University of West Bohemia
Faculty of Applied Sciences
Department of Mathematics

Fractals and Splines

Diploma Thesis

Plzeň, 2012 **Lenka Ptáčková**

I would like to thank my supervisor Dr Bohumír Bastl for his constant support and guidance, in particular, through the theory concerning splines. I would also like to express my sincere gratitude to my specialist consultant Professor Franco Vivaldi for his devoted approach and guidance. Without the insight of Dr Bastl and Professor Vivaldi, my understanding would never have reached its present level.

I do hereby declare that this Diploma Thesis is the result of my own work and that all external sources of information have been duly acknowledged.

. . . . . .

Lenka Ptáčková

Title: **Fractals and Splines**
Author: Lenka Ptáčková

Supervisor: Dr Bohumír Bastl
Department of Mathematics, Faculty of Applied Sciences
University of West Bohemia

Specialist consultant: Professor Franco Vivaldi
School of Mathematical Sciences
Queen Mary, University of London

Abstract: This diploma thesis deals with connections between subdivision algorithms of geometric modeling and iterated function systems (IFS) of fractal theory. These connections have recently appeared in the literature [13, 15]. We introduce IFS for subdivision curves, providing rigorous justification of the main constructions. We prove in a basic case that the subdivision algorithm for Bézier curves leads, under suitable scaling, to the Takagi fractal curve, and we argue that this property holds in general.

Key words: Fractal, spline, de Casteljau algorithm, Bézier curves, Bernstein polynomials, subdivision, IFS, dynamical systems, Takagi curve.

Název práce: **Fraktály a spline objekty**
Autor: Lenka Ptáčková

Vedoucí práce: Ing. Bohumír Bastl, Ph.D.
Kadedra matematiky, Fakulta aplikovaných věd
Západočeská univerzita

Konzultant: Prof. Franco Vivaldi
School of Mathematical Sciences
Queen Mary, University of London

Abstrakt: Tato diplomová práce se zabývá souvislostmi mezi subdivision algoritmy geometrického modelování a iteračním systémem funkcí (IFS) v teorii fraktálů. Tyto souvislosti se v současnosti objevily v literatuře [13, 15]. Představujeme zde IFS pro generování subdivision křivek a rigorózně zdůvodňujeme korektnost hlavních konstrukcí. Je zde dokázáno, že de Casteljau algoritmus pro Bézierovy křivky konverguje, za určité změny měřítka, k Takagi fraktální křivce.

Klíčová slova: Fraktál, spline, algoritmus de Casteljau, Bézierovy křivky, Bernsteinovy polynomy, subdivision, IFS, dynamické systémy, Takagi křivka.

Contents

Introduction

The video [14] by R. Goldman was the inspiration which made me decide for fractals and splines as a topic of my thesis. R. Goldman provides a brief introduction to splines and fractals, and presents some surprising connections between these two, seemingly unrelated geometric objects. Fractals are attractors of iterated function systems (IFS), splines are attractors of iterated subdivision process. Spline algorithms can be used to generate fractals with control points, spline curves can be generated by IFS.

The main aim of the thesis is to bridge the gap between the spline and fractal theory. The intention is to provide a treatment of the mathematics associated with fractals and splines at a level which is accessible to those who are not yet familiar with either of the subject. To the best of my knowledge, such a material is not available in the literature. The thesis investigates a wide variety of mathematical ideas that are related to fractals and splines. The theory is built systematically from minimal pre-requisites, in order to introduce more complex problems later. We also present the results of some original research.

In Chapter 1, we begin with spline theory, more precisely, with Bézier and B-spline curves, which are widely used in Computer Aided Design and Manufacturing (CAD and CAM), Geometric Modeling, Computer Graphics, etc. Bézier curves are, for example, implemented in Adobe Illustrator or AutoCAD software. We present examples of these curves and several methods for their construction, such as the de Casteljau algorithm for Bézier curves.

In recent years, the subdivision curves and surfaces have become very popular in computer graphics. We provide a brief introduction to subdivision curves in Chapter 2. The link between fractals and splines goes through subdivision, therefore the subdivision process is treated in detail. Generally, subdivision curve refinement schemes can be classified into two categories: interpolating and approximating. Interpolating curves go through the original input points, approximating curves are not necessarily incident with the starting set of points. The shape and smoothness of the resulting curve (or surface) depends on the chosen rules. Some of the properties we expect from such rules are:

- Efficiency: the new points should be computed with a small number of operations.

- Compact support: the region over which a point influences the shape of the final curve or surface should be small and finite.
- Local definition: the rules used to determine the location of new points should not depend on very distant points of the original polygon.
- Simplicity: the rules determining the new polygon should be simple and there should only be a small number of rules.
- Continuity: it should be possible to determine the continuity of the limit curve from the refinement rules.

There are many interesting connections between subdivision curves and fractal curves, more can be found also in Section 3.6.

Chapter 3 deals with iterated function systems and fractals. Before defining fractals, we introduce some basic dynamical systems terminology, metric spaces, and the notion of dimension of a set. Our main focus is on fractals generated by IFS consisting of affine transformations. We later introduce IFS for subdivision curves and prove that such an IFS has unique fixed point, see Section 3.6.

We present an IFS for B-spline curves and complex Bézier curves. Resorting to complex domain shows up to be very beneficial, since we can then generate well known fractals by the de Casteljau subdivision algorithm with complex parameter. Such complex Bézier curves seem to be related to many interesting and varied mathematical problems, but in the limited space of a diploma thesis, it is hardly possible to investigate all the directions. However, we provide a proof that the subdivision algorithm for Bézier curves leads, under suitable scaling, to the Takagi fractal curve (see Theorem 3.7.3).

Finally, in Chapter 4 we present concluding remarks. We also refer the interested reader in other sources of literature concerning our topic, which is reviewed in Appendix A.

Chapter 1

Splines

A spline is usually defined to be a sufficiently smooth piecewise-polynomial parametric curve. Even though there exist also exponential and other types of splines, in this work we will deal only with polynomial splines. More about exponential splines can be found in [24].

Only a small part of splines theory is presented; our main focus is on Bézier and B-spline curves.

In computer graphics, splines are used for free form modeling, since they are easy to construct and allow us to design and control the shape of complex curves and surfaces.

1.1 Bézier Curves

1.1.1 Bernstein polynomials

Bézier curves were widely popularized in 1962 by the French engineer Pierre Bézier, who worked for Renault. In 1970 R. Forrest showed the connection between the work of P. Bézier and the theory of Bernstein polynomials. Bernstein polynomials were first used by the Russian mathematician S. N. Bernstein in a constructive proof for the Stone–Weierstrass approximation theorem [19].

Definition 1.1.1. The $n+1$ polynomials defined by

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, i = 0, \ldots, n, \tag{1.1}$$

where $\binom{n}{i}$ is a binomial coefficient, are called the *Bernstein basis polynomials* of degree n.

They have the following properties:

- The standard polynomial basis $1, t, \ldots, t^n$ can be mapped to Bernstein polynomials $B_0^n, \ldots, B_n^n$ via a linear map:

$$\begin{pmatrix} B_0^n(t) \\ B_1^n(t) \\ \vdots \\ B_{n-1}^n(t) \\ B_n^n(t) \end{pmatrix} = \begin{pmatrix} 1 & -n & \ldots & n(-1)^{n-1} & (-1)^n \\ 0 & n & \ldots & (n^2-n)(-1)^{n-2} & n(-1)^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & n & -n \\ 0 & 0 & \ldots & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ t \\ \vdots \\ t^{n-1} \\ t^n \end{pmatrix}.$$

  We can see that the linear map is represented by an upper triangular matrix with non-zero diagonal elements, hence the map is invertible. Therefore, Bernstein polynomials form a basis for the vector space of polynomials of degree n.

- Non-negativity:

$$B_i^n(t) \geq 0, \quad t \in [0,1],\ i = 0, \ldots n. \tag{1.2}$$

- Recursive formula:

$$B_i^n(t) = (1-t)B_i^{n-1}(t) + tB_{i-1}^{n-1}(t), \quad n > 1,\ 0 < i < n, \tag{1.3}$$

  where $B_0^q(t) = (1-t)^q$ and $B_q^q(t) = t^q$ for $q = 1, \ldots n$. The formula follows from the recursive formula for binomial coefficients

$$\binom{n}{i} = \binom{n-1}{i-1} + \binom{n-1}{i}, \quad i, n > 0.$$

- Partition of unity:

$$\sum_{i=0}^{n} B_i^n(t) = \sum_{i=0}^{n} \binom{n}{i} t^i (1-t)^{n-i} = (t+1-t)^n = 1, \tag{1.4}$$

  which follows from the Binomial theorem.

- Symmetry:

$$B_i^n(t) = B_{n-i}^n(1-t),$$

  which follows from the symmetry of binomial coefficients $\binom{n}{i} = \binom{n}{n-i}$.

1.1.2 Bézier curves

Definition 1.1.2. A *Bézier curve* $\mathbf{c}(t)$ of degree n is defined by

$$\mathbf{c}(t) = \sum_{i=0}^{n} B_i^n(t)\mathbf{p}_i, \quad t \in [0,1], \tag{1.5}$$

where $\mathbf{p}_0, \ldots, \mathbf{p}_n$ are *control points* and $B_i^n(t)$ are the Bernstein basis polynomials of degree n. A polygon with vertices $\mathbf{p}_0, \ldots, \mathbf{p}_n$ is called the *control polygon*.

Example 1.1.1. A Bézier curve of degree 3 for the control points $\mathbf{p}_0 = [0,0], \mathbf{p}_1 = [\frac{1}{2}, 1], \mathbf{p}_2 = [1,1], \mathbf{p}_3 = [1,0]$ has the the parametrization as follows:

$$\mathbf{c}(t) = \left(\frac{3}{2}t - \frac{1}{2}t^3, 3t - 3t^2\right), \ t \in [0,1].$$

The curve is shown in Figure 1.1.

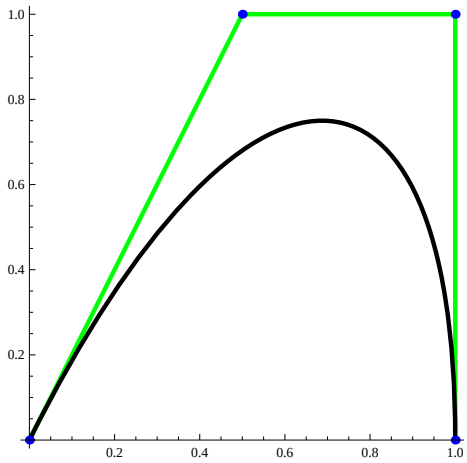

Figure 1.1: A Bézier curve of degree 3. The Bézier curve is black, its control polygon is green and its control points are blue.

Properties of Bézier curves

Proposition 1.1.1. *A Bézier curve interpolates the first and last control points of its control polygon.*

Proof. Since

$$\mathbf{c}(t) = (1-t)^n \mathbf{p}_0 + \sum_{i=1}^{n-1} \binom{n}{i} t^i (1-t)^{n-i}(t)\mathbf{p}_i + t^n \mathbf{p}_n, \ t \in [0,1],$$

it directly follows

$$\begin{array}{rcl} \mathbf{c}(0) & = & \sum_{i=0}^{n} B_i^n(0)\mathbf{p}_i = \mathbf{p}_0, \\ \mathbf{c}(1) & = & \sum_{i=0}^{n} B_i^n(1)\mathbf{p}_i = \mathbf{p}_n. \end{array} \tag{1.6}$$

□

Proposition 1.1.2. *A Bézier curve is tangent to its first and last control polygon legs. Moreover, for the first derivative of a Bézier curve with parametrization (1.5) it holds that*

$$\mathbf{c}'(0) = n(\mathbf{p}_1 - \mathbf{p}_0), \quad \mathbf{c}'(1) = n(\mathbf{p}_n - \mathbf{p}_{n-1}).$$

Proof. The first derivatives of the basis functions are

$$\left(B_i^n\right)'(t) \;=\; \tbinom{n}{i}[it^{i-1}(1-t)^{n-i} - (n-i)t^i(1-t)^{n-i-1}], \quad i = 1, \ldots, n-1.$$

Therefore,

$$\begin{aligned}\mathbf{c}'(t) = \; & n\Big[(1-t)^{n-1}(\mathbf{p}_1 - \mathbf{p}_0) - (n-1)t(1-t)^{n-2}\mathbf{p}_1 \\ & + \textstyle\sum_{i=2}^{n-2} \binom{n-1}{i}[it^{i-1}(1-t)^{n-i} - (n-i)t^i(1-t)^{n-i-1}]\mathbf{p}_i \\ & + (n-1)t^{n-2}(1-t)\mathbf{p}_{n-1} + t^{n-1}(\mathbf{p}_n - \mathbf{p}_{n-1})\Big],\end{aligned}$$

which yields

$$\mathbf{c}'(t) = \sum_{i=0}^{n-1} B_i^{n-1}(t) n(\mathbf{p}_{i+1} - \mathbf{p}_i). \tag{1.7}$$

Thus the first derivative[1] of a Bézier curve is again a Bézier curve. Substituting value 0 and 1 into the equation (1.7) completes the proof. □

Let $\mathbf{c}(t)$ and $\mathbf{d}(t)$ be two Bézier curves defined by equation (1.5). Let the first curve $\mathbf{c}(t)$ be given by $m+1$ control points $\mathbf{p}_0, \ldots, \mathbf{p}_m$ and the second curve $\mathbf{d}(t)$ be given by $n+1$ control points $\mathbf{q}_0, \ldots, \mathbf{q}_n$. We can join the two curves together with C^0 continuity by letting the last point of the first curve be incident with the first point of the second curve. Recalling Proposition 1.1.1 we have $\mathbf{c}(1) = \mathbf{d}(0) \Leftrightarrow \mathbf{p}_m = \mathbf{q}_0$. Further, provided that the tangent vector of the first curve at its last point is identical to the tangent vector of the second curve at its first point, i.e. $\mathbf{c}'(1) = \mathbf{d}'(0)$, the curves are joined together with C^1 continuity. Recalling Proposition 1.1.2 we obtain equivalent condition on control points $\mathbf{c}'(1) = \mathbf{d}'(0) \Leftrightarrow m(\mathbf{p}_m - \mathbf{p}_{m-1}) = n(\mathbf{q}_1 - \mathbf{q}_0)$.

Definition. The *convex hull* $H(\mathbf{P})$ of a set of points $\mathbf{P}$ is the intersection of all convex sets containing $\mathbf{P}$, i.e., it is given by the expression

$$H(\mathbf{P}) = \Big\{ \sum_{i=0}^{n} \mathbf{p}_i \alpha_i \mid \mathbf{p}_i \in \mathbf{P}, \alpha_i \geq 0, \sum_{i=0}^{n} \alpha_i = 1, i = 0, \ldots, n \Big\}. \tag{1.8}$$

Proposition 1.1.3. *A Bézier curve lies in the convex hull of its control points.*

Proof. We can substitute $B_i^n(t), t \in [0,1]$ for α_i in the equation (1.8), since Bernstein polynomials are also non-negative (1.2) and create partition of unity (1.4). Further, letting $\mathbf{P} = \{\mathbf{p}_0, \ldots, \mathbf{p}_n\}$ be Bézier control points we obtain

$$\Big\{ \sum_{i=0}^{n} \mathbf{p}_i B_i^n(t), t \in [0,1] \,|\, \mathbf{p}_i \in \mathbf{P} \Big\} \subseteq \Big\{ \sum_{i=0}^{n} \mathbf{p}_i \alpha_i \,|\, \mathbf{p}_i \in \mathbf{P}, \alpha_i \in \mathbb{R}^+, \sum_{i=0}^{n} \alpha_i = 1 \Big\} = H(\mathbf{P}).$$

Therefore, a Bézier curve lies in the convex hull of its control points. □

[1] The first derivative of a parametric curve is usually called hodograph in the context of geometric modeling.

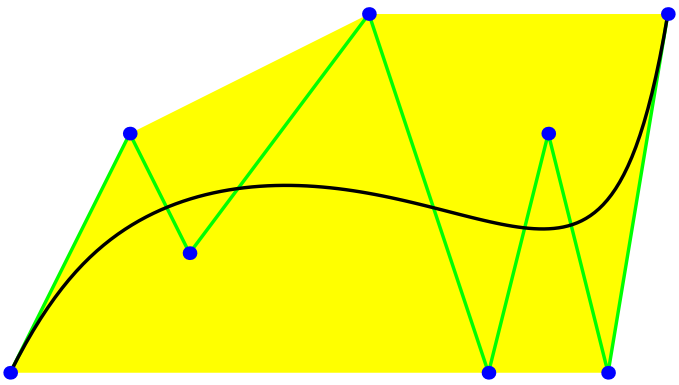

Figure 1.2: Convex hull of a Bézier curve

Figure 1.2 shows the convex hull of control points of a Bézier curve. According to [10], the importance of the convex hull property lies in what is known as *interference checking*. Suppose we want to know if two Bézier curves intersect each other. If the two convex hulls do not overlap, we are assured that the two curves do not intersect.

Proposition 1.1.4. *A Bézier curve is variation diminishing, i.e., the number of intersections of a straight line with a Bézier curve is no greater than the number of intersections of the line with its control polygon.*

Proof can be found in [10, Section 6.3].

This property guarantees that a Bézier curve oscillates less than its control polygon.

1.1.3 The de Casteljau algorithm

Points on a Bézier curve can be constructed by the de Casteljau algorithm, which is an alternative way that was introduced by Paul de Casteljau already in 1959. He worked for Citroën and his work was kept a secret by Citroën for a long time [11]. P. de Casteljau and P. Bézier had different approaches and they introduced Bézier curves independently, but P. Bézier could publish his work first.

