Synthetic Topiary

Przemyslaw Prusinkiewicz, Mark James, and Radomír Měch

Department of Computer Science
University of Calgary
Calgary, Alberta, Canada T2N 1N4

ABSTRACT

The paper extends Lindenmayer systems in a manner suitable for simulating the interaction between a developing plant and its environment. The formalism is illustrated by modeling the response of trees to pruning, which yields synthetic images of sculptured plants found in topiary gardens.

CR categories: F.4.2 **[Mathematical Logic and Formal Languages]**: Grammars and Other Rewriting Systems: *Parallel rewriting systems*, I.3.7 **[Computer Graphics]**: Three-Dimensional Graphics and Realism, I.6.3 **[Simulation and Modeling]**: Applications, J.3 **[Life and Medical Sciences]**: Biology.

Keywords: image synthesis, modeling of plants, L-system, topiary.

1 INTRODUCTION

One classification of visual plant models introduces a distinction between *structure-oriented* and *space-oriented* models [26]. The first class is characterized by the assumption that the developmental process and the resulting structure are under the control of *endogenous* mechanisms, inherent in the growing structure and internal to it. *Lineage*, or the transfer of information from mother to daughter *modules* (components of the model) at the time of daughter creation, is the most frequently simulated form of endogenous control, although many models have been formulated without referring to this term explicitly. For example, tree models proposed by Aono and Kunii [1], Bloomenthal [5], Reeves and Blau [32], Oppenheimer [25], and de Reffye and his collaborators [10, 19] are all controlled by lineage. In contrast, *interactive* mechanisms involve information flow between coexisting adjacent components of the developing structure. In a growing plant, this information may be represented by phytohormones, nutrients, or water. *Context-sensitive L-systems* [22] provide a formally defined framework for simulating interactive control mechanisms, and have been used for image synthesis purposes by Smith [33] and Prusinkiewicz *et al.* [28, 29]. Outside the domain of L-systems, models of interactive endogenous control have been investigated by Borchert and Honda [6].

In contrast to structure-oriented models, space-oriented models capture the entire environment of a growing plant, and emphasize *exogenous* control, in which information is transferred through the environment enclosing the modeled structure. This class includes the models of climbing plants introduced by Arvo and Kirk [2] and Greene [12], as well as the models of roots growing around obstacles, also created by Greene [13].

The dichotomy between the structure-oriented and space-oriented models makes it difficult to faithfully capture plants in which the internally controlled development is modified by environmental factors. Such a combination of endogenous and exogenous mechanisms is manifested, for instance, in plant responses to collisions with obstacles, presence or absence of light, pruning, and attacks by insects. Several techniques were proposed to simulate these phenomena. For example, Honda *et al.* [18] used proximity of branches as a factor inhibiting their further growth and bifurcation. Kanamaru *et al.* [21] devised a convincing model of tree architecture in which the development of individual branches is controlled by the amount and direction of incoming light. A variety of factors, including the availability of light and the presence of mechanical obstacles, was assumed by Dabadie [9]. Mechanical obstacles to the development of branching patterns were also considered by Kaandorp [20].

In spite of these results, the problem of specifying and constructing plant models that integrate endogenous and exogenous mechanisms has not yet been completely resolved, because the reported techniques do not combine exogenous and *interactive* endogenous control. We address this limitation by introducing an *environmentally-sensitive* extension of L-systems, based on earlier results by Prusinkiewicz and McFadzean [30], and MacKenzie [24]. According to this extension, selected modules of a growing structure may access information about their position and orientation in space. We illustrate the operation of environmentally-sensitive L-systems by modeling plant response to the extensive pruning found in topiary and knot gardens. This application is motivated by the visual appeal of the resulting forms and their potential relevance to computer-assisted landscape design.

The paper is organized as follows. Background information regarding L-systems is presented in Section 2. On this basis, environmentally sensitive L-systems are defined and illustrated using simple examples in Section 3. Section 4 introduces a more realistic tree model, needed to create synthetic topiary forms. A mechanism that governs the response to pruning is incorporated into this model in Section 5. The resulting topiary forms are presented in Section 6. Section 7 contains conclusions, and lists directions for future work.

2 L-SYSTEMS

As the point of departure, we use parametric L-systems with turtle interpretation, described in detail in [16, 27, 28]. The essential aspects of this formalism relevant to the environmentally-sensitive extension are summarized below.

An L-system is a parallel rewriting system operating on branching structures represented as *bracketed strings* of modules. Matching pairs of square brackets enclose branches. Simulation begins with an initial string called the *axiom*, and proceeds in a sequence of discrete *derivation steps*. In each step, *rewriting rules* or *productions* replace all modules in the predecessor string by successor modules. The applicability of a production depends on a predecessor's context (in context-sensitive L-systems), values of parameters (in productions guarded by conditions), and on random factors (in stochastic L-systems). In the most extensive case, a production has the format:

$$id : lc < pred > rc : cond \rightarrow succ : prob$$

where id is the production identifier (label), lc, $pred$, and rc are the left context, the strict predecessor, and the right context, $cond$ is the condition, $succ$ is the successor, and $prob$ is the probability of production application. The strict predecessor and the successor are the only mandatory fields. For example, the L-system given below consists of axiom ω and three non-identity productions p_1, p_2, and p_3.

