[To be read before The Royal Statistical Society in London at the Discussion meeting organized by the Research Section on Wednesday, 6 December 2023, Dr Maria De Iorio in the Chair].

Root and community inference on the latent growth process of a network

Harry Crane, Min Xu*
Department of Statistics
Rutgers University, New Brunswick, NJ, USA

February, 2023

Abstract

Many existing statistical models for networks overlook the fact that most real-world networks are formed through a growth process. To address this, we introduce the PAPER (Preferential Attachment Plus Erdős–Rényi) model for random networks, where we let a random network G be the union of a preferential attachment (PA) tree T and additional Erdős–Rényi (ER) random edges. The PA tree component captures the underlying growth/recruitment process of a network where vertices and edges are added sequentially, while the ER component can be regarded as random noise. Given only a single snapshot of the final network G, we study the problem of constructing confidence sets for the early history, in particular the root node, of the unobserved growth process; the root node can be patient zero in a disease infection network or the source of fake news in a social media network. We propose an inference algorithm based on Gibbs sampling that scales to networks with millions of nodes and provide theoretical analysis showing that the expected size of the confidence set is small so long as the noise level of the ER edges is not too large. We also propose variations of the model in which multiple growth processes occur simultaneously, reflecting the growth of multiple communities, and we use these models to provide a new approach to community detection.

1 Introduction

Network data is ubiquitous. To analyze networks, there are a variety of statistical models such as Erdős–Rényi, stochastic block model (SBM) (Abbe; 2017; Karrer and Newman; 2011; Amini et al.; 2013; Xu et al.; 2018), graphon (Diaconis and Janson; 2007; Gao et al.; 2015), random dot product graphs (Athreya et al.; 2017; Xie and Xu; 2019), latent space models (Hoff et al.; 2002), configuration graphs (Aiello et al.; 2000), and more. These models usually operate by specifying some structure, such as community structure in the case of SBM, and then adding independent random edges in a way that reflects the structure. The order in which the edges are added is of no importance to these models.

In contrast, real world networks are often formed from growth processes where vertices and edges are added sequentially. This motivates the development of Markovian preferential attachment (PA) models for networks (Barabási and Albert; 1999; Barabási; 2016) which produce a sequence of networks $\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n$ where $\boldsymbol{G}_1$ starts as a single node which we call the root node and, at

*Corresponding author: Min Xu, Department of Statistics, Rutgers University, New Brunswick, NJ 08854, USA (E-mail: *mx76@stat.rutgers.edu*).

each iteration, we add a new node and new edges. PA models naturally produce networks with sparse edges, heavy-tailed degree distributions, and strands of chains as well as pendants (several degree 1 vertices linked to a single vertex), which are important features of real world networks that are difficult to reproduce under a non-Markovian model, as observed by Bloem-Reddy and Orbanz (2018).

Although Markovian models are often more realistic, they have not been as widely used in network data analysis as, say SBM, because, whereas SBM is useful for recovering the community structure of a network, it is not obvious what structural information Markovian models could extract from a network. Recently however, seminal work from a series of applied probability papers (e.g. Bubeck, Devroye and Lugosi (2017); Bubeck et al. (2015)) demonstrate that Markovian models can indeed recover useful structure: these papers show that, surprisingly, when $\boldsymbol{G}_n$ is a random PA tree, one can infer the early history of $\boldsymbol{G}_n$, such as the root node, even as the size of the tree tends to infinity. Although these results are elegant, they are theoretical; their confidence set construction involves large constants that render the result too conservative. Moreover, most algorithms apply only to tree-shaped networks, which prohibitively limits their application since trees are rarely encountered in practice.

To overcome these problems, we propose a Markovian model for networks which we call Preferential Attachment Plus Erdős–Rényi, or PAPER for short. We say that $\boldsymbol{G}_n$ has the PAPER distribution if it is generated by adding independent random edges to a preferential attachment tree $\boldsymbol{T}$. The latent PA tree captures the growth process of the network whereas the ER random edges can be interpreted as additional noise. Given only a single snapshot of the final graph $\boldsymbol{G}_n$, we study how to infer the early history of the latent tree $\boldsymbol{T}$, focusing on the concrete problem of constructing confidence sets for the root node that can attain the nominal coverage. We give a visual illustration of the PAPER model and the inference problem in Figure 1.

Because we do not know which edges of $\boldsymbol{G}_n$ correspond to the tree and which are noise, most existing methods are not directly applicable. We therefore propose a new approach in which we first give the nodes new random labels which induce, for a given observation of the network $\boldsymbol{G}_n$, a posterior distribution of both the latent tree and the latent arrival ordering of the nodes. Then, we sample from the posterior distribution to construct a credible set for the inferential target, e.g. the root node. Bayesian inference statements usually do not have frequentist validity but we prove in our setting that that the level $1-\epsilon$ credible set for the root node has frequentist coverage at exactly the same level.

In order to efficiently sample from the posterior distribution of the latent ordering and the latent tree, we present a scalable Gibbs sampler that alternatingly samples the ordering and the tree. The algorithm to generate the latent ordering is based on our previous work (Crane and Xu; 2021) which studies inference in the tree setting. The algorithm to generate the latent tree operates by updating the parent of each of the nodes iteratively. The overall runtime complexity of one iteration of the outer loop is generally $O(m+n\log n)$ (where m is the number of edges) and the algorithm can scale to networks of up to a million nodes.

Since a trivial confidence set for the root node is the set of all the nodes, it is important to be able to bound the size of a confidence set. In particular, the presence of noisy Erdős–Rényi edges in the PAPER model motivates an interesting question: how does the size of the confidence set increase with the noise level? In this paper, we give an initial answer to this question under two specific settings of the preferential attachment mechanism: linear preferential attachment (LPA) and uniform attachment (UA). For LPA, we prove that the size of our proposed confidence set does not increase with the number of nodes n so long as the noisy edge probability is less than $n^{-1/2}$ and for UA, we prove that the size is bounded by n^{γ} for some $\gamma < 1$ so long as the noisy edge

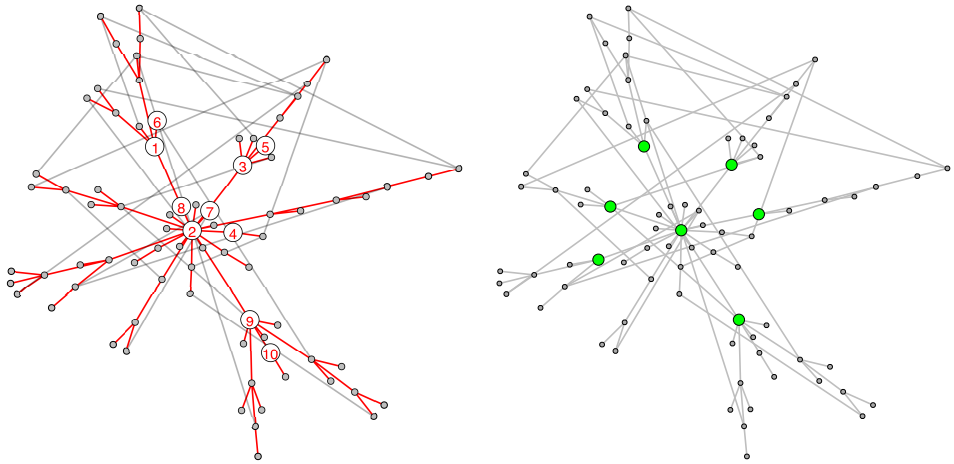


Figure 1: **Left**: illustration of PAPER model; nodes have latent time ordering (only first 10 orderings shown); the red edges form the latent tree while gray edges are Erdős–Rényi. **Right**: 80% confidence set for the root node (node number 1) constructed from the unlabeled graph.

probability is less than $\log(n)/n$. Our analysis shows that the phenomenon discovered by Bubeck, Devroye and Lugosi (2017), that there exists confidence sets for the root node of $O(1)$ size, is robust to the presence of noise.

Many real world networks often have community structures. In such cases, it would be unrealistic to assume that the network originates from a single root node. We therefore propose variations of the PAPER model in which K growth processes occur simultaneously from K root nodes. Each of K root nodes can be interpreted as being locally central with respect to a community subgraph. In the multiple roots model, there is no longer a latent tree but rather a latent forest (union of disjoint trees), where the components of the forest can naturally be interpreted as the different communities of the network. We provide model formulation that allows K to be either be fixed or random. To analyze networks with multiple roots, we use essentially the same inferential approach and Gibbs sampling algorithm that that we develop for the single root setting, with minimal modifications.

By looking at the posterior probability that a node is in a particular tree–community, we can estimate the community membership of each of the nodes. Compared with say the stochastic block model, the PAPER model approach to community recovery has the advantage that the inference quality improves with sparsity, that we can handle heavy-tailed degree distribution without a high-dimensional degree correction parameter vector, and that the posterior root probabilities also identify the important nodes in the community. Empirically, we show that our approach has competitive performance on two benchmark datasets and we find that our community membership estimate is more accurate for nodes with high posterior root probability than for the more peripheral nodes. We also use the PAPER model to conduct an extensive analysis of a statistician co-authorship network curated by Ji and Jin (2016) where we recover a large number of communities that accurately reflect actual research communities in statistics.

We have implemented our inference algorithm in a Python package called `paper-network`, which can be installed via command `pip install paper-network`. The code, example scripts, and documentation are all publicly available at `https://github.com/nineisprime/PAPER`.

Outline for the paper: In Section 2, we define the PAPER model in both the single root and

multiple roots setting. We also formalize the problem of root inference and review related work. In Section 3, we describe our approach to the root inference problem, which is to randomize the node labels and analyze the resulting posterior distribution. We also show that the Bayesian inferential statements have frequentist validity. In Section 4, we give a sampling algorithm for computing the posterior probabilities. In Section 5, we provide theoretical bounds on the size of our proposed confidence sets and in Section 6, we provide empirical study on both simulated and large scale real world networks.

We use the following notation throughout the paper:

- We take all graphs to be undirected. Given two labeled graphs $\boldsymbol{g}$ and $\boldsymbol{g}'$ defined on the same set of nodes, we write $\boldsymbol{g}+\boldsymbol{g}'$ as the resulting graph if we take the union of the edges in $\boldsymbol{g}$ and $\boldsymbol{g}'$ and collapse any multi-edges. We also write $\boldsymbol{g} \subset \boldsymbol{g}'$ if $\boldsymbol{g}$ is a subgraph of $\boldsymbol{g}'$.
- For a labeled graph $\boldsymbol{g}$, we write $D_{\boldsymbol{g}}(u)$ as the degree of node u in graph $\boldsymbol{g}$ and $N_{\boldsymbol{g}}(u)$ as the set of neighbors of u (all nodes directly connected to u) with respect to $\boldsymbol{g}$; we write $V(\boldsymbol{g})$ and $E(\boldsymbol{g})$ as the set of vertices and edges of $\boldsymbol{g}$ respectively.
- For an integer n, we write $[n] := \{1, 2, \ldots, n\}$. For a countable set A, we write $|A|$ as the cardinality of A. For two sets A, B of the same cardinality, we write $\mathrm{Bi}(A, B)$ as the set of bijections between them. For a vector π, we let $\pi_{1:K}$ be the sub-vector $(\pi_1, \pi_2, \ldots, \pi_K)$.
- Given a finite set V' of the same cardinality of $V(\boldsymbol{g})$ and given a bijection $\rho \in \mathrm{Bi}(V(\boldsymbol{g}), V')$, we write $\rho\boldsymbol{g}$ to denote a relabeled graph where a pair $(u', v') \in V' \times V'$ is an edge in $\rho\boldsymbol{g}$ if and only if $(u, v) \in V(\boldsymbol{g}) \times V(\boldsymbol{g})$ is an edge in $\boldsymbol{g}$.
- Throughout the paper, we use capital font (e.g. $\boldsymbol{G}$) to denote random objects and lower case font to denote fixed objects. Graphs are represented via bold font.

2 Model and Problem

We first describe the model and inference problem in the single root setting and then extend the definition to the setting of having fixed K roots and having random K roots.

2.1 PAPER model

Definition 1. The affine preferential attachment tree model, which we denote by $\mathrm{APA}(\alpha, \beta)$ for parameters $\alpha, \beta \in \mathbb{R}$, generates an increasing sequence $\boldsymbol{T}_1 \subset \boldsymbol{T}_2 \subset \ldots \subset \boldsymbol{T}_n$ of random trees where $\boldsymbol{T}_t$ is a tree with t nodes and where nodes are labeled by their arrival time so that $V(\boldsymbol{T}_t) = [t]$. The first tree $\boldsymbol{T}_1 = \{1\}$ is a singleton node, which we refer to as the *root node*, and for $t > 2$, we define the transition kernel $\mathbb{P}(\boldsymbol{T}_t \,|\, \boldsymbol{T}_{t-1})$ in the following way: given $\boldsymbol{T}_{t-1}$, we add a node labeled t and a random edge (t, w_t) to obtain $\boldsymbol{T}_t$, where the existing node $w_t \in [t-1]$ is chosen with probability

$$\frac{\beta D_{\boldsymbol{T}_{t-1}}(w_t) + \alpha}{\beta 2(t-2) + \alpha(t-1)}. \tag{1}$$

To ensure that (1) is always non-negative, we require either $\alpha, \beta \geq 0$ or, if $\beta < 0$, then $\alpha = -c\beta$ for some integer $c > 0$. We may verify that (1) describes a valid probability distribution by noting that $\boldsymbol{T}_{t-1}$ always has $t-2$ edges and $t-1$ nodes. Before continuing onto the PAPER model, we consider some specific examples of APA trees:

1. setting $\alpha = 1, \beta = 0$ means that we select w_t uniformly at random from $V(\boldsymbol{T}_{t-1})$. This yields the uniform attachment (UA) random tree. The resulting degree distribution has exponential tail and the maximum degree is of order $\log n$ (Na and Rapoport; 1970; Addario-Berry and Eslava; 2018).

2. Setting $\alpha = 0, \beta = 1$ means that we select w_t with probability proportional to the degree $D_{\boldsymbol{T}_{k-1}}(w_t)$. This yields the linear preferential attachment random (LPA) tree. LPA has heavy-tailed degree distribution and a maximum degree is of order $\sqrt{n}$ (Bollobás et al.; 2001; Peköz et al.; 2014).

3. We may also set β as -1 and α as some positive integer so that the maximum degree of any node is α. This may be interpreted as an uniform attachment tree growing on top of a background infinite α-regular tree (Khim and Loh; 2017).

We may generalize Definition 1 by defining a nonparametric function $\phi : \mathbb{N} \to [0, \infty)$ and choose w_t with probability proportional to $\phi(D_{\boldsymbol{T}_{t-1}}(w_t))$. In this paper however, we focus only on the case where ϕ is an affine function.

Definition 2. To model a general network, we define the PAPER(α, β, θ) (Preferential Attachment Plus Erdős–Rényi) model parametrized by $\alpha, \beta \in \mathbb{R}$ and $\theta \in [0, 1]$. We say that a random graph $\boldsymbol{G}_n$ distributed according to the PAPER(α, β, θ) model if

$$\boldsymbol{G}_n = \boldsymbol{T}_n + \boldsymbol{R}_n,$$

where $\boldsymbol{T}_n \sim$ APA(α, β) and $\boldsymbol{R}_n \sim$ Erdős–Rényi(θ) are independent random graphs defined on the same set of vertices $[n]$.

Since we collapse any multi-edges that occur when we add $\boldsymbol{R}_n$ to $\boldsymbol{T}_n$, we may view $\boldsymbol{R}_n$ equivalently as an ER random graph defined on potential edges excluding those already in the tree $\boldsymbol{T}_n$. The PAPER model can produce networks with either light tailed or heavy tailed degree distribution depending on the choice of the parameters α and β. It produces features that are commonly seen in real world networks but absent from non-sequential models like SBM, such as pendants (a node with several degree-1 node attached to it) and chains of nodes; see Figure 2. It also assigns a non-zero probability to any connected graph, in contrast to the general preferential attachment graph model where a fixed $m > 1$ edges are added at every iteration (Barabási and Albert; 1999). In computer science terminology, $\boldsymbol{G}_n$ is a *planted tree model* where the signal $\boldsymbol{T}_n$ is planted in an ER random graph $\boldsymbol{R}_n$ in the same sense that stochastic block model is often referred to as the planted partition model.

An alternative way to define the PAPER model is to specify the total number of edges m in the final graph and generate $\boldsymbol{R}_n$ as a uniformly random graph with $m - (n-1)$ edges (since a tree with n nodes always has $n - 1$ edges). This is equivalent to the PAPER(α, β, θ) model where we *condition* on the event that the final graph $\boldsymbol{G}_n$ has m edges. To simplify exposition, we use PAPER to refer to this conditional model as well.

Remark 1. We may view the PAPER(α, β, θ) model as a Markovian process over a sequence of networks $\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n$. We define the transition kernel $\mathbb{P}(\boldsymbol{G}_t \mid \boldsymbol{G}_{t-1})$ for $t \geq 3$ by first adding a new node labeled t, then adding a new tree edge (t, w_t) where w_t is chosen with probability (1), and then, for each existing node $j \in [t-1]$ not equal to w_t, we independently add a noise edge (t, j) with probability θ.

Interestingly, when $\alpha = 1$ and $\beta = 0$, we see that the PAPER model is the conditional distribution of an Erdős–Rényi graph $\boldsymbol{G}$ conditional on the event that, for some fixed ordering ρ of

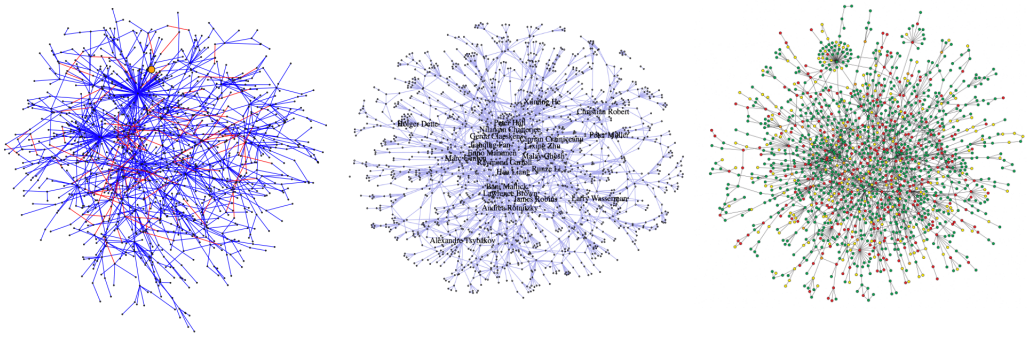


Figure 2: **Left**: PAPER graph with $\alpha = 1, \beta = 1$; **Center:** co-authorship graph from Ji and Jin (2016); **Right:** protein-protein interaction graph from Jeong et al. (2001).

the nodes, the sequence of induced subgraphs $\boldsymbol{G} \cap \{\rho_1, \ldots, \rho_t\}$ for $t \in [n]$ are all connected. In Section 2.3, we extend the PAPER model so that the noise edge probability is allowed to depend on the time t and the state of the tree at time t.

Remark 2. Under APA(α, β) model, the probability of generating a given tree has a closed form expression: $\mathbb{P}(\boldsymbol{T}_n = \boldsymbol{t}_n) = \frac{\prod_{v\in[n]} \prod_{j=1}^{D_{\boldsymbol{t}_n}(v)-1} (\beta j + \alpha)}{\prod_{t=3}^{n} 2(t-2)\beta + (t-1)\alpha}$. The important consequence is that the likelihood depends on the tree $\boldsymbol{t}_n$ only through its degree distribution $D_{\boldsymbol{t}_n}(\cdot)$. Hence, any two trees with the same degree distribution has the same likelihood; Crane and Xu (2021) refers to this property as *shape-exchangeability*. We give the likelihood expression for the multiple roots models and the PAPER model in Section S1.1 of the Appendix.

Remark 3. It is known that the degree distribution of an APA(α, β) tree has an asymptotic limit. For example, if $\beta = 1$ and $\alpha > 0$, then we have by Van Der Hofstad (2016, Theorem 8.2) that $\frac{1}{n}\sum_{t=1}^{n} \mathbb{1}\{D_{\boldsymbol{T}_n}(t) = k\} \to \frac{2+\alpha}{3+2\alpha} \prod_{j=1}^{k-1} \frac{j+\alpha}{j+3+2\alpha}$ as $n \to \infty$ uniformly over all k. The limiting distribution is approximately a power law where the number of nodes with degree k is proportional to $k^{-(3+\alpha)}$ (see Van Der Hofstad (2016, Section 8.4)). Since the ER graph $\boldsymbol{R}_n$ only adds an expected additional degree of at most $n\theta$ to every node, we see that, when θ is small, the PAPER graph can have heavy-tailed degree distribution without any additional degree correction parameters.

Single root inference problem: Let $\boldsymbol{G}_n \sim \text{PAPER}(\alpha, \beta, \theta)$ be a random graph. As the nodes of $\boldsymbol{G}_n$ are labeled by their arrival time, our observation is the unlabeled shape $\text{sh}(\boldsymbol{G}_n)$, that is, the network $\boldsymbol{G}_n$ with the labels removed. Our goal is to construct a subset of nodes that is guaranteed to contain the true root node (node with arrival time 1) with probability at least $1 - \epsilon$. Since we need to refer to specific nodes of $\text{sh}(\boldsymbol{G}_n)$, we give the nodes of $\text{sh}(\boldsymbol{G}_n)$ names from an arbitrary alphabet $\mathcal{U}_n$ of n elements to form a labeled graph $\boldsymbol{G}_n^*$ such that $V(\boldsymbol{G}_n^*) = \mathcal{U}_n$. We take $\boldsymbol{G}_n^*$ as our observation from this point on.

We note that there exists an unobserved label bijection $\rho \in \text{Bi}([n], \mathcal{U}_n)$ such that $\rho \boldsymbol{G}_n = \boldsymbol{G}_n^*$. This unobserved ρ captures precisely the arrival time of the nodes in that for any time $t \in [n]$, the node with label ρ_t in $\boldsymbol{G}_n^*$ is exactly node with arrival time t in $\boldsymbol{G}_n$. In particular, node ρ_1 of the observed graph $\boldsymbol{G}_n^*$ is the true root node. To illustrate the setting clearly, we provide a concrete example in Figure 3.

Definition 3. For $\epsilon \in (0, 1)$, we say that a set $C_\epsilon(\boldsymbol{G}_n^*) \subset \mathcal{U}_n$ is a level $1 - \epsilon$ confidence set for the root node if

$$\mathbb{P}\big(\rho_1 \in C_\epsilon(\boldsymbol{G}_n^*)\big) \geq 1 - \epsilon. \tag{2}$$

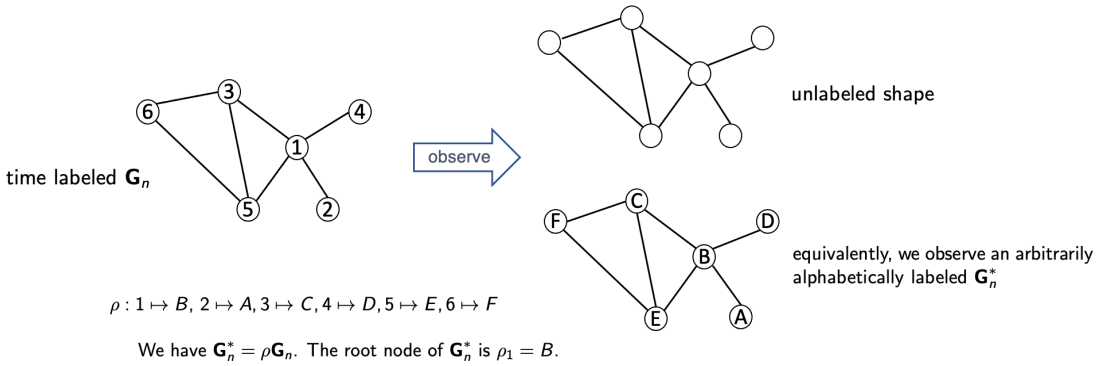


Figure 3: Our observation is the unlabeled shape or alphabetically labeled $\boldsymbol{G}_n^*$ instead of time labeled $\boldsymbol{G}_n$. There exists an unobserved ordering $\rho \in \mathrm{Bi}([n], \mathcal{U}_n)$ such that $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n$.

One may construct a trivial confidence set for the root nodes by taking the set of all the nodes. We aim therefore to make the confidence set $C_\epsilon(\cdot)$ as small as possible. Although we focus on the problem of root inference, the approach that we develop is applicable to more general problems such as inferring the first two or three nodes or inferring the arrival time of a particular node.

Remark 4. It is important to note that $\boldsymbol{G}_n^*$ may have multiple nodes that are indistinguishable once the node labels are removed, which may lead to the paradoxical scenario that which node of $\boldsymbol{G}_n^*$ correspond to the true root node depends on the choice of the label bijection ρ. Luckily, this is a technical issue that does not pose a problem so long as we restrict ourselves to confidence sets $C_\epsilon(\cdot)$ that are labeling equivariant in that they do not depend on the specific node labeling. Labeling equivariance is a very weak condition that only rules out confidence sets that can access side information about the nodes somehow.

Formally, we note that there may exist $\rho, \rho' \in \mathrm{Bi}([n], \mathcal{U}_n)$ where $\rho_1 \neq \rho_1'$ but both satisfy $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n = \rho' \boldsymbol{G}_n$; in other words, root node can only be well-defined up to an automorphism. We illustrate a concrete example in Figure 4. We define $C_\epsilon(\cdot)$ to be *labeling equivariant* if, for all $\tau \in \mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$, we have $\tau C_\epsilon(\boldsymbol{G}_n^*) = C_\epsilon(\tau \boldsymbol{G}_n^*)$; if the confidence set algorithm contains randomization (to break ties for example), then we say it is labeling equivariant if $\tau C_\epsilon(\boldsymbol{G}_n^*) \stackrel{d}{=} C_\epsilon(\tau \boldsymbol{G}_n^*)$ for all $\tau \in \mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$. If a confidence set $C_\epsilon(\cdot)$ is labeling equivariant, then for any $\rho, \rho' \in \mathrm{Bi}([n], \mathcal{U}_n)$ such that $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n = \rho' \boldsymbol{G}_n$, we have that $(\rho' \circ \rho^{-1}) \boldsymbol{G}_n^* = \boldsymbol{G}_n^*$ and hence,

$$\rho_1 \in C_\epsilon(\boldsymbol{G}_n^*) \Leftrightarrow (\rho' \circ \rho^{-1}) \rho_1 \in (\rho' \circ \rho^{-1}) C_\epsilon(\boldsymbol{G}_n^*) \Leftrightarrow \rho_1' \in C_\epsilon((\rho' \circ \rho^{-1}) \boldsymbol{G}_n^*) \Leftrightarrow \rho_1' \in C_\epsilon(\boldsymbol{G}_n^*).$$

Therefore, the coverage probability (2) does not depend on the choice of ρ.

2.2 Multiple roots models

Many real world networks have multiple communities that grow simultaneously form multiple sources. The APA model allows for only one root node in the graph but we can augment the model to describe networks that grow from multiple roots. When there are K roots, we start the growth process with an initial network of K singleton nodes and attach each new node to an existing node w_t with probability proportional to $\beta \cdot (\text{degree of } w_t) + \alpha$ as before.

However, one complication is that when $\alpha = 0$, the probability of attaching to a singleton node is 0. Thus, for convenience, we give each root node an unobserved imaginary self-loop edge for the purpose of computing the attachment probabilities.

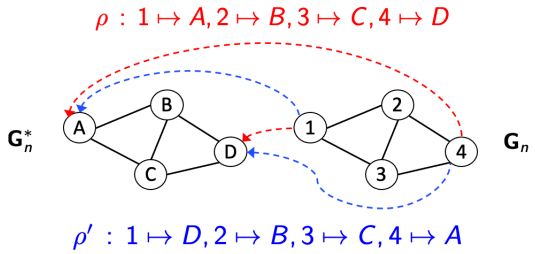


Figure 4: Both ρ (red) and ρ' (blue) are distinct bijections in $\mathrm{Bi}([n], \mathcal{U}_n)$ but they both satisfy $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n = \rho' \boldsymbol{G}_n$. The root node is D according to ρ but A according to ρ'. Note that nodes A and D are indistinguishable if the labels are removed.

Definition 4. We first define the $\mathrm{APA}(\alpha, \beta, K)$ model for a random forest of K disjoint component trees: let $K \in \mathbb{N}$ and for $t \in S := \{1, 2, \ldots, K\}$ (the set S is the set of root nodes), let $\boldsymbol{F}_t$ be the set of singleton nodes $1, 2, \ldots, t$. For $t > K$, we define the transition kernel $\mathbb{P}(\boldsymbol{F}_t \,|\, \boldsymbol{F}_{t-1})$ in the following way: given $\boldsymbol{F}_{t-1}$, we add a new node t and a new random edge (t, w_t) where the existing node $w_t \in [t-1]$ is chosen with probability

$$\frac{\beta D_{\boldsymbol{F}_{t-1}}(w_t) + 2\beta \mathbb{1}\{w_t \in S\} + \alpha}{(2\beta + \alpha)(t-1)}. \tag{3}$$

We then say that a random graph $\boldsymbol{G}_n \sim \mathrm{PAPER}(\alpha, \beta, K, \theta)$ if $\boldsymbol{G}_n = \boldsymbol{F}_n + \boldsymbol{R}_n$ where $\boldsymbol{F}_n \sim \mathrm{APA}(\alpha, \beta, K)$ and $\boldsymbol{R}_n \sim \mathrm{ER}_\theta$ is an Erdős–Rényi random graph independent of $\boldsymbol{F}_n$ defined on the same set of nodes $[n]$. We refer to this setting as the *fixed K setting*. In contrast, we refer to the $\mathrm{PAPER}(\alpha, \beta, \theta)$ model in Section 2.1 as the *single root setting*.

We can verify the normalization term (3) by noting that each root node starts with one imaginary self-loop and that we add one node and one edge at every iteration. The theory of Polya's urn immediately implies that the number of nodes in each of the K component trees, divided by n, has the asymptotic distribution of $\mathrm{Dirichlet}(\frac{1}{K}, \ldots, \frac{1}{K})$.

To deal with networks in which the number of roots K is unknown, we propose a variation of the PAPER model with random K number of roots. We can express the model as a sequential growth process where every newly arrived node has some probability of becoming a new root. Similar to the fixed K setting, we give each new root node an imaginary self-loop edge for the purpose of determining the attachment probabilities.

Definition 5. We first define the $\mathrm{APA}(\alpha, \beta, \alpha_0)$ model for a random forest graph: let $\boldsymbol{F}_1$ be a singleton node and let $S = \{1\}$. For $k > 1$, we define the transition kernel $\mathbb{P}(\boldsymbol{F}_t \,|\, \boldsymbol{F}_{t-1})$ in the following way: given $\boldsymbol{F}_{t-1}$, we add a new node t. With probability

$$\frac{\alpha_0}{(2\beta + \alpha)(t-1) + \alpha_0},$$

we let t be a new root node to form $\boldsymbol{F}_t$ and add t to set S. Or, we add a new edge (t, w_t) to $\boldsymbol{F}_{t-1}$ to obtain $\boldsymbol{F}_t$ where the existing node $w_t \in [t-1]$ is chosen with probability

$$\frac{\beta D_{\boldsymbol{F}_{t-1}}(w_t) + \alpha + 2\beta \mathbb{1}\{w_t \in S\}}{(2\beta + \alpha)(t-1) + \alpha_0}.$$

Note that the resulting set of root nodes $S \subset [n]$ of $\boldsymbol{F}_n$ is a random set.

We then say that a random graph $\boldsymbol{G}_n$ has the PAPER$(\alpha, \beta, \alpha_0, \theta)$ distribution if $\boldsymbol{G}_n = \boldsymbol{F}_n + \boldsymbol{R}_n$ where $\boldsymbol{F}_n \sim$ APA$(\alpha, \beta, \alpha_0)$ and $\boldsymbol{R}_n \sim$ ER(θ) is an Erdős–Rényi random graph independent of $\boldsymbol{F}_n$ defined on the same set of nodes $[n]$. We refer to this setting as the *random K setting*.

In the random K setting, each node has some probability of becoming a new root node and creating a new component tree in the same way as the Dirichlet process mixture model, which is often called the Chinese restaurant process. Therefore, the expected number of component trees is $(1 + o(1))\frac{\alpha_0}{(2\beta+\alpha)} \log n$ (Crane; 2016, Section 2.2).

Multiple roots inference problem: We observe $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n$ for an unknown label bijection $\rho \in \text{Bi}([n], \mathcal{U}_n)$. In both the APA$(\alpha, \beta, K)$ and the APA$(\alpha, \beta, \alpha_0)$ models, the root nodes is a set S which is fixed to be $[K]$ in the first model and random in the second model. Intuitively, we interpret S as a set of *local* roots, where each root is central with respect to a specific community or sub-network represented by a component tree in the forest $\boldsymbol{F}_n$ in Definition 4 or 5. The root inference problem is then, for a given $\epsilon \in (0, 1)$, to construct a confidence set $C_\epsilon(\boldsymbol{G}_n^*)$ such that

$$\mathbb{P}\big(\rho S \subseteq C_\epsilon(\boldsymbol{G}_n^*)\big) \geq 1 - \epsilon.$$

We illustrate this notion of local roots in a synthetic example in Figure 6.

Remark 5. **(Interpretation of community under the PAPER model)**

The disjoint component trees of $\boldsymbol{F}_n$ induce a community structure on the graph $\boldsymbol{G}_n$. This way of modeling community by adding Erdős–Rényi noise to disjoint subgraphs follows the same spirit as stochastic block model (SBM): a SBM with K communities, p as the within-community edge probability, and $q < p$ as the between-community edge probability can be similarly defined as first generating K disjoint ER$(\frac{p-q}{1-q})$ graphs on each of the communities and then taking the union of that with ER(q) noisy edges on all the nodes, collapsing multi-edges.

The PAPER notion of community is however different from that described by SBM. The PAPER notion of community is based on Markovian growth process and intuitively characterized by the imbalance of spanning trees on a network, that is, we believe a network to contain multiple communities if the spanning trees of the network tend to be highly imbalanced (see Figure 5), which would suggest that the network is very unlikely to have been formed from a single homogeneous growth process.

