

Bildgestaltung mit selbstähnlichen Fraktalen

Mathematische Grundlagen und
graphische Umsetzung

Maturaarbeit von Muriel Blum, 1MNb, November 2009
Betreut durch Herrn Heiner Rohner und Herrn Heinz Häsler
Gymnasium Bern-Neufeld

Abstract

Das Thema meiner Arbeit sind Fraktale, welche durch iterierte Funktionensysteme definiert sind. Solche Fraktale zeichnen sich besonders durch ihre Selbstähnlichkeit und ihre iterative Konstruktion aus. Bekannte Beispiele sind Cantor-Menge, Sierpinski-Dreieck, von Koch-Kurve und viele andere.

Was aus mathematischer Sicht interessiert, ist die Existenz einer eindeutigen Limesmenge für jedes iterierte Funktionensystem. Um diese nachzuweisen, werden verschiedene Begriffe und Sätze benötigt. Dazu gehören beispielsweise die Begriffe der Kontraktion, der kompakten Menge, des metrischen Raums, der Hausdorff-Metrik und der Banach'sche Fixpunktsatz.

Des Weiteren geht es in meiner Arbeit um die Darstellung der durch iterierte Funktionensysteme definierten Fraktale mit Hilfe des Computers. Es werden deshalb affin-lineare Transformationen, das „Chaosspiel“ sowie Masse und Verteilungen behandelt.

Meine praktische Arbeit war das Ausprobieren von verschiedenen Möglichkeiten, die Bilder zu gestalten. In diesem Zusammenhang wichtige Größen sind die Parameter des iterierten Funktionensystems, die Anzahl Iterationen, die Wahrscheinlichkeiten und natürlich die Werte für die Färbung. Da für Letztere Farbverläufe zusammengestellt werden müssen, was jedoch im RGB-Farbmodell nicht ideal ist, habe ich mich mit anderen Farbmodellen beschäftigt.

Das Endresultat meiner praktischen Arbeit ist ein selbst entworfenes Bild. Ich habe für dieses den Titel „Die vier Jahreszeiten“ gewählt.

Inhaltsverzeichnis

Abstract	i
Inhaltsverzeichnis	ii
Einleitung	iv
I Theorie	1
1 Fraktale – Wiedererkennen und Kennenlernen	1
1.1 Ein Beispiel aus der Natur	1
1.2 Einige Eigenschaften	1
2 Mathematische Grundlagen I: Abbildungen	4
2.1 Affin-lineare Transformationen der Ebene	5
2.1.1 Lineare Abbildungen	5
2.1.2 Lineare Abbildungen kombiniert mit Translationen	5
2.2 Distanzkontrahierende Abbildungen	6
2.3 Distanzkontrahierende affin-lineare Transformationen	6
2.4 IFS: Der Hutchinson-Operator \mathcal{F}	7
3 Mathematische Grundlagen II: Vollständige metrische Räume und Fixpunktsatz	7
3.1 Vollständige metrische Räume	7
3.1.1 Metrische Räume	8
3.1.2 Vollständigkeit von metrischen Räumen	11
3.2 Der vollständige metrische Raum $(\mathcal{C}(\mathbb{R}^2), h)$	11
3.2.1 Kompakte Mengen in \mathbb{R}^n	11
3.2.2 Die Hausdorff-Distanz	14
3.2.3 Der metrische Raum $(\mathcal{C}(X), h)$	15
3.2.4 Vollständigkeit von $(\mathcal{C}(\mathbb{R}^2), h)$	17
3.2.5 \mathcal{F} als Kontraktion auf $(\mathcal{C}(\mathbb{R}^2), h)$	18
3.3 Der Banach'sche Fixpunktsatz	19
3.4 Konsequenzen des Banach'schen Fixpunktsatzes	21
4 Mathematische Grundlagen III: Konzepte für die graphische Umsetzung	21
4.1 Deterministisch und zufällig iteriertes Funktionensystem	22
4.2 Masse und Verteilungen	24
4.2.1 Es werden nicht alle Punkte gleich häufig getroffen	24
4.2.2 Masse auf Teilmengen von \mathbb{R}^n	26
4.2.3 Verteilungen	27
4.2.4 Eine Methode zur Konstruktion einer Verteilung auf einer Teilmenge von \mathbb{R}^2	27
4.2.5 Mass beim Chaosspiel	28
5 Farbmodelle	28
5.1 Begriffe	28
5.2 Einige Farbmodelle	30

II	Anwendungen	34
6	Praktischer Teil	34
6.1	Verbindung mit der Theorie	34
6.2	Vorgehen	34
6.3	Gestaltungsmöglichkeiten	35
6.3.1	Anzahl Iterationen	35
6.3.2	Wahrscheinlichkeiten	35
6.3.3	Färbungen	36
7	Resultat	37
A	Anhang	41
A.1	IFS-Programm	41
A.2	C++-Programm IFS	42
A.3	C++-Programm Farbverlauf	44
	Literatur und Quellen	46
	Selbstständigkeitserklärung	49

Einleitung

Als Thema meiner Maturaarbeit habe ich Fraktale gewählt, weil diese mich faszinieren, seit wir sie im AM-Unterricht kennengelernt haben. Da Darstellungen von Fraktalen wegen deren besonderer Struktur sehr ästhetisch wirken, habe ich entschieden, mich mit der Bildgestaltung mit Fraktalen auseinanderzusetzen. Dabei wollte ich jedoch nicht nur Bilder kreieren, sondern auch deren mathematischen Hintergrund soweit als möglich verstehen.

Fraktale sind Gebilde oder geometrische Muster mit einem gewissen Grad an Selbstähnlichkeit. Sie enthalten Teile, welche Ähnlichkeiten mit dem Ganzen aufweisen. Solche Strukturen gibt es auch in der Natur, unter anderem bei Pflanzen. Ein Farnblatt beispielsweise besteht aus Teilen, welche mehr oder weniger als verkleinerte Exemplare des ganzen Blatts aufgefasst werden können. Ähnlich ist es beim Romanesco, denn dort besteht der Blütenstand aus massenhaft kleineren Versionen desselben.

Weil fraktale Strukturen in verschiedenen Zusammenhängen auftreten, sind sie ein interessantes Forschungsgebiet. Vielversprechend sind Fraktale beispielsweise für die Bildkomprimierung.

Meine Maturaarbeit behandelt Fraktale, welche durch sogenannte iterierte Funktionensysteme definiert sind. Ein iteriertes Funktionensystem ist eine Menge von Abbildungen mit bestimmten Eigenschaften. Betreffend das Farnblatt ist beispielsweise unter anderem diejenige Abbildung wesentlich, welche das ganze Blatt auf ein kleines Seitenblatt abbildet.

Es kann gezeigt werden, dass jedes iterierte Funktionensystem zu einer eindeutigen Menge führt. Der entsprechende Beweis und die für das Verständnis nötigen Begriffe und Sätze werden im theoretischen Teil der Arbeit behandelt.

Im praktischen Teil meiner Maturaarbeit habe ich mich mit der computerunterstützten Darstellung von Fraktalen beschäftigt. Dabei ist die Färbung ein wichtiger Aspekt. Weil das vom Computer verwendete RGB-Farbmodell nur mässig auf die menschliche Farbwahrnehmung abgestimmt ist, habe ich mich noch mit anderen Farbmodellen auseinandergesetzt, z. B. mit dem HSV-Farbmodell, das auf den für den Menschen natürlichen Grössen Farbton, Sättigung und Helligkeit basiert. Schliesslich habe ich ein Bild zum Thema „Die vier Jahreszeiten“ kreiert.

Für den theoretischen Teil meiner Arbeit habe ich unter anderem die Bücher [PEI/JÜR/SAU], [FAL] und [ZEI/PAG] beigezogen. Die Arbeitsinstrumente im praktischen Teil meiner Arbeit waren verschiedene Programme. Für das Experimentieren mit den Parametern der iterierten Funktionensysteme war das IFS-Programm [BAU/WEB] (s. Kap. A.2) von Beat Weber, Lehrer für Mathematik und Anwendungen der Mathematik am Gymnasium Köniz-Lerbermatt, überaus nützlich. Des Weiteren schrieb ich selbst einige C++-Programme (s. Kap. A.2 und Kap. A.3) und verwendete Adobe Photoshop ([ADO]) mit einem zusätzlichen Plugin ([ROS]).

In Kap. 1 werden verschiedene Fraktale vorgestellt und deren wichtigste Eigenschaften betrachtet.

Kap. 2 handelt von Abbildungen. Dabei geht es einerseits um die für den

praktischen Teil wichtigen affin-linearen Transformationen, andererseits um den Begriff des iterierten Funktionensystems.

Das Thema von Kap. 3 sind vollständige metrische Räume sowie der zentrale Satz in dieser Arbeit, der Banach'sche Fixpunktsatz.

In Kap. 4 geht es um die Visualisierung von Fraktalen mit Hilfe des Computers. Es werden Konzepte sowohl für die grundsätzliche Darstellung als auch für die Färbung vorgestellt.

Kap. 5 beinhaltet eine kurze Übersicht über verschiedene Farbmodelle, welche für das Zusammenstellen von Farbverläufen im praktischen Teil von Bedeutung sind.

In Kap. 6 wird aufgezeigt, was ich im praktischen Teil meiner Arbeit gemacht habe.

Kap. 7 schliesslich präsentiert das Resultat meiner Arbeit, das Bild mit dem Titel „Die vier Jahreszeiten“.

Neben dem gedruckten Anhang liegt dieser Arbeit eine CD als Anhang in digitaler Form bei. Darauf sind verschiedene Versionen des „Die vier Jahreszeiten“-Bildes, die verwendeten C++-Programme, IFS-Dateien, sämtliche in der Arbeit vorkommenden Abbildungen, die Internetquellen sowie die Arbeit im pdf-Format zu finden.

Diejenigen Bilder in der Arbeit, welche nicht mit einer Referenz versehen sind, habe ich selbst gezeichnet oder mit C++-Programmen hergestellt (s. beiliegende CD). Die Illustrationen im Anhang sind Screenshots.

Mein Dank geht an Heiner Rohner und Heinz Häsler für die gute Betreuung meiner Arbeit. Ebenfalls danke ich meiner Familie für die Unterstützung.

Teil I

Theorie

1 Fraktale – Wiedererkennen und Kennenlernen

Dieses erste Kapitel soll eine Vorstellung vom Wesen von Fraktalen vermitteln. Zu diesem Zweck betrachtet werden ein natürliches fraktales Gebilde sowie zwei berühmte Fraktale. Anhand dieser werden einige typische Eigenschaften von durch iterierte Funktionensysteme definierten Fraktalen erläutert.

1.1 Ein Beispiel aus der Natur

Ein sehr anschauliches Beispiel für fraktale Strukturen in der Natur ist die Farnpflanze (s. Abb. 1.1). Betrachtet man ein Blatt eines solchen Gewächses, fällt auf, dass die Form der einzelnen Teile viele Ähnlichkeiten mit derjenigen des ganzen Blatt hat.

Wenn man von einem Blatt die beiden untersten Seitenblätter abreisst, ist das verbleibende Stück immer noch wie das anfängliche Blatt, nur um einen bestimmten Faktor verkleinert. Auch die abgerissenen Stücke sähen fast wie das ganze Farnblatt aus, wenn man sie entsprechend vergrößern würde.

Diese gewisse Selbstähnlichkeit ist eine wichtige Eigenschaft von Fraktalen.



Abbildung 1.1: Ein Blatt einer Farnpflanze weist selbstähnliche Strukturen auf.

1.2 Einige Eigenschaften

Im Folgenden betrachten wir einige charakteristische Eigenschaften von Fraktalen und stützen uns dabei auf [FAL].

Fraktale haben ihren Namen von ihrer *gebrochenen Dimension* (lateinisch *fractus* = gebrochen). Es gibt zwar viele verschiedene Definitionen für die Dimension, doch bezüglich der meisten davon haben Fraktale eine nicht ganzzah-

lige Dimension. Dies ist ein Beispiel dafür, dass Fraktale zu irregulär sind, um mit traditionellen geometrischen Methoden beschrieben zu werden.

Viele Fraktale sind aus Teilen aufgebaut, die irgendwie ähnlich wie das Ganze aussehen. Dabei muss es sich nicht immer um Ähnlichkeit im strengen geometrischen Sinn handeln. Häufig besitzen Fraktale nur einen gewissen Grad an *Selbstähnlichkeit*.

Jedes Fraktal hat eine *Feinstruktur*. Dies bedeutet, dass ein Fraktal in allen Grössenordnungen Einzelheiten aufweist. Selbst in unendlicher Tiefe finden sich Details.

Viele Fraktale können mit einem *rekursiven Verfahren* erzeugt werden. Diese iterative Konstruktion ist geeignet, um eine graphische Annäherung an ein Fraktal zu machen. Jede weitere Iteration bedeutet eine grössere Genauigkeit.

Beispiel 1.1. Die *Cantor-Menge*¹ lässt sich folgendermassen konstruieren: Begonnen wird mit der Strecke $[0, 1]$. Daraus wird das mittlere Drittel entfernt. Im nächsten Schritt wird auch aus den beiden verbleibenden Strecken je das mittlere Drittel „herausgewischt“ und es bleiben vier Strecken der Länge $\frac{1}{9}$ übrig. Auch deren mittlere Drittel werden entfernt. So fortfahrend, erhält man einen immer besseren Eindruck von der Cantor-Menge (s. Abb. 1.2).

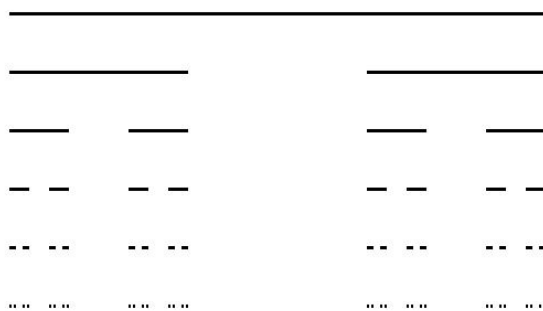


Abbildung 1.2: Die Konstruktion der Cantor-Menge. [1]

Die Konstruktion der Cantor-Menge kann man sich auch folgendermassen vorstellen: Die Startmenge, das Intervall $[0, 1]$, wird auf einen Drittel der Grösse verkleinert und dann einmal kopiert. Die beiden neuen Strecken werden so angeordnet, dass eine im Intervall $[0, \frac{1}{3}]$ liegt, die andere im Intervall $[\frac{2}{3}, 1]$. Diese Anordnung entspricht dem Bild der Startmenge unter den beiden Abbildungen $f_1(x) = \frac{1}{3} \cdot x$ und $f_2(x) = \frac{1}{3} \cdot x + \frac{2}{3}$. Werden diese beiden Abbildungen immer wieder angewendet, das heisst, wird die jeweils vorliegende Menge immer wieder in die beiden Intervalle abgebildet, ergibt sich ein immer besser werdendes Bild von der Cantor-Menge (s. Abb. 1.2).

Dieses Verfahren für die Konstruktion der Cantor-Menge ist ein *iteriertes Funktionensystem*, kurz IFS.

Ein iteriertes Funktionensystem funktioniert folgendermassen: Eine Startmenge A_0 wird mehrfach abgebildet. Die dabei entstehende Menge wird mit

¹Georg CANTOR (1845-1918)

A_1 bezeichnet. Sie ist die neue Startmenge, die wiederum mehrfach abgebildet wird, so dass A_2 entsteht. So fortfahrend, ergibt sich eine Folge von Mengen (A_n) . Die Limesmenge dieser Folge ist (häufig) ein Fraktal.

Im Falle der Cantor-Menge besteht A_n jeweils aus 2^n Intervallen der Länge 3^{-n} . Die Cantor-Menge A_∞ ist der Grenzwert der Folge von Mengen (A_n) .

Nun sollen die oben aufgeführten charakteristischen Eigenschaften von Fraktalen am Beispiel der Cantor-Menge betrachtet werden:

Die Struktur der Cantor-Menge lässt sich nicht einfach durch klassische Ausdrücke beschreiben, obwohl die eigentliche Definition der Cantor-Menge sehr einfach ist. Auch ihre Grösse ist nicht auf gewöhnliche Weise bestimmbar, denn bei jeder vernünftigen Definition hat die Cantor-Menge die Länge null. Dies ist natürlich nicht zufriedenstellend, vor allem, weil sie überabzählbar viele Elemente enthält. Vertraut ist man damit, dass eine Strecke ein eindimensionales Objekt ist, eine Fläche ein zweidimensionales usw. Einen Punkt kann man als nulldimensional betrachten. Die Cantor-Menge hat jedoch eine Selbstähnlichkeitsdimension von $\frac{\log(2)}{\log(3)} = 0.6309$, da sie etwas zwischen Punkt und Strecke ist.

Die Cantor-Menge ist (im strengen geometrischen Sinn) selbstähnlich. Die beiden Teile in den Intervallen $[0, \frac{1}{3}]$ und $[\frac{2}{3}, 1]$ sind je wie eine mit Faktor $\frac{1}{3}$ skalierte Cantor-Menge. Natürlich sind deswegen auch die in den vier Intervallen von A_2 liegenden Teile der Cantor-Menge ähnlich zu dieser, jedoch skaliert mit dem Faktor $\frac{1}{9}$. In dieser Weise geht es beliebig weiter, denn die Cantor-Menge enthält Kopien von sich selbst in unendlich vielen verschiedenen Grössen.

Durch diese Einzelheiten in beliebigen Grössenordnungen zeichnet sich die Feinstruktur aus.

Das Verfahren für die Konstruktion der Cantor-Menge ist rekursiv. Je mehr Schritte gemacht werden, desto besser ist die Annäherung an die Cantor-Menge.

Beispiel 1.2. Das *Sierpiński-Dreieck*² ist ebenfalls ein bekanntes Fraktal. Um einen Eindruck davon zu erhalten, startet man am besten mit einem gleichseitigen Dreieck und ersetzt dieses durch drei halb so grosse gleichseitige Dreiecke, so dass in der Mitte ein auf dem Kopf stehendes gleichseitiges Dreieck frei bleibt. Dieses Ersetzen wendet man danach auf alle drei entstandenen Dreiecke an, so dass die Figur aus neun Dreiecken besteht. Diese werden ebenfalls alle durch jeweils drei halb so grosse Dreiecke ersetzt. So fortfahrend, kommt man dem Sierpinski-Dreieck immer näher (s. Abb. 1.3).

Wiederum kann die Konstruktionsvorschrift als iteriertes Funktionensystem betrachtet werden. Im Falle des Sierpinski-Dreiecks umfasst das IFS drei Abbildungen. Alle sind Kontraktionen mit Faktor $\frac{1}{2}$. Eine beinhaltet eine Translation nach rechts, eine andere nach schräg oben. Konkret bedeutet dies

$$f_1 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \cdot x_1 \\ \frac{1}{2} \cdot x_2 \end{pmatrix}, \quad f_2 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \cdot x_1 + \frac{1}{2} \\ \frac{1}{2} \cdot x_2 \end{pmatrix}$$

und $f_3 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \cdot x_1 + \frac{1}{4} \\ \frac{1}{2} \cdot x_2 + \frac{\sqrt{3}}{4} \end{pmatrix}.$

Dieses IFS führt zu einem Sierpinski-Dreieck, dessen Grundlinie auf dem Intervall $[0, 1]$ liegt.

²Waclaw SIERPINSKI (1889-1969)

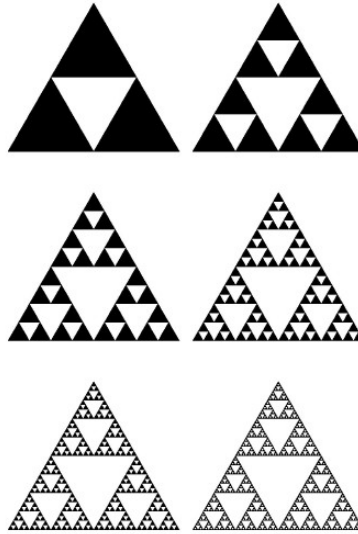


Abbildung 1.3: Die Konstruktion des Sierpinski-Dreiecks. [2]

Auch bei diesem Beispiel soll kurz ein Blick auf die typischen Eigenschaften eines durch ein iteriertes Funktionensystem definierten Fraktals geworfen werden:

Das Sierpinski-Dreieck ist nicht auf traditionelle Weise beschreibbar und geht dennoch auf eine sehr einfache Konstruktionsvorschrift zurück. Die Selbstähnlichkeitsdimension des Sierpinski-Dreiecks liegt bei $\frac{\log(3)}{\log(2)} = 1.585$.

Die Selbstähnlichkeit ist ebenfalls gegeben. Überall findet man kleine Kopien des ganzen Dreiecks.

Auch die Feinstruktur ist vorhanden, denn das Muster setzt sich in unendliche Tiefe fort.

Wiederum handelt es sich bei der Konstruktion um ein rekursives Verfahren.

Die charakteristischen Eigenschaften, welche anhand der beiden Beispiele betrachtet wurden, finden sich bei all den Fraktalen, um die es in dieser Arbeit geht. Zentral ist die rekursive Konstruktion mit Hilfe von iterierten Funktionensystemen.

In den nächsten Abschnitten geht es um die mathematischen Hintergründe der durch IFS definierten Fraktale. Die anhand der beiden Beispiele gemachte Feststellung, dass ein iteriertes Funktionensystem ein eindeutiges Fraktal definiert, soll bewiesen werden. Dazu müssen in den nächsten Kapiteln unter anderem der Begriff der Kontraktion, der kompakten Menge und des vollständigen metrischen Raumes eingeführt werden.

