

AW2 Hausarbeit
Simulation von natürlich vorkommenden Objekten

Malte Eckhoff

31. August 2014

Inhaltsverzeichnis

1	Motivation	3
2	Projekt i2e	3
3	Plastic Trees: Interactive Self-Adapting Botanical Tree Models [1]	4
3.1	Übersicht	4
3.2	Algorithmus	5
3.3	Relevanz für die Master-Arbeit	5
4	Interactive Authoring of Simulation-Ready Plants [2]	6
4.1	Übersicht	6
4.2	Algorithmus	7
4.2.1	Aufteilung des Input-Modells in Domänen	7
4.2.2	Instantiierung	8
4.2.3	Berechnung von verformbaren Körpern	8
4.3	Relevanz für die Master-Arbeit	9
5	Capturing and animating the morphogenesis of polygonal tree models	9
5.1	Übersicht	9
5.2	Algorithmus	9
5.2.1	Umgekehrte Wachstums-Simulation	11
5.3	Relevanz für die Master-Arbeit	11
6	Fazit	11

1 Motivation

Bäume in virtuellen Welten in aktuellen Anwendungen haben das Problem schlecht, bis gar nicht auf lokale Umwelteinflüsse zu reagieren. Konkret wird bspw. erwartet, dass die Blätter des Baumes auf Wind reagieren. Die Arbeiten aus der aktuellen Forschung, die hier behandelt werden, beschäftigen sich mit den folgenden Teilproblemen:

- Die Anpassung des Baumes an die Geometrie der Umgebung.
- Die effiziente Simulation der Reaktion des Baumes auf einwirkende interne und externe Kräfte, inklusive der Zerstörung bei zu großer mechanischer Belastung.
- Die Visualisierung des Wachstumsprozesses eines Baumes.

Das Ziel der Master-Arbeit ist es, eine Modellierung zu finden, die alle diese Teilprobleme auf einmal löst. Konkret soll mit der Modellierung die Simulation eines Baumes möglich sein, der über die Zeit wächst, dabei die umliegende Geometrie berücksichtigt und korrekt auf einwirkende Kräfte reagiert. Diese Modellierung soll auf den hier vorgestellten Arbeiten aufbauen. Für jede dieser Arbeiten soll untersucht werden:

- ob diese grundsätzlich für die Master-Arbeit relevant ist.
- welchen Bezug die Arbeit zum Thema der Master-Arbeit hat.
- welche Aspekte der Arbeit für die Master-Arbeit verwendet werden können.

Das Forschungsgebiet, auf das ich mich konzentriere, ist die Computer-Grafik. Es geht explizit nicht um die realistische biologisch korrekte Simulation, sondern um eine ansprechende Visualisierung. Naturgesetze (wie biologische Gesetze) sind nur soweit von Interesse, wie sie der Visualisierung dienen. Der einzige Zweck des Endergebnisses soll es sein, den Benutzer zu beeindrucken. Im folgenden möchte ich meine Forschungsgruppe i2e vorstellen.

2 Projekt i2e

Im Rahmen der Projektgruppe Projekt I2e werde ich im Master bei der Entwicklung einer kollaborativen mixed-reality-Anwendung helfen. Dabei arbeite ich mit folgenden Kommilitonen zusammen: Christian Blank, Iwer Petersen und Raimund Wege. Christian und Iwer kümmern sich um die Implementierung der Sensorik. Raimund kümmert sich um die Ausgabe auf geeigneten AR-Brillen. Ich übernehme die Visualisierung in einer 3D-Engine.

Da wir drei unterschiedliche Programmiersprachen für unser Projekt verwenden und die Anwendung verteilt sein soll, müssen wir eine Middleware implementieren, die die Kommunikation zwischen diesen Programmiersprachen und über das Netzwerk implementiert.

Die Idee ist, dass mehrere Benutzer zusammen an der virtuellen augmented reality Repräsentation einer Problemstellung, einer beliebigen Fachdomäne, arbeiten können. Jeder Benutzer trägt eine AR-Brille durch die er die Visualisierung sehen kann. Die Visualisierung erfolgt, wie in einer AR-Anwendung typisch, als Menge von virtuellen dreidimensionalen Objekten im Raum. Der Ankerpunkt für diese Darstellung im Raum ist dabei ein Tisch, um den die Sensorik herum aufgebaut ist. Die Benutzer können mit den dargestellten virtuellen Objekten über Gesten oder bestimmte physische Objekte (mixed reality) interagieren. Das Tracking des Benutzers und dessen Gesten erfolgt über Tiefenbild-Kameras (bspw. Kinect).

Mein Bereich ist die Logik der Fachdomäne und die Visualisierung der virtuellen Objekte. Außerdem bin ich für die Implementierung der Netzwerk Kommunikations-Middleware für C# verantwortlich.