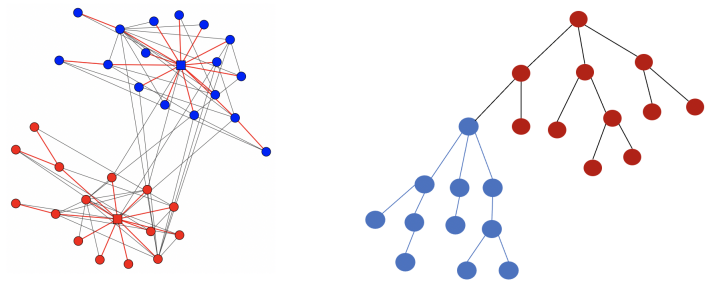


Figure 5: The karate club network (**left**) has two true communities. Most spanning trees of the whole karate club network would be imbalanced (such as the tree on the **right**), showing that the karate club network is very unlikely to have been formed from a single homogeneous growth process and hence very likely to contain multiple communities.

The PAPER model also produces more within-community edges than between-community edges because each community has a spanning tree. However, since a tree on n nodes only has $n-1$ edges,

the difference in the within-community edge density and the between-community edge density is diminishingly small when the noise level θ is of an order larger than $\omega(\frac{1}{n})$. In this case, the peripheral leaf nodes of a community-tree become impossible to cluster but it is still possible to recover the root node of each of the community-trees, as our experimental results show. One disadvantage of the PAPER notion of community is that it is not able to capture non-assortative clusters where nodes in the same clusters are unlikely to form edges.

The PAPER notion of community is appropriate in many application. For example, for a co-authorship network where there exists an underlying growth process, our empirical analysis in Section 6.5 shows that the PAPER model captures clusters that accurately reflect salient research communities. We can also combine both notions by a PAPER-SBM mixture model, where we generate a preferential attachment forest $\boldsymbol{F}_n$ via the mechanism described in Definition 4 or 5, then, for every pair of nodes u and v, we add a noisy edge (u, v) with probability θ_1 if u and v belong to the same tree in $\boldsymbol{F}_n$ and with a different probability θ_2 if u and v belong to different trees. The inference method and algorithm that we develop in this manuscript can extend to such a PAPER-SBM mixture model, but the computational run-time would be substantially slower. We relegate a detailed study of a PAPER-SBM mixture model to a future work.

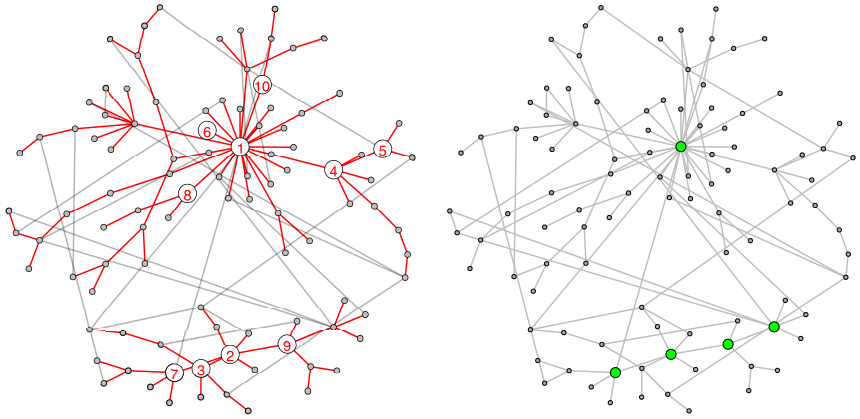


Figure 6: **Left**: illustration of PAPER model with $K = 2$ underlying trees; nodes have latent time ordering (only first 10 orderings shown); the red edges form the latent tree while gray edges are Erdős–Rényi. **Right**: 80% confidence set for the set of root nodes (node number 1 for tree 1 and node number 2 for tree 2) constructed from the unlabeled graph.

2.3 Sequential noise models

As suggested in Remark 1, PAPER model is a special case of a general Markovian process over a sequence of networks $\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_n$ based on a latent sequence of trees $\boldsymbol{T}_1, \boldsymbol{T}_2, \ldots, \boldsymbol{T}_n$. In the general framework, we specify the transition kernel $\mathbb{P}(\boldsymbol{G}_t \,|\, \boldsymbol{G}_{t-1})$ by specifying two stages:

1. (tree stage) $\mathbb{P}(\boldsymbol{T}_t \,|\, \boldsymbol{T}_{t-1}, \boldsymbol{G}_{t-1})$ which adds one node t and one tree edge and
2. (noise stage) $\mathbb{P}(\boldsymbol{G}_t \,|\, \boldsymbol{T}_t, \boldsymbol{G}_{t-1})$ which adds more random edges to obtain $\boldsymbol{G}_t$.

We can of course define $\mathbb{P}(\boldsymbol{G}_t \,|\, \boldsymbol{G}_{t-1})$ without having an underlying tree but the key insight of our approach is that augmenting the model with the latent tree $\boldsymbol{T}_n$ greatly facilitates the design of tractable models and inference algorithms because calculations on trees are easy and fast. In

addition, the latent tree has a real world interpretation as the recruitment history – a tree edge between nodes (u, v) implies that node u recruited node v into the network.

In the noise stage, if we independently adds noise edges between the new node t and the existing nodes with the same probability θ, then we get back the single root PAPER model. More generally, we can let the noise edge probability depend on the time t and the state of the graph at time t. We define the following extension which we refer to as the **seq-PAPER** model with parameters $(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta})$:

Definition 6. We start with a singleton root node $\boldsymbol{T}_1 = \boldsymbol{G}_1 = \{1\}$. At time $t = 2$, we add node 2 and attach it to node 1. At time $t \geq 3$:

1. (tree stage) We add new node t; we select node an existing node $w_t \in [t-1]$ with probability $\frac{\beta D_{\boldsymbol{T}_{t-1}}(w_t)+\alpha}{2(t-2)\beta+(t-1)\alpha}$ and add edge (t, w_t) to $\boldsymbol{T}_{t-1}$ to form $\boldsymbol{T}_t$;
2. (noise stage) for each existing node $j \in [t-1]$, we add edge (t, j) independently with probability

$$q_j := \theta \frac{\tilde{\beta} D_{\boldsymbol{T}_{t-1}}(j) + \tilde{\alpha}}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \wedge 1. \tag{4}$$

   It is possible that we add the tree edge (j, w_t) in the noise stage in which case we collapse the multi-edge.

In general, we may take $\tilde{\beta} = \beta$ and $\tilde{\alpha} = \alpha$ but we allow them to be distinct in the model definition for greater flexibility. We discuss parameter estimation in Section S3.5.4 of the Appendix.

When t is large, the independent Bernoulli generative process approximates a Poisson growth model (see e.g. Sheridan et al. (2008)) where we first generate $M \sim \text{Poisson}(\theta)$, and then repeat M times the procedure where we draw an existing node $j \in [t-1]$ with probability q_j (also with replacement) and then add the edge (t, j) to the random network, collapsing multi-edges if any are formed. We thus add an average of approximately θ noise edges at each time step. In contrast, under the PAPER model where the noise edge probability is θ, we add on average $(t-2) \cdot \theta$ noise edges at time t.

The approximation error between the Bernoulli mechanism and the Poisson mechanism, in each iteration t, converges to 0 in total variation distance as t increases; see rigorous statement and proof in Proposition S4 of Section S1.2 in the Appendix. However, it is important to note that the two mechanisms could still produce final random graphs whose overall distributions have total variation distance bounded away from 0. For example, UA or LPA trees are known to be sensitive to initialization so that different initial seeds could lead to very different distributions over the final observed graph, see e.g. Bubeck et al. (2015) and Curien et al. (2015). In this work, we prefer the Bernoulli generative process in order to simplify the inference algorithm. Even with the Bernoulli approximation however, inference under the sequential setting is much more computationally intensive than the vanilla PAPER model.

A more realistic extension of the seq-PAPER model is to replace the tree degree $D_{\boldsymbol{T}_{t-1}}(j)$ with the graph degree $D_{\boldsymbol{G}_{t-1}}(j)$ in the noise probability 4. This small change unfortunately leads to additional significant slowdown in the resulting inference algorithm; see Remark 9 for more detail. We note that an even more sophisticated model of sequential noise is one where the additional noise edges are generated by a random walk mechanism (Bloem-Reddy and Orbanz; 2018); Bloem-Reddy and Orbanz (2018) proposes a sequential Monte Carlo inference method which may not scale well to large networks.

We have so far considered additive noise where new edges are added to the network. We can also model deletion noise where each tree edge is removed from the observed network independently

with some probability $\eta > 0$. Having deletion noise under the vanilla PAPER model can adversely increase the size of the confidence set for the root node. However, the seq-PAPER model is much more resilient to deletion noise, especially when $\tilde{\beta} = \beta$ and $\tilde{\alpha} = \alpha$ since the noise edges also contain sequential information. To be precisely, we define the seq-PAPER$^*(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta}, \eta)$ as the model where we first generate $\boldsymbol{G}_n$ according to the seq-PAPER$(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta})$ model with latent spanning tree $\boldsymbol{T}_n$; we then remove each edge of $\boldsymbol{T}_n$ from the final graph $\boldsymbol{G}_n$ independently with probability η.

2.4 Related Work

Many researchers in statistics (Kolaczyk; 2009), computer science (Bollobás et al.; 2001), engineering, and physics (Callaway et al.; 2000) have been interested in the probabilistic properties of various random growth processes of networks, including the preferential attachment model (Barabási and Albert; 1999). Recently however, the specific problem of root inference on trees has received increased attention.

These efforts began with the ground-breaking work of Bubeck, Devroye and Lugosi (2017); Bubeck et al. (2015); Bubeck, Eldan, Mossel and Rácz (2017), which shows that, given an observation of an LPA or UA tree of size n, for any $\epsilon \in (0, 1]$, one can construct asymptotically valid confidence sets for the root node with size $K_{LPA}(\epsilon)$ and $K_{UA}(\epsilon)$ for LPA or UA trees respectively. Importantly and surprisingly, $K_{LPA}(\epsilon)$ and $K_{UA}(\epsilon)$ do not depend on n so that the confidence set have size that is $O(1)$. To construct the confidence sets, Bubeck, Devroye and Lugosi (2017) computes a centrality value for every node, which can for instance be based on inverse of the size of the maximum subtree of a node (a concepted sometimes called Jordan centrality on trees, different from the notion of a Jordan center, which is the node with the minimum farthest distance to the other nodes); they then sort the nodes by centrality and take the top $K(\epsilon)$ nodes where the size $K(\epsilon)$ is determined by probabilistic bounds.

Khim and Loh (2017) further extends these results to the setting of uniform attachment over an infinite regular tree. Banerjee and Bhamidi (2020) improves the analysis of Jordan centrality on trees and derives tight upper and lower bounds on the confidence set size. Devroye and Reddad (2018); Lugosi et al. (2019) study the more general problem of seed-tree inference instead of root node inference. The aforementioned results apply only to tree shaped networks but very recently, Banerjee and Huang (2021) studies confidence sets constructed from the degrees of the nodes which applies to preferential attachment models in which a fixed m edges are added at every iteration. After the completion of this paper, Briend et al. (2022) propose confidence sets for the root node on a class of uniform-attachment-based general Markovian graphs by detecting anchors of double-cycle subgraphs within the network; they show the confidence set sizes to be $O(1)$ and give explicit bounds in terms of confidence level ϵ.

A line of work in the physics literature also explores the problem of full or partial recovery of a tree network history (Young et al.; 2019; Cantwell et al.; 2019; Sreedharan et al.; 2019). In computer science and engineering, researchers have studied the related problem of estimating the source of an infection spreading over a background network Shah and Zaman (2011); Fioriti et al. (2014); Shelke and Attar (2019), with approaches that range from using Jordan centers, eigenvector centrality, and belief propagation (see survey in Jiang et al. (2016)).

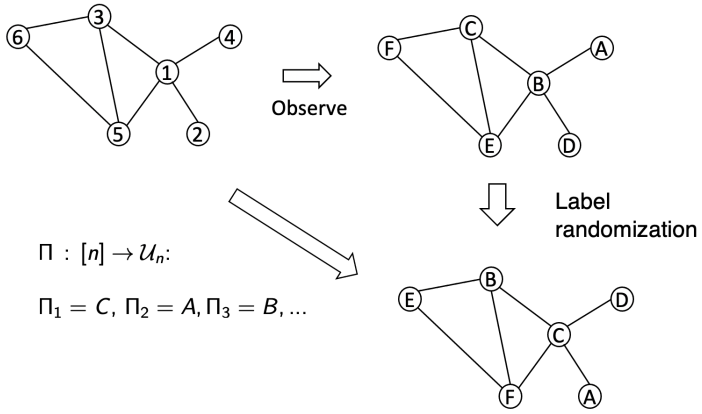


Figure 7: Label randomization induces a random latent arrival ordering Π.

3 Methodology

Our approach to root inference and related problems is to randomize the node labels, which induces a posterior distribution over the latent ordering.

3.1 Label randomization

Suppose $\boldsymbol{G}_n$ is a time labeled graph distributed according to a PAPER model and $\boldsymbol{G}_n^*$ is the alphabetically labeled observation where $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n$ for some label bijection $\rho \in \mathrm{Bi}([n], \mathcal{U}_n)$. We may independently generate a random bijection $\Lambda \in \mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$ and apply it to $\boldsymbol{G}_n^*$ to obtain a randomly labeled graph

$$\tilde{\boldsymbol{G}}_n := \Lambda \boldsymbol{G}_n^* = \underbrace{(\Lambda \circ \rho)}_{\Pi} \boldsymbol{G}_n.$$

By defining $\Pi = \Lambda \circ \rho$, we see that $\tilde{\boldsymbol{G}}_n = \Pi \boldsymbol{G}_n$ where Π is a random bijection drawn uniformly in $\mathrm{Bi}([n], \mathcal{U}_n)$ independently of $\boldsymbol{G}_n$ (see Figure 7). We define the randomly labeled latent forest $\tilde{\boldsymbol{F}}_n = \Pi \boldsymbol{F}_n$. We may view label randomization as an augmentation of the probability space. An outcome of a PAPER model is a time labeled graph $\boldsymbol{g}_n$ whereas an outcome after label randomization is a pair $(\tilde{\boldsymbol{g}}_n, \pi)$ where $\tilde{\boldsymbol{g}}_n$ is an alphabetically labeled graph and π is an ordering of the nodes. We now make two simple but important observations regarding label randomization.

Our first key observation is that, with respect to $\tilde{\boldsymbol{G}}_n$, the random labeling Π describes the arrival time of the nodes in the sense that if $\Pi_t = u$, then the node with alphabetical label u in $\tilde{\boldsymbol{G}}_n$ has the true arrival time t. Therefore, in the single root setting, we may infer the root node if we can infer Π_1; in the multiple roots setting, we may infer the set of root nodes if we can infer ΠS.

Our second key observation is that label randomization allows us to define the posterior distribution

$$\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) = \frac{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi)}{\sum_{\pi' \in \mathrm{Bi}([n], \mathcal{U}_n)} \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi')} \tag{5}$$

which follows because $\mathbb{P}(\Pi = \pi) = \frac{1}{n!}$. This posterior distribution is supported on the subset of bijection π such that $\pi^{-1}\tilde{\boldsymbol{g}}_n$ has non-zero probability under the PAPER model. In the case of the single root PAPER or seq-PAPER model, the support of (5) has a simple characterization: for

$\boldsymbol{G}_n$	time labeled graph (unobserved)	$\boldsymbol{F}_n$	latent time labeled forest
$\boldsymbol{G}_n^*$	observed alpha. labeled graph	$\boldsymbol{F}_n^*$	latent alpha. labeled forest
$\tilde{\boldsymbol{G}}_n$	randomly alpha. labeled graph	$\tilde{\boldsymbol{F}}_n$	latent randomly alpha. labeled forest
ρ	fixed unobserved ordering; $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n$	Π	latent random ordering; $\tilde{\boldsymbol{G}}_n = \Pi \boldsymbol{G}_n$
S	time labeled root nodes of $\boldsymbol{G}_n$	$\tilde{S}$	latent alpha. labeled root nodes; $\tilde{S} = \Pi S$

Table 1: Quick reference of important notation and definitions.

every time point $t \in [n]$, define $\pi_{1:t} \cap \tilde{\boldsymbol{g}}_n$ as the subgraph of $\tilde{\boldsymbol{g}}_n$ restricted to nodes in $\pi_{1:t}$. Then, $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) > 0$ if and only if $\pi_{1:t} \cap \tilde{\boldsymbol{g}}_n$ is connected for all $t \in [n]$.

From a Bayesian perspective, label randomization adds a uniform prior distribution on the arrival ordering of the nodes in the observed alphabetically labeled graph $\boldsymbol{G}_n^*$; this is sometimes used in Bayesian parameter inference on network models (Sheridan et al.; 2012; Bloem-Reddy et al.; 2018). This prior however is not subjective. Indeed, we will see in Theorem 7 that Bayesian inference statements in our setting directly have frequentist validity as well and, from Section S2.1, that the posterior root probability of a node is equal to the likelihood of that node being the root node up to normalization.

We describe how to compute (5) tractably in Section 4. For computation, we will also be interested in the posterior probability over both the ordering Π as well as the latent forest $\tilde{\boldsymbol{F}}_n$:

$$\mathbb{P}(\Pi = \pi, \tilde{\boldsymbol{F}} = \tilde{\boldsymbol{f}}_n \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n). \tag{6}$$

In the single root setting, $\tilde{\boldsymbol{f}}_n$ is actually a tree, which we may write as $\tilde{\boldsymbol{t}}_n$. It is then clear that (6) is non-zero only if $\tilde{\boldsymbol{t}}_n$ is a *spanning tree* of $\tilde{\boldsymbol{g}}_n$, i.e., $\tilde{\boldsymbol{t}}_n$ is a connected subtree of $\tilde{\boldsymbol{g}}_n$ that contains all the vertices.

3.2 Confidence set for the single root

To make the idea clear, we first consider the single root model. Since the root node is the node labeled Π_1 after label randomization, a natural approach is to first construct a level $1-\epsilon$ Bayesian *credible set* for the node Π_1 by using its posterior distribution, which we call the posterior root distribution.

More concretely, let $\tilde{\boldsymbol{g}}_n$ be an alphabetically labeled graph. For each node $u \in \mathcal{U}_n$ of $\tilde{\boldsymbol{g}}_n$, we define the posterior root probability as $\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$. We sort the nodes $u_1, \ldots, u_n$ so that

$$\mathbb{P}(\Pi_1 = u_1 \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq \mathbb{P}(\Pi_1 = u_2 \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \ldots \geq \mathbb{P}(\Pi_1 = u_n \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n),$$

and define

$$L_\epsilon(\tilde{\boldsymbol{g}}_n) = \min\left\{ k \in [n] \,:\, \sum_{i=1}^{k} \mathbb{P}(\Pi_1 = u_i \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq 1 - \epsilon \right\} \tag{7}$$

We then define the ϵ-credible set as

$$B_\epsilon(\tilde{\boldsymbol{g}}_n) = \left\{ u_1, u_2, \ldots, u_{L_\epsilon(\tilde{\boldsymbol{g}}_n)} \right\}, \qquad \text{(breaking ties at random)}. \tag{8}$$

By definition, $B_\epsilon(\tilde{\boldsymbol{g}})$ is the smallest set of nodes with Bayesian coverage at level $1-\epsilon$ in that $\mathbb{P}(\Pi_1 \in B_\epsilon(\tilde{\boldsymbol{g}}_n) \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq 1 - \epsilon$. In general, credible sets do not have valid frequentist confidence coverage. However, our next theorem shows that in our setting, the credible set B_ϵ is in fact an honest confidence set in that $\mathbb{P}\{\text{root node} \in B_\epsilon(\boldsymbol{G}_n^*)\} \geq 1 - \epsilon$.

Theorem 7. *Let $\boldsymbol{G}_n \sim$ PAPER(α, β, θ) or seq-PAPER$(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta})$ and let $\boldsymbol{G}_n^*$ be the alphabetically labeled observation. Let $\rho \in Bi([n], \mathcal{U}_n)$ be any label bijection such that $\rho \mathbf{G}_n = \mathbf{G}_n^*$. We have that, for any $\epsilon \in (0, 1)$,*

$$\mathbb{P}\big\{\rho_1 \in B_\epsilon(\mathbf{G}_n^*)\big\} \geq 1 - \epsilon.$$

The proof is very similar to that of Crane and Xu (2021, Theorem 1). Since the proof is short, we provide it here for readers' convenience.

Proof. We first claim that $B_\epsilon(\cdot)$ is labeling-equivariant (cf. Remark 4) in the sense that for any $\tau \in \mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$ and any alphabetically labeled graph $\tilde{\boldsymbol{g}}_n$, we have that $\tau B_\epsilon(\tilde{\boldsymbol{g}}_n) \overset{d}{=} B_\epsilon(\tau \tilde{\boldsymbol{g}}_n)$ (note that $B_\epsilon(\cdot)$ uses randomization to break ties). Indeed, since $(\Pi, \tilde{\boldsymbol{G}}_n) \overset{d}{=} (\tau^{-1} \circ \Pi, \tau^{-1} \tilde{\boldsymbol{G}}_n)$, we have that, for any $u \in \mathcal{U}_n$,

$$\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) = \mathbb{P}(\Pi_1 = \tau(u) \,|\, \tilde{\mathbf{G}}_n = \tau \tilde{\mathbf{g}}_n).$$

Therefore, for any $u, v \in \mathcal{U}_n$, we have that $\mathbf{P}(\Pi_1 = u \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) \geq \mathbf{P}(\Pi_1 = v \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n)$ if and only if $\mathbf{P}(\Pi_1 = \tau(u) \,|\, \tilde{\mathbf{G}}_n = \tau \tilde{\mathbf{g}}_n) \geq \mathbf{P}(\Pi_1 = \tau(v) \,|\, \tilde{\mathbf{G}}_n = \tau \tilde{\mathbf{g}}_n)$. Since $B_\epsilon(\mathbf{G}_n^*)$ is constructed by taking the top elements of $\mathcal{U}_n$ that maximize the cumulative posterior root probability, the claim follows.

Now, let $\rho \in \mathrm{Bi}([n], \mathcal{U}_n)$ be such that $\rho \mathbf{G}_n = \mathbf{G}_n^*$ and let Λ be a random bijection drawn uniformly in $\mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$ and let $\Pi = \Lambda \circ \rho$. Then,

$$\begin{aligned}
\mathbb{P}(\rho_1 \in B_\epsilon(\mathbf{G}_n^*)) &= \mathbb{P}(\rho_1 \in B_\epsilon(\rho \mathbf{G}_n)) \\
&= \mathbb{P}\big\{(\Lambda \circ \rho)_1 \in B_\epsilon((\Lambda \circ \rho)\mathbf{G}_n) \,|\, \Lambda = \mathrm{Id}\big\} \\
&= \mathbb{P}\big\{(\Lambda \circ \rho)_1 \in B_\epsilon((\Lambda \circ \rho)\mathbf{G}_n)\big\} \\
&= \mathbb{P}(\Pi_1 \in B_\epsilon(\tilde{\mathbf{G}}_n)) \geq 1 - \epsilon,
\end{aligned}$$

where the penultimate equality follows from the labeling-equivariance of B_ϵ and where the last inequality follows because $\mathbf{P}(\Pi_1 \in B_\epsilon(\tilde{\mathbf{G}}_n) \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) \geq 1 - \epsilon$ for any labeled tree $\tilde{\mathbf{g}}_n$ (with labels in $\mathcal{U}_n$) by the definition of B_ϵ. □

Remark 6. We show in Theorem S5 of the appendix that the posterior root probability $\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ is equal to the likelihood of node u being the root node on observing the unlabeled shape of $\tilde{\boldsymbol{g}}_n$. Therefore, the set $B_\epsilon(\tilde{\boldsymbol{g}}_n)$ is in fact the maximum likelihood confidence set. Because the likelihood in this setting is complicated to even write down, we leave all the details to Section S2.1 of the appendix.

Remark 7. One may see from the proof that Theorem 7 applies more broadly then just PAPER models. It in fact applies to any random graph $\boldsymbol{G}_n$ whose nodes are labeled by $\{1, 2, \ldots, n\}$. For the PAPER model, the integer labels encode arrival time and thus contain information about the graph. In a model where the integer labels are uninformative of the graph connectivity structure, Theorem 7 is still valid although the posterior probability $\mathbb{P}(\Pi_1 = \cdot \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ would be uniform. A reviewer of this paper also pointed out that Theorem 7 is related to the classical literature on invariant/equivariant estimation where credible sets constructed from uniform (Haar) priors may also be valid confidence sets; see e.g. Schervish (1995, Theorem 6.78).

3.3 Confidence set for multiple roots

First consider the fixed K setting where $\boldsymbol{G}_n \sim \mathrm{PAPER}(\alpha, \beta, \theta, K)$; let Π be a uniformly random ordering in $\mathrm{Bi}([n], \mathcal{U}_n)$ and let $\tilde{\boldsymbol{G}}_n = \Pi \boldsymbol{G}_n$. The latent set of root nodes of $\tilde{\boldsymbol{G}}_n$ in this case is

$\tilde{S} := \Pi S = \{\Pi_1, \ldots, \Pi_K\}$. We then define the posterior root probability for any node $u \in \mathcal{U}_n$ as

$$\mathbb{P}(u \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n),$$

that is, the probability that node u is an element of the latent root set $\tilde{S}$.

To form the credible set $B_\epsilon(\tilde{\boldsymbol{g}}_n) \subseteq \mathcal{U}_n$, we sort the nodes by the posterior root probabilities

$$\mathbb{P}(u_1 \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq \mathbb{P}(u_2 \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq \ldots \geq \mathbb{P}(u_n \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n). \tag{9}$$

We may then take $B_\epsilon(\tilde{\boldsymbol{g}}_n)$ to be the smallest set of nodes such that $P\big(\tilde{S} \subsetneq B_\epsilon(\tilde{\boldsymbol{g}}_n) \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n\big) \leq \epsilon$. More precisely, define the integer

$$L_\epsilon(\tilde{\boldsymbol{g}}_n) = \min\left\{k \in [n] \,:\, \sum_{i=k+1}^{n} \mathbb{P}(u_i \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \leq \epsilon\right\} \tag{10}$$

and then define the credible set as

$$B_\epsilon(\tilde{\boldsymbol{g}}_n) = \big\{u_1, u_2, \ldots, u_{L_\epsilon(\tilde{\boldsymbol{g}}_n)}\big\} \qquad \text{(breaking ties at random)}. \tag{11}$$

In the PAPER$(\alpha, \beta, \alpha_0, \theta)$ model where the number of roots K is random, the set of root nodes is $\tilde{S} = \Pi S$ which comprises, according to the ordering Π, of the node that is first to arrive in each of the component trees of $\tilde{\boldsymbol{F}}_n$. We may then sort the nodes as in (9), compute $L_\epsilon(\tilde{\boldsymbol{g}}_n)$ as in (10) and $B_\epsilon(\tilde{\boldsymbol{g}}_n)$ as in (11).

Similar to Theorem 7, we may show that $B_\epsilon(\cdot)$ in fact also has frequentist coverage at the same level $1 - \epsilon$.

Theorem 8. *Let $\boldsymbol{G}_n \sim$ PAPER$(\alpha, \beta, K, \theta)$ or PAPER$(\alpha, \beta, \alpha_0, \theta)$ and let $\boldsymbol{G}_n^*$ be the alphabetically labeled observation. Let $\rho \in$ Bi$([n], \mathcal{U}_n)$ be any label bijection such that $\rho \mathbf{G}_n = \mathbf{G}_n^*$ and let $S \subset [n]$ be the time labels of the root nodes (see Definitions 4 and 5). We have that, for any $\epsilon \in (0, 1)$,*

$$\mathbb{P}\big\{\rho S \subseteq B_\epsilon(\mathbf{G}_n^*)\big\} \geq 1 - \epsilon.$$

Proof. The proof is very similar to that of Theorem 7. First, since the random set $\tilde{S}$ is a function of the random ordering Π in the fixed K setting and a function of both the random ordering Π and the random forest $\tilde{\boldsymbol{F}}_n$, we write $\tilde{S}(\Pi)$ or $\tilde{S}(\Pi, \tilde{\boldsymbol{F}}_n)$ to be precise.

We then observe that $\tilde{S}(\Pi)$ in the fixed K setting or $\tilde{S}(\Pi, \tilde{\boldsymbol{F}}_n)$ in the random K setting, are labeling equivariant in that for any $\tau \in \text{Bi}(\mathcal{U}_n, \mathcal{U}_n)$, we have that $\tilde{S}(\tau^{-1}\Pi) = \tau^{-1}\tilde{S}(\Pi)$ or, in the random K setting, $\tilde{S}(\tau^{-1}\Pi, \tau^{-1}\tilde{\boldsymbol{F}}_n) = \tau^{-1}\tilde{S}(\Pi, \tilde{\boldsymbol{F}}_n)$. Therefore, since $(\Pi, \tilde{\boldsymbol{G}}_n) \overset{d}{=} (\tau^{-1}\Pi, \tau^{-1}\tilde{\boldsymbol{G}}_n)$ for any $\tau \in \text{Bi}(\mathcal{U}_n, \mathcal{U}_n)$, we have $\tilde{S}(\Pi, \tilde{\boldsymbol{F}}_n) \overset{d}{=} \tau^{-1}\tilde{S}(\Pi, \tilde{\boldsymbol{F}}_n)$ and thus, for any $u \in \mathcal{U}_n$,

$$\mathbb{P}(u \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) = \mathbb{P}(\tau(u) \in \tilde{S} \,|\, \tilde{\boldsymbol{G}}_n = \tau\tilde{\boldsymbol{g}}_n).$$

The rest the proof proceeds in an identical manner to that of Theorem 7. □

When there are multiple roots, an alternative way of inferring the root set is to construct the confidence set $B_\epsilon(\cdot)$ as a set of subsets of the nodes and then require that $\tilde{S} \in B_\epsilon$ with probability at least $1 - \epsilon$. We can take the same approach to construct such confidence set over sets but it becomes much more computationally intensive to compute them in practice.

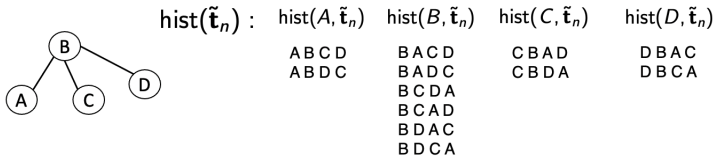


Figure 8: All histories of a tree with 4 nodes.

3.4 Combinatorial interpretation

Before we describe the Gibbs sampling algorithm for computing the posterior root probabilities $\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$, we provide an intuitive combinatorial interpretation of the posterior root probability in the single root PAPER model (Definition 2). The definitions and calculations here are also important for deriving the algorithm in Section 4.

The noiseless case: We first consider the simpler setting in which we can observe the tree $\tilde{\boldsymbol{T}}_n$ (with a single root) distributed according to the APA model. In this case, we have

$$\mathbb{P}(\Pi_1 = \cdot \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \sum_{\pi \,:\, \pi_1 = u} \mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n).$$

Recall that $\tilde{\boldsymbol{T}}_n = \Pi \boldsymbol{T}_n$ where $\boldsymbol{T}_n$ is a random time labeled tree with APA(α, β) distribution and Π is an independent uniformly random ordering in $\mathrm{Bi}([n], \mathcal{U}_n)$. The distribution $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)$ is supported on a subset of the the bijections $\mathrm{Bi}([n], \mathcal{U}_n)$ because $\pi^{-1}\tilde{\boldsymbol{T}}_n$ must be a valid time labeled tree (also called *recursive tree* in discrete mathematics). To be precise, we define the histories of $\tilde{\boldsymbol{t}}_n$ as

$$\begin{aligned} \mathrm{hist}(\tilde{\boldsymbol{t}}_n) &:= \big\{\pi \in \mathrm{Bi}([n], \mathcal{U}_n) \,:\, \mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n) > 0\big\}, \text{ and} \\ h(\tilde{\boldsymbol{t}}_n) &:= |\mathrm{hist}(\tilde{\boldsymbol{t}}_n)| \end{aligned}$$

as the number of distinct histories. Since the APA tree distribution assigns a non-zero probability to any valid time labeled trees, we see that $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$ contains the elements π of $\mathrm{Bi}([n], \mathcal{U}_n)$ such that for all $t \in [n]$, the subtree restricted only to nodes in $\pi_{1:t}$, i.e. $\tilde{\boldsymbol{t}}_n \cap \pi_{1:t}$, is connected. Thus, $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$ is the set of bijections π which represent a valid arrival ordering for the nodes of the given tree $\tilde{\boldsymbol{t}}_n$. Similarly, we define, for any node $u \in \mathcal{U}_n$,

$$\begin{aligned} \mathrm{hist}(u, \tilde{\boldsymbol{t}}_n) &:= \big\{\pi \in \mathrm{hist}(\tilde{\boldsymbol{t}}_n) \,:\, \pi_1 = u\big\} \\ h(u, \tilde{\boldsymbol{t}}_n) &:= |\mathrm{hist}(u, \tilde{\boldsymbol{t}}_n)|, \end{aligned}$$

as histories of $\tilde{\boldsymbol{t}}_n$ that start at node u. We illustrate an example of the set of histories for a simple tree in Figure 8.

