Random Triangulations (Extended Abstract)*

Luc Devroye
McGill University
luc@kriek.cs.mcgill.ca

Philippe Flajolet
INRIA - Rocquencourt
Philippe.Flajolet@inria.fr

Ferran Hurtado
Universitat Politecnica de Catalunya
hurtado@ma2.upc.es

Marc Noy
Universitat Politecnica de Catalunya
noy@ma2.upc.es

William Steiger
Rutgers University
steiger@cs.rutgers.edu

April 4, 1996

Abstract

Let T_n denote the set of triangulations of a convex polygon K with n sides. We study functions that measure very natural "geometric" features of a triangulation $\tau \in T_n$, for example $\Delta_n(\tau)$ which counts the maximal number of diagonals in τ incident to a single vertex of K. It is familiar that T_n is bijectively equivalent to B_n, the set of rooted binary trees with $n-2$ internal nodes, and also to P_n, the set of non-negative lattice paths that start at 0, make $2n-4$ steps X_i of size ± 1, and end at $X_1 + \cdots + X_{2n-4} = 0$. Δ_n and the other functions translate into interesting properties of trees in B_n, and paths in P_n, that seem not to have been studied before. We treat these functions as random variables under the uniform probability on T_n and can describe their behaviour quite precisely. A main result is that Δ_n is very close to $\log n$ (all logs are base 2). Finally we consider the random generation of elements triangulations and of triangulations with certain fixed properties. (KEYWORDS: triangulation, random binary tree, probabilistic analysis, computational geometry)

1 Introduction and Summary

Consider a convex polygon K with n sides. We label the vertices $v_i = i$, $i = 0, \ldots, n-1$, in clockwise order. A triangulation is a set of $n-3$ non-crossing diagonals $v_i v_j$ which partitions K into $n-2$ triangles. You can imagine constructing a triangulation τ recursively: taking the polygon edge $v_0 v_{n-1}$ as base, just choose the apex of the triangle of τ that it belongs to, say v_i, $0 < i < n-1$, and now continue in the same way on the two polygons $v_0, \ldots, v_i$ with $v_0 v_i$ as base and $v_i, \ldots, v_{n-1}$ with $v_i v_{n-1}$ as base (Figure 1 has $n = 8$ and $v_i = 2$, 4, and 5 respectively). This shows that t_n, the number of such triangulations, satisfies

$$t_n = t_2 t_{n-1} + t_3 t_{n-2} + \cdots + t_{n-1} t_2 \,, \; t_2 = t_3 = 1 \,,$$

*Part of the work of W. Steiger was supported by a grant XXX to UPC and part by NSF grant CCR-9111491. The work of L. Devroye is supported by NSERC grant A3456. The work of Philippe Flajolet is supported by the L.T.R. Project Alcom-IT (# 20244) of the European Union.

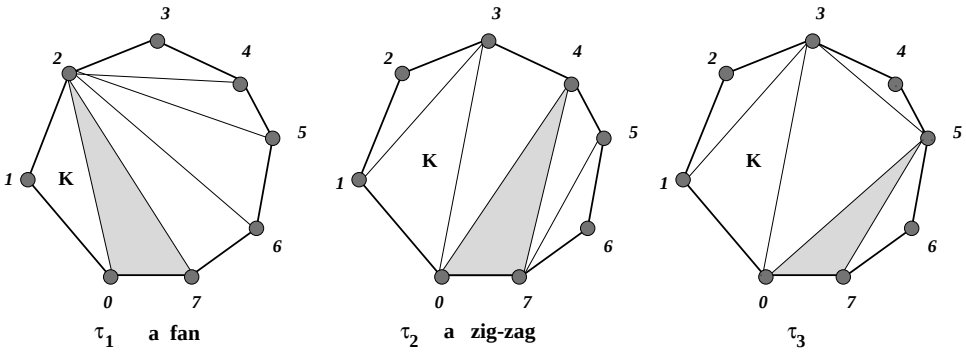

Figure 1: *Three triangulations:* $\Delta_8 = 5, 2,$ *and* $3,$ *respectively;* $\lambda_8 = 4, 4,$ *and* $3,$ *respectively.*

the recursion of the Catalan numbers. Therefore

$$t_n = C_{n-2} = \frac{1}{n-1}\binom{2n-4}{n-2} \tag{0}$$

is the size of T_n, the set of triangulations of K.

