RANDOM TREES IN RANDOM GRAPHS
E. A. BENDER AND N. C. WORMALD

(Communicated by Andrew Odlyzko)

ABSTRACT. We show that a random labeled n-vertex graph almost surely contains isomorphic copies of almost all labeled n-vertex trees, in two senses. In the first sense, the probability of each edge occurring in the graph diminishes as n increases, and the set of trees referred to as “almost all” depends on the random graph. In the other sense, the probability of an edge occurring is fixed, and the relevant set of trees is predetermined. The same method applies to show that almost all labeled n-tournaments contain isomorphic copies of almost all labeled oriented n-trees.

1. Introduction. Sumner [9] conjectured that every tournament on 2n — 2 vertices contains every oriented tree on n vertices. This was verified in [8] for some types of oriented trees and some types of tournaments, and in [10] the weaker result was established, in which the tournaments have n log_2(2n/e) vertices. In this note we modify the nature of the question by considering random tournaments. An easy result is that almost all tournaments with (2 + ε)n vertices contain all oriented n-trees for any ε > 0 (see Lemma 4.2(v)).

Our arguments are more easily presented when applied to graphs and unoriented trees rather than to tournaments and oriented trees, so we rarely mention tournaments directly. All our graphs are labeled unless otherwise specified. We shall be considering the probability spaces r_n of n-vertex trees (n-trees) and r_n,p of n-graphs. In r_n the probability distribution is uniform, whilst in r_n,p the probability of an m-edged graph is p^m q^{n(n-1)/2-m}. Here and henceforth, q denotes 1 — p. A random element of r_n is denoted by T, and of r_n,p by G. Notation such as ~ and —> denotes the passage of n to infinity, unless otherwise specified, and our convention is that s denotes an arbitrarily small positive constant, perhaps different at each occurrence. V(G) or V denotes the vertex set of G, N(v) denotes the set of vertices adjacent to v ∈ V(G), and N(R) denotes ∪_{v ∈ R} N(v) for R ⊆ V(G).

Preliminary results of independent interest concerning the numbers of branches of T of given isomorphism class are established in §2. In particular, we show that in a sense almost all vertices of a random tree are nonroot vertices of small branches (Lemma 2.3(ii)). In §3 we examine the question of when G almost surely (a.s.) contains an isomorphic copy of T. In their pioneering work on random graphs, Erdős and Rényi [2] showed amongst other things if p = c log n/n with c < 1, then G is a.s. disconnected. On the other hand, with c > 1 Korshunov [6] proved that G a.s. contains a Hamilton cycle (see also Komlós and Szemerédi [5]).
implies that \( G \) contains a very special spanning tree: a Hamilton path. We show in §3 that for \( c > 2e^2 \) the same result holds when the random tree \( T \) determines the isomorphism class of the spanning tree. It follows that for such \( c \), the number of labeled \( n \)-trees embeddable in \( G \) is a.s. asymptotic to \( n^{n-2} \). In §4 this result is modified to the extent that a set \( \tau^* \) of labeled trees is specified for which \( T \in \tau^* \) a.s. and \( G \) a.s. contains copies of all \( n \)-trees in \( \tau^* \). This is at the expense of "increasing" \( p \) to become constant.

2. Random trees. We first establish simple properties of almost all trees, for use in §§3 and 4. Recall Cayley's formula, \( |r_n| = n^{n-2} \). By "\( D_1 \sim D_2 \) for a.e. \( T \)" (respectively, "\( D_1 \leq D_2 \) for a.e. \( T \)") we mean that for all \( \varepsilon > 0 \), \( \Pr\{|D_1/D_2 - 1| < \varepsilon\} \to 1 \) (resp. \( \Pr\{D_1/D_2 - 1 \leq \varepsilon\} \to 1 \)). The maximum degree of a vertex in \( T \) is denoted by \( \Delta(T) \). Moon [7, Corollary 1] proved

**Lemma 2.1.** \( \Delta(T) \sim \log n / \log \log n \) for a.e. \( T \).

A branch of a tree \( T \) is one of the two components formed when an edge of \( T \) is deleted. A branch is regarded as being rooted at the vertex incident with the deleted edge. If \( R \) is a rooted tree, an \( R \)-branch of \( T \) is a branch of \( T \) for which there is some isomorphism from that branch to \( R \) which preserves the linear ordering of the vertex labels, and preserves the root-vertex. Given \( R \), the number of \( R \)-branches of \( T \) is a random variable, denoted by \( B(T,R) \).