According to [10], the de Casteljau algorithm is based on a generalization of the construction of a parabola by repeated linear interpolation for curves of higher degree. We deal with this case first.

Let $\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2$ be any three non-collinear points in $\mathbb{R}^3$, let $t \in \mathbb{R}$. Define

$$\begin{aligned} \mathbf{p}_0^1(t) &= (1-t)\mathbf{p}_0 + t\mathbf{p}_1, \\ \mathbf{p}_1^1(t) &= (1-t)\mathbf{p}_1 + t\mathbf{p}_2, \\ \mathbf{p}_0^2(t) &= (1-t)\mathbf{p}_0^1(t) + t\mathbf{p}_1^1(t). \end{aligned} \tag{1.9}$$

Inserting the first two equations into the third one, we obtain

$$\mathbf{p}_0^2(t) = (1-t)^2\mathbf{p}_0 + 2t(1-t)\mathbf{p}_1 + t^2\mathbf{p}_2. \tag{1.10}$$

Equation (1.10) is a quadratic expression in t and $\mathbf{p}_0^2(t)$ traces a parabola as t varies from $-\infty$ to $+\infty$. This construction consists of repeated linear interpolation and its geometry is illustrated in Figure 1.3. For $t \in [0,1]$, $\mathbf{c}(t)$ lies inside the triangle formed by $\mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2$.

Let us denote ratio$(\mathbf{a}, \mathbf{b}, \mathbf{c})$ the ratio of three collinear points $\mathbf{a}, \mathbf{b}, \mathbf{c}$, that is

$$\text{ratio}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \frac{\mathbf{a} - \mathbf{b}}{\mathbf{c} - \mathbf{b}}.$$

The ratios of points in 1.9 are

$$\text{ratio}(\mathbf{p}_0, \mathbf{p}_0^1, \mathbf{p}_1) = \text{ratio}(\mathbf{p}_1, \mathbf{p}_1^1, \mathbf{p}_2) = \text{ratio}(\mathbf{p}_0^1, \mathbf{p}_0^2, \mathbf{p}_1^1) = t/(1-t).$$

From analytic geometry we know the following theorem, which proves that our construction is correct. This theorem describes a property of parabolas, and the de Casteljau algorithm can be viewed as its constructive counterpart, see also [10, Section 4.1].

Theorem 1.1.1 (Three tangent theorem)**.** *Let* $\mathbf{a}, \mathbf{b}, \mathbf{c}$ *be three distinct points on a parabola. Let the tangent at* $\mathbf{b}$ *intersect the tangents at* $\mathbf{a}$ *and* $\mathbf{c}$ *in* $\mathbf{e}$ *and* $\mathbf{f}$*, respectively. Let the tangents at* $\mathbf{a}$ *and* $\mathbf{c}$ *intersect in* $\mathbf{d}$*. Then* $\text{ratio}(\mathbf{a}, \mathbf{e}, \mathbf{d}) = \text{ratio}(\mathbf{e}, \mathbf{b}, \mathbf{f}) = \text{ratio}(\mathbf{d}, \mathbf{f}, \mathbf{c})$.

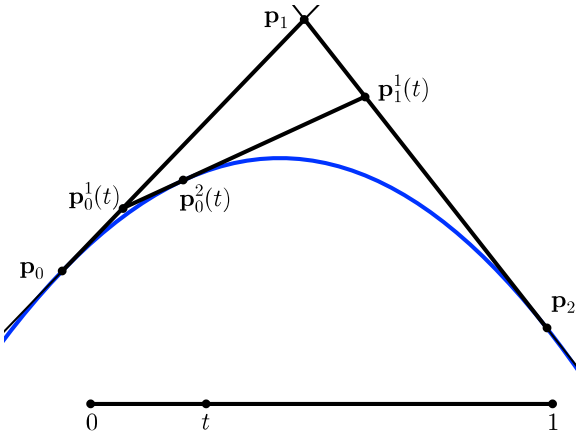

Figure 1.3: Construction of a parabola by repeated linear interpolation

The construction of a parabola can be generalized to generate a polynomial curve of arbitrary degree n. This gives rise to the *de Casteljau algorithm*:
Let $\mathbf{p}_0, \ldots, \mathbf{p}_n \in \mathbb{R}^3$ be given set of control points and $t \in [0,1]$, set

$$\begin{cases} \mathbf{p}_i^0(t) &= \mathbf{p}_i \\ \mathbf{p}_i^k(t) &= (1-t)\mathbf{p}_i^{k-1}(t) + t\mathbf{p}_{i+1}^{k-1}(t) \end{cases} \quad \begin{cases} k &= 1, \ldots, n \\ i &= 0, \ldots, n-k \end{cases} \tag{1.11}$$

Then $\mathbf{p}_0^n(t)$ is the point with parameter value t on the Bézier curve of degree n given by the control points $\mathbf{p}_0, \ldots, \mathbf{p}_n$.

The coefficients $\mathbf{p}_i^k(t)$ can be arranged into a schematic triangular array of points, called the de Casteljau scheme. The example of the cubic case is:

$$\begin{array}{llll} \mathbf{p}_0 & & & \\ \mathbf{p}_1 & \mathbf{p}_0^1 & & \\ \mathbf{p}_2 & \mathbf{p}_1^1 & \mathbf{p}_0^2 & \\ \mathbf{p}_3 & \mathbf{p}_2^1 & \mathbf{p}_1^2 & \mathbf{p}_0^3. \end{array}$$

1.2 B-spline curves

B-spline curves were originally investigated by N. Lobachevsky in the nineteenth century. They were constructed as convolutions of certain probability distributions. The modern theory of spline approximation started with a paper of I. J. Schoenberg in 1956; more about the history of B-spline curves and surfaces can be found in [10].

B-spline curves can be viewed as a generalization of Bézier curves because they consist of a sequence of polynomial curve segments, i.e., they consist of a sequence of Bézier curves.

1.2.1 B-spline basis functions

Definition 1.2.1. Let $\mathbf{T} = (t_0, t_1, \ldots, t_m)$ be a nondecreasing sequence of real numbers called a *knot vector*. Let $\mathbf{p}_0, \ldots, \mathbf{p}_j$ be given control points, let n be the degree of basis functions such that

$$n = m - j - 1.$$

Then the *B-spline basis functions* N_i^n of degree n are defined as

$$\begin{aligned} N_i^0(t) &= \begin{cases} 1 & \text{if } t_i \le t < t_{i+1}, \\ 0 & \text{otherwise} \end{cases} \\ N_i^p(t) &= \frac{t-t_i}{t_{i+p}-t_i} N_i^{p-1}(t) + \frac{t_{i+p+1}-t}{t_{i+p+1}-t_{i+1}} N_{i+1}^{p-1}(t), \end{aligned} \tag{1.12}$$

where $p = 1, 2, \ldots, n$ and $i = 0, \ldots, m-p-1$.

This definition encompasses the de Boor algorithm for B-spline basis functions. Different approaches defining B-spline basis functions can be found in [24] or [10].

The recursion formula (1.12) shows that a B-spline basis function of degree n is a linear combination of two lower-degree basis functions, see also Example 1.2.1.

We list the following properties of B-spline basis functions without proving them, since the proofs are not straightforward and can be found in [11].

- Partition of unity:
$$\sum_{i=0}^{j} N_i^n(t) = 1 \quad t \in [t_n, t_{m-n}].$$
- Positivity: $\forall n \in \mathbb{N}: \; N_i^n(t) > 0$ on (t_i, t_{i+n}).
- Local support: $\text{supp} N_i^n(t) = (t_i, t_{i+n})$.
- Smoothness: B-spline basis functions of degree n are C^{n-1}– continuous if all the inner knots are distinct.
- Basis functions are just shifted copies of each other if the knots are all distinct and equidistantly distributed.

Example 1.2.1. Let $\mathbf{T} = (0, 1, 2, 3)$, then the basis functions are

$$N_0^0(t) = \begin{cases} 1 & t \in [0,1) \\ 0 & \text{otherwise} \end{cases} \qquad N_1^0(t) = \begin{cases} 1 & t \in [1,2) \\ 0 & \text{otherwise} \end{cases} \qquad N_2^0(t) = \begin{cases} 1 & t \in [2,3] \\ 0 & \text{otherwise.} \end{cases}$$

We can see that $N_i^0(t)$ for $i = 0, 1, 2$ are piecewise constant functions. If we substitute them into the formula (1.12) again, we obtain

$$\begin{aligned} N_0^1(t) &= tN_0^0(t) + (2-t)N_1^0(t), \\ N_1^1(t) &= (t-1)N_1^0(t) + (3-t)N_2^0(t), \\ N_0^2(t) &= \tfrac{t}{2}N_0^1(t) + \tfrac{3-t}{2}N_1^1(t). \end{aligned}$$

After evaluation, the equations become

$$N_0^1(t) = \begin{cases} t & 0 \le t < 1 \\ 2-t & 1 \le t < 2 \\ 0 & \text{otherwise} \end{cases} \qquad N_1^1(t) = \begin{cases} -1+t & 1 \le t < 2 \\ 3-t & 2 \le t \le 3 \\ 0 & \text{otherwise.} \end{cases}$$

As expected, the functions $N_0^1(t)$, $N_1^1(t)$ are piecewise linear. Further,

$$N_0^2(t) = \begin{cases} \frac{t^2}{2} & 0 \le t < 1 \\ -\frac{3}{2} + 3t - t^2 & 1 \le t < 2 \\ \frac{1}{2}(-3+t)^2 & 2 \le t \le 3 \\ 0 & \text{otherwise} \end{cases}.$$

The function $N_0^2(t)$ is piecewise quadratic. The functions $N_0^1(t)$, $N_1^1(t)$, and $N_0^2(t)$ are shown in Figure 1.4.

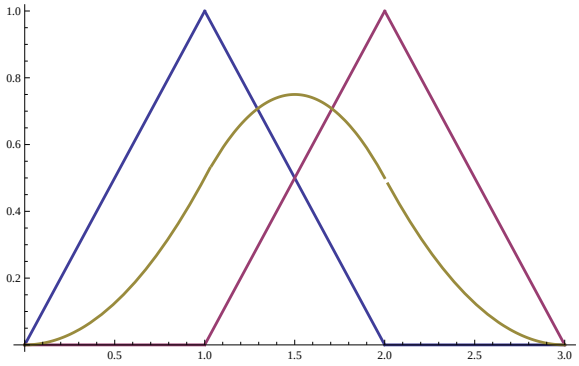

Figure 1.4: B-spline basis functions: N_0^1 (blue), N_1^1 (violet), and N_0^2 (ochroid).

Now, we explain the purpose of a knot vector $\mathbf{T} = (t_0, t_1, \ldots, t_m)$ from Definition 1.2.1. The knots subdivide the domain over which B-spline basis functions are defined. A basis function N_i^n is completely determined by the $n+2$ knots $t_i, \ldots, t_{i+n+1}$.

Therefore, the knot vector determines the properties of B-spline basis functions associated to it.

If a knot t_i appears r–times, i.e., $t_i = t_{i+1} = \cdots = t_{i+r-1}$, $r > 1$, then t_i is a *multiple* knot of multiplicity r. Otherwise it is a *simple* knot, see Examples 1.2.1 and 1.2.2. The basis function $N_i^k(t)$ is C^{k-r}– continuous at a knot of multiplicity r. Thus, increasing multiplicity decreases the level of continuity. On the other hand, increasing degree of basis functions increases continuity.

If the knot values are evenly spaced, the knot vector is called *uniform*, otherwise it is *non-uniform*. Further, if first and last knots have multiplicity $n+1$, the knot vector is called *non-periodic*. In contrary, if the first and last knots are simple, the knot vector is said to be *periodic*. See the table bellow with some examples of such knot vectors.

knot vector	periodic	non-periodic
uniform	$\mathbf{T} = (0, 1, 2, 3, 4)$	$\mathbf{T} = (0, 0, 0, \frac{1}{3}, \frac{2}{3}, 1, 1, 1)$
non-uniform	$\mathbf{T} = (0, 1, 5, 8, 15)$	$\mathbf{T} = (0, 0, 0, \frac{1}{5}, \frac{4}{5}, 1, 1, 1)$

Example 1.2.2. Let $\mathbf{T}_1 = (0, 1, 2, 3, 4, 5, 6, 7)$ be a uniform periodic knot vector and $\mathbf{T}_2 = (0, 0, 0, 1, 2, 3, 3, 3)$ be a uniform non-periodic knot vector. For both knot vectors, there are $(7-p-1)$ B-spline basis functions of degree p. The basis functions of degree 2 are shown in Figure 1.5.

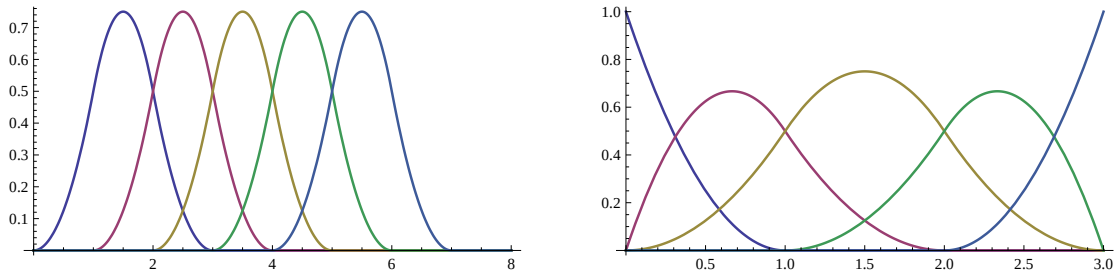

Figure 1.5: Basis functions of degree 2 for the knot vector $\mathbf{T}_1$ (left) and for the knot vector $\mathbf{T}_2$ (right).

1.2.2 B-spline curves

Definition 1.2.2. Let $\mathbf{T} = (t_0, t_1, \ldots, t_m)$ be a knot vector. Let $\mathbf{p}_0, \ldots, \mathbf{p}_j$ be given control points. Determine the degree of each curve segment as

$$n = m - j - 1.$$

Then the curve defined by

$$\mathbf{c}(t) = \sum_{i=0}^{j} \mathbf{p}_i N_i^n(t), \tag{1.13}$$

where $N_i^n(t)$ are B-spline basis functions of degree n, is a *B-spline curve* of degree n.

If the knot vector is not non-periodic, then the B-spline curve does not interpolate its first and last control points. Such a B-spline curve is called *floating*. Repeating the first and last knot $(n+1)$–times causes the curve to be tangent to the first and the last leg of its control polygon. Such curves are said *open B-spline curves*. A *periodic B-spline curve* of degree n is a B-spline curve which closes on itself. This requires that the first n control points are identical to the last n ones, and the lengths of the first n parameter intervals in the knot set are identical lengths of the last n intervals.

Example 1.2.3. Let $\mathbf{T} = (0, 0, 0, 1, 2, 3, 3, 3)$ be a uniform non-periodic knot vector as in Example 1.2.2. Let the control points be $\mathbf{p}_0 = [0, 0], \mathbf{p}_1 = [1, 4], \mathbf{p}_2 = [4, 4], \mathbf{p}_3 = [5, -1], \mathbf{p}_4 = [7, 0]$. The degree of each segment is $n = m - j - 1 = 7 - 4 - 1 = 2$. The B-spline curve for this knot vector and control points has parametrization

$$\mathbf{c}(t) = \sum_{i=0}^{4} \mathbf{p}_i N_i^2(t).$$

The curve is shown in Figure 1.6.

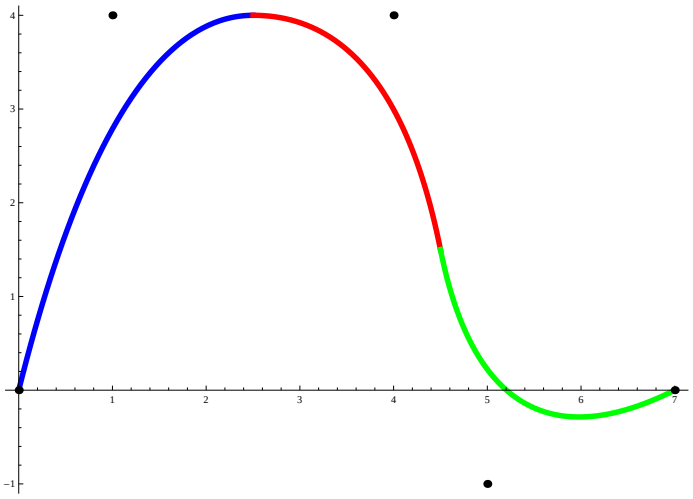

Figure 1.6: The B-spline from Example 1.2.3 is composed of three quadratic polynomial segments.

Properties of B-spline curves

- Convex hull property: B-spline curves lie in the convex hull of their control points [11].
- Variation diminution: A B-spline curve is variation diminishing, i.e., a B-spline curve has no more intersections with any line/plane than does its control polygon [11].
- Differentiation: Derivative of a B-spline curve is again a B-spline curve. Similarly as for Bézier curves, also for B-spline curves it holds

$$\mathbf{c}'(t) = \frac{\mathrm{d}}{\mathrm{d}t} \sum_{i=0}^{j} N_i^n(t)\mathbf{p}_i = \sum_{i=0}^{j-1} N_{i+1}^{n-1}(t) \frac{n}{t_{i+n+1} - t_{i+1}} (\mathbf{p}_{i+1} - \mathbf{p}_i), \tag{1.14}$$

  where n is the degree of the original B-spline curve. Therefore, the derivative of a B-spline curve is another B-spline curve of degree $n-1$ on the original knot vector with a new set of j control points $\mathbf{q}_0, \ldots, \mathbf{q}_{j-1}$, where $\mathbf{q}_i = \frac{n}{t_{i+n+1} - t_{i+1}} (\mathbf{p}_{i+1} - \mathbf{p}_i)$ [29].