It is natural to consider certain "geometric" features of a triangulation $\tau \in T_n$. Let d_i denote the *degree* of vertex v_i, the number of diagonals of τ incident with v_i. It is easy to see [10] that the degree sequence characterizes τ. In this paper we study

$$\Delta_n(\tau) = \max(d_i,\, i = 0, \ldots, n-1)\,, \tag{1}$$

the maximal degree of the vertices. $\Delta_n = 2$ when τ is a *zig-zag* and $n-3$ when it is a *fan* ($d_i = n-3$ for some vertex), as in Figure 1.

Define the length of a diagonal v_iv_j with $i > j$ to be $\|v_iv_j\| = \min(i-j, n-i+j)$, the (fewest) number of successive edges of K between the endpoints. Another geometric feature of τ that we look at is

$$\lambda_n(\tau) = \max(\,\|v_iv_j\| \,:\, v_iv_j \in t), \tag{2}$$

the length of the longest diagonal in the triangulation. It is clear that $n/3 \le \lambda_n \le n/2$.

To see how these functions behave across the family of triangulations we treat them as random variables under the uniform probability on T_n. By symmetry, each d_i has the same distribution, but they are not independent because, e.g., $d_0 + \cdots + d_{n-1} = 2(n-3)$. In view of the fact that the expected degree of each v_i is $2(1-3/n)$, it may be somewhat surprising that Δ_n is close to $\log n$ (all logs are base 2). The main result is

Theorem 1 *As* $n \to \infty$,

$$E(\Delta_n)/\log n \to 1\,.$$

In fact Δ_n *is strongly concentrated. For all* $c > 0$, *as* $n \to \infty$,

$$Prob\{\,|\Delta_n(\tau) - \log n| \le (1+c)\log\log n\} \to 1\,.$$

The upper bound is based on computing the distribution of d_i.

Lemma 1 *For each vertex* v_i, *the probability that its degree is* k *is given by*

$$Prob(d_i = k) = \binom{k+1}{2}\binom{n-1}{2n-5}\prod_{i=1}^{k}\frac{n-2-i}{2n-5-i}\,. \tag{3}$$

Remark 1: Since $\text{Prob}(d_i = k) \leq (k+1)2^{-(k+1)}$ when $n > 3$ (which we assume throughout), this says that d_i has tails that decrease geometrically fast. Theorem 1 indicates that their maximum is logarithmic, like the max of n independent geometric random variables (see also Final Remark 1). The proof makes these connections more explicit. It is interesting to wonder about the variance of Δ_n. Simulation indicates that it could be constant (like that of the max of n independent geometrics).

The key fact about the longest diagonal is

Lemma 2 *The distribution of the length of the longest diagonal is given by*

$$Prob(\lambda_n = k) = \frac{nC_{k-1}}{2C_{n-2}} \sum_{i=n-k}^{2k} {}_{(*)}C_{i-k-1}C_{n-i-1} \,. \tag{4}$$

This enables us to find the limit distribution of λ_n.

Theorem 2 *For each $x \in (1/3, 1/2)$, as $n \to \infty$, $Prob(\lambda_n \leq nx) \to$ to the distribution with density*

$$w(x) = \frac{1}{\pi} x^{-2}(1-x)^{-2}(3x-1)(1-2x)^{-1/2} \,.$$

In addition $E(\lambda_n)/n \to \alpha$, where

$$\begin{aligned} \alpha &= \frac{\sqrt{3}}{\pi} + \frac{1}{3} - \frac{\log(2+\sqrt{3})}{\pi} \\ &= 0.465461510476142649488548021603185025843925680109356093 3. \end{aligned}$$

