Structural Simulation of Tree Growth

Article *in* The Visual Computer · May 2003

DOI: 10.1007/s00371-002-0189-4 · Source: DBLP

3 authors, including:

John C. Hart
University of Illinois, Urbana-Champaign
151 PUBLICATIONS **5,595** CITATIONS

SEE PROFILE

Structural Simulation of Tree Growth and Response

John C. Hart
School of EECS
Washington State University
Pullman, WA 99164-2752
hart@eecs.wsu.edu

Brent Baker
Ogden Profession Services
NHEERL — Western Ecology Division
U.S. Environmental Protection Agency
Corvallis, OR 97333
brent@heart.cor.epa.gov

Manuscript under review: SIGGRAPH '96

Abstract

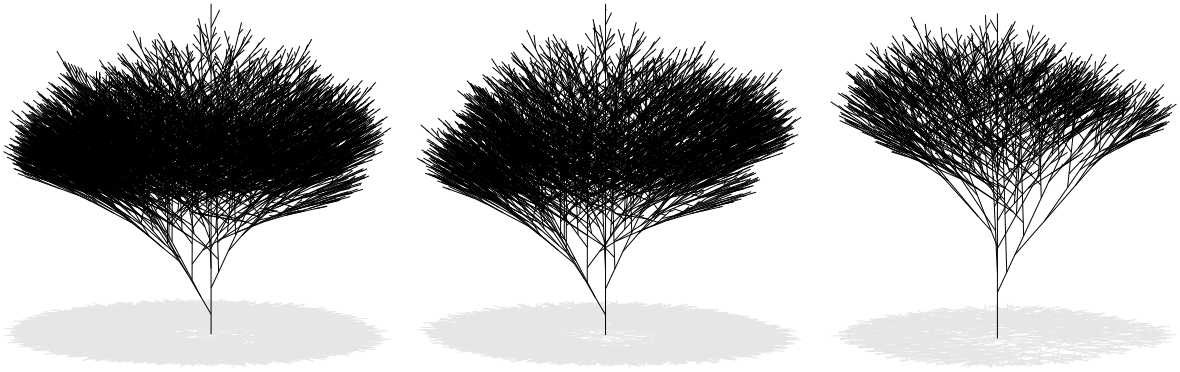
Each tree is unique because of the physical environment it experiences over the course of its life. Environmental factors shape a tree, within the bounds of its genotype. Only by modeling the environment influences can we create realistic models of trees. Our effort is motivated by a desire to construct a predictive tool that can be used by both those in computer graphics and forest management, with applications in image synthesis, dendrochronology, mensuration and the simulation of forest succession.

The structural simulation calculates the mass of each branch of the tree to emulate the mechanisms the tree uses to balance its weight, and estimates the photosynthesis return of the leaves to simulate phototropism. We have also created new techniques for blending branches and mapping bark onto the tree surface. Since trees are commonly identified by the appearance of their bark, we have also attempted to model bark that is typical of a particular species, rather than just realistic in general.

1 Introduction

Tree development is a continuous compromise between the maximization of available light and the minimization of structural stress. Computer graphics research has produced many models of trees and simulations of tree development based on genetic models of the branching process (e.g. L-systems). These trees have even incorporated some environmental influences such as geotropism (sagging branches).

This work separates the environmental information from the genetic information in tree development. While a tree grown in space might exhibit many of the self-similar traits predicted by their genetic growth models, real trees are exceptionally non-self-similar. Figure 1 illustrates how the external influences of gravity and sunlight affect tree form.



Genetic Balanced Phototropic

Figure 1: A purely genetic tree, a tree with its center of mass positioned over its trunk and a tree influenced by sunlight.

As made abundantly clear in [Thompson, 1942], trees (and most other natural forms) derive their strength from surface tension. The strength of a tree's trunk is therefore proportional to its cross-sectional area. As the tree grows, its mass increases cubicly, but its cross-sectional area (and therefore its support strength) increases quadratically. While young trees need not concern themselves with such mechanical constraints, a mature trees must not grow too tall lest its trunk collapse under its own weight.

The new simulation limits the growth of limbs to insure their supporting limbs are sufficiently strong. The model also centers the tree's center of mass over the trunk to reduce

rotational stress. The "genetic" tree in Fig. 1 might topple over because, unlike the "balanced" tree in this same figure, purely genetic development ignores this rotational stress.

The process of photosynthesis is also simulated, producing phototropic (light seeking) effects in the tree throughout the development process. By killing shaded limbs, the "phototropic" tree in Fig. 1 is able to collect approximately the same amount of light energy as the "genetic" tree, but with only 25% of the branches (and mass)[1].

A new blending model is devised to smoothly interpolate the branches of the tree and model the "reaction wood" added at branching points to counter the rotational stress caused by a limb's off-center weight. The mapping of bark onto the tree surface is accomplished by a particle-based flow simulation.

1.1 Applications

A structurally accurate tree model would find many applications, particularly in image synthesis, tree models for dendrochronology and mensuration, and in the simulation of forest succession.

Whereas the dinosaurs in "Jurassic Park" were computer models, their heavily-forested environments were real. Physical interactions between the dinosaurs and their surroundings were necessary for proper suspension of disbelief. The combination of synthetic dinosaurs with a real forest required carefully scripting and meticulously retouching. A highly realistic tree model would eliminate much of this work (and cost) by allowing designers to create realistic geometric models of forests that could interact more completely with computer generated characters.

Dendrochronology discerns past events, particularly weather, from the study of the tree's annual growth rings [Schweingruber, 1993]. The representation produced by this papers growth simulation is capable of producing such growth rings.

Mensuration estimates various forestry properties, e.g. lumber output, through the analysis of easily measured tree dimensions, e.g. trunk diameter at breast height [Avery &

[1]Fig 1 represents only the structure of the trees (using line segments) in winter (without leaves). Hence, the shadow of the "photosynthesis" tree is misleadingly sparse. With leaves, the shadows of all three trees are similarly dense.

Burkhart, 1994]. This paper's growth model utilizes these techniques to increase its accuracy, and in turn, returns a geometric model of mature trees to forest science for more accurate mensuration.

Forest succession is the complex process controlling the composition of trees in a forest. As the number of national "old-growth" forests declines, some have considered the task of re-creating old-growth effects in a new forest [Lorimer *et al.*, 1994]. Current measurements from existing old-growth forests could be used to simulate an old growth forest through selective human intervention in a new forest. The succession process of old-growth forests is a topic of current research. Detailed computer simulation provides new insight into the delicate equilibrium of old-growth forests. The developmental tree simulation described in this paper is a step toward such study.