**Lemma 2.2.** If \( R \) is a rooted \( j \)-tree \((j \geq 1)\), then \( B(T,R) \sim ne^{-j}/j! \) for a.e. \( T \).

**Proof.** We use \( E \) to denote expectation and \( \text{Var} \) for variance. To construct each \( n \)-tree once for each of its \( R \)-branches, use an edge to attach the root-vertex of \( R \) to the root-vertex of a rooted \((n-j)\)-tree and choose the labels of \( R \) from \( 1, \ldots, n \). Thus

\[
E(B(T,R)) = \binom{n}{j} \frac{(n-j)^{n-j-1}}{n^{n-2}} \sim ne^{-j}/j!.
\]

Similarly,

\[
\binom{n}{j} \binom{n-j}{j} (n-2j)^{n-2j}
\]

counts each \( n \)-tree once for each ordered pair of disjoint \( R \)-branches. Moreover, any two \( R \)-branches are disjoint for \( n \geq 2j \). Hence

\[
E(B(T,R)^2) = \binom{n}{j} \binom{n-j}{j} \frac{(n-2j)^{n-2j}}{n^{n-2}} + E(B(T,R)) \sim \frac{n^2e^{-2j}}{j!^2}.
\]

Thus \( \text{Var}(B(T,R)) = o(E(B(T,R)^2)) \), and so the lemma follows from applying Chebyshev's inequality

\[
\Pr\{|X - E(X)| \geq \varepsilon E(X)\} \leq \frac{\text{Var}(X)}{\varepsilon^2 E(X)^2}
\]

to \( B(T,R) = X \).
For $k \geq 2$, let $B_k(T)$ denote the number of vertices $v \in V(T)$ such that $v$ is in some branch of $T$ of order at most $k$, but not the root-vertex of that branch. Also let

$$S_k = \sum_{j=2}^{k} \frac{2(j-1)j^{j-2}e^{-j}}{j!}.$$ 

**Lemma 2.3.** (i) If $k$ is fixed, then $B_k(T) \sim nS_k$ for a.e. $T$.

(ii) If $k \to \infty$ as $n \to \infty$, then $B_k(T) \sim n$ for a.e. $T$.

**Proof.** A near-root vertex in a rooted tree is one adjacent to the root vertex. The expected degree of the root vertex in a random rooted $j$-tree is twice the number of edges divided by the number of vertices, or $2(j-1)/j$. Hence the total number of near-root vertices amongst all $j^{j-1}$ rooted $j$-trees is $2(j-1)j^{j-2}$. If $Y_j$ denotes the number of near-root vertices amongst all the branches of $T$ of order $j$ (fixed), Lemma 2.2 applied to all rooted $j$-trees $R$ now gives $Y_j \sim 2n(j-1)j^{j-2}e^{-j}/j!$ for a.e. $T$. For $n$ at least $2k$, each vertex counted in $B_k(T)$ is a near-root vertex in a unique branch of order at most $k$, so (i) follows.

Let $L(x)$ be the generating function for rooted, labeled trees. Since $L = xe^L$, $L' = L/x + LL'$, and so

$$L^2 = 2 \int L' - L/x \, dx = \sum_{j=2}^{\infty} \frac{2(j-1)j^{j-2}x^j}{j!}.$$ 

Hence

$$\lim_{k \to \infty} S_k = L^2(1/e) = 1.$$ 

This implies (ii).

### 3. Random trees in random sparse graphs

In the following theorem, the probability refers to the independent random choice of $G \in \gamma_{n,p}$ and $T \in \tau_n$. We assume $c$ is constant.

**Theorem 3.1.** For $p = c \log n/n$ where $c > 2e^2$,

$$\Pr\{U \subset G \text{ for some } U \equiv T\} \to 1.$$ 

**Proof.** Let $P_2$ denote a labeled path of length 1 rooted at the vertex labeled 1. By Lemmas 2.1 and 2.2 we may assume $\Delta(T) < (1 + \varepsilon)\log n/\log \log n$ and $|B(T, P_2) - n/2e^2| < \varepsilon n$. A suitable value for $\varepsilon$ will be chosen. Let $T^*$ be the tree obtained from $T$ by removing all vertices of degree 1 which lie in $P_2$-branches. Put $n - |V(T^*)| = wn > n/2e^2 - \varepsilon n$. We embark on the following plan for finding an embedding $f$ of $T^*$ into $G$, i.e., an adjacency-preserving isomorphism $f: V(T^*) \to V(G)$. Firstly, put $f(v) = u$ for any vertex $v$ of $T^*$ not of degree 1 and any vertex $u$ of $G$. Then repeatedly choose a vertex $v$ of $T^*$ which has already been embedded and embed all the remaining unembedded neighbors of $v$. If the number of such neighbors is no more than the number of available vertices of $G$ adjacent to $f(v)$, the step can be carried out. Otherwise, the plan fails. Since the edges of $G$ occur independently at random, the probability of success in any step depends only on the degree, say $d$, of the vertex being embedded and on the number, say $a$, of vertices already embedded: it is