L-system 1

$$\begin{array}{ll} \omega : & A(1)B(3)A(5) \\ p_1 : & A(x) \rightarrow A(x+1) : 0.4 \\ p_2 : & A(x) \rightarrow B(x-1) : 0.6 \\ p_3 : & A(x) < B(y) > A(z) : y < 4 \rightarrow B(x+z)[A(y)] \end{array}$$

The stochastic productions p_1 and p_2 replace module $A(x)$ either by $A(x+1)$ or by $B(x-1)$, with probabilities equal to 0.4 and 0.6, respectively. The context-sensitive production p_3 replaces a module $B(y)$ with left context $A(x)$ and right context $A(z)$ by module $B(x+z)$ supporting branch $A(y)$. The application of this production is guarded by condition $y < 4$. Consequently, the first derivation step may have the form:

$$A(1)B(3)A(5) \Longrightarrow A(2)B(6)[A(3)]B(4)$$

It was assumed that, as a result of random choice, production p_1 was applied to the module $A(1)$, and production p_2 to the module $A(5)$. Production p_3 was applied to the module $B(3)$, because it occurred with the required left and right context, and the condition $3 < 4$ was true.

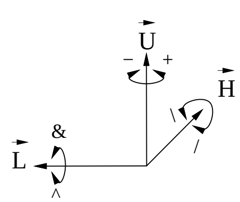

Figure 1: Controlling the turtle in three dimensions

In contrast to the parallel application of productions in each derivation step, the interpretation of the resulting strings proceeds sequentially, with the reserved modules acting as commands to a LOGO-style turtle [27, 28, 29]. At any time, the turtle is characterized by a position vector $\vec{P}$ and three mutually perpendicular orientation vectors $\vec{H}$, $\vec{U}$, and $\vec{L}$, indicating the turtle's heading, the up direction, and the direction to the left. Module F causes the turtle to draw a line in the current direction. Modules $+$, $-$, $\&$, $\wedge$, / and \ rotate the turtle around one of the vectors $\vec{H}, \vec{U}$, or $\vec{L}$, as shown in Figure 1. The length of the line and the magnitude of the rotation angle can be given globally or specified as parameters of individual modules. During the interpretation of branches, the opening square bracket pushes the current position and orientation of the turtle on a stack, and the closing bracket restores the turtle to the position and orientation popped from the stack. A two-symbol module @o draws a sphere at the current position. A special interpretation is reserved for the module $\%$, which cuts a branch by erasing all symbols in the string from the point of its occurrence to the end of the branch. The meaning of many symbols depends on the context in which they occur; for example, $+$ and $-$ denote arithmetic operators as well as modules that rotate the turtle.

3 ENVIRONMENTALLY-SENSITIVE L-SYSTEMS

The turtle interpretation of L-systems described above was designed to visualize models in a postprocessing step, with no effect on the L-system operation. Position and orientation of the turtle are important, however, while considering environmental phenomena, such as collisions with obstacles and exposure to light. Consequently, the *environmentally-sensitive extension* of L-systems makes these attributes accessible during the rewriting process. To this end, the generated string is interpreted after each derivation step, and turtle attributes found during the interpretation are returned as parameters to reserved *query modules* in the string. Each derivation step is performed as in parametric L-systems, except that the parameters associated with the query modules remain undefined. During the interpretation, these modules are assigned values that depend on the turtle's position and orientation in space. Syntactically, the query modules have the from $?X(x,y,z)$, where $X = P, H, U$, or L. Depending on the actual symbol X, the values of parameters x, y, and z represent a position or an orientation vector. In the two-dimensional case, the coordinate z may be omitted.

The operation of the query module is illustrated by a simple environmentally-sensitive L-system, given below.

L-system 2

$$\begin{array}{llll} \omega : & A & & \\ p_1 : & A & \rightarrow & F(1)?P(x,y) - A \\ p_2 : & F(k) & \rightarrow & F(k+1) \end{array}$$

The following strings are produced during the first three derivation steps.

$$\begin{array}{ll} \mu_0' : & A \\ \mu_0 : & A \\ \mu_1' : & F(1)?P(\star,\star) - A \\ \mu_1 : & F(1)?P(0,1) - A \\ \mu_2' : & F(2)?P(\star,\star) - F(1)?P(\star,\star) - A \\ \mu_2 : & F(2)?P(0,2) - F(1)?P(1,2) - A \\ \mu_3' : & F(3)?P(\star,\star) - F(2)?P(\star,\star) - F(1)?P(\star,\star) - A \\ \mu_3 : & F(3)?P(0,3) - F(2)?P(2,3) - F(1)?P(2,2) - A \end{array}$$

Strings μ_0', μ_1', μ_2', and μ_3' represent the axiom and the results of production application before the interpretation steps. Symbol $\star$ indicates an undefined parameter value in a query module. Strings μ_1, μ_2, and μ_3 represent the corresponding strings after interpretation. It has been assumed that the turtle is initially placed at the origin of the coordinate system, vector $\vec{H}$ is aligned with the y axis, vector $\vec{L}$ points in the negative direction of the x axis, and the angle

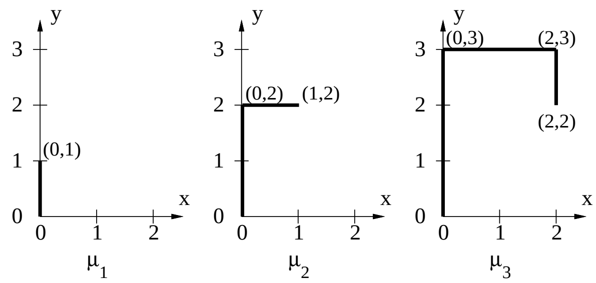

Figure 2: Assignment of values to query modules

of rotation associated with module "$-$" is equal to $90°$. Parameters of the query modules have values representing the positions of the turtle shown in Figure 2.

