UPPSALA
UNIVERSITET

U.U.D.M. Project Report 2013:23

The Brownian tree

Christoffer Cambronero

Examensarbete i matematik, 15 hp
Handledare och examinator: Jakob Björnberg
Augusti 2013

Department of Mathematics
Uppsala University

Contents

1 Introduction

The purpose of this paper will be to show how to generate different types of trees using basic probability. We start the paper by defining Brownian motion which is used to simulate different types of trees later. Brownian motion uses the normal distribution and thanks to that becomes a unique stochastic process, but more on that in chapter 2. Later, in chapter 3, the brownian motion will be our tool to simulate trees. In this paper we look at two types of trees, Galton Watson trees (chapter 3) and Brownian trees (chapter 4).

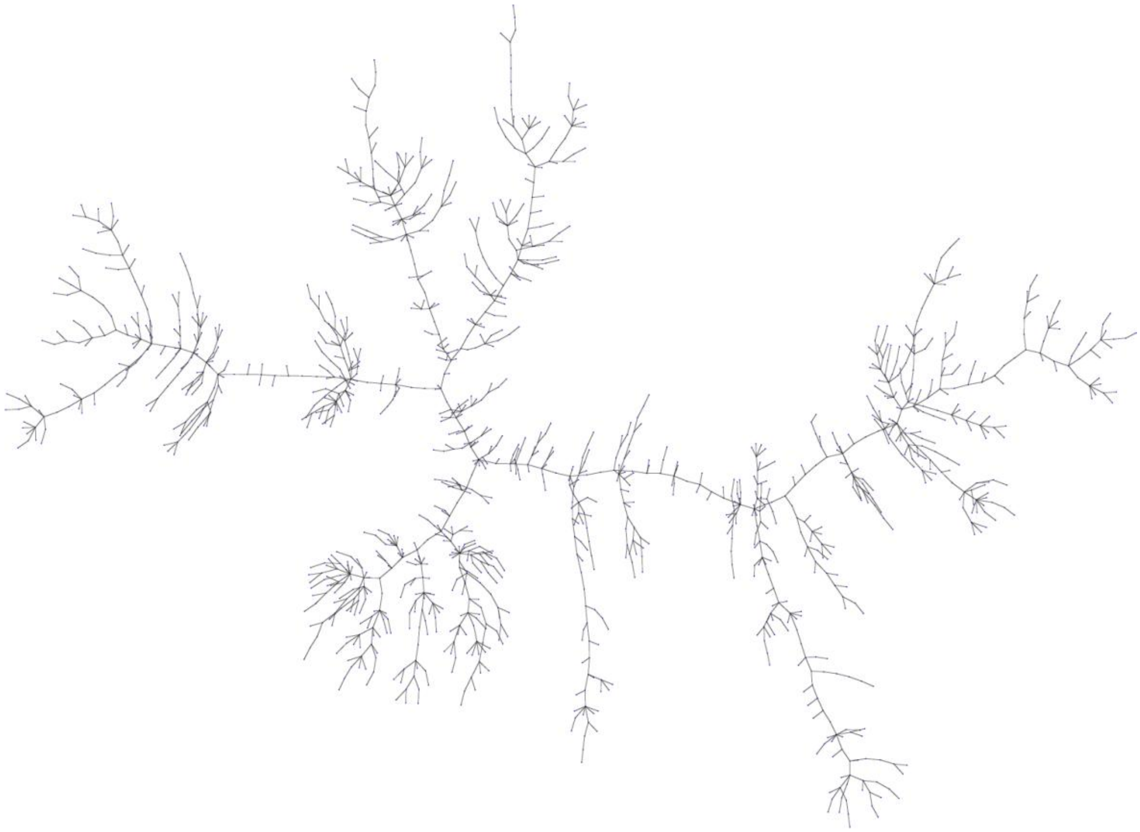

Figure 1.1. A tree containing 3000 nodes

2 Background

2.1 Brownian motion

The central object of this paper will be Brownian motion, which is defined as follows:

Brownian motion starting at x is the unique continuous stochastic process $\{B(t): t \geq 0\}$ such that the following holds:

- $B(0) = x\,, x \ \in \boldsymbol{R}$
- All the increments $B(t) - B(s), s < t$ are independent
- The increments $B(t) - B(s) \sim N(0, t - s)$

If $x = 0$ we say that the process $\{B(t): t \geq 0\}$ is a standard Brownian motion. In this paper we will only be talking about standard brownian motion.

There are many ways to construct a Brownian motion. In this paper Paul Lévy's and Donsker's construction will be presented.

2.2 Paul Lévy's construction

The Paul Lévy construction is a way to generate a Brownian motion on the interval $[0,1]$. The idea is to start by creating the values of the process at two points and approximate the process between these points by a straight line and then, step by step, create more points between the existing points and draw new lines between them. The amount of points and where they will be placed are determined by

$$D_n = \{\frac{k}{2^n} : 0 \ \leq k \ \leq \ 2^n \ \}$$

where n is the nth step of the construction, starting at step $n = 0$. Also let D be defined as

$$D = \bigcup_{n=0}^{\infty} D_n$$

and let $\{Z_d : d \in D\}$ be independent such that if $d \in D_n \backslash D_{n-1}$ then $Z_d \sim N(0, 2^{-(n+1)})$. $B^n(t)$ will be the value of t in step n. Lastly, let $\forall n \ B^n(0) = 0$.

