Linear-Time Calculation of the Expected Sum of Edge Lengths in Random Projective Linearizations of Trees

Lluís Alemany-Puig
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain
lluis.alemany.puig@upc.edu

Ramon Ferrer-i-Cancho
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain
rferrericancho@cs.upc.edu

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

The syntactic structure of a sentence is often represented using syntactic dependency trees. The sum of the distances between syntactically related words has been in the limelight for the past decades. Research on dependency distances led to the formulation of the principle of dependency distance minimization whereby words in sentences are ordered so as to minimize that sum. Numerous random baselines have been defined to carry out related quantitative studies on languages. The simplest random baseline is the expected value of the sum in unconstrained random permutations of the words in the sentence, namely, when all the shufflings of the words of a sentence are allowed and equally likely. Here we focus on a popular baseline: random projective permutations of the words of the sentence, that is, permutations where the syntactic dependency structure is projective, a formal constraint that sentences satisfy often in languages. Thus far, the expectation of the sum of dependency distances in random projective shufflings of a sentence has been estimated approximately with a Monte Carlo procedure whose cost is of the order of Rn, *where* n *is the number of words of the sentence and* R *is the number of samples; it is well known that the larger* R *is, the lower the error of the estimation but the larger the time cost. Here we present formulae to compute that expectation without error in time of the order of* n. *Furthermore, we show that star trees maximize it, and provide an algorithm to retrieve the trees that minimize it.*

1. Introduction

A successful way to represent the syntactic structure of a sentence is a dependency graph (Nivre 2006) that relates the words of a sentence by pairing them with syntactic links, as in Figure 1. Each link is directed and the arrow points from the **head word** to the **dependent word** (Figure 1). There are several conditions that are often imposed on

Action Editor: Giorgo Satta. Submission received: 7 July 2021; revised version received: 24 November 2021; accepted for publication: 18 January 2022.

https://doi.org/10.1162/coli_a_00442

© 2022 Association for Computational Linguistics
Published under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license

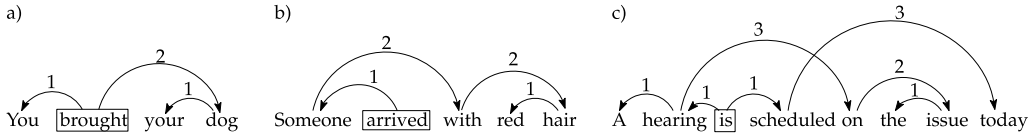

Figure 1
Examples of sentences and their syntactic dependency structure. Here arc labels indicate dependency distances. The word within a rectangle is the root of the sentence, and the number on top of each edge denotes its length. (a) Projective dependency tree (adapted from Groß and Osborne 2009). (b) Planar (but not projective) syntactic dependency structure (adapted from Groß and Osborne 2009). (c) Non-planar syntactic dependency structure (adapted from Nivre 2009).

the structure of dependency graphs (Nivre 2006). The first is **well-formedness**, namely, the graph is (weakly) connected. The second is **single-headedness**, that is, every word has at most one head. Another condition is **acyclicity**, that is, if two words, say w_i and w_j, are connected via following one or more directed links from w_i to w_j then there is no path of directed links from w_j to w_i. By definition, syntactic dependency trees always have a root vertex, that is, a vertex (word) with no head. The fourth condition is **projectivity**, often informally described as the situation where edges do not cross when drawn above the sentence and the root is not covered by any edge.

When a dependency graph is well-formed, single-headed, and acyclic, the graph is a directed tree, called a syntactic dependency tree (Kuhlmann and Nivre 2006; Gómez-Rodríguez, Carroll, and Weir 2011). In addition, a syntactic dependency structure is projective if, for every vertex v, all vertices reachable from v, that is, the yield of v, form a continuous substring within the linear ordering of the sentence (Kuhlmann and Nivre 2006). Equivalently, a syntactic dependency structure is projective if the yield of each vertex of the tree forms a contiguous interval of positions in the linear ordering of the vertices. Kuhlmann and Nivre (2006) define an interval (with endpoints i and j) as the set $[i, j] = \{k \mid i \leq k \text{ and } k \leq j\}$.

A linear arrangement of a graph is planar if it does not have edge crossings (Sleator and Temperley 1993; Kuhlmann and Nivre 2006). Then projectivity can be characterized as a combination of two properties: planarity and the fact that the root is not covered (Mel'čuk 1988). Planarity was, to the best of our knowledge, first thought of as **one-page embeddings** of trees by Bernhart and Kainen (1979). Figure 1 shows an example of a projective tree 1(a), a planar tree 1(b), and a non-planar tree 1(c) (see Bodirsky, Kuhlmann, and Möhl [2005] for further characterizations of syntactic dependency structures).

A free tree $T = (V, E)$ is an undirected acyclic graph (Figure 2(a)), where V is the set of vertices and E is the set of edges. Here we represent the syntactic dependency structure of a sentence as a pair consisting of a rooted tree and a linear arrangement of its vertices. A rooted tree $T^r = (V, E; r)$ is a free tree $T = (V, E)$ with one of its vertices, say $r \in V$, labeled as its root and with the edges oriented from r toward the leaves (Figure 2(b)). A linear arrangement π (also called **embedding**) of an n-vertex graph $G = (V, E)$ is a (bijective) function that assigns every vertex $u \in V$ to a position $\pi(u)$. Throughout this article, we use the terms "linear arrangement," "linear ordering," "arrangement," and "linearization" interchangeably. In addition, we assume that $\pi(u) \in [n] = \{1, \cdots, n\}$. Linear arrangements are often seen as determined by the labeling of the vertices (Chung 1984; Kuhlmann and Nivre 2006), but here we consider that the labeling of a graph and π are independent. In order to clarify our notion of linear arrangement of a labeled

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

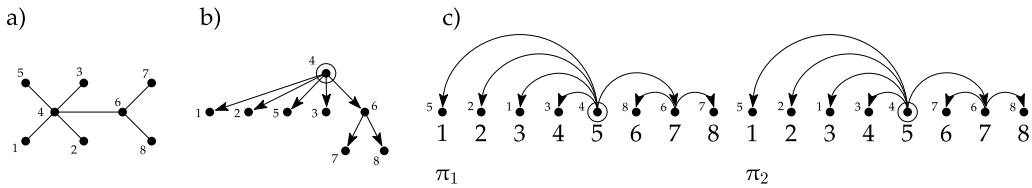

Figure 2
a) A free tree $T = (V, E)$. b) The tree T rooted at $r = 4$, yielding $T^r = (V, E; r)$ with $r = 4$. c) Two different projective linear arrangements of T^r: $\pi_1(5) = \pi_2(5) = 1$ (thus $\pi_1^{-1}(1) = \pi_2^{-1}(1) = 5$); $\pi_1(8) = 6$, $\pi_2(7) = 6$.

graph G, we say that two linear arrangements π_1 and π_2 of G are equal if and only if, for every vertex $u \in V$, it holds that $\pi_1(u) = \pi_2(u)$.

In any linear arrangement π of a graph $G = (V, E)$, one can define properties on the graph's edges and on the arrangement as a whole. The length of an edge between vertices u and v is their distance in the linear arrangement, usually defined as

$$\delta_{uv}(\pi) = |\pi(u) - \pi(v)| \tag{1}$$

Thus, the length of an edge in the arrangement is the number of vertices between its endpoints plus one, as in previous studies (Iordanskii 1987; Shiloach 1979; Chung 1984; Hochberg and Stallmann 2003; Ferrer-i-Cancho 2004; Gildea and Temperley 2007, 2010; Ferrer-i-Cancho 2019). A less commonly used definition of edge length is (Hudson 1995; Hiranuma 1999; Eppler 2005; Liu, Xu, and Liang 2017)

$$\delta^*_{uv}(\pi) = |\pi(u) - \pi(v)| - 1 \tag{2}$$

Here we use $D_\pi(G) = \sum_{uv \in E} \delta_{uv}(\pi)$ as the definition for the sum of edge lengths of G when it is linearly arranged by π, but we also derive some results for $D^*_\pi(G) = \sum_{uv \in E} \delta^*_{uv}(\pi)$.

There exists sizable literature on the calculation of baselines for the sum of edge lengths on trees. These baselines are crucial for research on the Dependency Distance Minimization (DDm) principle (Ferrer-i-Cancho 2004; Liu, Xu, and Liang 2017; Temperley and Gildea 2018). DDm was put forward by comparing actual dependency distances against a random baseline (Ferrer-i-Cancho 2004). Concerning the computation of the minimum baseline, Iordanskii (1987) and Hochberg and Stallmann (2003) independently devised an $O(n)$-time algorithm for planar (one-page) embedding of free trees. Gildea and Temperley (2007) sketched an algorithm for projective embeddings of rooted trees. Alemany-Puig, Esteban, and Ferrer-i-Cancho (2022) reviewed this problem and presented, to the best of our knowledge, the first $O(n)$-time algorithm. Polynomial-time algorithms for unconstrained embeddings were presented by Shiloach (1979), with complexity $O(n^{2.2})$, and later by Chung (1984), with complexities $O(n^2)$ and $O(n^\lambda)$, where λ is any real number satisfying $\lambda > \log 3/\log 2$. Concerning random baselines, the precursors are found in Zörnig's research on the distribution of the distance between repeats in a uniformly random arrangement of a sequence assuming that consecutive elements are at distance zero (Zörnig 1984) as in parallel research on syntactic dependency distances (Hudson 1995; Hiranuma 1999; Eppler 2005; Liu, Xu, and Liang 2017). Later,

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Ferrer-i-Cancho (2004, 2016) studied the expectation of the random variable $D(T)$ defined as

$$D(T) = \sum_{uv \in E} \delta_{uv} \tag{3}$$

in uniformly random arrangements, where δ_{uv} is a random variable defined over uniformly random unconstrained linear arrangements of the tree T, resulting in

$$\mathbb{E}[D(T)] = \frac{n^2 - 1}{3} \tag{4}$$

Notice that $\mathbb{E}[D(T)]$ does not depend on the topology of T.

While there are constant-time formulae for the expectation of $D(T)$ in unconstrained arrangements (Equation (4)), a procedure to calculate the expected value of $D(T)$ under projectivity is not forthcoming. Our primary goal is to improve the calculation of the expected sum of edge lengths in uniformly random projective arrangements with respect to the Monte Carlo method or random sampling method put forward by Gildea and Temperley (2007). Such a widely used procedure (Park and Levy 2009; Futrell, Mahowald, and Gibson 2015; Kramer 2021) estimates the expectation of D of an n-vertex tree with an error that is negatively correlated with R, the amount of arrangements sampled, while its cost is directly proportional to that amount, that is, $O(Rn)$. This raises the question of what the minimum value of R would be to obtain accurate-enough estimations of the expectation of $D(T)$ in projective arrangements. In recent research (Futrell, Mahowald, and Gibson 2015; Kramer 2021), $R = 10, R = 100$ were used. Here we demonstrate that there is no need to answer this question since we provide formulae to calculate its exact value.

