Permutations in Binary Trees and Split Trees

Michael Albert
Department of Computer Science, Otago University, New Zealand
malbert@cs.otago.ac.nz
 https://orcid.org/0000-0002-4587-1104

Cecilia Holmgren
Department of Mathematics, Uppsala University, Uppsala, Sweden
cecilia.holmgren@math.uu.se
 https://orcid.org/0000-0003-0717-4671

Tony Johansson
Department of Mathematics, Uppsala University, Uppsala, Sweden
tony.johansson@math.uu.se

Fiona Skerman
Department of Mathematics, Uppsala University, Uppsala, Sweden
fiona.skerman@math.uu.se
 https://orcid.org/0000-0003-4141-7059

──── **Abstract** ────

We investigate the number of permutations that occur in random node labellings of trees. This is a generalisation of the number of subpermutations occuring in a random permutation. It also generalises some recent results on the number of inversions in randomly labelled trees [3]. We consider complete binary trees as well as random split trees a large class of random trees of logarithmic height introduced by Devroye [4]. Split trees consist of nodes (bags) which can contain balls and are generated by a random trickle down process of balls through the nodes.

For complete binary trees we show that asymptotically the cumulants of the number of occurrences of a fixed permutation in the random node labelling have explicit formulas. Our other main theorem is to show that for a random split tree with high probability the cumulants of the number of occurrences are asymptotically an explicit parameter of the split tree. For the proof of the second theorem we show some results on the number of embeddings of digraphs into split trees which may be of independent interest.

1 Introduction and statement of results

Our main results are Theorem 2 on the distribution of the number of appearances of a fixed permutation in a random labelling of a complete binary tree and Theorem 4 which shows that for a random split tree with high probability (whp) the same result holds for the number of appearances of a fixed permutation in a random labelling of the balls of the tree. We write a complete introduction and statement of results in terms of complete binary trees first before defining split trees and stating our results for split trees.

Complete Binary trees

Let V_n denote the node set of the complete binary tree T_n of height m and $n = 2^{m+1} - 1$ nodes. Define a partial ordering on the nodes of the tree by saying that $a < b$ if a is an ancestor of b. Choose a uniform random labelling of the nodes $\pi : V_n \to [n]$.

We say that nodes a and b form an inversion if $a < b$ and $\pi(a) > \pi(b)$. The (random) number of inversions in random node labellings of fixed trees as well as some random models of trees were studied in a recent paper ([3]). This paper finds approximate extensions to some of these results.

The (random) number of inverted triples is $R(321, T) = \sum_{u_1 < u_2 < u_3} \mathbf{1}[\pi(u_1) > \pi(u_2) > \pi(u_3)]$ where the sum runs over all triples of nodes in T such that u_1 is an ancestor of u_2 and u_2 an ancestor of u_3. In general, we say a permutation σ appears on the $|\sigma|$-tuple of vertices $u_1, \dots, u_{|\sigma|}$, if $u_1 < \dots < u_{|\sigma|}$ and the induced order on $\pi(u) = (\pi(u_1), \dots, \pi(u_{|\sigma|}))$ is σ. Write $\pi(u) \approx \sigma$ to indicate the induced order is the same for example $527 \approx 312$. Define

$$R(\sigma, T) \stackrel{\text{def}}{=} \sum_{u_1 < \dots < u_{|\sigma|}} \mathbf{1}[\pi(u) \approx \sigma],$$

so in particular $R(21, T)$ counts the number of inversions in a random labelling of T.

We will generally be concerned with the centralised moments, e.g., $\mathbb{E}\left[(R(\sigma, T) - \mathbb{E}\left[R(\sigma, T)\right])^r\right]$. Let $d(v)$ denote the *depth* of v, i.e., the distance from v to the root ρ. For any $u_1 < \dots < u_{|\sigma|}$ we have $\mathbb{P}[\pi(u) = \sigma] = 1/|\sigma|!$ and so it immediately follows that,

$$\mathbb{E}\left[R(\sigma, T)\right] = \sum_{u_1 < \dots < u_{|\sigma|}} \mathbb{E}\left[\pi(u) = \sigma\right] = \frac{1}{|\sigma|!} \sum_v \binom{d(v)}{|\sigma| - 1}. \tag{1}$$

For length two permutations, e.g. inversions, $\mathbb{E}\left[R(21, T)\right] = \frac{1}{2}\Upsilon(T)$ where $\Upsilon(T) \stackrel{\text{def}}{=} \sum_v d(v)$ is called the *total path length* of T. We state our results in terms of a tree parameter $\Upsilon_r^k(T)$ which generalises the notion of total path length.

We define $\Upsilon_r^k(T)$ which allows us to generalize (1) to higher moments of $R(\sigma, T)$. For r nodes $v_1, \dots, v_r$ (not necessarily distinct), let $c(v_1, \dots, v_r)$ be the number of ancestors that they share $c(v_1, \dots, v_r) \stackrel{\text{def}}{=} |\{u \in V : u \le v_1, v_2, \dots, v_r\}|$ which is also the depth of the least common ancestor plus one. That is $c(v_1, \dots, v_r) = d(v_1 \vee \dots \vee v_r) + 1$ where we write $d(v)$ for the depth of v and $v_1 \vee v_2$ for the least common ancestor of v_1 and v_2. The 'off by one error' is because the root is in the set of common ancestors for any subsets of nodes but we use the convention the root has depth 0. Also define

$$\Upsilon_r^k(T) \stackrel{\text{def}}{=} \sum_{v_1, \dots, v_r} c(v_1, \dots, v_r) \prod_{i=1}^r \binom{d(v_i)}{k - 2}, \tag{2}$$

where the sum is over all ordered r-tuples of nodes in the tree and with the convention $\binom{x}{0} = 1$. For a single node v, $d(v) = c(v) - 1$, since v itself is counted in $c(v)$. So $\Upsilon(T) = \Upsilon_1^2(T) - |V|$; i.e., we recover the usual notion of total path length. The $k = 2$ case recovers the r-total common ancestors defined in [3], $\Upsilon_r^2(T) = \sum_{v_1, \dots, v_r} c(v_1, \dots, v_r)$.

