

---

# 6 Regelbasierte Objekterzeugung

## Interaktive Modellierung

---

Im letzten Kapitel wurde gezeigt, dass regelbasierte Verfahren eine sehr mächtige Methode zur Herstellung verschiedenster Pflanzen sind. Trotzdem gibt es viele alternative Modellierverfahren. Dies liegt zum einen an der Vielfalt der Modellierungsanforderungen, zum anderen an der wenig intuitiven Steuerung der L-Systeme. Selbst für einen erfahrenen Benutzer ist es aufwendig, eine spezifische Pflanze zu erzeugen. Zwar scheinen die jüngsten Entwicklungen bei den L-Systemen dieses Problem zu vermindern, ein für jedermann benutzbares System ist aber noch nicht erreicht. Prozedurale Methoden haben genau entgegengesetzte Charakteristika: Meist lässt sich mit einem Algorithmus nur eine sehr eingeschränkte Menge von Pflanzen erzeugen, die Handhabung der Prozeduren und deren Parametrisierung ist jedoch einfach und intuitiv.

Eine Kombination der beiden Ansätze verbindet die Intuitivität des prozeduralen Modellierens mit der Mächtigkeit regelbasierter Verfahren. Im xfrog-Modelliersystem [?, ?, ?, ?] wird dies auf der Basis der so genannten regelbasierten Objekterzeugung durchgeführt.

Eine Pflanze wird hier durch eine Kombination von Komponenten repräsentiert. Die Komponenten erzeugen über prozedurale Verfahren Teile der Pflanzengeometrie wie etwa Blätter, Stiele oder einfache Graphikprimitive. Spezielle Multiplikationskomponenten vervielfältigen die erzeugte Geometrie anderer Komponenten. Weitere Komponenten werden zum globalen Modellieren verwendet. Der Benutzer kann über spezielle graphische Benutzerschnittstellen die Parameter der einzelnen Komponenten einstellen.

So verwendet die Komponente zum Herstellen von Blättern einen Polygonzug, um den Umriss der Blätter zu spezifizieren, verschiedene Krümmungen können über Schieberegler entlang der Blattachsen eingestellt werden. Diese Art von Interfaces war das Vorbild für die interaktiven Editiermöglichkeiten für L-Systeme, die bereits in Abschnitt 5.9 vorgestellt wurden.

In vielen Fällen müssen Teile einer Pflanze algorithmisch verteilt werden. Ein klassisches Beispiel ist wiederum die Anordnung der Samen einer Sonnenblume nach dem goldenen Schnitt. Solche Verteilungen werden über Multiplikationskomponenten erzeugt, deren Algorithmen über die Anzahl der zu generie-

p-Graph und i-Baum →

renden Objekte, ihre Verteilungscharakteristik sowie ihre Ausrichtung parametrisiert werden.

Die Pflanze wird durch das Verbinden von Komponentenprototypen zu einem gerichteten Graphen (dem p-Graph) definiert. Der Graph repräsentiert das Regelsystem, seine Kanten beschreiben Erzeugungsregeln bzw. Erzeugungsabhängigkeiten. Ist die Geometrie einer Vaterkomponente erzeugt, stößt diese die Geometrieerzeugung aller Kinder an, bis der gesamte Baum durchlaufen ist. Hierbei können die Komponenten beliebig verbunden werden, auch Rekursionen sind möglich.

Der p-Graph wird zur Geometrieerzeugung traversiert und der sog. i-Baum wird aufgebaut. Dies ist ein temporärer Baum von Komponenteninstanzen, aus welchem schließlich die geometrischen Daten erzeugt werden. Der Zwischenschritt ist notwendig, da im p-Graph die Struktur einer Pflanze in zweierlei Weise repräsentiert ist: als Verbindungsstruktur der Kanten und über die Multiplikationskomponenten. Trotz dieses Zwischenschrittes ist die Geometrieerzeugung hinreichend schnell, so dass auch komplexe Objekte interaktiv am Bildschirm bearbeitet werden können. Durch diese Modelliermöglichkeiten konnten bis heute mehrere tausend Pflanzen erzeugt werden, die für unterschiedlichste Einsatzzwecke benutzt werden können.

Neben der intuitiven Bedienung des Systems durch Komponenten ist die zweifache Repräsentation von Struktur der Hauptunterschied zu klassischen Regelsystemen wie etwa den L-Systemen oder dem graphbasierten Objektinstanzierungsparadigma aus Abschnitt 5.11. Zwar sieht Prusinkiewicz in seinem „Virtuellen Laboratorium“ für L-Systeme [?] auch die Möglichkeit des Ausführens externer Prozeduren zur Erzeugung von Pflanzenteilen vor, in der regelbasierten Objekterzeugung sind die Algorithmen aber integraler Bestandteil der Modellierung und ermöglichen hierdurch die effiziente und flexible Erzeugung vieler Verzweigungsstrukturen. Um diesen Unterschied zu verdeutlichen, wird der Prozess der Multiplikation im Folgenden genauer beschrieben.

## 6.1 Algorithmische Multiplikation

Ist eine Multiplikationskomponente Teil des p-Graphen, wird während der Erzeugung des i-Baumes der nachfolgende Teilbaum so oft erzeugt, wie es in der Multiplikationskomponente als Parameter angegeben ist. Hierbei werden alle Kopien der Kindkomponenten mit der Kopie der Multiplikationskomponente verbunden.

Prototyp →

Aus den Komponentenprototypen werden bei der Generierung des i-Baumes Instanzen. Der Unterschied zwischen einem Prototyp und einer Instanz liegt in der Definition der Parameter, die an die Kindkomponenten weitergegeben werden, wie etwa deren lokale Koordinatensysteme. In einem Prototyp sind diese Parameter als Wertebereiche gespeichert, die bei der Vervielfältigung für jede erzeugte Kindinstanz individuell durch Interpolation berechnet werden.

Ist beispielsweise in einer Multiplikationskomponente die Größe der Kinder im Bereich  $[v_0, v_1]$  eingestellt, so wird dem  $i$ -ten Kind von  $n$  der Wert  $v_i = v_0 + i(v_1 - v_0)/(n - 1)$  durch die Interpolation zugewiesen. Auf die Interpolation kann gegebenenfalls vor deren Weitergabe eine vom Benutzer definierte Funktion angewendet werden. Auf diese Weise können Winkelfunktionen oder Zufälligkeiten in die Modelle integriert werden.

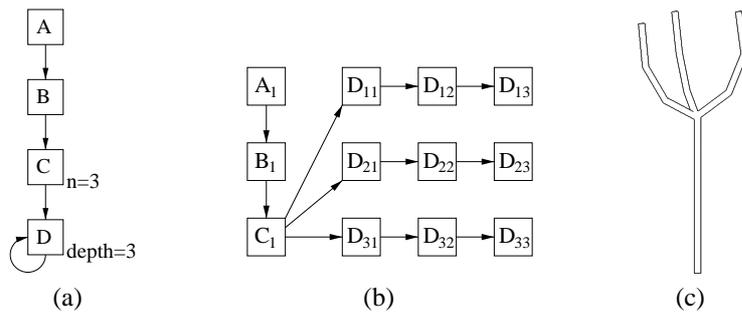
Ähnlich wie prozedurale Vervielfältigungen bei der Herstellung des  $i$ -Baumes expandiert werden, werden auch Rekursionen behandelt. In jeder Komponente ist die maximale Rekursionstiefe als Parameter definiert. Muss eine Rekursion expandiert werden, so wird nach der Erzeugung einer entsprechenden Anzahl von Instanzen abgebrochen. Nach den Bemerkungen aus Kapitel 2 über rekursive Algorithmen braucht an dieser Stelle aber nur erwähnt zu werden, dass diese Form des Modellierens nicht so häufig benötigt wird.

Ist der  $i$ -Baum aufgebaut, wird er von der Wurzel zu den Blättern hin durchlaufen. Die Wurzelkomponente erzeugt ihre Geometrie und stößt alle ihre Kinder an, dasselbe zu tun. Dies wird rekursiv fortgesetzt, bis der gesamte Baum durchlaufen ist.

Ein Beispiel soll den Prozess der Geometrieerzeugung weiter verdeutlichen: Sei  $A$  Komponentenprototyp der Wurzel des  $p$ -Graphen und  $B$  sei Prototyp einer geometriegenerierenden Komponente, die einen Stiel erzeugt.  $C$  sei eine Multiplikationskomponente, welche drei Instanzen des Nachfolgers erzeugen soll, aber keine Geometrie. Komponentenprototyp  $D$  erzeugt ebenfalls einen kleinen Stiel. Im Beispiel ist eine Rekursionstiefe von drei eingestellt. Der Graph wird so definiert, dass eine Rekursion auf  $D$  definiert ist.

← Rekursion

← Beispiel



*Abbildung 6.1*  
Herstellung von Geometrie: (a) Der Benutzer definiert den  $p$ -Graphen; (b) zur Generierung der Geometrie wird ein temporärer  $i$ -Baum erzeugt; (c) resultierende Geometrie

In Abb. 6.1(a) ist der  $p$ -Graph dargestellt, Abb. 6.1(b) zeigt den entsprechenden  $i$ -Baum, in dem die Instanzen jeder Komponente  $X$  mit  $X_i$  beschrieben sind. Die Rekursion auf  $D$  erzeugt im  $i$ -Baum drei Instanzen  $D_1, D_2$  und  $D_3$ , die aneinander gehängt werden. Diese Teilbäume werden durch die Multiplikationskomponente  $C$  dreifach kopiert, wobei jede Instanz  $D_{j1}$  durch den Interpolationsmechanismus ein individuelles Koordinatensystem zugewiesen bekommt und daher auch in unterschiedliche Richtung weisende Zweige erzeugen kann. Insgesamt erzeugt die Expansion also neun Instanzen von  $D$ , jeder der drei Teilbäume generiert einen Ast der Geometrie aus Abb. 6.1(c).

## 6.2 Komponententypen

Die Komponenten des Systems lassen sich in drei Klassen einteilen: Geometrieerzeugung, Multiplikation und globale Modellierung. Alle Komponenten haben einen Basissatz von Parametern, der es erlaubt, ein geometrisches Primitiv zu erzeugen, dieses relativ zum Vorgänger zu positionieren und die Geometrie nachfolgender Komponenten in Bezug zur eigenen anzuordnen.