The next example illustrates an abstract developmental process influenced by the environment.

L-system 3

$$
\begin{array}{lll}
\omega : & A & \\
p_1 : & A & \rightarrow \; [+B][-B]F?P(x,y)A \\
p_2 : & B & \rightarrow \; F?P(x,y)@oB \\
p_3 : & ?P(x,y) : 4x^2 + (y-10)^2 > 10^2 & \\
 & & \rightarrow \; [+(2y)F][-(2y)F]\%
\end{array}
$$

Module F represents a line of unit length, and modules $+$ and $-$ without parameters represent left and right turns of $60°$.

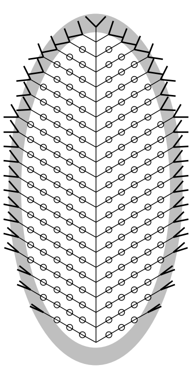

Figure 3: A branching structure pruned to an ellipse

The development begins with module A, which creates a sequence of opposite branches $[+B][-B]$ separated by internodes (branch segments) F (production p_1). The branches elongate by addition of segments F, delimited by markers $@o$ (production p_2). Both the main apex A and the lateral apices B create query modules $?P(x,y)$, which return the corresponding turtle positions. If a query module is placed beyond the ellipse $4x^2+(y-10)^2 = 10^2$, production p_3 creates a pair of "tentacles," represented by the substring $[+(2y)F][-(2y)F]$. The angle $4y$ between these tentacles depends on the vertical position y of the query module. Production p_3 also inserts cutting symbol $\%$, which terminates branch growth by removing its apex. In summary, L-system 3 produces a branching structure confined to an ellipse, with tentacles placed at the boundary of the structure, and the angle between the tentacles depending on the turtle's position in space, as shown in Figure 3.

The final example of this section presents a simple two-dimensional model of tree response to pruning. As described, for example, by Hallé *et al.* [15, Chapter 4] and Bell [3, page 298], during the normal development of a tree many buds do not produce new branches and remain dormant. These buds may be subsequently activated by the removal of leading buds from the branch system (*traumatic reiteration*), which results in an environmentally-adjusted tree architecture. The model given below represents the extreme case of this process, where buds are activated only as a result of pruning.

L-system 4

$$
\begin{array}{ll}
\omega : & FA?P(x,y) \\
p_1 : & A > ?P(x,y) : \; !\text{prune}(x,y) \rightarrow @oF/(180)A \\
p_2 : & A > ?P(x,y) : \; \text{prune}(x,y) \rightarrow T\% \\
p_3 : & F > T \rightarrow S \\
p_4 : & F > S \rightarrow SF \\
p_5 : & S \rightarrow \epsilon \\
p_6 : & @o > S \rightarrow [+FA?P(x,y)]
\end{array}
$$

The user defined function

$$\text{prune}(x,y) = (x < -L/2) \| (x > L/2) \| (y < 0) \| (y > L),$$

where $\|$ stands for the logical OR operator, defines a square *clipping box* of dimensions $L \times L$ that bounds the growing structure. According to axiom ω, the development begins with an internode F supporting apex A and query module $?P(x,y)$. The initial development of the structure is described by production p_1. In each step, the apex A creates a dormant bud $@o$ and an internode F. The module $/(180)$ rotates the turtle around its own axis (the heading vector $\vec{H}$), thus laying a foundation for an alternating branching pattern. The query module $?P(x,y)$, placed by the axiom, is the right context for production p_1 and returns the current position of apex A. When a branch extends beyond the clipping box, production p_2 removes apex A, cuts off the query module $?P(x,y)$, and generates the pruning signal T. In presence of this signal, production p_3 removes the last internode of the branch that extends beyond the clipping box and creates bud-activating signal S. Productions p_4 and p_5 propagate this signal basipetally (downwards), until it reaches a dormant bud $@o$. Production p_6 induces this bud to initiate a lateral branch consisting of internode F and apex A followed by query module $?P(x,y)$. According to production p_1, this branch develops in the same manner as the main axis. When its apex extends beyond the clipping box, it is removed by production p_2, and signal S is generated again. This process may continue until all dormant buds have been activated.

Selected phases of the described developmental sequence are illustrated in Figure 4. In derivation step 6 the apex of the main axis grows out of the clipping box. In step 7 this apex and the last internode are removed from the structure, and the bud-activating signal S is generated. As a result of bud activation, a lateral branch is created in step 8. As it also extends beyond the bounding box, it is removed in step 9 (not shown). Signal S is generated again, and in step 10 it reaches a dormant bud. The subsequent development of the lateral branches, shown in the middle and bottom rows of Figure 4, follows a similar pattern.

L-system 4 simulates the response of a tree to pruning using a schematic branching structure. A more realistic model is needed to synthesize visually convincing images of pruned trees. A suitable model of free-standing trees is presented in the next section, and applied to simulate the response to pruning in Sections 5 and 6.