We improve upon these techniques by providing closed-form formulae for the expected value of $D(T^r)$ in uniformly random projective arrangements of T^r that can be evaluated in $O(n)$-time. More formally, our goal in this article is to find closed-form formulae for $\mathbb{E}_{\text{pr}}[D(T^r)]$, the expectation of the random variable $D(T^r)$ conditioned to the set of projective arrangements, where the subscript "pr" indicates "projective linear arrangement." Notice that $\mathbb{E}[D(T)]$ in Equation (4) has no subscript to indicate **unconstrained linear arrangement**. An unconstrained linear arrangement is one of the $n!$ possible orderings. $\mathbb{E}_{\text{pr}}[D(T^r)]$ is a widely used random baseline for research on DDm (Park and Levy 2009; Gildea and Temperley 2010; Futrell, Mahowald, and Gibson 2015; Kramer 2021).

The structure of this article is as follows. We first derive, in Section 2, an arithmetic expression for $\mathbb{E}_{\text{pr}}[D(T^r)]$, given by

Theorem 1
Let $T^r = (V, E; r)$ be a tree rooted at $r \in V$. The expected sum of edge lengths $D(T^r)$ conditioned to uniformly random projective arrangements is

$$\mathbb{E}_{\text{pr}}[D(T^r)] = \frac{d_r(2n_r + 1) + n_r - 1}{6} + \sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[D(T^r_u)] \tag{5}$$

$$= \frac{1}{6}\left(-1 + \sum_{v \in V} n_v(2d_v + 1)\right) \tag{6}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

where n_u denotes the number of vertices of the subtree T^r_u rooted at $u \in V$, that is, $n_u = |V(T^r_u)|$, Γ_v denotes the set of children of vertex v, and $d_v = |\Gamma_v|$ is the out-degree of vertex v in the rooted tree. If $d_r = 0$ then $\mathbb{E}_{\text{pr}}[D(T^r)] = 0$.

Section 3 characterizes the class of trees that maximize $\mathbb{E}_{\text{pr}}[D(T^r)]$, detailed in Theorem 2.

Theorem 2
For any n-vertex rooted tree T^r, we have that $\mathbb{E}_{\text{pr}}[D(T^r)] \leq \mathbb{E}_{\text{pr}}[D(\mathcal{S}^h_n)]$ with equality if, and only if, $T^r = \mathcal{S}^h_n$, where $\mathcal{S}^h_n$ denotes the star tree of n vertices.

Then, a tight upper bound of $\mathbb{E}_{\text{pr}}[D(T^r)]$ is given by $\mathbb{E}[D(T)]$, as detailed in the next corollary.

Corollary 1
Given any n-vertex rooted tree $T^r = (V, E; r)$ rooted at $r \in V$, it holds that

$$\mathbb{E}_{\text{pr}}[D(T^r)] \leq \mathbb{E}_{\text{pr}}[D(\mathcal{S}^h_n)] = \mathbb{E}[D(T)] = \frac{n^2 - 1}{3} \tag{7}$$

where $\mathbb{E}[D(T)]$ is the expected sum of edge lengths in uniformly random (unconstrained) linear arrangements (Equation (4)) and T is the free tree variant of T^r.

Theorem 1 and Corollary 1 indicate that, for each n, a star tree rooted at its hub ($\mathcal{S}^h_n$) maximizes $\mathbb{E}_{\text{pr}}[D(T^r)]$, achieving $(n^2 - 1)/3$. Section 3 also shows that the minima can be calculated with a dynamic programming algorithm.

Section 4 compares our new method to calculate $\mathbb{E}_{\text{pr}}[D(T^r)]$ exactly against the Monte Carlo estimation method using dependency treebanks and finds that commonly used values of R can yield a large relative error in the estimation on a single tree. This new method is available in the Linear Arrangement Library (Alemany-Puig, Esteban, and Ferrer-i-Cancho 2021). We finally present some conclusions and propose future work in Section 5.

2. Expected Sum of Edge Lengths

We devote this section to characterize projective arrangements (Section 2.1) and to derive an arithmetic expression to calculate the sum of expected edge lengths in said arrangements (Section 2.3). We end this section with some instantiations of said expression for particular classes of trees (Section 2.4).

2.1 The Number of Random Projective Arrangements

The number of unconstrained arrangements of an n-vertex tree T is $\mathbf{N}(T) = |\mathbf{P}(T)| = n!$, where $\mathbf{P}(T)$ denotes the set of all $n!$ arrangements of T, hence $\mathbf{N}(T)$ is independent from the tree structure. The number of projective arrangements of a tree, however, depends on its structure, in particular on the out-degree sequence of the tree, as is shown later in this section. Counting the number of projective arrangements of a tree motivates a proper characterization that underpins the proof of Theorem 1. For this, we need to introduce some notation.

Henceforth we denote directed edges of a rooted tree $T^r = (V, E; r)$ as $uv = (u, v) \in E$; all edges are oriented toward the leaves. We denote the set of children of a vertex $v \in V$ as Γ_v, and thus the out-degree of v is $d_v = |\Gamma_v|$ in the rooted tree.

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

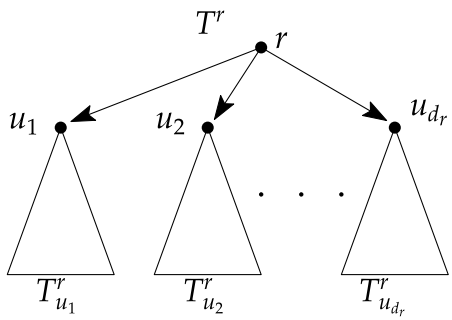

Figure 3
A tree T^r rooted at r. The children of the root are $\Gamma_r = \{u_1, u_2, \cdots, u_{d_r}\}$, where d_r is the out-degree of r. Each T^r_u, for $u \in \Gamma_r$, denotes the subtree of T^r rooted at u.

In particular, we refer to the root's children as $\Gamma_r = \{u_1, \cdots, u_{d_r}\} \subset V$. We denote the subtree of T^r rooted at $u \in V$ as T^r_u; we denote its size (in vertices) as $n_u = |V(T^r_u)|$; notice that $n_u \geq 1$. We say that T^r_v is an immediate subtree of T^r_u if uv is an edge of the tree. Figure 3 depicts a rooted tree and the immediate subtrees of T^r.

We provide a closed-form formula for the number of projective arrangements of a rooted tree, $\mathbf{N}_{\text{pr}}(T^r)$. This result helps us characterize said arrangements.

Proposition 1
Let $T^r = (V, E; r)$ be a tree rooted at $r \in V$.

$$\mathbf{N}_{\text{pr}}(T^r) = (d_r + 1)! \prod_{u \in \Gamma_r} \mathbf{N}_{\text{pr}}(T^r_u) \tag{8}$$

$$= \prod_{v \in V} (d_v + 1)! \tag{9}$$

where d_v is the out-degree of vertex v in the rooted tree. If $d_r = 0$, then $\mathbf{N}_{\text{pr}}(T^r) = 1$.

The fact that subtrees span over intervals (Kuhlmann and Nivre 2006) is central to the proof of Proposition 1. Because intervals are associated with a fixed pair of starting and ending positions in a linear sequence, we use the term **segment** of a rooted tree T^r_u to

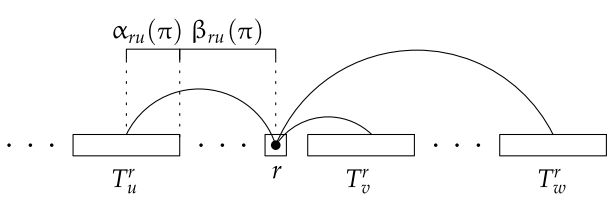

Figure 4
A permutation τ of the segments associated with r. Each rectangle represents the segment of the root r and those of the subtrees rooted at $u, v, w \in \Gamma_r$, denoted as T^r_u, T^r_v, T^r_w. The representative vertices of the segments are, from left to right: u, r, v and w. The anchor of edge ru (whose length is α_{ru}), and the coanchor of edge ru (whose length is β_{ru}), are delimited by the dotted lines above edge ru.

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

refer to a real segment within the linear ordering containing all vertices of T_u^r (Figure 4); technically, that segment is an interval of length n_u whose starting and ending positions are unknown until the whole tree is fully linearized. Thus, a segment is a movable set of vertices within the linear ordering. The concept of *segment* is equivalent to the notion of continuous *constituent* in headed phrase structure representations (Kuhlmann and Nivre 2006). Hereafter, for simplicity, we refer to the "segment of a tree in a linear arrangement" simply as "segment of a tree," assuming that such a segment is defined with respect to a linear arrangement.

Proof of Proposition 1. We can associate a set of segments to each vertex. The set of vertex u contains $d_u + 1$ segments: one segment corresponds to u (the only segment of length 1), and the remaining d_u segments correspond to the immediate subtrees of T_u^r. We obtain a projective linear arrangement by permuting the elements of each set for all vertices. Therefore, a projective arrangement can be seen as being recursively composed of permutations of sets of segments. Such "recursion" starts at the permutation of the set of segments associated with r. Note, then, that there are $(d_u + 1)!$ possible permutations of the segments associated to vertex u. For a fixed permutation of the segments associated with r, there are $\prod_{u \in \Gamma_r} \mathbf{N}_{\mathrm{pr}}(T_u^r)$ different projective arrangements of its immediate subtrees T_u^r, hence the recurrence in Equation (8). Equation (9) follows upon unfolding the recurrence. □

The proof of Proposition 1 can be used to devise a simple procedure to generate projective arrangements uniformly at random, and another to enumerate all projective arrangements, of a rooted tree. As explained in previous articles (Gildea and Temperley 2007; Futrell, Mahowald, and Gibson 2015), the former method consists of first generating a uniformly random permutation of the $d_v + 1$ segments associated with every vertex $v \in V$ and, afterward, constructing the arrangement using these permutations. When a tree is linearized using the permutations of the sets of segments, we say that each segment becomes an interval.

2.2 The Expected Sum of Edge Lengths in Random Arrangements

We first review the problem of computing $\mathbb{E}[D(T)]$—the expected value of $D(T)$ in uniformly random unconstrained arrangements—so as to introduce the methodology applied for $\mathbb{E}_{\mathrm{pr}}[D(T^r)]$. The calculation requires two steps: first, the calculation of $\mathbb{E}[\delta_{uv}]$, the expected length of an arbitrary edge joining vertices u and v, and second, the calculation of $\mathbb{E}[D(T)]$; henceforth we denote these values as $\mathbb{E}[D]$ and $\mathbb{E}[\delta]$ since they only depend on the size of T, not on its topology. For simplicity, we assume the definition of edge length in Equation (1).