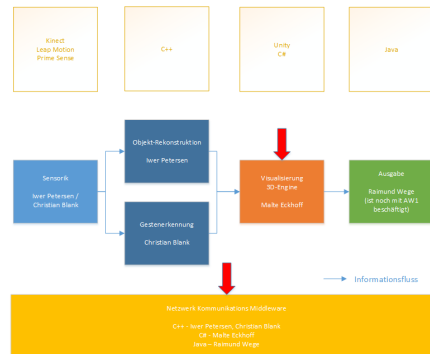


Abbildung 1: Verteilung der Aufgaben im Projekt i2e. Ich übernehme den Bereich der Visualisierung.

Quelle: *Eigene Abbildung*

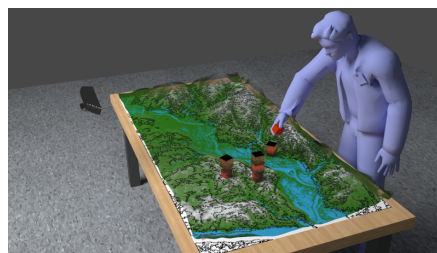


Abbildung 2: Der Benutzer interagiert über Gesten und physische Objekte mit der Anwendung.

Quelle: <http://i2e.informatik.haw-hamburg.de/>

3 Plastic Trees: Interactive Self-Adapting Botanical Tree Models [1]

3.1 Übersicht

Plastic Trees stellt eine dynamische Baum Modellierung und Visualisierung vor, die es komplexen Bäumen erlaubt, dynamisch auf die Geometrie ihrer Umgebung zu reagieren. Die Methode nutzt dabei Veränderungen in der Verfügbarkeit von Licht und die Nähe zu Hindernissen und anderen Bäumen um biologisch motivierte Transformationen zu approximieren. Die Transformationen werden auf den Stamm des Baums und dessen prozedural generiertes Blattwerk angewendet. Teile des Baums werden nur bei Bedarf transformiert. Daher ist der Ansatz sehr viel schneller als gebräuchliche Algorithmen wie Open L-Systeme oder space colonization Methoden. Als Input wird eine Skelett basierte Geometrie verwendet, die aus gewöhnlichen Baum Generierungs Systemen oder aus laser-scanning Methoden entstanden sind. Dieser Ansatz ermöglicht es Benutzern direkt in interaktiven Raten mit Bäumen zu interagieren und interaktiv visuell überzeugende Ökosysteme zu erschaffen.

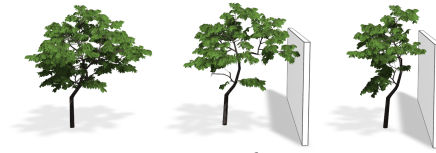


Abbildung 3: Der Plastic Trees Algorithmus erlaubt es Bäumen, sich dynamisch an die Geometrie ihrer Umgebung anzupassen. Der Algorithmus entfernt Äste, die durch die umgebende Geometrie im Schatten liegen und passt den Stamm entsprechend an.

Quelle: [1]

3.2 Algorithmus

Normalerweise benötigt die Anpassung eines Baumes an die Geometrie der Umgebung das Eingreifen des Benutzers. Dieser Algorithmus implementiert die dynamische prozedurale Anpassung des Baumes an die Geometrie seiner Umgebung. Dies erfolgt zur Laufzeit mit interaktiven Raten. Die grundlegende Annahme auf der der Algorithmus basiert ist, dass die Verfügbarkeit von Licht die Gestalt des Baumes bestimmt. Als Input werden fertig generierte Baummodelle¹ verwendet.

Die Gestalt eines Baumes ergibt sich aus zwei Faktoren: Den endogenen Informationen (die genetischen Informationen des Baumes) und den exogenen Einflüssen (der Umwelt des Baumes). Der Algorithmus approximiert die biologisch motivierten Transformationen nur, es erfolgt keine biologisch korrekte Simulation. Durch diese Vereinfachung wird eine hohe Effizienz erzielt.

Aus dem Stamm und den Ästen des Baumes wird ein zusammenhängender Graph generiert. Dieser Graph bildet quasi das Skelett des Baumes. Auf diesem Graphen wird im nächsten Schritt eine Menge von Transformationen B_{trans} definiert, die benutzt werden um die Gestalt des Baumes, in Anpassung an seine Umgebung, zu ändern. Die Blätter werden in Clustern zusammengefasst prozedural generiert. Die Äste in diesen Clustern werden mit Hilfe der GPU generiert und dienen nur optischen Zwecken. Sie sind nicht Teil des Graphen.

Für Bäume ist die Verfügbarkeit von Licht die entscheidende Einflussgröße, welche die Gestalt des Baumes bestimmt. Das Wachstum des Baums basiert darauf, mit einer möglichst geringen Oberfläche möglichst viel Licht einzufangen. In Volumina die durch die Geometrie der Umgebung im Schatten liegen, wird der Baum also nicht hinein wachsen.