4 A STOCHASTIC TREE MODEL

As a first approximation, the development of a free-standing woody plant — a tree or a shrub — can be described as a process in which new branches are successively added to the structure. Early tree

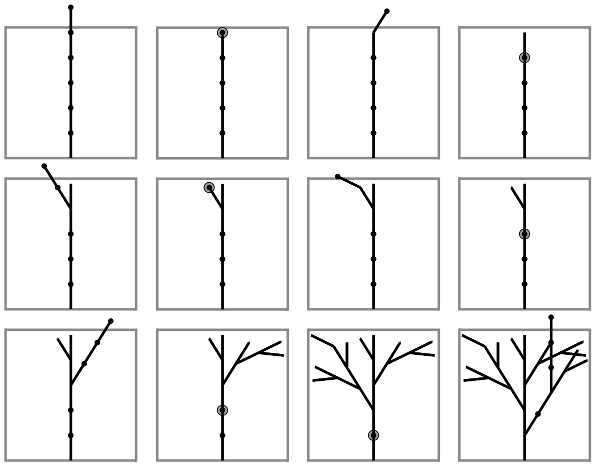

Figure 4: A simple model of a tree's response to pruning. Top row: derivation steps 6,7,8, and 10; middle row: steps 12, 13, 14, and 17; bottom row: steps 20, 40, 75, and 94. Small black circles indicate dormant buds, the larger circles indicate the position of signal S.

models emphasized the repetitive character of this process. For example, Honda [17] described a tree as a recursive branching structure, in which the *bifurcation ratio* (the number of branches originating at the mother branch), the branching angles, and the proportions between the lengths of the mother branch and the daughter branches do not depend on the position of the branches in the crown nor on the age of the simulated tree. Tree models of Aono and Kunii [1] and Oppenheimer [25] satisfy similar assumptions. The resulting structures are self-similar, which implies that the number of branches increases exponentially with the age of the structure.

Using morphometric data of young cottonwood (*Populus deltoides*) and observations of the tropical tree *Tabebuia rosea*, Borchert and Slade [7] showed that the exponential growth of the number of terminal branches yields unnaturally dense crowns in models of older trees. In reality, as soon as trees surpass a certain, relatively small size, the rate of branching decreases. Based on the analysis of this phenomenon presented by Borchert and Slade, we present here a model of trees suitable for computer graphics purposes. It is constructed to meet the following botanically justifiable postulates:

- The development begins in season $k = 1$ with the formation of a single nonbranching shoot (branch segment bearing leaves).
- In each subsequent growth season, new shoots grow from the buds situated near the distal ends of last year's segments. There is a constant, $b_{max} > 1$, that determines the maximum bifurcation ratio.
- All branch segments have approximately the same length l, independent of their position and the age of the tree, and reach out forming a hemispherical crown.
- Leaves are produced on the terminal (current year) branch segments, thus forming a hemispherical layer of leaves near the perimeter of the crown. There is a constant, σ_{min}, that determines the minimum area of leaves that must be exposed to light coming from the outside in order to create a viable shoot.

According to these postulates, the radius of a tree crown after $k \geq 1$ growth seasons is limited by $R_k = lk$. A hemispherical crown of this radius has surface area $S_k = 2\pi R_k^2 = 2\pi l^2 k^2$, and this value determines the upper bound on the crown area exposed to direct light. The number N_{k+1} of branch segments added to the tree in year $k+1$ is limited, on the one hand, by the number of last year's segments N_k multiplied by the maximum bifurcation ratio b_{max}, and on the other hand, by the maximum number $v_{k+1} = S_{k+1}/\sigma_{min}$ of shoots that may be produced without excessively obscuring each other. Thus,

$$N_{k+1} = \min\{b_{max}N_k, v_{k+1}\} = \min\{b_{max}N_k, \frac{2\pi l^2}{\sigma_{\min}}(k+1)^2\}.$$

Let us assume that the minimum leaf area exposed to light per shoot is small compared to the crown area, $\sigma_{min} \ll 2\pi l^2$. In a young tree (during the first few growth seasons), the maximum number of new shoots does not suffice to cover the available crown surface ($b_{max}N_k < v_{k+1}$), and the number of new shoots will increase exponentially with the age of the tree: $N_{k+1} = b_{max}N_k = b_{max}^k$. Since the crown area is proportional only to the square of the age of the tree, at some age t the potential number of new shoots will exceed the number that can be sufficiently exposed to direct light: $b_{max}N_t \geq v_t$. From then on, branching will be limited by the crown area, with the average bifurcation ratio b_k at age $k \geq t$ equal to:

$$b_k = \frac{N_{k+1}}{N_k} = \frac{2\pi l^2(k+1)^2/\sigma_{min}}{2\pi l^2 k^2/\sigma_{min}} = 1 + \frac{2k+1}{k^2}.$$

Different branching patterns may satisfy this general formula. For example, if each segment from the previous year gives rise either to one or to two new shoots, the fraction of segments supporting two shoots will be equal to:

$$\frac{N_{k+1} - N_k}{N_k} = \frac{2k+1}{k^2}. \quad (1)$$

The stochastic L-system below has been constructed to satisfy this equation.