1.2 Assumptions and Approximations

This paper is concerned with trees that have a single, non-hollow trunk, and exhibit and annual growth cycle. The branches in this model do not bend. Furthermore, once a branch has sprouted, its orientation remains fixed. Each branch is represented geometrically with a cone clipped by two planes perpendicular to its major axis.

Nature has arrived at very diverse solutions for some common problems, and the extremes of these solutions are outside the scope of this paper. Structural changes caused by disease or damage by insects are not modeled, nor are the effects of typical genetic anomalies such as spiral grain or burls. Although the living sapwood at the perimeter of each branch is somewhat weaker than the dead heartwood in its center, formal tests are not often conducted on the sapwood. Hence the heartwood measurements are used [Forest Products Laboratory, 1974] and the simulation treats the tree's volume uniformly.

2 Existing Tree Models in Computer Graphics

Trees possess an immense amount of detail, and some have worked on non-developmental models of mature trees that organize their geometry for best efficiency. Oppenheimer [1986]

modeled trees based on affine transformation of primitives for interactive design. Hart & DeFanti [1991] also used affine transformations to ray trace forests of highly-detailed tree models.

Trees pose difficult geometric and texturing problems. Bloomenthal [1985] simulated the maple tree using generalized cylinders, and developed a parametric blending model to maintain continuity in geometry and texture around branching points. A bark bump-map was made from an x-ray of real bark. Hart [1995] formulated an implicit blending model for interpolating a procedural bark displacement-map texture through a branching point. Section 5 describes a new technique, based on force flow approximation, for representing and texturing the surface of a tree.

Many have used observations documented in the botanical research community to construct non-developmental tree models. de Reffye *et al.* [1988] simulated trees branching and growth based on rules derived from the classifications and observations documented in the botanical research community. They mention the possibility of simulating external forces such as wind and gravity, but as a post-process to make the tree appear more realistic.

Weber & Penn [1995] continued this observation-based trend with a highly-detailed non-developmental tree model with a tremendous number of parameters to represent a wide variety of tree species. These parameters are based on various physical properties of the tree, and can be easily observed and recorded. They simulate several structural properties of trees, such as stem bending, but these actions were controlled by static input parameters.

Developmental simulations are more time consumptive and less space efficient than static models, but are necessary to capture the natural form of trees [Thompson, 1942].

Prusinkiewicz *et al.* [1988] used L-systems to encode biological observations to simulate the structure and growth of plants. They focused on non-woody plants, which better adhere to the "internal control mechanisms" as opposed to trees whose shape is determined by "the environment, competition between trees and tree branches, and accidents." Prusinkiewicz & Lindenmayer [1990] summarized a variety of previous L-systems controlling the basic shape of trees where the interaction with the environment was limited to rotating branches to face incoming light (planartropism) and adjusting branching directions to simulate the effect of gravity (geotropism).

An environmental query operator was later added to the L-system paradigm [Prusinkiewicz *et al.*, 1994] that allowed shrubs to be pruned to a predefined shape to simulate "topiary." Context-sensitivity in the L-system was used to pass a signal along the stem to emulate the plant's method of reacting to its environment. While each pruned shrub was topologically connected, no calculation was made to determine if the resulting sculpture was structurally viable.

Arvo & Kirk [1988] simulated the environmentally-sensitive growth of vines using ray-casting to determine proximity. Greene [1989] discretized space into a voxel array to more efficiently sample the environment at each step of the growth process of such vines. These vines actively grew toward light sources, avoiding shadowing obstacles when necessary, by sensing direct light from a moving sun and indirect light from the sky. Two striking figures illustrate the vined approximation without the supporting model, but lacked any calculation of the resulting structure's stability.

Viennot *et al.* [1989] examined the *ramification matrix,* used in the study of river networks, as a model for binary tree growth. They also simulated the reproduction of trees by an interpolation of their ramification matrices. Holton [1994] investigated the *strand model* (based on the pipe model used to study river networks and bronchi of the lungs) for simulating the vascular structure of trees. The strand model simulated many properties of a tree limb, such as diameter, length and branching angle, based on the number of strands in the limb. External forces were also simulated. The direction of trunk growth was adjusted in the direction of the central axis of the branching structure. Branches were also affected by the pull of gravity, and could be directed to grow upward, horizontal and/or planarly. Phototropism was elegantly approximated by having branches grow in directions away from the center of the tree.

3 The Physics of Trees

Whereas much of the physics used in computer graphics is optics and dynamics, trees are best understood through the use of *statics.* Such analysis reveals that the shape of a tree is significantly influenced by its response to the forces affecting its structure.

3.1 Statics

Dynamics (kinematics and kinetics) simulate an object in motion. Typically, objects are rigid bodies connected by springs, and forces alter position and orientation [Barzel & Barr, 1988]. Statics simulate objects at rest. Such objects are typically beams connected by pins into a rigid structure, and external forces that would ordinarily alter position and orientation are countered by equal and opposite internal forces. Statics focuses on the computation of these internal forces.

The following principles of statics [Beer & Johnston, 1984] use the notation $\mathbf{F}(\mathbf{x})$ to denote a force $\mathbf{F}$ acting on a point $\mathbf{x}$ of some rigid body. The result of two forces $\mathbf{F}_1, \mathbf{F}_2$ acting on the same point $\mathbf{x}$ of a rigid body is equivalent to that of a single force equal to the sum of the original forces

$$(\mathbf{F}_1 + \mathbf{F}_2)(\mathbf{x}) \equiv \mathbf{F_1}(\mathbf{x}) + \mathbf{F_2}(\mathbf{x}). \tag{1}$$

The result of two parallel forces $\mathbf{F}_1, \mathbf{F}_2$, acting on a rigid body along the same line of action is equivalent to that of a single force equal to the sum of the original forces

$$(\mathbf{F}_1 + \mathbf{F}_2)(\mathbf{x}_1) \equiv \mathbf{F_1}(\mathbf{x_1}) + \mathbf{F_2}(\mathbf{x_2}) \text{ iff } \mathbf{F}_1 \times \mathbf{F}_2 = \mathbf{0} \text{ and } (\mathbf{x}_2 - \mathbf{x}_1) \times \mathbf{F}_1 = \mathbf{0}. \tag{2}$$

The *principle of transmissibility* therefore states that a force at a point on a rigid body can be replaced by a force of equal magnitude and direction acting on a different point along the same direction of action

$$\mathbf{F}(\mathbf{x}) \equiv \mathbf{F}(\mathbf{x} + \alpha\mathbf{F}) \quad \forall\alpha. \tag{3}$$

The result of two forces acting on different points of the rigid body, along different lines of action, is equivalent to a force and a *couple.* A couple is a pair of opposite forces $\mathbf{F}_1 = -\mathbf{F}_2$ acting on different points $\mathbf{x}_1, \mathbf{x}_2$ of a rigid body. A couple causes a rigid body to rotate, and is typically represented by the *moment of the couple* $\mathbf{M}$

$$\mathbf{M} = \mathbf{F}_1 \times (\mathbf{x}_2 - \mathbf{x}_1). \tag{4}$$

The force of gravity acts simultaneously on every point in a rigid body. Using the above rules this gravitational force can be simplified, replaced by a single force acting on the *center*

of mass of the rigid body. The center of mass $\bar{\mathbf{x}}$ of an object of uniform density is given by

$$\bar{\mathbf{x}} = \frac{1}{v} \int \mathbf{x} \, dV \tag{5}$$

where $v = \int dV$ is the volume of the object.