By definition, $\mathbb{P}(\Pi = \cdot \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)$ is supported on $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$. For most values of α and β, the posterior distribution is in fact uniform over $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$:

Proposition 9. *(Crane and Xu; 2021, Theorem 4 and Proposition 3) Let α, β be two real numbers such that either (1) $\beta \geq 0$ and $\alpha \geq -\beta$ or (2) $\beta < 0$ and $\alpha = -D\beta$ for some integer $D \geq 2$. Suppose $\boldsymbol{T}_n \sim APA(\alpha, \beta)$. Let Π be a uniformly random ordering taking value in $Bi([n], \mathcal{U}_n)$ and let $\tilde{\boldsymbol{T}}_n = \Pi \boldsymbol{T}_n$. Then,*

$$\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \frac{1}{h(\tilde{\boldsymbol{t}}_n)} \mathbb{1}\{\pi \in \mathit{hist}(\tilde{\boldsymbol{t}}_n)\}. \tag{12}$$

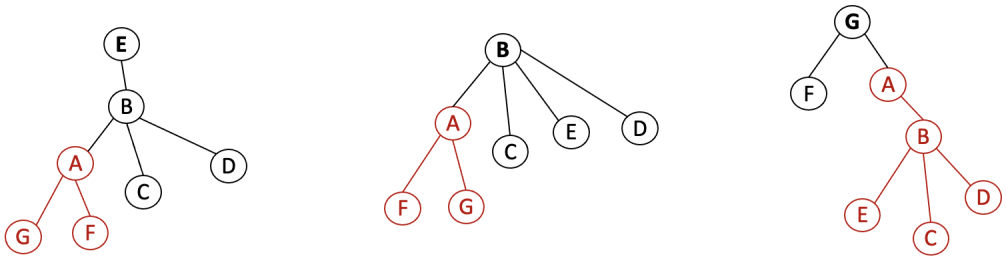


Figure 9: Same tree $\tilde{\boldsymbol{t}}_n$ in three rooted orientations. Left: $\tilde{\boldsymbol{t}}_n^{(E)}$ rooted at E; the subtree of A (denoted $\tilde{\boldsymbol{t}}_A^{(E)}$) contains nodes A, F, G; node A is the parent of F, G. Center: $\tilde{\boldsymbol{t}}_n^{(B)}$ rooted at B; the subtree of A (denoted $\tilde{\boldsymbol{t}}_A^{(B)}$) contains nodes A, F, G; node A is the parent of F, G. Right: $\tilde{\boldsymbol{t}}_n^{(G)}$ rooted at G; the subtree of A (denoted $\tilde{\boldsymbol{t}}_A^{(G)}$) contains nodes A, B, E, C, D; node A is the parent of B.

The full proof of Proposition 9 is in Crane and Xu (2021) but we give a short justification here: the posterior is uniform because $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \frac{\mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n \,|\, \Pi = \pi)\frac{1}{n!}}{\mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)} = \frac{\mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n)\frac{1}{n!}}{\mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)}$. Moreover, the probability $\mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n)$ is actually the same for any $\pi \in \mathrm{hist}(\tilde{\boldsymbol{t}}_n)$ by Proposition S1.

By Proposition 9, we have that

$$\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \frac{h(u, \tilde{\boldsymbol{t}}_n)}{h(\tilde{\boldsymbol{t}}_n)}.$$

Therefore, we need only count the histories $h(u, \tilde{\boldsymbol{t}}_n)$ for every node $u \in \mathcal{U}_n$. We give a well-known characterization of $h(u, \tilde{\boldsymbol{t}}_n)$ that leads to a linear time algorithm for counting the size of the histories: define, for any node $u, v \in \mathcal{U}_n$, the tree $\tilde{\boldsymbol{t}}_v^{(u)}$ as the subtree of node v where we view the whole tree as being rooted (hanging from) node u; $\tilde{\boldsymbol{t}}_u^{(u)}$ is thus the entire tree rooted at u. See Figure 9 for an example. We then have that, by Knuth (1997) or Shah and Zaman (2011),

$$h(u, \tilde{\boldsymbol{t}}_n) = n! \prod_{v \in \mathcal{U}_n} \frac{1}{|\tilde{\boldsymbol{t}}_v^{(u)}|}. \tag{13}$$

Therefore, we can compute $h(u, \tilde{\boldsymbol{t}}_n)$ by viewing $\tilde{\boldsymbol{t}}_n$ as being rooted at u and taking the product of the inverse of the sizes of all the subtrees. By using the fact that $h(u, \tilde{\boldsymbol{t}}_n)$ can be directly computed from $h(u', \tilde{\boldsymbol{t}}_n)$ for any neighbor u' of u, Shah and Zaman (2011) derive an $O(n)$ algorithm for computing the size of the histories over all roots $\{h(u, \tilde{\boldsymbol{t}}_n)\}_{u \in \mathcal{U}_n}$, which we give in Section S2 of the appendix for readers' convenience.

The general case: Now suppose we have the label randomized graph $\tilde{\boldsymbol{G}}_n$ from the PAPER model. We then have that

$$\begin{aligned}
\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) &= \sum_{\tilde{\boldsymbol{t}}_n \subseteq \tilde{\boldsymbol{g}}_n} \sum_{\pi \in \mathrm{hist}(u, \tilde{\boldsymbol{t}}_n)} \mathbb{P}(\Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \\
&\propto \sum_{\tilde{\boldsymbol{t}}_n \subseteq \tilde{\boldsymbol{g}}_n} \sum_{\pi \in \mathrm{hist}(u, \tilde{\boldsymbol{t}}_n)} \mathbb{P}(\Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) \underbrace{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n, \Pi = \pi)}_{\binom{n(n-1)/2-(n-1)}{m-(n-1)}^{-1}}. \\
&\propto \sum_{\tilde{\boldsymbol{t}}_n \subseteq \tilde{\boldsymbol{g}}_n} \sum_{\pi \in \mathrm{hist}(u, \tilde{\boldsymbol{t}}_n)} \mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n \,|\, \Pi = \pi) = \sum_{\tilde{\boldsymbol{t}} \subseteq \tilde{\boldsymbol{g}}_n} \sum_{\pi \in \mathrm{hist}(u, \tilde{\boldsymbol{t}})} \mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n),
\end{aligned} \tag{14}$$

where, in the outer summation, we require $\tilde{\boldsymbol{t}}_n$ to be a subtree of $\tilde{\boldsymbol{g}}_n$ with n nodes, that is, we require $\tilde{\boldsymbol{t}}_n$ to be a spanning tree of $\tilde{\boldsymbol{g}}_n$ (see (16)). If $\boldsymbol{T}_n$ has the uniform attachment distribution

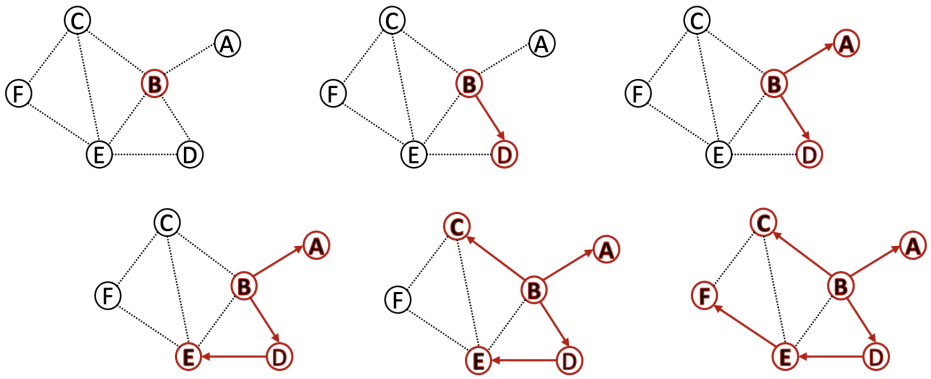


Figure 10: One possible growth realization starting from node B.

$(\alpha = 1, \beta = 0)$, then we have that $\mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n) = \frac{1}{(n-1)!}$ by Proposition S1 and hence,

$$\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \propto \sum_{\tilde{\boldsymbol{t}}_n \subseteq \tilde{\boldsymbol{g}}_n} h(u, \tilde{\boldsymbol{t}}_n).$$

Thus, the posterior root probability of u is simply proportional to the number of all possible realizations of growth process that start from node u and end up with graph $\tilde{\boldsymbol{g}}_n$; see Figure 10. When $\boldsymbol{T}_n$ has the LPA distribution $(\alpha = 0, \beta = 1)$, then $\mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n)$ depends on the degree sequence of the tree $\tilde{\boldsymbol{t}}_n$ so that the posterior root probability is proportional to a weighted count of all possible growth realizations.

4 Algorithm

The inference approach that we described in Sections 3.2 and 3.3 requires computing posterior probabilities such as the posterior root probability $P(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ for a fixed alphabetically labeled graph $\tilde{\boldsymbol{g}}_n$. In this section, we derive a Gibbs sampling algorithm to generate an ordering $\pi \in \mathrm{Bi}([n], \mathcal{U}_n)$ and a forest $\tilde{\boldsymbol{f}}_n$ according to the posterior probability

$$\mathbb{P}(\Pi = \pi, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n). \tag{15}$$

As discussed towards the end of Section 3.1, in the single root setting, the posterior probability (15) over $\Pi, \tilde{\boldsymbol{F}}_n$ is non-zero only if $\tilde{\boldsymbol{f}}_n$ is a spanning tree of the graph $\tilde{\boldsymbol{g}}_n$. We formally define the set of spanning trees of a connected graph $\tilde{\boldsymbol{g}}_n$ as

$$\mathcal{T}(\tilde{\boldsymbol{g}}_n) := \big\{\tilde{\boldsymbol{f}}_n \,:\, \tilde{\boldsymbol{f}}_n \text{ is connected subtree of } \tilde{\boldsymbol{g}}_n \text{ and } V(\tilde{\boldsymbol{f}}_n) = V(\tilde{\boldsymbol{g}}_n)\big\}. \tag{16}$$

We note that $\mathcal{T}(\tilde{\boldsymbol{g}}_n)$ is non-empty if and only if $\tilde{\boldsymbol{g}}_n$ is connected. For the multiple roots setting, we define the spanning forest of $\tilde{\boldsymbol{g}}_n$ with K components as

$$\mathcal{F}_K(\tilde{\boldsymbol{g}}_n) := \big\{\tilde{\boldsymbol{f}}_n \,:\, \tilde{\boldsymbol{f}}_n \text{ is sub-forest of } \tilde{\boldsymbol{g}}_n \text{ with } K \text{ disjoint component trees and } V(\tilde{\boldsymbol{f}}_n) = V(\tilde{\boldsymbol{g}}_n)\big\}$$

so that $\mathcal{F}_1(\tilde{\boldsymbol{g}}_n) = \mathcal{T}(\tilde{\boldsymbol{g}}_n)$. Then, for the fixed K roots model, the posterior probability (15) is non-zero only if $\tilde{\boldsymbol{f}}_n \in \mathcal{F}_K(\tilde{\boldsymbol{g}}_n)$ and for the random K roots model, probability (15) is non-zero only if $\tilde{\boldsymbol{f}}_n \in \mathcal{F}(\tilde{\boldsymbol{g}}_n) := \cup_{K=1}^{n} \mathcal{F}_K(\tilde{\boldsymbol{g}}_n)$.

The value of the posterior probability (15) depends on the parameters of the model, e.g. α, β, θ in the single root setting. We provide an estimation procedure for these parameters in Section S3.1 but for now, to keep the presentation simple, we assume that all parameters are known.

Our Gibbs sampler alternates between two stages:

(A) We fix the forest $\tilde{\boldsymbol{f}}_n$ and generate an ordering π with probability $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)$.

(B) We fix the ordering π and generate a new forest $\tilde{\boldsymbol{f}}_n$ by iteratively sampling a new parent for each of the nodes.

We give the details for stage A in the next section and for stage B in Section 4.2.

Remark 8. In Section S3.3, we give an alternative collapsed Gibbs sampling algorithm in which we collapse stage (A) so that we only sample the roots instead of the whole history π. The collapsed Gibbs sampler requires fewer iterations to converge but each iteration is more computationally intensive. Practically, the sampling algorithm that we present in Section 4.1 and 4.2 appears to be faster except for the random K roots model on some data sets.

4.1 Sampling the ordering

In this section, we provide an algorithm for the first stage of the Gibbs sampler where we sample an ordering. We fix a spanning forest $\tilde{\boldsymbol{f}}_n$ of the observed graph $\tilde{\boldsymbol{g}}_n$, let K be the number of component trees of $\tilde{\boldsymbol{f}}_n$, and let $m = |E(\tilde{\boldsymbol{g}}_n)|$ be the number of edges of $\boldsymbol{g}_n$. We have that

$$\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n) \propto \mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi). \tag{17}$$

Under the non-sequential noise PAPER models, since the non-forest edges of $\tilde{\boldsymbol{G}}_n$ are independent Erdős–Rényi random edges, we have $\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi) = \binom{\binom{n}{2}-(n-K)}{m-(n-K)}^{-1}$ and may thus ignore the non-forest edges and consider only on the posterior probability $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)$ when sampling π. In the sequential noise seq-PAPER model, the $\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi)$ term must be taken into account but can be computed efficiently. We give the detailed algorithms for each of the settings.

Single root setting: In the single root setting, $\tilde{\boldsymbol{f}}_n$ is connected and hence a tree; we thus change to the notation $\tilde{\boldsymbol{t}}_n := \tilde{\boldsymbol{f}}_n$ to be consistent with the notation used in Definition 1.

Hence, by our discussion in Section 3.4, sampling π according to $\mathbb{P}(\Pi = \cdot \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)$ is equivalent to sampling π uniformly from $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$. Crane and Xu (2021) and also Cantwell et al. (2021) derive a procedure to sample uniformly from $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$ and we provide a concise description of the procedure here for the readers' convenience.

To generate π uniformly from $\mathrm{hist}(\tilde{\boldsymbol{t}}_n)$, we generate the first node π_1 by taking the set of all nodes and drawing a node u with probability

$$\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \frac{h(u, \tilde{\boldsymbol{t}}_n)}{h(\tilde{\boldsymbol{t}}_n)}. \tag{18}$$

The entire collection $\{h(u, \tilde{\boldsymbol{t}}_n)\}_{u \in \mathcal{U}_n}$ can be computed in $O(n)$ time (c.f. Section 3.4 and S2) and thus we require at most $O(n)$ time to generate the first node π_1.

To generate the subsequent ordering $\pi_{2:n}$, we view the tree $\tilde{\boldsymbol{t}}_n$ as being rooted at π_1 and use the notation $\tilde{\boldsymbol{t}}_n^{(\pi_1)}$ make the root explicit. For each node $v \in \mathcal{U}_n$, we define $\tilde{\boldsymbol{t}}_v^{(\pi_1)}$ as the subtree of the node v, viewing the whole tree as being rooted at node π_1. We give an example of these definitions in Figure 9.

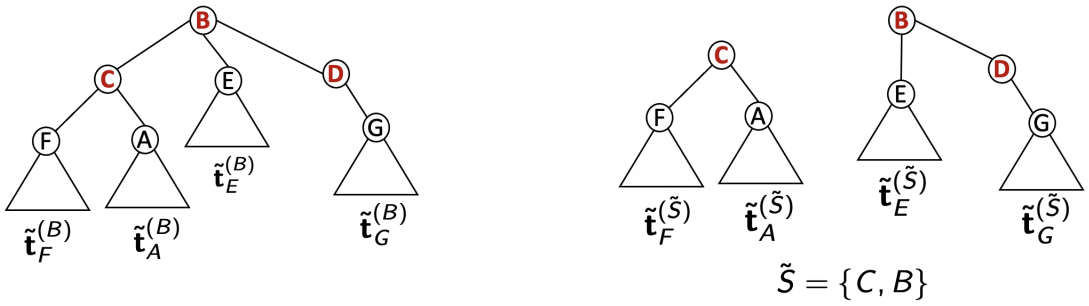


Figure 11: Example of sampling an ordering. In both cases, suppose $\pi_{1:3} = \{B, C, D\}$, then draw π_4 from the neighbors $\{F, A, E, G\}$ with probability proportional to the size of their subtrees.

Then, by Crane and Xu (Proposition 9 2021), for every $t \in [n-1]$,

$$\mathbb{P}(\Pi_{t+1} = v \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n, \Pi_{1:t} = \pi_{1:t}) = \begin{cases} \frac{|\tilde{\boldsymbol{t}}_v^{(\pi_1)}|}{n-t+1} & \text{if } v \text{ is a neighbor of } \pi_{1:t} \text{ in } \tilde{\boldsymbol{t}}_n \\ 0 & \text{else} \end{cases} \tag{19}$$

One may verify this by showing that the probability of generating a particular ordering is $\frac{1}{n!} \prod_{v \in \mathcal{U}_n} |\tilde{\boldsymbol{t}}_n^{(u)}| = \frac{1}{h(u, \tilde{\boldsymbol{t}}_n)}$ by (13).

Thus, we may generate π_2 by considering all neighbors of π_1 in $\tilde{\boldsymbol{t}}_n$ and drawing a node v with probability proportional to the size of its subtree $|\tilde{\boldsymbol{t}}_v^{(u_1)}|$ and similar for π_3, π_4, etc. The entire sampling process can be efficiently done by generating a permutation uniformly at random and modifying it in place so that it obeys the hist($\tilde{\boldsymbol{f}}_n$) constraint. We summarize this in Algorithm 1 with $K = 1$ and also give a visual illustration in Figure 11. The runtime of the sampling algorithm is upper bounded by $O(n\text{diam}(\tilde{\boldsymbol{t}}_n))$ (Crane and Xu; 2021, Proposition 10). Trees generated by the APA(α, β) model have diameter $O_p(\log n)$ (see e.g. Drmota (2009, Theorem 6.32) and Bhamidi (2007, Theorem 18)) and the overall runtime is therefore $O(n \log n)$. The computational complexity is the same under the fixed K setting and the random K setting.

Fixed K roots setting: For the PAPER$(\alpha, \beta, K, \theta)$ model, we may generate from $\mathbb{P}(\Pi = \cdot \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)$ in a similar way. In this case, $\tilde{\boldsymbol{f}}_n$ is a forest that contains K disjoint component trees, which we denote by $\tilde{\boldsymbol{t}}^1, \ldots, \tilde{\boldsymbol{t}}^K$. We first generate a root for each component tree. For each $k \in [K]$, we draw $u^k \in V(\tilde{\boldsymbol{t}}^k)$ with probability

$$\frac{h(u^k, \tilde{\boldsymbol{t}}^k)(\beta D_{\tilde{\boldsymbol{t}}^k}(u^k) + \beta + \alpha)(\beta D_{\tilde{\boldsymbol{t}}^k}(u^k) + \alpha)}{\sum_{v \in V(\tilde{\boldsymbol{t}}^k)} h(v, \tilde{\boldsymbol{t}}^k)(\beta D_{\tilde{\boldsymbol{t}}^k}(v) + \beta + \alpha)(\beta D_{\tilde{\boldsymbol{t}}^k}(v) + \alpha)}. \tag{20}$$

We note that (20) is different from the corresponding probability in the single tree setting (18) because we give each root node an imaginary self-loop edge. We leave the detailed derivation of (20) to Section S3.2 of the appendix.

We let $\tilde{s} = \{u^1, \ldots, u^k\}$ denote the set of roots that we have generated. By the definition of the PAPER$(\alpha, \beta, K, \theta)$ model (Definition 4), the root nodes $\tilde{s}$ occupy the first K positions of the ordering π and we thus let $\pi_{1:K}$ be the elements of $\tilde{s}$ placed in a random ordering.

Next, we view each component tree $\tilde{\boldsymbol{t}}^k$ as being rooted at u_k and, for every node $v \in V(\tilde{\boldsymbol{f}}_n)$, we denote the subtree of node v by $\tilde{\boldsymbol{t}}_v^{(\tilde{s})}$. We then generate $\pi_{(K+1):n}$ according to probability (19) where we use the size of the subtree $|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|$. This is equivalent to generating a full history (excluding the root node) for every tree and then interleaving them at random. We again summarize the whole

procedure in Algorithm 1.

Random K roots setting: Now consider the random K roots setting with the PAPER$(\alpha, \beta, \alpha_0, \theta)$ model and suppose $\tilde{\boldsymbol{f}}_n$ comprises of K disjoint trees $\tilde{\boldsymbol{t}}^1, \ldots, \tilde{\boldsymbol{t}}^K$. We again generate the set of roots $\tilde{s} = \{u^1, \ldots, u^K\}$ by drawing u^k from $\tilde{\boldsymbol{t}}^k$ with probability (20). In contrast with the fixed K roots setting, the root nodes $u^1, \ldots, u^K$ need not occupy the first K positions of the ordering π.

To generate the ordering π, we first choose $u^k \in \tilde{s}$ with probability $|\tilde{\boldsymbol{t}}^k|$ and set $\pi_1 = u^k$. We then draw $\pi_{2:n}$ iteratively using the conditional distribution

$$\mathbb{P}(\Pi_{t+1} = v \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}, \Pi_{1:t} = \pi_{1:t}) = \begin{cases} \frac{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|}{n-t+1} & \text{if } v \text{ is a neighbor of } \pi_{1:t} \text{ in } \tilde{\boldsymbol{f}}_n \text{ or if } v \in \tilde{s} \\ 0 & \text{else} \end{cases} \tag{21}$$

We note that for a root node $u^k \in \tilde{s}$, the subtree $\tilde{\boldsymbol{t}}_{u^k}^{(\tilde{s})}$ is precisely the whole tree $\tilde{\boldsymbol{t}}^k$. We summarize this procedure in Algorithm 1.

Sequential noise setting: Under the seq-PAPER model described in Section 2.3, we no longer have a direct sampling algorithm to draw from $\mathbb{P}(\Pi = \cdot \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)$ because we have to take into account the $\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n, \Pi = \pi)$ term in (17). For seq-PAPER models, we propose instead a Metropolis–Hastings algorithm to update π by sampling new transpositions.

Let π be the current sample of arrival ordering. To generate a new proposal π^*, we randomly choose a pair $j, k \in \{2, \ldots, n\}$ and construct π^* by swapping the j-th and the k-th entries of π, that is, $\pi_j^* = \pi_k$ and $\pi_k^* = \pi_j$ and all other entries are equal. If $\pi^* \notin \text{hist}(\tilde{\boldsymbol{t}}_n)$, then we reject the proposal; otherwise, we accept it with probability

$$1 \wedge \frac{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi^*, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)}{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)}, \tag{22}$$

which follows because $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \mathbb{P}(\Pi = \pi^* \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)$. The ratio in (22) has a complicated expression but can be computed in time proportional to only the degrees, with respect to $\tilde{\boldsymbol{g}}_n$, of π_j, π_k, and the parent nodes $\text{pa}(\pi_j), \text{pa}(\pi_k)$, where the notion of parent node is defined in (23). We give a detailed description of how to efficiently compute (22) and determine whether $\pi^* \in \text{hist}(\tilde{\boldsymbol{t}}_n)$ in Section S3.5 of the Appendix; in particular, see Section S3.5.2 which uses results from Section S3.5.1. Even with our efficient implementation however, updating π by sampling transpositions is considerably slower than sampling π directly via (19).

The transposition sampler does not change the root node since j, k are not allowed to take on the value 1. To sample a new root node, we fix $k_0 \in \mathbb{N}$ and generate a new proposal π^* by shuffling the first k_0 entries of π. We then accept π^* if it is a valid history and with probability (22). Finally, we note that under the seq-PAPER* model with tree edge removal, our method for sampling π is exactly the same. Since we condition on $\tilde{\boldsymbol{T}}_n$, it makes no difference whether we have deletion noise or not.

Remark 9. Sheridan et al. (2012) and Bloem-Reddy et al. (2018) use the idea of swapping *adjacent* elements of an ordering π for a Poisson growth attachment models and a sequential edge-growth model referred to as Beta NTL respectively. In contrast, under the seq-PAPER model, we can compute non-adjacent swap proposal probabilities efficiently and hence, we can explore the permutation space of π faster. This is because the seq-PAPER is a simpler model and also because we restrict ourselves to a spanning tree, which simplifies many parts of the calculations. We note that sampling π through non-adjacent pair swaps can also be used for the model $\boldsymbol{G}_n = \boldsymbol{T}_n + \boldsymbol{R}_n$ where $\boldsymbol{T}_n$ is not shape-exchangeable, for instance when the attachment probability is $\phi(D_{\boldsymbol{T}_{t-1}}(w_t))$ for

some non-affine function $\phi(\cdot)$ instead of the affine expression given in (1). Finally, We emphasize that inference for the vanilla PAPER model is significantly faster than any form of swapping-based Metropolis samplers since it directly samples the entire ordering.

Algorithm 1 Generating $\pi \in \text{hist}(\tilde{\boldsymbol{f}}_n)$ according to $\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)$ in ER noise settings.

Input: Labeled forest $\tilde{\boldsymbol{f}}_n$ with K trees, denoted $\tilde{\boldsymbol{t}}^1, \dots, \tilde{\boldsymbol{t}}^K$.
Output: $\pi \in \text{hist}(\tilde{\boldsymbol{f}}_n)$.

1: **for** $k = 1, 2, \dots, K$ **do**:
2: Choose node $u^k \in V(\tilde{\boldsymbol{t}}^{(k)})$ with probability (18) with PAPER(α, β, θ) model and with probability (20) under PAPER$(\alpha, \beta, K, \theta)$ or PAPER$(\alpha, \beta, \alpha_0, \theta)$.
3: **end for**
4: Let $\tilde{s} = \{u^1, u^2, \dots, u^K\}$ be the set of roots, and

- under PAPER(α, β, θ), let $\pi_1 = u^1$ and let $t_0 = 2$,
- under PAPER$(\alpha, \beta, K, \theta)$, let $\pi_{1:K} = \tilde{s}$ in a random ordering and let $t_0 = K + 1$.
- under PAPER$(\alpha, \beta, \alpha_0, \theta)$, choose $u^k \in \tilde{s}$ with probability $|\tilde{\boldsymbol{t}}^k|/n$, let $\pi_1 = u^k$, let $t_0 = 2$.

5: Generate $\pi_{t_0:n}$ as a uniformly random permutation of $\mathcal{U}_n \backslash \pi_{1:(t_0-1)}$.
6: **for** $t = t_0, t_0 + 1, \dots, n$ **do**:
7: Let $v_1 = \pi_t$, $v_2 = \text{pa}(v_1)$, $\dots, v_k = \text{pa}(v_{k-1})$ where k is the largest integer such that $v_1, v_2, \dots, v_k \notin \pi_{1:(t-1)}$. ▷ pa$(v)$ denotes the parent of v with respect to $\tilde{\boldsymbol{f}}_n$ rooted at $\tilde{s}$.
8: Set $\pi_t = v_k$, $t_k = \pi^{-1}(v_k)$, and $\pi_{t_k} = v_1$.
9: **end for**

4.2 Sampling the forest

In this section, we describe stage B of the Gibbs sampling algorithm. For a fixed ordering π and a spanning forest $\tilde{\boldsymbol{f}}_n$, we may obtain a set of roots $\tilde{s}$ for each of the component trees of $\tilde{\boldsymbol{f}}_n$ by taking the earliest node (according to π) of each tree. Viewing $\tilde{\boldsymbol{f}}_n$ as being rooted at $\tilde{s}$ induces parent-child relationships between all the nodes.

To define the parent-child relationship formally, let $\tilde{\boldsymbol{f}}_n$ be a forest with disjoint component trees $\tilde{\boldsymbol{t}}^1, \dots, \tilde{\boldsymbol{t}}^K$ and let $\tilde{s} = \{u^1, u^2, \dots, u^K\}$ be a set of root nodes such that $u^k \in V(\tilde{\boldsymbol{t}}^k)$. Let u be any node not in $\tilde{s}$ and suppose $u \in V(\tilde{\boldsymbol{t}}^k)$. There exists a unique node $v \in V(\tilde{\boldsymbol{t}}^k)$ such that v is a neighbor of u in $\tilde{\boldsymbol{f}}_n$ and that the unique path from u to the root u^k contains v. We say v the *parent node* of u and write

$$pa(u) \equiv pa_{\tilde{\boldsymbol{f}}_n^{(\tilde{s})}}(u) = \text{parent of } u \text{ with respect to } \tilde{\boldsymbol{f}}^{(\tilde{s})}. \tag{23}$$

For a root node $u \in \tilde{s}$, we let $pa(u) := \emptyset$ for convenience. Since every edge in $\tilde{\boldsymbol{f}}_n$ is between a node and its parent, the set of parents $\{pa(u)\}_{u \in \mathcal{U}_n}$ specifies the $n - K$ edges in $\tilde{\boldsymbol{f}}_n$ and hence uniquely specifies the forest $\tilde{\boldsymbol{f}}_n$ and the root nodes $\tilde{s}$.

Our Gibbs sampler updates the forest $\tilde{\boldsymbol{f}}_n$ by iteratively updating the parent of each of the nodes, which adds and removes a single edge from $\tilde{\boldsymbol{f}}_n$ (it is possible to add and remove the same edge so that the forest does not change) or, in the random K setting, we may remove a single edge and add a new root node or remove a root node and add a single edge.

To be precise, the latent tree $\tilde{\boldsymbol{F}}_n$ and root set $\tilde{S}$ induces a latent parent of each node which we denote $pa_{\tilde{\boldsymbol{F}}_n^{(\tilde{S})}}(\cdot)$. For every node u, we generate a new parent u' according to the conditional

distribution

$$Q_u(u') := \mathbb{P}\left(pa_{\tilde{\boldsymbol{F}}_n^{(\tilde{s})}}(u) = u' \,\middle|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n, \left\{ pa_{\tilde{\boldsymbol{F}}_n^{(\tilde{s})}}(v) = pa_{\tilde{\boldsymbol{f}}_n^{(\tilde{s})}}(v) \right\}_{v \neq u} \right), \tag{24}$$

and then replace the old edge $(u, pa(u))$ with (u, u'). Since we condition on the arrival ordering Π, probability (24) is non-zero only when u' arrives prior to u, i.e. $\pi^{-1}u' < \pi^{-1}u$, and $(u, u') \in E(\tilde{\boldsymbol{g}}_n)$. In other words, if $\pi^{-1}u = t$, then $Q_u(\cdot)$ is supported on the set of nodes $\pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(u)$. In the random K setting, u' is allowed to be empty in which case $Q_u(\cdot)$ is supported on $\{\emptyset\} \cup \big(\pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(u)\big)$ where $N_{\tilde{\boldsymbol{g}}}(u)$ is the set of neighbors of u on the graph $\tilde{\boldsymbol{g}}_n$. Our sampling procedure then generate the parents for $\pi_1, \pi_2, \pi_3, \ldots$ sequentially. In Figure 12, we illustrate how we may generate a new parent for π_5 (node C) by choosing one of the edges that connects π_5 with one of the earlier nodes $\pi_{1:4}$.

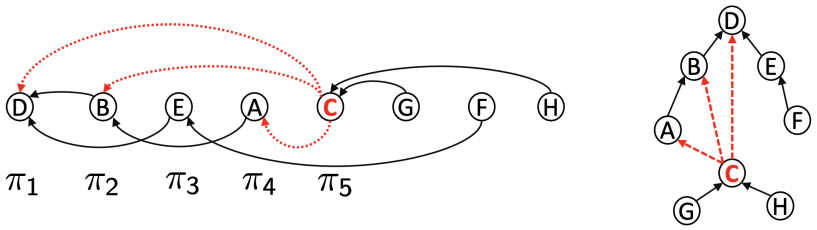


Figure 12: Sampling a parent for π_5 (node C).

At iteration t, to compute $Q_{\pi_t}(\cdot)$ with respect to π_t, for each node v in the support of $Q_{\pi_t}(\cdot)$, we let $\tilde{\boldsymbol{f}}_n^{(v,\pi_t)}$ denote the forest formed by removing the old edge $(\mathrm{pa}(\pi_t), \pi_t)$ and adding the new edge (v, π_t). We note that v is allowed to be the old parent so that we may have $\tilde{\boldsymbol{f}}_n = \tilde{\boldsymbol{f}}_n^{(v,\pi_t)}$. Then, for any w_t in the support of $Q_{\pi_t}(\cdot)$, we have

$$Q_{\pi_t}(w_t) = \frac{\mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n^{(w_t,\pi_t)} \,|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)}{\sum_v \mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n^{(v,\pi_t)} \,|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)}. \tag{25}$$

In the PAPER models with Erdős–Rényi edges, We can compute the conditional distribution $\mathbb{P}(\tilde{\boldsymbol{F}}_n = \cdot \,|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ by using the fact that once when we condition on $\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n$, the remaining edges of $\tilde{\boldsymbol{G}}_n$ are uniformly random and the fact that Π and $\boldsymbol{F}_n$ are independent. Thus,

$$\begin{aligned}
&\mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \\
&\quad \propto \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi) \mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n \,|\, \Pi = \pi) \\
&\quad = \binom{\binom{n}{2} - (n - K(\tilde{\boldsymbol{f}}_n))}{m - (n - K(\tilde{\boldsymbol{f}}_n))}^{-1} \mathbb{P}(\boldsymbol{F}_n = \pi^{-1}\tilde{\boldsymbol{f}}_n) \mathbb{1}\{\tilde{\boldsymbol{f}}_n \in \mathcal{F}(\tilde{\boldsymbol{g}}_n)\} \\
&\quad \propto \left\{ \prod_{k=1}^{K(\tilde{\boldsymbol{f}}_n)} \frac{n(n-1)/2 - n + k)}{m - n + k} \right\} \mathbb{P}(\boldsymbol{F}_n = \pi^{-1}\tilde{\boldsymbol{f}}_n) \mathbb{1}\{\tilde{\boldsymbol{f}}_n \in \mathcal{F}(\tilde{\boldsymbol{g}}_n)\}.
\end{aligned} \tag{26}$$

We now discuss the sampling procedure in detail in all the settings.

Single root setting: In the single root setting, we again use the notation $\tilde{\boldsymbol{t}}_n = \tilde{\boldsymbol{f}}_n$ to be consistent with Definition 1. The first term of (26) is a constant since $K(\tilde{\boldsymbol{t}}_n) = 1$ and may thus be ignored. Using the likelihood of APA trees (see Remark 2 as well as Proposition S1 from the Appendix)

and using the fact that $\mathbb{P}(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}}_n) > 0$ when $\pi \in \text{hist}(\tilde{\boldsymbol{t}}_n)$, we have that, for any $w_t \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$,

$$Q_{\pi_t}(w_t) = \frac{\beta D_{\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}}(w_t) + \alpha}{\sum_{v \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \beta D_{\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}}(v) + \alpha},$$

where $\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}$ is the disconnected graph obtained by removing the old edge $(\text{pa}(\pi_t), \pi_t)$ from $\tilde{\boldsymbol{t}}_n$. We summarize the resulting procedure in Algorithm 2. Since we visit every node once and, for a single node u, it takes time $O(D_{\tilde{\boldsymbol{g}}_n}(u))$ to generate a new parent, the overall runtime of the second stage of the algorithm is $O(m)$. The computational complexity is the same under the fixed K setting and the random K setting.

Fixed $K > 1$ setting: Since the number of trees K is fixed, the first term of (26) is again a constant. Using likelihood of APA trees again (see Proposition S2 from the Appendix), we have that for any $w_t \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$,

$$Q_{\pi_t}(w_t) = \frac{\beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(w_t) + 2\beta \mathbb{1}\{w_t \in \pi_{1:K}\} + \alpha}{\sum_{v \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(v) + 2\beta \mathbb{1}\{v \in \pi_{1:K}\} + \alpha},$$

where, as with the single root setting, $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}$ is the forest obtained by removing the old edge $(\text{pa}(\pi_t), \pi_t)$ from $\tilde{\boldsymbol{f}}_n$. The only difference from the single root setting is that we have a higher probability to attach to a root node because of the imaginary self-loop edge. We summarize the procedure in Algorithm 2.