Der Algorithmus prüft periodisch ob Teile des Baums im Schatten liegen. Falls dies der Fall ist, werden geeignete Transformationen aus der Menge B_{trans} angewendet, um Äste die im Schatten liegen zu entfernen. Bei Bedarf wird der Neigungswinkel des Stammes mittels dieser Transformationen angepasst (siehe Abbildung 3). Der Baum passt sich so dem Schattenwurf und der Geometrie der Umgebung an.

3.3 Relevanz für die Master-Arbeit

Plastic Trees ist für mein Master-Thema aus mehreren Gründen interessant. Am wichtigsten ist für mich die Modellierung der Reaktionen auf die Umwelteinflüsse. Die verwendete Modellierung abstrahiert stark von den biologischen Verhaltensweisen, schafft es aber gleichzeitig für den Benutzer glaubwürdig zu bleiben.

Die Anpassung an die Lichtverhältnisse und die Geometrie der Umgebung ist eine der sichtbarsten Anpassungen an die Umgebung, die von Bäumen durchgeführt wird. Da es das Ziel der Master-Arbeit ist, die Anpassung der Bäume an die Umwelt zu modellieren, muss diese Eigenschaft berücksichtigt werden.

¹Für die Generierung kann bspw. XFrog (<http://www.xfrog.com/>) verwendet werden.

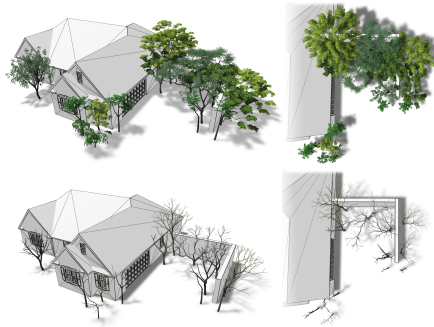


Abbildung 4: Die platzierte Vegetation passt sich mit Hilfe des Plastic Trees Algorithmus zur Laufzeit dynamisch an die komplexe Geometrie der Umgebung an.

Quelle: [1]

Der interessanteste Aspekt der Modellierung ist die fast vollständige Abstraktion von den biologischen Gesetzmäßigkeiten beim Wachstum des Baums. Grundsätzlich müssen nur die Lichtverhältnisse durch die umliegende Geometrie berücksichtigt werden, um eine glaubhafte Visualisierung zu erhalten. Die Arbeit von Pirk et. al. stellt außerdem viele interessante Optimierungen vor, um eine Aktualisierung der Bäume zur Laufzeit zu ermöglichen. Besonders interessant ist die Idee Änderungen in der Topologie des Baumes als Graph-Transformationen zu implementieren.

4 Interactive Authoring of Simulation-Ready Plants [2]

4.1 Übersicht

Physik basierte Simulationen ermöglichen glaubwürdige Reaktionen auf einwirkende externe und interne Kräfte, sowie die dynamische Zerstörung des Baumes bei zu großer mechanischer Belastung. Um ein 3D-Modell eines Baums in einer Physik-Simulation nutzen zu können, muss dieses zuerst manuell aufbereitet werden. Dies nimmt normalerweise viel Zeit in Anspruch. Die primäre Zielsetzung der vorgestellten Arbeit ist es, diese Zeit auf wenige Minuten zu reduzieren. Der verwendete Algorithmus erlaubt die Simulation der Reaktion des Baums auf einwirkende Kräfte, inklusive mechanischer Zerstörung mit interaktiven Raten.

Der Algorithmus teilt den Baum dafür in logische Abschnitte (Domänen) auf. Domänen sind bspw. Blätter, Äste oder Früchte. Im nächsten Schritt werden die Verbindungen zwischen den Domänen bestimmt. Auf dieser Grundlage wird ein Graph generiert, in der die Domänen als Knoten und die Verbindungen als Kanten repräsentiert werden. Ausgehend vom Wurzelknoten wird eine Hierarchie über den Domänen definiert.

Für die Optimierung wird die Erkennung von wiederkehrenden Pflanzenabschnitten wie bspw. Blättern unterstützt. Diese werden als Instanzen bezeichnet. Das Ergebnis der Berechnung für eine Instanz, während eines Schrittes der Simulation, kann dann für alle anderen Instanzen übernommen werden.

Das System skaliert mit der anatomisch korrekten Geometrie von erwachsenen Pflanzen, ist robust gegenüber sog. nicht mannigfaltiger ² (non manifold) Input-Geometrie, nicht verbundener Geometrie, fehlerhafter

²Ein finites zweidimensionales mesh ist genau dann mannigfaltig, wenn alle edges $e_0...e_n$ und triangles $t_0...t_n$ die einen Vertex v treffen ohne Wiederholungen wie folgt zyklisch angeordnet werden können: $t_0, e_0, t_1, \dots, t_n, e_n$ sodass e_i eine edge von t_i und t_{i+1} ist. Das impliziert, dass es für jede edge exakt zwei faces gibt, die diese enthalten [3].