3.2 Reaction Wood

There are two general classes of trees: gymnosperms and angiosperms. *Gymnosperms* are so named because their seeds are not enclosed, but are external. These include ginkos, pine, spruce, fir and other conifers whose seeds are carried in an open cone. The seeds of *angiosperms* are enclosed in a mature ovary, a fruit. These species include those called deciduous and most tropical species. The classification is important because the two groups have developed different mechanisms to resist mechanical stress.

Several factors can create an unbalanced load on a structural member of a tree. The most common is the tree's own growth. Branches are occasionally lost, due to external trauma or shade from higher branches. Trees counteract these forces by growing *reaction wood* [Mattheck, 1991].

Reaction wood either pushes a branch (*compression wood*) or pulls a branch (*tension wood*) to counteract rotational force. Compression wood is formed by gymnosperms whereas tension wood is formed by angiosperms[2] [Esau, 1960]. Both can be identified as a thickening, or eccentricity, of the annual ring on the side of the branch where it forms.

3.3 Structural Analysis of Branching Points

Statics provides the necessary tools to compute the internal forces of rigid bodies with connections involving various degrees of freedom, such as trusses, frames and machines. The tree is an example of a frame.

[2]Some authorities maintain that angiosperms form both tension wood and compression wood scattered around the perimeter of the branch, and that reaction wood in angiosperms does not always create a stem eccentricity [Gartner, 1995]. In this work, we assume angiosperms always react with tension wood.

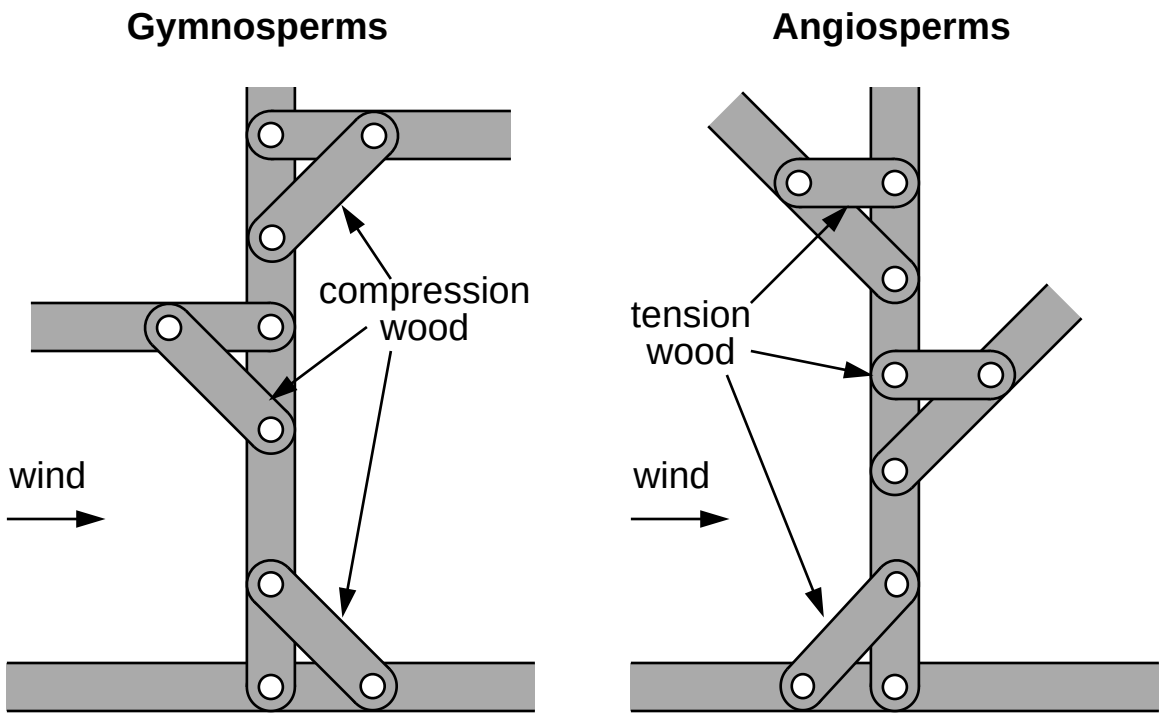


Figure 2: Structural analysis of reaction wood.

Figure 2 illustrates structural models of both kinds of trees and their appropriate reaction wood. Performing a static analysis at the branch points indicates how much counteracting reaction wood the tree needs to grow.

3.4 Mechanical Properties of Wood

Natural structures derive their strength from surface tension [Thompson, 1942]. Hence, the mass supported by a tree's trunk is proportional to its cross-sectional area [Avery & Burkhart, 1994].

Let

$$\text{supports}(R, d) = kdR^2 \tag{6}$$

be the maximum mass that a limb of basal radius R and density d can support. Let its inverse

$$\text{needstosupport}(m, d) = \sqrt{\frac{m}{kd}} \tag{7}$$

return the minimum radius necessary to support a mass m.

The constant of conversion k can be derived from wood strength tables, e.g. [Forest Products Laboratory, 1974]. A simpler definition comes from the observation that trees structurally cannot exceed 300 feet in altitude [Thompson, 1942]. This defines k such that it solves

$$\text{supports}(R, d) = dv(R, 0, 300\text{feet}), \tag{8}$$

specifically $k = \pi \cdot 100$ feet.

4 Simulating Tree Growth and Response

4.1 Growth and Response Algorithm

Each iteration of tree growth follows the following algorithm steps.

1. compute mass
2. compute center of mass
3. compute photosynthesis
4. compute growth rates
5. grow branches

The tree is simulated as a collection of limbs, organized in a hierarchical tree structure (of course). Table 1 lists the limb variables used in this process. The *trunk* is defined as the limb with index $i = 0$.

The limb's spatial variables are computed with respect to a canonical limb coordinate system. The branch's base is at the origin of this coordinate system, and the branch grows along the positive y axis. The instancing transformation T_i consists only of a rotation and a translation, and maps branch coordinate onto world coordinates[3].

