

Première approche de l'impression 3D

Francesco De Comité
Université des Sciences de Lille (France)



26 mai 2014

Résumé

On parle souvent de l'impression 3D et des prouesses qu'elle permet d'accomplir, dans les domaines médicaux ou scientifiques par exemple. Cette technique est présentée comme étant un peu magique, permettant de produire n'importe quel objet, facilement et pour pas cher. Le but des manipulations que nous vous proposons est de vous faire comprendre la chaîne complète qui va de l'idée initiale d'un objet à construire, passe par sa modélisation mathématique, sa programmation et enfin son impression.

Première partie Introduction

1 La chaîne de conception

Pour produire un objet, on utilise un logiciel de modélisation 3D pour le définir de façon virtuelle. Lorsque l'objet correspond à ce qu'on veut, on le sauvegarde dans un fichier (dans un format compris par le logiciel de l'imprimante), puis on l'envoie à l'imprimante. Un programme informatique envoie alors aux moteurs de l'imprimante les commandes qui permettent de créer l'objet.

2 Le logiciel de modélisation

Le logiciel de modélisation que nous allons utiliser est Blender¹ : c'est un logiciel ouvert, gratuit, très complet, en développement constant et disponible sur tous les systèmes d'exploitation. Blender est assez compliqué à apprendre et nous nous limiterons à ce qu'il est nécessaire de savoir pour le but qui est le nôtre : définir un objet 3D. Un point fort de Blender est qu'il est possible de définir des objets soit directement à la main

