

# Spezifikation der Simulation der Struktur und Dynamik von Pflanzenbeständen und Tierpopulationen mit sensitiven Wachstumsgrammatiken

Winfried Kurth

Brandenburgische Technische Universität Cottbus  
Institut für Informatik  
Lehrstuhl Praktische Informatik / Grafische Systeme  
Postfach 101344, 03013 Cottbus  
[wk@informatik.tu-cottbus.de](mailto:wk@informatik.tu-cottbus.de)

## Zusammenfassung:

Sensitive Wachstumsgrammatiken sind regelbasierte Systeme (Lindenmayer-Systeme) mit Erweiterungen, die die grafische Interpretation und die Interaktion mit der Umwelt betreffen. "Global sensitive Funktionen" ergänzen den stringbasierten Kontrollfluss durch direkte Rückwirkung der generierten virtuellen 3D-Strukturen auf den Prozess der Regelanwendung. L-Systeme wurden bisher hauptsächlich zur Spezifikation der endogen kontrollierten morphologischen Entwicklung einzelner Pflanzenindividuen eingesetzt. Die Erweiterungen ermöglichen es, Konkurrenzeffekte und, allgemeiner, exogene Einflüsse in regelbasierte Pflanzenmodelle einzubeziehen und den Ansatz auf Bestandes- und Biozönose-Simulationen auszudehnen. Wachstumsgrammatiken erhalten damit als formales Framework für räumliche Simulationen ein ähnliches, wenn nicht noch weitergehendes Anwendungspotenzial wie zelluläre Automaten (CA). Die Möglichkeiten werden demonstriert an einfachen Kronenkonkurrenz-Modellen für Waldbäume und an individuenorientierten Simulationen für Pflanzen-Herbivoren-Wechselwirkungen, einschließlich einer genetischen Weitergabe und Evolution der Nahrungssuche-Strategie der Herbivoren.

## 1 Einleitung

Individuenorientierte Simulationsmodelle, die die Struktur von Pflanzenbeständen sowie teilweise auch Tierpopulationen in ihrer Interaktion mit den Pflanzen abbilden, haben in den letzten Jahren wachsende Aufmerksamkeit gefunden (DeAngelis & Gross 1992; Pretzsch 1998). Wälder beispielsweise sind Ökosysteme mit einem hohen Grad an räumlicher Heterogenität. Die dreidimensionale Struktur setzt hier wichtige Randbedingungen für viele Prozesse, beispielsweise auch für die Ausbreitung von Insekten (Kaitaniemi et al. 1997). In der Forstwirtschaft ist die Veränderung der räumlichen Struktur die wichtigste Form des Eingreifens, um die Entwicklung von Beständen und von einzelnen Bäumen zu steuern (Bormann & Likens 1979).

In Anbetracht der großen Zahl unterschiedlicher Simulationsmodelle in der Literatur wäre es wünschenswert, über eine präzise und einheitliche Sprache für die Modellspezifikation zu verfügen. Einfache Modelle für die Dynamik von Pflanzenbeständen, die die räumliche Struktur vernachlässigen, können zwar oft noch in Form einer oder weniger einfacher Gleichungen ausgedrückt werden. Räumlich explizite Modelle dagegen erfordern meist einen umfangreichen Computer-Quellcode zu ihrer präzisen Spezifikation, welcher dann in der Regel auch mit zahlreichen technischen Konstrukten befrachtet ist, welche vom eigentlichen Kern des Modells ablenken. Umfangreiche Softwaresysteme dieser Art können in ihrem inneren Aufbau nicht leicht von Benutzern nachvollzogen werden, die keine Informatiker sind. Dies kann die fachliche Diskussion über Simulationsmodelle behindern und objektive Vergleiche zwischen verschiedenen Modellansätzen erschweren. Oft treten technische Aspekte wie die Wahl der Programmiersprache oder der Hilfsbibliotheken für die Benutzungsoberflächen-Gestaltung in den Vordergrund, wo es eigentlich um die den Programmen zugrundeliegenden biologischen Prämissen gehen müsste. Wenn andererseits Biologen oder Agronomen ohne adäquate Grundkenntnisse in Softwareentwicklung selbst solche Programme schreiben, werden sehr oft die Regeln des generischen und modularen Software-Entwurfs verletzt (Acock & Reynolds 1997).

Eine Chance, diese Probleme zu überwinden, bietet der Entwurf höherer Modellspezifikations-Sprachen, die an die spezifischen Bedürfnisse der Simulation biologischer Systeme angepasst sind und dem Fachwissenschaftler einen intuitiveren Zugang bieten. Modellspezifikationen, die in einer solchen Sprache abgefasst sind, können im günstigsten Fall unmittelbar von einer generischen Software interpretiert und in Simulationsläufe umgesetzt werden – ohne die Notwendigkeit, den Quellcode der Software jedesmal zu ändern und neu zu kompilieren, wenn einzelne Modellannahmen oder Parameter verändert werden. Nur die in der höheren Sprache angegebenen Spezifikationen müssen verändert werden, und dies kann auch durch „informierte Benutzer“ (Fachwissenschaftler) geschehen, die keine Informatiker sind (Abb. 1). Vergleiche zwischen verschiedenen Modellen werden vereinfacht, wenn die Basis-Software mit ihren technischen Details dieselbe bleibt.

Ein Beispiel-Formalismus, der die Rolle einer solchen „höheren“ Spezifikations-sprache übernehmen könnte (und für einige Anwendungen tatsächlich bereits übernommen hat), ist der der zellulären Automaten (Gerhardt & Schuster 1995). Ein zellulärer Automat wird durch eine Transitionsfunktion beschrieben, die den Zustand einer Zelle in Abhängigkeit vom vorherigen Zustand derselben Zelle und ihrer Nachbarzellen bestimmt. Es gibt mehrere generische Softwaresysteme für diesen Formalismus. Zelluläre Automaten wurden für verschiedene ökologische Anwendungen herangezogen (Ermentrout & Edelstein-Keshet 1993), auch für die Wald-Modellierung. Allerdings lässt sich im Einzelfall nur schwer ausschließen, dass die in einem Zellularautomaten-Modell inhärenten Vorzugsrichtungen, -geschwindigkeiten und die durch die Gitterauflösung vorgegebenen, festen Raum- und Zeitskalen das Modellergebnis auf unerwünschte Weise beeinflussen.