Indeed the distribution of the number of permutations in a fixed tree has already been studied in [3]. Let $\varkappa_r = \varkappa_r(X)$ denote the r-th cumulant of a random variable X (provided it exists); thus $\varkappa_1(X) = \mathbb{E}\left[X\right]$ and $\varkappa_2(X) = \text{Var}\left(X\right)$.

▶ **Theorem 1** (Thm 1 of Cai et al. [3]). *Let T be a fixed tree. Let $\varkappa_r = \varkappa_r(R(21, T))$ be the r-th cumulant of $R(21, T)$. Then for $r \geq 2$,*

$$\varkappa_r = \frac{B_r(-1)^r}{r}\left(\Upsilon_r^2(T) - |V|\right)$$

where B_r denotes the r-th Bernoulli number.

For the case of T a complete binary tree on n vertices we asymptotically recover this result for large n. Moreover we extend it to cover any fixed permutation σ for complete binary trees.

▶ **Remark.** In essence Theorem 1 of [3] shows the r-th cumulant of the number of inversions is a constant times $\Upsilon_r^2(T)$. Our main result on fixed trees, Theorem 2 (resp. Theorem 4 on split trees), shows that for any fixed permutation σ of length k for complete binary trees (and whp for split trees) the r-th cumulant is a constant times $\Upsilon_r^k(T_n)$ asymptotically. The exact constant is defined below and is a little more involved than for inversions but observe it is a function only of the moment r and the length of $k = |\sigma|$ together with the first element σ_1 of the permutation $\sigma = \sigma_1 \ldots \sigma_k$. With some work one can show $D_{12,r} = B_r(-1)^r/r$ and so Theorem 2 does asymptotically recover Theorem 1 for complete binary trees.

We now state our first main result.

▶ **Theorem 2.** *Let T_n be the complete binary tree of depth n and fix a permutation $\sigma = \sigma_1 \ldots \sigma_k$ of length k. Let $\varkappa_r = \varkappa_r(R(\sigma, T_n))$ be the r-th cumulant of $R(\sigma, T_n)$. Then for $r \geq 2$,*

$$\varkappa_r = D_{\sigma,r}\Upsilon_r^k(T_n) + o\left(\Upsilon_r^k(T_n)\right) \tag{3}$$

where

$$D_{\sigma,r} \stackrel{\text{def}}{=} \sum_{j=0}^{r}\left(\frac{-1}{k!}\right)^{r-j}\binom{r}{j}\frac{\left(j(\sigma_1 - 1)\right)!\left(j(k - \sigma_1 - 1)\right)!}{\left(j(k-1)+1\right)!\left((\sigma_1 - 1)!(k-\sigma_1)!\right)^j}. \tag{4}$$

This implies the following corollary.

▶ **Corollary 3.** *Let T_n be the complete binary tree of depth n. For permutations σ of length 3,*

$$\mathbb{V}(R(\sigma, T_n)) = \begin{cases} \frac{1}{45}\Upsilon_2^3(T_n)(1 + o(1)) & \text{for } \sigma = 123, 132, 312, 321 \\ \frac{1}{180}\Upsilon_2^3(T_n)(1 + o(1)) & \text{for } \sigma = 213, 231 \end{cases}$$

and more generally for $\sigma = \sigma_1\sigma_2 \ldots \sigma_k$,

$$\mathbb{V}(R(\sigma, T_n)) = \begin{cases} \frac{1}{((k-1)!)^2}\left(\frac{1}{2k-1} - \frac{1}{k^2}\right)\Upsilon_2^k(1 + o(1)) & \text{for } \sigma_1 \in \{1, k\} \\ \left(\frac{1}{(2k-1)(k-\sigma_1)!(k+\sigma_1-2)!} - \frac{1}{(k!)^2}\right)\Upsilon_2^k(1 + o(1)) & . \end{cases}$$

▶ **Remark.** The methods of proof are very different for inversions and general permutations. In [3], the method takes advantage of a nice independence property of permutations. For a node u let I_u be the number of inversions involving u as the top node: $I_u = |\{w : u < w, \pi(u) > \pi(w)\}|$. Then the $\{I_u\}_u$ are independent random variables and I_u is distributed as the uniform distribution on $\{0, \ldots, |T_u|\}$, see Lemma 1 of [3].

Without an obvious similar independence property for general permutations our route instead uses nice properties on the number of embeddings of small digraphs in both binary trees and, whp, in split trees. This property allows us to calculate the centralised r-th

moment of $R(\sigma, T)$ directly from a sum of products of indicator variables as most terms in the sum are zero or negligible by the embedding property. The centralised r-th moment is then approximately a function of the j-th cumulants for $j \leq r$ and we are able to deduce the r-th cumulant by induction.

We now define a particular notion of embedding small digraphs into a tree which will be important as discussed in the previous remark.

In the complete binary tree we have a natural partial order, the ancestor relation, where the root is the ancestor of all other nodes. Any fixed acyclic digraph also induces a partial order on its vertices where $v > u$ if there is a directed path from v to u. Define $[\vec{H}]_{T_n}$ to be the number of embeddings ι of $\vec{H}$ to distinct nodes in T_n such that the partial order of vertices in $\vec{H}$ is respected by the embedding to nodes in T_n under the ancestor relation.

$$[\vec{H}]_{T_n} \stackrel{\text{def}}{=} |\{\iota : V(\vec{H}) \to V(T_n) \text{ such that if } u < v \text{ in } \vec{H} \text{ then } \iota(u) < \iota(v) \text{ in } T_n\}|$$

Observe the inverse of embedding ι^{-1} need not respect relations. If $u \perp v$ in $\vec{H}$, i.e. u, v are incomparable in $\vec{H}$ then we can embed so that $\iota(u) < \iota(v)$, $\iota(u) > \iota(v)$ or $\iota(u) = \iota(v)$ in T_n. For an example of this take the digraph ⬦ and denote by P_ℓ the rooted path on ℓ nodes. Notice that in ⬦ two of the vertices are incomparable but the vertices of the digraph can be embedded into the nodes of a path which are completely ordered. The counts are $[⬦]_{P_4} = 2$ and in general $[⬦]_{P_\ell} = 2\binom{\ell}{4}$.

