In order to determine the weight of leaves on an individual tree we set out to develop a model to approximate the number of leaves on any given tree. However, given the large amount of variance between branch structures and the large amount of consistency within a single branch system, we decided to develop a program to model the different major branch systems (the sub-trees with a single large branch as its "trunk" or base) within a tree and the number of leaves that each branch structure develops to approximate the total number of leaves on a tree.

Branch systems have several striking properties that can be quantified and used to describe their physiology. We quantify some of the parameters more pertinent to branch and leaf growth and use them in our model. To best incorporate the inherently fractal nature of tree growth, we decided to use an algorithm that recursively produced auxiliary branch structures from the main branch. We also incorporated a stochastic factor into the algorithms to reflect the inherent randomness of tree growth. To counteract occasional extreme variances that arose from the randomness, we took a Monte Carlo approach: the program creates a large number of sample branches and leaf counts which are then averaged to produce our final estimation.

Our algorithm produced stunningly accurate leaf counts given the innate randomness of the problem. We did small scale testing on several tree branch systems from varying species of trees that we found nearby in different environments and were able to produce not only accurate leaf counts, but also comparable branch structures. We also did one large scale test on an entire oak tree by modeling each of the major branches and summing the results to produce the total leaf count. The results of our simulation produced an average that matches general estimates for leaf counts for mature oaks. Not only did our algorithm produce accurate leaf counts, it also reproduced the shapes and sizes of the branches that we try to model.

The algorithm does have some areas where we would like to improve it. One inherent weakness of our model is that we require the user to choose the parameters themselves, which can be difficult to do on site. However, our model depends a lot more on branch order and lengths, which are more easily estimated. As a result, it is still robust with regards to choices for those parameters which are hard to estimate. We also realize that our model does not take into account several properties related to leaf growth. Still, we feel that our model is a useful predictive tool for scientists who need to estimate the biomass of trees. The accuracy we obtained in small-scale and large-scale testing is rather resounding. Ultimately, we feel that our model provides a very strong starting point and that with some minor work, our model can have significant predictive power.

Branching Out:
Modeling Leaf Weight by Tree Growth Simulation

Control # 16647

February 14, 2012

Abstract

In this paper, we present a program which generates a model of a real branch based off its quantifiable parameters. We first examine how branches and leaves grow. From there we determine the parameters that affect leaf count the most, taking into account random distribution and normal variations of branching and leaf growth. This method reduces the problem from counting the leaves of a tree to counting the branches of a tree. Our model uses the aforementioned parameters to estimate the number of leaves on a branch. From the total number of leaves on the tree we can calculate the total mass of the leaves by weighing an average leaf. This method allows ecologists to quickly and accurately determine the biomass of a tree by simply taking measurements of its main branches. We test our method against actual tree branches by counting the leaves manually, inserting the branch's parameters into our program, and then comparing the results. The model successfully produced leaf counts similar to the actual amount that we manually counted.

Contents

1 Introduction

In this paper we provide a model for generating estimates for the total weight of leaves on a tree. We do so by modeling the approximate growth of the tree in question, and counting the number of leaves. We take some time to understand the dynamics of tree growth and other important properties of leaves, and after doing so, we construct a model which simulates how the tree could've grown. We then apply this model to various real-life trees and estimate how many leaves our tree has and how much they weigh, and demonstrate that our model is surprisingly accurate. We feel that our model will provide botanists a useful tool for calculating the biomass of trees.

1.1 Outline of our Approach

The first few sections of our paper will be devoted to developing the theory and framework for our computer model, and a brief outline of our computer model. Then we will model the behavior of leaves on pictures of various trees and branches. For each case study we do the following:

1. **Find approximately normal "branch system"** for each major branch of the tree. We want to identify the major branch structures on the tree so that the growth on each is regular, i.e. there is not too much variance in the growth pattern on the branch structure. Note that we may not want to choose the branches directly incident to the trunk if their growth is unpredictable. Rather, we may want to select multiple sub-branch systems for each branch.