A motivation for the present work - along with deep curiosity about how typical triangulations look - is the inherent interest of binary trees. It is familiar that T_n is bijectively equivalent to B_n, the set of rooted binary trees with $n-2$ internal nodes, each triangulation $\tau \in T_n$ corresponding to a particular tree $b(\tau) \in B_n$. The two features of triangulations that we study translate into interesting and natural properties of the corresponding trees. For example $\Delta_n(\tau)$ measures a property of $b(\tau)$ that we call the *external-node separation*, $\chi_n(b(\tau))$: this is the maximal distance in the tree between successive external nodes. $\lambda_n(\tau)$ measures a property of $b(\tau)$ that we call the *nearly half* measure, $H_n(b(\tau))$: it is the size of the largest subtree with not more than half the external nodes. Though trees have been studied intensively (e.g., [3], [4], [8], [11], [12]), we are unaware of any previous work on these two features. Theorem 1, Theorem 2, Lemma 1, and Lemma 2 thus appear to express interesting, new facts about trees, as well as about triangulations. In Section 2 we translate the functions Δ_n and λ_n into the context of binary trees. We also exploit the correspondence between T_n and non-negative lattice paths P_n; we interpret our functions in this set as well, to help with the proofs, which appear in Section 3.

In Section 4 we describe some linear-time algorithms to randomly generate elements of T_n. In addition let S_k denote the subset of triangulations in T_n with $\lambda_n(\tau) = k$. We show how to quickly generate triangulations restricted to S_k. Finally, if $d_i = 0$, the vertex v_i is called *an ear* of the triangulation. We show how to quickly generate triangulations with a given number of ears. This may be of some interest because ears of τ correspond to leaves of $b(\tau)$.

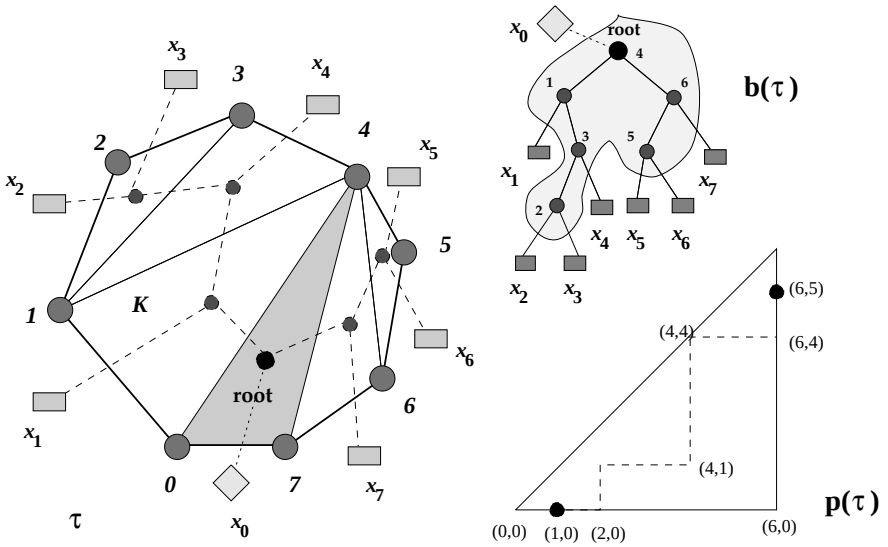

Figure 2: *A triangulation of K and two equivalent representations.*

Remark 2: If we regard the trees in B_n as binary-search-trees generated by permutations of $1, \ldots, n-2$, each permutation being equally likely, the bijection gives the (binary-search-tree) probability β, on T_n. Trees in B_n are well studied in this model (e.g., [4], [6], [11], [13]). In contrast to the situation in the uniform distribution, the vertex degrees in this model are not identically distributed. Actually $E_\beta(d_0) = E_\beta(d_{n-1}) = \Theta(\log n)$. We can prove that in this distribution $\Delta_n / \log n \to c > 1$ in probability. It seems difficult to analyze λ_n in this model.