2 Mathematische Grundlagen I: Abbildungen

Weil Abbildungen die Grundlage für iterierte Funktionensysteme sind, sollen sie in diesem Kapitel näher betrachtet werden. Zuerst geht es um affin-lineare Abbildungen, danach um den Begriff der Kontraktion und schliesslich um den Hutchinson-Operator, das Kernstück eines IFS.

2.1 Affin-lineare Transformationen der Ebene

Werden IFS zwecks Gestaltung von zweidimensionalen Bildern betrachtet, interessieren vor allem die sogenannten affin-linearen Transformationen der Ebene. Diese sind nämlich als Matrix darstellbar und deshalb gut geeignet für den Computer.

In diesem Abschnitt folgen wir mehrheitlich [PEI/JÜR/SAU].

2.1.1 Lineare Abbildungen

Definition 2.1. Eine Vektorfunktion $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ heisst *lineare Abbildung*, wenn für $\vec{a}, \vec{b} \in \mathbb{R}^n$ und $\alpha \in \mathbb{R}$ die folgenden Bedingungen erfüllt sind:

$$\begin{aligned} \text{(i)} \quad & f(\vec{a} + \vec{b}) = f(\vec{a}) + f(\vec{b}) && \text{(additiv)} \\ \text{(ii)} \quad & f(\alpha \cdot \vec{a}) = \alpha \cdot f(\vec{a}) && \text{(homogen)} \end{aligned} \tag{2.1}$$

Lineare Abbildungen lassen sich mit Matrizen darstellen:

$$[f] = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

ist die *Darstellungsmatrix* der Abbildung $f : \mathbb{R}^2 \rightarrow \mathbb{R}^2$.

Umgekehrt lässt sich jede Matrix als Darstellungsmatrix einer linearen Abbildung auffassen.

Zwei lineare Abbildungen $f : \mathbb{R}^n \rightarrow \mathbb{R}^k$ und $g : \mathbb{R}^k \rightarrow \mathbb{R}^m$ können verknüpft werden.

Das Produkt der Matrizen von g und f ist die Matrix der verknüpften Abbildung $g \circ f$:

$$[g \circ f] = [g] \cdot [f].$$

Das Vorhandensein einer solchen Darstellungsmatrix der Verknüpfung $g \circ f$ zeigt, dass eine Verknüpfung linearer Abbildungen ebenfalls linear ist. Ebenfalls ist die Verknüpfung linearer Abbildungen wie die Matrix-Matrix-Multiplikation assoziativ, jedoch nicht kommutativ.

2.1.2 Lineare Abbildungen kombiniert mit Translationen

Eine lineare Abbildung kann mit einer Verschiebung ergänzt werden. Dann spricht man von einer *affin-linearen Abbildungen* oder von einer *affin-linearen Transformation*.

Diejenigen Fraktale, um deren Darstellung es in dieser Arbeit geht, sind durch IFS mit ausschliesslich affin-linearen Transformationen der Ebene definiert. Da eine lineare Abbildung als Matrix und eine Translation als Vektor dargestellt werden kann, ist eine affin-lineare Abbildung $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ durch sechs reelle Zahlen eindeutig festgelegt. Diese werden in folgender Form geschrieben:

$$\left(\begin{array}{cc|c} a & b & e \\ c & d & f \end{array} \right) \tag{2.2}$$

Wenn eine solche erweiterte Matrix auf einen Vektor $\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}$ angewendet wird, ist das Resultat

$$\begin{pmatrix} a \cdot a_1 + b \cdot a_2 + e \\ c \cdot a_1 + d \cdot a_2 + f \end{pmatrix}.$$

2.2 Distanzkontrahierende Abbildungen

Der Begriff der distanzkontrahierenden Abbildung ist wesentlich für IFS, wie sich in Kap. 2.4 (s. S. 7) zeigen wird.

Für die Definition von Kontraktionen folgen wir [FAL].

Definition 2.2. Eine Abbildung $f : X \rightarrow X$ heisst eine *Kontraktion* des metrischen Raumes³ (X, d) , wenn es eine Konstante $q \in \mathbb{R}$ mit $0 < q < 1$ so gibt, dass für alle $A, B \in X$ gilt

$$d(f(A), f(B)) \leq q \cdot d(A, B). \quad (2.3)$$

Dies bedeutet, dass die Distanz zweier Bilder immer kleiner sein muss als diejenige der entsprechenden Originale.

Die Abbildung f heisst *distanzkontrahierend* mit *Kontraktionsfaktor* q .

Wenn Gleichheit herrscht, das heisst, wenn $d(f(A), f(B)) = q \cdot d(A, B)$ gilt, dann handelt es sich bei f um eine *Ähnlichkeitsabbildung*.

Dies muss jedoch überhaupt nicht der Fall sein, denn distanzkontrahierend können auch ganz andere Abbildungen sein, beispielsweise solche, die Winkel verändern oder in verschiedene Richtungen mit unterschiedlichen Faktoren stauchen.

2.3 Distanzkontrahierende affin-lineare Transformationen

Da für IFS distanzkontrahierende Abbildungen wichtig sind, soll im Hinblick auf die Darstellung von durch IFS definierten Fraktalen am Computer geklärt werden, wann eine affin-lineare Transformation distanzkontrahierend ist.

Eine affin-lineare Abbildung ist, wie in Kap. 2.1.2 (s. S. 5) gesehen, eine mit einer Translation kombinierte lineare Abbildung. Da Translationen Kongruenzabbildungen sind, hat der Translationsvektor einer affin-linearen Abbildung keinen Einfluss darauf, ob diese distanzkontrahierend ist oder nicht. Es kommt allein auf die Matrix an.

Um ablesen zu können, ob es sich um eine Kontraktion handelt, schreibt man die Matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ gemäss [PEI/JÜR/SAU] in folgender Form:

$$\begin{pmatrix} r \cdot \cos(\eta) & -s \cdot \sin(\vartheta) \\ r \cdot \sin(\eta) & s \cdot \cos(\vartheta) \end{pmatrix} \quad (2.4)$$

mit $r = \sqrt{a^2 + c^2}$ und $\eta = \arccos(\frac{a}{\sqrt{a^2 + c^2}})$ sowie analog $s = \sqrt{b^2 + d^2}$ und $\vartheta = \arccos(\frac{d}{\sqrt{b^2 + d^2}})$.

Massgebend für die Skalierung sind r und s . Wenn beide kleiner als 1 sind, handelt es sich um eine distanzkontrahierende Abbildung.

Zudem sind in dieser Darstellungsform Drehungen, Spiegelungen usw. besser erkennbar. Beispielsweise legt $s = r$ und $\eta = \vartheta$ eine Drehstreckung mit Faktor r und Winkel η fest. Eine Spiegelung an der y -Achse ergibt sich bei $\eta = \pi$ und $\vartheta = 0$.

³Die Definition dieses Begriffes folgt (s. Definition 3.1 auf S. 8)

2.4 IFS: Der Hutchinson-Operator \mathcal{F}

Nun soll das Konzept des iterierten Funktionensystems betrachtet werden. Dabei ist der Hutchinson-Operator der zentrale Begriff.

Wir folgen [PEI/JÜR/SAU], [FAL] und [WIK1].

Formal ist ein iteriertes Funktionensystem eine endliche Menge von Kontraktionen eines vollständigen⁴ metrischen Raumes.

$$\{f_i : X \rightarrow X : i = 1, 2, \dots, N\} \text{ mit } N \in \mathbb{N}$$

ist ein iteriertes Funktionensystem, wenn alle f_i Kontraktionen des vollständigen metrischen Raumes X sind.

Diese N Abbildungen werden zu einem Operator zusammengefasst:

Definition 2.3. Der *Hutchinson-Operator*⁵ ist

$$\mathcal{F}(A) = f_1(A) \cup f_2(A) \cup \dots \cup f_N(A) = \bigcup_{i=1}^N f_i(A). \quad (2.5)$$

Der Hutchinson-Operator steht für die Gesamtheit der N Transformationen. Wird er angewendet, bedeutet dies, dass N kleine Kopien des eingegebenen Bildes zu einem neuen Bild zusammengefügt werden. Weil dann jedem Originalpunkt mehrere Bildpunkte zugeordnet werden, spricht man nicht von einer Abbildung, sondern von einem Operator.

Die Iteration des Hutchinson-Operators \mathcal{F} ist der Kern eines iterierten Funktionensystems. \mathcal{F}^n steht dabei für die n -te *Iterierte* von \mathcal{F} . Diese ist definiert durch $\mathcal{F}^0(A) = A$ und $\mathcal{F}^n(A) = \mathcal{F}(\mathcal{F}^{n-1}(A))$ für $n \geq 1$.

3 Mathematische Grundlagen II: Vollständige metrische Räume und Fixpunktsatz

In diesem Kapitel geht es um den Beweis der in Kap.1.2 (s. S. 1) gemachten Feststellung, dass ein IFS eine eindeutige Limesmenge besitzt, welche durch fortgesetzte Iteration des Hutchinson-Operators erreicht wird. Zuerst werden verschiedene Beispiele zum Konzept des metrischen Raumes betrachtet, dann geht es um den vollständigen metrischen Raum $(\mathcal{C}(\mathbb{R}^2), h)$ und schliesslich um den Banach'schen Fixpunktsatz.

3.1 Vollständige metrische Räume

Bei der Betrachtung der Beispiele 1.1 und 1.2 (s. S. 2 und S. 3) haben wir festgestellt, dass die Folge (A_n) mit $A_n = \mathcal{F}^n(A_0)$ zur Limesmenge A_∞ konvergiert, welche meist ein Fraktal ist. Damit jedoch über Konvergenz oder Grenzwerte von Folgen gesprochen werden kann, ist eine Metrik, das heisst eine Funktion, die den Abstand misst, nötig. Deshalb geht es in diesem Kapitel um metrische Räume und deren Vollständigkeit, eine Eigenschaft, welche für die Anwendung des Banach'schen Fixpunktsatzes benötigt wird.

Für die Definitionen und einige der Beispiele in diesem Kapitel orientieren wir uns an [ZEI/PAG] und [PEI/JÜR/SAU].

⁴Die Definition dieses Begriffes folgt (s. Definition 3.2 auf S. 11)

⁵John HUTCHINSON

3.1.1 Metrische Räume

Ein gut bekannter Raum ist die Ebene \mathbb{R}^2 . Darin gibt es zu jedem Paar von Punkten $a = (a_1|a_2) \in \mathbb{R}^2$ und $b = (b_1|b_2) \in \mathbb{R}^2$ einen eindeutigen Abstand $d_E(a, b)$. Dieser berechnet sich folgendermassen:

$$d_E(a, b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2} \quad (\text{Pythagoras}).$$

Dieser *euklidische Abstand*⁶ $d_E(a, b)$ hat gewisse Eigenschaften: Erstens ist $d_E(a, b)$ immer eine nicht negative reelle Zahl, wobei $d_E(a, b) = 0$ nur gerade dann gilt, wenn der Radikand 0 und damit $a_1 = b_1$ sowie $a_2 = b_2$, das heisst $a = b$ ist. Zweitens spielt es keine Rolle, ob die Distanz von a nach b oder umgekehrt gemessen wird, denn allfällige Minuszeichen werden wegquadriert. Drittens gilt für diese Abstandsfunktion die sogenannte Dreiecksungleichung $d_E(a, b) + d_E(b, c) \geq d_E(a, c)$, das heisst, die Distanz kann nicht kürzer werden, wenn ein dritter Punkt als eine Art „Zwischenstation“ involviert wird.

Diese allgemein bekannten Eigenschaften der euklidischen Distanz in der Ebene sind die Bedingungen, welche eine Abstandsfunktion erfüllen muss, damit sie aus einer Menge einen metrischen Raum macht:

Definition 3.1. Eine (beliebige) Menge X heisst *metrischer Raum*, wenn es eine Funktion $d : X \times X \rightarrow \mathbb{R}_0^+$ gibt, die für alle $A, B, C \in X$ folgende Bedingungen erfüllt:

$$\begin{array}{lll} \text{(i)} & \text{a)} & d(A, B) \geq 0 & \text{(positiv definit)} \\ & \text{b)} & d(A, B) = 0 \Leftrightarrow A = B & \text{(koinzident)} \\ \text{(ii)} & & d(A, B) = d(B, A) & \text{(symmetrisch)} \\ \text{(iii)} & & d(A, B) + d(B, C) \geq d(A, C) & \text{(Dreiecksungleichung).} \end{array} \quad (3.1)$$

d wird *Metrik*, *Abstand* oder *Distanz* genannt.

Beispiel 3.1 (Taximetrik). (\mathbb{R}^2, d) mit $d(a, b) = |a_1 - b_1| + |a_2 - b_2|$
Überprüfen der Bedingungen:

(i) a) ist erfüllt, weil eine Summe von Beträgen nicht negativ sein kann.

(i) b) von rechts nach links ist erfüllt, weil für $a = b$ gilt: $d(a, b) = d(a, a) = |a_1 - a_1| + |a_2 - a_2| = |0| + |0| = 0$. Von links nach rechts ist die Bedingung (i) b) erfüllt, weil eine Summe von zwei Beträgen (≥ 0) nur genau dann 0 sein kann, wenn beide Beträge und damit auch ihre jeweiligen Argumente 0 sind. Es gilt also $a_1 - b_1 = 0$ und $a_2 - b_2 = 0$, woraus $a_1 = b_1$ und $a_2 = b_2$ und damit $a = b$ folgt.

(ii) ist erfüllt, weil $d(a, b) = |a_1 - b_1| + |a_2 - b_2| = | -(-a_1 + b_1) | + | -(-a_2 + b_2) | = |b_1 - a_1| + |b_2 - a_2| = d(b, a)$.

(iii) ist erfüllt, weil $d(a, b) + d(b, c) = |a_1 - b_1| + |a_2 - b_2| + |b_1 - c_1| + |b_2 - c_2| = \underbrace{|a_1 - b_1| + |b_1 - c_1|}_{\geq |a_1 - c_1| \quad (*)} + \underbrace{|a_2 - b_2| + |b_2 - c_2|}_{\geq |a_2 - c_2| \quad (*)} \geq |a_1 - c_1| + |a_2 - c_2| = d(a, c)$.

(*) Hier wird die Dreiecksungleichung für reelle Zahlen angewendet. Zum besseren Verständnis dieses Schritts kann man sich vorstellen, die Koordinaten a_1 , b_1 und c_1 seien auf der x_1 -Achse, also auf einer Geraden, eingezeichnet. Dann ist $|a_1 - c_1|$ die Länge der Strecke \overline{ac} , also die kürzeste Verbindung zwischen den

⁶Der *euklidische Abstand* von zwei Elementen $a, b \in \mathbb{R}^n$ mit den Koordinaten a_1, a_2, \dots, a_n bzw. b_1, b_2, \dots, b_n ist $d_E(a, b) := \sqrt{\sum_{i=1}^n (a_i - b_i)^2}$

zwei Punkten. Diese ist immer kürzer oder gleich lang wie die Längen der beiden Strecken \overline{ab} und \overline{bc} zusammengezählt. Gleichheit tritt auf, wenn der Punkt b im Intervall $[a, c]$, also „auf dem Weg“, liegt. Für die x_2 -Koordinaten sind die Überlegungen analog.

Es sind alle Bedingungen erfüllt, das heisst, (\mathbb{R}^2, d) ist ein metrischer Raum.

Beispiel 3.2 (Metrometrik). (\mathbb{C}, d) mit $d(a, b) = \begin{cases} |a - b| & \text{für } \arg(a) = \arg(b) \\ |a| + |b| & \text{für } \arg(a) \neq \arg(b) \end{cases}$

Überprüfen der Bedingungen:

(i) a) ist erfüllt, weil ein Betrag oder eine Summe von Beträgen nicht negativ sein kann.

(i) b) von rechts nach links ist erfüllt, weil für $a = b$ gilt: $d(a, b) = d(a, a) = |a - a| = |0| = 0$. Von links nach rechts ist die Bedingung (i) b) erfüllt, weil $|a - b| = 0$ bedingt, dass $a = b$ ist, respektive weil $|a| + |b|$ nur genau dann 0 ist, wenn $a = b = 0$ gilt.

(ii) ist erfüllt, weil (für $\arg(a) = \arg(b)$) $d(a, b) = |a - b| = |-(b - a)| = |b - a| = d(b, a)$, respektive weil (für $\arg(a) \neq \arg(b)$) $d(a, b) = |a| + |b| = |b| + |a| = d(b, a)$.

(iii) ist erfüllt, weil (für $\arg(a) = \arg(c)$) $d(a, c) = |a - c|$ die Länge der kürzesten Verbindung zwischen den beiden Zahlen in der komplexen Zahlenebene ist, respektive weil bei $\arg(a) \neq \arg(c)$ Folgendes gilt: Wenn b auf dem gleichen Strahl wie a ist, ergibt sich $d(a, b) + d(b, c) = |a - b| + |b| + |c|$. Für $|a| \geq |b|$ ist, weil die Vorzeichen von a und b immer gleich sind (sie liegen ja auf dem gleichen Strahl und damit im gleichen Quadranten), $|a - b| = |a| - |b|$ und deshalb $d(a, b) + d(b, c) = |a| - |b| + |b| + |c| = |a| + |c| = d(a, c)$. Für $|b| > |a|$ ist, ebenfalls wegen der gleichen Vorzeichen, $|a - b| = |b - a| = |b| - |a|$ und damit $|b| = |a| + |a - b|$, woraus folgt $d(a, b) + d(b, c) = |a - b| + |a| + |a - b| + |c| = |a| + |c| + 2 \cdot |a - b| > |a| + |c| = d(a, c)$. Wenn b auf dem gleichen Strahl ist wie c , sind die Überlegungen analog. Wenn b auf einem anderen Strahl liegt als a und als c , dann ist $d(a, b) + d(b, c) = |a| + |b| + |b| + |c| \geq |a| + |c| = d(a, c)$ (Gleichheit tritt nur bei $b = 0$ ein).

Es sind alle Bedingungen erfüllt, das heisst, (\mathbb{C}, d) ist ein metrischer Raum.

Beispiel 3.3. (X, d) mit X : Menge aller TeilnehmerInnen an einem bestimmten Rennen mit Massenstart und einem Ziel, das nur einzeln durchlaufen werden kann, und $d(A, B)$: Betrag der Differenz der Ankunftszeiten von A und B .

Überprüfen der Bedingungen:

(i) a) ist erfüllt, weil ein Betrag nicht negativ sein kann.

(i) b) von rechts nach links ist erfüllt, weil eine Person nur eine Ankunftszeit haben kann und deshalb die Differenz der Ankunftszeiten von $A = B$ immer 0 ist und damit $d(A, B) = 0$ gilt. Von links nach rechts ist die Bedingung (i) b) erfüllt, weil $d(A, B)$ nur genau dann 0 sein kann, wenn die beiden Ankunftszeiten übereinstimmen, was aber wegen der Art und Weise der Zeitmessung (Zieleinlauf einzeln) nur möglich ist, wenn es sich um dieselbe Person handelt.

(ii) ist erfüllt, weil $|t_A - t_B| = |t_B - t_A|$ gilt.

(iii) ist erfüllt, weil $|t_A - t_B| + |t_B - t_C| \geq |t_A - t_C|$ (Dreiecksungleichung für reelle Zahlen) gilt.

Es sind alle Bedingungen erfüllt, das heisst, (X, d) ist ein metrischer Raum.

Gegenbeispiel 3.4. Für den Raum (\mathbb{C}, d) sei $d(a, b)$ folgendermassen definiert: Für $a = b$ ist $d(a, b) = 0$. Wenn $a, b \neq a$ und 0 auf einer Geraden liegen, das

heisst, wenn $\arg(a) = \arg(b) + k \cdot 180^\circ (k \in \mathbb{Z})$, ist $d(a, b) = |a| + |b|$. In den anderen Fällen entspricht der Abstand von a und b der Länge des zwischen a und b liegenden und 0 beinhaltenden Bogenstücks des Umkreises des Dreiecks $ab0$. Ein Beispiel für eine solche Situation ist ein Schiff, das, sofern $a \neq b$ gilt (in diesem Fall muss es gar nicht fahren und die Distanz beträgt 0), von a aus via 0 nach b fahren soll. Es kann entweder geradeaus fahren oder eine gleichmässige Kurve ziehen. Ersteres findet Anwendung, wenn alle drei Orte auf einer Geraden liegen, Letzteres in den übrigen Fällen. Dabei ist die Kurve ein Bogenstück aus einem Kreis, auf dem alle drei Orte liegen. Dieser ist der Umkreis des Dreiecks $ab0$.

Überprüfen der Bedingungen:

(i) a) ist erfüllt, weil sowohl eine Summe von Beträgen als auch die Länge eines Bogenstücks nicht negativ sein kann.

(i) b) von rechts nach links ist definitionsgemäss erfüllt. Von links nach rechts ist die Bedingung (i) b) erfüllt, weil $d(a, b) = 0$ nur für $a = b$ gilt, da weder die Summe von Beträgen noch die Länge des Bogenstücks 0 sein kann, wenn $a \neq b$ gilt, was ja die Bedingung für die Anwendung dieser beiden Distanzdefinitionen ist.

(ii) ist erfüllt, weil $|a| + |b| = |b| + |a|$ gilt und weil die Länge des Kreisbogenstücks nicht von der Messrichtung abhängig ist.