$$\sum_{k=r}^{n-a} \binom{n-a}{k} p^k q^{n-a-k}.$$
where \( r = d \) for the first step and \( r = d - 1 \) for all subsequent steps. Thus the probability of failure in any given step is at most

\[
\sum_{k=0}^{d-1} \binom{wn}{k} p^k q^{wn-k} \quad \text{(as } n - a \geq wn) \\
\leq \exp[(wn - d) \log(wnq/(wn - d)) + d \log(wnp/d)]
\]

(by the Chernoff bound—see Erdös and Spencer [5, p. 18])

\[
\leq \exp[-wn \log n + O(\log n \log \log n/\log \log n)]
\]

(using \( \log q < -p \), which can be verified using Taylor’s Theorem)

\[
< n^{-1-\epsilon} \quad \text{(provided } \epsilon < (c/2e^2 - 1)/(c + 1)).
\]

Hence, the probability that the embedding of \( T^* \) in \( G \) can be found is asymptotic to 1. To complete the embedding of \( T \) we need to find a set of independent edges which match the set \( R \) of the remaining \( wn \) unused vertices of \( G \) with the set \( S \) of images of the \( wn \) root vertices of \( P_2 \)-branches of \( T \). Note that the process of embedding \( T^* \) does not consider edges joining these two sets, so such edges still occur independently with probability \( p \). Theorem 3 of Erdös and Rényi [3] implies that if the entries of a random \( n \times n \) \((0,1)\)-matrix \( M \) are independently chosen each with probability \( (1 + \epsilon) \log n/n \) of becoming 1, then the bipartite graph whose adjacency matrix is \( M \) a.s. contains a set of \( n \) independent edges. Hence the embedding of \( T \) can a.s. be completed. The same method yields a similar result for embedding oriented trees in digraphs.

4. Random graphs containing a set of almost all trees. In this section we consider \( G \in \gamma_{n,p} \) with \( p \) fixed, \( 0 < p < 1 \). For \( K \) and \( L \subset V \) we say \( K \) matches into \( L \) if there is a set of independent edges (i.e. a matching) of \( G \) joining each vertex in \( K \) to a vertex in \( L \). The following result is well known.

**Lemma 4.1** (Phillip Hall’s Marriage Theorem). If \( K, L \subset V \) with \( K \cap L \) empty, and \(|N(J) \cap L| \geq |J| \) for each \( J \subset K \), then \( K \) matches into \( L \).

For \( c > 0 \), a \( c \)-large set is a subset of \( V \) of cardinality at least \( cn \). If \( c \) is unimportant, we speak of a \( large \) set. An interval set is a subset of \( V \) containing just those vertices whose labels are in some closed interval \([i,j]\). The fact that there are few interval sets—at most \((n^2+1)/2\)—is important.

**Lemma 4.2.** Suppose \( c, \epsilon > 0 \) are fixed. Then \( G \) satisfies each of the following a.s.

(i) For all \( v \in V \) and all \( c \)-large interval sets \( X \), \(|N(v) \cap X| > |X|(p - \epsilon)\).

(ii) For all \( c \)-large sets \( W \) and all \( X \subset V \) with \(|X| \geq \lceil \log n \rceil \), \( N(X) \cap W \) is nonempty.

(iii) For all \( c \)-large disjoint sets \( X \) and \( W \), with \( W \) an interval set and \(|W| = |X| + \lceil \log n \rceil \), \( X \) matches into \( W \).

(iv) For all \( c \)-large disjoint interval sets \( X \) and \( W \) and all \( X' \subset X \) with \(|X'| = |W| \geq |X| - \lceil \log n \rceil \), \( X' \) matches into \( W \).

(v) For all \( c \)-large interval sets \( X \), every tree on at most \((p - \epsilon)|X| \) vertices is contained in the subgraph of \( G \) induced by \( X \).
PROOF. The probability that (i) fails is at most
\[ n \sum \binom{x}{j} p^j q^{x-j-1}, \]
where the sum is over all interval sets \( X \) with \( x = |X| \geq cn \), and over all \( j \leq x(p-\varepsilon) \). This gives the upper bound \( n^3 e^{cn} \) where \( c_1 < 1 \). Thus (i) holds a.s.