A particular star-like digraph $\vec{S}_{k,r}$ will be important. This is the digraph obtained by taking r directed paths of length k and fusing their source vertices into a single vertex. Alternatively we can state the theorem in terms of star counts as $[\vec{S}_{|\sigma|,r}]_{T_n} = \Upsilon_r^{|\sigma|}(T_n)(1 + o(1))$. See the beginning of the proof of the theorem for details.

Split trees

Split trees were first defined in [4] and were introduced to encompass many families of trees that are frequently used in algorithm analysis, e.g., binary search trees [6], m-ary search trees [8] and quad trees [5].

The random split tree T_n has parameters $b, s, s_0, s_1, \mathcal{V}$ and n. The integers b, s, s_0, s_1 are required to satisfy the inequalities

$$2 \leq b, \quad 0 < s, \quad 0 \leq s_0 \leq s, \quad 0 \leq bs_1 \leq s + 1 - s_0. \tag{5}$$

and $\mathcal{V} = (V_1, \ldots, V_b)$ is a random non-negative vector with $\sum_{i=1}^b V_i = 1$.

We may now define the random split tree as follows. Consider an infinite b-ary tree $\mathcal{U}$. The split tree T_n is constructed by distributing n balls (pieces of information) among nodes of $\mathcal{U}$. For a node u, let n_u be the number of balls stored in the subtree rooted at u. Once n_u are all decided, we take T_n to be the largest subtree of $\mathcal{U}$ such that $n_u > 0$ for all $u \in T_n$. Let $\mathcal{V}_u = (V_{u,1}, \ldots, V_{u,b})$ be the independent copy of $\mathcal{V}$ assigned to u. Let $u_1, \ldots, u_b$ be the child nodes of u. Conditioning on n_u and $\mathcal{V}_u$, if $n_u \leq s$, then $n_{u_i} = 0$ for all i; if $n_u > s$, then

$$(n_{u_1}, \ldots, n_{u_b}) \sim \text{Mult}(n - s_0 - bs_1, V_{u,1}, \ldots, V_{u,b}) + (s_1, s_1, \ldots, s_1),$$

where Mult denotes multinomial distribution, and b, s, s_0, s_1 are integers satisfying (5). Note that $\sum_{i=1}^b n_{u_i} \leq n$ (hence the "splitting"). Naturally for the root ρ, $n_\rho = n$. Thus the distribution of $(n_u, \mathcal{V}_u)_{u \in V(\mathcal{U})}$ is completely defined. For this paper we will also require that the internal node capacity s_0 is at least one so that there are some internal balls to receive labels.

This next theorem is our other main result.
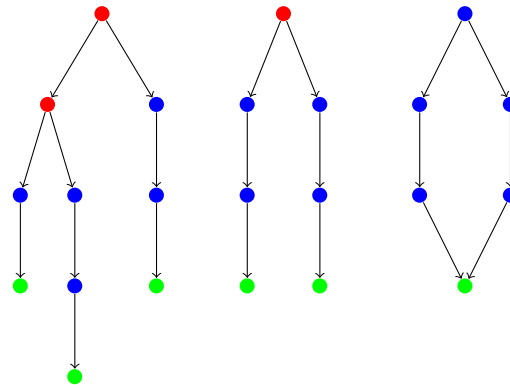
Figure 1 An example of a directed acyclic graph $\vec{H}$ with 'sink' (green), 'ancestor' (blue) and 'common-ancestor' (red) nodes indicated by colour. This particular digraph is in $\mathcal{G}_{4,7}$ and it appears in the seventh moment calculations of $R(\sigma, T)$ for $|\sigma| = 4$.

▶ **Theorem 4.** *Fix a permutation $\sigma = \sigma_1 \ldots \sigma_k$ of length k. Let T_n be a split tree with split vector $\mathcal{V} = (V_1, \ldots, V_b)$ and n balls. Let $\varkappa_r = \varkappa_r(R(\sigma, T_n))$ be the r-th cumulant of $R(\sigma, T_n)$. For $r \geq 2$ the constant $D_{\sigma,r}$ is defined in line (4). Whp the split tree T_n has the following property.*

$$\varkappa_r = D_{\sigma,r} \Upsilon_r^k(T_n) + o\big(\Upsilon_r^k(T_n)\big).$$

Our theorem says the following. Generate a random split tree T_n, whp it has the property that the random number of occurrences of any fixed subpermutation in a random ball labelling of T_n has variance and higher cumulant moments approximately a constant times a 'simple' tree parameter of T_n.

We may contrast this with Theorem 4 of [3]. This theorem states the distribution of the number of inversions in a random split tree; where the distribution is expressed as the solution of a system of fixed point equations. It is work in progress to find the distribution of $\Upsilon_r^k(T_n)$. This would extend Theorem 4 of [3] about inversions to general permutations.

2 Embeddings of small digraphs into the complete binary tree

Certain classes of digraphs will be important in the proof of Theorem 2, loosely those that may be obtained by taking r copies of the path $\vec{P}_k$ and iteratively fusing pairs of vertices together. It will also matter how many embeddings each digraph has into the complete binary tree. In Proposition 9 we show the counts for most digraphs in such a class are dwarfed by the counts of a particular digraph in the class. The main work in the proof of this proposition is to show that the number of embeddings of any digraph $\vec{H}$, up to a factor of n, depends only on the numbers of two types of vertices in $\vec{H}$. We separate this result out as a lemma, Lemma 5, which we show first before proving the proposition.