(iii) ist nicht in allen Fällen erfüllt, denn ein „Zwischenhalt“ bei 0 verkürzt die Strecke im Fall, dass a , b und 0 nicht auf einer Geraden liegen. Dann wird nämlich der Kreisbogen durch die beiden Sehnen von a zu 0 und von 0 zu b ersetzt, deren Gesamtlänge kürzer ist als diejenige des Bogenstücks. Auch scheinbare „Umwege“ über andere Punkte (z. B. in der Nähe von 0) können die Distanz verkleinern. Betreffend das Schiff bedeutet dies, dass sich der Weg unter Umständen verkürzt, wenn statt eine Kurve gefahren angehalten und an Ort eine Richtungsänderung vorgenommen wird.

Es sind nicht alle Bedingungen erfüllt, das heisst, es handelt sich hierbei nicht um einen metrischen Raum.

Gegenbeispiel 3.5. (X, d) mit X : Menge aller Flughäfen und $d(A, B)$: Preis des billigsten Flugtickets von A nach B .

Überprüfen der Bedingungen:

(i) a) ist erfüllt, weil man kein Flugticket zu einem negativen Preis (man würde Geld erhalten) kaufen kann.

(i) b) von rechts nach links ist erfüllt, weil man von A nach $B = A$ nichts bezahlen muss, weil man gar nicht fliegt. Von links nach rechts ist die Bedingung (i) b) erfüllt, weil man nur genau dann nichts bezahlen muss, wenn man nicht fliegt, das heisst, wenn $A = B$ ist.

(ii) ist meist nicht erfüllt, da sich die Preise auch nach der Nachfrage richten, also zum Beispiel zu Beginn der Ferien ein Flug von A aus mit einer beliebten Feriendestination B als Ziel teurer ist als der entsprechende Rückflug von B nach A .

(iii) ist in gewissen Fällen auch nicht erfüllt. Es kommt vor, dass ein Flug mit Zwischenlandung auf einem gewissen Flughafen B billiger ist als ein Direktflug von A nach C .

Es sind nicht alle Bedingungen erfüllt, das heisst, es handelt sich hierbei nicht um einen metrischen Raum.

Nach diesen Beispielen zur Veranschaulichung von metrischen Räumen nun wieder zurück zum Thema Konvergenz und Grenzwerte von Folgen:

Wenn eine Metrik d für einen Raum X vorliegt, kann über Grenzwerte von Folgen gesprochen werden. Sei (a_n) eine Folge von Elementen aus X und a_∞ ebenfalls ein Element aus X . Dann ist a_∞ der Grenzwert der Folge, falls $\lim_{n \rightarrow \infty} d(a_n, a_\infty) = 0$. Anders gesagt, für jedes $\varepsilon > 0$ gibt es ein a_n so, dass die Distanz zu a_∞ jedes nachfolgenden Folgenglieds kleiner als ε ist.

Wenn die Konvergenz einer Folge ohne Kenntnis des Grenzwerts, sondern nur mit Hilfe der Abstände aufeinanderfolgender Folgenglieder untersucht werden soll, muss der betreffende metrische Raum von besonderer Art sein. Um solche sogenannte vollständige metrische Räume geht es im nächsten Abschnitt.

3.1.2 Vollständigkeit von metrischen Räumen

Definition 3.2. Ein metrischer Raum (X, d) heisst *vollständig*, wenn jede Cauchy-Folge einen Grenzwert besitzt, der zu X gehört.

Eine *Cauchy-Folge*⁷ ist eine Folge $a_1, a_2, \dots \in X$, für welche es zu jedem $\varepsilon > 0$ ein $N \in \mathbb{N}$ so gibt, dass für alle $i, j \geq N$ gilt $d(a_i, a_j) < \varepsilon$. Bei einer Cauchy-Folge gibt es also zu jeder beliebig kleinen positiven Zahl ε ein bestimmtes Folgenglied, unter dessen Nachfolgern sich keine finden, die einen grösseren Abstand als ε voneinander haben.

Gegenbeispiel 3.6. (\mathbb{Q}, d_E) ist ein metrischer Raum. Die Folge $1, 1 + \frac{1}{2^2}, 1 + \frac{1}{2^2} + \frac{1}{3^2}, \dots, \sum_{k=1}^n \frac{1}{k^2}, \dots$ ist eine Cauchy-Folge, die bekanntlich gegen $\frac{1}{6}\pi^2$ konvergiert. Weil $\frac{1}{6}\pi^2 \notin \mathbb{Q}$ gilt, handelt es sich bei (\mathbb{Q}, d_E) nicht um einen vollständigen metrischen Raum.

3.2 Der vollständige metrische Raum $(\mathcal{C}(\mathbb{R}^2), h)$

Wenn es um zweidimensionale Bilder geht, interessieren kompakte Teilmengen von \mathbb{R}^2 . Was für die Untersuchung der Konvergenz der bei Iteration des Hutchinson-Operators entstehenden Bildfolge benötigt wird, ist eine Abstandsfunktion für kompakte Mengen. Eine solche ist die Hausdorff-Distanz h . Deshalb geht es in diesem Kapitel um kompakte Mengen und die Hausdorff-Distanz, welche zusammen einen vollständigen metrischen Raum bilden.

3.2.1 Kompakte Mengen in \mathbb{R}^n

In diesem Abschnitt stützen wir uns auf [HEU1], [HEU2] sowie stellenweise auf [DUD], [FAL] und [ZEI/PAG].

Eine Teilmenge A des \mathbb{R}^n heisst *abgeschlossen*, wenn der Grenzwert jeder konvergenten Folge aus A wieder in A liegt.

Eine Teilmenge A des \mathbb{R}^n ist abgeschlossen, wenn sie alle ihre Häufungspunkte enthält. Ein Punkt $P \in \mathbb{R}^n$ ist ein *Häufungspunkt*, wenn es in jeder ε -Umgebung von P mindestens einen Punkt $Q \in \mathbb{R}^n$ mit $Q \neq P$ gibt.

Intuitiv gesprochen, ist eine Menge abgeschlossen, wenn sie alle ihre Randpunkte enthält.

Eine Teilmenge A des \mathbb{R}^n ist *beschränkt*, wenn sie in einer (genügend grossen) Kugel (bzw. Kreisscheibe für $A \in \mathbb{R}^2$ bzw. Strecke für $A \in \mathbb{R}$) enthalten ist.

⁷Augustin Louis CAUCHY (1789-1857)

Formal bedeutet dies, dass A beschränkt ist, wenn es $P \in A$, $r \in \mathbb{R}^+$ so gibt, dass für alle $X \in A$ gilt, dass $d_E(X, P) < r$.

Definition 3.3. Eine Teilmenge A des \mathbb{R}^n heisst *kompakt*, wenn jede Folge aus A eine Teilfolge enthält, die gegen ein Element von A konvergiert.

Satz 3.1. Eine Teilmenge A des \mathbb{R}^n ist genau dann kompakt, wenn sie beschränkt und abgeschlossen ist.

Beweis. [1. Teil: Jede kompakte Teilmenge A des \mathbb{R}^n ist beschränkt und abgeschlossen]: Wäre die Menge A unbeschränkt, würde sie eine Folge (x_n) mit $|x_n| > n$ enthalten, deren Teilfolgen aber alle unbeschränkt und deshalb divergent wären. Da dies ein Widerspruch zur Kompaktheit von A wäre, muss A beschränkt sein.

Für die Abgeschlossenheit von A muss gezeigt werden, dass $x_\infty \in A$, wenn (x_n) eine Folge aus A mit Grenzwert x_∞ ist. Weil A kompakt ist, strebt definitionsgemäss eine gewisse Teilfolge von (x_n) gegen ein Element aus A . Da aber diese Teilfolge natürlich auch gegen x_∞ strebt, ist $x_\infty \in A$.

Damit ist gezeigt, dass jede kompakte Teilmenge A des \mathbb{R}^n sowohl beschränkt als auch abgeschlossen ist.

[2. Teil: Jede beschränkte und abgeschlossene Teilmenge A des \mathbb{R}^n ist kompakt]: Für diesen Beweis soll ein Hilfssatz beigezogen werden:

Satz 3.2 (Auswahlprinzip von Bolzano-Weierstraß⁸). Jede beschränkte Folge enthält eine konvergente Teilfolge.

Beweis. Es kann gezeigt werden, dass jede Folge (x_n) eine *monotone*, das heisst monoton wachsende oder monoton fallende, Teilfolge enthält:

m heisst eine *Gipfelstelle* von (x_n) , wenn für alle $n > m$ gilt, dass $x_n < x_m$. Wenn die Folge (x_n) unendlich viele Gipfelstellen $m_1 < m_2 < \dots$ besitzt, dann ist $x_{m_1} > x_{m_2} > \dots$ eine (streng) monoton fallende Teilfolge von (x_n) .

Andernfalls, das heisst, wenn es nur endlich viele Gipfelstellen gibt, findet sich eine Stelle n_1 , welche grösser ist als alle Gipfelstellen und deshalb selbst keine Gipfelstelle ist. Dies bedeutet aber, dass es irgend ein späteres Folgenglied geben muss, das mindestens so gross wie x_{n_1} ist. Es gibt also $n_2 > n_1$ mit $x_{n_2} \geq x_{n_1}$. Da auch n_2 keine Gipfelstelle sein kann, gibt es $n_3 > n_2$ mit $x_{n_3} \geq x_{n_2}$. So fortfahrend, erhält man eine monoton wachsende Teilfolge.

Damit ist gezeigt, dass jede Folge (x_n) eine monotone Teilfolge enthält.

Es gilt, dass eine monotone Folge genau dann konvergiert, wenn sie beschränkt ist.

Eine beliebige beschränkte Folge (x_n) enthält also eine (natürlich ebenfalls beschränkte) monotone Teilfolge, welche konvergiert.

Damit ist gezeigt, dass jede beschränkte Folge eine konvergente Teilfolge enthält. \square

Nun wieder zurück zum Beweis, dass jede beschränkte und abgeschlossene Teilmenge A des \mathbb{R}^n kompakt ist:

(x_n) sei eine Folge aus A . Es muss gezeigt werden, dass diese eine Teilfolge enthält, welche gegen ein Element von A konvergiert. Aus der Beschränktheit von A folgt, dass (x_n) beschränkt ist. Dies wiederum bedeutet, dass für jedes x_n die Summe der Beträge seiner Koordinaten $k_1^{(n)}, k_2^{(n)}, \dots, k_p^{(n)}$ (p steht für

⁸Bernhard BOLZANO (1781-1848) und Karl WEIERSTRASS (1815-1897)

die Anzahl Dimensionen, also \mathbb{R}^p) beschränkt ist. Nun wird die Komponente k_1 separat betrachtet: Die Folge $(k_1^{(n)}) = k_1^{(1)}, k_1^{(2)}, \dots$ ist natürlich ebenfalls beschränkt. Deswegen enthält sie gemäss dem Auswahlprinzip von Bolzano-Weierstraß (s. Satz 3.2) eine konvergente Teilfolge. Diese sei $(k_1^{(n_m)})$. Werden die korrespondierenden Folgenglieder der zweiten Koordinatenfolge $(k_2^{(n)})$ betrachtet, das heisst $(k_2^{(n_m)})$, hat man es ebenfalls mit einer beschränkten Folge zu tun. Auch diese hat eine konvergente Teilfolge $(k_2^{(n_{m_l})})$. Natürlich ist auch $(k_1^{(n_{m_l})})$ konvergent, denn es handelt sich ja dabei um eine Auswahl von Folgengliedern der konvergenten Folge $(k_1^{(n_m)})$. So fortfahrend, erhält man schliesslich eine Teilfolge $(k_p^{(n')})$ der p -ten Koordinatenfolge so, dass nicht nur $(k_p^{(n')})$ selbst, sondern auch alle korrespondierenden Folgen $(k_1^{(n')}), (k_2^{(n')}), \dots, (k_{p-1}^{(n')})$ konvergieren. Mit $k_i := \lim_{n' \rightarrow \infty} k_i^{(n')}$ für $1 \leq i \leq p$ erhält man die Koordinaten des Grenzwerts einer Teilfolge von (x_n) . Dieser Grenzwert liegt in A , weil A abgeschlossen ist.

Jede beliebige Folge (x_n) aus A enthält demnach eine gegen ein Element aus A konvergierende Teilfolge.

Damit ist gezeigt, dass eine beschränkte und abgeschlossene Teilmenge von \mathbb{R}^n kompakt ist. \square

An dieser Stelle soll ein weiterer Satz betreffend kompakte Mengen, welcher nicht nur für \mathbb{R}^n gilt, erwähnt werden. Der nicht allzu kurze Beweis kann beispielsweise in [HEU2] (S. 37) nachgelesen werden.

Satz 3.3 (Überdeckungssatz von Heine-Borel⁹). *Eine Menge A ist genau dann kompakt, wenn jede Überdeckung von A mit offenen Mengen eine endliche Teilüberdeckung enthält.*

Eine *Überdeckung* von A ist eine Auswahl von Mengen, deren Vereinigungsmenge A enthält:

$$A \subseteq \bigcup_i U_i.$$

Endliche Teilüberdeckung bedeutet, dass eine Auswahl von endlich vielen dieser U_i immer noch ganz A überdeckt, also dass gilt $A \subseteq \bigcup_{i=1}^N U_i$, wobei $N \in \mathbb{N}$.

Es muss eine Überdeckung mit offenen Mengen sein, weil sonst jeder Punkt einzeln mit einer abgeschlossenen Kugel mit Radius 0 überdeckt werden könnte und dadurch nur noch Mengen mit einer endlichen Anzahl Punkte kompakt wären.

Es soll noch einmal darauf hingewiesen werden, dass sich der Satz 3.1 (s. S. 12) auf Teilmengen des (\mathbb{R}^n, d_E) bezieht. Im Allgemeinen ist nicht jede abgeschlossene und beschränkte Menge zwingend kompakt, wie folgendes Beispiel ([WIK2]) zeigt:

Für einen beliebigen metrischen Raum (X, d) sei die Distanz definiert als $d(a, b) = 0$ für $a = b$ und $d(a, b) = 1$ für $a \neq b$. Dann sind zwar alle Teilmengen dieses Raums sowohl abgeschlossen (es gibt gar keine Häufungspunkte) als auch beschränkt (beliebige zwei Punkte haben höchstens einen Abstand von 1), aber

⁹Eduard HEINE (1821-1881) und Émile BOREL (1871-1956)

nur endliche Teilmengen von X sind kompakt in dem Sinne, dass alle ihre offenen Überdeckungen eine endliche Teilüberdeckung enthalten. Der Überdeckungssatz von Heine-Borel ist deshalb mächtiger als der Satz 3.1.

Gegenbeispiel 3.7. Die Menge $\{\frac{1}{n} : n \in \mathbb{N}\}$ ist nicht kompakt. Um dies zu zeigen, kann man sich folgende Überdeckung ansehen: Jedes $\frac{1}{n}$ wird mit der offenen Kugel $B_{\frac{1}{2}(\frac{1}{n}-\frac{1}{n+1})}^o(\frac{1}{n})$ überdeckt. Der Radius beträgt somit jeweils die Hälfte der Länge der kürzeren angrenzenden Strecke (diejenige näher bei 0). Von der anderen Strecke (diejenige näher bei 1) deckt die Kugel weniger als die Hälfte ab. Also sind die Kugeln alle *disjunkt* (das heisst, ihre Schnittmenge ist die leere Menge), weshalb das Fehlen einer Kugel nicht kompensiert werden könnte und die Menge $\{\frac{1}{n} : n \in \mathbb{N}\}$ in diesem Fall nicht mehr überdeckt wäre. Demnach hat diese Überdeckung mit offenen Mengen keine endliche Teilüberdeckung und die Menge $\{\frac{1}{n} : n \in \mathbb{N}\}$ ist nicht kompakt.

Beispiel 3.8. Die Menge $\{\frac{1}{n} : n \in \mathbb{N}\} \cup 0$ ist kompakt. Die offene Kugel, welche 0 überdeckt, hat einen gewissen Radius $\varepsilon > 0$. Es lässt sich immer ein $N \in \mathbb{N}$ finden, so dass $\frac{1}{N} < \varepsilon$ gilt. Dann wird $\{\frac{1}{n} : n \geq N\} \subset B_\varepsilon^o(0)$ gelten. Dies bedeutet aber, dass nur noch eine endliche Anzahl von Punkten nicht überdeckt ist. Auf diese Weise gibt es für jede Überdeckung von $\{\frac{1}{n} : n \in \mathbb{N}\} \cup 0$ mit offenen Mengen eine endliche Teilüberdeckung und die Menge ist deshalb kompakt.

Nach der ausführlichen Betrachtung des Begriffs der Kompaktheit nun zu einer Bezeichnung, die im Folgenden wichtig ist: Es steht $\mathcal{C}(X)$ für die Menge der nicht leeren kompakten Teilmengen des metrischen Raumes X . Die Bezeichnung $\mathcal{C}(\mathbb{R}^2)$ meint demnach die Menge aller nicht leeren kompakten Teilmengen von \mathbb{R}^2 .

3.2.2 Die Hausdorff-Distanz

Dieser Abschnitt basiert auf [PEI/JÜR/SAU].

Die Hausdorff-Distanz¹⁰ ist eine Funktion, die dazu dient, den Abstand zwischen zwei nicht leeren kompakten Teilmengen eines vollständigen metrischen Raumes (X, d) zu bestimmen. Sie ist deshalb geeignet für die Untersuchung der Distanz zwischen zweidimensionalen Bildern.

Vor der Definition der Hausdorff-Distanz muss ein Begriff eingeführt werden: $U_\delta(A)$ ist die sogenannte δ -parallele Erweiterung einer Menge $A \in \mathcal{C}(X)$. Diese kann man sich als „aufgeblasene“ Menge A vorstellen. Sie wird auch δ -Kragen von A genannt, obwohl Kragen im Sinn von Kranz nicht die richtige Vorstellung ist, da die Menge A dazugehört.

Formal ist die δ -parallele Erweiterung

$$U_\delta(A) := \{x \in X : d(x, a) \leq \delta \text{ für irgendein } a \in A\}, \text{ wobei } \delta \in \mathbb{R}_0^+.$$

Der Abstand zweier nicht leeren kompakten Mengen ist nun folgendermassen definiert:

Definition 3.4. Die *Hausdorff-Distanz* h zwischen zwei Mengen $A, B \in \mathcal{C}(X)$ ist

$$h(A, B) = \inf\{\delta : A \subseteq U_\delta(B) \text{ und } B \subseteq U_\delta(A)\}. \quad (3.2)$$

¹⁰Felix HAUSDORFF (1868-1942)

Dies bedeutet, dass die Hausdorff-Distanz definiert ist als das kleinste δ , für welches die Menge A in der δ -parallelen Erweiterung von B liegt und umgekehrt.

Auf Punkte x, y aus (\mathbb{R}^n, d_E) angewendet ergibt die Hausdorff-Distanz den gewohnten euklidischen Abstand, also $h(x, y) = d_E(x, y)$.

3.2.3 Der metrische Raum $(\mathcal{C}(X), h)$

Nun soll gezeigt werden, dass es sich bei $(\mathcal{C}(X), h)$ um einen metrischen Raum handelt. Dabei stützen wir uns auf [EDG] und [ROH].

Satz 3.4. *Die Menge $\mathcal{C}(X)$ aller nicht leeren kompakten Teilmengen eines metrischen Raums X in Kombination mit der Hausdorff-Distanz h ist ein metrischer Raum.*

Beweis. Überprüfen der Bedingungen (s. Definition 3.1 auf S. 8):

$h(A, B) \geq 0$ ist erfüllt, weil $\delta \in \mathbb{R}_0^+$.

$h(A, B) = 0 \Leftrightarrow A = B$ von rechts nach links ist erfüllt, weil für $A = B$ gilt, dass $A \subseteq U_0(B)$ und $B \subseteq U_0(A)$ und damit $h(A, B) = 0$. Von links nach rechts ist die Bedingung erfüllt, weil $h(A, B) = 0$ bedeutet, dass es für jeden Punkt aus A beliebig nahe einen Punkt aus B gibt und umgekehrt, das heisst, dass A dicht in B und B dicht in A liegt. Da die Mengen alle kompakt und damit gemäss dem Satz 3.1 (s. S. 12) alle abgeschlossen sind, folgt daraus $A = B$. An dieser Stelle soll noch einmal darauf hingewiesen sein, dass die Hausdorff-Distanz nur auf nicht leere kompakte Mengen anwendbar ist. Wie soeben gezeigt, ist die Eigenschaft abgeschlossen erforderlich, weil sonst beispielsweise die Mengen $[0, 1]$ und $]0, 1[$ die Hausdorff-Distanz 0 haben, obwohl sie nicht identisch sind. Die Eigenschaften beschränkt und nicht leer sind nötig, weil es in Fällen wie $A = 0; B = [0, \infty[$ oder $A = 0; B = \emptyset$ kein δ gibt, für das $A \subseteq U_\delta(B)$ und $B \subseteq U_\delta(A)$ gelten, und damit $d(A, B) = \inf\{\emptyset\} = \infty$ wäre, was nicht der Bedingung $\delta \in \mathbb{R}_0^+$ genügen würde.

$h(A, B) = h(B, A)$ ist erfüllt, weil der Teil $A \subseteq U_\delta(B)$ und $B \subseteq U_\delta(A)$ in der Definition 3.4 (s. S. 14) symmetrisch ist.