To establish (ii), note that if \( |W| > cn \) then the probability of failure is at most
\[ 2nq^{\log n^2}, \]
which tends to 0.

To show (iii), we may assume (i) and (ii). Let \( J \subseteq X \). Then \( |N(J) \cap W| \) is large by (i). Also, if \( J \) is large, then \( |V - N(J)| < \log n \) by (ii), and so \( |N(J) \cap W| > |W| - |\log n| = |X| \). Thus by Lemma 4.1, \( X \) matches into \( W \).

For (iv), let \( J \subseteq X' \). As with (iii), \( |N(J) \cap W| \geq |J| \) when \( |J| \leq |W| - \log n \). By Lemma 4.1 we are done if a.s. \( |N(J) \cap W| \geq |J| \) whenever \( |W| - |J| < \log n \). The probability that this fails for some \( X, W \) and \( J \) is, for large \( n \), at most \( n^A XX''(n)2q^{\log n^2} \) where the sum is over all \( i = |W| - |J| < \log n \) and \( c_1 \) is some positive constant. The factor \( q^{\log n^2} \) comes from the probability that \( N(J) \cap L \) is empty for some \( (i+1)\)-subset \( L \) of \( W \). The result follows.

The truth of (v) follows from (i), which implies that each vertex in \( X \) has degree at least \( (p - \varepsilon)|X| \) in \( X \). Thus the simple plan, for embedding a tree in a graph, given in Theorem 3.1, succeeds for a.e. \( G \) if the tree has at most \( (p - \varepsilon)|X| \) vertices.

This completes the proof of the lemma.

We now give a strengthened version of Theorem 3.1 for \( p \) fixed.

**Theorem 4.3.** There is a set \( \tau^* \) of trees such that \( T \) is a.s. in \( \tau^* \) and \( G \) a.s. contains isomorphic copies of all trees in \( \tau^* \).

**Proof.** In some of our statements we assume that \( n \) is sufficiently large. Choose \( \delta < p/12e^3 \). Then by Lemma 2.3 that there are finite \( k \) and positive \( \varepsilon \) such that \( S_k(1-\varepsilon) > 1 - \delta \). Denote the rooted trees with at least 2 and at most \( k \) vertices by \( R_1, \ldots, R_m \), where \( m = \sum_{j=2}^{k} j^{j-1} \). For each \( R_i \) we can calculate from Lemma 2.2 the expected number, say \( b_i \), of \( R_i \)-branches of \( T \) not contained in any other branch with at most \( k \) vertices. Such \( R_i \)-branches are called pure. Note that \( b_i \) is at least \( cn \) for some constant \( c \), because there is some rooted tree \( R \) with an \( R_i \)-branch contained in no bigger branch of \( R \) with at most \( k \) vertices, and at least \( cn \) \( R_i \)-branches occur a.s. in \( T \) by Lemma 2.2.

Let \( \tau^* \) denote the set of trees \( T \) with at least \( [b_i(1-\varepsilon)] \) pure \( R_i \)-branches for each \( i = 1, \ldots, m \). By Lemma 2.2, \( T \in \tau^* \) a.s. In the rest of the proof, we consider a fixed \( n \)-tree \( T \in \tau^* \). For each \( i \), choose a set \( H_i \) of \( [b_i(1-\varepsilon)] \) pure \( R_i \)-branches of \( T \), and put \( H = \bigcup H_i \). Let \( T^- \) denote the tree obtained from \( T \) by removing all nonroot vertices in all branches in \( H \). Define disjoint interval subsets \( L_1, \ldots, L_z \) of the vertices \( \{1, \ldots, n\} \) in \( V \), one \( L_i \) for each nonroot vertex of each of the \( R_j \), subject to the following constraints. If \( L_i \) corresponds to a vertex \( u \) of \( R_j \) then \( |L_i| = [b_j(1-\varepsilon)] \), unless \( u \) is adjacent to the root-vertex of \( R_j \), in which case \( |L_i| = [b_j(1-\varepsilon)] + [\log n] \). Also \( V - \bigcup L_i \), which we will call \( K \), must be an interval set.

We may speak of each nonroot vertex in an \( R_i \)-branch in \( H \) as corresponding to a unique \( L_i \). Roughly speaking, we will attempt to find a copy of a tree \( T \) in \( G \) so that vertices in the \( L_i \) are used as the vertices in \( R_j \)-branches in \( H \) to which they
correspond. Between two sets $L_i$ and from $K$ to the $L_i$, the part of $T$ to be found is a matching. There are two main modifications to this: the vertices adjacent to root-vertices of branches in $H$ will not quite use up all of their corresponding $L_i$, and some of $L_1$ will be used, together with $K$, to find a copy of $T^-$. Let $R_1$ denote one of the rooted labeled paths of length 3 in which the ends are labeled 1 and 4, with 1 as the root-vertex. We can assume that $L_1$ corresponds to the vertex 4 of $R_1$, and that $L_2$ is assigned to the vertex adjacent to root-vertices of branches in $H$.

