

Analyse von Anwendungsbereichen und Umsetzungswegen generativer Gestaltung. Durchführung visueller Produktionen mittels Houdini.

vorgelegt von **Waldemar Schmidt** (33774)

an der Hochschule der Medien Stuttgart

am 10.09.2018

zur Erlangung des akademischen Grades eines Master of Engineering

Erst-Prüferin: Prof. Katja Schmid (Hochschule der Medien Stuttgart)

Zweit-Prüferin: Prof. Dipl.-Designerin Sabine Hirtes (Hochschule Offenburg)

I. Eidesstattliche Versicherung

Hiermit versichere ich, **Waldemar Schmidt**, ehrenwörtlich, dass ich die vorliegende Masterarbeit mit dem Titel: **„Analyse von Anwendungsbereichen und Umsetzungswegen generativer Gestaltung. Durchführung visueller Produktionen mittels Houdini.“** selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Hilfsmittel benutzt habe. Die Stellen der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen wurden, sind in jedem Fall unter Angabe der Quelle kenntlich gemacht. Die Arbeit ist noch nicht veröffentlicht oder in anderer Form als Prüfungsleistung vorgelegt worden.

Ich habe die Bedeutung der ehrenwörtlichen Versicherung und die prüfungsrechtlichen Folgen (§ 23 Abs. 2 Master-SPO (3 Semester) der HdM) einer unrichtigen oder unvollständigen ehrenwörtlichen Versicherung zur Kenntnis genommen.

Freiburg, 10.09.2018, Waldemar Schmidt

II. Abstract

Die vorliegende Masterarbeit gibt einen Einblick in die Varianz generativer Gestaltung und analysiert dabei einige Branchen, in welchen diese Anklang findet. Hierbei wird deutlich, in welchem großen Spektrum generative Methoden angewandt werden. Darüber hinaus werden verschiedene Umsetzungswege mittels Houdini aufgezeigt. Mögliche Generierungen von Formen und Bewegungen werden dabei durch prozedurale Teilschritte erstellt. Die vorgestellten Umsetzungswege werden durch eigene Kreationen und Weiterentwicklungen auf ihre Durchführbarkeit hin überprüft.

The present Master's thesis provides an insight into the variance of generative design and analyses the branches where it catches on. Thereby it becomes clear in which huge spectrum generative methods are exercised. Furthermore several implementation methods are pointed out through Houdini. Feasible generations of forms and movements are created with procedural steps. The presented implementation methods are verified by own creations and further developments.

III. Inhaltsverzeichnis

1. Einführung.....	5
1.1 Zur Relevanz der Thematik.....	5
1.2 Was ist generative Gestaltung?	8
1.3 Begriffsdefinitionen	10
 2. Anwendungsbereiche und	
 Beispiele generativer Gestaltung.....	12
2.1 Natur / Fiktiv.....	12
2.2 Digitale Visualisierung.....	14
2.3 (Innen- /) Architektur	17
2.4 Kunst.....	18
2.5 Technik / Industrie	20
2.5.1 Fahrzeuge.....	20
2.5.2 Orthopädie.....	21
2.6 Mode.....	22
2.7 Exkurs: 3D-Druck	24
2.8 Zusammenführung.....	25
 3. Software - SideFXs Houdini.....	26
 4. Umsetzungswege.....	29
4.1 Lindenmayer System / Baumgenerierung.....	29
4.1.1 Vorüberlegungen	29
4.1.2 Wissenschaftliche Einbettung	30
4.1.3 Umsetzungsweg in Houdini.....	32
4.1.4 Resultat	34
4.2 Space Colonization Algorithmus / Baumgenerierung.....	36
4.2.1 Vorüberlegungen.....	36
4.2.2 Wissenschaftliche Einbettung.....	36
4.2.3 Umsetzung in Houdini.....	38
4.2.4 Resultat.....	43
4.3 Bewegung generieren / Spirograph	44
4.3.1 Vorüberlegungen.....	44
4.3.2 Wissenschaftliche Einbettung.....	44
4.3.3 Umsetzungsweg in Houdini	47
4.3.4 Resultat.....	49
4.4 Zusammenführung.....	51

5. Durchführung	52
5.1 Lindenmayer System	52
5.1.1 Städte.....	52
5.1.2 Organische Formen.....	53
5.1.3 Variation	55
5.2 Wachstum mit dem Space Colonization Algorithmus	57
5.2.1 Anatomie	57
5.2.2 Ranken.....	57
5.2.3 Variation.....	60
5.3 Generierte Bewegungen und Formen	62
5.3.1 Weiterentwicklung.....	62
5.3.2 Variation.....	66
5.4 Zusammenführung.....	67
6. Fazit	68
 IV. Literaturverzeichnis	 71
 V. Abbildungsverzeichnis.....	 75
V.I Einführung.....	75
V.II Anwendungsbereiche.....	75
V.III Umsetzungswege.....	76
V.IV Durchführung.....	77
 VI. Anhang.....	 80
VI.I Verwendet Software	80
VI.II Digitaler Anhang (DVD beigelegt)	80

1. Einführung

1.1 Zur Relevanz der Thematik

In der Kreativwirtschaft ist die Generierung visueller Darstellungen mittlerweile in vielen Produktionen für verschiedenste Einsätze zu finden und sehr angesagt. Die Einsatzmöglichkeiten umfassen dabei Werbung (Bewegtbild, Print, Web) wie auch Film (VFX, Simulation, Set Extension). Generative Gestaltungselemente lassen sich für die Erstellung unterschiedlichster Objekte, Oberflächenstrukturen und sogar ganzer fiktiver Welten nutzen. Somit ist es nicht verwunderlich, dass generative Gestaltung einen großen Aufschwung im Bereich der Medienbranche erlebt.

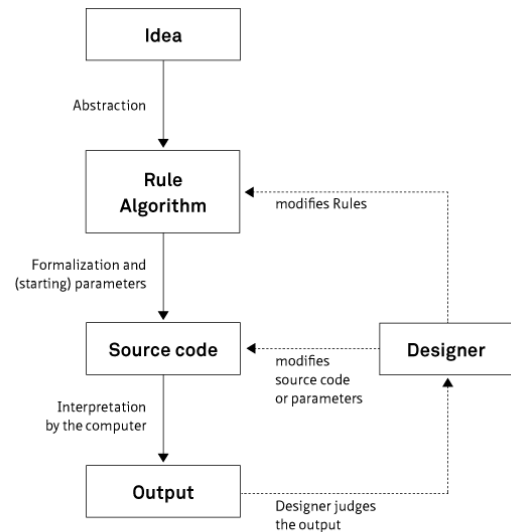
Die Einsatzmöglichkeiten generativ erstellter Objekte beschränkt sich aber nicht nur auf die Entertainmentbranche. Auch andere Disziplinen bedienen sich an Methoden der Generierung diverser Formgebungen (es sei bspw. an organische Formen gedacht). Der Schwerpunkt liegt hierbei hauptsächlich darin, die Nutzbarkeit und das Design verschiedener Produkte und Techniken zu verbessern. Beispielhaft seien hier zunächst nur einige Branchen erwähnt:

- ▢ Industrie
- ▢ Innen-/Architektur
- ▢ Mode
- ▢ Medizintechnik

Auch die Ausgestaltungsmöglichkeiten generativer Erzeugnisse wie bspw. Texturen und Formen sind vielfältig. So können diese rein fiktiv sein, um so etwas noch nie Dagewesenes, Einzigartiges bzw. Neues zu erstellen. Daneben kann aber auch die Realität widerspiegelt und ggf. weiterentwickelt werden. Dabei finden wissenschaftliche Erkenntnisse aus Natur und Physik Anwendung. Im gesamten Bereich generativer Gestaltung spielen mathematische Formeln und Regeln eine große Rolle. Generative Gestaltungsmethoden bauen häufig auf wissenschaftlich begründeten Vorüberlegungen auf und setzen sich aus einzelnen, logisch aufeinander aufbauenden Teilschritten zusammen. Dies ermöglicht den Aufbau einer Form oder einer Grafik nachvollziehbar darzustellen.

Einen wichtigen Aspekt generativer Gestaltung stellt die Nachbildung natürlicher Formen dar. Die Natur entsteht allerdings nicht willkürlich, sondern wird von äußeren Faktoren beeinflusst. Nimmt man etwa das Wachstum von Bäumen zum Beispiel, so wird klar, dass diese nicht einfach so wachsen, sondern dass ihre Form und Verzweigung von vielen Faktoren, wie bspw. Feuchtigkeit, Licht, umgebende Objekte etc. abhängig ist. Diese Faktoren haben direkten Einfluss auf das Wachstum und müssen damit auch bei der Generierung naturinspirierter Formen berücksichtigt werden.

Diese Überlegungen lassen sich auch auf die Erstellung fiktiver Formen und Strukturen übertragen. Bestimmte Faktoren können als einflussgebend bestimmt werden, die darüber hinausgehende Ausgestaltung wird dagegen dem Zufall überlassen. Somit birgt generative Gestaltung die Möglichkeit, das Aussehen eines Objektes oder eines Musters zwar in bestimmte Bahnen zu lenken, das tatsächliche Ergebnis kann aber in einer Vielzahl von Variationen erstellt werden. (vgl. Abb. 1)



Copyright Hartmut Bohnacker, Julia Laub, Benedikt Groß, Claudius Lazzeroni (2009)
Book „Generative Gestaltung“, www.generative-gestaltung.de

Generative Gestaltung lässt sich nun wie folgt genauer beschreiben:

Abb. 1: Workflow generativer Gestaltung

„Ein Bild wird [...] nicht mehr »von Hand« geschaffen, sondern dadurch, dass eine visuelle Idee in ein Regelwerk übersetzt und dann in einer Programmiersprache in Form eines Quellcodes umgesetzt wird. Die Folge ist, dass ein solches Programm nicht nur ein einzelnes Bild erzeugen kann, sondern dass durch Veränderung von Parametern ganze Bilderwelten entstehen.“¹

Dieses Zitat veranschaulicht die Grundidee generativer Gestaltung, nämlich die Möglichkeit, dass durch Erstellung eines bestimmten Regelwerkes und dessen Umsetzung unendliche Variationen ein und derselben Vorüberlegung möglich werden. Und dies nicht in mühsamer *Handarbeit*, sondern beinahe per *Knopfdruck*. Und diesen *Knopfdruck* soll die vorliegende Arbeit näher beleuchten, d.h. wie genau ist generative Gestaltung in der Praxis umsetzbar.

Zuvor sind jedoch noch einige theoretische Vorüberlegungen notwendig. Zunächst folgt eine genauere Definition generativer Gestaltung (vgl. Kapitel 1.2) und die Klärung einiger Begriffsdefinitionen (vgl. Kapitel 1.3). Daraufhin wendet sich die Arbeit möglichen Anwendungsbereichen zu (vgl. Kapitel 2). Bei genauerer Auseinandersetzung mit der Thematik wird deutlich, welches Potenzial generative Gestaltung für unzählige Disziplinen, weit über die Medienbranche hinaus, birgt. So finden sich in der Kunst, der Architektur, der Medizintechnik etc. eine ganze Bandbreite an Einsatzmöglichkeiten. Aber auch die Verwendungszwecke innerhalb der Medienschaffung sind kaum zählbar. Durch die Veranschaulichung einiger dieser Einsatzbereiche wird das ganze Spektrum generativer Gestaltung in verschiedensten wissenschaftlichen und künstlerischen Disziplinen deutlich.

Um generative Gestaltung nutzbar zu machen, bedarf es geeigneter Software. Die vorliegende Arbeit legt den Schwerpunkt auf Houdini. Daneben gibt es aber auch noch andere Programme, welche die Umsetzung generativer Gestaltung möglich machen. Kapitel 3 soll

1 [Über generative Gestaltung] o.J.

dazu dienen, diese Anwendungen kurz zu erwähnen. Außerdem wird auf die Grundideen, welche Houdini beinhaltet, genauer eingegangen.

Nachdem nun die Anwendungsbereiche und die zur Verfügung stehende Software betrachtet wurden, widmet sich Kapitel 4 den Umsetzungswegen generativer Gestaltung. Hierbei wurden einige Umsetzungswege (L-System, Space Colonization, generierte Bewegung) ausgewählt, welche auf Grundlage ihres wissenschaftlichen Aufbaus in ihrer Funktionsweise dargestellt werden. Ihre Anwendung in Houdini wird Schritt für Schritt an einem Beispiel erklärt, was für eine praktische Verortung der Thematik sorgt.

Zuletzt werden mögliche Anwendungsbereiche, die Software Houdini und die dargestellten Umsetzungswege in einer Durchführung (Kapitel 5) zusammengeführt. Hierbei werden die vorangegangenen Kapitel weitergedacht und in einen praktischen Zusammenhang gestellt.

Auf Grundlage all dieser Überlegungen analysiert die vorliegende Arbeit die Anwendungsbereiche und Umsetzungswege generativer Gestaltung und führt diese in visuellen Produktionen mittels Houdini aus.

1.2 Was ist generative Gestaltung?

„Generative design is a design methodology that differs from other design approaches insofar that during the design process the designer does not interact with materials and products in a direct („hands-on“) way but via a generative system.“²

Generative Gestaltung stellt damit wie eingangs (vgl. Kapitel 1.1) bereits erwähnt einen Prozess dar, bei welchem eine Idee mittels eines bestimmten Regelwerkes umgesetzt wird, ohne dass der Gestalter / die Gestalterin direkt in den Entstehungsprozess eingreift. Die Generierung eines Werkes geschieht durch die Verwendung von Zwischenschritten, welche in den Gesamtprozess eingebaut werden. Für die vorliegende Arbeit ist die generative Gestaltung damit immer als prozeduraler Workflow zu verstehen. Die darin enthaltenen Zwischenschritte beinhalten Regelwerke in Form von Algorithmen diverser Funktionen und sind ausschlaggebend für das Resultat. Der Gestalter / die Gestalterin kann oder könnte jeden einzelnen Prozess der Generierung beeinflussen, indem das Regelwerk und der Code so verändert werden, bis das Endprodukt der Erwartung entspricht. Er oder sie kann sich aber auch dazu entscheiden, bestimmte Prozesse dem Zufall zu überlassen.

„Generative Design beschreibt einen Gestaltungsprozess, bei dem das Ergebnis nicht mehr direkt durch den Designer erzeugt wird, sondern durch einen programmierten Algorithmus. Generative Design verändert damit nicht nur den Prozess des Gestaltens, sondern auch die Rolle des Designers fundamental.“³

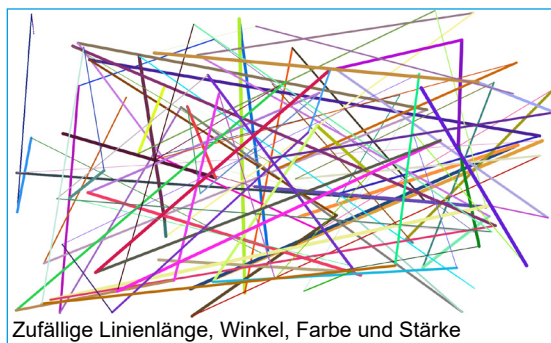


Abb. 2: Erstellung interaktiv generativer Bilder 1



Abb. 3: Erstellung interaktiv generativer Bilder 2

Generative Gestaltung muss sich dabei nicht zwingend auf eine digitale Umsetzung beschränken, sondern kann auch durch analoge Systeme umgesetzt werden.⁴ Eine der einfachsten Methoden zur analogen Generierung visueller Eindrücke ist der Einsatz von Stift und Papier. Solange bestimmte Abfolgen und Regeln beachtet werden, wird ein generatives Bild erstellt. Durch das Ändern der Abfolgen und Regeln wie bspw. der Strichlänge, der Form, der Farbe, des Abstands, des Drucks etc. entsteht ein völlig neues Bild, das Resultat ist immer ein anderes. Im Rahmen dieser Arbeit liegt das Hauptinteresse allerdings auf

² Fischer / Herr 2002.

³ Dertinger 2015.

⁴ vgl. Fischer / Herr 2002.

digitalen Produktionen. So lässt sich auch das Beispiel mit Stift und Papier auf computer-gestützte Generierung beziehen. Dies könnte in diesem Fall so aussehen, dass mit farbigen Pixelstrichen und farbigem Hintergrund gearbeitet wird. Dabei kann definiert werden, dass zu einer bestimmten Zeit oder bei einer bestimmten Aktion etwas Vordefiniertes passieren soll. Der Prozess kann aber auch rein zufällig ablaufen, indem Zufallszahlen im Bezug zu Strichstärke, Farbe, Winkel etc. festgelegt werden. (vgl. Abb. 2 und 3)

1.3 Begriffsdefinitionen

Generieren / generativ

Generieren wird laut Duden als *hervorbringen* und *erzeugen* definiert. Etwas zu generieren bedeutet, etwas zu produzieren oder dafür zu sorgen, dass etwas überhaupt erst entsteht oder auftritt.⁵ Als generativ wird somit etwas bezeichnet, dass in der Lage ist, etwas zu produzieren oder zu erstellen.⁶

Generieren bzw. generativ wird in dem hier dargestellten Zusammenhang als Umschreibung für einen während der Entstehung nicht manuell beeinflussten Prozess verwendet. D.h. der Weg bis hin zum fertigen Objekt wird zwar mittels eines Regelwerkes in gewissen Bahnen gelenkt, der Prozess an sich läuft aber ohne äußeren Einfluss nur anhand dieser Regeln ab, er ist damit generativ.

Gestalten

Laut Duden handelt es sich beim Gestalten darum, einer Sache eine bestimmte Form, ein bestimmtes Aussehen zu geben. Darüber hinaus lässt sich unter Gestaltung die Art und Weise verstehen, welche Eigenschaften ein Erzeugnis aufweist. Allerdings beschränkt sich das Gestalten nicht alleine auf das Aussehen, sondern auch auf den Entstehungsweg des jeweiligen Objektes bzw. der jeweiligen Struktur.

Fraktale

Fraktale sind Objekte, deren Spektrum und Vielfalt den Formen der Natur sehr nahe kommen. Sie haben Strukturen der Selbstähnlichkeit, was bedeutet, dass die Teilstrukturen des Fraktals meist die selbe Form aufweisen, wie das gesamte Objekt. Eine weitere Eigenschaft von Fraktalen ist ihre enorme Komprimierbarkeit, welche sich dadurch ergibt, dass nicht das fertige Objekt an sich gespeichert wird, sondern lediglich der zugrunde liegende Algorithmus bzw. die Anleitung zur Erstellung des Objektes. Diese einfachere Weiterverwertung wiederum ergibt sich aus der rekursiven und / oder iterativen Eigenart fraktaler Algorithmen.⁷

Fraktale können sich damit in unendlich vielen denkbaren Formen vervielfältigen und entwickeln. Durch die Möglichkeit, sie zu unzähligen Objekten *wachsen* zu lassen, sie aber auch wieder komprimieren zu können, sind sie der Grundstein für verschiedenste Umsetzungswege generativer Gestaltung.

⁵ vgl. Hornby 1995a, S. 470.

⁶ vgl. Hornby 1995b, S. 470.

⁷ vgl. Flake 1998, S. 59.

Design

Unter der deutschen Verwendung des Begriffes *Design* wird häufig die Ästhetik eines Produktes verstanden. Etwas hat ein bestimmtes Design, wenn es besonders gut oder speziell aussieht. Aus diesem Grund wird in der vorliegenden Arbeit auf die Verwendung des Begriffes *Design* so gut es geht verzichtet und stattdessen von *Gestaltung* gesprochen. *Gestaltung* bezieht sich nicht nur auf das reine Aussehen eines Objektes oder Musters, sondern hat auch dessen Nutzbarkeit, Funktionalität und Struktur im Blick. Da generative Gestaltung nicht unbedingt *schöne* Erzeugnisse produzieren soll, sondern es hauptsächlich um deren Entstehungsprozess und deren Umsetzbarkeit geht und das Aussehen je nach Anwendungsbereich und Verwendungszweck gestaltet werden soll, scheint diese Abgrenzung hier notwendig.

Prozedurale Programmierung

Von prozeduraler Programmierung spricht man dann, wenn die von einer Software zu lösende Gesamtaufgabe in kleinere Teilaufgaben zerlegt wird. Der Vorteil liegt darin, dass die einzelnen Teilaufgaben meist einfacher zu beschreiben, zu programmieren und zu testen sind. Hinzu kommt, dass die dabei entstehenden Programmcodes in anderen Programmen erneut genutzt werden können.⁸

Computergrafik

„[...] Verfahren der Informationstechnik zur Erfassung, Erstellung, Modellierung, Verarbeitung, Speicherung, Darstellung und Ausgabe von grafischen Daten.“⁹

Motion Design / Graphics

Unter Motion Design wird eine Synthese unterschiedlicher Entwicklungen im Bereich des Kommunikationsdesigns (filmisch bzw. künstlerisch wie auch grafisch und audiovisuell) verstanden. Motion Designer verfolgen dabei das Ziel, zu unterhalten, zu überraschen und zu begeistern.¹⁰

⁸ vgl. DATACOM Buchverlag GmbH 2011.

⁹ DATACOM Buchverlag GmbH 2018.

¹⁰ vgl. Jung 2018.

2. Anwendungsbereiche und Beispiele generativer Gestaltung

2.1 Natur/Fiktiv

Die Natur ist die Ausgangsbasis und das Vorbild für die Gestaltung einer Vielzahl von durch Menschen erschaffener Werke. So wird beispielsweise die Malerei seit Jahrtausenden von Landschaften und Pflanzen geprägt und inspiriert. Des Weiteren werden in zahllosen Bereichen die Bauarten der Natur kopiert und so umgestaltet, dass der Mensch einen Nutzen daraus ziehen kann. Diese Umsetzung natürlicher Systeme, Prozesse und Strukturen, auch Bionik bezeichnet, kann auch als das „*Lernen von der Natur als Anregung für eigenständig-technisches Gestalten*“¹¹ verstanden werden. So basiert die Formgebung von Fahrzeugen unter anderem auf der Form von schnell schwimmenden oder fliegenden Tieren.¹² Auch in der Architektur, der Textilproduktion, der Medizintechnik und vielen weiteren Disziplinen dient die Natur als Inspiration.

Eine besondere Rolle kommt in diesem Zusammenhang der goldenen Zahl zu.

Die Zahl **Phi (Φ)**, welche mit der Zahl 1,618033 beginnt, definiert den goldenen Schnitt (vgl. Abb. 4).

Phi wird in der Wissenschaft schon seit Jahrhunderten zur Erklärung verschiedenster Naturprozesse und natürlicher Strukturen herangezogen und wird auch als logische Ordnung der sonst so chaotisch wirkenden Welt verstanden. Diese Zahl ist einzigartig und hat viele Besonderheiten. In der Natur kommt ihr eine bedeutende Rolle zu. Zum Beispiel sind Blumen, wie die Akeleiblüte, die Glockenblume und die Heckenrose nach dem Prinzip des goldenen Schnitts aufgebaut. Die Blüten bilden ein perfektes Fünfeck, in der Mitte der Blüte ist der Stamm und nach außen hin führen fünf Blätter.¹³

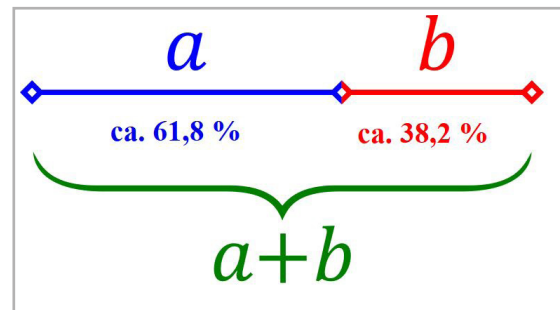


Abb. 4: Goldener Schnitt

Aber auch in der Medienbranche wird Phi als Gestaltungsregel für die Bildbearbeitung und die Layoutgestaltung unter Anwendung des goldenen Schnitts und der goldenen Spirale verwendet. Daneben eignet sie sich zur Erstellung diverser Pflanzen und Blumen in der dreidimensionalen Welt. Somit wird die goldene Zahl für die Gestaltung und Erschaffung natürlicher sowie naturempfundener Eindrücke genutzt. Daneben können auch Menschen

¹¹ Nachtigall 1997, S.1.

¹² vgl. ebd., S. 2.

¹³ vgl. Junghans o.J.

mithilfe der goldenen Proportion im dreidimensionalen Raum erstellt werden. Das Verhältnis menschlicher Glieder weist einen bestimmten Bezug zueinander auf. So ist es möglich, lediglich durch die Angabe einer Größe und unter Zuhilfenahme eines Algorithmus sowie eines bestimmten Regelwerks einen digitalen Menschen zu generieren.

Die Natur ist kein eigenständiger Anwendungsbereich generativer Gestaltung im eigentlichen Sinne, sondern bildet vielmehr die Grundlage bzw. Ausgangsbasis für die generative Gestaltung verschiedenster Objekte. Zum einen können natürliche Formen wie bspw. ein Blatt oder eine Schneeflocke mithilfe verschiedener Algorithmen generiert werden. Zum anderen werden natürliche Strukturen und Formen zur Generierung von Materialien, Formen, Strukturen etc. in unterschiedlichsten Anwendungsbereichen genutzt. Da die Natur eine Vielzahl an Möglichkeiten für die generative Gestaltung bereithält, wurde ihr an dieser Stelle ein eigenes Unterkapitel gewidmet, auch wenn es sich, wie bereits erwähnt, nicht um einen eigenständigen Anwendungsbereich generativer Gestaltung handelt.

2.2 Digitale Visualisierung

In Werbespots und Filmproduktionen gewinnt die generative Gestaltung immer mehr an Bedeutung. Mithilfe physikalischer und anderer wissenschaftlicher Erkenntnisse werden natürlich wirkende Visualisierungen generiert. Diese haben zum Ziel, die dargestellten Objekte oder Welten möglichst real erscheinen zu lassen. So werden in Filmen und Spots ganze Ozeane, Wälder, Städte und sogar abstrakte Welten oder aber auch Innenansichten von Produkten und deren Fertigungswege generiert.

Dabei geht es meist darum, Dinge darzustellen, die durch Realfilmaufnahmen nicht möglich wären. Bezieht sich ein Film beispielsweise auf eine eigene Welt, die sich stark von den Landschaften der Erde unterscheidet, ist es kaum möglich, diese Welt durch gewöhnliche Filmaufnahmen abzubilden. Generative Gestaltung bietet dabei die Möglichkeit, diese Welten zwar an die Formen und Strukturen unserer Welt anzugleichen (durch die Generierung von Ebenen und Bergen, Wüsten und Wälder etc.), sie aber dennoch einzigartig und nicht-irdisch zu gestalten. Daneben finden auch Vorgänge und Effekte, die durch Realbildaufnahmen nicht abbildbar wären, ihre Umsetzung in der generativen Gestaltung. So werden Spezialeffekte und Materialien prozedural am Computer erstellt. All diese Möglichkeiten wären auch mit konventionellen Methoden durch die Erstellung jedes einzelnen Objektes, Effektes oder Musters möglich. Der Vorteil generativer Gestaltung liegt aber in der schnelleren Produktionsweise und den breiten Variationsmöglichkeiten, die sich vergleichsweise einfach erstellen lassen.

