

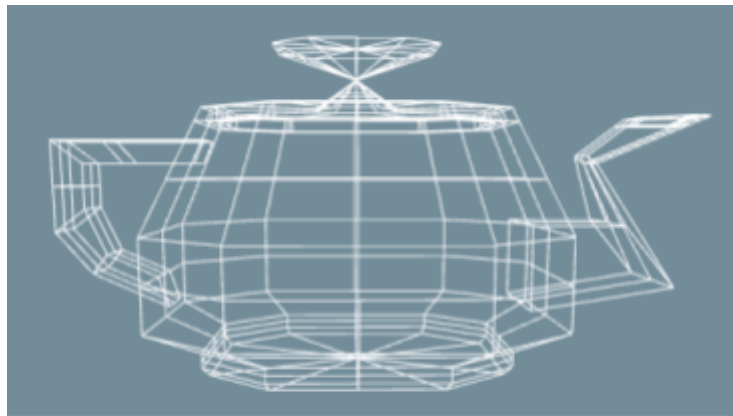
# Modélisation géométrique

## Modélisation

### Bézier

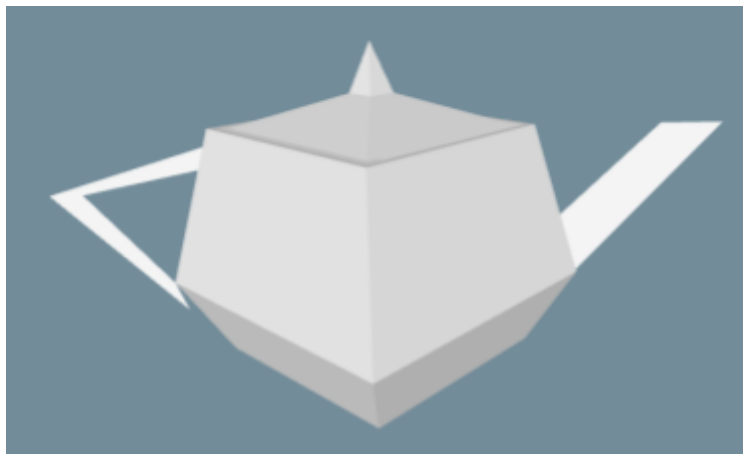
Dans les images suivantes figure la fameuse “Utah Teapot” originellement décrite par Martin Newell, en 1975, à l’aide de 32 carreaux de Bézier cubiques (4 par 4).

Les carreaaux sont manuellement définis de manière à être continus.



*Mesh des points de contrôles*

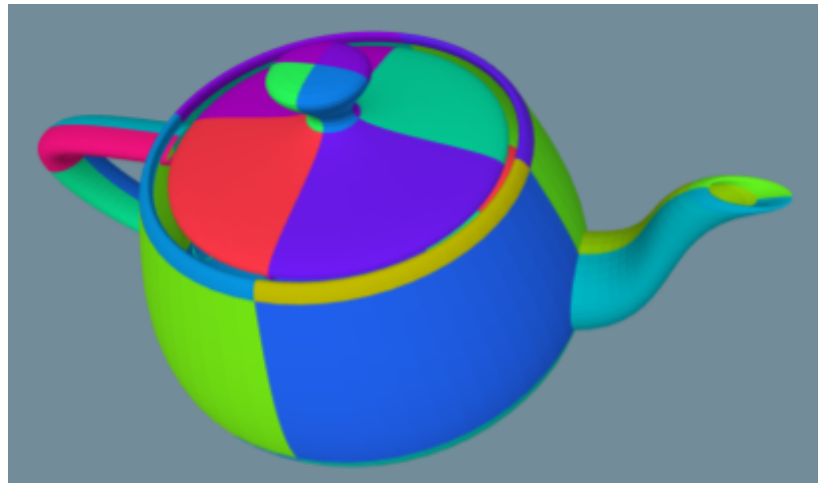
En reliant les extrémités des carreaux (soit une tessellation de résolution 2 par 2), on obtient le mesh suivant:



*Mesh reliant les quatres  
extrémités de chaque carreau.*

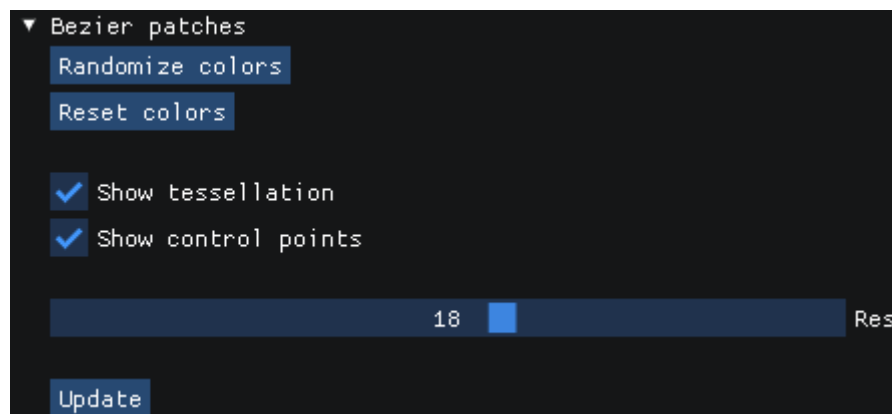
On observe qu'en procédant ainsi, on n'obtient pas nécessairement un objet volumique et que la contribution des points de contrôle intermédiaires est nulle.

Avec une résolution plus grande, on obtient bien un objet volumique et visuellement lisse (classe  $C^1$  du fait de carreaux cubiques):



*Coloration des différents carreaux tessellés  
avec une résolution de 18 par 18.*

Notre interface nous permet de changer la résolution des carreaux:



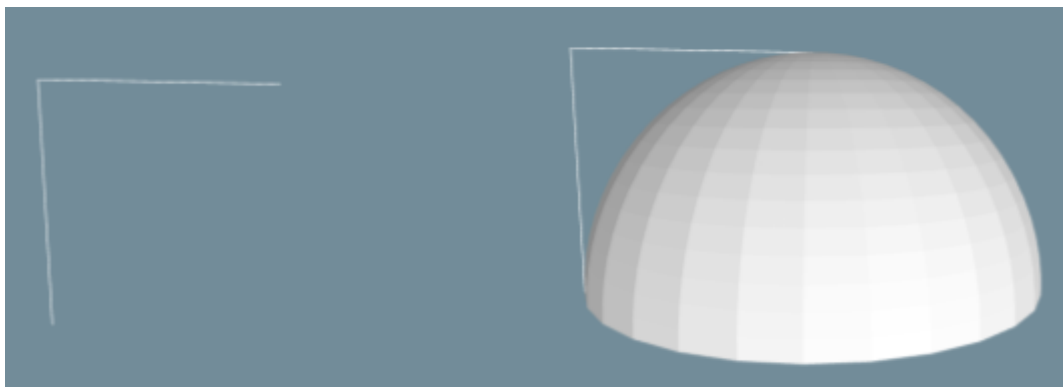
## Surface de révolution

À partir d'une courbe de Bézier et d'un axe de rotation, nous disposons de la faculté de générer une surface de révolution:



À gauche, courbe de contrôle avec 5 points; à droite, surface de révolution avec une résolution axiale de 30 et radiale de 30.

Nous avons également implémenté les courbes de Bézier rationnelles pour produire des courbes circulaires:



À gauche, courbe de contrôle avec 3 points de poids (1, 0.7, 1); à droite, la surface de révolution résultante: une hémisphère.

La courbe peut être modifiée interactivement à l'aide de notre interface.

▼

Revolution surface

☒ Show tessellation
 ☒ Show control points

3

# of control points

▼

Control points

X/Y/Weight:

2.000	0.000	1.000	#0
2.000	2.000	0.700	#1
0.000	2.000	1.000	#2

19

Axial resolution

30

Radial resolution

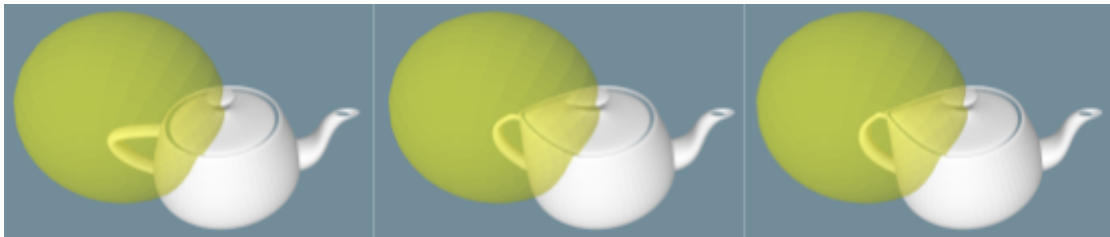
Update

# Déformation de modèle

Nous avons implémenté des transformations quelconques sur les modèles précédemment générés.