$h(A, B) + h(B, C) \geq h(A, C)$ ist erfüllt, weil für $h_1 := h(A, B)$ und $h_2 := h(B, C)$ definitionsgemäss $A \subseteq U_{h_1}(B)$ sowie $B \subseteq U_{h_2}(C)$ und damit $A \subseteq U_{h_1}(B) \subseteq U_{h_1}(U_{h_2}(C)) = U_{h_1+h_2}(C)$ gilt. Analog lässt sich $C \subseteq U_{h_1+h_2}(A)$ zeigen. Es folgt, dass $h(A, C) = \inf\{\delta : A \subseteq U_\delta(C) \text{ und } C \subseteq U_\delta(A)\}$ sicher nicht grösser als $h_1 + h_2$ ist. Also ergibt sich $h(A, C) \leq h_1 + h_2 = h(A, B) + h(B, C)$. Dabei ist $h(A, C)$ echt kleiner als $h(A, B) + h(B, C)$, wenn für $h_{AB} := \inf\{\delta : B \subseteq U_\delta(A)\}$, $h_{BA} := \inf\{\delta : A \subseteq U_\delta(B)\}$, $h_{BC} := \inf\{\delta : C \subseteq U_\delta(B)\}$ und $h_{CB} := \inf\{\delta : B \subseteq U_\delta(C)\}$ entweder $h_{AB} > h_{BA}$ und $h_{CB} > h_{BC}$ oder $h_{AB} < h_{BA}$ und $h_{CB} < h_{BC}$ gelten, weil unter diesen Umständen $h_1 + h_2$ nicht dem kleinstmöglichen Wert für δ entspricht. Dies liegt daran, dass beispielsweise für $h_2 = h_{BC} > h_{CB}$ (s. Abb. 3.1, 2)) der Wert von h_2 ein unnötig grosser ist, was das Erfüllen der Bedingung $B \subseteq U_{h_2}(C)$ und damit auch $A \subseteq U_{h_1+h_2}(C)$ betrifft. Da es aber nicht nur auf das kleinste δ mit $A \subseteq U_\delta(C)$, sondern auch auf dasjenige mit $C \subseteq U_\delta(A)$ ankommt, muss gleichzeitig $h_{AB} < h_{BA}$ gelten, damit auch von A ausgehend zur Einschliessung von B weniger als h_1 und damit für die gesamte Distanz weniger als $h_1 + h_2$ nötig ist. Für den anderen Fall (s. Abb. 3.1, 1)) lassen sich ähnliche Überlegungen machen.

Damit ist gezeigt, dass alle Bedingungen (s. Definition 3.1 auf S. 8) erfüllt sind und dass $(\mathcal{C}(X), h)$ ein metrischer Raum ist. \square

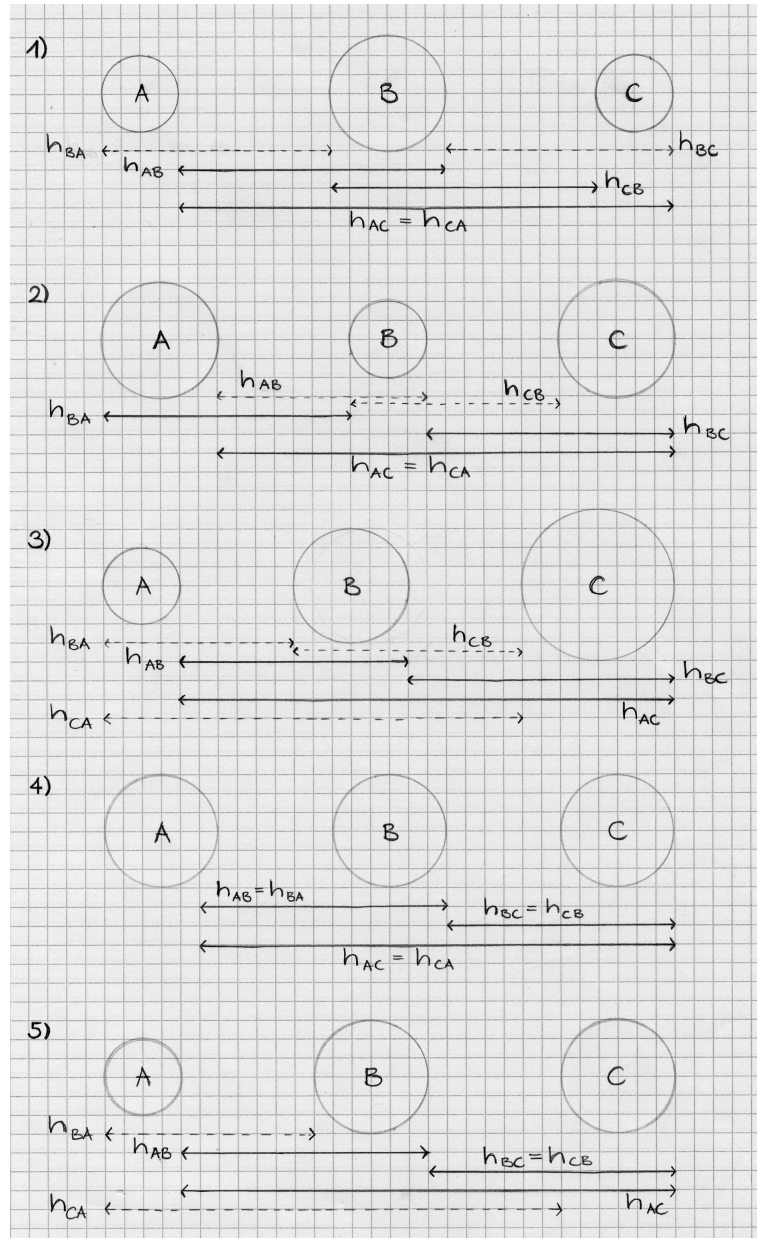


Abbildung 3.1: In 1) ($h_{AB} > h_{BA}$ und $h_{CB} > h_{BC}$) sowie in 2) ($h_{AB} < h_{BA}$ und $h_{CB} < h_{BC}$) ist $h(A, C)$ echt kleiner als $h(A, B) + h(B, C)$, in 3), 4) und 5) tritt Gleichheit auf.

Der soeben behandelte Satz 3.4 kann insbesondere auf den metrischen Raum (\mathbb{R}^2, d_E) übertragen werden. Dann folgt, dass $(\mathcal{C}(\mathbb{R}^2), h)$ ein metrischer Raum ist. Im Folgenden wird es um diesen Raum gehen.

3.2.4 Vollständigkeit von $(\mathcal{C}(\mathbb{R}^2), h)$

In diesem Abschnitt beweisen wir, dass es sich bei $(\mathcal{C}(\mathbb{R}^2), h)$ um einen vollständigen metrischen Raum (s. Kap. 3.1.2 auf S. 11) handelt, und folgen dabei im Wesentlichen [ROH].

Satz 3.5. *Weil (\mathbb{R}^2, d_E) ein vollständiger metrischer Raum ist, ist $(\mathcal{C}(\mathbb{R}^2), h)$ auch ein vollständiger metrischer Raum.*

Beweis. Zuerst wird gezeigt, dass die Schnittmenge $M_\infty = \bigcap_{i=1}^\infty M_i$ einer fallenden Folge $M_1 \supset M_2 \supset M_3 \supset \dots$ von nicht leeren kompakten Teilmengen von \mathbb{R}^2 ebenfalls ein Element aus $\mathcal{C}(\mathbb{R}^2)$ ist:

Mit einem indirekten Beweis wird gezeigt, dass M_∞ nicht leer ist: Es wird demnach angenommen, M_∞ sei leer. Nur unter dieser Bedingung ist

$$\bigcup_{i=1}^\infty U_i \text{ mit } U_i := M_1 \setminus M_{i+1}$$

eine Überdeckung von M_1 . Dabei sind alle U_i offen in M_1 , denn es sind die jeweiligen Komplementärmengen zu den M_{i+1} , welche alle kompakt und damit gemäss dem Satz 3.1 (s. S. 12) abgeschlossen sind.

Da $M_{i+1} \subset M_i$ und damit $M_1 \setminus M_{i+1} \supset M_1 \setminus M_i$ gilt, kommen mit grösser werdendem i immer mehr für die Überdeckung unerlässliche Punkte zur Vereinigung $\bigcup_{i=1}^\infty U_i$. Es braucht also unendlich viele U_i , um M_1 zu überdecken.

Somit ist $\bigcup_{i=1}^\infty U_i$ eine offene Überdeckung der kompakten Menge M_1 ohne endliche Teilüberdeckung. Dies ist ein Widerspruch zum Überdeckungssatz von Heine-Borel (s. S. 13). Also kann M_∞ nicht eine leere Menge sein.

Kompakt ist die Menge M_∞ , weil die Schnittmenge jeder Auswahl kompakter Mengen kompakt ist. Dies folgt daraus, dass eine Teilmenge einer kompakten Menge sicher beschränkt ist und dass abgeschlossene Mengen definitionsgemäss alle ihre Häufungspunkte beinhalten, weshalb eine Schnittmenge von abgeschlossenen Mengen auch alle ihre Häufungspunkte enthält, was bedeutet, dass sie abgeschlossen ist (s. Kap. 3.2.1 auf S. 11).

M_∞ ist demnach nicht leer und kompakt.

Nach dieser Feststellung kann nun gezeigt werden, dass es sich bei $(\mathcal{C}(\mathbb{R}^2), h)$ um einen vollständigen metrischen Raum handelt, dass also jede Cauchy-Folge darin einen Grenzwert besitzt, der zu $(\mathcal{C}(\mathbb{R}^2), h)$ gehört:

Die Folge (A_n) , wobei $A_n \in \mathcal{C}(\mathbb{R}^2)$, sei eine Cauchy-Folge bezüglich der Hausdorff-Distanz. Nun wird Folgendes definiert:

$$M_n := \overline{\bigcup_{i=n}^\infty A_i}.$$

M_n ist also jeweils der *Abschluss* der Vereinigung von A_n, A_{n+1}, \dots , das heisst die kleinste abgeschlossene Menge, die diese Vereinigungsmenge enthält. Je grösser n ist, desto weniger (nicht leere) A_i werden einbezogen, was bedeutet, dass (M_n) eine fallende Folge ist. Wenn also M_1 beschränkt ist, sind alle M_n

beschränkt. Da zudem alle M_n abgeschlossen (Definition) und nicht leer (kein A_i ist leer) sind, reicht ein Beweis, dass M_1 beschränkt ist, um zu zeigen, dass alle M_n nicht leere kompakte Teilmengen von \mathbb{R}^2 sind. Dies liegt natürlich daran, dass genau die abgeschlossenen und beschränkten Teilmengen von \mathbb{R}^2 kompakt sind (s. Satz 3.1 auf S. 12).

Die Beschränktheit von M_1 ist folgendermassen ersichtlich: Da (A_n) eine Cauchy-Folge (s. Kap. 3.1.2 auf S. 11) bezüglich der Hausdorff-Distanz ist, findet sich für jedes $\varepsilon > 0$ ein $N \in \mathbb{N}$ so, dass $h(A_N, A_n) < \varepsilon$ für alle $n \geq N$, was aber auch bedeutet, dass $A_n \subseteq U_\varepsilon(A_N)$ für alle $n \geq N$. Damit ist auch die Vereinigung all dieser A_n mit $n \geq N$ in $U_\varepsilon(A_N)$ enthalten. Diese Vereinigung $\bigcup_{i=N}^\infty A_i$ ist wie M_N ohne Abschluss und sei deshalb mit M_N^* bezeichnet. Es gilt also $M_N^* \subseteq U_\varepsilon(A_N)$, und weil die ε -parallele Erweiterung der kompakten Menge A_N beschränkt ist, dass M_N^* beschränkt ist. Die Menge M_1 ist gemäss Definition $M_1 = \overline{\bigcup_{i=1}^\infty A_i}$, was sich aber auch als $M_1 = \overline{\bigcup_{i=1}^{N-1} A_i} \cup M_N^*$ schreiben lässt. Dabei ist die Vereinigung $\bigcup_{i=1}^{N-1} A_i$ endlich vieler kompakter Mengen beschränkt. Also ist M_1 nichts Anderes als der Abschluss der Vereinigung zweier beschränkter Mengen. Somit ist gezeigt, dass M_1 beschränkt ist.

Nun ist also ebenfalls gezeigt, dass (M_n) eine fallende Folge von nicht leeren kompakten Teilmengen von \mathbb{R}^2 ist. Gemäss der ersten Feststellung aus diesem Beweis gibt es also eine nicht leere kompakte Menge A_∞ mit

$$A_\infty = \bigcap_{i=1}^\infty M_i.$$

Wie gesehen, ist $M_N^* \subseteq U_\varepsilon(A_N)$. Da A_N kompakt und damit abgeschlossen ist, ist auch $\overline{U_\varepsilon(A_N)}$ abgeschlossen. Wenn also M_N^* in $U_\varepsilon(A_N)$ enthalten ist, gilt dies auch für $\overline{M_N^*}$, was aber gerade M_N ist. Deshalb gilt $M_N \subseteq U_\varepsilon(A_N)$. Weil die Menge A_∞ die Schnittmenge aller M_i ist, ist sie sicher eine Teilmenge von M_N . Also gilt $A_\infty \subseteq M_N \subseteq U_\varepsilon(A_N)$.

Da (M_n) zu A_∞ konvergiert, gilt für genügend grosse N auch $M_N \subseteq U_\varepsilon(A_\infty)$. Weil M_N definitionsgemäss unter anderem A_N enthält, gilt $A_N \subseteq M_N \subseteq U_\varepsilon(A_\infty)$.

Aus den beiden Aussagen $A_\infty \subseteq U_\varepsilon(A_N)$ und $A_N \subseteq U_\varepsilon(A_\infty)$ folgt sofort $h(A_N, A_\infty) \leq \varepsilon$ für genügend grosse N , was bedeutet, dass $A_n \rightarrow A_\infty$ für $n \rightarrow \infty$. Also hat eine Cauchy-Folge (A_n) den Grenzwert $A_\infty \in \mathcal{C}(\mathbb{R}^2)$. Damit ist die Vollständigkeit von $(\mathcal{C}(\mathbb{R}^2), h)$ bewiesen. \square

3.2.5 \mathcal{F} als Kontraktion auf $(\mathcal{C}(\mathbb{R}^2), h)$

Für den folgenden Satz orientieren wir uns an [PEI/JÜR/SAU].

Satz 3.6. *In $(\mathcal{C}(\mathbb{R}^2), h)$ ist der Hutchinson-Operator \mathcal{F} eine distanzkontrahierende Abbildung, wenn die Abbildungen f_i in (\mathbb{R}^2, d_E) alle distanzkontrahierend sind.*

Beweis. Dass der Hutchinson-Operator in $(\mathcal{C}(\mathbb{R}^2), h)$ distanzkontrahierend ist, bedeutet, dass $h(\mathcal{F}(A), \mathcal{F}(B)) \leq q \cdot h(A, B)$ mit $0 < q < 1$ für zwei beliebige Mengen $A, B \in \mathcal{C}(\mathbb{R}^2)$.

Die Hausdorff-Distanz $h(A, B)$ sei δ . Dann liegt B definitionsgemäss im δ -Kragen von A , das heisst, es gilt $B \subseteq U_\delta(A)$. Werden die Transformationen

$\{f_1, f_2, \dots, f_N\}$ angewendet, ergibt sich

$$f_1(B) \subseteq f_1(U_\delta(A)), f_2(B) \subseteq f_2(U_\delta(A)), \dots, f_N(B) \subseteq f_N(U_\delta(A)).$$

Weil die Transformationen alle Kontraktionen sind, liegt das Bild unter f_i des δ -Kragens von A im $q_i \cdot \delta$ -Kragen von $f_i(A)$. Es gelten also

$$\begin{aligned} f_1(U_\delta(A)) &\subseteq U_{\delta \cdot q_1}(f_1(A)) \\ f_2(U_\delta(A)) &\subseteq U_{\delta \cdot q_2}(f_2(A)) \\ &\vdots \\ f_N(U_\delta(A)) &\subseteq U_{\delta \cdot q_N}(f_N(A)). \end{aligned}$$

Wird $q := \max(q_1, q_2, \dots, q_N)$ definiert, ergibt sich, dass $f_1(B) \cup f_2(B) \cup \dots \cup f_N(B)$ innerhalb des $q \cdot \delta$ -Kragens von $f_1(A) \cup f_2(A) \cup \dots \cup f_N(A)$ liegt. Dies ist gleichbedeutend wie $\mathcal{F}(B) \subseteq U_{q \cdot \delta}(\mathcal{F}(A))$.

Ein analoger Gedankengang führt auf $\mathcal{F}(A) \subseteq U_{q \cdot \delta}(\mathcal{F}(B))$.

Damit ist offensichtlich, dass die Hausdorff-Distanz $h(\mathcal{F}(A), \mathcal{F}(B))$ höchstens $q \cdot \delta$ beträgt. Da $q = \max(q_1, q_2, \dots, q_N)$ gilt, ist $0 < q < 1$ genau dann erfüllt, wenn die Abbildungen f_i alle distanzkontrahierend sind. In diesem Fall ist der Hutchinson-Operator distanzkontrahierend bezüglich der Hausdorff-Distanz. \square

Nachdem nun geklärt ist, unter welchen Bedingungen der Hutchinson-Operator \mathcal{F} eine Kontraktion auf $(\mathcal{C}(\mathbb{R}^2), h)$ ist, wollen wir von nun an nur noch solche Hutchinson-Operatoren betrachten:

Ab jetzt gilt, dass $\{f_1, f_2, \dots, f_N\}$ Kontraktionen $\mathbb{R}^2 \rightarrow \mathbb{R}^2$ sind, also dass

$$d_E(f_i(a), f_i(b)) \leq q_i \cdot d_E(a, b)$$

mit $0 < q_i < 1$ für jedes i und alle $a, b \in \mathbb{R}^2$.

Für den Hutchinson-Operator gilt natürlich immer noch $\mathcal{F}(A) = \bigcup_{i=1}^N f_i(A)$ (s. Definition 2.3 auf S. 7). Wie gesagt, ist er ab jetzt als eine distanzkontrahierende Abbildung auf $(\mathcal{C}(\mathbb{R}^2), h)$ zu betrachten.

3.3 Der Banach'sche Fixpunktsatz

Der Banach'sche Fixpunktsatz¹¹ ist für verschiedene Gebiete der Mathematik von fundamentaler Bedeutung. Er gilt für vollständige metrische Räume und distanzkontrahierende Abbildungen und ist demnach auf den vollständigen metrischen Raum $(\mathcal{C}(\mathbb{R}^2), h)$ und die distanzkontrahierende Abbildung \mathcal{F} anwendbar. Wir übernehmen diesen Satz aus [ZEI/PAG].

Satz 3.7. *Jede distanzkontrahierende Abbildung \mathcal{F} eines vollständigen metrischen Raumes (X, d) besitzt genau einen Fixpunkt A_∞ , das heisst, es gibt genau ein $A_\infty \in X$ so, dass $\mathcal{F}(A_\infty) = A_\infty$.*

Beweis. Es sei $A_n = \mathcal{F}(A_{n-1}) = \mathcal{F}^n(A_0)$. Weil \mathcal{F} eine distanzkontrahierende Abbildung ist, gilt für $k, p \in \mathbb{N}$

$$\begin{aligned} d(A_{k+p}, A_{k+p-1}) &\leq q \cdot d(A_{k+p-1}, A_{k+p-2}) \leq \\ &\leq q^2 \cdot d(A_{k+p-2}, A_{k+p-3}) \leq \\ &\vdots \\ &\leq q^{p-1} \cdot d(A_{k+1}, A_k) \end{aligned}$$

¹¹Stefan BANACH (1892-1945)

und weiter

$$\begin{aligned} d(A_{k+1}, A_k) &\leq q \cdot d(A_k, A_{k-1}) \leq \\ &\leq q^2 \cdot d(A_{k-1}, A_{k-2}) \leq \\ &\vdots \\ &\leq q^k \cdot d(A_1, A_0) \end{aligned}$$

Nun wird die Dreiecksungleichung auf die Distanz $d(A_{k+p}, A_k)$ angewendet:

$$d(A_{k+p}, A_k) \leq d(A_{k+p}, A_{k+p-1}) + d(A_{k+p-1}, A_{k+p-2}) + \dots + d(A_{k+1}, A_k)$$

Mit Hilfe der im ersten Schritt ablesbaren Ungleichungen

$$\begin{aligned} d(A_{k+p}, A_{k+p-1}) &\leq q^{p-1} \cdot d(A_{k+1}, A_k) \\ q \cdot d(A_{k+p-1}, A_{k+p-2}) &\leq q^{p-1} \cdot d(A_{k+1}, A_k) \\ \Leftrightarrow d(A_{k+p-1}, A_{k+p-2}) &\leq q^{p-2} \cdot d(A_{k+1}, A_k) \\ &\vdots \end{aligned}$$

ergibt sich

$$d(A_{k+p}, A_k) \leq (q^{p-1} + q^{p-2} + \dots + 1) \cdot d(A_{k+1}, A_k).$$

Unter Verwendung der Summenformel $s_n = a_1 \cdot \frac{1-q^n}{1-q}$ für geometrische Folgen, wobei hier $a_1 = 1$, $q = q$ und $n = p$ (von $1 = q^0$ bis q^{p-1} sind es p Glieder) ist, erhält man

$$d(A_{k+p}, A_k) \leq \frac{1-q^p}{1-q} \cdot d(A_{k+1}, A_k).$$

Da $0 < q < 1$ gilt, ist $1 - q^p < 1$, weshalb man schreiben kann:

$$d(A_{k+p}, A_k) \leq \frac{1}{1-q} \cdot d(A_{k+1}, A_k).$$

Wird nun die Ungleichung $d(A_{k+1}, A_k) \leq q^k \cdot d(A_1, A_0)$ aus dem ersten Schritt dieses Beweises angewendet, ergibt sich

$$d(A_{k+p}, A_k) \leq \frac{q^k}{1-q} \cdot d(A_1, A_0). \quad (3.3)$$

Nun findet sich für jede beliebig kleine positive reelle Zahl ε ein $k \in \mathbb{N}$ so, dass $d(A_{k+p}, A_k) < \varepsilon$. Damit ist A_0, A_1, A_2, \dots eine Cauchy-Folge (s. Kap. 3.1.2 auf S. 11). Diese konvergiert in einem vollständigen metrischen Raum X gegen ein Element A_∞ aus X . Also gilt

$$\lim_{n \rightarrow \infty} \mathcal{F}^n(A_0) = A_\infty. \quad (3.4)$$

Schliesslich beweist

$$A_\infty = \lim_{n \rightarrow \infty} A_{n+1} = \lim_{n \rightarrow \infty} \mathcal{F}(A_n) = \mathcal{F}(\lim_{n \rightarrow \infty} A_n) = \mathcal{F}(A_\infty),$$

dass A_∞ ein Fixpunkt der stetigen Abbildung \mathcal{F} ist.