Algorithm 2 Generating spanning forest $\tilde{\boldsymbol{f}}_n$ of $\tilde{\boldsymbol{g}}_n$ under either PAPER(α, β, θ) or PAPER$(\alpha, \beta, K, \theta)$

Input: Graph $\tilde{\boldsymbol{g}}_n$, ordering $\pi \in \text{Bi}([n], \mathcal{U}_n)$, and a spanning forest $\tilde{\boldsymbol{f}}_n$ with K component trees.
Effect: Modifies $\tilde{\boldsymbol{f}}_n$ in place.

1: **for** $t = K+1, \ldots, n$ **do**:
2: Remove old edge $(\pi_t, pa(\pi_t))$ from $\tilde{\boldsymbol{f}}_n$ to obtain $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}$.
3: Choose a node $w_t \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$ with probability proportional to

$$\begin{cases} \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(w_t) + \alpha & \text{under PAPER}(\alpha, \beta, \theta) \\ \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(w) + 2\beta \mathbb{1}\{w \in \pi_{1:K}\} + \alpha & \text{under PAPER}(\alpha, \beta, K, \theta) \end{cases}$$

4: Add new edge (π_t, w_t) to $\tilde{\boldsymbol{f}}_n$.
5: **end for**

Random K roots setting: Under the PAPER$(\alpha, \beta, \alpha_0, \theta)$ model, a node may become a new root in the sampling process and thus we must take into account the first term of (26). Moreover, in this setting, $Q_{\pi_t}(\cdot)$ for node π_t is supported on $\{\emptyset\} \cup \big(\pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)\big)$ since we may turn the node π_t into a new root node, in which case we set its parent to $\emptyset$ by convention. Define $\tilde{\alpha}_0 := \alpha_0 \frac{m-n+K+\mathbb{1}\{\pi_t \notin \tilde{s}\}}{n(n-1)/2-n+K+\mathbb{1}\{\pi_t \notin \tilde{s}\}}$; we then have that, by Proposition S3, for any $w_t \in \{\emptyset\} \cup \big(\pi_{1:(t-1)} \cap$

$N_{\tilde{\boldsymbol{g}}_n}(\pi_t)\big)$,

$$Q_{\pi_t}(w_t) = \frac{\tilde{\alpha}_0}{\tilde{\alpha}_0 + \sum_{v \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(v) + 2\beta \mathbb{1}\{v \in \tilde{s}\} + \alpha} \quad \text{if } w_t = \emptyset$$

$$\text{and } Q_{\pi_t}(w_t) = \frac{\beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(w_t) + 2\beta \mathbb{1}\{w_t \in S\} + \alpha}{\tilde{\alpha}_0 + \sum_{v \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(v) + 2\beta \mathbb{1}\{v \in \tilde{s}\} + \alpha} \quad \text{if } w_t \neq \emptyset,$$

where, if π_t is not a root node, $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}$ is the forest obtained by removing the old edge $(\pi_t, \mathbf{pa}(\pi_t))$ and if π_t is a root node, then $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)} = \tilde{\boldsymbol{f}}_n$. We summarize the resulting procedure in Algorithm 3.

Algorithm 3 Generating spanning forest $\tilde{\boldsymbol{f}}_n$ of $\tilde{\boldsymbol{g}}_n$ under PAPER$(\alpha, \beta, \alpha_0, \theta)$

Input: Graph $\tilde{\boldsymbol{g}}_n$, ordering $\pi \in \mathrm{Bi}([n], \mathcal{U}_n)$, and a spanning forest $\tilde{\boldsymbol{f}}_n$.
Effect: Modifies $\tilde{\boldsymbol{f}}_n$ in place.

1: Let $\tilde{s}$ be the set of root nodes.
2: **for** $t = 2, 3, \ldots, n$ **do**:
3: If $\pi_t \notin \tilde{s}$, remove edge $(\pi_t, pa(\pi_t))$ from $\tilde{\boldsymbol{f}}_n$ to get $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}$. Else, let $\tilde{s} = \tilde{s} \backslash \{w_t\}$ and let $\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)} = \tilde{\boldsymbol{f}}_n$.
4: Choose a node $w_t \in \{\emptyset\} \cup \big(\pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)\big)$ with probability proportional to

$$\begin{cases} \alpha_0 & \text{for } w_t = \emptyset \\ \beta D_{\tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}}(w_t) + 2\beta \mathbb{1}\{w_t \in \tilde{s}\} + \alpha & \text{for } w_t \neq \emptyset \end{cases}$$

5: If $w_t \neq \emptyset$, let $\tilde{\boldsymbol{f}}_n = \tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)} \cup (\pi_t, w_t)$. Otherwise, let $\tilde{s} = \tilde{s} \cup \{\pi_t\}$ and $\tilde{\boldsymbol{f}}_n = \tilde{\boldsymbol{f}}_n^{(\cdot,\pi_t)}$.
6: **end for**

Sequential noise setting: Under the seq-PAPER setting, we use the same sampling procedure but the sampling probabilities become more complicated. From (25), we see that, for $w \in N_{\tilde{\boldsymbol{g}}_n} \cap \pi_{1:(t-1)}$,

$$\begin{aligned} Q_{\pi_t}(w) &\propto \mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(w,\pi_t)} \,|\, \Pi = \pi, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \\ &\propto \underbrace{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(w,\pi_t)}, \Pi = \pi)}_{\text{noise term}} \mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(w,\pi_t)} \,|\, \Pi = \pi). \end{aligned}$$

Under the seq-PAPER model, the noise term also depends on w since choosing a new parent for π_t would change the tree degrees of some of the nodes. Naively computing $Q_{\pi_t}(w)$ takes time $O(n)$, but in Section S3.5.3 of the Appendix (using results from Section S3.5.1), we give a detailed algorithm to compute $Q_{\pi_t}(w)$ in time $O(D_{\tilde{\boldsymbol{g}}_n}(w))$ so that overall, we can sample a new parent for π_t in time proportional to the number of neighbors of neighbors of π_t.

When we have deletion noise, as the case of the seq-PAPER* model, the latent tree $\tilde{\boldsymbol{T}}_n$ need not be a subgraph of $\tilde{\boldsymbol{G}}_n$ and hence, when sampling a new parent for π_t, we must consider all of $\pi_{1:(t-1)}$ and not just graph neighbors of π_t. Thus, we draw $w \in \pi_{1:(t-1)}$ with probability $Q_{\pi_t}(w)$ and set $\mathrm{pa}(\pi_t) = w$. We give the detailed algorithm for computing $Q_{\pi_t}(w)$ in Section S3.5.3 of the Appendix.

4.3 Other aspects of the algorithm

Parameter estimation: To estimate α and β, we derive an EM algorithm in Section S3.1 of the Appendix. The noise level θ is easy to estimate via $\hat{\theta} = \frac{m-(n-1)}{n(n-1)/2-(n-1)}$ in the single root setting.

The inference algorithm in fact does not require knowledge of θ since it conditions on the number of edges m of the observed graph. We discuss some ways to select the number of trees K in the fixed K root setting and ways to estimate α_0 in the random K roots setting in Section S3.4 of the Appendix.

Inference from posterior samples: The Gibbs sampler described in Section 4.1 and Section 4.2 generates a Monte Carlo sequence $\{(\pi^{(j)}, \tilde{\boldsymbol{f}}_n^{(j)})\}_{j=1}^J$ where J is the number of Monte Carlo samples. A straightforward way to approximate the posterior root probability is to use the empirical distribution based on all the $\pi^{(j)}$'s. However, we can construct a much more accurate approximation by taking advantage of the fact that the posterior root probability is easy to compute on a tree.

Consider the single root setting for simplicity where the posterior root probability is $\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ for any node u. In this case, we may compute distributions $Q^{(1)}, Q^{(2)}, \ldots, Q^{(J)}$ over the nodes by

$$Q^{(j)} = \mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(j)}, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) = \mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(j)}) = \frac{h(u, \tilde{\boldsymbol{t}}_n^{(j)})}{h(\tilde{\boldsymbol{t}}_n^{(j)})}.$$

Then, we output $\frac{1}{J}\sum_{j=1}^J Q^{(j)}$ as our approximation of the posterior root distribution. In the multiple roots setting, we use the same procedure except that we compute $u \mapsto \mathbb{P}(u \in \tilde{S} \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n^{(j)})$ and then average across $j \in \{1, 2, \ldots, J\}$.

In the multiple roots setting, each Monte Carlo sample of the forest $\tilde{\boldsymbol{f}}_n^{(j)}$ contain either K disjoint trees in the fixed K setting or a random number of disjoint trees in the random K setting. These disjoint trees provide a posterior sample of the communities on the network and using them, we may estimate the community structure of the network. We provide details on one way of using posterior samples for community recovery in Section 6.3 and 6.4.

The Gibbs sampling algorithm scales to large networks. We are able to run it on networks of up to a million nodes (c.f. Section 6.2.2) on a single 2020 MacBook Pro laptop. To give a rough sense of the runtime, it takes about 1 second to perform one outer loop of the Gibbs sampler on a graph of 10,000 nodes and 20,000 edges. In Section S3.4 of the appendix, we provide more details on practical usage of the Gibbs sampler such as convergence criterion.

Initialization: In the single root setting, to initialize the Gibbs sampling algorithm, we recommend generating the initial tree $\tilde{\boldsymbol{t}}_n$ uniformly at random from the set of spanning trees $\mathcal{T}(\tilde{\boldsymbol{g}}_n)$ of the observed graph, which can be efficiently done via elegant random-walk-based algorithms such as the Aldous–Broder algorithm (Broder; 1989; Aldous; 1990) or Wilson's algorithm (Wilson; 1996). We then initialize π by drawing an ordering uniformly from the history of the initial tree. This initialization distribution is guaranteed to be overdispersed and works very well in practice. The same initialization works for the random K setting. For the fixed K setting, we can form the initial forest by constructing uniformly random spanning tree $\tilde{\boldsymbol{t}}_n$ and uniformly random ordering π as usual, taking the first K nodes of the π as the root nodes, and removing all tree edges between them to obtain an initial $\tilde{\boldsymbol{f}}_n$. We use Wilson's algorithm in our implementation.

5 Theoretical Analysis

We provide theoretical support for our approach by deriving bounds on the size of our proposed confidence sets when the observed graph has the PAPER distribution. In particular, we aim to quantify how the quality of inference deterioriates with the noise level θ, that is, how the size of the confidence set increases with θ. For simplicity, for consider only the single root setting and we do

not take into account approximation errors introduced by the Gibbs sampler, that is, we analyze the confidence set constructed from the exact posterior root probabilities.

We begin with a type of optimality statement which shows that the size of the confidence set $B_\epsilon(\cdot)$, as defined in (8), is of no larger order than any other asymptotically valid confidence set. Intuitively, this is because $B_\epsilon(\cdot)$ can be interpreted as a "Bayes estimator" for the root node.

Lemma 10. *Let ϵ be in $(0,1)$, let $\boldsymbol{G}_n \sim PAPER(\alpha,\beta,\theta)$, and let $\boldsymbol{G}_n^* = \rho \boldsymbol{G}_n$ be the observed alphabetically labeled graph for some $\rho \in Bi([n],\mathcal{U}_n)$. Let $B_\epsilon(\mathbf{G}_n^*)$ be defined as in (7) and (8). Fix any $\delta \in (0,1)$ and let $C_{\delta\epsilon}(\mathbf{G}_n^*)$ be any confidence set for the root node that is labeling-equivariant and has asymptotic coverage level $1-\delta\epsilon$, that is, $\limsup_{n\to\infty} \mathbb{P}(\rho_1 \notin C_{\delta\epsilon}(\mathbf{G}_n^*)) \le \delta\epsilon$. Then, we have that*

$$\limsup_{n\to\infty} \mathbb{P}\big(|B_\epsilon(\mathbf{G}_n^*)| \ge |C_{\delta\epsilon}(\mathbf{G}_n^*)|\big) \le \delta.$$

We provide the proof of Lemma 10 in Section S4 of the appendix.

Ideally, we would compare the size of $B_\epsilon(\cdot)$ with $C_\epsilon(\cdot)$ at the same level. It is however much easier to compare with the more conservative $C_{\delta\epsilon}(\cdot)$. In many cases, the size of a confidence set $|C_\epsilon(\cdot)|$ has bounds of the form $f(n)g(\epsilon^{-1})$ for some functions f and g (see e.g. Banerjee and Bhamidi (2020)) so that comparing with $C_{\delta\epsilon}(\cdot)$ adds only a multiplicative constant to the bound.

Lemma 10 is useful because it is difficult to directly bound the confidence set $B_\epsilon(\cdot)$ as a function of n and the parameters; Lemma 10 shows that we can indirectly upper bound it by analyzing a simpler asymptotically valid confidence set. Our strategy then is to construct confidence sets based on the degree of the nodes whose size is much easier to bound through well-understood probabilistic properties of preferential attachment trees. This leads to our next result which provides explicit bounds on the size of the confidence set $B_\epsilon(\cdot)$ when the underlying tree is LPA.

Theorem 11. *Let $\boldsymbol{G}_n \sim PAPER(\alpha,\beta,\theta)$ for $\beta = 1$, $\alpha = 0$, and $\theta \in [0,1]$. For $t \in [n]$, let $D_{\boldsymbol{G}_n}(t)$ be the degree of node with arrival time t and for $k \in [n]$, let k-$\max(D_{\boldsymbol{G}_n})$ be the k-th largest degree of $\boldsymbol{G}_n$. Let $\delta > 0$ be arbitrary and suppose $\theta \le n^{-\frac{1}{2}-\delta}$. Then, for any $\epsilon > 0$, there exists $L_\epsilon \in \mathbb{N}$ (dependent on δ but not on n) such that*

$$\limsup_{n\to\infty} \mathbb{P}\big\{D_{\boldsymbol{G}_n}(1) \le L_\epsilon\text{-}\max(D_{\boldsymbol{G}_n})\big\} \le \epsilon. \tag{27}$$

As a direct consequence, if $\theta = O(n^{-\frac{1}{2}-\delta})$ for any $\delta > 0$, then, for any $\epsilon \in (0,1)$,

$$|B_\epsilon(\boldsymbol{G}_n^*)| = O_p(1).$$

We relegate the proof of Theorem 11 in Section S4.1 of the appendix and provide a short sketch here: we use results from Peköz et al. (2014) which show that the degree sequence of an LPA tree, when normalized by $\frac{1}{\sqrt{n}}$, converges to a limiting distribution in the ℓ_q sequential metric sense, which shows that (27) holds for the tree degree $D_{\boldsymbol{T}_n}(\cdot)$, that is, the degree of the root node is one of the highest among all the nodes. Since $D_{\boldsymbol{G}_n} = D_{\boldsymbol{T}_n} + D_{\boldsymbol{R}_n}$, we show that if the noise level θ is less than $n^{-1/2-\delta}$ for some $\delta > 0$, then the degree of the noisy edges $D_{\boldsymbol{R}_n}$ has a second order effect and (27) remains valid.

We know from existing results (such as Bubeck, Devroye and Lugosi (2017, Theorem 6); see also Crane and Xu (2021, Corollary 7)) that $|B_\epsilon(\boldsymbol{T}_n^*)|$ is $O_p(1)$ in the $\theta = 0$ case where we observe the LPA tree $\boldsymbol{T}_n^*$. Theorem 11 shows that this phenomenon is quite robust to noise. Indeed, when $\theta = n^{-1/2-\delta}$, the observed graph would have approximately $n^{3/2-\delta}$ noisy edges and only $n-1$ tree edges.

The situation is different when the underlying latent tree has the UA distribution, where $\alpha = 1$ and $\beta = 0$. In this case, we have the following result:

Theorem 12. *Let $\boldsymbol{G}_n \sim PAPER(\alpha, \beta, \theta)$ for $\alpha = 1$, $\beta = 0$, and $\theta \in [0,1]$. For $t \in [n]$, let $D_{\boldsymbol{G}_n}(t)$ be the degree of node with arrival time t and for $k \in [n]$, let $k\text{-}\max(D_{\boldsymbol{G}_n})$ be the k-th largest degree of $\boldsymbol{G}_n$. Suppose $\theta = o\big(\frac{\log n}{n}\big)$ and let $\epsilon \in (0,1)$ be arbitrary. For any $\eta \in (0,1)$, define $L_{\eta,n,\epsilon} := n^{\eta} + \epsilon^{-1} n^{1-(2-\eta)h\left(\frac{\eta}{2-\eta}\right)}$ where $h(x) = (1+x)\log(1+x) - x$ for $x \geq 0$. Then, we have that*

$$\limsup_{n\to\infty} \mathbb{P}\big\{ D_{\boldsymbol{G}_n}(1) \leq L_{\eta,n,\epsilon}\text{-}\max(D_{\boldsymbol{G}_n}) \big\} \leq \epsilon. \tag{28}$$

As a direct consequence, if $\theta = o(\frac{\log n}{n})$, then, for some $\gamma \leq 0.8$, we have that

$$n^{-\gamma}\epsilon^{-1}|B_\epsilon(\boldsymbol{G}_n^*)| = O_p(1) \quad \text{for any } \epsilon \in (0,1).$$

We relegate the proof of Theorem 12 to Section S4.2 of the appendix. The proof technique is similar to that of Theorem 11 except that we use concentration inequalities to derive (28).

Comparing Theorem 12 with Theorem 11, we see two important differences. First, even if the noise level is small, we can no longer guarantee that $|B_\epsilon(\boldsymbol{G}_n^*)|$ is bounded even as n increases. Instead, we have the much weaker bound that $|B_\epsilon(\boldsymbol{G}_n^*)|$ is less than $O(n^\gamma)$ for some $\gamma < 0.8$. We believe this bound is not tight; we observe from simulations in Section 6.1 (see Figure 13) that the size of the confidence set $B_\epsilon(\cdot)$ is indeed $O_p(1)$ even when the noise level is of order $\frac{\log n}{n}$. The bound is sub-optimal because the degree of the nodes is not informative of their latent ordering when the latent tree has the UA distribution; hence, $B_\epsilon(\cdot)$ could be much smaller than confidence sets constructed solely from degree information. Intuitively, this is because largest degree nodes do not persist in uniform attachment as opposed to linear preferential attachment (Dereich and Mörters; 2009; Galashin; 2013).

The second difference is that the noise tolerance is much smaller. We require θ to be smaller than $\frac{\log n}{n}$ rather than $n^{-1/2}$. We conjecture that these rates are tight in the following sense:

Conjecture 13. *Let $\boldsymbol{G}_n \sim PAPER(\alpha, \beta, \theta)$ for $\alpha = 1$, $\beta = 0$, and $\theta \in [0,1]$.*

1. *Suppose $\alpha = 0$ and $\beta = 1$ (LPA). If $\theta = o(n^{-1/2})$, then $|B_\epsilon(\boldsymbol{G}_n^*)| = O_p(1)$ and if $\theta = \omega(n^{-1/2})$, then every asymptotically valid confidence set has size that diverges with n.*
2. *Suppose $\alpha = 1$ and $\beta = 0$ (UA). If $\theta = o(\frac{\log n}{n})$, then $|B_\epsilon(\boldsymbol{G}_n^*)| = O_p(1)$ and if $\theta = \omega(\frac{\log n}{n})$, then every asymptotically valid confidence set has size that diverges with n.*

We provide empirical support for this conjecture in Section 6.1, particularly Figure 13. In those experiments, we see that, when the latent tree has the LPA distribution and when $\theta = cn^{-1/2}$ where $c > 0$ is small, the size of B_ϵ does not increase with n; however, when c (and hence θ) is large, B_ϵ is larger when the size of the graph n is larger. The same phenomenon holds when the latent tree has the UA distribution when $\theta = c\frac{\log n}{n}$.

6 Empirical Studies

We have implemented the inference approach in Section 3 and the sampling algorithm in Section 4 in a Python package named `paper-network`, which can be installed via command line `pip install paper-network` on the terminal and then imported in Python via `import PAPER`. The source code of the package, along with examples and documentation, are available at the website `https://github.com/nineisprime/PAPER`. All the code used in this Section are also available there under the directory `paperexp`. We also give detailed sampler diagnostics information in Section S5.4 of the Appendix.

6.1 Simulation

Frequentist coverage in the single root setting: In our first simulation study, we empirically verify Theorem 7 by showing that a level $1-\epsilon$ credible set for the root node constructed from the posterior root probabilities has frequentist coverage at exactly the same level $1-\epsilon$. We consider three different settings of parameters: $\alpha = 0, \beta = 1$ (LPA), $\alpha = 1, \beta = 0$ (UA), and $\alpha = 8, \beta = 1$. We generate $\boldsymbol{G}_n^*$ according to the PAPER(α, β, θ) model with $n = 3,000$ nodes and $m = 7,500$ edges. We then estimate α and β using the method given in Section S3.1, compute the level $\epsilon \in \{0.2, 0.05, 0.01\}$ credible sets, and record whether they cover the true root node. We repeat the experiment over 300 independent trials and report the results in Table 2. We observe that the credible sets attain the nominal coverage and that the size of the credile sets are small compared to the number of nodes n.

(α, β)	(0, 1)	(1, 0)	(8, 1)	(0, 1)	(1, 0)	(8, 1)	(0, 1)	(1, 0)
Theoretical coverage	0.8	0.8	0.8	0.95	0.95	0.95	0.99	0.99
Empirical coverage	**0.8**	**0.823**	**0.82**	**0.937**	**0.943**	**0.94**	**0.983**	**0.993**
Ave. conf. set size	7	12	9	42	42	31	183	115

Table 2: Empirical coverage of our confidence set for the root node. We report the average over 300 trials. Graph has $n = 3000$ nodes and $m = 7,500$ edges in all cases.

Size of the confidence set: In our second simulation study, we study the effect of the sample size n and the magnitude of the noisy edge probability θ on the size of the confidence set. We let $\boldsymbol{G}_n^*$ be the observed graph with n nodes and m edges according to the PAPER(α, β, θ) model where we consider $(\alpha, \beta) = (0, 1)$ (LPA) or $(1, 0)$ (UA). Since a tree with n nodes always contains $n-1$ edges, $\frac{n^2}{2}\theta + n$ is approximately equal to the number of edges m in the observed graph $\boldsymbol{G}_n^*$.

We empirically show that the confidence set size does not depend on n so long as θ is much smaller than $n^{-1/2}$ for LPA and much smaller than $\frac{\log n}{n}$ for UA. To that end, we set $m = cn\sqrt{n}$ for $c \in \{0.1, 0.2, 0.4, 0.6, 0.8, 1\}$ for LPA and $m = cn \log n$ for $c \in \{0.15, 0.2, 0.4, 0.6, 0.8\}$ for UA. We then plot the average size of the confidence set with respect to c for $n \in \{5000, 10000\}$. We plot the curve for $n = 5,000$ and for $n = 10,000$ on the same figure and observe that, when c is small, the two curves overlap completely but when c is large, the $n = 10,000$ curve lies above the $n = 5,000$ curve. This provides empirical support to Theorem 11 and Theorem 12. In fact, this experiment shows that the bound of n^γ on the size of the confidence set in Theorem 12 is loose; the actual size does not increase with n. The fact that the confidence set size seems to diverge with n when c is larger supports Conjecture 13 and suggests that the problem of root inference exhibits a phase transition when $\theta \approx \frac{1}{\sqrt{n}}$ under the LPA model and $\theta \approx \frac{\log n}{n}$ under the UA model.

Frequentist coverage under sequential noise models: In our third simulation study, we verify Theorem 7 for the seq-PAPER model with sequential noise described in Section 2.3. We generate $\boldsymbol{G}_n^*$ according to both the seq-PAPER$(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta})$ model and the seq-PAPER$^*(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta}, \eta)$ model with deletion noise. We then construct the credible sets for the root node from posterior root probabilities computed via the algorithm given in Section 4. We repeat the experiment over 200 independent trials and report the results in Tables 3 and 4. We observe that the credible sets attain the nominal coverage. We also note that Table 4 shows that the seq-PAPER* model can tolerate tree deletion probability up to $\eta = 0.08$ without significant increase in the confidence set sizes.

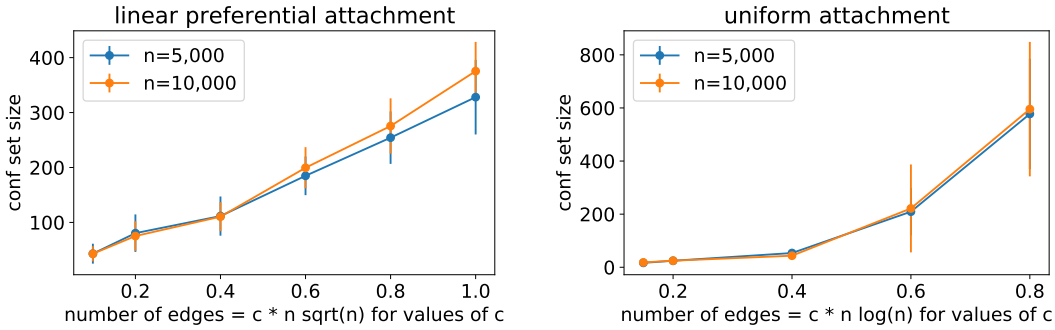


Figure 13: Size of the confidence set vs. the number of edges.

(α, β) (with $\tilde{\alpha} = \alpha$, $\tilde{\beta} = \beta$)	(0, 1)	(1, 0)	(0, 1)	(1, 0)	(0, 1)	(1, 0)
Theoretical coverage	0.8	0.8	0.95	0.95	0.99	0.99
Empirical coverage	**0.795**	**0.895**	**0.935**	**0.965**	**0.970**	**0.995**
Ave. conf. set size	7	7	25	16	56	28

Table 3: Empirical coverage of our confidence set for the seq-PAPER$(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta})$ model without deletion noise, with $\theta = 1.5$ and $\tilde{\alpha} = \alpha$ and $\tilde{\beta} = \beta$. We report the average over 200 trials. Graph has $n = 600$ nodes and around $m \approx 1500$ edges in all cases.

η (tree edge deletion probability)	0	0	0.04	0.04	0.08	0.08
Theoretical coverage	0.8	0.95	0.8	0.95	0.8	0.95
Empirical coverage	**0.825**	**0.96**	**0.84**	**0.95**	**0.85**	**0.98**
Ave. conf. set size	5.9	14.1	6.3	15.0	6.7	15.9

Table 4: Empirical coverage of our confidence set for the seq-PAPER$^*(\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta}, \eta)$ model with deletion noise, with $\alpha = 0, \beta = 1, \tilde{\alpha} = 8, \tilde{\beta} = 1, \theta = 1.5$ in all cases. We report the average over 200 trials. Graph has $n = 300$ nodes and around $m \approx 750$ edges in all cases.

Frequentist coverage for multiple roots: Our next simulation study is similar to the first except that we generate graphs from the PAPER$(\alpha, \beta, K, \theta)$ model with $K = 2$. We construct our credible sets as described in Section 3.3 and verify Theorem 8 by showing that the credible set at level $1 - \epsilon$ also has frequentist coverage at exactly the same level. We consider two different settings of parameters: $\alpha = 0, \beta = 1$ (LPA) and $\alpha = 1, \beta = 0$ (UA). We generate $\boldsymbol{G}_n^*$ according to the PAPER$(\alpha, \beta, K, \theta)$ model with $n = 700$ nodes, $m = 1,000$ edges, and $K = 2$. We then estimate α and β using the method given in Section S3.1, compute the level $\epsilon \in \{0.2, 0.05, 0.01\}$ credible sets, and record whether they contain the true set of root nodes. We repeat the experiment over 200 independent trials and report the results in Table 5. We observe that the credible sets attain the nominal coverage. In the LPA setting, the size of the credible sets are small but in the UA setting, the sizes of the credible sets become much larger. We relegate an in-depth analysis of this phenomenon to future work.

(α, β)	(0, 1)	(1, 0)	(0, 1)	(1, 0)	(0, 1)	(1, 0)
Theoretical coverage	0.8	0.8	0.95	0.95	0.99	0.99
Empirical coverage	**0.826**	**0.826**	**0.933**	**0.964**	**0.974**	**0.985**
Ave. conf. set size	5	57	12	155	31	295

Table 5: Empirical coverage of our confidence set for the set of $K = 2$ root nodes. We report the average over 200 trials. Graph has $n = 700$ nodes and $m = 1,000$ edges in all cases.

Posterior on K in the random K roots setting: In our last simulation experiment, we generate PAPER graphs with $K = 2$ roots but perform posterior inference using the PAPER$(\alpha, \beta, \alpha_0, \theta)$ model and study resulting posterior distribution over the number of roots K. We consider two different settings of parameters: $\alpha = 0, \beta = 1$ (LPA) and $\alpha = 1, \beta = 0$ (UA). We generate $\boldsymbol{G}_n^*$ according to the PAPER$(\alpha, \beta, K, \theta)$ model with $n = 700$ nodes, $m = 1,000$ edges, and $K = 2$. We report the posterior distribution over K, averaged over 20 independent trials, in Figure 14. We observe that, in both cases, the mode of the posterior distribution over K is 2, which is the true number of roots. However, the distributions exhibits high variance, which could be due to the fact that the two true latent trees may have significantly different sizes.

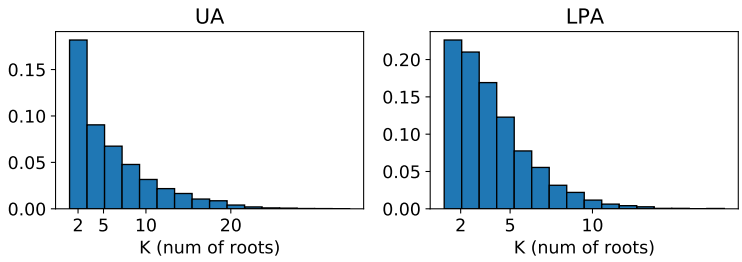


Figure 14: Posterior distribution over K averaged across 20 independent trials. **Left:** networks have two latent UA trees. **Right:** networks have two latent LPA trees.

6.2 Single root analysis on real data

We now apply the single root PAPER model on real world networks. In a few cases (Section 6.2.1), we can ascertain from domain knowledge that the network originated from a single root node but more often, we use the single root model to identify important nodes and subgraphs (Section 6.2.2).

6.2.1 Flu transmission network

We analyze a person-to-person contact network among 32 students in a London classroom during a flu outbreak (Hens et al.; 2012). We extract the data from Figure 3 in Hens et al. (2012) and illustrate the network in the left sub-figure of Figure 15. Public health investigation revealed that the outbreak originated from a single student, which is the true patient zero and shown as the orange node in Figure 15. We apply the PAPER model with a single root to this network. We estimate that $\beta = 1$ and $\alpha = 53.06$ using the method described in Section S3.1 and compute the $60\%, 80\%, 95\%$, and 99% confidence sets. All the confidence sets contain the true patient zero and

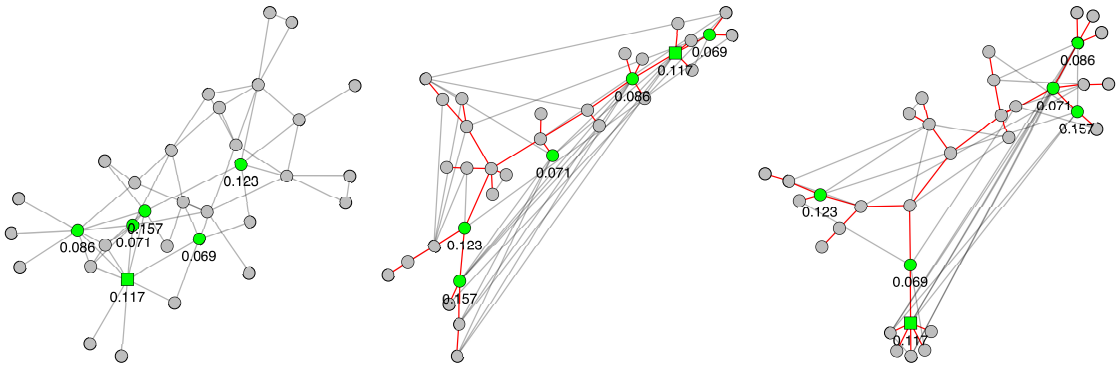


Figure 15: **Left:** contact network among 32 students in a flu outbreak. **Center and right:** two examples of the latent tree generated by the Gibbs sampler.

their sizes are as followed:

60%: 6 nodes 80%: 10 nodes 95%: 19 nodes 99%: 27 nodes.

We provide the approximate posterior root probabilities of the top 7 nodes in Figure 15. The true patient zero has a posterior root probability of 0.11 is the node with the 3rd highest posterior root probability. In the center and right sub-figure of Figure 15, we also show two of the latent trees $\tilde{\boldsymbol{T}}_n$ that were generated by the Gibbs sampler.

6.2.2 Visualizing central subgraphs

Large scale real graphs are difficult to visualize but one can often learn salient structural properties of a graph by visualizing a smaller subgraph that contains the most important nodes. In this section, we apply the single root PAPER model on four large networks and, for each graph, display the subgraph that comprises the 200 nodes with the highest posterior root probability. We see that the result reveals striking differences between the different graphs. Unfortunately, we do not have the node labels on any of these four graphs and can only make qualitative interpretations of the results.

MathSciNet collaboration network: We first consider a collaboration network of research publications from MathSciNet, which is publicly available in the Network Repository (Rossi and Ahmed; 2015) at the link `http://networkrepository.com/ca-MathSciNet.php`. This network has $n = 332,689$ nodes and $m = 820,644$ edges, with a maximum degree of 496. Using the method described in Section S3.1, we estimate $\beta = 1$ and $\alpha = 0$. The sizes of confidence sets are:

60%: 3 nodes 80%: 6 nodes 95%: 21 nodes 99%: 112 nodes.