The calculation of $\mathbb{E}[\delta]$ requires the calculation of $\mathbb{P}(\delta)$, that is, the probability that an edge linking two vertices has length δ. This is actually the proportion of unconstrained linear arrangements such that the two vertices are at distance δ. Because arrangements are unconstrained, said probability, and the corresponding expectation, do not depend on the edge. There are $\mathbf{N}(T) = n!$ unconstrained linear arrangements and $2(n-\delta)(n-2)!$ unconstrained arrangements where the pair of vertices are at distance δ, hence

$$\mathbb{P}(\delta) = \frac{2(n-\delta)(n-2)!}{n!} = \frac{2(n-\delta)}{n(n-1)} \tag{10}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

as expected from previous research (Zörnig 1984; Ferrer-i-Cancho 2004). Then, the expected length of an edge in an unconstrained random arrangement is (Zörnig 1984; Ferrer-i-Cancho 2004)

$$\mathbb{E}[\delta] = \sum_{\delta=1}^{n-1} \delta \mathbb{P}(\delta) \tag{11}$$

$$= \frac{2}{n(n-1)} \left(n \sum_{\delta=1}^{n-1} \delta - \sum_{\delta=1}^{n-1} \delta^2 \right) \tag{12}$$

$$= \frac{2}{n(n-1)} \left(n \frac{1}{2}(n-1)n - \frac{1}{6}(n-1)n(2n-1) \right) \tag{13}$$

$$= \frac{n+1}{3} \tag{14}$$

The third equality follows from well-known formulae on the sum of integer numbers. The second step is the calculation of $\mathbb{E}[D]$, the expected value of D in an unconstrained arrangement. Since a tree has $n-1$ edges and applying linearity of expectation, $\mathbb{E}[D] = (n-1)\mathbb{E}[\delta]$, which gives Equation (4). Note that neither $\mathbb{E}[\delta_{uv}]$ nor $\mathbb{E}[D(T)]$ depend on T's topology (excluding n, the size of the tree).

2.3 The Expected Sum of Edge Lengths in Random Projective Arrangements

To obtain an arithmetic expression for $\mathbb{E}_{\text{pr}}[D(T^r)]$, we follow again a two-step approach. First, we calculate the expected length of an edge in uniformly random projective arrangements. However, unlike the unconstrained case, the edge must be incident to the root. Second, we calculate the expected $\mathbb{E}_{\text{pr}}[D(T^r)]$ applying the result of the first step. Before we proceed, we need to introduce some notation.

An edge connecting the root of the tree (r) with one of its children (u) can be decomposed into two parts: its **anchor** (Shiloach 1979; Chung 1984) and its **coanchor** (Figure 4). Such decomposition is also found in Gildea and Temperley (2007) and Park and Levy (2009), but using different terminology. In the context of projective linear arrangements, we define $\alpha_{ru}(\pi)$ as the length of the anchor, that is, the number of positions of the linear arrangement covered by the edge ru in the segment of T^r_u including the end of the edge $\pi(u)$; similarly, we define $\beta_{ru}(\pi)$ as the number of positions of the linear arrangement that are covered by that edge in segments other than that of T^r_u and r. Put differently, $\alpha_{ru}(\pi)$ is the width of the part of T^r_u covered by the edge ru including the end of the edge $\pi(u)$; similarly, β_{ru} is the total width of T^r_v over all r's children v that fall between r and u. Then the length of an edge connecting r with u is $\delta_{ru}(\pi) = |\pi(r) - \pi(u)| = \alpha_{ru}(\pi) + \beta_{ru}(\pi)$.

The next lemma shows that the expected value of δ_{ru} depends only on the size of the whole tree (n_r) and the size of the subtree rooted at the child (n_u).

Lemma 1
Let $T^r = (V, E; r)$ be a tree rooted at $r \in V$. Given an edge $ru \in E$, its anchor's expected length in uniformly random projective arrangements is

$$\mathbb{E}_{\text{pr}}[\alpha_{ru}] = \frac{n_u + 1}{2} \tag{15}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

and the expected length of its coanchor is

$$\mathbb{E}_{\text{pr}}[\beta_{ru}] = \frac{n_r - n_u - 1}{3} \tag{16}$$

Therefore, the expected length of an edge $ru \in E$ in such arrangements is

$$\mathbb{E}_{\text{pr}}[\delta_{ru}] = \mathbb{E}_{\text{pr}}[\alpha_{ru} + \beta_{ru}] = \frac{2n_r + n_u + 1}{6} \tag{17}$$

Proof. By the law of total expectation,

$$\begin{aligned}\mathbb{E}_{\text{pr}}[\alpha_{ru}] = \mathbb{E}_{\text{pr}}[\alpha_{ru}|\pi(u) < \pi(r)]\mathbb{P}_{\text{pr}}(\pi(u) < \pi(r)) \\ + \mathbb{E}_{\text{pr}}[\alpha_{ru}|\pi(u) > \pi(r)]\mathbb{P}_{\text{pr}}(\pi(u) > \pi(r))\end{aligned} \tag{18}$$

where $\mathbb{P}_{\text{pr}}(\pi(u) < \pi(r))$ is the probability that u precedes r in a random projective linear arrangement and $\mathbb{P}_{\text{pr}}(\pi(u) > \pi(r)) + \mathbb{P}_{\text{pr}}(\pi(u) < \pi(r)) = 1$. As any projective linear arrangement such that u precedes r has a reverse projective arrangement where u follows r, then $\mathbb{P}_{\text{pr}}(\pi(u) < \pi(r)) = 1/2$ and

$$\mathbb{E}_{\text{pr}}[\alpha_{ru}] = \frac{1}{2}\left(\mathbb{E}_{\text{pr}}[\alpha_{ru}|\pi(u) < \pi(r)] + \mathbb{E}_{\text{pr}}[\alpha_{ru}|\pi(u) > \pi(r)]\right) \tag{19}$$

Now, let q_u be the relative position of vertex u in its segment in the (projective) linear arrangement (the ith vertex of said segment is at relative position i). If u precedes r in the linear arrangement ($\pi(u) < \pi(r)$) as in Figure 4, then $\alpha_{ru} = n_u - q_u + 1$. If u follows r in the linear arrangement ($\pi(r) < \pi(u)$), $\alpha_{ru} = q_u$. Applying these two results, we obtain

$$\mathbb{E}_{\text{pr}}[\alpha_{ru}] = \frac{1}{2}\left(\mathbb{E}_{\text{pr}}[n_u - q_u + 1|\pi(u) < \pi(r)] + \mathbb{E}_{\text{pr}}[q_u|\pi(u) > \pi(r)]\right) \tag{20}$$

By symmetry, $\mathbb{E}_{\text{pr}}[q_u|\pi(u) < \pi(r)] = \mathbb{E}_{\text{pr}}[q_u|\pi(u) > \pi(r)]$ and then $\mathbb{E}_{\text{pr}}[\alpha_{ru}] = (n_u + 1)/2$, hence Equation (15).

In order to calculate β_{ru}, we define s_{ru} as the number of intermediate segments between r and the segment of T_u^r in the linear arrangement. Therefore, β_{ru} can be decomposed in terms of the lengths of each of these segments. The length of the ith segment in, say, the left-to-right order, is denoted as $\varphi_{ru}^{(i)}$. Formally, β_{ru} can be decomposed as

$$\beta_{ru} = \sum_{i=1}^{s_{ru}} \varphi_{ru}^{(i)} \tag{21}$$

By the law of total expectation,

$$\mathbb{E}_{\text{pr}}[\beta_{ru}] = \sum_{s=1}^{d_r - 1} \mathbb{E}_{\text{pr}}[\beta_{ru}|s_{ru} = s]\mathbb{P}_{\text{pr}}(s_{ru} = s) \tag{22}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

where $\mathbb{E}_{\text{pr}}[\beta_{ru}|s_{ru} = s]$ is the expectation of β_{ru} given that u and r are separated by s segments, and $\mathbb{P}_{\text{pr}}(s_{ru} = s)$ is the probability that u and r are separated by s segments, both in uniformly random projective arrangements. On the one hand,

$$\mathbb{E}_{\text{pr}}[\beta_{ru}|s_{ru} = s] = \mathbb{E}_{\text{pr}}\left[\sum_{i=1}^{s} \varphi_{ru}^{(i)}\right] = s\mathbb{E}_{\text{pr}}\left[\varphi_{ru}^{(i)}\right] \tag{23}$$

where

$$\mathbb{E}_{\text{pr}}\left[\varphi_{ru}^{(i)}\right] = \frac{n_r - n_u - 1}{d_r - 1} \tag{24}$$

is the average length of the segments excluding those of r and T_u^r. On the other hand, $\mathbb{P}_{\text{pr}}(s_{ru} = s)$ is the proportion of projective linear arrangements where the segments of T_u^r and that of r are separated by s segments in the linear arrangement, that is

$$\mathbb{P}_{\text{pr}}(s_{ru} = s) = \frac{2(d_r - s)(d_r - 1)! \prod_{u \in \Gamma_r} \mathbf{N}_{\text{pr}}(T_u^r)}{(d_r + 1)! \prod_{u \in \Gamma_r} \mathbf{N}_{\text{pr}}(T_u^r)} = \frac{2(d_r - s)}{(d_r + 1)d_r} \tag{25}$$

Plugging Equations (23) and (25) into Equation (22), one finally obtains Equation (16),

$$\mathbb{E}_{\text{pr}}[\beta_{ru}] = 2\frac{n_r - n_u - 1}{(d_r + 1)d_r(d_r - 1)} \sum_{s=1}^{d_r - 1} s(d_r - s) = \frac{n_r - n_u - 1}{3} \tag{26}$$

□

Now we can derive an arithmetic expression for $\mathbb{E}_{\text{pr}}[D(T^r)]$.

Proof of Theorem 1 (stated on page 494). Consider the random variable $D(T^r)$ over the probability space of uniformly random (unconstrained) linear arrangements of T^r, as defined above. This variable can be decomposed into two summations

$$D(T^r) = \sum_{u \in \Gamma_r} D(T_u^r) + \sum_{u \in \Gamma_r} \delta_{ru} \tag{27}$$

The first summation groups the edges by subtrees of T^r. The second summation groups the edges incident to the root r. Then, we can use linearity of expectation to obtain

$$\mathbb{E}_{\text{pr}}[D(T^r)] = \sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[D(T_u^r)] + \sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[\delta_{ru}] \tag{28}$$

The recurrence in Equation (5) follows easily from applying Lemma 1 to Equation (28), which gives

$$\sum_{ru \in E} \mathbb{E}_{\text{pr}}[\delta_{ru}] = \frac{1}{6} \sum_{ru \in E} (2n_r + 1) + \frac{1}{6} \sum_{ru \in E} n_u = \frac{d_r(2n_r + 1) + n_r - 1}{6} \tag{29}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Equation (6) follows upon unfolding the recurrence. □

In the proof above we implicitly use our definition of edge length $\delta_{uv}(\pi)$ (Equation (1)). Nevertheless, the expression in Equation (6) can be easily adjusted to use different definitions of edge length, for example, $\delta^*_{uv}(\pi)$ (Equation (2)). It suffices to find an appropriate transformation of our definition of $D(T^r)$ (Equation (3)) into the one desired, namely, $D^*(T^r)$. The next corollary gives the solution.

Corollary 2
Let $T^r = (V, E; r)$ be a tree rooted at $r \in V$. We have that

$$\mathbb{E}_{\text{pr}}[D^*(T^r)] = \frac{d_r(2n_r - 5) + n_r - 1}{6} + \sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[D^*(T^r_u)] \tag{30}$$

$$= \frac{1}{6}\left(5 - 6n_r + \sum_{v \in V} n_v(2d_v + 1)\right) \tag{31}$$

Proof of Corollary 2. The fact that $\mathbb{E}_{\text{pr}}[D^*(T^r)] = \mathbb{E}_{\text{pr}}[D(T^r)] - (n_r - 1)$, transforms Equation (5) into Equation (30) immediately as well as Equation (6) into Equation (31) thanks to

$$\sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[D(T^r_u)] = \sum_{u \in \Gamma_r} (\mathbb{E}_{\text{pr}}[D^*(T^r_u)] + n_u - 1) = n_r - 1 - d_r + \sum_{u \in \Gamma_r} \mathbb{E}_{\text{pr}}[D^*(T^r_u)] \tag{32}$$

□

It is easy to see that Equations (6) and (31) can both be evaluated in $O(n)$-time and $O(n)$-space, where n is the number of vertices of the tree: One only needs to compute the values n_v in $O(n)$-time, store them in $O(n)$ space, and then evaluate the formula, also in $O(n)$-time using those values. In the analysis above, we are assuming that the values of d_v are already computed; depending on the data structure used to represent the tree, the cost of computing d_v might be relevant.

