The expected sum of edge lengths in planar linearizations of trees

Lluís Alemany-Puig and Ramon Ferrer-i-Cancho
Universitat Politècnica de Catalunya (UPC)

ABSTRACT

Dependency trees have proven to be a very successful model to represent the syntactic structure of sentences of human languages. In these structures, vertices are words and edges connect syntactically-dependent words. The tendency of these dependencies to be short has been demonstrated using random baselines for the sum of the lengths of the edges or their variants. A ubiquitous baseline is the expected sum in projective orderings (wherein edges do not cross and the root word of the sentence is not covered by any edge), that can be computed in time $O(n)$. Here we focus on a weaker formal constraint, namely planarity. In the theoretical domain, we present a characterization of planarity that, given a sentence, yields either the number of planar permutations or an efficient algorithm to generate uniformly random planar permutations of the words. We also show the relationship between the expected sum in planar arrangements and the expected sum in projective arrangements. In the domain of applications, we derive a $O(n)$-time algorithm to calculate the expected value of the sum of edge lengths. We also apply this research to a parallel corpus and find that the gap between actual dependency distance and the random baseline reduces as the strength of the formal constraint on dependency structures increases, suggesting that formal constraints absorb part of the dependency distance minimization effect. Our research paves the way for replicating past research on dependency distance minimization using random planar linearizations as random baseline.

Keywords: *dependency grammar, projectivity, planarity, syntactic dependency distance minimization*

1 INTRODUCTION

The structure of a natural language sentence can be represented as a (labelled) graph indicating the syntactic relationships between words together with the encoding of the words' order. In such a graph, the edge labels indicate the type of syntactic relationship between the words. Such a combination of graph and linear ordering, as in Figure 1, is known as syntactic dependency structure (Nivre 2006). When the graph is (1) *well-formed*, namely, the graph is weakly connected, (2) is *acyclic*, that is, there are no cycles in the graph, (3) is *single-headed*, that is, every node has a single head (except for the root node), and (4) there is only one root node (one node with no head) in the graph, then it is called a syntactic dependency tree (Nivre 2006). There exist *formal constraints* that are often imposed on dependency structures. One such constraint is projectivity: a dependency structure is projective if, for every vertex v, all vertices reachable from v in the underlying graph form a continuous substring within the sentence (Kuhlmann and Nivre 2006). Projectivity implies that (1) the root word of the sentence (the root of the underlying syntactic dependency structure) is never covered (as in Figure 1(a)) and (2) planarity, namely absence of edge crossings (Figure 1 (a) and (b)). Indeed planarity is another constraint that generalizes projectivity by allowing the root to be covered by one or more edges (as in Figure 1(b)). Figure 1(c) shows a sentence that is neither projective nor planar.

In this article, we study statistical properties of syntactic dependency structures under the planarity constraint. Such structures are represented in this article as a pair consisting of a (free or rooted) tree and a linear arrangement of its vertices. Free trees are denoted as $T = (V, E)$, and rooted trees as $T^r = (V, E; r)$, where V is the set of vertices, E the set of edges, and $r \in V$ denotes the root vertex. Unless stated otherwise $n = |V|$, that is, n denotes the number of vertices which is equal to the number of words in the sentence. A linear arrangement π (also called *embedding*) of a tree is a (bijective) function ($\pi : V \to \{1, \dots, n\}$) that maps every vertex u of a tree to a unique position in $\{1, \dots, n\}$, which is denoted by $\pi(u)$.

Projectivity, as well as planarity, can be alternatively defined on linear arrangements using the concept of edge crossing. We say that

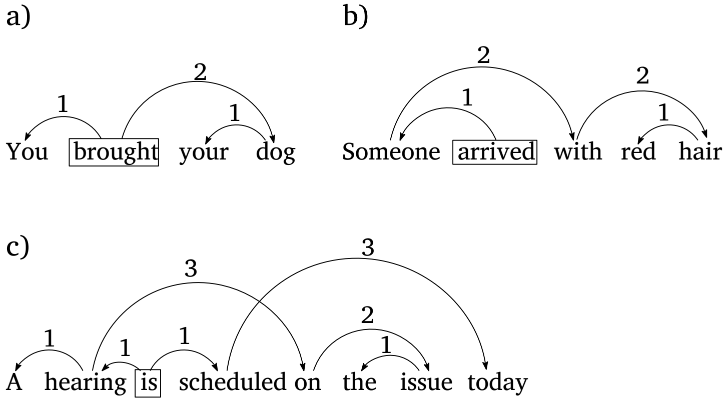


Figure 1: Examples of sentences with their syntactic dependency structures; arc labels indicate dependency distance (in words) between linked words. The rectangles denote the root word in each sentence. a) A projective dependency tree (adapted from Groß and Osborne 2009). b) Planar (but not projective) syntactic dependency structure (adapted from Groß and Osborne 2009). c) Non-projective and non-planar syntactic dependency structure (adapted from Nivre 2009)

any two (undirected) edges $\{s,t\}$, $\{u,v\}$ cross if the positions of their vertices interleave. More formally, assume, without loss of generality, that $\pi(s) < \pi(t)$, $\pi(u) < \pi(v)$ and $\pi(s) < \pi(u)$. Then, edges $\{s,t\}$, $\{u,v\}$ cross in the linear ordering defined by π if $\pi(s) < \pi(u) < \pi(t) < \pi(v)$.[1] We denote the total number of edge crossings in an arrangement π as $C_\pi(T)$. Then, an arrangement π of a rooted tree T^r is *planar* if $C_\pi(T^r) = 0$ and is *projective* if (a) it is planar and (b) the root of the tree is not covered, that is, there is no edge $\{s,t\}$ such that $\pi(s) < \pi(r) < \pi(t)$ or $\pi(t) < \pi(r) < \pi(s)$. Planarity is a relaxation of projectivity where the root can be covered (Sleator and Temperley 1993; Kuhlmann and Nivre 2006). Planar arrangements are also known in the literature as *one-page book embeddings* (Bernhart and Kainen 1979).

In this article, the main object of study is the expectation of the sum of edge lengths (or syntactic dependency distances) in planar arrangements of free trees. The length of an edge connecting two syntactically-related words, also known as dependency distance, is usually[2] defined as the number of intervening words between u and v

---

[1] Notice that this notion of crossing does not depend on edge orientation.

[2] Another popular definition is $\delta_{uv}(\pi) = |\pi(u) - \pi(v)| - 1$ (Liu *et al.* 2017).

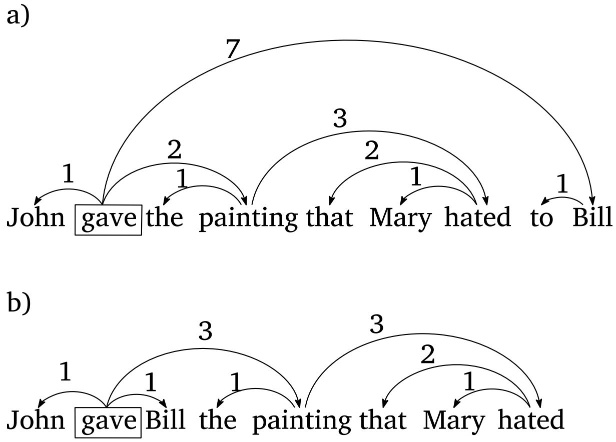


Figure 2: Examples of sentences with their syntactic dependency structures; arc labels indicate dependency distance. The rectangles denote the root word in each sentence. Examples adapted from Morrill 2000. The sum of edge lengths are $D = 18$ for (a) and $D = 12$ for (b)

in the sentence plus 1 (Figure 1). It is defined mathematically as

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

We define the total sum of edge lengths in π as

$$(1) \qquad D_\pi(T) = \sum_{uv \in E} \delta_{uv}(\pi).$$

Close attention has been paid to this metric in modern linguistic research since its causal relationship with cognitive cost was first put forward, to the best of our knowledge, by Hudson 1995. The main causal argument is that the longer the dependency, the greater the memory burden arising from decay of activation and interference (Hudson 1995; Liu *et al.* 2017). A number of studies have exposed the general tendency in languages to reduce D, the total sum of edge lengths, a reflection of a potentially universal cognitive force known as the Dependency Distance Minimization principle (DDm) (Ferrer-i-Cancho 2004; Liu 2008; Futrell *et al.* 2015; Liu *et al.* 2017; Ferrer-i-Cancho *et al.* 2022). As an example of such cognitive cost, consider the sentences in Figures 2(a) and 2(b): it is not surprising that the latter is preferred over the former due to smaller total sum of edge lengths (Morrill 2000), the former's being $D = 18$ and the latter's being $D = 12$.

Statistical evidence of the DDm principle has been provided showing that dependency distances are smaller than expected by chance in syntactic dependency treebanks (Ferrer-i-Cancho 2004; Liu 2008; Park and Levy 2009; Gildea and Temperley 2010; Futrell *et al.* 2015; Liu *et al.* 2017; Ferrer-i-Cancho *et al.* 2022; Kramer 2021). Typically, the random baseline is defined as a random shuffling of the words of a sentence. To the best of our knowledge, the first known instance of such an approach was done by Ferrer-i-Cancho 2004, who established the DDm principle by comparing the average real $D(T)$ of sentences against the corresponding expected value in a uniformly random permutation of sentences' words. More formally, Ferrer-i-Cancho 2004 calculated the expected value of $D(T)$ when the words of the sentence are shuffled uniformly at random (u.a.r.), that is, when all $n!$ permutations are equally likely. This value is denoted here as $\mathbb{E}\left[D(T)\right]$. Ferrer-i-Cancho 2004 found that

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

In spite of the simplicity of Equation 2, the majority of researchers have used as random baseline the expected sum of edge lengths conditioned to projective arrangements (Temperley 2008; Park and Levy 2009; Gildea and Temperley 2010; Futrell *et al.* 2015; Kramer 2021) which we denote here as $\mathbb{E}_{\mathrm{pr}}\left[D(T^r)\right]$. However, this baseline has been computed approximately via random sampling of projective arrangements. For these reasons, a formula to calculate the exact value of $\mathbb{E}_{\mathrm{pr}}\left[D(T^r)\right]$ in linear time was derived by Alemany-Puig and Ferrer-i-Cancho 2022

(3) $$\mathbb{E}_{\mathrm{pr}}\left[D(T^r)\right] = \frac{1}{6}\sum_{u \in V} s_r(u)(2d_r(u)+1) - \frac{1}{6},$$

where $s_r(u)$ denotes the size (in vertices) of the subtree of T^r rooted at u, and $d_r(u)$ is the out-degree of u in T^r. In spite of its extensive use, the projective random baseline has some limitations. First, the percentage of non-projective sentences in languages ranges between 18.2 and 26.4 (Gómez-Rodríguez 2016) or between 6.8 and 36.4 (Gómez-Rodríguez and g 2010) (see also Havelka 2007). The limited coverage of projectivity raises the question if the projective baseline should be used for sentences that are not projective as it is customary in research on dependency distance minimization. In addition, projectivity *per se* implies a reduction in dependency distances, which raises the question if that rather strong constraint may mask the effect of the dependency distance minimization principle under investigation (Gómez-

Rodríguez *et al.* 2022). Here we aim to make a step forward by considering planarity, a generalization of projectivity, so as to increase the coverage of real sentences and reduce the bias towards dependency minimization in the random baseline. The percentage of non-planar sentences in languages ranges between 14.3 and 20.0 (Ferrer-i-Cancho *et al.* 2018) or between 5.3 and 31 (Gómez-Rodríguez and g 2010). The latter range is consistent with earlier estimates (Havelka 2007).

This article is part of a research program on the statistical properties of $D(T)$ under constraints on the possible linear arrangements (Ferrer-i-Cancho 2019; Alemany-Puig *et al.* 2022; Alemany-Puig and Ferrer-i-Cancho 2022). The remainder of the article is divided into two main parts: theory (Section 2) and applications (Section 3).

The theory part (Section 2) is structured as follows. In Section 2.1, we introduce notation used throughout that part. In Section 2.2, we first present a characterization of planar arrangements so as to identify their underlying structure, which we apply to count their number for a given free tree, and later on in Section 2.3, to generate them u.a.r. by means of a novel $O(n)$-time algorithm. In Section 2.4, we use said characterization to prove the main result of the article, namely that expectation of $D(T)$ in planar arrangements can be calculated from the expectation of projective arrangements, as the following theorem indicates.

THEOREM 1 *Given a free tree $T=(V,E)$,*

$$\mathbb{E}_{\text{pl}}[D(T)] = \frac{1}{n}\sum_{u\in V}\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)] \tag{4}$$

$$= \frac{(n-1)(n-2)}{6n} + \frac{1}{n}\sum_{u\in V}\mathbb{E}_{\text{pr}}[D(T^u)], \tag{5}$$

where $\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)]$ is the expected value of $D(T^u)$ in uniformly random projective arrangements π of T^u such that $\pi(u)=1$ and $\mathbb{E}_{\text{pr}}[D(T^u)]$ (Equation 3) is the expected value of $D(T^u)$ in uniformly random projective arrangements of T^u, the free tree T rooted at u.