Une fonction de transformation est associée à une fonction d'atténuation. La fonction d'atténuation d'une transformation globale est une constante.

Nous avons réalisé, en premier lieu, une déformation d'attraction; les points sont attirés vers le centre de la transformation:



*La sphère jaune représente la zone d'effet de la transformation. À gauche, aucune transformation; au milieu, atténuation linéaire; à droite, atténuation cubique.*

Nous avons également implémenté une sorte d'extrusion avec le "bend" de Barr:



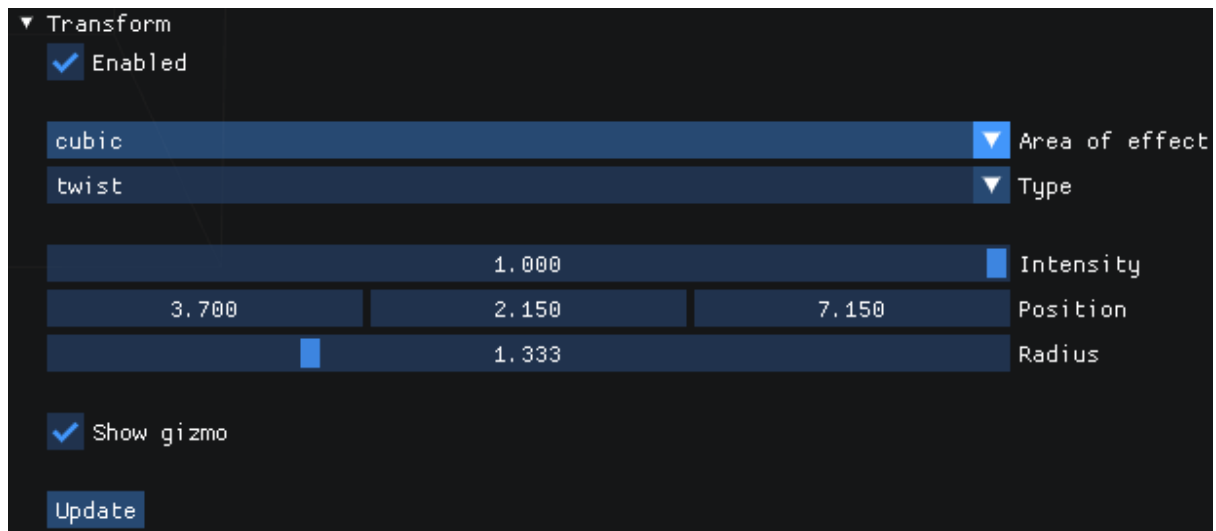
**Extrusion.** *À gauche, aucune transformation; au milieu, atténuation linéaire; à droite, atténuation cubique.*

Ainsi que le "twist" de Barr:



***Twist.** À gauche, modèle de référence; au milieu, twist global; à droite, twist local.*

Les déformations peuvent être interactivement manipulées avec notre interface:

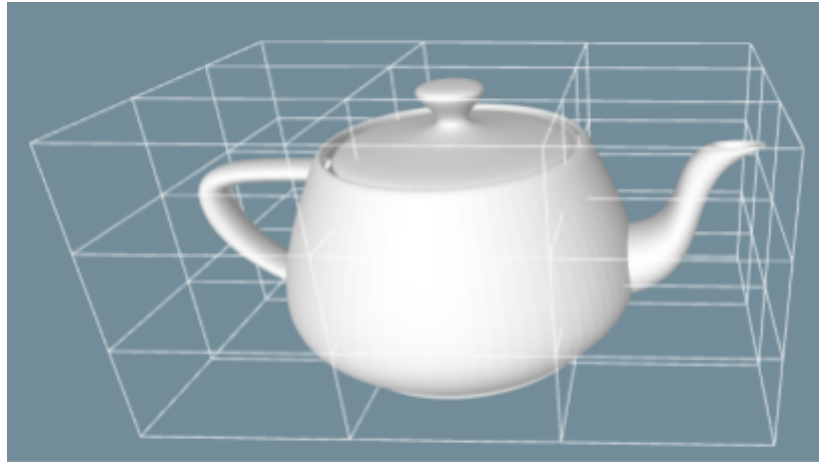


## Free-form deformation

Nous avons implémenté la “free-form déformation” introduite par Sederberg et Parry en 1986.