1. Disponible à l'adresse blender.org

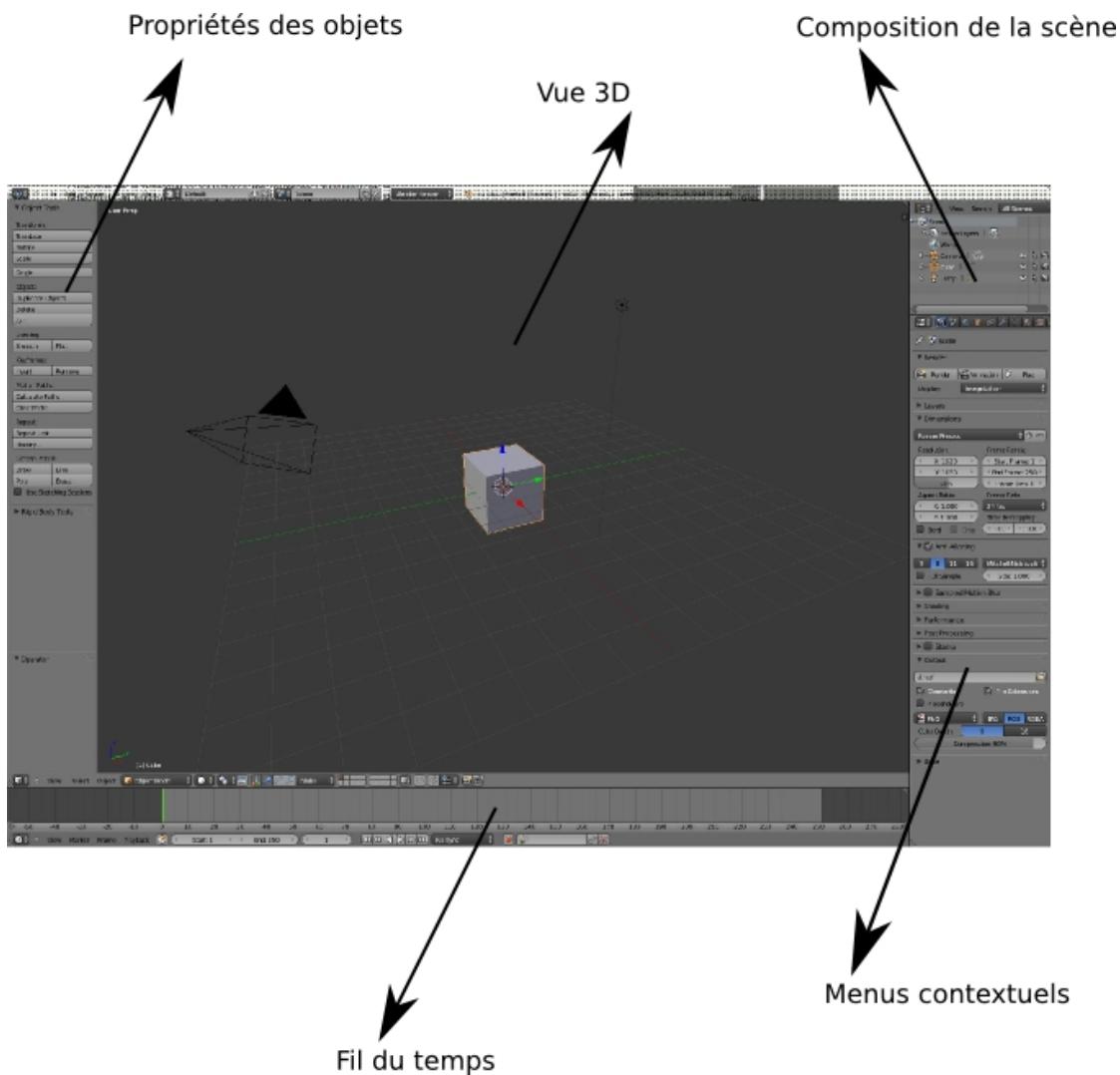


FIGURE 1: L'écran de Blender

(en sculptant une forme dans une fenêtre 3D), soit par programme : on peut écrire des programmes (appelés scripts) en Python pour construire des objets. Cela nous ouvre des perspectives nouvelles : nous allons être capables de définir des objets mathématiques, les programmer et leur donner forme !

3 Plan de la manipulation

La manipulation se divise en deux parties : La première partie vous apprend l'utilisation basique de Blender et la façon de programmer un objet simple. La seconde met en pratique ces notions pour la construction d'un objet plus compliqué, et sans doute impossible à réaliser proprement par d'autres techniques.

Deuxième partie

Les bases de Blender pour l'impression 3D

4 Mise en route

Blender est présent par défaut sous toutes les distributions Linux, et il existe des installateurs automatiques pour Windows et Mac. De nouvelles versions apparaissent souvent. Nous utiliserons des variantes de la version 2.6.

Après avoir lancé Blender, vous obtenez l'écran de la figure 1. Cet écran contient quatre zones principales :

- La vue 3D qui contient la scène en cours de définition. Cette scène est composée des objets suivants :
 - Un cube, au centre de l'écran. Il est entouré d'un liseré orange : il a été sélectionné, c'est un objet actif (celui sur lequel on peut appliquer des actions).
 - Une caméra (la pyramide à gauche de la vue) : elle définit ce qui sera vu lorsqu'on fera un rendu(une photo) de la scène.
 - Une lampe, hors de vue pour le moment.
- A gauche, la fenêtre des propriétés de l'objet : on peut y définir différentes transformations de l'objet. On ne l'utilisera pas dans le cadre de ce TP.
- En haut à droite, la liste des objets présents dans la scène. On y retrouve le cube, la caméra et la lampe, ainsi que le monde, qui contient les définitions globales (couleur du ciel ...)
- Juste en dessous, la fenêtre des propriétés qui permet d'agir sur la scène de beaucoup de façons différentes. Les actions possibles dans cette fenêtre dépendent de l'objet sélectionné.