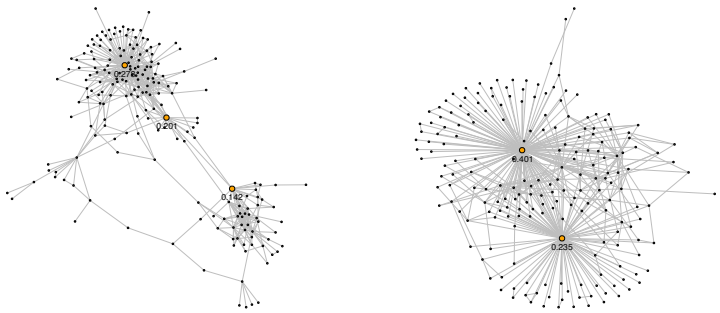
We display the subgraph containing the 200 nodes with the highest posterior root probability in Figure 16a. We observe that the subgraph reveals a cluster structure that may represent the different academic disciplines.

University of Notre Dame website network: We study a network of hyperlinks between webpages of University of Notre Dame (Albert et al.; 1999), which is publicly available at the website `https://snap.stanford.edu/data/web-NotreDame.html`. This network has $n = 325,729$

nodes and $m = 1,090,108$ edges, with a maximum degree of 10,721. Using the method described in Section S3.1, we estimate $\beta = 1$ and $\alpha = 0$. The sizes of confidence sets are:

$$60\%\text{: 2 nodes} \qquad 80\%\text{: 21 nodes} \qquad 95\%\text{: 524 nodes} \qquad 99\%\text{: 3498 nodes}\ .$$

We observe that the central subgraph (shown in Figure 16b) reveals two hub nodes with many sparsely connected "spokes".



(a) MathSciNet subgraph (b) Notre Dame subgraph

Figure 16: Subgraph of the 200 nodes with highest posterior root probabilities.

6.3 Community recovery with the fixed K model

In this section, we show that we can use the PAPER model with multiple roots for community recovery on real world networks. To estimate the community membership from the posterior samples, we use a greedy matching procedure. To be precise, our Gibbs sampler outputs a sequence of forests $\tilde{\boldsymbol{f}}_n^{(1)}, \ldots, \tilde{\boldsymbol{f}}_n^{(J)}$ where J is the number of Monte Carlo samples. Each forest $\tilde{\boldsymbol{f}}_n^{(j)}$ contains K component trees which we denote $\tilde{\boldsymbol{t}}^{(1,j)}, \tilde{\boldsymbol{t}}^{(2,j)}, \ldots, \tilde{\boldsymbol{t}}^{(K,j)}$. We write $Q_k^{(j)}(\cdot) := \mathbb{P}(\Pi_1 = \cdot \,|\, \tilde{\boldsymbol{T}} = \tilde{\boldsymbol{t}}^{(k,j)})$ as the posterior root distribution of the k-th tree of the j-th Monte Carlo sample. Since the tree labels may switch from sample to sample, we use the following matching procedure: we maintain K distributions $Q_1(\cdot), Q_2(\cdot), \ldots, Q_K(\cdot)$ and initially set $Q_k = Q_k^{(1)}$ for all $k \in [K]$. Then, for $j = 2, 3, \ldots, J$, we use the Hungarian algorithm to compute a one-to-one matching $\sigma : [K] \to [K]$ that minimizes the overall total variation distance

$$\sum_{k=1}^{K} \mathrm{TV}(Q_k^{(j)}, Q_{\sigma(k)}).$$

Once we compute the matching, we then update $Q_{\sigma(k)} \leftarrow \frac{j-1}{j} Q_{\sigma(k)} + \frac{1}{j} Q_k^{(j)}$.

In this way, we interpret $Q_1, \ldots, Q_K$ as the average posterior root distributions for the K trees across all the Monte Carlo samples and using the matching, we may also compute the posterior probability $\mathbb{P}(\ u \text{ in tree } k \ \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$, which allows us to perform community detection – we put node u in cluster k if $\mathbb{P}(\ u \text{ in tree } k \ \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \geq \mathbb{P}(\ u \text{ in tree } k' \ \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n)$ for all $k' \neq k$. We use the greedy matching procedure for computational efficiency – slower but more principles approaches are studied by e.g. Wade and Ghahramani (2018).

6.3.1 Karate club network

We apply the PAPER model to Zachary's karate club network Zachary (1977), which is publicly available at `http://www-personal.umich.edu/ mejn/netdata/`. The karate club network has

$n = 34$ nodes and $m = 76$ edges, where two individuals share an edge if they socialize with each other. The network has two ground truth communities, one led by the instructor and one led by the administrator (shown as rectangular nodes in Figure 17. These two communities later split into two separate clubs. In this case, we apply the PAPER model with $K = 2$ roots. For every node u, we consider the community membership probability $\mathbb{P}(u \text{ in tree } 1 \,|\, \tilde{\boldsymbol{G}}_n)$ and assign u to community 1 if and only if this value is greater than 0.5. We show the result in in Figure 17, where each node has a color that reflects its community membership probability.

We correctly cluster all but one node, which matches the performance of degree-corrected SBM Karrer and Newman (2011); Amini et al. (2013) (DCSBM)–the current the state of the art model for community detection. The node that we misclassify has a posterior probability $\mathbb{P}(u \text{ in tree } 1 \,|\, \tilde{\boldsymbol{G}}_n) = 0.47$, indicating that the model is indeed unsure of whether it belong in community 1 or 2. We note that the PAPER model requires only 3 parameters whereas the DCSBM for this network requires 38 parameters because each node has a degree correction parameter. SBM without degree correction performs badly Karrer and Newman (2011).

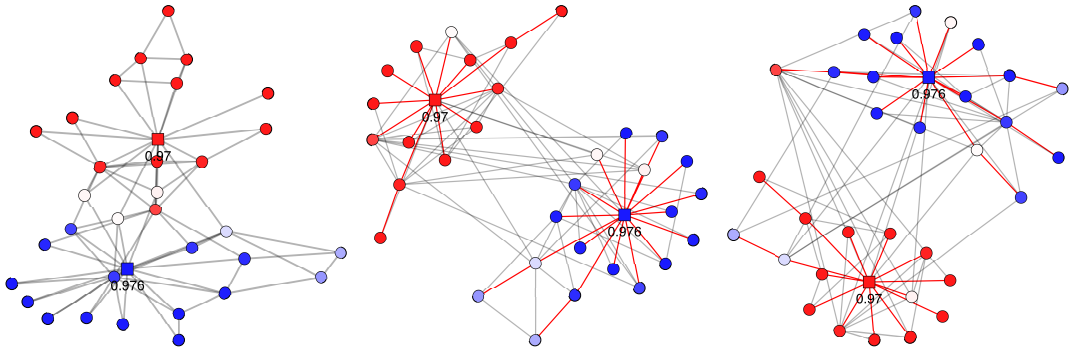


Figure 17: **Left:** karate club network where node color reflects community membership probability. **Center and right:** two examples of the latent forest generated by the Gibbs sampler.

6.3.2 Political blogs network

Next, we analyze a political blogs network (Adamic and Glance; 2005) that is frequently used as a benchmark for network clustering algorithms; the full network is publicly available at the website `http://www-personal.umich.edu/ mejn/netdata/`. This network contains $m = 16,714$ edges between $n = 1,222$ blogs, where two blogs are connected if one contains a link to the other. For simplicity, we treat the network as undirected.

The network again has two ground truth communities, one that comprise of left-leaning blogs and one that comprises of right-leaning blogs. We again apply the PAPER model with $K = 2$ roots and for every node u, we compute the community membership probability $\mathbb{P}(u \text{ in tree } 1 \,|\, \tilde{\boldsymbol{G}}_n)$ and assign u to community 1 if and only if this value is greater than 0.5. We show the result in in Figure 18, where each node has a color that reflects its community membership probability.

Our overall misclustering error rate is **9.1%**, which is high compared to current state of the art approaches; for example, the SCORE method (Jin; 2015) attains an error rate of about 5%. However, we compute the misclustering error rate with respect to only the top 400 nodes with the highest posterior root probabilities, which can be interpreted as the most important nodes in the graph, our misclustering error rate drops to **3.5%**. This confirms our intuition that the PAPER model, when used for clustering, is more reliable for central nodes than for peripheral nodes.

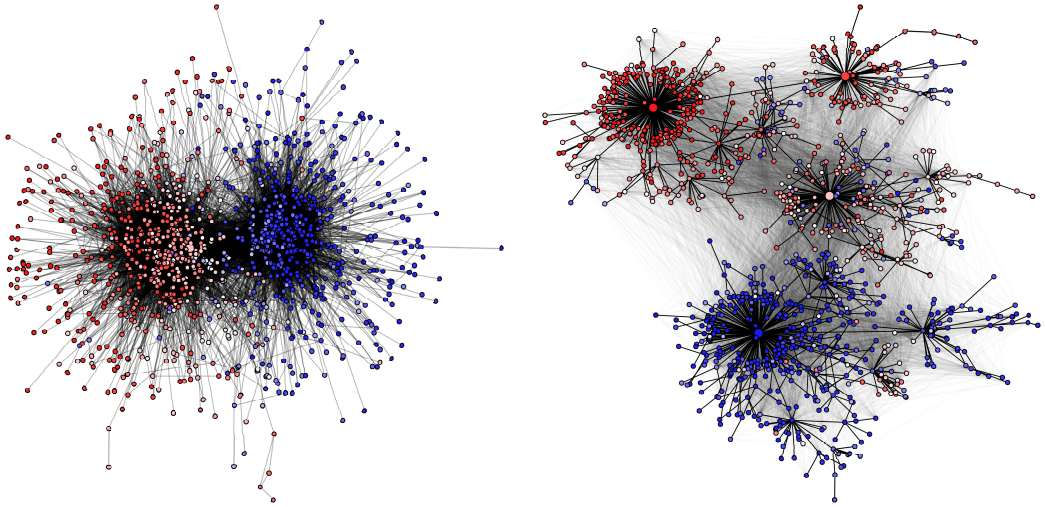


Figure 18: **Left:** political blog network where node color reflects community membership probability. **Right:** one example of a forest generated by the Gibbs sampler. The 5 nodes with the larger marker comprise the 95% confidence set for the roots.

6.4 Community discovery with the random K model

For networks with an unknown number of small and possibly overlapping communities, the random K model PAPER$(\alpha, \beta, \alpha_0, \theta)$ can be useful for discovering complex community structures. To extract community information from the posterior samples, we again use a greedy matching procedure. To be precise, in the random K setting, our proposed Gibbs sampler outputs a sequence of forests $\tilde{\boldsymbol{f}}_n^{(1)}, \ldots, \tilde{\boldsymbol{f}}_n^{(J)}$ where J is the number of Monte Carlo samples. We write each forest $\tilde{\boldsymbol{f}}_n^{(j)}$, for $j \in [J]$, as a collection of trees $\{\tilde{\boldsymbol{t}}^{(1,j)}, \ldots, \tilde{\boldsymbol{t}}^{(K_j,j)}\}$ where K_j is the number of trees in $\tilde{\boldsymbol{f}}_n^{(j)}$. For $j \in [J]$ and $k \in [K_j]$, we write $Q_k^{(j)}(\cdot) = \mathbb{P}(\Pi_1 = \cdot \,|\, \tilde{\boldsymbol{T}} = \tilde{\boldsymbol{t}}^{(k,j)})$ as the posterior root distribution of the k-th tree in the j-th Monte Carlo sample. To summarize the output in an interpretable way, we do the following:

1. We initialize $K_{\text{all}} = \max_{j \in [J]} K_j$ and $Q_k = Q_k^{(1)}$ for $k = 1, 2, \ldots, K_1$. For $k = K_1 + 1, \ldots, K_{\text{all}}$, we initialize $Q_k(\cdot) = 0$.

2. For $j = 2, 3, \ldots, J$, we match $\{Q_1, \ldots, Q_{K_{\text{all}}}\}$ with $\{Q_1^{(j)}, \ldots, Q_{K_j}^{(j)}\}$ by computing a one-to-one matching $\sigma : [K_j] \to [K_{\text{all}}]$ that minimizes

$$\sum_{k=1}^{K_j} \text{TV}(Q_k^{(j)}, Q_{\sigma(k)}).$$

   For every $k \in [K_j]$, if the total variation distance between the k-th pair of the matching is too large, that is $\text{TV}(Q_k^{(j)}, Q_{\sigma(k)}) > 0.75$, then we create a new set $K_{\text{all}} \leftarrow K_{\text{all}} + 1$ and set $Q_{K_{\text{all}}+1} \leftarrow Q_k^{(j)}$; otherwise, we perform the update $Q_{\sigma(k)} \leftarrow \frac{j-1}{j} Q_{\sigma(k)} + \frac{1}{j} Q_k^{(j)}$.

3. We output $\{Q_1, \ldots, Q_{K_{\text{all}}}\}$ as the discovered clusters, represented as posterior root probability distributions.

For all of our experiments, we only include trees that contain at least 1% of the total number of nodes. For each discovered cluster Q_ℓ for $\ell \in [K_{\text{all}}]$, we also compute ρ_{Q_ℓ} as the number of Monte

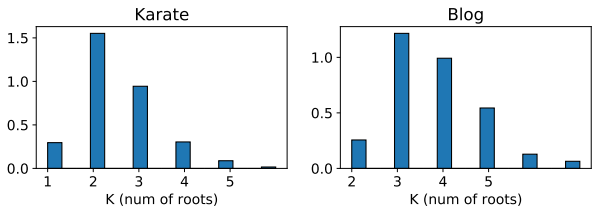


Figure 19: Posterior over K using the random K roots model on the karate club network (**left**) and the political blog network (**right**).

Carlo iteration $j \in [J]$ where we match Q_ℓ with $Q_k^{(j)}$, i.e. $\sigma(k) = \ell$, and update Q_ℓ. We then compute $\frac{\rho_{Q_\ell}}{J}$ as the posterior frequency of cluster Q_ℓ.

In order to check that the random K model is reasonable, we first apply it to the karate club and the political blog networks, which we know contain two underlying clusters, and analyze the resulting posterior distribution over the number of cluster-trees K. We provide the results for the karate club network in the left part of Figure 19, in which we see that the posterior mode is at $K = 2$. For the political blog network, the Gibbs sampler tends to produce a few large clusters and many tiny clusters of fewer than 10 nodes. Therefore, to compute the posterior over K, we count only clusters that have at least 12 nodes (1% of the total number of nodes) and give the results in the right part of Figure 19. The posterior mode in this case is $K = 3$, which is reasonably close to the ground truth.

We also analyze an air route network (Guimera et al.; 2005) of $n = 3,618$ airports and $m = 14,142$ edges where two airports share an edge if there is a regularly scheduled flight between them. We remove the direction of the edges and treat the network as undirected. The dataset is publicly available at `http://seeslab.info/downloads/air-transportation-networks/`. Using the random K model, we discover a large central cluster containing major airports around the world and various small clusters that correspond to more remote regions such as airports on Pacific and Polynesian islands, airports in Alaska, and airports in the Canadian Northwest Territories. For sake of brevity, we defer the detailed results to Section S5.2 of the Appendix.

6.5 Analysis of statistician co-authorship network

We now apply PAPER models to perform an extensive analysis of a statistician co-authorship network constructed by Ji and Jin (2016). In this network, each node corresponds to a statistician and two nodes u and v have an edge between them if they have co-authored 1 or more papers in either *Journal of Royal Statistical Society: Series B*, *Journal of the American Statistical Association*, *Annals of Statistics*, or *Biometrika* from 2002 to 2013. We consider only the largest connected component which has $n = 2263$ nodes and $m = 4388$ edges. Ji and Jin (2016) in their manuscript (Section 4.3) refers to this network as "Coauthorship Network (B)". We emphasize that since the data reflect only coauthorship in 4 journals in the period 2002-2013, the results that we produce cannot be used to compare researchers—we use this network only to illustrate PAPER models in a setting where we can more easily assess whether the output is meaningful or not.

Single root analysis: We first use the single root PAPER(α, β, θ) model where we estimate $\alpha = 0$, $\beta = 1$ using the EM algorithm described in Section S3.1. We find that the following 4 nodes have the highest posterior root probabilities: (1) Raymond Carroll with root probability 0.32, (2) Peter Hall with root probability 0.26, (3) Jianqing Fan with root probability 0.086, and (4) James

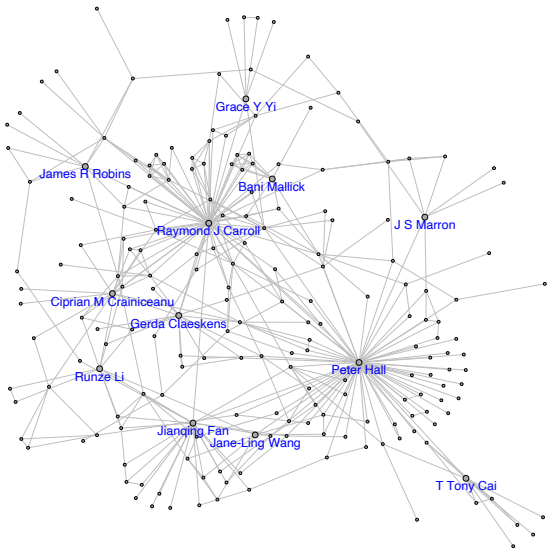


Figure 20: Subgraph of the co-authorship graph comprising the 200 nodes with the highest posterior root probabilities. We label the 12 nodes with the highest root probabilities.

Robins with root probability 0.048. The root probability ranking align closely with betweenness centrality ranking, in which Raymond Carroll, Peter Hall, and Jianqing Fan are also the top 3 most central nodes; see Table 2 of Ji and Jin (2016). Both the root probability ranking and the betweenness ranking differ significantly from degree ranking. We also display the subgraph of the 200 nodes with the highest posterior root probabilities in Figure 20 where we labeled the top 12 nodes with the highest root probabilities.

Community detection with random K roots model: Using our inference algorithm and the greedy matching procedure in 6.4, we compute clusters $\{Q_1, \ldots, Q_{K_{\text{all}}}\}$ where we find about $K_{\text{all}} \approx 40$ significant clusters. We order the clusters by their posterior frequencies and display the top 9 clusters in Figure 21, along with labels that we curated; we display the nodes in the cluster as word clouds in which the word size is proportional to the posterior root probabilities. We display 18 additional clusters in Section S5.3 of the Appendix. We note that the clusters can overlap since they are constructed from a sequence of posterior samples by matching; see the first paragraph of Section 6.4.

Ji and Jin (2016) on the same network uses scree plot to conclude that there are $K = 3$ clusters, which are shown in Figures 9, 10, and 11 in their paper. They refer to the three clusters as a "high-dimensional" supercluster, a "biostatistics" cluster, and a "Bayes" cluster. We find a giant supercluster, but we also find a large number of smaller clusters which accurately reflect actual research communities in statistics. For example, we find the same "Bayes" cluster in Ji and Jin (2016) (see Figure 21b), but we also discover other Bayesian clusters such as ones shown in Figure 21c. Similarly, we find the "biostat" community in Ji and Jin (2016) (see Figure 21f) but we find other biostat clusters as well such as the one shown in Figure 21h and the one centered on Jason Fine and Michael Korsorok in Figure 27 in the Appendix. In addition, we find many other meaningful communities, such as the experimental design community or the high-dimensional statistics community shown in Figure 22, or the survey and theory community in Figure 27 in the Appendix. We believe that PAPER model gives highly coherent clusters for this network because

(a) Central super-cluster (b) Bayesian (c) Bayesian

(d) Theory (e) Multivariate Analysis (f) Biostat

(g) Computation/UK (h) Biostat (i) Graphical Models

Figure 21: Nine of the clusters that most frequently appear in the posterior samples. Word sizes are proportional to the posterior root probability with respect to the cluster.

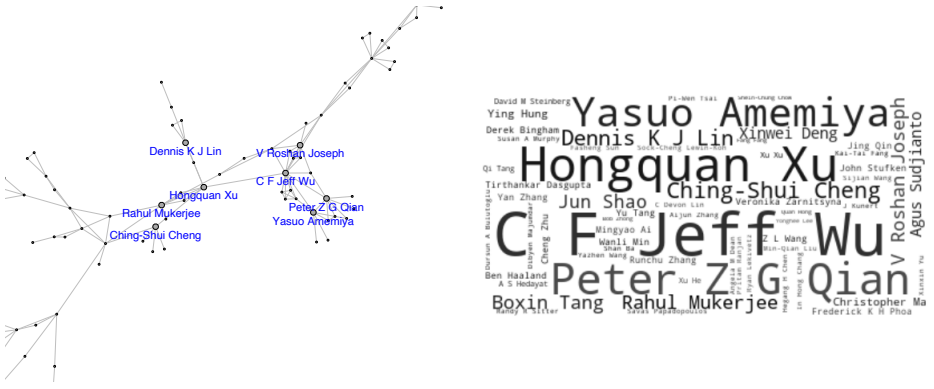
the network itself is locally tree-like, as shown in two cluster subgraphs that we display in Figure 22.

7 Discussion

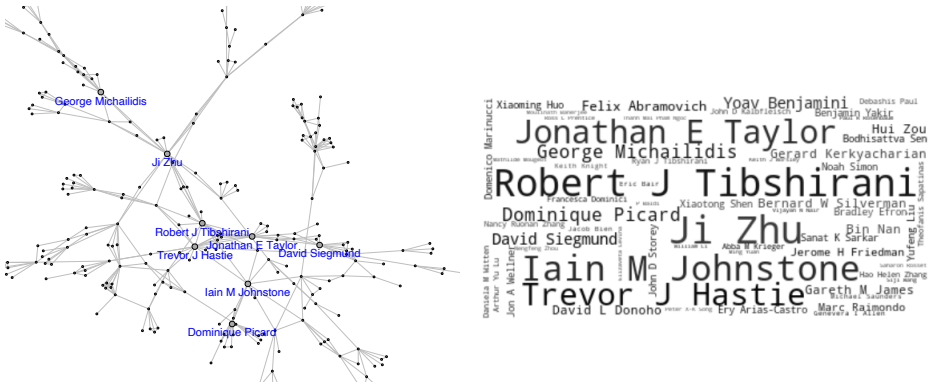
In this paper, we present the PAPER model for networks with underlying formation processes and formalize the problem of root inference. We extend the PAPER model to the setting of multiple roots to reflect the growth of multiple communities. There are a number of important open questions from modeling, theoretical, and algorithmic perspectives.

From a modeling perspective, an interesting direction is to suppose that the graph start not as singleton nodes but as a small subgraph. The goal then is to infer the seed-graph instead of the root node (c.f. Devroye and Reddad (2018)). Model extensions such as the PAPER-SBM mixture described in Remark 5 are also interesting; in these models, a subtle question is to what extend we have to estimate the parameters of the noise model well in order to recover the root nodes of the latent forest.

There are many open theoretical questions related to PAPER model and root inference. For instance, in Conjecture 13, we hypothesize that the size of the optimal confidence set for the root node is of a constant order if so long as the noise level is below a certain threshold. If the noise level is above the threshold, then every confidence set has size that diverges with n. The lower bound of this conjecture seems especially difficult and may require new techniques. Another interesting



(a) Experimental design community



(b) High-dimensional statistics community

Figure 22: Two additional clusters along with the subgraphs that correspond to the clusters. In the subgraph, we label the 8 nodes with the highest posterior root probability with respect to that cluster. We observe that the subgraphs are tree-like.

theoretical question is the analysis of community recovery using the PAPER model with multiple roots. Intuitively, we expect be able to correctly cluster the early nodes since they tend to have more central positions in the final graph. The late arriving nodes on the other hand would be more peripheral and difficult to cluster.

Algorithmically, we observe that the Gibbs sampler that we derived in Section 4 converges very quickly in practice (see Section S5.4). It would be interesting to study its mixing time, especially how the mixing time depends on the noise level.

Acknowledgement

This work is supported by the U.S. National Science Foundation DMS grant #2113671. The second author is grateful to Justin Khim for insightful discussions in the early stage of the work and to Rong Chen for helpful feedback and comments. The authors would like to thank anonymous referees for insightful comments which helped improve the paper.

Data Availability: The flu transmission network is extracted from Figure 3 in Hens et al. (2012). The MathSciNet network is available at `http://networkrepository.com/ca-MathSciNet.php`.

The University of Notre Dame website network is available at

https://snap.stanford.edu/data/web-NotreDame.html. The karate club network is available at http://www-personal.umich.edu/ mejn/netdata/. The political blog network is available at http://www-personal.umich.edu/ mejn/netdata/.

The airport network is available at http://seeslab.info/downloads/air-transportation-networks/. The statistician co-authorship network is available at http://zke.fas.harvard.edu/MADStat.html.

References

Abbe, E. (2017). Community detection and stochastic block models: recent developments, *The Journal of Machine Learning Research* **18**(1): 6446–6531.

Adamic, L. A. and Glance, N. (2005). The political blogosphere and the 2004 us election: divided they blog, *Proceedings of the 3rd international workshop on Link discovery*, pp. 36–43.

Addario-Berry, L. and Eslava, L. (2018). High degrees in random recursive trees, *Random Structures & Algorithms* **52**(4): 560–575.

Aiello, W., Chung, F. and Lu, L. (2000). A random graph model for massive graphs, *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pp. 171–180.

Albert, R., Jeong, H. and Barabási, A.-L. (1999). Diameter of the world-wide web, *Nature* **401**(6749): 130–131.

Aldous, D. J. (1990). The random walk construction of uniform spanning trees and uniform labelled trees, *SIAM Journal on Discrete Mathematics* **3**(4): 450–465.

Amini, A. A., Chen, A., Bickel, P. J. and Levina, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks, *Ann. Statist.* **41**(4): 2097–2122.
URL: *https://doi.org/10.1214/13-AOS1138*

Athreya, A., Fishkind, D. E., Tang, M., Priebe, C. E., Park, Y., Vogelstein, J. T., Levin, K., Lyzinski, V. and Qin, Y. (2017). Statistical inference on random dot product graphs: a survey, *The Journal of Machine Learning Research* **18**(1): 8393–8484.

Banerjee, S. and Bhamidi, S. (2020). Root finding algorithms and persistence of jordan centrality in growing random trees, *arXiv preprint arXiv:2006.15609* .

Banerjee, S. and Huang, X. (2021). Degree centrality and root finding in growing random networks, *arXiv preprint arXiv:2105.14087* .

Barabási, A.-L. (2016). *Network science*, Cambridge university press.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks, *Science* **286**(5439): 509–512.

Bhamidi, S. (2007). Universal techniques to analyze preferential attachment trees: Global and local analysis.

Bloem-Reddy, B., Foster, A., Mathieu, E. and Teh, Y. W. (2018). Sampling and inference for beta neutral-to-the-left models of sparse networks, *Proceedings of UAI 2018* .

Bloem-Reddy, B. and Orbanz, P. (2018). Random-walk models of network formation and sequential monte carlo methods for graphs, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **80**(5): 871–898.

Bollobás, B., Riordan, O., Spencer, J. and Tusnády, G. (2001). The degree sequence of a scale-free random graph process, *Random Structures & Algorithms* **18**(3): 279–290.

Briend, S., Calvillo, F. and Lugosi, G. (2022). Archaeology of random recursive dags and cooper-frieze random networks, *arXiv preprint arXiv:2207.14601* .

Broder, A. (1989). Generating random spanning trees, *30th Annual Symposium on Foundations of Computer Science* .

Bubeck, S., Devroye, L. and Lugosi, G. (2017). Finding Adam in random growing trees, *Random Structures & Algorithms* **50**(2): 158–172.

Bubeck, S., Eldan, R., Mossel, E. and Rácz, M. Z. (2017). From trees to seeds: on the inference of the seed from large tree in the uniform attachment model, *Bernoulli* **23**(4A): 2887–2916.

Bubeck, S., Mossel, E. and Rácz, M. Z. (2015). On the influence of the seed graph in the preferential attachment model, *IEEE Transactions on Network Science and Engineering* **2**(1): 30–39.

Callaway, D. S., Newman, M. E., Strogatz, S. H. and Watts, D. J. (2000). Network robustness and fragility: Percolation on random graphs, *Physical review letters* **85**(25): 5468.

Cantwell, G. T., St-Onge, G. and Young, J.-G. (2019). Recovering the past states of growing trees, *arXiv preprint arXiv:1910.04788* .

Cantwell, G. T., St-Onge, G. and Young, J.-G. (2021). Inference, model selection, and the combinatorics of growing trees, *Physical Review Letters* **126**(3): 038301.

Crane, H. (2016). The ubiquitous Ewens sampling formula, *Statistical Science* **31**(1): 1–39.

Crane, H. and Xu, M. (2021). Inference on the history of a randomly growing tree, *Journal of Royal Statistical Society, series B* (to appear.).

Curien, N., Duquesne, T., Kortchemski, I. and Manolescu, I. (2015). Scaling limits and influence of the seed graph in preferential attachment trees, *Journal de l'École polytechnique—Mathématiques* **2**: 1–34.

Dereich, S. and Mörters, P. (2009). Random networks with sublinear preferential attachment: degree evolutions, *Electronic Journal of Probability* **14**: 1222–1267.

Devroye, L. and Reddad, T. (2018). On the discovery of the seed in uniform attachment trees, *arXiv preprint arXiv:1810.00969* .

Diaconis, P. and Janson, S. (2007). Graph limits and exchangeable random graphs, *arXiv preprint arXiv:0712.2749* .

Drmota, M. (2009). *Random trees: an interplay between combinatorics and probability*, Springer Science & Business Media.

Fioriti, V., Chinnici, M. and Palomo, J. (2014). Predicting the sources of an outbreak with a spectral technique, *Applied Mathematical Sciences* **8**: 6775–6782.

Galashin, P. (2013). Existence of a persistent hub in the convex preferential attachment model, *arXiv preprint arXiv:1310.7513* .

Gao, C., Lu, Y. and Zhou, H. H. (2015). Rate-optimal graphon estimation, *The Annals of Statistics* **43**(6): 2624–2652.

Guimera, R., Mossa, S., Turtschi, A. and Amaral, L. N. (2005). The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles, *Proceedings of the National Academy of Sciences* **102**(22): 7794–7799.

Hens, N., Calatyud, L., Kurkela, S., Tamme, T. and Wallinga, J. (2012). Robust reconstruction and analysis of outbreak data: influenza a(h1n1)v transmission in a school-based population, *American Journal of Epidemiology* **176**(3): 196–203.

Hoff, P. D., Raftery, A. E. and Handcock, M. S. (2002). Latent space approaches to social network analysis, *Journal of the american Statistical association* **97**(460): 1090–1098.

Jeong, H., Mason, S. P., Barabási, A.-L. and Oltvai, Z. N. (2001). Lethality and centrality in protein networks, *Nature* **411**(6833): 41.

Ji, P. and Jin, J. (2016). Coauthorship and citation networks for statisticians, *The Annals of Applied Statistics* **10**(4): 1779–1812.

Jiang, J., Wen, S., Yu, S., Xiang, Y. and Zhou, W. (2016). Identifying propagation sources in networks: State-of-the-art and comparative studies, *IEEE Communications Surveys & Tutorials* **19**(1): 465–481.

Jin, J. (2015). Fast community detection by SCORE, *The Annals of Statistics* **43**(1): 57–89.

Karrer, B. and Newman, M. E. (2011). Stochastic blockmodels and community structure in networks, *Physical review E* **83**(1): 016107.

Khim, J. and Loh, P.-L. (2017). Confidence sets for the source of a diffusion in regular trees, *IEEE Transactions on Network Science and Engineering* **4**(1): 27–40.

Knuth, D. E. (1997). *The Art of Computer Programming: Volume 1: Fundamental Algorithms*, Addison-Wesley Professional.

Kolaczyk, E. D. (2009). *Statistical Analysis of Network Data: Methods and Models*, Springer Series in Statistics.

Leskovec, J., Lang, K. J., Dasgupta, A. and Mahoney, M. W. (2009). Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters, *Internet Mathematics* **6**(1): 29–123.

Lugosi, G., Pereira, A. S. et al. (2019). Finding the seed of uniform attachment trees, *Electronic Journal of Probability* **24**.

Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P. and Bhattacharjee, B. (2007). Measurement and Analysis of Online Social Networks, *Proceedings of the 5th ACM/Usenix Internet Measurement Conference (IMC'07)*, San Diego, CA.

Na, H. S. and Rapoport, A. (1970). Distribution of nodes of a tree by degree, *Mathematical Biosciences* **6**: 313–329.

Peköz, E. A., Röllin, A. and Ross, N. (2014). Joint degree distributions of preferential attachment random graphs, *arXiv preprint arXiv:1402.4686* .

Pollard, D. (2002). *A user's guide to measure theoretic probability*, number 8, Cambridge University Press.

Rossi, R. A. and Ahmed, N. K. (2015). The network data repository with interactive graph analytics and visualization, *AAAI*.
URL: *http://networkrepository.com*

Schervish, M. J. (1995). *Theory of statistics*, Springer Series in Statistics.

Shah, D. and Zaman, T. (2011). Rumors in a network: Who's the culprit?, *IEEE Transactions on information theory* **57**(8): 5163–5181.

Shelke, S. and Attar, V. (2019). Source detection of rumor in social network–a review, *Online Social Networks and Media* **9**: 30–42.

Sheridan, P., Yagahara, Y. and Shimodaira, H. (2008). A preferential attachment model with poisson growth for scale-free networks, *Annals of the institute of statistical mathematics* **60**(4): 747–761.

Sheridan, P., Yagahara, Y. and Shimodaira, H. (2012). Measuring preferential attachment in growing networks with missing-timelines using markov chain monte carlo, *Physica A: Statistical Mechanics and its Applications* **391**(20): 5031–5040.

Sreedharan, J. K., Magner, A., Grama, A. and Szpankowski, W. (2019). Inferring temporal information from a snapshot of a dynamic network, *Scientific reports* **9**(1): 1–10.

Van Der Hofstad, R. (2016). *Random graphs and complex networks*, Vol. 1, Cambridge university press.

Wade, S. and Ghahramani, Z. (2018). Bayesian cluster analysis: Point estimation and credible balls (with discussion), *Bayesian Analysis* **13**(2): 559–626.

West, M. (1992). *Hyperparameter estimation in Dirichlet process mixture models*, Duke University ISDS Discussion Paper# 92-A03.

Wilson, D. B. (1996). Generating random spanning trees more quickly than the cover time, *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 296–303.

Xie, F. and Xu, Y. (2019). Optimal bayesian estimation for random dot product graphs, *arXiv preprint arXiv:1904.12070* .

Xu, M., Jog, V. and Loh, P.-L. (2018). Optimal rates for community estimation in the weighted stochastic block model, *Annals of Statistics, Accepted, to appear* .

Young, J.-G., St-Onge, G., Laurence, E., Murphy, C., Hébert-Dufresne, L. and Desrosiers, P. (2019). Phase transition in the recoverability of network history, *Physical Review X* **9**(4): 041056.