[3]Since the branches' orientation and base position remain fixed with respect to world coordinates in this simulation, the instancing transformation is not necessary. It is included to support future simulations that would allow branches (and their children) to change orientation.

Constant across tree	
d	density of wood
Variable for each limb i	
T_i	instancing transformation
l_i	length of branch
R_i, r_i	branch base (tip) radius
t_i	age of branch
v_i	volume of branch
m_i, M_i	mass of limb (everything supported)
e_i	limit on mass of new growth
$\bar{\mathbf{x}}_i, \bar{\mathbf{X}}_i$	center of mass of limb (everything supported)
o_i	adjustment to balance tree
g_i, G_i	rate of growth for limb (and children)
p_i, P_i	photosynthesis of limb (everything supported)
J_i	indices of children

Table 1: Parameters and variables used during tree growth and response.

Compute Mass: The mass of a branch $m_i = dv_i$ is the product of the density of its wood d with its volume v_i, where

$$v_i = v(l_i, R_i, r_i) = \frac{1}{3}\pi l_i \frac{R_i^3 - r_i^3}{R_i - r_i} \tag{9}$$

is the volume of a truncated cone. The mass a branch supports M_i is recursively the mass of the branch plus the mass its children support

$$M_i = m_i + \sum_{j \in J_i} M_j. \tag{10}$$

Compute Center of Mass: The center of mass for a branch is the point

$$\bar{\mathbf{x}}_i = T_i \ (0, \frac{1}{4} l_i \frac{R^4 - r^3(4R - 3r)}{(R - r)(R^3 - r^3)}, 0, 1)^T. \tag{11}$$

For a cone ($r = 0$) the center of mass is simply $(0, \frac{1}{4}l_i, 0)$.

The center of mass for everything a branch supports is computed recursively

$$\bar{\mathbf{X}}_i = \frac{1}{M_i} \left(\bar{\mathbf{x}}_i m_i + \sum_{j \in J_i} \bar{\mathbf{X}}_j M_j \right). \tag{12}$$

Compute Offset: The offset o_i indicates the effect of branch growth on the tree's center of mass

$$o_i = T_i \mathbf{y} \cdot \bar{\mathbf{X}}_0 \tag{13}$$

where $\mathbf{y} = (0, 1, 0, 0)^T$ indicates the limb's growth direction in limb coordinates. This value is then clamped to $[-1, 1]$.

Compute Photosynthesis: Photosynthesis is simulated by rendering the tree from the light source's point of view. Instead of shading the leaves their usual color, they are instead colored with the index i of the limb they are attached to. The limbs themselves gather no light and so are rendered in the background color (e.g. a limb index of -1). The amount of photosynthetic nutrition returned to each limb by its leaves p_i is proportional to the percentage of pixels in the rendered image whose index matches the limb.

Trees receive light directly from a moving sun, and indirectly from the sky[4]. Hence, for each growth iteration we take a single stochastic sample of photosynthesis, taking samples

[4]This model neglects the indirect light received from diffuse interreflection and transmission.

from the sun's path more than other locations proportionately to the ratio of the sun's intensity to the sky's intensity[5].

The total photosynthetic nutrition returned to each limb by its leaves and its children is accumulated recursively

$$P_i = p_i + \sum_{j \in J_i} P_j. \tag{14}$$

Phototropism, the desire of the tree to grow towards light, results from two effects. First, the simulation kills limbs that do not return sufficient photosynthetic nutrition. Second, it increases the growth rate of branches whose leaves return more nutrients.

The shape and size of leaves affects the shape and size of tree crowns [Horn, 1971]. Leaves are represented by polygons that sprout from limbs at fixed intervals where their radius is sufficiently small. The leaves are oriented about their stem axis to face up.

Compute Growth Rate: The growth rate g_i indicates how much the tree should increase the length of limb i. The growth rate is initialized with some minimum growth (e.g. $g_i = 0.1$). The rate is then incremented by the center of mass offset value (e.g. $g_i = g_i + 0.45 + 0.45o_i$) for a maximum total of 1. The rate is also incremented slightly by the branch photosynthesis nutrition $g_i = g_i + p_i$ which can push it above 1.

The total growth rate for the branch and its children is computed recursively as

$$G_i = g_i + \sum_{j \in J_i} G_j. \tag{15}$$

The ratio g_i/G_i dictates the limb's allocation of new growth resources. The rest it must pass on to its children.

Grow Branches: Growth begins at the trunk. Each increase in the trunk radius allows it to support more weight. The first priority is that the trunk be allowed to support its own weight. The remaining residual mass is passed on to the branches to limit their growth.

[5]Greene [1989] used an 80%/20% sun-to-sky weighting. However, Horn [1971] (Fig. 5.1) shows that the nutrients from photosynthesis remains fairly constant from 20% to 100% of full sunlight intensity. While trees may grow toward the brightest point in the sky, direct sunlight is not necessary for effective photosynthesis.

If the current limb is the trunk, then the radius of its base increases amount proportionally to the ratio of nutrition returned by photosynthesis to the mass of the tree

$$R_0 = R_0 + \alpha \text{photosynthesis} \frac{m_0}{M_0}. \tag{16}$$

(The prime indicates the post-growth variable.) The extra mass supported by the trunk's growth is given by

$$e_0 = \text{supports}(R_0) - M_0. \tag{17}$$

If the current limb is not the trunk, then the radius of the current limb increases to support the extra mass made available by the trunk's growth

$$R_i' = R_i + \text{needstosupport}(M_i + e_i). \tag{18}$$

Enough of the extra mass needs to be allocated to expanding the current branch to its new radius. Hence the extra mass is reduced as

$$e_i' = e_i - \Delta m_i \tag{19}$$

where $\Delta m_i = m_i' - m_i$ is the change in the mass of the branch due to the increased base radius. This increase in mass only accounts for the change in base radius and does not lengthen the branch.

The mass of the branch is further increased by extending the branch's length proportionally to its growth rate

$$m_i' = m_i' + \frac{g_i}{G_i} e_i'. \tag{20}$$

Given the new branch mass m_i', the new branch length l_i' is

$$l_i' = \frac{3m_i'}{\pi R_i'^2}. \tag{21}$$

The rest of the mass is pass along to the child branches

$$e_j = \frac{G_j}{G_i} e_i' \tag{22}$$

for each child branch $j \in J_i$.

5 Modeling the Tree Surface

While the union of truncated cones suffices to represent a trees general form [Max, 1990], the realism of the tree's appearance is greatly increased by the blending of branches and the application of bark.

5.1 Adding Reaction Wood

Trees grow reaction wood to counter the rotational moment created by the limb's off-center mass. The amount of reaction wood needed requires static analysis of the supporting structure.

