

## Modélisation Géométrique : Filtrage

*L'archive du TP se trouve sur la page du cours.*

On partira du code du programme réalisé au TP précédent. La philosophie générale de cette partie et des suivantes est d'associer des opérateurs de traitement géométrique à différentes touches du clavier dans le programme. On notera la présence d'une classe `Mesh` : l'objectif du TP est de l'enrichir de nouveaux opérateurs. Quelques méthodes simples sont fournies, notamment la méthode de chargement de maillages depuis un fichier OFF (`loadOFF`) et une méthode simple de calcul des normales (`recomputeNormales`). Tous les traitements implémentés seront appliqués au maillage courant à l'aide de touches clavier, afin de pouvoir enchaîner plusieurs traitements à la suite. On modifiera la fonction `void keyboard (unsigned char, int, int)` du programme pour activer les traitements, et on s'assurera que le texte affiché par la touche 'h' est à jour (i.e., indique quelle touche active quel opérateur).

### I. Filtrage passe-bas laplacien

On se propose d'implémenter un opérateur de lissage de maillage sous la forme d'un filtrage laplacien. On pourra le tester sur le modèle `max_50k.off`.

1. Implémenter une méthode `void laplacianFilter ()` dans la classe `Mesh` qui réalise un filtrage laplacien du maillage. Cette méthode modifiera la position de chaque point en les déplaçant le long du « vecteur laplacien » (topologique).

Principe :

1. Calculer le barycentre du 1-voisinage de chaque sommet
2. Déplacer chaque sommet vers son barycentre associé
3. Recalculer les normales

2. Remplacer le laplacien topologique par un laplacien géométrique à l'aide, notamment, des poids co-tangent (c.f. cours). En profiter pour modifier la méthode de calcul des normales, de manière à prendre en compte les angles formés par les faces incidentes aux sommets.

3. Ajouter un paramètre de contrôle `alpha`, compris entre 0 et 1 permettant de moduler le filtrage (1 = déplacement complet). Tester en associant les touches '1', '2' et '3' à l'application d'un lissage avec `alpha` égal à 0.1, 0.5 et 1.0. Penser à rajouter un raccourci clavier permettant de recharger l'objet d'origine.

### **Bonus 1 : Structure en demi-arête**

Enrichir la classe `Mesh` d'un vecteur de `HalfEdge` (cf. cours) et les exploiter dans l'opérateur de filtrage.

### **Bonus 2 : Filtrage bilatéral**

Ajouter un second filtre, `void bilateralFilter (float rangeSupport, float domainSupport)` à la classe `Mesh`, qui préserve mieux les arêtes vives et les détails géométriques. Pour cela, on implémentera la méthode de filtrage bilatéral non-itérative suivante :

### **Non-Iterative, Feature-Preserving Mesh Smoothing**

Thouis R. Jones, Frédo Durand, Mathieu Desbrun  
SIGGRAPH 2003  
<http://people.csail.mit.edu/thouis/JDD03.pdf>