Nun muss noch gezeigt werden, dass es höchstens einen Fixpunkt gibt: Angenommen, es gebe zwei verschiedene Fixpunkte $S, T \in X$, dann gilt $d(S, T) = d(\mathcal{F}(S), \mathcal{F}(T))$, weil beides Fixpunkte sind, die sich unter der Abbildung \mathcal{F} nicht verändern. Weil die Abbildung \mathcal{F} distanzkontrahierend ist, gilt $d(\mathcal{F}(S), \mathcal{F}(T)) \leq q \cdot d(S, T)$. Nun ist aber $q < 1$ und damit $q \cdot d(S, T) < d(S, T)$. Wird nun zusammengefasst, ergibt sich der Widerspruch $d(S, T) < d(S, T)$. Damit ist gezeigt, dass die distanzkontrahierende Abbildung \mathcal{F} genau einen Fixpunkt besitzt. \square

3.4 Konsequenzen des Banach'schen Fixpunktsatzes

Nun übertragen wir die Ergebnisse des Banach'schen Fixpunktsatzes auf iterierte Funktionensysteme in der euklidischen Ebene, das heisst auf den vollständigen metrischen Raum $(\mathcal{C}(\mathbb{R}^2), h)$ und den distanzkontrahierenden Hutchinson-Operator \mathcal{F} . Dabei stützen wir uns auf [PEI/JÜR/SAU] und [FAL].

Zentral ist natürlich, dass ein eindeutiger Fixpunkt $\in \mathcal{C}(\mathbb{R}^2)$ der Abbildung \mathcal{F} existiert und dass dieser durch fortgesetzte Iteration erreicht wird (s. Gleichung (3.4)). Der Fixpunkt ist eine nicht leere kompakte Teilmenge von \mathbb{R}^2 und entspricht der in Kap. 1.2 (s. S. 1) kennengelernten *invarianten* Limesmenge des entsprechenden IFS, also beispielsweise der Cantor-Menge oder dem Sierpinski-Dreieck.

A_∞ ist die Limesmenge der Folge A_0, A_1, A_2, \dots mit $A_{n+1} = \mathcal{F}(A_n)$ von nicht leeren kompakten Teilmengen von \mathbb{R}^2 , das heisst, die Hausdorff-Distanz $h(A_n, A_\infty)$ strebt für $n \rightarrow \infty$ gegen 0. Dies bedeutet, dass $\mathcal{F}^n(A_0)$ für grösser werdendes n eine zunehmend gute Approximation für A_∞ liefert. Diese Annäherung wird *Präfraktal* genannt, falls A_∞ ein Fraktal ist.

Zudem ist es wichtig, zu erkennen, dass mit einer beliebigen Startmenge $A_0 \in \mathcal{C}(\mathbb{R}^2)$ begonnen werden kann. Durch fortgesetzte Iteration wird die Invariantmenge erreicht, und das unabhängig von der Startmenge!

Wenn die Invariantmenge A_∞ als Startmenge gewählt wird, geschieht gar nichts, denn sie ist ja der Fixpunkt der Abbildung \mathcal{F} .

Was ebenfalls aus dem Beweis herausgelesen werden kann, ist eine a priori Abschätzung für die Entfernung von A_n zur Invariantmenge: Geht in Gleichung (3.3) p gegen ∞ , dann ergibt sich $d(A_n, A_\infty) \leq \frac{q^n}{1-q} \cdot d(A_1, A_0)$. Damit ist bei bekanntem Kontraktionsfaktor und bekanntem Abstand von A_0 und A_1 , das heisst nach einer Iteration, voraussagbar, wie gross n sein muss, wenn A_n innerhalb eines vorgegebenen Abstands von A_∞ liegen soll. Da der Kontraktionsfaktor von \mathcal{F} dem grössten Kontraktionsfaktor q_i der N Abbildungen f_i entspricht (s. Kap. 3.2.5 auf S. 18), ist die Schnelligkeit der Konvergenz nur von der „schlechtesten“ Kontraktion abhängig.

Mit Hilfe des Banach'schen Fixpunktsatzes wurde nun also bewiesen, *dass iterierte Funktionensysteme eindeutige (nicht leere) kompakte Mengen definieren*. Dies bedeutet, dass sich Invariantmengen als IFS darstellen lassen und umgekehrt. Die Invariantmengen sind häufig Fraktale, doch mit einem IFS kann beispielsweise auch ein gewöhnliches Quadrat erzeugt werden. Was die entstehenden Bilder jedoch immer sind, ist *selbstaffin*, das heisst, sie sind aus Teilen aufgebaut, welche Ähnlichkeiten mit dem Ganzen aufweisen.

Weil sich das Endbild A_∞ im Falle von affin-linearen Transformationen mit $6 \cdot N$ reellen Zahlen kodieren lässt, sind IFS sehr vielversprechend für die Bildkomprimierung.

4 Mathematische Grundlagen III: Konzepte für die graphische Umsetzung

Dieser Teil behandelt die Grundlagen für die Visualisierung von Fraktalen am Computer. Dabei wird es hauptsächlich um das Chaosspiel und die Möglichkeit der Einfärbung gehen.

4.1 Deterministisch und zufällig iteriertes Funktionensystem

Obwohl ein Fraktal wegen seiner Feinstruktur in unendlicher Tiefe natürlich auch mit Computer nicht dargestellt werden kann, lassen sich zumindest Näherungen sehr gut visualisieren.

Bei der Verwendung von IFS liegt es nahe, ausgehend von einer beliebigen Startmenge den Computer eine gewisse Anzahl Iterationen durchführen und die n -te Iterierte darstellen zu lassen. In diesem Fall spricht man von einem *deterministisch iterierten Funktionensystem*. Diese Methode wird jedoch sehr schnell sehr rechenintensiv, denn nach n Schritten besteht das zu einem IFS mit N Abbildungen gehörende Präfraktal bereits aus N^n Teilmengen. Selbst wenn als Startmenge nur ein Punkt genommen wird, steigt die Anzahl Teile derart, dass die Präfraktale nur für kleine n berechnet werden können.

Deutlich effizienter ist folgendes Vorgehen, das *zufällig iterierte Funktionensystem*, welches wir hier [PEI/JÜR/SAU] und teilweise [FAL] folgend betrachten: Es wird ein beliebiger Startpunkt $z_0 \in \mathbb{R}^2$ definiert. Aus $\{f_1, f_2, \dots, f_N\}$ wird zufällig eine Abbildung ausgewählt und auf z_0 angewendet. Das Resultat ist der Punkt z_1 . Auch auf diesen wird eine zufällig ausgewählte Abbildung angewendet usw. Es ist also $z_n := f_i(z_{n-1})$, wobei i zufällig aus $\{1, 2, \dots, N\}$ ausgewählt wird. Die Punkte liegen bald zufällig über das gesamte Fraktal verteilt und gelangen zudem beliebig nahe an die Punkte des echten Fraktals. Wird diese Iteration genügend lange fortgesetzt, erhält man eine gute Annäherung des Fraktals.

Diese Methode zur Darstellung von durch iterierte Funktionensysteme definierten Fraktalen wird wegen der zufälligen Wahl der jeweiligen Abbildung auch *Chaosspiel* genannt.

Es ist wünschenswert, dass der Startpunkt zum Fraktal gehört, denn in diesem Fall liegen auch alle weiteren Punkte im Fraktal, weil dieses als Fixpunkt des Hutchinson-Operators unter den Transformationen $\{f_1, f_2, \dots, f_N\}$ unverändert bleibt.

Nun soll begründet werden, weshalb das Chaosspiel funktioniert. Dazu zuerst eine Vorüberlegung (s. Abb. 4.1):

Ein IFS mit zwei Abbildungen f_1 und f_2 sei gegeben. Die Startmenge A_0 kann beliebig gewählt werden, solange sie die Invariantmenge vollständig überdeckt.¹²

Nach einem Iterationsschritt liegt die Menge A_1 , bestehend aus den beiden Bildern $f_1(A_0)$ und $f_2(A_0)$, vor. Diese seien mit U_1 und U_2 bezeichnet.

Wird der Hutchinson-Operator auf A_1 angewendet, liefert dies die aus vier Teilen aufgebaute Menge A_2 . Diese Teile werden mit U_{11}, U_{12}, U_{21} und U_{22} bezeichnet. Dabei ist beispielsweise der Teil U_{21} entstanden, indem auf das Bild von A_0 unter f_1 die Abbildung f_2 angewendet wurde, das heißt $f_2(f_1(A_0)) = f_2 \circ f_1(A_0)$. Die Menge U_{21} ist in Abb. 4.1 mit Punkten markiert.

Nach n Iterationsschritten entsprechen die 2^n Teile, aus denen das Präfraktal aufgebaut ist, allen $U_{i_1 i_2 \dots i_n}$, die sich mit $1 \leq i_j \leq 2$ konstruieren lassen, oder,

¹²Dies muss gefordert werden, damit im Verlauf der Konstruktion eine gewisse Eigenschaft sichergestellt ist (s. S. 23)

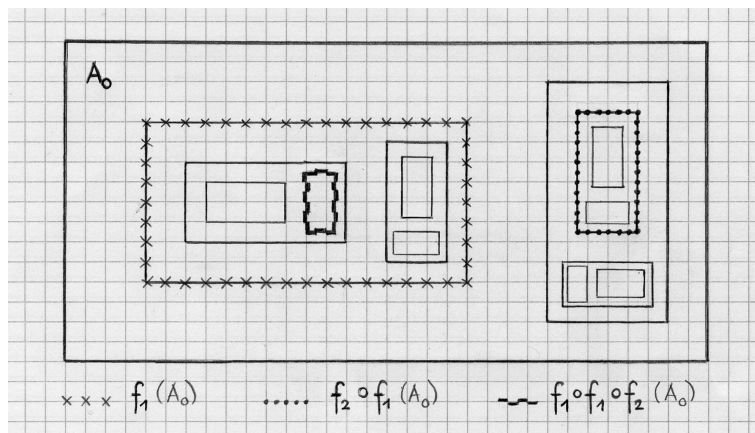


Abbildung 4.1: Die Resultate der ersten drei Iterationsschritte eines IFS mit zwei Abbildungen. Hervorgehoben sind U_1 (Kreuze), U_{21} (Punkte) und U_{112} (Balken).

anders gesagt, allen U_{\dots} mit einem n -stelligen Index, der aus den Ziffern 1 und oder 2 aufgebaut ist.

Der Index gibt dabei an, wie der entsprechende Teil zu Stande gekommen ist: Die hinterste Zahl ist die zuerst ausgeführte Abbildung, die vorderste die letzte. Es gilt deshalb $U_{i_1 i_2 \dots i_n} = f_{i_1} \circ f_{i_2} \circ \dots \circ f_{i_n}(A) = f_{i_1}(f_{i_2}(\dots(f_{i_n}(A))))$.

Wird der Index von vorne nach hinten gelesen, ist er eine Art „Adresse“. Sucht man z. B. U_{112} , schaut man im unter f_1 entstandenen Bild der Startmenge, innerhalb dessen wiederum im „Teil 1“ und findet darin letztlich am Platz des „Teils 2“ das gesuchte U_{112} , welches in Abb. 4.1 mit Balken markiert ist.

Bei Verallgemeinerung dieser Vorüberlegung erhält man für die n -te Iterierte eines IFS mit N Abbildungen

$$\mathcal{F}^n(A_0) = \bigcup_{I_n} f_{i_1} \circ f_{i_2} \circ \dots \circ f_{i_n}(A_0) = \bigcup_{I_n} f_{i_1}(f_{i_2}(\dots(f_{i_n}(A_0))))$$

I_n ist dabei die Menge aller Folgen (i_1, i_2, \dots, i_n) (mit n Gliedern), deren Glieder natürliche Zahlen nicht grösser als N sind.

Weil A_∞ durch fortgesetzte Iteration erreicht wird, gibt es für jeden Punkt $x \in A_\infty$ mindestens eine Folge (i_1, i_2, \dots) , so dass x für jede Iterationsstufe n im Teil $U_{i_1 i_2 \dots i_n}$ liegt, also so, dass $x \in f_{i_1} \circ f_{i_2} \circ \dots \circ f_{i_n}(A_0)$ gilt.¹³

Die Folge muss übrigens nicht eindeutig sein. Wenn die Vereinigung der unter den N Abbildungen entstehenden Bilder nicht disjunkt ist, gibt es Punkte, zu denen mehrere „Wege“ führen. Nur wenn sich die Bilder jeweils weder berühren noch überschneiden, entsteht eine *total unzusammenhängende* Limesmenge, deren Punkte eindeutige „Adressen“ besitzen. Eine solche Limesmenge ist beispielsweise die Cantor-Menge (s. Bsp. 1.1 auf S. 2).

Zu jedem beliebigen Punkt $x \in A_\infty$ existiert aber immer mindestens eine

¹³Für die Gewährleistung dieser Eigenschaft mussten wir voraussetzen, dass die Startmenge die gesamte Invariantmenge überdeckt (s. S. 22)

Folge (i_1, i_2, \dots) . Deshalb gilt

$$A_\infty = \bigcup \{x_{i_1 i_2 \dots}\} \text{ mit } x_{i_1 i_2 \dots} = \lim_{n \rightarrow \infty} f_{i_1} \circ f_{i_2} \circ \dots \circ f_{i_n}(A_0). \quad (4.1)$$

Dieser Ausdruck für $x_{i_1 i_2 \dots}$ ist unabhängig von A_0 , wenn $f_i(A_0)$ für alle i in A_0 enthalten ist.

Dass die Punktfolge bei einer genügend grossen Anzahl Iterationen jedem beliebigen Punkt $x_{i_1 i_2 \dots}$ der Invariantmenge beliebig nahe kommen kann, lässt sich folgendermassen zeigen: Gesucht ist ein z_n , das von $x_{i_1 i_2 \dots}$ nicht weiter als $\varepsilon > 0$ entfernt ist. Der Punkt $x_{i_1 i_2 \dots}$ ist für jedes n in der Menge $U_{i_1 i_2 \dots i_n} = f_{i_1} \circ f_{i_2} \circ \dots \circ f_{i_n}(A_\infty)$ enthalten. Es gilt $U_{i_1} \supset U_{i_1 i_2} \supset \dots$. Mit Hilfe der Kontraktionsfaktoren q_1, q_2, \dots, q_N der einzelnen Transformationen kann der jeweilige Durchmesser von $U_{i_1 i_2 \dots i_n}$ abgeschätzt werden. Er ist nie grösser als $q_{i_1} \cdot q_{i_2} \cdot \dots \cdot q_{i_n}$ -mal der Durchmesser der Startmenge. Weil alle Kontraktionsfaktoren kleiner als 1 sind, wird dieser Durchmesser bei genügend grossem n kleiner als ε . Dies bedeutet, dass alle Punkte, deren „Adressen“ mit $i_1 i_2 \dots i_n$ beginnen, einen Abstand von höchstens ε vom vorgegebenden Punkt $x_{i_1 i_2 \dots}$ haben. Die Wahrscheinlichkeit, dass eine bestimmte Sequenz von ausgelosten Abbildungen der Länge n mit $i_1 i_2 \dots i_n$ übereinstimmt, ist $\frac{1}{N^n} > 0$. Bei genügend Iterationsschritten kann also jedem Punkt der Limesmenge beliebig nahe gekommen werden.

Aufgrund dieser Überlegungen ist nun ersichtlich, dass das oben beschriebene Chaosspiel zu guten Darstellungen von Fraktalen führt. Die nach und nach durch zufällig gewählte Abbildungen erzeugten Punkte füllen die Limesmenge fast sicher dicht aus.

Natürlich wird für jede Iterationsstufe und damit „Index-Länge“ n nur ein einziger Punkt aus jeweils N^n Möglichkeiten getroffen, doch wegen der beschränkten Auflösung fallen schnell einmal so viele Punkte zusammen, dass dies gar keine Rolle mehr spielt: Wenn der Durchmesser der U_{\dots} unter die Seitenlänge der Pixel fällt, können die Ziffern am rechten Ende der Adresse weggelassen werden, denn diese geben ja nur über den noch präziseren Ort Auskunft. Demnach läuft das zufällig iterierte Funktionensystem am Computer darauf hinaus, dass ein „Fenster“ mit einer gewissen Anzahl s von Stellen über eine Zufallsfolge geschoben wird. Da bei einer sehr langen Zufallsfolge davon ausgegangen werden kann, dass ein grosser Teil der N^s Möglichkeiten auftreten, werden so gut wie alle gesuchten Pixel getroffen.

Ist also die Anzahl Punkte genügend gross gewählt, erhält man einen sehr guten Eindruck vom Fraktal.

4.2 Masse und Verteilungen

In diesem Kapitel geht es um die mathematischen Hintergründe der Färbung von fraktalen Bildern. Dabei spielt der Begriff der Verteilung eine zentrale Rolle. Für den Einstieg betrachten wir jedoch zuerst eine Modellvorstellung.

4.2.1 Es werden nicht alle Punkte gleich häufig getroffen

Wird bei der Umsetzung des Chaosspiels in jedem Iterationsschritt ein schwarzer Punkt ins getroffene Pixel gesetzt, erhält man zwar eine Darstellung des Fraktals, jedoch eine einfarbige. Dabei ist nicht erkennbar, ob ein Punkt mehrmals

getroffen wurde. Um eine Färbung zu gewinnen, ist dies jedoch von Interesse, wie folgendes Beispiel zeigt:

Als Startmenge wird ein Quadrat mit Mittelpunkt im Ursprung und Ecken auf den Koordinatenachsen gewählt. Das IFS besteht aus fünf affin-linearen Abbildungen. Alle sind Kontraktionen mit Kontraktionsfaktor 0.5, unterscheiden sich also nur durch ihre Translationsvektoren. Bei einer Abbildung findet keine Translation statt, bei den vier anderen jeweils um die halbe Diagonalenlänge des Startquadrats entlang einer der Achsenhälften. Der erste Iterationsschritt liefert folgendes Bild: Vier aneinander grenzende Quadrate füllen den Platz des Startquadrats und in der Mitte befindet sich noch ein fünftes Quadrat der gleichen Grösse. Die Limesmenge dieses IFS ist natürlich das Startquadrat und damit nicht fraktal. Bei anderer Betrachtungsweise sind jedoch sehr wohl fraktale Strukturen erkennbar:

Den Iterationsvorgang kann man sich modellhaft als Einschmelzen und Neuformen von irgend einem Material vorstellen. Betreffend das Beispiel von oben bedeutet dies, dass mit einem quadratischen Werkstoffstück begonnen wird. Dieses hat, im Unterschied zu kompakten Teilmengen der euklidischen Ebene, eine gewisse Dicke d und damit auch eine bestimmte Masse. Nun wird das Werkstoffstück eingeschmolzen. Mit je einem Fünftel der Masse wird ein neues Quadrat mit halber Seitenlänge geformt. Die fünf Quadrate werden so zusammengefügt, dass ein grosses Quadrat mit einem „aufgeklebten“ Quadrat in der Mitte entsteht (s. Abb. 4.2, links). Diese Anordnung weist in der Mitte eine Dicke von $\frac{8}{5} \cdot d$ und sonst von $\frac{4}{5} \cdot d$ auf.

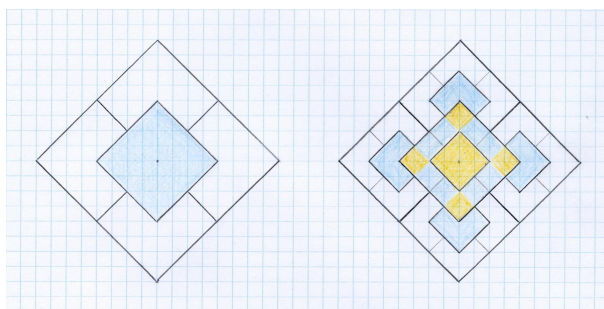


Abbildung 4.2: Die ersten beiden Schritte der beschriebenen Iteration. Blau bedeutet zwei, orange drei Schichten und keine Farbe entspricht einer Schicht.

Der nächste Schritt ist das erneute Einschmelzen des Materials. Wiederum wird die Masse gefünftelt und es werden neue Stücke mit halber Seitenlänge geformt. Diesmal sind es aber nicht einfach Quadrate, sondern Quadrate mit einem „aufgeklebten“ Quadrätchen in der Mitte, also Miniaturen der vorhergehenden Form. Dabei ist jedoch zu beachten, dass das Verhältnis von Seitenlänge zu Dicke nicht gleich bleibt, denn je ein Fünftel der Masse wird zu einem Stück mit einem Viertel der Fläche geformt. Demnach beträgt die Dicke nur noch $\frac{4}{5}$ der vorherigen. Die fünf Stücke werden wiederum zusammengefügt (s. Abb. 4.2, rechts). Dabei darf man sich das Material nicht starr vorstellen, denn es dürfen keine Hohlräume gelassen werden. Wo also das mittlere Stück nicht auf den unteren aufliegt, wird es hinuntergedrückt. Das neue Gebilde hat an unterschiedlichen Stellen je nach Anzahl Schichten verschiedene Dicken von $\frac{16}{25} \cdot d$,

$\frac{32}{25} \cdot d$ oder $\frac{48}{25} \cdot d$.

