# On Splines and Their Use in Computer Graphics and Generating Curves

Simon Abrelat

002129–0004

Math HL IA

May 2019

Words:

## Abstract

Splines are used everywhere around us, from fonts to animations. They are the way that computers deal with curves and because of that have a large range of possible uses. Since they are so useful, there are also a lot of ways to generate these functions and lots of formats they come in. In general, they are fairly computationally cheap which makes them so useful and can be extended in a multitude of ways. In this paper, the goal would be to use splines to smoothly interpolate values either for control or for path generation.

# Contents

# 1 Introduction

Splines are not a specific equation, they are more a class of equations that have similar properties. They are simply piecewise polynomials, and are often parameterized so that they could “loop over themselves” and break the vertical line test. One of the interesting aspects of splines is that they keep this property even at low degrees. The word spline is derived from wooden splines that curve given some constraints and are often used in things like braces for ships. The major benefit of splines is how they are suited for computers. Almost all curves on a computer from fonts to animation paths are described quickly and accurately by Bèzier curves and other such splines. This means that these splines are around us all of the time and are almost invisible if you do not know where to look. There are many different kinds of splines with different methods of formulation. However, this paper will mostly be focused on Hermite and Bèzier curves.

# 2 Bèzier Curves

## 2.1 Theory

Bèzier curves can be made with a linear combination of Bernstein polynomials. Bernstein polynomials were first used in the proof of the Stone-Weierstrass theorem which states that every continuous function defined on a closed interval $[a, b]$ can be approximated as closely as desired by a polynomial (Pinkus, 2000). A Bernstein polynomial (2) is defined by a linear combination of Bernstein basis functions (1).

Bernstein Polynomials:

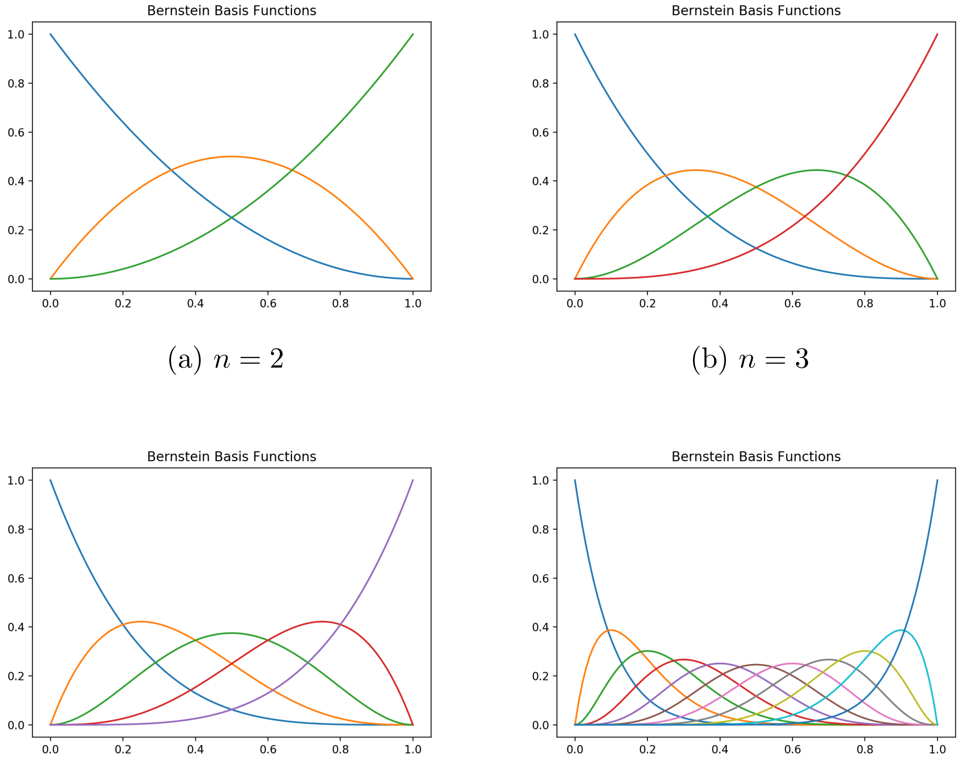
$$b_{\nu,n} = \binom{n}{\nu} x^{\nu}(1-x)^{\nu-n} \tag{1}$$

$$B_n(x) = \sum_{\nu=0}^{n} \beta_{\nu} b_{\nu,n}(x) \tag{2}$$

$\nu$: is the predicted rate for user $x$ on item $i$

$n$: is the rate of song $i$ given by user $k$

Figure 1: Basis functions for different degrees



(a) $n = 2$

(b) $n = 3$

(a) $n = 4$

(b) $n = 10$

## 2.2 Application

# References

Pinkus, A. (2000). Weierstrass and approximation theory. *Journal of Approximation Theory*, 107:1–66.