A vertex in a directed graph is a *sink* if it has zero out-degree. For a directed acyclic graph $\vec{H}$ we define $A_i \subseteq V(\vec{H})$ to be the vertices with exactly i descendents in $\vec{H}$ which are sinks. In particular A_0 is the set of sink vertices. We will call vertices in A_1 *ancestors* as they are ancestors of a single sink and those in A_i for $i \geq 2$ *common-ancestors* as they are the common ancestor of at least two sinks (see Figure 1). Observe if $\vec{H}$ is a directed forest then the sinks are the leaves but a sink may have indegree more than one as in the rightmost sink in Figure 1

The next lemma shows that the numbers of sinks and ancestors in $\vec{H}$ determine the number of ways to map $\vec{H}$ into the complete binary tree T_n on n vertices to within a factor of $\ln n$.

▶ **Lemma 5.** *Let $\vec{H}$ be a fixed directed acyclic graph and let T_n be the complete binary tree of height m with $n = 2^{m+1} - 1$ vertices. Then writing $|A_0| = |A_0(\vec{H})|$ for the number of sink (green) vertices and $|A_1| = |A_1(\vec{H})|$ for the number of ancestor (blue) vertices*

$$\Omega(n^{|A_0|}(\ln n)^{|A_1|}) = [\vec{H}]_{T_n} = o(n^{|A_0|}(\ln n)^{|A_1|+1}).$$

Proof of upper bound. The key observation is that for most pairs of nodes in T_n their least common ancestor is very near the root. Let the nodes at depth d be $w_1, \ldots, w_{2^d}$. Fix a node u in the tree. Provided the depth of node u is at least d, i.e. $h(u) \geq d$ then if $c(u, v) \geq d$ it must be that u and v are in the same subtree T_{w_i} for some i. If $h(u) \geq d$ let $w(u)$ be the node at depth d which is either node u itself or an ancestor of u. Thus

$$\begin{aligned}
\sum_{u,v} \mathbf{1}[c(u,v) \geq d] &\leq \sum_v \mathbf{1}[v \in T_{w(u)})] \sum_u \mathbf{1}[d(u) \leq d] \\
&\leq 2(m + 2^{m-d+1} - 1)(2^{m+1} - 1) \\
&\leq 2^{2m-d+2+1} + m2^{m+1} + 2^{2d+2} \\
&= n^2 2^{-d+3} + mn
\end{aligned} \tag{6}$$

Fix $\epsilon > 0$ such that $|A_2|\epsilon < 1/2$. Let B be the set of $|A_0|$-tuples of vertices so that some pair of them have an ancestor at depth $> n^\epsilon$. By (6) the set is B is small: $|B| \leq |A_0|^2 n^{|A_0|} \cdot 2^{-n^\epsilon}$.

Given an embedding of A_0 into T_n the number of ways to extend an embedding of $\vec{H}$ into T_n is at most $m^{|A_1|+|A_2|}$. This is because each vertex in $A_1 \cup A_2$ must be embedded as an ancestor of the embedding of a vertex in A_0 and each vertex in T_n has at most m ancestors. And in particular, if A_0 is embedded to a $|A_0|$-tuple not in B there are at most $m^{|A_1|+\epsilon|A_2|}$ ways to extend to an embedding of $\vec{H}$. Thus

$$[\vec{H}]_T \leq n^{|A_0|} m^{|A_1|+\epsilon|A_2|} + n^{|A_0|-\epsilon} m^{|A_1|+|A_2|} = o(n^{|A_0|}(\ln n)^{|A_1|+1}),$$

where the second inequality follows because $m = \Theta(\ln n)$. ◀

Proof of lower bound. We restrict attention to embeddings where all common-ancestors are embedded very near the root of T_n, the sink vertices are embedded to leaves of T_n and the ancestor vertices are placed on the path between the root of T_n and the leaf in to which their descendent sink was embedded (see Figure 2). There are sufficiently many such embeddings to obtain the lower bound. In fact we restrict a little further to make it easy to check all the embeddings are valid.

By an abuse in notation denote by A_2 the union $\cup_{i \geq 2} A_i$. As $\vec{H}$ is an acyclic digraph the directed edges define a partial order on all vertices of $\vec{H}$ and in particular for those in A_2. Thus this relation can be extended to a total order. Fix some total order $<_*$ on $V(\vec{H})$ and relabel vertices in A_2 so that $v_1 <_* \ldots <_* v_{|A_2|}$. Thus we may embed v_1 to the root ρ in T_n and each v_{i+1} to a child of the node to which v_i was embedded and the relation between vertices in $\vec{H}$ will be preserved by their embedding in T_n; i.e. we may embed A_2 to the nodes on the path from ρ to some u^* at depth $|A_2| - 1$. Fix such a node u^* and let T^* be the subtree of T_n from u^*.

Label the sinks $A_0 = \{s_1, \ldots, s_{|A_0|}\}$ and vertices in A_1 according to which sink they are the ancestors of $A_1^i \overset{\text{def}}{=} \{v \in A_1 \ : \ v < s_i\}$.