2. **Estimate the size and growth pattern of the branch system.** In our program we need estimates on the length of the total growth, the size of one segment of the branch, branch factor, how fast the main branch grows and how fast auxiliary branches grow. We provide precise definitions later on when we describe our model in more detail.

3. **Generate random twigs and leaves** in these growth clusters based on estimates above and knowledge of the fractal nature of tree growth. We repeat this process numerous times and obtain an average leaf count and average branch count.

At this point, our case studies terminate, as we will compare our leaf count to the leaf count of the actual branch. To find an estimate for the weight of the leaves, one can at this point simply multiply by the average weight of a single leaf.

1.2 Assumptions

Due to the large variation between individual trees, we assume that we have some visual model of the tree, as otherwise it seems to us a fool's errand to try

to count the leaves on the tree. For our case studies, we decided to ignore pine trees, as the growth habits of their branches and leaves are markedly different from deciduous trees. We also assume the following about the physiology of the tree:

- **The tree is in "full leaf",** that is, it already has most of its leaves and has not begun to extensively lose leaves. This model was not designed for estimating how many leaves a deciduous tree has in the late fall or the winter.

- **The tree is generally healthy.** Our model actively uses how trees grow normally to predict how many leaves there will be. Hence our model is less accurate for a tree if it's very sickly or if it's been heavily pruned.

- **The growth properties of the branch systems are predictable.** In our model we assume that the growth of the tree does not change drastically (which could happen if for instance an outside agent applies hormones which inhibit tree growth, or if the environment rapidly changes).

Our model, without too much difficulty, can be extended for some of the exceptional cases (we discuss this later). In the wild, most trees should fit these assumptions, so we decided to focus on trees whose growth had not been hampered by human interference or disease.

2 Understanding Tree Growth and Leaf Properties

In this section we provide some basic background on how trees grow; these insights will provide the motivation for our model that generates tree growth. We define a **branch system** or **branch structure** to be a sub-tree of the tree with a single branch being the substitute for the trunk. We will use this term to represent a single branch on which we are running our model. We also define a **segment** of a branch to be the part of the branch between two nodes at which the branch system splits. In §2.1 we lay out the basics of tree growth; these form the basis for our tree simulation. In §2.2 we discuss the relationships between leaf shape, branching structure, and nutrient management, and answer questions such as why are leaves shaped the way they are.

2.1 Fundamentals of Tree Growth

2.1.1 Apical Dominance and Growth Decay

We found that there were two key parameters that largely determine the structure and size of the resulting branch. The *apical dominance* is defined to be the ratio between the the size of a segment on the main branch and the size of a segment of a lateral branch that splits off at the node the segment ends. In the notation of Fig 2.1 the apical dominance (of that tiny branch) would be A/C ([1], [3], [4]). We also define the *growth decay* to be the ratio between successive

segments on the same branch; in the notation of Fig 2.1 the growth decay would be B/A ([1]). On a larger branch system with more branches and sub-branches this will generally not be entirely constant, however in our research we found that these terms are generally considered constants ([1], [3], [4]). Intuitively, the apical dominance represents how skinny the tree is; the higher the apical dominance, the smaller the side branches tend to be, and the growth decay represents how short the tree is; the lower the growth decay, the shorter the tree will be.
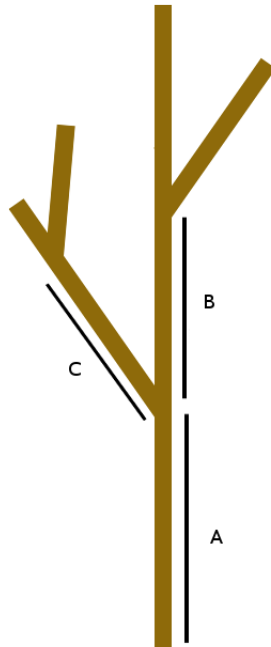

Figure 2.1: Defining Lengths for Apical Dominance and Growth Decay.