Zachary, W. W. (1977). An information flow model for conflict and fission in small groups, *Journal of anthropological research* **33**(4): 452–473.

List of Figures

Supplementary material to "Root and community inference on latent network growth processes using noisy attachment models"

Harry Crane and Min Xu

S1 Supplement for Section 2

S1.1 Model Likelihood

We first give the likelihood of any time labeled tree under the APA(α, β) model. Define, for any integer $k \geq 1$,

$$\psi_{\alpha,\beta}(k) := \begin{cases} \prod_{j=1}^{k-1}(\beta j + \alpha) & \text{if } k \geq 2, \\ 1 & \text{if } k = 1. \end{cases}$$

Proposition S1. *Let $\boldsymbol{T}_n \sim APA(\alpha, \beta)$. Then, for any time labeled tree $\boldsymbol{t}_n$, we have that*

$$\mathbb{P}(\boldsymbol{T}_n = \boldsymbol{t}_n) = L_{\alpha,\beta}(\boldsymbol{t}_n) := \frac{\prod_{v\in[n]} \psi_{\alpha,\beta}(D_{\boldsymbol{t}_n}(v))}{\prod_{t=3}^{n}(2(t-2)\beta + (t-1)\alpha)}. \tag{S1.1}$$

The fact that the likelihood depends on the tree $\boldsymbol{t}_n$ only through its degree distribution $D_{\boldsymbol{t}_n}(\cdot)$ remains true in the multiple roots setting except that the likelihood also depends on the root nodes. One complication with the multiple roots setting is that we give each root node an imaginary self-loop. To deal with this, we first define $\psi^r_{\alpha,\beta}(k) := \prod_{j=2}^{k+1}(\beta j + \alpha)$.

Proposition S2. *Let $\boldsymbol{F}_n \sim APA(\alpha, \beta, K)$. Then, for any time labeled forest $\boldsymbol{f}_n$, we have that*

$$\mathbb{P}(\boldsymbol{F}_n = \boldsymbol{f}_n) = L_{\alpha,\beta,K}(\boldsymbol{f}_n) := \frac{\prod_{v\in\pi_{1:K}} \psi^r_{\alpha,\beta}(D_{\boldsymbol{f}_n}(v)) \prod_{v\notin\pi_{1:K}} \psi_{\alpha,\beta}(D_{\boldsymbol{f}_n}(v))}{\prod_{t=K+1}^{n}(2(t-1)\beta + (t-1)\alpha)}. \tag{S1.2}$$

In the random K setting, the likelihood is very similar except that the set of root nodes is not necessarily $\pi_{1:K}$.

Proposition S3. *Let $\boldsymbol{F}_n \sim APA(\alpha, \beta, \alpha_0)$. Then, for any time labeled forest $\boldsymbol{f}_n$ with K component trees, we have that*

$$\mathbb{P}(\boldsymbol{F}_n = \boldsymbol{f}_n) = L_{\alpha,\beta,S}(\boldsymbol{f}_n) := \frac{\prod_{v\in S} \psi^r_{\alpha,\beta}(D_{\boldsymbol{f}_n}(v)) \prod_{v\notin S} \psi_{\alpha,\beta}(D_{\boldsymbol{f}_n}(v))}{\prod_{t=K+1}^{n}(2(t-1)\beta + (t-1)\alpha)}. \tag{S1.3}$$

where S is the set of root nodes of $\boldsymbol{f}_n$, that is, a node is in S if and only if it has the earliest arrival time in its component tree.

Under the PAPER model, the complete data likelihood is also simple owing to the fact that any non-forest edge of the random graph $\boldsymbol{G}_n$ is Erdős–Rényi and any forest with K component trees has exactly $n - K$ edges. Therefore, for a time labeled graph $\boldsymbol{g}_n$ with m edges and a time labeled sub-forest $\boldsymbol{f}_n$, we have that, under the PAPER model and conditional on $\boldsymbol{G}_n$ having m edges,

$$\mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}_n, \boldsymbol{F}_n = \boldsymbol{f}_n) = \binom{n(n-1)/2 - (n-K)}{m - (n-K)}^{-1} \mathbb{P}(\boldsymbol{F}_n = \boldsymbol{f}_n).$$

We do not observe the forest of course. This is one of the main hurdles that we address in Section 4.

S1.2 Poisson attachment approximation

Proposition S4. *Let $t \in \mathbb{N}$, and let $q_1, \ldots, q_t \in [0,1]$ satisfy $\sum_{j=1}^t q_j \leq 1$, and let $\theta > 0$ such that $\theta q_j \leq 1$ for all $i \in [t]$. Let X and Y denote two **random subsets** of $[t]$, where X is generated by adding each $j \in [t]$ independently with probability θq_j, and where Y is generated by first drawing $M \sim$ Poisson(θ) and then repeating M times the procedure where we randomly choose $j \in [t]$ with probability q_j and with replacement and add j to Y. Let $P^{(X)}$ and $P^{(Y)}$ denote the distribution of X and Y respectively. Then, we have that*

$$d_{TV}(P^{(X)}, P^{(Y)}) \leq \theta^2 \max_{j \in [t]} q_j.$$

Proof. For $j = 1, \ldots, n$, define $X_j = \mathbb{1}\{j \in X\}$ and Y_j as the number of copies of element j in Y. Direct calculation then shows that $X_1, \ldots, X_t$ are independent where $X_j \sim \text{Ber}(\theta q_j)$ and that $Y_1, \ldots, Y_t$ are independent where $Y_j \sim \text{Poisson}(\theta q_j)$.

Therefore, by a coupling argument (see e.g. Example 2 in Chapter 10.1 of Pollard (2002)) and the fact that $\sum_{j=1}^t q_j \leq 1$, we have

$$\begin{aligned} d_{\text{TV}}(P^{(X)}, P^{(Y)}) &\leq \mathbb{P}(X \neq Y) = \sum_{j=1}^t \mathbb{P}(X_j \neq Y_j) \\ &\leq \sum_{j=1}^t \theta^2 q_j^2 \leq \theta^2 \max_j q_j, \end{aligned}$$

as desired. □

If we apply Proposition S4 to the independent Bernoulli noise model described in Section 2.3, where X and Y denote the random set of edges added under the Bernoulli noise model and the Poisson noise model respectively and where $q_j = \frac{\tilde{\beta} D_{\boldsymbol{T}_{t-1}}(j) + \tilde{\alpha}}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}}$, then we may use the fact that $\max_{j \in [t]} q_j = O_p(\frac{1}{\sqrt{t}})$ (see e.g. Section 8.7 in Van Der Hofstad (2016)) to see that the two noise models are approximately equivalent for large t.

S2 Supplement for Section 3

Recall that for an alphabetically labeled tree $\tilde{\boldsymbol{t}}_n$, we define the hist$(\tilde{\boldsymbol{t}}_n)$ as the set of all label ordering $\pi \in \text{Bi}([n], \mathcal{U}_n)$ such that $\pi^{-1}\tilde{\boldsymbol{t}}_n$ is a time labeled tree that has a positive probability over the APA model (Definition 1). For a node u, we also define hist$(u, \tilde{\boldsymbol{t}}_n)$ as all $\pi \in \text{hist}(\tilde{\boldsymbol{t}}_n)$ such that $\pi_1 = u$ and $h(u, \tilde{\boldsymbol{t}}_n) = |\text{hist}(u, \tilde{\boldsymbol{t}}_n)|$. Shah and Zaman (2011) derives an $O(n)$ runtime algorithm that computes the whole collection $\{h(u, \tilde{\boldsymbol{t}}_n)\}_{u \in \mathcal{U}_n}$, which is shown as Algorithm 4.

S2.1 Equivalence to maximum likelihood

Before deriving the likelihood formally, it is useful to have the following standard definitions. For two labeled graphs $\boldsymbol{g}, \boldsymbol{g}'$, we say that $\boldsymbol{g} \sim \boldsymbol{g}'$ if there exists $\rho \in \text{Bi}(V(\boldsymbol{g}), V(\boldsymbol{g}'))$ such that $\rho \boldsymbol{g} = \boldsymbol{g}'$. In this case, we say that $\boldsymbol{g}$ and $\boldsymbol{g}'$ are isomorphic, or that they have the same shape, or that they are equivalent as unlabeled graphs. The $\sim$ relationship defines equivalence classes on the set of all labeled graphs, which we refer to as the *unlabeled shape* or just *shape* for short. We write

$$I(\boldsymbol{g}, \boldsymbol{g}') := \{\rho \in \text{Bi}(V(\boldsymbol{g}), V(\boldsymbol{g}')) \,:\, \rho \boldsymbol{g} = \boldsymbol{g}'\}.$$

Algorithm 4 Computing $\{h(u, \tilde{\mathbf{t}}_n)\}_{u \in \mathcal{U}_n}$ (Shah and Zaman; 2011)

Input: a labeled tree $\tilde{\mathbf{t}}_n$.
Output: $h(u, \tilde{\mathbf{t}}_n)$ for all nodes $u \in \mathcal{U}_n$.

Arbitrarily select root $u_0 \in \mathcal{U}_n$.
for $u \in \mathcal{U}_n$ **do**
  Compute and store $n_u^{(u_0)} := |\tilde{\mathbf{t}}_u^{(u_0)}|$.
end for
Compute $h(u_0, \tilde{\mathbf{t}}_n) = n! \prod_{u \in \mathcal{U}_n} \frac{1}{|\tilde{\mathbf{t}}_u^{(u_0)}|}$.
Set $\mathcal{S} = \{\text{Children}(u_0)\}$.
while $\mathcal{S}$ is not empty **do**
  Remove an arbitrary node $u \in \mathcal{S}$.
  Compute $h(u, \tilde{\mathbf{t}}_n) = h(\text{pa}(u), \tilde{\mathbf{t}}_n) \frac{n_u^{(u_0)}}{n - n_u^{(u_0)}}$
  Add Children(u) to $\mathcal{S}$
end while

Note that $I(\boldsymbol{g}, \boldsymbol{g})$ is the set of automorphisms of the graph $\boldsymbol{g}$. To represent an unlabeled shape, we write $\text{sh}(\boldsymbol{g})$ where $\boldsymbol{g}$ an arbitrary representative element in the equivalence class.

Similarly, given a node $u \in V(\boldsymbol{g})$ and $u' \in V(\boldsymbol{g}')$, we say that $(\boldsymbol{g}, u) \sim_0 (\boldsymbol{g}', u')$ if there exists $\rho \in \text{Bi}(V(\boldsymbol{g}), V(\boldsymbol{g}'))$ such that $\rho \boldsymbol{g} = \boldsymbol{g}'$ and $\rho(u) = u'$. In this case, we say that $(\boldsymbol{g}, u)$ and $(\boldsymbol{g}', u')$ have the same *rooted shape*. The $\sim_0$ relationship defines an equivalence class on the pairs $(\boldsymbol{g}, u)$. We write

$$I(\boldsymbol{g}, u, \boldsymbol{g}', u') := \{\rho \in \text{Bi}(V(\boldsymbol{g}), V(\boldsymbol{g}')) \,:\, \rho \boldsymbol{g} = \boldsymbol{g}', \rho(u) = u'\}.$$

We have the following facts:

1. $I(\boldsymbol{g}, \boldsymbol{g}')$ is non-empty if and only if $\boldsymbol{g}, \boldsymbol{g}'$ have the same shape. Moreover, the cardinality of $I(\boldsymbol{g}, \boldsymbol{g}')$ depends only on that shape. For instance, $|I(\boldsymbol{g}, \boldsymbol{g}')| = |I(\boldsymbol{g}, \boldsymbol{g})|$ if the former is non-zero. In discrete mathematics, this cardinality is referred to as the size of the automorphism group of $\boldsymbol{g}$.

2. $I(\boldsymbol{g}, u, \boldsymbol{g}', u')$ is non-empty if and only if $(\boldsymbol{g}, u), (\boldsymbol{g}', u')$ have the same shape. Moreover, the cardinality of $I(\boldsymbol{g}, u, \boldsymbol{g}', u')$ depends only on that shape.

Now, for a labeled graph $\boldsymbol{g}$ and a node $u \in V(\boldsymbol{g})$, we define

$$\text{Eq}(u, \boldsymbol{g}) = \{u' \in \boldsymbol{g} \,:\, (\boldsymbol{g}, u) \sim_0 (\boldsymbol{g}, u')\}.$$

Nodes in $\text{Eq}(u, \boldsymbol{g})$ are indistinguishable from node u once the node labels are removed.

On observing an unlabeled graph $\text{sh}(\tilde{\boldsymbol{g}}_n)$, the likelihood of a node u being the root node is therefore

$$\mathcal{L}(u, \tilde{\boldsymbol{g}}_n) := \frac{1}{|\text{Eq}(u, \tilde{\boldsymbol{g}}_n)|} \sum_{\boldsymbol{g}_n \text{ time labeled}} \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}_n) \mathbb{1}\{(\boldsymbol{g}_n, 1) \sim_0 (\tilde{\boldsymbol{g}}_n, u)\},$$

where $\boldsymbol{G}_n$ has the PAPER(α, β, θ) distribution. It is straightforward to check that $\mathcal{L}(u, \tilde{\boldsymbol{g}}_n)$ depends only on the unlabeled shape of $(\tilde{\boldsymbol{g}}_n, u)$. We give a concrete example of the likelihood in Figure 23.

Theorem S5. *For any alphabetically labeled graph $\tilde{\boldsymbol{g}}_n$, we have*

$$\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) = \frac{\mathcal{L}(u, \tilde{\boldsymbol{g}}_n)}{\sum_{v \in \mathcal{U}_n} \mathcal{L}(v, \tilde{\boldsymbol{g}}_n)}.$$

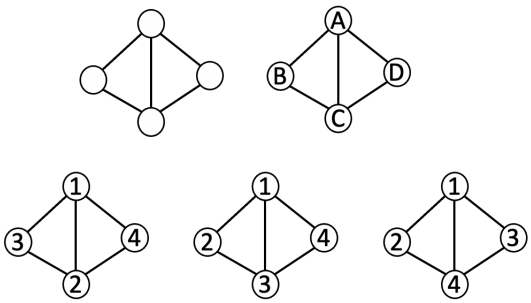


Figure 23: Viewing the top right graph as $\tilde{\boldsymbol{g}}$ and the bottom graphs as $\boldsymbol{g}^1, \boldsymbol{g}^2, \boldsymbol{g}^3$, we have $\mathrm{Eq}(A, \tilde{\boldsymbol{g}}) = \{A, C\}$ and $\mathcal{L}(A, \tilde{\boldsymbol{g}}_n) = \frac{1}{2}\big\{\mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}^1) + \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}^2) + \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}^3)\big\}$.

Proof. We have that

$$\begin{aligned}
\mathbb{P}(\Pi_1 = u \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) &\propto \sum_{\pi \in \mathrm{Bi}([n], \mathcal{U}_n),\, \pi_1 = u} \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi) \frac{1}{n!} \\
&\propto \sum_{\pi \in \mathrm{Bi}([n], \mathcal{U}_n),\, \pi_1 = u} \mathbb{P}(\boldsymbol{G}_n = \pi^{-1} \tilde{\boldsymbol{g}}_n) \\
&= \sum_{\boldsymbol{g}_n \text{ time labeled}} \sum_{\substack{\pi \in \mathrm{Bi}([n], \mathcal{U}_n), \\ \pi_1 = u, \pi \boldsymbol{g}_n = \tilde{\boldsymbol{g}}_n}} \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}_n) \\
&= \sum_{\boldsymbol{g}_n \text{ time labeled}} |I(\boldsymbol{g}_n, 1, \tilde{\boldsymbol{g}}_n, u)| \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}_n) \\
&= \frac{|I(\tilde{\boldsymbol{g}}_n, \tilde{\boldsymbol{g}}_n)|}{|\mathrm{Eq}(u, \tilde{\boldsymbol{g}}_n)|} \sum_{\boldsymbol{g}_n \text{ time labeled}} \mathbb{P}(\boldsymbol{G}_n = \boldsymbol{g}_n) \mathbb{1}\{(\boldsymbol{g}_n, 1) \sim_0 (\tilde{\boldsymbol{g}}_n, u)\},
\end{aligned}$$

where the second equality follows by the definition of $I(\boldsymbol{g}, 1, \tilde{\boldsymbol{g}}_n, u)$ and the final equality follows by Lemma S6. The desired conclusion immediately follows.

□

Lemma S6. *For any labeled graphs $\boldsymbol{g}, \boldsymbol{g}'$ and nodes $u \in V(\boldsymbol{g})$, $u' \in V(\boldsymbol{g}')$, if $(\boldsymbol{g}, u) \sim_0 (\boldsymbol{g}', u')$, then*

$$|I(\boldsymbol{g}, \boldsymbol{g}')| = |I(\boldsymbol{g}, u, \boldsymbol{g}', u')||Eq(u, \boldsymbol{g})|.$$

Proof. Suppose $|I(\boldsymbol{g}, u, \boldsymbol{g}', u')| > 0$. We note for any node $v \in V(\boldsymbol{g})$, we have that $|I(\boldsymbol{g}, v, \boldsymbol{g}', u')|$ is either zero or equal to $|I(\boldsymbol{g}, u, \boldsymbol{g}', u')|$. Moreover, it is non-zero if and only if $v \in \mathrm{Eq}(u, \boldsymbol{g})$.

Therefore, using the fact that $I(\boldsymbol{g}, \boldsymbol{g}') = \cup_{v \in V(\boldsymbol{g})} I(\boldsymbol{g}, v, \boldsymbol{g}', u')$, we have

$$|I(\boldsymbol{g}, \boldsymbol{g}')| = \sum_{v \in V(\boldsymbol{g})} |I(\boldsymbol{g}, v, \boldsymbol{g}', u')| = |\mathrm{Eq}(u, \boldsymbol{g}_n)||I(\boldsymbol{g}, u, \boldsymbol{g}', u')|,$$

as desired. □

S3 Supplement for Section 4

S3.1 Parameter estimation for the PAPER model via EM

The PAPER models are parametrized by α, β which control the attachment mechanism, by θ which is the noise level, and by either K or α_0 in the multiple roots setting. We discuss some ways to

select the number of trees K in the fixed K root setting and ways to estimate α_0 in the random K roots setting in Section S3.4 of the appendix.

In this section therefore, we consider only the estimation of the parameters α and β. We assume that $\beta > 0$, in which case, without loss of generality, we may assume $\beta = 1$ so that we only need to estimate α. We note that assuming $\beta > 0$ does not exclude uniform attachment if we allow $\alpha = \infty$. We first consider the single root setting. For any tree $\tilde{\boldsymbol{t}}_n$, by Proposition S1 that

$$\begin{aligned}\mathbb{P}_\alpha(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) &= \sum_{\pi \in \mathrm{hist}(\tilde{\boldsymbol{t}}_n)} \mathbb{P}_\alpha(\boldsymbol{T}_n = \pi^{-1}\tilde{\boldsymbol{t}} \,|\, \Pi = \pi)\mathbb{P}(\Pi = \pi) \\ &= h(\tilde{\boldsymbol{t}}_n)\frac{\prod_{v\in\mathcal{U}_n}\prod_{j=1}^{D_{\tilde{\boldsymbol{t}}_n}(v)-1}(j+\alpha)}{\prod_{k=3}^{n}(2(k-2)+(k-1)\alpha)}\frac{1}{n!}. \end{aligned} \tag{S3.4}$$

Therefore, keeping only terms that depend on α, we have that the log-likelihood is

$$\begin{aligned}\ell(\alpha; \tilde{\boldsymbol{T}}_n) &= \sum_{v\in\mathcal{U}_n}\sum_{j=1}^{\infty}\log(j+\alpha)\mathbb{1}\{j < D_{\tilde{\boldsymbol{T}}_n}(v)\} - \sum_{k=3}^{n}\log\big(2(k-2)+(k-1)\alpha\big) \\ &= \sum_{j=1}^{\infty}\log(j+\alpha)W_{\tilde{\boldsymbol{T}}_n}(j) - \sum_{k=3}^{n}\log\big(2(k-2)+(k-1)\alpha\big),\end{aligned}$$

where we define $W_{\tilde{\boldsymbol{T}}_n}(j) := |\{v \in \mathcal{U}_n \,:\, D_{\tilde{\boldsymbol{T}}_n}(v) > j\}|$. We note that, in this case, the log-likelihood of α depends on the tree $\tilde{\boldsymbol{T}}_n$ only through its degree sequence.

In the PAPER model where $\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{T}}_n + \tilde{\boldsymbol{R}}_n$, for every node $v \in \mathcal{U}_n$, we have that $D_{\tilde{\boldsymbol{G}}_n}(v) = D_{\tilde{\boldsymbol{T}}_n}(v) + D_{\tilde{\boldsymbol{R}}_n}(v)$ where the tree degree $D_{\tilde{\boldsymbol{T}}_n}(v)$ is now latent. We propose an approximate EM algorithm in this setting.

The complete data log-likelihood in this case is

$$\ell(\alpha; D_{\tilde{\boldsymbol{G}}_n}, D_{\tilde{\boldsymbol{T}}_n}) = \sum_{j=1}^{\infty}\log(j+\alpha)\sum_{v}\mathbb{1}\{j < D_{\tilde{\boldsymbol{T}}_n}(v)\} - \sum_{k=s}^{n}\log\big(2(k-2)+(k-1)\alpha\big).$$

For a given value α', the EM update is then to maximize

$$\begin{aligned}M(\alpha|\alpha') &:= \mathbb{E}_{\alpha'}\left[\sum_{j=1}^{\infty}\log(j+\alpha)\sum_{v}\mathbb{1}\{j < D_{\tilde{\boldsymbol{T}}_n}(v)\}\,\middle|\,\tilde{\boldsymbol{G}}_n\right] - \sum_{k=s}^{n}\log\big(2(k-2)+(k-1)\alpha\big) \\ &= \sum_{j=1}^{\infty}\log(j+\alpha)\sum_{v}\mathbb{P}_{\alpha'}\left\{j < D_{\tilde{\boldsymbol{T}}_n}(v)\,\middle|\,\tilde{\boldsymbol{G}}_n\right\} - \sum_{k=s}^{n}\log\big(2(k-2)+(k-1)\alpha\big).\end{aligned} \tag{S3.5}$$

The conditional probability term $\mathbb{P}_{\alpha'}(j < D_{\tilde{\boldsymbol{T}}_n}(v) \,|\, \tilde{\boldsymbol{G}}_n)$ can be computed by Gibbs sampling, but we can significantly reduce the computation time by approximating $\mathbb{P}_{\alpha'}(j < D_{\tilde{\boldsymbol{T}}_n}(v) \,|\, \tilde{\boldsymbol{G}}_n)$ with $\mathbb{P}_{\alpha'}(j < D_{\tilde{\boldsymbol{T}}_n}(v) \,|\, D_{\tilde{\boldsymbol{G}}_n}(v))$, which ignores the mild dependence between the degrees of all the nodes. To further improve the quality of the approximation, we observe that

$$\sum_{j=1}^{\infty}\sum_{v}\mathbb{P}_{\alpha'}\big(j < D_{\tilde{\boldsymbol{T}}_n}(v)\,\big|\,\tilde{\boldsymbol{G}}_n\big) = \sum_{v}(D_{\tilde{\boldsymbol{T}}_n}(v) - 1) = n - 2$$

while the sums of the approximate conditional probabilities $\sum_{j=1}^{\infty}\sum_{v}\mathbb{P}_{\alpha'}(j < D_{\tilde{\boldsymbol{T}}_n}(v) \,|\, D_{\tilde{\boldsymbol{G}}_n}(v))$ may be different. Thus, we normalize $\mathbb{P}_{\alpha'}(j < D_{\tilde{\boldsymbol{T}}_n}(v) \,|\, D_{\tilde{\boldsymbol{G}}_n}(v))$ by defining $\tilde{W}_{\tilde{\boldsymbol{G}}_n}(j) = (n-2)\frac{\mathbb{P}_{\alpha'}(j<D_{\tilde{\boldsymbol{T}}_n}(v)\,|\,D_{\tilde{\boldsymbol{G}}_n}(v))}{\sum_{j=1}^{\infty}\mathbb{P}_{\alpha'}(j<D_{\tilde{\boldsymbol{T}}_n}(v)\,|\,D_{\tilde{\boldsymbol{G}}_n}(v))}$ so that $\sum_{j=1}^{\infty}\tilde{W}_{\tilde{\boldsymbol{G}}_n}(j) = n-2$ and, instead of maximizing (S3.5), we

update

$$\tilde{M}(\alpha|\alpha') := \sum_{j=1}^{\infty} \log(j+\alpha)\tilde{W}_{\tilde{\boldsymbol{G}}_n}(j) - \sum_{k=s}^{n} \log\big(2(k-2)+(k-1)\alpha\big). \tag{S3.6}$$

In practice, we find that the normalization significant improves the quality of the approximation.

To compute $\tilde{W}_{\tilde{\boldsymbol{G}}_n}$, we have by Bayes rule that for any $k \in [n]$ and $s \le k$,

$$\mathbb{P}_{\alpha'}(D_{\tilde{\boldsymbol{T}}_n}(v) = s \,|\, D_{\tilde{\boldsymbol{G}}_n}(v) = k) = \frac{\mathbb{P}_{\alpha'}(D_{\tilde{\boldsymbol{T}}_n}(v) = s, D_{\tilde{\boldsymbol{R}}_n}(v) = k-s)}{\sum_{t=1}^{k} \mathbb{P}_{\alpha'}(D_{\tilde{\boldsymbol{T}}_n}(v) = t, D_{\tilde{\boldsymbol{R}}_n}(v) = k-t)} \tag{S3.7}$$

$$= \frac{P_{\mathrm{Bin}(n-s,\theta)}(k-s)\mathbb{P}_{\alpha'}(D_{\tilde{\boldsymbol{T}}_n}(v) = s)}{\sum_{t=1}^{k} P_{\mathrm{Bin}(n-t,\theta)}(k-t)\mathbb{P}_{\alpha'}(D_{\tilde{\boldsymbol{T}}_n}(v) = t)}, \tag{S3.8}$$

where $P_{\mathrm{Bin}(n-s,\theta)}(\cdot)$ denotes the probability of a binomial distribution with $n-s$ trials and success probability θ. The exact distribution of the degree $D_{\tilde{\boldsymbol{T}}_n}(v)$ of a node v under the $\mathrm{APA}_{\alpha',1}$ is intractable but we can approximate it by its limiting distribution

$$P_{\alpha'}(s) := (2+\alpha')\frac{\Gamma(s+\alpha')\Gamma(3+2\alpha')}{\Gamma(s+3+2\alpha')\Gamma(1+\alpha')} = \frac{2+\alpha'}{3+2\alpha'}\prod_{j=1}^{s-1}\frac{j+\alpha'}{j+3+2\alpha'}.$$

By Van Der Hofstad (Theorem 8.2 2016), we have that, for any node v,

$$\sup_{s\in\mathbb{N}}\big|\mathbb{P}_{\alpha'}\big(D_{\tilde{\boldsymbol{T}}_n}(v) = s\big) - P_{\alpha'}(s)\big| \le C_{\alpha'}\sqrt{\frac{\log n}{n}}$$

with probability that tends to 1 as $n \to \infty$. Therefore, we may replace $\mathbb{P}_{\alpha'}\big(D_{\tilde{\boldsymbol{T}}_n}(v) = s\big)$ with $P_{\alpha'}(s)$ in (S3.8) to obtain a tractable approximation which is accurate in the limit.

To summarize, our estimation procedure generates a sequence α^j where α^j maximizes $\tilde{M}(\cdot\,|\,\alpha^{j-1})$ and where $\tilde{M}$ is computed using (S3.8). Although we approximate $M(\cdot\,|\,\cdot)$ by $\tilde{M}(\cdot\,|\,\cdot)$ and approximate the distribution of the random degree $D_{\tilde{\boldsymbol{T}}_n}(v)$ by its asymptotic limit, we find empirically that the resulting procedure always converges and performs well. We test the estimation procedure on simulated PAPER graphs of $n = 3,000$ nodes and $m = 15,000$ edges and report the estimation performance in Table 6. We find that the estimator is biased upwards when α is large, which is possibly because the likelihood (S3.4) is much less sensitive to a change in α when α is large than when α is small. In our simulation studies (Section 6.1), we show that the confidence sets constructed with the estimated parameters still attain their nominal coverage so that estimation error does not significantly impact the inference quality.

True α	0	1	3	6	∞ (UA)
Estimated α	0.03 (0.04)	1.04 (0.2)	3.3 (1.34)	10.7 (13.57)	85.4 (20.9)

Table 6: Mean and standard deviation of the estimated α computed on 200 independent trials on graphs with $n = 3,000$ nodes and $m = 15,000$ edges.

We use the same estimator in the fixed $K > 1$ setting and the variable K setting. In these cases, the log-likelihood is slightly different because the root nodes have imaginary self-loop edges. However, if the number of root nodes is small, the log-likelihood is virtually identical.

S3.2 Derivation of root sampling probability (20) for the fixed K and random K setting

Let $\tilde{\boldsymbol{f}}_n$ be an alphabetically labeled forest with component trees $\tilde{\boldsymbol{t}}^1, \ldots, \tilde{\boldsymbol{t}}^K$. For a specific tree $\tilde{\boldsymbol{t}}^k$ and a node $u \in V(\tilde{\boldsymbol{t}}^k) \subset \mathcal{U}_n$, we compute the probability, under the PAPER$(\alpha, \beta, K, \theta)$ model and label randomization, that u^k is the first node of $\tilde{\boldsymbol{t}}^k$ given $\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n$.

To formally derive this, denote the K random component trees of the random forest $\tilde{\boldsymbol{F}}_n$ by $\tilde{\boldsymbol{T}}^1, \ldots, \tilde{\boldsymbol{T}}^K$, and define Π^k as the random latent *relative* ordering of the nodes in the k-th random component tree $\tilde{\boldsymbol{T}}^k$. In other words, Π^k takes value in $\mathrm{Bi}([n^k], V(\tilde{\boldsymbol{T}}^k))$ (where $n^k = |V(\tilde{\boldsymbol{T}}^k)|$) and $\Pi^k_t = v$ implies that v is the t-th node, among the nodes of $\tilde{\boldsymbol{T}}^k$, to arrive in $\tilde{\boldsymbol{T}}^k$.

Then, we have that, for any $u \in V(\tilde{\boldsymbol{t}}^k)$,

$$\begin{aligned}
\mathbb{P}(\Pi^k_1 = u \,|\, \tilde{\boldsymbol{T}}^k = \tilde{\boldsymbol{t}}^k) &= \sum_{\pi^k \in \mathrm{hist}(u, \tilde{\boldsymbol{t}}^k)} \mathbb{P}(\Pi^k = \pi^k \,|\, \tilde{\boldsymbol{T}}^k = \tilde{\boldsymbol{t}}^k) \\
&\propto \sum_{\pi^k \in \mathrm{hist}(u, \tilde{\boldsymbol{t}}^k)} \mathbb{P}(\tilde{\boldsymbol{T}}^k = \tilde{\boldsymbol{t}}^k \,|\, \Pi^k = \pi^k) \\
&\propto h(u, \tilde{\boldsymbol{t}}^k) \prod_{j=2}^{D_{\tilde{\boldsymbol{t}}^k}(u)+1} (\beta j + \alpha) \prod_{v \neq u, v \in V(\tilde{\boldsymbol{t}}^k)} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}^k}(v)-1} (\beta j + \alpha) \\
&= h(u, \tilde{\boldsymbol{t}}^k)(\beta D_{\tilde{\boldsymbol{t}}^k}(u) + \beta + \alpha)(\beta D_{\tilde{\boldsymbol{t}}^k}(u) + \alpha) \prod_{v \in V(\tilde{\boldsymbol{t}}^k)} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}^k}(v)-1} (\beta j + \alpha) \\
&\propto h(u, \tilde{\boldsymbol{t}}^k)(\beta D_{\tilde{\boldsymbol{t}}^k}(u) + \beta + \alpha)(\beta D_{\tilde{\boldsymbol{t}}^k}(u) + \alpha),
\end{aligned}$$

where the third proportionality (equality up to multiplicative factor that is constant with respect to u) follows from Proposition S2. Formula (20) thus follows.

S3.3 Collapsed Gibbs sampler

We give an alternative Gibbs sampler in which we sample only a set of root nodes instead of sampling an entire history π. More precisely, we alternate between the following two stages:

(A) We fix the forest $\tilde{\boldsymbol{f}}$ and sample a set of root nodes $\tilde{s}$ with probability

$$\mathbb{P}(\tilde{S} = \tilde{s} \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}) \propto \mathbb{P}(\tilde{S} = \tilde{s} \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n), \tag{S3.9}$$

where $\tilde{s}$ comprise of a single node from each of the component trees of $\tilde{\boldsymbol{f}}_n$.

(B) We fix the root set $\tilde{s}$ and generate a new forest $\tilde{\boldsymbol{f}}_n$ by iteratively sampling a new parent for each of the nodes.

To sample the root set for the first stage of the Gibbs sampler, we write $\tilde{\boldsymbol{t}}^1, \ldots, \tilde{\boldsymbol{t}}^K$ as the K disjoint trees of the fixed forest $\tilde{\boldsymbol{f}}_n$. Then, to generate the root set $\tilde{s}$, we generate, for each tree $\tilde{\boldsymbol{t}}^k$, the root node u^k with probability (20).

For the second stage of the Gibbs sampler, we place the nodes in some arbitrary order and for each node u, we generate a parent $\tilde{u}$, which could be equal to the old parent, according to the distribution

$$\mathbb{P}\big\{pa(u) = \tilde{u} \,|\, \{pa(v)\}_{v \neq u}, \tilde{S} = \tilde{s}, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n\big\}. \tag{S3.10}$$

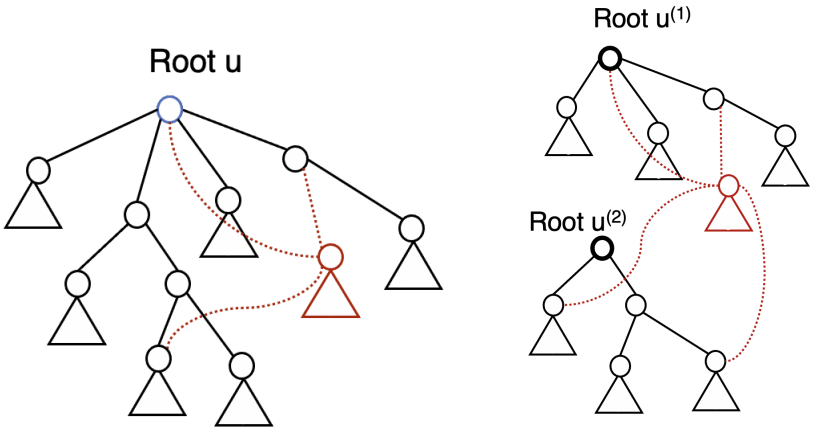


Figure 24: Selecting a new parent for a node. **Left:** the single root setting. **Right:** the multiple roots setting.