2 Preliminaries

We first describe the explicit correspondences between triangulations, trees, and paths that we will use. The standard way to associate a tree with a triangulation uses the dual graph of $\tau \in T_n$. This gives a binary tree with $n-2$ internal nodes, one for each triangle of τ; adjacent triangles of τ correspond to nodes joined by an edge of the tree. The triangle with edge $v_0 v_{n-1}$ is associated with the root of the tree. If v_i is the apex of this triangle in τ, label the root with i. The left subtree has $i-1$ internal nodes (the number of vertices of K between v_0 and v_i) and corresponds to the triangulation of subpolygon $v_0, \ldots, v_i$ in τ; the right subtree has $n-2-i$ internal nodes and corresponds to the triangulation of subpolygon $v_i, \ldots, v_{n-1}$ in τ, and now continue recursively in the two subpolygons (subtrees). Once the $n-2$ internal nodes are placed, external nmodes are added so internal nodes have outdegree 2. Call this (binary-search) tree $b(\tau)$. It has $n-1$ external nodes whose *inorder* traversal corresponds to the edges $v_{i-1} v_i$, $i = 1, \ldots, n-1$. The root misses an external node corresponding to $v_0 v_{n-1}$. We label them x_i, $i = 1, \ldots, n-1$ and the missing external node, x_0 (see Figure 2). This scheme defines a bijection $T_n \leftrightarrow B_n$.

For $\tau \in T_n$ construct a path $p(\tau) \in P_n$ as follows (we think of elements of P_n as up-right rectilinear paths joining points in the integer lattice in R^2 contained in the triangle bounded by the x-axis, $x = n-2$, and $y = x$). Paths start at $(1,0)$ and end at $(n-2, n-3)$. Suppose τ has j_0 internal diagonals incident to v_0; then the path moves *right* j_0 steps. In general, let j_i denote the number of diagonals from v_i to a higher number vertex. We are currently at vertex v_0. We move clockwise in K to the next vertex v_i with $j_i > 0$. The path moves *up* to the line

$y = i$ and then moves *right* for j_i steps. It is easy to see that this procedure gives a path in P_n and that every such path comes from a distinct triangulation. These bijections are frequently exploited when studying the combinatorics in one of these sets (see especially [14]), and also for the task of randomly generating elements from one of the sets (e,g,. [2], [5], [13]).

To understand what Δ_n says about trees, imagine the diagonal v_iv_j in τ as directed from the smaller numbered vertex of K to the larger one. Take $0 < i < n-1$ and move counter-clockwise along the circumference of a sufficiently small circle centered at vertex v_i from edge $v_{i-1}v_i$ to edge v_iv_{i+1}. First we meet diagonals (if any) coming from lower vertices *into* v_i and then we meet diagonals (if any) going *out from* v_i to higher vertices. This shows that the degree of v_i in τ is the number of nodes in $b(\tau)$ *between* x_i and α, and the number of nodes between x_{i+1} and α, α being the root of the smallest subtree containing x_i and x_{i+1}; thus it is the path distance in $b(\tau)$ (number of internal nodes) from x_i to x_{i+1}, minus 1. Similarly, and because x_0 is missing from rooted binary trees, d_0 and d_{n-1} count the number of internal nodes between the root and x_1 and the root and x_{n-1}, respectively.

Given a rooted binary tree b with $n-2$ internal nodes and external nodes $x_1, \ldots, x_{n-1}$, tack an external node x_n ($\equiv x_0$) onto the root and define the *external-node separation* by

$$\chi_n(b) = \max(\, \|x_ix_{i+1}\|\,,\ i = 0, \ldots, n-1), \tag{5}$$

where $\|x_ix_{i+1}\|$ counts the path distance minus 1 between the external nodes. We just argued that

Lemma 3 *Given $\tau \in T_n$, $\Delta_n(\tau) = \chi_n(b(\tau))$.*

It is more difficult to interpret Δ_n on paths. From the construction of $p(\tau)$ the width j_i of the step along $y = i$ is the outdegree of vertex v_i, $i = 0, \ldots, n-3$ ($j_i = 0$ means the path has no step at level i). The outdegrees of v_{n-2} and v_{n-1} are zero. Similarly the indegrees of v_0 and v_1 are zero. The other indegrees are more complicated, except for v_{n-1}, where the indegree equals d_{n-1}, and both count the number of times the path meets $y = x$, from $x = 1$ to $x = n-3$. Also both d_{n-2} and the indegree of v_{n-2} can be determined from the intersections of the path with $y = x - 1$. However in general, d_i seems not to be an easily "seen" feature of the path.

Given a path $p \in P_n$, define its step-width by

$$s_n(p) = \max(j_i, i = 0, \ldots, n-2), \tag{6}$$

where j_i is the width of the step of p at height $y = i$. Since $d_i \geq j_i$,

Lemma 4 *Given a triangulation $\tau \in T_n$, $\Delta_n(\tau) \geq s_n(p(\tau))$.*

Therefore probabilistic lower bounds for step-width imply lower bounds for the maximum degree.