Chapter 2

Subdivision curves

Subdivision curves are defined recursively. The goal of the subdivision process is to obtain effectively a limit curve that is smooth and has given properties.

The process starts with a given set of control points, a refinement scheme is applied to them, and new vertices are generated. New vertices create a new control polygon and the process is repeated.

Iteratively repeating this process infinitely many times yields the limit subdivision curve. But in geometric modeling, where the subdivision curves are mostly used, it is usually sufficient to approximate the curve within tolerance by its control polygon after several iterations.

There are many refinement schemes, we present subdivision schemes for Bézier and B-spline curves, and also four-point subdivision. The important part of this chapter is Section 2.1, where subdivision matrices are studied.

2.1 Binary subdivision schemes

Subdivision schemes for curves are defined by a set of rules that take in a set of control points $\mathbf{P}^k$ as input and produce a new, refined set of control points $\mathbf{P}^{k+1}$ as output. In this text, we consider binary subdivision schemes for curves which are defined by two sets of rules of the general form

$$\begin{aligned} \mathbf{p}_{2i}^{k+1} &= \textstyle\sum_j \alpha_j \mathbf{p}_{j+i}^k, \\ \mathbf{p}_{2i+1}^{k+1} &= \textstyle\sum_j \beta_j \mathbf{p}_{j+i}^k, \end{aligned}$$

where α_j and β_j are numerical coefficients and

$$\sum_j \alpha_j = \sum_j \beta_j = 1. \tag{2.1}$$

These binary subdivision rules can also be written in matrix form as

$$\mathbf{S} = \begin{pmatrix} \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \beta_0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 & \dots \\ \dots & \alpha_{-1} & \alpha_0 & \alpha_1 & \alpha_2 & \alpha_3 & \dots \\ \dots & \beta_{-1} & \beta_0 & \beta_1 & \beta_2 & \beta_3 & \dots \\ \dots & \alpha_{-2} & \alpha_{-1} & \alpha_0 & \alpha_1 & \alpha_2 & \dots \\ \dots & \beta_{-2} & \beta_{-1} & \beta_0 & \beta_1 & \beta_2 & \dots \\ \dots & \alpha_{-3} & \alpha_{-2} & \alpha_{-1} & \alpha_0 & \alpha_1 & \dots \\ \dots & \beta_{-3} & \beta_{-2} & \beta_{-1} & \beta_0 & \beta_1 & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{pmatrix}. \tag{2.2}$$

Then we can write

$$\mathbf{P}^k = \mathbf{S}\mathbf{P}^{k-1} = \mathbf{S}^k\mathbf{P}^0. \tag{2.3}$$

Denote $\lambda_0 \geq \lambda_1 \geq \dots \lambda_j$ the eigenvalues of the matrix $\mathbf{S}$. With respect to the value λ_0, we can think about these 3 cases:

- $\lambda_0 > 1 \Rightarrow \mathbf{P}^k \to \infty$ and the subdivision process does not converge,
- $\lambda_0 < 1 \Rightarrow |\lambda_i| < 1 \Rightarrow \mathbf{P}^k \to \mathbf{0}$ and the control polygon converges to single zero point,
- $\lambda_0 = 1 > \lambda_1 \geq \dots \geq \lambda_j$ yields that the subdivision process converges to the limit curve [26].

The representation of the subdivision in the form (2.2) allows to build an iterated function system (IFS) corresponding to the curve with control points $\mathbf{P}^0$ and subdivision matrix $\mathbf{S}$. More about IFS for subdivision curves can be found in Section 3.6.

2.2 The de Casteljau subdivision

The de Casteljau algorithm, which was introduced in Subsection 1.1.3, can be viewed as a subdivision scheme, a method for finding new control points $\mathbf{q}_0(t), \dots, \mathbf{q}_n(t)$ and $\mathbf{r}_0(t), \dots, \mathbf{r}_n(t)$ from the original control points $\mathbf{p}_0, \dots, \mathbf{p}_n$. These new control points represent the original Bézier curve restricted to the parameter intervals $[0, t]$ and $[t, 1]$, respectively.

The new control points can be computed from the formulas

$$\begin{aligned} \mathbf{q}_k(t) &= \sum_{j=0}^{k} B_j^k(t)\mathbf{p}_j, & k = 0, \dots, n, \\ \mathbf{r}_k(t) &= \sum_{j=k}^{n} B_{n-j}^{n-k}(t)\mathbf{p}_{k+n-j}, & k = 0, \dots, n. \end{aligned} \tag{2.4}$$

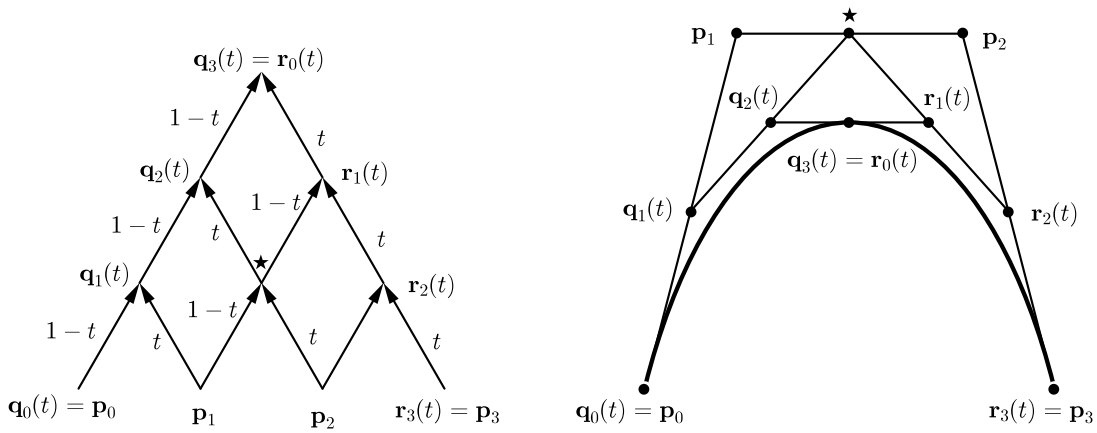

Figure 2.1: The de Casteljau algorithm for a cubic Bézier curve. On the left side is an illustration of the data flow. The Bézier curve is on the right.

The de Casteljau subdivision for a cubic Bézier curve $\mathbf{c}(t), t \in [0,1]$ is shown in Figure 2.1. The schematic picture on the left shows the data flow; each interior node is computed by adding two previous points multiplied by $(1-t)$ and t, respectively. For instance, $\mathbf{q}_1(t) = (1-t)\mathbf{p}_0 + t\mathbf{p}_1$. The points $\mathbf{q}_0(t), \ldots, \mathbf{q}_3(t)$ on the left side of the triangle are the new control points for the segment of the original Bézier curve in the interval $[0,t]$. The points $\mathbf{r}_0(t), \ldots, \mathbf{r}_3(t)$ on the right side of the triangle are the control points for the segment of the original Bézier curve in the interval $[t,1]$. The picture on the right illustrates the geometric interpretation of the algorithm. The two identical points $\mathbf{q}_3(t), \mathbf{r}_0(t)$ lie on the Bézier curve, i.e. $\mathbf{q}_3(t) = \mathbf{r}_0(t) = \mathbf{c}(t)$.

Matrix form The de Casteljau subdivision algorithm given in equation (2.4) can be rewritten into the following matrix form:

$$\begin{aligned}
\begin{pmatrix} \mathbf{q}_0 \\ \vdots \\ \mathbf{q}_n \end{pmatrix} &= \begin{pmatrix} B_0^0(t) & 0 & \ldots & 0 \\ B_0^1(t) & B_1^1(t) & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ B_0^n(t) & B_1^n(t) & \ldots & B_n^n(t) \end{pmatrix} \begin{pmatrix} \mathbf{p}_0 \\ \vdots \\ \mathbf{p}_n \end{pmatrix} = \mathbf{L}(t) \cdot \mathbf{P} \\
\begin{pmatrix} \mathbf{r}_0 \\ \vdots \\ \mathbf{r}_n \end{pmatrix} &= \begin{pmatrix} B_0^n(t) & B_1^n(t) & \ldots & B_n^n(t) \\ 0 & B_0^{n-1}(t) & \ldots & B_{n-1}^{n-1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & B_0^0(t) \end{pmatrix} \begin{pmatrix} \mathbf{p}_0 \\ \vdots \\ \mathbf{p}_n \end{pmatrix} = \mathbf{R}(t) \cdot \mathbf{P}
\end{aligned} \tag{2.5}$$

The matrices $\mathbf{L}(t)$, $\mathbf{R}(t)$ represent left and right subdivision scheme for Bézier curves. Starting with the original control points and applying these matrices repeatedly generates a sequence of control polygons that converge to the original Bézier curve. Example 2.2.1 illustrates this approach.

The matrices $\mathbf{L}(t)$, $\mathbf{R}(t)$ are square invertible matrices, therefore we can investigate their eigenvectors and eigenvalues. Their eigenvalues for $t = \frac{1}{2}$ are

$$\lambda_i = \left(\frac{1}{2}\right)^i, \quad i = 0, \ldots, n,$$

the corresponding eigenvector for $\lambda_0 = 1$ is eigenvector of ones $\mathbf{v}_0 = (1, 1, \ldots, 1)$. Further, for convergent subdivision schemes, the subdivision matrices have eigenvalues of the form $1 > \lambda_1 \geq \lambda_2 \ldots$, which in our case holds.

Example 2.2.1. As in Example 1.1.1 we construct a Bézier curve of degree 3 for the control points $\mathbf{P} = \{[0, 0], [0.5, 1], [1, 1], [1, 0]\}$, but we use the equation (2.5) of the de Casteljau subdivision algorithm. The first level of iteration is

$$\mathbf{Q}^1 = \mathbf{L}(\tfrac{1}{2}) \cdot \mathbf{P} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ \frac{1}{2} & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\mathbf{R}^1 = \mathbf{R}(\tfrac{1}{2}) \cdot \mathbf{P} = \begin{pmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 \\ \frac{1}{2} & 1 \\ 1 & 1 \\ 1 & 0 \end{pmatrix}.$$

Applying the matrices $\mathbf{L}(\frac{1}{2})$, $\mathbf{R}(\frac{1}{2})$ on the new control points $\mathbf{Q}^1$, $\mathbf{R}^1$ gives control points at level 2. Repeating this process iteratively generates control polygons that converge to the Bézier curve. Some of the iterations are shown in Figure 2.2.

2.3 B-spline subdivision

Such as in the case of Bézier curves, there exist subdivision schemes converging to uniform B-spline curves of arbitrary degree. The following theorem is due to J. M. Lane and R. F. Riesenfeld [22].

Theorem 2.3.1. *Let* $\mathbf{P} = \{\mathbf{p}_0, \ldots, \mathbf{p}_j\}$ *be a control polygon and* $\mathbf{T} = (t_0, t_1, \ldots, t_m)$ *be a uniform knot vector with mesh size* 1. *Let*

$$\mathbf{c}(t) = \sum_{i=0}^{j} \mathbf{p}_i N_i^n(t), \quad t \in \left[\frac{n}{2}, j - \frac{n}{2}\right]$$

be the uniform B-spline curve of degree n *with* $2 \leq n \leq j$. *Then*

$$\mathbf{c}(t) = \mathbf{c}_1(t) = \sum_{i=0}^{2j-n+2} \mathbf{p}_i^n N_i^n(t), \quad t \in \left[\frac{n}{2}, j - \frac{n}{2}\right],$$

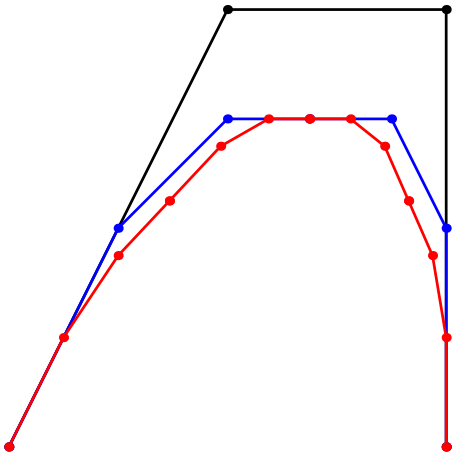

Figure 2.2: The de Casteljau subdivision algorithm. The original control polygon is black, the new control polygon after one iteration is blue, control polygon after two iterations is red.

where $\mathbf{T}_1 = (t_0, \ldots, t_{2m+1})$ *is a new uniform knot vector with mesh size* $\frac{1}{2}$*, and the points* $\mathbf{p}_i^m$ *are defined recursively by*

$$\begin{cases} \mathbf{p}_i^2 &= \begin{cases} \mathbf{p}_{i/2} & i \text{ even} \\ \dfrac{\mathbf{p}_{(i-1)/2} + \mathbf{p}_{(i+1)/2}}{2} & i \text{ odd} \end{cases} \qquad i = 0, 1, \ldots, 2j, \\ \mathbf{p}_i^n &= \dfrac{\mathbf{p}_i^{n-1} + \mathbf{p}_{i+1}^{n-1}}{2} \qquad i = 0, 1, \ldots, 2j+n+2, n > 2. \end{cases} \tag{2.6}$$

Proof can be found in [22].
Thus, given the B-spline curve of degree n with integral knot spacing and the control polygon $\mathbf{P}$, the control points for the same curve in terms of the B-spline basis over the refined mesh $0, 1/2, 1, \ldots, j - 1/2, j$ are given by (2.6).

The recursion equation (2.6) can be solved explicitly and the Lane-Riesenfeld algorithm can be thus derived.

Lane-Riesenfeld algorithm Given an initial set of control points $\mathbf{P}^0$, the control points at level $\mathbf{P}^{k+1}$ are given by the rules

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \dfrac{1}{2^n} \displaystyle\sum_{j=0}^{\lfloor \frac{n+1}{2} \rfloor} \binom{n+1}{2j} \mathbf{p}_{i-1+j}^k \\ \mathbf{p}_{2i+1}^{k+1} &= \dfrac{1}{2^n} \displaystyle\sum_{j=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n+1}{2j+1} \mathbf{p}_{i+j}^k. \end{cases} \tag{2.7}$$

The limit curve is a uniform B-spline curve of degree n, which is C^{n-1}– continuous (see [31]).

Example 2.3.1. From equations (2.7) it follows that the subdivision scheme for uniform quadratic B-spline curves is

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \frac{3}{4}\mathbf{p}_i^k + \frac{1}{4}\mathbf{p}_{i+1}^k \\ \mathbf{p}_{2i+1}^{k+1} &= \frac{1}{4}\mathbf{p}_i^k + \frac{3}{4}\mathbf{p}_{i+1}^k. \end{cases} \tag{2.8}$$

This subdivision scheme is also known as Chaikin algorithm and it was presented in 1974 by George Chaikin [31]. He introduced this effective method for generating smooth curves based on some geometric rules between points, in contrast to exact mathematical models based on parametrization.

Example 2.3.2. By equations (2.7), the subdivision scheme for uniform cubic B-spline curves is

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \frac{1}{8}\mathbf{p}_{i-1}^k + \frac{3}{4}\mathbf{p}_i^k + \frac{1}{8}\mathbf{p}_{i+1}^k \\ \mathbf{p}_{2i+1}^{k+1} &= \frac{1}{2}\mathbf{p}_i^k + \frac{1}{2}\mathbf{p}_{i+1}^k. \end{cases} \tag{2.9}$$

The related local[1] subdivision matrix is

$$\mathbf{S} = \begin{pmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{pmatrix}.$$

Let $\mathbf{P}^0$ be initial control points, then control points at level k can be expressed as:

$$\mathbf{P}^k = \mathbf{S}\mathbf{P}^{k-1} = \mathbf{S}^k\mathbf{P}^0.$$

The first level of iteration for the cubic case is

$$\mathbf{P}^1 = \begin{pmatrix} \mathbf{p}_0^1 \\ \mathbf{p}_1^1 \\ \mathbf{p}_2^1 \\ \mathbf{p}_3^1 \\ \mathbf{p}_4^1 \end{pmatrix} = \begin{pmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{pmatrix} \begin{pmatrix} \mathbf{p}_{-1}^0 \\ \mathbf{p}_0^0 \\ \mathbf{p}_1^0 \\ \mathbf{p}_2^0 \\ \mathbf{p}_3^0 \end{pmatrix}.$$

Eigenvalues of the matrix $\mathbf{S}$ are $\lambda_0 = 1, \lambda_1 = 1/2, \lambda_2 = 1/4$ and $\lambda_3 = \lambda_4 = 1/8$, which satisfies the condition for convergent subdivision scheme.

[1]Local subdivision matrix is a basic representation matrix of the binary subdivision rules without respect to number of control points.