Now that everything has been defined the construction can begin. For the first step, $n = 0$, the set of points will be $D_0 = \{0,1\}$ and we define

$$B^0(t) = \begin{pmatrix} Z_1 \quad if\ t = 1 \\ 0 \quad if\ t = 0 \\ and\ linear\ in\ between \end{pmatrix}$$

For the next steps, $n \geq 1$,

$$B^n(t) = \begin{pmatrix} 2^{-(n+1)/2} * Z_t \quad if\ t \in D_n \backslash D_{n-1} \\ B^{n-1}(t) \qquad if\ t \in D_{n-1} \\ and\ linear\ between\ the\ points\ in\ D_n \end{pmatrix}$$

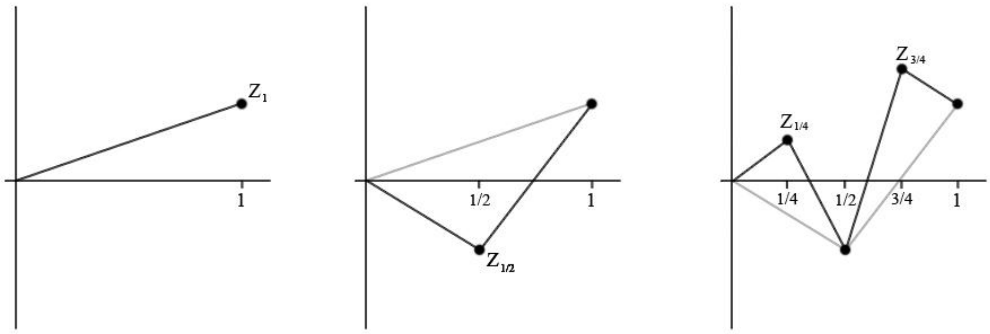

Figure 2.1 The first three steps in the Lévy constrction

It can be shown [1] that the functions $(B^{(n)}(t): t \in [0,1])$ converge to a standard Brownian motion as stated more precisely in 2.4 below.

2.3 Donsker construction

Let X_n be independent with the same distribution, such that $\forall i\ E(X_i) = \mu, V(X_i) = \sigma^2 < \infty$. Let $S_0 = 0$ and create the random walk

$$S_n = \sum_{i=1}^{n} X_i$$

and let $S(t)$ for $t \geq 0$ be given by interpolating between the integer points, i.e.

$$S(t) = S_{\lfloor t \rfloor} + (t - \lfloor t \rfloor) * (S_{\lfloor t \rfloor + 1} - S_{\lfloor t \rfloor})$$

[1] Mörters and Peres. Brownian Motion, chapter 1

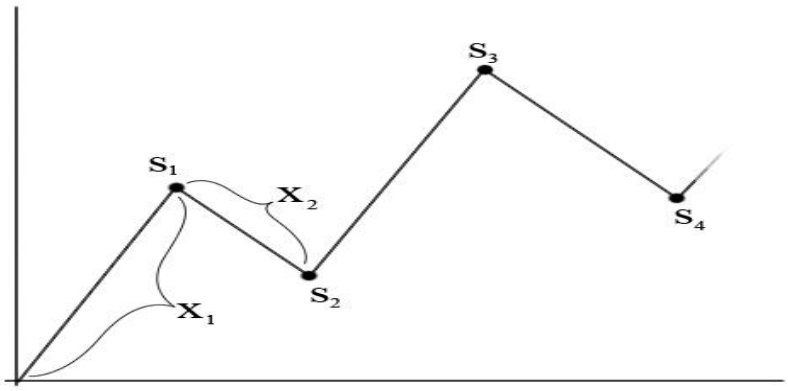

Figure 2.2 The first steps of the Donsker construction

Assume now that $\mu = 0$ and $\sigma^2 = 1$, this can easily be done by considering $\frac{X_i - \mu}{\sigma}$ and this way we don't lose any generality. Let us now define

$$B^{(n)}(t) = \frac{S(n * t)}{\sqrt{n}} \text{ ,} n \ \geq 1$$

For $t = 1$ we have

$$B^{(n)}(1) = \frac{S(n)}{\sqrt{n}} \to N(0,1)$$

when $n \to \infty$, thanks to the central limit theorem. Donsker's theorem[2] states that the entire procress $(B^{(n)}(t): t \in [0,1])$ converges to a standard Brownian motion as stated more precisely in 2.4 below.

2.4 Theorem

In both these approximations $B^n(t): t \in [0,1]$ when $n \to \infty$ converges to $B(t)$ in the sense that

$$P\left(\lim_{n\to\infty} \max_{t\in[0,1]} \left|B^{(n)}(t) - B(t)\right| = 0\right) = 1$$

Observe that if $B_{1,} B_2, \ldots$ are independent brownian motions on the interval [0,1] then one can concatenate these to create a new brownian motion on $[0, \infty)$

[2] Mörters and Peres. Brownian Motion, theorem 5.22

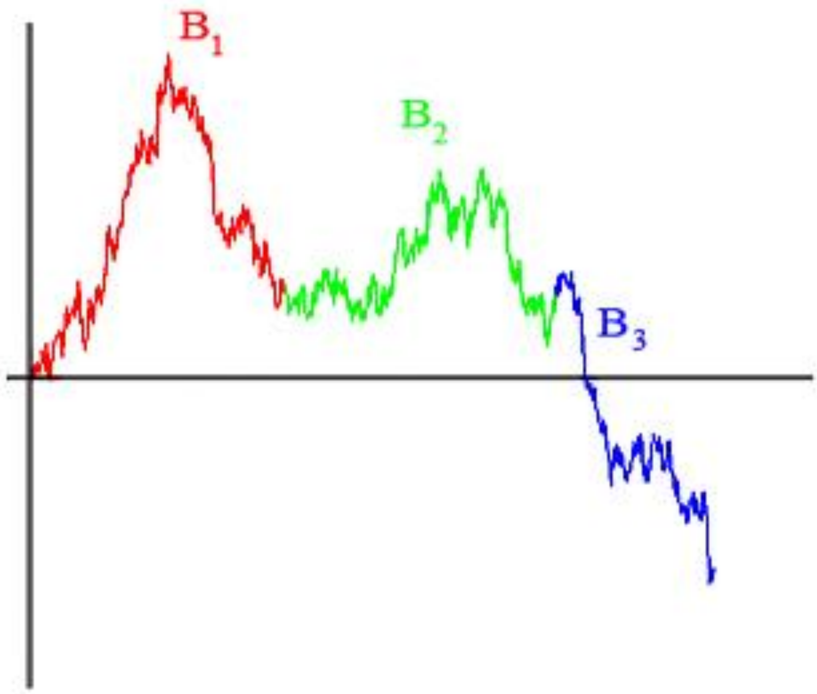

Figure 2.3. A concatenation of three Brownian motions

3 Galton Watson trees

3.1 Galton Watson trees definition

A tree in graph theory is defined as an undirected connected graph, free of any loops. In this section we will discuss Galton Watson trees. Galton Watson trees are defined as trees that start with one node and with some random variable gets a number of children. By children we mean nodes that are connected with the node which had the children. Each of these children has with the same random variable a number of children independently of the rest of the tree.