It is straightforward to interpret λ_n. From the construction of $b(\tau)$ from τ, each internal node in the tree other than the root corresponds to the part of τ restricted to some subpolygon $v_i, \ldots, v_j$, $i < j-1$. Therefore $\|v_iv_j\|$ corresponds to the number of external nodes in the subtree rooted at that particular internal node. Given a tree $b \in B_n$, denote its non-root internal nodes by ν_i and define $\|\nu_i\|$ as the number of external nodes in the subtree rooted at ν_i. The "nearly-half measure" of b is defined by

$$H_n(b) = \max(\min(\, \|\nu_i\|, n - \|\nu_i\|\,)\,,\ i = 1, \ldots, n-3). \tag{7}$$

Its the size of the largest subtree with not more than half the external nodes. Because $\lambda_n(\tau) = H_n(b(\tau))$, Lemma 3 gives the distribution of this random variable on trees.

3 Proofs

We sketch the proofs of the results mentioned previously. Some details may be omitted until the final version. We will apply the ballot theorem (see [7], p. 73) which says that the number of lattice paths that start at $(0,0)$, make i unit steps to the right, and $j \leq i$ unit steps up, and preserve $y \leq x$ is

$$N_{ij} = \frac{i+1-j}{i+1+j}\binom{i+1+j}{j}. \tag{8}$$

Proof of Lemma 1: Since the degrees are identically distributed, we only have to consider vertex v_0. If $d_0 = k$ in τ, the corresponding path (a good path) must start at $(1,0)$, pass through $(k+1,0)$ and then $(k+1,1)$, and finally continue to $(n-2,n-2)$. The number of ways a path can continue through $(k+1,1)$ to $(n-2,n-2)$ is

$$N = \frac{k+1}{2n-5-k}\binom{2n-5-k}{n-2},$$

by the ballot theorem. Since there is only one way a path in P_n can get from $(1,0)$ to $(k+1,0)$, N is also the number of good paths. Therefore $\text{Prob}(d_0 = k) = N/C_{n-2}$. Simplification gives (2). ∎

We prove Theorem 1 in two steps. For the upper bound we want to determine a k so $\text{Prob}(\Delta_n \geq k) \to 0$. Observe that $\text{Prob}(\Delta_n \geq k) = \text{Prob}(\bigcup_{i=0}^{n-1}\{d_i \geq k\})$ which, by Bonferroni's inequality ([7], p. 110) is bounded by $n\text{Prob}(d_0 \geq k)$. Lemma 1 shows

$$\text{Prob}(\Delta_n \geq k) \leq n(k+1)2^{-k}$$

which $\to 0$ for $k \geq \log n + c\log\log n$, $c > 1$.

The lower bound is

Lemma 5 *For any $c > 0$, Prob$(\Delta_n \leq k) \to 0$ for $k \leq \log n - (1+c)\log\log n$*

Proof: This is the only tricky part, because the d_i are dependent. From Lemma 4, Δ_n is larger than the size of the largest horizontal step, s_n, in the corresponding path, so we just need to determine k so that $\text{Prob}(\Delta_n(\tau) \leq k) \leq \text{Prob}(s_n(p(\tau)) \leq k) \to 0$.

Let $U_1, \ldots, U_{2n-4}$ be a sequence of i.i.d. uniform random variables on $[0,1]$. We describe (X_m, Y_m), the coordinates of a point on a random path in P_n, after $m < 2n-4$ steps of size 1, each up or right, starting from $(1,0) \equiv (X_0, Y_0)$. Of the C_{n-2} paths in P_n, C_{n-3} pass through $(1,1)$, the rest through $(2,0)$. Therefore, $I_{[A]}$ denoting the indicator of A,

$$X_1 = I_{[U_1 \leq C_3/C_2]}$$

$$Y_1 = 2 - X_1$$

will be $(1,1)$ or $(2,0)$ with the correct probabilities. Next suppose $(X_m, Y_m) = (i,j)$ is a point on p after $m = i+j-1 < 2n-5$ steps from $(1,0)$. By the ballot theorem there are