2.4 Four-point subdivision

The four-point interpolating subdivision scheme was introduced by Dyn et al. [6]. It is defined as follows: for the initial set of control points $\mathbf{P}^0 = \{\mathbf{p}_j^0 \in \mathbb{R}^d\}$, let $\mathbf{P}^k = \{\mathbf{p}_j^k\}_{j=-2}^{2^k n+2}$ be the set of control points at level $k \in \mathbb{N}^0$, and $\mathbf{P}^{k+1} = \{\mathbf{p}_j^{k+1}\}_{j=-2}^{2^{k+1} n+2}$ be defined recursively by the following rule:

$$\begin{cases} \mathbf{p}_{2j}^{k+1} &= \mathbf{p}_j^k, & -1 \le j \le 2^k n + 1, \\ \mathbf{p}_{2j+1}^{k+1} &= (\frac{1}{2} + \omega)(\mathbf{p}_j^k + \mathbf{p}_{j+1}^k) - \omega(\mathbf{p}_{j-1}^k + \mathbf{p}_{j+2}^k), & -1 \le j \le 2^k n, \end{cases} \tag{2.10}$$

where ω is a tension parameter.

Geometric meaning of the tension parameter ω is illustrated in Figure 2.3. The midpoint vector is marked by $\mathbf{e} = \frac{1}{2}(\mathbf{p}_j^k + \mathbf{p}_{j+1}^k) - \frac{1}{2}(\mathbf{p}_{j-1}^k + \mathbf{p}_{j+2}^k)$.

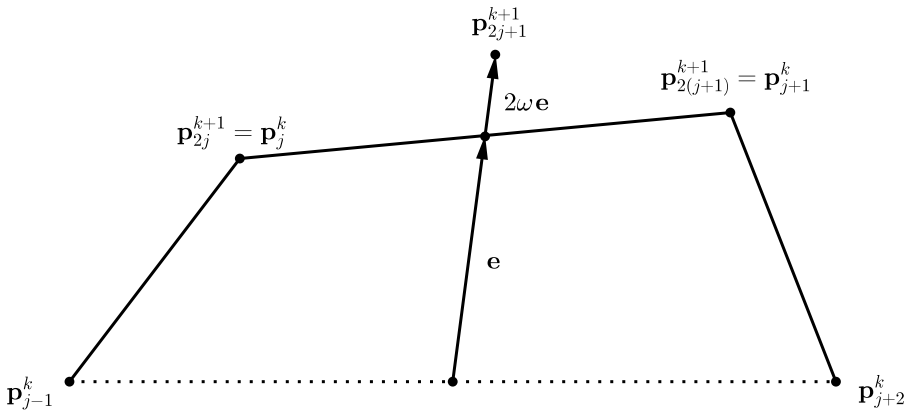

Figure 2.3: Geometric interpretation of the tension parameter $\omega > 0$.

Continuity of the limit curves depends on the tension parameter ω. It is proved that the four-point scheme for the value $\omega = \frac{1}{16}$ gives limit curves which are almost C^2-continuous.

But this subdivision method is not only an important tool for the fast generation of smooth curves from initial control points. It can be also an efficient tool for the fast generation of fractals. The limit curve of the four-point binary subdivision process can be fractal for some special values of the tension parameter ω. Authors in [17] analyze the fractal properties of this subdivision scheme and apply the obtained results to the generation of fractal curves and surfaces.

Theorem 2.4.1. *For $-\frac{1}{2} < \omega < 0$ or $\frac{1}{4} \le \omega < \frac{1}{2}$, the limit curve of the four-point scheme is a fractal curve.*

Proof of this theorem can be found in [17].

Nevertheless, for $\omega = \frac{1}{16}$, the limit curve of the four-point scheme is C^1-continuous and can reproduce cubic polynomials.

Chapter 3

Iterated function systems

In this chapter we define iterated function systems and study some of their properties. We first recall some standard terminology from dynamical systems and contraction mappings.

3.1 Basic dynamical systems terminology

Let X be any set, and let $f : X \to X$ be a mapping. Let f^k denote the kth iterate of f, where

$$\begin{array}{rcl} f^0(x) & = & x \\ f^n(x) & = & f(f^{n-1}(x)) \text{ for } n = 1, 2, \dots . \end{array} \tag{3.1}$$

An iterative scheme (f^n) given by (3.1) is called a *discrete dynamical system*. Further, the sequence $(x_n)_{n=0}^{\infty}$ on X given by

$$\begin{array}{rcl} x_0 & \in & X \\ x_n & = & f(x_{n-1}) \text{ for } n = 1, 2, \dots . \end{array} \tag{3.2}$$

is called the (forward) *orbit* through x_0.

A *fixed point* of a mapping $f : X \to X$ of a set X into itself is an $x \in X$ which is mapped onto itself, that is

$$f(x^*) = x^*,$$

the image $f(x)$ coincides with x.

A point $x \in X$ is said to be *periodic point* if $f^T(x) = x$ for some $T > 0$. An orbit (x_n) is called *periodic* if all its points are periodic, that is $\forall k \in \mathbb{N},\ x_k = x_{k+T}$. The smallest positive T is called the (minimal) *period*. A point $x \in X$ is *eventually periodic* if there exist a positive integer N such that $x_N = f^N(x)$ is periodic or the set $\{x_1, x_2, \dots\}$ is finite. An orbit is *eventually periodic* if all its points are eventually periodic.

Thus a fixed point is a periodic orbit with period $T = 1$.

A closed set $A \subset X$ is an *attractor* for f, if $f(A) = A$ and there exists a neighborhood U of A and a positive integer N such that $f^N(U) \subset U$ and

$$A = \bigcap_{t=1}^{\infty} f^t(U).$$

The set U is called a *fundamental neighborhood* of A. Further, the open set

$$B = \bigcup_{t>0} (f^t)^{-1}(U)$$

is called the *basin of attraction* of A.

Example 3.1.1. Let $X = \mathbb{R}$ and $f : \mathbb{R} \to \mathbb{R}$ be a linear mapping in the form $f(x) = \lambda x$, $\lambda \in \mathbb{R}$. The point $x^* = 0$ is a fixed point for all values of $\lambda \in \mathbb{R}$, since $f(0) = \lambda 0 = 0$. The nth iteration of f is $f^n(x) = \lambda^n(x)$, that is,

$$x_n = \lambda^n x. \tag{3.3}$$

If $|\lambda| < 1$, then the sequence (3.3) converges to 0 for all $x \in \mathbb{R}$; therefore $x^* = 0$ is the global attractor with basis of attraction equal to the whole set $\mathbb{R}$.
If $\lambda = -1$, then we have $x_n = (-1)^n x$ and all points $x \in \mathbb{R} - \{0\}$ are periodic with period $T = 2$.

Example 3.1.2. Let $X = \mathbb{R}^m$ and $f : \mathbb{R}^m \to \mathbb{R}^m$ be a linear mapping in the form $f(\mathbf{x}) = \mathbf{M}\,\mathbf{x}$, $\mathbf{M} \in \mathbb{R}^{m \times m}$ is a real square matrix. Equation (3.2) is now of the form

$$\begin{array}{rcll} \mathbf{x}_0 & = & (x_1, x_2, \ldots, x_m)^\top & \\ \mathbf{x}_n & = & \mathbf{M}^n \mathbf{x}_0 & n = 1, 2, \ldots . \end{array} \tag{3.4}$$

Notice that $\mathbf{x}$ are vectors of m real variables. For a fixed point the following condition must hold

$$\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = \begin{pmatrix} x_{1,1} & \ldots & x_{1,m} \\ \vdots & \ldots & \vdots \\ x_{m,1} & \ldots & x_{m,m} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}. \tag{3.5}$$

Vector $\mathbf{x}^* = (0, \ldots, 0)^\top$, which is called the zero vector, is a fixed point of the mapping. Further, if there exists an eigenvalue $\lambda_1 = 1$ such that $\mathbf{M}\mathbf{v}_1 = \lambda_1 \mathbf{v}_1 = \mathbf{v}_1$, then the entire eigenspace[1] of λ_1 consists of fixed points. Generally, there does not have to be such an eigenvalue.
On the other hand, if all eigenvalues of $\mathbf{M}$ are $|\lambda_i| < 1,\ i = 1, \ldots, m$, then the powers $\mathbf{M}^n$ of $\mathbf{M}$ converge to zero matrix [33]. More precisely:

$$\lim_{n \to \infty} \mathbf{M}^n = \mathbf{0} \Leftrightarrow |\lambda_i| < 1,\ i = 1, \ldots, m. \tag{3.6}$$

Therefore, the sequence $\mathbf{x}_n$ given by equation (3.5) converges to the zero vector for any $\mathbf{x} \in \mathbf{R}^m$. The zero vector is the global attractor with basin of attraction equal to the whole space $\mathbf{R}^m$.

[1]Eigenspace of λ_1 is the union of the zero vector $\mathbf{0}$ and the set of all eigenvectors corresponding to eigenvalue λ_1.

3.2 Complete metric spaces

When introducing dynamical systems no particular conditions are imposed on the set X, neither we needed any notion of distance on X. For our purpose, we need to define a distance between two sets, as well as the notion of dimension of a set. Therefore we introduce metric spaces and, in particular, complete metric spaces, where all Cauchy sequences are convergent. We start with a definition of a metric space.

Definition. A *metric space* is a pair (X, d), where X is a set and d is a *metric on* X, that is, a function defined on $X \times X$ such that for all $x, y, z \in X$ we have:

(M1) $d(x, y) \geq 0$ and $d(x, y) = 0 \Leftrightarrow x = y$.

(M2) $d(x, y) = d(y, x)$ (Symmetry).

(M3) $d(x, y) \leq d(x, z) + d(z, y)$ (Triangle inequality).

In what follows we will work with complete metric spaces, since, as noticed above, they are necessary for our investigation.

Definition. A sequence x_n in a metric space $X = (X, d)$ is said to be *Cauchy* if for every $\epsilon > 0$ there is an $N = N(\epsilon)$ such that

$$d(x_m, x_n) < \epsilon \quad \forall m, n > N. \tag{3.7}$$

The space X is said to be *complete* if every Cauchy sequence in X converges (that is, has a limit which is an element of X).

Definition 3.2.1. Let $X = (X, d)$ be a metric space. A mapping $f : X \to X$ is called a *contraction on* X if there is a non-negative real number $s < 1$ such that for all $x, y \in X$

$$d(f(x), f(y)) \leq s \cdot d(x, y), \qquad 0 \leq s < 1. \tag{3.8}$$

The smallest such number s is called the *contractivity factor* for f. If the equality holds, i.e. if $d(f(x), f(y)) = s \cdot d(x, y)$, then f is called a *contracting similarity*.

If f is a linear contracting similarity, then f transforms sets into geometrically similar sets. An iterative scheme of a contraction f given by (3.1) is a special case of a dynamical system, whose importance is expressed in the next theorem.

Theorem 3.2.1 (Banach fixed point theorem)**.** *Let $X = (X, d)$, $X \neq \emptyset$, be a complete metric space and let $f : X \to X$ be a contraction on X. Then f has precisely one fixed point. Such a fixed point is an attractor, whose basin of attraction is the whole space X.*

The proof can be found in [21, Section 5.1].

Further, we work with compact sets in order to introduce metric space $(\mathcal{H}(X), d_H)$, which is the ideal space for studying fractal geometry. Let us recall that a subset A of a metric space is said to be *compact* if every sequence in A has a convergent subsequence whose limit is an element of A. A subset A of $\mathbb{R}^n$ is compact iff A is closed and bounded.

Definition. Let (X, d) be a complete metric space. The collection of all nonempty compact subsets of X is called the *hyperspace of compact subsets* (of X) and is denoted by $\mathcal{H}(X)$.

We are going to define a metric on $\mathcal{H}(X)$ which turns it into complete metric space.

Definition. Let (X, d) be a complete metric space, $x \in X, B \in \mathcal{H}(X)$. Define

$$d(x, B) = \min\{d(x, y) \,|\, y \in B\}. \tag{3.9}$$

The quantity $d(x, B)$ is called the *distance from* the point x to the set B. Further, let $A, B \in \mathcal{H}(X)$. Define

$$d(A, B) = \max\{d(x, B) \,|\, x \in A\}. \tag{3.10}$$

$d(A, B)$ is called the *distance from* the set $A \in \mathcal{H}(X)$ to the set $B \in \mathcal{H}(X)$.

It is easy to verify that $d(A, B) \neq d(B, A)$ in general (e.g., if A is a proper subset of B). To make this distance symmetric, we define Hausdorff metric.

Definition. We define on the set $\mathcal{H}(X)$ a function $d_H : \mathcal{H}(X) \times \mathcal{H}(X) \to \mathbb{R}$ by

$$d_H(A, B) = \max\{d(A, B), d(B, A)\}, \tag{3.11}$$

where d is given by (3.9) and (3.10). The function d_H is *Hausdorff metric* on $\mathcal{H}(X)$.

It can be shown that the metric space $(\mathcal{H}(X), d_H)$ is complete (the interested reader is referred to [2, Section 7]).

3.3 Iterated function systems

We now define a contraction mapping on $\mathcal{H}(X)$.

Definition. Let (X, d) be a complete metric space and let $1 < N \in \mathbb{N}$. Let $f_i : X \to X,\ i = 1, \dots N$ be a collection of contraction mappings with respective contractivity factors s_i. Define a mapping $F : \mathcal{H}(X) \to \mathcal{H}(X)$ such that

$$F(A) = \bigcup_{i=1}^{N} f_i(A) \quad \forall A \in \mathcal{H}(X). \tag{3.12}$$

The system $(X; F)$ is called a (hyperbolic) *iterated function system (IFS).*

The operator $F : A \mapsto \bigcup_{i=1}^{N} f_i(A)$ defined above is also known as *Hutchinson operator* (or Barnsley operator).
Any finite set of contraction mappings on a complete metric space defines a hyperbolic iterated function system (IFS). Even though the notation IFS does not necessarily implies contraction mappings, i.e., IFS does not have to be hyperbolic, we will restrict ourself to the hyperbolic case and further will use notation IFS in the meaning of hyperbolic IFS.

Theorem 3.3.1 (Hutchinson)**.** *Let $(X; F)$ be an IFS. The transformation $F : \mathcal{H}(X) \to \mathcal{H}(X)$, as defined above, is a contraction mapping on the complete metric space $(\mathcal{H}(X), d_H)$ with contractivity factor $s = \max\{s_i \mid i = 1, \ldots, N\}$.*

The proof can be found in [18].

Remark. Let us recall that for a contraction mapping F with contractivity factor s, the following must hold

$$d_H(F(B), F(C)) \leq s \cdot d_H(B, C) \quad \forall B, C \in \mathcal{H}(X).$$

Therefore, according to Banach theorem (Theorem 3.2.1), F has its unique fixed point $A^* \in \mathcal{H}(X)$, which obeys

$$A^* = F(A^*) = \bigcup_{i=1}^{N} f_i(A^*),$$

and is given by

$$A^* = \lim_{i \to \infty} F^i(B) \text{ for any } B \in \mathcal{H}(X). \tag{3.13}$$

Definition. The fixed point $A^* \in \mathcal{H}(X)$ described in the remark above is called the *attractor of the IFS* $(X; F)$.

If f_i are linear contracting similarities (see Definition 3.2.1), the unique attractor A is the union of smaller copies of itself.

3.4 Fractals

To deal with fixed points of IFS, we need to reconsider the notion of dimension of a set. More precisely, we will define Hausdorff and box dimension. Such dimensions tell us how densely an object occupies the metric space in which it lies. Intuitively, we think about a line as about 1-dimensional object, whereas a smooth surface is 2-dimensional. But how about the Siérpinski gasket? It apparently occupies more space than a smooth curve but less space that a surface. It is necessary to quantify such a property and for that reason introduce new notion of dimension.

We define the so called box dimension or box-counting dimension, which was introduced by Russian mathematician A. N. Kolmogorov and it is also know as Kolmogorov capacity. It uses covering by closed cubes, but closed balls can be used equivalently.

Definition. Let M be a bounded subset of $\mathbb{R}^n, n \in \mathbb{N}$, and let $\epsilon > 0$ be given. For $x_0 \in M$, denote by

$$W_\epsilon(x_0) = \{x \in \mathbb{R}^n \mid \parallel x - x_0 \parallel = \epsilon/2\} \tag{3.14}$$

the n-dimensional sphere with radius $\epsilon > 0$ and center x_0. Denote by $\mathcal{N}(M)$ the minimum number of such spheres required to cover the set M, i.e.,

$$\mathcal{N}_\epsilon(M) = \inf\{n \in \mathbb{N} \mid M \subseteq \bigcup_{i=1}^{n} W_\epsilon(x_i)\,;\, x_i \in M\}$$

Then the *box dimension* of the set M is defined by

$$dim_B(M) = \lim_{\epsilon \to 0+} \frac{\ln(\mathcal{N}_\epsilon(M))}{\ln(1/\epsilon)}, \tag{3.15}$$

provided the limit exists.

Box dimension is easy to use and it is adequate for our purpose. But it is loose in some cases, e.g., the set $M = \mathbb{Q} \cap [0, 1]$ is a set of zero measure, but $dim_B(M) = 1$.

For completeness of this text we define the Hausdorff dimension, which was introduced in 1918 by the German mathematician Felix Hausdorff. As stated in [9], Hausdorff dimension has the advantage of being defined for any set. A major disadvantage is that in many cases it is hard to calculate or to estimate by computational methods. According to [24], the explicit computation of the Hausdorff dimension is rather difficult since it involves taking the infimum over covers consisting of balls of radius less than or equal to a given $\epsilon > 0$. Whereas the box dimension involves only covers by balls of radius equal to ϵ or covers by cubes, therefore a slight simplification.