A Galton Watson tree can be used as a model to simulate populations. Let the probability of offsprings be $p = (p_0, p_1, p_2, \dots)$ and let X be a random variable such that $P(X = k) = p_k$. Also let the expected value be

$$\mu = \sum_{k=0}^{\infty} k * p_k$$

and assume that the variance $\sigma^2 = V(X) < \infty$. We let Z_n define the total number of individuals in generation n defined as follows. By default $Z_0 = 1$ and the unique individual at generation 0 has k children with probability p_k. Now let $(X_j^{(n)}; n, j \geq 1)$ be independent with $P\left(X_j^{(n)} = k\right) = p_k \ \forall k$. We interpret $X_j^{(n)}$ as the number of offspring in the *n*th generation by the *j*th individual in that generation, and we recursively define

$$Z_n = \sum_{j=1}^{Z_{n-1}} X_j^{(n)}$$

The first question one may ask is whether the the tree dies out, meaning that it's finite. For the tree to die out an entire generation has to have zero offsprings, $m = P(\exists n : Z_n = 0)$.

3.1.1 Theorem

$$m = 1 \ if \ \mu \leq 1 \ (called \ subcritical \ and \ critical \ cases)$$
$$m < 1 \ if \ \mu > 1 \ (called \ the \ supercritical \ case)$$
$$in \ cases \ where \ p_k > 0 \ for \ some \ k \geq 2.$$

3.1.2 Proof

Let

$$G(s) = E(s^{z_1}) = \sum_{k \geq 0} s^k * p_k \quad for \ 0 \leq s \leq 1$$

and let $m_n = P(Z_n = 0)$. Note that $m_n \leq m_{n+1}$ because if $Z_n = 0$ then $Z_{n+1} = 0$. Hence

$$m = \lim_{n \to \infty} m_n$$

we have

$$m_1 = p_0 = G(0)$$

$$E(s^{Z_2}) = E\left(s^{\sum_{j=1}^{Z_1} Z_j^{(1)}}\right) = E(E\left(s^{Z_1}\right)^{Z_1^{(1)}}\right) = G(G(s))$$

where $Z_1, Z_j^{(1)}$ are independent, all with distribution p. So

$$m_2 = P(Z_2 = 0) = G\big(G(0)\big) = G(m_1)$$

and in the same way

$$m_n = G(m_{n-1})$$

G is continuous in s, so letting $n \to \infty$ implies that $m = G(m)$.

Note that $G(s)$ is increasing in s, and in fact convex in $s_{\in(0,1)}(G''(s) \geq 0)$

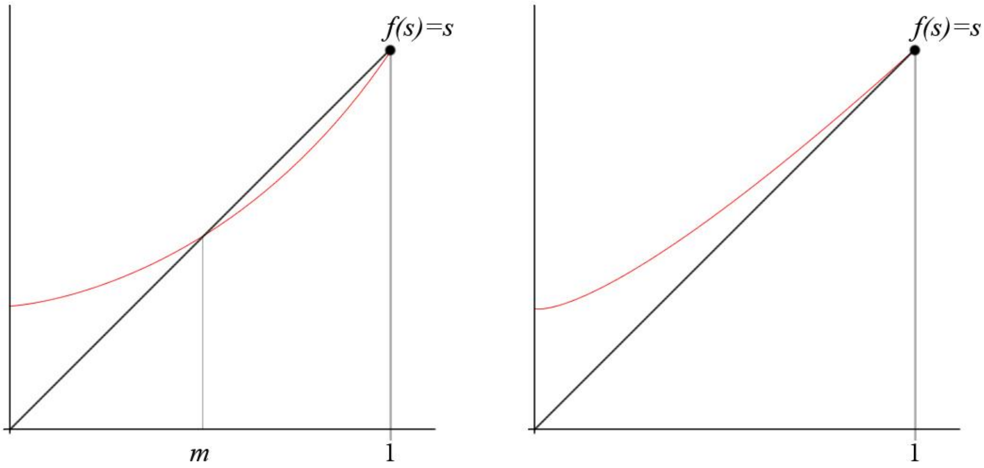

Figure 3.1. The graph to the right shows $G'(1) > 1$, the one to the left shows $G'(1) \leq 1$

By looking at the figures above one can see that the only way for $m < 1$ is if $G'(1) > 1$. And since

$$G'(s) = \sum_{k=0}^{\infty} k * s^{k-1} * p_k$$

and set $s = 1$ gives

$$G'(1) = \sum_{k=0}^{\infty} k * 1 * p_k = \mu$$

i.e.

$$m < 1 \; iff \; \mu > 1$$

3.2 Critical case

So if $\mu \leq 1$ the tree will be finite and a very interesting case is when $\mu = 1$, the critical case. In the subcritical case, when $\mu < 1$, the tree is usually "small" with high probability. Assuming sufficient moments we have that $P\left(\sum_{j\geq 0} Z_j > n\right) \leq e^{-\alpha n}\ for\ some\ \alpha > 0$. For the critical case on the other hand, $P((\sum_{j\geq 0} Z_j > n) \approx n^{-\beta}$, meaning that when $n \to \infty$ the critical case decreases slower than the subcritical case regardless of the precise values of $\alpha > 0, \beta > 0$.

$$\frac{e^{-\alpha n}}{n^{-\beta}} = \frac{n^{\beta}}{e^{\alpha n}} \to 0\ when\ n \to \infty$$

i.e. the tree is finite but usually "very big" for the critical case compared to the subcritical case.