L-system 5

ω : $FA(1)$
p_1 : $A(k) \rightarrow /(\phi)[+(\alpha)FA(k+1)] - (\beta)FA(k+1)$: $\min\{1, (2k+1)/k^2\}$
p_2 : $A(k) \rightarrow /(\phi)B - (\beta)FA(k+1)$: $\max\{0, 1-(2k+1)/k^2\}$

The generation of the tree begins with a single internode F terminated by apex $A(1)$. The parameter of the apex acts as a counter of derivation steps. Production p_1 describes the creation of two new branches, while production p_2 describes the production of a branch segment and a dormant bud B. Probabilities of these events are equal to $p = \min\{1, (2k+1)/k^2\}$, and $q = 1 - p$, respectively. This corresponds to the assumption that the departure from exponential bifurcation occurs in step $k = 3$, and in subsequent steps the probability of bifurcation is determined by Equation 1. Figure 5 shows side views of three sample trees after 18 derivation steps. The branching angles, equal to $\phi = 90°, \alpha = 32°$, and $\beta = 20°$, yield a sympodial branching structure (new shoots do not continue the growth direction of the preceding segments). This structure is

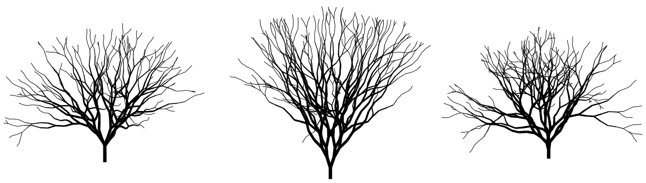

Figure 5: Sample tree structures generated using L-system 5

representative to the Leeuwenberg's model of tree architecture identified by Hallé *et al.* [15], although no attempt to capture a particular tree species was made. The same values of branching angles can be found in all the tree models shown in this paper.

5 SIMULATION OF PRUNING

L-system 5 generates structures with many dormant buds, and therefore can be used to simulate tree response to pruning in a manner similar to that implemented in L-system 4. The resulting integrated model is given below.

L-system 6

$$
\begin{array}{ll}
\omega : & FA(1)?P(x,y,z) \\
p_1 : & A(k) > ?P(x,y,z) : \text{!prune}(x,y,z) \rightarrow \\
& \quad /(\phi)[+(\alpha)FA(k+1)?P(x,y,z)] - (\beta)FA(k+1) : \\
& \qquad \min\{1, (2k+1)/k^2\} \\
p_2 : & A(k) > ?P(x,y,z) : \text{!prune}(x,y,z) \rightarrow \\
& \quad /(\phi)B(k+1,k+1) - (\beta)FA(k+1) : \\
& \qquad \max\{0, 1-(2k+1)/k^2\} \\
p_3 : & A(k) > ?P(x,y,z) : \ \text{prune}(x,y,z) \rightarrow T\% \\
p_4 : & F > T \rightarrow S \\
p_5 : & F > S \rightarrow SF \\
p_6 : & S \rightarrow \epsilon \\
p_7 : & B(m,n) > S \rightarrow [+(\alpha)FA(am+bn+c)?P(x,y,z)] \\
p_8 : & B(m,n) > F \rightarrow B(m+1,n)
\end{array}
$$

According to axiom ω, the development begins with a single internode F supporting apex $A(1)$ and query module $?P(x,y,z)$. Productions p_1 and p_2 are similar to those in L-system 5 and describe the spontaneous growth of the tree within the volume characterized by a user-defined clipping function prune(x,y,z). Productions p_3 to p_7 are similar to productions p_2 to p_6 in L-system 4. Specifically, production p_3 removes the apex $A()$ after it has crossed the clipping surface, cuts off the query module $?P(x,y,z)$, and creates pruning signal T. Next, p_4 removes the last internode of the pruned branch and initiates bud-activating signal S, which is propagated basipetally by productions p_5 and p_6. When S reaches a dormant bud $B()$, production p_7 transforms it into a branch consisting of an internode F, apex $A()$, and query module ?P(x,y,z).

The parameter value assigned by production p_7 to apex $A()$ is derived as follows. According to production p_2, both parameters associated with a newly created bud $B()$ are set to the age of the tree at the time of bud creation (expressed as the the number of derivation steps). Production p_8 updates the value of the first parameter (m), so that it always indicates the actual age of the tree. The second parameter (n) remains unchanged. The initial *biological age* [3, page 315] of the activated apex $A()$ in production p_7 is a linear combination of parameters m and n, calculated using the expression $am + bn + c$. Since rule p_1 is more likely to be applied for young apices (for small values of parameter k), by manipulating constants a, b, and c it is possible to control the bifurcation frequency of branches created as a result of traumatic reiteration. This is an important feature of the model, because in nature the reiterated branches tend to be more juvenile and vigorous than the remainder of the tree [3, page 298].

The operation of this model is illustrated in Figure 6. The clipping form is a cube with an edge length 12 times longer than the internode lengh. The constant values used in production p_7 are $a = 0$, $b = 1$, and $c = -5$. The structures shown have been generated in 3, 6, 9, 13, 21, and 27 steps. Leaves were defined using Bézier surfaces, as described in [28, Section 5.1].

The impact of constants a, b, and c on tree structures is further illustrated in Figure 7. All trees have been generated in 31 steps. In the pair of trees shown on the left-hand side, the initial age of the activated apices is equal to the actual age of the tree minus 5 ($a = 1$, $b = 0$, and $c = -5$). In the middle pair, the initial age is equal to the time of bud creation minus 5 ($a = 0$, $b = 1$, and $c = -5$). Finally, in the rightmost pair the reiterated branches are assigned an initial age of 1 ($a = 0, b = 0, c = 1$). In all cases, the density of the branches is increased near the boundary of the clipping box, compared to a non-pruned tree. As a result, a pruned tree acquires a shape that closely resembles that of its bounding volume, defined by the clipping function. This effect is most pronounced when the reiterated branches are assigned an initial age of 1, which results in the most vigorous branching.