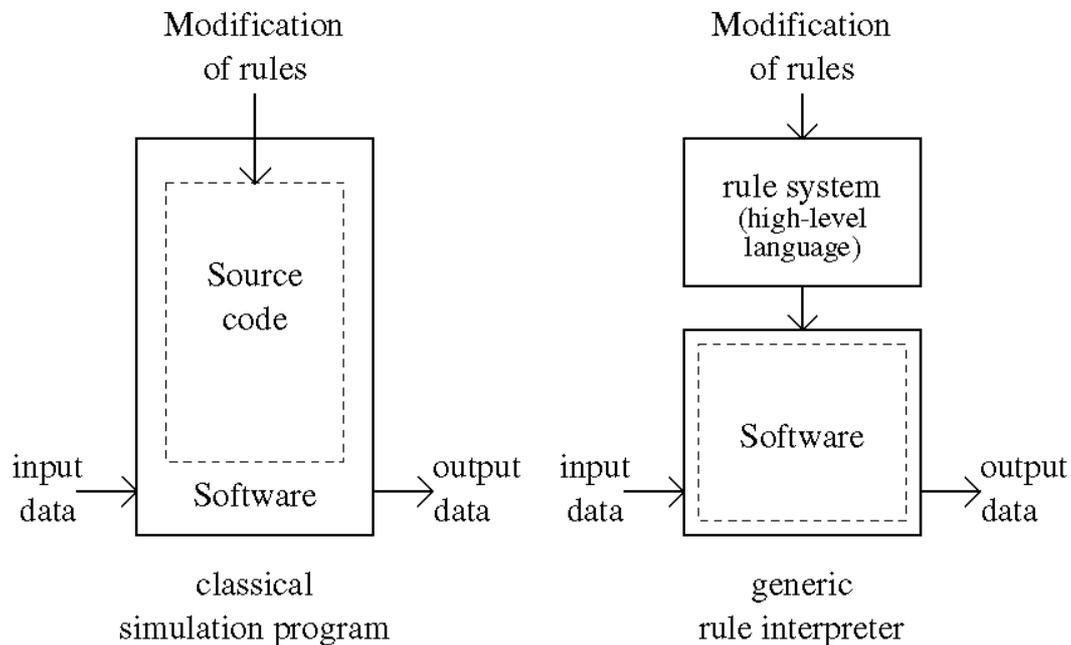


Abbildung 1. Vergleich der Architektur eines klassischen Simulationsmodells (links), wo jede Änderung einen Eingriff in den Quellcode erfordert, mit einer generischen Software für eine höhere Modellspezifikationssprache (wie z.B. erweiterte L-Systeme; rechts).

Eine andere recht allgemeine Spezifikationsprache für biologische Wachstumsprozesse ist die regelbasierte Sprache der L-Systeme (Lindenmayer-Systeme, benannt nach dem Botaniker Aristid Lindenmayer, 1925-1989). Den Informatikern sind diese Regelsysteme aus der Theorie formaler Sprachen geläufig. Sie wurden ursprünglich aber entwickelt, um Wachstumsregeln einfacher, filamentöser Pflanzen zu beschreiben (Lindenmayer 1968). Seitdem wurden zahlreiche Erweiterungen zum Formalismus hinzugefügt (Überblicke bei Prusinkiewicz et al. 1997 und Kurth 1999). Das Hauptanwendungsfeld dieser Sprache blieb aber die Spezifikation der Architekturentwicklung einzelner Pflanzen-Individuen (Prusinkiewicz & Lindenmayer 1990). Im vorliegenden Artikel soll angedeutet werden, welche Möglichkeiten durch sensitive Wachstumsgrammatiken (erweiterte L-Systeme) für die Modellspezifikation auf der Ebene der Pflanzenbestände und Tierpopulationen entstehen – also auf einer höheren Skalenebene als der der Einzelpflanze. Die Architektur des Individuums wird daher in unseren Beispielen mehr oder weniger stark vereinfacht werden; das Interesse konzentriert sich auf Anordnung und Konkurrenz der Einzelpflanzen und ihre Wechselwirkung mit Phytophagen. Wir werden aber auch sehen, dass sich die vereinfachte Geometrie der Individuen relativ leicht und im Rahmen desselben Formalismus „aufbessern“ lässt, z.B. für Zwecke der Visualisierung. – Wie im Falle der Einzelpflanzen-Architektur wird die Entwicklungsdynamik, die das Modell simuliert, ein Ergebnis lokaler Regelanwendungen sein: Das heißt, es wird keine globale Regel für Selbstausdünnung oder für andere aggregierte Bestandesparameter in die

Modellspezifikationen hineingesteckt. Die Dynamik resultiert als emergente Eigenschaft aus der vielfachen Anwendung lokaler Regeln (vgl. Breckling 1996).

## 2 Sensitive Wachstumsgrammatiken

Ein L-System besteht aus einer Menge von Symbolen, einem Startsymbol  $\alpha$  und einer Menge von Ersetzungsregeln der Form „Symbol  $\rightarrow$  String aus Symbolen“. Zusätzlich wird als Semantik eine geometrische Interpretation der Strings definiert, welche diese Symbol-Strings in 3-dimensionale Strukturen übersetzt. Üblicherweise verwendet man für diese Interpretation die Konventionen der sogenannten Turtle-Geometrie (Schildkröten-Geometrie; Abelson & diSessa 1982): Einige der Symbole werden als Kommandos für ein virtuelles Zeichen- oder Konstruktionsgerät, die „Turtle“, verwendet. Die Turtle ist in der Lage, sich zu bewegen (auf das Kommando „f“), zylindrische Elemente bei der Bewegung zu erzeugen (Kommando „F“), sich zu drehen (Kommando „RU“) oder Parameter ihres internen Zustands, welche die als nächstes erzeugten Elemente beeinflussen, zu ändern (z.B. Kommandos „L“ für die Länge, „D“ für den Durchmesser des nächsten Strukturelements; siehe Kurth (1994, 1999) für weitere Details). Trifft die Turtle im Befehls-String auf eine geöffnete eckige Klammer „[“, so wird der aktuelle Zustand mit allen Parameterwerten (einschließlich der Position und Orientierung) auf einem Stack gespeichert und bei Erreichen der zugehörigen schließenden Klammer „]“ dort wieder entnommen. Hierdurch ist es möglich, verzweigte Strukturen zu spezifizieren.