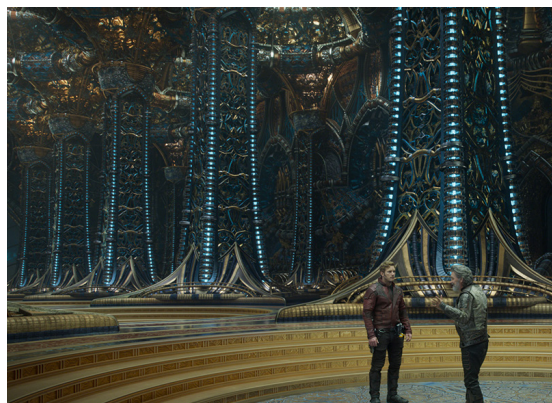


Abb. 5: Fraktale Welt von Guardians of the Galaxy Vol.2

In der Filmproduktion hält generative Gestaltung immer weiter Einzug. Wie bereits erwähnt können dabei unter anderem auch ganze Welten erschaffen werden. Ein Beispiel hierfür ist der Film „The Guardians of the Galaxy Vol. 2“. Bei diesem wurden Fraktale in Houdini erzeugt, welche dann in eine Art Gerüst traditioneller Architektur bzw. Struktur eingesetzt wurden, um Gebäude, Objekte, Pflanzen und sogar das Innenleben eines Menschen abzubilden. (vgl. Abb. 5)

Ein weiteres Beispiel für die Nutzung generativer Gestaltung im Bereich des Bewegtbildes sind Commercials von Bekleidung. Immer mehr dieser Werbespots verwenden generative Animationen, welche den



Abb. 6: Generativer Aufbau eines Schuhs

Aufbau, die Funktionalität oder das Design verschiedener Stoffe und Webtechniken veranschaulichen. (vgl. Abb. 6) Teilweise werden die analogen Fertigungsmethoden in die digitale Animation übernommen. Aber auch andere Methoden der Visualisierung sind denkbar. Beispielsweise kann das Wachstum eines Materialgeflechts, basierend auf natürlichen Beobachtungen und Erkenntnissen aus der Natur, dargestellt werden. Aber auch die Generierung aus künstlerisch-abstrakten und fiktiven Methoden kann zum Einsatz kommen. Dies vermittelt dem / der potenziellen Käufer / Käuferin den Eindruck eines gut durchdachten und individuell designten Produktes und veranschaulicht dessen Aufbau. Insbesondere im Bereich der Funktionskleidung (vgl. Kapitel 2.6) können die funktionalen Eigenschaften anschaulich dargestellt werden.

Die Möglichkeiten sind beinahe unbegrenzt und es ist davon auszugehen, dass wir in Zukunft immer mehr generierte Film- und Werbeproduktionen sehen werden. Diese bieten den Vorteil, bisher kaum oder nur sehr aufwendig umsetzbare Objekte digital darzustellen und damit dem Zuschauer / der Zuschauerin neue Einblicke in unbekannte Welten zu geben.

In der On- und Offlinewelt finden sich immer häufiger generierte Bilder und Elemente. Diese haben zum einen, wie bereits vorangegangen beschrieben, zum Ziel, Produkte für potenzielle Käufer / Käuferinnen attraktiver darzustellen und damit zum Kauf anzuregen. Daneben können sie aber auch zur vereinfachten Informations- und Datendarstellung dienen. So können komplexe Funktionsweisen und Fertigungsverfahren gestalterisch ansprechend und verständlicher dargestellt werden. Diese Funktionen können nicht nur zu Werbezwecken, sondern auch zur Wissensvermittlung genutzt werden. So sind zu verschiedenste Datensätze zurückgreifen und binden. (vgl. Abb. 7) Auch die vereinfachte Darstellung mittels generativer Gestaltung erreicht werden kann, um interaktiv anbieten. Der Nutzer / die Nutzerin steuert den Verlauf.

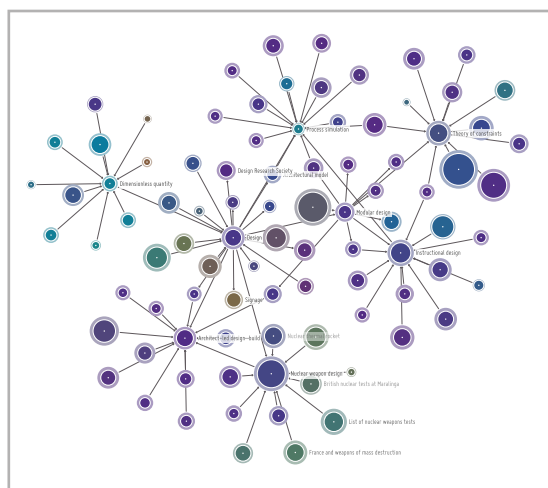


Abb. 7: Dynamische Datenstruktur

sensvermittlung genutzt werden. So sind zum Beispiel Infografiken denkbar, welche auf verschiedenste Datensätze zurückgreifen und diese mittels Algorithmen miteinander verbinden. (vgl. Abb. 7) Auch die vereinfachte Darstellung komplexer Zusammenhänge könnte mittels generativer Gestaltung erreicht werden. Gerade im Web lassen sich die Inhalte auch interaktiv anbieten. Der Nutzer / die Nutzerin hat dabei Einfluss auf die Darstellung oder auf den Verlauf.

Auch im Bereich des Game-Designs nimmt die Bedeutung generativer Gestaltung stetig zu. Als Beispiel hierfür kann das Spiel *No Man's Sky* von Hello Games erwähnt werden. Für dieses wurde eine komplette Galaxie prozedural generiert, welche mehr als 18 Trillionen Planeten enthält. Keiner dieser Planeten ist dabei gleich, so ist jede Kreatur, geologische Formation und Pflanze einzigartig.¹⁴

14 vgl. Hello Games No Man's Sky o.J.

Selbst im Audibereich macht die Generierung nicht halt. Generatives Audiodesign ist in den unterschiedlichsten Zusammenhängen einsetzbar und ermöglicht die Herstellung lebhafter Klangkulissen. Verwendung kann generatives Audiodesign bspw. in Museen und bei Musikveranstaltungen finden.

Klänge können als digitale Information in grafische Formen und Muster umgewandelt werden. Diese Visualisierung von Musik kann häufig in Mediaplayern gefunden werden. Dabei richtet sich die Formgebung meist nach der Lautstärke und der Frequenz der jeweiligen Klänge. Daneben können aber auch ganze Songs als Grafik dargestellt und damit die Komplexität von musikalischen Werken visualisiert werden.¹⁵

¹⁵ vgl. Kretzer 2016.

2.3 (Innen-) Architektur

Im Bereich der Architektur und des Städtebaus können mittels generativer Gestaltung verschiedenste Aufgaben angegangen werden. Angefangen bei der Generierung unterschiedlicher Raummuster, bis hin zur Planung ganzer Stadtviertel, können computerbasierte Systeme unterschiedliche Entwürfe generieren und damit langwierige manuelle Entwurfsphasen ersetzen.¹⁶ Dabei ist es möglich, unter Berücksichtigung bestimmter Einschränkungen und Vorgaben wie bspw. Tageslicht, Wärmeaustausch und Statik, zahlreiche Gestaltungsoptionen eines Gebäudes zu generieren.¹⁷ Gebäude und Stadtviertel lassen sich somit noch effektiver an ihre natürliche und bebaute Umgebung anpassen.

Es sind ganz neue Konzepte von Außenfassaden (vgl. Abb. 8) und Innenwänden denkbar, welche z.B. auch klimatische Aspekte weiter berücksichtigen könnten. Auch das generelle Design eines Gebäudes kann revolutioniert werden. Bisher kaum denkbare Formen können mittels generativer Gestaltung unter Beachtung statischer Parameter entstehen.

Neben der Architektur kommen Methoden der generativen Gestaltung auch im Bereich des Möbelbaus zum Einsatz. Für diverse Möbelstücke wie Stühle, Liegen und Lampenschirme können damit völlig neue Formen und Strukturen entwickelt werden. Daneben ist der Einsatz im allgemeinen Innenausbau, aber auch im Messe- und

Ladenbau denkbar. Der Vorteil generativer Gestaltung und Fertigung in diesen Bereichen liegt in der neu entstandenen Formfreiheit, welche mit konventionellen Mitteln kaum oder nur sehr umständlich planbar war. Spektakuläre und Aufsehen erregende Designs waren aufgrund ihrer komplexen Geometrie und der damit verbundenen viel zu aufwendigen Plandarstellung beinahe nicht umsetzbar. Prozedurale Generierung bringt für diese Problematiken neue Möglichkeiten insbesondere im Bereich der Formfindung und Konstruktion. Daraus kann schlussgefolgert werden, dass generative Methoden die gestalterischen und konstruktiven Grenzen der Architektur und des Produktdesigns zu sprengen vermögen.¹⁸



Abb. 8: Alibaba Hauptquartier

¹⁶ vgl. König 2012, S. 17f.

¹⁷ vgl. Alderton 2017.

¹⁸ vgl. Behaneck 2011.

2.4 Kunst

Immer mehr Künstler und Künstlerinnen bedienen sich an Methoden der generativen Kunst. Darunter wird eine Praxis verstanden, bei welcher der Künstler / die Künstlerin ein System nutzt, das er oder sie zwar in Bewegung setzt, dieses dann aber mit einem gewissen Grad an Autonomie ein Kunstwerk erschafft. Ein solches System kann unter anderem auch eine Maschine oder ein Computerprogramm sein.¹⁹

Die gemeinsame Vorgehensweise generativer Kunst und generativer Gestaltung ist in der Nutzung eines Systems bzw. Werkzeuges zu sehen. Dieses wird dazu verwendet, ein Ergebnis (ein Kunstwerk, ein Objekt, eine Animation etc.) zu erzeugen. Allerdings bleibt bei generativer Kunst der Prozess ergebnisoffen, d.h. die Umsetzung wird zwar durch ein bestimmtes Regelwerk vorgegeben, der Künstler / die Künstlerin beeinflusst das Ergebnis aber nicht durch aktives Eingreifen in den eigentlichen Entstehungsprozess des Werkes. Dagegen werden bei der generativen Gestaltung verschiedene Parameter und Regeln immer weiter angepasst, bis das gewünschte Ergebnis vorliegt.²⁰

In der generativen Kunst wird also im eigentlichen künstlerischen Prozess auf eine Steuerung verzichtet und die Erstellung eines Werkes wird einem mehr oder weniger autonomen System überlassen. Der Künstler / die Künstlerin kann den Prozess lediglich in eine bestimmte Richtung lenken. Änderungen der vorgegebenen Regeln erzeugen einen anderen Gestaltungsprozess und ein anderes Kunstwerk. Der künstlerische Erschaffungsprozess bewegt sich zwar innerhalb der vorgegebenen Grenzen und Regeln, das Ergebnis ist davon abgesehen aber nicht vorhersehbar.²¹ (vgl. Abb. 9 und Abb. 10)

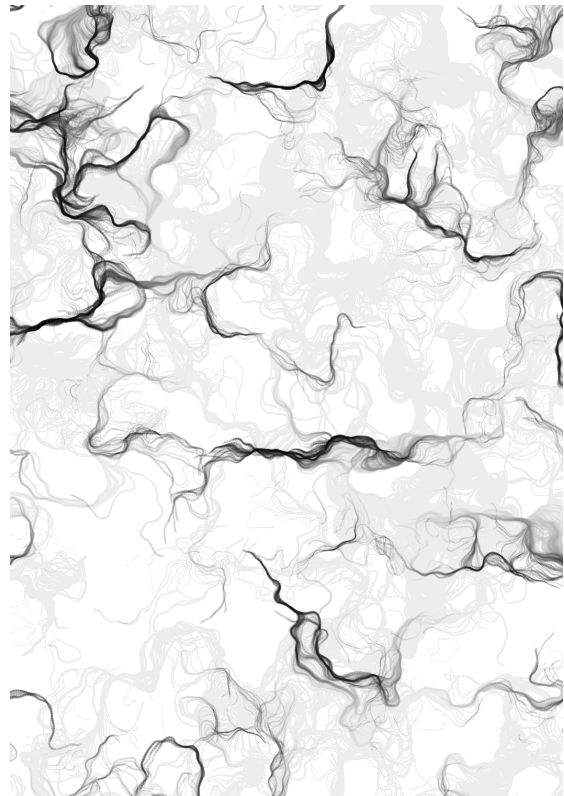


Abb. 9: Dreidimensionales Rauschen



Abb. 10: Kunst in Echtzeit

¹⁹ vgl. Galanter 2003, S. 228.

²⁰ vgl. Kizny o.J.

²¹ vgl. Generative Kunst o.J.

Kunst, welche sich generativer Methoden bedient, unterscheidet sich von klassischer Kunst dadurch, dass nicht das Werk im Mittelpunkt steht. Vielmehr liegt der künstlerische Aspekt in den Prozessen, die zur Generierung des Resultats geführt haben. Sie bestimmen das Ergebnis und bilden damit die eigentliche Kunst, sowohl für den Künstler / die Künstlerin als auch für die Betrachter / Betrachterinnen.

Zusammenfassend kann generative Kunst als Unterform generativer Gestaltung verstanden werden, da sie sich der gleichen Methoden bedienen, generative Kunst aber weiterhin ergebnisoffen bleibt und damit den künstlerischen Gedanken und nicht das Kunstwerk bzw. Ergebnis in den Mittelpunkt stellt. Die Methoden generativer Gestaltung lassen sich dabei für allerlei Kunstrichtungen sei es Malerei, Installationskunst, Bildhauerei und noch viele mehr einsetzen.

2.5 Technik/Industrie

2.5.1 Fahrzeuge

In der Fahrzeugindustrie und im Flugzeugbau wird schon länger im Bereich der Generierung von praktischen Formen geforscht. Auch hier wird häufig auf Ideen aus der Natur zurückgegriffen. Wie eingangs (Vgl. Kapitel 2.1) bereits erwähnt, werden Fahrzeuge in ihrer Form häufig besonders stromlinienförmigen Tieren angeglichen, um den Luftwiderstand zu minimieren und damit ein besseres Fahrerlebnis zu gewähren. Neben diesen eher aerodynamischen Überlegungen spielen auch Materialien und deren Form eine wichtige Rolle im Fahrzeugbau. So kann die generative Erstellung bestimmter Fahrzeugteile unter dem Aspekt der Materialreduzierung und unter Beachtung statischer Vorgaben Vorteile mit sich bringen. Ziel dabei ist, die verwendeten Rohstoffe im Vergleich zur herkömmlichen Bauweise auf einen Bruchteil zu reduzieren und damit das Gewicht deutlich zu senken, dabei aber die Stabilität und Funktionalität nicht einzuschränken.

Ein Beispiel hierzu ist die Forschung von Schäfer (eine Gruppe der Airbus-Gruppe „*Emerging Technologies and Concepts*“ in Deutschland) an einer bionischen Trennwand (vgl. Abb. 11) für einen Airbus A320. Die Trennwand wurde unter Zuhilfenahme verschiedener Ansätze aus der Natur generiert. Die verwendeten Algorithmen beziehen sich auf die Wachstumsmuster von Schleimpilzen und von Säugetierknochen. Schleimpilze können in ihrem Wachstum mehrere Punkte effizient miteinander verbinden, Säugetierknochen wachsen an stark belasteten Stellen dichter. Aus diesen Vorgaben wurde ein Wachstumsmuster generiert, welches sowohl Materialeffizienz und -einsparung, als auch Stabilität zum Ziel hat.²²

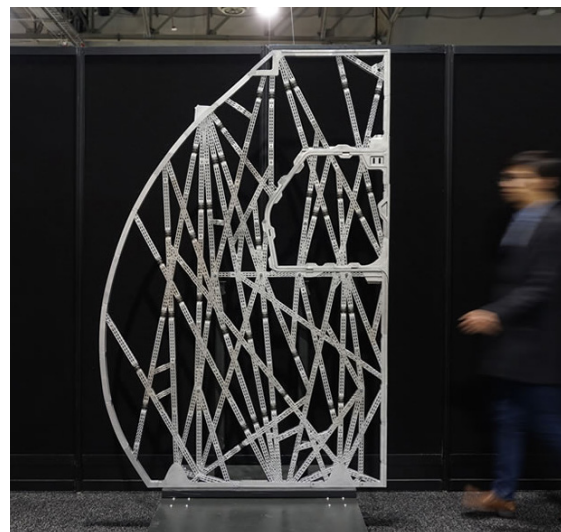


Abb. 11: Bionische Trennwand

Durch diese Generierung, den Simulationen und dem anschließenden 3D-Druck ergeben sich viele Vorteile. Unter anderem verringert sich das Gewicht der bionischen Trennwand im Gegensatz der konventionell gefertigten um 30 kg. Dadurch ergibt sich eine Ersparnis an Treibstoff von 3 180 kg pro Trennwand und Jahr, sowie 166 Tonnen CO-Emissionen weniger pro Jahr und Flugzeug. Zudem werden 95% weniger Rohstoffe zur Herstellung benötigt.²³

Auch im Bereich des Fahrzeugbaus sind noch viele weitere Möglichkeiten der Nutzung generativer Gestaltungsmethoden denkbar. Durch die fortschreitende Forschung und einem

²² vgl. Autodesk, Inc. o.J.

²³ vgl. ebd.

hohen Interesse an mehr Sicherheit, wie auch an der Gewinnmaximierung und Ressourceneinsparung, ist zu erwarten, dass zukünftig zahlreiche Weiterentwicklungen u.a. durch generative Gestaltung erzielt werden.

2.5.2 Orthopädie

Auch in der Orthopädie sind die Methoden der generativen Gestaltung auf dem Vormarsch. Ähnlich wie in der Fahrzeugindustrie kann mit diesen Material eingespart und Gewicht reduziert werden. Hier steht die Wahrung der Funktionalität und Qualität gleichermaßen im Mittelpunkt, wenn nicht sogar deren Revolutionierung. Denkbar sind bspw. mehr Elastizität, eine verbesserte Handhabung aber auch eine flexiblere Anpassung an die individuellen Bedürfnisse der Trägers / der Trägerin und eine schnellere Produktion.

Durch generative Gestaltungsprozesse können im Bereich der Orthopädie auf relativ einfachem Wege maßgeschneiderte Prothesen geschaffen werden. Außerdem ergibt sich der Vorteil, dass diese nicht durch wiederholtes Anprobieren immer weiter angepasst werden müssen, sondern dass die notwendigen Daten und Maße direkt am Computer in die Konstruktion der Prothese integriert werden. Daneben können komplexe Geometrien und innere Strukturen generiert werden, die bei manuellen bzw. konventionellen Entwicklungs- und Fertigungsverfahren kaum denkbar wären.²⁴

Daneben wird in der Orthopädie mittlerweile mehr Wert auf gestalterische Aspekte gelegt, um beispielsweise Prothesen unter Erfüllung der funktionalen Eigenschaften auch ästhetisch ansprechend zu gestalten. Generative Fertigungsverfahren helfen dabei, gestalterische Ansprüche bei der Prothesenfertigung mit zu berücksichtigen. Prothesen sind keine Modeaccessoires, sondern in aller Regel werden sie von der betreffenden Person benötigt, um ihrem alltäglichen Leben nachgehen zu können. Sie sollten sich daher nicht nur in ihrer Funktionsweise sondern auch in ihrer Gestaltung an den Lebensstil des Trägers / der Trägerin anpassen. Die Verbindung dieser beiden Attribute ist durch generative Technologien auf einfachere Weise realisierbar geworden.²⁵

Durch die Methoden generativer Gestaltung lassen sich Prothesen immer besser an die Bedürfnisse der Träger und Trägerinnen anpassen, Bewegungsabläufe werden verbessert und es ist eine Annäherung an den natürlichen menschlichen Bewegungsapparat erkennbar. Auch wenn diese Entwicklung auch ohne generative Prozesse stattfindet, so scheinen prozedurale Entwicklungs- und Fertigungsverfahren doch eine Möglichkeit zu bieten, schnell realisierbare und individuell anpassbare Hilfsmittel zu erstellen. Neben der Erstellung von Prothesen sind auch weitere Bereiche wie beispielsweise die Entwicklung von Orthesen oder gar Mensch-Roboter-Hybridsystemen in Form von Exoskeletten als Einsatzbereich generativer Gestaltung auf dem Vormarsch.

²⁴ vgl. Breuninger / Gottinger 2011.

²⁵ vgl. ebd.

2.6 Mode

Schon seit einiger Zeit werden viele Produktionsprozesse der Modebranche durch computergestützte Maschinen durchgeführt, der Prozess des Entwerfens von Mustern und Schnitten wird in den meisten Fällen aber noch immer händisch von Modedesignern und -designerinnen durchgeführt. Der Prozess des manuellen Erstellens von Schnitt- oder Stoffmustern ist dabei immer von den konkreten Ideen der Modeschaffenden abhängig. Dadurch entstehen Schablonen, basierend auf designten Inspirationen, welche in aller Regel für hunderte oder gar tausende Kleidungsstücke verwendet werden.

Generative Gestaltung in der Modebranche birgt gleich mehrere Vorteile. Der Designer / die Designerin wird in die Lage versetzt, durch die Vorgabe einiger Parameter zahllose Stoff- und Schnittmuster generieren zu lassen. Mithilfe dieser Parameter kann eine Vielzahl an automatisch generierten Versionen eines bestimmten Kleidungsstyps, Farbmusters etc. erstellt werden. Bei der herkömmlichen Modeschöpfung dagegen, muss jede minimale Änderung manuell durchgeführt werden. Außerdem entfällt durch Rückgriff auf die Methoden generativer Gestaltung zumindest teilweise die Begrenzung durch die eigenen Inspirationen. Die Automatisierung der Generierung verschiedenster Varianten führt theoretisch zu einer unendlichen Masse an generierbaren Versionen eines bestimmten Modestückes.²⁶

Damit lassen sich durch prozedurale Generierungen Unikate in Massenproduktion erstellen. So können Formen und Muster erstellt werden, die zwar dem gleichen vorgegebenen Regelwerk unterliegen, in ihrer Ausfertigung aber unterschiedlich sind.²⁷ (vgl. Abb. 12)

„Durch generative Gestaltung können Muster entwickelt werden, die sich zufällig und dennoch kontrolliert verändern. Diese Gestaltungstechnik bietet die Möglichkeit einen Unikatcharakter [...] industriell in grossen [sic!] Stückzahlen umzusetzen. Womit ‹Einzigartigkeit in Serie› nicht länger widersprüchlich ist.“²⁸

²⁶ vgl. Häberle 2013, S. 2f.

²⁷ vgl. Häberle 2011, S. 19.

²⁸ ebd.



Abb. 12: Einzigartige Modestücke | generativ erstellte Muster

Neben den ästhetischen und individualistischen Überlegungen kann generative Gestaltung in der Modebranche auch im Bereich der Funktionskleidung eine Rolle spielen. Dabei ist denkbar, durch die Berücksichtigung verschiedener Parameter bestimmte Aspekte wie bspw. Atmungsfähigkeit oder Stützfunktion weiter zu verbessern und zu individualisieren.

Under Armour hat bereits eine Schuhdämpfung mittels einer generativen Gestaltungssoftware entwickelt, welche sich am Wurzelwerk von Bäumen orientiert. Die Idee hierbei war, dass Wurzeln durch die Art ihrer Verzweigungen dem Gewicht eines Baumes gut standhalten können. Diese Überlegung sollte auf eine Schuhdämpfung übertragen werden, welche ein hohes Gewicht durch sportliche Aktivitäten aushält, aber dennoch einen hohen Tragekomfort aufweist. Ziel dabei war es, einen Schuh zu entwickeln, der für verschiedene Sportarten geeignet ist und damit flexibel und stabil sein sollte und gleichzeitig genügend Halt und Dämpfung für alle Arten sportlichen Trainings bietet. Bei der Entwicklung der Dämpfung wurde mit einer generativen Gestaltungssoftware eine Gitterstruktur erstellt, welche durch Kombination verschiedenster Muster und Zwischenräume variierte. Dabei konnten verschiedenste Entwürfe verglichen und getestet werden, bis das gewünschte Ergebnis erreicht war.²⁹

Generative Gestaltung im Bereich der Mode- und Textilproduktion birgt die Chance, zukünftig mehr individuelle, aber auch funktionalere Mode zu erschwinglicheren Preisen zu erschaffen. Allerdings bleibt zu beachten, dass nicht alle Muster- und Schnittvarianten, welche mittels generativer Methoden dabei erstellt werden, auch den ästhetischen und praktischen Wünschen der Kunden und Kundinnen bzw. der Modeschaffenden entsprechen werden.³⁰ Diese unerwünschten Ergebnisse sind durch eine Verfeinerung der Parameter zumindest eindämmbar, wenn auch nicht komplett vermeidbar. Die Zeit wird zeigen, welche neuen Kreationen und Ideen Modeschaffende unter Rückgriff auf generative Gestaltung in Zukunft erstellen werden. Die Modebranche steht in diesem Feld erst am Anfang und es ist kaum absehbar, welche Innovationen noch zu erwarten sind.

²⁹ vgl. Brown 2016.

³⁰ vgl. Häberle 2011, S. 19.

2.7 Exkurs: 3D-Druck

Alle der genannten Themen lassen sich sehr gut in den immer mehr an Bedeutung gewinnenden Bereich des 3D-Drucks übernehmen. So ist der 3D-Druck für die Entwicklung von Prototypen (Prototyping) bestens geeignet. Vor der finalen Erstellung des Produktdesigns werden die verschiedenen Entwürfe gedruckt und weiter überarbeitet. So lassen sich das Produkt an sich, aber auch seine Gestaltung, Funktion und weitere Eigenschaften besser beurteilen und testen. Fehler können früher erkannt und beseitigt werden. Diese Herangehensweise bei der Produktfinalisierung spart Zeit und Kosten.

Auch in der Architektur kommt der 3D-Druck immer häufiger zum Einsatz. So können zum Beispiel schon ganze Häuser gedruckt werden. Auch wenn diese eher konventionell als generativ erstellt wurden, so ist dennoch die Möglichkeit der freien Gestaltung gegeben und die Einbeziehung generativer Gestaltungsmethoden ist denkbar. (vgl. Kapitel 2.3)

Im audiovisuellen Bereich wird der 3D-Druck häufig in Stopmotion-Animationen verwendet. Individuelles und manuelles Anpassen einzelner Objekte oder Charaktere kann damit wegfallen. Durch die generative Gestaltung dieser Formen lassen sich verschiedenste Versionen in kürzester Zeit generieren und mittels 3D-Druck ausdrucken. Hierin steckt eine enorme Zeitersparnis und Entlastung für die ohnehin sehr aufwendigen Stopmotion-Produktionen.

Selbst Schmuck, seien es Anhänger, Armbänder oder Ringe, kann mittlerweile mittels 3D-Druck erstellt werden. Hierbei wird häufig auf generativ gestaltete Formen zurückgegriffen, da diese schnell anpassbar sind und so den Druck verschiedenster Muster ermöglichen.

2.8 Zusammenführung

Wie bereits eingangs beschrieben (vgl. Kapitel 2.1), bildet die Natur die Vorgabe und Inspiration für viele mittels generativer Gestaltung umgesetzte Ideen. Aber auch in allen anderen Wissenschaften und Fachbereichen spielt die Natur seit jeher eine wichtige Rolle. Bei allen Erfindungen, wissenschaftlichen Überlegungen und neuen Entwicklungen orientieren sich Menschen an der Natur. Was dabei für den Menschen umsetzbar ist und was nicht, bestimmen die zur Verfügung stehenden Werkzeuge und das Verständnis für den Aufbau der natürlichen Muster. Generative Gestaltung beinhaltet damit eine kleine Revolution für die Erforschung und Umsetzung neuer (natur-) basierter Entwicklungen. Bisher nicht oder nur sehr aufwendig plan- und umsetzbare Erfindungen und Werke können erstellt und angepasst werden. Muster können individualisiert werden, bis sie den gewünschten Effekt erzielen.