3.3 Representations of Galton Watson tree

In this section we will look at two different ways of representing a tree with a finite sequence of numbers. Both methods are used for different reasons and both are useful depending on what you are after.

3.3.1 Dyck paths

Start with a Galton Watson tree with $\mu = 1$. The size of the tree will be defined as the number of nodes in it. We will define excursion based on the tree by walking $2(n-1)$ steps along the edges as follows:

For every step we take along the edges that leads "further away" in the graph distance from the root the excursion goes up one and as we "gets closer" to the root the excursion goes down one.

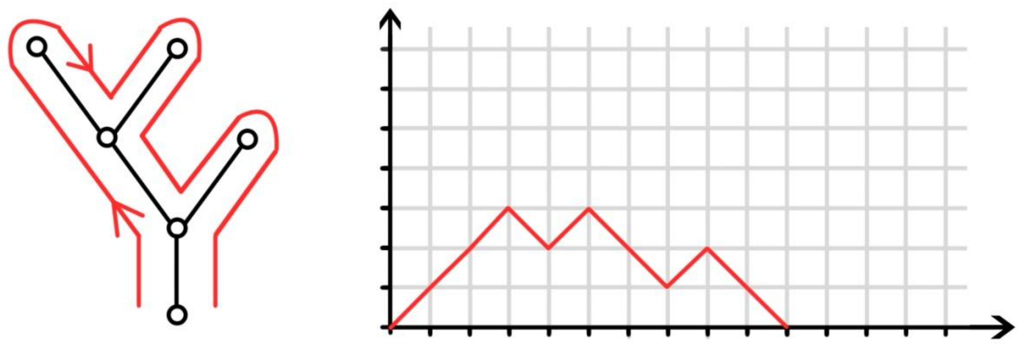

Figure 3.2. A tree with 6 nodes and an excursion that has walked 10 steps.

3.3.2 Depth first search (DFS)

The DFS representation provides a useful method for simulating a size-conditioned Galton Watson tree. The DFS will unlike Dyck paths excursion keep track of how many offspring each node has and will from that create its own excursion. Lets say a tree of size n is desired, then a sequence $(\xi_1, \xi_2, \dots, \xi_n)$ is needed where ξ_i is the numbers of offsprings on the ith node. Here the nodes are numbered by walking along the edges starting on the left side of the root, much like the Dyck path. Should a node which was already visited be visited again it will not be counted again.

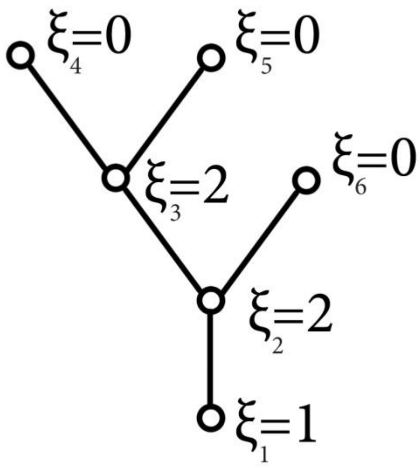

Figure 3.3. A tree with 6 nodes all numbered according to DFS

We also define $U_k = "\,The\ number\ of\ nodes\ still\ to\ explore\ after\ k\ steps\,"$.

$$U_0 = 1$$
$$U_1 = U_0 + \xi_1 - 1$$
$$\vdots$$
$$U_k = U_{k-1} + \xi_k - 1 = U_0 + \sum_{j=1}^{k} (\xi_j - 1)$$

Or simplified $U_k = 1 + \sum_{j=1}^{k} (\xi_j - 1)$. To obtain a tree of size n two criteria needs to be met.

$$(1) \qquad U_n = 0$$
$$(2) \qquad U_k > 0 \ \forall k < n$$

Lets rewrite (1) to easier use it

$$U_n = 1 + \sum_{j=1}^{n} (\xi_j - 1) \rightarrow \sum_{j=1}^{n} \xi_j = n - 1.$$

If we now obtain a sequence $(\xi_1, \xi_2, \dots, \xi_n)$ that meets (1) there exist a unique rotation that

$$(\xi_l, \xi_{l+1}, \dots, \xi_n, \xi_1, \xi_2, \dots, \xi_{l-1}) \rightarrow (\xi_1^1, \dots, \xi_n^1)$$

that meets (2), i.e. $U_k > 0 \; \forall k < n$. (This follows from the so-called Dvoretzky-Motzkin cycle lemma[3])

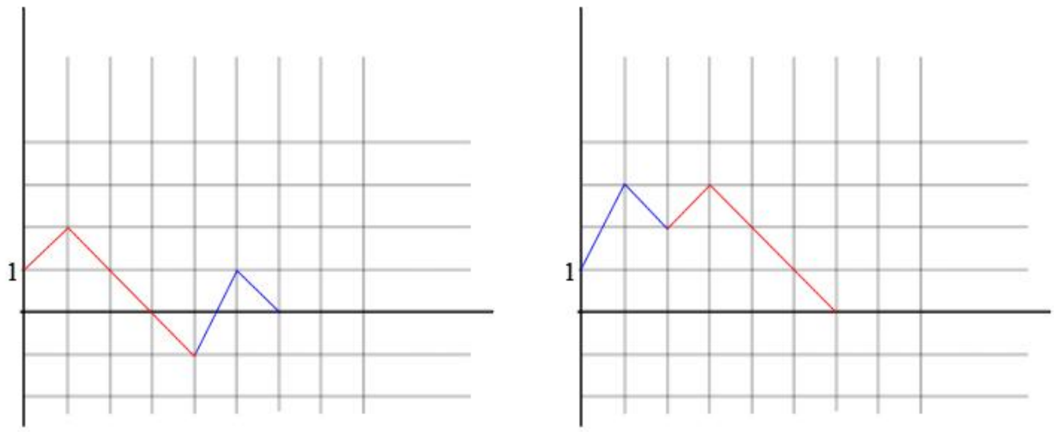

Figure 3.4. Shows a DFS before and after rotation

As seen in fig 3.4, where $n = 6$, although $U_6 = 0$ both U_3 and U_4 fail on criterion (2). After the rotation all U_k $k < n$ fulfill (2).