2.4 Formulae for Classes of Trees

Here we consider three kinds of free trees that are later transformed into rooted trees (Harary 1969; Valiente 2021). First, **linear** (or **path**) trees are trees in which the maximum degree is 2. **Star** trees consist of a vertex connected to $n - 1$ leaves; also, a complete bipartite graph $K_{1,n-1}$. A **quasi-star** tree is a star tree in which one of its edges has been subdivided once with a vertex in the middle.[1] For the following analyses, we define the hub of a rooted tree as the vertex of the underlying free tree that has the highest degree; we also use the term "leaf" to refer to a leaf in the underlying free tree. We now instantiate Equations (6) and (9) for several classes of trees (Table 1): Star trees, $\mathcal{S}_n$, rooted at the hub, $\mathcal{S}^h_n$, and at a leaf, $\mathcal{S}^l_n$; Quasi-star trees, $\mathcal{Q}_n$, rooted at the hub, $\mathcal{Q}^h_n$, at a leaf adjacent to the hub, $\mathcal{Q}^{hl}_n$, at the leaf not adjacent to the hub $\mathcal{Q}^e_n$,

1 Alternatively, an n-vertex quasi-star tree is obtained by joining to a 2-vertex complete graph, K_2, a pendant vertex to one end and $n - 3$ pendant vertices to the other end of K_2; a quasi-star tree is a particular case of bistar tree (San Diego and Gella 2014).

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Table 1
Instantiations of $\mathbf{N}_{\text{pr}}(T^r)$ and $\mathbb{E}_{\text{pr}}[D(T^r)]$ for several classes of trees (Figure 5).

Class of tree	T^r	$\mathbf{N}_{\text{pr}}(T^r)$	$\mathbb{E}_{\text{pr}}[D(T^r)]$
Star	$\mathcal{S}_n^h$	$n!$	$(n^2-1)/3$
	$\mathcal{S}_n^l$	$2(n-1)!$	$n(2n-1)/6$
Quasi-star ($n \geq 4$)	$\mathcal{Q}_n^h$	$2(n-1)!$	$(2n^2-2n+3)/6$
	$\mathcal{Q}_n^e$	$4(n-2)!$	$(2n^2-2n+3)/6$
	$\mathcal{Q}_n^b$	$6(n-2)!$	$(2n^2-3n+7)/6$
	$\mathcal{Q}_n^{hl}$	$4(n-2)!$	$(2n^2-3n+7)/6$
Linear	$\mathcal{L}_n^0$	2^{n-1}	$(n-1)(n+2)/4$
	$\mathcal{L}_n^k$, $(k>0)$	$3 \cdot 2^{n-2}$	$[(n-1)(3n+10)+6k(k+1-n)]/12$

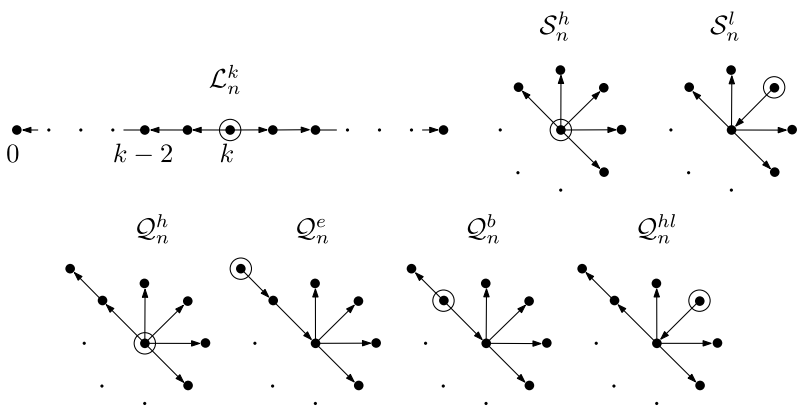

Figure 5
Linear trees ($\mathcal{L}_n$), star trees ($\mathcal{S}_n$), and quasi-star trees ($\mathcal{Q}_n$) of n vertices. Labels 0 and k in $\mathcal{L}_n^k$ denote the distance of the labeled vertex from the same leaf. A circled dot marks a tree's root.

and at the only internal vertex that is not the hub, $\mathcal{Q}_n^b$; Linear trees when rooted at a vertex at distance $k \geq 0$ from one of the leaves, $\mathcal{L}_n^k$. Each class of tree is depicted in Figure 5. These classes of trees are chosen for graph theoretic reasons. Linear trees minimize the variance of the degree (Ferrer-i-Cancho 2013); star graphs maximize it (Ferrer-i-Cancho 2013) and all their unconstrained linear arrangements are planar; concerning $\mathcal{S}_n^h$, all its linear arrangements are projective; quasi-star trees maximize the variance of the degree among trees whose set of unconstrained arrangements contains some non-planar arrangement (Ferrer-i-Cancho 2016).

We choose linear trees to illustrate how one can instantiate Equation (6). In order to ease this task, we rewrite Equation (6) using vectorial notation as

$$\mathbb{E}_{\text{pr}}[D(T^r)] = \frac{1}{6}\left(-1 + 2\sum_{v \in V} n_v d_v + \sum_{v \in V} n_v\right) = \frac{1}{6}\left(-1 + 2\overrightarrow{n_v} \cdot \overrightarrow{d_v} + \overrightarrow{n_v} \cdot \overrightarrow{1}\right) \tag{33}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

For $\mathcal{L}_n^0$, we have

$$\begin{aligned}\overrightarrow{n_v} &= (n, n-1, n-2, \cdots, 3, 2, 1)\\ \overrightarrow{d_v} &= (1, 1, \quad 1, \quad \cdots, 1, 1, 0)\end{aligned}$$

Then

$$\mathbb{E}_{\text{pr}}[D(\mathcal{L}_n^0)] = \frac{1}{6}\left(-1 + 2\sum_{i=2}^{n} i + \sum_{i=1}^{n} i\right) = \frac{(n-1)(n+2)}{4} \tag{34}$$

For $\mathcal{L}_n^k$ with $k > 0$, we have

$$\begin{aligned}\overrightarrow{n_v} &= (1, 2, \cdots, k-1, k, n, n-k-1, n-k-2, \cdots, 2, 1)\\ \overrightarrow{d_v} &= (0, 1, \cdots, 1, \quad 1, 2, 1, \quad 1, \quad \cdots, 1, 0)\end{aligned}$$

and then

$$\mathbb{E}_{\text{pr}}[D(\mathcal{L}_n^k)] = \frac{1}{6}\left[-1 + 2\left(2n + \sum_{j=2}^{k} j + \sum_{j=2}^{n-k-1} j\right) + \left(n + \sum_{j=1}^{k} j + \sum_{j=1}^{n-k-1} j\right)\right] \tag{35}$$

$$= \frac{(n-1)(3n+10) + 6k(k+1-n)}{12} \tag{36}$$

Regarding Equation (9) for $\mathcal{L}_n^0$ and $\mathcal{L}_n^k$, we have that

$$\mathbf{N}_{\text{pr}}(\mathcal{L}_n^0) = (0+1)! \prod_{i=1}^{n-1} (1+1)! = 2^{n-1} \tag{37}$$

$$\mathbf{N}_{\text{pr}}(\mathcal{L}_n^k) = (2+1)!(0+1)!(0+1)! \prod_{i=1}^{n-3} (1+1)! = 3 \cdot 2^{n-2} \tag{38}$$

3. Maxima and Minima

In this section, we tackle the problem of computing the minima and characterizing the maxima of $\mathbb{E}_{\text{pr}}[D(T^r)]$, both over all n-vertex root trees (keeping n constant). In particular, we give a closed-form formula for the maximum value of $\mathbb{E}_{\text{pr}}[D(T^r)]$ and characterize the trees that maximize it, as well as outline a dynamic programming algorithm to compute the minima. Henceforth, we use $\mathcal{T}_n$ to denote the set of n-vertex (unlabeled) rooted trees. Evidently, any tree that maximizes (respectively, minimizes) $\mathbb{E}_{\text{pr}}[D(T^r)]$ also maximizes (respectively, minimizes) $\mathbb{E}_{\text{pr}}[D^*(T^r)]$; thus we restrict our study to the former. Throughout this section, we use n_i to refer to the size of the subtree rooted at the ith child of the root for $1 \leq i \leq d_r$.

The construction of projective minimum linear arrangements has optimal substructure: Optimal arrangements are composed of optimal arrangements of subtrees (Hochberg and Stallmann 2003; Gildea and Temperley 2007; Alemany-Puig, Esteban, and Ferrer-i-Cancho 2022). Similarly, the construction of n-vertex rooted trees $T^r \in \mathcal{T}_n$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

that maximize (respectively, minimize) $\mathbb{E}_{\text{pr}}[D(T^r)]$ also has optimal substructure. The following lemma proves this claim.

Lemma 2
Let $\mathcal{T}_n$ be the set of all unlabeled rooted trees of n vertices. $f(n)$, the optimal value of $\mathbb{E}_{\text{pr}}[D(T^r)]$ satisfies

$$f(n) = \operatorname*{opt}_{T^r \in \mathcal{T}_n} \{\mathbb{E}_{\text{pr}}[D(T^r)]\} \tag{39}$$

$$= \operatorname*{opt}_{0 \le d_r \le n-1} \left\{ \frac{d_r(2n+1)+n-1}{6} + \operatorname*{opt}_{\substack{n_1+\cdots+n_{d_r}=n-1 \\ n_i \ge 1}} \left\{ \sum_{i=1}^{d_r} f(n_i) \right\} \right\} \tag{40}$$

Proof. We can construct an n-vertex optimal tree using optimal subtrees. We can obtain the right-hand side of Equation (40) using the recurrence in Equation (5). An optimal n-vertex tree is one whose cost value is optimal among all optimal n-vertex trees whose root has fixed degree $d_r \in \{1, \cdots, n-1\}$. Given a fixed root degree d_r such that $1 \le d_r \le n-1$, an optimal n-vertex rooted tree can be built by constructing an optimal $(n-1)$-vertex forest of d_r rooted trees, each of size n_i vertices and optimal among the n_i-vertex trees. Choosing the d_r trees to be n_i-optimal makes the sum of their costs, $\sum_i f(n_i)$, optimal. □

In subsequent paragraphs we use $f_M(n)$ to denote the *maximization* variant and $f_m(n)$ to denote the *minimization* variant of $f(n)$, respectively. Theorem 2 characterizes the maxima of $\mathbb{E}_{\text{pr}}[D(T^r)]$. Perhaps not so surprisingly, the only maximum of $\mathbb{E}_{\text{pr}}[D(T^r)]$ is obtained by $\mathcal{S}_n^h$.