Zur praktischen Einbettung dieser Arbeit sollte mit dem vorangegangenen Kapitel ein, bei Weitem nicht vollständiger, Überblick über die möglichen Anwendungsbereiche generativer Gestaltung gegeben werden. Auch wenn sich die vorliegende Arbeit im weiteren Verlauf verstärkt Generierungen in Medienproduktionen zuwendet, wurde bei der Übersicht der Anwendungsbereiche bewusst auch auf andere Einsatzmöglichkeiten eingegangen, um einen Eindruck der vielfältigen Ausprägungen prozeduraler Generierungen zu geben. Die digitale Visualisierung ist einer der Grundsteine in den genannten Anwendungsbereichen und somit in allen verwendbar. Generative Gestaltung hält Einzug in allerlei Fachbereiche und entwickelt sich somit zu einer fächerübergreifenden Disziplin. Dabei ist kaum absehbar, in welchen Branchen sie sich durchsetzen und zum Standardrepertoire werden wird und in welchen Bereichen sie eine untergeordnete Rolle spielen oder gar ganz verschwinden wird. Auch ist kaum vorhersehbar, welche weiteren Disziplinen in Zukunft noch von den Methoden generativer Gestaltung profitieren werden.

3. Software - SideFXs Houdini

Im Folgenden soll die für die vorliegende Arbeit verwendete Software Houdini des Unternehmens Side Effects Software (SideFX) vorgestellt werden. Es geht dabei nicht darum, Houdini in all seinen Funktionen zu erklären, sondern lediglich deren grundlegende Vorteile im Bereich der generativen Gestaltung kurz darzustellen.

Mehrfach ausgezeichnet und führend in der Branche Film bietet SideFX seit 1987 Künstlern und Künstlerinnen Werkzeuge zur prozeduralen Generierung von 3D-Animationen und visuellen Effekten in den Bereichen Bewegtbild, Kunst und Wissenschaft an.³¹

Das Besondere an Houdini ist der einzigartige prozedurale Ansatz, den die Software verfolgt. Dabei ist es möglich, ein ganzes System zur Gestaltung eines bestimmten Objektes zu erstellen und nicht einfach nur das Objekt selbst. Mit diesem prozeduralen Aufbau wird ein Modell kreiert, in welchem sich alle Aspekte und Teilbereiche kontrollieren lassen. Damit muss nicht jeder abweichende Entwurf von Neuem erstellt werden, sondern es besteht die Möglichkeit, über Nodes einzelne Aspekte des Objektes zu verändern. Diese Nodes lassen sich in ihrem Effekt auf so viele andere Nodes wie gewünscht anwenden. Und genau hier setzt das System an. Mit der Node-basierten Architektur lassen sich Projekte wie eine Maschine steuern. In anderen Programmen werden Änderungen als Verlauf der Arbeit meist nur temporär gespeichert. In Houdini dagegen werden dieser Verlauf und die Nodes Teil des Workflows und erlauben eine flexible Produktion.³²

Die Software kann man sich wie einen Baukasten vorstellen. Für jeden Einsatz gibt es das richtige Werkzeug. Über die grafische Oberfläche lassen sich schon sehr viele Funktionen finden und einstellen. Die Benutzeroberfläche mit ihrer Vielzahl an Symbolen, Beschriftungen und Reitern wirkt auf den ersten Blick erschlagend. Dafür sind jedoch selbst für fortgeschrittene Modellierungen, Animationen und Simulationen enorme Vorgaben vorhanden.

VEX ist die Scriptsprache, welche in Houdini hauptsächlich verwendet wird. Die Sprache ist leistungsfähig, ähnlich wie ein kompilierter C/C++ Code. VEX ist eine kleine, effiziente Allzweckssprache zum Schreiben von Shadern und benutzerdefinierten Nodes. VEX basiert lose auf C, nimmt aber auch Ideen aus C++ sowie der RenderMan Shading-Sprache auf.³³

Diese Eigenschaften machen Houdini im Bereich der generativen Gestaltung zu einer vielseitig einsetzbaren Software, die insbesondere durch ihren prozeduralen Ansatz überzeugt. Hervorzuheben ist dabei die *nicht destruktive* Vorgehensweise, d.h. die Eigenschaft, Änderungen als eigenständige Abschnitte zu speichern und somit vorherige Versionen zu einem späteren Zeitpunkt wieder aufrufen und verändern zu können.

³¹ vgl. Side Effects Software Inc. 2018.

³² vgl. Lindblad 2018.

³³ vgl. Side Effects Software Inc. o.J.a

Neben Houdini gibt es weitere Software, welche zur Umsetzung generativer Gestaltung genutzt werden kann. Einige dieser Programme sollen im Folgenden kurz erwähnt werden.

Processing³⁴, eine frei zugängliche Software, interpretiert die Programmierung der Kreativschaffenden mittels Java (p5.js → Javascript) und setzt diese um. In dem Programm wird auf eine Oberfläche mit diversen Buttons und Einstellungsmöglichkeiten verzichtet. Die in Processing verwendete Programmiersprache ist relativ leicht zu erlernen und in Bezug auf visuelle Effekte einfach umzusetzen. Die Software wird häufig in Agenturen aber auch in der Wissenschaft und Bildung zur Visualisierung von Informationen, Forschungsergebnissen, zur Bildverarbeitung, für interaktive Inhalte wie auch für prozedurale Kunstwerke in 2D oder 3D verwendet. Außerdem beinhaltet Processing die Möglichkeit, auf andere Hard- und Softwarekomponenten zuzugreifen und so eine weitaus größere Vielfalt an Einsatzmöglichkeiten zu schaffen.

Bei der Vorstellung weiterer Software, die zur generativen Gestaltung genutzt werden kann, darf die Erwähnung des Unternehmens Autodesk³⁵ nicht fehlen. Dieses hat es sich zur Aufgabe gemacht, diverse Bereiche der Industrie- und Kreativbranche mit ihrem Softwareangebot abzudecken. Software für 3D-Design, Konstruktion, Planung und Entertainment finden sich in Autodesks Repertoire. Maya und 3ds Max sind zwei weitverbreitete Angebote zur Erstellung visueller Erzeugnisse, welche unter anderem für Film- und Fernsehproduktionen genutzt werden. Houdini ermöglicht sogar einen Austausch der Digital Assets mit Maya, so können die Vorteile beider Softwares genutzt werden. Besonders hervorzuheben ist allerdings die Software Fusion 360 Ultimate³⁶ mit der die Möglichkeit besteht, Produkte generativ zu gestalten und dabei durch Simulationen in ihrer Funktion und Statik nicht zu beeinträchtigen. Diese Erweiterung um Funktionen generativer Gestaltung entstand durch die Erkenntnisse des Projektes *Dreamcatcher*. Dieses basiert zum Teil auf Forschungsergebnissen in Zusammenhang mit der Airbus-Trennwand, die in Kapitel 2.5.1 bereits erwähnt wurde.

Eine weitere Software, welche in Zusammenhang mit generativer Gestaltung genannt werden sollte, ist Blender³⁷. Blender ist eine Open Source Anwendung, die mittlerweile eine große Community rund um den Globus hat und immer mehr Anklang in der Design- und Bewegtbildbranche findet. In dieser Arbeit wurde zum Teil Blender verwendet, um die Beispiele der mit Houdini generierten Geometrien zu rendern. Diese Software ermöglicht dem Anwender / der Anwenderin 2D- und 3D-Animation zu erstellen, modellieren, texturieren, riggen und vieles Weiteres. Seit geraumer Zeit enthält Blender die Funktion, Materialien prozedural auf Basis eines Nodes-Systems zu generieren. Außerdem beinhaltet Blender mittlerweile auch die Möglichkeit, mittels dem Animation Node System, Einfluss auf Geometrien und Animationen zu nehmen, was auch in Houdini gegeben ist. In der Umsetzung

34 vgl. Fry / Reas o.J.

35 vgl. Autodesk, Inc. 2018a.

36 vgl. Autodesk, Inc. 2018b.

37 vgl. Blender Foundation o.J.

generativer Gestaltung ist Blender nicht so stabil und umfangreich wie Houdini, dennoch findet dieses Programm immer mehr Anklang und wird durch die Community sowie der Blender Foundation stetig weiterentwickelt.

Jede der genannten Softwares hat ihre Daseinsberechtigung. Sie sind für unterschiedliche Zwecke ausgelegt und mehr oder minder mit einer grafischen Oberfläche oder mit Script- / Programmiersprachen zu bedienen. Es gibt weitere Softwares, die sich auf das Prinzip der Nodesysteme stützen und somit den Trend der *nicht destruktiven* Komposition verfolgen. Zu nennen sind hier ein paar führende Programme wie *Substance Designer* zur Erstellung von Texturen, *DaVinci Resolve* für Farbkorrekturen von Bewegtbild und *Nuke* für Bewegtbildbearbeitung und Compositing.

In dieser Arbeit wird Houdini verwendet, weil diese Software sehr mächtig im Bereich der visuellen Produktion und darüber hinaus ist. Zahlreiche Simulationen, Animationen, Programmierungen und Generierungen können damit umgesetzt werden. Zudem ist Houdini führend in der Erstellung von Visual Effects und gewinnt immer mehr Bedeutung in der Spielebranche, wo sie besonders zur prozeduralen Generierung ganzer Welten und Objekte eingesetzt wird.

4. Umsetzungswege

4.1 Lindenmayer System / Baumgenerierung

4.1.1 Vorüberlegungen

Ein möglicher Umsetzungsweg generativer Gestaltung ist das L-System, benannt nach Astrid Lindenmayer (1925-1989). Dieses macht die Generierung komplexer Formen durch die Nutzung von Iterationen möglich. Durch eine mathematische Sprache wird zunächst eine Zeichenfolge definiert, welche anschließend mit vorgegebenen Regeln verglichen und wiederholt ausgewertet wird. Daraus resultieren diverse generative Formen und Objekte, bei welchen jede weitere Auswertung die Basis für die nächste Iteration ist. Dadurch entsteht der Eindruck, das generierte Objekt bzw. die generierte Form würde wachsen.³⁸

Das wesentliche Prinzip, welches das L-System ausmacht, ist die Umschreibung. Bei diesem Prozess werden die Teile einer simplen Ausgangsdefinition durch komplexere Folgedefinitionen ersetzt. Die Ersetzungs- und Produktionsregeln werden dabei im Voraus festgelegt und bestimmen damit die Parameter, welche vorgeben, wie sich die zu generierende Form entwickeln wird.³⁹

Das L-System ist ein vielfältig einsetzbarer Umsetzungsweg generativer Gestaltung und kann für die Erstellung von Fraktalen genutzt werden. Nahezu jede Ausgangsform lässt sich vervielfältigen, durch Verfeinerungen ersetzen und weiterentwickeln. Die Steuerung durch vorab festgelegte Parameter erfüllt den Zweck, die zu generierende Form an gewisse Bedingungen zu knüpfen. Die verschiedensten damit erzeugbaren Variationen sind dennoch so facettenreich und händisch kaum umsetzbar. Damit birgt das L-System einen enormen Wert für die Entwicklung generativer Objekte.

Für die einfachere Veranschaulichung der Umsetzung des L-Systems bedarf es zunächst einiger Vorüberlegungen, um zu verstehen, nach welchen Prinzipien und Grundregeln dabei gearbeitet wird. Auf diese soll im Folgenden genauer eingegangen werden.

Um ein Wachstum durch ein L-System zu beschreiben, ist zuerst eine speziellen Zelle vonnöten, welche die Ausgangsbasis für das Wachstum darstellt. Mittels einer Beschreibung kann dann bestimmt werden, wie eine neu hinzukommende Zelle auf Basis der alten Zelle generiert werden soll. Die Ausgangszelle wird Axiom genannt und die Beschreibung, wie neue Zellen aus alten Zellen wachsen, wird Produktionsregel genannt.⁴⁰

Neben diesen Definitionen bedarf es zur Umsetzung von L-Systemen und der damit verbundenen Generierung von Fraktalen in der Computergrafik einer Methode, die es ermöglicht,

³⁸ vgl. Side Effects Software Inc. o.J.b.

³⁹ vgl. Linendmayer / Prusinkiewicz 2004, S. 1.

⁴⁰ vgl. Flake 1998, S. 78.

Anweisung und Befehle simpel zu veranschaulichen. Eine dafür gut nutzbare Methode ist unter der Bezeichnung *Turtle Grafik* zu finden. Diese ist eine bildliche Vorüberlegung dafür, wie eine Schildkröte Befehle zu ihrer Fortbewegung ausführt. So kann definiert werden, wann sie sich zum Beispiel vorwärts und rückwärts bewegen soll und über welche Strecke sie dies tut. Auch Drehungen zu einer bestimmten Seite und um einen bestimmten Winkel sind denkbar. Indem man sich nun vorstellt, dass die Schildkröte eine Spur hinterlässt, die ihren bisher gegangenen Weg abbildet, wird deutlich, dass durch die Turtle Grafik geometrische Figuren erstellt werden können.⁴¹

4.1.2 Wissenschaftliche Einbettung

L-Systeme ermöglichen die Definition komplexer geometrischer Formen durch geregelte Wiederholungen und Ersetzungen. Mittels einer mathematischen Sprache werden anfängliche Zeichenfolgen auf Grundlage verschiedener Regeln ersetzt und ausgewertet. Die daraus resultierenden Ergebnisse werden dazu benutzt, geometrische Formen zu generieren.⁴² (vgl. Abb. 13)

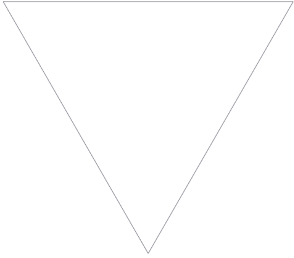
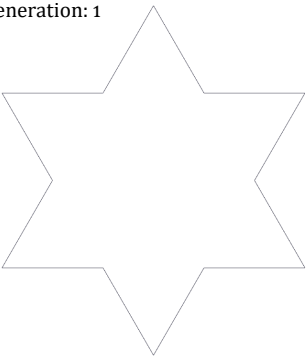
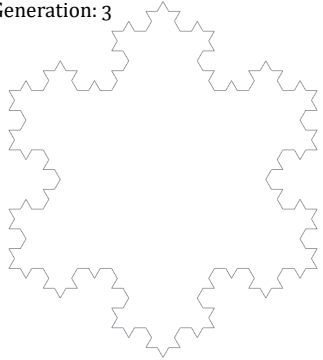
Kochsche-Schneeflocke Winkel: 60 ° Axiom: F + + F + + F Regel: F = F - F + + F - F			
Generation: 0	Generation: 1	Generation: 3	
			
F Schritt vor	+ Rechtsdrehung	- Linksdrehung	= Ersetzen

Abb. 13: Turtlegrafik der Kochschen Schneeflocke

⁴¹ vgl. Abelson / diSessa 1986, S. 3.

⁴² vgl. Side Effects Software Inc. o.J.b.

Das Prinzip, nach welchem sich die vorangegangene Grafik aufbaut, wird in Abb. 14 genauer dargestellt:

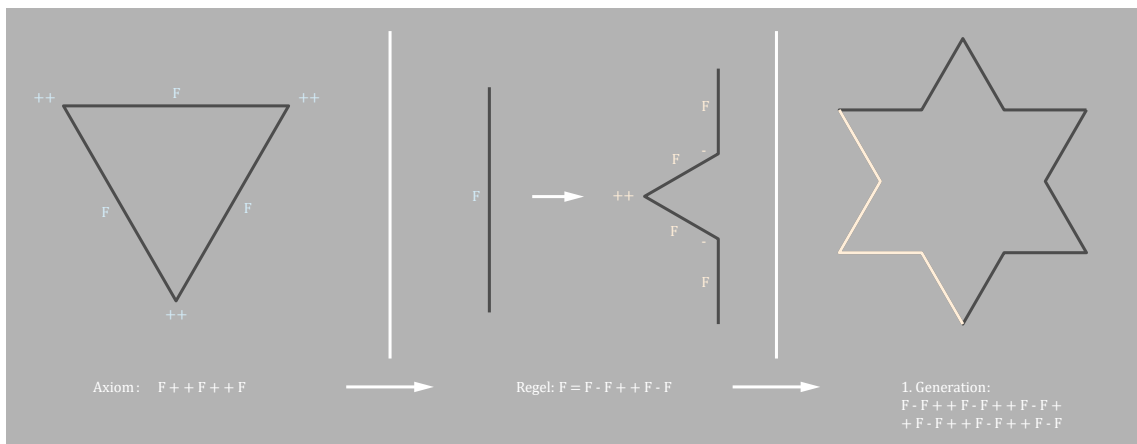


Abb. 14: Prinzip der Ersetzung des L-Systems

Mit dem L-System lassen sich diverse fraktale Formen generieren. Nicht nur Formen wie diese Kochsche-Schneeflocke oder das Sierpinski-Dreieck sind denkbar. Es lassen sich auch organische Verzweigungen für Pflanzen wie Bäume und Büsche, komplexe Systeme und Vernetzungen, wie zum Beispiel Nervenbahnen realisieren.

Durch das Aneinanderreihen von definierten Wegen, den Regeln, an einem Axiom und / oder der vorherigen Zelle wird ein immer größeres Geflecht gebildet. Hier kommt ein System mit Speicher ([,]) zum Einsatz, welches definiert, an welche Stelle nach einer bestimmten Aktion zurückgekehrt werden soll. Zur Veranschaulichung von pflanzenähnlichen Fraktalen dient das folgende Beispiele in Abb. 15:

Pflanzenähnliches Fraktal	Winkel: 45°	Axiom: F	Regel: F = F[-F]F[+F]F
Generation: 0 	Generation: 1 	Generation: 4 	
[Merke die aktuelle Position] Kehre zur zuletzt gemerkten Position zurück			

Abb. 15: Turtlegrafik einer Pflanze

Die vorangegangene Grafik wird mittels dem Prinzip der Aneinanderreihung (Abb. 16) erzeugt und folgend dargestellt:

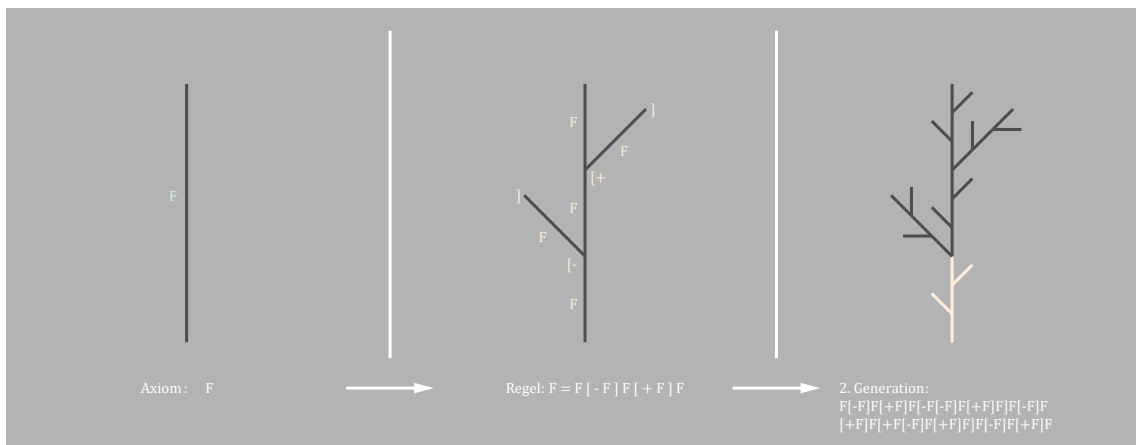


Abb. 16: Prinzip der Aneinanderreihung des L-Systems

4.1.3 Umsetzungsweg in Houdini

Um die Realisierung von L-Systemen nun noch konkreter zu machen und praktisch zu veranschaulichen, wird im Folgenden die Umsetzung in Houdini aufgezeigt. Hierfür wird eine vordefinierte Schnittstelle benötigt. Auf Basis des Tutorials von Ari Danesh⁴³ werden die einzelnen Schritte, ausgehend von der vordefinierten Schnittstelle, aufgezeigt und umgesetzt.

In Houdini muss zunächst im Reiter *Create* das L-System ausgewählt werden. Ein einfaches dreidimensionales System steht direkt zur Verfügung. Innerhalb der Geometrie des L-Systems kann auf der Szenenebene auf die Einstellungen zugegriffen werden. (vgl. Abb. 17)

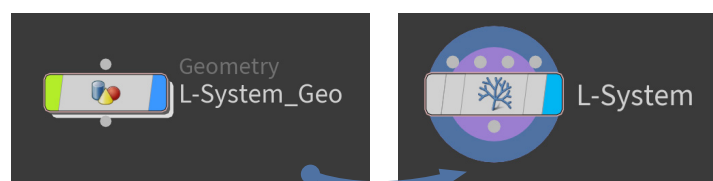


Abb. 17: Das L-System innerhalb des *Geometry Nodes*

Houdini hat die Anwendung von L-Systemen für die Nutzer und Nutzerinnen stark vereinfacht, indem schon diverse Voreinstellung (Zahnrad) für eine schnellere Umsetzung zur Verfügung gestellt werden. Um mit einem leeren L-System zu beginnen, sollten zunächst ein paar Einstellungen vorgenommen und als Grundeinstellung gespeichert werden (vgl. Abb. 18, Abb. 19 und Abb. 20).

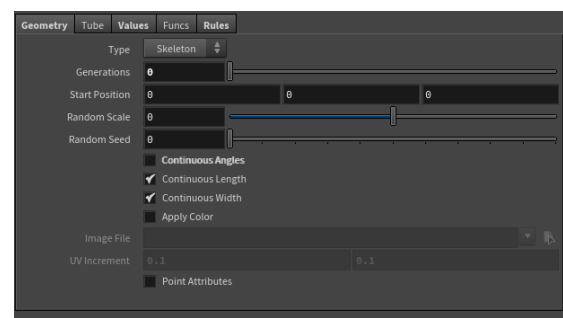


Abb. 18: Geometrieparameter des L-Systems in Houdini

⁴³ vgl. Danesh 2012.

In Houdini wird das Axiom *Premise* genannt. Trägt man nun im Reiter *Rules* unter *Premise* das Axiom und unter *Rule1* usw. die Produktionsregel ein, wird das gewünschte L-System generiert. Um das Wachstum zu steuern, findet sich im Reiter *Geometry* der Regler *Generations*. Je höher man diesen setzt, um so mehr wird das System iteriert. Unter dem Tab *Values* sind viele weitere Einstellungsmöglichkeiten zu finden. Die Gradzahl eines System wird mit dem Parameter *Angle* eingestellt.

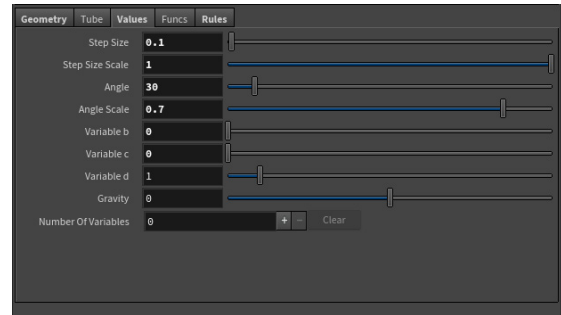


Abb. 19: Weitere Parameter des L-Systems in Houdini

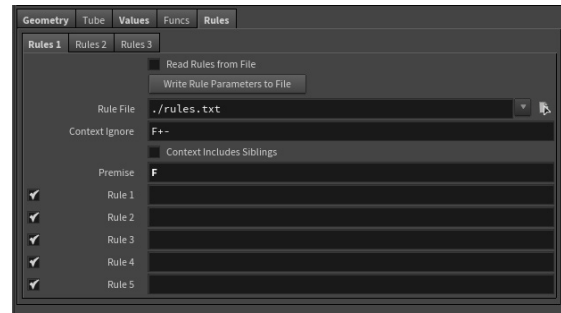


Abb. 20: Regeleingabe für das L-System in Houdini

Wenn ein dreidimensionaler Baum generiert werden soll, werden weitere Anweisungen benötigt. Hinzu kommen Befehle wie Drehen (roll) und Neigen (pitch). Zudem kann der Einsatz der Selbstreferenz eine weitere Komplexität erzeugen (vgl. Abb. 21).⁴⁴

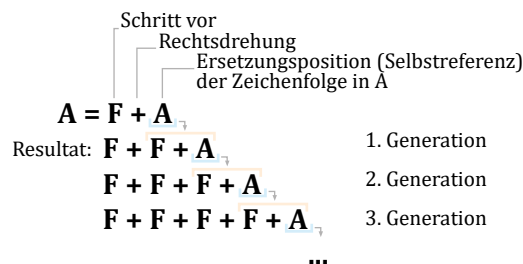
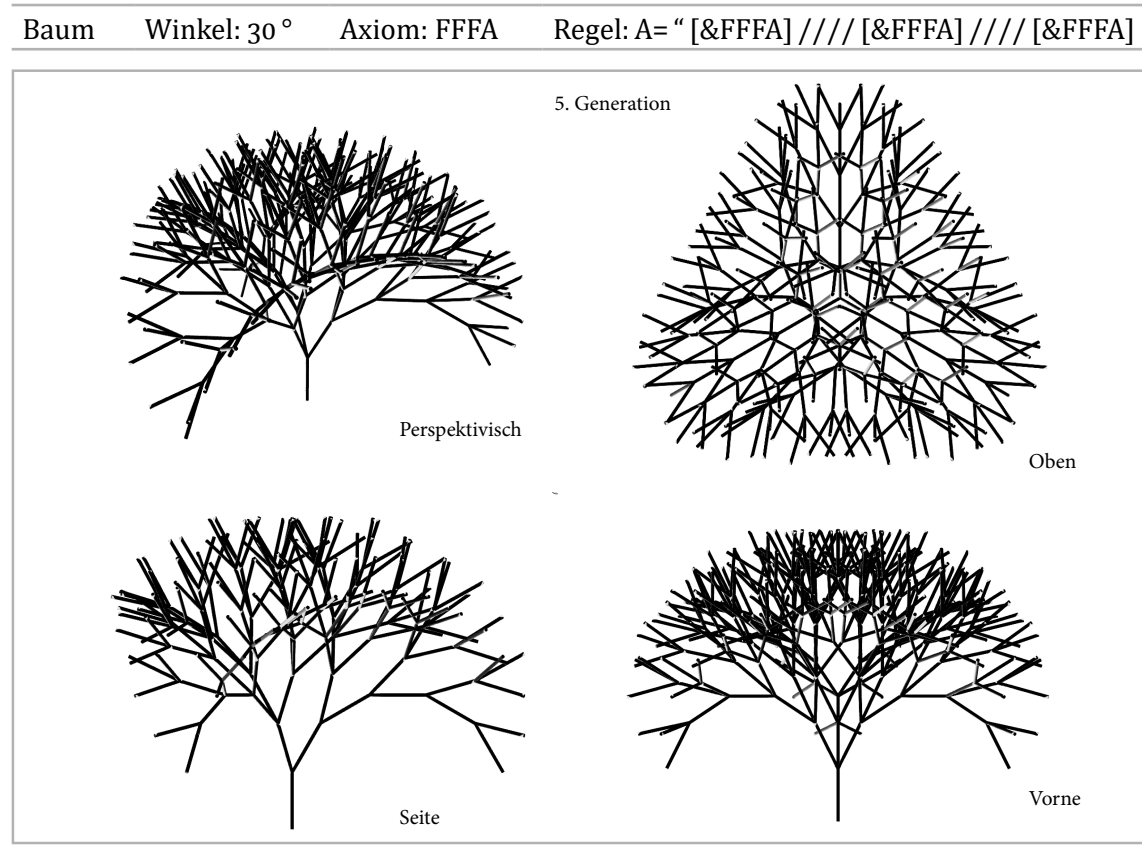


Abb. 21: Beispiel einer Selbstreferenz

4.1.4 Resultat

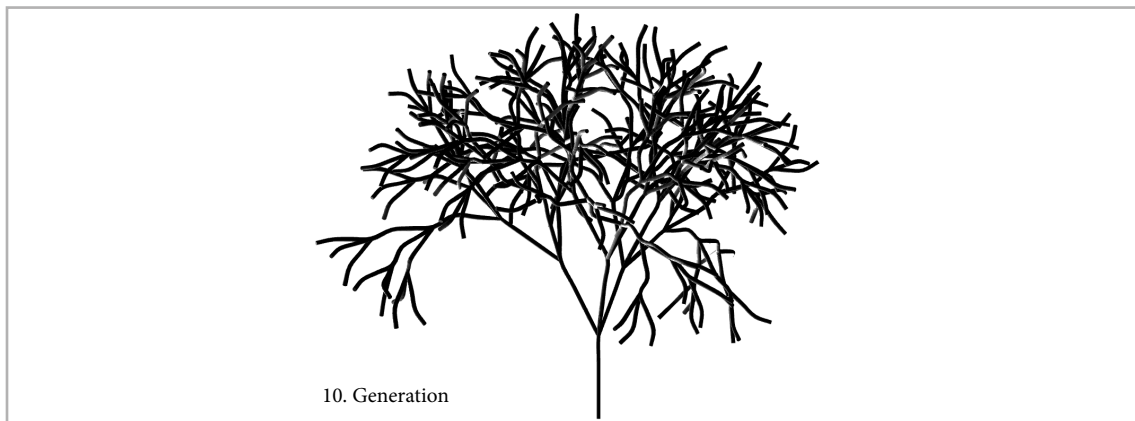


" Multiplikation der aktuellen Länge & Neigen nach oben
/ Gegen den Uhrzeigersinn rollen

Abb. 22: 3D-Turtlegrafik eines Baumes

Um die in Abb. 22 dargestellte Baumform noch realistischer und mehr der Natur entsprechend zu generieren, müssen weitere Variablen und Anweisungen verwendet werden (vgl. Abb. 23):

Baum 2	Winkel: 30 °	Axiom: FFFA	Regeln:
			A:t<5!="[B]////[B]////B
			A:t>=5!="[C]\\&[C]////C
			B=&FFFAJ
			C=T^FF-FAK



t Iterationen (Generation)	! Multiplikation der aktuellen Dicke
: Bedingte Anweisung	B, C Ersetzungsposition ^ Neigen nach unten
J, K Eingang für weitere Geometrie (Leaf input)	T Gravitation
\\ Im Urzeigersinn rollen	

Abb. 23: 3D-Turtlegrafik eines Baumes mit weiteren Variablen

Da in den Regeln eine Anweisung zur Dicke verwendet wurde, entfaltet sich dieser Effekt wenn der *Typ* des L-Systems unter dem Tab *Geometry* von *Skeleton* auf *Tube* gestellt wird. Der Reiter *Tube* ist danach verwendbar und es kann damit die gewünschte Dicke eingestellt werden. Durch Ändern der Parameter kann die Beschaffenheit, die Geometrie und das Aussehen des Baumes variiert werden. Da der Parameter *Gravity* in einer der vorgegebenen Regeln verwendet wird, kann er unter *Values* angepasst werden, was wiederum der realistischeren Darstellung zu Gute kommt.