Table 1 summarizes the theoretical results obtained in previous articles and those presented in this article.

The applications part (Section 3) is structured as follows. In Section 3.1, we apply Theorem 1 to derive a $O(n)$-time algorithm to calculate $\mathbb{E}_{\text{pl}}[D(T)]$. Since Alemany-Puig and Ferrer-i-Cancho 2022

Table 1: Summary of the main mathematical results for increasing constraints on linear orders. Results for the unconstrained and projective cases are borrowed from previous research (Ferrer-i-Cancho 2004 and Alemany-Puig and Ferrer-i-Cancho 2022, respectively). Results for the planar case are a contribution of this article. $\mathbf{N}_{\mathbf{pr}}(T^r)$, $\mathbf{N}_{\mathbf{pl}}(T)$ and $\mathbf{N}(T)$ denote the number of distinct projective, planar and unconstrained linear arrangements, respectively, of a rooted tree T^r or of a free tree T. $\mathbb{E}_{\text{pr}}[\delta_{uv}]$, $\mathbb{E}_{\text{pl}}[\delta_{uv}]$ and $\mathbb{E}[\delta_{uv}]$ denote the expected length of an edge in random linear arrangement for the projective, planar and unconstrained cases, respectively. $\mathbb{E}_{\text{pr}}[\delta_{uv} \mid s]$ is the expected value of δ_{uv} conditioned to having vertex s as root of the tree. In $\mathbb{E}_{\text{pr}}[\delta_{uv}]$ the root is vertex r

Unconstrained (T)	$\mathbf{N}(T)$	$n!$
	$\mathbb{E}[\delta_{uv}]$	$\frac{n+1}{3}$
	$\mathbb{E}[D(T)]$	$\frac{n^2-1}{3}$
Planar (T)	$\mathbf{N}_{\mathbf{pl}}(T)$	$n\prod_{u\in V} d(u)!$
	$\mathbb{E}_{\text{pl}}[\delta_{uv}]$	$1+\frac{1}{n}\sum_{s\in V\setminus\{u,v\}} \mathbb{E}_{\text{pr}}[\delta_{uv} \mid s]$
	$\mathbb{E}_{\text{pl}}[D(T)]$	$\frac{(n-1)(n-2)}{6n}+\frac{1}{n}\sum_{u\in V}\mathbb{E}_{\text{pr}}[D(T^u)]$
Projective (T^r)	$\mathbf{N}_{\mathbf{pr}}(T^r)$	$\prod_{u\in V}(d_r(u)+1)!$
	$\mathbb{E}_{\text{pr}}[\delta_{uv}]$	$\frac{1}{6}(2s_r(u)+s_r(v)+1)$
	$\mathbb{E}_{\text{pr}}[D(T^r)]$	$\frac{1}{6}\left(-1+\sum_{v\in V}s_r(v)(2d_r(v)+1)\right)$

showed that $\mathbb{E}_{\text{pr}}[D(T^r)]$ can be evaluated in time $O(n)$, Equation 5 naturally leads to a $O(n^2)$-time algorithm if it is evaluated 'as is'. However, we devise a $O(n)$-time algorithm to calculate $\mathbb{E}_{\text{pl}}[D(T)]$. In Section 3.2, we apply this and previous research on the projective case (Alemany-Puig and Ferrer-i-Cancho 2022) to a parallel syntactic dependency treebank. We find that the gap between the actual dependency distance and that of the random baseline reduces as the strength of the formal constraint on dependency structures chosen for the random baseline increases, suggesting that formal constraints absorb part of the dependency distance minimization effect.

Finally, in Section 4, we review all the findings and make suggestions for future research.

From this point onwards, the article is organized to ease reading by readers of distinct profiles. Readers interested in the analysis of syntactic dependency treebanks can jump directly to Section 3.2. Readers interested in the algorithm for computing $\mathbb{E}_{\text{pl}}[D(T)]$ can jump directly to Section 3.1, after reading Section 2.1. Readers whose primary interest is applying the algorithms have ready-to-use code: both methods to generate planar arrangements (Section 2.3) and the $O(n)$-time calculation of $\mathbb{E}_{\text{pl}}[D(T)]$ (Section 3.1) are freely available in the Linear Arrangement Library[3] (Alemany-Puig *et al.* 2021).

2 THEORY

2.1 *Definitions and notation*

We use u, v, w, z to denote vertices, r to always denote a root vertex, and i, j, k, p, q to denote integers. The edges of a free tree are undirected, and denoted as $\{u, v\} = uv$; those of rooted trees are directed, denoted as (u, v), and oriented away from r towards the leaves.

Let $\Gamma(u)$ denote the set of neighbors of $u \in V$ in the free tree T, and let $\Gamma_r(u)$ denote the out neighbors (also, children) of $u \in V$ in T^r. Notice that, $\Gamma_r(u) \subseteq \Gamma(u)$ with equality if, and only if $u = r$. Let $d_r(u) = |\Gamma_r(u)|$ denote the out-degree of vertex u of a rooted tree T^r, and let $d(u) = |\Gamma(u)|$ denote the degree of u in a free tree T. Notice that $d_r(u) = d(u) - 1$ when $u \neq r$ and $d_r(r) = d(r)$. Furthermore, we denote the subtree rooted at v with respect to root u as T^u_v (obviously $T^r_r = T^r$), and its size as $s_u(v) = |V(T^u_v)|$ (Figure 3). We call this *directional* size (Hochberg and Stallmann 2003; Alemany-Puig *et al.* 2022). Note that $s_v(u) + s_u(v) = n$ for any $uv \in E$.

[3] `https://github.com/LAL-project/linear-arrangement-library/`

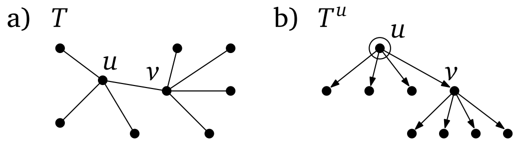


Figure 3: a) A free tree T, where $d(u) = 4$, and $d(v) = 5$; in this tree, $s_u(v) = 5$ and $s_v(u) = 4$. b) The free tree T rooted at u, denoted as T^u, where $d_u(u) = d_{T^u}(v) = d(u) = 4$, and where $4 = d_u(v) = d_{T^u}(v) < d(v) = 5$. Figure borrowed from Hochberg and Stallmann 2003 and Alemany-Puig *et al.* 2022

As in previous research, we also decompose an edge (r, u) in a projective arrangement π into two parts: its *anchor* and its *coanchor*, as in Figure 4 (Shiloach 1979; Chung 1984; Alemany-Puig and Ferrer-i-Cancho 2022). Informally, $\alpha_{ru}(\pi)$ is the number of vertices in π covered by (r, u) in the segment of T^r_u including vertex u (Figure 4); similarly, $\beta_{ru}(\pi)$, is the number of vertices of π covered by (r, u) in segments that fall between r and u (Figure 4). The length of an edge connecting r with u can be expressed with the formula

$$\delta_{ru}(\pi) = |\pi(r) - \pi(u)| = \alpha_{ru}(\pi) + \beta_{ru}(\pi),$$

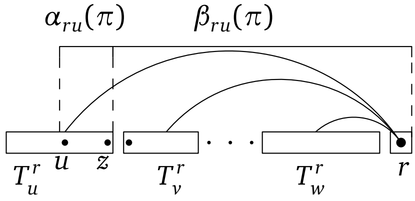
where $\alpha_{ru}(\pi)$ is the length of the anchor and $\beta_{ru}(\pi)$ is the length of the coanchor. The length of the anchor and coanchor can be formally defined as

$$\begin{aligned}\alpha_{ru}(\pi) &= |\pi(u) - \pi(z)| + 1\\ \beta_{ru}(\pi) &= |\pi(z) - \pi(r)| - 1,\end{aligned}$$

where $z \in V(T^r_u)$ is the vertex of T^r_u closest to r in π (Figure 4). The same notation with π omitted, α_{ru} and β_{ru} denote random variables. Furthermore, it will be useful to define the operator $\diamond$, which we use to condition expected values and constrain sets of arrangements of a rooted tree, in both cases to arrangements π where (only) the root is fixed at the leftmost position of π. For instance, if S is a set of arrangements π of a rooted tree T^r then $S^\diamond = \{\pi \in S \mid \pi(r) = 1\}$. Moreover, if X is defined on uniformly random arrangements from S then $\mathbb{E}^\diamond[X]$ is the expected value of X in uniformly random arrangements from $S^\diamond$.

Finally, in this article we consider that two arrangements π and π' of the same tree T are *different* if there is (at least) one vertex u for which $\pi(u) \neq \pi'(u)$.

Figure 4:
Illustration of an edge's anchor $\alpha_{ru}(\pi)$ and coanchor $\beta_{ru}(\pi)$. In this figure, $u, v, w \in \Gamma(r)$. Figure adapted from Alemany-Puig and Ferrer-i-Cancho 2022



2.2 *Counting planar arrangements*

It is well known that the number of unconstrained arrangements of an n-vertex tree is $n!$. This is true given that arrangements are simply permutations, and unconstrained arrangements are not subject to any particular constraint, thus all vertex orderings are possible. Building on the fact that projective arrangements span over contiguous intervals (Kuhlmann and Nivre 2006), Alemany-Puig and Ferrer-i-Cancho 2022 studied the expected value of the random variable $D(T^r)$ in such arrangements by defining, as usual, a set of *segments* Φ_u associated to each vertex u, consisting of the segments associated to the subtrees $T^r_{u_1}, \dots, T^r_{u_p}$ and u. A *segment* of a rooted tree T^r_u is a segment within the linear ordering containing all vertices of T^r_u, an interval of length $s_r(u)$ 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 (Alemany-Puig and Ferrer-i-Cancho 2022). For a vertex u, the set Φ_u is constructed from vertex u's segment and the segments of its children $\Gamma_r(u) = \{u_1, \dots, u_k\}$ (Figure 5). Decomposing every vertex and its segments from the root to the leaves linearizes T^r into a projective arrangement (Figure 5). This characterization led to a straightforward derivation of the number of projective arrangements of a rooted tree T^r (Table 1)

(6) $$\mathbf{N}_{\mathbf{pr}}(T^r) = \prod_{u \in V} (d_r(u) + 1)!.$$

Using the structure of segments summarized above, we present a characterization of planar arrangements of free trees which helps to devise a method to generate planar arrangements u.a.r. (Section 2.3.3) and to prove Theorem 1 (Section 2.4). To this aim, we define $\mathbf{P}^{\diamond}_{\mathbf{pr}}(T^r)$ as the set of projective arrangements of a rooted tree T^r such that $\pi(r) = 1$, and denote its size as $\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^r) = |\mathbf{P}^{\diamond}_{\mathbf{pr}}(T^r)|$. Notice that

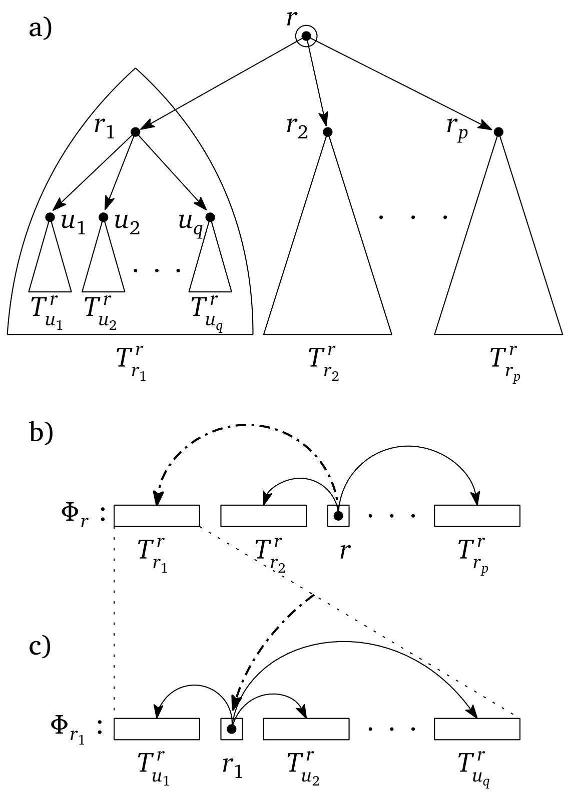


Figure 5:
a) A rooted tree T^r where $\Gamma(r) = \{r_1, \ldots, r_p\}$ are the p children of r. The subtree $T^r_{r_1}$ has been circled for clarity. b) An example of a permutation of the segments in Φ_r associated to the root. c) An example of a permutation of the segments in Φ_{r_1} associated to r_1, the segment at the leftmost position in the example in (b). The dash-dotted edge in (b) and in (c) represent the same edge of the tree. In (b) and (c), respectively, r and r_1 are segments of length 1

when a vertex u is fixed to the leftmost position, the planar arrangements in $\mathbf{P}^{\diamond}_{\mathbf{pr}}(T^u)$ are obtained by arranging the subtrees T^u_v, $v \in \Gamma(u)$, projectively to the right of u in the linear arrangement. It is important to bear in mind that the operator $\diamond$ only fixes the root vertex r to the leftmost position of the arrangement: the other vertices can be placed freely as long as the result is projective.