Proof of Theorem 2 (stated on page 495). We prove this by induction on n using the formalization of the optimum in Equation (40). The base cases can be easily obtained by an exhaustive enumeration of the (unlabeled) rooted trees of n vertices for some small n. Indeed, the only tree that maximizes $\mathbb{E}_{\text{pr}}[D(T^r)]$ for $n \le 2$ are the one-vertex tree and the two-vertex tree, which are both star trees.

In order to prove that $\mathbb{E}_{\text{pr}}[D(T^r)] \le \mathbb{E}_{\text{pr}}[D(\mathcal{S}_n^h)]$ for $n \ge 3$, it suffices to prove that

$$\mathbb{E}_{\text{pr}}[D(\mathcal{S}_n^h)] = \frac{n^2-1}{3} > \max_{0 \le d_r \le n-2} \left\{ \frac{d_r(2n+1)+n-1}{6} + L(n, d_r) \right\} \tag{41}$$

where

$$L(n, d_r) = \max_{\substack{n_1+\cdots+n_{d_r}=n-1 \\ n_i \ge 1}} \left\{ \sum_{i=1}^{d_r} f_M(n_i) \right\} \tag{42}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

For this we need to know the maximum value of $L(n, d_r)$. Applying the induction hypothesis (each maximum subtree of a tree of n vertices is a star tree), we have that

$$L(n, d_r) = \max_{\substack{n_1+\cdots+n_{d_r}=n-1 \\ n_i \geq 1}} \left\{ \sum_{i=1}^{d_r} \frac{n_i^2 - 1}{3} \right\} = \frac{1}{3} \left(-d_r + \max_{\substack{n_1+\cdots+n_{d_r}=n-1 \\ n_i \geq 1}} \left\{ \sum_{i=1}^{d_r} n_i^2 \right\} \right) \tag{43}$$

Notice that any vector $(n_1, \cdots, n_i, \cdots, n_{d_r})$, such that $1 \leq n_1 \leq \cdots \leq n_{d_r}$, can be transformed into another vector $(n_1, \cdots, n_i - 1, \cdots, n_{d_r} + 1)$, for any $n_i \geq 2$, such that the sum of squared components is strictly larger while the sum of the components remains constant. Therefore, the maximum sum of squares is obtained by choosing $n_{d_r} = n - d_r$ and $n_i = 1$ for $1 \leq i < d_r$, yielding

$$L(n, d_r) = \frac{1}{3} \left[-d_r + (n - d_r)^2 + (d_r - 1) \right] = \frac{(n - d_r)^2 - 1}{3} \tag{44}$$

and the theorem holds if, and only if

$$\frac{n^2 - 1}{3} > \max_{0 \leq d_r \leq n-2} \left\{ \frac{d_r(2n + 1) + n - 1}{6} + \frac{(n - d_r)^2 - 1}{3} \right\} \tag{45}$$

After rearranging the terms that do not depend on d to the left-hand side we obtain

$$-n + 1 > \max_{0 \leq d_r \leq n-2} \{ d_r(2d_r - 2n + 1) \} \tag{46}$$

The right-hand side of the inequality is maximized for $d_r = n - 2$. Because this last inequality holds true when $n > 5/2$, we are done. □

It is easy to see that $\mathbb{E}_{\text{pr}}[D(T^r)]$ is bounded above by the expected value of $D(T)$ as stated in Corollary 1 and justified in its proof below.

Proof of Corollary 1 (stated on page 495). Because of Theorem 2, the maximum value of $\mathbb{E}_{\text{pr}}[D(T^r)]$ is maximized by $\mathcal{S}_n^h$, formally $\mathbb{E}_{\text{pr}}[D(T^r)] \leq \mathbb{E}_{\text{pr}}[D(\mathcal{S}_n^h)]$, which, in turn, becomes $\mathbb{E}_{\text{pr}}[D(\mathcal{S}_n^h)] = (n^2 - 1)/3$, as shown in Table 1. Finally, recall that $\mathbb{E}[D(T)] = (n^2 - 1)/3$ (Equation (4)), and thus $\mathbb{E}_{\text{pr}}[D(\mathcal{S}_n^h)] = \mathbb{E}[D(T)]$. □

We devised a dynamic programming algorithm based on Lemma 2 to calculate the distinct trees up to isomorphism minimizing $\mathbb{E}_{\text{pr}}[D(T^r)]$. The method to obtain said values and trees is outlined in Algorithm 1. That algorithm has two parameters: n, the number of vertices, and H a hash table whose keys are natural numbers k and the value associated with each key is a pair formed by $f_m(k)$ and the k-vertex trees T^r that attain that $f_m(k)$. Notice that n is an input parameter, while H is an input/output parameter. In order to calculate the minimum n-vertex trees, the values of the parameters of the

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Algorithm 1: Calculate the minimum value and trees of $\mathbb{E}_{\text{pr}}[D(T^r)]$.

Input: $n \in \mathbb{N}$, the number of vertices of the trees. H is a hash table whose keys are natural numbers $k \in \mathbb{N}$ and the value of each key is the pair formed by $f_m(k)$ and the k-vertex trees T^r for which $\mathbb{E}_{\text{pr}}[D(T^r)] = f_m(k)$.
Output: The hash table H updated to contain the results for $n' \leq n$.

```
Function MINIMUM_E_PROJECTIVE(n, H) is
    if hash table H contains key n then Stop.
    f_m ← (n^2 − 1)/3 // The minimum value initialized at a maximum
    T* ← ∅ // The set of n-optimal trees
    if n ≤ 2 then
        T* ← {the only rooted tree T^r of n vertices}
        f_m ← E_pr[D(T^r)]
    else
        for each d_r ∈ [1, n − 1] do
            P ← The set of partitions of 'n − 1' in d_r summands.
            for each partition P = {n_1, ··· , n_{d_r}} ∈ P do
                // Initialize the cost of the trees
                C ← (d_r(2n + 1) + n − 1)/6
                // Evaluate the partition recursively
                for each n_i ∈ P do
                    // Calculate the n_i-minimum trees
                    MINIMUM_E_PROJECTIVE(n_i, H)
                    f_m^(i), {T*}^(i) ← H[n_i] // Retrieve the trees from H
                    C ← C + f_m^(i)
                    if C > f_m then
                        Stop evaluating P and move on to the next partition
                // Modified Cartesian product of all {T*}^(i) (see text)
                F ← {T*}^(1) × ··· × {T*}^(d)
                // W is the set of (n, d, P)-optimal trees
                W* ← add a root to every forest in F
                if C < f_m then f_m ← C, T* ← W*
                else if C = f_m then T* ← T* ∪ W*
    Store ⟨ f_m, T* ⟩ in H[n]
```

first call to Algorithm 1 are the value n and an empty H. Now, Algorithm 1 is a direct evaluation of Equation (40), that is, for every value of out-degree of the root d_r ($1 \leq d_r \leq n-1$), it computes the value $f_m(n)$ by finding the partition of $n-1$ into d summands that minimizes

$$\frac{d_r(2n+1)+n-1}{6} + \operatorname*{opt}_{\substack{n_1+\cdots+n_{d_r}=n-1 \\ n_i \geq 1}} \left\{ \sum_{i=1}^{d_r} f_m(n_i) \right\} \tag{47}$$

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Table 2
The columns indicate, from left to right, the number of (unlabeled) rooted trees (Sloane 1964–2022), the number of trees that minimize $\mathbb{E}_{\text{pr}}[D(T^r)]$, and the value of $\mathbb{E}_{\text{pr}}[D(T^r)]$ for such trees. The trees yielding these values are displayed in Figure 7. The horizontal line over a decimal digit denotes it is an infinitely repeating digit; thus $1/3 = 0.\overline{3}$.

n	# trees	# opt trees	$\mathbb{E}_{\text{pr}}[D(T^r)]$	n	# trees	# opt trees	$\mathbb{E}_{\text{pr}}[D(T^r)]$
1	1	1	0	11	1,842	1	22
2	1	1	1	12	4,766	1	$25.1\overline{6}$
3	1	1	2.5	13	12,486	2	$28.\overline{3}$
4	4	2	4.5	14	32,973	1	31.5
5	9	1	$6.\overline{3}$	15	87,811	1	$34.\overline{6}$
6	20	1	$8.\overline{6}$	16	235,381	1	38
7	48	1	11	17	634,847	1	41.5
8	115	2	$13.8\overline{3}$	18	1,721,159	2	45
9	286	1	16.5	19	4,688,676	2	48.5
10	719	2	$19.\overline{3}$	20	12,826,228	2	52

where $f_m(n_i)$ is calculated recursively and stored in the hash table H. Therefore, the algorithm's complexity heavily depends on the number of partitions of $n-1$, denoted as $p(n-1)$. The worst-case complexity of Algorithm 1, then, is superpolynomial in n due to the exponential nature of $p(n)$ (Hardy and Ramanujan 1918). Finally, notice that the "Modified Cartesian" product in line 19 of Algorithm 1 must ensure that no repeated trees are produced. Repeated trees arise in the standard Cartesian product due to the fact that some partitions may have repeated parts. As an example, consider the partition of 37 with two repeated parts $(11, 13, 13)$; such parts have 1 and 2 non-isomorphic minimum trees, respectively (Figure 7). Let $t_{11} = \{T_A\}$ be the unique 11-vertex minimum tree, and $t_{13} = \{T_B, T_C\}$ be the two 13-vertex minimum trees. The Cartesian product $t_{11} \times t_{13} \times t_{13}$ produces four forests, two of them being isomorphic: (T_A, T_B, T_C) and (T_A, T_C, T_B); the other two forests are (T_A, T_B, T_B) and (T_A, T_C, T_C). In order to obtain the unique n-vertex trees (up to isomorphism) that attain $f_m(n)$, we modify the standard Cartesian product. Let $\{T^*\}^{(i)}$ be the list of n_i-vertex minimum trees. Thus, one element in the Cartesian product is obtained by choosing the trees in the j_1th, $\cdots$, j_{d_r}th positions of the lists, that is,

$$(\{T^*\}^{(1)}_{j_1}, \cdots, \{T^*\}^{(d_r)}_{j_{d_r}}) \in \{T^*\}^{(1)} \times \cdots \times \{T^*\}^{(d_r)} \tag{48}$$

where $1 \leq j_i \leq |\{T^*\}^{(i)}|$ for all $i \in [1, d_r]$. Now, tree uniqueness is ensured by forcing indices of every pair of lists $\{T^*\}^{(i)}, \{T^*\}^{(i+1)}$ such that $n_i = n_{i+1}$ is $j_i \leq j_{i+1}$.

In this article, we do not characterize the minima of $\mathbb{E}_{\text{pr}}[D(T^r)]$ because, unlike the number of maxima, which shows a clear regularity, the number of minima varies with n in a non-monotonic fashion. Table 2 shows that the number of these minima oscillates between 1 and 2 for $n \leq 20$ and Figure 6 shows the number of minimum trees in linear-log scale for $n \leq 178$. Figure 7 suggests that the shape of the trees does not seem to fit into a simple class. Moreover, Figure 8 shows the values of $f_m(n)$ in log-linear scale. The straight line that is found in that scale for sufficiently large n suggests $f_m(n) = O(n \log n)$ asymptotic behavior, to be confirmed in future research.

