State of the Art in the Realistic Simulation of Plant Leaf Venation Systems

Julia Taylor-Hell   Gladimir Baranoski
Department of Computer Science, University of Waterloo
200 University Avenue West, Waterloo, Ontario, N2L 3G1, Canada

April 17, 2002
Technical Report CS-2002-17

1 Introduction

In order to create a realistic portrayal of plant leaves in computer graphics, the veins on these leaves must be represented. In the computer graphics industry, an artist may draw a branching vein structure as a texture and paste it onto the leaf model. Because natural scenes are so frequently represented, it would be useful to devise an automatic and predictable technique for simulating leaf venation systems and embedding them to the leaf's geometric model. This technical report examines this problem and reviews the state of the art in the simulation of venation systems.

The next section of this report will provide a background on relevant biological concepts. Section 3 will discuss the requirements for an acceptable modeling technique for plant veins, and Section 4 will examine several potential solutions, analyzing them based on the stated criteria.

2 Background in Botany and Definition of Terms

In this section we present and define terms from biology and botany which are used during our discussion. The main body of a leaf is called its blade, or lamina [15, page 94]. The blade begins from the petiole, which is the stalk of the leaf, and extends to its apex, or tip. The term venation is used to describe the arrangement of vascular bundles of xylem[1] and phloem[2] within the leaf. These veins lie in the mesophyll, a "diffusing and pigmented structure" [18, page 660] which provides the photosynthentic ground tissue of the leaf.

As well as providing a channel for nutrient conduction, venation systems provide the basic structure of the leaf, and are instrumental in the plant's growth. A leaf's venation pattern is useful in species recognition or classification, and is often considered as a plant's "fingerprint" [9, page 9].

2.1 Monocotyledons and Dicotyledons

Two types of plants that will be discussed are monocotyledons and dicotyledons. Monocots have one cotyledon, or seed leaf, whereas dicots have two. Monocot leaves have a parallel or striate venation system, in which equally sized primary veins traverse the lamina lengthwise, coming together at the blade's apex. Smaller veins connect the larger parallel ones, and also run parallel to each other [13, page 628]. Experiments performed by Woolley [18] suggest that a leaf reflects light more diffusely when viewed from a direction perpendicular to the veins. Hence, the parallel venation system of monocotyledons simplifies the simulation of their anisotropy [1, page 7].

In dicots, venation is described as netted or reticulate. Dicot veins follow a branching pattern in which the central vein, or midrib, begins from the petiole and traverses the leaf, branching into successively smaller, secondary and tertiary veins in a recursive fashion [13, page 631]. The secondary veins often extend to the edges of the leaf, and tertiary and higher-order veins create a complex network throughout the mesophyll [4, page 97], creating polygonal subdivisions of lamina called areoles. Within these areoles even smaller veinlets come to their ends in the mesophyll, within approximately 0.25 millimeters of each other [4, page 631]. An observation of the branching angle between subveins usually ranges between 30 to 60 degrees in dicots. The dicot's branching venation patterns are those most typically associated with plant leaves, and are more difficult to generate than the parallel venation of monocots. As such, the pattern generation portion of this report focuses on dicot veins.

A study in botany provides much in the way of expectations for a biologically-based representation of plant leaf venation systems. The goal

---

[1] Xylem is the name given to the plant vascular tissue which conducts water and dissolved nutrients upwards from the root.

[2] Phloem is the name given to the plant vascular tissue which conducts sugar and other metabolic products downwards from the leaves.

Figure 1: *Striate venation of a monocotyledon leaf (redrawn from [4, page 104])*

in realistic simulation of leaf veins is to find a solution which will model all of these observable traits. This report presents a list of criteria for a biologically-based solution, and analyze several potential solutions with respect to these requirements.