2.1.2 Growth Mechanics

The living tissue responsible for the outward growth of a branch is concentrated near the end of the branch ([3]). A group of cells located at the apex of the branch is responsible for the continued growth of the branch, and each year they bud to continue growth along the branch. Lateral growth occurs later at this buds (sometimes up to a year later). The apex of the branch releases a class of hormones called auxins down the branch. These hormones promote the production of buds, which ultimately cause lateral growth ([6]). The concentration of these hormones is much higher near the apex, which causes lateral growth to most often occur near the end of the branch ([7]). As these lateral buds grow,

they become new branches with their own apexes. Notably, as hormones travel down the branches of a tree and not up ([3], [6]), these buds lose all means of communication with the old apex.

Leaf production occurs as the apex produces organs which ultimately make a leaf or flower ([4]). At some point the apex of the branch stops devoting energy to producing lateral branches and uses it all for the production of such organs, and at that point the growth of the branch terminates ([4]). Therefore if we consider these organs to be another type of bud, we notice that the apex will branch a certain number of times before the sub-branches are actually leaves. Empirically, it appears to us that this number is almost constant throughout a single branch system.

This growth behavior is self-similar, and naturally induces a "tree fractal", as noted in [1] and by Mandelbrot in [5], although in the latter reference Mandelbrot approaches the problem from a distinctly more mathematical perspective. This is a key observation that serves as the basis for our program.

2.2 Leaf Layering and Sizes

We begin by examining how leaves are layered within a tree. In this section, we assume leaves can be described as belonging purely in flat horizontal layers. Thus, the area of a leaf is given by a circle whose radius is $r = \frac{(r_1 + r_2)}{2}$, where r_1 is the radius of the largest circle that can be inscribed in the leaf and r_2 is the radius of the smallest circle that circumscribes the leaf.

We define an ideal *monolayer tree* as a tree whose leaves are arranged in a single horizontal layer. On the other hand, an ideal *multilayer tree* is a tree whose leaves are arranged in many distinct vertical layers far enough apart such that no portion of any leaf in a lower layer is obscured by the shadow of a leaf in an upper layer ([2]).

2.2.1 Ideal Monolayer Trees

In an ideal monolayer tree, if the uniform leaf density per unit of ground area ρ becomes too large, the leaves tend to overlap. Any leaf shaded by another is a net loss for the tree, and thus new leaves tend to fill in the gaps in the layer. The surface area of the leaves above a unit ground area is given by $\rho \pi r^2$, which does not account for any overlap between leaves. To account for the overlap, we use the Poisson probability distribution to express the amount of sunlight that passes through the layer over an unit of ground as $e^{-\rho \pi r^2}$ ([2]). Thus, the remaining portion of the unit area of ground must be due to the projection of leaves onto the ground. We can then express the surface area of the leaves projected onto a unit area of ground as

$$1 - e^{\rho \pi r^2}.$$

Using the Taylor expansion for e^x,

$$1 - e^{-\rho\pi r^2} = \rho\pi r^2 - \left[\frac{(\rho\pi r^2)^2}{2!} - \frac{(\rho\pi r^2)^3}{3!} + \cdots\right] < \rho\pi r^2.$$

In other words, the projection of the leaves onto the ground is always less than the total surface area of the leaves. Since there are no leaves in any lower layers, it follows that ideal monolayer trees would want to maximize this projection to the ground level such that its leaves capture all incident sunlight. To accomplish this, it follows that ideal monolayer trees would have leaves with larger radii; see, for example, species of palm.

2.2.2 Ideal Multilayer Trees

In a multilayer tree it's important that light penetrates upper layers to reach the lower layers. We assume the leaves of each layer are independently spaced, and thus the area of the leaves on each of the n layers is given by $\frac{\rho}{n}\pi r^2$. The proportion of average incident solar light intensity L_0 that penetrates layer n, assuming leaves have opacity σ, is given by $L_0\left(1 - \sigma\frac{\rho}{n}\pi r^2\right)^n$. To represent an ideal multilayer tree, we let each leaf represent a single layer which would imply that n approaches infinity. If we assume an ideal multilayer tree, the fraction of light on the lowest layer is:

$$\lim_{n\to\infty} L_0\left(1 - \frac{\sigma\rho\pi r^2}{n}\right)^n = L_0 e^{-\sigma\rho\pi r^2}$$