PROPOSITION 1 *The number of planar arrangements of an n-vertex free tree $T = (V, E)$, with $V = \{u_1, \cdots, u_n\}$ is*

$$\text{(7)} \qquad \mathbf{N}_{\mathbf{pl}}(T) = n\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^{u_1}) = \cdots = n\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^{u_n}) = n \prod_{u \in V} d(u)!.$$

PROOF Given a free tree T, and any two distinct vertices u, v, it holds that $\mathbf{P}^{\diamond}_{\mathbf{pr}}(T^u) \cap \mathbf{P}^{\diamond}_{\mathbf{pr}}(T^v) = \emptyset$ because the vertices in the first positions are different. This lets us partition $\mathbf{P}_{\mathbf{pl}}(T)$ into the non-empty pairwise-disjoint sets $\mathbf{P}^{\diamond}_{\mathbf{pr}}(T^u)$ and see that

$$\mathbf{N}_{\mathbf{pl}}(T) = \sum_{u\in V} \mathbf{N}^{\diamond}_{\mathbf{pr}}(T^u).$$

It is easy to see that

$$\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^u) = d(u)! \prod_{v\in\Gamma(u)} \mathbf{N}_{\mathbf{pr}}(T^u_v) = \prod_{v\in V} d(v)!.$$

We used Equation 6 in the second equality. Notice that

$$\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^{u_1}) = \cdots = \mathbf{N}^{\diamond}_{\mathbf{pr}}(T^{u_n}),$$

since the value $\mathbf{N}^{\diamond}_{\mathbf{pr}}(T^u)$ does not depend on the root vertex u. Therefore, Equation 7 follows immediately. □

Obviously, there are more planar arrangements of a free tree T than projective arrangements of any 'rooting' T^r of T, formally $\mathbf{N}_{\mathbf{pl}}(T) \geq \mathbf{N}_{\mathbf{pr}}(T^r)$. We can see this by noticing that, when given a 'rooting' of T at $r \in V$,

$$\frac{\mathbf{N}_{\mathbf{pl}}(T)}{\mathbf{N}_{\mathbf{pr}}(T^r)} = \frac{nd(r)! \prod_{u\in V\setminus\{r\}} d(u)!}{(d(r)+1)! \prod_{u\in V\setminus\{r\}} d(u)!} = \frac{n}{d(r)+1} \geq 1,$$

with equality when T is a star tree[4] and r is its vertex of highest degree.

2.3 *Generating arrangements uniformly at random*

Arrangements can be generated freely, that is, by imposing no constraint on the possible orderings, where all the $n!$ possible orderings are equally likely, or by imposing some constraint on the possible orderings. Generating unconstrained arrangements is straightforward: it is well known that a permutation of n elements can be generated u.a.r. in time $O(n)$ (Cormen *et al.* 2001). It can be done as follows. Assume we are given a set of n vertices, say $V = \{u_1, \dots, u_n\}$, and let $i = 1$. Repeat the following steps n times:

[4] An n-vertex star tree consists of a vertex connected to $n-1$ leaves; it is also a complete bipartite graph $K_{1,n-1}$.

1. select u.a.r. a vertex from V; the vertex is chosen with probability $1/(n-i+1)$. Let u_i be said vertex,
2. place u_i in the arrangement at position i, that is, let $\pi(u_i) = i$,
3. remove u_i from V,
4. increment i by 1.

The product of all probabilities of vertex choice gives that the probability of producing a certain linear arrangement is

$$\prod_{i=1}^{n} \frac{1}{n-i+1} = \frac{1}{n!}$$

thus the arrangement is constructed uniformly at random. Since the removal of a vertex from the set and uniformly random choice of vertex can both be implemented in constant time (using arrays), the running time is $O(n)$.

When constraints are involved, projectivity is often the preferred choice (Gildea and Temperley 2007; Liu 2008; Futrell *et al.* 2015). First, we present a $O(n)$-time procedure to generate projective arrangements u.a.r. (Section 2.3.1) and review methods used in past research (Section 2.3.2). Then we present a novel $O(n)$-time procedure to generate planar arrangements u.a.r. (Section 2.3.3) which in turn involves the generation of random projective arrangements of a subtree.

Generating projective arrangements 2.3.1

The method we will present in detail here was outlined first by Futrell *et al.* 2015. Here we borrow from recent theoretical research summarized above (Alemany-Puig and Ferrer-i-Cancho 2022) to derive a detailed algorithm to generate projective arrangements and prove its correctness. In order to generate projective arrangements u.a.r., simply make random permutations of a vertex u and its children $\Gamma_r(u)$, that is, choose one of the possible $(d_r(u)+1)!$ permutations u.a.r. Algorithm 1 formalizes this brief description. The proof that Algorithm 1 produces projective arrangements of a rooted tree T^r u.a.r. is simple. The first call takes the root and its dependents and produces a uniformly random permutation with probability $1/(d(r)+1)!$. Subsequent recursive calls (in Algorithm 2) produce the corresponding

Algorithm 1: Generating projective arrangements u.a.r

Function RANDOM_PROJECTIVE_ARRANGEMENT(T^r) **is**
  Input: T^r a rooted tree.
  Output: A projective arrangement π of T^r chosen u.a.r.
  $\pi \leftarrow$ empty n-vertex arrangement
  `// Algorithm 2`
  RANDOM_PROJECTIVE_ARRANGEMENT_SUBTREE($T^r, r, 1, \pi$)
  return π

Algorithm 2: Generating projective arrangements u.a.r. of a subtree

Function RANDOM_PROJECTIVE_ARRANGEMENT_SUBTREE(T^r, u, p, π) **is**
  Input: T^r a rooted tree, u any vertex of T^r, p the starting position to arrange the vertices of T^r_u, π partially-constructed without T^r_u.
  Output: π partially-constructed with T^r_u.
  $\Phi_u \leftarrow$ a random permutation of $\Gamma_r(u) \cup \{u\}$
  for $v \in \Phi_u$ **do**
    if $v = u$ **then**
      $\pi(v) \leftarrow p$
      $p \leftarrow p + 1$
    else
      RANDOM_PROJECTIVE_ARRANGEMENT_SUBTREE(T^r, v, p, π)
      $p \leftarrow p + s_r(v)$

permutations each with its respective uniform probability, hence the probability of producing a particular permutation is the product of individual probabilities. Using Equation 6, we easily obtain that the probability of producing a certain projective arrangement is

$$\prod_{u \in V} \frac{1}{(d_r(u)+1)!} = \frac{1}{\mathbf{N_{pr}}(T^r)}.$$

2.3.2 Generation of projective arrangements in past research

Algorithm 1 is equivalent to the "fully random" method used by Futrell *et al.* 2015 as witnessed by the implementation of their code available on Github,[5] in particular in file `cliqs/mindep.py`[6] (function `_randlin_projective`). Notice that Futrell *et al.* 2015 outline

[5] `https://github.com/Futrell/cliqs/tree/44bfcf2c42c848243c264722b5eccdffec0ede6a`

[6] `https://github.com/Futrell/cliqs/blob/44bfcf2c42c848243c264722b5eccdffec0ede6a/cliqs/mindep.py`

(though vaguely) that a projective arrangement is generated randomly by "Starting at the root node of a dependency tree, collect[ing] the head word and its dependents and order[ing] them randomly".

Futrell *et al.* 2015 present their method to generate random projective arrangements as though it were the same as that by Gildea and Temperley 2007, 2010, who introduced a method to generate random linearizations of a tree which consists of "choosing a random branching direction for each dependent of each head,[7] and – in the case of multiple dependents on the same side – randomly ordering them in relation to the head" (Gildea and Temperley 2010). However, Futrell *et al.* 2015 do not actually implement Gildea and Temperley's method as witnessed by their code. Critically, Gildea and Temperley's method does not produce uniformly random linearizations as we show with a counterexample.

Consider a star tree rooted at its hub. Let X be a random variable for the position of the root in a random projective linear arrangement ($1 \leq X \leq n$). We have $\mathbb{P}(X = x) = 1/n$ for all $x \in [1, n]$, therefore X follows a uniform distribution and hence $\mathbb{E}[X] = (n+1)/2$ and $\mathbb{V}[X] = (n^2 - 1)/12$ (Mitzenmacher and Upfal 2017). Let X' be a random variable for the position of the root according to Gildea and Temperley's method. It is easy to see that $X' - 1$ follows a binomial distribution with parameters $n-1$ and $1/2$. Namely, $\mathbb{P}(X' - 1 = x) = \binom{n-1}{x}/2^{n-1}$. We have that $\mathbb{E}[X'] = 1 + \mathbb{E}[X' - 1] = (n+1)/2 = \mathbb{E}[X]$, but $\mathbb{V}[X'] = \mathbb{V}[X' - 1] = (n-1)/4$. Therefore, the variance in a truly uniformly random projective linear arrangement is $\Theta(n^2)$ while Gildea and Temperley's method results in $\Theta(n)$, a much smaller dispersion. As $n \to \infty$, $X' - 1$ converges to a Gaussian distribution.

Gildea and Temperley's method was introduced as a random baseline for the distance between syntactically-related words in languages and has been used with that purpose (Gildea and Temperley 2007, 2010; Temperley and Gildea 2018). Interestingly, the minimum baseline, namely, the minimum sum of dependency distances, results from placing the root at the center (Shiloach 1979; Chung 1984). The example above shows that Gildea and Temperley's baseline tends to put the root at the center of the linear arrangement

[7] That is, as explained by Temperley and Gildea 2018, "choose a random assignment of each dependent to either the left or the right of its head."

with higher probability than the truly uniform baseline. That behavior casts doubts on the power of that random baseline to investigate dependency distance minimization in languages since it tends to place the root at the center of the sentence, as expected from an optimal placement under projectivity (Gildea and Temperley 2007; Alemany-Puig *et al.* 2021) and does it with much lower dispersion around the center than in truly uniformly random linearizations.

2.3.3 Generating planar arrangements

Proposition 1 leads to a method to generate planar arrangements u.a.r. for any free tree T. The method we propose is detailed in Algorithm 3.

Algorithm 3: Generating planar arrangements u.a.r.

Function RANDOM_PLANAR_ARRANGEMENT(T) **is**
    Input: T a free tree.
    Output: A planar arrangement π of T chosen u.a.r.
    $\pi \leftarrow$ empty n-vertex arrangement
    $u \leftarrow$ a vertex of T chosen u.a.r.
    $\pi(u) \leftarrow 1$
    $\Phi_u \leftarrow$ a random permutation of $\Gamma(u)$
    $p \leftarrow 2$
    for $v \in \Phi_u$ **do**
        `// Algorithm 2`
        RANDOM_PROJECTIVE_ARRANGEMENT_SUBTREE(T^u, v, p, π)
        $p \leftarrow p + s_u(v)$
    return π

It is easy to see that Algorithm 3 has time complexity $O(n)$. Now we show that it generates planar arrangements uniformly at random. Firstly, choose a vertex, say $u \in V$, u.a.r., and place it at one of the arrangement's ends, say, the leftmost position; this vertex acts as a root for T. Secondly, choose u.a.r. one of the $d(u)!$ permutations of the segments of the subtrees T^u_v u.a.r. Lastly, recursively choose u.a.r. a projective linearization of every subtree T^u_v for $v \in \Gamma(u)$ (Algorithm 2). These steps generate a planar arrangement u.a.r. since the probability of producing a certain planar arrangement following these steps is, then,

$$\frac{1}{n}\frac{1}{d(u)!}\prod_{v\in\Gamma(u)}\frac{1}{\mathbf{N_{pr}}(T^u_v)} = \frac{1}{n}\frac{1}{d(u)!}\prod_{v\in V\setminus\{u\}}\frac{1}{d(v)!} = \frac{1}{\mathbf{N_{pl}}(T)}.$$

The equalities follow from Proposition 1.

Expected sum of edge lengths 2.4

In this section we derive an arithmetic expression for $\mathbb{E}_{\text{pl}}[D(T)]$. First, we prove Theorem 1. To this aim, we define $\mathbb{E}^{\diamond}_{\text{pr}}[\alpha_{uv} \mid r] = \mathbb{E}_{\text{pr}}[\alpha_{uv} \mid \pi(r) = 1]$ as the expected value of α_{uv} conditioned to the projective arrangements π of T^r such that $\pi(r) = 1$; we define $\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid r]$ likewise. The root is specified as a parameter of the expected value because we want to be able to use various roots. In the following proofs we rely heavily on Linearity of Expectation (Mitzenmacher and Upfal 2017, Theorem 2.1) and the Law of Total Expectation (Mitzenmacher and Upfal 2017, Lemma 2.5).