À cette fin, nous calculons la boîte englobante du modèle courant et définissons naïvement une paramétrisation  $[0, 1]^3$  sur celle-ci. Chaque sommet du modèle peut par conséquent être transformé en calculant son correspondant dans un volume de Bézier à partir de sa paramétrisation dans l’englobant.

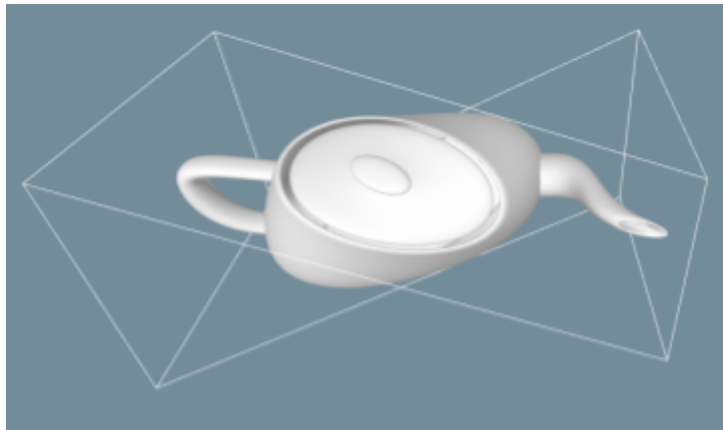
Le volume de Bézier peut être découpé un nombre arbitraire de fois selon chaque dimension:



*Division de l'englobant sur une grille 4x4x4.*

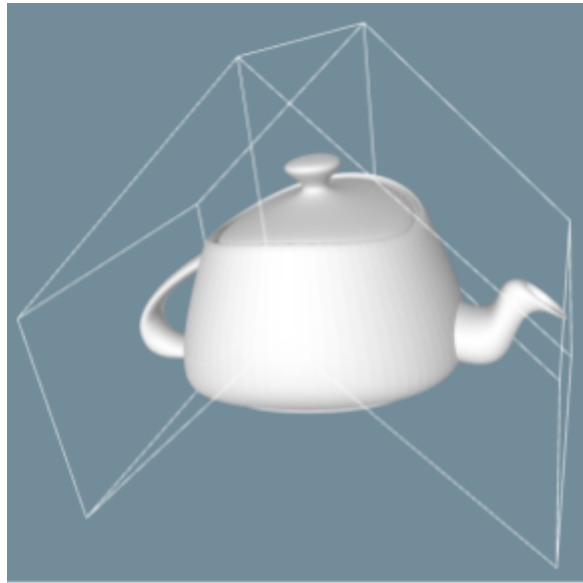
Des transformations de natures très différentes sont possibles grâce à ce modèle.

Une transformation similaire au “twist” de Barr peut être obtenue en transformant les coins de l'englobant:



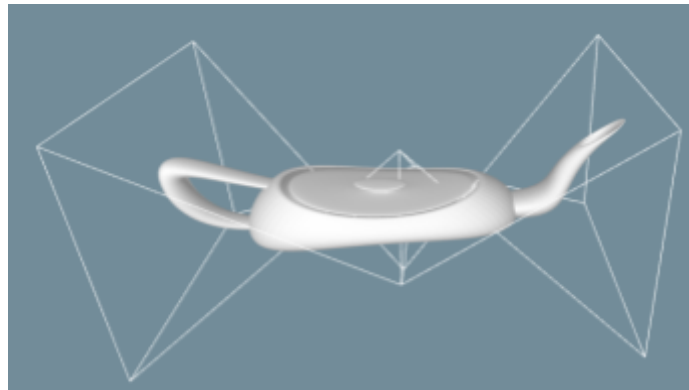
*Rotation de 90 degrés  
entre deux faces opposées.*