L-Systeme sind zeitlich diskrete Systeme. Alle Regeln werden parallel auf Anwendbarkeit auf die Symbole eines zum Zeitpunkt  $t$  bestehenden Strings geprüft und ggf. angewandt. Das Ergebnis dieser parallelen Ersetzung ist ein neuer String am Zeitpunkt  $t+1$ . Man verwendet die Konvention, dass Symbole, auf die keine der Regeln anwendbar sind, unverändert übernommen werden. Dieser Rewriting-Prozess wird normalerweise mehrfach wiederholt, und man erhält eine (potenziell unendliche) Folge von Strings  $s_0, s_1, s_2, \dots$ , worin  $s_{t+1}$  aus  $s_t$  durch Regelanwendung zustandekam und  $s_0 = \alpha$  ist. Für die Ersetzung ist es dabei unerheblich, ob die zu ersetzenden Symbole eine Bedeutung als Turtle-Kommandos haben oder lediglich als Hilfszeichen für die weitere Entwicklung der Struktur dienen. Der String  $s_t$  wird dann als Befehlsfolge für die Turtle interpretiert (unter Vernachlässigung aller Symbole, die keine Bedeutung für die Turtle haben). Es ergibt sich eine (nicht notwendig zusammenhängende) geometrische Struktur, die ein Modell einer Pflanze oder einer Population zum Zeitpunkt  $t$  repräsentieren kann. Die zeitliche Schrittweite kann einem Jahr oder kleineren oder auch größeren Zeitabständen entsprechen. Das Ergebnis ist also eine Entwicklungssequenz von geometrischen Strukturen. Abb. 2 demonstriert dieses Modellprinzip an einem sehr einfachen, klassischen L-System, das das Wachstum einer Verzweigungsstruktur beschreibt. Das Startsymbol ist dort „a“. Das L-System enthält nur zwei Regeln:  $a \rightarrow F [ RU45 b ] a$  und  $b \rightarrow F b$ . Die Symbole  $a$  und  $b$  sollen für die apikalen Knospen der Hauptachse bzw. der Seitenzweige stehen; sie würden normalerweise als reine Hilfssymbole nicht von der

Turtle in geometrische Einheiten umgesetzt. Zur Verdeutlichung des botanischen Sinngehalts des Modells haben wir in der Grafik aber die entsprechenden Knospen durch Ovale visuell dargestellt. Eine solche visuelle Interpretation kann durch eine Erweiterung des Formalismus realisiert werden, nämlich durch einen zusätzlichen Typ von Regeln, sogenannte Interpretationsregeln (Kurth 1994). – Horizontale Pfeile stehen in Abb. 2 für den Prozess der Regelanwendung, vertikale Pfeile für die Interpretation der Strings durch die Turtle. Das L-System enthält ein Symbol mit Parameter: RU, wobei die darauffolgende Zahl den Drehwinkel in Grad angibt. Reellwertige Parameter können auch an andere Symbole, wie  $a$  oder  $b$ , angefügt werden (Prusinkiewicz & Lindenmayer 1990). – Einige andere Erweiterungen des klassischen L-System-Formalismus werden in den folgenden Beispielen an den Stellen vorgestellt, wo sie benötigt werden. Zusammen führten diese Erweiterungen zum Begriff der „stochastischen, sensitiven Wachstumsgrammatiken“ (Kurth 1994).

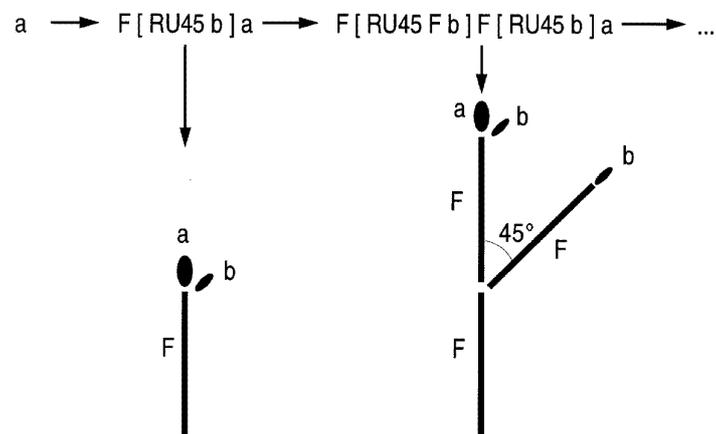


Abbildung 2. Entwicklung geometrischer Strukturen, spezifiziert durch die Wachstumsgrammatik  $a \rightarrow F [RU45 b] a$ ,  $b \rightarrow F b$  (siehe Text).

### 3 Modelle der Dynamik von Baumkronen unter Konkurrenz

Klassische L-Systeme, wie durch das obige Beispiel repräsentiert, sind dazu geeignet, die endogen festgelegten Muster der Ontogenese pflanzlicher Strukturen wiederzugeben. Der Informationsfluss verläuft ausschließlich von einem Symbol zu seinen Nachfolgersymbolen bei der Regelanwendung (*lineage control*). Frühzeitig wurden *kontextsensitive Regeln* eingeführt, um auch eine Informationsweitergabe zwischen topologisch benachbarten Segmenten der dreidimensionalen Struktur abzubilden (siehe Prusinkiewicz & Lindenmayer 1990 für eine genaue Beschreibung des Formalismus). Damit wurde es möglich, beispielsweise auch die Dynamik von Konzentrationsgradienten innerhalb von Verzweigungssystemen und eine hormonale Steuerung von morphogenetischen Vorgängen zu modellieren. Allerdings reichen

kontextsensitive Regeln nicht aus, um *Fernwirkungen* zwischen Pflanzenteilen oder zwischen Pflanzen und anderen Objekten darzustellen. Hierfür wurden *global-sensitive L-Systeme* (Kurth 1994) bzw. (unabhängig, aber mit ganz ähnlichem Ergebnis) "environmentally-sensitive L-systems" (Prusinkiewicz et al. 1994) eingeführt. Das Prinzip der global-sensitiven L-Systeme besteht darin, vor der Regelanwendung Bedingungen zu überprüfen, die Funktionen oder Variablen aufrufen, deren Werte von der gesamten bis dahin erzeugten geometrischen Struktur (und eventuellen externen Steuergrößen) abhängen können. Wir lösen uns damit vom Prinzip einer ausschließlich auf Strings operierenden Grammatik und nähern uns den Graph-Grammatiken an (Rozenberg 1997). Solche global-sensitiven Regeln können benutzt werden, um Kollisionen zu entdecken oder Konkurrenzeffekte durch räumliche Bedrängung oder Überschattung zu modellieren. In dem zweidimensionalen Beispiel in Abbildung 3 wird eine sensitive Funktion verwendet, die einen Winkelsektor oberhalb der jeweils betrachteten Knospe überprüft. Wenn sich in diesem Sektor potenziell „beschattende“ Objekte befinden, treibt die Knospe nicht aus, d.h. eine Wachstumsregel wird nicht auf das betreffende Symbol angewandt. Weitere sensitive Regeln sorgen dafür, dass Seitenzweige, die nicht weiterwachsen, nach einem vorgegebenen Zeitintervall abgeworfen werden, und dass eine Pflanze, die keine „lebenden“ Seitenzweige mehr hat, umstürzt. Im Beispiel-Simulationslauf in Abb. 3 wird die mittlere Pflanze von ihren Konkurrenten überschattet. Das verwendete L-System ist bei Kurth (1999) ausführlich dokumentiert. Es enthält auch eine stochastische Komponente, nämlich eine Variation der Abstände der Seitenzweige entlang der Hauptachse der Pflanze. Nur hierdurch kommt das unregelmäßige Aussehen der modellierten Pflanzen zustande.