PROOF [Proof of Theorem 1] We first prove Equation 4. By the Law of Total Expectation,

$$\mathbb{E}_{\text{pl}}[D(T)] = \sum_{u \in V} \mathbb{E}_{\text{pl}}[D(T) \mid \pi(u) = 1] \, \mathbb{P}_{\text{pl}}(\pi(u) = 1).$$

Notice, quite simply, that

$$\mathbb{E}_{\text{pl}}[D(T) \mid \pi(u) = 1] = \mathbb{E}_{\text{pr}}[D(T^u) \mid \pi(u) = 1] = \mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)],$$

that is, the expected value of D conditioned to planar arrangements of T such that u is fixed at the leftmost position, $\mathbb{E}_{\text{pl}}[D(T) \mid \pi(u) = 1]$, is equal to the expected value of D conditioned to projective arrangements of T^u such that vertex u is fixed at the leftmost position, which is denoted as $\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)]$. By noticing, given a fixed vertex u, that $\mathbb{P}_{\text{pl}}(\pi(u) = 1) = \frac{1}{n}$, which is the proportion of planar arrangements of T in which $\pi(u) = 1$ (Proposition 1), Equation 4 follows immediately. Notice that Equation 4 expresses the expected value of D conditioned to planar arrangements of a free tree T as the average of each of the expected values of D conditioned to projective arrangements of T^u (for all $u \in V$) such that the root is fixed at the leftmost position.

Now we aim to write $\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)]$ as a function of $\mathbb{E}_{\text{pr}}[D(T^u)]$. We start by decomposing $\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)]$ into a summation of expected values of the individual edge lengths, and group the edges of every subtree T^u_v of T^u (where uv is a (directed) edge of the tree) into one single expected value for each subtree and leave the edges incident to the root u in the same summation as follows:

$$\mathbb{E}^{\diamond}_{\text{pr}}[D(T^u)] = \sum_{vw \in \Gamma(u)} \left(\mathbb{E}^{\diamond}_{\text{pr}}[\delta_{vw} \mid u] + \mathbb{E}_{\text{pr}}\left[D(T^u_v)\right] \right).$$

Now, it is important to notice that we did not write $\mathbb{E}^{\diamond}_{\mathrm{pr}}\left[D(T^u_v)\right]$ in the summation above since the conditioning imposed by the operator $\diamond$ in $\mathbb{E}^{\diamond}_{\mathrm{pr}}[D(T^u)]$ only applies to the root u. The root of the subtrees can be placed freely in the arrangement as long as the result is projective. Now we decompose all (directed) edges uv of T^r in the first summation into anchor and coanchor, and we get

$$\mathbb{E}^{\diamond}_{\mathrm{pr}}[D(T^u)] = \sum_{v\in\Gamma(u)} \left(\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} + \beta_{uv} \mid u] + \mathbb{E}_{\mathrm{pr}}\left[D(T^u_v)\right]\right).$$

Although the root u is clear in this context, we have made it explicit in $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} + \beta_{uv} \mid u]$ so as to be able to keep track of it in the following derivations. By linearity of expectation,

$$\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} + \beta_{uv} \mid u] = \mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid u] + \mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid u].$$

Now, notice that the length of the anchor of any given directed edge (u,v), where u is the head and v is the dependent, is invariant to the position of u, that is, it only changes if we change the position of v within its interval. Therefore, fixing the head to the leftmost position of the arrangement (or any position outside the segment of v) does not affect the value of $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid u]$ and we simply have that $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid u] = \mathbb{E}_{\mathrm{pr}}[\alpha_{uv} \mid u]$ and thus

$$\begin{aligned}\mathbb{E}^{\diamond}_{\mathrm{pr}}[D(T^u)] = \sum_{v\in\Gamma(u)} \big(&\mathbb{E}_{\mathrm{pr}}[\alpha_{uv} \mid u] + \mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid u] \\ &+ \mathbb{E}_{\mathrm{pr}}\left[D(T^u_v)\right]\big).\end{aligned}$$

The next step is to find the value of $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid u]$. Notice now that the length of the coanchor of any directed edge (u,v) *is* affected by the position of the head u and, as such, $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid u]$ need not be exactly equal to $\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u]$. The derivation is found in Appendix 4.3 since it is merely an adaptation of the proof by Alemany-Puig and Ferrer-i-Cancho 2022, Lemma 1; it gives

$$\mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid u] = \frac{3}{2}\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u].$$

Thus,

$$\begin{aligned}\mathbb{E}^{\diamond}_{\mathrm{pr}}[D(T^u)] &= \sum_{v\in\Gamma(u)} \Big(\mathbb{E}_{\mathrm{pr}}[\alpha_{uv} \mid u] + \frac{3}{2}\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u] \\ &\qquad + \mathbb{E}_{\mathrm{pr}}\left[D(T^u_v)\right]\Big) \\ &= \sum_{v\in\Gamma(u)} \Big(\mathbb{E}_{\mathrm{pr}}[\delta_{uv} \mid u] + \mathbb{E}_{\mathrm{pr}}\left[D(T^u_v)\right] \\ &\qquad + \frac{1}{2}\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u]\Big) \\ &= \mathbb{E}_{\mathrm{pr}}[D(T^u)] + \frac{1}{2}\sum_{v\in\Gamma(u)} \mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u]. \qquad (8)\end{aligned}$$

In the third equality we have used the identity by Alemany-Puig and Ferrer-i-Cancho 2022, Equation 28, which states that in a rooted tree T^r

$$\mathbb{E}_{\mathrm{pr}}[D(T^r)] = \sum_{v\in\Gamma(r)} \Big(\mathbb{E}_{\mathrm{pr}}[\delta_{rv}] + \mathbb{E}_{\mathrm{pr}}\left[D(T^r_v)\right]\Big).$$

In this equation, we have not specified the expected values as being conditioned by the root r since this is clear from the context. Plugging Equation 8 into Equation 4 we get

$$\mathbb{E}_{\mathrm{pl}}[D(T)] = \frac{1}{2n}\sum_{u\in V}\sum_{v\in\Gamma(u)} \mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u] + \frac{1}{n}\sum_{u\in V}\mathbb{E}_{\mathrm{pr}}[D(T^u)]. \qquad (9)$$

We can use the following result by Alemany-Puig and Ferrer-i-Cancho 2022, Equation 16:

$$\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u] = \frac{s_u(u) - s_u(v) - 1}{3} = \frac{n - s_u(v) - 1}{3}$$

to further simplify Equation 9 and, after proving that

$$\begin{aligned}\sum_{v\in\Gamma(u)} \mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid u] &= \sum_{v\in\Gamma(u)} \frac{s_u(u) - s_u(v) - 1}{3} \\ &= \frac{(n-1)(d(u)-1)}{3},\end{aligned}$$

$$\sum_{u\in V} \frac{1}{3}(n-1)(d(u)-1) = \frac{(n-1)(n-2)}{3},$$

we obtain

$$\frac{1}{2n}\sum_{u\in V}\sum_{v\in\Gamma(u)}\mathbb{E}_{\mathrm{pr}}[\beta_{uv}\mid u]=\frac{(n-1)(n-2)}{6n}. \tag{10}$$

Hence Equation 5. □

For the sake of comprehensiveness, we also provide an arithmetic expression for the expected length of an edge uv of a free tree in uniformly random planar arrangements. To this aim, we further define $\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv}\mid r]=\mathbb{E}_{\mathrm{pl}}[\delta_{uv}\mid \pi(r)=1]$ to be the expected value of the length of edge $uv\in E(T)$ when the vertex $r\in V(T)$ is fixed to the leftmost position in planar arrangements of T. Similarly, given a rooting of T at r, let $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\delta_{uv}\mid r]=\mathbb{E}_{\mathrm{pr}}[\delta_{uv}\mid \pi(r)=1]$ to be the expected value of the length of edge $uv\in E(T^r)$ when vertex r acts as the root of the tree and it is fixed to the leftmost position in projective arrangements of T^r. The root vertex r may be vertex u, vertex v, or neither. In the expected value $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\delta_{uv}\mid r]$ we assume that the edge uv is directed from u to v in accordance with the orientation defined by the root vertex r. Therefore, when r is neither u nor v, the vertex of edge uv closest to r is always vertex u, and the farthest is always vertex v.

LEMMA 2 *Given a free tree $T=(V,E)$, for any $uv\in E$ it holds that*

$$\mathbb{E}_{\mathrm{pl}}[\delta_{uv}]=1+\frac{1}{n}\sum_{r\in V\setminus\{u,v\}}\mathbb{E}_{\mathrm{pr}}[\delta_{uv}\mid r], \tag{11}$$

where as per Alemany-Puig and Ferrer-i-Cancho 2022

$$\mathbb{E}_{\mathrm{pr}}[\delta_{uv}\mid r]=\frac{2s_r(u)+s_r(v)+1}{6}. \tag{12}$$

PROOF Following the characterization of planar arrangements described in Section 2.2, we have that $\mathbb{P}_{\mathrm{pl}}(\pi(r)=1)=1/n$. Then applying the Law of Total Expectation

$$\begin{aligned}\mathbb{E}_{\mathrm{pl}}[\delta_{uv}]&=\sum_{r\in V}\mathbb{E}_{\mathrm{pl}}[\delta_{uv}\mid \pi(r)=1]\,\mathbb{P}_{\mathrm{pl}}(\pi(r)=1)\\ &=\frac{1}{n}\sum_{r\in V}\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv}\mid r]. \end{aligned}\tag{13}$$

Now we calculate $\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv}\mid r]$ by cases. When $r\notin\{u,v\}$,

$$\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv}\mid r]=\mathbb{E}^{\diamond}_{\mathrm{pr}}[\delta_{uv}\mid r]=\mathbb{E}_{\mathrm{pr}}[\delta_{uv}\mid r]. \tag{14}$$

When $r \in \{u, v\}$, by linearity of expectation,

$$\begin{aligned}\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv} \mid r] &= \mathbb{E}^{\diamond}_{\mathrm{pr}}[\delta_{uv} \mid r] \\ &= \mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} + \beta_{uv} \mid r] \\ &= \mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid r] + \mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid r].\end{aligned}$$

By denoting $\overline{r}$ the only vertex in $\{u, v\} \setminus \{r\}$, then

$$\mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid r] = \mathbb{E}_{\mathrm{pr}}[\alpha_{uv} \mid r] = \frac{s_r(\overline{r}) + 1}{2}. \tag{15}$$

Equation 15 relies on the fact that in a rooted tree T^r, the expected length of the anchor of an edge incident to the root, say $rw \in E(T^r)$, is given by $\mathbb{E}_{\mathrm{pr}}[\alpha_{rw} \mid r] = (s_r(w) + 1)/2$ (Alemany-Puig and Ferrer-i-Cancho 2022). An arithmetic expression for $\mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid r]$ can be found by modifying the proof of Alemany-Puig and Ferrer-i-Cancho 2022, Lemma 1. Then, as before, we get (see Appendix 4.3),

$$\mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid r] = \frac{3}{2}\mathbb{E}_{\mathrm{pr}}[\beta_{uv} \mid r] = \frac{n - s_r(\overline{r}) - 1}{2}. \tag{16}$$

Therefore, by adding Equations 15 and 16 we obtain

$$\begin{aligned}\mathbb{E}^{\diamond}_{\mathrm{pl}}[\delta_{uv} \mid r] &= \mathbb{E}^{\diamond}_{\mathrm{pr}}[\alpha_{uv} \mid r] + \mathbb{E}^{\diamond}_{\mathrm{pr}}[\beta_{uv} \mid r] \\ &= \frac{s_r(\overline{r}) + 1}{2} + \frac{n - s_r(\overline{r}) - 1}{2} \\ &= \frac{n}{2}. \qquad (17)\end{aligned}$$

Equation 11 follows immediately after inserting Equations 17 and 14 in Equation 13. □

APPLICATIONS 3

A linear-time algorithm to compute $\mathbb{E}_{\mathrm{pl}}[D(T)]$ 3.1

Here we consider algorithms of increasing efficiency. First, since $\mathbb{E}_{\mathrm{pr}}[D(T^u)]$ can be calculated in $O(n)$-time for any n-vertex rooted tree T^u (Alemany-Puig and Ferrer-i-Cancho 2022, Theorem 1), the evaluation 'as is' of Equation 5 leads to a $O(n^2)$-time algorithm.