4.2 Space Colonization Algorithmus/Baumgenerierung

4.2.1 Vorüberlegungen

Ein Problem, welches beim L-System früher oder später auftritt, ist die Tatsache, dass *blind* Regeln ersetzt werden, ohne dass externe Gegebenheiten beachtet werden. Auf einfachem Wege ist es nicht möglich, bspw. Sonnenlicht, benachbarte Objekte oder sogar eine Überschneidung der eigenen Geometrie zu berücksichtigen. Einen Algorithmus zu erstellen, der das L-System nutzt, um einen realitätsnahen Baum zu generieren, ist kein leichtes Unterfangen. Die Methode der Verzweigung eines Baums durch das L-System kann zwar leicht übertragen werden, doch das System berücksichtigt nicht das natürliche Wachstum, welches den Baum wie einen Baum aussehen lässt.⁴⁵

Abhilfe kann hier ein autonomes System mit dem Space Colonization Algorithmus (im Weiteren SCA abgekürzt) schaffen. Mit dem SCA können weitere fraktale Geometrien generiert werden, welche noch organischer und realer wirken. Am Beispiel der Erstellung von Bäumen ermöglicht der Algorithmus

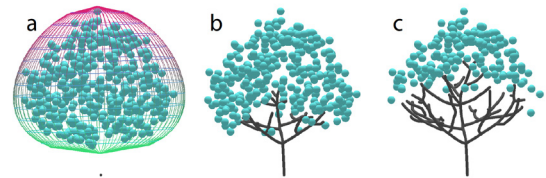


Abb. 24: Methode des SCA

die Bildung eines Geflechtes innerhalb eines vorgegebenen Volumens. Wie in Abb. 24 dargestellt, liegt das Grundprinzip darin, dass bestimmte Punkte (händisch oder zufällig gesetzt) innerhalb eines Raumes (a) eine Vernetzung bilden. Dabei sucht sich ein Ausgangspunkt (a), welcher außerhalb oder innerhalb des Raumes liegen kann, durch eine Definition den besten Weg zu den äußersten Punkten (b). Dabei entstehen während der Wegfindung von Punkt zu Punkt Verzweigungen (c).⁴⁶

4.2.2 Wissenschaftliche Einbettung

Der Space Colonization Algorithmus von Runions, Lane & Prusinkiewicz ist eine dreidimensionale Erweiterung basierend auf der vorherigen wissenschaftlichen Arbeit der Autoren zur Generierung von Blattmustern.

Die Bedingung des SCAs beginnt mit der Initialisierung von N Anziehungspunkten, in der Regel hunderte oder tausende, und mit einem oder wenigen Baumknoten. Der Baum lässt sich iterativ generieren. Während jeder Iteration kann ein Anziehungspunkt unter bestimmten Bedingungen Einfluss auf den Baumknoten ausüben. Wenn der Abstand zwischen dem Punkt und Knoten am geringsten und innerhalb des Einflussradius (d_i) ist, wird als Abschnitt (v, v') ein neuer Knoten (v') gebildet und an (v) angefügt. Mehrere Punkte können Einfluss auf einen Baumknoten v haben, was als ($S(v)$) bezeichnet wird. Der Knoten v' ist in einem

⁴⁵ vgl. Space Colonization 2011.

⁴⁶ vgl. Runions / Lane / Prusinkiewicz 2007.

Abstand (D) von v positioniert, dessen Richtung durch den Durchschnitt der normierten Vektoren zu allen Quellen ($s \in S(v)$) definiert wird.⁴⁷

Folglich, $v' = v + D \hat{n}$, wo $\hat{n} = \frac{\vec{n}}{\|\vec{n}\|}$ und $\vec{n} = \sum_{s \in S(v)} \frac{s - v}{\|s - v\|}$.

Optional kann die Wachstumsrichtung durch den Vektor \vec{g} beeinflusst werden. Dabei werden die Richtung der Verzweigung und die Tropismen unter Verwendung folgender Gleichung kombiniert:

$$\tilde{n} = \frac{\hat{n} + \vec{g}}{\|\hat{n} + \vec{g}\|}$$

Wenn die neuen Knoten hinzugefügt wurden, findet eine Überprüfung statt, um zu testen, welche der Anziehungspunkte, wenn es welche gibt, aufgrund der Nähe zu den Ästen, die zu diesen Punkten hin gewachsen sind, entfernt werden sollen. Entfernt wird ein Anziehungspunkt s dann, wenn es mindestens einen Baumknoten (v) gibt, welcher näher an s ist als der Schwellenwert „kill distance“ d_k .⁴⁸

In Abb. 25 ist der Algorithmus illustriert, welcher mit sechs Knoten (schwarz) und vier Anziehungspunkten (blau) beginnt (a). Jeder Punkt ist mit dem Knoten verbunden, welcher ihm am nächsten ist. Als Bedingung gilt allerdings, dass der Knoten innerhalb des Einflussradius liegen muss (b, blaue Linie). Die normierten Vektoren, von jedem Knoten zu jeder Quelle, welche Einfluss auf den Knoten haben, werden gefunden (c,

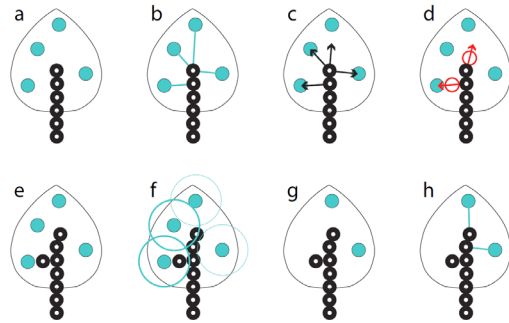


Abb. 25: Ablauf des Algorithmuses

schwarze Pfeile). Diese Vektoren werden hinzugefügt und deren Summe wird wiederum normiert (d, rote Pfeile). Dadurch ergibt sich die Grundlage zur Lokalisierung neuer Baumknoten (d, rote Kreise). Die neuen Knoten werden in die Baumstruktur eingefügt und führen zu einer Verlängerung der Hauptachse und zum Wachstum eines seitlichen Astes (e). Nun wird die Umgebung der Anziehungspunkte (blaue Kreise) darauf getestet, ob Baumknoten darin eingeschlossen sind (f). Die Radien der zwei Punkte links außen schneiden die neuen Knoten (dicke schwarze Kreise). Diese Punkte werden daher entfernt (g). Die Baumknoten, welche den Punkten am nächsten sind, werden identifiziert (h) und die nächste Iteration des Algorithmus beginnt.⁴⁹

⁴⁷ vgl. Runions / Lane / Prusinkiewicz 2007.

⁴⁸ vgl. ebd.

⁴⁹ vgl. ebd.

4.2.3 Umsetzung in Houdini

Die Umsetzung in Houdini erfolgt durch die Implementierung des Algorithmus in VEX-Code und die Anwendung dieses Codes auf eine Geometrie. Der Aufbau und Ablauf wird folgend in Anlehnung an das Tutorial *VEX in Houdini: Space Colonization*⁵⁰ von Manuel Casasola Merkle dargestellt.

Zu Beginn wird eine Geometrie definiert, in welcher der Baum heranwachsen soll. Für die vorliegende Umsetzung wurde eine Kugel (Center: 0, 2, 0) gewählt. Um Geometrie als Raum, in welchem ein Wachstum stattfindet, nutzen zu können, muss dieses zunächst in ein Volumen umgewandelt werden. In der Geometrie-Ebene der Kugel wird ein *IsoOffset* Node mit dem der Kugel verbunden. Der Wert *Uniform Sampling Divs* kann ein wenig erhöht werden, um die ursprüngliche Form mehr hervorzuheben.

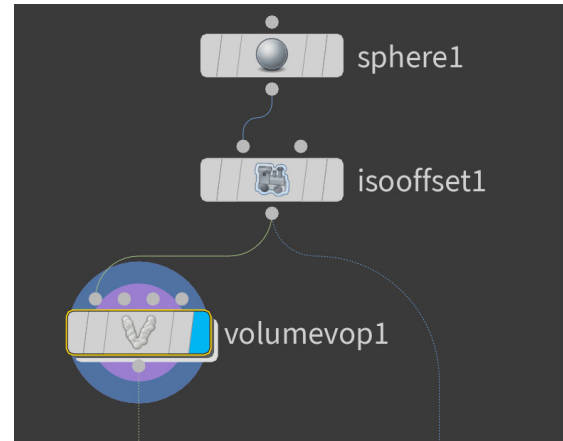


Abb. 26: Volumenerstellung in Houdini

Unter *Name* sollte *density* eingetragen werden, da später auf dieses Volumen zugegriffen wird. Um nun für mehr Variation in der Dichte zu sorgen, wird ein *Volume VOP* Node angehängt. Dieses Node hat innerhalb der *VEX Builder* Ebene (Doppelklick auf das Node) einen Ausgang namens *density* und kann im Zusammenspiel mit dem *IsoOffset* verwendet werden. (vgl. Abb. 26)

In dem VEX-BUILDER wird nun der Ausgang *P* (Position) mit einem *Anti-Aliased Noise* Node am Eingang *pos* verbunden, um die Lokalisierung des Rauschens zu bestimmen. Die Werte von *3D Frequency* werden hier jeweils auf 10 gestellt und *Roughness* auf 0.6 erhöht. Der Ausgang *noise* wird mit einem *Fit Range* Node am Eingang *val* verbunden. Hier werden die Werte *Source Min* und *Source Max* zu -0.5 und 0.5 eingestellt, da sich das Rauschen des *Anti-Aliased Noise* Nodes in diesem Bereich bildet. Daraufhin wird ein *Ramp Parameter* Node am Ausgang *shift* mit dem Eingang *input* verbunden, um damit die Steuerung des Rauschens und der Variation der Dichte besser einzustellen. Der Name wird in *ramp_noise* geändert und *Default Interpolation* auf *Spline Ramp (float)* umgestellt. Nun wird ein *Multiply* Node benötigt, welcher wie eine Maske dafür sorgt, dass das Rauschen nur auf die Dichte anzuwenden ist. Der Ausgang *density* des *Volume VOP Global Parameters* Nodes wird mit dem Eingang *input1* und der Ausgang *ramp_noise* des *Ramp Parameter* Nodes mit dem Eingang *input2* des *Multiply* Nodes verbunden. (vgl. Abb. 27)

⁵⁰ vgl. Merkle 2017.

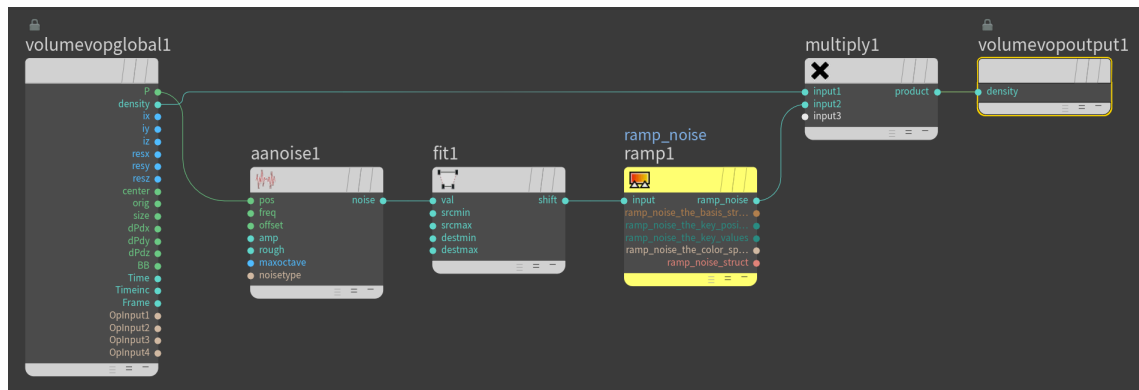


Abb. 27: Rauschen innerhalb des Volumens

Zurück auf der Geometrieebene kann nun am *Volume VOP* die *ramp noise* eingestellt werden, sodass die Dichte ein gewisses Rauschen erhält. (vgl Abb. 28)

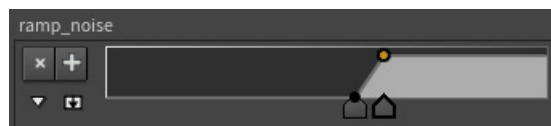


Abb. 28: Verlaufseinstellung der Rauschintensität/-varianz

Um nun aus dem Volumen Punkte zu generieren, wird nach dem *Volume VOP* Node ein *Scatter* Node angefügt. Hier wird der Wert *Force Total Count* auf ca. 50 000 geändert und das Häkchen bei *Relax Iterations* für eine zufällige Anordnung der Punkte entfernt. Da der Algorithmus auf Anziehungspunkte angewiesen ist und durch das Rauschen evtl. zu große Kluften entstehen, sollte zusätzlich noch ein zweites *Scatter* Node an *Iso-Offset* verbunden werden. Außerdem sollten weniger Punkte (ca. 1000) gleichmäßig im Volumen gesetzt werden (*Relax Iterations* beibehalten). Diese zwei *Scatter* Nodes wer-

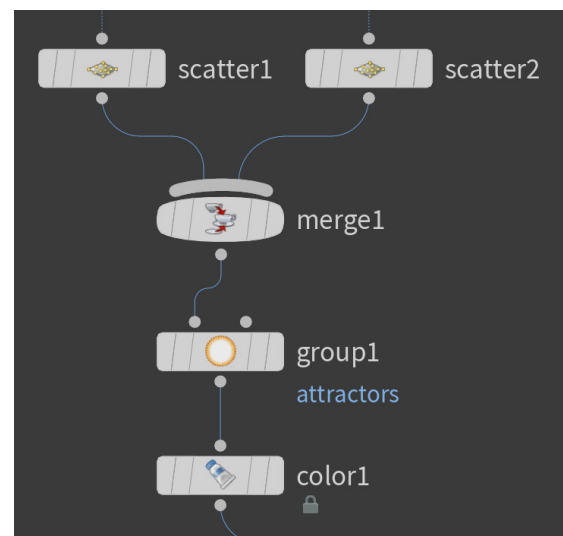


Abb. 29: Erstellung der Anziehungspunkte

den mit einem *Merge* Node zusammengelegt. Daraufhin folgt ein *Group Create* Node, um die erstellten Punkte in eine Gruppe einzuordnen. Diese Gruppe erhält den Namen *attractors*. Für eine visuelle Unterscheidung zwischen den Punkten und den Knoten, wird ein *Color* Node angefügt und eine Farbe ausgewählt. (vgl. Abb. 29)

Für den Baumknoten, also den Startpunkt, wird separat ein *Add* Node eingefügt. Mit diesem Node könnten nun mehrere Knoten erstellt werden, im vorliegenden Beispiel wird es allerdings bei einem Ursprung belassen (0,0,0). Dieser Knoten wird ebenso mit dem Namen *nodes* einer Gruppe zugewiesen. Hierfür wird ein zweites *Group Create* Node erstellt und verbunden. Zudem kann ein erneutes *Color* Node zur weiteren Unterscheidung verwendet werden. (vgl. Abb. 30)

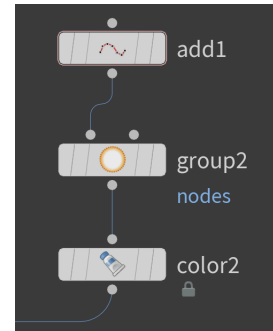


Abb. 30: Erstellung der zu emittierenden Punkte

Die bis hierhin erstellten beiden Stränge werden mit einem *Merge* Node verbunden. Daraufhin wird ein *Solver* Node verwendet, um den Algorithmus zu implementieren. Dieses Node ermöglicht es, über die Zeitspanne (Frames) iterative Effekte zu erstellen. Innerhalb des *Solver* Nodes sind fünf Nodes zu finden. Der *Dop Import* gibt ein Feedback des vorherigen Frames aus, die anderen vier Nodes sind die Eingänge des *Solver* Nodes. An den *Dop Import* Node werden vier *Point Wrangle* Nodes angehängt. (vgl. Abb. 31)

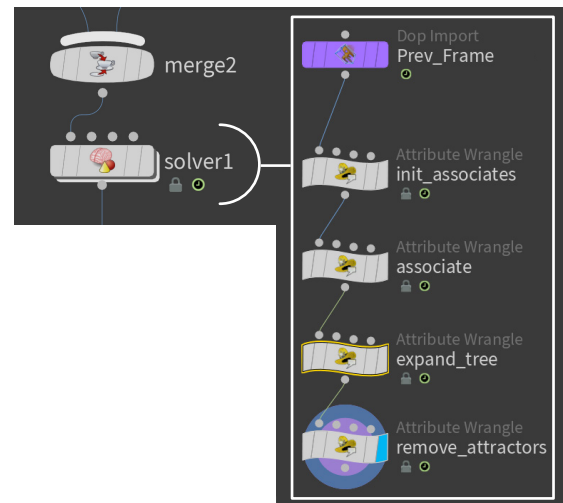


Abb. 31: Anwenden des SCAs

Das erste wird in *init_associates* umbenannt und erhält folgenden VEX-Code:

```
i[]@associates = array(); // Array-Attribut zum Speichern
                        // der am nächsten liegenden Knoten
```

In jedem Schritt wird, sobald der Solver aufgerufen wird, ein leeres Array angelegt, um dieses später mit Werten zu füllen.

Der zweite Node heißt *associate*, der Gruppenname ist *attractors* und beinhaltet diesen Code:

```
float infrad = ch(„infrad“); // Parameter „radius of influence“  $d_i$ 

int nearpnt = nearpoint(0, „nodes“, @P, infrad); /* In der „node“ Gruppe
wird nach den nächstgelegenen Knoten gesucht und in die Variable „nearpo-
int“ gespeichert */

int valarr[] = array(@ptnum); /* Integer können nicht in ein Array
geschrieben werden, somit muss ein Array in ein Array gespeichert
werden. Der aktuelle Punkt „ptnum“ wird als Array in die Variable
„valarr[]“ geschrieben */

setpointattrib(0, „associates“, nearpnt, valarr, „append“);
/* Aktueller Punkt „valarr“ wird an das Atribut „associates“ des Knotens
„nearpnt“ angehängt */
```

Der Gruppenname *attractors* sorgt dafür, dass nur die Anziehungspunkte in der gleichnamigen Gruppe angesprochen bzw. verarbeitet werden. Der Parameter *infrad* wird nun auf 1.3 gesetzt.

Der dritte Node mit dem Namen *expand_tree* und dem Gruppennamen *nodes* führt diesen Code aus:

```
float rad = ch("rad"); // Parameter - Radius der individuellen Knoten.
int myassociates[] = i[@associates; /* Die Werte des Attributs
"associates" werden in eine lokale Variable geladen */

if (len(myassociates) > 0){ /* Alle Verbindungen werden durchsucht und
nur wenn sie vorhanden sind (größer 0) berücksichtigt */

    vector sum = {0,0,0}; /* Wird für die Zusammentragung der späteren
Vektoren benötigt (wenn mehr als eine Verbindung besteht) */

    foreach(int node; myassociates){ // Für jeden Knoten im Array
        vector nodepos = point(0, "P", node); /* Position der
Verbindungen werden geladen */
        vector dir = normalize(nodepos - @P); /* Die Vektoren werden
normiert und so die Richtung bestimmt */
        sum += dir; // Die Richtung wird zusammengetragen
    }

    sum = normalize(sum); // Die Summe der Richtung wird normiert

    int nnode = addpoint(0, @P+sum*rad); /* Ein neuer Knoten wird
mit der Skalierung von "rad" erstellt */

    setpointgroup(0, "nodes", nnode, 1, "set"); /* Der Knoten wird
zur Gruppe hinzugefügt */
    setpointattrib(0, "Cd", nnode, {1,1,1}, "set"); /* Der Knoten
wird eingefärbt */

    int nprim = addprim(0, "polyline"); /* Die Knoten werden mit
einer Linie verbunden */
    addvertex(0, nprim, @ptnum); /* Ein Vertecie des aktuellen
Knotens wird erstellt */
    addvertex(0, nprim, nnode); /* Ein Vertecie für den neuen Knoten
wird erstellt */

}
```

Der Gruppenname *nodes* sorgt dafür, dass nur die Knoten in der gleichnamigen Gruppe angesprochen bzw. verarbeitet werden. Der Parameter *rad* wird auf 0.05 gestellt.

Der Code für den vierten Node mit dem Namen *remove_attractors* und dem Gruppennamen *nodes* lautet:

```
float killrad = ch("killrad"); // Parameter - Radius "kill distance"  $d_k$ 

int nearpnts[] = nearpoints(0, "attractors", @P, killrad); /* Es werden
alle Punkte innerhalb des Radius "killrad" gesucht */

foreach(int pnt; nearpnts){ /* Über alle gefundenen Punkte wird ein
Befehl ausgeführt */

    removepoint(0, pnt); // Die gefunden Punkte werden entfernt
}
```

In diesem Node sorgt der Gruppenname *nodes* dafür, dass nur die Punkte in der gleichnamigen Gruppe angesprochen bzw. verarbeitet werden. Der Parameter *killrad* wird auf 0.3 gestellt.

Zurück auf der *Geometry*-Ebene wird die Animation nun gestartet und der Baum wächst. (vgl. Abb. 32)

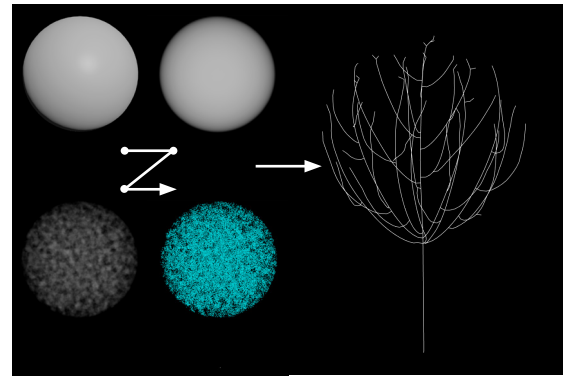


Abb. 32: Teilschritte der Generierung

4.2.4 Resultat

Es ist darauf hinzuweisen, dass in dem Tutorial von Merkle, auf welchem die vorangegangene Darstellung aufbaut, nicht explizit die dargestellten Formeln verwendet werden. Es wird lediglich die Methode mit Houdinis Nodesystem und VEX-Codes nachempfunden. Nicht direkt eingegangen wird auf die optionale Einstellbarkeit der Wuchsrichtung und die Möglichkeit, die Dicke der Äste zu bestimmen. Daher sollen an dieser Stelle diese Erweiterungen noch dargestellt werden.

Durch die Verwendung eines *PolyWires* oder *Wire Nodes* sowie eines *Deletes*, welcher die *attractors* Gruppe nach der Verwendung löscht, kann die wachsende Geometrie sichtbar gemacht werden.

Je nach Einstellung des *kill radius* können mehr oder weniger Verzweigungen und Äste gebildet werden. (vgl. Abb. 33)



Abb. 33: Resultate des SCAs

4.3 Bewegung generieren/Spirograph

4.3.1 Vorüberlegungen

Um Bewegungen zu visualisieren, muss zunächst überlegt werden, was abgebildet werden soll. Für die vorliegende Arbeit wurde hierzu das Beispiel des Spirographen (vgl. Abb. 34) herangezogen. Mit dem beliebigen Kinderspielzeug lassen sich verschiedenste Spiralen und Kurven mit bestimmten Werkzeugen und gleichförmigen Bewegungen erstellen. Die Werkzeuge sind diverse Kreise oder Ovale mit Zacken und Löchern.



Abb. 34: Spirograph (Spielzeug)

Ein Blatt Papier wird auf die Malfläche des Spirographen gelegt. Diese ist ebenfalls rund und weist an ihrem Rand nach innen gerichtete Zacken auf. Einer der kleinen Kreise oder Ovale wird nun auf die Malfläche gelegt. Ein Stift wird in eines der Löcher des Kreises / Ovals gesteckt. Bewegt man nun den Stift in Kreisbewegungen so, dass die Zacken des kleinen Kreises sich mit den Zacken der Malfläche verzahnen, ergibt sich ein ebenmäßiges Gemälde, ähnlich einem Mandala. Je nach gewähltem Kreis oder Oval und der Position des ausgewählten Loches, entstehen diverse Muster.