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

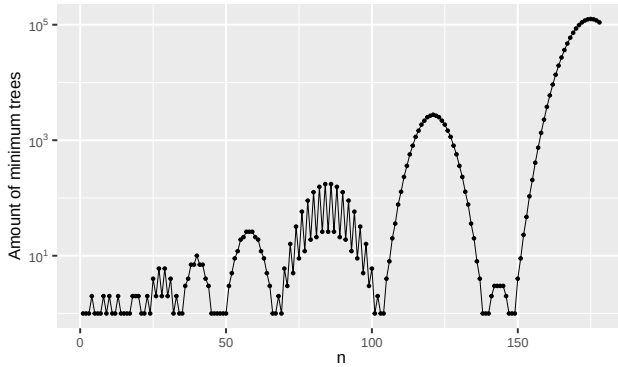

Figure 6
Number of distinct n-vertex (unlabeled) trees T^r that minimize $\mathbb{E}_{\text{pr}}[D(T^r)]$ (that is, $\mathbb{E}_{\text{pr}}[D(T^r)] = f_m(n)$) for $n \leq 178$. Notice the logarithmic scale for the y-axis. The number is computed via Algorithm 1.

4. Exact versus Approximate Calculation of the Expected Sum of Dependency Distances

In previous research the random baseline for D under projectivity has been estimated with a Monte Carlo approximation method[2] (Gildea and Temperley 2007; Park and Levy 2009; Gildea and Temperley 2010; Futrell, Mahowald, and Gibson 2015; Kramer 2021) whose cost $O(Rn)$, where R is the number of random arrangements sampled and $O(n)$ is the cost of generating a random arrangement and computing D on it. Our method, as well as providing exact values, has a much lower complexity $O(n)$ as there is no need for random sampling. We used the *UD2.5* (de Marneffe et al. 2019) treebanks for Catalan, English, and German to measure the relative error of the Monte Carlo approximation method for values of R used in past research. English and German are selected because they have been used in previous research utilizing projective random baselines (Gildea and Temperley 2007; Park and Levy 2009; Gildea and Temperley 2010). Catalan is included as the native language of the present authors. For each sentence T^r in a treebank, we calculated $\mathbb{E}_{\text{pr}}[D(T^r)]$ in two ways. First, exactly with Equation (6). Second, approximately by averaging the value of D obtained in $R = 10^i$, for $1 \leq i \leq 4$, uniformly random projective arrangements (denoted as $\tilde{\mathbb{E}}_{\text{pr}}[D(T^r)]$). More precisely, given $\{\pi_i\}_{i=1}^{R}$ random projective arrangements of a given rooted tree,

$$\tilde{\mathbb{E}}_{\text{pr}}[D(T^r)] = \frac{1}{R}\sum_{i=1}^{R} D_{\pi_i}(T^r) \tag{49}$$

2 This Monte Carlo method consists of averaging the values of D obtained via random sampling of R random projective arrangements of a rooted tree. It is well known that the error of such methods decreases as the number of samples increases.

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

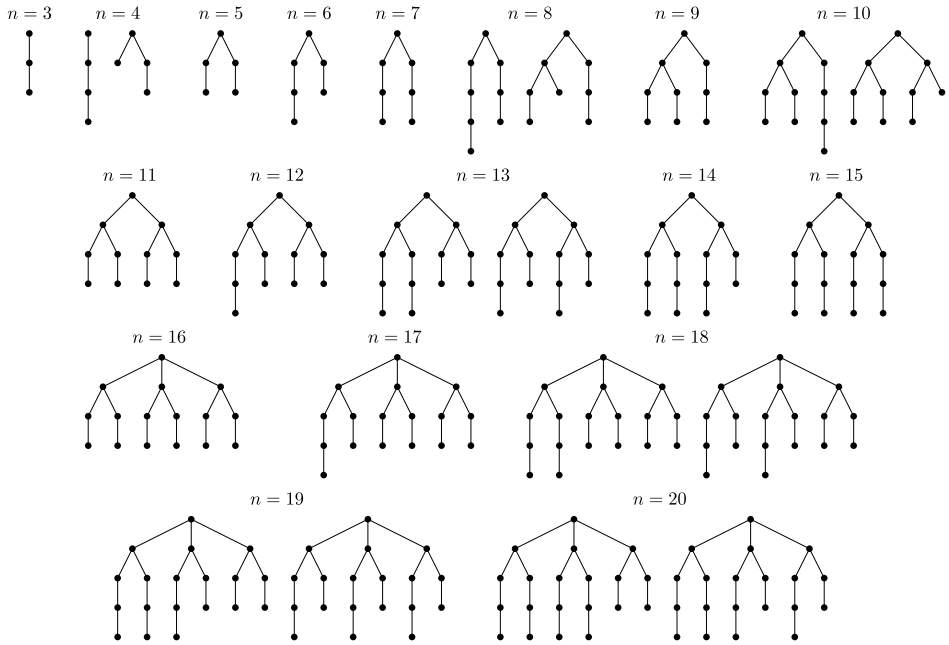

Figure 7
The trees that minimize $\mathbb{E}_{\text{pr}}[D(T^r)]$ for $n \leq 20$. Their values of $\mathbb{E}_{\text{pr}}[D(T^r)]$ are given in Table 2. Roots are drawn atop each tree and each edge should be regarded as oriented away from the root.

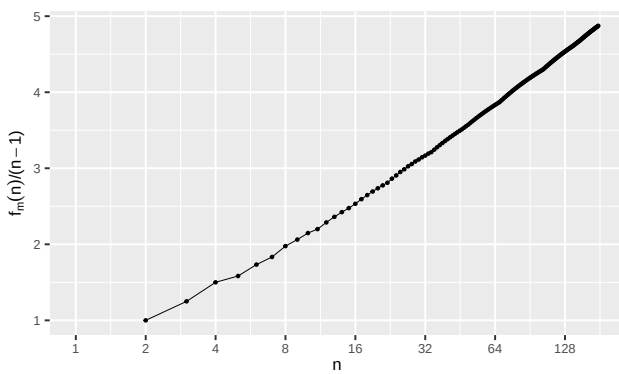

Figure 8
Values of $f_m(n)/(n-1)$ in log-linear scale for $n \leq 178$, computed via Algorithm 1.

Using these values we calculated the relative error $\varepsilon_{rel}(T^r)$ for every tree as

$$\varepsilon_{rel}(T^r) = \frac{\tilde{\mathbb{E}}_{\text{pr}}[D(T^r)] - \mathbb{E}_{\text{pr}}[D(T^r)]}{\mathbb{E}_{\text{pr}}[D(T^r)]} \tag{50}$$

$\varepsilon_{rel}(T^r) > 0$ indicates that $\tilde{\mathbb{E}}_{\text{pr}}[D(T^r)]$ overestimates $\mathbb{E}_{\text{pr}}[D(T^r)]$; $\varepsilon_{rel}(T^r) < 0$ indicates underestimation error. Figure 9 shows the average, minimum, maximum, and confidence interval of the relative error as a function of sentence length (see the Appendix for

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

a parallel analysis of the standard deviation of $D(T^r)$). This figure can be seen as a confirmation of the correctness of Theorem 1 via simulation.

In previous research, the value of R is sometimes indicated, for example, $R = 10$ (Kramer 2021) and $R = 100$ (Futrell, Mahowald, and Gibson 2015), and sometimes not reported (Park and Levy 2009; Gildea and Temperley 2010). When $R = 10$, Figure 9 shows that the relative error peaks between $n = 10$ and $n = 20$ (close to the mean sentence length in English [Rudnicka 2018]) and tends to decay from then onward. However, the confidence interval of the relative error clearly broadens for sufficiently large n, for example, $n = 30$ or $n = 40$ onward. These behaviors smooth out for $R = 100$ while the range of variation and the confidence interval are narrower. As expected, the relative error reduces dramatically for larger R. In sum, Figure 9 indicates that a large R must be used to estimate $\mathbb{E}_{\text{pr}}[D(T^r)]$ with high numerical precision. However, that increases the computation time. An implication of that figure is that the best solution in terms of numerical precision and speed is provided by Equation (31).

5. Conclusions and Future Work

In this article, we have derived several simple closed-form formulae related to projective arrangements of rooted trees in Section 2. First, we have given a simple closed-form formula to calculate the number of *different* projective arrangements a rooted tree admits, $\mathbf{N}_{\text{pr}}(T^r)$ (Proposition 1). The proof reveals a straightforward way of enumerating such arrangements for an input rooted tree, and of generating this kind of arrangement uniformly at random without a rejection method. Second, and more importantly, we have provided a way of calculating, for any given rooted tree, the expected value of the sum of edge lengths over the space of uniformly random *different* projective arrangements, $\mathbb{E}_{\text{pr}}[D(T^r)]$ (Theorem 1 and Corollary 2). This means that future studies in which such value is calculated approximately via random sampling of arrangements can now be calculated exactly and much faster. The $O(Rn)$ Monte Carlo method to estimate true expectation with an error that tends to zero as R tends to infinity can now be replaced by our fast $O(n)$ with zero error. Moreover, these formulae can be instantiated in particular classes of trees (as shown in Section 2.4). In Section 3, we have characterized the trees that maximize $\mathbb{E}_{\text{pr}}[D(T^r)]$ and proven that there exists a dynamic programming method to calculate the minima of $\mathbb{E}_{\text{pr}}[D(T^r)]$. A precise characterization of the minima should be the subject of future research. Finally, in Section 4 we have highlighted the obvious advantages of an exact and fast calculation of $\mathbb{E}_{\text{pr}}[D(T^r)]$ for future quantitative dependency syntax research.

The present article is part of a research program on the calculation of random baselines for D via formulae or exact algorithms under formal constraints on linear arrangements that started about two decades ago with the unconstrained case, for example Equation (4) (Zörnig 1984; Ferrer-i-Cancho 2004). Here we have covered the projective case. In a forthcoming article, we will focus on planar linearizations of (free) trees—those in which there are no edge crossings—and obtain a closed-form formula, and a $O(n)$-time algorithm, to calculate $\mathbb{E}_{\text{pl}}[D(T)]$, the expected value of $D(T)$ in uniformly random planar arrangements of a given (labeled free) tree T (Alemany-Puig and Ferrer-i-Cancho 2022). In the future, the problem of the calculation of the variance of $D(T^r)$ in random projective arrangements should also be considered. The analysis of the distribution of dependency distances—for example, their first and second moments of $D(T^r)$, in random arrangements that are not uniformly random (but still projective and planar)—could benefit from applying general-purpose algorithmic frameworks (Eisner 2002; Li and Eisner 2009; Wang and Eisner 2018).