Second, we could calculate the value $\mathbb{E}_{\mathrm{pr}}[D(T^u)]$ for all $u \in V$ in $O(n)$-time and $O(n)$-space with the following procedure:

1. Precompute $s_u(v)$ in $O(n)$-time (Alemany-Puig *et al.* 2022);
2. Choose an arbitrary vertex w;
3. Calculate $\mathbb{E}_{\mathrm{pr}}[D(T^w)]$ in $O(n)$-time (Alemany-Puig and Ferrer-i-Cancho 2022); and, finally,
4. Perform a Breadth First Search (BFS) traversal of T starting at w. In this traversal, when going from vertex u to vertex v, the value of $\mathbb{E}_{\mathrm{pr}}[D(T^v)]$ is calculated applying the precomputed value of $\mathbb{E}_{\mathrm{pr}}[D(T^u)]$ to the following equation:

   $$\mathbb{E}_{\mathrm{pr}}[D(T^u)] = \mathbb{E}_{\mathrm{pr}}[D(T^v)] + \Delta,$$

   where Δ is equal to the difference $\mathbb{E}_{\mathrm{pr}}[D(T^u)] - \mathbb{E}_{\mathrm{pr}}[D(T^v)]$. We can obtain a formula for this difference by manipulating Equation 3. We get

   $$\begin{aligned}\Delta &= \mathbb{E}_{\mathrm{pr}}[D(T^u)] - \mathbb{E}_{\mathrm{pr}}[D(T^v)] \\ &= \frac{1}{6}\big[s_u(v)(2d(v)-1) + 2n(d(u)-d(v)) \\ &\qquad - s_v(u)(2d(u)-1)\big].\end{aligned}$$

   Notice that the value of Δ can be computed in constant time for any two vertices u and v (here we are interested in the value of Δ for pairs of adjacent vertices) and, crucially, without knowledge of either $\mathbb{E}_{\mathrm{pr}}[D(T^u)]$ or $\mathbb{E}_{\mathrm{pr}}[D(T^v)]$. That is, if the value of $\mathbb{E}_{\mathrm{pr}}[D(T^u)]$ is known then the value of $\mathbb{E}_{\mathrm{pr}}[D(T^v)]$ for any $v \in \Gamma(u)$ can be calculated in constant time as

   $$\mathbb{E}_{\mathrm{pr}}[D(T^v)] = \mathbb{E}_{\mathrm{pr}}[D(T^u)] - \Delta.$$

Third, we propose an alternative that is also $O(n)$-time yet simpler and faster in practice, based on Proposition 2.

PROPOSITION 2 *Given a free tree $T = (V, E)$,*

$$\begin{aligned}\mathbb{E}_{\mathrm{pl}}[D(T)] &= \frac{(n-1)(3n^2+2n-2)}{6n} \\ &\quad - \frac{1}{6n}\sum_{v\in V}(2d(v)-1)\sum_{u\in\Gamma(v)} s_v(u)^2.\end{aligned} \tag{18}$$

PROOF Here we simplify the summation in Equation 5, which becomes (as per Alemany-Puig and Ferrer-i-Cancho 2022)

$$\frac{1}{n}\sum_{u\in V}\mathbb{E}_{\mathrm{pr}}[D(T^u)] = \frac{1}{6n}(f(T)-n)$$

with

$$f(T) = \sum_{u\in V}\sum_{v\in V} s_u(v)(d_u(v)+1).$$

Now we simplify $f(T)$ by first replacing the term $d_u(v)$ by $d(v)$ after the necessary transformations so that we can swap the order of the summations afterwards, that is,

$$\begin{aligned}
f(T) &= \sum_{u\in V}\Big(s_u(u)(2d_u(u)+1) + \sum_{v\in V\setminus\{u\}} s_u(v)(2d_u(v)+1)\Big)\\
&= \sum_{u\in V} n(2d(u)+1) + \sum_{u\in V}\sum_{v\in V\setminus\{u\}} s_u(v)(2d(v)-1)\\
&= n(5n-4) - \sum_{u\in V} s_u(u)(2d(u)-1)\\
&\quad + 2\sum_{u\in V}\sum_{v\in V} s_u(v)d(v) - \sum_{u\in V}\sum_{v\in V} s_u(v)\\
&= 2n^2 + g(T) - h(T) \qquad (19)
\end{aligned}$$

with

$$g(T) = 2\sum_{u\in V}\sum_{v\in V} s_u(v)d(v), \qquad (20)$$

$$h(T) = \sum_{u\in V}\sum_{v\in V} s_u(v). \qquad (21)$$

In the preceding derivation, the second equality holds due to $d_u(v) = d(v)-1$ for $v \neq u$; the third and fourth steps, we apply the Handshaking lemma.[8] These lead to

$$\frac{1}{n}\sum_{u\in V}\mathbb{E}_{\mathrm{pr}}[D(T^u)] = \frac{1}{6n}(n(2n-1) + g(T) - h(T)). \qquad (22)$$

[8] The Handshaking lemma (Gunderson 2014) states that the sum of the degrees of all vertices of a graph equals twice the number of its edges.

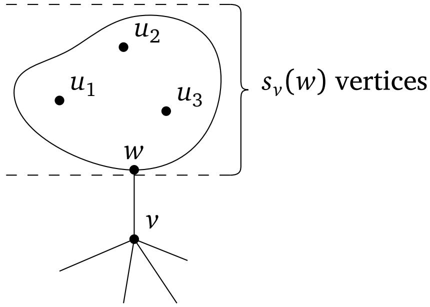


Figure 6:
Proof of Proposition 2. The value $s_u(v)$ is the same for all vertices of T^v_w denoted as $\{u_1,\dots,u_k\}$ in the figure and the proof

It remains to simplify Equations 20 and 21. We start by changing the order of the summations in Equation 20,

$$g(T) = 2\sum_{v\in V}\sum_{u\in V} s_u(v)d(v) = 2\sum_{v\in V} d(v)\sum_{u\in V} s_u(v),$$

and continue simplifying the inner summation. Consider a fixed $v \in V$. We have that

$$\underbrace{\sum_{u\in V} s_u(v)}_{(1)} = n + \underbrace{\sum_{u\in V\setminus\{v\}} s_u(v)}_{(2)} = n + \sum_{w\in\Gamma(v)} s_w(v)s_v(w).$$

The summation (1) adds up the size of all subtrees T^w_v with respect to a 'moving' root w. In the first equality we have simply taken out the case $s_u(u)$. To understand the second equality, focus for now on a single subtree T^v_w such that $wv \in E$. The summation (2) contains summands that correspond to all the vertices in T^v_w, say vertices $u_1,\dots,u_k$ (assume, without loss of generality, that $w = u_k$). These summands are $s_{u_1}(v),\dots,s_{u_k}(v)$, which are all equal to $s_w(v)$ (Figure 6). Moreover, there are $s_v(w)$ vertices in T^v_w thus $k = s_v(w)$, and this holds for all $w \in \Gamma(v)$, hence the equality. Finally,

$$\sum_{u\in V} s_u(v) = n + \sum_{u\in\Gamma(v)} (n - s_v(u))s_v(u) = n^2 - \sum_{u\in\Gamma(v)} s_v(u)^2, \tag{23}$$

thanks to the identity $s_u(v) + s_v(u) = n$. Then,

$$g(T) = 4n^2(n-1) - 2\sum_{v\in V} d(v) \sum_{u\in\Gamma(v)} s_v(u)^2. \tag{24}$$

We use the result in Equation 23 to simplify Equation 21,

$$h(T) = \sum_{v\in V}\sum_{u\in V} s_u(v) = n^3 - \sum_{v\in V}\sum_{u\in\Gamma(v)} s_v(u)^2. \tag{25}$$

By combining Equations 24 and 25 into Equation 22 and, after some effort, we obtain

$$\mathbb{E}_{\mathrm{pl}}[D(T)] = \frac{(n-1)(n-2)}{6n} + \frac{1}{6n}\Bigg(n(n-1)(3n+1) - \sum_{v\in V}(2d(v)-1)\sum_{u\in\Gamma(v)} s_v(u)^2\Bigg),$$

which leads directly to Equation 18. □

LEMMA 3 *For any given free tree T, Algorithm 4 calculates $\mathbb{E}_{\mathrm{pl}}[D(T)]$ in time and space $O(n)$.*

PROOF The pseudocode to calculate $\mathbb{E}_{\mathrm{pl}}[D(T)]$ based on Proposition 2 is given in Algorithm 4. This algorithm first calculates $s_u(v)$ for all edges $uv \in E$, for the given tree T in $O(n)$ time using the pseudocode by Alemany-Puig *et al.* 2022, Algorithm 2.1. Then it uses these values to calculate the sums of $s_v(u)^2$ for every vertex $v \in V$. Such sums are then used to evaluate Equation 18 hence calculating $\mathbb{E}_{\mathrm{pl}}[D(T)]$ in time $O(n)$. □

Function COMPUTE_EXPECTED_PLANAR(T) **is**
  Input: T free tree.
  Output: $\mathbb{E}_{\mathrm{pl}}[D(T)]$.
  `// Alemany-Puig et al. 2022, Algorithm 2.1`
  $S \leftarrow$ COMPUTE_S_FT(T)
  $L \leftarrow \{0\}^n$ `// a vector of n zeroes.`
  for $(u, v, s_u(v)) \in S$ **do** $L[u] \leftarrow L[u] + s_u(v)^2$
  return $((n-1)(3n^2+2n-2) - \sum_{u\in V}(d(u)-1)L[u])/6n$

Algorithm 4: Calculation of $\mathbb{E}_{\mathrm{pl}}[D(T)]$. Cost $O(n)$-time, $O(n)$-space

3.1.1 A simple application

Let $\mathbb{E}_{\geq 1}[D(T)]$ be the expected value of the sum of edge lengths conditioned to arrangements π such that $C_\pi(T) \geq 1$. That is, arrangements such that the number of edge crossings is at least 1. An immediate consequence of Lemma 3 is that $\mathbb{E}_{\geq 1}[D(T)]$ can be computed easily as the following corollary states.

COROLLARY 3 *For any free tree T, $\mathbb{E}_{\geq 1}[D(T)]$ can be computed in time and space $O(n)$ thanks to the fact that*

$$\mathbb{E}_{\geq 1}[D(T)] = \frac{\mathbb{E}[D(T)] - \mathbb{E}_{\text{pl}}[D(T)]\,\mathbb{P}(C(T)=0)}{\mathbb{P}(C(T)\geq 1)} \tag{26}$$

with $\mathbb{P}(C(T) \leq 0) = \mathbf{N}_{\mathbf{pl}}(T)/n!$ and $\mathbb{P}(C(T) \geq 1) = (n! - \mathbf{N}_{\mathbf{pl}}(T))/n!$.

PROOF Due to the Law of Total Expectation,

$$\mathbb{E}[D(T)] = \mathbb{E}_{\text{pl}}[D(T)]\,\mathbb{P}(C(T)=0) + \mathbb{E}_{\geq 1}[D(T)]\,\mathbb{P}(C(T)\geq 1), \tag{27}$$

and hence Equation 26. $\mathbf{N}_{\mathbf{pl}}(T)$ can be computed in $O(n)$-time with Equation 6 and $\mathbb{E}_{\text{pl}}[D(T)]$ can be computed in time and space $O(n)$ (Lemma 3). Hence all the components in the right hand side of Equation 26 can be computed in time and space $O(n)$. □

3.2 *Real syntactic dependency distances versus random baselines*

Evidence that dependency distances are smaller than expected by chance can be obtained by random baselines of varying strength:

- None, $\mathbb{E}[D(T)]$, the expectation of $D(T)$ in unconstrained random linear arrangements (Ferrer-i-Cancho 2004).
- Planarity, $\mathbb{E}_{\text{pl}}[D(T)]$, the expectation of $D(T)$ in planar random linear arrangements (this article).
- Projectivity, $\mathbb{E}_{\text{pr}}[D(T^r)]$, the expectation of $D(T)$ in projective random linear arrangements (Alemany-Puig and Ferrer-i-Cancho 2022; Gildea and Temperley 2007).

This raises the questions of what would be the most appropriate baseline for research on dependency distance minimization. $\mathbb{E}_{\text{pr}}[D(T^r)]$ is by far the most widely used random baseline (Gildea and Temperley 2007; Liu 2008; Park and Levy 2009; Futrell *et al.* 2015).

Since planarity is a weaker condition than projectivity, $\mathbb{E}_{\text{pl}}[D(T)]$ implies a gain in coverage. Accordingly, there are more planar sentences than projective sentences in real texts (Havelka 2007; Gómez-Rodríguez and g 2010, Table 1) and also in artificially-generated syntactic dependency structures (Gómez-Rodríguez *et al.* 2022, Figure 2).

However, surprisingly, $\mathbb{E}_{\text{pl}}[D(T)]$ has never been used in research on the principle of dependency distance minimization. Here we aim to test the hypothesis that formal constraints mask the effects of the principle, a hypothesis that has already been confirmed on artificially-generated syntactic dependency structures (Gómez-Rodríguez *et al.* 2022).

Given the natural growth of dependency distance as sentence length increases (Ferrer-i-Cancho and Liu 2014; Ferrer-i-Cancho *et al.* 2022), we measure, for each sentence, the average dependency distance, namely $\langle d \rangle = D(T)/(n-1)$ instead of the raw total sum $D(T)$ (a sentence of n vertices has $n-1$ syntactic dependencies when the structure is a tree). As, in addition to such a growth, the manifestation of the principle also depends on sentence length (the statistical bias towards shorter distances may disappear or become a bias in the opposite direction in short sentences; Ferrer-i-Cancho and Gómez-Rodríguez 2021; Ferrer-i-Cancho *et al.* 2022), we compare the actual dependency distances against the values predicted by the baselines in sentences of the same length.