4.3.2 Wissenschaftliche Einbettung

Die Funktionsweise eines Spirographen lässt sich mathematisch und geometrisch definieren. Diese Definitionen können in die digitale Welt übernommen werden.

Die mathematische Funktion eines Spirographenbildes wird von H. R. Schneebeli und H. R. Vollmer⁵¹ vereinfacht und allgemein so dargestellt:

$$\vec{p}(t) = (R+r) \begin{pmatrix} \cos(\omega t + \tau) \\ \sin(\omega t + \tau) \end{pmatrix} + s \begin{pmatrix} \cos\left(\left(1 + \frac{R}{r}\right)\omega t + \theta\right) \\ \sin\left(\left(1 + \frac{R}{r}\right)\omega t + \theta\right) \end{pmatrix}$$

⁵¹ vgl. Schneebeli / Vollmer 1998, S.39ff.

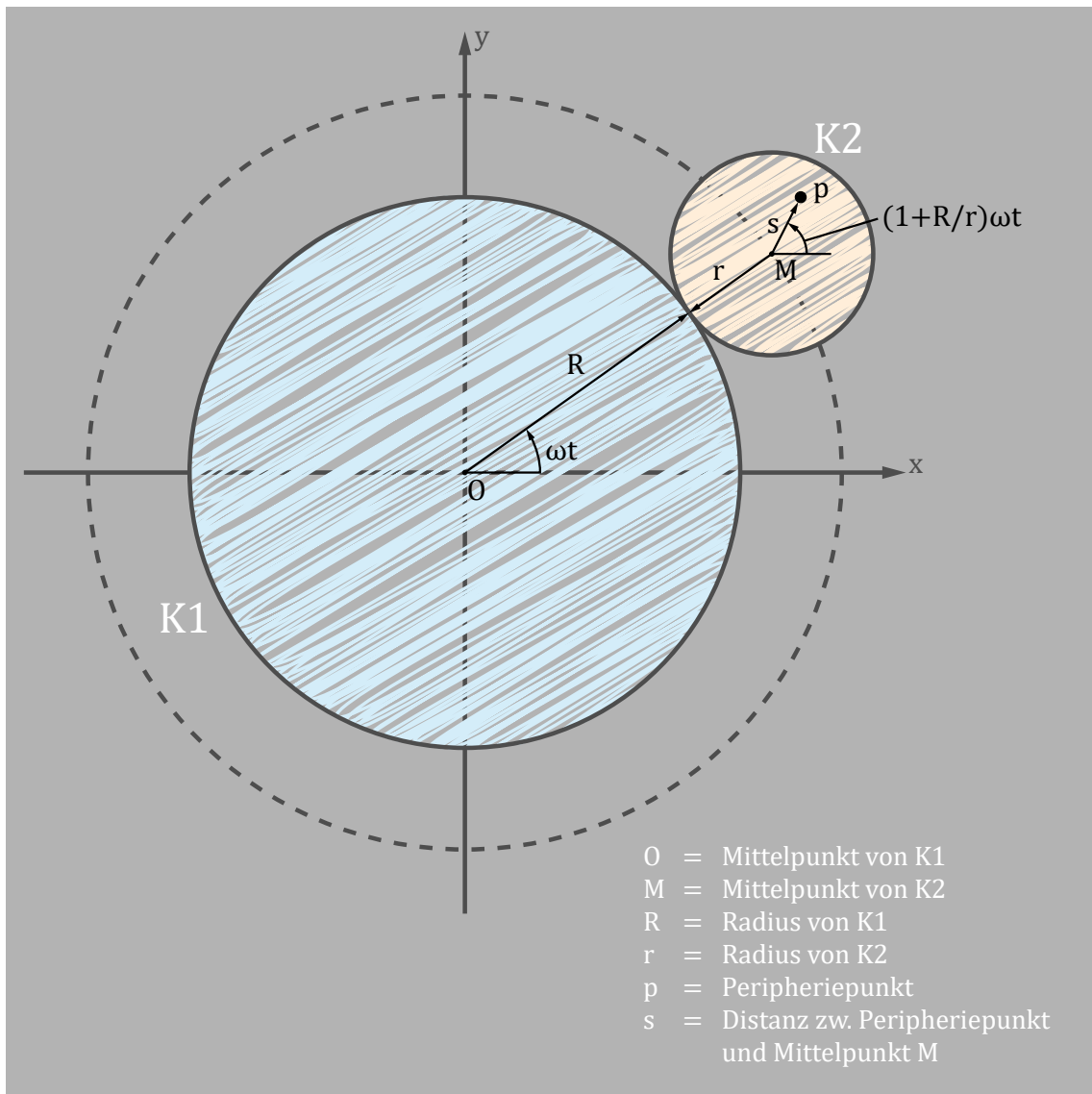


Abb. 35: Veranschaulichung der Funktionsweise des Spirographen

Abb. 35 bezieht sich auf die vorangegangene Formel und es wird deutlich, dass diese nicht die Darstellung eines Spielzeug-Spirographen abbildet. Der kleinere Kreis befindet sich hier außerhalb des größeren. Bei dem Spielzeug-Spirographen hingegen befindet sich der kleinere Kreis innerhalb des größeren Kreises.

Um die Formel möglichst an die Original-Funktionsweise des Spielzeuges anzugleichen, wurde sie für die vorliegende Arbeit ein wenig abgeändert. Anstelle der Addition der Radien des kleinen und des großen Kreises wird der Radius r vom Radius R subtrahiert. So erhält man die Formel, welche dem Spielzeug-Spirographen entspricht:

$$\vec{p}(t) = (R-r) \begin{pmatrix} \cos(\omega t + \tau) \\ \sin(\omega t + \tau) \end{pmatrix} + s \begin{pmatrix} \cos\left(\left(1 + \frac{R}{r}\right)\omega t + \theta\right) \\ \sin\left(\left(1 + \frac{R}{r}\right)\omega t + \theta\right) \end{pmatrix}$$

Die Darstellungsform ändert sich damit wie in Abb. 36 dargestellt:

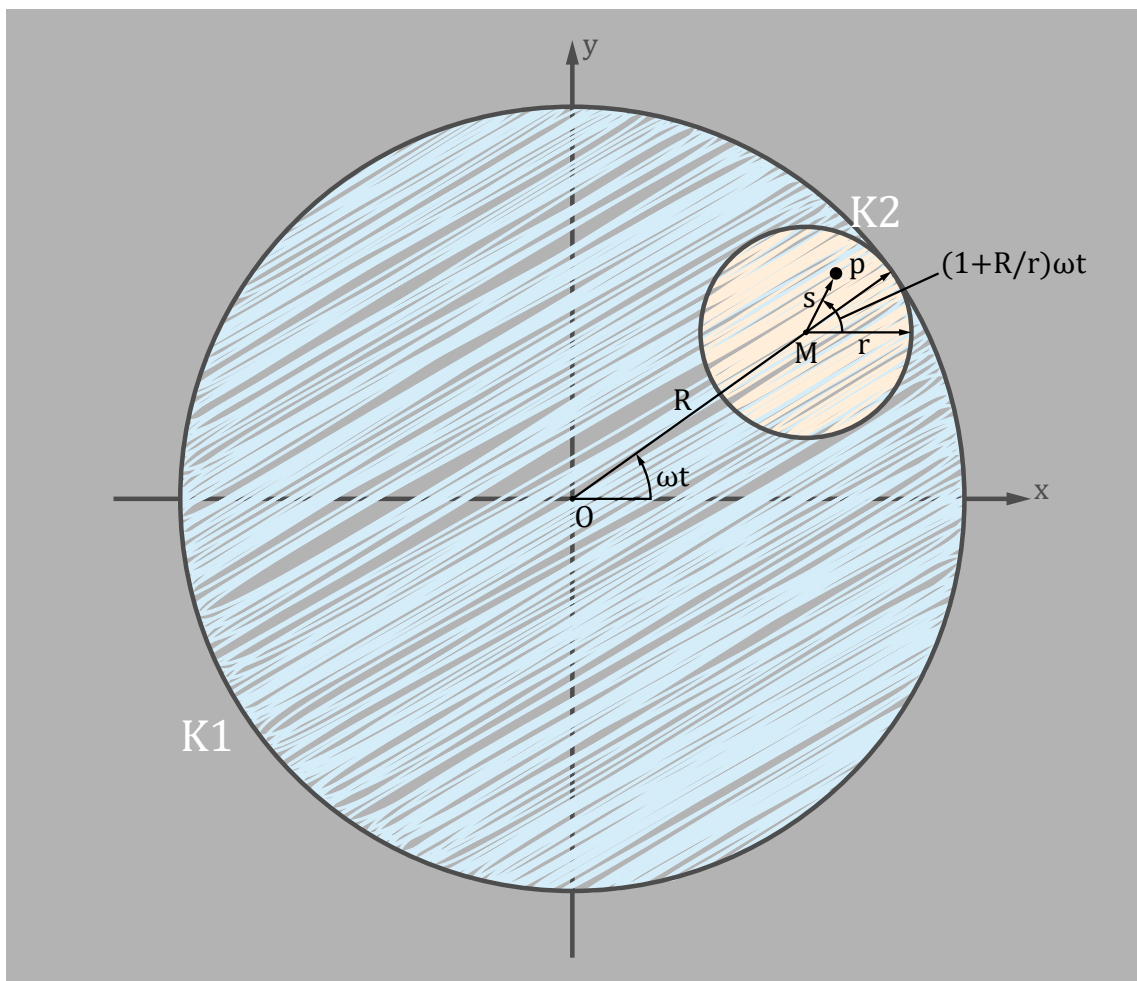


Abb. 36: Veranschaulichung der Funktionsweise des Spirographen (geändert)

Es ist zu beachten, dass bei einem echten Spielzeug-Spirographen die Distanz (s) des Loches (p) zum Mittelpunkt (M) des kleinen Kreises ($K2$), niemals größer sein kann als der Radius (r). Somit gilt: $|s| < r$. „Die Geschwindigkeit des Zeichenvorganges wird mit ω gesteuert und die Phasenwinkel τ , θ bestimmen die Position des Punktes zur Zeit t .“⁵²

4.3.3 Umsetzungsweg in Houdini

Um das Spirographenbild in Houdini auf einer zweidimensionalen Ebene darzustellen, muss die im vorangegangenen Kapitel 4.3.2 erstellte Formel aufgeteilt werden, sodass aus dem Vektor p zur Zeit t die Punkte p_x , auf der X-Achse, und p_y , auf der y-Achse, zur Zeit t werden. Dies vereinfacht die Implementierung der Formel in der jeweiligen Software und Programmiersprache. Aus diesen Überlegungen ergeben sich folgende Formeln:

$$p_x(t) = (R-r) \cos(\omega t + \tau) + s \cos\left(\left(1 + \frac{R}{r}\right) \omega t + \theta\right)$$

$$p_y(t) = (R-r) \sin(\omega t + \tau) + s \sin\left(\left(1 + \frac{R}{r}\right) \omega t + \theta\right)$$

Auf Basis des Tutorials von Ari Danesh⁵³ und der bisher erwähnten Theorie, wird im Folgenden der Aufbau einer möglichen Generierung der Kreisbewegung eines Spirographen erläutert und die Visualisierung eines darauf aufbauenden Bildes demonstriert.

Zuerst sollte überlegt werden, welche Parameter benötigt werden, um das Regelwerk des Spirographenbildes zu steuern. Aus der Formel werden folgende Variablen herausgenommen:

Formel	Bezeichnung	Parameter	Name (Variable)	Label
R	Radius von K1	float	fixed_radius	Fixed Radius
r	Radius von K2	float	movable_radius	Movable Radius
ω	Geschwindigkeit	float	speed	Speed
θ	Anfangswinkel zur Zeit 0	float	theta	Theta
τ	Anfangswinkel zur Zeit 0	float	tau	Tau
s	Distanz von M zu p	float	pen_offset	Pen_Offset

Zudem kann noch eine Variable hinzugefügt werden, die für eine spätere Skalierung des gesamten Bildes wichtig ist:

Skalierung	float	spiro_scale	Spiro Scale
------------	-------	-------------	-------------

⁵³ vgl. Danesh 2015.

Wird nun diese Überlegung als VEX-Code implementiert, ergibt sich folgender Aufbau:

```
@P.x=((ch(„fixed_radius“)-ch(„movable_radius“))*cos(ch(„speed“)*
@Time+ch(„tau“))+ch(„pen_offset“)*cos((1+ch(„fixed_radius“)/
ch(„movable_radius“))*ch(„speed“)*@Time+ch(„theta“)))*ch(„spiro_scale“);

@P.y=((ch(„fixed_radius“)-ch(„movable_radius“))*sin(ch(„speed“)*
@Time+ch(„tau“))+ch(„pen_offset“)*sin((1+ch(„fixed_radius“)/
ch(„movable_radius“))*ch(„speed“)*@Time+ch(„theta“)))*ch(„spiro_scale“);
```

Diese unübersichtliche Abbildung kann noch etwas strukturiert und vereinfacht werden. Des Weiteren wird die Variable für die Geschwindigkeitsteuerung (ω) in Bezug zu der Zeit (t) gesetzt.

Der vereinfachte Programmcode, welcher die Absicherung enthält, dass der Radius der Kreise nicht 0 wird, sieht dann wie folgt aus:

```
float R = ch(„fixed_radius“);
float r = ch(„movable_radius“);
float v = ch(„speed“)*@Time; // Omega*Zeit
float s = ch(„pen_offset“);
float tau = ch(„tau“); // Anfangswinkel zur Zeit 0
float theta = ch(„theta“); // Anfangswinkel zur Zeit 0
float scale = ch(„spiro_scale“); // Skalieren des Spirographen

if(r==0) r = 1; // Wenn movable_radius 0 werden sollte, wird es zu 1
if(R==0) R = 1; // Wenn fixed_radius 0 werden sollte, wird es zu 1

@P.x = ((R-r)*cos(v+tau)+s*cos((1+R/r)*v+theta))*scale; // X-Achse
@P.y = ((R-r)*sin(v+tau)+s*sin((1+R/r)*v+theta))*scale; // Y-Achse
```

Der dazugehörige Aufbau in Houdini wird in der folgenden Abb. 37 dargestellt:

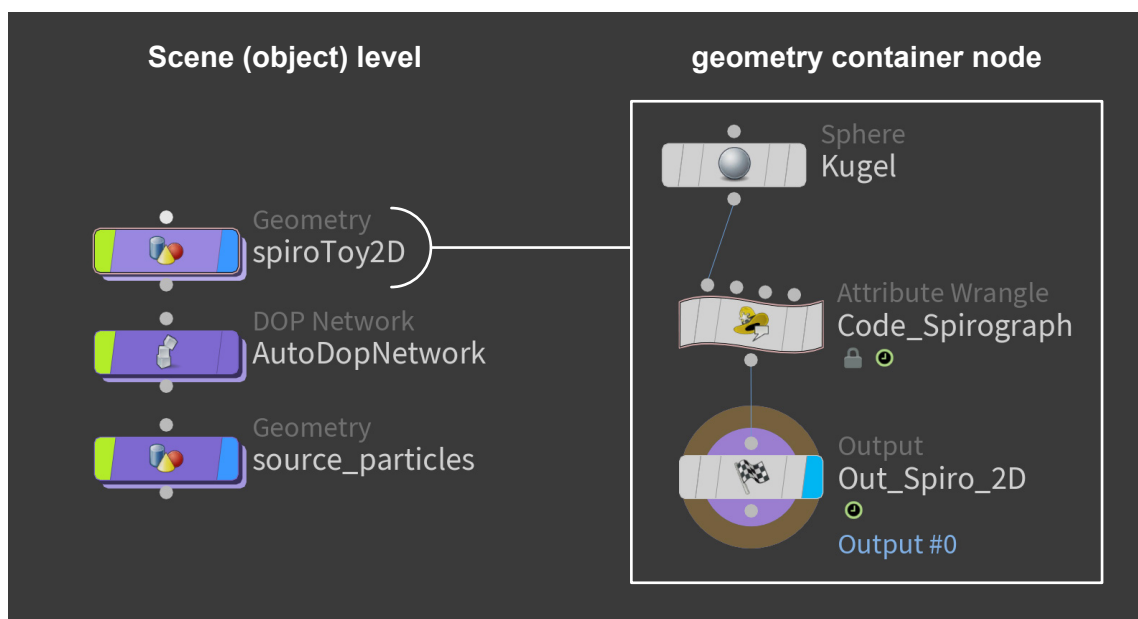


Abb. 37: Aufbau des Spirographen in Houdini

Innerhalb der Kugelgeometrie werden im *Attribute Wrangle* Node die vorher definierten Parameter angelegt und der VEX-Code verwendet. Dies bewirkt, dass die Kugel sich entsprechend des Peripheriepunktes bewegt.

Um eine visuelle Verdeutlichung zu erreichen und ein Spirographenbild zu erstellen, wird in der Szene eine Partikelsimulation auf die Kugel angewandt. Der zurückgelegte Weg der Kugel wird ersichtlich. Damit nicht

nur Punkte zu sehen sind, werden innerhalb der Geometrie der Partikelquelle die Punkte mit einem *add* Node (Polygons / By Group / Add:All points) verbunden und mit einem zusätzlichen *Attribute Wrangle* Node und VEX-Code die Stärke der Linien zwischen den Partikeln definiert. (vgl. Abb. 38)

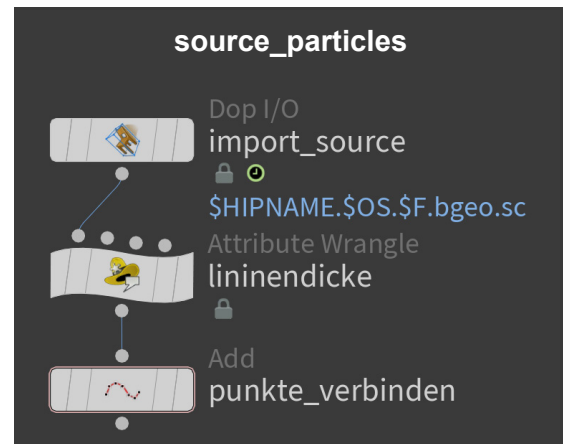


Abb. 38: Ausgabe des Partikelsystems

```
@width = ch(„size_part“); // Liniendicke
```

4.3.4 Resultat

Um zu gewährleisten, dass sich die Bewegung genau so verhält wie das Spielzeug, müssen bestimmte Anforderungen vorliegen:

$$\nabla 0 < r < R$$

$$\nabla |s| < r$$

Folgende Voreinstellung der Parameter wurden vorgenommen und ergeben das entsprechende Resultat (vgl. Abb. 39):

- ▢ Fixed Radius (R): 2.5
- ▢ Movable Radius (r): 1
- ▢ Pen Offset (s): 0.9
- ▢ Theta: 0.1
- ▢ Tau: 0.1
- ▢ Spiro Scale: 1
- ▢ Size Part: 0.1

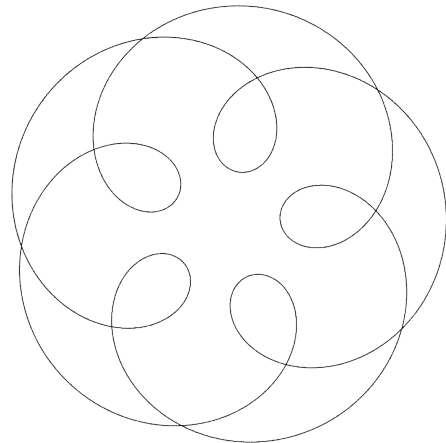


Abb. 39: Resultat des Spirographen

Das Testen weiterer Werte hat folgende Varianten als Resultat (vgl. Abb. 40):

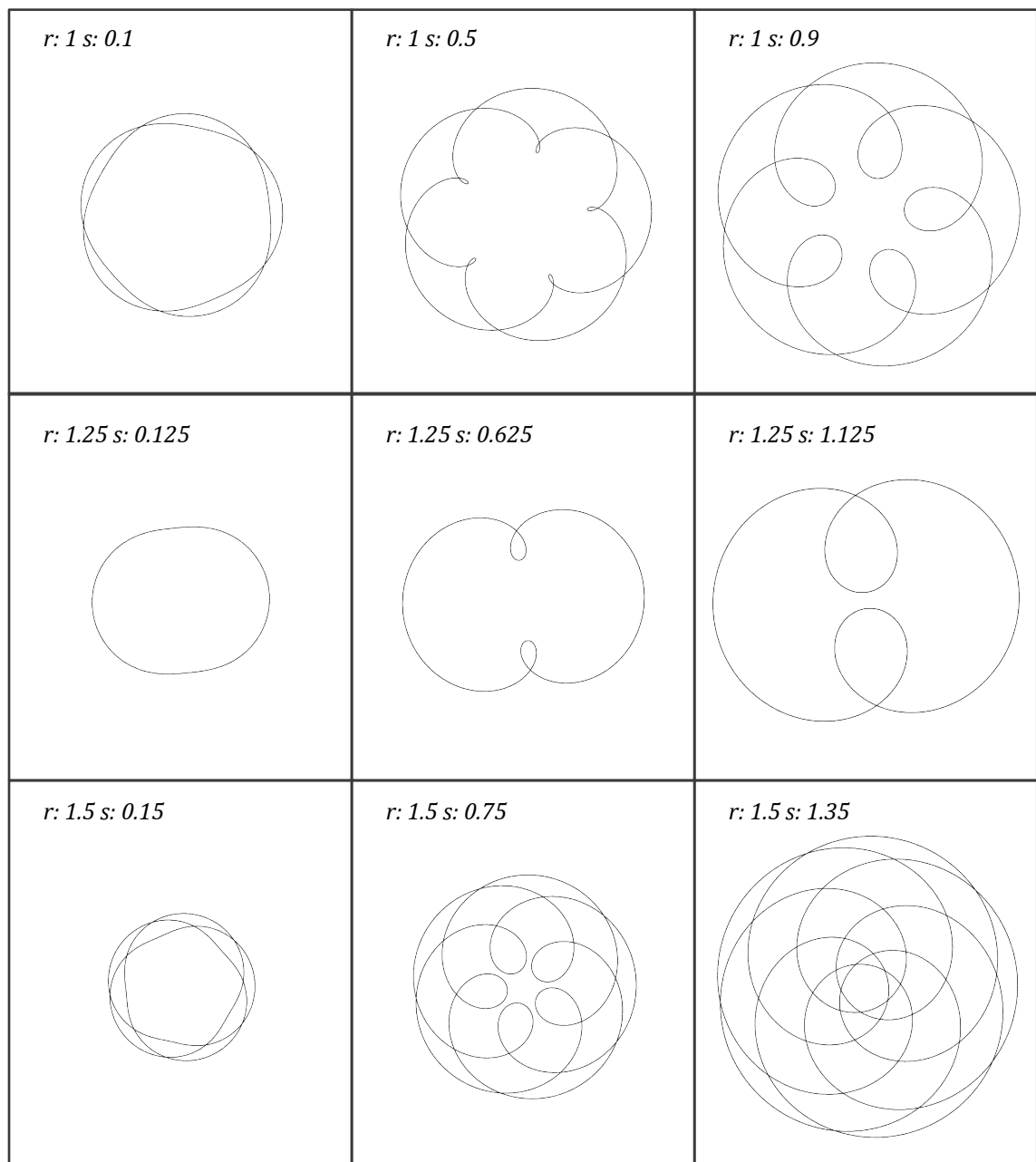


Abb. 40: Spirographvarianten

4.4 Zusammenführung

Im vorangegangenen Kapitel sollte zunächst durch die Beleuchtung des L-Systems (vgl. Kapitel 4.1) die Möglichkeit der Gestaltung mittels Fraktalen dargestellt werden. Dadurch wurde die Generierung von Bäumen näher erklärt. Das L-System bietet eine gute Methode, um Bäume und andere organische oder sich selbst ähnelnde Strukturen zu erstellen. Dennoch zieht dieser Umsetzungsweg einige Nachteile bei der Berücksichtigung externen Faktoren nach sich. Daher wurde der Space Colonization Algorithmus angeführt (vgl. Kapitel 4.2). Dieser ermöglicht auf relativ einfache Weise, die Umgebung des zu generierenden Objektes miteinzubeziehen und damit noch realer bzw. natürlicher wirkende Objekte zu erzeugen. Zuletzt wurde am Beispiel des Spirographen (vgl. Kapitel 4.3) die generative Gestaltung von Bewegung dargestellt. Dies veranschaulicht die Möglichkeit, Formen, Strukturen und Objekte als Resultat von Bewegungen zu generieren und stellt damit im Vergleich zu den anderen vorgestellten Methoden noch einen gänzlich anderen Ansatz generativer Gestaltung dar.

Die hier erläuterten Umsetzungswege mittels Houdini bilden nur einen kleinen Ausschnitt der Vielfältigkeit ab, welche generative Gestaltung in sich birgt. Es sind zahlreiche weitere Umsetzungswege, aber auch Variationen denkbar. Für den dargestellten Zusammenhang wurden speziell Umsetzungswege gewählt, die Wachstum und Bewegung darstellen und damit für visuelle Produktionen nutzbar sind. Insbesondere ist hier an den Bereich der Animation zu denken. Die Auswahl der vorgestellten Möglichkeiten sollte aber dem Anspruch Rechnung tragen, verschiedene Ansätze darzustellen und nicht nur auf die vorgegebenen Methoden in Houdini zurückzugreifen. Das L-System ist als ein derartiges, vordefiniertes Regelwerk seitens der Software zu betrachten und findet als solches Eingang in die geschilderten Umsetzungswege. Daneben ist der SCA wiederum eine Erzeugung durch Einsatz der Scriptsprache VEX. Zuletzt stellt der Spirograph eine Kombination aus VEX-Code und dem definierten Partikelsystems dar und kann somit als Kombinationsmöglichkeit verschiedener Ansätze betrachtet werden.

5. Durchführung

5.1 Lindenmayer System

5.1.1 Städte

Das L-System ist aufbauend auf den vorangegangenen Ausführung (vgl. Kapitel 4.1) und vielen weiteren Überlegungen in zahlreichen Bereichen einsetzbar. So ist zum Beispiel die Erstellung von Städten, Vierteln oder ganzen Straßenzügen mit Häusern möglich. Einsetzbar sind solche generativen Erstellungen beispielsweise in Filmproduktionen sowie in Spielen und der Architekturvisualisierung.

Um die Umsetzungsmöglichkeiten des L-Systems durch eine eigene Arbeit zu testen, verfolgt das nachstehende Beispiel das Ziel, mit einem L-System ein Straßenviertel zu generieren.