### Kamerakomponente

Die Kamerakomponente ist die Grundkomponente des Systems, muss also immer als erstes Element im Strukturbaum stehen. Sie beherbergt alle Parameter, die für die Ansicht des Modells benötigt werden. Dies sind Dinge wie etwa die Lage des Kamerakoordinatensystems und der Öffnungswinkel der Kamera (beide zusammen bestimmen die perspektivische Abbildung auf dem Bildschirm), die Position und Art von Lichtquellen für die Beleuchtungssimulation sowie weitere Darstellungsparameter.



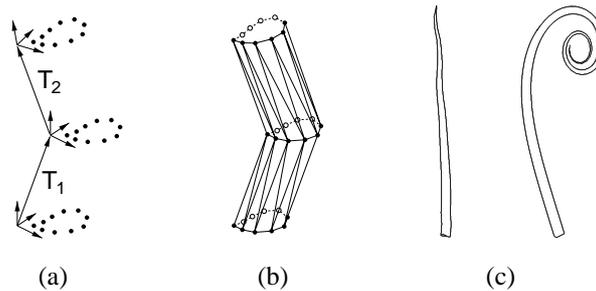
### Basiskomponente

In der Basiskomponente ist nur der grundlegende Parametersatz vorhanden, alle anderen Komponenten sind hiervon abgeleitet und bieten zusätzliche Funktionalität. Die Basiskomponente selbst gehört zu der Klasse von Komponenten, welche zur Geometrieerzeugung gedacht sind.

geometrische Primitive →

Als geometrische Primitive lassen sich Objekte wie Würfel, Kugeln, Zylinder oder Tori erzeugen, aber auch Mengen diskreter Punkte. Diese können als offene Punktmenge (Area) oder geschlossene Punktmenge (Tube) definiert werden und liegen üblicherweise in einer Ebene. Wird an die Komponente eine weitere angehängt, welche ebenfalls eine Punktmenge definiert, so werden die beiden Punktmenge trianguliert und bilden auf diese Weise eine Oberfläche. In der unten beschriebenen Hornkomponente wird dieser Vorgang intern angewendet, um aus einer Folge triangulierter Punktmenge Äste und Stiele zu erzeugen. Abb. 6.2 zeigt den Vorgang bei der Erzeugung von stielartigen Objekten.

Abbildung 6.2  
Geometriedefinition eines Stieles:  
a) Punktmenge werden relativ  
zueinander spezifiziert; b) durch  
Triangulation wird die Oberfläche  
erzeugt; c) unterschiedlich geformte  
Stiele



Das von der Komponente erzeugte Primitiv kann mit einer Farbe oder Textur versehen werden. Wird eine Textur gewählt, so lassen sich die Texturkoordinaten sowie die Transparenz einstellen, um die Textur optimal zu positionieren und darzustellen. Texturen werden üblicherweise für Blätter verwendet, um deren Realismus zu steigern und eine aufwendige Randkonstruktion zu vermeiden, aber auch Rinde wird auf diese Weise erzeugt.

Innerhalb der Geometrieerzeugung stehen zweierlei Arten von Transformationen zur Verfügung. Einerseits lässt sich das Primitiv in Lage und Größe verändern, andererseits kann die Veränderung auf das lokale Koordinatensystem angewendet werden und betrifft auf diese Weise auch die Geometrie aller nachfolgenden Komponenten.

Ein weiterer Parameter ist die Stärke des Fototropismus. Sie bestimmt, ob und wie stark sich das Primitiv in Bezug auf ein externes Lichtfeld ausrichtet. Dies ist besonders für Blätter interessant, deren Oberflächen auf diese Weise transversal fototrop (senkrecht zum Licht) ausgerichtet werden können.

Der bereits erwähnte Rekursionsparameter bestimmt, wie oft eine Rekursion im p-Graphen angewendet werden soll. Als weitere wichtige Modelliermöglichkeit kann der Benutzer in einem Feld auswählen, ob die Geometrie der Komponente bei der Vorschaltung eines Multiplikators generiert wird oder nicht. Jede Multiplikationskomponente gibt allen erzeugten Kindinstanzen eine individuelle Nummer. Diese Nummer wird mit den im Feld gespeicherten Werten verglichen. Der Benutzer kann auf diese Weise vorgeben, eine Anzahl von Komponentengeometrien bei der Multiplikation nicht zu erzeugen. Dieser Mechanismus wird für die Generierung von Ausnahmen verwendet, die in jeder natürlichen Pflanze vorkommen. Beispiele sind abgestorbene Äste oder beschädigte Blätter.

← Ausnahmen

Die Definition von Ausnahmen erzeugt eine Art Kontextsensitivität im bislang kontextfreien System, da hier das Erscheinen eines Objektes an eine Umgebung gekoppelt ist, die durch den Schalter und die Objektzahl definiert wird. In Abschnitt 5.5 wurde diese Problematik bereits behandelt, in Abschnitt 6.5 wird der Prozess anhand von Beispielen näher erläutert.

## Rotationskörperkomponente



Diese Komponente erzeugt ein weiteres geometrisches Primitiv: einen Rotationskörper. Der Benutzer kann die Silhouette als polygonale Kurve editieren sowie die Auflösung in der Drehrichtung bestimmen. Da die Editierung einen speziellen Dialog erforderlich macht, wurde die Komponente von der Basis-komponente getrennt.

## Blattkomponente

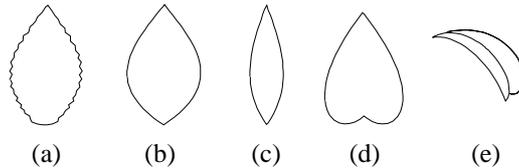


Blattkomponenten werden für alle Arten von Blättern und Blütenblättern verwendet. Die Blattoberfläche wird durch eine Folge von Area-Primitiven er-

zeugt, die anschließend trianguliert werden. Verschiedene Parameter bestimmen das Aussehen des Blattes. Die geometrische Komplexität wird durch die Anzahl Punkte pro Area-Primitiv und die Anzahl Area-Primitive pro Blatt definiert. Ein Parameter bestimmt, inwieweit das Blatt herzförmig gebogen werden soll (siehe Abb. 6.3), ein anderer legt dessen Breite fest.

Die Blattoberfläche kann entlang oder quer zur Hauptachse gebogen werden. Das Profil der Blattoberfläche kann editiert werden, um quer zur Hauptachse auftretende Verformungen (Abb. 6.3(e)) zu erzeugen. Die Umrandung wird durch eine polygonale Kurve bestimmt, der Benutzer kann hiermit gezackte oder irregulär geformte Blätter definieren. Der im Basis-Parametersatz einstellbare Fototropismus kann an dieser Stelle verwendet werden, um die Blattoberseite unabhängig von der Stellung zum Ast dem einfallenden Licht entsprechend auszurichten.

Abbildung 6.3  
Definition verschiedener  
Blattgeometrien: a) gezackter Rand;  
b) Standardform; c) schmales Blatt;  
d) herzförmig; e) profiliert und  
gebogen



Textur → In der Praxis zeigt sich, dass es in den meisten Fällen besser ist, eine einfache Geometrie mit wenigen Dreiecken zu verwenden<sup>1</sup> und sie mit einer Textur in Form einer Fotografie eines realen Blattes zu versehen. Der Grund liegt in der Struktur heutiger Graphikhardware, bei der die Bildrate von der Gesamtzahl erzeugter Dreiecke bestimmt wird, das Auftragen von Texturen kostet hierbei kaum Zeit. Daher ist es besser, wenige Dreiecke mit aufwendigen Texturen zu verwenden anstelle vieler Dreiecke ohne Texturen.



## Hornkomponente

Die von der Hornkomponente erzeugte Geometrie ist die Grundlage für alle Arten von Stielen, Ästen oder Stämmen, kann aber auch zur Herstellung anderer organischer Objekte verwendet werden [?, ?]. Sie wird verschiedentlich als generalisierter Zylinder bezeichnet und entsteht aus der Triangulation von Tube-Primitiven, wie sie in Abb. 6.2 gezeigt wurden. Neben Tube-Primitiven können aber auch andere Primitive verwendet und entlang einer Kurve angeordnet werden.

Die Hornkomponente ermöglicht es ferner, die Geometrie nachfolgender Komponenten zu vervielfältigen. In diesem Fall werden die lokalen Koordinatensysteme der Kindkomponenten an die Orte entlang der Kurve geheftet, die zusammen mit den Tube-Primitiven den Zylinder definiert. In diesem Sinne ist sie sowohl geometrieezeugend als auch Multiplikator, genauso wie die nun folgende Baumkomponente.

<sup>1</sup>Im Extremfall sogar nur mit einem Dreieck, in diesem Fall kann das Blatt aber nicht gebogen werden, was den visuellen Eindruck oft wesentlich verschlechtert.

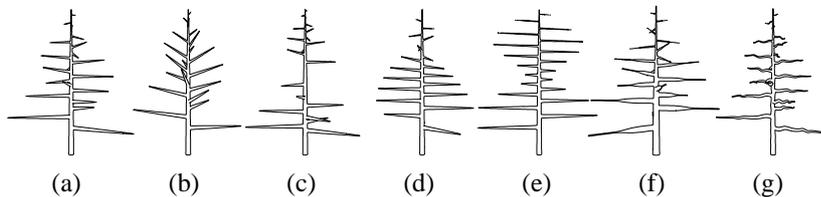
## Baumkomponente

Wie bei der Hornkomponente wird auch durch die Baumkomponente ein generalisierter Zylinder erzeugt<sup>2</sup>. Der Unterschied zur Hornkomponente liegt in der Art, wie nachfolgende Komponenten vervielfältigt werden und über welche Parameter die Form des Zylinders beeinflusst werden kann.

Üblicherweise wird ein Baum als Kaskade von Baumkomponenten aufgebaut oder über eine Rekursion definiert. Im letzteren Fall können allerdings die Parameter nicht individuell für jede Verzweigungsebene eingestellt werden, was oft nachteilig ist.