Data and methods 3.2.1

We use the Parallel Universal Dependencies 2.6 collection (Zeman *et al.* 2020) for experimentation. To control for annotation style, we consider two versions of the collection: the collection with its original content-head annotation (PUD) and its transformation into Surface-Syntactic Universal Dependencies 2.6 (hereafter PSUD). By doing so, we cover two major competing annotation styles (Gerdes *et al.* 2018).

We borrow the preprocessing methods from previous research (Ferrer-i-Cancho *et al.* 2022). The main features of the processing are that nodes that are punctuation marks are removed and that the corpus remains fully parallel after the removal (Ferrer-i-Cancho *et al.* 2022). The preprocessed data is freely available as ancillary materials of the Linear Arrangement Library website.[9]

With respect to previous accounts (Havelka 2007; Ferrer-i-Cancho *et al.* 2018; Gómez-Rodríguez and g 2010), our collections exhibit some remarkable statistical differences. First, the proportion of projective and planar sentences is higher in PUD, where the proportion of

[9] `https://cqllab.upc.edu/lal/universal-dependencies/`

Table 2: Proportion (%) of projective and planar sentences in the PUD collection

Language	Projective	Planar	Language	Projective	Planar
Arabic	96.2	96.3	Italian	99.3	99.3
Czech	89.6	89.8	Japanese	99.7	99.7
Chinese	99.4	99.4	Korean	93.6	95.2
German	86.3	86.7	Polish	94.8	95.3
English	95.5	95.9	Portuguese	96.7	96.8
Finnish	96.4	96.7	Russian	97.6	98.0
French	98.3	98.3	Spanish	95.5	95.7
Hindi	74.3	76.3	Swedish	96.5	96.9
Icelandic	96.2	96.9	Thai	97.2	97.2
Indonesian	98.7	99.0	Turkish	93.5	94.1

Table 3: Proportion (%) of projective and planar sentences in the PSUD collection

Language	Projective	Planar	Language	Projective	Planar
Arabic	83.6	83.9	Italian	94.5	94.6
Czech	86.6	87.2	Japanese	35.8	35.8
Chinese	42.0	46.1	Korean	75.8	77.1
German	72.3	72.7	Polish	88.2	89.7
English	93.6	94.1	Portuguese	87.3	87.7
Finnish	88.8	89.4	Russian	95.1	95.5
French	90.5	90.6	Spanish	80.2	80.9
Hindi	43.6	44.3	Swedish	93.0	93.7
Icelandic	90.7	92.0	Thai	85.6	86.8
Indonesian	90.5	91.8	Turkish	87.6	88.3

non-projective or non-planar sentences does not exceed 10% in most cases (Tables 2 and 3). This proportion increases in PSUD; wherein, in two exceptional languages, Chinese and Hindi, it becomes larger than 50% (Table 3). Second, the difference between the proportion of non-projective and non-planar sentences is smaller than in previous reports (Gómez-Rodríguez and g 2010; Havelka 2007). Having said that, notice that our collections are fully parallel, and special care has been taken to keep annotation consistent across languages.

Given formal constraint 'c' (either 'none', 'planarity' ($c = \text{pl}$) or 'projectivity' ($c = \text{pr}$)) and sentence length n,

1. We calculate $D(T^r)$ for each T^r and also calculate the expected

sum of edge lengths under 'c' different constraints (none, Equation 2; planarity, Equation 5; projectivity, Equation 3).

2. Then, for each sentence, we divide each by $n-1$, to produce the mean length of its dependencies

$$\langle d_c \rangle = \frac{D}{n-1}$$

and the expected mean of length of its dependencies under some constraint 'c'

$$\mathbb{E}\left[\langle d_c \rangle\right] = \frac{\mathbb{E}_c\left[D\right]}{n-1}.$$

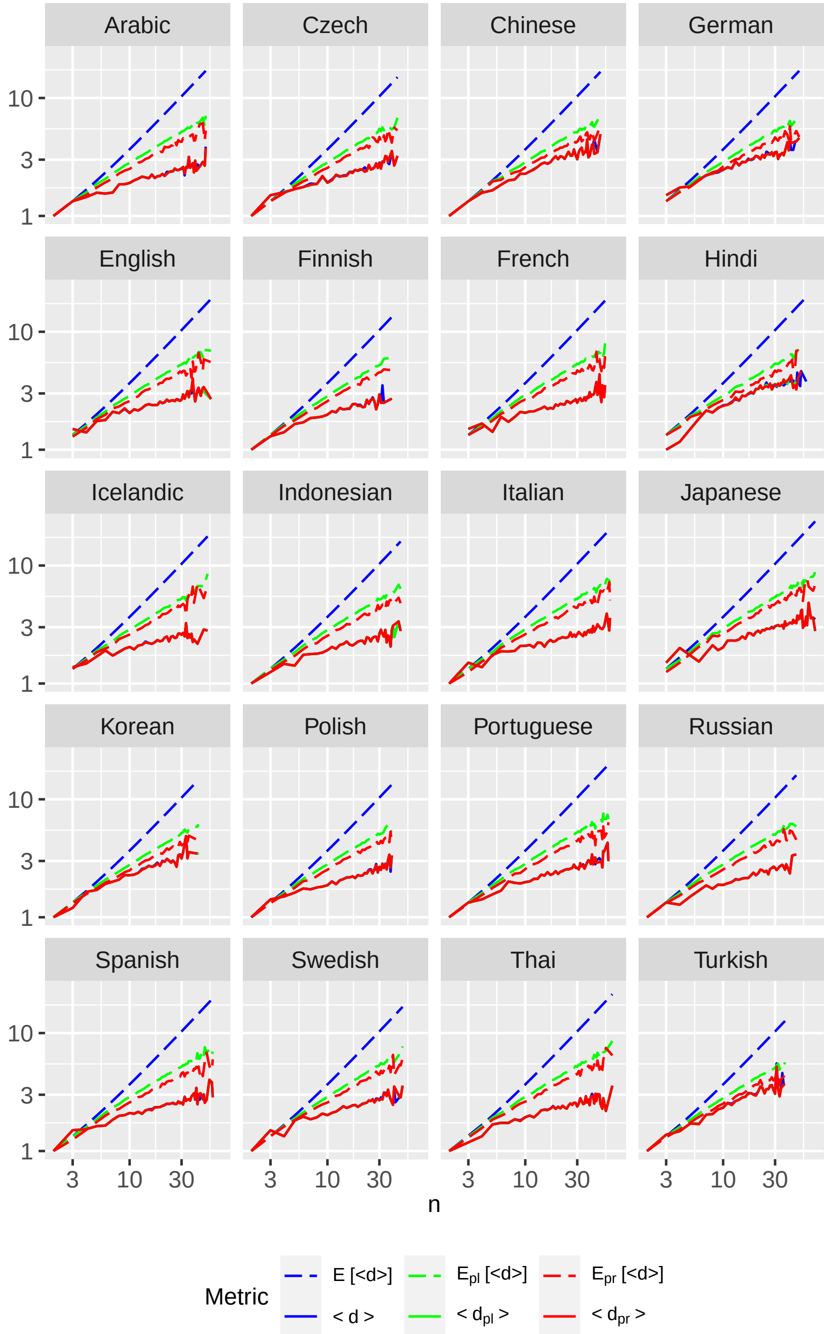
3. Finally, we compute the average $\langle d_c \rangle$ and the average $\mathbb{E}\left[\langle d_c \rangle\right]$ over all sentences of length n satisfying constraint 'c'.

Results 3.2.2

Figures 7 and 8 show the scaling of mean dependency distance as a function of sentence length in real sentences and in their corresponding random baselines. Concerning the random baselines (dashed lines), we find that the stronger the formal constraint on syntactic dependency structures, the lower the value of the random baseline. In contrast, the actual mean sentence length (solid lines) is practically the same independently of the formal constraint (none, planarity and projectivity). This is due to the fact the proportion of sentences that are lost by imposing some formal constraint is small in the PUD and PSUD collections, namely, the baselines $\langle d \rangle$, $\langle d_{\mathrm{pl}} \rangle$ and $\langle d_{\mathrm{pr}} \rangle$ are extremely similar in value. The overwhelming majority of sentences are planar and the proportion of planar sentences that are not projective is really small (Table 2 and 3). Thus, selecting sentences satisfying a certain formal constraint has a negligible impact on the estimation of mean dependency distance.

Concerning the relationship between the actual mean dependency distance and the random baselines, we find that the average $\langle d \rangle$ is below the average value of the random baselines for sufficiently large n in all languages. The only exception is Turkish, where the actual average $\langle d \rangle$ is just slightly below the average of the projective baseline (Figures 7 and 8).

Figure 7: The scaling of $\langle d \rangle$, the mean dependency distance of a sentence as a function of sentence length (n) for languages in the PUD collection for formal constraints of increasing strength: none (blue), planarity (green) and projectivity (red). Lines indicate the average value over all sentences of the same length. Solid lines are used for real sentences and dashed lines are used for the corresponding random baseline. Solid lines overlap so much that only one of them can be seen in most cases



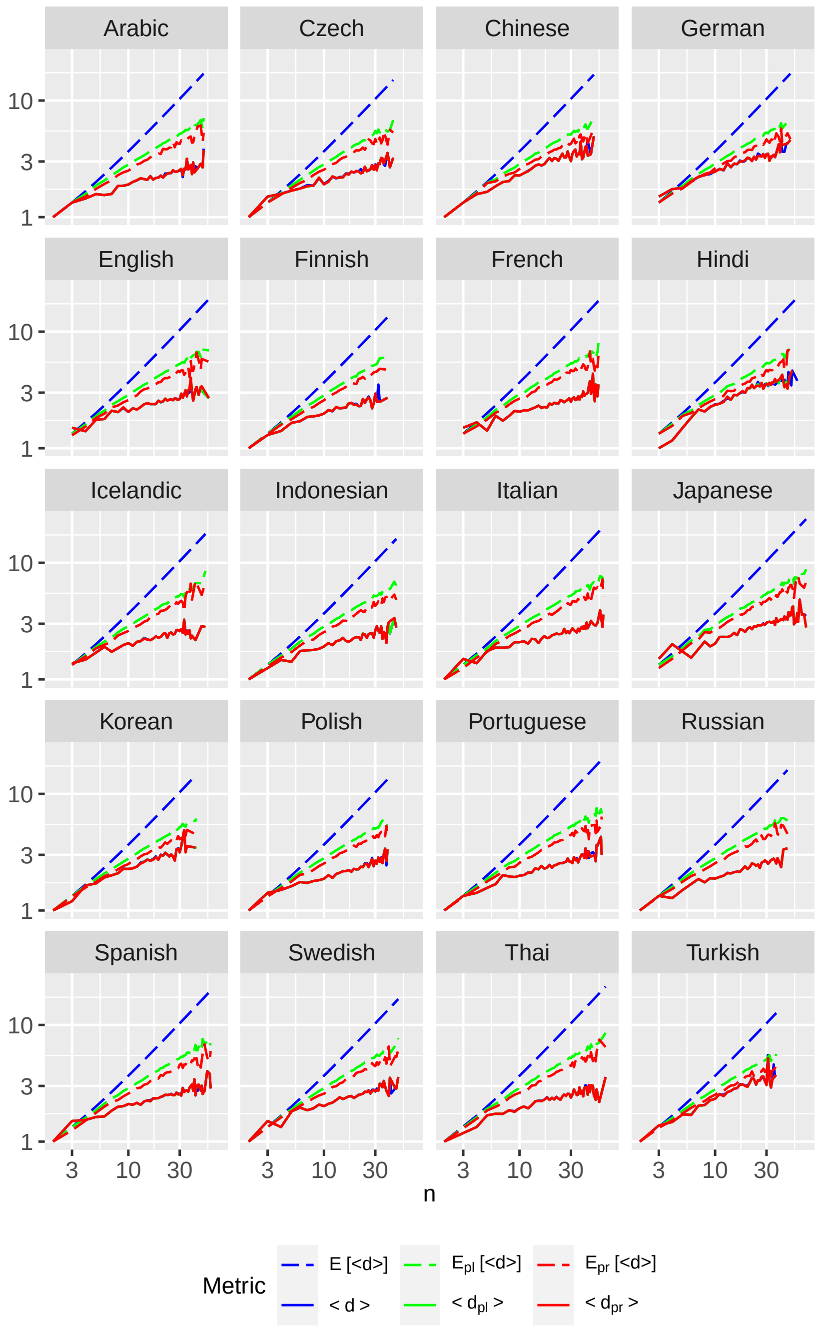


Figure 8: The scaling of $\langle d \rangle$, the mean dependency distance of a sentence as a function of sentence length (n) for languages in the PSUD collection for formal constraints of increasing strength. Format is the same as in Figure 7. Again, solid lines overlap so much that only one of them can be seen in most cases

These findings are consistent between PUD and PSUD, in spite of their differences in proportions of projective and planar sentences commented above.