By changing the clipping function, one can shape plant models generated by L-system 6 to a variety of artificial forms. Selected examples are presented in the next section. In all cases the initial age of activated apices is calculated using the set of parameters $a = 0$, $b = 1$, and $c = -5$.

6 EXAMPLES OF SYNTHETIC TOPIARY

The term *topiary* denotes the art (or craft) of clipping suitable trees and shrubs into elaborate ornamental shapes, most frequently freestanding [23, pages 132, 183]. These shapes range from purely geometric ones, such as spheres, cones, or spirals, to depictions of "hunting scenes, fleets of ships, and imitations of real objects." [14, page 11]. Related to topiary is the ornamental use of hedges, which includes tall structures intended to obscure the view in *labyrinths*, and intricate patterns of low shrubs designed to be viewed from above in *knot gardens* or *parterres*.

Given a flexible model of tree response to pruning, the main remaining issue is the specification of the clipping surface. *Implicit surfaces* [4, 36] are particularly suitable for this purpose, since they provide a simple method for checking whether a query point lies inside or outside the defined surface. The clipping forms can be blended together and combined using constructive solid geometry operations [35].

The Levens Hall garden in England, laid out at the beginning of the 18th century, is considered the most famous topiary garden in the world [8, pages 52–57]. It contains many geometric forms, two of which have been reproduced in Figures 8 and 9. Specifically, Figure 8 illustrates the use of constructive solid geometry operations to define the clipping form, in this case, the union of a parallelepiped and a cylinder. The spirals shown in Figure 9 have been obtained by pruning a tree to the shape of a seashell. An implicit repre-

Figure 6: Simulation of tree response to pruning

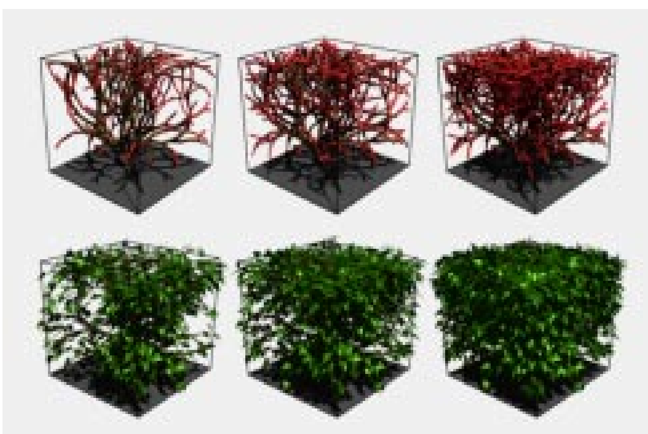

Figure 7: Impact of vigor of reiterated branches (shown in red) on the appearance of a pruned tree

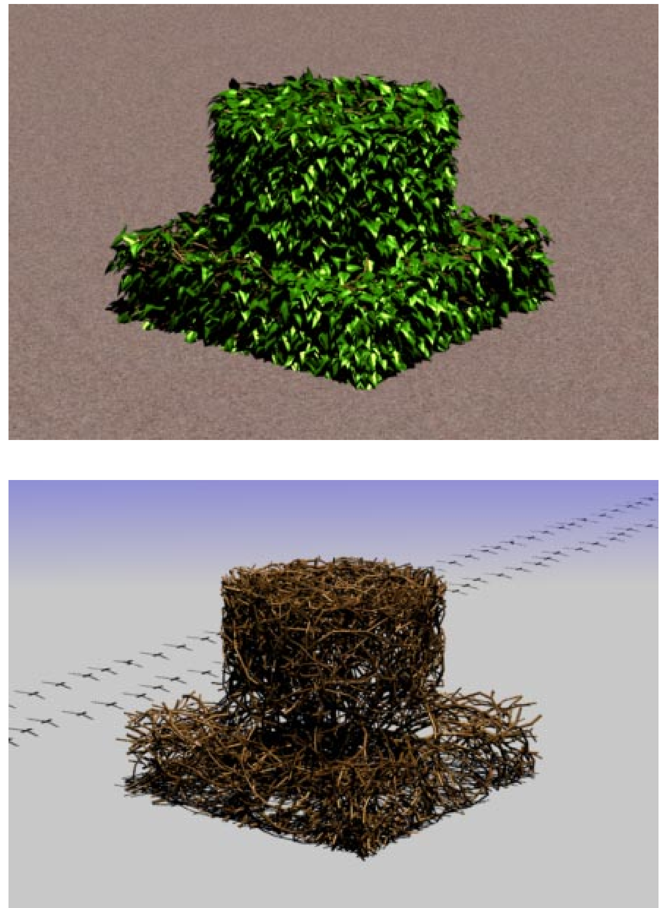

Figure 8: A tree pruned to a union of a parallelepiped and a cylinder

Figure 9: Trees pruned to a spiral shape

sentation of the seashell was obtained by converting the parametric representation described by Fowler *et al.* [11].

Another application of implicit surface definition is shown in Figures 10 and 11. In this case, the clipping form of a "topiary dinosaur" was obtained as an implicit surface defined by a skeleton of lines and ellipsoids. Two trees were used to facilitate the growth of branches into the elongated shapes of neck and tail.