We obtain a subcount of $[\vec{H}]_{T_n}$ by embedding A_2 onto the path from ρ to u^*, embedding A_0 to leaves of T^* and then for each i in turn embedding vertices in A_i on the path from u^*
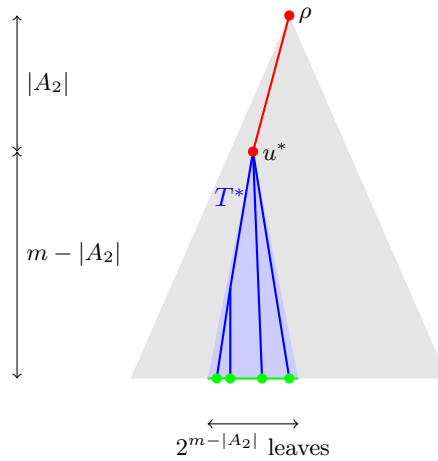
Figure 2 Schematic for the lower bound construction in Lemma 5. The colours indicate the positions in the binary tree to which the common-ancestor (red), ancestor (blue) and sink (green) vertices are embedded. Recall $A_2 = A_2(\vec{H})$ denotes the set of common-ancestor vertices of $\vec{H}$.

to the embedding of s_i. There are $m - |A_2| - 1$ vertices on the path from s_i to u^* and at most $|A_1|$ of them already have an ancestor vertex embedded onto to them (i.e. from A_1^j for some $j < i$). Thus

$$[\vec{H}]_{T_n} \geq \binom{2^{m-|A_2|}}{|A_0|} \prod_i \binom{m - |A_2| - |A_1| - 1}{|A_1^i|}$$

where the first binomial counts the number of ways to embed A_0 and the i-th binomial in the product counts the ways to embed A_1^i. Now because $\vec{H}$ is fixed $|A_2| = O(1)$ and the product over i is at least $\binom{m-|A_2|-1}{|A_1|}$ so the lower bound follows. ◄

3 Embeddings of small digraphs into the split trees

In this section we show upper and lower bounds on the number of embeddings of a fixed digraph $\vec{H}$, thought of as constant, into a random split tree with n balls. We begin by briefly listing some results on split trees from the literature that will be useful for us.

For split vector $\mathcal{V}$ define $\mu = \sum_i \mathbb{E}[V_i \ln V_i]$. The average depth of a ball is $\sim \frac{1}{\mu} \ln n$ [7][Cor 1.1]. Moreover almost all balls are very close to this depth. Define a ball v to be *good* if it has depth

$$\left| d(v) - \frac{1}{\mu} \ln n \right| \leq \ln^{0.6} n$$

and then whp $n - o(n)$ of the balls in the split tree are good [2][Thm 1.2]. That whp in a split tree all good balls have a $\Theta(n)$ depth and almost all balls are good is the only result about split trees required for the proof of the lower bound on $[\vec{H}]_{T_n}$ in Lemma 8. For the upper bound we need a bit more.

It is known that the height of a split tree with split vector $\mathcal{V}$ is whp $(c + o(1)) \ln n$ for a (known) constant c; for details see [1][Thm 2]. We write T_u to denote the subtree from bag (node) u and $|T_u|$ the number of balls in the subtree.

▶ **Lemma 6.** *Fix k. Let U be the set of bags at depth $\lfloor \alpha \ln \ln n \rfloor$ for some large enough constant $\alpha = \alpha(k)$. Then whp*

$$\sum_{u \in U} |T_u|^2 = o\left(\frac{n^2}{(\ln n)^k}\right).$$

We omit the proof of the lemma but note that it follows the same steps as Lemma 3.5 of [2].

Similarly for binary trees we show that the number of embeddings of a fixed acyclic digraph $\vec{H}$, to a good approximation, depends only on the number of 'sink' and 'ancestor' vertices in $\vec{H}$. It is a little trickier to prove the corresponding statement to the upper bound Lemma 5 in the case of split trees. However, we are rewarded by a tighter bound on the number of embeddings is determined by the numbers of 'sink' and 'ancestor' vertices up to $\ln \ln n$ factors.

▶ **Lemma 7.** *Let $\vec{H}$ be a fixed directed acyclic graph and let T_n be a split tree with split vector $\mathcal{V}$ and n balls. Then writing $|A_0| = |A_0(\vec{H})|$ for the number of sink (green) vertices, $|A_1| = |A_1(\vec{H})|$ for the number of ancestor (blue) vertices and $|A_2| = |A_2(\vec{H})|$ for the number of common-ancestor (red) vertices whp*

$$[\vec{H}]_{T_n} = O(n^{|A_0|}(\ln n)^{|A_1|}(\ln \ln n)^{|A_2|}).$$

Proof. The idea of the proof is to show that any way of embedding $A_0(\vec{H})$ into the tree can only be extended to an embedding of all the vertices in $\vec{H}$ in a limited number of ways. Note

$$[\vec{H}]_{T_n} = \sum_{\mathbf{v} = v_1, \ldots, v_{|A_0|}} f(\mathbf{v}) \tag{7}$$

where $f(\mathbf{v})$ is the number of ways to extend an embedding of $A_0(\vec{H})$ to an embedding $V(\vec{H}) \to V(T_n)$. Formally label the vertices in $A_0(\vec{H})$ by $s_1, \ldots, s_{|A_0|}$ and define

$$f(\mathbf{v}) \quad \stackrel{\text{def}}{=} \quad |\{\iota : \iota(s_j) = v_j \text{ for each } j = 1, \ldots, |A_0| \text{ and}$$
$$\iota : V(\vec{H}) \to V(T_n) \text{ such that if } u < v \text{ in } \vec{H} \text{ then } \iota(u) < \iota(v) \text{ in } T_n\}|.$$

We claim first that for any $\mathbf{v}$, whp $f(\mathbf{v}) = O\big((\ln n)^{|A_1| + |A_2|}\big)$ and indeed will later show a stronger bound holds for most $\mathbf{v}$.

To see this first claim recall that whp the height of a split tree on n balls is $\Theta(\ln n)$. In particular the depth of each ball v_j is $O(\ln n)$ and so v_j has $O(\ln n)$ balls as ancestors. Each vertex in $A_1(\vec{H}) \cup A_2(\vec{H})$ must be embedded to a ball which is the ancestor of some v_j (and possibly further restricted to balls which are ancestors of some set of v_j's but we will not need this). Hence there are at most $O(\ln n)$ choices of where to embed each vertex in $A_1(\vec{H}) \cup A_2(\vec{H})$ which finishes the claim.