5 La vue 3D

Quelques notions de base sur la navigation dans la fenêtre Vue 3D :

- Assurez-vous que le curseur de la souris soit dans cette fenêtre (la fenêtre active est toujours celle où se trouve la souris).
- En bougeant la souris tout en maintenant le bouton central enfoncé, on effectue une rotation dans l'espace. Tourner la molette centrale permet de zoomer/dézoomer.
- Shift-tourner la molette centrale déplace la scène verticalement.
- CTRL+tourner la molette centrale déplace la scène horizontalement.
- Click droit sur un objet le sélectionne.
- Click gauche sur un objet le translate (évitez d'utiliser cette méthode pour déplacer les objets, si besoin est, nous vous en donnerons d'autres).
- La touche 'A' sélectionne tout ou désélectionne tout.
- Les touches du pavé numérique modifient l'angle de vue. En particulier, la touche '0' (zéro) donne la vue de la caméra.
- La combinaison CTRL-ALT-q scinde la vue 3D en 4 fenêtres donnant 4 vues différentes (bien pratique pour voir où on en est dans un espace à 3 dimensions).
- CTRL-flèche vers le haut met la fenêtre courante en plein écran (ça marche pour toutes les fenêtres).
- CTRL-flèche vers le bas restitue la taille initiale de la fenêtre.
- CTRL-Z rattrape les bêtises en annulant la dernière action.

6 Les objets

6.1 Manipulation

Quelques notions de base pour la manipulation des objets :

- Assurez-vous que votre souris est bien dans la fenêtre Vue 3D, et sélectionnez le cube (et seulement le cube!).
- La touche 'g' (pour Grab) permet de déplacer l'objet, soit en mode libre (bouger la souris), soit en contraignant le déplacement sur l'un des trois axes : 'g x' permet un déplacement le long de l'axe des X (dont l'orientation est donné par le petit schéma en bas à gauche de la vue 3D). De même 'g y' et 'g z' contiennent le mouvement sur l'axe correspondant.
- La touche 'r' fait de même pour les rotations de l'objet.
- la touche 'n' ouvre une fenêtre à droite de la vue 3D, qui contient toutes les infos concernant l'objet : position, orientation, taille... On peut par exemple y changer la taille de l'objet, en jouant sur les paramètres **scale**. Une deuxième frappe sur la touche 'n' referme cette fenêtre.

6.2 Comment sont définis les objets

Avant d'aller plus loin et définir nos propres objets, voyons comment Blender les définit. Dans la fenêtre Vue 3D, sélectionnez le cube. En bas à gauche de la fenêtre, une case



FIGURE 2: Bas de Vue 3D en mode Objet



FIGURE 3: Bas de Vue 3D en mode édition



FIGURE 4: Pour choisir les composants de base

contient 'Object Mode'. Cliquez sur cette case, et dans la liste qui apparaît, choisissez 'Edit Mode'. Remarquez comment la barre qui contient ce bouton a changé (figures 2 et 3)

En mode édition, on a accès aux composants primitifs d'un objet, à savoir les sommets (vertex/vertices), les arêtes (edges) et les faces. La définition d'objets 3D se ramène souvent à la description de ces composants et de leurs relations. En mode édition, la barre du bas de la fenêtre Vue 3D contient trois boutons (figure 4) qui permettent de travailler, de gauche à droite, sur les sommets, les arêtes ou les faces. Le quatrième bouton permet de choisir de voir ou non les parties cachées. Comme en mode objet, on peut sélectionner un composant (sommet, arête ou face selon le bouton choisi). On peut aussi sélectionner plusieurs composants :

- Les uns à la suite des autres : sélectionnez le premier (click droit) puis maintenez CTRL enfoncé en sélectionnant les suivants.
- Par bloc : Touche 'b', puis cliquez du bouton central, maintenez le bouton enfoncé pendant que vous définissez un rectangle.

Outre les transformations déjà rencontrées (translation et rotation), on peut maintenant étirer un composant sélectionné avec la commande 'e' (pour Extrude). C'est de cette façon qu'on peut, avec un peu de patience, sculpter des objets, en utilisant ensuite d'autres outils pour la finition.

On peut aussi écrire un programme qui définit un objet dans Blender : on crée les sommets de l'objet, on donne ensuite la liste des faces de l'objet en indiquant quels sont les sommets qui la définissent. En lançant le programme, on voit apparaître l'objet programmé dans la fenêtre Vue 3D. C'est ce que nous allons faire à partir de maintenant.