2.2 Thickness and Pigmentation

The presence of veins in a plant leaf alters its surface shape and pigment concentration of the leaf, which in turn affects its interaction with light. Contrary to many representations in computer graphics, veins do not reside on the surface of the leaf but inside the mesophyll layer, under the cuticle and epidermis layers. However, the veins do protrude from the leaf surface, adding to the thickness of the leaf. This fact has numerous ramifications in computer graphics [2]. First, it affects the required physical representation. A silhouette view of a plant leaf's edge would need to show the physical protrusions in order to achieve realism. Secondly, veins affect the masking and shadowing of light on a leaf since the physical presence of veins creates obstacles for the light interacting with the foliar tissues.

The different level of pigmentation found in veins has further consequences in the area of light interaction with the leaf. McClendon and Fukshansky [10, page 205] estimated that the light flux through veins was 10

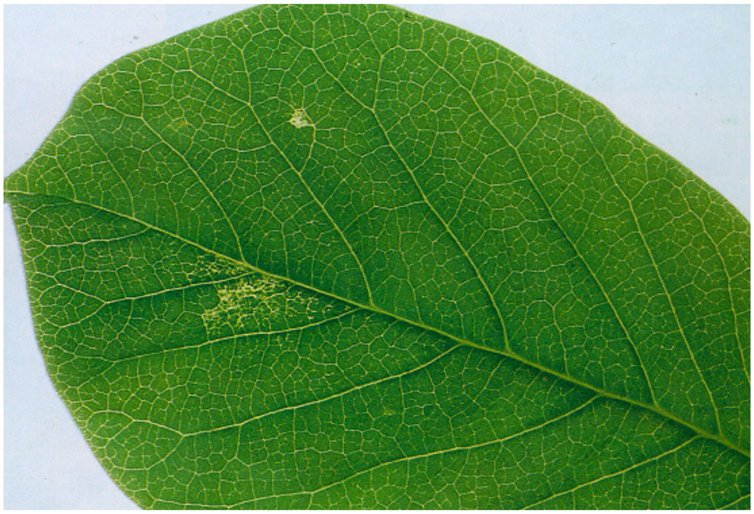

Figure 2: *Reticulate venation of a dicotyledon leaf (redrawn from [4, page 104]).*

times that of the flux through the neighbouring dense mesophyll of the leaf, because of lower concentration of pigment in veins. Furthermore, as mentioned earlier, a leaf's anisotropy depends on its venation pattern.

3 Requirements for a Biologically-Based Solution

In the previous section we summarized some results from botany which describe various factors affecting the appearance of plant leaf venation systems. From the above examination we can glean several requirements which must be fulfilled by an automatic and predictable representation of leaf veins which claims to be biologically-based. In this section we propose a set of requirements for such a solution. The next section will analyze several potential solutions with respect to the stated criteria.

1. The goal of this report is to automate the procedure for representing venation, and as such, using an artist's drawing of veins does not qualify as such a procedure. The method used must provide a way to design the dicot veins' branching pattern automatically, taking into account the above-described angles and inter-vein spacing in the mesophyll. As the venation pattern differs slightly between species, the procedure should also allow for parameters which can modify the pattern. The

solution must also give an appropriate volume to this structure, taking into account the alteration in size between primary and higher-order veins, and their curved trajectories and smooth/bulged branching. A representation using straight, constant-sized cylinders will not suffice. The solution must also provide for the variations between branching patterns in different species of dicots.

2. The solution must accomplish the prominence of veins from the leaf's surface, such that a close view of the leaf's silhouette shows their protrusion.

3. The method used must allow these veins to interact with light correctly, as described in the previous section. This includes variations in pigment and thickness, masking and shadowing effects, and anisotropy.

4. Whichever method is used to create the veins, an overall solution must keep in mind the changing level of detail necessary in different scenes. In a closely viewed leaf, the veins must be accounted for in the physical leaf model, for reasons described above. For scenes at mid-range, the protrusion of the veins is not a noticeable effect and thus an equivalent texture map, (possibly generated from the geometry itself before it is removed) may replace the geometry. As the leaves increase in distance from the viewpoint, even these textures may not be seen and it suffices to consider the veins' effects on the overall illumination of the leaf.

5. If the chosen solution places veins onto existing leaf surface, it must provide a way to allow for movement or growth of the leaf, with the vein remaining "attached" to the surface and with both moving together realistically. This is an interesting problem, and an elegant solution is presented in the final criterion of this list.