Die Parameter der Baumkomponente sind in drei Gruppen aufgeteilt. Eine Gruppe bestimmt die Anordnung der verzweigenden Äste. Hier können der Verzweigungswinkel (Abb. 6.4(b)) sowie die Verzweigungscharakteristik eingestellt werden. Diese bestimmt, wie viele Äste pro Längeneinheit entlang des Stammes erzeugt werden (Abb. 6.4(c)), sie wird mittels einer Dichtekurve vom Benutzer eingestellt.

Auch der Deviationswinkel zwischen benachbarten Ästen kann modifiziert werden. Standardmäßig wird hier der Fibonacci-Winkel bzw. der goldene Schnitt verwendet (siehe Abschnitt 3.4).



## Abschnitt 6.2

### KOMPONENTENTYPEN



Abbildung 6.4

Parametervariation in einer Baumkomponente: (a) Standardform; (b) vertikaler Winkel; (c) Verzweigungsdichte; (d) horizontaler Winkel zwischen Ästen; (e) Größe; (f) Dicke entlang Sprossachse; (g) Knorzeligkeit

Weitere Parameter bestimmen die Größe der Äste (Abb. 6.4(e)), deren Dicke entlang der Sprossachse (f), die Knorzeligkeit (g) sowie die Richtungsänderung des Stammes bei der Abzweigung eines Astes. Außerdem kann der Verlauf des Stammes auch direkt über einen Spline definiert werden.

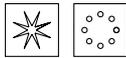
Wie bei den Blättern können auch auf die generalisierten Zylinder des Stammes und der Äste Fototropismen und Gravitropismen einwirken. Hierzu werden die erzeugten Tube-Primitive lokal in Richtung des Lichtfeldes gedreht, was insgesamt eine Biegung der Geometrie erzeugt. Das Aussehen vieler Bäume ist maßgeblich durch diese Parameter bestimmt, der Effekt kann aber auch zum Modellieren von Verformungen durch Windeinwirkung verwendet werden.

Bei der Modellierung von Bäumen tritt ein weiterer entscheidender Unterschied zwischen regelbasierten Systemen und der regelbasierten Objekterzeugung zutage. Während bei einem regelbasierten System das Aussehen einer Pflanze über die Veränderung lokaler Parameter gesteuert wird, werden hier globale Aspekte modelliert. Ein Beispiel ist die Größe der abzweigenden Äste, die als Funktion über die Länge des Stammes angegeben wird. Das Modellieren mit L-Systemen ist an diesem Punkt schwierig, da kleinste lokale Änderun-

← Tropismen

<sup>2</sup>In diesem Falle ist allerdings nur der Einsatz von Tubes als Primitive erlaubt.

gen über mehrere Rekursionsstufen große Unterschiede im Aussehen erzeugen können. Wie schon erwähnt, wird diese Problematik allerdings durch die editierbaren Funktionen aus Abschnitt 5.9 abgemildert.



### Hydra- und Wreathkomponente

Die Hydrakomponente multipliziert alle an sie angehängten Komponenten des p-Graphen und platziert sie in einem sternförmigen Arrangement. Der Benutzer kann hierbei die Anzahl und Größe der Nachfolger, den Öffnungswinkel des Sterns sowie die Verdrehung der Nachfolger bezüglich der Hauptachse als Wertebereiche definieren. In Abb. 6.5(a) sind die Hauptachsen der Vaterkomponente und der multiplizierten Komponenten wiedergegeben.

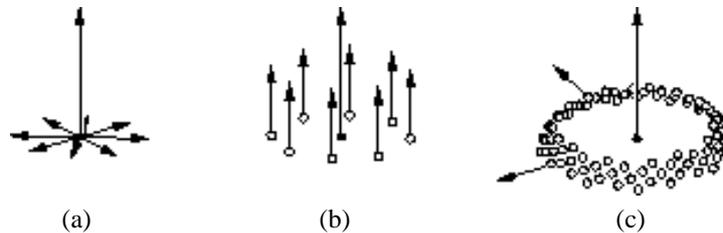
Die Wreathkomponente ordnet ihre Nachfolger auf einem Ring an. Eingabeparameter sind dessen Durchmesser sowie die Anzahl der Nachfolger.



### Phiballkomponente

Werden Komponenten über eine Phiballkomponente multipliziert, so ordnet diese die Geometrien nach der Regel des goldenen Schnittes auf einer Kugeloberfläche an. Der Benutzer stellt die Anzahl und Größe der zu multiplizierenden Komponenten sowie den Öffnungswinkel der Kugel ein. Dieser erlaubt es, die Geometrien auch auf einem Kugelschnitt anzuordnen, was in vielen Fällen verwendet werden kann. Wird nur die Kappe der Kugel benötigt und ein großer Radius eingegeben, können kreisförmige Arrangements erzeugt werden. Dies wird beispielsweise bei den Samen einer Sonnenblume angewendet, um diese in ihrer typischen Anordnung zu erzeugen. Ein seitlicher Schnitt wie in Abb. 6.5(c) kann für Blütenblätter benutzt werden.

Abbildung 6.5  
 Positionen und Hauptachsen  
 multiplizierter Komponenten  
 a) Multiplikation über  
 Hydrakomponente;  
 b) Wreathkomponente;  
 c) Phiballkomponente



Die Berechnung solcher Anordnungen kann auf verschiedene Weise erfolgen (siehe Abschnitt 3.4). Hier wird eine analytische Lösung für den Kugelausschnitt gegeben.

Um die entsprechenden Gleichungen zu erhalten, wird zuerst von einer uniformen Größe der multiplizierten Objekte ausgegangen. Es sollen  $n$  Objekte auf einem Kugelausschnitt mit Rotationsachse  $\vec{z}$  verteilt werden. Die Objekte werden um die Achse  $\vec{z}$  mit dem Winkel

$$\Phi_i = i * d\Phi, \quad d\Phi = 2\pi / ((1 + \sqrt{5})/2)^2$$

verteilt, wobei  $i$  der Nummer des multiplizierten Objektes entspricht. Der Winkel  $d\Phi \approx 137,5^\circ$  ist hierbei der Fibonacci-Winkel.

---

**Abschnitt 6.2**  
KOMPONENTENTYPEN

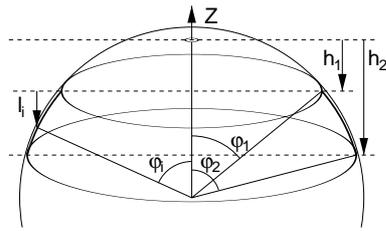


Abbildung 6.6  
Verteilung nach dem goldenen Schnitt  
auf einem Kugelausschnitt

Der vertikale Winkel  $\theta_i$  wird auf folgende Weise berechnet: Zuerst wird die Höhe des Ausschnittes berechnet, der Platz hat für  $n$  Objekte mit Flächenbedarf  $A_E$ . Auf einer Kugel mit Radius  $R$  benötigt der Schnitt  $S(\theta_1, \theta_2)$ ,  $0 \leq \theta_1 < \theta_2 \leq \pi$  eine Fläche von

$$A = 2\pi R h \quad \text{mit} \quad h = R(\cos(\theta_1) - \cos(\theta_2)).$$

Um  $n$  Objekte mit Fläche  $A_n = nA_E$  aufzunehmen, muss eine Kugel mit Radius

$$R = \sqrt{\frac{A_n}{2\pi(\cos(\theta_1) - \cos(\theta_2))}}$$

verwendet werden. Zur Bestimmung des vertikalen Winkels  $\theta_i$  für das  $i$ -te Objekt wird die bisher benötigte Fläche  $A_i = iA_E$  als Funktion der Höhe des Kugelausschnitts angegeben <sup>3</sup>:

$$A_i = 2\pi R h_i \quad \text{bzw.} \quad h_i = \frac{A_i}{2\pi R} = \frac{A_i h}{A_n}.$$

Nun wird  $\theta_i$  berechnet über:

$$\begin{aligned} \theta_i &= \arcsin\left(\frac{R - (h_1 + h_i)}{R}\right) \\ &= \arcsin\left(\frac{R - R(1 - \cos(\theta_1)) - \frac{A_i h}{A_n}}{R}\right) \\ &= \arcsin\left(\cos(\theta_1) - \frac{A_i}{A_n}(\cos(\theta_1) - \cos(\theta_2))\right). \end{aligned}$$

Haben die Objekte unterschiedliche Größe, so sei  $a_j$  die Fläche von Objekt  $j$ . Nun müssen die Variablen  $A_i$  und  $A_n$  in den obigen Gleichungen berechnet werden über

$$A_N = \sum_{j=1}^N a_j \quad \text{und} \quad A_i = \sum_{j=1}^i a_j.$$

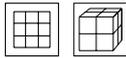
---

<sup>3</sup>Hierbei wird keine Annahme über die Form der Objekte gemacht und vorausgesetzt, dass diese dicht gepackt sind. Dies ist natürlich nur eine Approximation, die unbedeckte Fläche kann in  $A_i$  aber heuristisch eingearbeitet werden.

Für Objekte mit konstanter Größe gilt

$$\theta_i = \arcsin\left(\cos(\theta_1) - \frac{i}{N}(\cos(\theta_1) - \cos(\theta_2))\right).$$

Diese Methode ist im Gegensatz zu den in Abschnitt 4.13 vorgestellten kollisionsbasierten Verfahren ein rein deskriptives Modell mit einer Vorgehensweise ähnlich der von Ridley aus Abschnitt 3.4. Anstatt auf der Ebene den Radius über die Wurzelfunktion wachsen zu lassen, wird hier der vertikale Winkel in geeigneter Weise verändert.



## FFD- und Hyperpatchkomponente

Freiform-Deformation →

Beide Komponenten gehören zu der Gruppe von Komponenten, mit denen die globale Gestalt einer Pflanze beeinflusst wird. Mit beiden lässt sich eine Freiform-Deformation (FFD) definieren, die auf den geometrischen Daten oder aber nur auf den zugrunde liegenden Sprossachsen arbeitet.

Es werden zwei Arten der Definition von Deformationen verwendet. Während bei der FFD-Komponente vom Benutzer Funktionen  $D_x(x)$ ,  $D_y(y)$  und  $D_z(z)$  eingegeben werden müssen, arbeitet die Hyperpatchkomponente mit dreidimensionalen Bézierfunktionen vom Grad eins bis drei. Diese definieren einen Würfel mit einer Reihe von Kontrollpunkten, die der Benutzer verschieben kann. Mit einer entsprechenden graphischen Schnittstelle sind direkt-manipulative Eingriffe möglich.