Troisième partie

Programmer un objet

Il nous faut un programme qui calculera un objet et l'affichera dans la fenêtre Vue 3D. Pour cela, nous devons ouvrir une nouvelle fenêtre : la fenêtre de script, aussi appelée **Text Editor** :

- Placez-vous en haut de la fenêtre Vue 3D. Lorsque le curseur de la souris devient une flèche à deux pointes, cliquez droit et choisissez **split area** (séparer la zone). Vous pouvez choisir une séparation horizontale ou verticale à l'aide de la touche de tabulation.
- Bougez la souris jusqu'à obtenir une fenêtre de taille convenable (vous pourrez l'ajuster plus tard).
- Ouvrez le menu en bas à gauche de cette nouvelle fenêtre, et choisissez **Text Editor**
- Ouvrez le menu **Text** du bandeau inférieur de cette fenêtre, et choisissez **Open text block** puis allez chercher le fichier **icosaedre.py**

Le fichier s'affiche dans la nouvelle fenêtre, c'est du code Python. Il est facile à comprendre, les lignes intéressantes ici sont reprises dans l'annexe ?? page ??.

Ligne 2 : déclaration de la fonction `icosaedre()`.

Ligne 3 : un `mesh` est un solide défini par des sommets et des faces. On crée un nouveau `mesh` dont on va calculer les caractéristiques.

Lignes 4, 5 et 7 : Création de deux tableaux pour contenir les sommets et les faces.

Ligne 6 : on a besoin du nombre d'or ($\frac{1+\sqrt{5}}{2}$) pour calculer les coordonnées des sommets.

Ligne 8 à 19 : définition des 12 sommets de l'icosaèdre.

Lignes 20 à 39 : définition des 20 faces triangulaires de l'icosaèdre.

Lignes 40 et 41 : Ranger ses données dans le `mesh`.

Ligne 42 : retour au programme principal.

Ligne 45 : appel de la fonction de création d'un icosaèdre, rangé dans l'objet `myMesh`

Lignes 44,46 et 47 : intégration de cet objet à la scène Blender.

Les lignes manquantes dans ce poly concernent soit l'initialisation de Python ou Blender, soit la mise en couleur des faces.

Placez-vous en mode **Object mode** dans la fenêtre Vue 3D, sélectionnez le cube, et demandez sa suppression avec la touche **suppr** du clavier. En plaçant la souris dans la fenêtre Text Editor, cliquez droit, puis choisissez **Run Script** : l'icosaèdre apparaît à la place du cube. Pour voir le cube avec les couleurs que le programme a affecté à chacune de ses faces, passez de **Object Mode** à **Texture Paint** dans la fenêtre Vue 3D.

6.3 Prendre une photo

Vérifiez que l'icosaèdre est bien dans le champ de vision de la caméra (c'est le cas si vous n'avez rien modifié ou si vous relancez Blender) en choisissant le point de vue correspondant à la touche '0' (point de vue de la caméra) pour la Vue 3D. La photo (on parle de *Rendu*) est obtenu en cliquant sur le bouton **Render** en haut à gauche de

la fenêtre globale... pas de chance, c'est en noir et blanc ! Pour avoir les couleurs sur le rendu, vous devez :

- Dans la fenêtre des propriétés (à droite), choisir le bouton **Material**.
- Cliquez sur **new**
- Dans **Options**, cochez **Vertex Col Paint**
- Vous pouvez maintenant demander un rendu avec les bonnes couleurs.
- Le menu **Image**, en bas de la fenêtre où figure la photo, vous permet de sauvegarder l'image.
- Le menu **Render** dans la fenêtre des propriétés permet de régler la qualité de l'image.