Wird dieses Einschmelzen und Neuformen fortgesetzt, wird das Gebilde an gewissen Stellen immer dicker, an andern immer dünner (s. Abb. 4.3).

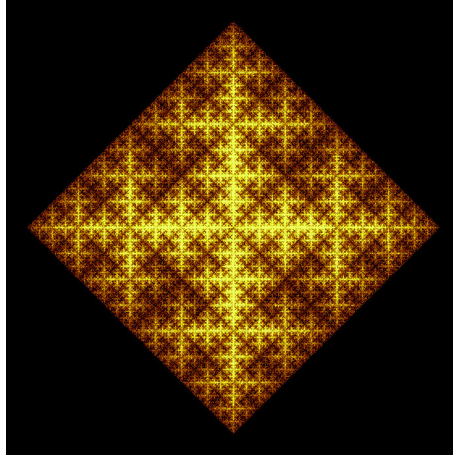


Abbildung 4.3: Die fraktale Struktur der Limesmenge des beschriebenen IFS kommt zum Vorschein, wenn – um in der Sprache der Modellvorstellung zu sprechen – für unterschiedliche Dicken des Materials verschiedene Farbtöne verwendet werden.

Was in diesem Beispiel als Masse des Werkstoffs beschrieben wurde, kann auch mathematisch erfasst werden.

4.2.2 Masse auf Teilmengen von \mathbb{R}^n

In diesem Teil folgen wir [FAL].

Einfach gesagt, ist ein Mass eine Funktion, welche Mengen eine „Grösse“ zuordnet.

Definition 4.1. μ ist ein *Mass* auf \mathbb{R}^n , wenn μ jeder Teilmenge von \mathbb{R}^n eine nicht negative reelle Zahl oder ∞ zuordnet, so dass gilt:

$$\begin{aligned}
 \text{(i)} \quad & \mu(\emptyset) = 0 \\
 \text{(ii)} \quad & \mu(B) \leq \mu(A), \text{ falls } B \subseteq A \text{ ist} \\
 \text{(iii)} \quad & \mu\left(\bigcup_{i=1}^{\infty} A_i\right) \leq \sum_{i=1}^{\infty} \mu(A_i) \text{ für eine abzählbare} \\
 & \text{(oder endliche) Folge von Mengen } A_1, A_2, \dots
 \end{aligned} \tag{4.2}$$

In (i) wird festgelegt, dass die leere Menge das Mass 0 hat. Bedingung (ii) bedeutet, dass das Mass einer Menge B , welche Teilmenge einer Menge A ist, nicht grösser sein kann als dasjenige von A . Bedingung (iii) besagt, dass das Mass einer Vereinigung abzählbar oder endlich vieler Mengen A_i nicht grösser sein kann als die Summe der Masse der einzelnen Mengen.

In (iii) tritt Gleichheit auf, wenn die Mengen A_i paarweise disjunkt sind, das heisst, wenn jeweils zwei Mengen keine gemeinsamen Elemente besitzen. Dies ist natürlich vernünftig, denn wenn eine Menge in endlich oder abzählbar viele sich nicht überlappende Stücke zerteilt wird, dann soll die Summe der Grösse der einzelnen Stücke gleich der Grösse der Menge sein.

Demnach gilt $\mu(A \setminus B) = \mu(A) - \mu(B)$ für $A \supseteq B$, denn A kann als disjunkte Vereinigung $A = B \cup (A \setminus B)$ dargestellt werden, was gemäss der Feststellung von oben $\mu(A) = \mu(B) + \mu(A \setminus B)$ bedeutet.

Der Träger von μ ist die kleinste abgeschlossene Menge T , so dass $\mu(\mathbb{R}^n \setminus T) = 0$ gilt. Den Träger kann man sich als diejenige Menge vorstellen, auf der das Mass „versammelt“ ist. Es wird von einem Mass *auf* einer Menge X gesprochen, wenn X den Träger von μ enthält.

Beispiel 4.1. Das zählende Mass sei folgendermassen definiert: Für jede Teilmenge A von \mathbb{R}^2 sei $\mu(A)$ die Anzahl Punkte in A , wenn A endlich ist, und andernfalls ∞ . In diesem Falle ist der Träger von μ gerade \mathbb{R}^2 . Die Menge $\{1, 2, \dots, 10\}$ hat das Mass 10. Die Kreisscheibe mit Radius 1 um den Ursprung hat das Mass ∞ , denn sie enthält unendlich viele Punkte.

4.2.3 Verteilungen

Eine *Verteilung* ist ein Mass auf einer beschränkten Teilmenge von \mathbb{R}^n , welches die Bedingung $0 < \mu(\mathbb{R}^n) < \infty$ erfüllt. Man kann sich das Mass einer Menge beispielsweise als die Masse der Menge im physikalischen Sinn vorstellen. In dieser Vorstellung kann eine Verteilung konstruiert werden, indem eine endliche Masse in irgendeiner Weise über einer Menge X verstreut wird.

Beispiel 4.2. Es wird eine Verteilung konstruiert, indem ein Punkt a aus \mathbb{R}^2 bestimmt und Folgendes definiert wird: $\mu(A) = 1$, wenn die Menge A den Punkt a enthält, und ansonsten $\mu(A) = 0$. Diese Verteilung kann man sich als in a konzentrierte Punktmasse vorstellen.

4.2.4 Eine Methode zur Konstruktion einer Verteilung auf einer Teilmenge von \mathbb{R}^2

Es sei A_0 eine einzelne beschränkte Menge aus \mathbb{R}^2 mit dem Mass $0 < \mu(A_0) < \infty$. Durch wiederholtes Unterteilen wird eine Verteilung auf A_0 konstruiert:

Mit \mathfrak{A}_n sei jeweils eine Auswahl von Mengen bezeichnet. Für $n = 0$ beinhaltet die Auswahl \mathfrak{A}_0 nur gerade die Menge A_0 . \mathfrak{A}_1 ist eine endliche Auswahl disjunkter Teilmengen U_1, U_2, \dots, U_N von A_0 . Aus jedem U_j werden für die nächste Stufe, das heisst für \mathfrak{A}_2 , endlich viele Mengen $U_{j1}, U_{j2}, \dots, U_{jN}$ ausgewählt. Allgemein ist \mathfrak{A}_n eine Auswahl $U_{\dots 1}, U_{\dots 2}, \dots, U_{\dots N}$, so dass jede Menge $U_{\dots i}$ in einer der Mengen aus \mathfrak{A}_{n-1} enthalten ist und ihrerseits eine endliche Anzahl von Mengen aus \mathfrak{A}_{n+1} enthält.

Die ursprüngliche Masse $\mu(A_0)$ wird auf die Mengen U_1, U_2, \dots, U_N aus \mathfrak{A}_1 so verteilt, dass $\sum_{i=1}^N \mu(U_i) = \mu(A_0)$ gilt. Dann wird jeder der Mengen $U_{j1}, U_{j2}, \dots, U_{jN}$ aus \mathfrak{A}_2 , welche jeweils in einem bestimmten U_j aus \mathfrak{A}_1 liegt, ein Mass zugeordnet, so dass $\sum_{i=1}^N \mu(U_{ji}) = \mu(U_j)$ für jedes j erfüllt ist. Dieses Prinzip wird fortgesetzt.

So wird schliesslich eine Verteilung erreicht. Diese kann nun zu einem Mass auf ganz \mathbb{R}^2 ausgebaut werden:

Für jedes n bezeichne A_n die Vereinigung der Mengen aus \mathfrak{A}_n (A_n ist also selbst *eine* Menge, während \mathfrak{A}_n immer eine *Auswahl* von Mengen meint). Nun wird $\mu(\mathbb{R}^2 \setminus A_n) = 0$ für alle n definiert. Mit \mathfrak{A}_{gesamt} sei die Auswahl von all jenen Mengen bezeichnet, die entweder für irgend ein n zu \mathfrak{A}_n gehören oder die

$\mathbb{R}^2 \setminus A_n$ für irgend ein n sind. Nach all diesen Definitionen ist für jede Menge M aus \mathfrak{A}_{gesamt} ein Mass $\mu(M)$ festgelegt.

Mit $\mu(M) = \inf\{\sum_i \mu(T_i) : M \subseteq \bigcup_i T_i \text{ mit } T_i \in \mathfrak{A}_{gesamt}\}$ ist μ ein Mass auf ganz \mathbb{R}^2 . Es werden also für die Menge, deren Mass gefragt ist, alle Überdeckungen mit Mengen aus \mathfrak{A}_{gesamt} angesehen. Die kleinste Summe von Massen der einzelnen Teile einer Überdeckung ist das gesuchte Mass von M .

Der Träger von μ ist in $\overline{A_\infty}$ enthalten.

4.2.5 Mass beim Chaosspiel

Die Auszählung der relativen Häufigkeit der Punkte, welche im Verlauf des Chaosspiels in Teilmengen der Invariantmenge des IFS fallen, begründet ein Mass. Dieses Mass ist *invariant* und hat als Träger die Invariantmenge. Da es jeder Teilmenge von \mathbb{R}^n einen Wert zuordnet, ist es geeignet für das Einfärben der Darstellung eines Fraktals.

Es kann also mit Hilfe der Wahrscheinlichkeitsverteilung der im Verlauf des Chaosspiel gesetzten Punkte eine Färbung bestimmt werden: In Abhängigkeit der Anzahl Punkte, die in ein Pixel fallen, wird die Farbe gewählt. Die Verteilung der Punkte hat viele Gemeinsamkeiten mit der in Kap. 4.2.1 (s. S. 24) untersuchten Verteilung der Masse des Werkstoffs, denn beide werden durch Iteration erreicht. Demnach ist Abb. 4.3 (s. S. 26) einerseits eine Darstellung der Dicken des Werkstoffs nach zahllosen Iterationsschritten, andererseits aber zugleich eine Visualisierung der Wahrscheinlichkeitsverteilung der Punkte, also des invarianten Masses.

Wegen der iterativen Konstruktion ist die Wahrscheinlichkeitsverteilung der Punkte und damit das invariante Mass wie die Invariantmenge selbstaffin.

Mehr zu diesem invarianten Mass kann in [PEI/JÜR/SAU] (S. 391) nachgelesen werden.

5 Farbmodelle

Im Folgenden sollen einige Farbmodelle vorgestellt werden. Sie sind von Bedeutung für den praktischen Teil dieser Arbeit, denn für das Entwerfen von Farbverläufen gibt es besser geeignete Farbmodelle als das vom Computer verwendete RGB-Farbmodell, denn dieses ist nicht gut an die menschliche Farbwahrnehmung angepasst. Leichter zu handhaben ist beispielsweise das auf den drei Grössen Farbton, Sättigung und Helligkeit basierende HSV-Farbmodell.

5.1 Begriffe

Für diesen Abschnitt stützen wir uns mehrheitlich auf [WIK3].

Mit *Farbraum* bezeichnet man die Menge all derjenigen Farben, die von einem bestimmten Ein- oder Ausgabegerät (Sehsinn, Kamera, Scanner, Bildschirm, Projektor, Drucker usw.) erkannt bzw. dargestellt werden können.

Farbräume sind fast immer dreidimensional. Dies hängt mit der Physiologie des menschlichen Auges zusammen. In diesem gibt es drei verschiedene Arten von Farbrezeptoren, *Zapfen* genannt. Ein *Farbreiz*, das heisst die spektrale Zusammensetzung des gesehenen Lichts, wird deshalb auf drei Grössen, die so-

genannte *Farbvalenz*, abgebildet. Dies ist im ersten Graßmann'schen Gesetz¹⁴ festgehalten, welches besagt, dass jeder Farbeindruck mit genau drei Grundgrößen vollständig beschrieben werden kann.

Die Farben, präziser gesagt Farbvalenzen, eines Farbraumes werden mit Hilfe eines *Farbsystems* quantifiziert. Ein Farbsystem ist meist ein dreidimensionales Koordinatensystem, kann aber auch ein Farbkatalog sein, wie z. B. der Pantone-Farbkatalog. In diesem Fall ist allerdings nur eine diskrete und nicht eine kontinuierliche Darstellung der Farben möglich.

Häufig werden ein Farbsystem „und der entsprechende Farbraum, auf dem das System basiert, nicht unterschieden, sondern zusammengefasst als Farbmodell bezeichnet“ ([GRA]). Ein *Farbmodell* ist demnach der Farbraum in Gestalt eines mathematischen Messraums.

In einem Farbmodell hat jede Farbvalenz ihren *Farbort*. Dieser ist ein Punkt im Raum und deshalb auch als dreidimensionaler Vektor darstellbar.

Ein Zahlentripel legt demnach eine eindeutige Farbvalenz fest, wenn klar ist, in welchem Farbsystem.

Wenn sämtliche Farborte eines Farbmodells in Form eines geometrischen Körpers (Würfel, Kugel, Kegel usw.) dargestellt sind, spricht man von einem *Farbkörper*.

Der Körper des Farbraums wird als *Gamut* bezeichnet. Wird der Gamut im zum Farbraum gehörenden Farbsystem dargestellt, entspricht er dem Farbkörper. In einem anderen Farbsystem hat er jedoch eine andere Form (s. Abb. 5.1).

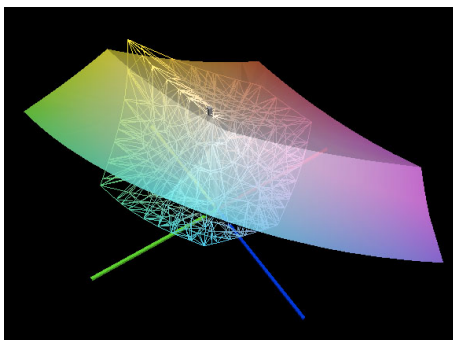


Abbildung 5.1: Der Gamut eines sRGB-Bildschirms (farbig) und der Gamut eines CMYK-Druckers (Gitter), dargestellt im CIELAB-System (a^* -Achse grün, b^* -Achse blau, L^* -Achse nicht sichtbar, wäre senkrecht dazu). Farben, die außerhalb eines Gamuts liegen, können mit der entsprechenden Methode nicht dargestellt werden. Der Drucker kann beispielsweise das helle Grün und das reine Rot des Monitors nicht reproduzieren. Dieser kommt aber seinerseits auch nicht an die Grüntöne der Natur heran, denn der Gamut des menschlichen Auges ist in diesem Bereich noch viel ausgedehnter. [3]

¹⁴Hermann GRASSMANN (1809-1877)

5.2 Einige Farbmodelle

Der Mensch sieht Licht mit Wellenlängen von 380 nm (Violett) bis 780 nm (Rot). Die drei Zapfentypen sind aufgrund der Lage ihres Empfindlichkeitsmaximums im Spektrum benannt: S-Zapfen (kurze (*short*) Wellenlängen, blau), M-Zapfen (mittlere Wellenlängen, grün) und L-Zapfen (lange Wellenlängen, rot). Da der Sehsinn nicht bei jedem Menschen genau gleich ist, hat jeder seinen individuellen Zapfenfarbraum, auch *LMS-Farbraum* genannt.

Viele Farbmodelle basieren auf dem durchschnittlichen Farbraum des menschlichen Sehens. Sie sind nicht von einem spezifischen Gerät abhängig, sondern enthalten sämtliche sichtbaren Farben und zum Teil auch noch theoretische. Ihre Farbräume werden oft als *absolut* bezeichnet. Sie eignen sich als Referenzfarbräume, z. B. für die verlustfreie Konvertierung von Farbinformationen aus einem Farbsystem in ein anderes, wie sie beim Übergang zwischen verschiedenen Geräten erforderlich sein kann.

CIE-Normvalenzsystem:

Ein wichtiges Farbmodell wurde im Jahr 1931 von der CIE (*commission internationale de l'éclairage*, internationale Beleuchtungskommission) entwickelt. Die Grundlage dafür waren aus einem Experiment gewonnene Messwerte. 17 Versuchspersonen mussten die Helligkeit einer roten, einer grünen und einer blauen Lichtquelle so verändern, dass deren Mischfarbe (additive Farbmischung mittels überlappender Projektion) mit einer gegebenen Farbprobe übereinstimmte. Das beanspruchte Sichtfeld betrug 2° , was im Auge der Zone mit der höchsten Dichte der Zapfen entspricht. Mit diesem Experiment wurde der Bereich der vom Menschen wahrnehmbaren Farbreihe erfasst. Deshalb enthält das Farbmodell alle sichtbaren Farben. Die Achsen im CIE-Normvalenzsystem sind mit X , Y und Z bezeichnet.

Genauer zum CIE-Normvalenzsystem kann in [WIK4] nachgelesen werden.

Ergänzend zum CIE-Normvalenzsystem gibt es die CIE-Normfarbtafel (s. Abb. 5.2). Sie ist eine zweidimensionale Darstellung des CIE-XYZ-Farbraums bei konstanter Helligkeit. Die dritte Komponente wird aus den beiden abgelesenen Komponenten x und y und aus der Gleichung $x + y + z = 1$ berechnet.

Ein Nachteil von CIE-Normvalenzsystem und CIE-Normfarbtafel ist, dass sie *nicht gleichabständig* sind. Das bedeutet, dass gleiche Distanzen zwischen Farborten je nach Lage im Koordinatensystem anderen empfindungsgemässen Farbunterschieden entsprechen.

CIELAB-Farbmodell:

Um diesen Nachteil zu beseitigen, entwickelte die CIE im Jahr 1976 das CIELAB-Farbmodell. Dieses ging unter Anwendung einer nicht linearen Transformation aus dem CIE-Normvalenzsystem hervor. Die Gleichabständigkeit ist im CIELAB-Farbmodell ziemlich gut gewährleistet. Der *Farbabstand* ΔE ist ganz einfach die euklidische Distanz. Zudem stimmt die Komponente L^* recht gut mit der nicht linearen Helligkeitswahrnehmung des Menschen überein. Deshalb ist das CIELAB-Farbmodell gut auf die Farbwahrnehmung des Menschen abgestimmt. Jedoch gibt es darin auch „imaginäre“ Farben, also solche, die ausserhalb des Gamuts der menschlichen Wahrnehmung liegen.

Die drei Achsen des CIELAB-Farbsystems werden L^* , a^* und b^* genannt, denn L , a und b sind die Bezeichnungen im *Hunter Lab-Farbraum* von 1948.

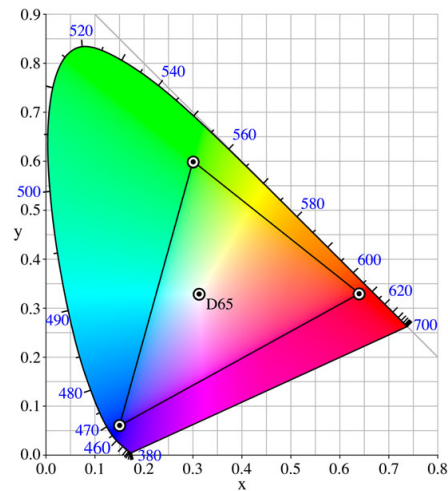


Abbildung 5.2: Die CIE-Normfarbtafel enthält die Gesamtheit der für den Mensch sichtbaren Farben, jedoch ohne Beachtung der Varianten betreffend Helligkeit. Auf der gebogenen *Spektralfarblinie* (blaue Zahlen entsprechen Wellenlängen in nm) liegen die reinen Farben, in der Mitte der von der Beleuchtungssituation (hier: D65) abhängige *Weisspunkt*. Das eingezeichnete Dreieck ist der Gamut des sRGB-Farbraums. [4]

L^* steht für die Helligkeit (Luminanz), a^* und b^* sind die Bezeichnungen für die beiden Achsen von Grün nach Rot bzw. von Blau nach Gelb (s. Abb. 5.3). Diese Anordnung geht auf die Gegenfarbentheorie zurück, nach der sich Gelb und Blau bzw. Grün und Rot gegenseitig ausschliessen. Auf der L^* -Achse liegen die Grautöne von Schwarz bis Weiss. L^* kann Werte von 0 bis 100 annehmen, a^* von -150 bis 100 und b^* von -100 bis 150 .

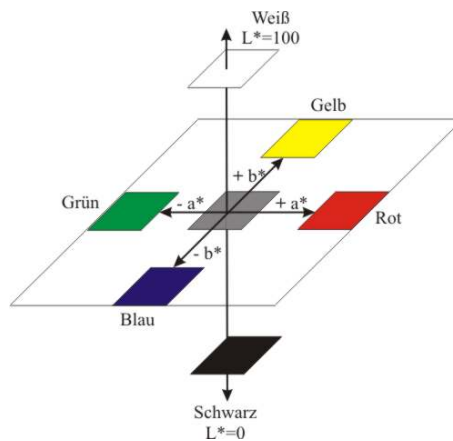


Abbildung 5.3: Schematische Darstellung des CIELAB-Farbsystems. [5]

CIELAB-Werte definieren nur bei Angabe des Weisspunkts (s. Abb. 5.2) konkrete Farben. Der Weisspunkt ist vom Licht abhängig, beispielsweise ist es

für CIE-Normbeleuchtung E der Punkt mit $X = Y = Z$.

Mehr zum CIELAB-Farbmodell kann in [WIK5] und [WIK6] nachgelesen werden.