6. Solutions seen so far approach the problem of creating venation systems by "pasting" the veins onto a preexisting leaf model, so that the veins are a surface feature which depend on the leaf surface's placement and shape. This approach is contrary to nature, in which this dependency is reversed. In reality, the leaf veins are embedded inside the mesophyll layer, and leaf tissue surrounds the veins to obtain the nutrients they supply. This may seem to be a subtle difference with respect to computer graphics since the role of nutrient supplier is not important to vein representation, yet it does become important when the final leaf model is folded or otherwise manipulated. In current approaches, the veins would have to be made to follow the leaf, whereas a

more biologically-based approach would use the veins themselves as a sort of skeleton which controls leaf movement. Bloomenthal [3] points this out by stating that model design "makes more sense inside out". This "inside-out" design is more intuitive and realistic, and as such could handle the challenge of leaf movement more gracefully. Yet this final criterion applies to very sophisticated solutions where great effort is being made to ensure biological realism, and effective-looking solutions may exist which ignore this requirement. The main implications are in leaf movement and in leaf growth, which are only briefly touched upon in this technical report.

4 Computer Graphics Approaches

Several existing computer graphics techniques can be applied to this problem. In this section we examine seven approaches, beginning with the most basic representation which fail on most of the criteria outlined above, and move toward more robust solutions.

4.1 Texture Mapping and Bump Mapping

Texture mapping is the simplest technique available in dealing with the representation of leaf veins, and one of the methods most commonly used in industry. In industry practice, an artist will manually create a texture of a leaf's venation system and map this texture onto the leaf model. Clearly, texture mapping alone is not an acceptable solution to the problem. First, the creation of the texture is manual and ad-hoc, whereas our goal is to make the process predictable. Secondly, even if the texture can be automatically generated, simple pasting of a texture onto a model does not provide a precise enough representation in that it does not allow for realistic light interaction with the object during rendering. Also, texture mapping does not solve the problem set out in the second criteria above, of portraying the veins' protrusion at a close view of the blade's silhouette.

We can introduce bump mapping, or the manipulation of surface normals according to the surface features the texture is trying to convey, in order to cause light to interact with the model roughly as it would if real modulations existed on the surface. Although this is a distinct advantage and can be accomplished simply, it also fails to provide an adequate solution. As in texture mapping, the side view of the leaf blade does not show veins protruding from its surface. Also, some lighting effects such as shadowing and masking will still not be easily available, since a surface geometry is not physically present.

4.2 Displacement Mapping

Displacement mapping is little more than another extension of texture and bump mapping, but unlike the above methods it is capable of producing results which are realistic with respect to many of the above-stated criteria. Displacement mapping resembles bump mapping in that a map of alterations is created for an object model, which will be applied as the object is rendered. However, the displacement map does not only contain perturbations of surface normals, but adjustments for the height of the surface at each point in the map. The map actually constructs a scalar height field $h(x, y)$ whose values are added to points on the model's surface, $s(x, y)$, during rendering [17, page 1]. Thus, this method allows for the addition of geometry to a scene during rendering. In traditional displacement mapping, these height values can only be added to the model's surface in the direction normal to the surface, $N(x, y)$. So the surface resulting from rendering is $S'(x, y) = S(x, y) + h(x, y)N(x, y)$ [*ibid.*]. Pedersen's approach of using 3D flow fields to form the direction of displacement [11] provides a way to bypass the restriction on displacement direction if needed. However, his refinement is not needed for the purposes of veins on the leaf blade. For this problem, adjustment in the normal direction is adequate, and greatly reduces the storage needed for this algorithm.

Displacement mapping is a relatively simple way of adding modulation to an underlying smooth surface. Since it creates a real geometry, it will cause correct interaction with light, including shadowing and masking effects. The geometry produced will be visible in a silhouette view of the blade. This method also suggests a simple procedure for changing the level of detail in a scene: if only a textured representation of veins is needed the height map could be interpreted as a texture in some predefined manner. Also, although this method is among the simplest available, it does satisfy the fifth criteria. Because the offset surface is added to the underlying surface values, the veins will be correctly attached to the leaf surface, even if it is not flat. They will remain attached to the leaf when it moves or deforms. This solution may be the optimal choice for some applications since it is not expensive computationally and still satisfies many of our criteria.