6.4 Exporter l'objet pour l'impression 3D

Les objets en couleur doivent avoir des parois d'au moins trois millimètres pour être imprimables par les machines actuelles. On peut vérifier les dimensions globales de l'objet en ouvrant la fenêtre des propriétés de l'objet (tapez 'n' dans la fenêtre Vue 3D) : vous voyez que ses dimensions sont de 3.236 mm (tiens ! deux fois le nombre d'or !) dans les trois directions de l'espace (les dimensions dans Blender sont en millimètres par défaut). S'il était trop petit, on pourrait, toujours dans la fenêtre des propriétés de l'objet, changer sa taille en changeant les paramètres de **Scale**.

Pour exporter l'objet en vue d'une impression 3D :

- passez en mode **Texture Paint**
- Sélectionnez l'objet à exporter.
- Dans le menu **File**, choisissez **Export**
- Pour une version couleur, choisissez le format **x3D** : c'est le choix à faire pour ce stage.
- Pour une version monochrome, choisissez **stl**.

Il faudrait encore ici vérifier la viabilité de votre modèle, mais les programmes correspondants ne sont pas disponibles de base sur nos machines. On peut citer Meshlab et netfabb, qui proposent des versions gratuites de leurs logiciels. Vous pouvez aussi charger votre fichier sur les sites de Sculpteo ou Shapeways, deux compagnies qui proposent des services d'impression 3D, et qui vous diront au chargement de votre fichier, s'il est imprimable ou non. Vous aurez aussi une idée de son coût.

Quatrième partie

Un objet plus sérieux

L'objet que nous vous proposons de réaliser est plus compliqué qu'un simple icosaèdre, mais il ne devrait pas vous prendre beaucoup de temps à concrétiser, puisque l'on va suivre exactement la même procédure que pour l'icosaèdre. Pour partir sur des bases saines, quittez puis relancez Blender.

- Ouvrez une fenêtre de script.
- Chargez le programme **villarceau.py**
- Lancez le script.

– Pour visualiser ou photographier les couleurs, référez-vous à la manipulation précédente. Vous avez obtenu ce que nous appellerons un tore de Villarceau. Une définition détaillée est donnée en annexe. Dans un premier temps, on peut se contenter de quelques explications basiques.

Le tore de Villarceau est composé d'un certain nombre de tranches inclinées, elle-mêmes composées de deux quartiers de lune (on parlera de lunules) de couleurs différentes. On peut jouer sur les paramètres suivants (entre les lignes 129 et 152 du programme) :

- Nombre de tranches.
- Epaisseur des lunules (sauf si vous pensez changer d'échelle, gardez 3mm)
- Angle d'inclinaison des tranches.
- Taille des cercles composant les lunules.
- Ecart entre les centres des cercles.
- Nombre de segments définissant une lunule.
- Couleurs des lunules en RGB. Une couleur codée en RGB est un vecteur de trois valeurs comprises entre 0 et 1 : les proportions de rouge, vert et bleu (Red, Green, Blue) définissant une couleur. Dans le fichier de base, les couleurs codées sont bleu (0,0,1) et rouge (1,0,0).

Vous pouvez modifier ces paramètres, effacer le tore précédent et relancer le programme. Normalement, ce que vous avez appris dans les chapitres précédents devrait vous permettre de définir votre version du tore de Villarceau (taille, couleur, aperçu avant impression, sauvegarde d'une image). Lorsque vous êtes satisfaits du résultat, sauvegardez le fichier et rendez-le aux enseignants, qui se chargeront de l'imprimer et de vous le faire parvenir. Bon courage et bonne chance² !

Cinquième partie

Compléments

7 Le tore de Villarceau

7.1 Le cadre mathématique

Sur une sphère, on peut tracer un système de cercles tel que par tout point de la sphère passent exactement deux cercles : les parallèles et les méridiens. On peut faire de même sur un tore, en le coupant avec un plan horizontal ou vertical passant par le centre du tore. Pour le tore, il existe un deuxième système de cercles : les cercles de Villarceau. On les obtient en coupant le tore par un plan passant par le centre du tore et tangent deux fois au tore (voir figure 8). Si on regarde la coupe du tore par ce plan, on voit qu'elle a la forme de deux *lunules*). Cette structure a donné l'idée à Yoshinobu Miyamoto, un professeur d'architecture japonais, de réaliser une approximation du tore à partir d'un ensemble de lunules en papier à imbriquer (figures 6 et 7)³.