Note that some authors refer to Hausdorff dimension as *Hausdorff-Besicovitch dimension.*

Hausdorff dimension

Let M be a bounded subset of $\mathbb{R}^n$ and $x_0 \in M$. Let

$$B_r(x_0) = \{x \in M \mid \parallel x - x_0 \parallel < r\} \tag{3.16}$$

be the ball of radius $r > 0$ with center at x_0. Here $\parallel . \parallel$ denotes Euclidean norm on $\mathbb{R}^n$. Choose $\epsilon > 0$ and cover M by balls of radius $0 < r_i \leq \epsilon$ centered at points $x_i \in M$:

$$M \subseteq B_{r_i}(x_i).$$

For $s \geq 0$, consider the quantity

$$\mathfrak{H}^s_\epsilon(M) = \inf\left\{ \sum_{i=1}^{\infty} |r_i|^s \;\mid\; M \subseteq B_{r_i}(x_i);\, 0 < r_i \leq \epsilon \right\} \tag{3.17}$$

and define

$$\mathfrak{H}^s(M) = \lim_{\epsilon \to 0+} \sup \mathfrak{H}^s_\epsilon(M). \tag{3.18}$$

Since $\mathfrak{H}^s_{\epsilon_1}(M) \geq \mathfrak{H}^s_{\epsilon_2}(M)$, for $\epsilon_2 > \epsilon_1 > 0$, the above limit superior exists and is an element of $[0, \infty]$. According to [24], it is easy to verify that the function $s \mapsto \mathfrak{H}^s_\epsilon(M)$ is nondecreasing for $0 < \epsilon < 1$.
Now assume that $0 < \epsilon < 1$ and that $t > s \geq 0$. Then,

$$\sum_{i=1}^{\infty} |r_i|^t \leq \epsilon^{t-s} \sum_{i=1}^{\infty} |r_i|^s$$

and, therefore,

$$\mathfrak{H}^t_\epsilon(M) \geq \epsilon^{t-s} \mathfrak{H}^s_\epsilon(M).$$

If $\mathfrak{H}^s(M) < \infty$, then

$$0 \leq \lim_{\epsilon \to 0+} \sup \mathfrak{H}^t_\epsilon(M) \leq \big(\lim_{\epsilon \to 0+} \epsilon^{t-s} \big) \big(\lim_{\epsilon \to 0+} \sup \mathfrak{H}^s_\epsilon(M) \big) = 0.$$

On the other hand, if $\mathfrak{H}^t(M) < \infty$, then

$$\lim_{\epsilon \to 0+} \sup \mathfrak{H}^s_\epsilon(M) \geq \big(\lim_{\epsilon \to 0+} \epsilon^{s-t} \big) \big(\lim_{\epsilon \to 0+} \sup \mathfrak{H}^t_\epsilon(M) \big) = \infty.$$

The quantity $\mathfrak{H}^s(M)$ thus exhibits a $0 - \infty$ behavior. Based on these observations, one defines the *Hausdorff dimension* of M by

$$dim_H(M) = \inf\{s \geq 0 \mid \mathfrak{H}^s(M) = 0\} = \sup\{s \geq 0 \mid \mathfrak{H}^s(M) = \infty\}, \tag{3.19}$$

i.e., as such a value of s at which $\mathfrak{H}^s(M)$ jumps from 0 to ∞.

Now we can define fractals. We are concerned with fractals generated from IFS consisting of affine transformations, these fractals are generally self-similar and it is possible to count their box dimension.

Definition. A subset of $\mathbb{R}^n$ is called *fractal* if its box dimension is not an integer.

In our context, fractals are invariably fixed points of IFSs. However, this construction is more general.

3.5 Affine IFS

Affine transformations are an important class of iterated function systems, with which many of the best-known fractals are generated. In this section, we look closer on this class of transformations.

Definition. A transformation $f : \mathbb{R}^n \to \mathbb{R}^n$ of the form

$$f(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}, \tag{3.20}$$

where $\mathbf{A}$ is a square regular matrix $n \times n$ and $\mathbf{b}$ is n-dimensional vector, is called an *affine transformation.*

Thus an affine transformation is a linear transformation represented by matrix $\mathbf{A}$ followed by a translation in the form of a real vector $\mathbf{b} \in \mathbb{R}^n$. The fixed point equation for f reads

$$\mathbf{x}^* = \mathbf{A}\mathbf{x}^* + \mathbf{b}.$$

A simple manipulation yields

$$\mathbf{x}^* = (\mathbf{I} - \mathbf{A})^{-1}\mathbf{b}, \tag{3.21}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix. The matrix $(\mathbf{I} - \mathbf{A})$ is invertible if and only if its determinant is not zero, or equivalently, 1 is not an eigenvalue of $\mathbf{A}$.

We want to represent f in matrix form in homogenous coordinates (since we use this form for IFS of subdivision curves). For this purpose we use an augmented matrix and an augmented vector[2], and define a new $(n+1) \times (n+1)$ matrix $\mathbf{M}$ of the form:

$$\mathbf{M} = \begin{pmatrix} \mathbf{A} & \mathbf{b} \\ 0, \ldots, 0 & 1 \end{pmatrix}.$$

The equation (3.20) is then as follows:

$$f\begin{pmatrix} x_1 \\ \vdots \\ x_n \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{b} \\ 0, \ldots, 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \\ 1 \end{pmatrix}. \tag{3.22}$$

It is useful to work with equation (3.22), since the iterations of the function f can be written as powers of the augmented matrix $\mathbf{M}$, i.e.,

$$f^k(\mathbf{x}) = \mathbf{M}^k\mathbf{x}.$$

Using the new form (3.22), for a fixed point the following must hold:

$$\begin{pmatrix} \mathbf{x}^* \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}^* \\ 1 \end{pmatrix}.$$

After multiplication we obtain

$$\begin{aligned} \mathbf{x}^* &= (\mathbf{A} \quad \mathbf{b})(\mathbf{x}^* \quad 1)^\top = \mathbf{A}\mathbf{x}^* + \mathbf{b} \\ 1 &= (\mathbf{0} \quad 1)(\mathbf{x}^* \quad 1)^\top = \mathbf{0} \cdot \mathbf{x}^* + 1 \cdot 1 = 1, \end{aligned}$$

[2]An augmented vector is a vector that is augmented with and extra dimension.

Therefore, provided the fixed point exists, it is identical to the fixed point in equation (3.21).

We would like to know when a set of affine transformations has unique fixed point.

Definition. A map f is *eventually contractive* if there exist a positive integer n such that f^n is contractive. Such a smallest exponent n is called the exponent of eventual contractivity [12].

Moreover, an IFS F is eventually contractive if there exists some positive integer n such that F^n is contractive. The integer n specifies the number of iterations required of F before the system is contractive [20]. All contractive maps are eventually contractive, but not vice–versa.

Theorem 3.5.1. *An affine transformation f given by matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ and translation vector $\mathbf{b} \in \mathbb{C}^n$ is eventually contractive if all eigenvalues of $\mathbf{A}$ are within unit circle.*

Proof. If $|\lambda_i| < 1,\ i = 1, \ldots, n$, then the powers $\mathbf{A}^k$ of $\mathbf{A}$ converge to zero matrix [33], that is:

$$\lim_{k\to\infty} \mathbf{A}^k = \mathbf{0} \Leftrightarrow |\lambda_i| < 1,\ i = 1, \ldots, n.$$

□

The eventual contractivity ensures that the fixed point from equation (3.21) is unique. To this end we state the following theorem.

Theorem 3.5.2. *If X is a complete metric space and $f : X \to X$ is a mapping such that some iterate $f^N : X \to X$ is a contraction, then f has a unique fixed point. Moreover, the fixed point of f can be obtained by iteration of f starting from any $x_0 \in X$ [4].*

The proof can be found in [4, Section 3]. Since the metric space $(\mathbb{C}^n, d_H)$ is complete, an affine transformation satisfying the conditions of Theorem 3.5.1 has a unique fixed point, which obeys the equation (3.21).

3.6 IFS for subdivision curves

In this section we deal with IFS created from binary subdivision schemes. We show that an IFS can be constructed for all binary subdivision curves. Informally, given a subdivision matrix $\mathbf{S}$ constructed by the rules (2.2), we break the matrix into two suitable square matrices $\mathbf{S}_1, \mathbf{S}_2$ of smaller size, and we apply all products of $\mathbf{S}_i$ of length k to the control points $(\mathbf{p}_0, \ldots, \mathbf{p}_n)^\top$, then the control polygon converges to the the limit curve as k approaches ∞.

Such an IFS is constructed from transformations $f_i : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ by defining

$$f_i(X) = X\mathbf{P}^{-1}\mathbf{S}_i\mathbf{P} = X\mathbf{M}_i, \tag{3.23}$$

where $\mathbf{P}$ is a square matrix which was created from the matrix of control points $(\mathbf{p}_0, \ldots, \mathbf{p}_n)^\top$ by adding columns from identity matrix and a column of ones corresponding to homogenous component of the coordinates.

The equation $\mathbf{M}_i = \mathbf{P}^{-1}\mathbf{S}_i\mathbf{P}$ from (3.23) corresponds to a change of basis of the vector space $\mathbb{R}^{n+1}$, where $\mathbf{P}$ is the change-of-basis matrix, $\mathbf{S}_i$ represents the linear transformation f_i in the old basis, and $\mathbf{M}_i$ represents f_i in the new basis.

Because the elements in a row of $\mathbf{S}_i$ sum up to 1 (see equation (2.1)), and the last column of $\mathbf{P}$ is a column of ones, the last column of $\mathbf{S}_i\mathbf{P}$ is also a column of ones. Further, the last column of $\mathbf{P}^{-1}\mathbf{S}_i\mathbf{P}$ is a column of the identity matrix because the last column of $\mathbf{P}$ is a column of ones and $\mathbf{P}^{-1}\mathbf{P} = \mathbf{I}$. Thus the matrix $\mathbf{M}_i = \mathbf{P}^{-1}\mathbf{S}_i\mathbf{P}$ has the following form:

$$\mathbf{M}_i = \begin{pmatrix} m_{0,0} & \ldots & 0 \\ \vdots & \vdots & \vdots \\ m_{n-1,0} & \ldots & 0 \\ m_{n,0} & \ldots & 1 \end{pmatrix}. \tag{3.24}$$

The last row is the translational vector. Thus, equation (3.23) is the transpose of an affine transformation of the form (3.22). Therefore we can rewrite (3.23) as follows

$$f_i(\mathbf{x} \quad 1) = (\mathbf{x} \quad 1) \begin{pmatrix} \mathbf{A}_i & \mathbf{0} \\ \mathbf{b}_i & 1 \end{pmatrix}. \tag{3.25}$$

According to Theorem 3.5.1, the mapping f_i is eventually contractive if all eigenvalues of the matrix $\mathbf{A}_i$ are within the unit circle. If this is the case, then Theorem 3.5.2 implies that the fixed point given by

$$\mathbf{x}^* = \mathbf{b}_i(\mathbf{I} - \mathbf{A}_i)^{-1}$$

is unique.

In order to show that our mappings have unique fixed points, we prove the following theorem.

Theorem 3.6.1. *The eigenvalues of the matrix $\mathbf{A}_i$ from equation (3.25) are all within the unit circle.*

Proof. As we stated in Section 2.1, in the case of convergent subdivision schemes, the subdivision matrices $\mathbf{S}_i$ have eigenvalues of the form $\lambda_0 = 1 > \lambda_1 \geq \cdots \geq \lambda_n$. The matrix $\mathbf{M}_i = \mathbf{P}^{-1}\mathbf{S}_i\mathbf{P}$ from (3.23) has the same eigenvalues as the matrix $\mathbf{S}_i$, since the equation corresponds just to a change of basis and such a change does not alter the eigenvalues of the matrices.
Further, let us recall that the characteristic equation of the matrix $\mathbf{M}_i$ reads

$$\det(\mathbf{M}_i - \lambda\mathbf{I}) = 0.$$

Because the matrix $\mathbf{M}_i$ is always of the form (3.24), its eigenvalue $\lambda_0 = 1$ is associated to its last column, which is a column of the identity matrix.
Therefore, the matrix $\mathbf{A}_i$ has precisely the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n$, which are all within the unit circle. □

3.6.1 IFS for uniform B-spline curves

Let us recall that binary subdivision scheme for a uniform B-spline curves of degree n is

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \dfrac{1}{2^n} \displaystyle\sum_{j=0}^{\lfloor \frac{n+1}{2} \rfloor} \binom{n+1}{2j} \mathbf{p}_{i-1+j}^{k} \\ \mathbf{p}_{2i+1}^{k+1} &= \dfrac{1}{2^n} \displaystyle\sum_{j=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n+1}{2j+1} \mathbf{p}_{i+j}^{k} \end{cases}$$

It is possible to construct IFS for any uniform B-spline curves, where the knot vector is uniform and periodic.

Let us assume that a uniform B-spline curve is given by $m+1$ control points $\mathbf{p}_0, \ldots, \mathbf{p}_m$, then the subdivision matrix $\mathbf{S}$ from equation (2.2) is always $(2m+2-n) \times (m+1)$ and we break it into two matrices $\mathbf{S}_1$, $\mathbf{S}_2$ such that the last n rows of $\mathbf{S}_1$ are identical to first n rows $\mathbf{S}_2$ and both are $(m+1) \times (m+1)$ matrices. The algorithm is illustrated in Examples 3.6.1 and 3.6.2.

Example 3.6.1. The subdivision scheme for uniform cubic B-spline curves is

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \frac{1}{8}\mathbf{p}_{i-1}^{k} + \frac{3}{4}\mathbf{p}_{i}^{k} + \frac{1}{8}\mathbf{p}_{i+1}, \\ \mathbf{p}_{2i+1}^{k+1} &= \frac{1}{2}\mathbf{p}_{i}^{k} + \frac{1}{2}\mathbf{p}_{i+1}^{k}. \end{cases} \tag{3.26}$$

We will generate uniform cubic B-spline curve with control points $\mathbf{P} = \{\mathbf{p}_0 = [0,0], \mathbf{p}_1 = [1,2], \mathbf{p}_2 = [3,2.5], \mathbf{p}_3 = [3.5,0]\}$. Using the equations (3.26), resorting to matrix form and breaking the matrix into two matrices, we obtain

$$\mathbf{S}_1 = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{pmatrix}, \qquad \mathbf{S}_2 = \begin{pmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

We can now construct IFS consisting of two transformations f_1, f_2 for our subdivision curve

$$\begin{aligned} f_1(X) &= X\mathbf{P}^{-1}\mathbf{S}_1\mathbf{P} = X\mathbf{M}_1, \\ f_2(X) &= X\mathbf{P}^{-1}\mathbf{S}_2\mathbf{P} = X\mathbf{M}_2, \end{aligned}$$

where

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 1 & 2 & 0 & 1 \\ 3 & 2.5 & 0 & 1 \\ 3.5 & 0 & 0 & 1 \end{pmatrix}.$$

The matrices $\mathbf{M}_1, \mathbf{M}_2$ have the form

$$\mathbf{M}_1 = \begin{pmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{b}_1 & 1 \end{pmatrix}, \mathbf{M}_2 = \begin{pmatrix} \mathbf{A}_2 & \mathbf{0} \\ \mathbf{b}_2 & 1 \end{pmatrix}.$$

Eigenvalues of matrices $\mathbf{A}_1, \mathbf{A}_2$ are all within unit circle, therefore f_1 and f_2 are eventually contractive mappings and the IFS has unique attractor, which is the the B-spline subdivision curve.

To generate the subdivision curve, we apply the IFS on any initial set of points. We choose the set of points $X_0 = \{[4,0],[4,4]\}$ and iteratively repeat the process. The process is illustrated in Figure 3.1. The black curve in both the figures is the subdivision curve which was created from its parametrization, the pink dots are points $X_{i+1} = f_1(X_i) \bigcup f_2(X_i)$. We see that the set X_0 converges to the subdivision curve fast.

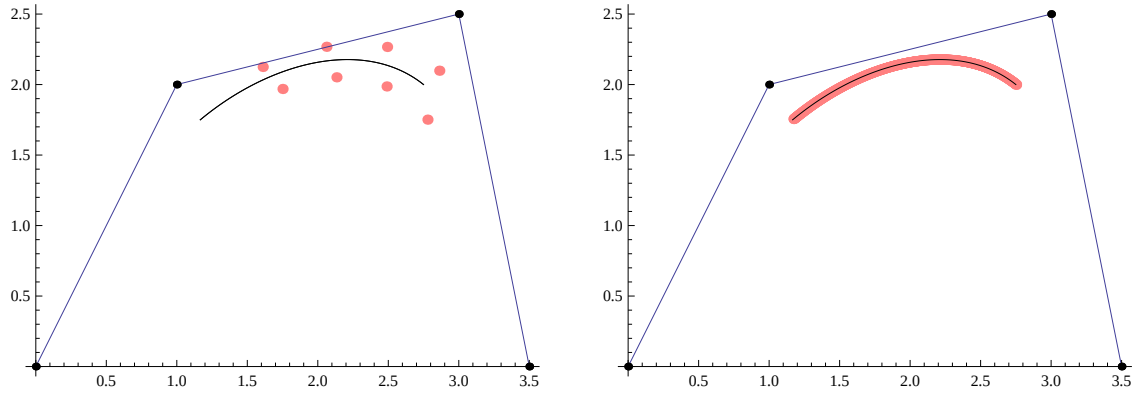

Figure 3.1: The subdivision curve is black, its control polygon is green, the pink dots are points X, left figure shows points X after two iterations, on the right are points X after 8 iterations.