The rotational moment at the branch point is equal to the sum of the rotational moments due to the off-center weight of the branch less the unknown supporting force of the wood applied at some offset δ_i. The equilibrium of the frame's support structure demands the rotational moment at the branch point vanish, therefore

$$\mathbf{W}_i \times T_i \bar{\mathbf{X}}_i = -\mathbf{F}_R \times T_i(0, \delta_i, 0, 0). \tag{23}$$

Since we are only concerned with the scalar δ_i and the magnitude of $\mathbf{F}_R$, we can replace (23) by its scalar equivalent

$$W_i \bar{X}_i \sin\theta_{\bar{X}} = \delta_i F_R \sin\theta_d. \tag{24}$$

The value $\theta_{\bar{X}}$ indicates the angle that the branch's center of mass is offset from above the base of the branch. The value θ_d is related to the branch's divergence/azimuth angle.

The value f_R is the magnitude of the upward component of the available support force of the reaction wood. The actual supporting force of the reaction wood needs to be $F_R/\cos\theta_R$ where θ_R is the angle the reaction wood makes with respect to the up direction ($-\mathbf{W}$). We can assume the reaction wood has the same strength as the wood at the branch's base, so the strength of the reaction wood can be determined using the supports function (6).

Thus, the only unknown is δ_i, whose solution

$$\delta_i = \frac{W_i \bar{X}_i \sin\theta_{\bar{X}}}{F_R \sin\theta_d} \tag{25}$$

indicates the displacement along to branch to place the reaction wood.

5.2 Blending Cones

Bloomenthal & Wyvill [1990] developed a blend of generalized cylinders of constant radius. Given a point $\mathbf{x}$, Let L_1 be the skeletal line segment closest to $\mathbf{x}$, let $\mathbf{x}_1 \in L_1$ be the point on this line segment closest to $\mathbf{x}$. Define L_2 and $\mathbf{x}_2$ similarly for the second-closest skeletal line segment. Let L be the line segment connecting $\mathbf{x}_1$ with $\mathbf{x}_2$.

The cylindrical blending procedure defined d as the distance from $\mathbf{x}$ to L and returned $d - r$ where r was a constant offset radius of the skeleton.

This blending procedure can be extended to blend cones. Let $t_1 \in [0, 1]$ parameterize the cone's major axis line segment L_1. At the base, $t_1 = 0$, the radius is R_1 and at the tip $t_1 = 1$ the radius is r_1. Define t_2, R_2 and r_2 likewise for the second cone. The new blend returns $d - r$ as before, but r is no longer constant. Instead it is defined as follows

$$d_1 = ||\mathbf{x} - \mathbf{x}_1|| \tag{26}$$

$$r_1 = (1 - t_1)R_1 + t_1 r_1 \tag{27}$$

$$d_2 = ||\mathbf{x} - \mathbf{x}_2|| \tag{28}$$

$$r_2 = c(1 - t_2)R_2 + t_2 r_2 \tag{29}$$

$$r = \frac{d_2 r_1 + d_1 r_2}{d_1 + d_2}. \tag{30}$$

Figure 3 demonstrates this blend on a simple branching configuration.

From the force magnitude $F_R / \cos\theta_R$ we can find the cross-sectional length Δr the reaction wood needs to add to the existing wood to offset the branch's moment of rotation. Assuming branch i is the nearest branch to $\mathbf{x}$, let $\Delta\mathbf{x} = \mathbf{x} - \mathbf{x}_1$. The operation

$$r_1 = r_1 + \Delta r c(\frac{t_1 l_i}{\delta_i}) \frac{1 + \mathbf{y} \cdot \Delta\mathbf{x}/||\Delta\mathbf{x}||}{2} \tag{31}$$

increases the radius in the direction of the unit vector $\mathbf{y}$. For gymnosperms, $\mathbf{y}$ points down, and the radius is increased on the bottom side of the base of the branch, thereby modeling compression wood. For angiosperms, the basal radius at the top side of the branch is increased to model tension wood. Figure 4 demonstrates the same configuration as Figure 3 but with reaction wood simulated.

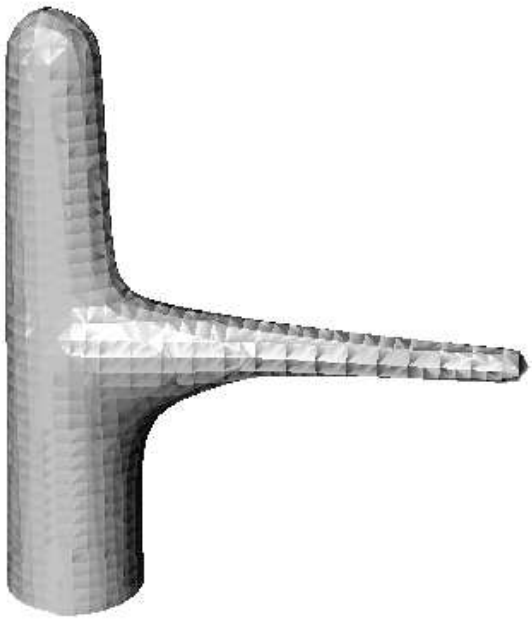


Figure 3: Blend of a gymnosperm branch simulates by two cones.

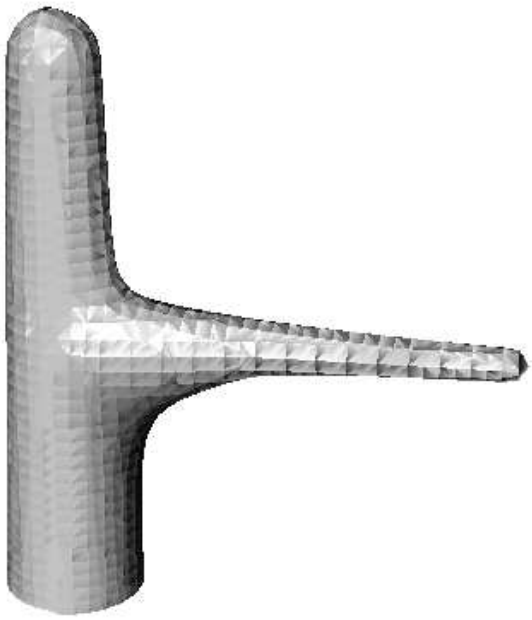


Figure 4: Blend of a gymnosperm branch with reaction wood.

5.3 Particle-System Force-Flow Approximation

Following Szeliski & Tonnesen [1992], Witkin & Heckbert [1994] and Fleischer *et al.* [1995], we texture the tree using particles constrained to the implicit surface. Instead of covering the entire surface with particles as previous techniques have, we create a flow of particles from the base of the trunk to the tip of the branches.