2. L'informatique comporte toujours une petite part de hasard et de magie...

3. téléchargeable ici : https://www.flickr.com/photos/yoshinobu_miyamoto/5555289192

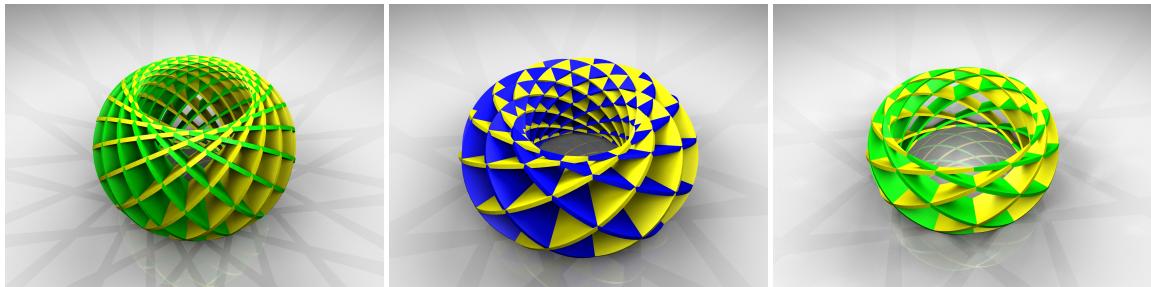


FIGURE 5: Quelques exemples des variations possibles



FIGURE 6: Un tore en kit ...

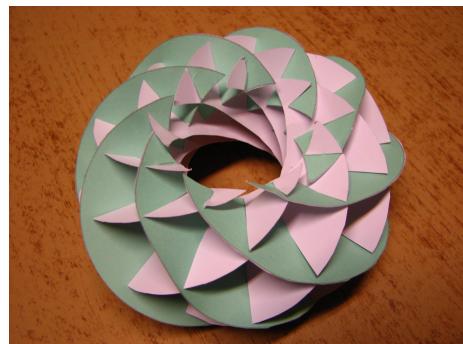


FIGURE 7: ...et sa réalisation.

Pour construire une lunule, on a besoin de connaître deux choses : le diamètre des cercles qui se croisent, et la distance entre les centres de ces deux cercles. Ce sont les variables `rayon` et `distance` utilisés dans le programme `villarceau.py` (voir figure 9)

7.2 De la surface mathématique à la lunule physique

Pour pouvoir imprimer une lunule en trois dimensions, on a besoin de :

- Premièrement de la définir comme un volume (il faut lui donner une épaisseur).
- Deuxièmement, de la définir comme une union de triangles élémentaires. En effet, les représentations d'objets tri-dimensionnels manipulés par les logiciels des imprimantes 3D ne connaissent que ce genre d'objet.

7.2.1 Gonfler la lunule

Pour passer de la surface à un volume "équivalent", on va lui donner une épaisseur : on aura maintenant 4 surfaces : les deux faces de la nouvelle lunule, et les côtés qui les relient (voir figure 10). Cette nouvelle variable, l'épaisseur, devient un des paramètres de notre programme.