Die Komponenten können auch als Schalter zum Ausschalten des Einflusses einer Deformation verwendet werden. In diesem Fall wird keine Deformation angegeben, aber ein spezieller Parameter zum Ausschalten verwendet. Auf diese Weise kann im p-Graphen an einer Stelle eine FFD definiert werden, der Effekt aber an einer anderen Stelle bei den Kindkomponenten wieder aufgehoben werden. In Abschnitt 6.5 wird darauf näher eingegangen und an einem Beispiel auch der Unterschied zwischen Deformation der Geometrie und der Sprossachsen gezeigt.



## Weltkomponente

Tropismen →

Die Ausrichtung aller Geometrien und speziell von Blatt- und Baumkomponenten kann durch Foto- und Gravitropismen beeinflusst werden. Diese arbeiten in Bezug auf ein Lichtfeld, welches per Standardeinstellung aus vertikaler Richtung kommt.

Mit der Weltkomponente kann dieses Feld beliebig definiert werden, indem für die  $x$ -,  $y$ - und  $z$ -Komponente des Richtungsvektors jeweils eine Funktion in Abhängigkeit vom Ort angegeben wird. Die Einsatzmöglichkeiten dieser Felder sind vielfältig, wie in den folgenden Beispielen gezeigt werden wird.

## 6.3 Kombination von Komponenten

### Abschnitt 6.3

#### KOMBINATION VON KOMPONENTEN

Zur Herstellung von Pflanzen werden die Komponenten vom Benutzer zum p-Graphen verbunden, er stellt auch die jeweiligen Parameter der Komponenten ein. Um einen p-Graphen aufzubauen, werden die Komponenten über ein Auswahlmü gewählt und mit den schon erzeugten Komponenten verbunden. Hierbei sind drei Arten von Verbindungen (Kanten im p-Graph) möglich:

- **Child-Link:** Hierbei handelt es sich um die Standardverbindung. Die Geometrie der Komponente wird relativ zur Geometrie der Vorgängerkomponente platziert. Verbindungen dieses Typs werden über dünne Linien dargestellt (Abb. 6.7(a)).
- **Branch/Rib-Link:** Die Komponente wird als Verzweigung einer Baumkomponente oder als Rippe einer Hornkomponente multipliziert. Bei anderen Komponenten wird die Verbindung immer als Child-Link interpretiert. Die Darstellung erfolgt über dicke Kanten (siehe Abb. 6.7(b)).
- **Leaf-Link:** Ist die Vaterkomponente Teil einer Rekursion, wird die Geometrie der Kindkomponente nur nach Abbruch der Rekursion über den Rekursionsparameter erzeugt. Sie ist also Blatt des i-Baumes im strukturellen Sinne. Diese Verbindungen werden durch gestrichelte Kanten dargestellt.

In den folgenden Abbildungen werden die Richtungen der Kanten in den p-Graphen weggelassen, da diese immer von der Wurzelkomponente wegzeigen. Zur Darstellung von Rekursionen werden doppelte Kanten verwendet, außerdem wird die erste Komponente zweimal dargestellt, um Anfang und Ende der Rekursion im Graphen anzuzeigen.

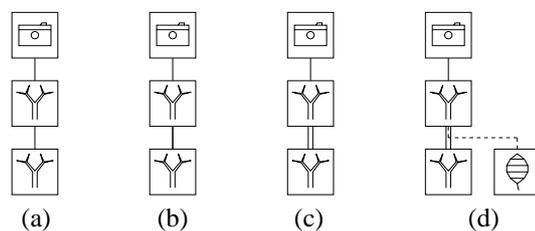


Abbildung 6.7

Verbindungsarten von Komponenten:  
(a) Child-Link; (b) Branch/Rib-Link;  
(c) rekursive Kombination;  
(d) rekursive Kombination mit  
Leaf-Link

Dies erlaubt die Darstellung des p-Graphen als Baum und umgeht auf diese Weise viele der Probleme des Graph-Zeichnens, wodurch die Darstellung anschaulicher wird. Als Manko dieser Darstellung ist anzusehen, dass überkreuzende Rekursionen nicht angezeigt werden können. Da diese aber auch aus anderen Gründen vom System nicht handhabbar sind, werden sie im p-Graphen nicht zugelassen<sup>4</sup>.

<sup>4</sup>In der in Anhang B beschriebenen PC-Variante des Modelliersystems sind Rekursionen generell nicht zugelassen.

## 6.4 Beispiele

Nach der Beschreibung von Komponenten und deren Kombination zu p-Graphen soll nun an drei Beispielen demonstriert werden, wie Pflanzen erzeugt werden. Eine Sonnenblume, ein Rhododendron und eine Kastanie zeigen die vielfältigen Kombinationsmöglichkeiten der Komponenten und gleichzeitig die Mächtigkeit des Gesamtansatzes.

Sonnenblume →

Zur Herstellung der Sonnenblume werden Fotos oder gescannte Vorlagen der Blätter und Blütenblätter benötigt. Diese werden vorab mit einem Bildverarbeitungsprogramm so bearbeitet, dass der Hintergrund transparent wird. Hierzu wird der so genannte Alpha-Kanal gängiger Bildformate verwendet, mit dem sich Transparenzen als eine Art eigene Farbwerte definieren lassen. Üblicherweise werden für Texturen von Blättern nur die Werte null (ganz durchsichtig) oder eins (ganz undurchsichtig) benötigt.

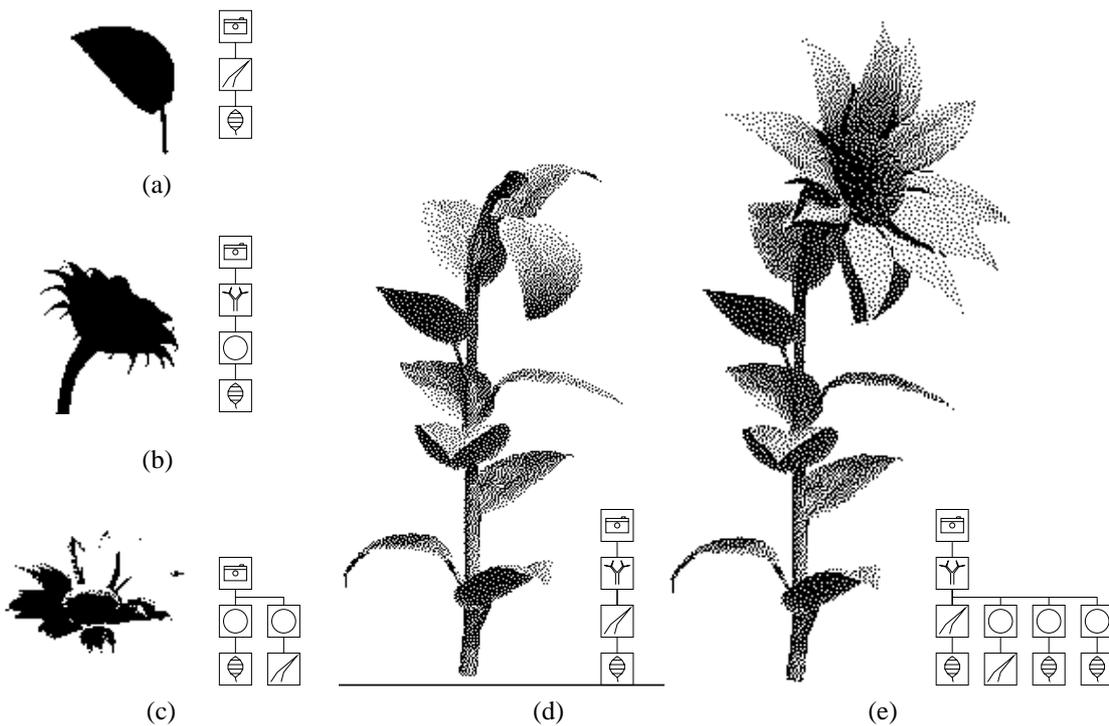


Abbildung 6.8  
 Teile einer Sonnenblume mit  
 entsprechenden p-Graphen: a) Blatt mit  
 Stiel; b) Blütengrund; c) Blüte mit  
 Samen; d) Stiel; e) fertige Pflanze

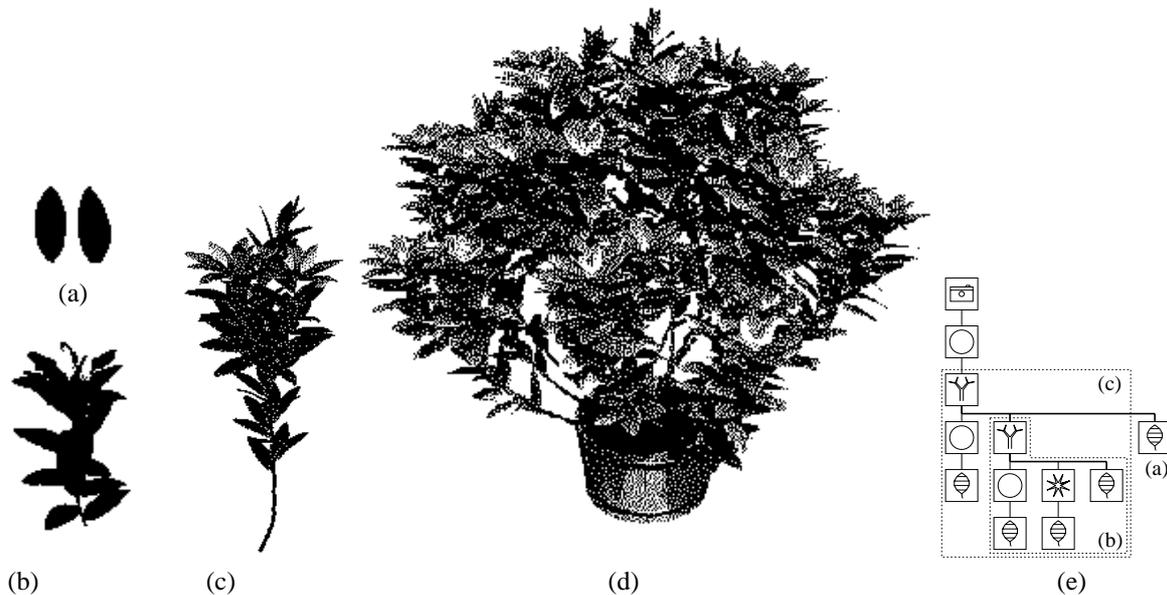
Das Bild des Blattes wird als Textur auf die Geometrie einer Blattkomponente aufgebracht und ein kleiner Stiel wird mit ihr verbunden. In Abb. 6.8(a) ist der entsprechende p-Graph zu sehen, der aus der Kamera, einer Hornkomponente und einer Blattkomponente besteht. Über Krümmungswerte und Transformationen werden die Komponenten so parametrisiert, dass das Aussehen eines natürlichen Blattes entsteht.