4 CONCLUSIONS AND FUTURE WORK

4.1 *Theory*

In Section 2.2, we have characterized planar arrangements of a given free tree T using the concept of segment (Alemany-Puig and Ferrer-i-Cancho 2022). Employing said characterization, we have shown that the number of planar arrangements of a free tree depends on its degree sequence (Proposition 1), similar to the manner in which projective arrangements of a rooted tree do (Alemany-Puig and Ferrer-i-Cancho 2022). Moreover, we have given a procedure to generate u.a.r. planar arrangements of a given free tree in Section 2.3 (Algorithm 3) which can be easily adapted to generate such arrangements exhaustively. Notably, our algorithm to generate planar arrangements is based on the generation of projective arrangements of a rooted subtree. For the sake of completeness, we have detailed a procedure to generate u.a.r. projective arrangements of a given rooted tree (Algorithm 1).

4.2 *Applications*

Having identified the underlying structure of planar arrangements, we have derived an arithmetic expression, in Section 2.4, for $\mathbb{E}_{\text{pl}}[D(T)]$ (Theorem 1). We have also devised a $O(n)$-time algorithm to calculate this value (Proposition 1, Algorithm 4).

In Section 3, we have applied the theory developed up until that point to investigate the effect of formal constraints of increasing strength (none, planarity, projectivity) in a parallel collection and reported two main findings. First, the average dependency distance in real sentences remains practically the same while the strength of the formal constraint increases. We believe that this result stems from the high proportion of planar sentences (and the very low proportion of planar sentences that are not projective) of the PUD collection. Higher proportions of non-planar sentences have been reported in other collections (Gómez-Rodríguez and Ferrer-i-Cancho 2017). Second, the tendency of the random baseline to have a smaller value in stronger formal constraints indicates that the strength of the dependency distance minimization effect depends on the choice of the formal constraint for the random baseline. As these formal constraints may be a side effect of dependency distance minimization (Ferrer-i-Cancho 2006; Gómez-Rodríguez and Ferrer-i-Cancho 2017; Gómez-Rodríguez *et al.* 2022; Yadav *et al.* 2022), this phenomenon suggests that

1. Formal constraints absorb the dependency distance effect.
2. A fairer evaluation of the actual degree of optimization of dependency distances or a more accurate measurement of the power of the effect of dependency distance minimization requires considering not only the magnitude of the effect with respect to some random baseline but also the formal constraint, as the latter may hide part of the dependency distance minimization effect.

In past research on syntactic dependency distance minimization, $\mathbb{E}_{\mathrm{pr}}[D(T^r)]$ has been the most widely used random baseline (Gildea and Temperley 2007; Liu 2008; Park and Levy 2009; Futrell *et al.* 2015). However, projectivity has a lower coverage than planarity in real sentences (Havelka 2007; Gómez-Rodríguez and g 2010). Projectivity is at risk of underestimating the strength of the dependency distance minimizaton principle (Ferrer-i-Cancho 2004) because of the significant reduction in the value of the random baseline (Figures 7 and 8) or the reduction of the actual dependency distances (Gómez-Rodríguez *et al.* 2022, Figure 2) that it introduces. Thanks to the research in this article, we have paved the way for replicating past research replacing $\mathbb{E}_{\mathrm{pr}}[D(T^r)]$ with $\mathbb{E}_{\mathrm{pl}}[D(T)]$.

Future work 4.3

Planarity is a relaxation of projectivity but future work should address the problem of the expected value of $D(T)$ in classes of formal constraints with even more coverage (Ferrer-i-Cancho *et al.* 2018). A promising step is the investigation of $\mathbb{E}_{\leq k}[D(T)]$, the expected value of $D(T)$ conditioned to arrangements π such that $C_\pi(T) \leq k$, that is, in arrangements such that the number of edge crossings is at most k. Notice that $\mathbb{E}_{\leq 0}[D(T)] = \mathbb{E}_{\text{pl}}[D(T)]$. In real languages, the average number of crossings ranges between 0.40 and 0.62 (Ferrer-i-Cancho *et al.* 2018), suggesting that $\mathbb{E}_{\leq k}[D(T)]$ with $k = 1$ or a small k would suffice.

ACKNOWLEDGMENTS

We are grateful to J. L. Esteban for helpful comments and C. Gómez-Rodríguez advice on the computational linguistics literature. LAP is supported by Secretaria d'Universitats i Recerca de la Generalitat de Catalunya and the Social European Fund and a Ph.D. contract extension funded by Banco Santander. This research is supported by a recognition 2021SGR-Cat (01266 LQMC) from AGAUR (Generalitat de Catalunya) and the grants AGRUPS-2022 and AGRUPS-2023 from Universitat Politècnica de Catalunya.

APPENDIX
DERIVATION OF $\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u]$

Here we derive the expected length of the coanchor of a (directed) edge $uv \in E(T^u)$ in uniformly random projective arrangements of T^u conditioned to $\pi(u) = 1$. Following Alemany-Puig and Ferrer-i-Cancho (2022), we decompose the length of the coanchor of the (directed) edge uv, β_{uv}, as the sum of the lengths of the segments in-between u and v (Figure 4). Here we use k_{uv} to denote the number of segments in-between u and v, and $\varphi^{(i)}_{uv}$ to denote the size of the ith segment, yielding (Alemany-Puig and Ferrer-i-Cancho 2022),

$$\beta_{uv} = \sum_{i=1}^{k_{uv}} \varphi^{(i)}_{uv}.$$

By the Law of Total Expectation, we have that

$$\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u] = \sum_{k=1}^{d(u)-1} \mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u, k_{uv} = k] \mathbb{P}^{\diamond}_{\text{pr}}(k_{uv} = k \mid u), \tag{28}$$

where $\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u, k_{uv} = k]$ is the expectation of β_{uv} given that u is the root of the tree (fixed at the leftmost position), and that u and v are separated by k segments, and $\mathbb{P}^{\diamond}_{\text{pr}}(k_{uv} = k \mid u)$ is the probability that u and v are separated by k intermediate segments, both in uniformly random projective arrangements π conditioned to $\pi(u) = 1$, both conditioned to the root of the tree being vertex u. On the one hand,

$$\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u, k_{uv} = k] = \mathbb{E}^{\diamond}_{\text{pr}}\left[\sum_{i=1}^{k} \varphi^{(i)}_{uv} \mid u\right] = \frac{n - s_u(v) - 1}{d(u) - 1} k. \tag{29}$$

Notice that this is the same result as that obtained in Alemany-Puig and Ferrer-i-Cancho 2022. Lastly, the proportion of arrangements in which the segment of v is at position $k_{uv} + 1$ equals $(d(u) - 1)!$, therefore,

$$\mathbb{P}^{\diamond}_{\text{pr}}(k_{uv} = k \mid u) = \frac{(d(u) - 1)! \prod_{v \in \Gamma(u)} \mathbf{N}_{\mathbf{pr}}(T^u)}{d(u)! \prod_{v \in \Gamma(u)} \mathbf{N}_{\mathbf{pr}}(T^u)} = \frac{1}{d(u)}. \tag{30}$$

Recalling that (Alemany-Puig and Ferrer-i-Cancho 2022)

$$\mathbb{E}_{\text{pr}}[\beta_{uv} \mid u] = \frac{s_u(u) - s_u(v) - 1}{3},$$

and plugging the results of Equations 29 and 30 into Equation 28, we get

$$\begin{aligned}
\mathbb{E}^{\diamond}_{\text{pr}}[\beta_{uv} \mid u] &= \frac{n - s_u(v) - 1}{d(u) - 1} \frac{1}{d(u)} \sum_{k=1}^{d(u)-1} k \\
&= \frac{s_u(u) - s_u(v) - 1}{2} \\
&= \frac{3}{2} \mathbb{E}_{\text{pr}}[\beta_{uv} \mid u].
\end{aligned}$$

REFERENCES

Lluís ALEMANY-PUIG, 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)*, pp. 1–16, Association for Computational Linguistics, Sofia, Bulgaria, `https://aclanthology.org/2021.quasy-1.1`.

Lluís ALEMANY-PUIG, Juan Luis ESTEBAN, and Ramon FERRER-I-CANCHO (2022), Minimum projective linearizations of trees in linear time, *Information Processing Letters*, 174:106204, doi:10.1016/j.ipl.2021.106204.

Lluís ALEMANY-PUIG and Ramon FERRER-I-CANCHO (2022), Linear-time calculation of the expected sum of edge lengths in projective linearizations of trees, *Computational Linguistics*, 48(3):491–516, doi:10.1162/coli_a_00442.

Frank BERNHART and Paul C. KAINEN (1979), The book thickness of a graph, *Journal of Combinatorial Theory, Series B*, 27(3):320–331, doi:10.1016/0095-8956(79)90021-2.

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

Thomas H. CORMEN, Charles E. LEISERSON, Ronald L. RIVEST, and Clifford STEIN (2001), *Introduction to algorithms*, The MIT Press, Cambridge, MA, USA, 2nd edition.

Ramon FERRER-I-CANCHO (2004), Euclidean distance between syntactically linked words, *Physical Review E*, 70:056135, doi:10.1103/PhysRevE.70.056135.

Ramon FERRER-I-CANCHO (2006), Why do syntactic links not cross?, *Europhysics Letters (EPL)*, 76(6):1228–1235, doi:10.1209/epl/i2006-10406-0.

Ramon FERRER-I-CANCHO (2019), The sum of edge lengths in random linear arrangements, *Journal of Statistical Mechanics*, 2019(5):053401, doi:10.1088/1742-5468/ab11e2.

Ramon FERRER-I-CANCHO and Carlos GÓMEZ-RODRÍGUEZ (2021), Anti dependency distance minimization in short sequences. a graph theoretic approach, *Journal of Quantitative Linguistics*, 28(1):50–76, doi:10.1080/09296174.2019.1645547.

Ramon FERRER-I-CANCHO, Carlos GÓMEZ-RODRÍGUEZ, and Juan Luis ESTEBAN (2018), Are crossing dependencies really scarce?, *Physica A: Statistical Mechanics and its Applications*, 493:311–329, doi:10.1016/j.physa.2017.10.048.

Ramon FERRER-I-CANCHO, Carlos GÓMEZ-RODRÍGUEZ, Juan Luis ESTEBAN, and Lluís ALEMANY-PUIG (2022), Optimality of syntactic dependency distances, *Physical Review E*, 105:014308, doi:10.1103/PhysRevE.105.014308.

Ramon FERRER-I-CANCHO and Haitao LIU (2014), The risks of mixing dependency lengths from sequences of different length, *Glottotheory*, 5:143–155, doi:10.1515/glot-2014-0014.

Richard FUTRELL, 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, doi:10.1073/pnas.1502134112.

Kim GERDES, Bruno GUILLAUME, Sylvain KAHANE, and Guy PERRIER (2018), SUD or surface-syntactic universal dependencies: an annotation scheme near-isomorphic to UD, in *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pp. 66–74, Association for Computational Linguistics, Brussels, Belgium, doi:10.18653/v1/W18-6008.

Daniel GILDEA and David TEMPERLEY (2007), Optimizing grammars for minimum dependency length, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 184–191, Association for Computational Linguistics, Prague, Czech Republic, `https://www.aclweb.org/anthology/P07-1024`.

David GILDEA and David TEMPERLEY (2010), Do grammars minimize dependency length?, *Cognitive Science*, 34(2):286–310, doi:10.1111/j.1551-6709.2009.01073.x.

Carlos GÓMEZ-RODRÍGUEZ (2016), Restricted non-projectivity: Coverage vs. efficiency, *Computational Linguistics*, 42(4):809–817, doi:10.1162/COLI_a_00267.

Carlos GÓMEZ-RODRÍGUEZ, Morten H. CHRISTIANSEN, and Ramon FERRER-I-CANCHO (2022), Memory limitations are hidden in grammar, *Glottometrics*, 52:39–64, doi:10.53482/2022_52_397.

Carlos GÓMEZ-RODRÍGUEZ and Ramon FERRER-I-CANCHO (2017), Scarcity of crossing dependencies: a direct outcome of a specific constraint?, *Physics Review E*, 96:062304, doi:10.1103/PhysRevE.96.062304.

Carlos GÓMEZ-RODRÍGUEZ and Joakim G (2010), A transition-based parser for 2-planar dependency structures, in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 1492–1501, Association for Computational Linguistics, Uppsala, Sweden, `https://aclanthology.org/P10-1151`.

Thomas GROSS and Timothy OSBORNE (2009), Toward a practical dependency grammar theory of discontinuities, *SKY Journal of Linguistics*, 22:43–90, `http://www.linguistics.fi/julkaisut/sky2009.shtml`.

David S. GUNDERSON (2014), *Handbook of Mathematical Induction: Theory and Applications*, Discrete Mathematics and Its Applications, CRC Press, ISBN 9781420093643, `https://www.routledgehandbooks.com/doi/10.1201/9781420093650`.