Photosynthesis is at maximum capacity when incident light is at least 20% of initial incident light intensity[2]. So for an ideal multilayer tree,

$$L_0 e^{-\sigma\rho\pi r^2} \geq 0.2$$

which implies

$$r^2 \geq \frac{\ln(5)}{\sigma\rho\pi L_0}$$

for the lowest layer. In higher layers, the incident light is more intense, which allows for smaller leaf radii. Since trees tend to produce leaves efficiently, the top layers will have smaller leaves than the bottom layers. This behavior is seen, for example, in the leaves of the Black Oak ([2]).

2.2.3 Non-ideal Trees

Since leaves in a real tree don't grow in perfect horizontal planes, the concept of 'layers' in non-ideal trees becomes vague. Non-ideal trees also don't tend toward the extremes of mono- and multilayer trees. The model still holds in principle

if we define a 'layer' to be the approximate horizontal plane of a branch. Trees with fewer layers tend to the monolayer analysis, whereas trees with more layers tend toward the multilayer analysis.

In non-ideal trees, leaves are allowed to droop instead of remaining horizontal. This has the effect of decreasing the incident light on the upper layers as the sun travels across the sky; the leaves on those layers have a lower heat load and more light passes through these upper layers to the lower layers. Since the lower layers receive less incident light, they tend to become more horizontal ([2]).

2.3 Leaf Shapes and Tree Habits

Now that we've established some correlation between leaf size and leaf distribution in ideal trees, we turn to examining the correlation between leaf shape and the habits of the tree.

2.3.1 Ideal Monolayer Trees

As stated above in §2.2.2, for maximum photosynthesis to occur, leaves must receive at least 20% of average solar light intensity. Since leaves in a monolayer tree don't obscure sunlight from each other, these leaves can exist in the undergrowth of forests. Covered in shade, their leaves can afford to be larger since they receive less heat from the sun and so they don't have to worry about drying out. Therefore, there is a tendency for monolayer trees to have low apical dominance and grow laterally instead of vertically ([2]).

2.3.2 Multilayer Trees

Multi-layer trees have a tendency to have lobed and complex leaf structures. As a result, the circles inscribed in each leaf have smaller radii. In the upper layers of a tree, the lobed shape helps dispose of the excess heat load. Also, the shape decreases overlap and allows for light to pass through more easily to lower layers ([2]).

3 The Branch System Model

3.1 Motivation for the Algorithm

We want to capture the self-similar growth pattern of trees as described in §2.1. Suppose the branch we are at just budded as we described in §2.1.1, and we have chosen a bud to follow. As described in [6], the bud grows for some amount of time until it too buds. The apex then continues growing, and this growth stimulates the growth of auxiliary branches at the budding point. If we assume that the auxiliary and main branches grow at approximately the same rate, this explains why the auxiliary branches tend to be smaller: they simply start growing later. We could not find sources for exactly how the apex releases

the hormone which stimulates growth; however empirically we determined that the most accurate model for apical dominance is in fact a constant, and the literature supports this ([1], [4]).

We also make the assumption that the auxiliary branches and the main branch begin budding at about the same time, and in particular we assume that the auxiliary branch will begin growing before the main branch buds again. Behavior of this sort is implied in some of the literature ([1], [6]) but ultimately it is probably a simplifying assumption.

In this framework however it is easy to model the growth of the tree recursively: because we assume that the axuiliary branches will start growing before the main branch buds again, for the purposes of modeling growth this is roughly equivalent to assuming that they start growing simultaneously but the auxiliary branch simply grows at a slower rate (i.e. one over the apical dominance). At the same time, since consecutive segments of the main branch tend to be smaller and smaller, we assume that later segments grow proportionally slower ([1]). This is proportion is exactly the growth decay. Hence when we reach a budding point we simply recurse to produce the auxiliary branch systems as well as continue growing the main branch. The end of the recursion is also intuitive: it should end when the main branch has hit the current branch size or when the branch has budded a number of times equal to the average branching order of the branch. With these considerations in mind we devised the recursive method for generating one branch.