CMYK- und RGB-Farbmodell:

Andere Farbmodelle basieren auf einem von drei Farben, *Primärvalenzen* genannt, aufgespannten Farbraum. Ein solcher ist allgemein ein Farbraum eines Ausgabegeräts, welches alle Farben aus drei Grundfarben mischt. Zu beachten ist, dass es zwei grundsätzlich verschiedene Arten von Mischen gibt: additive und subtraktive Farbmischung (s. Abb. 5.4). Additiv mischt sich farbiges Licht. Die spektralen Zusammensetzungen, das heisst die Intensitäten in Abhängigkeit von der Wellenlänge, addieren sich. Anders ist es bei sogenannten Körperfarben, Farben von Gegenständen. In diesem Fall ist die Farbmischung subtraktiv. Eine Körperfarbe absorbiert bestimmte Wellenlängen des beleuchtenden Lichts. Wird eine andere Körperfarbe darüber gelegt, absorbiert sie noch weitere Wellenlängen. Ein Bildschirm fällt unter die additive Farbmischung, ein Drucker hingegen unter die subtraktive.

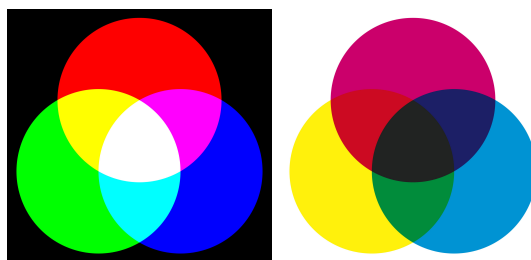


Abbildung 5.4: Additive und subtraktive Farbmischung unterscheiden sich grundsätzlich: Die Primärvalenzen für die additive Mischung sind Rot, Grün und Blau (links), für die subtraktive Cyan, Magenta und Yellow (rechts). Jeweils zwei der Primärvalenzen zusammen ergeben eine Primärvalenz der anderen Art der Farbmischung (Schnittmenge von jeweils zwei Kreisen). Rot, Grün und Blau additiv gemischt entsprechen Weiss, Cyan, Magenta und Yellow zusammen sind Schwarz (Schnittmenge aller drei Kreise). [6] und [7]

Die Primärfarben eines Druckers sind typischerweise Cyan, Magenta und Yellow (CMY). Alle drei zusammen absorbieren das gesamte Licht und ergeben deshalb Schwarz. Dies gilt jedoch nur theoretisch und nicht praktisch. Deshalb gibt es den Vierfarbdruck mit Schwarz (*Key*, da es auch Tiefe schafft) und das dazu gehörende CMYK-Farbmodell.

Der folgende Abschnitt basiert auf [WIK7] und [WIK8].

Bei einem Bildschirm sind die drei Primärfarben Rot, Grün und Blau (RGB). Dies führt zu den RGB-Farbmodellen. Je nach eingesetzten Leuchtstoffen sind die RGB-Farbräume unterschiedlich, doch das Farbsystem ist immer das Gleiche: Die drei Achsen sind R , G und B . Der Wertebereich der Achsen kann beispielsweise zwischen 0 und 1 oder zwischen 0 und 255 liegen (s. Abb. 5.5).

Der Gamut eines RGB-Farbraums in der CIE-Normfarbtafel ist das Dreieck, das man beim Verbinden der drei Farborte der eingesetzten Primärfarben erhält (s. Abb. 5.2 auf S. 31). Die Farben innerhalb des Dreiecks können dargestellt werden, diejenigen ausserhalb nicht.

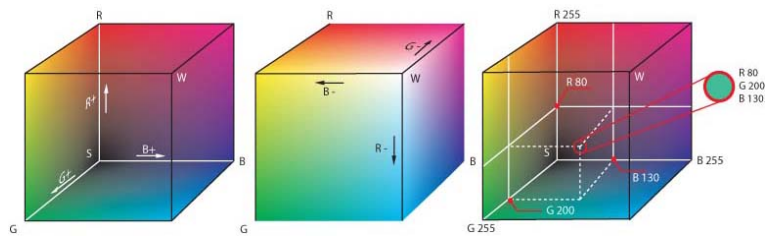


Abbildung 5.5: Ein RGB-Farbraum wird als Würfel visualisiert. Im Ursprung des Koordinatensystems liegt Schwarz, in der gegenüberliegenden Ecke Weiss. Auf der Diagonale dazwischen befinden sich die Unbunttöne, also Grau in all seinen Abstufungen. Entlang der drei Achsen sind Rot-, Grün- und Blauanteil aufgetragen. Demnach sind in den drei auf den Achsen liegenden Ecken Rot, Grün und Blau zu finden. In den drei verbleibenden Ecken liegen Gelb, Cyan und Magenta. [8]

Für die Festlegung eines konkreten RGB-Farbraums braucht es drei definierte Primärvalenzen und definierte Maximalwerte der Intensität. Die Maximalwerte sind nötig, weil die Komponenten nicht absolute Werte, sondern Rot-, Grün- und Blauanteil sind.

HSV-Farbmodell:

Für diesen Abschnitt folgen wir im Wesentlichen [WIK9].

Ein weiteres Farbmodell ist das HSV-Farbmodell. Mit den drei Grundgrößen Farbton (*hue*), Sättigung (*saturation*) und Hellwert (*value*) ist es gut auf das Farbempfinden abgestimmt. Es ist im HSV-System leichter, eine Farbe zu finden, als beispielsweise im RGB-System.

HSV-Farbsysteme gibt es verschiedene. Je nach System ist der Farbkörper ein Zylinder, eine Halbkugel, ein Kegel usw. Natürlich sind auch die Farborte je nach System anders.

Hier soll dasjenige HSV-Farbsystem vorgestellt werden, das als Zylinder visualisiert wird (s. Abb. 5.6). Der Farbton wird als Farbwinkel auf dem Farbkreis, das heisst als Winkel bis 360° , angegeben. Sättigung und Hellwert werden in Prozent oder von 0 bis 1 angegeben. Je kleiner die Sättigung, desto mehr pures Weiss wird hinzugemischt.

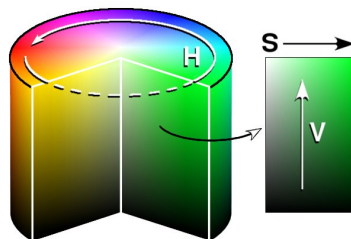


Abbildung 5.6: Das HSV-Farbmodell dargestellt als Zylinder. H steht für den Farbton, S für die Sättigung und V für den Hellwert. [9]

Teil II

Anwendungen

6 Praktischer Teil

Im praktischen Teil meiner Arbeit habe ich mich mit der Bildgestaltung mit Hilfe von IFS beschäftigt. Ich habe viele verschiedene IFS und Färbungen ausprobiert. Schliesslich habe ich mich dafür entschieden, auf das Ziel, ein Bild zum Thema „Die vier Jahreszeiten“ zu entwickeln, hinzuarbeiten.

6.1 Verbindung mit der Theorie

Wie mit Hilfe des Banach'schen Fixpunktsatzes gezeigt wurde, definiert jedes IFS mit einem Hutchinson-Operator, welcher eine Kontraktion auf $(\mathcal{C}(\mathbb{R}^2), h)$ ist, eine eindeutige Invariantmenge $\in \mathcal{C}(\mathbb{R}^2)$, also ein Bild. Man kann demnach mit IFS Bilder gewinnen und diese mit dem Chaosspiel visualisieren. Ein IFS kann zusammengestellt werden, indem $N \in \mathbb{N}$ Kontraktionen auf (\mathbb{R}^2, d_E) ausgewählt werden, denn diese bilden, wie in Kap. 3.2.5 (s. S. 18) gesehen, einen auf $(\mathcal{C}(\mathbb{R}^2), h)$ distanzkontrahierenden Hutchinson-Operator. Für den Fall, dass die Transformationen $\{f_1, f_2, \dots, f_N\}$ alle affin-linear sind, haben wir in Kap. 2.3 (s. S. 6) geklärt, wann es sich um Kontraktionen handelt. Zudem sind affin-lineare Abbildungen für den Computer geeignet, da sie sich mit einer erweiterten Matrix darstellen lassen (s. Gleichung (2.2) auf S. 5).

Aus diesen Gründen habe ich mich im praktischen Teil meiner Arbeit mit aus affin-linearen Abbildungen bestehenden IFS auseinandergesetzt.

Für die Darstellung der Fraktale habe ich ein C++-Programm geschrieben (s. Kap. A.2 auf S. 42). Dieses ist eine Umsetzung des in Kap. 4.1 (s. S. 22) beschriebenen Chaosspiels. Es führt also eine Iteration durch, bei der jeweils eine Abbildung zufällig ausgewählt und dann auf den aktuellen Punkt angewendet wird, was zum neuen aktuellen Punkt führt.

Dass der Startpunkt im Fraktal liegt, wird sichergestellt, indem der Fixpunkt der Transformation f_1 gewählt wird, denn die Fixpunkte der einzelnen Transformationen f_i eines IFS gehören immer zur entsprechenden Limesmenge und sind zudem durch Auflösen des Gleichungssystems $x = a \cdot x + b \cdot y + e; y = c \cdot x + d \cdot y + f$ berechenbar. Es ist deshalb geeignet, einen von ihnen als Startpunkt zu wählen.

In jedem Iterationsschritt erfolgt die Anwendung einer ausgelosten Matrix auf den aktuellen Ortsvektor. In einem Array mit Werten für jedes Pixel wird jeweils die zum getroffenen Pixel gehörende Zahl um 1 erhöht.

Dieses Array wird schliesslich ausgewertet und jedem Pixel wird in Abhängigkeit des Wertes, also der Anzahl Treffer, seine Farbe zugeordnet. So entsteht ein Bild.

6.2 Vorgehen

Als Erstes experimentierte ich mit verschiedenen IFS. Dank des Computerprogramms [BAU/WEB] war diese Arbeit ziemlich anschaulich. Ich konnte herumspielen und zu Bildern gelangen, welche meinen Vorstellungen entsprechen.

Wenn ich jeweils ein IFS gefunden hatte, das ich weiterverfolgen wollte, konnte ich bei jeder Abbildung die 6 reellen Zahlen für die erweiterte Matrix (s. (2.2) auf S. 5) herauskopieren.

Das IFS-Programm ist im Anhang (s. Kap. A.1 auf S. 41) mit Illustrationen noch etwas genauer beschrieben.

Die erweiterten Matrizen setzte ich dann jeweils in meinem C++-Programm (s. Kap. A.2 auf S. 42) für das zufällig iterierte Funktionensystem ein. Die RGB-Tripel mit der Farbinformation für jedes einzelne Pixel werden von diesem Programm in eine ppm-Datei ausgegeben.

Aus der ppm-Datei konnte ich jeweils mit Hilfe des Plugins [ROS] für Adobe Photoshop ([ADO]) ein JPEG-Bild gewinnen.

6.3 Gestaltungsmöglichkeiten

Es gibt verschiedene Grössen, welche die entstehenden Bilder beeinflussen. Am wichtigsten sind natürlich die Transformationen, also die Matrizen. Wie gesagt, habe ich diese mit Hilfe des IFS-Programms gewählt. Doch auch mit meinem C++-Programm (s. Kap. A.2 auf S. 42) hatte ich verschiedene Gestaltungsmöglichkeiten. Dazu gehören einerseits verschiedene Einstellungen für gewisse Parameter, andererseits unterschiedliche Farbverläufe.

6.3.1 Anzahl Iterationen

Eine der einstellbaren Grössen ist die Anzahl Iterationen und damit die Anzahl Punkte. Diese ist dafür verantwortlich, wie dicht die Invariantmenge überdeckt wird (s. Kap. 4.1 auf S. 22).

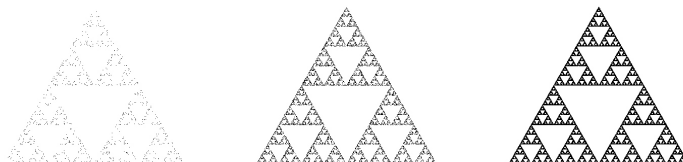


Abbildung 6.1: Die Anzahl Iterationen entspricht der Anzahl der gesetzten Punkte. Diese ist nicht unwesentlich für die Darstellung. Links: 1'000. Mitte: 10'000. Rechts: 100'000. Bildgrösse jeweils 500×500 .

Wenn eine Färbung definiert ist, beeinflusst die Einstellung der Anzahl Iterationen natürlich auch die Farben, denn diese hängen ja davon an, in welchem Pixel wieviele Punkte landen.

6.3.2 Wahrscheinlichkeiten

Wenn für die verschiedenen Abbildungen $\{f_1, f_2, \dots, f_N\}$ unterschiedliche Wahrscheinlichkeiten p_1, p_2, \dots, p_N mit $0 \leq p_i \leq 1$ und $\sum_{i=1}^N p_i = 1$ definiert werden, können beeindruckende Effekte betreffend Punktdichte und Färbung erzielt werden. Ausführlichere Informationen dazu finden sich z. B. in [PEI/JÜR/SAU].

In der Modellvorstellung aus Kap. 4.2.1 (s. S. 24) bedeuten unterschiedliche Wahrscheinlichkeiten, dass die Masse nach dem Einschmelzen ungleichmässig verteilt wird.

Beim Chaosspiel geht die Iteration so wie zuvor, jedoch mit dem Unterschied, dass die Abbildungen mit einer bestimmten Wahrscheinlichkeit ausgewählt werden.

In gewissen Teilen des Fraktals kommen auf diese Weise mehr Punkte zu liegen, in anderen weniger als bei gleichen Wahrscheinlichkeiten für alle Abbildungen. In einem Teil $U_{i_1 i_2 \dots i_n}$ liegt derjenige Anteil der Gesamtanzahl Punkte, der $p_{i_1} \cdot p_{i_2} \cdot \dots \cdot p_{i_n}$ entspricht.

Demnach ist das Festlegen von Wahrscheinlichkeiten für die verschiedenen Transformationen ein geeignetes Mittel, um das Bild zu gestalten, vor allem auch, weil damit bei einer definierten Färbung die Farbverteilung beeinflusst wird.

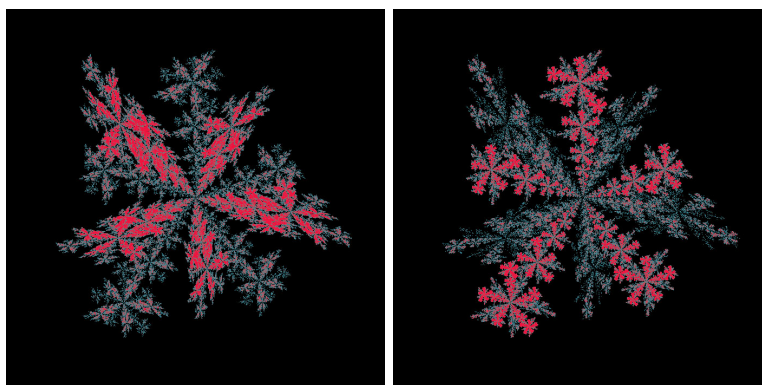


Abbildung 6.2: Die gewählten Wahrscheinlichkeiten für die unterschiedlichen Abbildungen haben einen ziemlich starken Einfluss auf das entstehende Bild. Links: $p_1 = 0.80, p_2 = 0.03, p_3 = 0.17$. Rechts: $p_1 = 0.80, p_2 = 0.17, p_3 = 0.03$.

Teilweise ist die Wahl von verschiedenen Wahrscheinlichkeiten nötig, da andernfalls kein befriedigendes Ergebnis erzielt wird. Dies ist vor allem bei deutlich unterschiedlichen Kontraktionsfaktoren der Fall, beispielsweise beim Farn.

6.3.3 Färbungen

Das Ausprobieren von verschiedenen Farbverläufen war natürlich im Hinblick auf die Gestaltung von Bildern zum Thema „Die vier Jahreszeiten“ ziemlich wichtig.

Weil das Bestimmen von Farbverläufen im nicht gleichabständigen RGB-Farbmodell nicht ideal ist, habe ich das CIELAB- und das HSV-Farbmodell beigezogen (s. Kap. 5 auf S. 28).

Ich habe ein Programm geschrieben, welches Farbverläufe berechnet, indem es von einem Farbort im CIELAB-System ausgehend für jede weitere Farbe jeweils einen festgelegten Vektor dazuzählt. Auf diese Weise sind die euklidischen Abstände und damit die Farbabstände immer gleich, was zu einem regelmässigen Farbverlauf führt. Schliesslich werden die CIELAB-Werte in RGB-Werte umgerechnet. Das Programm befindet sich auf der beiliegenden CD.

Ein ähnliches Programm habe ich für das HSV-Farbmodell geschrieben. Auch dieses gibt RGB-Werte aus. Es ist im Kap. A.3 (s. S. 44) abgedruckt.

Mit diesen beiden Programmen konnte ich also diverse RGB-Farbverläufe erzeugen und diese jeweils im C++-Programm für das zufällig iterierte Funktionensystem als Grundlage für die Einfärbung der Darstellung des Fraktals verwenden.

Einige wenige Farbverläufe stellte ich mit Hilfe des Farbwählers in Microsoft Word zusammen.

7 Resultat

Für mein Bild habe ich das Thema „Die vier Jahreszeiten“ gewählt, weil diese mich immer wieder faszinieren. Ich mag natürlich gewisse Eigenheiten von Frühling, Sommer, Herbst und Winter, vor allem aber die Abwechslung.

Zudem finde ich es schön, dass man mit den verschiedenen Jahreszeiten ganz unterschiedliche Farben assoziiert. Dieser Umstand ist interessant für die Gestaltung eines viergeteilten Bildes. Ebenfalls finden sich gewisse Formen, welche man sofort einer bestimmten Jahreszeit zuordnet.

Das Bild ist auf S. 39 zu finden.

Für den *Frühling* habe ich ein Bild mit fünfblättrigen Blüten gestaltet. Die Färbung enthält viel Grün, eine Farbe, die ich ganz klar dem Frühling und den dazu gehörenden frischen Blättern zuordne. Da aber die Blüten natürlich nicht grün sein sollen, gibt es einen farbigen Übergang, der schliesslich bei hellem Blau endet. Ein weiterer Aspekt des Bildes ist die aus dem Mittelpunkt hervorquellende Form, die meiner Meinung nach gut zum Frühling passt.

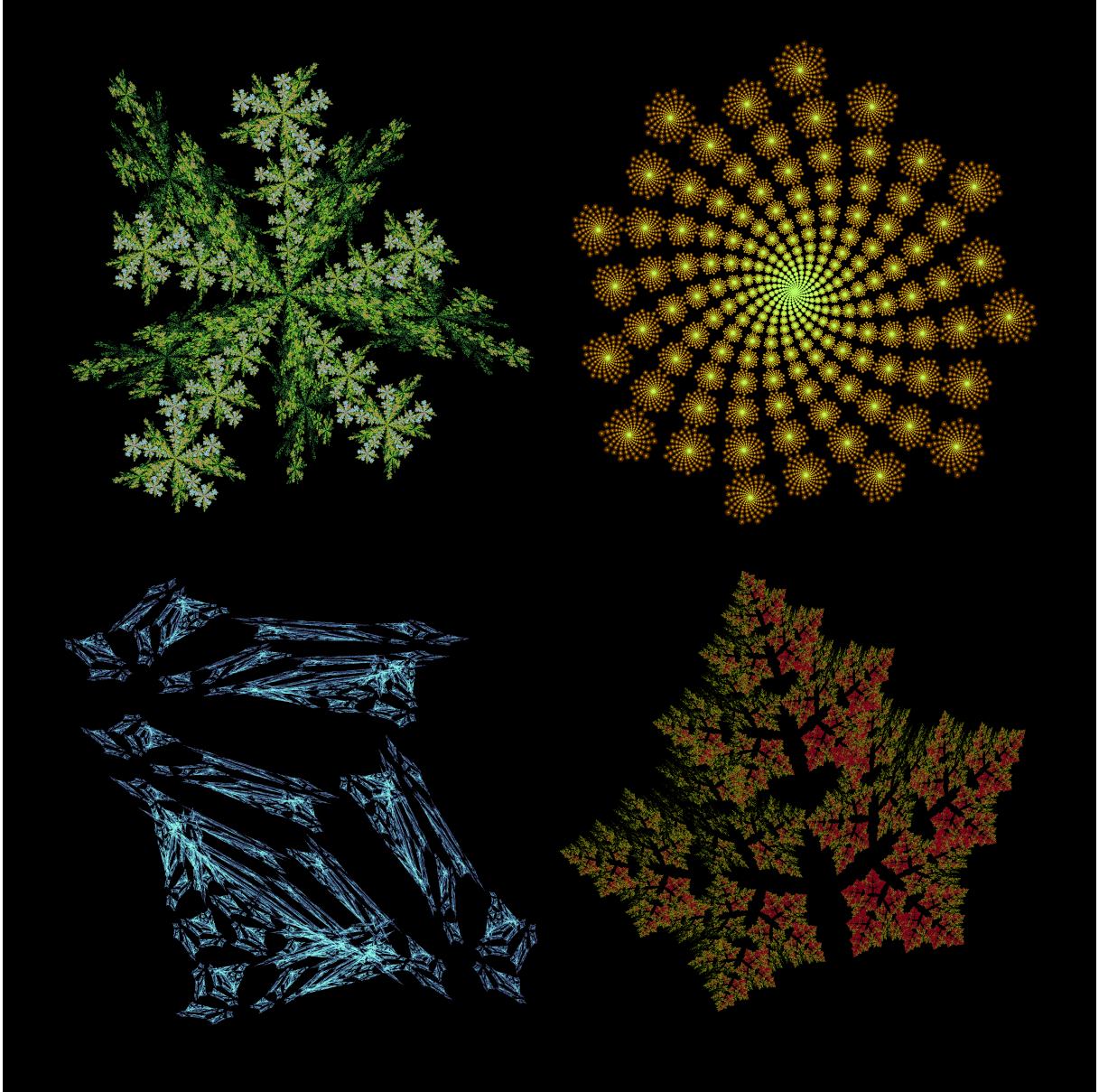
Das Bild zum *Sommer* ist in denjenigen Farben gehalten, welche ich mit der Sonne und der Wärme assoziiere. Damit das Gleissende der Sonne etwas zum Ausdruck kommt, geht der Farbverlauf bis zu sehr hellem Gelb. Diese Idee fortführend, habe ich eine runde und regelmässige Form gewählt. Diese hat zugleich Ähnlichkeiten mit dem Innern einer Sonnenblume.