Since we can generate sequences of length n until we get one that meets (1), $\sum_{j=1}^{n} \xi_j = n - 1$, and after that rotate the sequence to fulfill (2) we can now create a Galton Watson tree with the desired size n with the rotated sequence.

Observe that if we look at the critical case when $E(\xi_j) = 1$ it will lead to $E\left(\sum_{j=1}^{n} \xi_n\right) = n$ which are very close to the sought number $n - 1$. Meaning our generations of sequence won't take so long before fulfilling criterion (1).

[3] Luc Devroye, Simulating size-constrained Galton-Watson Trees, p. 5

4 Brownian tree

We will start by explaining how to construct a tree from a continuous function. Start by letting $f(t)$ be a continuous function on the interval [0,1] with $f(0) = f(1) = 0$ and $f(t) \geq 0 \forall t \in [0,1]$. To construct a tree we say that we "glue together" any point x, y such that $f(x) = f(y)$ and $f(t) > f(x) \forall t \in (x, y)$. Upon doing so every local minimum will act as a branching point and every local maximum will act as an ending point.

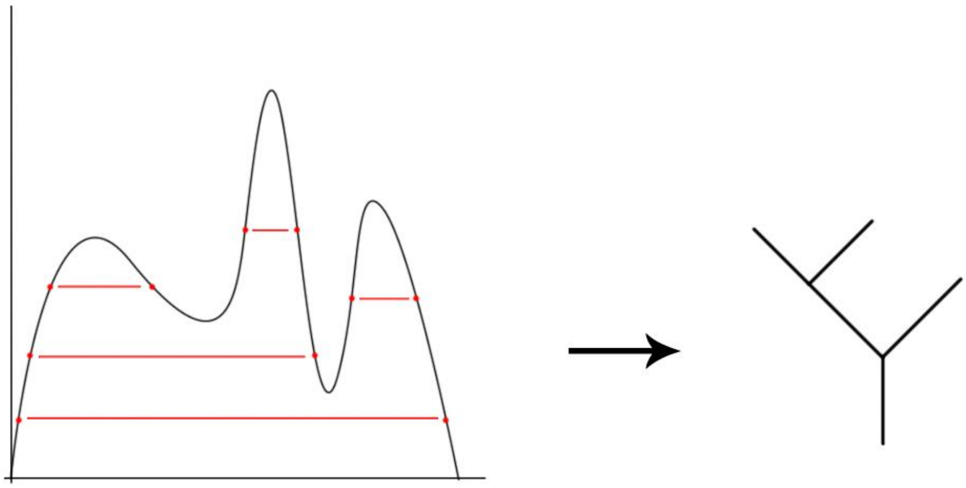

Figure 4.1. A tree constructed from a function

To now create a brownian tree we need to use a function f with the properties above, which is derived from brownian motion. A brownian motion $B(t)$ itself however does not meet the requirement of $B(t) \geq 0 \forall t \in [0,1]$. To solve this problem we start by defining the brownian bridge and then the brownian excursion.

4.1 Brownian bridge and brownian excursion

One can "force" a brownian motion to return to 0 at a given time. Lets define

$$\text{ƀ}_t = B(t) - \mathrm{t} * B(1)$$

Then ƀ_t is called a brownian bridge. By defining ƀ_t as above we obtain $\text{ƀ}_0 = B(0) = 0$ and $\text{ƀ}_1 = B(1) - B(1) = 0$ i.e. a brownian motion that returns to 0 at time 1.