The action of generating a new parent is equivalent to replacing the edge between u and its old parent with a new one between u and $\tilde{u}$. Because we do not condition on the ordering Π, the new parent $\tilde{u}$ can be any node in the network connected to u that is not a descendant of u–that is, we only require that $\tilde{u}$ is not in the subtree $\tilde{\boldsymbol{t}}_u^{(\tilde{s})}$ of node u, where we view $\tilde{s}$ as the roots for the whole forest.

Another way to think of the second stage is that we take the subtree $\tilde{\boldsymbol{t}}_u^{(\tilde{s})}$ and *graft* it onto another part of the forest. In the multiple roots setting, a subtree may be transferred from one component tree to another. In the random K setting, two disjoint subtrees may be merged into a single tree or, a subtree may be split and forms a new component. See Figure 24 for a visual illustration.

In contrast with (24), we do not condition on Π and must therefore sum over all histories when computing (S3.10):

$$
\begin{aligned}
&\mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n \,|\, \tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n, \tilde{S} = \tilde{s}) \\
&\quad \propto \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n)\mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \tilde{S} = \tilde{s}) \\
&\quad = \binom{n(n-1)/2 - n + K}{m - n + k}^{-1} \sum_{\pi \in \mathrm{hist}(\tilde{s}, \tilde{\boldsymbol{f}}_n)} \mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi) \\
&\quad \propto \prod_{k=1}^{K} \frac{m - n + k}{n(n-1)/2 - n + k} \sum_{\pi \in \mathrm{hist}(\tilde{s}, \tilde{\boldsymbol{f}}_n)} \mathbb{P}(\boldsymbol{F}_n = \pi^{-1}\tilde{\boldsymbol{f}}_n \,|\, \Pi = \pi) \\
&\quad \propto \prod_{k=1}^{K} \frac{m - n + k}{n(n-1)/2 - n + k} h(\tilde{s}, \tilde{\boldsymbol{f}}_n) \begin{cases} \tilde{L}_{\alpha,\beta}(D_{\tilde{\boldsymbol{f}}_n}) & \text{if single root} \\ \tilde{L}_{\alpha,\beta,K}(\tilde{s}, D_{\tilde{\boldsymbol{f}}_n}) & \text{if fixed } K \text{ roots} \\ \tilde{L}_{\alpha,\beta,\alpha_0}(\tilde{s}, D_{\tilde{\boldsymbol{f}}_n}) & \text{if random } K \end{cases}
\end{aligned}
$$

where we have that

$$\tilde{L}_{\alpha,\beta}(D_{\tilde{\boldsymbol{f}}_n}) = \prod_{v} \prod_{j=1}^{D_{\tilde{\boldsymbol{f}}_n}(v)-1} \beta j + \alpha$$

$$\tilde{L}_{\alpha,\beta,K}(\tilde{s}, D_{\tilde{\boldsymbol{f}}_n}) = \prod_{v\in\tilde{s}} \prod_{j=2}^{D_{\tilde{\boldsymbol{f}}_n}(v)+1} (\beta j + \alpha) \prod_{v\notin\tilde{s}} \prod_{j=1}^{D_{\tilde{\boldsymbol{f}}_n}(v)-1} (\beta j + \alpha)$$

$$\tilde{L}_{\alpha,\beta,\alpha_0}(\tilde{s}, D_{\tilde{\boldsymbol{f}}_n}) = \alpha_0^K \prod_{v\in\tilde{s}} \prod_{j=2}^{D_{\tilde{\boldsymbol{f}}_n}(v)+1} (\beta j + \alpha) \prod_{v\notin\tilde{s}} \prod_{j=1}^{D_{\tilde{\boldsymbol{f}}_n}(v)-1} (\beta j + \alpha).$$

We may characterize the count of the history as follows:

$$h(\tilde{s}, \tilde{\boldsymbol{f}}_n) = \begin{cases} n! \prod_v \frac{1}{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|} & \text{if single root} \\ (n-K)! \prod_{v\notin\tilde{s}} \frac{1}{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|} & \text{if fixed } K \text{ roots} \\ (n-1)! \prod_v \frac{1}{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|} & \text{if random } K \text{ roots} \end{cases}$$

We summarize the resulting procedure in Algorithm 5 and 6. These are similar to Algorithm 2 and 3 except that we take into account how the choice of the graft affects the size of the history of the resulting forest.

Algorithm 5 Collapsed Gibbs sampler for fixed K or single root settings

Input: labeled forest $\tilde{\boldsymbol{f}}_n$, a set of K root nodes $\tilde{s}$.
Effect: Modifies $\tilde{\boldsymbol{f}}_n$ in place.

for each node $u \in \mathcal{U}_n$ **do**:
  if $u \in \tilde{s}$, continue.
  Remove the edge $(u, p(u))$ from $\tilde{\boldsymbol{f}}_n$.
  Generate a node $w \in N_{\tilde{\boldsymbol{g}}_n} \backslash V(\tilde{\boldsymbol{t}}_u^{(\tilde{s})})$ with probability proportional to

$$w \mapsto \prod_{v\in A_{\tilde{\boldsymbol{f}}_n}(w), v\notin\tilde{s}} \frac{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|}{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}| + |\tilde{\boldsymbol{t}}_u^{(\tilde{s})}|} (\beta D_{\tilde{\boldsymbol{f}}_n}(w) + \underbrace{2\beta \mathbb{1}\{w \in \tilde{s}\}}_{\text{only for } K>1} + \alpha),$$

  where $A_{\tilde{\boldsymbol{f}}_n}(w)$ is the set of ancestors (parent, parent of parent, etc) of w *including* w itself.
  Add edge (u, w) to $\tilde{\boldsymbol{f}}_n$.
end for

S3.4 Practical details on the Gibbs sampler

Convergence criterion: We use a simple convergence criterion where we run two chains simultaneously and keep track of the resulting posterior root distributions, which we denote Q^A and Q^B for the two chains. We continue the chain until the distance (we use Hellinger distance or total variation distance in all the experiments) between Q^A and Q^B is smaller than some threshold τ. We find that $\tau = 0.1$ suffices to generate accurate confidence sets for the root node in the single root setting. However, in the multiple roots setting, we require $\tau = 0.01$ or smaller. We observe in our experiments that the UA setting ($\alpha = 1, \beta = 0$) requires far more iterations to converge than the LPA model ($\alpha = 0, \beta = 1$). It is important to note that the chains A and B are initialized with uniformly random spanning trees and uniformly random orderings on those trees so that the initialization is guaranteed to be overdispersed.

Algorithm 6 Collapsed Gibbs Sampler for the random K setting

Input: labeled forest $\tilde{\boldsymbol{f}}_n$, a set of root nodes $\tilde{s}$.
Effect: Modifies $\tilde{\boldsymbol{f}}_n$ and $\tilde{s}$ in place.

for each node $u \in \mathcal{U}_n$ **do**:

If $u \in \tilde{s}$ and $|\tilde{s}| = 1$, continue.

If $u \in \tilde{s}$ and $|\tilde{s}| > 1$, set $\tilde{s} = \tilde{s} \backslash \{u\}$; else, remove the edge $(u, p(u))$ from $\tilde{\boldsymbol{f}}_n$.

Generate $w \in \{\emptyset\} \cup \big(N_{\tilde{\boldsymbol{g}}_n} \backslash V(\tilde{\boldsymbol{t}}_u^{(\tilde{s})})\big)$ with probability proportional to

$$\begin{cases} w \mapsto \prod_{v \in A_{\tilde{\boldsymbol{f}}_n}(w)} \frac{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}|}{|\tilde{\boldsymbol{t}}_v^{(\tilde{s})}| + |\tilde{\boldsymbol{t}}_u^{(\tilde{s})}|} (\beta D_{\tilde{\boldsymbol{f}}_n}(w) + 2\beta \mathbb{1}\{w \in \tilde{s}\} + \alpha), & \text{if } w \in N_{\tilde{\boldsymbol{g}}_n} \backslash V(\tilde{\boldsymbol{t}}_u^{(\tilde{s})}) \\ w \mapsto \alpha_0 \frac{m - n + |\tilde{s}|}{n(n-1)/2 - n + |\tilde{s}|} & \text{if } w = \emptyset, \end{cases}$$

where $A_{\tilde{\boldsymbol{f}}_n}(w)$ is the set of ancestors (parent, parent of parent, etc) of w *including* w itself.

If $w \in N_{\tilde{\boldsymbol{g}}_n} \backslash V(\tilde{\boldsymbol{t}}_u^{(\tilde{s})})$, add edge (u, w) to $\tilde{\boldsymbol{f}}_n$. Else, if $w = \emptyset$, let $\tilde{s} = \tilde{s} \cup \{w\}$.

end for

Estimating K in the fixed K roots setting: one way to select K is by maximum likelihood. For $K = 1, 2, 3, \ldots$, let $\tilde{\boldsymbol{G}}_n$ be distributed according to PAPER$(\alpha, \beta, K, \theta)$ and let

$$\begin{aligned} \mathcal{L}(K) &:= \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n) \\ &= \sum_{\tilde{\boldsymbol{f}}_n \in \mathcal{F}_K(\tilde{\boldsymbol{g}}_n), \pi} \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi) \mathbb{P}(\tilde{\boldsymbol{F}}_n = \tilde{\boldsymbol{f}}_n, \Pi = \pi) \\ &= \binom{n(n-1)/2 - (n-K)}{m - (n-K)}^{-1} \sum_{\tilde{\boldsymbol{f}}_n \in \mathcal{F}_K(\tilde{\boldsymbol{g}}_n), \pi} \mathbb{P}(\boldsymbol{F}_n = \pi^{-1} \tilde{\boldsymbol{f}}_n) \frac{1}{n!}. \end{aligned}$$

Using the Gibbs sampler, we would then evaluate $\mathcal{L}(K)$ for all $K \in [n]$. This however would be computationally intensive. We therefore recommend the random K model in settings where K is unknown and potentially large.

Estimating α_0 in the random K roots setting: We estimate α_0 by adding one more step in the Gibbs sampler where, after we generate a new forest and potentially a new K, we sample α_0 from the posterior distribution $\mathbb{P}(\alpha_0 \,|\, K)$. To that end, we use an Exponential(λ) prior on α_0 (we use $\lambda = 0.1$ yielding a variance of 100 in all experiments) and follow West (1992) to generate posterior samples from $\mathbb{P}(\alpha_0 \,|\, K)$. We find that the resulting estimate is insensitive to the choice of the hyperparameter λ and performs well in practice.

S3.5 Details for algorithm under sequential noise models

Let $\tilde{\boldsymbol{g}}_n$ be an alphabetically labeled graph and $\tilde{\boldsymbol{t}}_n$ be a spanning tree of $\tilde{\boldsymbol{g}}_n$. Write $\tilde{\boldsymbol{r}}_n = \tilde{\boldsymbol{g}}_n \backslash \tilde{\boldsymbol{t}}_n$ as the subgraph of $\tilde{\boldsymbol{g}}_n$ that comprises of the noise edges. We then have

$$\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n) = \prod_{j=3}^{n-1} \prod_{k=j+1}^{n} Q_{jk},$$

$$\text{where } Q_{jk} \equiv Q_{jk}(\pi, \tilde{\boldsymbol{t}}_n) = \left\{ \frac{\theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{k-1}}(\pi_j) + \tilde{\alpha})}{2(k-2)\tilde{\beta} + (k-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(\pi_j, \pi_k) \in \tilde{\boldsymbol{r}}_n\}}$$

$$\left\{ \frac{2(k-2)\tilde{\beta} + (k-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{k-1}}(\pi_j) + \tilde{\alpha})}{2(k-2)\tilde{\beta} + (k-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(\pi_j, \pi_k) \notin \tilde{\boldsymbol{g}}_n\}}.$$

In some cases, it is convenient to refer to nodes through alphabetical labels $\mathcal{U}_n$. Let $u, v \in \mathcal{U}_n$ be a pair of nodes and suppose $\pi_u^{-1} < \pi_v^{-1}$; we write

$$Q_{uv} \equiv Q_{uv}(\pi, \tilde{\boldsymbol{t}}_n) := \left\{ \frac{\theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{\pi_v^{-1}-1}}(u) + \tilde{\alpha})}{2(\pi_v^{-1}-2)\tilde{\beta} + (\pi_v^{-1}-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,v) \in \tilde{\boldsymbol{r}}_n\}} \tag{S3.11}$$

$$\left\{ \frac{2(\pi_v^{-1}-2)\tilde{\beta} + (\pi_v^{-1}-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{\pi_v^{-1}-1}}(u) + \tilde{\alpha})}{2(\pi_v^{-1}-2)\tilde{\beta} + (\pi_v^{-1}-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,v) \notin \tilde{\boldsymbol{g}}_n\}}. \tag{S3.12}$$

For simplicity, we leave implicit the dependence of Q_{uv} on π, $\tilde{\boldsymbol{g}}_n$, and $\tilde{\boldsymbol{t}}_n$.

S3.5.1 Preliminary calculations

To simplify notation, for two positive integers $j < k$, we write $[j,k] := \{j, j+1, \ldots, k\}$, $[j,k) := \{j, j+1, \ldots, k-1\}$, and $(j,k] := \{j+1, j+2, \ldots, k\}$.

We first describe a fast algorithm to compute, for a particular node u and a time interval $[j,k]$ where $\pi_u^{-1} < j$, the quantity

$$\prod_{t \in [j,k]} Q_{u,\pi_t}, \tag{S3.13}$$

which can be interpreted as the part of the noise likelihood associated with node u on a time interval $[j,k]$. We first observe that

$$\prod_{t \in [j,k]} Q_{u,\pi_t} = \prod_{t \in [j,k]} \left\{ \frac{\theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{t-1}}(u) + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t) \in \tilde{\boldsymbol{r}}_n\}}$$

$$\left\{ \frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\tilde{\boldsymbol{t}}_{t-1}}(u) + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t) \notin \tilde{\boldsymbol{g}}_n\}}.$$

We extract the term $C_1 := \prod_{t \in [j,k]} \frac{1}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}}$ to obtain

$$\prod_{t \in [j,k]} Q_{u,\pi_t} = C_1 \prod_{t \in [j,k]} \left\{ \theta(\tilde{\beta} D_{\boldsymbol{t}_{t-1}}(u) + \tilde{\alpha}) \right\}^{\mathbb{1}\{(u,\pi_t) \in \tilde{\boldsymbol{r}}_n\}} \left\{ 2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{t-1}}(u) + \tilde{\alpha}) \right\}^{\mathbb{1}\{(u,\pi_t) \notin \tilde{\boldsymbol{g}}_n\}}$$

$$\left\{ 2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} \right\}^{\mathbb{1}\{(u,\pi_t) \in \tilde{\boldsymbol{t}}_n\}}.$$

We divide the time interval $[j,k]$ into sub-intervals in which $D_{\boldsymbol{t}_{t-1}}(u)$ is constant. To that end, define $j = t_0 < t_1 < \ldots < t_M = k+1$ such that

$$\begin{aligned}
&D_{\tilde{\boldsymbol{t}}_{t-1}}(u) = d_0 \text{ for all } t \in [t_0, t_1),\\
&D_{\tilde{\boldsymbol{t}}_{t-1}}(u) = d_1 = d_0 + 1 \text{ for all } t \in [t_1, t_2),\\
&\qquad \ldots\\
&D_{\tilde{\boldsymbol{t}}_{t-1}}(u) = d_{M-1} = d_0 + M - 1 \text{ for all } t \in [t_{M-1}, t_M).
\end{aligned}$$

Then, we have that

$$\begin{aligned}
\prod_{t\in[j,k]} Q_{u,\pi_t} &= C_1 \prod_{\ell=0}^{M-1} \prod_{t\in[t_\ell,t_{\ell+1})} \left\{\theta(\tilde{\beta}d_\ell + \tilde{\alpha})\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}}\\
&\qquad \left\{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})\right\}^{\mathbb{1}\{(u,\pi_t)\notin\tilde{\boldsymbol{g}}_n\}} \left\{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}}\\
&= C_1 \prod_{\ell=0}^{M-1} \prod_{t\in[t_\ell,t_{\ell+1})} \left\{\frac{\theta(\tilde{\beta}d_\ell + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})}\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}}\\
&\qquad \left\{\frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})}\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{t}}_n\}} \left\{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})\right\}.
\end{aligned}$$

To simplify, we observe that $2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} = (2\tilde{\beta} + \tilde{\alpha})t - (4\tilde{\beta} + \tilde{\alpha})$ and hence,

$$\begin{aligned}
&\prod_{t\in[t_\ell,t_{\ell+1})} \left\{(2\tilde{\beta} + \tilde{\alpha})t - (4\tilde{\beta} + \tilde{\alpha}) - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})\right\}\\
&\qquad = (2\tilde{\beta} + \tilde{\alpha})^{t_{\ell+1}-t_\ell} \frac{\Gamma(t_{\ell+1} - \frac{(4\tilde{\beta}+\tilde{\alpha})+\theta(\tilde{\beta}d_\ell+\tilde{\alpha})}{2\tilde{\beta}+\tilde{\alpha}})}{\Gamma(t_\ell - \frac{(4\tilde{\beta}+\tilde{\alpha})+\theta(\tilde{\beta}d_\ell+\tilde{\alpha})}{2\tilde{\beta}+\tilde{\alpha}})}.
\end{aligned}$$

Therefore, we may re-write $\prod_{t\in[j,k]} Q_{u,\pi_t}$ as follows:

$$\begin{aligned}
\prod_{t\in[j,k]} Q_{u,\pi_t} &= C_1 \prod_{\ell=0}^{M-1} (2\tilde{\beta} + \tilde{\alpha})^{t_{\ell+1}-t_\ell} \frac{\Gamma(t_{\ell+1} - \frac{(4\tilde{\beta}+\tilde{\alpha})+\theta(\tilde{\beta}d_\ell+\tilde{\alpha})}{2\tilde{\beta}+\tilde{\alpha}})}{\Gamma(t_\ell - \frac{(4\tilde{\beta}+\tilde{\alpha})+\theta(\tilde{\beta}d_\ell+\tilde{\alpha})}{2\tilde{\beta}+\tilde{\alpha}})}\\
&\prod_{t\in[t_\ell,t_{\ell+1})} \left\{\frac{\theta(\tilde{\beta}d_\ell + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})}\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}} \left\{\frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}d_\ell + \tilde{\alpha})}\right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{t}}_n\}}.
\end{aligned}$$

The quantities $\{t_\ell, d_\ell\}_{\ell=0}^{M-1}\}$ can be readily computed by iterating through the neighbors of u in $\tilde{\boldsymbol{g}}_n$. Therefore, this entire expression can be computed in time at most $O(D_{\tilde{\boldsymbol{g}}_n}(u))$, as $M \le D_{\tilde{\boldsymbol{g}}_n}(u)$. This concludes the description of the algorithm for computing (S3.13).

Now, suppose u is a node such that $\pi_u^{-1} \le k$ and that $D_{\boldsymbol{T}_{t-1}}(u) = 1$ for all $t \in [\pi_u^{-1}+1, k]$. We now give an efficient method to compute

$$\prod_{t=1}^{k} Q_{u,\pi_t}. \tag{S3.14}$$

This is the part of the noise likelihood associated with node u on the time interval $[1,k]$. We have that

$$\prod_{t=1}^{k} Q_{u,\pi_t} = \underbrace{\prod_{t=1}^{\pi_u^{-1}-1} Q_{u,\pi_t}}_{\text{first term}} \underbrace{\prod_{t=\pi_u^{-1}+1}^{k} Q_{u,\pi_t}}_{\text{second term}}. \tag{S3.15}$$

To compute the first term of (S3.15), we have

$$\prod_{t=1}^{\pi_u^{-1}-1} Q_{u,\pi_t} = \prod_{t=1}^{\pi_u^{-1}-1} \left\{ \frac{\theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\pi_t) + \tilde{\alpha})}{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}} \left\{ \frac{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\pi_t) + \tilde{\alpha})}{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\notin\tilde{\boldsymbol{g}}_n\}}.$$

Define $C_2 = \prod_{t=1}^{\pi_u^{-1}-1} \frac{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\pi_t) + \tilde{\alpha})}{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha}}$. Then,

$$\prod_{t=1}^{\pi_u^{-1}-1} Q_{u,\pi_t} = C_2 \left\{ \frac{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha}}{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\mathrm{pa}(u)) + \tilde{\alpha})} \right\} \prod_{t=1}^{\pi_u^{-1}-1} \left\{ \frac{\theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\pi_t) + \tilde{\alpha})}{2(\pi_u^{-1}-2)\tilde{\beta} + (\pi_u^{-1}-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{\pi_u^{-1}-1}}(\pi_t) + \tilde{\alpha})} \right\}^{\mathbb{1}\{(\pi_t,u)\in\tilde{\boldsymbol{r}}_n\}}.$$

Since it takes at most $O(D_{\tilde{\boldsymbol{g}}_n}(\mathrm{pa}(u)))$ time to compute $D_{\boldsymbol{T}_{\pi_u^{-1}-1}}(\mathrm{pa}(u))$, we see that the above expression, excluding C_2, can be computed in time at most $O(D_{\tilde{\boldsymbol{g}}_n}(\mathrm{pa}(u)) \vee D_{\tilde{\boldsymbol{g}}_n}(u))$. We do not need to compute the C_2 term in practice as we care only about ratios of likelihoods.

For the second term of (S3.15), we have that

$$\prod_{t=\pi_u^{-1}+1}^{k} Q_{u,\pi_t} = \prod_{t=\pi_u^{-1}+1}^{k} \left\{ \frac{\theta(\tilde{\beta} D_{\boldsymbol{t}_{t-1}}(u) + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}} \left\{ \frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta} D_{\boldsymbol{t}_{t-1}}(u) + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\notin\tilde{\boldsymbol{g}}_n\}}.$$

Since we assume that (π_t, u) is not a tree edge for every $t = \pi_u^{-1}+1, \ldots, k$, we have that $D_{\boldsymbol{t}_{t-1}}(u) = 1$ for all $t \in (\pi_u^{-1}, k]$ and thus,

$$\prod_{t=\pi_u^{-1}+1}^{k} Q_{u,\pi_t} = \prod_{t=\pi_u^{-1}+1}^{k} \left\{ \frac{\theta(\tilde{\beta} + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}} \left\{ \frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta} + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}^{\mathbb{1}\{(u,\pi_t)\notin\tilde{\boldsymbol{g}}_n\}}.$$

Define $C_3 = \prod_{t=\pi_u^{-1}+1}^{k} \left\{ \frac{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta}+\tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha}} \right\}$, we then have

$$\prod_{t=\pi_u^{-1}+1}^{k} Q_{u,\pi_t} = C_3 \prod_{t=\pi_u^{-1}+1}^{k} \left\{ \frac{\theta(\tilde{\beta} + \tilde{\alpha})}{2(t-2)\tilde{\beta} + (t-1)\tilde{\alpha} - \theta(\tilde{\beta} + \tilde{\alpha})} \right\}^{\mathbb{1}\{(u,\pi_t)\in\tilde{\boldsymbol{r}}_n\}}.$$

S3.5.2 Calculation for transposition sampling

In this section, we provide an efficient way to compute the acceptance probability in the Metropolis–Hastings algorithm for updating our sample of π. For clarity, we write $Q_{jk}(\pi) \equiv Q_{jk}(\pi, \tilde{\boldsymbol{t}}_n)$ to highlight the dependence of Q_{jk} on π. We first state a Lemma that gives an easy way to check if a proposed π^* is a valid history with respect to a given tree $\tilde{\boldsymbol{t}}_n$.

Lemma S7. *Let $\pi \in hist(\pi_1, \tilde{\boldsymbol{t}}_n)$. Let π^* be equal to π except that nodes u and v, neither equal to π_1, are swapped. Assume without loss of generality that $\pi_u^{-1} < \pi_v^{-1}$. Then, $\pi^* \in hist(\pi_1, \tilde{\boldsymbol{t}}_n)$ if and only*

(1) For any child w of u, we have $\pi_w^{-1} > \pi_v^{-1}$ and

(2) the parent $pa(v)$ satisfies $\pi_{pa(v)}^{-1} < \pi_v^{-1}$.

Proof. If π^* is in $\text{hist}(\pi_1, \tilde{\boldsymbol{t}}_n)$, it is clear that it must satisfy the two conditions (1) and (2).

Now assume conditions (1) and (2), we aim to show that $\pi^* \in \text{hist}(\pi_1, \tilde{\boldsymbol{t}}_n)$. Since π is a valid history, condition (1) implies that v cannot be a descendant (e.g. child, grand-child, etc) of u. Moreover, (2) implies that all ancestors of v have a π-position earlier than u. Therefore, it follows that swapping u and v yields a valid history π^*. The lemma follows as desired. □

We choose a pair $u = \pi_j$ and $v = \pi_k$ and define a new π^* equal to π except that

$$\pi_j^* = v, \quad \pi_k^* = u.$$

Suppose π^* satisfies the conditions of Lemma S7 so that $\pi^* \in \text{hist}(\pi_1, \tilde{\boldsymbol{t}}_n)$.

For a pair of nodes $x, y \in \mathcal{U}_n$, recall the definition of $Q_{x,y}(\pi)$ from (S3.12), where we now explicitly state the dependence of $Q_{x,y}$ on π. We have that

$$\frac{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi^*, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)\mathbb{P}(\Pi = \pi^* \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)}{\mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)\mathbb{P}(\Pi = \pi \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)} = \prod_{(x,y)} \frac{Q_{xy}(\pi^*)}{Q_{xy}(\pi)}.$$

We claim that $\frac{Q_{xy}(\pi^*)}{Q_{xy}(\pi)} = 1$ for all x, y that satisfy one of the following three conditions:

1. both $x, y \notin \{u, v, \text{pa}(u), \text{pa}(v)\}$;
2. $x \in \{u, v, \text{pa}(u), \text{pa}(v)\}$ and $\pi_y^{-1} > k$;
3. $x \in \{\text{pa}(u), \text{pa}(v)\}$ and $\pi_y^{-1} < j$.

This follows from the definition of $Q_{xy}(\pi)$. Therefore, we have that

$$\begin{aligned}\prod_{(x,y)} \frac{Q_{xy}(\pi^*)}{Q_{xy}(\pi)} = &\prod_{\substack{y\,:\,\pi_y^{-1} \le k,\\ y \notin \{\text{pa}(u), \text{pa}(v)\}}} \frac{Q_{uy}(\pi^*)}{Q_{uy}(\pi)} \prod_{\substack{y\,:\,\pi_y^{-1} \le k,\\ y \notin \{\text{pa}(u), \text{pa}(v), v\}}} \frac{Q_{vy}(\pi^*)}{Q_{vy}(\pi)} \\ &\prod_{y\,:\,\pi_y^{-1} \in [j,k]} \frac{Q_{\text{pa}(u),y}(\pi^*)}{Q_{\text{pa}(u),y}(\pi)} \prod_{y\,:\,\pi_y^{-1} \in [j,k]} \frac{Q_{\text{pa}(v),y}(\pi^*)}{Q_{\text{pa}(v),y}(\pi)}. \end{aligned} \tag{S3.16}$$

The first two terms on the RHS of (S3.16) are of the form (S3.14). The second two terms of the RHS of (S3.16) are of the form (S3.13). Therefore, the whole expression (S3.16) can be computed in time at most $O(D_{\tilde{\boldsymbol{g}}_n}(u) \vee D_{\tilde{\boldsymbol{g}}_n}(v) \vee D_{\tilde{\boldsymbol{g}}_n}(\text{pa}(u)) \vee D_{\tilde{\boldsymbol{g}}_n}(\text{pa}(v)))$.

S3.5.3 Calculations for tree sampling

For seq-PAPER model without deletion of tree edges:

For clarity, we write $Q_{jk}(\tilde{\boldsymbol{t}}_n) \equiv Q_{jk}(\pi, \tilde{\boldsymbol{t}}_n)$ to highlight the dependence of Q_{jk} on $\tilde{\boldsymbol{t}}_n$. For convenience, let us define

$$F(\tilde{\boldsymbol{t}}_n) := \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)\mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n \,|\, \Pi = \pi) \tag{S3.17}$$

$$= \prod_{x,y} Q_{xy}(\tilde{\boldsymbol{t}}_n) \frac{\prod_{v \in \mathcal{U}_n} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}_n}(v)-1} (\beta j + \alpha)}{\prod_{t=3}^{n} 2(t-2)\beta + (t-1)\alpha}. \tag{S3.18}$$

We iterate $t = 2, 3, \ldots, n$ and sample a new parent for π_t among the candidate set $\pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$. For each $w \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$, define $\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}$ as the disconnected graph that results from removing the edge $(\mathrm{pa}(\pi_t), \pi_t)$ from $\tilde{\boldsymbol{t}}_n$, and define $\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}$ as the tree that results from adding the edge (w, π_t) to $\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}$.

For $t = 1, 2, \ldots, n$, we then sample a new parent for π_t by removing $(\mathrm{pa}(\pi_t), \pi_t)$ and then randomly choosing $w \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$ with probability

$$\frac{F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)})}{\sum_{u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} F(\tilde{\boldsymbol{t}}_n^{(u,\pi_t)})}. \tag{S3.19}$$

Calculating $F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)})$ naively according to (S3.18) takes time $O(n^2)$. We give a faster algorithm here.

We start by noting that if **(1)** $x, y \notin \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$ or **(2)** $x \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$ and y is such that $\pi_y^{-1} < t$, then the tree degree of x at time $\pi_y^{-1} - 1$ (or the tree degree of y at time $\pi_x^{-1} - 1$) is the same under both $\tilde{\boldsymbol{t}}_n$ and $\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}$ for any w and hence, $Q_{xy}(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) = Q_{xy}(\tilde{\boldsymbol{t}}_n)$. Therefore, we have that

$$F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) = C\big\{\beta(D_{\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}}(w) - 1) + \alpha\big\} \prod_{u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \prod_{y\,:\,\pi_y^{-1} \geq t} Q_{uy}(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}), \tag{S3.20}$$

where C is a term that does not depend on w; more precisely, we have that

$$C = \Bigg\{ \prod_{x,y \notin \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} Q_{xy}(\tilde{\boldsymbol{t}}_n) \Bigg\} \Bigg\{ \prod_{u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \prod_{y\,:\,\pi_y^{-1} < t} Q_{uy}(\tilde{\boldsymbol{t}}_n) \Bigg\}$$
$$\Bigg\{ \frac{\prod_{v \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}}(v)-1} (\beta j + \alpha) \prod_{v \notin \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}_n}(v)-1} (\beta j + \alpha)}{\prod_{t=3}^{n} 2(t-2)\beta + (t-1)\alpha} \Bigg\}.$$

We make one further simplication. Since $Q_{uy}(\tilde{\boldsymbol{t}}^{(w,\pi_t)})$ depends on the tree $\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}$ only through its degree sequence across time, we observe that, for an arbitrary fixed $u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$, the quantity $\prod_{y\,:\,\pi_y^{-1} \geq t} Q_{uy}(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)})$ depends on w only through the binary value of whether $w = u$ or $w \neq u$. Therefore, for any $u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$, we write

$$\begin{aligned} B(u) &= \prod_{y\,:\,\pi_y^{-1} \geq t} Q_{uy}(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) \quad \text{for any } w \neq u \\ A(u) &= \prod_{y\,:\,\pi_y^{-1} \geq t} Q_{uy}(\tilde{\boldsymbol{t}}_n^{(u,\pi_t)}). \end{aligned} \tag{S3.21}$$

Then, by defining $C' = \prod_{u \in \pi_{1:(t-1)} \cap N_{\tilde{\boldsymbol{g}}_n}(\pi_t)} B(u)$, we have that

$$F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) = C \cdot C' \cdot \frac{A(w)}{B(w)} \big\{\beta(D_{\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}}(w) - 1) + \alpha\big\}.$$

The terms $A(w), B(w)$ are of the form (S3.13) and can thus be computed in time proportional to the degree $D_{\tilde{\boldsymbol{g}}_n}(w)$. Therefore, the whole term $F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)})$ can be, up to constants C, C' which do not depend on w, computed in time $O(D_{\tilde{\boldsymbol{g}}_n}(w))$.

For seq-PAPER* model with potential deletion of tree edges:

With deletion noise, we must incorporate the likelihood of tree edge removal into (S3.18). We denote $E(\tilde{\boldsymbol{t}}_n)$ and $E(\tilde{\boldsymbol{g}}_n)$ as the sets of edges of $\tilde{\boldsymbol{t}}_n$ and $\tilde{\boldsymbol{g}}_n$ respectively and define

$$F(\tilde{\boldsymbol{t}}_n) := \mathbb{P}(\tilde{\boldsymbol{G}}_n = \tilde{\boldsymbol{g}}_n \,|\, \Pi = \pi, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n)\mathbb{P}(\tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n \,|\, \Pi = \pi) \tag{S3.22}$$

$$= \prod_{x,y} Q_{xy}(\tilde{\boldsymbol{t}}_n) \frac{\prod_{v \in \mathcal{U}_n} \prod_{j=1}^{D_{\tilde{\boldsymbol{t}}_n}(v)-1} (\beta j + \alpha)}{\prod_{t=3}^{n} 2(t-2)\beta + (t-1)\alpha} (1-\eta)^{|E(\tilde{\boldsymbol{t}}_n) \cap E(\tilde{\boldsymbol{g}}_n)|} \eta^{|E(\tilde{\boldsymbol{t}}_n) \setminus E(\tilde{\boldsymbol{g}}_n)|}. \tag{S3.23}$$

Define $\tilde{\boldsymbol{t}}_n^{(\cdot,\pi_t)}$ as the disconnected graph that results from removing $(\mathrm{pa}(\pi_t), \pi_t)$ just as in the discussion following (S3.18) and, for $w \in \pi_{1:(t-1)}$, define $\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}$ as the tree that results from adding (w, π_t). Note that we do not require $w \in N_{\tilde{\boldsymbol{g}}_n}(\pi_t)$, i.e. (w, π_t) need not be an edge in $\tilde{\boldsymbol{g}}_n$, and hence, $\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}$ may not be a subgraph of $\tilde{\boldsymbol{g}}_n$.