Example 3.6.2. The subdivision scheme for uniform quadratic B-spline curves is

$$\begin{cases} \mathbf{p}_{2i}^{k+1} &= \frac{1}{4}\mathbf{p}_{i-1}^{k} + \frac{3}{4}\mathbf{p}_{i}^{k}, \\ \mathbf{p}_{2i+1}^{k+1} &= \frac{3}{4}\mathbf{p}_{i}^{k} + \frac{1}{4}\mathbf{p}_{i+1}^{k}. \end{cases} \tag{3.27}$$

We will generate uniform quadratic B-spline curve with control points $\mathbf{P}_0 = \{\mathbf{p}_0 = [0,0], \mathbf{p}_1 = [1,4], \mathbf{p}_2 = [4,4], \mathbf{p}_3 = [5,-1], \mathbf{p}_4 = [7,0]\}$. Using the equations (3.27) we construct subdivision matrix $\mathbf{S}$, which is 8×5, and break it into two matrices 5×5:

$$\mathbf{S}_1 = \begin{pmatrix} \frac{3}{4} & \frac{1}{4} & 0 & 0 & 0 \\ \frac{1}{4} & \frac{3}{4} & 0 & 0 & 0 \\ 0 & \frac{3}{4} & \frac{1}{4} & 0 & 0 \\ 0 & \frac{1}{4} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & 0 \end{pmatrix}, \qquad \mathbf{S}_2 = \begin{pmatrix} 0 & \frac{1}{4} & \frac{3}{4} & 0 & 0 \\ 0 & 0 & \frac{3}{4} & \frac{1}{4} & 0 \\ 0 & 0 & \frac{1}{4} & \frac{3}{4} & 0 \\ 0 & 0 & 0 & \frac{3}{4} & \frac{1}{4} \\ 0 & 0 & 0 & \frac{1}{4} & \frac{3}{4} \end{pmatrix}.$$

We can now construct IFS consisting of two transformations f_1, f_2 for our subdivision curve

$$\begin{aligned} f_1(X) &= X\mathbf{P}^{-1}\mathbf{S}_1\mathbf{P} = X\mathbf{M}_1, \\ f_2(X) &= X\mathbf{P}^{-1}\mathbf{S}_2\mathbf{P} = X\mathbf{M}_2, \end{aligned}$$

where

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 4 & 0 & 1 & 1 \\ 4 & 4 & 0 & 0 & 1 \\ 5 & -1 & 0 & 0 & 1 \\ 7 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

The matrices $\mathbf{M}_1, \mathbf{M}_2$ have the form

$$\mathbf{M}_1 = \begin{pmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{b}_1 & 1 \end{pmatrix}, \mathbf{M}_2 = \begin{pmatrix} \mathbf{A}_2 & \mathbf{0} \\ \mathbf{b}_2 & 1 \end{pmatrix}.$$

Eigenvalues of matrices $\mathbf{A}_1, \mathbf{A}_2$ are all within unit circle, therefore f_1 and f_2 have unique fixed points. The unique fixed point of transformation f_1 is

$$\mathbf{x}_1^* = \mathbf{b}_1(\mathbf{I} - \mathbf{A}_1)^{-1} = \left(\frac{1}{2}, 2, \frac{1}{2}, \frac{1}{2}\right),$$

which corresponds to the point $[\frac{1}{2}, 2]$ in $\mathbb{R}^2$. The unique fixed point of transformation f_2 is

$$\mathbf{x}_2^* = \mathbf{b}_2(\mathbf{I} - \mathbf{A}_2)^{-1} = \left(6, -\frac{1}{2}, 0, 0\right),$$

which corresponds to the point $[6, -\frac{1}{2}]$. The limiting curve in $\mathbb{R}^2$ has to interpolate these two fixed points.

We iterate the IFS with chosen initial conditions $X_0 = \{[0,0], [1,4], [2,4], [4,0], [5,0]\}$ to generate the subdivision curve, see Figure 3.2. The black curve in both the figures is the subdivision curve which was created from its parametrization, the pink dots are points $X_{i+1} = f_1(X_i) \bigcup f_2(X_i)$. We see that the set X_0 converges to the subdivision curve, which interpolates the fixed points $\left[\frac{1}{2}, 2\right]$ and $\left[6, -\frac{1}{2}\right]$, as expected.

3.7 IFS for complex Bézier curves

The de Casteljau algorithm gives a subdivision scheme for Bézier curves with real parameter r, as introduced in Section 2.2. In this section we extend the de Casteljau subdivision scheme to the case of complex parameter $t \in \mathbb{C}$.

Resorting to complex domain gives interesting results, and many famous fractals can be generated by this method. We introduce an IFS for complex Bézier curves as follows. The IFS is constructed from transformations f_1 and f_2 by defining

$$f_1(X) = \mathbf{P}\mathbf{L}^\top(t)\mathbf{P}^{-1}X = \mathbf{L}X, \qquad f_2(X) = \mathbf{P}\mathbf{R}^\top(t)\mathbf{P}^{-1}X = \mathbf{R}X, \tag{3.28}$$

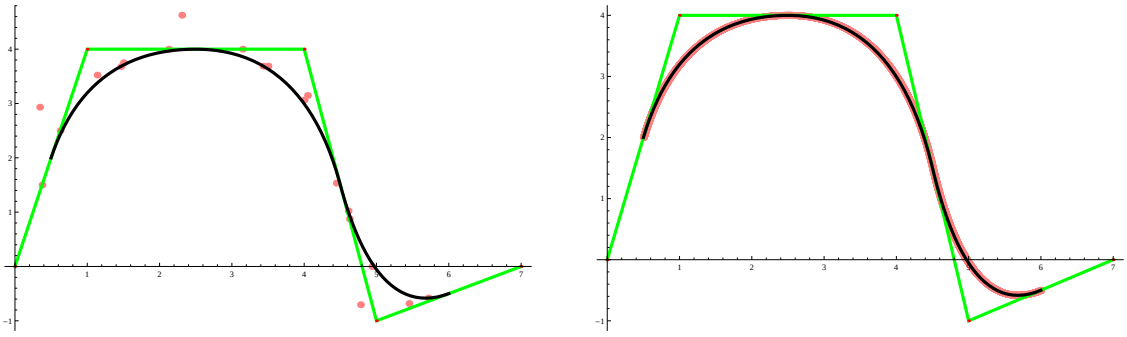

Figure 3.2: The subdivision curve is black, its control polygon is green, the pink dots are points X, left figure shows points X after two iterations, on the right are points X after 8 iterations.

where $\mathbf{L}(t), \mathbf{R}(t)$ are the de Casteljau subdivision matrices (2.5) and $\mathbf{P}$ is a square matrix which was created from the matrix of control points $(\mathbf{p}_0, \ldots, \mathbf{p}_n)$ by adding rows from identity matrix and a row of ones corresponding to homogenous component of the coordinates.

Equation (3.28) is just a transposed equivalent to equation (3.23), therefore all the statements valid for subdivision IFS are right also for the case of IFS for Bézier curves.

We want to show that the attractor of the de Casteljau IFS for unit segment is connected for all

$$t \in \mathbb{C}, \quad |t| < 1 \ \wedge \ |1 - t| < 1.$$

We will prove this by using the following theorem.

Theorem 3.7.1. *Let $\{X; f_1, f_2\}$ be a hyperbolic IFS with attractor A. Let f_1 and f_2 be one-to-one*[3] *on A. If*

$$f_1(A) \cap f_2(A) = \emptyset,$$

then A is totally disconnected. If

$$f_1(A) \cap f_2(A) \neq \emptyset,$$

then A is connected.

Proof can be found in [2, Chapter 2].

In order to use Theorem 3.7.1 we have to check the contractivity of f_1 and f_2. By equation (3.28), the subdivision matrices for a Bézier segment with control points 0 and 1, and subdivision parameter t generate the transformations $f_1, f_2 : \mathbb{C} \to \mathbb{C}$ as follows:

$$f_1 \begin{pmatrix} z \\ 1 \end{pmatrix} = \begin{pmatrix} t & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z \\ 1 \end{pmatrix}, \qquad f_2 \begin{pmatrix} z \\ 1 \end{pmatrix} = \begin{pmatrix} 1-t & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} z \\ 1 \end{pmatrix}. \tag{3.29}$$

[3] injective function

These equations represent the affine transformations f_1, f_2 in augmented matrix form (3.22), where $\mathbf{A}_1 = t$, $\mathbf{b}_1 = 0$, $\mathbf{A}_2 = 1 - t$, and $\mathbf{b}_2 = t$. For the mappings f_1, f_2 to be contractive, the following must hold:

$$\begin{aligned} \| \mathbf{A}_1 \| &= |t| < 1 \\ \| \mathbf{A}_2 \| &= |1 - t| < 1. \end{aligned}$$

That is, for all $t \in \mathbb{C}$, $|t| < 1 \wedge |1 - t| < 1$, the IFS consisting of f_1 and f_2 is a hyperbolic IFS and has a unique attractor A. We illustrate the domain of t over which is the IFS hyperbolic in Figure 3.3.

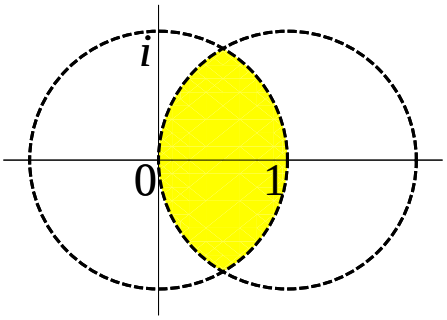

Figure 3.3: The domain of t, over which is IFS $\{\mathbb{C}; f_1, f_2\}$ hyperbolic, is the intersection of two open spheres $|t| < 1$ and $|1 - t| < 1$.

Further, by equation (3.21), fixed points of f_1, f_2 are

$$\begin{aligned} z_1^* &= (\mathbf{I} - \mathbf{A}_1)^{-1}\mathbf{b}_1 = (1 - t)^{-1}0 = 0 \\ z_2^* &= (\mathbf{I} - \mathbf{A}_2)^{-1}\mathbf{b}_2 = (t)^{-1}t = 1. \end{aligned}$$

The fixed points 0 and 1 are naturally elements of the attractor A. Now we are ready to state and prove the following theorem.

Theorem 3.7.2. *Let f_1 and f_2 be transformations defined by subdivision matrices for a Bézier segment with control points 0 and 1, and subdivision parameter $t \in \mathbb{C}$, as in equation (3.29). Let $|t| < 1 \wedge |1 - t| < 1$. Then the attractor A of the IFS $\{\mathbb{C}; f_1, f_2\}$ is connected.*

Proof. We have shown in the previous paragraphs that f_1 and f_2 are contractions for all $t \in \mathbb{C}$, $|t| < 1 \wedge |1 - t| < 1$. Since f_1 and f_2 are linear transformations, they are one-to-one on A. Further, point $z_1^* = 0$ is the fixed point of f_1 and $z_2^* = 1$ is the fixed point of f_2, and both of them are elements of A. According to Theorem 3.7.1, A is connected if $f_1(A) \cap f_2(A) \neq \emptyset$. We have

$$\begin{aligned} f_1 \begin{pmatrix} z_2^* \\ 1 \end{pmatrix} &= \begin{pmatrix} t & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} t \\ 1 \end{pmatrix}, \\ f_2 \begin{pmatrix} z_1^* \\ 1 \end{pmatrix} &= \begin{pmatrix} 1 - t & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} t \\ 1 \end{pmatrix}. \end{aligned}$$

That is, $f_1(z_2^*) = f_2(z_1^*) \Rightarrow f_1(A) \cap f_2(A) \neq \emptyset$, which completes the proof. □

Remark. Equations (3.29) can be articulated differently. The mappings f_1, f_2 as linear transformations in one variable $z \in \mathbb{C}$ are of the form

$$f_1(z) = tz \qquad f_2(z) = (1-t)z + t, \tag{3.30}$$

where $|t| < 1$ and $|1-t| < 1$. These transformations and their extensions have been studied by the Swiss mathematician G. De Rham. More can be found in [8, Chapter 16].

Example 3.7.1. In this example we generate the attractor A of the IFS introduced in this section. We start with a segment between the points 0 and 1, and plot the resulting curve after 15 iterations for various $t \in \mathbb{C}$ in Figures 3.4 and 3.5. Some more cases can be found in Example 3.7.2.

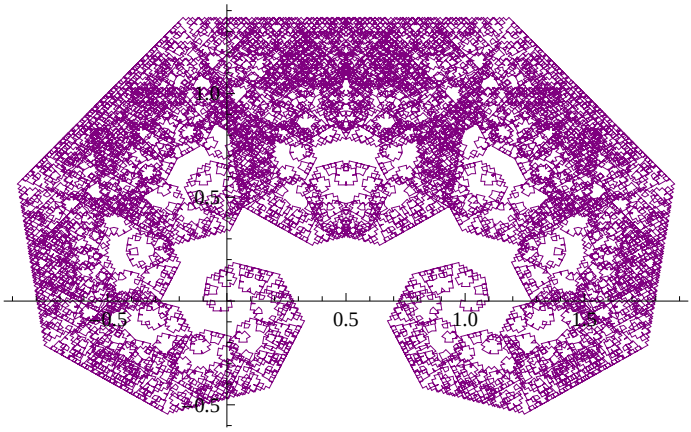

Figure 3.4: For $t = \frac{1}{2} + \frac{\sqrt{3}}{3}i$, the limiting curve resembles the Lévy C curve.

3.7.1 The Takagi curve

The Takagi fractal curve, also called Blancmange function, is a continuous function which is nowhere differentiable. The Takagi function is defined on the unit interval by

$$\mathcal{T}(x) = \sum_{n=0}^{\infty} \frac{\sigma(2^n x)}{2^n},$$

where $\sigma(x)$ is defined by $\sigma(x) = \min_{n \in \mathbb{Z}} |x - n|$, that is, $\sigma(x)$ is the distance from x to the nearest integer.

The Takagi curve can be approximated by a complex Bézier curve, which we show in the following paragraphs.

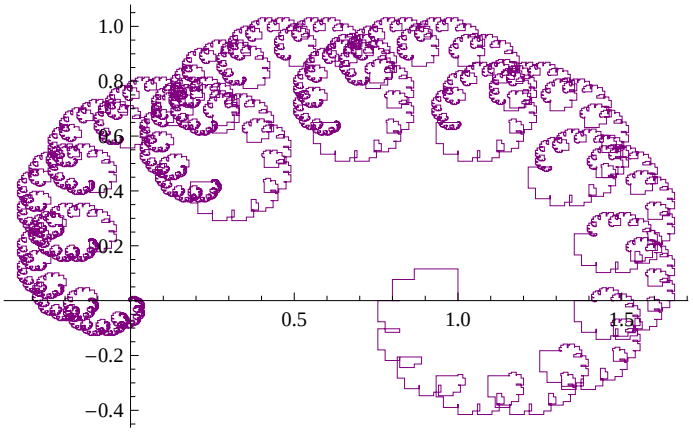

Figure 3.5: Here $t = \frac{1}{4} + \frac{\sqrt{3}}{4}$. The resulting curve after 15 iterations is not quite the limiting curve, i.e. the attractor of given IFS. The closer is t to the boundary of the domain of t over which are f_1 and f_2 contractive, the slower is the convergence to the limiting curve.

Let d be a finite binary code $d = [d_1, d_2, ..., d_n],\ d_k \in \{0, 1\}$. We associate two rational numbers to it,

$$r(d) = \sum_{k=1}^{n} \frac{d_k}{2^k} = \frac{j}{2^n}, \quad j \in \mathbf{N} \tag{3.31}$$

and

$$u(d) = \sum_{k=1}^{n} d_k, \tag{3.32}$$

that is, u is the number of ones in the binary expansion of j.

The following lemma will be needed in our analysis.

Lemma 3.7.1. *Let $d = [d_1, ..., d_n],\ n \leq m$ and $r(d) = \frac{j}{2^n}\,,\ j < 2^n$. Then*

$$\mathcal{T}\big(r(d) + \frac{1}{2^m}\big) - \mathcal{T}\big(r(d)\big) = \frac{m - 2u(d)}{2^m}. \tag{3.33}$$

The proof can be found in [1, p. 19].

The connection between the de Casteljau algorithm and the Takagi curve rests on the following lemma.

Lemma 3.7.2. *Let $d = [d_1, d_2, ..., d_n], d_k \in \{0, 1\}$ be a finite binary code and $r(d)$ be a rational number associated to it as in equations (3.31) and (3.32). Let $\mathbf{L}, \mathbf{R}$ be the de Casteljau subdivision matrices (3.28) with control points in homogenous coordinates $\mathbf{p}_0 = (0, 1)^\top, \mathbf{p}_1 = (1, 1)^\top$, and complex parameter $t = \frac{1}{2} + iy$, $|y|$ small enough. Define matrix $\mathbf{M}(d)$ as*

$$\mathbf{M}(d) = \mathbf{M}(d_1)\mathbf{M}(d_2)\ldots\mathbf{M}(d_n),$$

where $\mathbf{M}(0) = \mathbf{L}$ *and* $\mathbf{M}(1) = \mathbf{R}$.
Further, let $s(d)$ *be the first coordinate of* $\mathbf{M}(d)\mathbf{p}_0$. *Then*

$$s(d) = r(d) + 2\,i\,y\,\mathcal{T}(r(d)) + O(y^2). \tag{3.34}$$

Proof. We proof the equation (3.34) by mathematical induction on the length n of d. The de Casteljau complex subdivision matrices for the corresponding IFS (3.28) are

$$\begin{aligned} \mathbf{L} &= \begin{pmatrix} t & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} + iy & 0 \\ 0 & 1 \end{pmatrix}, \\ \mathbf{R} &= \begin{pmatrix} 1-t & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} - iy & \frac{1}{2} + iy \\ 0 & 1 \end{pmatrix}. \end{aligned}$$

1. The base case $n = 1$**:** Either $d = [0]$ or $d = [1]$, for $d = [0]$ we have

$$\begin{pmatrix} s([0]) \\ 1 \end{pmatrix} = \mathbf{L} \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

which yields

$$0 = 0 + 2\,i\,y\,\mathcal{T}(0),$$

that is, $0 = 0$ and equation (3.34) holds. If $d = [1]$, we obtain

$$\begin{pmatrix} s([1]) \\ 1 \end{pmatrix} = \mathbf{R} \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

which yields

$$\frac{1}{2} + iy = \frac{1}{2} + 2\,i\,y\,\frac{1}{2},$$

hence equation (3.34) holds as well.