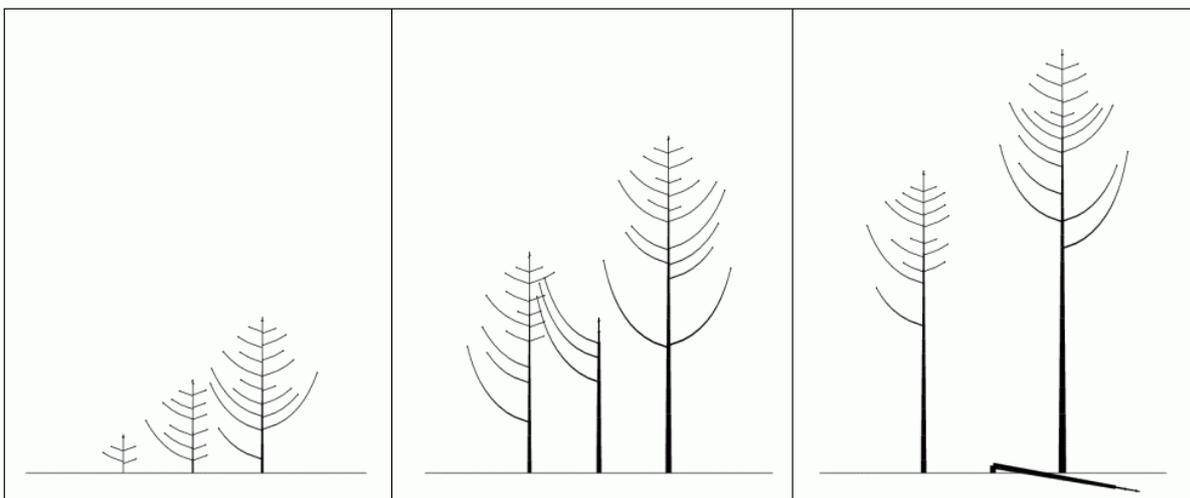


Abbildung 3. Ergebnis einer global-sensitiven Wachstumsgrammatik nach 12, 25 und 31 Schritten. Zum Weiterwachsen einer Achse muss ein  $40^\circ$ -Winkelsektor oberhalb der Knospe (an der Spitze) frei von Objekten sein. Die mittlere Pflanze erliegt der Konkurrenz ihrer beiden Nachbarn.

Entsprechende Lichtkonkurrenz-Grammatiken wurden auch für dreidimensionale Modellpflanzen entwickelt (siehe Kurth 1999). Für eine Modellierung großer Bestände ist dieser Ansatz aber noch zu aufwändig. Im Forstbereich wird meistens nicht eine detaillierte 3D-Aufnahme der Kronenstruktur einzelner Individuen durchgeführt, sondern man erhebt im Feld höchstens die Ausdehnung der Krone in vorgegebenen Himmelsrichtungen (meistens 8 an der Zahl) durch Ablotung. Entsprechend verwenden ad hoc entwickelte Modelle der Strukturodynamik in Waldbeständen (die also nicht auf Grammatiken zurückgreifen, sondern unmittelbar in Allzweck-Programmiersprachen implementiert sind) vereinfachte zweidimensionale Projektionen der Baumkronen als Darstellung der Individuen. Die Dynamik der horizontalen Kronenausdehnung wird dann in Abhängigkeit von den Abständen zu den Nachbarbäumen modelliert (Pretzsch 1992).

Zur Umsetzung dieses Ansatzes in einer sensitiven Wachstumsgrammatik wird eine sensitive Funktion  $sf$  benutzt, die den Abstand zum nächsten Strukturelement innerhalb eines  $45^\circ$ -Winkelsektors entlang der Richtung des betrachteten Kronenradius zurückgibt. Wenn diese Distanz groß genug, d.h. größer als ein vorgegebener Schwellenwert  $ds$  ist, geht der betrachtete Kronenradius über in einen Zustand „wachsend“ (Parameter  $c = 1$ ) und vergrößert seine Länge von  $r$  auf  $r+1$ :

$$(sf(22.5) > ds) s(c, r) \rightarrow s(1, r+1),$$

andernfalls wird ein Zustand des „Schrumpfens“ ( $c = 2$ ) angenommen und der Radius wird um 0,3 Längeneinheiten verkürzt:

$$s(c, r) \rightarrow s(2, r-0.3).$$

Die zweite Regel wird in all den Fällen angewandt, wenn die erste nicht anwendbar ist. Im folgenden Beispiel werden darüberhinaus für jeden Baum die „schrumpfenden“ Kronenradien gezählt (in einer lokalen Variablen  $x$ , die mit jedem Baum assoziiert ist); wenn diese Anzahl 5 (von 8) übersteigt, wird der Baum aufgrund von zu starker Unterdrückung durch seine Nachbarn als tot angesehen und entfernt. Letzteres geschieht durch die Transformation eines Hilfssymbols in einen cut-Operator (durch das fest vorgegebene Symbol „%“ dargestellt), welcher die Turtle-Interpretation nachfolgender Kommandos (bis zum Verlassen der aktuellen Klammer-Ebene) unterbindet. Tabelle 1 zeigt die komplette Grammatik, wie sie von der generischen Software GROGRA (Growth Grammar Interpreter; Kurth 1994) eingelesen und interpretiert werden kann. Einige der verwendeten Konstrukte wurden noch nicht erklärt: Mit „\const“ werden Konstanten, mit „\var“ Variablen deklariert; Variablen vom Typ „uniform“ sind Zufallsvariablen (auch z.B. „normal“, „poisson“ oder „binomial“ wären hier als Spezifikation möglich), die zur Laufzeit mittels eines internen Pseudozufallszahlengenerators erzeugt werden. Mit „&(15) < s >“ wird der String  $s$  fünfzehn Mal wiederholt. Die mit dem Typ „index“ deklarierte Variable  $i$  darf dabei innerhalb des Strings  $s$  als Laufvariable in arithmetischen Ausdrücken verwendet werden. „sum“ ist ein Operator, der innerhalb der generierten Struktur über alle Tochter-Zweige summiert und dabei eine Bedingung (hier:  $col = 2$ , also eine

Farb-Einschränkung) berücksichtigt. „ $A(x, \text{Ausdruck})$ “ ist ein Zuweisungs- (assignment-) Operator, der den spezifizierten Ausdruck auswertet und das Ergebnis der lokalen Variablen  $x$  zuweist. (Man beachte, dass jedes Pflanzenindividuum über ein eigenes  $x$  verfügt.) Die letzte Regel schließlich ist eine Interpretationsregel und daher mit einem Doppelpfeil statt des einfachen Pfeils versehen.