Similarly to the proof for the case of binary trees we now exploit the fact that in split trees most pairs of balls have their least common ancestor in a bag very near the root. This will allow us to define a large set of $\mathbf{v}$ for which $f(\mathbf{v})$ is small.

Say a tuple of balls $\mathbf{v}$ is *inbred* if some pair of balls has a common ancestor at depth greater than $L \stackrel{\text{def}}{=} \lfloor \alpha \ln \ln n \rfloor$ for some α such that Lemma 6 holds with $k = |A_2|$. Denote the set of these tuples by $\mathcal{I}$. We claim that whp

$$|\mathcal{I}| \leq |A_0|^2 n^2 (\ln n)^{-|A_2|}. \tag{8}$$

Before proving claim (8) let us show that it implies the theorem. If a tuple of balls is not inbred, $\mathbf{v} \notin \mathcal{I}$, then any ancestor of any pair of balls has depth at most $L = O(\ln \ln n)$. Thus whp there are at most $O(\ln \ln n)$ choices of where to embed each vertex in $A_2(\vec{H})$ when extending an embedding in which $A_0(\vec{H})$ was embedded to $\mathbf{v} \notin \mathcal{I}$. So for non inbred $\mathbf{v}$,

$$\max_{\mathbf{v} \notin \mathcal{I}} f(\mathbf{v}) = O((\ln n)^{|A_1|}(\ln \ln n)^{|A_2|}).$$

We are almost finished (modulo the claim). By (9) and recalling there are less than $n^{|A_0|}$ possible tuples of balls we get

$$[\vec{H}]_{T_n} = \sum_{\mathbf{v} \in \mathcal{I}} f(\mathbf{v}) + \sum_{\mathbf{v} \notin \mathcal{I}} f(\mathbf{v}) \leq |\mathcal{I}|O\big((\ln n)^{|A_1|+|A_2|}\big) + O(n^{|A_0|}(\ln n)^{|A_1|}(\ln \ln n)^{|A_2|}) \quad (9)$$

and so the claim $|\mathcal{I}| = O(n^{|A_0|}(\ln n)^{-|A_2|})$ does imply the theorem.

It now remains to prove the claim. Let $c(v_1, v_2)$ be the depth of the bag which is the least common ancestor of balls v_1 and v_2. To prove the claim it suffices to show

$$\sum_{v_1,v_2} \mathbf{1}[c(v_1, v_2) \geq L] \leq \frac{n^2}{(\ln n)^{|A_2|}}.$$

Trivially, if $c(v_1, v_2) \geq L$ then both v_1 and v_2 must be at depth at least L. Also notice if v_1 and v_2 have their least common ancestor at depth at least L they must have some common ancestor, u say, at depth exactly L. Let U be the set of bags at depth L. Then

$$\mathbf{1}\big[c(v_1, v_2) \geq L\big] = \mathbf{1}\big[v_1, v_2 \in T_u \text{ for some } u \in U\big]$$

and so we may apply Lemma 6 directly

$$\sum_{v_1,v_2} \mathbf{1}\big[c(v_1, v_2) \geq L\big] \ \leq \ \sum_u |T_u|^2 \ \leq \ \frac{n^2}{(\ln n)^{|A_2|}}$$

which establishes the claim. ◄

▶ **Lemma 8.** *Let $\vec{H}$ be a fixed directed acyclic graph and let T_n be a split tree with split vector $\mathcal{V} = \{V_1, \ldots, V_b\}$ and n balls. Then writing $|A_0| = |A_0(\vec{H})|$ for the number of sink (green) vertices and $|A_1| = |A_1(\vec{H})|$ for the number of ancestor (blue) vertices whp*

$$[\vec{H}]_{T_n} = \Omega(n^{|A_0|}(\ln n)^{|A_1|}).$$

Proof. *(sketch)* We describe a strategy to embed $\vec{H}$ into T_n. The details of the proof are then to show that whp this strategy can be followed to obtain a valid embedding of $\vec{H}$ and that there are sufficiently many different such embeddings to achieve the lower bound.

First embed 'common-ancestor' vertices along a path to some node u^* with $\tilde{n} = \Omega(n)$ balls. Now consider a split tree with $\tilde{n}$ balls and embed 'ancestor' and 'sink' vertices into that. Embed 'sink' vertices to 'good' balls in the tree (i.e. depth very close to the expected depth) and the 'ancestor' vertices to balls which along the path between u^* and the embedding of their descendent. See Figure 3.

We embed the common-ancestor vertices, $A_2(\vec{H})$, to the balls in the nodes on the path between a node, u^* say, at depth $|A_2| - 1$ and the root, using one ball per node. This is so far effectively the same as in the binary case. And we will later embed the 'sink' and
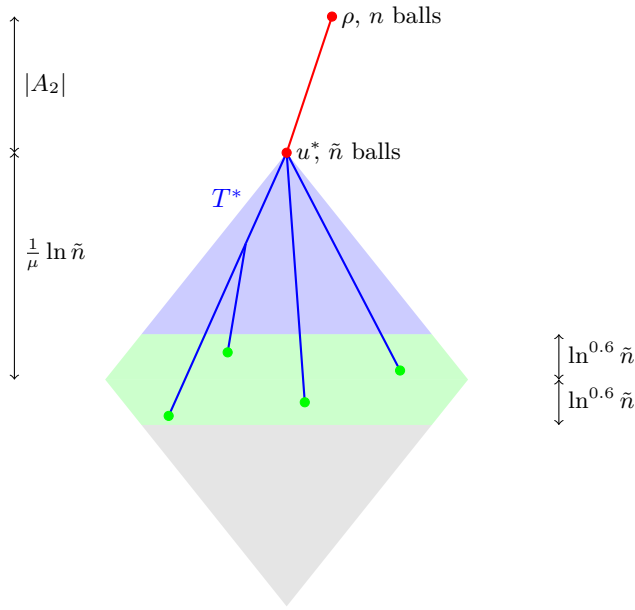
■ **Figure 3** Schematic for the construction in Lemma 8. The colours indicate the positions in the split tree to which the common-ancestor (red), ancestor (blue) and sink (green) vertices are embedded. Recall $A_2 = A_2(\vec{H})$ denotes the set of common-ancestor vertices of $\vec{H}$.