However, several drawbacks exist for displacement mapping with respect to criteria mentioned. First, this method gives no direction as to how the map itself will be generated. Setting these heights based on data carefully obtained from a real leaf does not satisfy the requirement that the procedure be automated. The sections below will examine some available methods for automatically generating such maps. Secondly, like its predecessors, texture and bump mapping, displacement mapping deals with a venation system as a surface feature instead of as an essential component of the leaf's structure.

Another difficulty with this method is in determining the sampling density needed to represent the height map. Large sections of leaf with no veins should not be sampled as heavily as sections at the edges of veins. To deal with this problem, Wang proposes a method called "feature-based" displacement mapping for localizing sampling density based on discovered features in the displacement map [17, page 3]. His method uses the computation of a feature metric and an orientation for each point in the low-density sample. Discarding the sample points which are not meaningful, the procedure then adjusts the locations of remaining sampling pointed to focus on the regions where features are determined to lie.

4.3 L-Systems

The Lindermeyer systems, or L-systems, use a context-sensitive grammar to automatic generate plant geometry and to portray branching patterns in trees [12, page 2]. They satisfy the first criterion in that they can automatically design predictable branching patterns. A realistic pattern can be achieved and the grammar can be written to account for differences arising between species.

However, some geometry must be created to implement this pattern and it is not immediately clear how to do this realistically. A branching structure of line segments or straight cylinders will be too geometric and not give the impression of a natural structure. Algin developed a "generalized cylinder" which can be swept across a space curve and can vary in size of shape of its cross-section [3, chapter 5]. Because of these qualities, the generalized cylinder is ideal for representing natural forms. It may be used to bring geometry to the L-system, or else the L-system may be created using simple line or curve segments and serve as input to some other method described below.

L-systems are an important step in the direction of a solution for this problem, but several issues must be addressed if we intend them to satisfy all of our criteria. For instance, we must develop a method of attaching this L-system to the surface of the leaf blade, and ensuring that it lies correctly inside the bounds of the leaf margin. A possible approach is to use the blade model as input to the process of L-system development to provide constraints for the grammar.

4.4 Cellular Texture Basis Functions

Another procedure for generating a texture is the use of a cellular texture basis function as suggested by Burge [5, page 1]. Feature line segments can be used to link cells in a parent-child hierarchy, and various techniques may be applied to create leaf-specific features such as tapering towards the apex. Burge successfully modeled a texture for plant veins using this approach, and the results are shown in the following figure.

Figure 3: *A procedurally generated vein texture (redrawn from [5, page 1]).*

Another benefit in Burge's approach is his suggestion for extending this texture to a 3D model. He discusses placing feature geometry in each of the cells, which would allow a geometry to be rendered for the generated pattern [*ibid.*]. Exactly how this would be done or how precise the result is unclear, but this is a promising direction for future research. Such an approach would satisfy the criterion for shifting level of detail, since a textured leaf could be reinterpreted to obtain the geometry representation when the viewer moves closer to the leaf.

4.5 Implicit Surfaces

A promising method for vein representation is found in the idea of implicit surfaces. As proposed by Bloomenthal [3, chapter 3], the design of a model using implicit surfaces is achieved from the inside out, basing an outlying surface on an internal skeletal structure. Bloomenthal argues that this method is particularly suitable for the representation of natural forms, in which structure and movement depend on some form of skeleton.

Implicit surfaces are defined by functions which specify a volume surrounding a skeleton. Points in space which are roots to these functions are considered to lie on the object's surface. Thus, implicit surfaces are useful in skeletal design since the distance from skeleton to the outermost tissue is volumetric [*ibid.*].

The approach can be useful in two ways. First, it can automatically construct a geometry for veins based on a preexisting pattern made up of simple line or curve segments. This pattern could be generated by use of L-systems or some other method. Even if an L-system uses line segments to create the skeleton, Bloomenthal shows that the implicit surface based on such a skeleton will still look natural. His demonstration is reproduced in the figure below.