Tabelle 1. Eine sensitive Wachstumsgrammtik für die Kronenradien-Dynamik in gleichaltrigen Waldbeständen.

```

\const ds 3,           /* threshold distance for competition */
\const dp 12,          /* distance between the planting positions */
\const ang 22.5,       /* opening angle of sensitive cone */
\var rr uniform 0 360, /* random rotation */
\var i index,
\var sf function 9 1,  /* competition function */
\var x local 0,        /* count of shrinking crown radii
                       of each tree */

\var col color,
* → &(15) < [ &(12) < [ F(0) K(x) tree ] RU90 f(dp)
      RU-90 > ] RL-90 f(dp) RL90 >,
tree → A(x, sum(col=2, 1)) cut RU90 RL(rr)
      &(8) < [ RL(i*45) s(1, 1) ] >,
(sf(ang) > ds) s(c, r) → s(1, r+1),
s(c, r) → s(2, r-0.3),
(x >= 6) cut → %,
s(c, r) ⇒ Pl(c) F(r)

```

Abbildung 4 zeigt das Ergebnis der Anwendung dieser Grammatik auf einen regelmäßig „angepflanzten“ Bestand von 180 Bäumen. Jeder Baum wird durch seine 8 Kronenradien repräsentiert. Man beachte, dass die initiale Ausrichtung der Radien in der Ebene die einzige stochastische Komponente in dem Modell ist (gleichverteilter Winkel  $rr$ ). Dies genügt, um eine zufällige Verteilung der Lücken im älteren Bestand zu induzieren.

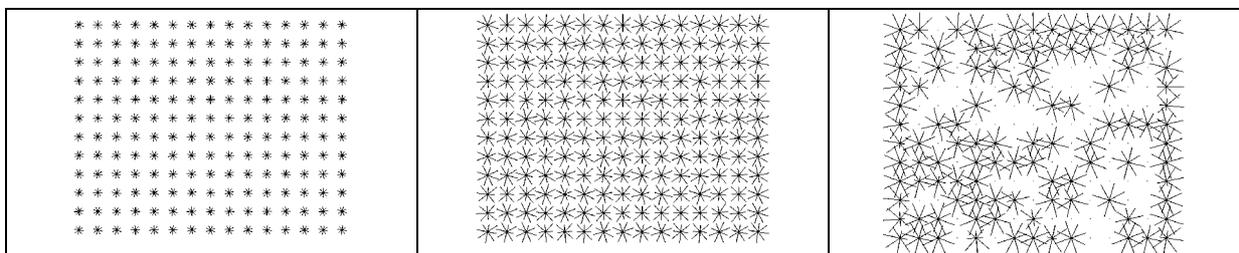


Abbildung 4. Ergebnis der in Tabelle 1 gezeigten sensitiven Grammatik nach 4, 8 und 12 Schritten. Im rechts gezeigten Zustand sind mehrere Bäume bereits aufgrund zu vieler schrumpfender Radien durch Konkurrenz eingegangen, und andere Bäume beginnen die entstandenen Lücken mit ihren Kronenradien zu erobern (aus Kurth 1999).

Zum Zweck der Visualisierung können für die vereinfachten Pflanzen eines solchen Bestandesmodells im Nachhinein wieder detaillierte morphologische Modelle eingesetzt werden, wie sie für verschiedene Baumarten mit Wachstumsgrammatiken entwickelt wurden (Kurth 1999). Es genügt dabei, nur eines oder wenige Individuen detailliert zu modellieren und deren Kopien auf die Positionen im Bestandesmodell zu verteilen ("object instancing", vgl. Deussen et al. 1998). Abbildung 5 zeigt mit einem L-System produzierte Fichten an einem Hang, die sämtlich Kopien desselben detaillierten Baumes sind; die Instanzen unterscheiden sich lediglich in Position, Orientierung und Skalierung.

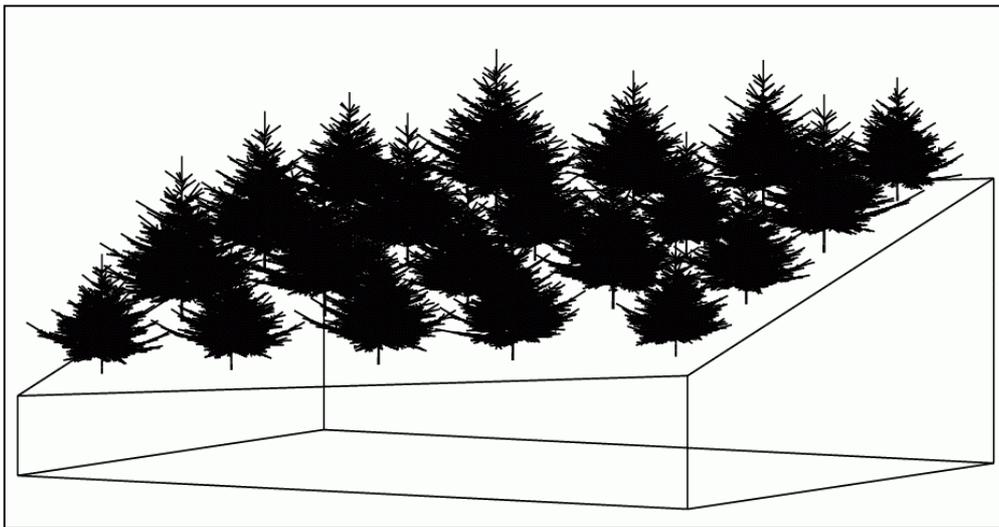


Abbildung 5. Mit einer Wachstumsgrammatik unter Verwendung des "object"-Konstrukts erzeugter Fichtenbestand (aus Kurth 1999). Alle Individuen sind Kopien desselben Pflanzen-Prototyps.