Geometrie und Selbst-Kollision in der Input-Geometrie.

Die Ergebnisse der Arbeit werden mithilfe einer Finiten Elemente Methode (FEM) Simulation demonstriert.

4.2 Algorithmus

4.2.1 Aufteilung des Input-Modells in Domänen

Der Algorithmus basiert auf der Vorgehensweise, den Baum in logische Komponenten, sog. Domänen aufzuteilen. Eine Domäne ist eine logische Einheit des Baumes. Dabei kann es sich bspw. um einen Ast, eine Menge von Blättern oder eine Frucht handeln.

Über diesen Domänen wird, ausgehend vom Stamm, ein Graph definiert. Zwei Domänen sind genau dann verbunden, wenn sie sich in Nachbarschaft zueinander befinden. Eine Frucht ist bspw. mit einem Ast verbunden, der wiederum mit dem Stamm verbunden ist. Die Idee ist es jede Domäne während jedes Zeitschritts der Simulation einzeln zu berechnen. Um die Verbindung der Domänen und die gegenseitige Beeinflussung sicherzustellen, müssen bei dieser Berechnung entsprechende Kupplungs-Kräfte angewendet werden. So kann bspw. jede Frucht des Baums einzeln berechnet werden. Durch die angewandten Kupplungs-Kräfte wird sichergestellt, dass die Früchte am Ast hängen bleiben. Es wird auch sichergestellt, dass die Bewegungen des Astes Auswirkungen auf die verbundenen Früchte haben. Mit dieser Vorgehensweise ist es ebenfalls möglich den Bruch der Verbindung zwischen Domänen, bei zu großer Krafteinwirkung, zu implementieren. Die Frucht würde in diesem Fall vom Ast abreißen und zu Boden fallen.

Es wird zwischen flexiblen (F) und statischen Domänen (R1 oder R2) unterschieden. Es wird zwischen zwei statischen Domänen unterschieden, um Fälle abbilden zu können, in dem eine statische Domäne einer anderen untergeordnet werden soll.

Flexible Domänen werden für Teile des Baums verwendet, die sich zur Laufzeit verformen werden. Ein gutes Beispiel sind der Stamm oder die Äste. Statische Domänen werden für logische Teile des Baumes verwendet, die zur Laufzeit starr sein werden. Diese werden bspw. für Blätter oder Früchte verwendet. Der Rechenaufwand für die Deformation von diesen Domänen würde in keinem Verhältnis zum visuellen Verbesserung stehen. Außerdem erlaubt diese Optimierung die Verwendung der Instantiierungs-Optimierung die in Abschnitt 4.2.2 erläutert wird.

Die Geometrie eines Baum-3D-Modells ist im Normalfall unorganisiert. D.h. es sind keine Informationen hinterlegt, welche Geometrie zu welchen logischen Teilen des Baums (bspw. Blätter, Äste oder Früchte) gehört. Außerdem gibt es normalerweise keine Informationen wie diese logischen Teile miteinander verbunden sind.

Hinzu kommt, dass die Input-Modelle Fehler, wie nicht verbundene Äste oder überflüssige Geometrie, enthalten können. Diese Probleme erschweren eine automatische Verarbeitung soweit, dass eine Markierung der Domänen durch den Benutzer notwendig ist.

Der Benutzer wählt die Dreiecke des Input-Baum-Modells aus die zu einer Domäne gehören sollen, und kategorisiert diese als flexibel (F) oder starr (R1 oder R2). Wobei jedes Dreieck genau einer Domäne zugeordnet sein muss und jede Domäne genau einer Kategorie angehört.

Nachdem der Benutzer eine Wurzel-Domäne gewählt hat, generiert das System automatisch den Domänen-Graphen. Wenn es während der Generierung zu Zyklen im Graphen kommt, werden diese visualisiert und an einer, vom Benutzer gewählten, Stelle aufgetrennt.

4.2.2 Instanziierung

Viele Pflanzen bestehen zu großen Teilen aus sich wiederholenden gleichen Teilen, wie bspw. Blätter und Früchten. Diese verschiedenen Instanzen sind meistens nur rotierte, skalierte oder nichtlinear deformierte Kopien voneinander. Es reicht in diesem Fall, nur eine Instanz aufzubereiten, weil das Ergebnis für alle Instanzen verwendet werden kann.

Die Technik der Instanziierung verringert die Aufbereitungszeit und den Speicherbedarf zur Laufzeit, weil jede Instanz nur einmal aufbereitet und gespeichert werden muss. Die Bearbeitung durch den Benutzer wird ebenfalls stark vereinfacht, weil dieser für jede Instanz die Anker-Punkte (diese werden in folgenden erklärt) nur einmal setzen muss.