$$N_{i,j} = \frac{i-j+1}{2n-3-i-j}\binom{2n-3-i-j}{n-1-j}$$

continuations from (i,j) to $(n-2,n-2)$, of which $N_{i+1,j}$ go through $(i+1,j)$. Therefore the probability that the path at $(X_m,Y_m)=(i,j)$ moves RIGHT at step m is

$$p_m = \frac{N_{i+1,j}}{N_{i,j}} = \left(\frac{i+2-j}{i+1-j}\right)\left(\frac{n-2-i}{2n-4-i-j}\right), \tag{9}$$

which is 0 when $i=n-2$, and 1 when $i=j$, as required. If we define

$$X_{m+1} = X_m + I_{[U_{m+1}\le p_m]}$$

$$Y_{m+1} = Y_m + (1 - I_{[U_{m+1}\le p_m]})$$

our path will move from (i,j) to (X_m+1,Y_m) or (X_m,Y_m+1) with the correct probabilities. We use $m=i+j-1$ in the equation for p_m and simplify to see

$$p_m \ge \frac{1}{2}\left(1-\frac{1+m-2j}{2n-5-m}\right) \ge \frac{1}{2}\left(1-\frac{1+m}{2n-5-m}\right) \ge \frac{1}{2}\left(1-\frac{1+m^*}{2n-5-m^*}\right), \tag{10}$$

where $m^* > m$ is a bound on the number of steps taken.

Now define

$$Z_i = \sum_{j=1}^{k} I_{[U_{(i-1)k+j}]} \ , i=1,\ldots,(2n-4)/k$$

and observe that $[(maxZ_i)\ge k]$ implies that a random path has a horizontal step of size at least k. Therefore $\text{Prob}(\Delta_n < k) \le \text{Prob}(\max Z_i < k) \le \text{Prob}(\max[Z_i, i\le m^*/k] < k)$, the last step because max increases. Since $[Z_i<k]$ only if $[U_{(i-1)k+j} > p_{(i-1)k+j}]$ for some j, we get from (10)

$$\text{Prob}(\Delta_n \le k) \le \left(1-\left[\frac{1}{2}\left(1-\frac{1+m^*}{2n-5-m^*}\right)\right]^k\right)^{\frac{m^*}{k}}$$

$$\le \exp\left(-\frac{m^*}{k}2^{-k}\left(1-\frac{1+m^*}{2n-5-m^*}\right)^k\right) \le \exp\left(-\frac{m^*}{k}2^{-k}\left(1-\frac{k(1+m^*)}{2n-5-m^*}\right)\right).$$

Now just take $m^* = n/(\log n)^{c/2}$ and $k = \log n - (1+c)\log\log n$ and observe that the right hand side $\to 0$. ∎

Proof of Lemma 2: First observe that there are $C_{k-1}C_{i-k-1}C_{n-i-1}$ triangulations in T_n containing $\triangle v_0v_kv_i$, $i>k$. To count the number of triangulations with $\lambda_n = k$, suppose v_0v_k is a diagonal in τ (its kength is k) and that v_i is the apex of its triangle. If $n-k \le i \le 2k$ neither $\|v_0v_i\|$ nor $\|v_iv_k\|$ exceeds k. So we sum the products $C_{k-1}C_{i-k-1}C_{n-i-1}$ over i, from $n-k$ to $2k$, and multiply this sum by n to reflect the fact that the longest diagonal could as well be $v_1v_{k+1}, v_2v_{k+2}, \ldots$. Finally the * in $\Sigma_{(*)}$ in (4) means "multiply the summand by 1/2 when $i=2k$ and when $i=n-k$ (these triangles have two edges of length k and would be counted twice), unless $3k=n$, when we multiply by 1/3". This counts each good triangulation only once. ∎

Proof of Theorem 2: The first observation is that sum (disregarding the meaning of (*)) in (4) has the closed form

$$\sum_{i=n-k}^{2k} C_{i-k-1}C_{n-i-1} = \frac{(n-2k)(3k+1-n)}{(n-k)(n-k-1)(2n-4k-1)}\binom{2k}{k}\binom{2n-4k}{n-2k},$$

which can be verified easily. Multiply this sum by nC_{k-1}/C_{n-2}, approximate $\binom{2m}{m}$ by $4^m/\sqrt{\pi m}$ and observe that $w(x)$ is the limit as k and $n \to \infty$, with $k/n \to x \in (1/3, 1/2)$. The use of Σ in place of $\Sigma_{(*)}$ has no effect on this analysis. The constant α arises from direct evaluation of

$$\int_{1/3}^{1/2} x w(x) dx.$$

It is also possible to compute higher moments exactly. ∎

4 Algorithms

The proof of Theorem 1 has the germ of a fast algorithm for generating a random element of T_n. Uniform means "generate a uniform $[0,1]$ random number".