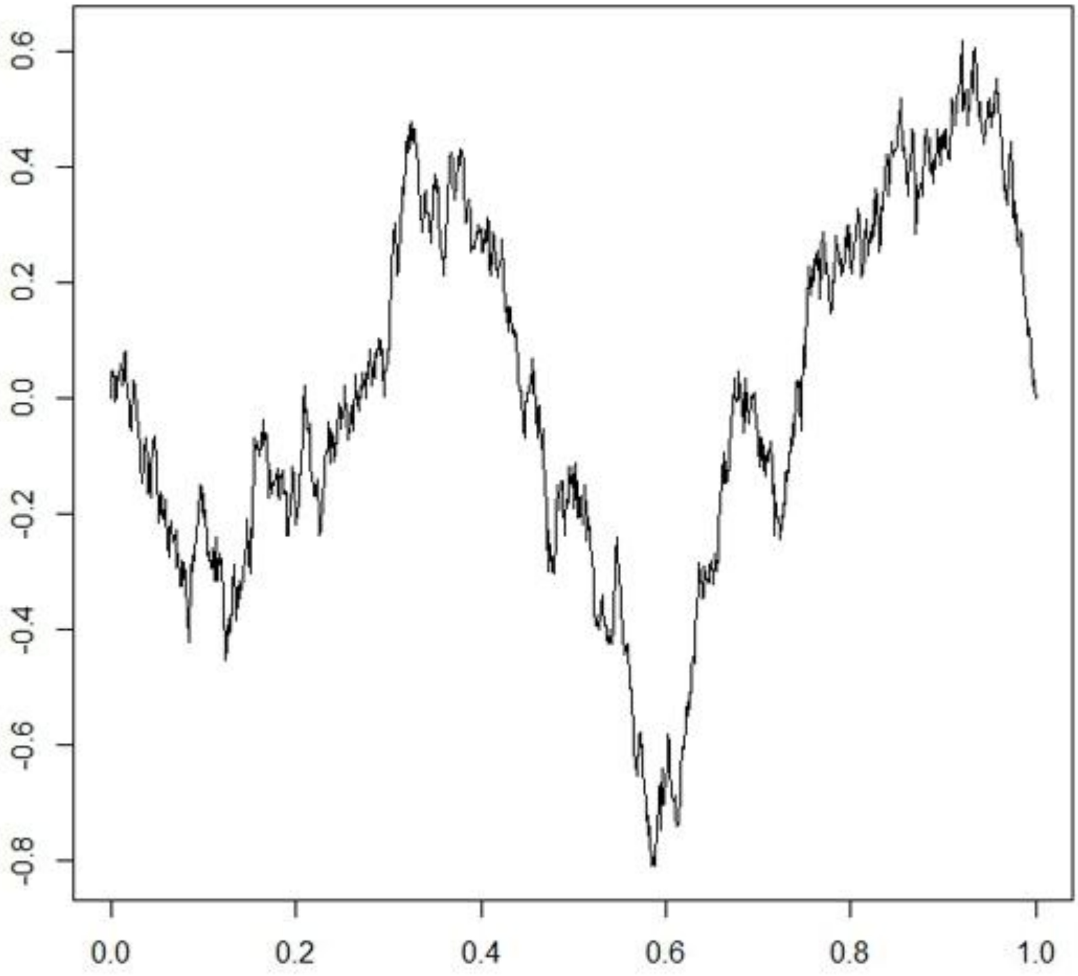

Figure 4.2. A brownian bridge

In order to obtain a "Brownian" function that is $\geq 0 \forall t \in [0,1]$ we will now use the fact that a brownian bridge returns to 0 at time 1. We define the Brownian excursion $\bar{e}_t$ as following:

$$\bar{e}_t = ƀ_{\tau+t(mod\ 1)} - ƀ_\tau$$

where $ƀ_\tau$ is the minimum of ƀ(this is in fact unique with probability 1). The Brownian excursion will now be a continuous positive brownian function on the interval [0,1].

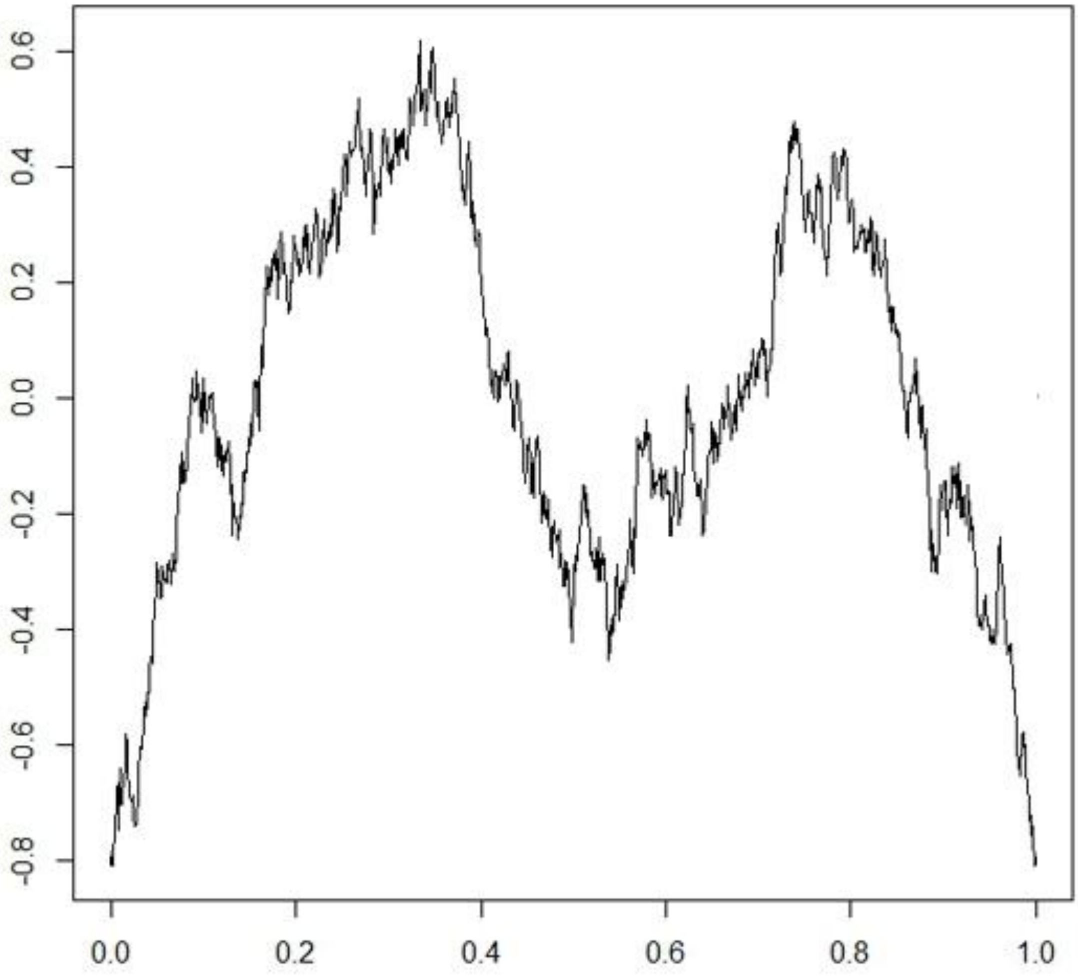

Figure 4.3. A brownian excursion

By using the brownian excursion, $\bar{e}_t$, as our function for creating a tree a brownian tree will be obtained.

4.2 Convergence theorems

Let T_n be a Galton Watson tree where we condition on the size n of the tree. Also let $ex_{T_n}(t)$ be the excursion from 3.3.1, i.e. $ex_{T_n}(t)$ is the distance from the root at the tth step. Aldous proved[4]([DA, Theorem 23]) that if we let $n \to \infty$ then

$$\sigma * n^{-\frac{1}{2}} * ex_{T_n}(t) \to 2 * \bar{e}_t$$

In this sense the size conditioned Galton Watson trees T_n converges to the Brownian tree.