Das hierfür erstellte Regelwerk zeigt ein Wachstum des Systems, welches Straßenverläufen ähnelt. (vgl. Abb. 41)

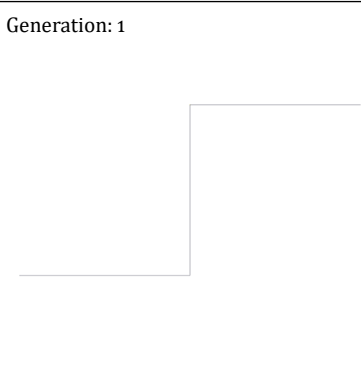
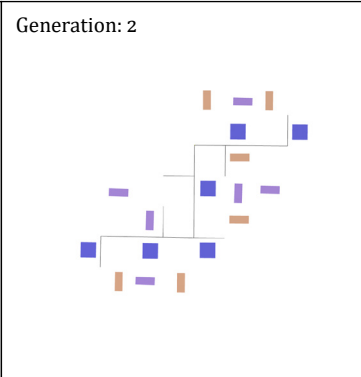
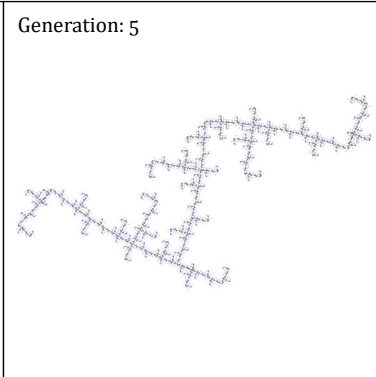
Stadtviertel	Winkel: 90 ° b: 0.5	Axiom: A	Regeln: A=[+F]F[-F] F=M-(b)F][-FK]FMF][+FK]
Generation: 1	Generation: 2	Generation: 5	
			
J,K,M (Leaf)Inputs für Geometrien		b Parameter (einstellbar)	

Abb. 41: Turtlegrafik eines Stadtviertels

Mit der Variable *b* werden die Linien in ihrer Rechtwinkligkeit unterbrochen und sorgen damit für eine dynamischere Entwicklung der einzelnen Straßen. Die Eingänge des L-Systems werden mit drei verschiedenen Box-Varianten verbunden und in ihrer Position angepasst. Die *Fuse Nodes* dienen dazu, Punkte, welche sich in einem bestimmten Radius befinden, zusammenzuführen. Mit zwei *Delete Nodes* werden die Linien und die Geometrien separiert. An den Linien wird eine angepasste Fläche mit dem *Sweep Node* verbunden. Mit diesem Node wird die Fläche mehrfach kopiert und der Linie entlang verlegt. Zum Schluss werden die Gebäude und die Straße mit dem *Merge Node* verbunden. (vgl. Abb. 42)

Dieses Beispiel zeigt nur einen Bruchteil dessen, was nötig ist, um eine ganze Stadt mit ihrer Variation an Gebäuden, Straßen, geschweige denn des Stadtbildes, wie Bäumen, Straßenlaternen, Parks usw. zu generieren. Aber dennoch bietet sich dadurch eine einfache Möglichkeit, um zum Beispiel bei Set Extensions Städte für den Bildhintergrund zu generieren. (vgl. Abb. 43)

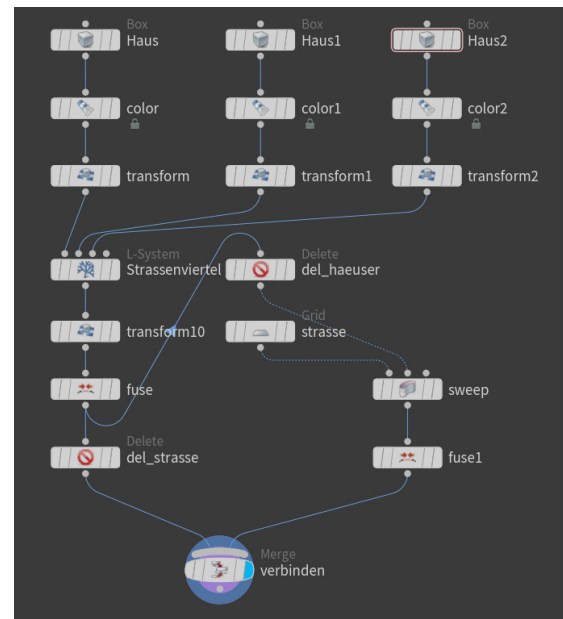


Abb. 42: Aufbau der Generierung des Stadtviertels

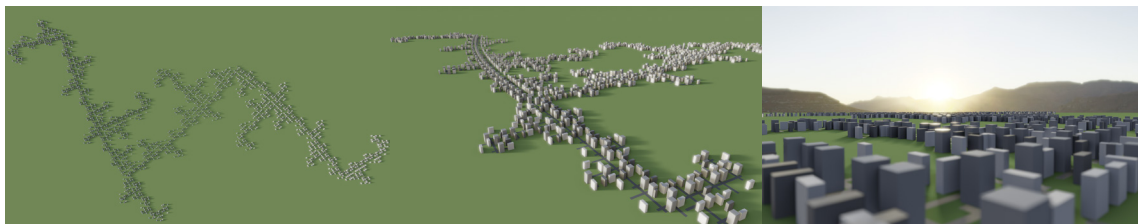


Abb. 43: Visualisierungen des Stadtviertels

Zu erwähnen ist hier die wissenschaftliche Ausarbeitung *Procedural Modeling of Cities* von Y. I. H. Parish, P. Müller aus dem Jahre 2001. Basierend auf dem L-System wurden dabei Autobahnen, Straßen und Gebäude generiert, indem Bildinformationen von Wasserkarten und Bevölkerungsdichte eingegeben wurden.⁵⁴

5.1.2 Organische Formen

Nicht nur Bäume haben Verzweigungen, diverse Büsche, aber auch Muster oder das Innenleben von Lebewesen weisen solch ein System des Selbstbildes auf. Daher kann das L-System auch auf diese Bereiche angewandt werden.

Das folgende Beispiel soll eine korallenähnliche Form darstellen, welche unter anderem für Unterwasserszenarien oder als fiktive Pflanze einer anderen Welt in der visuellen Medienproduktion verwendet werden kann.

⁵⁴ vgl. Müller / Parish 2001.

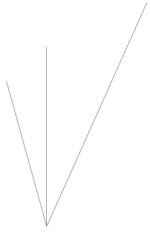
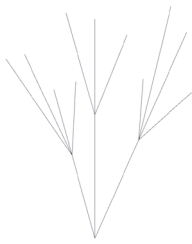
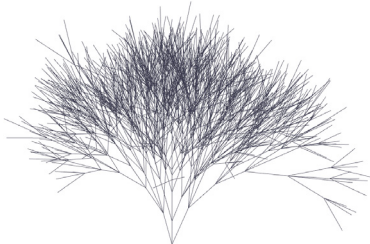
Korallenähnliche Form	Winkel: 23 °	Axiom: A	Regeln: A=[&THLA]/////[&THLA]/////[&THLA] F=S/////F S=FL H=H[H]
Generation: 1	Generation: 2	Generation: 6	
			
L Aktuelle Länge nimmt zu	H Halber Weg vorwärts	T Einfluss von Gravitation	

Abb. 44: Turtlegrafik einer Koralle/eines Baumes

Abgesehen von der Formgebung liegt die Herausforderung bei der Generierung der Koralle darin, dass sie nicht wie ein Baum oder Busch aussehen soll. Das System erweckt zunächst den Anschein eines Busches oder Baumes, da die Verzweigungen deutlich voneinander getrennt sind (vgl. Abb. 44). Nimmt man aber bspw. die Feuerkorallen zum Vorbild, so wird deutlich, dass die Äste zum Teil miteinander verwachsen sind und Löcher bilden. Für ein derartiges Aussehen bedarf es somit einer Weiterverarbeitung.

Mit einem *PolyWire* Node wird die Dicke der Äste bestimmt. Das *VDB from Polygons* Node wandelt die Geometrie in ein *volume primitive* aus *Voxels* um. Dies erlaubt, die Größe der *Voxels* einzustellen und somit eine Überschneidung des Volumens bzw. der Äste zu realisieren. Daraufhin wird die Form in ein Polygon mit dem Node *Convert VDB* umgewandelt. Da nun eine enorme Anzahl an Vertices vorhanden ist, wird diese mit einem *Remesh* Node reduziert. Das bringt auch den Effekt mit sich, dass das Geflecht organischer wirkt, da die Verzweigungen abgerundet werden. Mit dem *Smooth* Node werden die scharfen Kanten ein wenig geglättet und zum Schluss werden mit dem *Normal* Node die restlichen sichtbaren scharfen Kanten weicher dargestellt. (vgl. Abb. 45, Abb. 46 und Abb. 47)

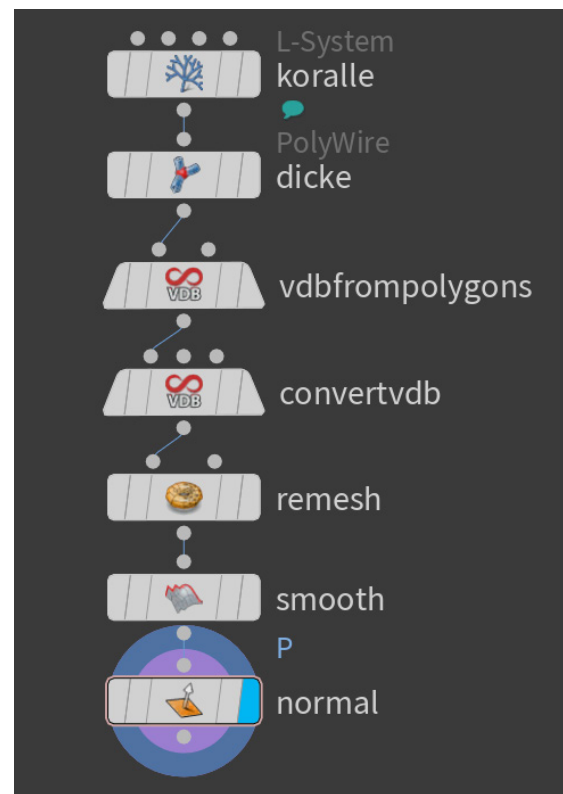


Abb. 45: Aufbau der Generierung einer Koralle

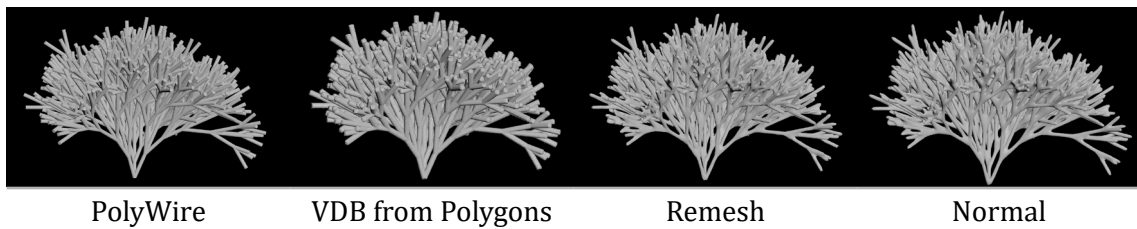


Abb. 46: Teilschritte des Prozesses der Geometrieänderung



Abb. 47: Shadingbeispiel der Koralle

Ändern sich die Werte der Gravitation, der Winkel, der Länge oder sogar die Regeln, werden verschiedenste Gebilde generiert.

5.1.3 Variation

Überdenkt man zunächst das Thema der Baumgestaltung noch einmal, so wird klar, dass durch das Verwenden von Kommandos und Variablen und der Regulierung der Parameter verschiedenste Bäume generierbar werden. Kommen dazu noch die Eingänge des L-Systems zum Einsatz, können zum Beispiel Blätter und Früchte an den angewiesenen Positionen angefügt werden. (vgl. Abb. 48)

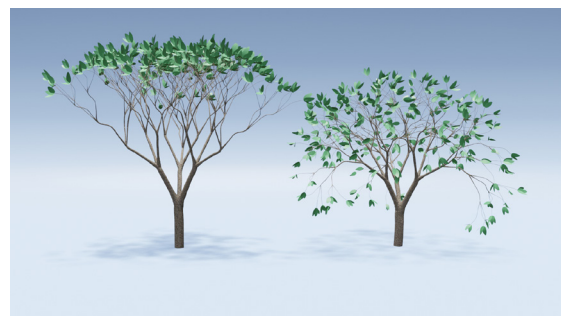


Abb. 48: Variationen durch Ändern der Gravitation

Neben der Generierung von Bäumen kann das L-System in vielen anderen Bereichen verwendet werden. So lassen sich unter anderem abstrakte und organische Geometrien prozedural erstellen (vgl. Abb. 49). Es kann bei Vegetationen, Visualisierung von Vernetzungen (vgl. Abb. 50) etc. zum Einsatz kommen. Außerdem wäre auch die Generierung von Blitzen für VFX Einsätze oder von



Abb. 49: Abstrakt und organisch

Labyrinthen für Game-Leveldesigns denkbar (vgl. Abb. 51). Die Möglichkeiten sind beinahe unbegrenzt und der Fantasie der Anwender und Anwenderinnen überlassen.

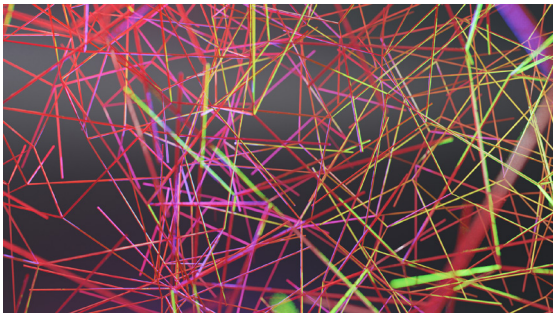


Abb. 50: Willkürliche Vernetzung



Abb. 51: Labyrinth

5.2 Wachstum mit dem Space Colonization Algorithmus

5.2.1 Anatomie

Im Folgenden sollen nun Möglichkeiten des Space Colonization Algorithmus abseits der Baumgestaltung getestet werden. Da diese Methode gezielte aber auch willkürlich gesetzte Punkte im Raum verbinden kann, ist es bspw. auch möglich, anatomische Strukturen, wie z.B. Blutgefäße zu visualisieren. Aufbauend auf den Darstellungen dieses Umsetzungswegs aus Kapitel 4.2 wird daher nun eine Möglichkeit der Erstellung von menschlichen Adern dargestellt (vgl. Abb. 52).

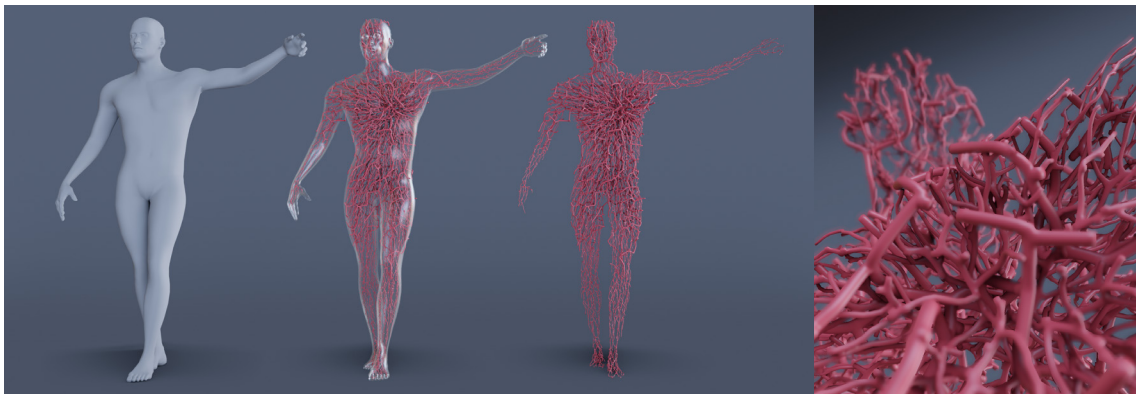


Abb. 52: Menschliches Adersystem

Der Algorithmus wird dafür so verwendet, dass zunächst dickere Adern ausgehend von der ungefähren Position des Herzens in die gesamte Geometrie eines menschlichen Körpers wachsen. Die dünneren Adern werden von der selben Position mit viel mehr Anziehungspunkten generiert und anschließend mit den dickeren Adern verbunden. (vgl. Abb. 53)

Variiert man nun die Anzahl der Punkte und die Parameter des SCAs, erhält man so unterschiedlichste Verzweigungen. Diese können auch für diverse weitere Visualisierungen und Objekte verwendet werden.

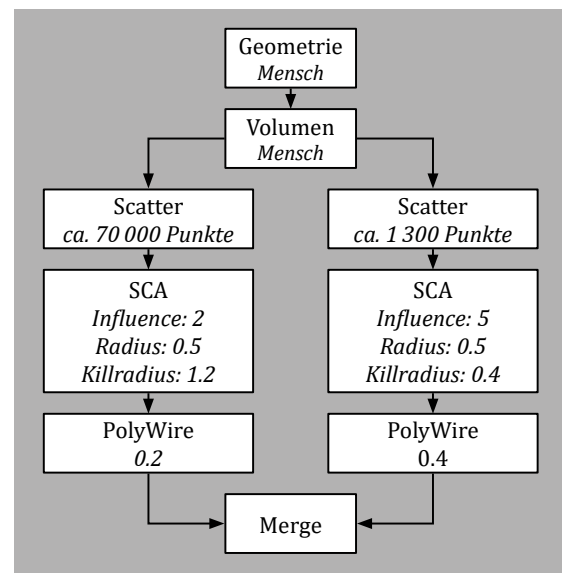


Abb. 53: Kombination zweier SCAs

5.2.2 Ranken

Das besondere an der Methode des SCAs ist, dass ein Volumen nicht zwingen notwendig ist, um ein Wachstum zu generieren. Für den Algorithmus werden lediglich Punkte benötigt

und diese können sich auch auf einer Oberfläche befinden. Das folgende Beispiel zeigt solch einen Unterschied. (vgl. Abb. 54)

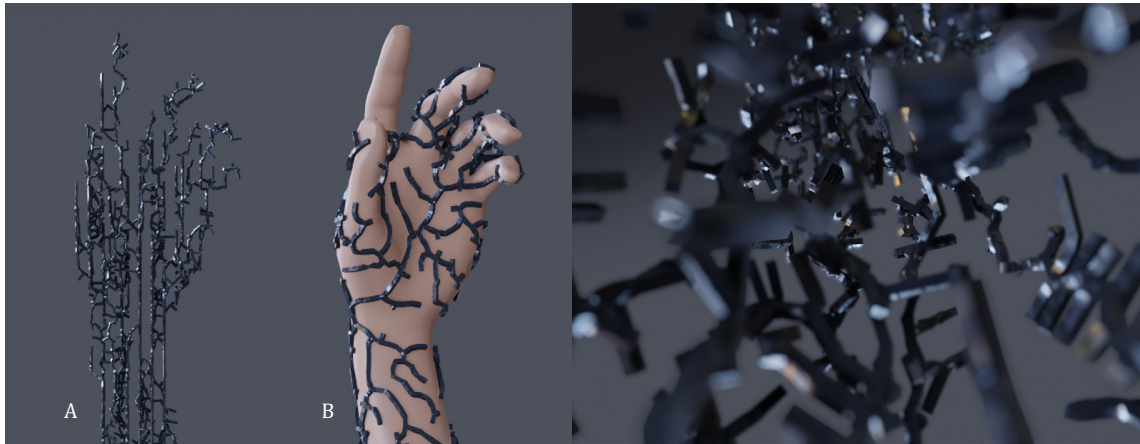


Abb. 54: Technisch wirkender SCA innerhalb und außerhalb einer Geometrie

Hier wird der Algorithmus so verwendet, dass die Ranken sich innerhalb einer Geometrie wie auch außerhalb auf deren Oberfläche ausbreiten. Zur Realisierung der linken Abbildung (Abb. 54-A), wird eine andere Art der Punktgenerierung innerhalb des Objekts verwendet, das *Points from Volume* Node. Diese Node setzt Punkte in gleichgroßen Abständen zueinander, was es ermöglicht, ein etwas technischer wirkendes Wachstum zu generieren. Die rechte Abbildung (Abb. 54-B), zeigt das Wachstum auf der Oberfläche des Objektes. Hier wird lediglich das *Scatter* Node verwendet, ohne zuvor ein Volumen zu erstellen. Dieses Node sorgt dafür, dass Punkte auf der Oberfläche erstellt werden.

Eine Kombination der beiden Methoden funktioniert ähnlich wie im vorherigen Beispiel, der Generierung von Adern, erklärt. Folgend wird ein Beispiel des SCAs gezeigt, welches sowohl Punkte innerhalb einer Geometrie, als auch auf deren Oberfläche berücksichtigt. Sogar die Überschneidung zwischen dem Geflecht und anderen Objekten ist durch *Boolesche Operatoren* möglich. (vgl. Abb. 55 und Abb. 56)

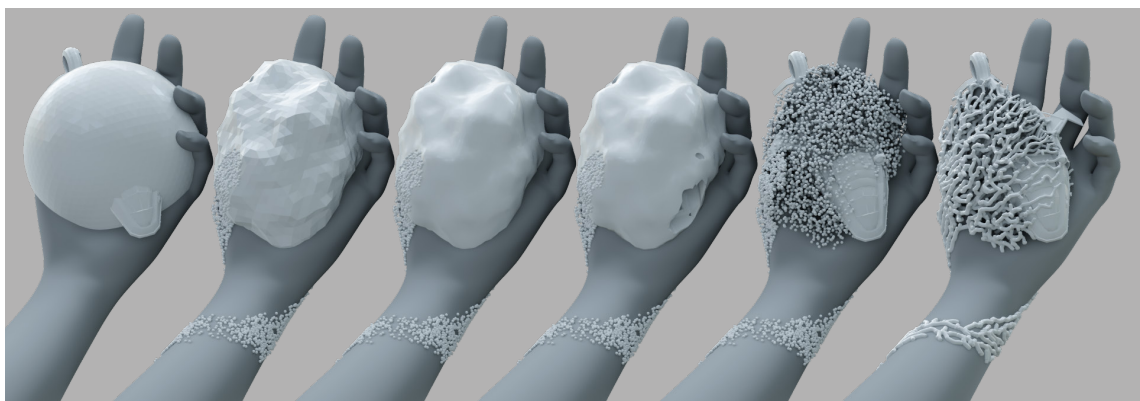


Abb. 55: Ablauf des Booleschen Operators und des SCAs

Zunächst muss die Geometrie festgelegt werden, in und auf welcher sich die Ranken ausbreiten können. Das hier dargestellte Beispiel (vgl. Abb. 57) basiert auf der Überlegung, ein Flugobjekt in einer Hand mit Ranken zu umschließen. Zudem sollen sich die Ranken dabei um den Unterarm wickeln. Für das Umschließen des Flugobjekts wird eine Kugel verwendet, welche durch ein *Mountain* Node deformiert wird, um eine natürliche Wirkung des Volumens für das Wachstum zu erzielen. Der Arm und die Kugel werden verbunden und anschließend werden die Anziehungspunkte händisch auf der Oberfläche des Unterarms und der Kugel mit dem *Spray Paint* Node erzeugt. Für die anderen Punkte innerhalb der Kugel mit Ausschluss des Flugobjektes und der Hand müssen zuerst die Geometrien des Objektes und der Hand durch *Boolean* Nodes von der Kugel subtrahiert werden. So entstehen innerhalb der Kugel Aussparungen der anderen Objekte. Daraufhin wird das *Points from Volume* Node mit rauschender Anordnung der Punkte verwendet. Zum Schluss werden die so erzeugten Punkte mit dem SCA zu Ranken generiert und mit dem Arm und Flugobjekt verbunden. (vgl. Abb. 56)

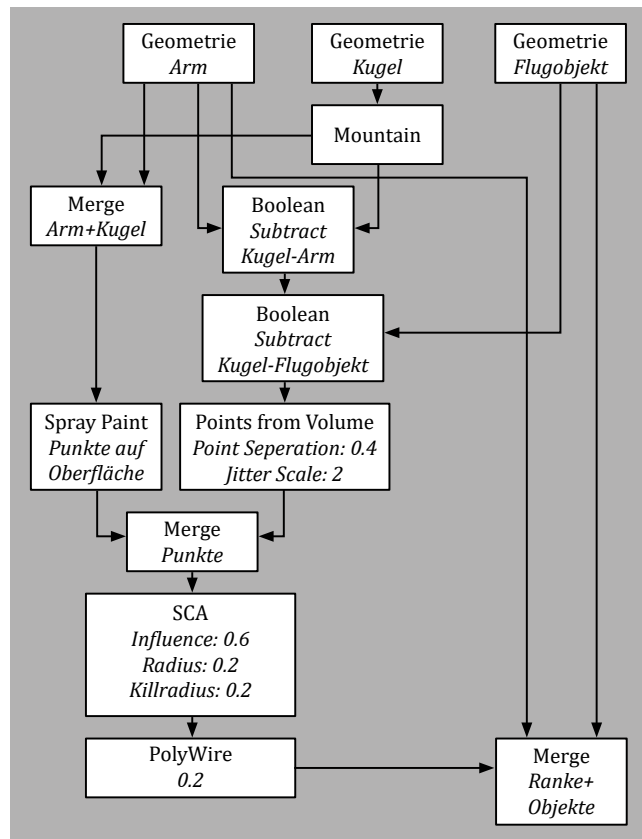


Abb. 56: Kombinierte Objekte mit anschließendem SCA

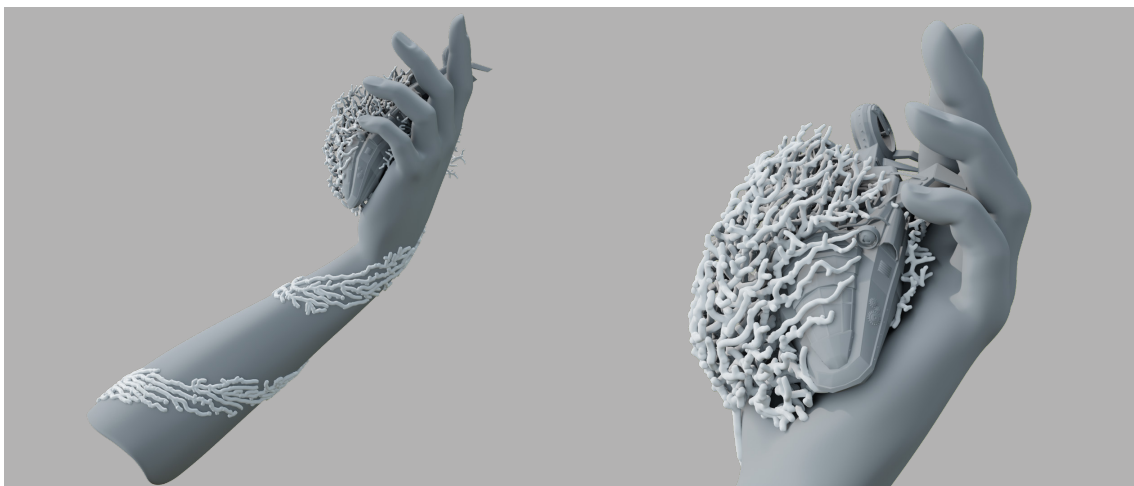


Abb. 57: Flugobjekt in einer Hand mit Ranken umschlossen

5.2.3 Variation

Sehr interessant ist diese Methode bei Visualisierungen, in welchen sich etwas aufbaut oder etwas statisches durch etwas organisches ersetzt wird (vgl. Abb. 58). Es können diverse andere Formen als Vorlage für dieses Wachstum verwendet werden. Es bestehen Kombinationsmöglichkeiten mit weiteren Effekten und Nodes, um noch interessantere Gebilde und Formen zu generieren. (vgl. Abb. 59 und Abb. 60)

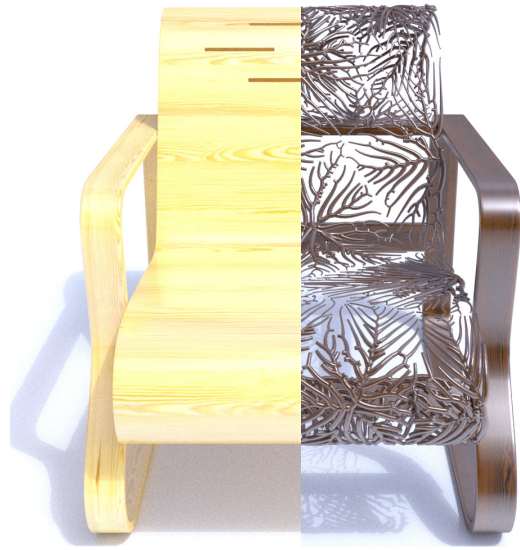


Abb. 58: Verändern eines bestehenden Objektes

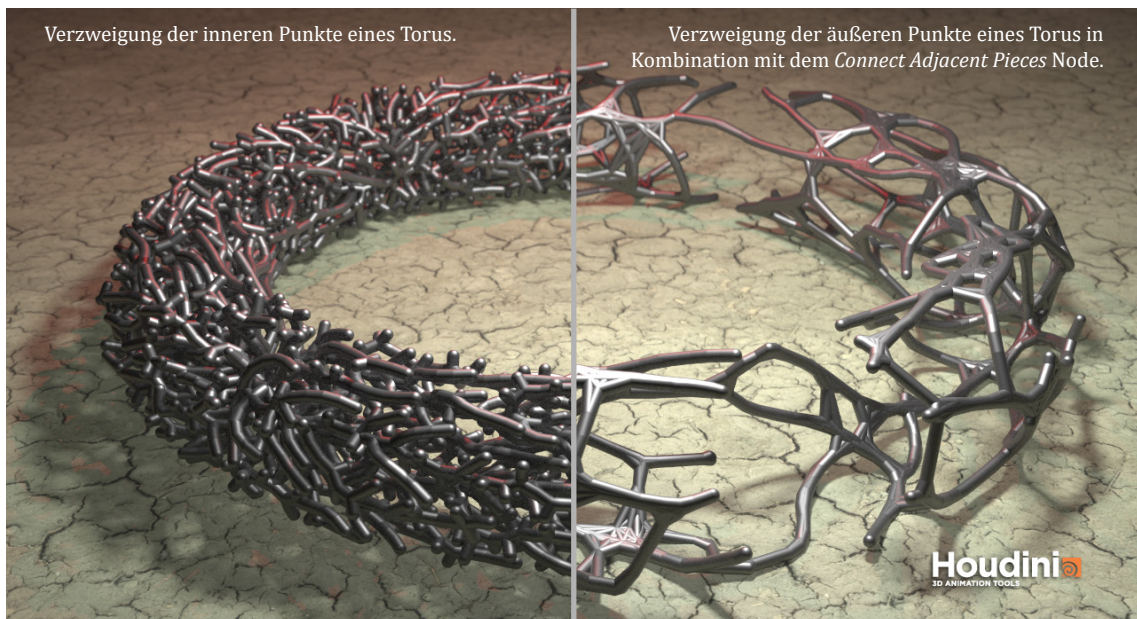


Abb. 59: Torus mit zwei Einsatzmöglichkeiten des SCAs

Mit dem SCA lassen sich bspw. auch Dschungelumgebungen, mit Lianen und Ranken um die Bäume herum sowie eben den Bäumen selbst, erzeugen. Aber auch abstrakte Aufbauten können generiert werden. Durch das Ändern der Anzahl der Anziehungspunkte, deren Anordnung sowie der Parameter des SCAs sehen die Geflechte stets unterschiedlich aus. Außerdem können biologisches Wachstum, Vernetzungen und vieles mehr umgesetzt und kombiniert werden.