Jiří HAVELKA (2007), Beyond projectivity: multilingual evaluation of constraints and measures on non-projective structures, in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 608–615, Association for Computational Linguistics, Prague, Czech Republic, `https://aclanthology.org/P07-1077`.

Robert A. HOCHBERG and Matthias F. STALLMANN (2003), Optimal one-page tree embeddings in linear time, *Information Processing Letters*, 87(2):59–66, doi:10.1016/S0020-0190(03)00261-8.

Richard HUDSON (1995), Measuring syntactic difficulty, *Unpublished paper*, `https://dickhudson.com/wp-content/uploads/2013/07/Difficulty.pdf`.

Alex KRAMER (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, pp. 359–365, doi:10.7275/pz9g-d780.

Marco KUHLMANN and Joakim NIVRE (2006), Mildly non-projective dependency structures, in *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, COLING-ACL '06, pp. 507–514, doi:10.3115/1273073.1273139.

Haitao LIU (2008), Dependency distance as a metric of language comprehension difficulty, *Journal of Cognitive Science*, 9(2):159–191, doi:10.17791/jcs.2008.9.2.159.

Haitao LIU, Chunshan XU, and Junying LIANG (2017), Dependency distance: a new perspective on syntactic patterns in natural languages, *Physics of Life Reviews*, 21:171–193, doi:10.1016/j.plrev.2017.03.002.

Michael MITZENMACHER and Eli UPFAL (2017), *Probability and computing. Randomization and probabilistic techniques in algorithms and data analysis*, Cambridge University Press, ISBN 978-1-107-15488-9.

Glyn MORRILL (2000), Incremental processing and acceptability, *Computational Linguistics*, 25(3):319–338, doi:10.1162/089120100561728.

Joakim NIVRE (2006), Constraints on non-projective dependency parsing, in Diana MCCARTHY and Shuly WINTNER, editors, *11th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 73–80, Association for Computational Linguistics, Trento, Italy, `https://aclanthology.org/E06-1010`.

Joakim NIVRE (2009), Non-projective dependency parsing in expected linear 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*, ACL '09, pp. 351–359, Association for Computational Linguistics, Stroudsburg, PA, USA, `https://aclanthology.org/P09-1040`.

Y. Albert PARK and Roger P. LEVY (2009), Minimal-length linearizations for mildly context-sensitive dependency trees, in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 335–343, Association for Computational Linguistics, Stroudsburg, PA, USA, https://aclanthology.org/N09-1038/.

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

Daniel SLEATOR and Davy TEMPERLEY (1993), Parsing English with a link grammar, in *Proceedings of the Third International Workshop on Parsing Technologies (IWPT'93)*, pp. 277–292, ACL/SIGPARSE, https://dblp.uni-trier.de/db/journals/corr/corr9508.html#abs-cmp-lg-9508004.

David TEMPERLEY (2008), Dependency-length minimization in natural and artificial languages, *Journal of Quantitative Linguistics*, 15(3):256–282, doi:10.1080/09296170802159512.

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

Himanshu YADAV, Samar HUSAIN, and Richard FUTRELL (2022), Assessing corpus evidence for formal and psycholinguistic constraints on nonprojectivity, *Computational Linguistics*, pp. 1–27, doi:10.1162/coli_a_00437.

Daniel ZEMAN, Joakim NIVRE, Mitchell ABRAMS, Elia ACKERMANN, Noëmi AEPLI, Željko AGIĆ, Lars AHRENBERG, Chika Kennedy AJEDE, Gabrielė ALEKSANDRAVIČIŪTĖ, Lene ANTONSEN, Katya APLONOVA, Angelina AQUINO, Maria Jesús ARANZABE, Gashaw ARUTIE, Masayuki ASAHARA, Luma ATEYAH, Furkan ATMACA, Mohammed ATTIA, Aitziber ATUTXA, Liesbeth AUGUSTINUS, Elena BADMAEVA, Miguel BALLESTEROS, Esha BANERJEE, Sebastian BANK, Verginica BARBU MITITELU, Victoria BASMOV, Colin BATCHELOR, John BAUER, Kepa BENGOETXEA, Yevgeni BERZAK, Irshad Ahmad BHAT, Riyaz Ahmad BHAT, Erica BIAGETTI, Eckhard BICK, Agnė BIELINSKIENĖ, Rogier BLOKLAND, Victoria BOBICEV, Loïc BOIZOU, Emanuel BORGES VÖLKER, Carl BÖRSTELL, Cristina BOSCO, Gosse BOUMA, Sam BOWMAN, Adriane BOYD, Kristina BROKAITĖ, Aljoscha BURCHARDT, Marie CANDITO, Bernard CARON, Gauthier CARON, Tatiana CAVALCANTI, Gülşen CEBIROĞLU ERYIĞIT, Flavio Massimiliano CECCHINI, Giuseppe G. A. CELANO, Slavomír ČÉPLÖ, Savas CETIN, Fabricio CHALUB, Ethan CHI, Jinho CHOI, Yongseok CHO, Jayeol CHUN, Alessandra T. CIGNARELLA, Silvie CINKOVÁ, Aurélie COLLOMB, Çağrı ÇÖLTEKIN, Miriam CONNOR, Marine COURTIN, Elizabeth DAVIDSON, Marie-Catherine DE MARNEFFE, Valeria DE PAIVA, Elvis DE SOUZA, Arantza DIAZ DE ILARRAZA, Carly DICKERSON, Bamba DIONE, Peter DIRIX, Kaja DOBROVOLJC, Timothy DOZAT, Kira DROGANOVA, Puneet DWIVEDI, Hanne

Eckhoff, Marhaba Eli, Ali Elkahky, Binyam Ephrem, Olga Erina, Tomaž Erjavec, Aline Etienne, Wograine Evelyn, Richárd Farkas, Hector Fernandez Alcalde, Jennifer Foster, Cláudia Freitas, Kazunori Fujita, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Moa Gärdenfors, Sebastian Garza, Kim Gerdes, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökirmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta González Saavedra, Bernadeta Griciūtė, Matias Grioni, Loïc Grobol, Normunds Grūzītis, Bruno Guillaume, Céline Guillot-Barbance, Tunga Güngör, Nizar Habash, Jan Hajič, Jan Hajič jr., Mika Hämäläinen, Linh Hà Mỹ, Na-Rae Han, Kim Harris, Dag Haug, Johannes Heinecke, Oliver Hellwig, Felix Hennig, Barbora Hladká, Jaroslava Hlaváčová, Florinel Hociung, Petter Hohle, Jena Hwang, Takumi Ikeda, Radu Ion, Elena Irimia, Ọlájídé Ishola, Tomáš Jelínek, Anders Johannsen, Hildur Jónsdóttir, Fredrik Jørgensen, Markus Juutinen, Hüner Kaşıkara, Andre Kaasen, Nadezhda Kabaeva, Sylvain Kahane, Hiroshi Kanayama, Jenna Kanerva, Boris Katz, Tolga Kayadelen, Jessica Kenney, Václava Kettnerová, Jesse Kirchner, Elena Klementieva, Arne Köhn, Abdullatif Köksal, Kamil Kopacewicz, Timo Korkiakangas, Natalia Kotsyba, Jolanta Kovalevskaitė, Simon Krek, Sookyoung Kwak, Veronika Laippala, Lorenzo Lambertino, Lucia Lam, Tatiana Lando, Septina Dian Larasati, Alexei Lavrentiev, John Lee, Phương Lê Hồng, Alessandro Lenci, Saran Lertpradit, Herman Leung, Maria Levina, Cheuk Ying Li, Josie Li, Keying Li, KyungTae Lim, Yuan Li, Nikola Ljubešić, Olga Loginova, Olga Lyashevskaya, Teresa Lynn, Vivien Macketanz, Aibek Makazhanov, Michael Mandl, Christopher Manning, Ruli Manurung, Cătălina Mărănduc, David Mareček, Katrin Marheinecke, Héctor Martínez Alonso, André Martins, Jan Mašek, Hiroshi Matsuda, Yuji Matsumoto, Ryan McDonald, Sarah McGuinness, Gustavo Mendonça, Niko Miekka, Margarita Misirpashayeva, Anna Missilä, Cătălin Mititelu, Maria Mitrofan, Yusuke Miyao, Simonetta Montemagni, Amir More, Laura Moreno Romero, Keiko Sophie Mori, Tomohiko Morioka, Shinsuke Mori, Shigeki Moro, Bjartur Mortensen, Bohdan Moskalevskyi, Kadri Muischnek, Robert Munro, Yugo Murawaki, Kaili Müürisep, Pinkey Nainwani, Juan Ignacio Navarro Horñiacek, Anna Nedoluzhko, Gunta Nešpore-Bērzkalne, Lương Nguyễn Thị, Huyền Nguyễn Thị Minh, Yoshihiro Nikaido, Vitaly Nikolaev, Rattima Nitisaroj, Hanna Nurmi, Stina Ojala, Atul Kr. Ojha, Adédayọ̀ Olúòkun, Mai Omura, Emeka Onwuegbuzia, Petya Osenova, Robert Östling, Lilja Øvrelid, Şaziye Betül Özateş, Arzucan Özgür, Balkız Öztürk Başaran, Niko Partanen, Elena Pascual, Marco Passarotti, Agnieszka Patejuk, Guilherme Paulino-Passos, Angelika Peljak-Łapińska, Siyao Peng, Cenel-Augusto Perez, Guy Perrier, Daria Petrova, Slav Petrov, Jason Phelan, Jussi Piitulainen, Tommi A

Pirinen, Emily Pitler, Barbara Plank, Thierry Poibeau, Larisa Ponomareva, Martin Popel, Lauma Pretkalniņa, Sophie Prévost, Prokopis Prokopidis, Adam Przepiórkowski, Tiina Puolakainen, Sampo Pyysalo, Peng Qi, Andriela Rääbis, Alexandre Rademaker, Loganathan Ramasamy, Taraka Rama, Carlos Ramisch, Vinit Ravishankar, Livy Real, Petru Rebeja, Siva Reddy, Georg Rehm, Ivan Riabov, Michael Riessler, Erika Rimkutė, Larissa Rinaldi, Laura Rituma, Luisa Rocha, Mykhailo Romanenko, Rudolf Rosa, Valentin Roşca, Davide Rovati, Olga Rudina, Jack Rueter, Shoval Sadde, Benoît Sagot, Shadi Saleh, Alessio Salomoni, Tanja Samardžić, Stephanie Samson, Manuela Sanguinetti, Dage Särg, Baiba Saulīte, Yanin Sawanakunanon, Salvatore Scarlata, Nathan Schneider, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Mo Shen, Atsuko Shimada, Hiroyuki Shirasu, Muh Shohibussirri, Dmitry Sichinava, Aline Silveira, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Maria Skachedubova, Aaron Smith, Isabela Soares-Bastos, Carolyn Spadine, Antonio Stella, Milan Straka, Jana Strnadová, Alane Suhr, Umut Sulubacak, Shingo Suzuki, Zsolt Szántó, Dima Taji, Yuta Takahashi, Fabio Tamburini, Takaaki Tanaka, Samson Tella, Isabelle Tellier, Guillaume Thomas, Liisi Torga, Marsida Toska, Trond Trosterud, Anna Trukhina, Reut Tsarfaty, Utku Türk, Francis Tyers, Sumire Uematsu, Roman Untilov, Zdeňka Urešová, Larraitz Uria, Hans Uszkoreit, Andrius Utka, Sowmya Vajjala, Daniel van Niekerk, Gertjan van Noord, Viktor Varga, Eric Villemonte de la Clergerie, Veronika Vincze, Aya Wakasa, Lars Wallin, Abigail Walsh, Jing Xian Wang, Jonathan North Washington, Maximilan Wendt, Paul Widmer, Seyi Williams, Mats Wirén, Christian Wittern, Tsegay Woldemariam, Tak-sum Wong, Alina Wróblewska, Mary Yako, Kayo Yamashita, Naoki Yamazaki, Chunxiao Yan, Koichi Yasuoka, Marat M. Yavrumyan, Zhuoran Yu, Zdeněk Žabokrtský, Amir Zeldes, Hanzhi Zhu, and Anna Zhuravleva (2020), Universal Dependencies 2.6, `http://hdl.handle.net/11234/1-3226`, LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Lluís Alemany-Puig
0000-0002-3874-991X
lluis.alemany.puig@upc.edu

Quantitative, Mathematical and Computational Linguistics Research Group. Departament de Ciències de la Computació, Universitat Politècnica de Catalunya (UPC), Barcelona, Catalonia, Spain

Ramon Ferrer-i-Cancho
0000-0002-7820-923X
rferrericancho@cs.upc.edu

Quantitative, Mathematical and Computational Linguistics Research Group. Departament de Ciències de la Computació, Universitat Politècnica de Catalunya (UPC), Barcelona, Catalonia, Spain

Lluís Alemany-Puig and Ramon Ferrer-i-Cancho (2024), *The expected sum of edge lengths in planar linearizations of trees*, Journal of Language Modelling, 12(1):1–42
https://dx.doi.org/10.15398/jlm.v12i1.362