

Maillage par raffinement de Delaunay 3D

Le projet fourni est un projet QT4/QGLViewer donc en théorie multi-plateforme.

Vous pouvez compiler en ouvrant le fichier CMakeLists.txt avec cmake.

Vous devez avoir des variables d'environnement définies :

- CGAL_DIR -> répertoire de CGAL
- PATH doit contenir les répertoires lib et bin de qt
- QTDIR -> répertoire de Qt (fonctionne avec 4.7.0)

La démo Qt4 est minimaliste : elle instancie un `complex-3-in-triangulation-3` défini dans `c3t3.h` (voir `scene.h`), et utilisé par le mailleur 3D (cf http://www.cgal.org/Manual/latest/doc_html/cgal_manual/Mesh_3/Chapter_main.html). Le fichier `mesh.ui` peut être ouvert avec Qt Designer pour ajouter des menus. Les fichiers `window.*` montrent des exemples de gestion de menus.

- Parcourir le code des fichiers `Scene.*` et `c3t3.h`. La scène contient les données (ici un `c3t3`, le viewer les affiche et la classe `MainWindow` gère l'interface principale). La démo permet de mailler une fonction implicite simpliste : une sphère unité comme illustré dans la doc : http://www.cgal.org/Manual/latest/doc_html/cgal_manual/Mesh_3/Chapter_main.html#Subsection_50.3.1, ladite fonction étant définie dans le fichier `function_sphere.h`. Les paramètres principaux pour le maillage sont modifiables via SHIFT-P (menu Algorithms / Parameters).
- Lancer des tests avec différents paramètres sur la sphère.
- Ecrire une fonction implicite pour un ellipsoïde (fichier `function_ellipsoide.h`), et tester. Idem pour un tore (fichier `function_torus.h`).
- Pour une fonction choisie, expérimenter en lançant plusieurs générations de maillages en faisant varier les paramètres de forme. Ecrire une fonction dans la classe scène qui génère en boucle une série de maillages pour différents paramètres d'erreur d'approximation, et tracer le nombre de sommets du maillage en fonction de ce paramètre.
- Ecrire une fonction implicite pour un cube, et observer le comportement du mailleur au voisinage des arêtes vives.
- Etant donné un ensemble de points 3D (à définir en dur). Mettre au point une fonction implicite qui représente une union de boules centrées sur ces points.
- Ecrire une fonction implicite multi-domaine en combinant plusieurs domaines par des opérations booléennes, et afficher les cellules de différentes couleurs suivant l'index du domaine (cf fonction `subdomain_index()` depuis http://www.cgal.org/Manual/latest/doc_html/cgal_manual/Mesh_3_ref/Concept_MeshComplex_3InTriangulation_3.html).