Note that $|K| \sim n(1 - S_k(1 - \varepsilon))$, $|L_1| \sim (1 - \varepsilon)n/6e^3$ and $\delta n < np/12e^3$. If necessary, decrease the size of $\varepsilon$ so that $|L_1| > n/8e^3$ ($\varepsilon < \frac{1}{4}$ would do). Let $C \subseteq L_1$ with $|C| = [|[K]/p|]$, such that $C$ and $C \cup K$ are interval sets (if necessary rearranging $K$, $L_1$, ..., $L_2$). All the sets defined so far are large, as is $L_1 - C$. We now choose a fixed $n$-graph $G$ which satisfies all the properties (i) to (v) stated in Lemma 4.2 (for suitable $\varepsilon$ and suitable constants in the references to large sets). In the rest of the proof we refer to these properties of $G$ simply as (i) to (v). Since $|V(T^-)| \sim |K|$, (v) gives that $G$ contains a copy of $T^-$ with vertices in $C \cup K$. Denote $V(T^-)$ at this point by $L_0$. The completion of $T^-$ to a copy of $T$ is performed in three stages. In the first stage, we treat an $L_i$ which corresponds to a vertex adjacent to the root-vertex of an $R_1$. Denote the vertices of $T^-$ which come from root-vertices of $R_1$-branches in $H$ by $\overline{L}_i$. Then $|\overline{L}_i| = [b_j(1 - \varepsilon)] = |L_i| - [\log n]$ and so by (iii), $\overline{L}_i$ matches into $L_i$. All $[\log n]$ vertices of $L_i$ unused by this matching are thrown into a set $D$, and we update $T^-$ by adding to it the edges in the matching. This also adds to $T^-$ the vertices of $T$ corresponding to $L_i$. This step is repeated for each $i$ as above, each time adding to $D$. When all such $i$ have been treated we move to the second stage. Here we consider $L_i$ and $L_l$ corresponding to adjacent vertices in an $R_1$, with $\{i, l\} \neq \{1, 2\}$. If $L_i$ or $L_l$ was treated in stage 1 then the difference between their cardinalities is $[\log n]$; otherwise they are equal. Thus by (iv) one set matches appropriately into the other, involving all vertices except perhaps $[\log n]$ in $D$. This allows us to update $T^-$ by adding the edges joining vertices of $T$ corresponding to $L_i$ and $L_l$. When all such $i$ and $l$ have been treated, we pass to the third stage. Here we find in $G$ the vertices of $T$ corresponding to $L_1$ (although we use vertices other than those in $L_1$ to do it). For this we need to show that $L_2$ matches into $D \cup L_1 \cup K - L_0$. Note that these two sets must have the same cardinality since $|V(G)| = |V(T)| = n$. First, choose $F \subseteq L_2$ with $|F| = |D \cup C \cup K - L_0| + [\log n]$ such that $F$ and $L_2 - F$ are interval sets. Then $F$ is large and so by (iii), $D \cup C \cup K - L_0$ matches into a subset, say $L_2^-$, of $L_2$. Let $L_2^+ = L_2 - L_2^-$. Then all that is left is to show that $L_2^+$ matches into $L_1 - C$. Since the latter is an interval set and both are large, (i) and (ii) imply $|N(W) \cap (L_1 - C)| \geq |W|$ for each $W \subseteq L_2^+$ with $|W| \leq |L_2^+| - [\log n]$ (as in the proof of Lemma 4.2(iii)). Since $L_2^+$ contains a large interval set, this also holds for all other $|W| \leq |L_2^+|$ by (i) applied separately to each vertex in $L_1 - C$. This completes the embedding of $T$ in $G$, and the proof.

Clearly, for random labeled tournaments, in which arcs are oriented randomly with probability $\frac{1}{2}$ for each direction, the same method yields the following. (Recall that an oriented tree is a tree in which each edge has been oriented.)

**Theorem 4.4.** There is a set $\tau^*$ containing almost all oriented labeled trees, such that a random labeled $n$-tournament almost surely contains at least one isomorphic copy of all oriented $n$-trees in $\tau^*$. 

It seems likely that this theorem remains true when "a random" is replaced by "every" and "almost surely" is deleted.

REFERENCES

9. D. Sumner, private communication with K. B. Reid.