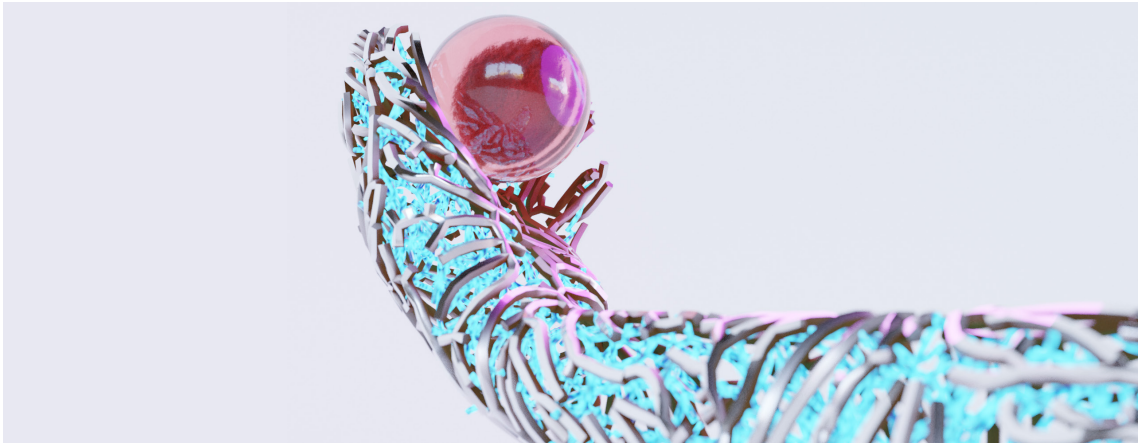


Abb. 60: SCA als Haut und Innenleben eines Arms

5.3 Generierte Bewegungen und Formen

5.3.1 Weiterentwicklung

Das Ziel dieses Unterkapitels ist es, zu zeigen, dass sich der Ansatz der generativen Bewegungserzeugung (vgl. Kapitel 4.3) weiter ausbauen lässt. Mit dem dargestellten Umsetzungsweg und der Verwendung des Partikelsystems in Houdini lassen sich neben der Form des Spirographen noch andere Formen erzeugen. Zudem kann die dritte Dimension mit einbezogen werden, sodass wiederum diverse weitere Formen entstehen.

Noch einen Schritt weitergedacht können mit der Implementierung des Umsetzungsweges aus Kapitel 4.3 diverse Muster generiert werden, welche praktisch nicht mehr mit dem Spielzeug-Spirographen umsetzbar wären. Dazu können die speziell für den Spirographen definierten Anforderung missachtet werden. Stattdessen könnte für den Radius des kleinen Kreises negative Zahlen oder größere Zahlen als Radius des größeren Kreises verwendet werden. Auch die Distanz des Peripheriepunktes kann beliebig verändert werden. Zudem können auch diverse Objekte als Partikel verwendet oder weitere Effekte mit dem Partikelsystem kombiniert werden.

Nachfolgend werden zwei Formeln verwendet, um darzustellen, wie das System mit weiteren Funktionen verwenden werden kann. Zudem ist es möglich ein Tool zu erstellen, mit welchem schnell und einfach unterschiedlichste Bewegungen von Mustern generiert werden können. Die zusätzlich verwendeten Formeln für dieses Beispiel sind die Superformel und der Attraktor, genauer der Dequan Li Attraktor.

Mithilfe des Partikelsystems entstehen Punkte, welche zu einem Muster verbunden werden. An die Stelle der Punkte können nun auch Objekte, also andere Geometrien wie bspw. Kugeln, Würfel oder sogar eigens modellierte Formen gesetzt und somit aneinandergereiht werden. Die so gesetzten Objekte lassen sich wiederum in ihrer Größe, Position und Rotation ändern. Zudem kann die Liniendicke und die Qualität eingestellt werden sowie der Verlauf dieser Dicke. Alle Parameter können zusätzlich animiert werden, so lässt sich selbst der Aufbau der Formen variieren. Die Eigenschaften eines derartigen Tools könnten auch als Digital Asset in Houdini verwendet werden. Dies ermöglicht eine immense, wenn nicht gar eine unendliche Varianz der so generierten Muster, Formen und Eindrücke. (vgl. Abb. 61)

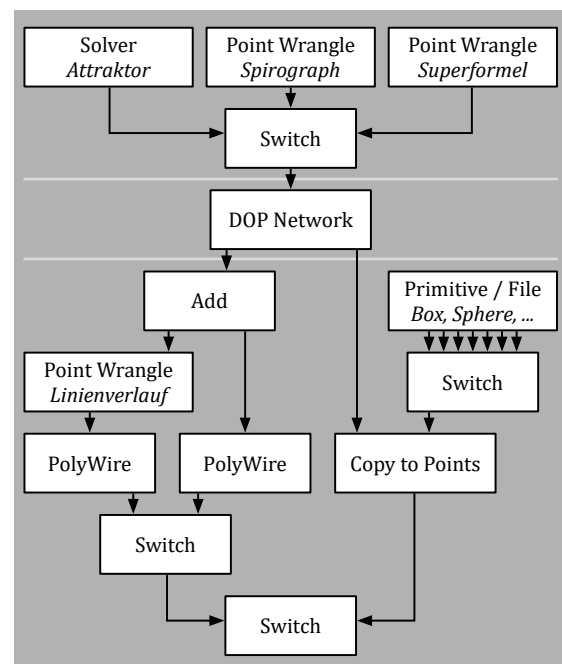


Abb. 61: Aufbau eines möglichen Tools

Die folgenden mathematischen Interpretationen des Spirographen, der Superformel bzw. des Super Shapes und eines Attraktors sollen in die Weiterentwicklung übernommen werden:

Um die Z-Achse bei einem Spirographen miteinzubeziehen und so eine dreidimensionale Generierung einer Kreisbewegung (vgl. Abb. 62) zu erzeugen, wurde die Implementierung aufbauend auf der Formel von Owenelton⁵⁵ ergänzt:

$$p_x(t) = (R-r) \cos(\omega t + \tau) + s \cos\left(\left(1 + \frac{R}{r}\right) \omega t + \theta\right)$$

$$p_y(t) = (R-r) \sin(\omega t + \tau) + s \sin\left(\left(1 + \frac{R}{r}\right) \omega t + \theta\right) \cos(a + \tau)$$

$$p_z(t) = (R-r) \sin(\omega t + \tau) + s \sin\left(\left(1 + \frac{R}{r}\right) \omega t + \theta\right) \sin(a + \tau)$$

Der zusätzliche Wert a gibt an, wie oft sich das System um die jeweilige Achse bewegt.

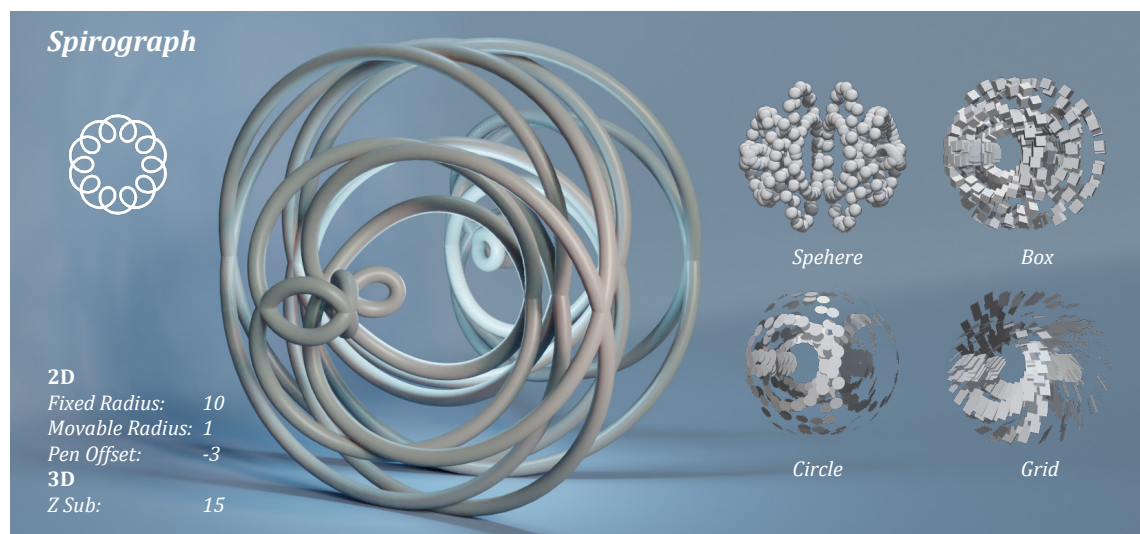


Abb. 62: Spirograph

⁵⁵ vgl. Owenelton 2012.

Die Superformel bzw. Super Shape ist eine weitere Methode, um diverse Formen in zwei und drei Dimensionen darzustellen (vgl. Abb. 63). Jede Änderung der Parameter sorgt für ein anderes Ergebnis. Die Implementierung dieser Formel sowie der Super Shapes bezieht sich auf die Überlegungen von Paul Bourke⁵⁶.

$$r(\varnothing) = \left[\left| \frac{1}{a} \cos(m\varnothing/4) \right|^{n_2} + \left| \frac{1}{a} \sin(m\varnothing/4) \right|^{n_3} \right]^{-1/n_1}$$

$$x = r_1(\theta) \cos(\theta) r_2(\varphi) \cos(\varphi)$$

$$y = r_1(\theta) \cos(\theta) r_2(\varphi) \sin(\varphi)$$

$$z = r_2(\varphi) \sin(\varphi)$$

$$\varphi: -\pi \leq \pi$$

$$\theta: -\pi/2 \leq \pi/2$$

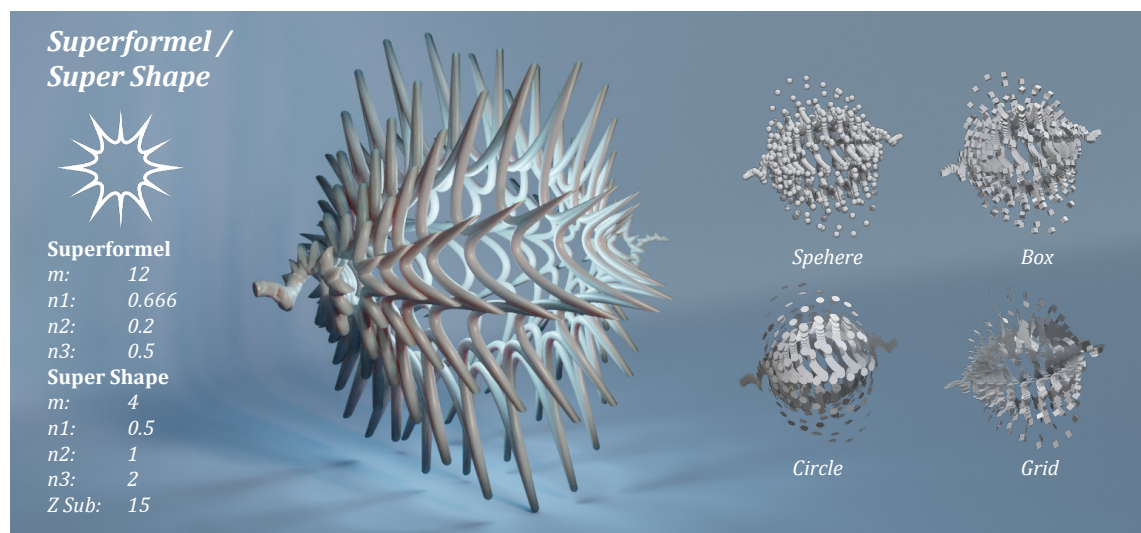


Abb. 63: Superformel/Super Shape

Eine weitere sehr interessante Formgenerierung eines dynamischen Systems ist der Attraktor (vgl. Abb. 64). Als Grundlage für die Umsetzung der Gleichung eines Attraktors wurden das Tutorial von Junichiro Horikawa⁵⁷ sowie die Kunstsammlung von Istvan⁵⁸ verwendet. In diesem Beispiel wird der Dequan Li Attraktor verwendet, dessen Gleichung sich wie folgt definieren lässt:

$$\frac{dx}{dt} = \alpha (y - x) + \delta xz$$

$$\frac{dx}{dt} = \rho x + \zeta y - xz$$

$$\frac{dx}{dt} = \beta z + xy - \varepsilon x^2$$

Wichtig ist hierbei, dass der Attraktor in einem *Solver* Node ablaufen muss, da die Position des zu generierenden Punkts in Bezug zu dem vorherigen Punkt steht. Zudem darf sich das Objekt, welches in seiner Bewegung durch den Code beeinflusst wird (vgl. Abb. 37), nicht auf der Position 0 befinden, da weiterführend keine Positionsänderung zustandekommen würde.

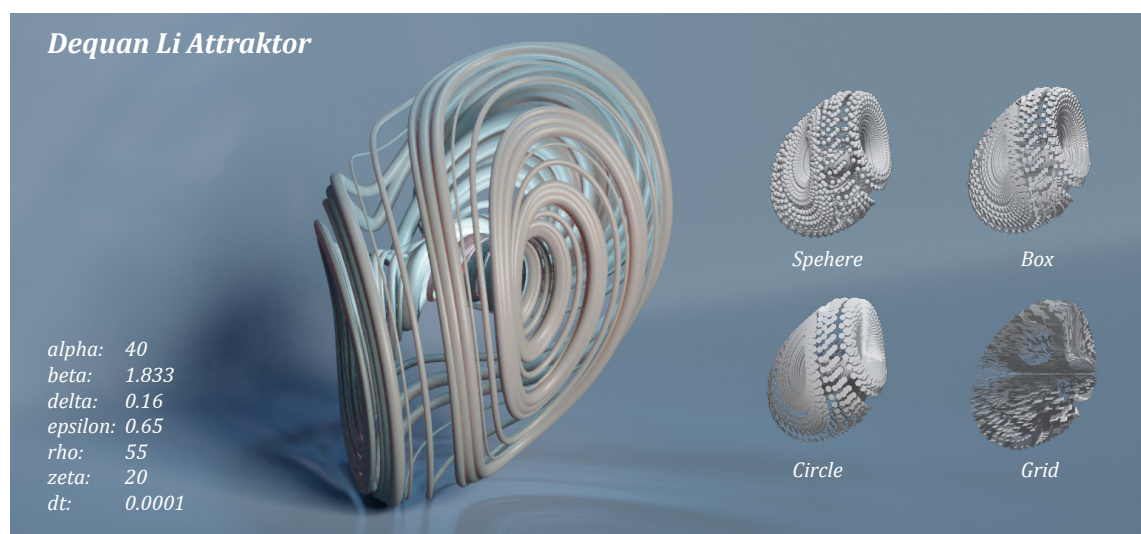


Abb. 64: Dequan Li Attraktor

Die Generierung weiterer Attraktoren ist durchaus möglich, es gibt unzählige Gleichungen, die in dieses Tool übernommen werden könnten.

Die Weiterführung und Erklärung zur Erstellung eines solchen Tools oder sogar eines Digital Assets würden an dieser Stelle den Rahmen sprengen. Es sei so viel gesagt, dass alle Variablen und Parameter an einer zentralen Stelle abgelegt werden könnten, um direkten Zugriff auf die verwendeten Formen und Variationen zu haben.

⁵⁷ vgl. Horikawa 2018.

⁵⁸ vgl. Istvan 2013.

5.3.2 Variation

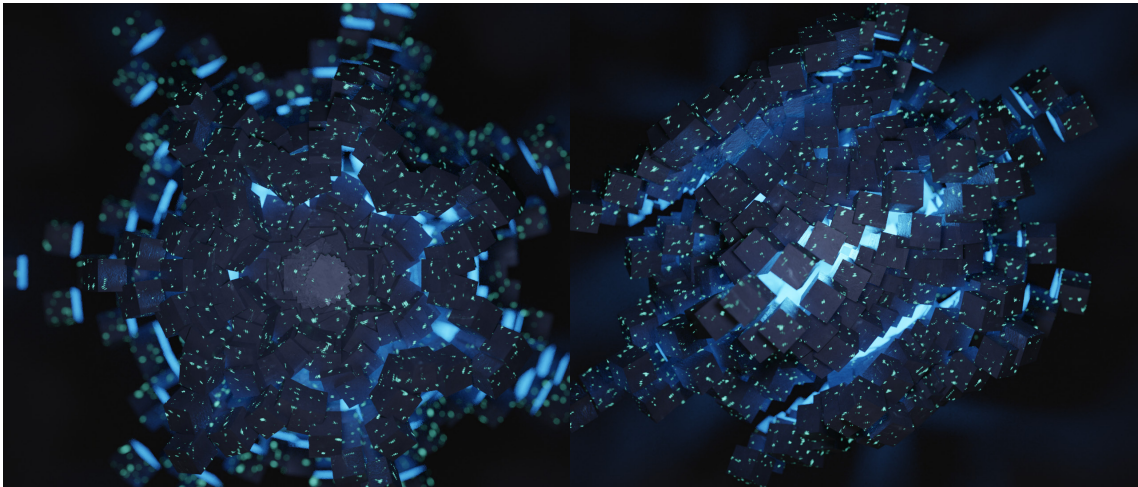


Abb. 65: Super Shape - Beispiel Stern

Den Variationsmöglichkeiten der Erstellung prozeduraler Formen und Bewegungen setzt letztlich nur die Kreativität eine Grenze. Einsetzbar sind die hier generierten Objekte bspw. für Elemente des Science Fiction-Genres. So wären überirdische Designs von Raumschiffen und deren Einzelteilen denkbar. Oder auch Galaxien könnten damit abgebildet werden. (vgl. Abb. 65 und Abb. 66)

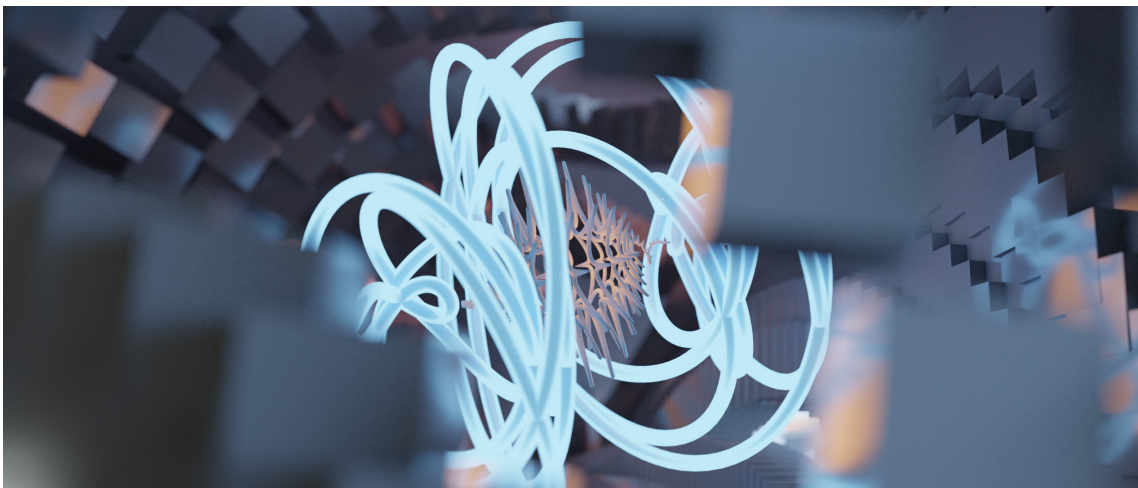


Abb. 66: Formbeispiel generierter Bewegung

Der Kunst- und Medienbereich profitiert von dieser Varianz und Kombinierbarkeit. Diese können auch Eingang in Architektur, Bildende Kunst, Mode, Produktdesign usw. finden. In der Bewegtbildproduktion sind die Animationen insbesondere für Motion Graphics einsetzbar.

5.4 Zusammenführung

Es ist möglich, jeden Ansatz der Methoden zur Realisierung fiktiver und organischer Formen noch weiter auszubauen und sogar den Einsatzbereich zu erweitern. Die oben vorgestellten Erweiterungen decken nur einen kleinen Bereich der prozeduralen Erzeugung von Visualisierungen in Houdini ab, zeigen aber dennoch die unterschiedlichen Wege auf, mit welchen es möglich ist, *nicht destruktiv* zu arbeiten. Jeder Schritt im Regelwerk kann beeinflusst und verändert werden, sei es in und mit dem VEX-Code oder der zu Verfügung stehenden Methode.

Durch die Umsetzung verschiedenster Objekte und Formen durch das L-System, den Space Colonization Algorithmus und der generierten Bewegung sollten die in Kapitel 4 vorgestellten Umsetzungswege durch eigene Arbeiten in ihrer Verwertbarkeit in der Praxis überprüft werden. Dabei wurde deutlich, dass die einzelnen Ansätze eine enorme Variationsbreite beinhalten.

So kann das L-System nicht nur für die Generierung organischer Formen verwendet werden. Es eignet sich auch gut für die Darstellung von allerlei Objekten, die dem Prinzip des Selbstbilds folgen. Mit relativ geringem Aufwand können komplexe Systeme gestaltet werden.

Der Space Colonization Algorithmus zeigt die Anwendung von VEX-Code und der Erstellung realistischer und natürlicher Systeme durch den Einsatz logischer Abläufe. Es können sowohl natürliche, als auch fiktive Formen generiert werden. Wachstum ist hierbei das zentrale Thema.

Die Möglichkeit, komplexe und prozedurale Formen durch generierte Bewegungen zu erstellen, erleichtern dem/der Kreativschaffenden die Arbeit enorm. Kreationen, welche durch händisches Animieren und Modellieren schier unmöglich sind, werden umsetzbar. Der Verwendungszweck solcher Formen ist zudem sehr breit gefächert.

Wie bereits erwähnt, dient dieses Kapitel zur Überprüfung der Einsetzbarkeit der drei vorgestellten Umsetzungswege und soll keinesfalls als abschließende Darstellung der Verwendungsmöglichkeiten verstanden werden. Je nachdem, welches Ziel verfolgt wird, bietet sich die eine oder die andere Methode an und es muss im Einzelfall abgewägt werden, welcher Umsetzungsweg gewählt wird. Da auch die im vorangegangenen Kapitel dargestellten Umsetzungswege bei Weitem nicht die einzigen Möglichkeiten generativer Gestaltung abdecken, sollten im Einzelfall auch andere, hier nicht berücksichtigte Methode hinzugezogen werden.

6. Fazit

Die vorliegende Arbeit verfolgt, aufbauend auf den zuvor geschilderten Teilabschnitten, das Ziel, mögliche Anwendungsbereiche und Umsetzungswege generativer Gestaltung vorzustellen und genauer zu beleuchten. Die praktische Durchführung mittels Houdini sollte die Anwendbarkeit und Effektivität verschiedener Methoden veranschaulichen. Der Schwerpunkt wurde dabei auf den Anwendungsbereich visueller Produktionen gelegt.

Rückblickend auf die Darstellung der Anwendungsbereiche (vgl. Kapitel 2) wird deutlich, dass generative Gestaltung in vielen Bereichen Zuspruch findet und eine moderne und intuitive Methode ist, um einzigartige Produkte und Effekte zu erstellen. Beeindruckend ist bspw. die Vorstellung, dass virtuell einzigartige Welten erstellbar sind, die in Filmen als Handlungsort dienen können. Aber die möglichen Anwendungsbereiche generativer Gestaltung erstrecken sich auf weitaus mehr, als auf die Abbildung unserer Welt. Prozedurale Erzeugnisse bringen eine Vielzahl neuer und noch nie dagewesener Gestaltungen zu Tage, sei es organisch mit Kurven und Ellipsen oder technisch mit Ecken und Kanten und dies in beinahe unzähligen Bereichen. Die Einsatzbereiche ermöglichen enorme Spielräume und beschränken sich bei Weitem nicht nur auf digitale Visualisierungen.

In Kapitel 3 wurde die Software Houdini in ihrem grundlegendem Ansatz erläutert. Dabei wurde deutlich, dass Houdini durch die nicht destruktive, prozedurale Arbeitsweise und eine Benutzeroberfläche, die eine Vielzahl an Einstellungen ermöglicht, besonders geeignet für die Umsetzung generativer Gestaltung ist. Daneben wurden noch weitere Software erwähnt, die sich für prozedurale Generierungen eignet.