#### 4 Ein Modell für interagierende Pflanzen- und Tierpopulationen

Das folgende Beispiel orientiert sich an einem Modell von Breckling (1990), der aber keine Grammatiken zur Spezifikation verwendete. In weiteren Einzelheiten ist es bei Kurth (2000) erläutert. Im Gegensatz zum obigen Kronenkonkurrenz-Modell wird nun die natürliche Reproduktion der Pflanzen mitberücksichtigt, und zwar in Form eines integrierten Samenausbreitungs-Modells. Eine Pflanze wird durch das Symbol  $p$  repräsentiert und hat zwei Parameter, das Alter  $t$  und den Radius  $r$ . Im Vergleich zum vorangehenden Beispiel werden die Individuen jetzt noch weiter vereinfacht und nur noch als Kreise mit Radius  $r$  dargestellt. Der Radius wird als proportional zum Energiegehalt der Pflanze angenommen. Es ist kein detailliertes Modell des Energiestoffwechsels intendiert; für die Mortalität wird auf heuristische Regeln zurückgegriffen: Eine Pflanze stirbt, wenn sie ihr maximales Alter erreicht hat, oder wenn ihr Energiegehalt unter Null fällt ( $r < 0$ ), oder wenn sie von einer größeren Pflanze überschattet wird (d.h. wenn ihr Mittelpunkt innerhalb des Kronenradius einer größeren Pflanze liegt). Um die letztgenannte Bedingung zu testen, wird wieder eine sensitive

Funktion in der Grammatik aufgerufen. Wenn sie nicht stirbt, wächst eine Pflanze in jedem Zeitschritt um einen festen Betrag. Ferner gibt es eine Regel für die Vermehrung. Diese wird angewandt, wenn die Pflanze eine von zwei festen, vorgegebenen Altersstufen erreicht hat und wenn ihr Radius zugleich über einem Schwellenwert liegt. Die Anzahl der Samen, die produziert und in zufälligen Abständen und in zufällige Richtungen verteilt werden, ist proportional zum Radius (bzw. Energiegehalt) der Pflanze. Die Positionierung der Samen (symbolisiert durch  $p(0, 0)$ ) erfolgt einfach durch den  $f$ -Operator. Insgesamt kommt man mit 5 Regeln für das Verhalten der Pflanzen aus, zuzüglich einer Interpretationsregel für ihre grafische Darstellung. Abbildung 6 fasst das Verhalten der Pflanzen grafisch zusammen.

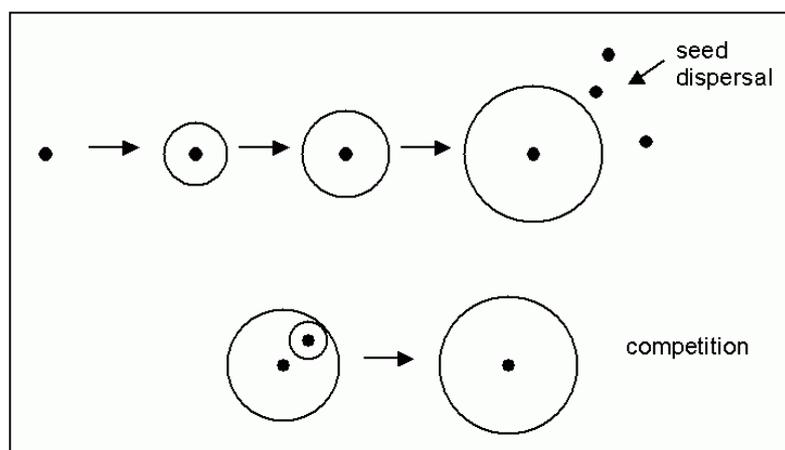


Abbildung 6. Verhalten der Pflanzen in der Phytophagen-Wachstumsgrammatik.

Schon mit dieser einfachen Grammatik erhält man reichhaltige räumliche Strukturen. Es bilden sich Cluster kleinerer Pflanzen und große Lücken, die später wieder besiedelt werden. Abbildung 7 zeigt als Momentaufnahme aus einem Simulationslauf den 54. Entwicklungsschritt; gestartet worden war diese Simulation mit einer einzelnen Pflanze, die ungefähr in der Mitte des Bildes positioniert war.

Tiere beziehen ihre Energie zum Leben von den Pflanzen. Ein Tier wird in der Grammatik durch  $a(t, e)$  symbolisiert, wobei  $t$  wieder das Alter und  $e$  die eigene Energiereserve ist. Grafisch werden die Tiere durch kleine Kreise repräsentiert (in einer anderen Farbe als die Pflanzen). Um die Regel für die Vermehrung einfach zu halten, wird hier angenommen, dass die Vermehrung durch Teilung erfolgt. Ein Tier stirbt, wenn sein Energievorrat kleiner oder gleich 0 wird, und vermehrt sich, wenn dieser Energievorrat größer als ein Schwellenwert wird. In jedem Zeitschritt macht ein Tier, welches mit keiner Pflanze in Kontakt ist und sich nicht gerade vermehrt, einen „Suchschritt“ und verbraucht einen festen Energiebetrag:

$$a(t, e) \rightarrow RH(rr) + f(w) - a(t+1, e - \text{respi}),$$

wobei + und – Kurzkommandos für Rotationen und  $RH(rr)$  eine Zufallsdrehung um die vertikale Achse bezeichnen. „*resp*“ ist der Respirationsbetrag (Energieverbrauch),  $w$  die Schrittweite der Eigenbewegung.

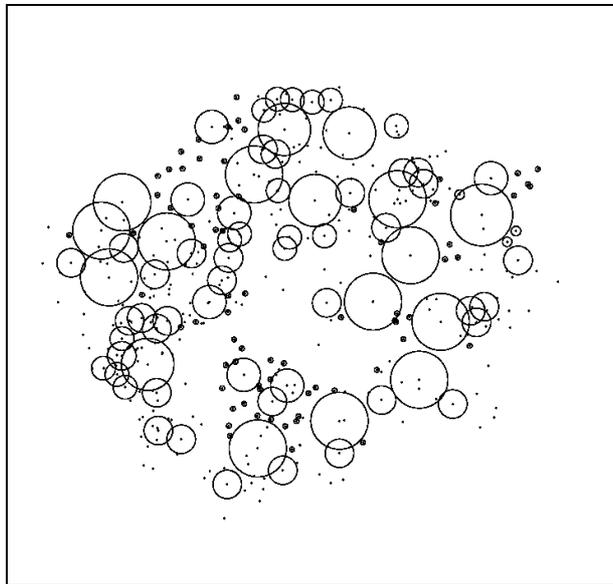


Abbildung 7. Ungestörte Bestandesentwicklung (ohne Pflanzenfresser) nach 54 Zeitschritten. Die Interaktion zwischen Pflanzenindividuen gehorcht der im Text und in Abb. 6 beschriebenen Überschattungsregel. Punkte ohne umgebende Kreise sind Samen, die im nächsten Zeitschritt „austreiben“ werden.

Wenn ein Tier in Kontakt mit einer Pflanze gekommen ist (d.h. sich innerhalb des Kronenradius aufhält, was von einer sensitiven Regel geprüft wird), verringert sich seine Bewegungsdistanz und es entnimmt der Pflanze in jedem Zeitschritt einen festen Energiebetrag, der dem eigenen Energievorrat hinzugefügt wird. Der Radius der Wirtspflanze schrumpft um diesen Betrag. Abbildung 8 fasst das Verhalten der Tiere grafisch zusammen.