[4] David Aldous. The Contunuum Random Tree III, theorem 23

4.3 An interesting choice of p

As mentioned earlier in theorem 3.1.1, $m = 1 \; if \; \mu \leq 1$ where m was the probability that the tree died out, we shall now look at the critical case when $\mu = 1$. We have already established that trees where $\mu = 1$ are usually big, but there are many choices of p to obtain $\mu = 1$. What we will look at now is called the geometric distribution.

$$p_k = 2^{-(k+1)} \; (k \geq 0)$$

Observe that choosing p_k as above gives

$$\mu = \sum_{k=0}^{\infty} p_k = \sum_{k=0}^{\infty} 2^{-(k+1)} = \frac{1}{2}\sum_{k=0}^{\infty} 2^{-k} = \frac{1}{2} * \frac{1}{1-\frac{1}{2}} = 1$$

The geometric distribution, according to Aldous, will have an easy excursion description.

4.4 Theorem

Let $X_1, \; X_2, \dots$ be independently ± 1 with probability $\frac{1}{2}$.

$$ex_{T_n}(k) \; will \; have \; the \; same \; distribution \; as \; X_1 + X_2 + \cdots + X_k \; conditioned \; on:$$

$$1) \sum_{j=1}^{2n} X_j = 0$$

$$2) \sum_{j=1}^{l} X_j \geq 0 \; \forall l \leq 2n$$

4.4.1 Proof

Call a sequence $X_1, \dots, X_{2n}$ accepted if (1) and (2) hold. Call a Galton Watson tree with

p as above accepted if it has size n. Let A_n denote the set of accepted sequences, i.e

$$A_n = \{Sequences \; a_1, \dots, a_{2n} \; so \; that \; (i) \sum_{j=1}^{2n} a_j = 0 \; and \; (ii) \sum_{j=1}^{l} a_j \geq 0 \; \forall l \leq 2n \}$$

We denote the size of the set A_n by C_n. In fact C_n is a Catalan number. Thus A_n is the set of accepted sequences of length $2n$. We hace two ways of sampling elements of A_n, lets call them way 1 and way 2.

Way 1

Sample $X_1, \dots, X_{2n}$ as above conditioned on (i) and (ii):

- Sequences $x_1, \dots, x_{2n}$ that are ± 1 and independent probabilities

Way 2

- Sample a Galton Watson tree with distribution ξ, and condition on size n for the tree.
- Create an excursion from the tree as in section 3.3.1

Both these ways gives the probabilities to all elements in $\underline{a} \in A_n$. Define $P_1(\underline{a})$ and $P_2(\underline{a})$ as the probability to get an element from A_n using way 1 respective way 2. We now want to show that

$$P_1(\underline{a}) = P_2(\underline{a})\ for\ \forall \underline{a} \in A_n$$

We do this by showing that both $P_1(\underline{a})$ and $P_2(\underline{a})$ do not depend on a. It follows that $P_1(\underline{a}) = P_2(\underline{a}) = \frac{1}{C_n}$.

We start by considering way 1.

Let's fix $a_1, \dots, a_{2n}$so that $\sum_{j=1}^{2n} a_j = 0$ and $\sum_{j=1}^{l} a_j \geq 0\ \forall l \leq 2n$.

$$P_1(\underline{a}) = P\left(X_1 = a_1, \dots, X_{2n} = a_{2n} \middle| \sum_{j=1}^{2n} X_j = 0, \sum_{j=1}^{l} X_j \geq 0\ \forall l \leq 2n\right) =$$

$$= \frac{P(X_1 = a_1, \dots, X_{2n} = a_{2n}, \sum_{j=1}^{2n} X_j = 0, \sum_{j=1}^{l} X_j \geq 0\ \forall l \leq 2n)}{P(\sum_{j=1}^{2n} X_j = 0, \sum_{j=1}^{l} X_j \geq 0\ \forall l \leq 2n)}$$

In the numerator the second and third expressions are implied by the first leading to

$$P_1(\underline{a}) = \frac{P(X_1 = a_1, \dots, X_{2n} = a_{2n})}{P(\sum_{j=1}^{2n} X_j = 0, \sum_{j=1}^{l} X_j \geq 0\ \forall l \leq 2n)} =$$

$$= \frac{2^{-2n}}{\sum P(X_1 = b_1, \dots, X_{2n} = b_{2n})}$$

Where the sum is over all $b_1, \dots, b_{2n}$ meeting the criteria (1) and (2). For each fixed sequence $b_1, \dots, b_{2n}$ in the sum we have $P(X_1 = b_1, \dots, X_{2n} = b_{2n}) = 2^{-2n}$ and the number of such sequences is what we called C_n.

$$\frac{2^{-2n}}{2^{-2n} * C_n} =$$

$$= \frac{1}{C_n}$$

We now know that the probability of all accepted sequence is $P_1(\underline{a}) = \frac{1}{C_n}$.

We now want to show that all accepted trees have the same probability. Since the number of trees of size n equals the number of accepted sequences, this will prove the claim.

To show this we will use DFS to represent a Galton Watson tree. To generate a tree of size n is the same as generating a sequence $(\xi_1, \xi_2, \dots, \xi_n)$ conditioned on $U_n = 0$ and $U_k > 0 \ \forall k < n$. So what is the probability of a given sequence? If we define $b_1, \dots, b_{2n}$ as numbers such that $(iii) \sum_{j=1}^{n}(b_j - 1) = 0$ and $(iv) \sum_{j=1}^{k}(b_j - 1) > 0 \ \forall k \le n$ then

$$P(\xi_1 = b_1, \dots, \xi_n = b_n | (iii) \ and \ (iv) \ for \ \xi_1, \xi_2, \dots, \xi_n) =$$

$$\frac{P(\xi_1 = b_1, \dots, \xi_n = b_n, (iii) \ and \ (iv) \ for \ \xi_1, \xi_2, \dots, \xi_n)}{P\big((iii) \ and \ (iv) \ for \ \xi_1, \xi_2, \dots, \xi_n\big)} =$$

Once more the second expression in the numerator is determed by the first and the denominator is some function based on n

$$\frac{1}{f(n)} * P(\xi_1 = b_1, \dots, \xi_n = b_n) =$$

$$\frac{1}{f(n)} * 2^{-n} * 2^{-\sum_{j=1}^{n} b_j}$$

but from (iii) we get

$$\frac{2^{-2n+1}}{f(n)} = p(n)$$

which is some function that does not depend on b. So we must have $P_1(\underline{a}) = P_2(\underline{a}) = \frac{1}{C_n}$ as discussed above.