Im nächsten Schritt werden die Blätter als Verzweigungen einer Baumkomponente verwendet (Abb. 6.8(d)). Der Oberteil des Stiels wird geöffnet, um den Kopf der Sonnenblume aufzunehmen.

Die Herstellung der Blüte aus den Samen und Blütenblättern zeigen Abb. 6.8(b) und (c). Blütenblätter werden mit einer Phiballkomponente multipliziert, wobei die Parametrisierung so gewählt wird, dass die Blätter kranzförmig angeordnet sind. Auf ähnliche Weise werden kleine Stiele mit einer weiteren Phiballkomponente multipliziert, um die Samen der Blume zu erzeugen. Beide Teilbäume werden mit dem Stiel der Sonnenblume über Child-Links verbunden, als Resultat entsteht nach weiteren Parameteranpassungen die komplette Blume.

Ein Rhododendron ist ein Beispiel für einen mittelgroßen Busch. Die geometrische Komplexität ist mit etwa 30 000 Dreiecken größer als bei der Sonnenblume. Der Fokus liegt hier weniger auf der Verzweigungsstruktur als auf den geometrischen Eigenschaften der Blüten und des Blattwerks.

← Rhododendron



Im ersten Schritt wird wieder ein reales Blatt aufgenommen und als Textur auf eine Blattkomponente aufgebracht (siehe Abb. 6.9(a)), deren p-Graph in Teilbild (e) dargestellt ist. Nun wird ein kleiner Zweig hergestellt, wobei die Spitze separat erzeugt wird, da sich die Blätter um die Blüte auf eine besondere Weise anordnen. Eine Phiballkomponente multipliziert die Blätter und eine Hydrakomponente ordnet die Blütenblätter an (Abb. 6.9(b)). Der p-Graph besteht jetzt aus vier Komponenten: einem Phiball, um die normalen Blätter zu multiplizieren, sowie der Hydrakomponente zur Multiplikation der Blütenblätter. Beide werden über einen Child-Link mit der Baumkomponente verbunden, die den Zweig erzeugen soll.

Abbildung 6.9

Aufbau eines Rhododendron: a) Blatt mit Textur; b) kleiner Zweig mit Blüte; c) Zweig mit verzweigenden Blättern und zwei kleinen Zweigen; d) gesamter Busch; e) p-Graph mit markierten Teilgraphen

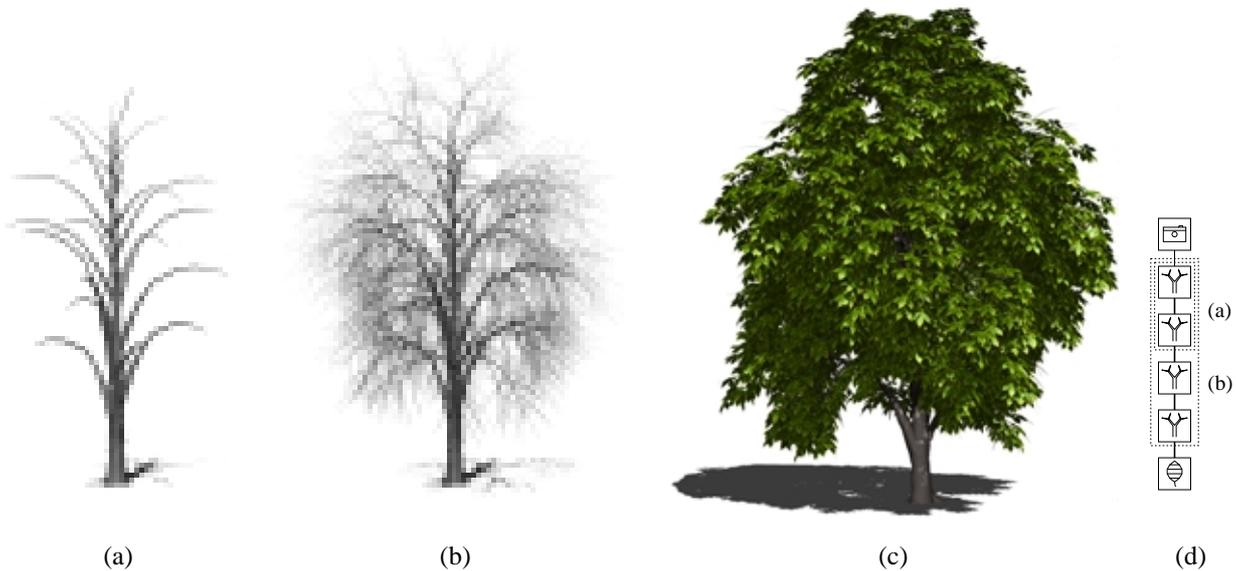
Bei der Herstellung des gesamten Zweiges ergibt sich ein weiteres Problem: Vom Zweig sollen sowohl Blätter abzweigen als auch wenige kleine Zweige. An diesen Abzweigungen sollen aber keine Blätter generiert werden. Diese Art von Ausnahmen wird mit dem Feld bearbeitet, das in der Basiskomponente beschrieben wurde. Hiermit werden einige Blattverzweigungen abgeschaltet, und es werden zusätzlich die Komponenten für die Erzeugung einer kleinen Verzweigung angehängt. Bei der Baumkomponente werden nur für diejenigen Verzweigungen, bei denen die Blätter abgeschaltet wurden, die Zweige als zu erzeugend eingetragen. Auf diese Weise kann der Ast auf zweierlei Arten verzweigen (siehe Abb. 6.9(c)).

Im letzten Schritt werden zwanzig dieser Zweige über eine Phiballkomponente multipliziert, die Komponente wird etwas unterhalb der Erde platziert und die Äste erhalten einen leichten Fototropismus, der sie nach oben biegt (Abb. 6.9(d)).

Abbildung 6.10

Ein Baum wird über eine Sequenz von Baumbausteinen erzeugt: a) zwei Komponenten, nach Parametrisierung; b) vier Komponenten; c) die Blätter werden als Verzweigung an die letzte Komponente gehängt; d) p-Graph mit markierten Teilgraphen

Als drittes Beispiel soll ein Baum erzeugt werden. Wie schon erwähnt, ist die generelle Struktur der p-Graphen für die meisten Bäume ähnlich. Eine Kaskade von Baumkomponenten repräsentiert die Verzweigungsstruktur, während Blätter, Blüten oder Nadeln als Verzweigungen an die unterste Baumkomponente gehängt werden.



Um eine Kastanie als Beispiel für einen mittelgroßen Baum herzustellen, werden wieder Texturen eines Blattes und der Rinde erzeugt und auf die Geometrie einer Blatt- und einer Baumkomponente aufgebracht. Werden zwei Baumkomponenten verbunden und die Parameter entsprechend eingestellt, so ergibt sich ein Baum wie in Abb. 6.10(a). Es wurde ein großer Verzweigungswinkel in Kombination mit einem hohen Gravotropismus eingestellt, um die Form der Hauptäste zu erzeugen. Nun werden zwei weitere Astebenen angehängt

und entsprechend parametrisiert<sup>5</sup>. Im letzten Schritt werden die Blätter mit den Texturen an die Äste gehängt.

Obwohl es einfach klingt, ist das Einstellen der Parameter gerade für die kleineren Äste nicht trivial. Die Parameter sind nicht unabhängig voneinander und erzeugen ein zum Teil komplexes Wechselspiel. Trainierte Benutzer benötigen mehrere Stunden zur Herstellung eines natürlich wirkenden größeren Baumes. Da die Daten jedoch auf einfache Weise wiederverwendet werden können, lassen sich ähnliche Bäume dann sehr schnell erzeugen.



*Abbildung 6.11*  
Ahorn aus insgesamt sechs Ebenen

## 6.5 Modellierung der Gesamtgestalt

In vielen Fällen soll nicht nur eine Pflanzenart, sondern ein Exemplar mit ganz spezifischem Aussehen erzeugt werden. Beispiele hierfür sind Bäume, die an einer Wand wachsen, oder solche, die vom Wind verbogen wurden. In beiden Fällen muss Einfluss auf die Gesamtgestalt genommen werden, um eine charakteristische Form zu erhalten.

Insgesamt fünf Mechanismen zur Modellierung der Gestalt wurden implementiert. Eine Möglichkeit besteht in der Definition von Ausnahmen aus Abschnitt 6.2 innerhalb der Basiskomponente. Die vier anderen sind funktionales Model-

---

<sup>5</sup>Um Daten einzusparen, kann die Geometrie der letzten Astebene für viele Anwendungen auch weggelassen werden.

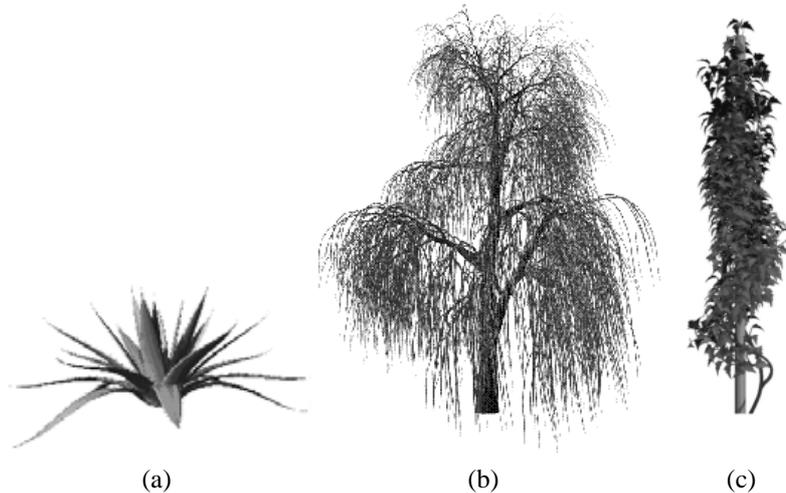
funktionales Modellieren →

lieren, Tropismen, Freiform-Deformationen sowie Beschneidung. Sie werden in den nächsten Abschnitten erläutert.