The bark of trees follows the lines of force flow from the structure of the tree [Mattheck, 1991]. While these force flow lines would result from a finite-element structure analysis, such an expensive procedure would be overkill. Instead, we simulate the formation of bark through a significantly simpler flow approximation.

Particles are constrained to the surface, beginning at the base of the trunk with an initial upward velocity. The momentum of these particles carries them to the tips of the branches, flowing naturally through branching points. The trails of these particles simulates the grain of the bark.

The constraint of a particle to an implicit surface that is itself defined as an offset to a skeleton is equivalent to connecting the particle to the skeleton with a beam, even if this length of this beam depends on the location with respect to the skeleton.

Zahlten & Jurgens [1991] used chains of voxels to polygonize implicit surfaces. A chain is a connected ring of cells, where cells could be cubes or simplices (tetrahedra). This chain is pulled across a surface, and its intersection is polygonized.

We can similarly use a chain of particles to interrogate the implicit tree surface. Instead of voxels, particles are constrained to lie on the implicit surface. These particles are connected into a ring by springs of zero rest length. Giving the particles an initial velocity pushes the chain across the implicit surface, thereby interrogating it.

Surfaces such as trees, while simply connected, require the chain to subdivide into several loops for proper interrogation. The reconnection algorithm is listed in Table 2, and Figure 5 demonstrates this algorithm before and after a reconnection. The function limb(a) returns the limb that particle a is attached to.

The parameters of the chain-of-particles are quite delicate and will require some experimentation for reasonable performance. Our trees surfaces were approximated by the

relinked = FALSE

$b_0 = b$

$a_0 = a$

while limb(a_0) $\neq$ limb(b_0)

repeat

$a = b$

b = limb(b)

if $b = a_0$ then return(relinked)

until limb(a) $\neq$ limb(b) & $||\mathbf{x}(b_0) - \mathbf{x}(a)||^2 < ||\mathbf{x}(b) - \mathbf{x}(a)||^2$

relinked = TRUE

unlink(a_0, b_0)

link(a, b_0)

unlink(a, b)

link(a_0, b)

$b_0 = b$

return(relinked)

Table 2: Relinking algorithm.

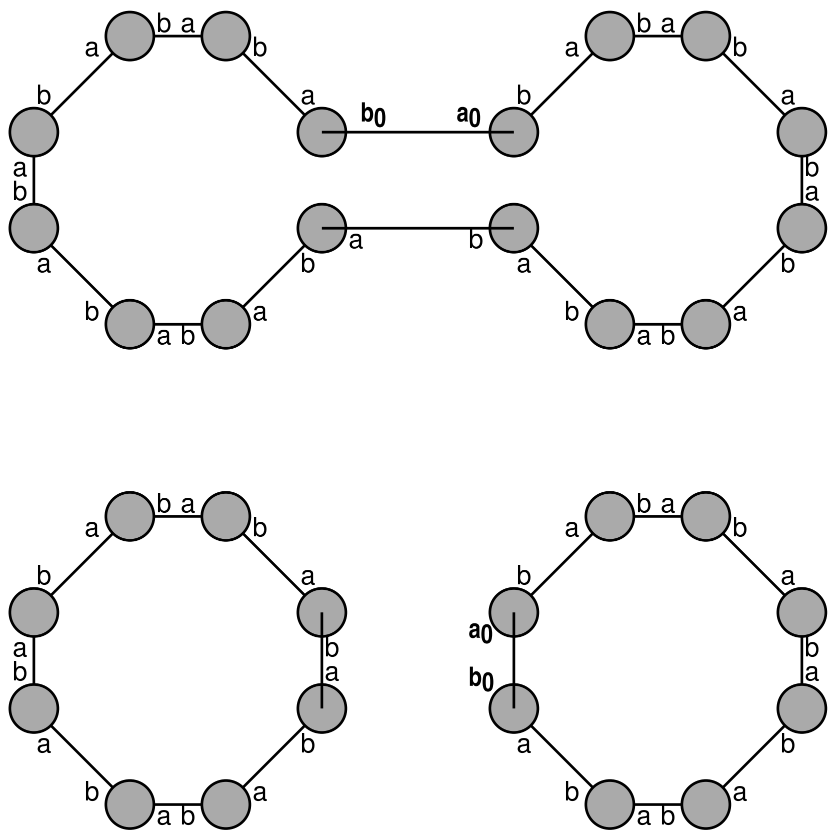


Figure 5: Before and after relinking.

chain-of-particles with the following parameters:

- Time steps per second: 100.
- Initial velocity of particles: 2 units/sec up.
- Maximum Link Length: .15 units.
- Minimum Link Length: .05 units.
- Spring Constant Between Links: 100.
- Dampening Constant Between Links: 1.
- Spring Constant Between Links and Tree Skeleton: 100.
- Dampening Constant Between Links and Tree Skeleton: 10.

Figure 6 demonstrates the result of the particle flow on an unblended branching point.

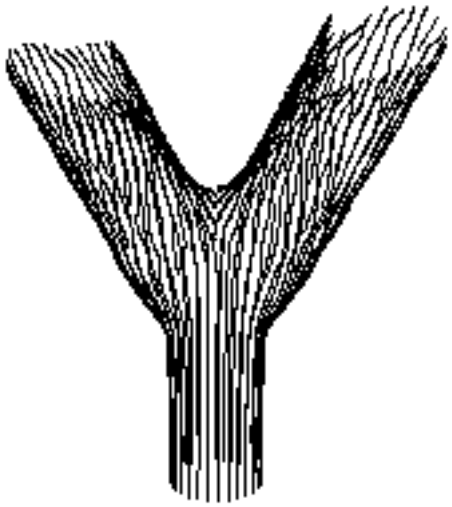


Figure 6: Particle flow around a branching point.

5.4 Modeling Bark

We have attempted to model the bark of the tree realistically. The appearance of bark is often used to identify a species growing in the forest. The appearance of bark changes as

the tree ages, in some ways analogous to the aging of human skin. In many species the bark of younger trees is fairly smooth. As the tree ages the bark becomes much more heavily textured, and can shift in color. We have used [Little, 1994] as a guide for the appearance of the bark of the species we model.

Generating the bark texture takes place in two phases. The first phase attaches a "background" polygon to each particle from the dynamic particle flow stage of the model. Currently, this polygon is a quadrilateral. Its size is adjusted until the surface of the tree is covered, essentially polygonizing the surface of the tree. Distances between particles are small enough that these polygons can be flat shaded. The second phase attaches a "foreground" feature, made up of several polygons, to some or all of the background polygons, depending on a density parameter. Figure 7 provides a general example.

Figure 7: A general bark texturing example.

We've classified bark appearance into two general categories, *spotted* and *ribbon*. This grouping is based solely on appearance and is not connected to the tree's biological classification.