2. The induction hypothesis for $n = k$**:** Assume that for $d = [d_1, \ldots, d_k]$, equation (3.34) holds. Denote $r_k = r(d)$, $s_k = s(d)$, and $\mathbf{M}_k = \mathbf{M}(d)$. Equation (3.34) can be rewritten as

$$s_k = r_k + 2\,i\,y\,\mathcal{T}(r_k) + O(y^2). \tag{3.35}$$

Further, denote u_k the number of ones in the vector $d = [d_1, \ldots, d_k]$, that is,

$$u_k = \sum_{i=1}^{k} d_i.$$

Notice that the matrix $\mathbf{M}_k$ has always the following form:

$$\mathbf{M}_k = \begin{pmatrix} \left(\frac{1}{2} - iy\right)^{u_k} \left(\frac{1}{2} + iy\right)^{k-u_k} & s_k \\ 0 & 1 \end{pmatrix}. \tag{3.36}$$

3. The inductive statement for $n = k+1$: We prove that if the equation (3.35) holds, then for $d' = [d_1, \ldots, d_k, d_{k+1}]$ the equation (3.34) is also true. We have

$$\begin{aligned} r(d') &= \sum_{i=1}^{k+1} \frac{d_i}{2^i} = r_k + \frac{d_{k+1}}{2^{k+1}} \\ \mathbf{M}(d') &= \mathbf{M}_k \mathbf{M}(d_{k+1}). \end{aligned}$$

Either $d' = [d_1, \ldots, d_k, 0]$ or $d' = [d_1, \ldots, d_k, 1]$. The case $d = [d_1, \ldots, d_k, 0]$ is immediate, because $r(d') = r_k$, $\mathbf{M}(0) = \mathbf{L}$, and we can write

$$\begin{pmatrix} s(d') \\ 1 \end{pmatrix} = \mathbf{M}_k \begin{pmatrix} \frac{1}{2} + iy & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \mathbf{M}_k \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} s_k \\ 1 \end{pmatrix},$$

that is, $s(d')$ for $d' = [d_1, \ldots, d_k, 0]$ is equal to s_k and the statement (3.34) holds.

The proof for the case $d' = [d_1, \ldots, d_k, 1]$ is not so straightforward. For $d_{k+1} = 1$, $\mathbf{M}(1) = \mathbf{R}$, and we have

$$r(d') = r_k + \frac{1}{2^{k+1}} \tag{3.37}$$

and

$$\mathbf{M}(d') = \mathbf{M}_k \mathbf{R} = \begin{pmatrix} \left(\frac{1}{2} - iy\right)^{u_k+1} \left(\frac{1}{2} + iy\right)^{k-u_k} & \left(\frac{1}{2} - iy\right)^{u} \left(\frac{1}{2} + iy\right)^{k+1-u} + s_k \\ 0 & 1 \end{pmatrix},$$

which yields

$$\begin{pmatrix} s(d') \\ 1 \end{pmatrix} = \mathbf{M}(d') \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \left(\frac{1}{2} - iy\right)^{u_k} \left(\frac{1}{2} + iy\right)^{k+1-u_k} + s_k \\ 1 \end{pmatrix}.$$

Hence $s(d')$ satisfies the equation

$$s(d') = \left(\frac{1}{2} - iy\right)^{u_k} \left(\frac{1}{2} + iy\right)^{k+1-u_k} + s_k. \tag{3.38}$$

We must prove that

$$s(d') = r(d') + 2\,i\,y\,\mathcal{T}(r(d')) + O(y^2). \tag{3.39}$$

From (3.37), $r(d') = r_k + \frac{1}{2^{k+1}}$. We also substitute $s(d')$ in equation (3.39) by expression in (3.38). Thus equation (3.39) becomes

$$\left(\frac{1}{2} - iy\right)^{u_k} \left(\frac{1}{2} + iy\right)^{k+1-u_k} + s_k = r_k + \frac{1}{2^{k+1}} + 2\,i\,y\,\mathcal{T}\left(r_k + \frac{1}{2^{k+1}}\right) + O(y^2).$$

Furthermore, from induction hypothesis (3.35), $s_k = r_k + 2\,i\,y\,\mathcal{T}(r_k)$. After substitution we obtain

$$\left(\frac{1}{2} - iy\right)^{u_k} \left(\frac{1}{2} + iy\right)^{k+1-u_k} + r_k + 2\,i\,y\,\mathcal{T}(r_k) = r_k + \frac{1}{2^{k+1}} + 2\,i\,y\,\mathcal{T}\left(r_k + \frac{1}{2^{k+1}}\right) + O(y^2).$$

In the next step we use the binomial theorem to express

$$\left(\frac{1}{2}-iy\right)^{u_k}\left(\frac{1}{2}+iy\right)^{k+1-u_k}=\frac{1}{2^{k+1}}+\frac{k+1-2u_k}{2^k}iy+O(y^2),$$

and we substitute this expression in the previous equation, that is,

$$\frac{1}{2^{k+1}}+\frac{k+1-2u_k}{2^k}iy+O(y^2)+2\,i\,y\,\mathcal{T}(r_k)=\frac{1}{2^{k+1}}+2\,i\,y\,\mathcal{T}\left(r_k+\frac{1}{2^{k+1}}\right)+O(y^2).$$

After some manipulation, we obtain

$$\frac{k+1-2u_k}{2^{k+1}}iy+i\,y\,\mathcal{T}(r_k)=i\,y\,\mathcal{T}\left(r_k+\frac{1}{2^{k+1}}\right)+O(y^2).$$

From Lemma 3.7.1, we know that the following equation holds:

$$\frac{k+1-2u}{2^{k+1}}+\mathcal{T}(r_k)=\mathcal{T}\left(r_k+\frac{1}{2^{k+1}}\right),$$

which completes the proof.

□

In order to formulate the statement about approximation of the Takagi curve, we define the following scaling map

$$g:\mathbb{C}\times\mathbb{R}\to\mathbb{C}\qquad g(z,y)=\mathrm{Re}(z)+y\,i\,\mathrm{Im}(z). \tag{3.40}$$

Theorem 3.7.3. *Let $A^*(y)\in\mathbb{C}$ be the attractor of the de Casteljau IFS (3.28) for the Bézier curve with control points 0 and 1, and complex subdivision parameter $t=\frac{1}{2}+iy$, as in Lemma 3.7.2. Let $T=\{x+i\mathcal{T}(x)\mid x\in[0,1]\}$ be the graph of the Takagi function. Then the set*

$$A^*=\lim_{y\to 0}g\big(A^*(y),\frac{1}{2y}\big) \tag{3.41}$$

contains the set T.

Proof. According to Theorem 3.7.2, the curve A^* is connected. The Takagi curve is continuous. Therefore it is sufficient to prove pointwise convergence of the complex Bézier curve A^* to the Takagi curve T on a dense set of points. The control points 0 and 1 are both in A^* and T. Also the points $s(d)$ from equation (3.34) are all in $A^*(y)$ (as we have proven in Lemma 3.7.2), thus points $g(s(d),\frac{1}{2y})$ are in A^*. As d runs through all finite binary codes, the points $g(s(d),\frac{1}{2y})$ are dense in A^*, and A^* approximates T in infinitely many points.

□

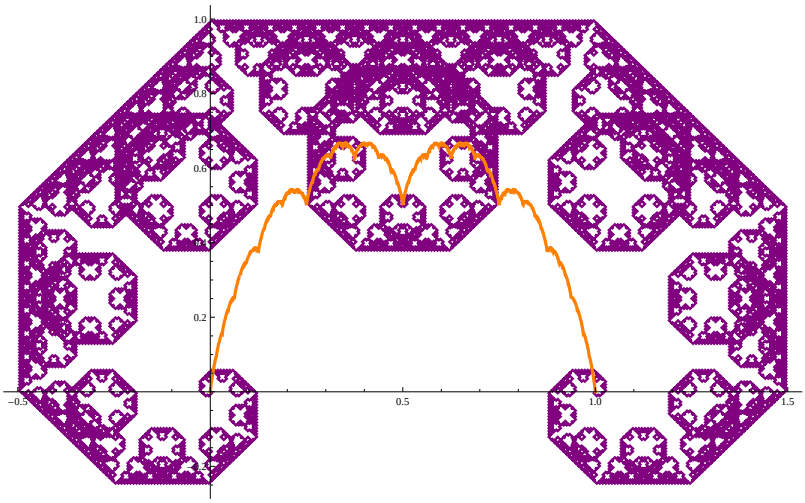

Figure 3.6: The purple curve is $g(A^*(y), \frac{1}{2y})$, which for $y = \frac{1}{2}$ becomes the Lévy C curve. The orange curve is the graph of Takagi curve T.

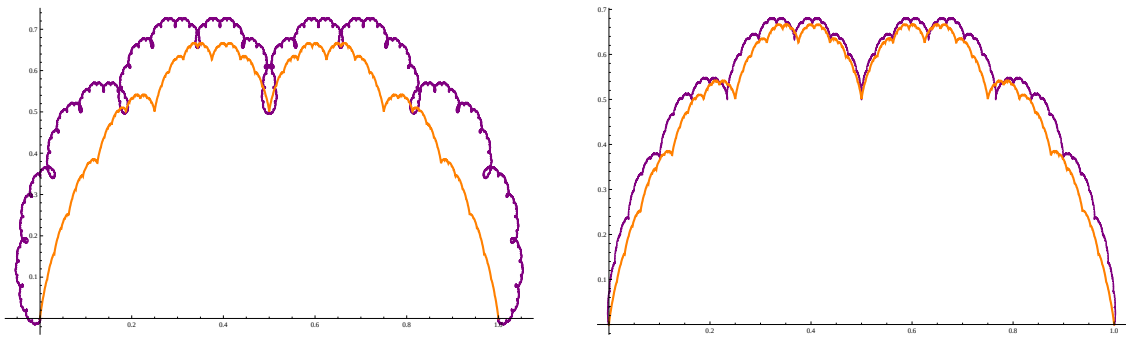

Figure 3.7: Left figure shows $g(A^*(y), \frac{1}{2y})$ (purple) for $y = \frac{1}{4}$, on the right is $y = \frac{1}{8}$, the orange curve is the Takagi curve.

Example 3.7.2. The properties of the attractor $A^*(y)$ depend strongly on y. In Figures 3.6–3.8 we plot $g(A^*(y), \frac{1}{2y})$ for various values of y, and T for comparison. In all cases, the algorithm is iterated 15 times.

For $y = \frac{1}{4}$, the curve $A^*(y)$ still creates crunodes. On the other hand, $A^*(y)$ seems to start creating cusps instead of crunodes for $y = \frac{1}{8}$.

Remark. The experimental evidence suggests that $A^* = T$, but in order to prove this statement we would have to establish uniform convergence. We consider the following.
Let $A^*(y) \in \mathbb{C}$ be the attractor of the de Casteljau IFS (3.28) for the Bézier curve with control points 0 and 1, and complex subdivision parameter $t = \frac{1}{2} + iy$. Let $T^*(y) = \{x + 2\,i\,y\,\mathcal{T}(x) \mid x \in [0,1]\}$. Then, as $y \to 0$,

$$d_H(A^*(y), T^*(y)) = O(y^2), \tag{3.42}$$

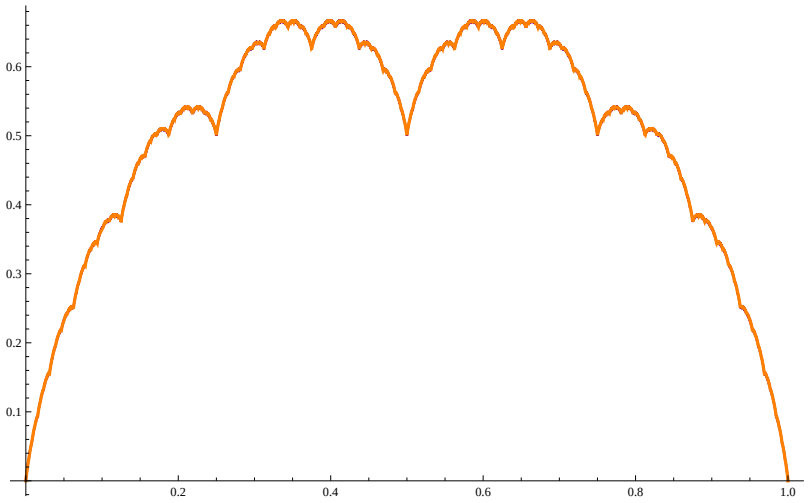

Figure 3.8: Here $y = 2^{-10}$ and $g(A^*(y), \frac{1}{2y})$ is incident with T. That is, we can only see T, which is the orange curve on the picture.

where d_H is the Hausdorff distance in $\mathbb{C}$. The error term $O(y^2)$ is probably limited, its real part dominates, but it is not clear where the maximum error is achieved.

In Conjecture 3.7.1, we argue that the Takagi curve appears to be present in every Bézier curve where the subdivision parameter has a vanishing imaginary part.

Conjecture 3.7.1. *Let $\mathbf{c} = \mathbf{c}(\alpha)$ be a Bézier curve with complex control points $\mathbf{P}$. Let $A^*(y)$ be the attractor of the de Casteljau IFS (3.28) with control points $\mathbf{P}$ and complex subdivision parameter $t = \frac{1}{2} + iy$, as in Theorem 3.7.3. Then*

$$d_H\big(A^*(y), \mathbf{c}(g(T, 2y))\big) = O(y), \qquad y \to 0, \tag{3.43}$$

where d_H is the Hausdorff distance in $\mathbb{C}$.

We illustrate the above conjecture with the following example.

Example 3.7.3. We construct the attractor $A^*(y)$ for the de Casteljau IFS (3.28) with complex control points $\mathbf{P} = \{\mathbf{p}_0 = 0, \mathbf{p}_1 = \frac{1}{2} + 2i, \mathbf{p}_2 = \frac{1}{2}\}$ and complex subdivision parameter $t = \frac{1}{2} + iy$, in this example $y = 10^{-2}$. We denote $T^*(y) = g(T, 2y)$. We construct $\mathbf{c}(T^*(y))$, where $\mathbf{c}(\alpha)$ is a Bézier curve with the same control points $\mathbf{P}$. That is,

$$\mathbf{c}(T^*(y)) = \sum_{k=0}^{2} \binom{2}{k} \big(x + 2\,i\,y\,\mathcal{T}(x)\big)^k \big(1 - x - 2\,i\,y\,\mathcal{T}(x)\big)^{2-k} \mathbf{p}_k, \; x \in [0, 1]. \tag{3.44}$$

The attractor $A^*(y)$ and the Bézier curve $\mathbf{c}(T^*(y))$ are shown in Figure 3.9.

In Figure 3.9 we can not quite see any difference between $A^*(y)$ and $\mathbf{c}(T^*)$. Even if we look on a small scaled detail, we will not see any difference after sufficiently

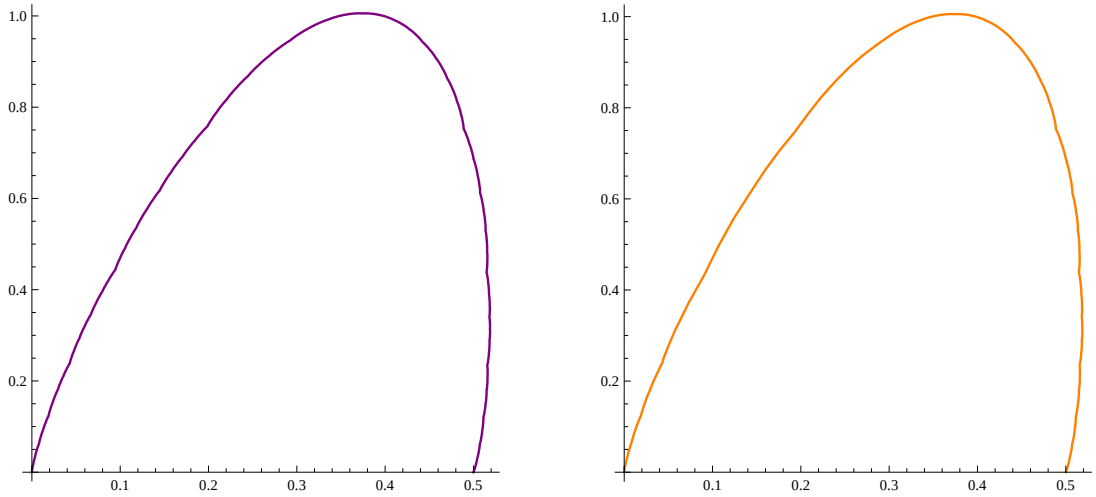

Figure 3.9: Left figure shows points $A^*(y)$ after 12 iterations, on the right is curve $\mathbf{c}(T^*)$.

many iterations of $A^*(y)$. In Figure 3.10 we plot a small segment of $\mathbf{c}(T^*)$, as it is expressed in equation (3.44), for $x \in [0, \frac{1}{10}]$, and for comparison $A^*(y)$ after several iterations. Obviously, $A^*(y)$ approximates $\mathbf{c}(T^*)$ quite well.