Für die Erkennung der Instanzen wird eine Kombination aus dem shape matching Ansatz von Müller et. al. [4] und der Analyse der uv-Koordinaten des texture mappings der Instanzen verwendet. Für die Texturierung wird angenommen, dass jedes mesh jeder Instanz zu genau einer Textur gemapped ist (siehe Abbildung 5 (a)).

Auf dieser Grundlage werden die uv-mappings der meshes der Domänen verglichen. Sind diese gleich und stimmt die Topologie der Domänen überein, werden diese als Kopien der selben Instanz klassifiziert.

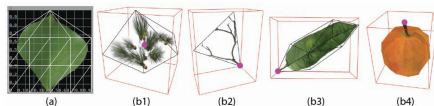


Abbildung 5: Instanzen und Ankerpunkte: (a) Eins zu eins texture-map mit Transparenz. (b) Durch den Benutzer ausgewählte Ankerpunkte (in lila).

Quelle: [2]

Alle als Instanzen klassifizierte Domänen werden in Mengen zusammen gefasst. Für jede dieser Mengen wird eine repräsentative Instanz ausgewählt (die erste Instanz, die im Input-Modell vorkommt). Für jede dieser repräsentativen Instanzen wählt der Benutzer einen Ankerpunkt (siehe Abbildung 5 (b)). Dieser dient im weiteren als Verbindungspunkt mit den angrenzenden Domänen. Normalerweise sollte der Ankerpunkt ans Ende der jeweiligen botanischen Komponente gesetzt werden (bspw. der Stiel eines Blattes oder einer Frucht).

4.2.3 Berechnung von verformbaren Körpern

Für die Berechnung der Verformung der flexiblen Domänen (F-Domänen) wird der Squashing cubes Algorithmus von James et. al. verwendet [5]. Dieser Algorithmus benötigt kein Eingreifen des Benutzers und kann auf beliebige Input-Modelle angewendet werden. In diesem Fall handelt es sich beim Input-Modell um eine F-Domäne (bspw. einen Ast). Basierend auf dem Input-Modell wird eine Menge von Voxeln generiert, die die Gestalt des Modells approximieren. Bei den Voxeln handelt es sich um kleine verformbare Würfel (die Squashing Cubes). Durch die Parametrisierung der Masse und des Verformungs-Widerstandes der Squashing Cubes lassen sich viele verschiedene Materialien simulieren. In diesem Fall ist das simulierte Material das Holz des Baums.

Das Squashing Cube Modell kann in einer Physik-Simulation verwendet werden, um die resultierende Deformation des Input-Modells (in diesem Fall der Domäne) durch externe Kräfte zu berechnen. Die resultierende Deformation des Squashing Cube Modells wird in jedem Zeitschritt der Simulation zurück auf

das Input-Modell interpoliert. Damit wird erreicht, dass sich Äste und Stamm des Baumes bei genügender Krafteinwirkung verbiegen.

4.3 Relevanz für die Master-Arbeit

Die sichtbarste Reaktion auf Umwelteinflüsse ist die Reaktion des Baums auf externe und interne Kräfte. Dadurch hat die Arbeit von Zhao et. al. eine hohe Relevanz für die Master-Arbeit.

Die Aufteilung des Baumes in Domänen und die dadurch ermöglichten Optimierungen ist ein Konzept, das für die Master-Arbeit übernommen werden wird. Ebenfalls wird die Arbeit von James et. al. [5] zur effizienten Berechnung von verformbaren Körpern genutzt werden.

Die Arbeit von Zhao et. al. hat Lösungen für die Verarbeitung von fehlerhaften und/oder unvollständigen Input-Modellen vorgestellt. Diese werden für die Master-Arbeit jedoch nicht verwendet werden, da manuell sichergestellt werden wird, dass die Input-Modelle fehlerfrei und vollständig sind.

5 Capturing and animating the morphogenesis of polygonal tree models

5.1 Übersicht

In großen Baumbeständen in virtueller Welten kann sich, bei einer zu geringen Anzahl an verwendeten Baummodellen, das Problem ergeben, dass viele Bäume sehr ähnlich oder identisch aussehen. Dies schadet der Glaubwürdigkeit der virtuellen Welt. Weitere Baummodelle zu erstellen oder vorhandene zu modifizieren, bringt aber normalerweise einen hohen manuellen Aufwand mit sich.

Pirk et. al. [6] stellen eine Lösung vor, die es erlaubt aus einem Input-Baummodell automatisiert eine hohe Anzahl an Variationen zu erzeugen. Die Lösung erlaubt es, aus einem statischen Baummodell Wachstumsstufen abzuleiten, die das natürliche Wachstum des Baums approximieren.

Als Erstes wird der Baum analysiert und eine graphen-basierte Beschreibung des Skeletts erstellt. Auf Grundlage der Eigenschaft der Selbstähnlichkeit von Bäumen ist es möglich zu bestimmen, an welchen Stellen des Baumes, während des Wachstums, Äste abgestorben oder weggeschnitten wurden. Diese fehlenden Äste werden dem Graphen hinzugefügt.