Figure 4: *Implicit surface for a skeleton of line segments (redrawn from [3, chapter 5]).*

Concavity in the skeleton will result in tangent discontinuity in the surface, which appears in the form of an unnatural crease [*ibid.*]. This can be remedied by blending primitive volumes instead of taking the union of the surfaces. The blends produce bulging instead of creasing where two pieces intersect [3, page 25], which produces a more natural effect for plant veins.

The second potential use for implicit surfaces is as a way of basing the leaf blade structure and movement on its venation system. To achieve this, the function which defines the skeleton would be the automatically-generated branching pattern, and the surrounding implicit surface would be the leaf blade which envelops the veins, with raised bumps directly over vein lines. This effect may be achieved by the blending of two implicit surfaces based on the same skeleton, one for the veins themselves and one for the blade.

Bloomenthal used this approach, attempting to model the leaf itself as an implicit surface on the vein skeleton, even recreating the leaf margin based on vein ending points. His method had considerable success, although the margins were not perfect, and his approach had the advantage of being automatically generated by the vein branching pattern. He dealt only with primary veins and created a texture for the higher order veins. This texture was created by "exploding" the blade's triangle mesh into triangles, whose edges he warped slightly to give a more natural impression [3, chapter 7]. The spaces in between these exploded triangles became the texture for small veins on the leaf. There is much ingenuity in his method; it is efficient and does represent a branching structure bounded within the leaf. Yet his texture-generation method is not useful to us as it does not allow for sufficient control over the branching pattern achieved, and certainly does not allow us to differentiate the pattern between species of plants.

In general, however, Bloomenthal's approach has great promise. The use of implicit surfaces for the leaf blade gracefully satisfies our last criterion for the blade's dependence on the venation system. Movement of the entire leaf is made simple by automatically assigning weights for a certain vein's influence on the neighbouring surface. His method also allows for altering levels of detail. In combination with the use of generalized cylinders and blending, we can achieve realistic-looking geometries for the veins themselves.

4.6 Particle Systems

Structured particle systems have been used to devise models for natural forms such as trees or clouds. Instead of drawing a family of particles constantly in motion, the entire trajectory of each particle over its lifetime is drawn [14, page 3]. Thus structured particles can be used to create a static image.

The main benefit of structured particle systems to our problem is that this single method can be responsible for both pattern and geometry generation. The system can be structured so as to force the particles to follow a constrained branching pattern, and to remain on the level of the leaf surface at every point in its path. The particles can be given a spherical shape so as to have a semi-sphere protruding from the leaf at each point on a vein. Their diameter can be made to decrease with time, or with each bend in its path. The particles can die when their diameter falls below a certain threshold, or be explicitly killed off when they come within a few millimeters of the leaf margin.

Witkin and Herbert [7] propose a general method of combining particle systems with implicit surfaces. In their approach, particles and a surface are constrained together, while both the particles and the surface may move. The particle-surface constraints are based on classical mechanics and their method involves solving for derivatives of either the particles or the surface, given the velocity of the other's movement. This interesting method fulfills the last two criteria in constraining veins to the blade, even when the leaf moves or changes shape. With this method, leaf motion can also be initiated by the particles representing the venation system – the veins may be used to move the blade. However, this system is complex and designed to handle models for which real-time evaluation of changes must be done. If movement of the leaf is not a major priority, this solution may be more computationally expensive than necessary and the result may look less precise than desired because of the use of particles.

4.7 Cellular Texture Generation

Another method allowing predictable generation of textures along with physical surface features was suggested by Fleischer [6]. Cellular texture generation was developed as a way of producing textures to model surface details. It combines a biologically-motivated cellular development simulator with a constraint to keep the surface features located on a preexisting surface. Fleischer intended this method to be used in cases where texture maps are insufficient, but where by-hand model manipulation is inconvenient. His method is based on "cells", which resemble particles in that they have surface characteristics such as local curvature and colour, but whose behaviour depends heavily upon its neighbouring elements and the "extracellular environment". These cells interact as they mature in their environment, making used of reaction-diffusion properties and adhesion of cells to each other [6, page 240]. All these factors and cell constraints combine to form a first order differential equation term which will modify the cell state. An ordinary differential equation is created from the combination of cell terms and environment variables.