4.5 Connections

Aldous proved in the general critical case that the excursion when we let $n \to \infty$ will converge to a brownian excursion. This can be understood when one thinks about how the Donsker construction works and the condition of never letting the sum take negative values and end in value 0. Also the tree generated from this will tend to some kind of brownian tree.

Intuitively the same should apply independently of the choice of p. Especially when one thinks of how Donsker construction works, recall that all that is stated is that they need to be same distributed. It's true that this holds, but we don't get it as easily as in theorem 3.4.

5. Simulation

There are many ways to simulate trees thanks to the countless numbers of programs today. But by just simulate a tree in the critical- or subcritical case, when the chance of the tree dying out is 1, usually tends to give a small tree.

Lets say a tree with 700 nodes is desired. One way could be to simulate a tree and count the nodes and if it's not 700 nodes rerun the simulation untill the tree contains 700 nodes. This method could however take very long time and use much computer power. That's why, as mentioned before, the DFS is a good method when simulating trees.

The first thing one has to do now is just to simulate a sequence of 700 numbers and see if the sum of them is 699. Since the expected value of each number is 1 (in the subcritical case) acquire a sum of 699 won't take to long. After that is done the sequence must be rotated untill $U_k > 0 \; \forall k < 700$, see 3.3.2 for more details.

Now a sequence is obtained that can be used for DFS. The hard part can be how to use the sequence, remember how the nodes are number in DFS. One way is to write a program that keep track of which nodes are connected to which nodes. Finally use a program(for example graphviz) that draws up the tree using the pair nodes as mentioned above.

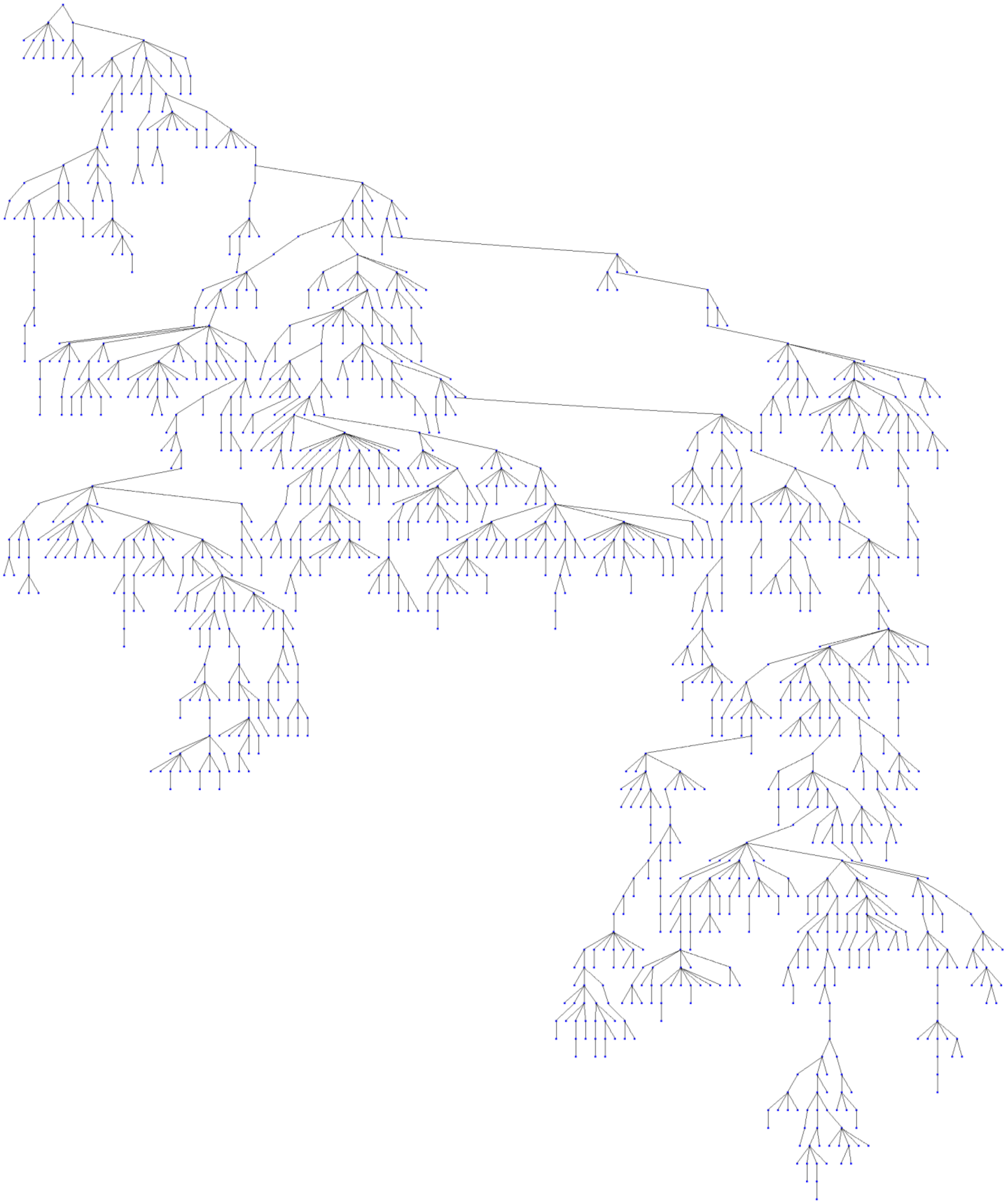

Figure 5.1. A Galton Watson tree containing 700 nodes

Bibliography

[1] Peter Mörters and Yuval Peres, *Brownian Motion*. Cambridge University Press, New York, 2010

[2] David Aldous, *The Continuum Random Tree III*, University of California, Berkeley, 1993

Published online

[3] Luc Devroye, *Simulating size-constrained Galton-Watson Trees*. http://luc.devroye.org/gw-simulation.pdf (Retrieved 2013-08-13)