'common-ancestor' vertices to balls in the subtree T_{u^*}. We need to confirm there is some node u^* at depth $L = |A_2| - 1$ with $\tilde{n}$ balls in its subtree. Each node (bag) has capacity at most s_0 or s and at most $(b^{L+1} - 1)$ nodes, a constant number, at depth less than L, so $n - O(1)$ balls remaining. These balls are shared between b^L, a constant, number of subtrees T_u. Hence by pigeon-hole principle some vertex u^* has $\tilde{n} = \Theta(n)$ balls in its subtree.

Now work in the split tree $T_{\tilde{n}}$. Embed the 'sink' vertices to any good balls $v_1, \ldots, v_{|A_0|}$ in the split trees. There are $\Theta(\tilde{n}^{|A_0|})$ ways to embed them. Label the 'sink' vertices $s_1, \ldots, s_{|A_0|}$ and $A_1^j \subset A_1^j(\vec{H})$ to be the 'ancestor' vertices with s_j as their lone descendent. Vertices in A_1^j can be embedded to balls anywhere between v_j and u^* and so there are $\Theta((\ln \tilde{n})^{|A_1^j|})$ ways to do that for each j. All up there are $\Omega(\tilde{n}^{|A_0|}(\ln \tilde{n})^{|A_1|})$ ways to embed $A_0(\vec{H}) \cup A_1(\vec{H})$ into balls of $T_{\tilde{n}}$. But now as $\tilde{n} = \Theta(n)$ we are done.   ◄

4   Star counts

After having proved the required properties of our two classes of trees, binary trees and split trees, we show these imply the desired results on cumulants of the number of appearances of a permutation in the node labellings of binary trees, respectively ball labellings in split trees.

Say a sequence of trees T_n with n nodes (resp. balls) is *explosive* if for any fixed acyclic digraph $\vec{H}$

$$\Omega(n^{|A_0|}(\ln n)^{|A_1|}) = [\vec{H}]_{T_n} = o(n^{|A_0|}(\ln n)^{|A_1|+1}).$$

Thus Section 2 was devoted to showing binary trees are explosive and Section 3 to showing split trees are explosive whp. This section proves the cumulant results using only this explosive property of the tree classes.

Now we introduce some notation in order to state Proposition 9. We use a notion of subgraph on an ordered set of vertices. For a k-tuple of vertices $V_i = (v_i^1, \ldots, v_i^k)$
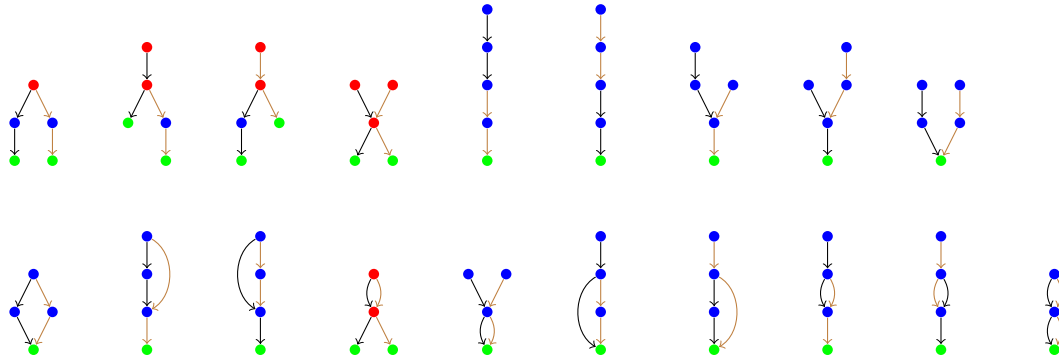
Figure 4 The set $\mathcal{G}'_{3,2}$. Labels of the first path $V_1 = (v_1^1, v_1^2, v_1^3)$ indicated by black arrows between the nodes and respectively brown arrows for labels of the second path $V_2 = (v_2^1, v_2^2, v_2^3)$. Colours of nodes indicate 'sink' (green), 'ancestor' (blue) and 'common-ancestor' (red) nodes respectively. These labelled directed acyclic graphs appear in variance calculations of $R(\sigma)$ for $|\sigma| = 3$.

we say $\vec{H}|_{V_i} = \vec{P}_k$ if the subgraph of $\vec{H}$ induced on V_i has precisely the directed edges $v_i^1 v_i^2, v_i^2 v_i^3, \ldots, v_i^{k-1} v_i^k$.

The set $\mathcal{G}_{k,r}$ is the set of acyclic digraphs which may be obtained by taking r copies of the path $\vec{P}_k$ and iteratively fusing pairs of vertices together such that each path is involved in at least one fusing operation. Likewise labelled $\vec{H}'$ in $\mathcal{G}'_{k,r}$ are those obtained by fusing together j labelled paths $\vec{P}_k$ keeping both sets of labels when a pair of vertices are fused. The set $\mathcal{G}'_{4,2}$ is illustrated in Figure 4.

Formally let $\mathcal{G}_{k,r}$ be the set of directed acyclic graphs $\vec{H}$ such that we can find (non-disjoint) vertex subsets $V_1, \ldots, V_r$ where for each i we have $\vec{H}|_{V_i} = \vec{P}_k$ and $\exists j \neq i$ with $V_i \cap V_j = \varnothing$. (The second condition is to ensure each i-th path is involved in a fusing operation.) For $\vec{H} \in \mathcal{G}_{k,r}$ write $\vec{H}'$ for $\vec{H}$ together with a labelling $V_1, \ldots, V_r$ (note some vertices have multiple labels). Likewise write $\mathcal{G}'_{k,r}$ for the labelled set of graphs.