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

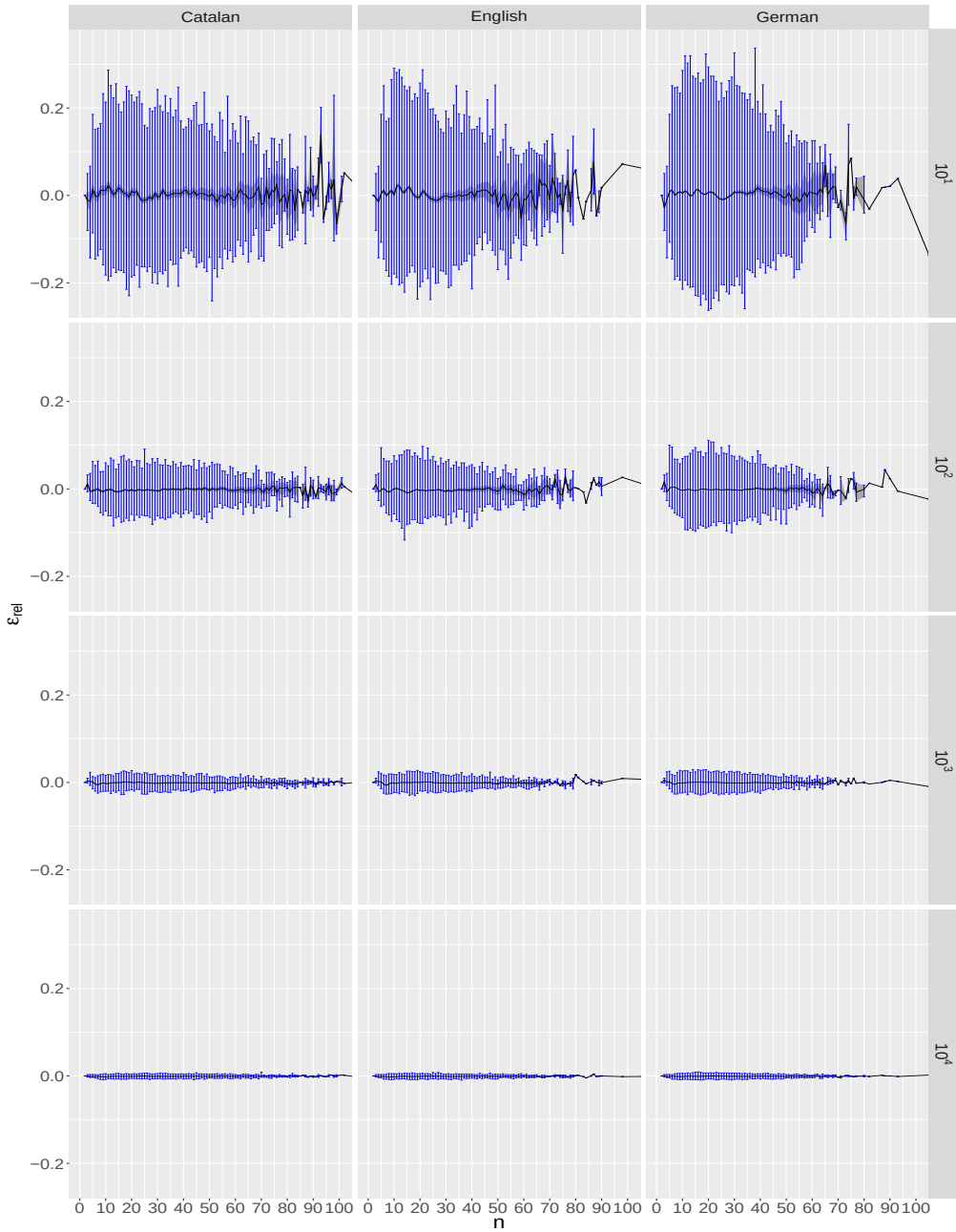

Figure 9
The statistical properties of ε_{rel}, the relative error in the estimation of $\mathbb{E}_{\text{pr}}[D(T^r)]$ calculated with a Monte Carlo method (Equation 50), as a function of the size of the tree n. For all sentences of the same length, we show the average ε_{rel} (solid black line), its 99% confidence interval calculated with a bootstrap method (shaded gray region), and the maximum and minimum ε_{rel} (vertical blue bars). The end of each row indicates R, that is, the number of random projective arrangements used to estimate $\mathbb{E}_{\text{pr}}[D(T^r)]$ (from $R = 10$ for the top row to $R = 10^4$ for the bottom row).

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Our research paves the way to investigate the optimality of dependency distances of languages under projectivity. Recently, that optimality has been evaluated in 93 languages from 19 families with the help of a new score, $\Omega(T)$, which is defined with respect to the minimum and random baseline in unconstrained linear arrangements. $\Omega(T)$ is defined as (Ferrer-i-Cancho et al. 2022)

$$\Omega(T) = \frac{\mathbb{E}[D(T)] - D(T)}{\mathbb{E}[D(T)] - m[D(T)]} \tag{51}$$

where $m[D(T)]$ is the minimum $D_\pi(T)$ over all unconstrained linear arrangements π, known as the Minimum Linear Arrangement problem (Garey, Johnson, and Stockmeyer 1976; Shiloach 1979; Chung 1984). However, projectivity is the most widely used constraint to investigate dependency distances and to define the corresponding minimum and random baselines (Futrell, Mahowald, and Gibson 2015; Gulordava and Merlo 2015; Futrell, Levy, and Gibson 2020). With the result in Theorem 1, one could replicate the aforementioned study under the projectivity constraint by redefining the score as

$$\Omega_{\text{pr}}(T^r) = \frac{\mathbb{E}_{\text{pr}}[D(T^r)] - D(T^r)}{\mathbb{E}_{\text{pr}}[D(T^r)] - m_{\text{pr}}[D(T^r)]} \tag{52}$$

where the minimum sum of edge lengths under the projectivity constraint is denoted as $m_{\text{pr}}[D(T^r)]$ (Hochberg and Stallmann 2003; Gildea and Temperley 2007), and is linear-time computable (Alemany-Puig, Esteban, and Ferrer-i-Cancho 2022). While $\Omega(T) \leq 1$ holds for any sentence (Ferrer-i-Cancho et al. 2022), as it is equivalent to $D(T) \geq m[D(T)]$, the statement $\Omega_{\text{pr}}(T^r) \leq 1$ only holds when applied to projective sentences. In other words, $\Omega_{\text{pr}}(T^r) \leq 1$ needs not hold because we can have that $D(T^r) > m_{\text{pr}}[D(T^r)]$. In the absence of any word order constraint on a sentence, Ω is expected to be zero while $\Omega_{\text{pr}}(T^r)$ is expected to be zero if the only word order constraint is projectivity. Formally, $\mathbb{E}[\Omega(T)] = 0$ and $\mathbb{E}_{\text{pr}}[\Omega_{\text{pr}}(T^r)] = 0$. Thanks to our article, such investigation of the optimality of dependency distances can be carried out, reducing the computational cost and maximizing numerical precision with respect to an approach based on a Monte Carlo estimation of $\mathbb{E}_{\text{pr}}[D(T^r)]$.

Here we have focused on $\mathbb{E}_{\text{pr}}[D(T^r)]$, the expectation of $D(T^r)$ on random arrangements of an individual tree. Finally, future research should consider the problem of $\mathbb{E}[\mathbb{E}_{\text{pr}}[D]]$, the expectation of $\mathbb{E}_{\text{pr}}[D]$ on ensembles of random trees of a fixed size n. $\mathbb{E}[\mathbb{E}_{\text{pr}}[D]]$ is indeed the average value of $\mathbb{E}_{\text{pr}}[D(T^r)]$ among all n-vertex rooted trees T^r. Although there are at least two ensembles possible, that is, uniformly random *labeled* trees and uniformly random *unlabeled* trees, a closed-form formula (or algorithm) to calculate $\mathbb{E}[\mathbb{E}_{\text{pr}}[D]]$ seems easier to obtain in the former.

Acknowledgments

We are thankful to the anonymous reviewers for invaluable comments and suggestions. LAP is supported by Secretaria d'Universitats i Recerca de la Generalitat de Catalunya and the Social European Fund. RFC is also supported by the recognition 2017SGR-856 (MACDA) from AGAUR (Generalitat de Catalunya). RFC and LAP are supported by the grant TIN2017-89244-R from MINECO (Ministerio de Economía, Industria y Competitividad).

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

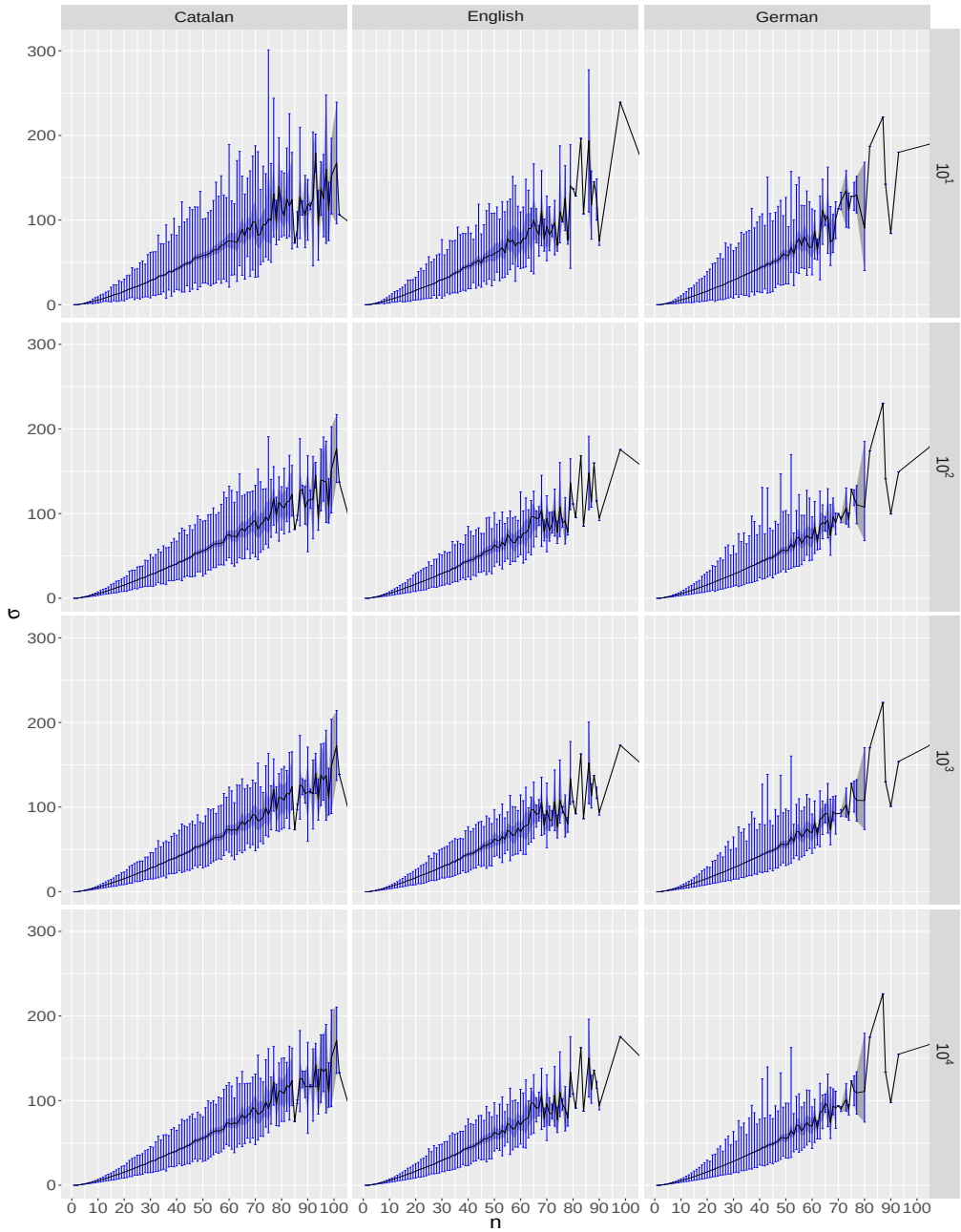

Figure A.1
The statistical properties of σ, the standard deviation in the estimation of $\mathbb{E}_{\text{pr}}[D(T^r)]$ calculated with a Monte Carlo method (Equation A.1), as a function of the size of the tree n. This figure has the same format as Figure 9. For all sentences of the same length, we show the average σ (solid black line), its 99% confidence interval calculated with a bootstrap method (shaded gray region), and the maximum and minimum ε_{rel} (vertical blue bars). The end of each row indicates R, that is, the number of random projective arrangements used to estimate $\mathbb{E}_{\text{pr}}[D(T^r)]$ (from $R = 10$ for the top row to $R = 10^4$ for the bottom row).