However, because of the amount of variation that could occur in the generation of one branch (as each choice gets compounded with every branch), a naive application of this approach could produce leaf counts that are non-representative of the actual leaf counts. But as we assume that the branch system we are modeling is roughly typical (by splitting into multiple smaller branch systems if necessary), if we instead do many of these simulations and take the average leaf count over all these simulations, we should obtain the leaf count for a typical branch system, which therefore should be a good estimate for the leaf count of the actual branch system.

3.2 Required Parameters

To generate a single branch, our program requires the following parameters:

- The **average branch order** B of the branch. We define the branch order of a twig to be the number of segments between it and the trunk. To get the average branch order of the branch we simply average the branch orders of all the twigs.

- The **minimum twig length** T of any twig on the branch.

- The **average apical dominance** A. For any group of segments resembling those in Fig 2.1, we can get its apical dominance, and to get the apical dominance of the whole branch, we again take the average of the

apical dominance over all groups of segments like those in Fig 2.1 in the branch.

- The **average growth decay** G. To obtain the average, we do the same as we did for the apical dominance.

- The **trunk length** of the branch L_T. We define this quantity to be the size of the last segment on the main branch before major splitting occurs. For instance, if Fig 2.1 was the entire branch, then the trunk length of the branch would be A.

- The **total branch length** L_B.

They would have to be measured by hand. We realize that some of these properties are impossible to calculate exactly in the field; however, because our program is relatively robust, most (if not all) assignments of the parameters which are close to the actual assignments will produce accurate results. In our testing, at least, this has never really been a problem.

We also note that for our algorithm to run correctly the units of trunk length, total branch length, and minimum twig length don't matter as long as the user maintains the correct (or at least close to correct) ratios of these quantities to one another. We note that if for instance we multiply all three quantities by any positive constant the leaf counts that our program produces are exactly the same.

3.3 The Algorithm

To model an entire tree, we first split the tree into branch systems which have internally consistent growth parameters. Then we apply this model to each branch system. After estimating values for the parameters above, we input them into the algorithm and it does the following:

1. The algorithm checks if the branch we're on has branched about B times. Using a normal distribution centered at B with standard deviation 1, we randomly chose whether or not to make a leaf. If the algorithm chooses to continue to branch, it generates a segment with length approximately the trunk length. This simulates the growth that occurs at the apex of the branch.

2. The algorithm then simulates the budding at the new apex by choosing some random number of lateral buds to produce. We decided to choose how many buds to create from a normal distribution with mean 3 and standard deviation 2. We determined that empirically these give the desired values, but of course these are subject to change.

3. The algorithm then recursively builds sub-branches. For each auxiliary branch, if l was the size of the original segment, it creates a segment of size l/A, and for the main branch, if the length of the main branch is less than L, it creates a segment of size Gl. This is fine, as we argue in §2.1.1.

We then repeat this process multiple times to produce an average leaf count for the branch system in question. Then we simply sum the leaf counts for all of the branch systems. To find a leaf weight, we can then multiply by an average leaf weight; this can be determined just by sampling throughout the tree.

4 Case Studies

Here we briefly describe how we did our tests and our results. In general the results we obtained were very accurate, even on the large scale testing.

4.1 Small Scale Testing

We first tested our model on several branches that we found nearby. We considered each branch a single branch structure and ran our model on them after roughly estimating the required parameters, to reflect the nature of how this kind of work would be done in the field. The results and the parameters we used are given in Table 4.1.

Branch 1 is pictured in Fig 4.1. While the difference in leaf count may seem appalling, we realized that the tree had significant pruning. After estimating the leaf loss due to pruning, it seems as though we accurately modeled the branchs natural growth and state. Furthermore, our model has a similar structure to the actual branch, if we account for the pruned section.