Algorithm 1 Rand-Tri($\mathbf{v_0}, \ldots, \mathbf{v_{n-1}}$).

1. $\mathbf{m} \leftarrow \mathbf{0}$; $(\mathbf{X_m}, \mathbf{Y_m}) \leftarrow (\mathbf{1}, \mathbf{0})$
2. $\mathbf{p_{m+1}} \leftarrow [(\mathbf{X_m + 2 - Y_m})(\mathbf{n - 2 - X_m})]/[(\mathbf{X_m - 1 - Y_m})(\mathbf{2n - 4 - X_m - Y_m})]$
3. $\mathbf{U_{m+1}} \leftarrow$ **uniform;** $\mathbf{X_{m+1}} \leftarrow \mathbf{X_m} + \mathbf{I}_{[\mathbf{U_{m+1}} \le \mathbf{p_{m+1}}]}$
   $\mathbf{Y_{m+1}} \leftarrow \mathbf{Y_m} + (\mathbf{1} - \mathbf{I}_{[\mathbf{U_{m+1}} \le \mathbf{p_{m+1}}]})$
4. $\mathbf{m} \leftarrow \mathbf{m + 1}$; **IF** $\mathbf{m < 2n - 5}$, $\rightarrow \mathbf{2}$
5. **decode path**

The algorithm is correct. Step 2, given that $(X_m, Y_m) = (i, j)$, computes p_{m+1} as in (9). Therefore (X_{m+1}, Y_{m+1}) is either $(X_m + 1, Y_m)$ or $(X_m, Y_m + 1)$, each with the correct probability. By the remark immediately following (9), this gives a path in P_n. By induction each path has equal probability.
The algorithm is $O(n)$. Since each p_m is computed in constant time the $2n - 5$ steps take linear time. It is straightforward to obtain - in linear time - the triangulation that corresponds to the path just generated. This is similar to the method of Arnold and Sleep [1] that is mentioned in [5]. A different approach is used in [2].

Suppose we take $m = 0$ and $(X_0, Y_0) = (k + 1, 0)$, and begin with Step 2. If $k \ge (n-3)/2$ the path corresponds to a random triangulation with $\Delta_n = k$ and whose max-degree vertex is v_0. Decoding the path and rotating the polygon a random number of vertices gives a random triangulation with max-degree k, each one being equally likely.

The proofs of Lemma 2 and Theorem 2 suggest an approach for fast uniform generation of triangulations with $\lambda = k$: given $v_0 v_k$ as the diagonal of max length $(= k)$, generate the apex v_i of its triangle, with $i \in [n - k, 2k]$ chosen according to the correct probability. The counting argument used for Lemma 2 shows that v_i should be chosen with probability S_i/S_{2k}, where $S_m = \sum_{j=n-k}^{m} C_{j-k-1} C_{n-j-1}$. Finally, use Algorithm 1 to randomly triangulate the polygons defined by $(v_0, \ldots, v_k)$, $(v_k, \ldots, v_i)$, and $(v_i, \ldots, v_0)$, and then rotate so edge $v_j v_{j+k}$ has probability $1/n$ to be the longest.

Algorithm 2 Random-max-diag$(\mathbf{k}; \mathbf{v_0}, \ldots, \mathbf{v_{n-1}})$.

1. $\mathbf{U} \leftarrow$ **uniform**
2. $\mathbf{i} \leftarrow \min(\mathbf{j} : \mathbf{S_j} \geq \mathbf{U} * \mathbf{S})$
3. **Rand-Tri**$(\mathbf{v_0}, \ldots, \mathbf{v_k})$; **Rand-Tri**$(\mathbf{v_k}, \ldots, \mathbf{v_i})$; **Rand-Tri**$(\mathbf{v_i}, \ldots, \mathbf{v_0})$.
4. **decode and rotate polygon randomly**

The algorithm to uniformly generate a triangulation that has k ears will be described in the final paper.