Benefits of this approach are that it is biologically-motivated and can take into account environmental factors or properties of the underlying surface. It also guarantees that the cellular texture will remain attached to the underlying surface. The cells have constraints which can be used to make them repel each other at certain points, or stick together at certain points, potentially based on which region of the surface they lie on. Fleischer's method produced convincing representations of furry of thorny surfaces. It may not be straightforward to adapt the cellular development simulator to accommodate patterns as structured as venation branching, but the idea is worth examining. The major drawback of this method is that it treats veins as a surface feature instead of as a basis of leaf structure.

5 Conclusions

This technical report has described several approaches to the problem of a plant venation system representation, and it has proposed a specified set of requirements to ensure that the solution is biologically-based and realistic. Although some of these approaches have never been used as a means of creating venation systems, they present different aspects that could be explored to fulfill the requirements listed. Investigating these techniques within the context of leaf venation systems will create an opportunity to assess their drawbacks and advantages more closely, and will contribute to further the state of the art in the realistic and predictable simulation of leaf venation systems.

References

[1] Baranoski G, Rokne J. Efficiently Simulating Scattering of Light by Leaves. *The Visual Computer.* 17:8, 2001:491-505.

[2] Baranoski G, Rokne J. Light Interaction with Plants. 2002 *ACM Siggraph 2002 Course Notes*, (26).

[3] Bloomenthal J. (1995) Skeletal Design of Natural Forms. Doctoral dissertation, University of Calgary, Dept. Computer Science.

[4] Bowes B. (1996) *A Colour Atlas of Plant Structure.* Iowa State University Press, Ames.

[5] Burge T. A Branching Cellular Texture Basis Function. *SIGGRAPH 2000 Sketch*, ACM.

[6] Fleischer K, Laidlaw D, Bena L, Currin, Barr A. (1995) Cellular Texture Generation. *Computer Graphics Proceedings*, 1995: 239- 248.

[7] Witkin A, Heckbert P. Using Particles to Sample and Control Implicit Surfaces. Computer Graphics, Proc. SIGGRAPH 1994.

[8] Hetherington A. (1997) *Leaf Development and Function.* Cambridge University Press, Cambridge.

[9] Kluckling E. (1986) Leaf Venation Patterns. *Gebruder Borntraeger*, Berlin, Volume 1.

[10] McClendon J, Fukshansky L (1989) On the Interpretation of Absorption Spectra of Leaves. *Photochemistry and Photobiology*, 51,2:211-216.

[11] Pedersen H. Displacement Mapping Using Flow Fields. *SIGGRAPH*, 1994: 279-286.

[12] Prusinkiewicz P, Hammel M, Mjolsness E. (1996) *L-systems: From the Theory to Visual Models of Plants.* CSIRO Publishing.

[13] Raven P, Evert R, Eichhorn S. (1999) *Biology of Plants.* W.H. Freeman, New York, Sixth Edition.

[14] Reeves W. (1983) Particles: A Technique for Modeling a Class of Fuzzy Objects. *Computer Graphics*, 13:3.

[15] Semple J. (1998) *An Introduction to Fungi, Algae and Plants.* University of Waterloo, Waterloo.

[16] Smits B, Shirley P, Stark M .(2000) Direct Ray tracing of Displacement Mapped Triangles. Tech. Rep. UUCS–00–008, Computer Science Department, University of Utah.

[17] Wang X, Maillot J, Fiume E. (2000) Feature-Based Displacement Mapping. *Eurographics Workshop on Rendering,* June 2000.

[18] Woolley J. (1971) Reflectance and Transmittance of Light by Leaves. *Plant Physiology*, 47: 656-662.

[19] Wyvill B. (2000) Better Blending of Implicit Objects at Different Scales. *SIGGRAPH 2000 Sketch*, ACM.