	Branch 1	Branch 2	Branch 3	Branch 4
Avg. Branching Order	6	8	6	7
Min. Twig Length	20	15	2	5
Apical Dominance	1.35	1.2	2.6	1.25
Growth Decay	0.85	0.8	0.35	0.8
Branch Len. / Trunk Len.	4	3	2.16	2.8
Estimated Leaf Count	**131**	**105**	**120**	**126**
Actual Leaf Count	**90**	**98**	**115**	**125**

Table 4.1: The parameters we used and the results we obtained from small-scale testing.

We also tested our model against several branches that grew outdoors which were naturally less pruned. The only branch where we saw any appreciable difference between our estimated leaf counts and the actual leaf count was Branch 2 which is pictured alongside a tree we generated in Fig 4.2, although even here the difference is only a little more than 7 %. This difference was actually surprisingly small because we noticed that the main branch seemed to have been cut off at some point (so there should've been more leaves and so we would

Figure 4.1: A picture of Branch 1 (rotated 90 degrees) alongside a sample tree generated by our model. In all our sample trees, the green lines denote leaves.

expect our model to predict more leaves), but it appears that the total branch length parameter allows our model to compensate as we wanted.

For the last two branches, we note that the leaf counts are approximate because some of the leaves had already fallen off due to the season. For these we did do a double-blind approach to eliminate any possibility of bias: one member of our team collected the data and obtained an approximate leaf count, and another input the parameters into the model only then compared results. We tried to find branches with extreme parameters to test our model's robustness; in particular we found Branch 3, which is pictured in Fig 4.3, and compared it to the results we got for the "regular" branch Branch 4, pictured in Fig 4.4. It is somewhat difficult to see from the picture, but the apical dominance of Branch 3 is extremely small but at the same time the growth decay is also low, as the "trunk" of the branch system is very long and only starts branching late. Our model still produces very accurate leaf counts; however, the picture we generate for Branch 3 (also in Fig 4.3) is somewhat strange, due to the high apical dominance and low growth decay. For Branch 4 however both the numbers and the sample trees are almost exactly-spot on. It is also possible that the bizarre picture for Branch 3 occurs because all our leaves are near the ends of the branches; we discuss this later in §5.
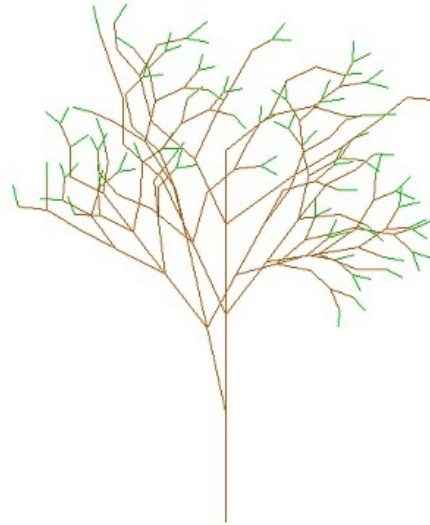
Figure 4.2: A picture of Branch 2 (rotated 90 degrees) after we stripped off the leaves alongside a sample tree generated by our model.

Figure 4.3: A picture of Branch 3 (rotated 90 degrees) alongside a sample tree generated by our model.
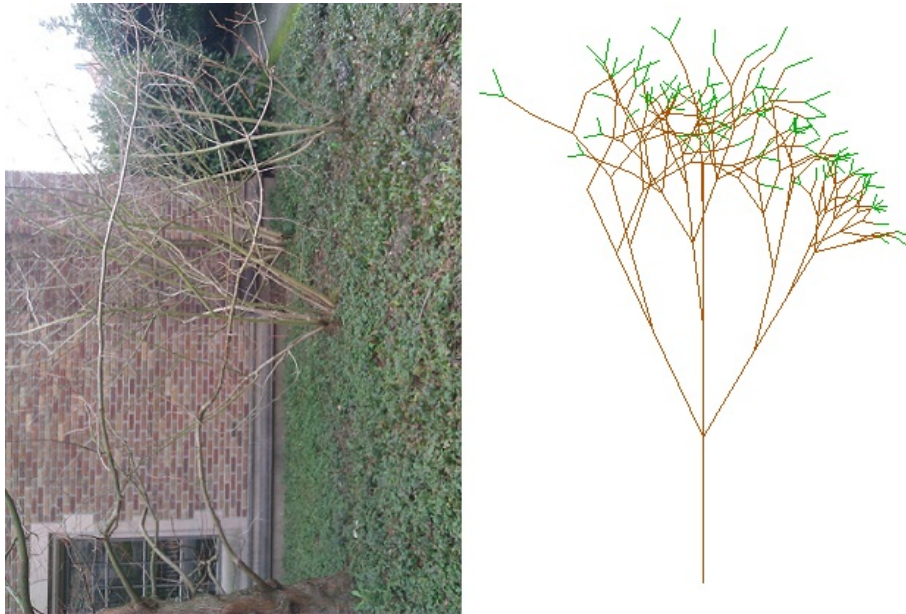
Figure 4.4: A picture of Branch 4 (rotated 90 degrees) alongside a sample tree generated by our model. The branch we modeled is the one to the left.