5 Final Remarks

This paper studied the behaviour of two properties - Δ_n, the maximal degree and λ_n, the length of the longest diagonal - in a random triangulation of a convex $n-$gon. Δ_n and λ_n correspond to interesting features of binary trees and our results on triangulations give new information about random trees. Some other points are:

1. Following the idea in Remark 1, as $n \to \infty$

$$\text{Prob}(d_i = k) \to (k+1)2^{-(k+2)},$$

the distribution of G_2, the sum of two independent geometric(1/2) random variables. It is interesting to wonder whether Δ_n converges in distribution to the limit distribution of the maximum of n independent copies of G_2.

2. According to Remark 2, $E(d_i) = 2(1 - 3/n)$ for uniform triangulations but $E_\beta(d_0) = E_\beta(d_{n-1}) = \Theta(\log n)$ in search-tree probability, a real difference between the two distributions. At the same time however, $E(\Delta_n / \log n) \to 1$ in the uniform case in and to $c > 1$ in the other. Some known properties of random binary trees in the two distributions are:

   (a) *height* h_n, the maximal depth of a node. $E(h_n)$ is asymptotic to $2\sqrt{\pi n}$ in the uniform case [8] and concentrated about $4.31107 \ln n$ in the other [4].

   (b) *leaves*. In the limit one expects $n/4$ leaves for uniform trees and $n/3$ for binary search trees (see [9]).

3. We note that it is easy to generate random triangulations in the search-tree probability. When constructing the triangulation as on page 1, just choose v_i uniformly from $1, \ldots, n-2$, etc. The complexity is $O(n)$.

4. An outstanding question concerns the set $T(K)$ of triangulations of a set K of n points not necessarily in convex position. As opposed to (0), it is not known how to count or even to approximate $|T(K)|$.

Acknowledgment: We thank Imre Bárány, Alistair Sinclair, and Jean-Marc Fedou for valuable conversations.

References

[1] D. Arnold and M. Sleep. Uniform Random Number Generation of n Balanced Parenthesis Strings. *ACM Trans. on Prog. Languages and Systems 2*,122-128, 1980.

[2] M. Atkinson and J. Sack. Generating Binary Trees at Random. *Inf. Proc. Letters 41*, 21-23, 1992.

[3] N. Dershowitz and S. Zaks. Patterns in Trees. *Disctere Applied Math. 25*, 241-255, 1989.

[4] L. Devroye. A Note on the Height of Binary Search Trees. *J.A.C.M. 33*,489-498, 1986.

[5] L. Devroye. *Non-Uniform Random Variate Generation*, Springer-Verlag, New York, 1986.

[6] L. Devroye. Application of the Theory of Records in the Study of Random Trees. *Acta Informatica 26*,123-130, 1988.

[7] W. Feller. *An Introduction to the Theory of Probability and Its Applications, Vol. 1, Third Ed.*, John Wiley, 1968.

[8] P. Flajolet and A. Odlyzko. The Average Height of Binary Trees and Other Simple Trees. *J.Comp. Syst. Sciences,25*, 171-213, 1982.

[9] P. Flajolet and R. Sedgewick. *An Introduction to the Analysis of Algorithms.* Addison-Wesley, Reading, Mass. 1996.

[10] F. Hurtado and M. Noy. The Graph of Triangulations of a Convex Polygon. *ACM Symposium on Computational Geometry (to appear)*, 1996.

[11] H. Mahmoud. *Evolution of Random Search Trees.* John Wiley, 1992.

[12] A. Meir and J. Moon. On the Altitude of Nodes in Random Trees. *Canad. J. Math. 30*,997-1015, 1978.

[13] B. Pittel. On Growing Random Binary Trees. *J. Math. Anal. Appl. 103*, 461-480, 1984.

[14] D. Sleator, R. Tarjan, and W. Thurston. Rotation Distance, Triangulations, and Hyperbolic Geometry. *J. Amer. Math. Soc. 1*, 647-681, 1988.