Quaking aspen (Figure 8, and many birch and alder species have spotted bark. Each spot is a fan of triangles whose outer edge touches the background and whose peak is an

adjustable distance above the background. For other species, the entire spot can be raised above the background, and the size increased to almost cover the surface. We used spotted bark where the height to width ratio of the foreground features does not exceed 2:1.

Figure 8: A Quaking Aspen example of spotted bark.

Ribbon bark consists of a long, narrow segmented ribbon that extends above and below the particle it is attached to. Several layers of differing heights can be generated and overlayed. Ribbons are aligned with the growth axis of the trunk/limb, but with a controlled amount of random twist introduced. In many cases, ribbon foreground completely obscures the background. Most variations of maple have ribbon bark (Figure 9, as well as many oaks, junipers, cedars, and redwood.

6 Conclusion

Analysis of the mechanical properties of tree growth has provided a simulation with parameters more physically meaningful than previous techniques. While the simulation's cones do not photorealistically represent the bending nature of real trees, they do capture their general balanced form and composition, which has been a missing component in previous

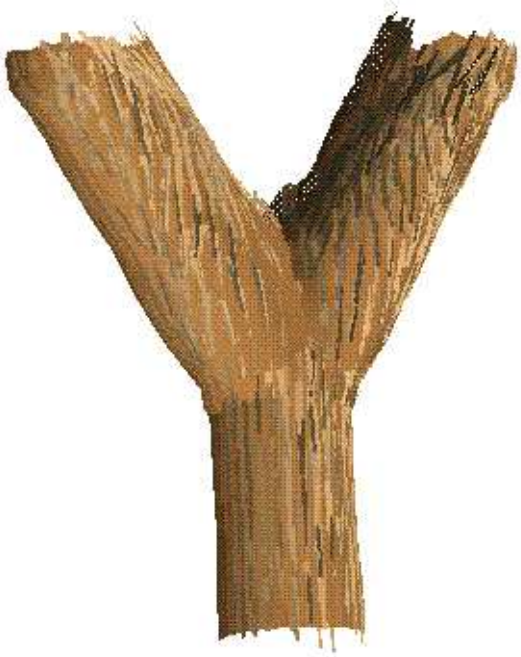


Figure 9: A Maple-like example of ribbon bark.

work.

Saving all of the tree surfaces created during the growth of the tree yields a "solid" representation of the tree. Slicing such a solid representation yields a model of the tree rings resulting from the tree's annual growth cycle. Since the growth rate of the tree is affected by balance and available light, certain dendrochronological elements of the tree's simulated history are revealed by this growth ring model.

As tree models become more physically-based and more accurately represent real tree structure, their dimensions also become more meaningful. Forest mensuration attempts to measure the lumber yield of entire forests. Currently these models are constructed through destructive sampling techniques. A physically-meaningful computational model of the tree permits the non-destructive accurate prediction of forest yield.

The texturing of bark has long been problematic in computer graphics. The physically-inspired bark simulation strives for a more realistic depiction of specific bark. Beyond its obvious uses in image synthesis, it may find use as a multimedia tool for the identification of tree species based on an interactive query of bark appearance. The rough texture of bark changes appearance under different lighting conditions, simulating similar lighting may aid

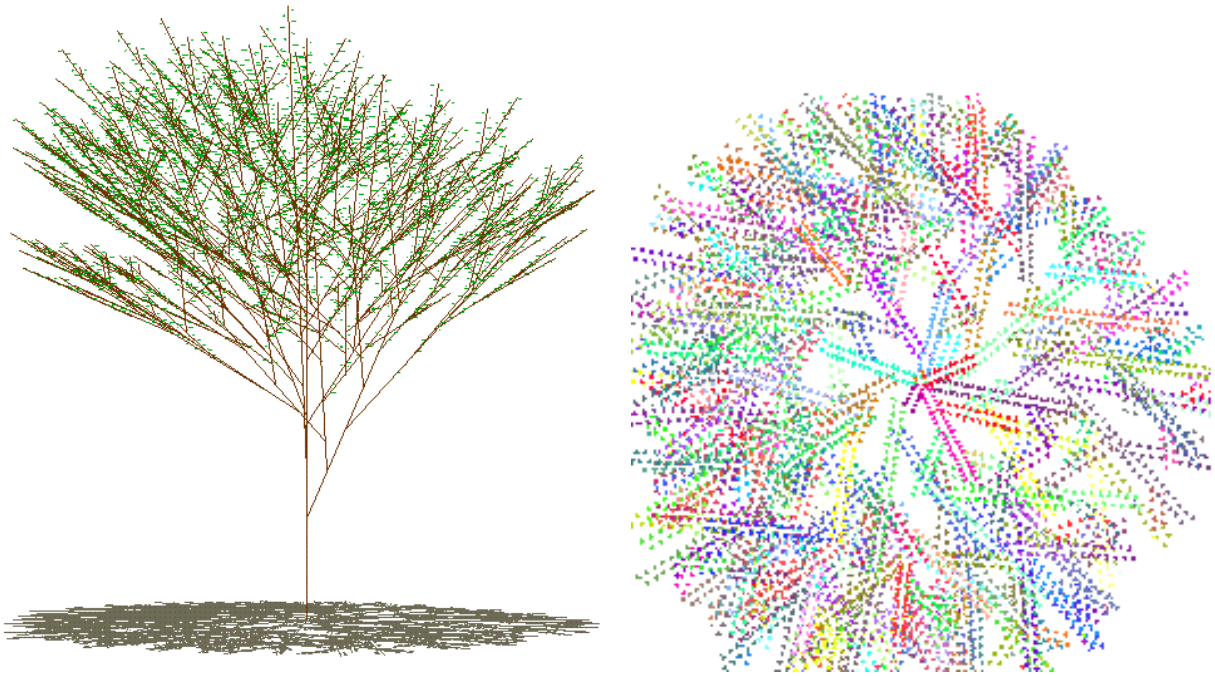


Figure 10: View of tree development (left) and photosynthesis computation (right).

in visually matching an unknown tree bark with its computationally cataloged counterpart.

The current system supports single tree growth. Simulating the simultaneous development of a forest of trees as they battle for resources should result in further insight into the forest succession process.

6.1 Implementation Notes

The tree simulation was implemented in an interactive multiwindow system, using the polygon-rendering and user-interaction tools available in the OpenGL (Mesa) and GLUT libraries.

The tree growth window currently displays a perspective view of a wireframe model of the tree, shown in Figure 10 (left). During growth, the user can click on a branch to prune it. The change in the center of mass causes the tree simulation to respond by encouraging its existing branches to compensate.