Für die Darstellung des *Herbstes* habe ich mich für ein Blatt entschieden. Der Übergang von Grün nach Rot ist an die natürliche Verfärbung der Blätter angelehnt. Der ausfransende Rand des Blattes steht für den herbstlichen Prozess des Verfallens.

Zum *Winter* gehören für mich blaue, hellblaue und natürlich weisse Farben, welche man vor allem bei Schnee und Eis findet. Wenn das Eis zudem noch zersplittert ist, wirkt es für mich noch kälter. Deshalb habe ich für das Bild des Winters eine Form gewählt, die mich an zersplittertes Eis erinnert. Die Fragilität gehört für mich ebenfalls zum Winter, weil beispielsweise die Schönheit von Winterlandschaften wegen äusseren Einflüssen sehr schnell zerstört werden kann.

Eines meiner Ziele bei der Gestaltung des Bildes war auch, möglichst verschiedene durch iterierte Funktionensysteme definierte Fraktale zu kreieren. Diese sind zum Teil regelmässig, zum Teil auf den ersten Blick überhaupt nicht, einige haben ein klares Zentrum, andere wirken eher flächig, teilweise wirken sie stark verästelt und kompliziert, teilweise eher nicht, einige erscheinen natürlich, andere nicht usw. Diese ungeheure Vielfalt sowie die immer auftretende Selbstähnlichkeit faszinieren mich sehr.

Wer sich mit Fraktalen beschäftigt, entdeckt immer wieder etwas Neues!



A Anhang

A.1 IFS-Programm

Im Folgenden wird das IFS-Computerprogramm ([BAU/WEB]) näher vorgestellt. Dazu soll der Ablauf beim Entwerfen eines IFS geschildert werden:

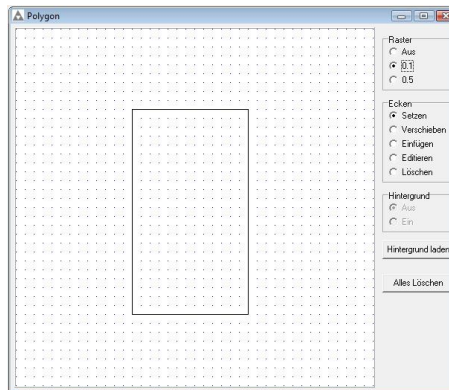


Abbildung A.1: Zuerst zeichnet man ein Polygon als Startmenge. Da diese für die Limesmenge unwesentlich ist, handelt es sich dabei in erster Linie um eine Hilfskonstruktion für das Wählen der Abbildungen.

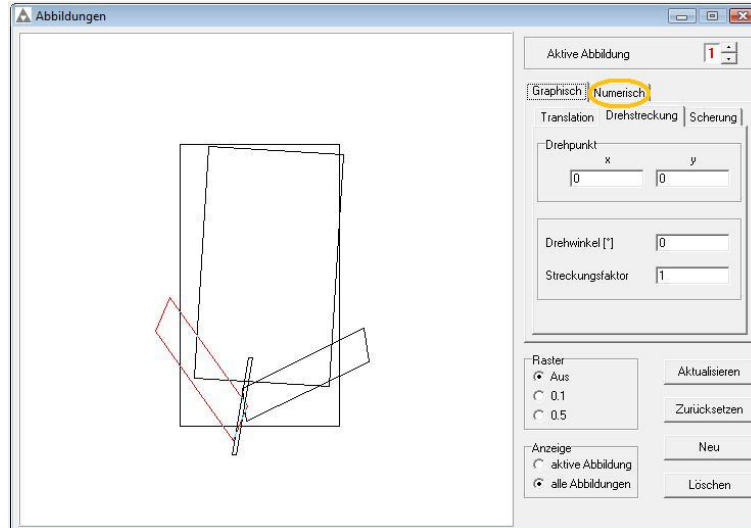


Abbildung A.2: Als Nächstes werden die Abbildungen bestimmt. Dank benutzerfreundlichen Werkzeugen für Translationen, Drehstreckungen und Scherungen ist dies praktisch und anschaulich. Ein weiterer grosser Vorteil des IFS-Programms ist, dass die Transformationen auch numerisch angegeben sind und man deshalb bei „numerisch“ (orange Ellipse) die jeweils 6 reellen Zahlen herauskopieren kann.

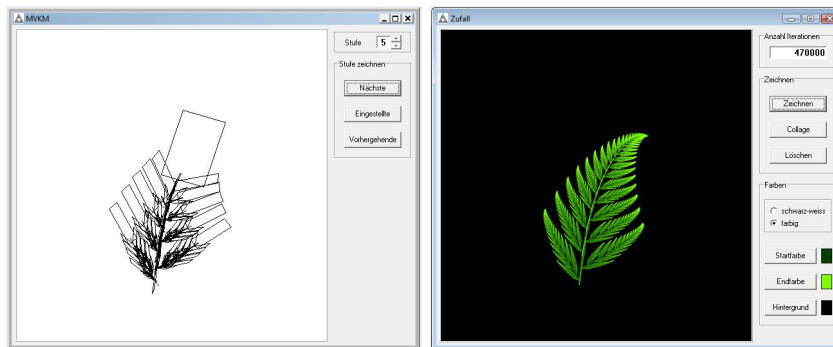


Abbildung A.3: Nachdem man die affin-linearen Transformationen gewählt hat, kann man sich einerseits eine gewisse Iterationsstufe des deterministisch iterierten Funktionensystems (links) und andererseits das mit dem zufällig iterierten Funktionensystem erzeugte Bild (rechts) anzeigen lassen. Letzteres ist hilfreich, wenn es darum geht, einen Eindruck von der Limesmenge zu erhalten. Wenn man sich jedoch die einzelnen Iterationsstufen anschaut, erkennt man besser, was im Verlauf der Iteration passiert und was man anders einstellen muss, um das Bild auf die gewünschte Weise zu verändern.

A.2 C++-Programm IFS

Das hier abgedruckte Programm enthält die Werte für das Bild zum Winter. Die Werte für die anderen Bilder sind auf der CD im C++-Programm „IFS.cpp“ zu finden.

```

1 #include <champ.h>
2 #include "ifs.rh"
3 #include <stdlib.h> // für random (Zufallszahlen)
4 #include <fstream.h> // für ofstream (Ausgabe in Textdatei)
5
6 void gmain ()
7 {
8     // Anzahl Iterationsschritte, d.h. Anzahl Punkte
9     const unsigned long ANZIT = 2000000;
10
11     // Transformationen: jeweils Wahrscheinlichkeit [%] und a, b, c, d, e, f
12     // für die N Kontraktionen
13     const int N = 3;
14     const double T[3][7] = {{25, 0.220809996372191, -0.769436320680939,
15                             0.251487467657931, 0.204661604335217,
16                             0.175407364358446, 0.832017626271714},
17                             {60, -0.349467704459924, 0.780885236582184,
18                             0.66423188802189, 0.0849105247907188,
19                             0.0686686975549493, -0.107045741512638},
20                             {15, -0.0288301927246291, -0.537487726282925,
21                             0.153894147951297, -0.0961505815638572,
22                             0.275454989314539, -0.470224199330359}};
23
24     // Koordinaten des Bildausschnitts
25     const float XMIN = -1.15;
26     const float XMAX = 1.25;
27     const float YMIN = -1;
28
29     // Auflösung
30     const int BREITE = 2000;
31     const int HOEHE = 2000;
32
33     const float YMAX = HOEHE / BREITE * ( XMAX - XMIN ) + YMIN;
34
35     // Seitenlänge eines Pixels

```

```

35     const double d = ( XMAX - XMIN ) / BREITE;
36
37     int i, j; // Zählvariablen
38     unsigned long ilong; // Zählvariable speziell für diejenige for-Schleife,
        die "ANZIT" Male durchlaufen wird
39     int zufallszahl; // Variable für den von der random-Funktion
        zurückgegebenen Wert
40     int nr; // Variable für die Nummer der durchzuführenden Abbildung
41     int xk, yk; // Variablen für x-, y-Wert des <<Kästchens>> im Array "felder"
42     double x, y; // Variablen für den jeweils aktuellen Punkt
43     double xneu, yneu; // Zwischenspeichervariable für neu berechnete Werte
44     int unten, oben; // Variablen, die im Rahmen der Auslosung von "nr"
        gebraucht werden
45
46     // Array für die Auszählung der Treffer pro Pixel
47     // [Zeile][Spalte], [0][0] ist oben links
48     int ** felder = new int * [HOEHE];
49     for( i = 0; i < HOEHE; i++ )
50     {
51         felder[i] = new int[BREITE];
52     }
53
54     // alles auf 0 stellen
55     for( i = 0; i < HOEHE; i++ )
56     {
57         for( j = 0; j < BREITE; j++ )
58         {
59             felder[i][j] = 0;
60         }
61     }
62
63     // ITERATION
64     // Startpunkt wird bestimmt (Fixpunkt der Abbildung 1)
65     double det = ( T[0][1] - 1 ) * ( T[0][4] - 1 ) - T[0][2] * T[0][3];
66     if( det != 0 ) // sollte eigentlich immer so sein...
67     {
68         x = ( - T[0][5] * ( T[0][4] - 1 ) + T[0][2] * T[0][6] ) / det;
69         y = ( - T[0][6] * ( T[0][1] - 1 ) + T[0][3] * T[0][5] ) / det;
70     }
71     else // zur Absicherung
72     {
73         x = 0.1;
74         y = 0.1;
75     }
76
77     randomize();
78     // eigentliche Iteration
79     for( ilong = 0; ilong < ANZIT; ilong++ )
80     {
81         zufallszahl = random( 100 );
82
83         // Auslosung der Abbildung
84         for( i = 0; i < N; i++ )
85         {
86             for( j = 0; j < i; j++ )
87             {
88                 unten += T[j][0];
89             }
90             for( j = 0; j < ( i + 1 ); j++ )
91             {
92                 oben += T[j][0];
93             }
94
95             if( zufallszahl >= unten && zufallszahl < oben )
96             {
97                 nr = i;
98             }
99
100            unten = 0;
101            oben = 0;
102        }
103
104        // Berechnung des neuen Punktes
105        xneu = T[nr][1] * x + T[nr][2] * y + T[nr][5];

```

```

106     yneu = T[nr][3] * x + T[nr][4] * y + T[nr][6];
107     x = xneu;
108     y = yneu;
109
110     // Umrechnung, in welchem <<Kästchen>> hochgezählt werden muss
111     xk = ( x - XMIN ) / d;
112     yk = ( YMAX - y ) / d;
113
114     // wenn dieses <<Kästchen>> existiert, wird sein Wert um 1 erhöht
115     if( 0 <= xk && xk < BREITE && 0 <= yk && yk < HOEHE )
116     {
117         felder[yk][xk]++;
118     }
119 }
120
121 // AUSGABE
122 ofstream ausgabe( "C:\\Users\\Muriel Blum\\Desktop\\PPM\\ausgabe.ppm" );
123 // Kopf des ppm-Files
124 ausgabe << "P3\n";
125 ausgabe << "# fraktal.ppm\n";
126 ausgabe << BREITE << " " << HOEHE << "\n";
127 ausgabe << "255\n";
128
129 for( i = 0; i < HOEHE; i++ )
130 {
131     for( j = 0; j < BREITE; j++ )
132     {
133         switch( felder[i][j] )
134         {
135             case 0: ausgabe << "0 0 0 "; break;
136             case 1: ausgabe << "80 80 140 "; break;
137             case 2: ausgabe << "80 97 153 "; break;
138             case 3: ausgabe << "80 115 165 "; break;
139             case 4: ausgabe << "80 134 178 "; break;
140             case 5: ausgabe << "80 155 191 "; break;
141             case 6: ausgabe << "80 177 204 "; break;
142             case 7: ausgabe << "80 196 213 "; break;
143             case 8: ausgabe << "90 209 214 "; break;
144             case 9: ausgabe << "100 222 213 "; break;
145             case 10: ausgabe << "120 245 212 "; break;
146             default: ausgabe << "150 250 230 ";
147         }
148     }
149     ausgabe << "\n";
150 }
151
152 CP::msgBoxD( "Fertig!" );
153 }

```

A.3 C++-Programm Farbverlauf

Hier ist das Programm für das Zusammenstellen von Farbverläufen mit Hilfe des HSV-Farbmodells abgedruckt, wobei die Einstellungen diejenigen für einen Farbverlauf von Orange nach Hellgelb sind. Das ähnlich aufgebaute Programm für das CIELAB-Farbmodell („LABZRGB2.cpp“) befindet sich auf der CD, ebenso die Werte für einige Farbverläufe.

```

1 #include <champ.h>
2 #include "hsvzrgb.rh"
3 #include <fstream.h>
4
5 void hsvzrgb( double h, double s, double v);
6 double h, s, v;
7 double f, p, q, t; // Variablen für die Umrechnung
8 int hi;
9 double R, G, B;
10
11 void gmain ()
12 {
13     ofstream ausgabe( "C:\\Users\\Muriel Blum\\Desktop\\PPM\\farbe.txt" );

```



```

14 ginit();
15
16 h = 20; // von 0 bis 360 [°]
17 s = 1; // von 0 bis 1
18 v = 0.5; // von 0 bis 1
19
20 float anz = 10; // Anzahl Abstufungen (wegen int-Division nicht int)
21
22 for ( int i = 0; i < anz; i++ )
23 {
24     // Verlauf:
25     h += 5;
26     s += 0;
27     v += 0.05;
28
29     // h (Farbton) zyklisch
30     if( h < 0 )
31     {
32         while( h < 0 )
33         {
34             h += 360;
35             CP::yield();
36         }
37     }
38     if( h > 360 )
39     {
40         while( h > 360 )
41         {
42             h -= 360;
43             CP::yield();
44         }
45     }
46
47     hsvzurgb( h, s, v ); // Umrechnung
48
49 // für die Betrachtung im Graphikfenster
50 gfillColor( RGB( R, G, B ) );
51 grectangle( 0, i / anz, 1, (i+1) / anz );
52
53 // für den für die Färbung benötigten Programmcode
54 ausgabe << "case " << ( i + 1 ) << ": ausgabe << \" " << (int)R << " " <<
    (int)G << " " << (int)B << " \"; break;\n";
55 }
56
57 CP::msgBoxD("OK?");
58 gend();
59 }
60
61 void hsvzurgb( double h, double s, double v ) // Umrechnung von HSV zu RGB
62 {
63     hi = floor( h / 60 ); // Grundfarbenintervall
64     f = h / 60 - hi; // Wert innerhalb des Intervalls
65     p = v * ( 1 - s );
66     q = v * ( 1 - s * f );
67     t = v * ( 1 - s * ( 1 - f ) );
68
69     switch( hi )
70     {
71     case 1: R = q; G = v; B = p; break;
72     case 2: R = p; G = v; B = t; break;
73     case 3: R = p; G = q; B = v; break;
74     case 4: R = t; G = p; B = v; break;
75     case 5: R = v; G = p; B = q; break;
76     case 6: case 0: R = v; G = t; B = p; break;
77     }
78
79     R = R * 255;
80     G = G * 255;
81     B = B * 255;
82 }

```

Literatur

- [ADO] Adobe Photoshop Elements (Version 3.0) [Computerprogramm]. Adobe Systems Incorporated: 2004.
- [BAU/WEB] Baumann, Kurt / Weber, Beat: Iterierte Funktionensysteme (Version 3.0) [Computerprogramm]. Gymnasium Köniz-Lerbermatt, CH-3098 Köniz: März 2005.
- [DUD] *Schülerduden Mathematik II*. 5., neu bearbeitete Auflage. Herausgegeben und bearbeitet von der Redaktion Schule und Lernen. Wissenschaftliche Bearbeitung: Prof. Dr. Harald Scheid und Dieter Kindinger. Mannheim·Leipzig·Wien·Zürich: Dudenverlag 2004.
- [EDG] Edgar, Gerald: *Measure, Topology, and Fractal Geometry*. New York: Springer 1990.
- [FAL] Falconer, Kenneth J.: *Fraktale Geometrie. Mathematische Grundlagen und Anwendungen*. Aus dem Englischen von Jens Meyer. Heidelberg·Berlin·Oxford: Spektrum Akademischer Verlag 1993 [Orig.: *Fractal Geometry. Mathematical Foundations and Applications*. 1990].
- [GRA] *Farbraum - Grafik-Fachbegriffe Online Lexikon*
<http://www.at-mix.de/farbraum.htm>
[Stand: 29.10.09]
- [HEU1] Heuser, Harro: *Lehrbuch der Analysis. Teil 1*. Stuttgart: B. G. Teubner 1980.
- [HEU2] Heuser, Harro: *Lehrbuch der Analysis. Teil 2*. Stuttgart: B. G. Teubner 1981.
- [PEI/JÜR/SAU] Peitgen, Heinz-Otto / Jürgens, Hartmut / Saupe, Dietmar: *Bausteine des Chaos. Fraktale*. Deutsch von Ernst F. Gucker in Zusammenarbeit mit Thomas Eberhardt. Reinbek bei Hamburg: Rowohlt 1998 [Orig.: *Fractals for the Classroom Part 1*. 1992].
- [ROH] Rohner, Heiner (2004): „Dimension Formulae and Open Set Condition for Self-Similar Fractals“. Diploma Thesis, philosophisch-naturwissenschaftliche Fakultät der Universität Bern.
- [ROS] Rosenman, Richard: Portable PixMap Importer / Exporter (Version 2.2) [Plugin für Photoshop]. Richard Rosenman Advertising & Design: 2008 (Release date: 11/03/03, Updated 09/28/08).
http://itunes.pluginworld.com/plugins_362/adobe/photoshop/richard_rosenman/portable_pixmap_importer_exporter.html
[Stand: Juni 2009].

- [WIK1] *Iterated function system - Wikipedia, the free encyclopedia*
http://en.wikipedia.org/wiki/Iterated_function_system
 [Stand: 24.10.09]
- [WIK2] *Satz von Heine-Borel - Wikipedia*
http://de.wikipedia.org/wiki/Satz_von_Heine-Borel
 [Stand: 29.10.09]
- [WIK3] *Farbraum - Wikipedia*
<http://de.wikipedia.org/wiki/Farbraum>
 [Stand: 25.09.09]
- [WIK4] *CIE-Normvalenzsystem - Wikipedia*
<http://de.wikipedia.org/wiki/XYZ-Farbraum>
 [Stand: 25.09.09]
- [WIK5] *Lab color space - Wikipedia, the free encyclopedia*
http://en.wikipedia.org/wiki/Lab_color_space
 [Stand: 25.09.09]
- [WIK6] *Lab-Farbraum - Wikipedia*
<http://de.wikipedia.org/wiki/Lab-Farbraum>
 [Stand: 25.09.09]
- [WIK7] *RGB color model - Wikipedia, the free encyclopedia*
http://en.wikipedia.org/wiki/RGB_color_model
 [Stand: 25.09.09]
- [WIK8] *RGB-Farbraum - Wikipedia*
<http://de.wikipedia.org/wiki/RGB-Farbraum>
 [Stand: 25.09.09]
- [WIK9] *HSV-Farbraum - Wikipedia*
<http://de.wikipedia.org/wiki/HSV-Farbraum>
 [Stand: 25.09.09]
- [ZEI/PAG] Zeitler, Herbert / Pagon Dušan: *Fraktale Geometrie. Eine Einführung. Für Studienanfänger, Studierende des Lehramtes, Lehrer und Schüler.* Braunschweig·Wiesbaden: Vieweg 2000.

Bildnachweis

- [1] <http://cordelia.mclean.org/~lowen/cantor.gif>
 [Stand: 15.10.09]
- [2] <http://www.mathaware.org/mam/00/master/essays/B3D/2/JPG/figure22.jpg>
 [Stand: 15.10.09]
- [3] http://upload.wikimedia.org/wikipedia/commons/9/98/Gamut_ISOcoated_and_sRGB.png
 [Stand: 28.09.09]

- [4] http://www.astronomie.at/Workshop/2005/CIExy1931_sRGB.png
[Stand: 01.10.09]
- [5] <http://www.webmasterpro.de/design/article/farblehrender-lab-farbraum.html/image/lab.jpg>
[Stand: 25.09.09]
- [6] <http://commons.wikimedia.org/w/thumb.php?f=Synthese%2B.svg&width=2000px>
[Stand: 28.09.09]
- [7] <http://commons.wikimedia.org/w/thumb.php?f=Synthese.svg&width=2000px>
[Stand: 28.09.09]
- [8] http://upload.wikimedia.org/wikipedia/commons/0/03/RGB_farbwuerfel.jpg
[Stand: 24.09.09]
- [9] http://dic.academic.ru/pictures/dewiki/72/HSV_cylinder.jpg
[Stand: 01.10.09]

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Das Informationsblatt „Plagiats-erkennung“ ist mir bekannt und somit auch die Konsequenzen eines Teil- oder Vollplagiates.

Muriel Blum
Bern, November 2009