The large number of primitives representing individual plants makes it difficult to combine them into complex scenes. The reuse (instantiation) of models provides a simple solution in the case of repetitive designs. For example, the hedges shown in Figure 12 have been composed of rectangular and circular segments, replicated to create the complete scene. The images are relatively faithful synthetic representations of the knot garden at Moseley Old Hall, reconstructed in England in 1960 from a seventeenth-century design [34, page 50] (see also [23, page 122]).

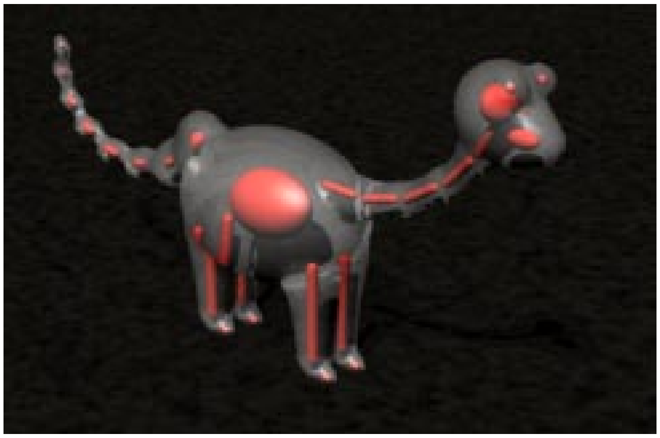

Figure 10: An implicit surface defined by a skeleton of lines and ellipsoids

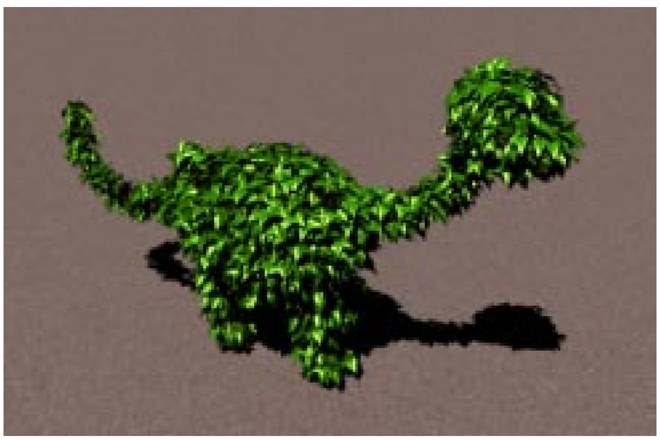

Figure 11: Topiary dinosaur

7 DISCUSSION

In this paper, we extended L-systems with a mechanism for simulating the impact of the environment on plant development. The resulting formalism was explained using simple geometric examples, then applied to simulate plant response to pruning. A biologically-motivated tree model incorporating this mechanism served as a basis for creating models of sculptured plants found in topiary gardens. One prospective application of such models is in computer-assisted landscape design.

We have not found much published information characterizing the impact of pruning on tree architecture. More data would be necessary to construct faithful models of particular tree species. As described in [31], construction of visual models of plants provides a valuable guideline for collecting field data. Consequently, the mathematical framework introduced in the present paper may assist in biological studies of the effects of pruning on plant development.

Pruning is only one of a range of phenomena that can be modeled using environmentally-sensitive L-systems. The values returned by the query modules may not only indicate whether a query point is inside or outside a clipping volume, but also to return, through properly defined *field functions,* other values characterizing the space in which the plant develops. For example, in a model of roots, the

Figure 12: A model of the knot garden at Moseley Old Hall

field values may represent concentrations of nutrients and water in soil. This field may be assumed to be stationary, or change dynamically to reflect the absorption of substances by the growing plant. Above the ground, a dynamically changing field may be used to distinguish areas exposed to light from those in shade, and specify areas occupied by other objects for collision detection purposes. We hope that such simulations will lead to a better understanding of the underlying phenomena, increase the predictive value of plant models, and result in more realistic synthetic images of plants.

Acknowledgements

We would like to thank Jim Hanan for his essential work on the plant modeling software *cpfg,* which provided the basis for our implementation of the environmentally-sensitive extension of L-systems. Brian Wyvill created the implicitly defined pruning surface for the dinosaur. All images were rendered using the ray-tracer `rayshade` by Craig Kolb. Brian Gaines kindly made his compute server available to us. Keith Ferguson, Lynn Mercer, and the anonymous referees provided many useful comments, which we tried to incorporate into the final version of the paper. This research was sponsored by operating and equipment grants from the Natural Sciences and Engineering Research Council of Canada, and a grant from the Josefa, Marie a Zdeňky Hlávkových Foundation.

REFERENCES

[1] M. Aono and T. L. Kunii. Botanical tree image generation. *IEEE Computer Graphics and Applications,* 4(5):10–34, 1984.

[2] J. Arvo and D. Kirk. Modeling plants with environment-sensitive automata. In *Proceedings of Ausgraph '88,* pages 27–33, 1988.

[3] A. Bell. *Plant form: an illustrated guide to flowering plants.* Oxford University Press, Oxford, 1991.

[4] J. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3), 1982.

[5] J. Bloomenthal. Modeling the Mighty Maple. Proceedings of SIGGRAPH '85, in *Computer Graphics*, 19(3):305–311, 1985.

[6] R. Borchert and H. Honda. Control of development in the bifurcating branch system of *Tabebuia rosea*: A computer simulation. *Botanical Gazette*, 145(2):184–195, 1984.