The phototropism window (Figure 10 right) displays the indexed orthographic view of the leaves from a random position in the sky. The edges of this window form a natural aperature

of available light such as a tree might receive if growing in a forest clearing. Obstacles can be placed in the scene that obscure this window's view, thereby simulating various phototropic effects that control tree shape.

6.2 Further Research

This first step at mechanically simulating tree growth inspires an immense number of new directions. Many of the mechanical computations operated at the simplest level of statics analysis. More rigorous static analysis of specific tree structural members will lead to improved accuracy.

The most immediate next step is to simulate other structural forces, specifically wind. Trees at the edge of a forest grow shorter to reduce the moment of rotation on the trunk caused by wind resistance. Trees in the middle of a forest are shielded from the wind by the perimiter trees, and are free to grow much taller and fuller. When a road is cut through a forest, the force of wind resisitance is immediately introduced to middle trees, which may then topple.

We are also interested in simulating the way the bark on trees heals its wounds. Such healing would require the tree surface be completely covered with active particles, instead of the chain-of-particles approach used in this paper.

An initial tangential velocity component can cause the bark particles to simulate spiral bark, but it is currently unclear how such bark spins its way around a branching point.

References

[Arvo & Kirk, 1988] Arvo, J. and Kirk, D. Modeling plants with environmentally-sensitive automata. Proc. of *Ausgraph '88*, 1988, pp. 27–33.

[Avery & Burkhart, 1994] Avery, T. E. and Burkhart, H. E. *Forest Measurements*. McGraw-Hill, New York, 1994.

[Barzel & Barr, 1988] Barzel, R. and Barr, A. H. A modeling system based on dynamic constraints. *Computer Graphics* 22(4), Aug. 1988, pp. 179–188.

[Beer & Johnston, 1984] Beer, F. P. and Johnston, Jr., E. R. *Vector Mechanics for Engineers: Statics and Dynamics.* McGraw-Hill, New York, 1984.

[Bloomenthal & Wyvill, 1990] Bloomenthal, J. and Wyvill, B. Interactive techniques for implicit modeling. *Computer Graphics* 24(2), Mar. 1990, pp. 109–116.

[Bloomenthal, 1985] Bloomenthal, J. Modeling the mighty maple. *Computer Graphics* 19(3), July 1985, pp. 305–311.

[de Reffye *et al.*, 1988] de Reffye, P., Edelin, C., Francon, J., Jaeger, M., and Puech, C. Plant models faithful to botanical structure and development. *Computer Graphics* 22(4), August 1988, pp. 151–158.

[Esau, 1960] Esau, K. *Anatomy of Seed Plants.* Wiley, 1960.

[Fleischer *et al.*, 1995] Fleischer, K. W., Laidlaw, D. H., Currin, B. L., and Barr, A. H. Cellular texture generation. In *Computer Graphics (Annual Conference Series)*, Aug. 1995, pp. 239–248.

[Forest Products Laboratory, 1974] Forest Products Laboratory. *Wood Handbook, Wood as an Engineering Material.* Forest Service, U.S. Dept. of Agriculture, 1974.

[Gartner, 1995] Gartner, B. Personal communication, 1995. Oregon State University, Dept. of Forest Products.

[Greene, 1989] Greene, N. Voxel space automata: Modeling with stochastic growth processes in voxel space. *Computer Graphics* 23(3), July 1989, pp. 175–184.

[Hart & DeFanti, 1991] Hart, J. C. and DeFanti, T. A. Efficient antialiased rendering of 3-D linear fractals. *Computer Graphics* 25(3), 1991.

[Hart, 1995] Hart, J. C. Implicit representations of rough surfaces. In Proc. of *Implicit Surfaces '95.* (Eurographics Workshop), April 1995, pp. 33–44.

[Holton, 1994] Holton, M. Strands, gravity and botanical tree imagery. *Computer Graphics Forum* 13(1), 1994, pp. pp 57–67.

[Horn, 1971] Horn, H. S. *The Adaptive Geometry of Trees.* Princeton University Press, Princeton, NJ, 1971.

[Little, 1994] Little, E. L. *National Audubon Society Field Guide to North American Trees (Western Region).* Knopf, New York, 1994.

[Lorimer *et al.*, 1994] Lorimer, N. D., Haight, R. G., and Leary, R. A. The fractal forest: Fractal geometry and applications in forest science. Technical Report NC-170, USDA Forest Service North Central Forest Experiment Station, 1994.

[Mattheck, 1991] Mattheck, C. *Trees: The Mechanical Design.* Springer-Verlag, New York, 1991.

[Max, 1990] Max, N. Cone-spheres. *Computer Graphics* 24(4), August 1990, pp. 59–62.

[Oppenheimer, 1986] Oppenheimer, P. E. Real time design and animation of fractal plants and trees. *Computer Graphics* 20(4), Aug. 1986, pp. 55–64.

[Prusinkiewicz & Lindenmayer, 1990] Prusinkiewicz, P. and Lindenmayer, A. *The Algorithmic Beauty of Plants.* Springer-Verlag, New York, 1990.

[Prusinkiewicz *et al.*, 1988] Prusinkiewicz, P., Lindenmayer, A., and Hanan, J. Developmental models of herbaceous plants for computer imagery purposes. *Computer Graphics* 22(4), August 1988, pp. 141–150.

[Prusinkiewicz *et al.*, 1994] Prusinkiewicz, P., James, M., and Mech, R. Synthetic topiary. In *Computer Graphics (Annual Conference Series)*, July 1994, pp. 351–358.

[Schweingruber, 1993] Schweingruber, F. H. *Trees and Wood in Dendrochronology.* Springer-Verlag, New York, 1993.

[Szeliski & Tonnesen, 1992] Szeliski, R. and Tonnesen, D. Surface modeling with oriented particle systems. *Computer Graphics* 26, July 1992, pp. 185–194.

[Thompson, 1942] Thompson, D. *On Growth and Form.* University Press, Cambridge, 1942. abridged edition (1961) edited by J. T. Bonner.

[Viennot *et al.*, 1989] Viennot, X. G., Eyrolles, G., Janey, N., and Arques, D. Combinatorial analysis of ramified patterns and computer imagery of trees. *Computer Graphics* 23(3), July 1989, pp. 31–40.

[Weber & Penn, 1995] Weber, J. and Penn, J. Creation and rendering of realistic trees. *Computer Graphics* Annual Conference Series, August 1995, pp. 119–128.

[Witkin & Heckbert, 1994] Witkin, A. P. and Heckbert, P. S. Using particles to sample and control implicit surfaces. In *Computer Graphics (Annual Conference Series)*, July 1994, pp. 269–277.

[Zahlten & Jurgens, 1991] Zahlten, C. and Jurgens, H. Continuation methods for approximating isovalued complex surfaces. In *Proc. of Eurographics '91*, 1991.