Mit Hilfe von botanischen Wachstumsmodellen und allometrischen³ Regeln erlaubt der Algorithmus außerdem die Generierung von überzeugenden Wachstums-Animationen von einem jungen- bis zum Input-Baum. Alle diese Wachstums-Zwischenstufen stehen dem Benutzer zur Verfügung. Dies ermöglicht die Population einer Szene mit vielen unterschiedlichen Bäumen auf Grundlage einer kleinen Anzahl an Input-Baummodellen.

5.2 Algorithmus

Der Algorithmus arbeitet auf einer Graph-basierten-Datenstruktur. Diese wird aus dem Input-Modell generiert. Typischerweise werden diese Modelle als eine gruppierte Liste von Vertices mit Verbindungsinformationen übergeben. Die Gruppen repräsentieren dabei die Organe der Pflanze, wie Äste, Blätter oder

³Die Allometrie beschäftigt sich mit dem Verhältnis zwischen der Größe eines Körpers und dessen Form, Anatomie, Physiologie und Verhalten. Diese Wissenschaft findet in der Statistik sowie in der Biologie Anwendung [7,8]. Für diesen Anwendungsfall besonders von Interesse sind die Aussagen die diese Wissenschaft über die Größenverhältnisse wachsender Entitäten macht.

Blüten.

Als Erstes werden die Äste des Baum-Input-Modells in eine Graph-Darstellung überführt. Pirk et. al. nutzen dafür einen Algorithmus von Au et. al. [9]. Dieser Algorithmus erlaubt die Generierung eines sog. *curve-skeletons*⁴ aus einem beliebigen *input-mesh*. Der große Vorteil ist, dass dieser Algorithmus, im Gegensatz zu den meisten anderen *skeleton-extraction* Algorithmen, direkt auf das *Input-Mesh* angewendet werden kann. Der Algorithmus kontrahiert das *Input-Mesh* mit Hilfe eines *Laplacian-smoothing-Operators* in ein *Null-Volumen-Mesh*. Diese Kontrahierung verändert nicht die Verbindungen innerhalb des *meshes* und behält die Hauptmerkmale des *input-meshes* bei. Das kontrahierte *mesh* wird dann in ein eindimensionales Skelett überführt. Aus diesem eindimensionalen Skelett wird dann in die zuvor erwähnte Graphenstruktur der Äste generiert.

Während des Wachstums ändern sich die Verzweigungswinkel mit der Zeit. Dies liegt an verschiedenen Arten von Tropismen⁵. Es ist auch möglich, dass der Baum an bestimmten Stellen mehr Holz entwickelt, um externen Kräften entgegen zu wirken.

Um diese Änderungen effizient zu modellieren, wird ein *Zweig-Representation* genutzt, die die Koordinaten des Zweiges mit zwei Winkeln α und β relativ zum Vater-Ast angibt. Äste können auch mit mehr als einer Kante repräsentiert werden, weil konsekutive Sequenzen von Kanten ohne Verzweigungen in dieser Darstellung erlaubt sind.

Botanische Strukturen werden oft durch die Designierung eines Hauptastes und der Ordnung der Zweige durch eine *Gravelius* Ordnung beschrieben [13]. Die *Gravelius* Ordnung wurde initial für die Beschreibung binärer Flusssysteme entwickelt, wurde aber auch von Holton erfolgreich für die Modellierung von Bäumen eingesetzt [14].

Dem Wurzelknoten der Graphenstruktur wird die Stufe $g_{,root} = 1$ zugewiesen. Allen Kanten mit dem geringsten Verzweigungswinkel wird die gleiche Stufe zugewiesen. Kanten mit einem größeren Verzweigungswinkel wird eine um eins höhere Stufe zugewiesen. Dies wird für alle Knoten des Graphen wiederholt. Mit dieser *Gravelius* Ordnung lassen sich der Hauptstamm und die Neben-Äste des Baumes bestimmen. Außerdem ist es möglich Ketten von Kanten im Graphen zu definieren: Eine Kette von Kanten ist eine Sequenz von aufeinanderfolgenden Kanten ohne Veränderung der Stufe.

Zusätzlich werden Informationen über die durchschnittliche Verzweigungsrate von Verzweigungen mit Kanten verschiedener Stufen und die durchschnittliche Länge von Internodium⁶ Abschnitten in terminalen Ästen gesammelt (die durchschnittliche Entfernung zwischen zwei Blättern).

Auf Grundlage dieses Graphen wird eine Wachstumssimulation in umgekehrter Richtung durchgeführt. Dies umfasst die Berechnung der individuellen Wachstumsraten der einzelnen Äste, Interpolation der Ast Radien, Aktualisierung der Ast-Radien, Hinzufügen von abgeschnittenen Ästen und das Abschneiden von vorhandenen Ästen. Bei dieser Berechnung wird davon ausgegangen, dass die Umweltbedingungen über die Zeit, in der der Baum gewachsen ist, gleichbleibend sind.