[7] R. Borchert and N. Slade. Bifurcation ratios and the adaptive geometry of trees. *Botanical Gazette*, 142(3):394–401, 1981.

[8] P. Coats. *Great gardens of the Western world*. G. P. Putnam's Sons, New York, 1963.

[9] P. Dabadie. *Contribution à la modélisation et simulation de la croissance des végétaux*. PhD thesis, Université Montpellier II, October 1991.

[10] P. de Reffye, C. Edelin, J. Françon, M. Jaeger, and C. Puech. Plant models faithful to botanical structure and development. Proceedings of SIGGRAPH '88, in *Computer Graphics*, 22(4):151–158, 1988.

[11] D. R. Fowler, H. Meinhardt, and P. Prusinkiewicz. Modeling seashells. Proceedings of SIGGRAPH '92, in *Computer Graphics*, 26(2):379–387, 1992.

[12] N. Greene. Voxel space automata: Modeling with stochastic growth processes in voxel space. Proceedings of SIGGRAPH '89, in *Computer Graphics*, 23(4):175–184, 1989.

[13] N. Greene. Detailing tree skeletons with voxel automata. SIGGRAPH '91 Course Notes on Photorealistic Volume Modeling and Rendering Techniques, 1991.

[14] M. Hadfield. *Topiary and ornamental hedges*. A. & C. Black, London, 1971.

[15] F. Hallé, R. A. A. Oldeman, and P. B. Tomlinson. *Tropical trees and forests: An architectural analysis*. Springer-Verlag, Berlin, 1978.

[16] J. S. Hanan. *Parametric L-systems and their application to the modelling and visualization of plants*. PhD thesis, University of Regina, June 1992.

[17] H. Honda. Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body. *Journal of Theoretical Biology*, 31:331–338, 1971.

[18] H. Honda, P. B. Tomlinson, and J. B. Fisher. Computer simulation of branch interaction and regulation by unequal flow rates in botanical trees. *American Journal of Botany*, 68:569–585, 1981.

[19] M. Jaeger and P. de Reffye. Basic concepts of computer simulation of plant growth. *Journal of Biosciences*, 17(3):275–291, 1992.

[20] J. Kaandorp. Modelling growth forms of sponges with fractal techniques. In A. Crilly, R. Earnshaw, and H. Jones, editors, *Fractals and chaos*. Springer-Verlag, 1991.

[21] N. Kanamaru and K. Takahashi. CG simulation of natural shapes of botanical trees based on heliotropism. *The Transactions of the Institute of Electronics, Information, and Communication Engineers*, J75-D-II(1):76–85, 1992. In Japanese.

[22] A. Lindenmayer. Mathematical models for cellular interaction in development, Parts I and II. *Journal of Theoretical Biology*, 18:280–315, 1968.

[23] R. Llewellyn. *Ellegance and eccentricity: Ornamental and architectural features of historic British gardens*. Ward Lock Limited, London, 1989.

[24] C. MacKenzie. Artificial evolution of generative models in computer graphics. Master's thesis, University of Calgary, 1993.

[25] P. Oppenheimer. Real time design and animation of fractal plants and trees. Proceedings of SIGGRAPH '86, in *Computer Graphics*, 20(4):55–64, 1986.

[26] P. Prusinkiewicz. Modeling and visualization of biological structures. In *Proceedings of Graphics Interface'93*, pages 128–137, 1993.

[27] P. Prusinkiewicz and J. Hanan. Visualization of botanical structures and processes using parametric L-systems. In D. Thalmann, editor, *Scientific Visualization and Graphics Simulation*, pages 183–201. J. Wiley & Sons, Chichester, 1990.

[28] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*. Springer-Verlag, New York, 1990. With J. S. Hanan, F. D. Fracchia, D. R. Fowler, M. J. M. de Boer, and L. Mercer.

[29] P. Prusinkiewicz, A. Lindenmayer, and J. Hanan. Developmental models of herbaceous plants for computer imagery purposes. Proceedings of SIGGRAPH '88, in *Computer Graphics*, 22(4):141–150, 1988.

[30] P. Prusinkiewicz and D. McFadzean. Modeling plants in environmental context. In *Proceedings of the Fourth Annual Western Computer Graphics Symposium*, pages 47–51, Banff, 1992.

[31] P. Prusinkiewicz, W. Remphrey, C. Davidson, and M. Hammel. Modeling the architecture of expanding *Fraxinus pennsylvanica* shoots using L-systems. To appear in the *Canadian Journal of Botany*, 1994.

[32] W. T. Reeves and R. Blau. Approximate and probabilistic algorithms for shading and rendering structured particle systems. Proceedings of SIGGRAPH '85, in *Computer Graphics*, 19(3):313–322, 1985.

[33] A. R. Smith. Plants, fractals, and formal languages. Proceedings of SIGGRAPH '84, in *Computer Graphics*, 18(3):1–10, 1984.

[34] R. Verey. *Classic garden design*. J. Murray Limited, London, 1989.

[35] G. Wyvill. Implicit surfaces in CSG systems. In *SIGGRAPH 1990 course notes: Modeling and animating with implicit surfaces*, pages 3.1–3.7, 1990.

[36] G. Wyvill, C. McPheeters, and B. Wyvill. Data structure for soft objects. *The Visual Computer*, 2(4):227–234, 1986.