Wie bereits in der Einleitung erwähnt, werden bei der Herstellung des temporären i-Baumes Komponenten-Prototypen in Instanzen verwandelt, wobei die Parameter, die insbesondere in Multiplikationskomponenten oft als Wertebereiche gespeichert werden, individuell für jede Instanz berechnet werden. Hierfür wird eine Interpolation verwendet, wobei auf das Ergebnis eine Funktion angewendet werden kann.

Abbildung 6.12

Modellieren der Gesamtgestalt:  
a) funktionales Modellieren: Die Krümmung der Blätter wird in Abhängigkeit von der Iterationsnummer und einer Zufallsfunktion berechnet; b) eine Trauerweide wird maßgeblich durch ihren starken Gravitropismus geformt; c) ein Philodendron wird durch ein zylindrisches Feld um einen Stab geführt (Bilder: B. Lintermann)



An dieser Stelle können sowohl Zufallsfunktionen in die Herstellung der Daten eingebracht werden als auch andere funktional spezifizierbare Parameter. Variationen werden durch Addition eines kleinen zufälligen Wertes zur Krümmung oder zur Größe erzeugt; zur funktionalen Spezifikation stehen alle wesentlichen mathematischen Funktionen zur Verfügung.

Um diese Funktionen sinnvoll parametrisieren zu können, ist eine Reihe von Variablen in Kombination mit dem aktuellen, für die jeweilige Instanz interpolierten Wert einsetzbar. So können Position und Orientierung, aber auch Rekursionstiefe und die aktuelle Iterationsnummer der Instanz verwendet werden.

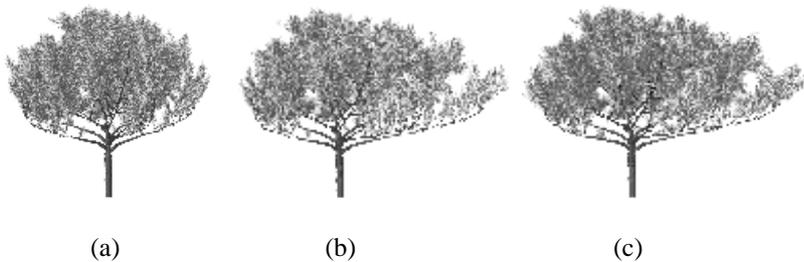
In Abb. 6.12(a) wurde eine Agave unter Verwendung dieser Technik modelliert. Jedes einzelne Blatt ist individuell gekrümmt, wobei die Iterationsnummer und ein Zufallswert den Grad der Krümmung bestimmen. Die zuerst generierten Blätter mit kleiner Iterationsnummer werden von der verwendeten Phi-Ballkomponente am Kopf der Kugel erzeugt und erhalten daher in der Funktion eine kleine Krümmung, für die späteren wird diese kontinuierlich erhöht.

Tropismen →

Tropismen wurden schon bei den Beschreibungen von Basis-, Blatt-, Baum- und Weltkomponente erwähnt. Das Modellieren mit Tropismen kann verwendet werden, um eine Anzahl unterschiedlicher Effekte zu erzeugen. Ein nach unten gerichtetes Gravitationsfeld wurde zur Herstellung der Trauerweide aus Abb. 6.12(b) benutzt. Wird das Feld zylindrisch in einer Weise definiert, dass

außerhalb des Zylinders die Richtungsvektoren nach innen zeigen und innerhalb des Zylinders nach außen, so lässt sich die Verzweigungsstruktur auf eine Zylinderform zwingen. Dies wurde beim Philodendron aus Abb. 6.12(c) angewendet. Weitere Beispiele sind Windeinfluss, für den ein seitliches Feld verwendet wird, und Hindernisse, die durch Felder realisiert werden, die den Eintritt in einen Raumbereich verhindern.

Ein weiterer Weg zur Veränderung der Gesamtgestalt einer Pflanze sind Freiform-Deformationen. Wie bereits bei der Vorstellung der Hyperpatchkomponente angesprochen, wird hierfür eine dreidimensionale Bézierfunktion verwendet, deren Kontrollpunkte vom Benutzer interaktiv verschoben werden.



## Abschnitt 6.5

### MODELLIERUNG DER GESAMTGESTALT

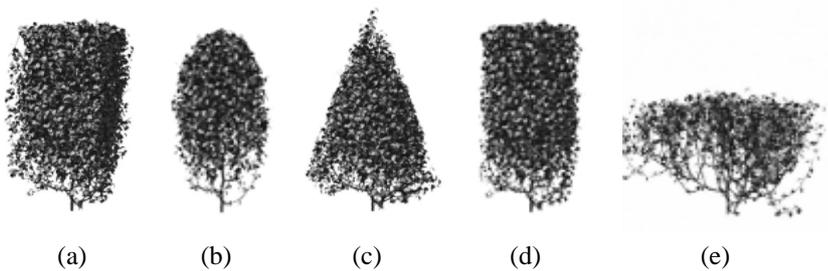
← Freiform-Deformationen

*Abbildung 6.13*  
Freiform-Deformation, angewandt auf einen Nadelbaum: (a) Ausgangsmodell; (b) Deformation der Zweige, die Nadeln bleiben undeformiert; (c) Deformation des ganzen Baumes

Abb. 6.13 zeigt ein Beispiel für die Verwendung von Freiform-Deformationen. Während in Abb. 6.13(c) der gesamte Baum deformiert ist, wurden in Abb. 6.13(b) nur die Äste, nicht aber die Nadeln deformiert. Insbesondere im stark verformten rechten Teil der Pflanze sind die Nadeln natürlicher, da sie keiner Verformung unterworfen wurden.

Durch menschliche Einwirkungen werden vielen Pflanzen Formen gegeben, die sie in der Natur nie annehmen würden. Neben Heckenpflanzen trifft dies für die meisten Bäume zu, die allein stehend eher als Sträucher wachsen würden und nur durch Beschneidung der unteren Äste baumförmigen Wuchs annehmen.

← Beschneidung



*Abbildung 6.14*  
Beschneidung einer Pflanze:  
a) würfelförmig; b) elliptisch;  
c) pyramidenförmig; d) zylindrisch;  
e) halbebenenförmig  
(Bilder B. Lintermann)

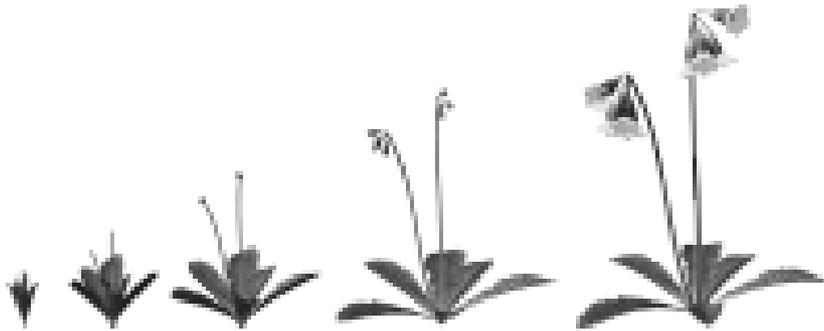
Beschneidungen werden durch einen Parameter realisiert, der in der Baumkomponente das Wachstum steuert. Gleichzeitig werden Basiskomponenten definiert, deren geometrische Primitive als Begrenzungskörper dienen. Bei der Geometrieerzeugung wird dann die Pflanzengeometrie von diesen Körpern begrenzt. In Abb. 6.14 ist zur Verdeutlichung ein Busch auf verschiedene Weise

beschnitten worden. Die Modelle können zur Definition von einfachen Hecken bis hin zu komplexen Renaissance-Gärten verwendet werden.

Je nach verwendetem Modell soll die Pflanze abgeschnitten erscheinen oder aber natürlich innerhalb ihres Begrenzungsvolumens wirken. Es wurden daher zwei Arten von Interaktionen mit dem Begrenzungsvolumen implementiert: Entweder werden die Äste abgeschnitten oder aber der Ast wächst bis zur Begrenzung; dann wird seine Länge ermittelt und in einem zweiten Durchlauf seine Dicke und insbesondere seine Verjüngung an diese Länge angepasst. Ist dies geschehen, wirkt die Pflanze natürlich innerhalb des Begrenzungskörpers.

## 6.6 Animation

In den beiden vorhergehenden Kapiteln wurde das Thema Animation von Pflanzen bereits verschiedentlich angeschnitten. Problematisch ist hierbei immer das Ineinanderspielen von geometrischer und topologischer Änderung beim Wachstum einer Pflanze. Prusinkiewicz verwendet differentielle L-Systeme, deren Modelländerungen durch L-Systeme beschrieben werden. Dazwischen liegende Wachstumsvorgänge werden durch Differentialgleichungen mit relativ komplexen Randbedingungen modelliert. Einige der prozeduralen Ansätze erlauben ebenfalls das Wachstum der erzeugten Bäume. Hier ist dies aufgrund der eingeschränkten Modellpalette (meist einfache Bäume) einfacher zu implementieren.