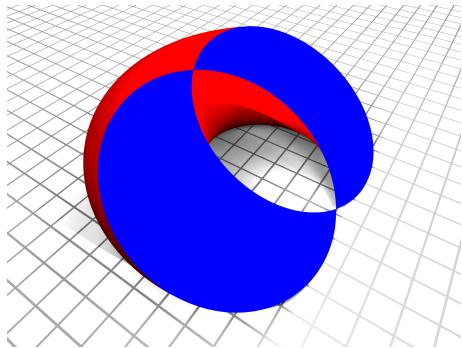


FIGURE 8: Les deux lunules dans l'espace

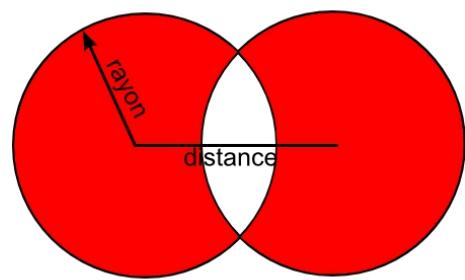


FIGURE 9: Les paramètres des lunules

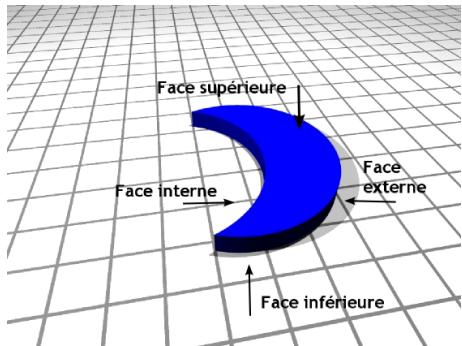


FIGURE 10: Gonfler la lunule.

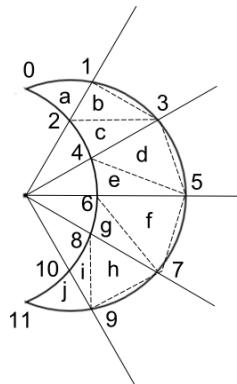


FIGURE 11: Décomposer une face de lunules en triangles élémentaires.

7.2.2 Décomposer une lunule (gonflée) en triangles élémentaires.

Considérons le centre de symétrie de l'image de la figure 9. A partir de ce centre, on trace des droites tournant d'un angle constant. Chacune de ces droites croise la lunule en deux points. Les coordonnées de ces points sont faciles à calculer (vous pouvez retrouver le calcul dans le code du programme...). Sur l'exemple de la figure 11, on utilise 5 droites, qui fournissent 10 points (plus deux points pour les cornes de la lunule), correspondant à 10 triangles élémentaires. Les points de 12 à 23, situés verticalement sous les points 0 à 11, définissent les triangles de la face inférieure. Pour les faces latérales, on ne crée pas de nouveaux points, on utilise les points déjà créés pour définir les nouveaux triangles. Toutes ces manipulations, un peu laborieuses peuvent se lire facilement dans le code de la fonction créant une lunule. Evidemment, le résultat sera d'autant plus précis que le nombre de droites utilisées sera grand (au moins une centaine). Cependant, il n'est pas nécessaire de demander trop de précision : votre œil ne discernera plus la différence en dessous d'une certaine échelle.

7.2.3 De la lunule au tore

Une fois définie la lunule, le reste du travail est simple :

- Créer une lunule dans le plan horizontal, dans la première couleur (figure 12).
- Créer une deuxième lunule de l'autre couleur, et la faire tourner de 180 degrés par rapport à l'axe vertical (figure 13).
- Lier les deux lunules, afin que Blender les considère comme n'étant qu'un seul objet.
- Incliner le couple de lunules de l'angle choisi (figure 14) .
- Le faire tourner de $\frac{360 \times i}{n}$ degrés autour de l'axe vertical.
- Créer la lunule suivante, après avoir augmenté i de 1 (figure 15 à 17).

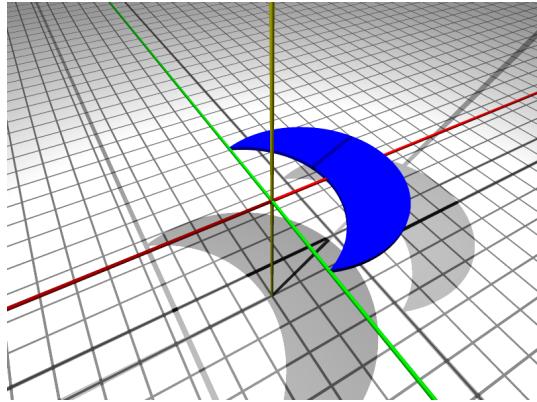


FIGURE 12: Créer une lunule.