Die Beschreibung von Umsetzungsweisen (vgl. Kapitel 4) berücksichtigt nur einen Bruchteil der Möglichkeiten, wie die generierte Gestaltung von Objekten bzw. Formen vonstatten gehen kann. Die Anzahl an wissenschaftlichen Fundierungen und die konkreten Herangehensweisen ist so enorm, dass es eine ganze Palette an möglichen Umsetzungsweisen gibt. Es bestehen keine einheitlichen Regeln der Erstellung, sondern viele Wege und Teilschritte. Als Gemeinsamkeit haben alle Umsetzungsweisen, so unterschiedlich sie in ihrer Ausgestaltung auch sein mögen, das Prinzip der Generierung durch Teilschritte. Die aufgezeigten Wege wurden speziell wegen ihrer unterschiedlichen Ausführbarkeit gewählt. Das L-System (vgl. Kapitel 4.1) wird in Houdini als eigenständiges Paket bereitgestellt, durch welches man schnell und einfach die sogenannten Turtle-Grafiken umsetzen kann. Die Verwendung des Space Colonization Algorithmus (vgl. Kapitel 4.2) sollte aufzeigen, dass durch den Einsatz von VEX-Code eine logische und wissenschaftliche Erkenntnis angewandt werden kann. Bei der generierten Bewegung (vgl. Kapitel 4.3) sollte die Kombinationsmöglichkeit von gegebenen Werkzeugen in Houdini und der Verwendung von VEX-Code dargestellt werden.

Durch das Weiterführen der Umsetzungsweisen eröffnen sich viele Möglichkeiten der Generierungen von Erzeugnissen und Einsatzzwecken, insbesondere in Bezug auf die Bewegtbildproduktion. Einige Ideen sollten in einer eigenen Durchführung (vgl. Kapitel 5) in die

Praxis umgesetzt werden. Dabei wurde auf die theoretischen Überlegungen aus Kapitel 4 aufgebaut. Die daraus gezogenen Erkenntnisse wurden auf eigene Arbeiten angewandt und im Einzelfall auch weiterentwickelt. Die dabei entstandenen Ergebnisse können bspw. für Set Extensions und Visual Effects verwendet werden.

In Bezug auf die Wahl der Software Houdini lässt sich zusammenfassen, dass diese, wie auch andere nodebasierte Systeme, sich ideal für eine logische und einfache Vorstellung von prozeduralen Workflows eignen. Bei der ebenenbasierten Arbeitsweise, wie sie bspw. in AfterEffects oder auch Blender verwendet werden, sind destruktive Eingriffe auf einzelne Prozesse kaum zu vermeiden oder nur komplex, evtl. durch Verschachtelungen, zu umgehen. Nodes können mehrere Eingänge haben, sodass verschiedene vorangegangene Prozessstränge berücksichtigt und kombiniert werden können, was das Arbeiten mit umfassenden und verflochtenen Prozessen vereinfacht.

Wie bereits erwähnt, wurden die vorgestellten Umsetzungswege nach ihren unterschiedlichen Ansätzen ausgewählt. Daneben gibt es noch zahlreiche weitere Umsetzungswege, welche in weiterführenden Arbeiten genauer beleuchtet werden könnten. Für die hier beschriebenen Ansätze kann noch erwähnt werden, dass die generierten Formen des Space Colonization Algorithmus sowie die der Spirographen, der Superformel und des Attraktors nicht nur in Bezug auf die Zeit sondern auch durch Iterationen innerhalb des Systems erstellt werden können, ähnlich wie bei den L-Systemen. Das würde auf *Knopfdruck* die Geometrie erzeugen. Dadurch eröffnen sich weitere Anwendungsmöglichkeiten. Wie im gesamten Bereich generativer Gestaltung geht es also darum, zu entscheiden, welche Methode sich für das angestrebte Ziel gut eignet. Ggf. können auch mehrerer Methoden getestet werden, um die passendste herauszufinden. Auch eine Kombination verschiedener Umsetzungswege ist denkbar.

Die Durchführung der vorgestellten Umsetzungswege zeigt nur einen Ausschnitt der Einsatzmöglichkeiten, die mittels der genannten Methoden umsetzbar wären, auf. So könnte noch viel mehr auf die Interaktivität der Prozesse innerhalb eines Regelwerks eingegangen werden. Wie könnte bspw. eine Beeinflussung anderer Systeme oder Objekte aussehen? Die vorgestellten Beispiele eignen sich zum Teil hervorragend für Animationen, der Veranschaulichung von Wachstum oder des Aufbaus verschiedener Strukturen. Auch weitere Anwendungsbereiche sind denkbar. Daher sollen diese Durchführungen keinesfalls als abschließende Darstellung betrachtet werden, sondern dienen lediglich dazu, einen ersten Eindruck des Machbaren zu vermitteln.

Abschließend soll aufbauend auf den angeführten Erkenntnissen eine umfassende Beschreibung generativer Gestaltung, die alle Eigenschaften benennt, die in den vorangegangenen Ausführungen als relevant erscheinen, nochmals zusammengefasst werden. Generative Gestaltung ist zunächst einmal als die Erstellung und Formgebung eines Produkts oder einer Visualisierung mit künstlerischen Aspekten zu verstehen. Die Abgrenzung zu anderen Methoden visueller Produktionen liegt insbesondere darin, dass vordefinierte oder

zufällig gewählte Parameter in prozesstechnischen Teilschritten ein nicht destruktives Ergebnis erzeugen. Der/die Kreativschaffende bestimmt die Rahmenbedingungen des Erzeugnisses, greift aber nicht direkt in den Entstehungsprozess ein. Computergestützte Funktionen, Algorithmen und gesetzte Bedingungen fertigen aufbauend auf den vorangegangenen Prozessen das jeweilige Resultat.

Dabei ist die Umsetzung prozeduraler Erzeugnisse nicht so einfach, wie es zunächst erscheinen mag. Sollen diverse Einflussmöglichkeiten auf das Erstellen generativer Formen möglich sein, so ist das eben nicht mit einem *Knopfdruck* getan. Das System kann zwar so aufgebaut werden, dass eine Form mit geringem Aufwand erstellt wird, auf diese Weise wird allerdings auf die Möglichkeit der Varianz und Einflussnahme verzichtet. Am Beispiel der L-Systeme ist zu sehen, dass ein Umsetzungssystem zwar schon vordefiniert gegeben ist, es allerdings dennoch einiger Überlegungen bedarf, welche Regeln verwendet werden sollen. Umso komplexer das Ergebnis sein soll, desto umfangreicher werden die Regeln. Zudem sind viele Parameter vorhanden, die das Regelwerk beeinflussen. So ist vielleicht die eigentliche Generierung mit einem Knopfdruck getan, es sind aber trotzdem eigene Vorüberlegungen und ggf. ein weiteres Justieren des Resultats notwendig.

IV. Literaturverzeichnis

[Über generative Gestaltung] (o.J.). Online unter:

<http://www.generative-gestaltung.de/1/> [Stand 02.09.2018].

Abelson, Harold / diSessa, Andrea(1986). *Turtle Geometry: The Computer as a Medium for Exploring Mathematics*, Cambridge / London: The MIT Press.

Alderton, Matt (2017). *Drei Schlüssel zur erfolgreichen Zukunft von BIM bei Bauprojekten*, Redshift by AUTODESK (Hrsg.), online unter: <https://www.autodesk.de/redshift/zukunft-von-bim/> [Stand 05.08.2018].

Autodesk, Inc. (2018a). *Autodesk (Homepage)*, online unter: <https://www.autodesk.de/> [Stand 03.09.2018].

Autodesk, Inc. (2018b). *Fusion 360*, online unter: <https://www.autodesk.de/products/fusion-360/overview> [Stand 03.09.2018].

Autodesk, Inc. (o.J.). *Die Zukunft des Flugreisens neu definiert*, online unter: <https://www.autodesk.de/customer-stories/airbus> [Stand 30.07.2018].

Behaneck, Marian (2011). *Generatives Design – Das Unplanbare planen*, online unter: <https://www.bm-online.de/wissen/edv/das-unplanbare-planen/#slider-intro-4> [Stand 05.08.2018].

Blender Foundation (o.J.). *Blender (Webseite)*, online unter: <https://www.blender.org/> [Stand 03.09.2018].

Bourke, Paul (2002). *Supershapes (Superformula)*, online unter: <http://paulbourke.net/geometry/supershape/> [Stand 03.09.2018].

Breuninger, Jannis / Gottinger, Ferdinand (2011). *Additive Fertigung – Bessere Beinprothesen generativ gefertigt*, online unter: <https://www.scope-online.de/additive-fertigung/neues-interessantes---bessere-beinprothesen--generativ-gefertigt-.htm> [Stand 04.08.2018].

Brown, PJ (2016). *Under Armour bringt mit generativer Gestaltung und 3D-gedruckten Schuhen frischen Wind in die Branche*, Redshift by AUTODESK (Hrsg.), online unter: <https://www.autodesk.de/redshift/3d-gedruckte-schuhe-%e2%80%8e/> [Stand 04.08.2018].

Danesh, Ari (2012). *BONUS LESSON / L-SYSTEMS*, Side Effects Inc. (Hrsg.), online unter: <https://www.sidefx.com/tutorials/bonus-lesson-l-systems/> [Stand 03.09.2018].

- Danesh, Ari (2015). *PROJECT SPIRO TOY*, Side Effects Inc. (Hrsg.), online unter: <https://https://www.sidefx.com/tutorials/project-spiro-toy/> [Stand 03.09.2018].
- DATAKOM Buchverlag GmbH (2011). *Prozedurale Programmierung*, online unter: <https://www.itwissen.info/Prozedurale-Programmierung-procedural-programming.html> [Stand 07.05.2018].
- DATAKOM Buchverlag GmbH (2018). *Computergrafik*, online unter: <https://www.itwissen.info/Computergrafik-computer-graphics-CG.html> [Stand 07.05.2018].
- Dertinger, Heiko (2015). *Generative Design*, online unter: <http://www.digitalwiki.de/generative-design/> [Stand 08.03.2018].
- Fischer, Thomas / Herr, Christiane M. (2002). *Teaching Generative Design*, online unter: http://www.generativeart.com/on/cic/ga2001_pdf/fischer.pdf [Stand 26.08.2018].
- Flake, Gray William (1998). *The Computational Beauty of Nature. Computer Explorations of Fractals, Chaos, Complex Systems, and Adaption*, Cambridge / London: The MIT Press.
- Fry, Ben / Reas, Casey (o.J.). *Processing (Webseite)*, online unter: <https://processing.org/> [Stand 03.09.2018].
- Galanter, Philip (2003). *What is generative art? Complexity theory as a context for art theory*, in: Soddu, Celestino (2003). GA 2003 – proceedings of 6th international conference GENERATIVE ART, Milan, 10-13 December 2003, Milan: Alea Design Publisher, online unter: http://www.artscience-ebookshop.com/ebooks_free/GA2003ebook.pdf [Stand 03.08.2018].
- Generative Kunst – Kunst Element Technologie* (o.J.), online unter: <https://hisour.com/de/generative-art-21197/> [Stand 03.08.2018].
- Häberle, Janine (2013). *The Meticulous Way of Design Thinking*, in: *Proceedings of the 1st International Conference on Digital Fashion*, 16-18 May 2013, London, Online unter URL: http://janinehaeberle.ch/wp-janine/wp-content/uploads/2013/12/Paper_Haerberle-HSLU_DigitalFashionConference_2013.pdf [Stand 04.08.2018].
- Häberle, Janine (2011). *Einzigartigkeit in Serie. Individuelle Bekleidung durch generatives Textildesign*. Theoretische Bachelorarbeit, Hochschule Luzern – Design & Kunst. Online unter: http://janinehaeberle.ch/wp-janine/wp-content/uploads/2013/12/Thesis_JH_EinzigartigkeitinSerie_WEB.pdf [Stand 24.07.2018].
- Hello Games No Man's Sky (o.J.). *Press Kit*, online unter: <https://www.nomanssky.com/press/> [Stand 26.08.2018].

- Horikawa, Junichiro (2018). *[Houdini Tutorial] 0034 Strange Attractor (Slow version)*, online unter: <https://www.youtube.com/watch?v=saA6-edb-OE> [Stand 03.09.2018].
- Hornby, Albert Sidney (1995a). *Generate*, in: Crowther, Jonathan (1995). *Oxford Advanced Learner's Dictionary of Current English*, Oxford: Oxford University Press.
- Hornby, Albert Sidney (1995b). *Generative*, in: Crowther, Jonathan (1995). *Oxford Advanced Learner's Dictionary of Current English*, Oxford: Oxford University Press.
- Istvan (2013). *Math:Rules, Strange Attractors*, online unter: <https://chaoticatmospheres.com/mathrules-strange-attractors> [Stand 03.09.2018].
- Jung, Richard (2018). *Was ist eigentlich Motion Design?*, online unter: <https://page-online.de/branche-karriere/was-ist-eigentlich-motion-design/> [Stand 24.07.2018].
- Junghans, Christian (o.J.). *Mathematische Überraschungen in der Natur*, online unter: <https://www.was-darwin-nicht-wusste.de/wunder/mathematische-ueberraschungen.html> [Stand 30.07.2018].
- Kizny, Patryk (o.J.). *Generative Design – Opening the Pandora's Box*, online unter: <http://kizny.com/know-how/introduction-to-generative-design/> [Stand 03.08.2018]
- König, Reinhard (2012), 1. *Projekteinführung*, in: Donath, Dirk; König, Reinhard; Petzold, Frank (2012). *KREMLAS: Entwicklung einer kreativen evolutionären Entwurfsmethode für Layoutprobleme in Architektur und Städtebau*, Weimar: Verlag der Bauhaus-Universität Weimar.
- Kretzer, Manuel (2016). *Processing - Generative Design Tutorial instructions for the creation of computational art - Sound Mapping*, online unter: http://responsivedesign.de/wp-content/uploads/2016/05/tutorial-06_processing-soundmapping2.pdf [Stand 26.08.2018].
- Lindblad, Mason (2018). *Houdini Overview: VFX and 3D Modeling Software*, Online unter: <https://filtergrade.com/houdini-overview-vfx-3d-modeling-software/> [Stand 31.08.2018].
- Lindenmayer, Aristid / Prusinkiewicz, Przemyslaw (2004). *The Algorithmic Beauty of Plants. The Virtual Laboratory, [PDF]*, New York, Springer Verlag, online unter: algorithmicbotany.org <<http://algorithmicbotany.org/papers/abop/abop.pdf>> [Stand 07.08.2018].
- Merkle, Manuel Casasola (2017). *VEX in Houdini: Space Colonization*, Entagma (Hrsg.), online unter: <http://www.entagma.com/1028-2/> [Stand 03.09.2018].

- Müller, Pascal / Parish, Yoav I H (2001). *Procedural Modeling of Cities*, online unter: https://graphics.ethz.ch/Downloads/Publications/Papers/2001/p_Par01.pdf [Stand 02.09.2018].
- Nachtigall, Werner (1997). *Vorbild Natur – Bionik-Design für funktionelles Gestalten*, Heidelberg: Springer.
- Owenelton (2012). *If Spirograph were 3D*, online unter: <http://matheminutes.blogspot.com/2012/01/if-spirograph-were-3d.html> [Stand 03.09.2018].
- Runions, Adam / Lane, Brendan / Prusinkiewicz, Przemyslaw (2007). *Modeling Trees with a Space Colonization Algorithm*, in: Ebert, D. / Mérillou, S. (Hrsg.) (2007). *Eurographics Workshop on Natural Phenomena*, Calgary: Department of Computer Science, University of Calgary, online unter: <http://algorithmicbotany.org/papers/colonization.egwnp2007.large.pdf> [Stand 11.08.2018].
- Schneebeli, H. R. / Vollmer, H. R. (1998). *Skalarprodukte, Schwingungen, Signale, Themen zu Anwendungen der Mathematik*, Zürich: sabe Verlag.
- Side Effects Software Inc. (2018). SideFX (Homepage), online unter: <https://www.sidefx.com/> [Stand 03.09.2018].
- Side Effects Software Inc. (o.J.a). *VEX*, online unter: <http://www.sidefx.com/docs/houdini/vex/index.html> [Stand 07.05.2018].
- Side Effects Software Inc. (o.J.b). *L-System geometry node*. <http://www.sidefx.com/docs/houdini/nodes/sop/lssystem.html> [Stand 07.08.2018].
- Space Colonization* (2011). Online unter: <http://procworld.blogspot.com/2011/02/space-colonization.html> [Stand 12.08.2018].

V. Abbildungsverzeichnis

V.I Einführung

Abb. 1: Workflow generativer Gestaltung. Bohnacker, Hartmut / Groß, Benedikt / Lazzeroni, Claudius (2009). *Generative Gestaltung*, www.generative-gestaltung.de, online unter: https://en.wikipedia.org/wiki/Generative_design#/media/File:Generative_Design_Process.png [Stand 02.09.2018].

Abb. 2: Erstellung interaktiv generativer Bilder 1. Eigene Abbildung mittels Processing.

Abb. 3: Erstellung interaktiv generativer Bilder 2. Eigene Abbildung mittels Processing.

V.II Anwendungsbereiche

Abb. 4: Goldener Schnitt. Kmhkmh (2012). *Golden ratio line percentages*, online unter: https://commons.wikimedia.org/wiki/File:Golden_ratio_line_percentages.svg [Stand 02.09.2018].

Abb. 5: Fraktale Welt von Guardians of the Galaxy Vol.2. Marvel (2017), online unter: http://www.artofvfx.com/GuardiansGalaxy2/GOTG2_AnimalLogic_ITW_03.jpg. [Stand 02.09.2018].

Abb. 6: Generativer Aufbau eines Schuhs. ManvsMachine (o.J.). *Nike LeBron XIII*, online unter: <https://static1.squarespace.com/static/5661de61e4b00ea0da1c158c/5668c2194bf1187dc34619b2/5668ed5069a91a928ff5df21/1449717094164/2.jpg?format=1000w>. Über: <http://www.simonholmedal.com/#/lebron/> [Stand 02.09.2018].

Abb. 7: Dynamische Datenstruktur. Eigene Abbildung. Aufbauend auf dem Code online unter: <http://www.generative-gestaltung.de/1/code> [Stand 03.09.2018]. Zu finden im „Code package for Processing 3.x“ mit dem Namen M_6_4_01_TOOL.

Abb. 8: Alibaba Hauptquartier. HASSELL Ltd, online unter: https://www.hassellstudio.com/cms_images/123_22-07-2011_6786.png. Über: <https://www.hassellstudio.com/en/cms-projects/detail/alibaba-headquarters/> [Stand 06.09.2018].

Abb. 9: Dreidimensionales Rauschen. Eigene Abbildung. Aufbauend auf dem Code online unter: <http://www.generative-gestaltung.de/1/code> [Stand 03.09.2018]. Zu finden im „Code package for Processing 3.x“ mit dem Namen M_1_5_03_TOOL.

Abb. 10: Kunst in Echtzeit. Eigene Abbildung. Aufbauend auf dem Code online unter: <http://www.generative-gestaltung.de/1/code> [Stand 03.09.2018]. Zu finden im „Code package for Processing 3.x“ mit dem Namen P_4_3_3_02.

Abb. 11: Bionische Trennwand. *Die bionische Trennwand nimmt Gestalt an*, online unter: <https://damassets.autodesk.net/content/dam/autodesk/www/customer-stories/airbus/epic-story-airbus-final-partition-1024x683.jpg>. Über: <https://www.autodesk.de/customer-stories/airbus> [Stand 06.09.2018].

Abb. 12: Einzigartige Modestücke | generativ erstellte Muster. *Sweaters*, online unter: http://casualdata.com/newsknitter/images/nk_sweaters_full.jpg [Stand 02.09.2018].

V.III Umsetzungswege

Abb. 13: Turtlegrafik der Kochschen Schneeflocke. Eigene Abbildung mittels Houdini.

Abb. 14: Prinzip der Ersetzung des L-Systems. Eigene Abbildung.

Abb. 15: Turtlegrafik einer Pflanze. Eigene Abbildung mittels Houdini.

Abb. 16: Prinzip der Aneinanderreihung des L-Systems. Eigene Abbildung.

Abb. 17: Das L-System innerhalb des Geometry Nodes. Eigene Abbildung mittels Houdini.

Abb. 18: Geometrieparameter des L-Systems in Houdini. Eigene Abbildung mittels Houdini.

Abb. 19: Weitere Parameter des L-Systems in Houdini. Eigene Abbildung mittels Houdini.

Abb. 20: Regeleingabe für das L-System in Houdini. Eigene Abbildung mittels Houdini.

Abb. 21: Beispiel einer Selbstreferenz. Eigene Abbildung anhand der Vorlage online unter: <http://www.sidefx.com/docs/houdini/nodes/sop/lssystem.html> [Stand 03.09.2018].

Abb. 22: 3D-Turtlegrafik eines Baumes. Eigene Abbildung mittels Houdini.

Abb. 23: 3D-Turtlegrafik eines Baumes mit weiteren Variablen. Eigene Abbildung mittels Houdini.

Abb. 24: Methode des SCA. Vgl. Runions / Lane / Prusinkiewicz 2007.

Abb. 25: Ablauf des Algorithmuses. Vgl. Runions / Lane / Prusinkiewicz 2007.

Abb. 26: Volumenerstellung in Houdini. Eigene Abbildung mittels Houdini.

Abb. 27: Rauschen innerhalb des Volumens. Eigene Abbildung mittels Houdini.

Abb. 28: Verlaufseinstellung der Rauschintensität/-varianz. Eigene Abbildung mittels Houdini.

Abb. 29: Erstellung der Anziehungspunkte. Eigene Abbildung mittels Houdini.

Abb. 30: Erstellung der zu emittierenden Punkte. Eigene Abbildung mittels Houdini.

Abb. 31: Anwenden des SCAs. Eigene Abbildung mittels Houdini.

Abb. 32: Teilschritte der Generierung. Eigene Abbildung mittels Houdini.

Abb. 33: Resultate des SCAs. Eigene Abbildung mittels Houdini.

Abb. 34: Spirograph (Spielzeug). Kungfuman (2007). *Spirograph, not Original*, online unter: <https://commons.wikimedia.org/wiki/File:Spirograph3.jpg> [Stand 03.09.2018].

Abb. 35: Veranschaulichung der Funktionsweise des Spirographen. Eigene Abbildung, Aufbauend auf Schneebeli / Vollmer 1998, S.41.

Abb. 36: Veranschaulichung der Funktionsweise des Spirographen (geändert). Eigene Abbildung, Aufbauend auf Schneebeli / Vollmer 1998, S.41.

Abb. 37: Aufbau des Spirographen in Houdini. Eigene Abbildung mittels Houdini.

Abb. 38: Ausgabe des Partikelsystems. Eigene Abbildung mittels Houdini.

Abb. 39: Resultat des Spirographen. Eigene Abbildung mittels Houdini.

Abb. 40: Spirographvarianten. Eigene Abbildung mittels Houdini.

V.IV Durchführung

Abb. 41: Turtlegrafik eines Stadtviertels. Eigene Abbildung mittels Houdini.

Abb. 42: Aufbau der Generierung des Stadtviertels. Eigene Abbildung mittels Houdini.

Abb. 43: Visualisierungen des Stadtviertels. Eigene Abbildung mittels Houdini und Blender. HDRI, online unter: <https://hdrihaven.com/hdri/> [Stand 03.09.2018].

Abb. 44: Turtlegrafik einer Koralle / eines Baumes. Eigene Abbildung mittels Houdini.

Abb. 45: Aufbau der Generierung einer Koralle. Eigene Abbildung mittels Houdini.

Abb. 46: Teilschritte des Prozesses der Geometrieänderung. Eigene Abbildung mittels Houdini.

Abb. 47: Shadingbeispiel der Koralle. Eigene Abbildung mittels Houdini und Blender.

Abb. 48: Variationen durch Ändern der Gravitation. Eigene Abbildung mittels Houdini und Blender.

Abb. 49: Abstrakt und organisch. Eigene Abbildung mittels Houdini und Blender.

Abb. 50: Willkürliche Vernetzung. Eigene Abbildung mittels Houdini und Blender.

Abb. 51: Labyrinth. Eigene Abbildung mittels Houdini und Blender. Texturen, online unter: <https://texturehaven.com/textures/> [Stand 03.09.2018].

Abb. 52: Menschliches Adersystem. Eigene Abbildung mittels Houdini und Blender. Menschliche Geometrie generiert mit der Software MakeHuman 1.1.1.

Abb. 53: Kombination zweier SCAs. Eigene Abbildung.

Abb. 54: Technisch wirkender SCA innerhalb und außerhalb einer Geometrie. Eigene Abbildung mittels Houdini und Blender. Der Arm ist Teil einer menschlichen Geometrie, generiert mit der Software MakeHuman 1.1.1.

Abb. 55: Ablauf des Booleschen Operators und des SCAs. Eigene Abbildung mittels Houdini und Blender. Der Arm ist Teil einer menschlichen Geometrie, generiert mit der Software MakeHuman 1.1.1. Das Flugobjekt ist eine eigene Arbeit des Kurzfilms „*Schatten der Erde*“.

Abb. 56: Kombinierte Objekte mit anschließendem SCA. Eigene Abbildung.

Abb. 57: Flugobjekt in einer Hand mit Ranken umschlossen. Eigene Abbildung mittels Houdini und Blender. Der Arm ist Teil einer menschlichen Geometrie, generiert mit der Software MakeHuman 1.1.1. Das Flugobjekt ist eine eigene Arbeit des Kurzfilms „*Schatten der Erde*“.

Abb. 58: Verändern eines bestehenden Objektes. Eigene Abbildung mittels Houdini und Blender. Ursprungsobjekt online unter: http://store.chocofur.com/free-01-wood/blender_model/blender-3d-model-cycles-object-furniture-lounge-wood-armchair-paimio-chair [Stand 03.09.2018].

Abb. 59: Torus mit zwei Einsatzmöglichkeiten des SCAs. Eigene Abbildung mittels Houdini.

Abb. 60: SCA als Haut und Innenleben eines Arms. Eigene Abbildung mittels Houdini und Blender. Der Arm ist Teil einer menschlichen Geometrie, generiert mit der Software MakeHuman 1.1.1.

Abb. 61: Aufbau eines möglichen Tools. Eigene Abbildung.

Abb. 62: Spirograph. Eigene Abbildung mittels Houdini und Blender.

Abb. 63: Superformel / Super Shape. Eigene Abbildung mittels Houdini und Blender.

Abb. 64: Dequan Li Attraktor. Eigene Abbildung mittels Houdini und Blender.

Abb. 65: Super Shape - Beispiel Stern. Eigene Abbildung mittels Houdini und Blender.

Abb. 66: Formbeispiel generierter Bewegung. Eigene Abbildung mittels Houdini und Blender.

VI. Anhang

VI.I Verwendete Software

Houdini 16.5 Apprentice: Generierung und Rendering.

Blender 2.79 und 2.8: Rendering.

Processing 3.3.7: Programmierung.

Photoshop CC 2018: Bildbearbeitung.

InDesign CC 2018: Layout und Erstellung der Thesis.

VI.II Digitaler Anhang (DVD beigelegt)

Thesis als PDF.

Eigene Abbildungen.

Umsetzungsweg L-System: 4-1_Baum_L-System.hipnc.

Umsetzungsweg Space Colonization Algorithmus: 4-2_Baum_SCA.hipnc.

Umsetzungsweg Bewegung generieren: 4-3_Spirograph_2D.hipnc.

Durchführung Städte: 5-1-1_Stadviertel_L-System.hipnc.

Durchführung Organische Formen: 5-1-2_Koralle_L-System.hipnc.

Durchführung Anatomie: 5-2-1_Adern_SCA.hipnc.

Durchführung Ranken: 5-2-2_Ranken_SCA.hipnc.

Durchführung Generierte Bewegungen und Formen: 5-3_GenBew.hipnc.