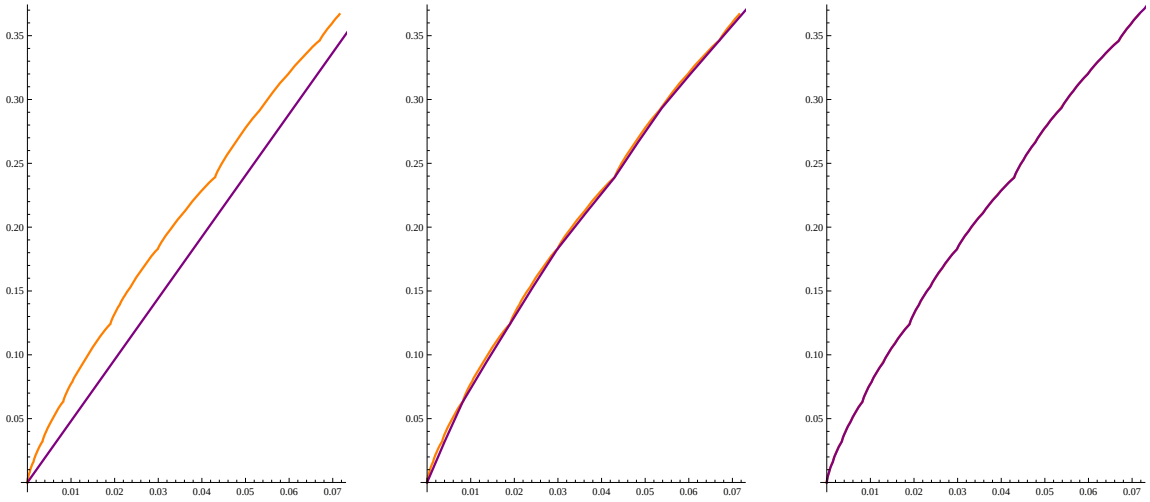

Figure 3.10: The orange curve is $\mathbf{c}(T^*)$, the purple curve is $A^*(y)$. In the left figure is $A^*(y)$ after two iterations, in the middle is $A^*(y)$ after six iterations, and on the right is after ten iterations.

Chapter 4

Summary

The aim of the thesis is to show some connections between subdivision algorithms of geometric modeling and iterated function systems of fractal theory. In addition, we provide rigorous treatment of material that has appeared recently in the literature.

We start with Bézier and B-spline curves and describe subdivision methods for the construction of these curves. These include the Casteljau algorithm, our main interest, and the four–point subdivision method, which generates fractal curves for some tension parameter.

After introducing some basic concepts of dynamical systems, we deal with metric spaces, the Banach fixed point theorem, hyperbolic iterated function systems (IFS), and the notion of dimension of a set. Fractals are then introduced as attractors of IFS, whose box dimension is not an integer.

Then, we deal with affine transformations and IFS for subdivision curves, which turn out to be IFS consisting of affine transformations. One of the main contributions of the thesis is a proof that an IFS for subdivision curves has unique fixed point (Section 3.6). In order to do so, we use the fact that given submatrices of subdivision matrices are eventually contractive (Theorem 3.5.1).

IFS for complex Bézier curves give rise to a new way of generating fractals. We prove that the curves generated by IFS for complex Bézier curve with control points 0 and 1 are connected (Theorem 3.7.2), and then show that a complex Bézier curve approximates the Takagi fractal curve (Theorem 3.7.3), in a suitable limit. We conjecture that the Takagi curve is present in every Bézier curve (of higher degree as well), if the subdivision parameter has vanishing imaginary part and the real part is equal to $\frac{1}{2}$.

The fractals generated by the complex de Casteljau subdivision algorithm have control points, with which the shape of a curve can be controlled. We illustrate this process for the case of the Takagi curve (Example 3.7.3), thereby generating fractals with control points.

A natural question is whether this fact finds applications in geometric modeling. More explicitly, we ask:

- Is there some practical use of IFSs for subdivision curves and surfaces?
- What are the advantages of using IFSs instead of classical subdivision algorithms or parametrization?
- Can we create IFS for complex Bézier surfaces?
- Can we use these fractals with control points for geometric modeling, e.g., modeling of mountain ranges?
- Does this method allow us to unify modeling of fractal-like objects and smooth surfaces?

When rendering attractors of IFSs for subdivision curves, we used a deterministic algorithm, based on recursive computation of a sequence of sets, starting from an initial set of points. There is an alternative random iteration algorithm, which is worth exploring. It allows one to define addresses for points on fractals. These are of great interest, because they encode a Bézier curve as a map of the unit interval into the (possibly complex) plane or space. We could also use random algorithms to generate fractals with "shades", which, mathematically, would correspond to non-trivial measures.

Further questions arising in this context are:

- Is the random iteration algorithm more effective than the classical subdivision algorithm for subdivision curves/surfaces?
- What fractal curves and surfaces can be generated by the complex de Casteljau algorithm?

In the limited space of a diploma thesis, we could not answer some of the question that appeared during our work. These questions might be interesting topics for further research.

Appendix A

A short review of related literature

Deterministic splines and stochastic fractals are both techniques for free-form shape modeling. The paper [32] deals with developing constrained fractals, a hybrid of splines and fractals, which can be used to model realistic terrains. Mathematical models for generating free-form shapes can be classified as deterministic or stochastic. Splines offer precise shape control through adjusting their control points. But splines are smooth and therefore less useful for modeling natural objects such as mountain ranges. Contrariwise, stochastic fractals provide fine detail modeling, therefore can be used for modeling of complex natural objects. On the other hand, the classical approach for their modeling offers only limited control over the shape.

Constrained fractals is a technique which provides both detail and control. Authors define controlled–continuity splines as splines with rigidity and tension parameter functions, which provide local control over continuity and smoothness of a spline. Controlled–continuity splines form the deterministic component of constrained fractals. For instance, some of the approaches in stochastic subdivision techniques are a random perturbing of points at each subdivision step, or a refinement of random midpoint displacement.

Authors propose a multi-resolution stochastic relaxation algorithm. Their physical model with energy functions presents control over the shape by determining the local energy. A constrained fractal is a physically–based model using the energy minimization principles (this is the deterministic component), and bombarding the spline shape with point masses impacting at random velocities (this is the stochastic component).

Author of [25] provides a brief summary of spline (mostly B–splines) and fractal function theory, and shows some of the relationship between splines and fractal functions.

Fractal functions can be constructed via fixed points of Read-Bajraktarevic operators $\mathcal{T}$ on a compact subset of $\mathbb{R}^n$. Such a contractive operator has its unique fixed point $\mathfrak{F}$, which is called an ($\mathbb{R}$-valued) fractal function. In short, the Read-Bajractarevic operators consist of real polynomials p_i^M and contractive mappings

u_i. The graphs of fractal functions may be fractal sets with non-integral Hausdorff-Besicovitch dimension, but there also exist fixed points that are of class $C^m, m \in \mathbb{N}$. Such a fractal function depends on the set of polynomials p_i^M. Fractal analogs of B-spline functions are fractal functions, where the polynomials p_i are B-spline basis functions.

Author also introduces Besov space $B_q^s(L^p)$ and Triebel-Lizorkin space $F_q^s(L^p)$, the theory of which has rich applications to partial differential equations and approximation theory. Furthermore, it is proved that under certain conditions, a fractal function $\mathfrak{F}$ is an element of $F_q^s(L^p)$ and $B_q^s(L^p)$.

Fractals are attractors, fixed points of iterated function systems. R. Goldman in [13] shows on several examples that Bézier curves are attractors of the de Casteljau subdivision algorithm and presents a new algorithm for rendering Bézier curves.

Fractals from IFS are self-similar curves, they are generated from scaled down copies of themselves. Each segment of a Bézier curve is itself a Bézier curve, from this perspective they are self-similar. Therefore, according to the author, Bézier curves are also fractal curves. The de Casteljau subdivision algorithm is used to split a Bézier curve into two Bézier segments. R. Goldman rewrites the de Casteljau subdivision as two matrices $\mathbf{L}$, $\mathbf{M}$. Starting with the original Bézier control points, and applying these matrices repeatedly generates a sequence of control polygons that converge to the Bézier curve. In order to converge to the Bézier curve, we have to start with the Bézier control polygon, which is the main difference between this iteration and IFS for fractals.

Next, author resorts to matrices $\mathbf{L}_p$, $\mathbf{M}_p$, which represent, according to the author, contractive mappings. Thus, the set $\{\mathbf{L}_p, \mathbf{M}_p\}$ is an iterated function system. Starting the iteration with any compact set, the process will converge to the Bézier curve.

Subdivision schemes generate self-similar curves and surfaces. Authors in [15] derive the IFS for many different subdivision curves and surfaces without extraordinary vertices, such as B-splines and piecewise Bézier subdivision surfaces. They also show how to build a subdivision schemes to generate self-similar fractals such as the Siérpinski gasket, and present fractals with control points. Any curve generated by an arbitrary stationary subdivision scheme can be generated by an IFS. Authors show how this method of generating splines by IFS can handle end-point conditions if the knot spacing is non-uniform. The IFS for an arbitrary spline consist of functions f_i, which are derived from control points and subdivision matrices. Authors provide a method for creating such an IFS for many subdivision curves and surfaces.

Authors of [16] deal with the so called "inverse problem" of fractal theory, i.e., with the determination of a model for approximating natural data, creation of a geometrical model for a given natural object. The model has to be simple, compact,

and easy to manipulate. Among models satisfying these demands is fractal interpolation function model (FIF), which was introduced by Barnsley in [2]. This article presents an approximation method based on a model that generalizes IFS model and free form curves, and synthesize both.

The quality of the approximation is given by an evaluation criterion, which is usually a given distance between the two comparable objects. Mostly Hausdorff distance between two sets is used in fractal modeling. Hausdorff distance is difficult to compute and takes some constraints on both models. In order to measure distance between two curves, the curves have to have unified representation (e.g., parametric curve).

Authors therefore propose a new distance χ^2 between two functions G_Q, G'_Q, which is based on the parametrization of the curves to compare. This new measure allows to compare two curves Q, Q' which can be obtained by different sampling functions. This new criterion is used to approximate natural rough curves using the presented fractal model, more specifically, fractal free form curve model.

Projected IFS attractors are then defined in barycentic coordinate system. The iteration semigroup constituted by matrices with barycentric columns allows this model resorting to projected IFS attractors in order to unify the IFS model and the free form representation used in CAGD [1]. Authors test their method on a model of mountains.

Fractal properties of selected interpolatory subdivision schemes and their application in fractal generation are studied in [17]. Authors are concerned with fractal properties of the four-point binary and three-point ternary interpolatory subdivision scheme. Depending on the tension parameter ω of the four-point scheme, the subdivision scheme can generate fractal curves. Authors analyze this property and prove that the limit curves for some specific values of ω are fractal curves.

[1] Computer Aided Geometric Design

Bibliography

[1] ALLAART, P. C., KAWAMURA, K. *The Takagi function: a survey*. University of North Texas, Department of Mathematics, 2011.

[2] BARNSLEY, M. *Fractals everywhere*. 2nd edition. San Francisco: Morgan Kaufmann, 1993. ISBN 0-12-079069-6.

[3] BASTL, B. *Applications of Geometry 1*. Lecture notes. University of West Bohemia, Faculty of Applied Sciences, 2011. [in Czech].

[4] CONRAD, K. *The Contraction Mapping Theorem*. Expository paper. University of Connecticut, College of Liberal Arts and Sciences, Department of Mathematics.

[5] DEVANEY, R. L. *An Introduction to Chaotic Dynamical Systems*. 2nd edition. Boulder (Colorado USA): Westview Press, 2003. ISBN 978-0813340852.

[6] DYN, N. Subdivision schemes in computer-aided geometric design. In *Advances in numerical analysis, vol.2*. Edit. by W. Light. Clarendon Press, 1992. p. 36-104.

[7] DYN, N., GREGORY J. A., LEVIN, D. Analysis of uniform binary subdivision schemes for curve design. Maths Technical Papers, Brunel University. September 1988, p. 1-28. [online], available: http://bura.brunel.ac.uk/handle/2438/2080 [accessed 26 Jan 2012]

[8] EDGAR, G. A. *Classics on Fractals*. Edited by G. A. Edgar. Westview Press, 2004. ISBN 0-8133-4153-1.

[9] FALCONER, K. *Fractal Geometry: Mathematical Foundations and Application*. 2nd edition. Chichester (UK): John Willey and Sons, 2003. ISBN 0-470-84861-8.

[10] FARIN, G. *Curves and Surfaces for CAGD: A Practical Guide*. 5th edition. San Francisco: Morgan Kaufmann, 2002. ISBN 1-55860-737-4.

[11] FARIN, G., HOSCHEK, J., KIM, M.-S. *Handbook of Computer Aided Geometric Design*. 1st edition. Amsterodam: North-Holland, 2002. ISBN 0-444-51104-0.

[12] FISHER, Y., JACOBS, E. W., BOSS, R. D. *Iterated Transform Image Compression*. San Diego: Naval Ocean Systems Center, 1991.

[13] GOLDMAN, R. The Fractal Nature of Bézier Curves. In *Proceedings of the Geometric Modeling and Processing 2004 (GMP'04)*. IEEE Computer Society. ISBN 0-7695-2078-2/04.

[14] GOLDMAN R. The Marriage of Fractals and Splines: Fractals with Control Points, Splines as Attractors. *Google Tech Talks* [video online], available: http://www.youtube.com/watch?v=jEjZl2nqcAU [accessed 6 Apr 2011].

[15] GOLDMAN, R. LEVIN, D. SCHAEFER, S. Subdivision Schemes and Attractors. In *Eurographics Symposium on Geometry Processing 2005*. Edit. by M. Desbrun, H. Pottmannn. The Eurographics Association, 2005.

[16] GUERIN E., TOSAN E., BASKURT A. A Fractal Approximation of Curves. In *Fractals*, Vol. 9, No. 1. World Scientific Publishing Company, 2001.

[17] HONGCHAN ZHENG, ZHENGLIN YE, YOUMING LEI, XIAODUNG LIU. Fractal properties of interpolatory subdivision schemes and their application in fractal generation. In *Chaos, Solitons and Fractals*. Volume 32, Issue 1, April 2007. p. 113-123.

[18] HUTCHINSON, J. E. Fractals and self similarity. In *Indiana University Mathematics Journal*. Volume 30, No. 5., 1981

[19] JEŽEK, F. *Geometrické a počítačové modelování*. Lecture notes. University of West Bohemia in Plzen, Faculty of Applied Sciences, Department of Mathematics, 2009. [in Czech].

[20] KOMINEK, J. *Convergence of Fractal Encoded Images*. University of Waterloo, Department of Applied Mathematics.

[21] KREYSZIG, E. *Introductory Functional Analysis with Applications*. 1st edition. New York: John Willey and Sons, 1989. ISBN 0-471-50459-9.

[22] LANE, J. M., RIESENFELD, R. F. A Theoretical Development for the Computer Generation and Display of Piecewise Polynomial Surfaces. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Volume PAMI-2, No. 1, January 1980. p. 35-46.

[23] LAURENT,M. Matrix completion problems. In *The Encyclopedia of Optimazation*. Volume III. Editors: C. A. Floudas, P. M. Pardalos. Kluwer, 2001. p. 221-229.

[24] MASSOPUST, P. *Interpolation and Approximation with Splines and Fractals*. 1st edition. Oxford: Oxford Press, 2010. 336 p. ISBN 0-19-533654-2.

[25] MASSOPUST, P. Splines, Fractal Functions, and Besov and Triebel-Lizorkin Spaces. In *Fractals in Engineering*. Edit. by Jacques Levy-Vehel, Evelyne Lutton. Springer London, 2005. p. 21-32.

[26] SAMAVATI, F. F. *Advanced Geometric Modeling*. Lecture notes. University of Calgary, Department of Computer Science. [online], available: http://pages.cpsc.ucalgary.ca/ samavati/cpsc789/slides/ [accessed 17 Feb 2012].

[27] SCHRÖDER, P. ZORIN, D. Subdivision for modeling and animation. Course notes, Siggraph, 2000.

[28] SCHRÖDER, P. *Topics in Geometric Modeling*. Lecture notes. California Institute of Technology, Department of Computing and Mathematical Sciences, 2006.

[29] SHENE, C.-K. *Introduction to Computing with Geometry Notes*. Lecture notes. Michigan Technological University, Department of Computer Science. [online], available: http://www.cs.mtu.edu/ shene/COURSES/cs3621/NOTES/ [accessed 19 Jan 2012].

[30] SEDERBERG, T. W. *An Introduction to B-Spline Curves*. Lecture notes. Brigham Young University, 2005. [online], available: http://tom.cs.byu.edu/ 455/bs.pdf [accessed 24 Jan 2012].

[31] SLABÁ, K. *Interpolatory subdivision schemes for curves*. Master thesis. University of West Bohemia in Plzen, Faculty of Applied Sciences, Department of Mathematics, 2011. [in Czech].

[32] SZELISKI R., TERZOPOULOS D. From Splines to Fractals. In *Computer Graphics*. Volum 23, Number 3, July 1989, p. 51-60.

[33] THEYS, J. *Joint Spectral Radius: theory and approximation*. PhD thesis. Université Catholique de Louvain, Center for Systems Engineering and Applied Mechanics, 2005.

[34] WISSTEIN, E. W. B-Spline. MathWorld - A Wolfram Web Resource. [online], available: http://mathworld.wolfram.com/B-Spline.html [accessed 20 Jan 2012].