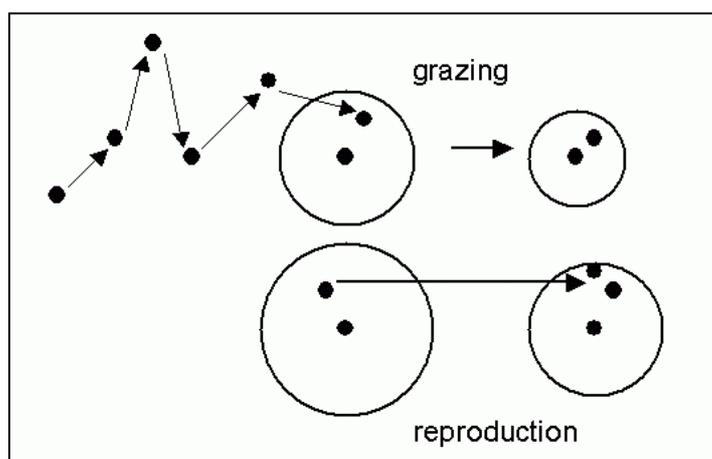


Abbildung 8. Verhalten der Tiere in der Phytophagen-Wachstumsgrammatik.

Zu den bisher beschriebenen L-System-Regeln für das Verhalten der Pflanzen und Tiere kommen nur noch Startregeln, eine Interpretationsregel für die Darstellung der Tiere und einige Variablen- und Konstantendeklarationen hinzu. Mit dieser relativ einfachen Spezifikation erhält man bereits Simulationsläufe, die komplexe räumliche und zeitliche Muster generieren. Je nach Wahl der numerischen Parameter (Schwellenwerte, Ausbreitungs- und Suchdistanzen, Samenanzahl der Pflanzen) können sich unterschiedliche Dynamiken ergeben: Die Pflanzen (und anschließend auch die Tiere) können wegen „Überweidung“ aussterben, die Tiere können vor den Pflanzen aussterben und einen „ungestörten“ Bestand wie in Abb. 7 zurücklassen, oder es kann sich eine längerdauernde Koexistenz mit Fluktuationen der Individuenzahlen einstellen. Auch die räumliche Musterbildung wird von der Wahl der Parameter beeinflusst, wie Abb. 9 zeigt.

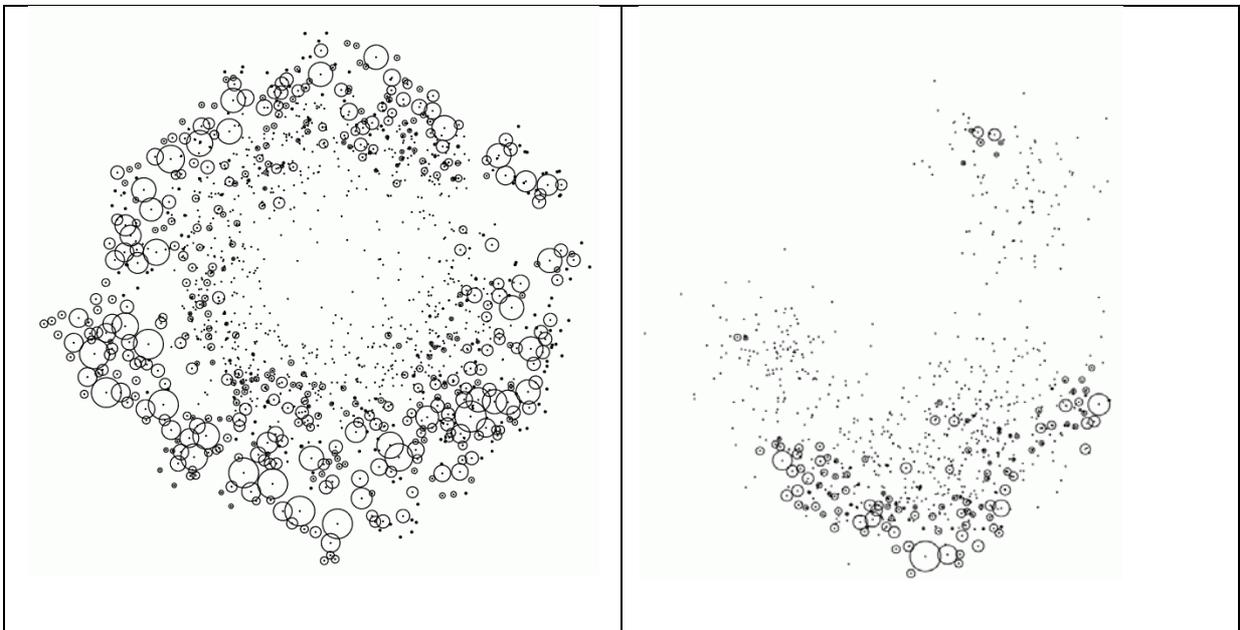


Abbildung 9. Ergebnisse der Phytophagen-Grammatik (beschrieben im Text) nach jeweils 200 Zeitschritten. Beide Simulationsläufe starteten mit einer Pflanze und einem Tier und unterscheiden sich in der Parameterbelegung. Im rechten Beispiel wurden die Tiere mit geringerer Respiration und längeren Suchschrittweiten ausgestattet.

## 5 Simulation evolvierender Verhaltensstrategien

Eine zusätzliche Möglichkeit, die die L-System-Regeln bieten, ist die Weitergabe von genetischer Information für Morphologie und Verhalten. Hier soll dies am Beispiel der Tiere aus obigem Phytophagen-Modell demonstriert werden, die als dritten Parameter (neben Alter und Energiegehalt) die Schrittweite  $w$  zugeordnet bekommen, welche sie im Suchmodus (also ohne Kontakt mit einer Pflanze) für die Eigenbewegung verwenden. Man kann diesen Parameter als eine einfache Be-

schreibung einer Nahrungssuche-Strategie ansehen. Ein Tier wird dann also codiert durch  $a(t, e, w)$ . Bei der Regel für die Reproduktion wird nun dafür gesorgt, dass dieser Parameter  $w$  mit einer kleinen, zufälligen Veränderung  $eps$  (in unserem Beispiel uniform zwischen  $-1$  und  $+1$ ) an die beiden Nachkommen weitergegeben wird (Mutation).