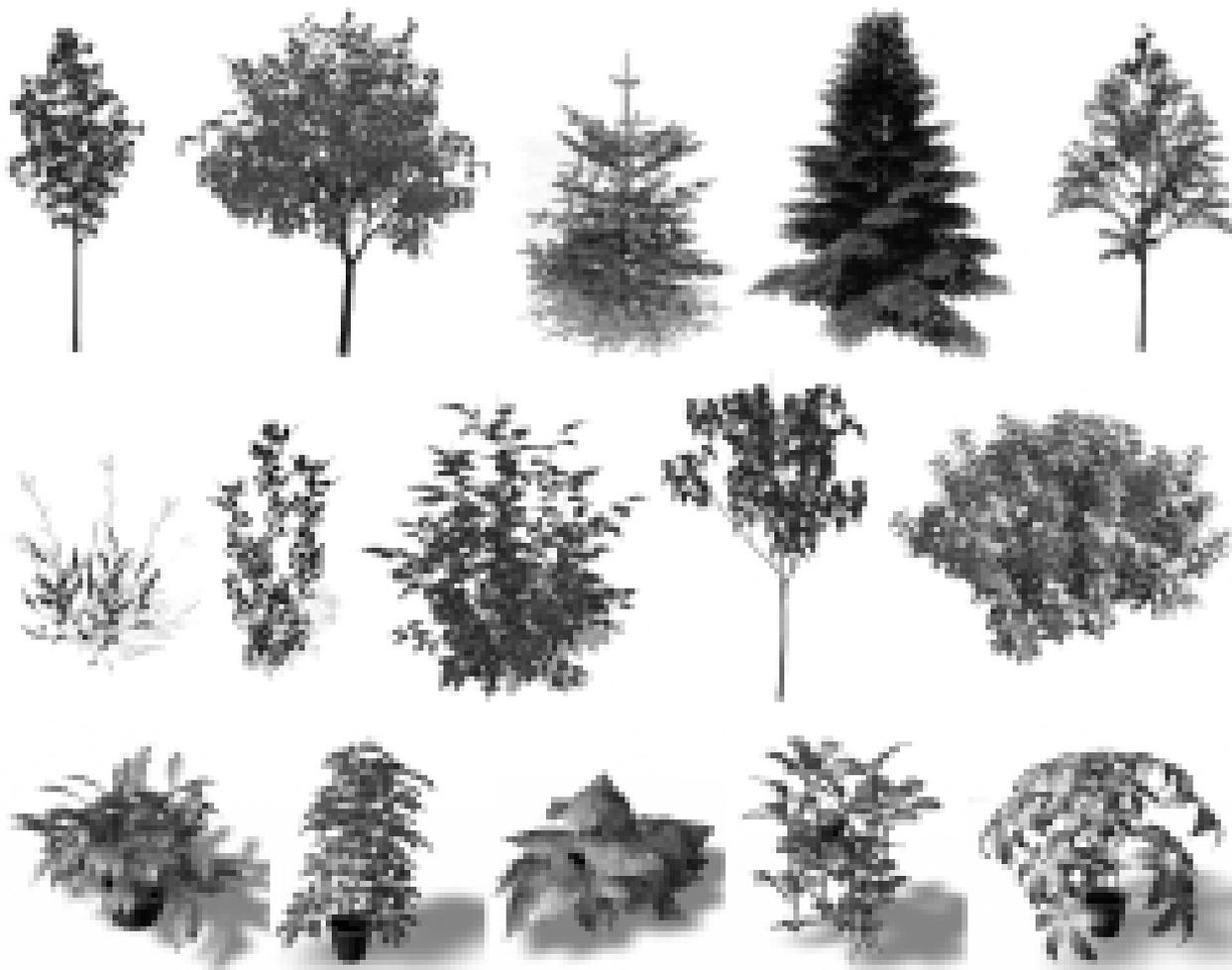


*Abbildung 6.15*  
Wachstum einer Schlüsselblume durch  
Veränderung der Parameterwerte und  
parameterbasiertes Keyframing

Bei der regelbasierten Objekterzeugung macht man sich zunutze, dass ein Großteil der Modelltopologie – so etwa die Frage, wie viele Objekte von einem Multiplikationsbaustein erzeugt werden – nicht in der Topologie des p-Graphen gespeichert ist. Der p-Graph kann daher ohne allzu große Einschränkung als konstant während des Verlaufs einer Animation angenommen werden.

In diesem Fall werden die Parameter, die innerhalb der Komponenten spezifiziert werden, als Parametersatz zu einem Zeitpunkt aufgefasst. Werden unterschiedliche Parametersätze für unterschiedliche Zeitpunkte definiert, so können die Zwischenwerte der einzelnen Parameter durch Interpolation bestimmt wer-

den. Es handelt sich also um ein Keyframing-Konzept auf der Basis von Parameterwerten.



Durch kaskadiertes Keyframing wird lokale Zeit implementiert. Hierbei kann eine Keyframing-Sequenz in eine andere Sequenz eingebunden werden. Wird die Sequenz im Lauf der Animation durch einen Multiplikationsbaustein erzeugt, so startet ihre lokale Zeit mit null und läuft parallel zur globalen Zeit. Auf diese Weise können Vorgänge wie beispielsweise das Öffnen der Blätter in einer Blüte ein einziges Mal spezifiziert werden und zu verschiedenen Zeiten und an verschiedenen Orten unabhängig voneinander ablaufen.

Zur Kaskadierung von Keyframing-Sequenzen wird ein eigener Baustein verwendet, der eine vollständige Sequenz mit Modellbeschreibung und Parameterwerten zu verschiedenen Zeitpunkten speichert. Dieser Baustein wird wie ein normaler Baustein in den p-Graphen der Pflanze eingehängt.

*Abbildung 6.16*  
Synthetische Pflanzen, Oben:  
kleiner Ahorn, kleiner Affenbrotbaum,  
Tanne, Blaufichte, kleine Eiche;  
Mitte: unbekannt, Barbarakraut, Busch,  
Flieder, Busch;  
Unten: Wunderstrauch, Strahlenaralie,  
Buntwurz, Kastanienwein,  
Regenschirmpflanze

Beispielpflanzen →

Mit dem System wurde eine große Vielfalt von Pflanzen erzeugt. Die erzeugten Bäume bestehen aus einer Kaskade von Baumbausteinen und besitzen vier bis sieben Verzweigungsebenen. Die geometrische Komplexität reicht von einigen tausend oder wie im Falle des Ahorns bis zu mehreren Millionen Polygonen pro Pflanze (siehe Abb. 6.11).

Die Zeiten zur Herstellung der Bäume betragen zwischen einer Stunde und einem Tag für den großen Ahorn. Generell nimmt der Modellieraufwand zum Einstellen der Parameter mit der Anzahl der Verzweigungsebenen zu. Sind drei bis vier Ebenen noch ohne großen Aufwand herstellbar, so muss für große Bäume nicht nur wegen der größeren Komplexität der Daten ein längerer Zeitaufwand veranschlagt werden.

Eine Ursache für diesen höheren Aufwand besteht in der Überlagerung vieler Parameter in hohen Verzweigungsebenen, so dass das Verhalten des Systems oft nicht mehr sehr intuitiv ist. Hier müssen geeignete Verfahren entwickelt werden, Parametereinflüsse zu begrenzen. Die geometrische Komplexität der Darstellung hingegen kann über den Ausnahmemechanismus aus Abschnitt 6.2 (Basiskomponente) begrenzt werden. Beispielsweise können alle Äste bis auf einen abgeschaltet werden, dieser wird modelliert, und nur am Schluss werden die anderen wieder aktiviert.

In Abb. 6.16 sind weitere Modelle zu sehen, die mit dem System erzeugt wurden. Die Baummodelle zeigen das Spektrum verschiedenartiger Verzweigungsstrukturen, die Büsche sind weitere Beispiele für Modelle mittlerer Komplexität. In der untersten Zeile von Abb. 6.16 sind diverse Zimmerpflanzen zu sehen, die ebenfalls mit dem System erstellt wurden. Diese Modelle wurden nach der Generierung in ein professionelles Animationssystem übertragen, in dem auch die Bildgenerierung stattfand.

## 6.7 Resümee

Im Gegensatz zu vielen der bisher vorgestellten Methoden erlaubt die regelbasierte Objekterzeugung die schnelle und intuitive Modellierung von Pflanzen. Um diese Aussage zu untermauern, wurde eine kleine Benutzerstudie mit achtzehn Personen durchgeführt.

Nach einer 10-minütigen Einweisung in das System mussten zehn Personen innerhalb von 30 Minuten den Kopf einer Sonnenblume modellieren, acht Personen sollten eine komplette Tulpe herstellen. Hierfür waren 45 Minuten Zeit gegeben. Die Modellieraufgabe der ersten Gruppe erforderte eine stärkere Beschäftigung mit den Multiplikationsbausteinen, die zweite Gruppe wurde eher mit geometrischen Modellierproblemen konfrontiert. Während des Versuchs durften die Personen Fragen an die Betreuer stellen, beide Gruppen erhielten außerdem Fotografien der zu modellierenden Objekte.

In Abb. 6.17 sind die Resultate der anschließenden Befragung angegeben. Alle Werte sind mit 90%-Konfidenzintervallen für eine Normalverteilung angegeben. Trotz der relativ großen Konfidenzintervalle lässt sich aus den Ergebnis-

sen eine intuitive, verständliche und leicht zu editierende Repräsentation einer Pflanze mittels der Komponenten ableiten.

**Abschnitt 6.7**  
RESÜMEE

	1	2	3	4	5
Intuitivität des Modellierprozesses			┌─○─┐		
Intuitivität der Komponenten				┌─○─┐	
Intuitivität der Parameter		┌─○─┐			
Verständlichkeit der Komponenten				┌─○─┐	
Verständlichkeit der Parameter		┌─○─┐			
Verständlichkeit des p-Graphen				┌─○─┐	
Editierung des p-Graphen				┌─○─┐	
Editierung der Parameter		┌─○─┐			
Voraussehbarkeit von Änderungen		┌─○─┐			
Zielorientiertes Modellieren				┌─○─┐	

Abbildung 6.17  
Bewertung untrainierter Benutzer  
(1=schlecht/gering, 5=sehr gut/hoch)

Der p-Graph ist ebenfalls sehr intuitiv in seinem Aufbau. Problematisch ist die Vielzahl und die Verständlichkeit der Parameter. Hier tritt das Problem aller Modellersysteme zutage, in denen Modelliermächtigkeit fast immer mit vielen Parametern einhergeht.

Aufgrund der unterschiedlichen Herangehensweise der verschiedenen Ansätze zur Pflanzenmodellierung aus den letzten drei Kapiteln ist ein direkter Vergleich aller Verfahren relativ schwierig. Die Verfahren werden daher anhand einzelner Kriterien bewertet, die sich in der praktischen Arbeit als wichtig erwiesen haben.