⁴Ein *curve-skeleton* ist eine geometrische Entität, die das Volumen eines Objekts abstrahiert.

⁵Bei Tropismen handelt es sich um Bewegungen die von Pflanzen als Reaktion auf externe Reize ausgeführt werden. Bspw. dem Wachstum hin zu Lichtquellen (*Phototropismus*) oder der Reaktion auf die Gravitation (*Gravitropismus*) [10–12].

⁶Das *Internodium* bezeichnet in der Botanik den Teil einer Sprossachse, zwischen zwei Knoten (*Nodi*), die definitionsgemäß keine Blätter trägt [15]. In diesem Fall sind die Äste des Baums die Sprossachsen. Die Knoten entsprechen den Stellen an denen sich der Ast verzweigt oder Blätter wachsen.

5.2.1 Umgekehrte Wachstums-Simulation

Für die Interpolation früherer Wachstumsstufen ist die Bestimmung der **individuellen Wachstumsrate der Äste**, die **Interpolation der Radien der Äste** und das **Hinzufügen abgeschnittener Äste** wichtig.

Die **Wachstumsrate eines Astes** kann aus der Relation der Länge zu anderen Ästen berechnet werden. Es wird davon ausgegangen, dass die Äste mit einer konstanten Geschwindigkeit wachsen. Das ist nicht biologisch korrekt, führt aber zu Ergebnissen, die glaubwürdig genug aussehen.

Leonardo Da Vinci hat herausgefunden, dass der Flächeninhalt, und damit auch der **Radius eines Astes**, der Summe des Flächeninhalts seiner Kind-Äste entspricht [16]. Genauere empirische Untersuchungen fanden heraus, dass sich der Umfang des Vater Astes c zu dem seiner Kind-Äste c_1 und c_2 wie folgt verhält $c^{2,49} = c_1^{2,49} + c_2^{2,49}$ [17].

Ein gegebener Baum mag initial für eine bestimmte Umgebung modelliert worden sein. In einer urbanen Umgebung wären bspw. die unteren Äste gekürzt. Das bedeutet, dass für die Simulation des Wachstums diese **Äste hinzugefügt** werden müssen. Um das zu erreichen, nutzt man die Selbstähnlichkeits-Eigenschaft von Bäumen. Der Algorithmus kopiert den größtmöglichen zusammen hängenden Teil des Astwerks aus dem oberen Teil des Baums und fügt diesen im unteren leeren Teil des Baumes ein. Die danach noch fehlenden Teile des Baumes werden dann mit der geringst möglichen Anzahl an Kopieroperationen ergänzt.

5.3 Relevanz für die Master-Arbeit

Die Arbeit von Pirk et. al. [6] löst das Problem der Approximierung der Wachstumsprozesse eines Baumes. Die Lösung erlaubt es auch den Wachstumsprozess indirekt zu beeinflussen, indem das Input-Modell entsprechend modifiziert wird. Während des Wachstums kann es aber zu temporär inkorrekten Baum-Konfigurationen kommen, weil Geometrie ohne Berücksichtigung der Umgebung hinzugefügt wird.

Die Arbeit beschäftigt sich nicht mit der Reaktion auf lokale Umwelteinflüsse. Dieser Aspekt wird aber durch die beiden anderen vorgestellten Arbeiten abgedeckt. Es müsste möglich sein von Plastic Trees modifizierte Baummodelle als Input für diese Arbeit zu verwenden. Das Input-Modell wäre damit bereits an die Geometrie der Umgebung angepasst. Der Wachstumsprozess, der zu diesem Baum geführt hat, kann dann approximiert werden. Für das Problem der temporär inkorrekten Baum-Konfigurationen muss in diesem Fall natürlich noch eine Lösung gefunden werden. Dies ist unter anderem der Gegenstand weiter Forschung in der Master-Arbeit. Eine andere mögliche Weiterentwicklung ist die Modifikation des Algorithmus, um wechselnde Umweltbedingungen während des Wachstumsprozesses zu berücksichtigen.

6 Fazit

Alle vorgestellten Arbeiten sind für die Master-Arbeit von hoher Relevanz. Plastic Trees und Interactive Authoring of Simulation-Ready Plants haben einen sehr direkten Bezug zum Thema der Verbesserung der Reaktion auf Umwelteinflüsse. Die Arbeit Capturing and animating the morphogenesis of polygonal tree models behandelt zwar nicht die Reaktion auf Umwelteinflüsse, löst aber das wichtige Problem der Visualisierung des Wachstums der Bäume. In Kombination mit Plastic Trees sollte es möglich sein, die Geometrie der Umgebung während des Wachstumsprozesses zu berücksichtigen.