4.2   Large Scale Testing

Our model performed well in small-scale testing so we decided to perform a far more rigorous test: counting the leaves of a mature oak tree. Since we didn't have any oak trees nearby, we found and worked with the oak pictured in Fig 4.5. We divided the oak into 13 major branch systems as shown in the picture and began our testing on one branch.

We estimated the growth decay to be 90 %, meaning the branches decayed in length very slowly; and the apical dominance to be 135 % which meant that the main branches were longer than the auxiliary ones, but not by too much. Being an oak, the branching order was huge, and we estimated an average of 14. We note that even though this was definitely a rough estimate, even branch orders of 13 and 15 would also produce very similar numbers. After collecting dozens of samples, the average leaf count was 17517 leaves.

After estimating the desired parameters for the other branch systems, we noticed that they were very similar to each other. Since we didnt have an exact number to compare to, we decided to assume that the branch systems had similar amounts of leaves. Thus, the total leaf count our model produced for the oak tree was 227,721 leaves. The most reliable sources we found noted 200,000 to half a million leaves for a mature oak, but again these numbers are still speculative and at most a highly educated estimate.

Figure 4.5: The oak tree we modeled with the branch systems we used.

4.3 Assessment of Results

The results of the small-scale testing were incredibly accurate. Although our model failed for the first branch, it failed in a manner that we predicted, and for the other branches, the estimates differed from the real counts by almost nothing.

For the large scale testing, since there are no reliable sources for the number of leaves on an oak tree, the fact that our results were very similar to the best guesses people have must be taken with a grain of salt. At the same time, this accuracy is very promising, and likely demonstrates that our model produces estimates which are reasonable even at very large scales. This reflects both the accuracy and robustness of our algorithm. Had it been possible we would have liked to have done more large scale testing, but as it is no reliable data exists.

5   Improving the Model

5.1   Tree Properties

5.1.1   Additional Growth Properties

Our entire idea was to accurately model the growth of tree. Therefore if we could take into account more properties of tree growth, the accuracy of our model would improve. One major property we seem to have disregard is the leaf density of the tree. For instance if the twigs create dense clusters of leaves, we would count that as one leaf. To account for this error and produce a correct leaf count, we would need to alter our program. We also assume that leaves tend to grow near the end of the branches, which limits our model to mostly deciduous trees.

5.1.2   Leaf Sizes

For our program, to calculate the total leaf weight we simply assumed we could calculate an average leaf weight; however just by looking at a tree we notice that there can be significant variation in leaf size and therefore leaf weight. It could be argued that these varying sizes follow a predictable structure: for instance, smaller leaves probably occur near new growth.

In general, we found this to be too difficult to characterize. Ultimately we decided to ignore these factors. Instead, our model focused on getting an accurate leaf count. A more refined model would certainly consider leaf size and weight in its calculations of the total weight of leaves on a tree.

5.1.3   Distinction Between Twigs and Branches

Initially we had planned to differentiate between twigs and branches. We tried to categorize twigs as the actively growing parts of a tree and branches as the remaining woody matter. This is an interesting distinction because leaf and vertical growth occur almost exclusively on twigs. In the end we abandoned this distinction because unable to define a consistent criteria for what constitutes a twig,. However, this idea is definitely worth exploring.