Denote by $\vec{S}_{k,j}$ the digraph composed by taking j copies of the path $\vec{P}_k$ and fusing the j source vertices into a single vertex. Also define $\mathcal{S}^*_{k,r} = \cup_i \vec{S}_{k,r_i}$ where the disjoint union is over all S_{k,r_i} with $\sum_i r_i = r$ and $r_i \geq 2$. Observe $\mathcal{S}_{k,r} \subset \mathcal{G}_{k,r}$.

▶ **Proposition 9.** *Fix k, r and let $\vec{H} \in \mathcal{G}_{k,r}$. Suppose T_n is explosive. If $\vec{H} \notin \mathcal{S}_{k,r}$ then*

$$[\vec{H}]_{T_n} = o\Big([\vec{S}_{k,r}]_{T_n}\Big).$$

Proof. First observe that $\vec{S}_{k,r}$ has r sink vertices, $(k-2)r$ ancestor vertices and exactly one common-ancestor vertex. Thus by the explosive property of T_n

$$[\vec{S}_{k,r}]_{T_n} = \Omega(n^r (\ln n)^{(k-2)r}).$$

Fix $\vec{H} \in \mathcal{G}_{k,r} \backslash \mathcal{S}_{k,r}$ and fix a labelling $V_1, \ldots, V_r$ on $\vec{H}$. Again by the explosive property

$$[\vec{H}]_{T_n} = o(n^{|A_0(\vec{H})|}(\ln n)^{|A_1(\vec{H})|+1}). \tag{10}$$

Hence if $|A_0(\vec{H})| \leq r - 1$ then $[\vec{H}]_{T_n} = o([\vec{S}_{k,r}])$ and so we would be done. Thus we may assume that $A_0(\vec{H}) = r$ and it will suffice to show that $A_1(\vec{H}) < (k-2)r$. Consider the path labelled $V^i = (v_1^i, \ldots, v_k^i)$. We know v_k^i is a sink vertex and not fused with any other vertex otherwise we would have $A_0(\vec{H}) < r$. If vertex v_j^i is fused with another vertex, it must be a vertex on a different path to avoid a cycle, and so v_j^i and $v_{j-1}^i, \ldots, v_1^i$ would

become common-ancestors. Thus if v_j^i is fused to another vertex there are at most $(k-j-1)$ ancestor vertices in path V_i. Hence if $A_1(\vec{H}) = (k-2)r$ then we must have only fused the source vertices of each path but this means that $\vec{H} \in \mathcal{S}_{k,r}$ and so we are done.   ◀

By exploiting only the explosive property of binary and (whp) of split trees we prove the moments result for both classes at once. In particular observe that Theorems 2 and 4 are both implied by taking Proposition 10 along with the lemmas proving binary trees are explosive and split trees are whp explosive.

▶ **Proposition 10.** *Suppose T_n is explosive. Let $\varkappa_r = \varkappa_r(R(\sigma, T_n))$ be the r-th cumulant of $R(\sigma, T_n)$. Then for $r \geq 2$,*

$$\varkappa_r = D_{\sigma,r} \Upsilon_r^{|\sigma|}(T_n) + o(\Upsilon_r^{|\sigma|}(T_n)).$$

Proof sketch. The proof proceeds by induction on r with $r = 2$, the variance, as the base case. The variance calculation is also a simpler version of the calculations for higher r and so illustrates the key steps we use for the inductive step.

We give a rough idea of these steps. The variance (and higher centralised moments) can be written as a sum over indicator random variables for a subpermutation occuring on a set of $|\sigma|$ nodes. Almost all terms in this sum are zero or negligible. Firstly if the indicators concern disjoint sets of vertices they are independent and because we calculate centralised moments these terms drop away. This leaves only terms in the sum in which the nodes of indicator variables overlap. We group the terms by how the vertices in these sets overlap and the results about numbers of embeddings then show most groups are negligible.

For the variance only one group is non-negligible and so we will be done at this step. In the inductive step the centralised r-th moment has only one 'new' group (not occuring in smaller moment calculations) which is non-negligible as well as non-negligible groups which appeared in smaller cumulants for $j \leq r$. This occurs in such a way that we can prove this new group approximates the r-th cumulant.   ◀

References

1   Nicolas Broutin, Luc Devroye, and Erin McLeish. Weighted height of random trees. *Acta Informatica*, 45(4):237, 2008. `doi:10.1007/s00236-008-0069-0`.

2   Nicolas Broutin and Cecilia Holmgren. The total path length of split trees. *The Annals of Applied Probability*, 22(5):1745–1777, 2012. `doi:10.1214/11-aap812`.

3   Xing Shi Cai, Cecilia Holmgren, Svante Janson, Tony Johansson, and Fiona Skerman. Inversions in split trees and conditional Galton–Watson trees. *arXiv preprint arXiv:1709.00216*, 2017.

4   Luc Devroye. Universal limit laws for depths in random trees. *SIAM Journal on Computing*, 28(2):409–432, 1998. `doi:10.1137/s0097539795283954`.

5   R. Finkel and J. Bentley. Quad trees a data structure for retrieval on composite keys. *Acta informatica*, 4(1):1–9, 1974. `doi:10.1007/bf00288933`.

6   C. Hoare. Quicksort. *The Computer Journal*, 5(1):10–16, 1962. `doi:10.1145/366622.366644`.

7   C. Holmgren. Novel characteristics of split trees by use of renewal theory. *Electronic Journal of Probability*, 17, 2012. `doi:10.1214/ejp.v17-1723`.

8   R. Pyke. Spacings. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 395–449, 1965.