Plastic Trees löst das Problem der Anpassung eines Baumes an die Geometrie seiner Umgebung. Interactive Authoring of Simulation-Ready Plants ermöglicht die Reaktion auf einwirkende Kräfte und die dynamische

Zerstörung bei zu hoher mechanischer Belastung. Capturing and animating the morphogenesis of polygonal tree models ermöglicht die Approximation und Visualisierung des Wachstumsprozesses eines gegebenen Baums. Alle diese Lösungen verbessern die Reaktionen des simulierten Baums auf externe Einflüsse. Jede der vorgestellten Arbeiten besitzt Aspekte, die Einfluss auf die Master-Arbeit haben. Die wichtigsten sind die abstrakte Modellierung der biologischen Prozesse von Plastic Trees, die Optimierungen der Physik-Simulation von Interactive Authoring of Simulation-Ready Plants und die Visualisierung des Wachstumsprozesses in Capturing and animating the morphogenesis of polygonal tree models.

Literatur

- [1] S. Pirk, O. Stava, J. Kratt, and M. Said, “Plastic trees: interactive self-adapting botanical tree models,” *ACM Transactions on Graphics*, 2012. [Online]. Available: http://graphics.uni-konstanz.de/publikationen/2012/plastic_trees/website/
- [2] Y. Zhao and J. Barbic, “Interactive authoring of simulation-ready plants,” *ACM TRANSACTIONS ON GRAPHICS*, no. Figure 1, 2013. [Online]. Available: <http://run.usc.edu/botanical/>
- [3] A. Van Dam and S. K. Feiner, *Computer graphics: principles and practice*. Pearson Education, 2014.
- [4] M. Müller and B. Heidelberger, “Meshless deformations based on shape matching,” *ACM Transactions on Graphics*, 2005. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1073216>
- [5] D. James, J. Barbič, and C. Twigg, “Squashing cubes: Automating deformable model construction for graphics,” *ACM SIGGRAPH 2004 Sketches*, vol. 23, p. 1997, 2004. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1186271>
- [6] S. Pirk, T. Niese, O. Deussen, and B. Neubert, “Capturing and animating the morphogenesis of polygonal tree models,” *ACM Transactions on Graphics*, vol. 31, no. 6, p. 1, Nov. 2012. [Online]. Available: http://graphics.uni-konstanz.de/publikationen/2012/tree_growth/website/
- [7] J. Damuth, “Scaling of growth: Plants and animals are not so different,” *Proceedings of the National Academy of Science*, vol. 98, pp. 2113–2114, Feb. 2001.
- [8] C. Small, *The statistical theory of shape*. New York: Springer, 1996.
- [9] O. Au, C. Tai, and H. Chu, “Skeleton extraction by mesh contraction,” *ACM Transactions on Graphics*, 2008. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1360643>
- [10] N. Campbell, *Biologie*. Heidelberg Berlin: Spektrum, Akad. Verl, 1997.
- [11] U. Lüttge, *Botanik*. Weinheim: Wiley-VCH-Verl, 2005.
- [12] E. Strasburger, *Lehrbuch der Botanik für Hochschulen*. Heidelberg etc: Spektrum Akademischer Verlag, 1998.
- [13] H. Gravelius, *Grundriss der gesamten Gewässerkunde: in vier Bänden*. G.J. Göschen’sche Verlagsbuchhandlung, 1914. [Online]. Available: <http://books.google.de/books?id=tGmCnQEACAAJ>
- [14] M. Holton, “Strands, Gravity and Botanical Tree Imagery,” *Computer Graphics Forum*, vol. 13, no. 1, pp. 57–67, Feb. 1994. [Online]. Available: [/citations?view_op=view_citation&continue=/scholar?hl=en&as_sdt=0,5&scilib=1&scioq=J.+Weber+and+J.+Penn.+Creation+and+rendering+of+realistic+trees.&citilm=1&citation_for_view=-wBMwZgAAAAJ:qjMakFHDy7sC&hl=en&oi=phttp://doi.wiley.com/10.1111/1467-8659.1310057](http://citations?view_op=view_citation&continue=/scholar?hl=en&as_sdt=0,5&scilib=1&scioq=J.+Weber+and+J.+Penn.+Creation+and+rendering+of+realistic+trees.&citilm=1&citation_for_view=-wBMwZgAAAAJ:qjMakFHDy7sC&hl=en&oi=phttp://doi.wiley.com/10.1111/1467-8659.1310057)
- [15] G. Wagenitz, *Wörterbuch der Botanik*, 2nd ed. Heidelberg u.a: Spektrum Akad. Verl, 2003.
- [16] J. P. RICHTER, *The Notebooks of Leonardo da Vinci vol. 1*, new york. ed. Dover Publications Inc, 1970.

- [17] C. D. Murray, "a Relationship Between Circumference and Weight in Trees and Its Bearing on Branching Angles." *The Journal of general physiology*, vol. 10, no. 5, pp. 725–9, May 1927. [Online]. Available: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2140927&tool=pmcentrez&rendertype=abstract>