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Appendix A. The Standard Deviation of $D(T^r)$ in Projective Arrangements

Besides estimating the relative error of a Monte Carlo method to approximate $\mathbb{E}_{\text{pr}}[D(T^r)]$, we also calculated the standard deviation $\sigma(T^r)$ of $D_\pi(T^r)$ over R random projective arrangements π via

$$\sigma^2(T^r) = \frac{1}{R-1}\sum_{i=1}^{R}\left(D_{\pi_i}(T^r) - \mathbb{E}_{\text{pr}}[D(T^r)]\right)^2 \tag{A.1}$$

Notice that σ is calculated using $\mathbb{E}_{\text{pr}}[D(T^r)]$ and not $\tilde{\mathbb{E}}_{\text{pr}}[D(T^r)]$. Figure A.1 shows that σ increases with n (as expected) but that it does not decrease as R increases; furthermore, the apparent linear trend for sufficiently large n suggests an $O(n)$ growth of σ, to be confirmed in future research.

References

Alemany-Puig, Lluís, Juan Luis Esteban, and Ramon Ferrer-i-Cancho. 2021. The Linear Arrangement Library. A new tool for research on syntactic dependency structures. In *Proceedings of the Second Workshop on Quantitative Syntax (Quasy, SyntaxFest 2021)*, pages 1–16.

Alemany-Puig, Lluís, Juan Luis Esteban, and Ramon Ferrer-i-Cancho. 2022. Minimum projective linearizations of trees in linear time. *Information Processing Letters*, 174:106204. https://doi.org/10.1016/j.ipl.2021.106204

Alemany-Puig, Lluís and Ramon Ferrer-i-Cancho. 2022. Linear-time calculation of the expected sum of edge lengths in planar linearizations of trees. *In preparation*. https://doi.org/10.1162/coli_a_00442

Bernhart, Frank and Paul C. Kainen. 1979. The Book Thickness of a Graph. *Journal of Combinatorial Theory, Series B*, 27(3):320–331. https://doi.org/10.1016/0095-8956(79)90021-2

Bodirsky, Manuel, Marco Kuhlmann, and Mathias Möhl. 2005. Well-nested drawings as models of syntactic structure. In *Proceedings of the 10th Conference on Formal Grammar and 9th Meeting on Mathematics of Language*, pages 195–204, http://cslipublications.stanford.edu/FG/2005/FGMoL05.pdf#page=207

Chung, Fan R. K. 1984. On optimal linear arrangements of trees. *Computers and Mathematics with Applications*, 10(1):43–60. https://doi.org/10.1016/0898-1221(84)90085-3

de Marneffe, Marie-Catherine, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher Manning, Ryan McDonald, Joakim Nivre, Slav Petrov, Sampo Pyysalo, Sebastian Schuster, Natalia Silveira, Reut Tsarfaty, Francis Tyers, and Dan Zeman. 2019. Universal dependencies. https://universaldependencies.org/. Accessed: 2021-06-07.

Eisner, Jason. 2002. Parameter estimation for probabilistic finite-state transducers. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 1–8. https://doi.org/10.3115/1073083.1073085

Eppler, Eva Maria. 2005. *The Syntax of German-English Code-Switching*. Ph.D. thesis, University College, London.

Ferrer-i-Cancho, Ramon. 2004. Euclidean distance between syntactically linked words. *Physical Review E*, 70(5):5. https://doi.org/10.1103/PhysRevE.70.056135

Ferrer-i-Cancho, Ramon. 2013. Hubiness, length, crossings and their relationships in dependency trees. *Glottometrics*, 25:1–21.

Ferrer-i-Cancho, Ramon. 2016. Non-crossing dependencies: least effort, not grammar. In Alexander Mehler, Andy Lücking, Sven Banisch, Philippe Blanchard, and Barbara Job, editors, *Towards a Theoretical Framework for Analyzing Complex Linguistic Networks*. Springer, Berlin, pages 203–234. https://doi.org/10.1007/978-3-662-47238-5_10

Ferrer-i-Cancho, Ramon. 2019. The sum of edge lengths in random linear arrangements. *Journal of Statistical Mechanics*, 2019(5):053401. https://doi.org/10.1088/1742-5468/ab11e2

Ferrer-i-Cancho, Ramon, Carlos Gómez-Rodríguez, Juan Luis Esteban, and Lluís Alemany-Puig. 2022. Optimality of syntactic dependency distances. *Physical*

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

Review E, 105:014308. https://doi.org/10.1103/PhysRevE.105.014308
Futrell, Richard, Roger Park Levy, and Edward Gibson. 2020. Dependency locality as an explanatory principle for word order. *Language*, 96(2):371–412. http://dx.doi.org/10.1353/lan.2020.0024
Futrell, Richard, Kyle Mahowald, and Edward Gibson. 2015. Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341. https://doi.org/10.1073/pnas.1502134112
Garey, Michael R., David Stifler Johnson, and Larry J. Stockmeyer. 1976. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1:237–267. https://doi.org/10.1016/0304-3975(76)90059-1
Gildea, Daniel and David Temperley. 2007. Optimizing grammars for minimum dependency length. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 184–191.
Gildea, David and David Temperley. 2010. Do grammars minimize dependency length? *Cognitive Science*, 34(2):286–310. https://doi.org/10.1111/j.1551-6709.2009.01073.x
Gómez-Rodríguez, Carlos, John Carroll, and David Weir. 2011. Dependency parsing schemata and mildly non-projective dependency parsing. *Computational Linguistics*, 37(3):541–586. https://doi.org/10.1162/COLI_a_00060
Groß, Thomas and Timothy Osborne. 2009. Toward a practical dependency grammar theory of discontinuities. *SKY Journal of Linguistics*, 22:43–90.
Gulordava, Kristina and Paola Merlo. 2015. Diachronic trends word order freedom and dependency length in dependency-annotated corpora of Latin and Ancient Greek. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 121–130. http://www.aclweb.org/anthology/W15-2115
Harary, Frank. 1969. *Graph Theory*. Addison-Wesley, Reading, MA. https://doi.org/10.21236/AD0705364
Hardy, Godfrey Harold and Srinivasa Ramanujan. 1918. Asymptotic formulaæ in combinatory analysis. *Proceedings of the London Mathematical Society*, s2-17(1):75–115. https://doi.org/10.1112/plms/s2-17.1.75
Hiranuma, So. 1999. Syntactic difficulty in English and Japanese: A textual study. *UCL Working Papers in Linguistics*, 11:309–322.
Hochberg, Robert A. and Matthias F. Stallmann. 2003. Optimal one-page tree embeddings in linear time. *Information Processing Letters*, 87(2):59–66. https://doi.org/10.1016/S0020-0190(03)00261-8
Hudson, Richard. 1995. Measuring syntactic difficulty. Unpublished paper. Available at https://dickhudson.com/wp-content/uploads/2013/07/Difficulty.pdf
Iordanskii, Mikhail Anatolievich. 1987. Minimal numberings of the vertices of trees—Approximate approach. In *Fundamentals of Computation Theory*, pages 214–217, Springer Berlin Heidelberg, Berlin, Heidelberg. https://doi.org/10.1007/3-540-18740-5_44
Kramer, Alex. 2021. Dependency lengths in speech and writing: A cross-linguistic comparison via YouDePP, a pipeline for scraping and parsing YouTube captions. In *Proceedings of the Society for Computation in Linguistics*, volume 4, pages 359–365.
Kuhlmann, Marco and Joakim Nivre. 2006. Mildly non-projective dependency structures. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, COLING-ACL '06, pages 507–514. https://doi.org/10.3115/1273073.1273139
Li, Zhifei and Jason Eisner. 2009. First- and second-order expectation semirings with applications to minimum-risk training on translation forests. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 40–51. https://doi.org/10.3115/1699510.1699517
Liu, Haitao, Chunshan Xu, and Junying Liang. 2017. Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of Life Reviews*, 21:171–193. https://doi.org/10.1016/j.plrev.2017.03.002
Mel'čuk, Igor. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press, Albany, NY.
Nivre, Joakim. 2006. Constraints on non-projective dependency parsing. In *EACL 2006 - 11th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference*, pages 73–80.
Nivre, Joakim. 2009. Non-projective dependency parsing in expected linear

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022

time. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, pages 351–359. https://doi.org/10.3115/1687878.1687929

Park, Y. Albert and Roger Park Levy. 2009. Minimal-length linearizations for mildly context-sensitive dependency trees. In *Proceedings of the 10th Annual Meeting of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT) Conference*, pages 335–343. https://doi.org/10.3115/1620754.1620803

Rudnicka, Karolina. 2018. Variation of sentence length across time and genre: Influence on the syntactic usage in English. In Richard J. Whitt, editor, *Diachronic Corpora, Genre, and Language Change*. John Benjamins, pages 219–240. https://doi.org/10.1075/scl.85.10rud

San Diego, Immanuel T. and Frederick S. Gella. 2014. The *b*-chromatic number of bistar graph. *Applied Mathematical Sciences*, 8(116):5795–5800. https://doi.org/10.12988/ams.2014.47529

Shiloach, Yossi. 1979. A minimum linear arrangement algorithm for undirected trees. *SIAM Journal on Computing*, 8(1):15–32. https://doi.org/10.1137/0208002

Sleator, Daniel and Davy Temperley. 1993. Parsing English with a link grammar. In *Proceedings of the Third International Workshop on Parsing Technologies (IWPT93)*, pages 277–292. https://dblp.uni-trier.de/db/journals/corr/corr9508.html#abs-cmp-lg-9508004.

Sloane, Neil James Alexander. 1964–2022. The on-line encyclopedia of integer sequences – number of unlabeled rooted trees. https://oeis.org/A000081. Accessed: 2022-01-20.

Temperley, David and Daniel Gildea. 2018. Minimizing syntactic dependency lengths: Typological/cognitive universal? *Annual Review of Linguistics*, 4(1):67–80. https://doi.org/10.1146/annurev-linguistics-011817-045617

Valiente, Gabriel. 2021. *Algorithms on Trees and Graphs*, second edition. Springer, New York. https://doi.org/10.1007/978-3-030-81885-2

Wang, Dingquan and Jason Eisner. 2018. Synthetic data made to order: The case of parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1325–1337. https://aclanthology.org/D18-1163

Zörnig, Peter. 1984. The distribution of the distance between like elements in a sequence I. *Glottometrika*, 25(6):1–15.

Downloaded from http://direct.mit.edu/coli/article-pdf/48/3/491/2040451/coli_a_00442.pdf by guest on 03 November 2022