5.2   Environmental Factors

Our model assumes near-constant growth rates, but in the real world this is not the case. Environmental factors can strongly influence the growth rate of trees. If it were possible to obtain data related to these factors, our model could adjust branch growth to mimic these fluctuating growth rates and be more accurate. In larger trees, these variations can result in larger changes in leaf count and structure over the years. Thus, the additional parameters would certainly be very helpful when dealing with larger trees. However, these additions in our model might be less useful for smaller trees since they are less affected by these environmental factors.

6   Conclusion

The problem of weighing the leaves on a tree proved to be a very difficult problem. Even our deconstruction of the problem into the problem of counting all the leaves on a tree proved to be non-trivial, as one would expect. However, by taking into account the nature of tree growth, we managed to create a model which produced results that we had previously thought to be impossible, given the incredible variance innate to the problem. While our algorithm requires quite a few assumptions on the state of the tree, we found that during our small scale testing the results were very impressive, especially when in the healthier branches. Even in the large scale testing, where the data was unreliable, our results made sense and demonstrated a high degree of robustness and stability. We feel that with minor alterations and some added functionality, our algorithm would provide an useful tool to scientists out in the field.

7 Letter to the Editor

February 12, 2012

To whom it may concern,

After several tests and recalibrations, we have developed a program that models the growth of tree branches and estimates the total number of leaves on a single branch. By running the model on the larger branch systems of a tree and summing the results, our model can actually produce the total leaf count for an entire tree. We have confirmed its accuracy with several small and large scale tests. Our model would be useful for botanists or ecologists who wish to survey the biomass of leaves on a tree to monitor tree health as well as other ecological phenomena.

Our model uses several important and unique physiological properties of branches to simulate and reproduce its growth. These variables include *growth decay*, *apical dominance*, *minimum twig length*, *total branch length*, and *initial segment length*. The apical dominance and growth decay represent the ratio of lengths between adjacent segments of the branch, while the lengths were used to limit branch growth and ensure leafing occurred at the right time. Our model is not based on how large the branch is, but instead compares the length of the initial segment to the total length and minimum twig length to monitor and limit growth. With accurate enough parameters, our model can compute the number of leaves on the branch with a very high degree of accuracy.

In addition to modeling the growth of branches, we examined the properties of leaves and their distribution within the volume of a tree. We found that in *mono–* and *multilayer trees*, the shape and size of leaves indeed correlate with their distribution. Monolayer trees tend to have larger leaves and be more shade tolerant. Multilayer trees, on the other hand, have a wider variety leaf sizes: smaller ones at the top of their distribution and larger ones at the bottom. They also tend to dominate the canopy, unlike their shade tolerant counterparts. Due to the different vertical niches they strive in, these distributions maximize the photosynthetic potential of their leaves.

Our scientific model can play a significant role in the future of botany related studies by aiding surveyors with its reliable and convenient estimates. In the hands of expert dendrologists who can properly quantify the parameters necessary for our model, the results can only be more accurate and useful.

Best regards,
16647

References

[1] Department of Environmental Studies at the University of Virginia. *Tree Architecture and a Simple Tree Model.* Last Revised 2/1/2008, Accessed 2/9/2012. `http://www.evsc.virginia.edu/~evscta/EVSC430L/ TreeArchLab/L6javatree.html`.

[2] Henry S. Horn. *The Adaptive Geometry of Trees.* Princeton University Press ©1971.

[3] A. Carl Leopold. *Plant Growth and Development.* McGraw-Hill Book Company ©1975.

[4] R. F. Lyndon. *Plant Development.* Unwin Hyman Ltd. ©1990.

[5] Benoit Mandelbrot. *The Fractal Geometry of Nature.* W.H. Freeman and Company ©1983.

[6] David Suzuki & W. Grady. *Tree: A Life Story.* GreyStone Books ©2004.

[7] Hugh Wilcox. *Cambial Growth Characteristics,* Tree Growth. ed. Theodore Kozlowski, The Ronald Press Company ©1962.