Als Ergebnis hat sich in mehreren Simulationsläufen nach 200 Schritten eine größere Individuenzahl der Tiere ergeben als bei dem Modell ohne Mutation der Schrittweite. Auch zeigt sich nach dieser Schrittzahl eine signifikante Verschiebung des Parameters  $w$  vom Startwert 15 weg zu größeren Schrittweiten (Abb. 10). Dies sind vorläufige Ergebnisse, die noch einer systematischeren Überprüfung bedürfen. Sie zeigen aber bereits, dass es mit sensitiven Wachstumsgrammatiken relativ leicht möglich ist, Evolutionsmechanismen zu simulieren. Man beachte auch, dass in diesem Beispiel keine explizite Fitnessfunktion vorgegeben wurde. Die Fitness (Reproduktionschance) eines Individuums ergibt sich aus der Wechselwirkung mit den simulierten Pflanzen. Auch eine Koevolution von Pflanzen und Tieren wäre im Modell denkbar, wenn man auch die Pflanzen mit genetischer Information ausstattet, die der Mutation unterliegt.

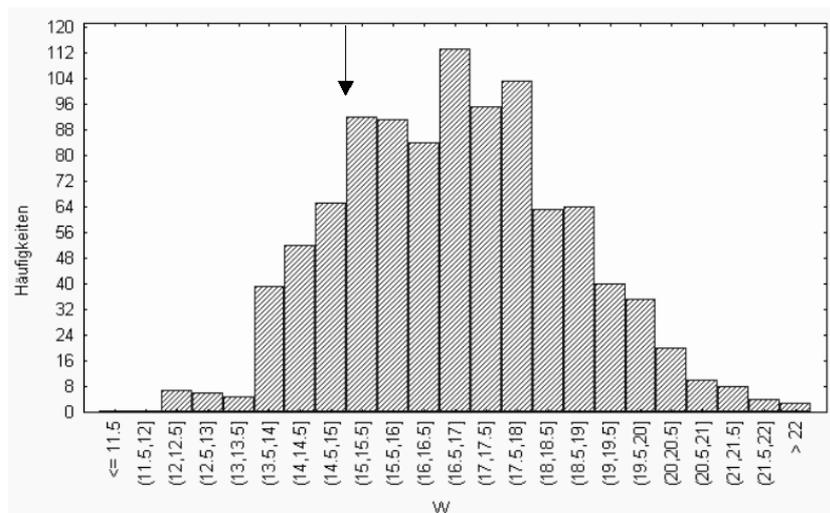


Abbildung 10. Histogramm der in der Tierpopulation nach 200 Zeitschritten vorhandenen Schrittweiten (mutierter Parameter  $w$ ) nach einem Simulationslauf der durch Evolution von  $w$  ergänzten Phytophagen-Grammatik. Der Pfeil kennzeichnet den Startwert 15. Der Trend zu größeren Werten als 15 ist hochsignifikant (t-Test).

## Danksagung

Der Verfasser dankt Broder Breckling, Hans Pretzsch und Helge Dzierzon für wichtige Anregungen. Das Forschungsvorhaben wurde zeitweise vom BMBF und zeitweise von der DFG gefördert; der Verfasser dankt für die Unterstützung.

## Literaturangaben

- Abelson, H.; diSessa, A.A. (1982): *Turtle Geometry*. MIT Press, Cambridge.
- Acock, B.; Reynolds, J.F. (1997): Introduction: Modularity in Plant Models. *Ecological Modelling* **94** (1997) 1–6.
- Bormann, F.H.; Likens, G.E. (1979): *Pattern and Process in Forested Ecosystems*. Springer, New York.
- Breckling, B. (1990): Singularität und Reproduzierbarkeit in der Modellierung ökologischer Systeme. Dissertation, Universität Bremen, FB2 (Biologie / Chemie).
- Breckling, B. (1996): An Individual Based Model for the Study of Pattern and Process in Plant Ecology: An Application of Object Oriented Programming. *EcoSys* **4** (1996) 241–254.
- DeAngelis, D.L.; Gross, L.J. (Eds.) (1992): *Individual-based Models and Approaches in Ecology: Populations, Communities and Ecosystems*. Chapman & Hall, New York.
- Deussen, O.; Hanrahan, P.; Lintermann, B.; Mech, R.; Pharr, M.; Prusinkiewicz, P. (1998): Realistic modeling and rendering of plant ecosystems. *Proceedings SIGGRAPH98, Computer Graphics Proceedings, Annual Conference Series, 1998*, 275–286.
- Ermentrout, G.B.; Edelstein-Keshet, L. (1993): Cellular Automata Approaches to Biological Modelling. *Journal of Theoretical Biology* **160** (1993) 97–133.
- Gerhardt, M.; Schuster, H. (1995): *Das digitale Universum. Zelluläre Automaten als Modelle der Natur*. Vieweg, Braunschweig.
- Kaitaniemi, P.; Ruohomäki, K.; Haukioja, E. (1997): Consequences of Defoliation on Phenological Interaction between *Epirrita autumnata* and its Host Plant, Mountain Birch. *Functional Ecology* **11** (1997) 199–208.
- Kurth, W. (1994): Growth Grammar Interpreter GROGRA 2.4. *Forschungszentrum Waldökosysteme der Universität Göttingen, Ser. B* **38**.
- Kurth, W. (1999): *Die Simulation der Baumarchitektur mit Wachstumsgrammatiken*. Wissenschaftlicher Verlag Berlin, Berlin.
- Kurth, W. (2000): Spezifikation räumlicher Bestandes- und Populationsmodelle mit sensitiven Grammatiken. In: Saborowski, J.; Sloboda, B. (Hrsg.): *Deutscher Verband Forstlicher Forschungsanstalten, Sektion Forstliche Biometrie u. Informatik, 12. Tagung 1999*. Biotechnische Fakultät Ljubljana, 259–278.
- Lindenmayer, A. (1968): Mathematical Models for Cellular Interactions in Development. *Journal of Theoretical Biology* **18** (1968) 280–299, 300–315.
- Pretzsch, H. (1992): Modellierung der Kronenkonkurrenz von Fichte und Buche in Rein- und Mischbeständen. *Allg. Forst- u. Jagdzeitg.* **163** (1992) 203–213.
- Pretzsch, H. (1998): Models for Pure and Mixed Forests. In: Mies, E. (ed.): *Natural and Socioeconomic Analysis and Modelling of Forest and Agroforestry Systems in Southeast Asia*. DSE/ZEL, Feldafing, 1–25.
- Prusinkiewicz, P.; Lindenmayer, A. (1990): *The Algorithmic Beauty of Plants*. Springer, New York.
- Prusinkiewicz, P.; Hammel, M.; Hanan, J.; Mech, R. (1997): Visual Models of Plant Development. In: Rozenberg, G.; Salomaa, A. (eds.): *Handbook of Formal Languages. Vol. 3: Beyond Words*. Springer, Berlin, 535–597.
- Rozenberg, G. (Ed.) (1997): *Handbook of Graph Grammars and Computing by Graph Transformation. Vol.1: Foundations*. World Scientific Publ., Singapore.