← Vergleich der Verfahren

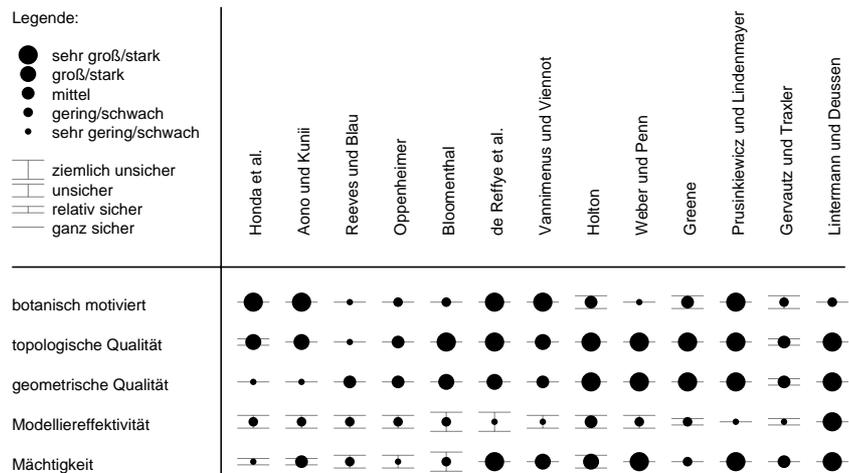
Wie schon in der Einleitung angesprochen, sind die Motive zur Nachbildung von Pflanzenmorphologien vielfältig. Botaniker sind an den zugrunde liegenden Regelmechanismen interessiert, die Geometrie der Modelle dient nur zur Validierung der Erzeugungsmechanismen oder als Datenbasis zur Erforschung weiter gehender Parameter wie etwa der Lichtausnutzung. Für die Computergraphik steht die Form im Vordergrund: „Erlaubt ist, was gut aussieht“. Dieser Satz schließt sogar bewusst botanisch inkorrekte Formen ein, solange damit ein gewünschter visueller Effekt erzielt werden kann. In diesem Sinn soll sowohl die botanische Genauigkeit, also die Frage, inwieweit botanische Effekte nachmodelliert wurden, als auch die topologische und geometrische Qualität der Darstellung bewertet werden.

Ferner spielt in der Computergraphik der Modellieraspekt eine wichtige Rolle. Es ist entscheidend, wie schnell der Benutzer zu einer gewünschten Form gelangt. Ein wichtiger Bewertungspunkt ist also die Modelliereffektivität und außerdem die Mächtigkeit, die angibt, welches Spektrum an Pflanzen erzeugt werden kann. In Tabelle 6.18 sind die Bewertungen zusammengefasst.

Da das Material zu den einzelnen Verfahren in unterschiedlicher Qualität und Quantität vorlag, ist zusätzlich zur jeweiligen Bewertung die Einschätzung bezüglich der Sicherheit der Bewertung aufgeführt. So ist in fast allen Fällen die Modelliereffektivität der Verfahren mit relativ großer Unsicherheit angegeben, da meist nur Papiere vorlagen und kein Experimentieren mit dem jeweiligen Ansatz möglich war. Leider lassen sich die Autoren fast niemals auf die Angabe von Modellierzeiten ein.

Das Gesamtbild zeigt durchaus unterschiedliche Stärken und Schwächen der einzelnen Verfahren, was den Schluss zulässt, dass diese für unterschiedliche Zwecke programmiert wurden und der Fokus sich im Lauf der Zeit gewandelt hat.

*Abbildung 6.18*  
 Bewertung von Verfahren zur  
 Pflanzenmodellierung. Es sind jeweils  
 die Werte und die entsprechenden  
 Intervalle für Sicherheit der Bewertung  
 angegeben.



Die frühen Verfahren von Honda et al. sowie von Aono und Kunii sind botanisch motiviert, legen Wert auf die Modellierung der Topologie und vernachlässigen die geometrischen Aspekte. Die darauf folgenden Arbeiten von Reeves und Blau, Oppenheimer und Bloomenthal stellen im Kontrast dazu die Botanik in den Hintergrund, das realistische Aussehen der Pflanzen wird zum eigentlichen Ziel. De Reffye scheint einen guten Kompromiss zwischen einem botanisch motivierten Vorgehen und der notwendigen Mächtigkeit des Ansatzes gefunden zu haben.

Obwohl die visuellen Ergebnisse von Vannimenus und Viennot nicht wirklich hochqualitativ sind, ist ihre kombinatorische Vorgehensweise methodisch interessant und wiegt daher die Schwächen in der Darstellung auf. Auch Holton verfolgt ein interessantes Prinzip und ist darüber hinaus auch in der Lage, realistische Bilder zu liefern.

Weber und Penn lösen sich bewusst von botanischen Gesetzmäßigkeiten und modellieren die Gesamtgestalt von Pflanzen. Die Ergebnisse sind beeindruckend, insbesondere da auch verschiedene Detaillierungsgrade automatisch erzeugt werden können.

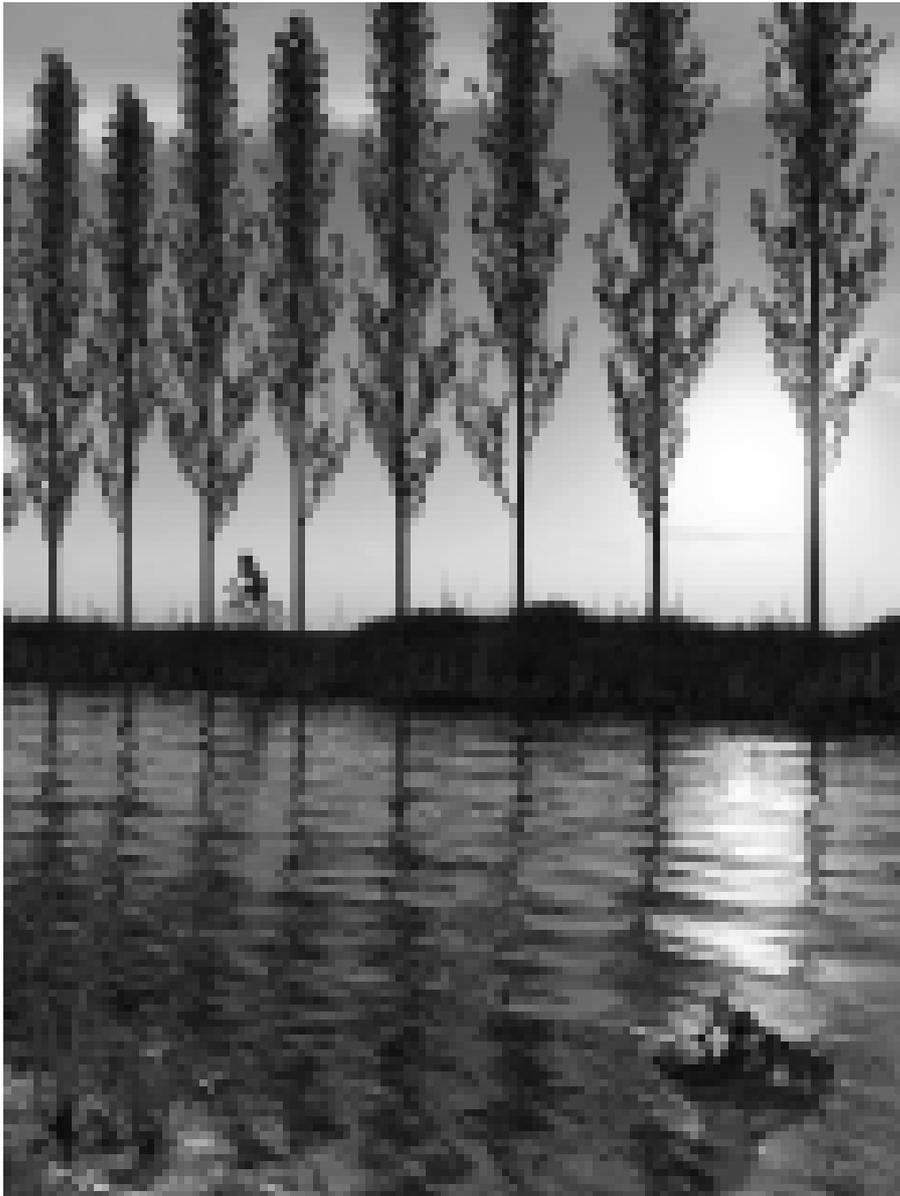
Das sicherlich am weitesten durchdrungene Forschungsfeld im Zusammenhang mit der Modellierung von Pflanzen sind L-Systeme. Mit den beschriebenen Erweiterungen gelang es der Gruppe von Premyslaw Prusinkiewicz an der Universität Calgary, L-Systeme Stück für Stück zu erweitern, so dass diese annähernd alle wichtigen Prozesse des Pflanzenwachstums nachbilden können. Einziges Manko bleibt die weniger intuitive Struktur der Systeme, zumindest was Aufbau und Zusammenspiel von Teilsystemen betrifft. Dies macht viel Erfahrung notwendig, um L-Systeme effizient einsetzen zu können.

Die regelbasierte Objekterzeugung stellt einen Kompromiss zwischen notwendiger Offenheit und Intuitivität einer Modelliermethode dar. Mit dem Ansatz ist nichts modellierbar, was nicht auch in irgendeiner Weise mit parametrischen L-Systemen programmierbar wäre. Durch die Komponenten ist die Modellierung aber dem normalen Benutzer leichter nahe zu bringen, globale und lokale Modelliermethoden erlauben es, die Gestalt einer Pflanze leichter zu beeinflussen. Mit einer kleinen Anzahl verschiedener Bausteine kann vieles modelliert werden, was die Botanik bietet – und vieles darüber hinaus.

---

**Abschnitt 6.7**

## RESÜMEE



*Abbildung 6.19*  
„Pappeln“ von Gilles Tran, hergestellt  
mit POV-Ray unter Verwendung von  
xfrog-Pflanzen



*Abbildung 6.20*  
„2CV“ von Gilles Tran, siehe auch  
<http://www.oyonale.com>



*Abbildung 6.21*  
„Evergreen“ von Gilles Tran

