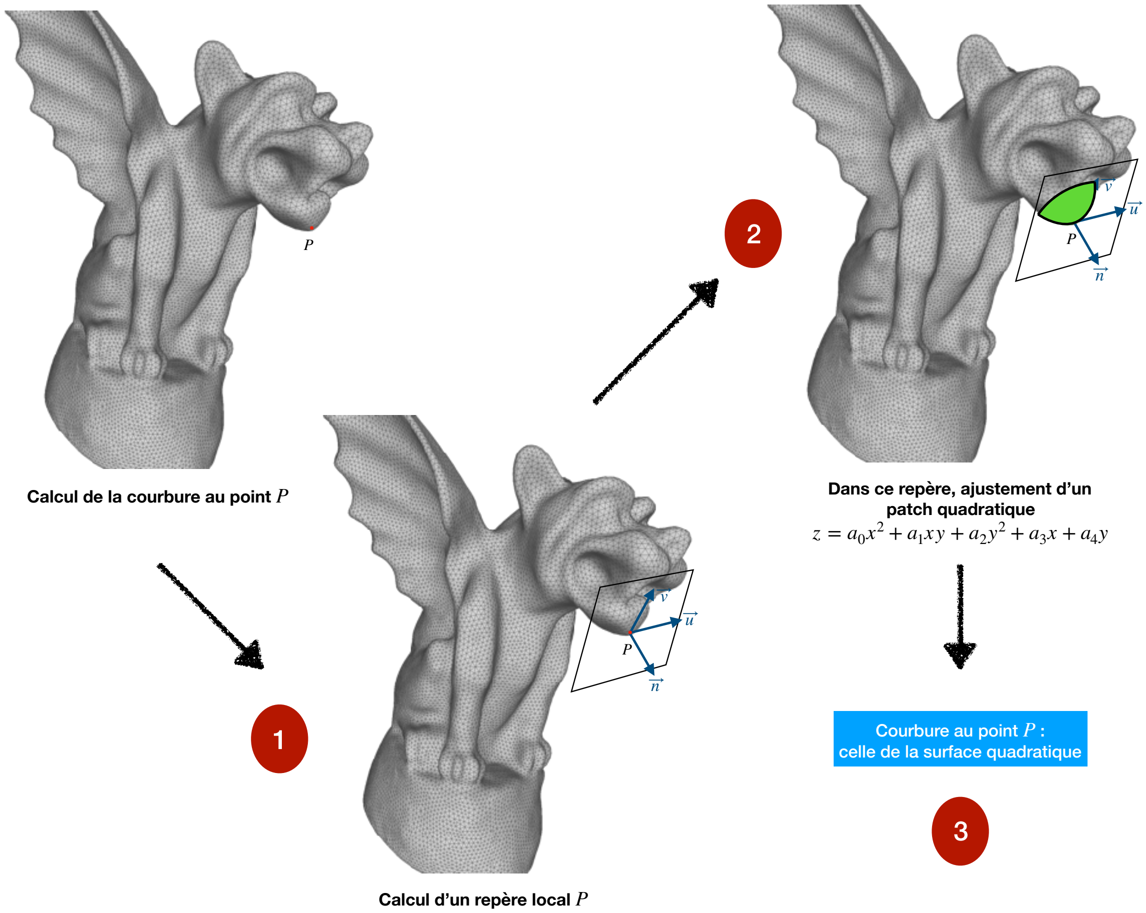


FIGURE 1 – Etapes du calcul de la courbure par approximation quadratique

On note $\{X_i = (x_i, y_i, z_i)\}_{i=1\ldots N\}}$ cet ensemble de points (P et ses voisins). L'erreur au $i$ème point est évaluée comme :

$$\varepsilon_i = (g(x_i, y_i) - z_i)$$

On va donc déterminer la fonction g (dépendant des paramètres $(a_0, \ldots, a_4)$) minimisant (moindres carrés) :

$$J(a_0, \ldots, a_4) = \sum_{i=1}^{N} \varepsilon_i^2$$

Déterminez la matrice A et le vecteur B tels que le vecteur d'erreur s'écrive :

$$\begin{pmatrix} \varepsilon_1 \\ \vdots \\ \vdots \\ \vdots \\ \varepsilon_N \end{pmatrix} = A \cdot \underbrace{\begin{pmatrix} a_0 \\ \vdots \\ a_4 \end{pmatrix}}_{X} - B$$

Le problème d'optimisation aux moindres carrés s'écrit alors :

$$J(a_0, \ldots, a_4) = \|AX - B\|^2$$

Le calcul le plus efficace et stable du minimum de ce problème de moindres carrés linéaires se fait par décomposition en valeurs singulières, ie. SVD (ou transformée QR, c'est équivalent). Vous trouverez cette résolution dans la fonction `fit_quad`. On obtient alors les coefficients $X = (a_0, \ldots, a_4)$ optimaux.

(i) En fonction du cours, pour la surface paramétrique correspondante :

$$\begin{array}{rcl} \varphi : \mathbb{R}^2 & \to & \mathbb{R}^3 \\ (x, y) & \mapsto & \begin{pmatrix} x \\ y \\ a_0x^2 + a_1xy + a_2y^2 + a_3x + a_4y \end{pmatrix} \end{array}$$

Calculez le tenseur de courbure (dans quel repère du plan tangent est-il exprimé ?). Pourquoi peut-on approximer H et K par :

$$K = 4a_0a_2 - a_1^2 \qquad H = a_0 + a_2$$

quand a_3, a_4 sont petits.

(ii) Afficher la carte de courbure pour K puis H

2 Suite du TP 1/2

Vous choisirez l'un des deux "addon" suivant et continuerez votre TP dans cette perspective :

2.1 Visualisation

On souhaite mieux visualiser l'ajustement quadratique et les résultats de coubures obtenus pour le maillage. Vous implémenterez donc une visualisation de la surface quadratique en un sommet P donné (un peu comme sur l'image 1) :

(i) on fera ressortir d'une couleur particulière, sur le maillage d'origine, les sommets voisins de P intervenant dans l'approximation

(ii) puis on génèrera un maillage du patch quadratique au voisinage de P (attention de bien réappliquer translation et rotation pour vous retrouver dans le repère initial) - idéalement le sommet P sera sélectionné par picking sur le maillage

(iii) vous testerez alors l'impact sur la surface quadratique de la taille du voisinage (premiers ou premiers+seconds voisins)

2.2 Courbures

On a calculé la matrice de courbures ... il est tout de même dommage de l'avoir fait pour n'extraire que H et K :

(i) Comment calculer les courbures principales et directions principales et comment replacer ces dernières dans le repère global (grâce à $\vec{t}$ et r) ?

(ii) Afficher en chaque sommet en rouge la direction principale correspondant à $\max(\kappa_1, \kappa_2)$ et en vert la direction correspondant au minimum.

(iii) Implémenter une fonction colorant les sommets par type en fonction des courbures principales (une couleur par type)