Similairement au “bend” de Barr:



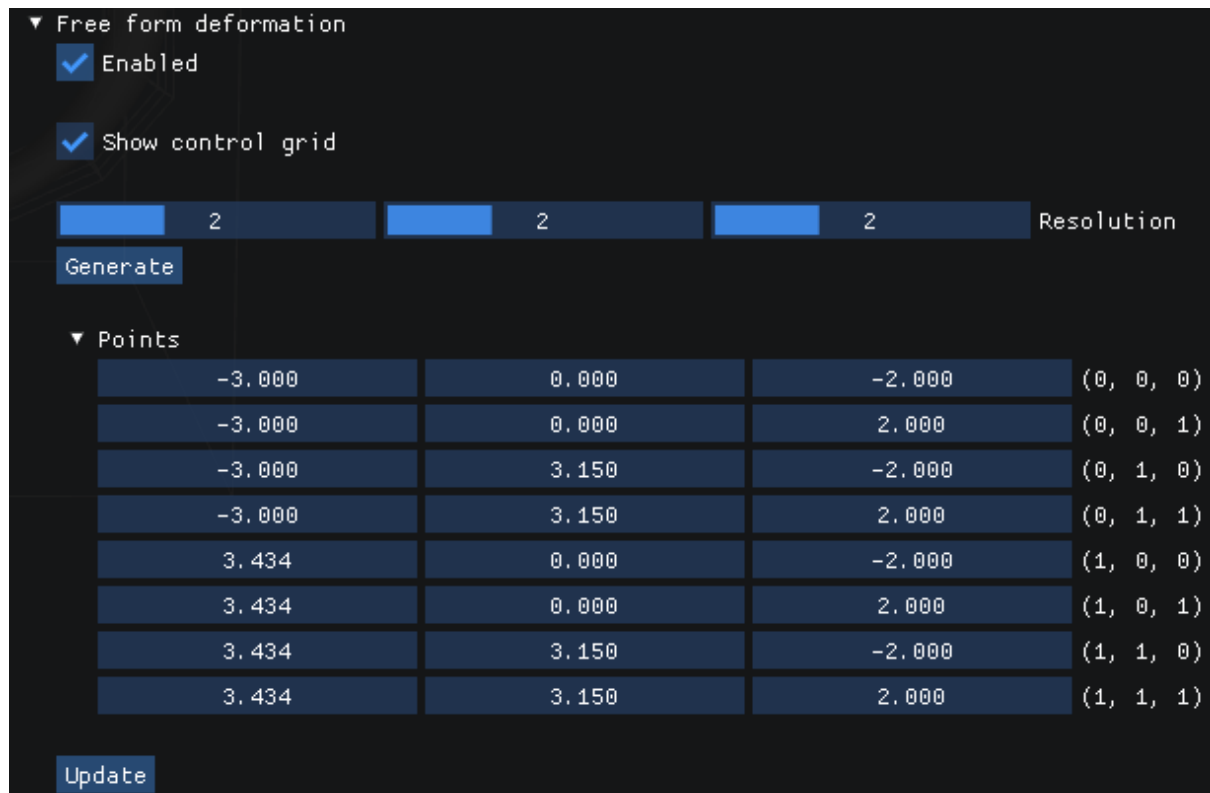
*Translation des noeuds internes.*

Et enfin au “taper” de Barr:



*Écrasement des noeuds internes  
sur grille 3x2x2.*

Nous avons pris soin de développer une interface pour l'édition de la grille afin de pouvoir itérer rapidement et tester de nombreuses configurations.



## Implémentation

En somme, nous supportons, dans notre implémentation, les courbes, surfaces et volumes de Bézier.

Le calcul d'un point prend en entrée une grille de points de contrôle de dimension  $N$  et un vecteur dans l'espace paramétrique  $[0, 1]^N$ .

Nous avons implémenté une grille comme un tableau associé à un schéma d'indexation définissant la dimension de la grille et la résolution le long de chaque axe.

Tous nos algorithmes sont basés sur cette structure.

Toutes les opérations proposées par l'interface sont perceptuellement instantanées.

L'application a été réalisée à l'aide de bibliothèques C++ développées personnellement utilisant Dear ImGui pour l'interface.