

Pierre Bézier



The Mathematical Basis of the
UNISURF CAD
System

LIBRARY COPY
DO NOT REMOVE



BUTTERWORTHS

Butterworths



for free3d.com



Les Courbes de Bézier

1. Introduction

Au début des années 60, les machines à commande numérique ne savaient usiner de façon précise que des courbes simples comme des droites, des paraboles ou des ellipses.

Une seconde catégorie d'objets, au contraire, offrait une forme a priori peu précise, déterminée expérimentalement : Les hélices d'avions, les coques de bateaux et les carrosseries de voitures étaient tracées à main levée, sans que l'on puisse décrire leurs formes par une formule mathématique.

Les machines à commande numérique de cette époque offrant une programmation limitée, on savait les alimenter avec des nombres pour définir des déplacements élémentaires comme des droites, des arcs de cercle, et à la rigueur des ellipses. Mais il n'était pas question de programmer des courbes quelconques, tracées à la main, faute d'une définition numérique (mathématique) de celles-ci.

Pierre Bézier (1910-1999), ingénieur français employé chez Renault, chercha donc comment traduire mathématiquement une courbe, puis une surface, dessinées à main levée. Il lui fallait concevoir un système capable de gérer des courbes gauches, c'est-à-dire de manipuler des surfaces en 3D, d'où la nécessité de définir un modèle mathématique qui ne soit pas limité à des courbes en deux dimensions.

Sa préoccupation était d'offrir au dessinateur un moyen simple et puissant pour créer des formes et pour faciliter la programmation des machines à commande numérique. Afin d'utiliser concrètement les courbes et surfaces de Bézier, ces courbes furent introduites à partir de 1962 dans un logiciel développé par Renault et nommé Unisurf. Il est à la base de tous les logiciels créés par la suite, dont Catia. Les concepts de CAO et de CFAO venaient ainsi de prendre forme.

Cependant, les recherches de Bézier n'étaient pas entièrement originales. Dès 1958, un mathématicien employé par Citroën, Paul De Casteljau, s'était attaqué au même problème. Paul De Casteljau avait été chargé de numériser une courbe, une fois celle-ci tracée, sans se poser la question d'une correction a posteriori. Il définissait ses courbes comme caractérisées par des pôles, d'une façon nettement moins parlante que les points de contrôle de Bézier. Ces travaux, publiés comme des rapports techniques tenus très au secret par Citroën restèrent inconnus jusqu'en 1975 quand W. Böhm en a pris connaissance et les a rendus publiques. L'algorithme de De Casteljau a été très utile pour l'informatique qui utilise les courbes de

Bézier dans de nombreux cas (logiciels de dessin, de modélisation, ...), et sans lequel le développement de l'utilisation des courbes de Pierre Bézier n'aurait pas pu se faire.

Plus tard, un problème s'est posé à Apple : trouver un moyen de définir de façon mathématique une courbe, comme le tracé d'un caractère, avant de l'envoyer à l'imprimante laser conçue pour le micro-ordinateur MacIntosh apparu en 1984. John Warnock, co-fondateur en 1982 avec Charles Geschke de la société Adobe Systems Inc, connaissait ces travaux et il utilisa les courbes de Bézier comme base du langage PostScript qui fut choisi par Apple pour son imprimante laser.

Microsoft adopta à son tour les polices true-type (format de fontes vectoriel également basé sur les courbes de Bézier et créé par Apple) à partir de Windows 3.1.

Les **courbes de Bézier** constituent une modélisation qui permet, à l'époque, de concevoir les formes des différents éléments de carrosserie d'une voiture de manière plus interactive. Elles furent donc inventées pour répondre à un besoin technique.

Par la suite, après quelques publications scientifiques, les **courbes de Bézier** trouvèrent une place incontournable dans la plupart des logiciels de Conception Assistée par Ordinateur (CAO) et de Dessin Assisté par Ordinateur (DAO) et dans la commande numérique de machines.

A l'heure actuelle les **courbes de Bézier** sont non seulement encore utilisées dans l'industrie automobile pour concevoir les formes des voitures de demain mais elles sont présentes dans tous les domaines du design, de l'infographie et de la conception. Elles servent à représenter des objets aux formes complexes, méthode parfois préférée par rapport à un simple échantillonnage de l'objet.

On les retrouve en architecture, en mécanique, dans l'industrie aéronautique, dans les polices de caractères True-type, etc.

Un autre exemple de l'utilisation des **courbes de Bézier** est leur application dans une industrie un peu particulière qui est la conception de sous-marins nucléaires. Le but premier de tels engins étant de pouvoir scruter les fonds marins sans se faire repérer, il faut pour cela minimiser le bruit de l'appareil produit lors de ses déplacements.

Il s'agit ainsi d'un problème physique d'écoulement des fluides. Les **courbes de Bézier**, ou plutôt les **surfaces de Bézier** sont ici intéressantes car étant de classe C^∞ (ce sont des courbes infiniment dérivables), elles ne présentent pas de cassures ce qui conduit à créer un sous-marin au contour aérodynamique permettant un écoulement plus facile de l'eau sans trop de turbulences et donc de bruit.

Dans le domaine de la conception (design), les **courbes de Bézier** sont utilisées non seulement en retouche et synthèse d'images mais aussi en morphing (le morphing consiste à déformer des images à partir d'un tramage de départ).

Parmi les logiciels les plus connus faisant appels aux **courbes de Bézier** pour de telles applications on peut citer entre autres Paint, Photoshop, Blender ou encore The Gimp.

Les **courbes de Bézier** sont des courbes polynomiales paramétriques. Elles ont donné naissance à de nombreux outils mathématiques, tels que les NURBS (Non-Uniform Rational B-Spline).

Avant que les **courbes de Bézier** ne soient inventées, on utilisait d'autres courbes d'ajustement (utilisées dans le lissage des courbes expérimentales) appelées "splines". Le problème rencontré avec les splines c'est qu'elles changent d'aspect lorsqu'on effectue une rotation dans le repère. C'est pourquoi elles sont inutilisables en CAO.

Les **courbes de Bézier** ne présentent pas ce défaut. Pour effectuer la rotation d'une **courbe de Bézier** il suffit d'appliquer la rotation aux points qui la définissent et de tracer la nouvelle courbe. Les calculs pour une rotation sont donc peu nombreux et par conséquent très rapides. Il existe aujourd'hui des splines conformes aux principes de **Bézier**, elles sont nommées B-splines.

2. Courbes de Bézier

La définition classique des courbes de Bézier s'appuie sur les polynômes de Bernstein, utilisés couramment en mathématique pour l'approximation polynômiale des fonctions. Concernant les courbes de Bézier, les polynômes de Bernstein sont utilisés pour calculer les points de la courbe à tracer.

2.1. Définition du polynôme de Bernstein

Soit n appartenant à $\mathbb{N}^*$ (ensemble des nombres entiers non nuls).

Pour tout i variant de 0 à n ($i \in [0, n]$), le polynôme de Bernstein de degré n et d'indice i , noté B_n^i , est défini par la formule

$$B_n^i(t) = C_n^i \cdot t^i \cdot (1 - t)^{n-i} \quad (1)$$