Following the same derivation as (S3.20), we have that

$$F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) = C\big\{\beta(D_{\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}}(w) - 1) + \alpha\big\} \eta^{\mathbb{1}\{(w,\pi_t) \notin \tilde{\boldsymbol{g}}_n\}} (1-\eta)^{\mathbb{1}\{(w,\pi_t) \in \tilde{\boldsymbol{g}}_n\}} \prod_{u \in \pi_{1:(t-1)}} \prod_{y \,:\, \pi_y^{-1} \geq t} Q_{uy}(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}), \tag{S3.24}$$

where C is a term that does not depend on w.

Defining $A(\cdot)$ and $B(\cdot)$ as in (S3.21), we then have

$$F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}) = C \cdot C' \cdot \frac{A(w)}{B(w)} \big\{\beta(D_{\tilde{\boldsymbol{t}}_n^{(w,\pi_t)}}(w) - 1) + \alpha\big\} \cdot \eta^{\mathbb{1}\{(w,\pi_t) \in \tilde{\boldsymbol{g}}_n\}} (1-\eta)^{\mathbb{1}\{(w,\pi_t) \notin \tilde{\boldsymbol{g}}_n\}} \tag{S3.25}$$

Since $A(w)$ and $B(w)$ can be computed in time $O(D_{\tilde{\boldsymbol{g}}_n})$, we have that $F(\tilde{\boldsymbol{t}}_n^{(w,\pi_t)})$ can be computed in time $O(D_{\tilde{\boldsymbol{g}}_n})$ as well.

The overall procedure is then to sample $w \in \pi_{1:(t-1)}$ with probability proportional to (S3.25) and replacing the edge $(\mathrm{pa}(\pi_t), \pi_t)$ with (w, π_t) in the tree $\tilde{\boldsymbol{t}}_n$.

S3.5.4 Parameter sampling for the seq-PAPER model

Although it may be possible to derive an EM algorithm to estimate the parameters $\alpha, \beta, \theta, \tilde{\alpha}, \tilde{\beta}$ in the seq-PAPER model, we propose to take a full Bayesian approach where we impose a prior and sample the parameters after sampling the ordering π and the tree $\tilde{\boldsymbol{\pi}}_n$ in the Gibbs sampler.

As in Section S3.1, we assume that $\beta, \tilde{\beta} > 0$ so that we may assume without loss of generality that $\beta = \tilde{\beta} = 1$ and only estimate α and $\tilde{\alpha}$. We propose to use an Exponential(λ) prior for α, θ, and $\tilde{\alpha}$ with $\lambda = 0.1$. Conditional on the ordering π and the tree $\tilde{\boldsymbol{t}}_n$, the likelihood for α is the same as that of $\ell(\alpha; \tilde{\boldsymbol{T}}_n)$ in Section S3.1; the likelihood for $\tilde{\alpha}$ and θ is, writing $\tilde{\boldsymbol{r}}_n = \tilde{\boldsymbol{g}}_n \backslash \tilde{\boldsymbol{t}}_n$,

$$\ell(\tilde{\alpha}; \tilde{\boldsymbol{g}}_n, \tilde{\boldsymbol{t}}_n, \pi) = \prod_{j=3}^{n-1} \prod_{k=j+1}^{n} Q_{jk}$$

where $Q_{jk} = \big\{\frac{\theta(D_{\tilde{\boldsymbol{t}}_{k-1}}(\pi_j) + \tilde{\alpha})}{2(k-2) + (k-1)\tilde{\alpha}}\big\}^{\mathbb{1}\{(\pi_j, \pi_k) \in \tilde{\boldsymbol{r}}_n\}} \big\{1 - \frac{\theta(D_{\tilde{\boldsymbol{t}}_{k-1}}(\pi_j) + \tilde{\alpha})}{2(k-2) + (k-1)\tilde{\alpha}}\big\}^{\mathbb{1}\{(\pi_j, \pi_k) \notin \tilde{\boldsymbol{g}}_n\}}$ is the contribution to the likelihood from the pair (π_j, π_k).

Conditionally on $\tilde{\boldsymbol{t}}_n$ and π, it is still intractable to directly sampling $\alpha, \theta, \tilde{\alpha}$ so we propose Metropolis updates where we generate the new proposal either by adding a draw from Unif$[-\delta, \delta]$ or by multiplying with log-normal e^Z for $Z \sim N(0, \delta)$, where the ratio of proposal probabilities is easy to compute with a Jacobian adjustment.

S4 Proof of results in Section 5

We first give the proof of the optimality lemma for $B_\epsilon(\cdot)$.

Proof. (of Lemma 10)
Fix $\epsilon, \delta \in (0,1)$ and suppose that $C_{\delta\epsilon}(\cdot)$ is a labeling-equivariant (see Remark 4) confidence set for the root node with asymptotic coverage $1-\delta\epsilon$, that is, there exists a sequence $\mu_n \to 0$ such that $\mathbb{P}(\rho_1 \in C_{\delta\epsilon}(\mathbf{G}_n^*)) \geq 1-\delta\epsilon-\mu_n$.

Let Λ be a random permutation drawn uniformly from $\mathrm{Bi}(\mathcal{U}_n, \mathcal{U}_n)$ and write $\Pi = \Lambda \circ \rho$ so that $\tilde{\boldsymbol{G}}_n := \Lambda \boldsymbol{G}_n^* = \Pi \boldsymbol{G}_n$ is the randomly labeled graph. Then, there exists a real-valued sequence $\mu_n \to 0$ such that

$$\begin{aligned}
&\mathbb{P}\big\{\Pi_1 \in C_{\delta\epsilon}(\tilde{\mathbf{G}}_n)\big\} \\
&= \sum_{\pi \in \mathrm{Bi}([n], \mathcal{U}_n)} \mathbb{P}\big\{\pi_1 \in C_{\delta\epsilon}(\pi \mathbf{G}_n) \,|\, \Pi = \pi\big\} \mathbb{P}(\Pi = \pi) \\
&= \mathbb{P}(\rho_1 \in C_{\delta\epsilon}(\rho \mathbf{G}_n)) \\
&= \mathbb{P}(\rho_1 \in C_{\delta\epsilon}(\mathbf{G}_n^*)) \geq 1-\delta\epsilon+\mu_n,
\end{aligned} \tag{S4.26}$$

where the penultimate equality follows from the labeling-equivariance of $C_{\delta\epsilon}(\cdot)$.

For any labeled graph $\tilde{\mathbf{g}}_n$, we have from definition (11) that $B_\epsilon(\tilde{\mathbf{g}}_n)$ is the smallest labeling-equivariant subset of $\mathcal{U}_n$ such that $\mathbb{P}(\Pi_1 \in B_\epsilon(\tilde{\mathbf{t}}_n) \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) \geq 1-\epsilon$. Then, if $|B_\epsilon(\tilde{\mathbf{g}}_n)| > |C_{\delta\epsilon}(\tilde{\mathbf{g}}_n)|$, then it must be that $\mathbb{P}(\Pi_1 \in C_{\delta\epsilon}(\tilde{\mathbf{G}}_n) \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) < 1-\epsilon$.

Therefore, we have from (S4.26) that

$$\begin{aligned}
1-\delta\epsilon+\mu_n &\leq \mathbb{P}(\Pi_1 \in C_{\delta\epsilon}(\tilde{\mathbf{G}}_n)) \\
&= \sum_{\tilde{\mathbf{g}}_n} \mathbb{P}(\Pi_1 \in C_{\delta\epsilon}(\tilde{\mathbf{G}}_n) \,|\, \tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) \mathbb{P}(\tilde{\mathbf{G}}_n = \tilde{\mathbf{g}}_n) \\
&\leq \mathbb{P}\big\{|B_\epsilon(\tilde{\mathbf{G}}_n)| \leq |C_{\delta\epsilon}(\tilde{\mathbf{G}}_n)|\big\} + (1-\epsilon)\mathbb{P}\big\{|B_\epsilon(\tilde{\mathbf{G}}_n)| > |C_{\delta\epsilon}(\tilde{\mathbf{G}}_n)|\big\}.
\end{aligned}$$

We then obtain by algebra that

$$\mathbb{P}\big\{|B_\epsilon(\tilde{\mathbf{G}}_n)| > |C_{\delta\epsilon}(\tilde{\mathbf{G}}_n)|\big\} \leq \delta + \mu_n/\epsilon,$$

which yields the desired conclusion. □

S4.1 Proof of results in LPA setting

Next, we give the proof of all statements regarding the LPA setting.

Proof. (of Theorem 11)

Since $\boldsymbol{G}_n = \boldsymbol{T}_n + \boldsymbol{R}_n$ for a linear preferential attachment tree $\boldsymbol{T}_n$ and an Erdős–Rényi graph $\boldsymbol{R}_n$, we have that $D_{\boldsymbol{G}_n} = D_{\boldsymbol{T}_n} + D_{\boldsymbol{R}_n}$.

By Peköz et al. (2014), we have that, for any $q > 2$,

$$\frac{1}{\sqrt{n}}(D_{\boldsymbol{T}_n}(1), D_{\boldsymbol{T}_n}(2), \ldots, D_{\boldsymbol{T}_n}(n), 0, 0, \ldots) \xrightarrow{d} (Y_1, Y_2, Y_3, \ldots),$$

in distribution with respect to the ℓ_q metric where $(Y_1, Y_2, \ldots)$ is a random sequence satisfying $\sum_{j=1}^{\infty} \mathbb{E}Y_j^q < \infty$ and each random variable Y_j has a density with respect to the Lebesgue measure.

We first claim that, for any $q > \frac{1}{\delta}$, if $\theta \le n^{-\frac{1}{2}-\delta}$, then

$$\frac{1}{\sqrt{n}}(D_{\boldsymbol{R}_n}(1), D_{\boldsymbol{R}_n}(2), \dots, D_{\boldsymbol{R}_n}(n), 0, 0 \dots) \to (0, 0, 0 \dots)$$

in ℓ_q metric. Indeed, we have

$$\begin{aligned}\mathbb{E}\|n^{-1/2}(D_{\boldsymbol{R}_n}(1), D_{\boldsymbol{R}_n}(2), \dots, D_{\boldsymbol{R}_n}(n), 0, 0, \dots)\|_q^q &= n^{-\frac{q}{2}} \sum_{k=1}^{n} \mathbb{E} D_{\boldsymbol{R}_n}(k)^q \\ &\overset{(a)}{\le} n^{1-\frac{q}{2}} \mathbb{E}\big(\mathrm{Bin}(n-1, \theta)^q\big) \overset{(b)}{\le} n^{1-\frac{q}{2}}((2\theta n)^q + C_q) \\ &\le 2^q n^{1-\frac{q}{2}} n^{\frac{q}{2}-q\delta} + C_q n^{1-\frac{q}{2}} = 2^q n^{1-q\delta} + C_q n^{1-\frac{q}{2}},\end{aligned}$$

where the inequality (a) follows since $D_{\boldsymbol{R}_n}(k)$ is Binomial with $n - D_{\boldsymbol{T}_n}(k)$ trials and hence stochastically dominated by $\mathrm{Bin}(n-1, \theta)$ and where inequality (b) follows from Lemma S8.

Since $q > 2 \vee 1/\delta$ by assumption, we have that

$$\limsup_{n\to\infty} \mathbb{E}\|n^{-1/2}(D_{\boldsymbol{R}_n}(1), D_{\boldsymbol{R}_n}(2), \dots, D_{\boldsymbol{R}_n}(n), 0, 0, \dots)\|_q^q = 0$$

and thus $\frac{1}{\sqrt{n}}(D_{\boldsymbol{R}_n}(1), D_{\boldsymbol{R}_n}(2), \dots, D_{\boldsymbol{R}_n}(n), 0, 0 \dots) \to (0, 0, \dots)$ in distribution.

Since $D_{\boldsymbol{G}_n}(k) = D_{\boldsymbol{T}_n}(k) + D_{\boldsymbol{R}_n}(k)$ for all $k \in [n]$, we have by Slutsky's lemma that

$$\frac{1}{\sqrt{n}}(D_{\boldsymbol{G}_n}(1), D_{\boldsymbol{G}_n}(2), \dots, D_{\boldsymbol{G}_n}(n), 0, 0, \dots) \xrightarrow{d} (Y_1, Y_2, Y_3, \dots).$$

We claim that, for any $\epsilon \in (0, 1)$, there exists $L_\epsilon \in \mathbb{N}$ such that $\mathbb{P}(Y_1 \le L_\epsilon\text{-}\max(\{Y_n\})) \le \epsilon$. To see this, recall that Y_1 has a density $q(\cdot)$ on $[0, \infty)$ with respect to the Lebesgue measure and, fixing some $q > 2$, that $\mathbb{E}Y_j^q \to 0$ as $j \to \infty$. Therefore, choosing any $\delta > 0$ such that $\mathbb{P}(Y_1 \le \delta) \le \frac{\epsilon}{2}$ and L_ϵ such that $\mathbb{E}Y_{L_\epsilon}^q \le \frac{\epsilon}{2}\delta^j$, we have by Markov's inequality that

$$\begin{aligned}\mathbb{P}(Y_1 \le Y_{L_\epsilon}) &\le \int_0^\infty \mathbb{P}(Y_{L_\epsilon} > t) q(t)\, dt \\ &\le \int_0^\delta q(t)\, dt + \int_\delta^\infty \mathbb{P}(Y_{L_\epsilon} > t) q(t)\, dt \\ &\le \mathbb{P}(Y_1 \le \delta) + \mathbb{P}(Y_{L_\epsilon} > \delta) \int_\delta^\infty q(t)\, dt \\ &\le \frac{\epsilon}{2} + \frac{\mathbb{E}Y_{L_\epsilon}^q}{\delta^q} \le \epsilon.\end{aligned}$$

Since $L_\epsilon\text{-}\max(\cdot)$ function on sequences is continuous with respect to ℓ_q, we have by continuous mapping theorem and Portmanteau lemma that

$$\limsup_{n\to\infty} \mathbb{P}\big\{D_{\boldsymbol{G}_n}(1) \le L_\epsilon\text{-}\max(D_{\boldsymbol{G}_n})\big\} \le \mathbb{P}\big\{Y_1 \le L_\epsilon\text{-}\max(\{Y_n\})\big\} \le \epsilon.$$

This proves the first conclusion of Theorem 11.

To obtain the second conclusion, note that $C_\epsilon(\boldsymbol{G}_n^*) := \big\{1\text{-}\max(D_{\tilde{\boldsymbol{G}}_n}), 2\text{-}\max(D_{\tilde{\boldsymbol{G}}_n}), \dots, L_\epsilon\text{-}\max(D_{\tilde{\boldsymbol{G}}_n})\big\}$ is a labeling-equivariant confidence set for the root at asymptotical level $1-\epsilon$. The second conclusion follows from Lemma 10.

□

Lemma S8. *Let X be a random variable with $Bin(n, \theta)$ distribution. For any $q \ge 1$, $\theta \in [0, 1]$ and any $n \in \mathbb{N}$, we have that*

$$\mathbb{E}X^q \le (2\theta n)^q + C_q,$$

where $C_q > 0$ is a constant that depends only on q.

Proof. Write X as a random variable with the $\mathrm{Bin}(n,\theta)$ distribution. Then,

$$\begin{aligned}\mathbb{E}X^q &= \int_0^\infty \mathbb{P}(X^q \geq t)dt \\ &\leq (2\theta n)^q + \int_{(2\theta n)^q}^\infty \mathbb{P}(X^q \geq t)dt. \end{aligned} \tag{S4.27}$$

We note that $\mathrm{Var}X \leq \theta n$. By Bernstein's inequality, we have that for all $t \geq (2\theta n)^q$,

$$\begin{aligned}\mathbb{P}(X^q \geq t) &= \mathbb{P}(X - \theta n \geq t^{1/q} - \theta n) \\ &\leq \exp\left(-\frac{1}{2}\frac{(t^{1/q}-\theta n)^2}{(t^{1/q}-\theta n)+\theta n}\right) \\ &\leq \exp(-\frac{1}{8}t^{1/q}).\end{aligned}$$

Therefore, we may bound the second term of (S4.27) as

$$\begin{aligned}\int_{(2\theta n)^q}^\infty \mathbb{P}(X^q \geq t)dt &\leq \int_{(2\theta n)^q}^\infty e^{-\frac{t^{1/q}}{8}}dt \\ &\leq \int_0^\infty q s^{q-1} e^{-\frac{s}{8}} ds.\end{aligned}$$

□

S4.2 Proof of results in UA setting

Proof. (of Theorem 12)

Let $\boldsymbol{T}_n$ be a random recursive tree with the UA distribution. Let $s \in [n]$ be a node with arrival time s and assume that $s \geq n^\eta$. For any integer $i \geq 1$, we define the random variable

$$Z_i^{(s)} := \begin{cases} 1 & \text{if node } i+1 \text{ is attached to node } 1 \\ -1 & \text{if node } i+1 \text{ is attached to node } s \\ 0 & \text{else} \end{cases}$$

We note then that $\{Z^{(s)}\}_{i=1}^n$ are independent. If $i \geq s$, then $\mathbb{E}Z_i^{(s)} = 0$ and $\mathrm{Var}Z_i^{(s)} = \frac{2}{i}$, and if $i < s$, then we cannot attach to node s and hence, $\mathbb{E}Z_i^{(s)} = \frac{1}{i}$ and $\mathrm{Var}Z_i^{(s)} \leq \frac{1}{i}$. Define $Z^{(s)} = \sum_{i=1}^n Z_i^{(s)}$ so that

$$Z^{(s)} = D_{\boldsymbol{T}_n}(1) - D_{\boldsymbol{T}_n}(s).$$

Then, we have that

$$\begin{aligned}\mathbb{E}Z^{(s)} &= \sum_{i=1}^n \mathbb{E}Z_i^{(s)} = \sum_{i=1}^s \frac{1}{i} \geq (1+\mu_1)\log s \\ \mathrm{Var}Z^{(s)} &= \sum_{i=1}^n \mathrm{Var}Z_i^{(s)} \leq \sum_{i=2}^s \frac{1}{i} + \sum_{i=s+1}^n \frac{2}{i} \leq (1+\mu_2)\big\{\log s + 2\big(\log n - \log s\big)\big\}.\end{aligned}$$

where we use μ_1, μ_2 to represent terms that are $o(1)$ as $n \to \infty$. Therefore, we obtain that

$$\mathbb{E}\big(D_{\boldsymbol{G}_n}(1) - D_{\boldsymbol{G}_n}(s)\big) = \mathbb{E}Z^{(s)} + \mathbb{E}\big(D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s)\big) \leq (1+\mu_1)\log s,$$

where the inequality follows since $D_{\boldsymbol{R}_n}(s)$ has the $\mathrm{Bin}(n - D_{\boldsymbol{T}_n}(s), \theta)$ distribution; since $D_{\boldsymbol{T}_n}(1)$ stochastically dominates $D_{\boldsymbol{T}_n}(s)$, we have that $D_{\boldsymbol{R}_n}(s)$ stochastically dominates $D_{\boldsymbol{R}_n}(1)$. We also have the following bound on the variance of $D_{\boldsymbol{G}_n}(1) - D_{\boldsymbol{G}_n}(s)$:

$$\begin{aligned}
\mathrm{Var}\big(D_{\boldsymbol{G}_n}(1) - D_{\boldsymbol{G}_n}(s)\big) &= \mathrm{Var}\bigg(\sum_{i=1}^{n} Z_i^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s)\bigg) \\
&\le \mathbb{E}\,\mathrm{Var}\bigg(\sum_{i=1}^{n} Z_i^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s)\bigg| D_{\boldsymbol{R}_n}(1), D_{\boldsymbol{R}_n}(s)\bigg) \\
&\qquad + \mathrm{Var}\,\mathbb{E}\bigg[\sum_{i=1}^{n} Z_i^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}}(s)\bigg| D_{\boldsymbol{R}_n}(s), D_{\boldsymbol{R}_n}(1)\bigg] \\
&\le (1+\mu_2)\{\log s + 2(\log n - \log s)\} + 2n\theta \\
&\le (1+\mu_3)(2-\eta)\log n.
\end{aligned}$$

Hence, we have by Proposition S9 that

$$\begin{aligned}
\mathbb{P}(D_{\boldsymbol{G}_n}(s) \ge D_{\boldsymbol{G}_n(1)}) &= \mathbb{P}\bigg(\sum_{i=1}^{n} Z_i^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s) \le 0\bigg) \\
&\le \mathbb{P}\bigg(\sum_{i=1}^{n} Z_i^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s) - \mathbb{E}\big[Z^{(s)} + D_{\boldsymbol{R}_n}(1) - D_{\boldsymbol{R}_n}(s)\big] \le -(1+\mu_1)\log s\bigg) \\
&\le 2\exp\bigg(-(1+\mu_3)(2-\eta)\log n \cdot h\bigg(\frac{(1+\mu_1)\eta\log n}{(1+\mu_3)(2-\eta)\log n}\bigg)\bigg) \\
&\le 2(1+\mu_4)n^{-(2-\eta)h(\frac{\eta}{2-\eta})}.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
&\mathbb{P}\big(|\{s \ge n^\eta \,:\, D_{\boldsymbol{G}_n}(s) > D_{\boldsymbol{G}_n}(1)\}| \le 2\epsilon^{-1}n^{1-(2-\eta)h(\frac{\eta}{2-\eta})}\big) \\
&\quad \le \epsilon n^{-1+(2-\eta)h(\frac{\eta}{2-\eta})}\mathbb{E}|\{s \ge n^\eta \,:\, D_{\boldsymbol{G}_n}(s) > D_{\boldsymbol{G}_n}(1)\}| \\
&\quad \le \epsilon n^{-1+(2-\eta)h(\frac{\eta}{2-\eta})} \sum_{s=\lfloor n^\eta \rfloor}^{n} \mathbb{P}(D_{\boldsymbol{G}_n}(s) \ge D_{\boldsymbol{G}_n}(1)) \\
&\quad \le \epsilon(1+\mu_4).
\end{aligned}$$

Hence, we have that with probability at least $1 - (1+\mu_4)\epsilon$,

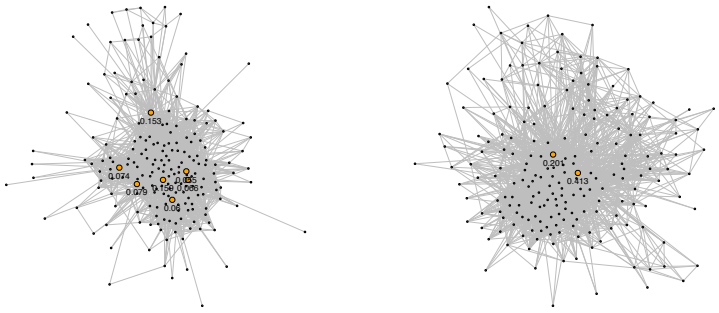
$$D_{\boldsymbol{G}_n}(1) \ge L_{\eta,n,\epsilon}\text{-}\max(D_{\boldsymbol{G}_n}).$$

By optimizing η, we have that for some $\gamma < 0.8$ and universal constant $C > 0$, with probability at least $1 - (1+\mu_4)\epsilon$,

$$D_{\boldsymbol{G}_n}(1) \ge \frac{C}{\epsilon}n^\gamma\text{-}\max(D_{\boldsymbol{G}_n}).$$

Therefore, we may form a level $1-\epsilon$ asymptotically valid confidence set for the root node by taking the $\frac{C}{\epsilon}n^\gamma$ nodes with the highest degree in the observed alphabetically labeled graph $\boldsymbol{G}_n^*$. The second claim of the theorem follows directly from Lemma 10. □

The next concentration inequality is standard.



(a) Enron email subgraph (b) Youtube subgraph

Figure 25: Subgraph of the 200 nodes with highest posterior root probabilities.

Proposition S9. *(Bennett's inequality)*
Let $X_1, \dots X_n$ *be independent random variables such that* $|X_i| \leq b$. *Let* $V \geq \sum_{i=1}^n Var(X_i)$. *Then, for any* $t \geq 0$,

$$\mathbb{P}\left(\left|\sum_{i=1}^n X_i - \mathbb{E}X_i\right| > t\right) \leq 2\exp\left(-\frac{V}{b^2}h\left(\frac{bt}{V}\right)\right),$$

where $h(z) = (1+z)\log(1+z) - z$.

S5 Supplement for Section 6

S5.1 Additional results for central subgraph visualization

Enron email network: This dataset consists of email exchanges between members of the Enron corporation shortly before its bankruptcy and the network is publicly available at the website `https://snap.stanford.edu/data/email-Enron.html` (c.f. Leskovec et al. (2009)) for more details on the network). This network has $n = 33,696$ nodes and $m = 180,811$ edges, with a maximum degree of 1,383. We estimate $\beta = 1$ and $\alpha = 0$ and the sizes of confidence sets are:

60%: 7 nodes 80%: 11 nodes 95%: 42 nodes 99%: 2393 nodes .

The central subgraph of this network (shown in Figure 25a) exhibits a large central cluster with many nodes that have relatively large posterior root probabilities. These nodes may correspond to leadership personnel in the company.

Youtube social network: This dataset consists of friendship links between users in Youtube (Mislove et al.; 2007) and it is publicly available at `https://snap.stanford.edu/data/com-Youtube.html`. This network has $n = 1,134,890$ nodes and $m = 2,987,624$ edges, with a maximum degree of 28,754. We estimate $\beta = 1$ and $\alpha = 0$ and the sizes of confidence sets are:

60%: 2 nodes 80%: 35 nodes 95%: 1874 nodes 99%: 16368 nodes .

The central subgraph of this network (shown in Figure 25b) also contains a large central cluster, which may contain the most popular accounts on Youtube.

S5.2 Random K roots analysis on air route network

We analyze an air route network (Guimera et al.; 2005) of $n = 3,618$ airports and $m = 14,142$ edges where two airports share an edge if there is a regularly scheduled flight between them. We remove the direction of the edges and treat the network as undirected. The dataset is publicly available at `http://seeslab.info/downloads/air-transportation-networks/`.

We perform our inference algorithm and display the top 12 community–trees in Figure 26. That is, we take $\{Q_1, \ldots, Q_{K_{\text{all}}}\}$ and display the 12 that has the largest posterior probability of occuring. The first 6 community–trees represent the same community, basically of all the major airports in the world, centered at various potential root nodes (Paris, London, Moscow, Tokyo, Chicago, Frankfurt).

The 7th community–tree comprise of regional airports in the remote Northwest Territories province of Canada and it is centered at Yellowknife, which is the capital of the province. This is not surprising because most regional airports in Northern Canada are very small and are built only to connect remote settlements to larger nearby cities such as Yellowknife.

The 8th community–tree comprise of regional airports on various Pacific and Polynesian islands and it is centered at Port Moresby, the capital of Papua New Guinea. The 9th community–tree is the Australia/Southeast Asia cluster centered at Sydney. This result is sensible again because most airports in the pacific islands are built only to connect the small islands to larger nearby cities such as Port Moresby or Cairns. From a network respectively, these remote airports are reachable only through a few cities such as Port Moresby.

The 10th to 12th community–trees comprise of airports in Alaska, many of which are regional. The 10th community–tree is the whole Alaska cluster centered at Anchorage while the 11th community–tree and the 12th community–tree represent, respectively, Western Alaska (centered at Bethel, AK) and Northern Alaska (centered at Fairbanks, AK).

S5.3 Additional clusters for statistician co-authorship network

In this section, we give 18 additional clusters discovered on the statistican co-authorship network in Figure 27, expanding the results given in Section 6.5 of the main paper.

S5.4 Sampler diagnostic information

In this section, we give detailed sampler diagnostic information of the Gibbs sampling algorithm proposed in Section 4. We use a simulation setting where we generate a PAPER network with $n = 2000$ nodes and $m = 4000$ edges with $K = 1$ and we also use the statistician co-authorship network analyzed in Section 6.5, which has $n = 2263$ nodes and $m = 4388$ edges.

Recall that our Gibbs sampler produces a sequence of samples of a spanning tree $\tilde{\boldsymbol{t}}_n^{(j)}$ and ordering $\pi^{(j)}$ for $j = 1, 2, \ldots, J$ where J is the number of Gibbs outer iterations. We use $\tilde{\boldsymbol{t}}_n^{(j)}$ to compute the "sampled" posterior root probability $Q^{(j)}(\cdot) = \mathbb{P}(\Pi_1 = \cdot \,|\, \tilde{\boldsymbol{T}}_n = \tilde{\boldsymbol{t}}_n^{(j)})$. For the simulation setting, we then construct trace plot and auto-correlation plot based on the sequence $\{Q^{(j)}(\text{true root})\}_{j=1}^J$. For the statistician co-authorship network, we use construct the plots based on $\{Q^{(j)}(\text{ Raymond Carroll })\}_{j=1}^J$. Figures 28a, 28b, 28c, and 28d suggest that the sampler is able to converge to the stationary distribution and has no significant autocorrelation.

As described in Section S3.4, to assess convergence, we run two parallel chains A and B with corresponding posterior root probability estimates $Q^{A(1:J)}(\cdot) = \frac{1}{J}\sum_{j=1}^J Q^{A(j)}(\cdot)$ and $Q^{B(1:J)}(\cdot) = \frac{1}{J}\sum_{j=1}^J Q^{A(j)}(\cdot)$. We then compute the Hellinger distance $d_{\text{H}}(Q^{A(1:J)}, Q^{B(1:J)})$ and increase J until the distance is small enough. In Figures 29a and 29b, we show that $d_{\text{H}}(Q^{A(1:J)}, Q^{B(1:J)})$ indeed

Figure 26: Top 12 community–trees on the air route network; first 6 trees reflect the hub of major global airports centered at different cities; tree 7 contains remote regional airports in the Northwest Territories province of Canada; tree 8 contains remote regional airports in southeast Asian Pacific islands; tree 9 contains Australia/Southeast Asia airports; tree 10 contains Alaskan airports while tree 11 and 12 contain western Alaskan and Northern Alaskan airports respectively.

converges to 0 quickly as J increases. We emphasize that the chains A and B are initialized with a uniformly random spanning tree and a uniformly random ordering on that tree so that the initialization is guaranteed to be overdispersed.

(a) Bayesian

(b) Experimental design

(c) Biostat/Survey

(d) Biostat

(e) Survey

(f) Causal

(g) Econometrics

(h) High dimensional

(i) High dimensional

(j) Theory

(k) Empirical likelihood/Inference

(l) High dim/Multivariate

(m) Sequential

(n) Spatial/Image

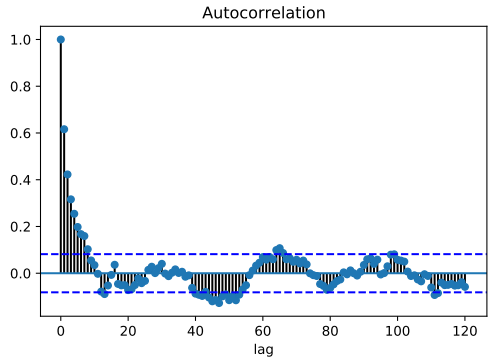
(o) Dimensionality reduction

(p) Time series

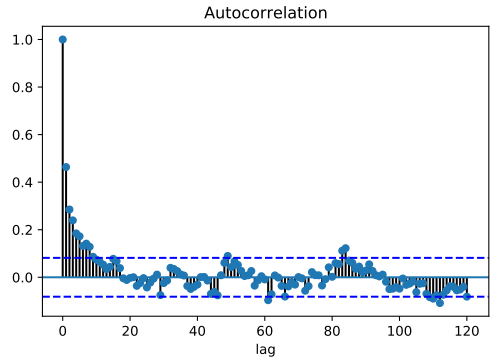
(q) Shape constrained

(r) Theory

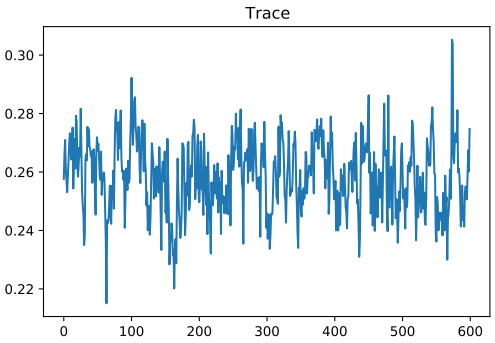
Figure 27: Additional clusters from the statistician co-authorship network. We hand label a subset.



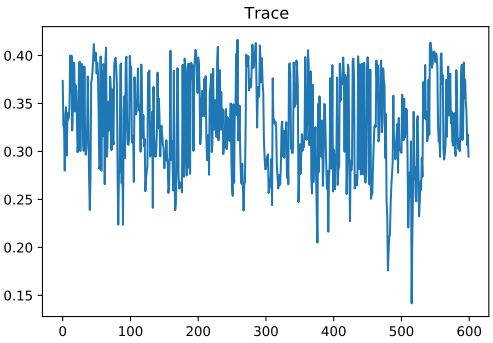
(a) ACF plot for a simulated network



(b) ACF plot for the co-authorship network

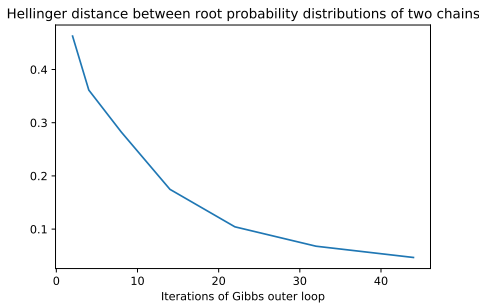


(c) Trace plot for a simulated network

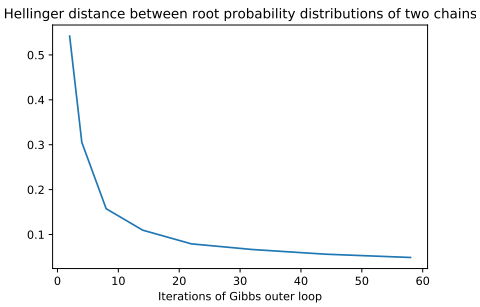


(d) Trace plot for the co-authorship network

Figure 28



(a) Hellinger distance between the root probability distributions of two chains on a simulated network.



(b) Hellinger distance between the root probability distributions of two chains on the co-authorship network.

Figure 29