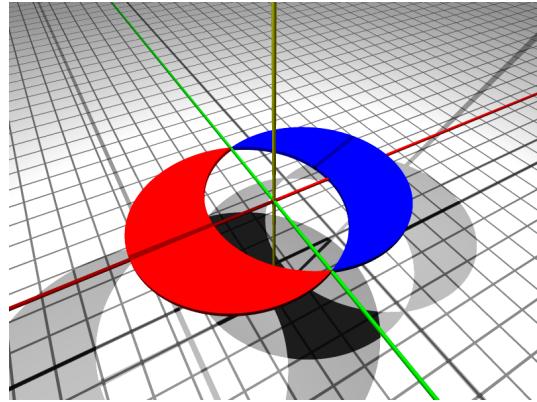


FIGURE 13: Créer la lunule symétrique.

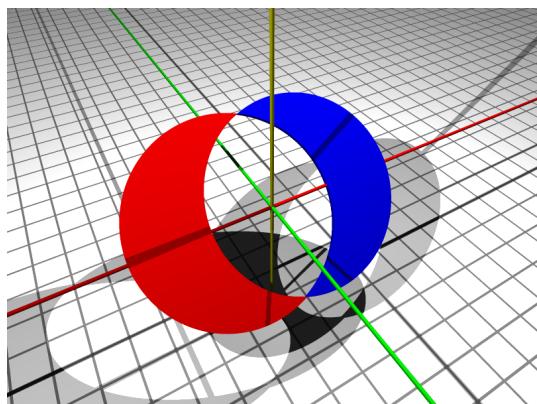


FIGURE 14: Incliner le couple de lunules.

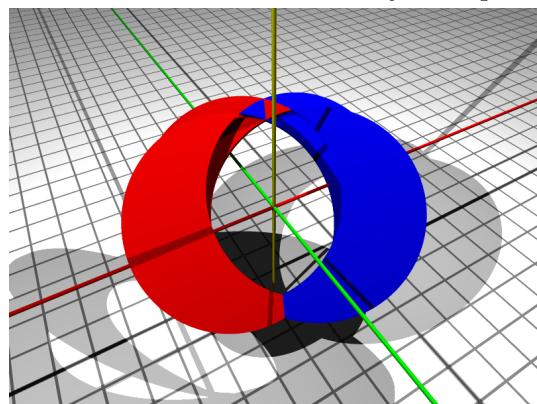


FIGURE 15: Ajouter un nouveau couple.

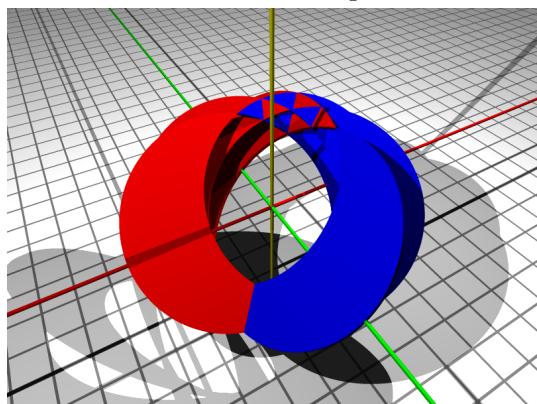


FIGURE 16: Encore un.

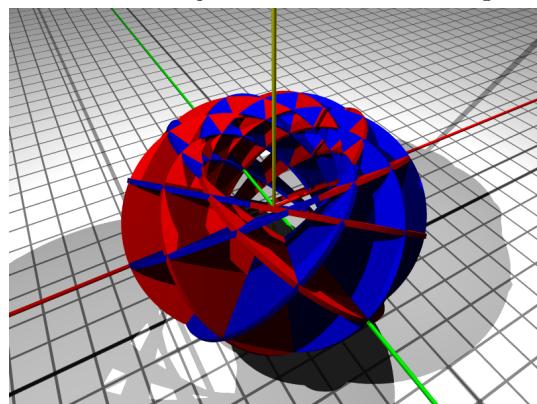


FIGURE 17: Le tore terminé.